Music City Code Edition

MUSIC CITY CODE

MICROSERVICES

Lessons from the Trenches

@gbworld

MUSIC CITY CODE

Thanks to our Sponsors

Hall of Famers

INGRAM®

Availity®

Rockstars

iVision®

#buildwhatmatters
LIFEWAY LABS
NASHVILLE

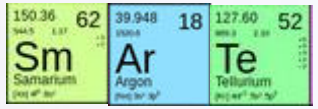
TEKsystems
Our people make IT possible

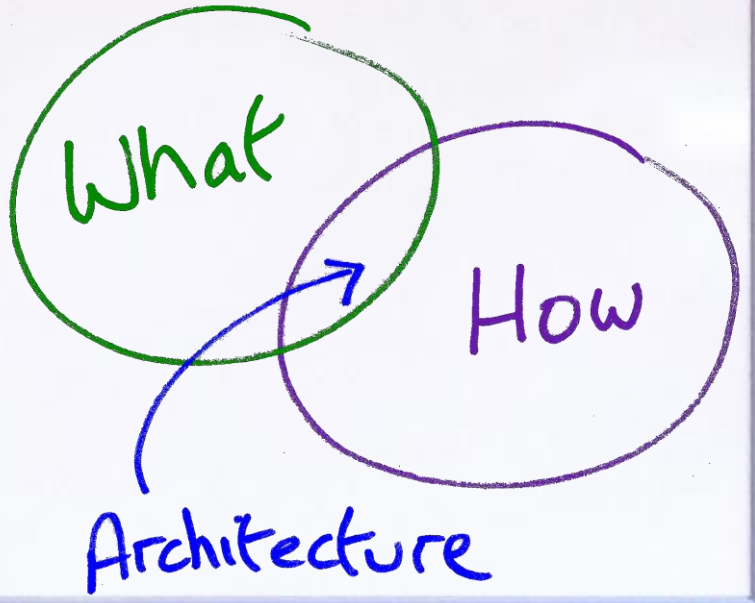
MUSIC CITY CODE

Who?

Gregory A. Beamer

- Senior Architect, UST Global
  - Microservices Practice
  - Senior Architect
    - Modernization
    - Integration
    - Service Enablement
    - Microservices
  - Other
    - Agile
    - DevOps





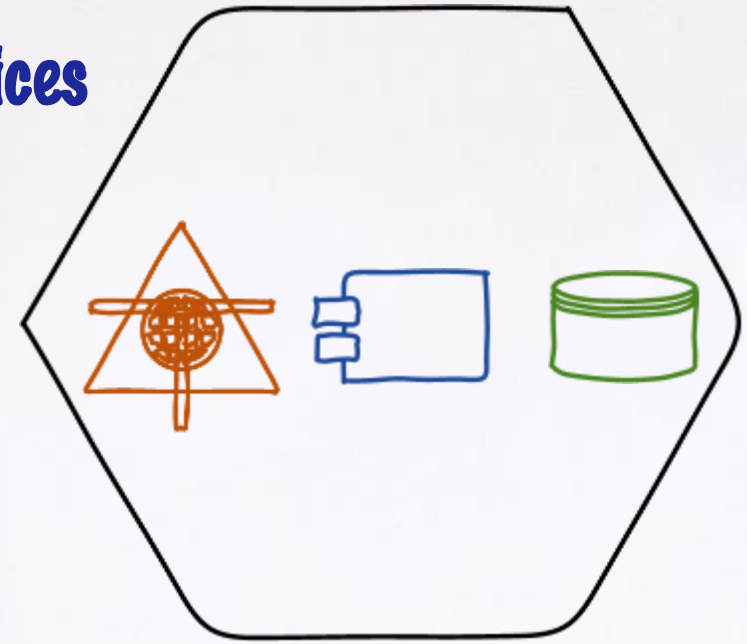
Opening Remarks

- Insert Greg's Random Bull Here
  Possibilities
  - Architectural Styles
  - Other Bull

MUSIC CITY CODE

Agenda

- What are Microservices?
  A Brief Introduction
- Why use Microservices?
  Benefits and breaking some myths about microservices
- How do we do this?
  Prepping for & Implementing Microservices
- Here in the Real World
  Inspired by Real Events



MUSIC CITY CODE

MUSIC CITY CODE

What are Microservices?

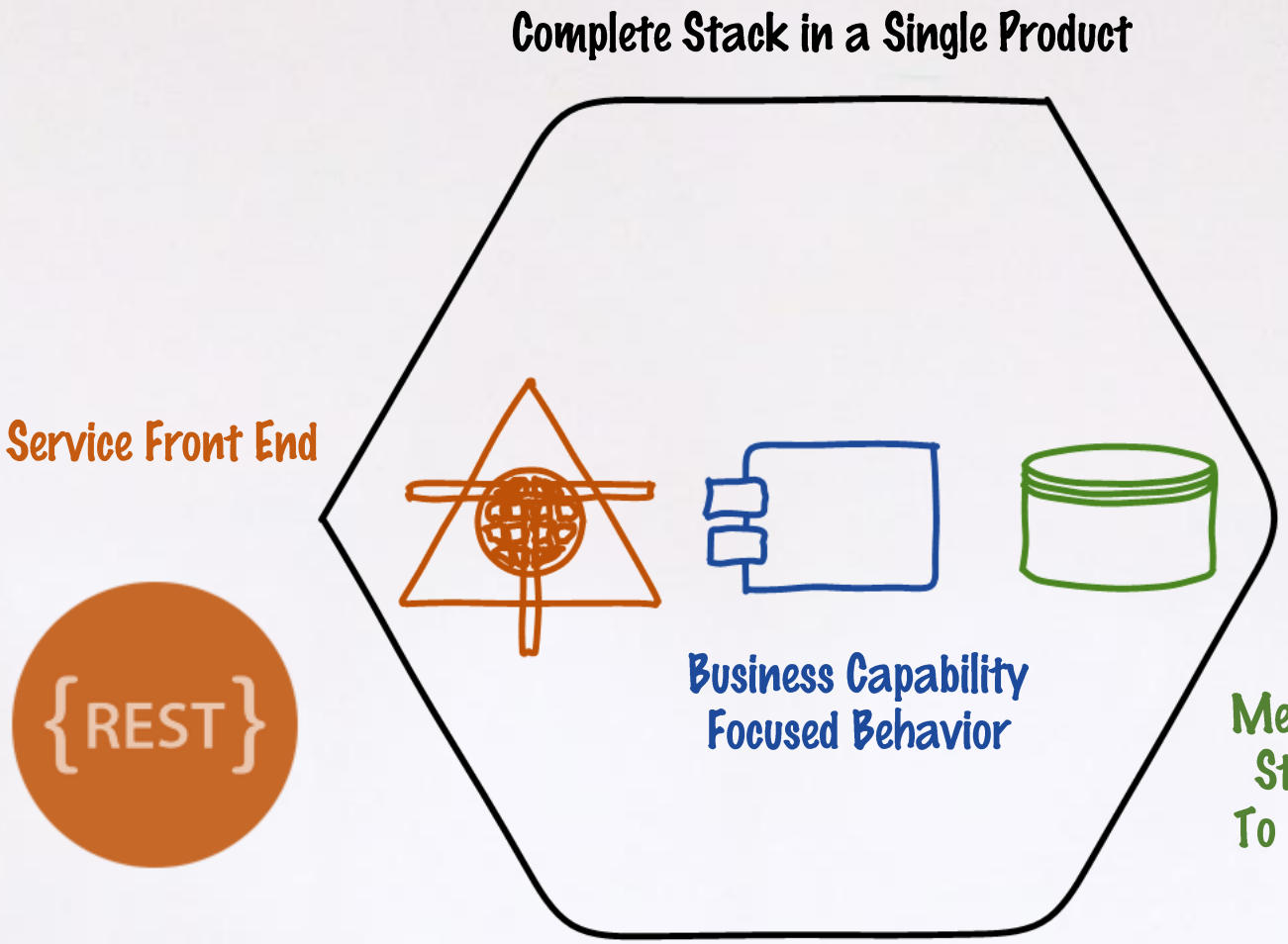
A Brief Introduction

MUSIC CITY CODE

What?

In a Nutshell

PURIST VIEW



Architectural Style

MUSIC CITY CODE

What?

Architectural Style

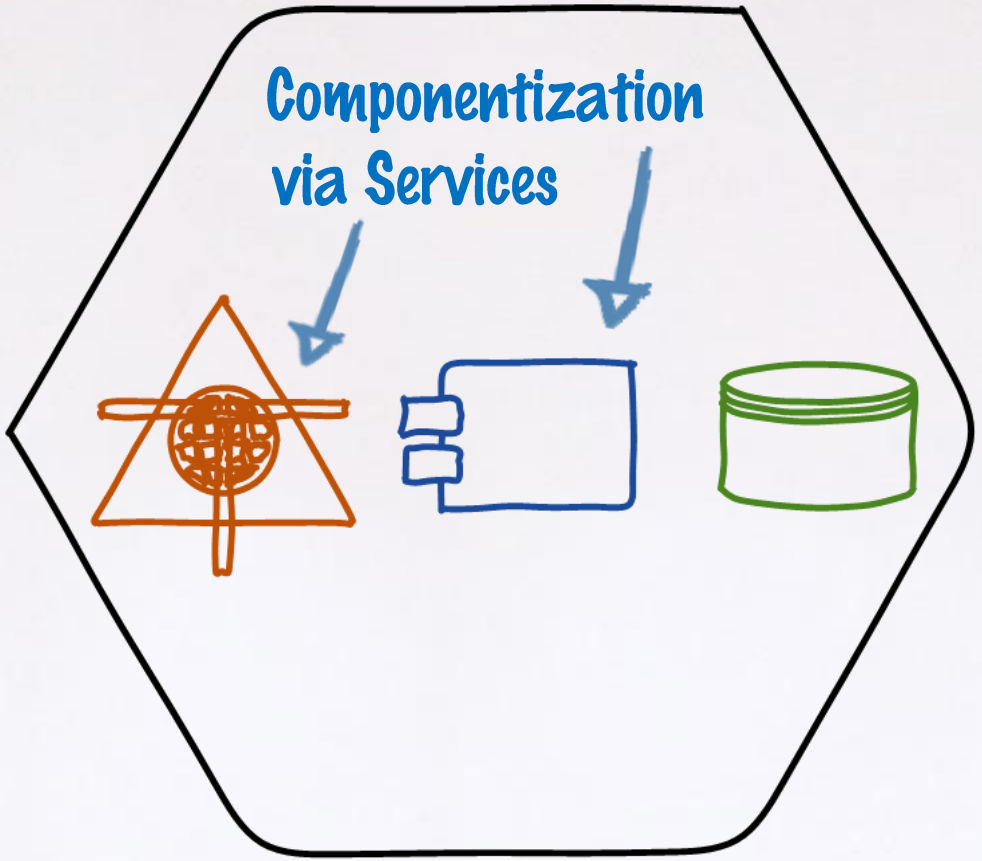
In short, the microservice architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms, often an HTTP resource API

From "Characteristics of a Microservice"-
Martin Fowler
http://goo.gl/AZwucZ

MUSIC CITY CODE

What?

Characteristics

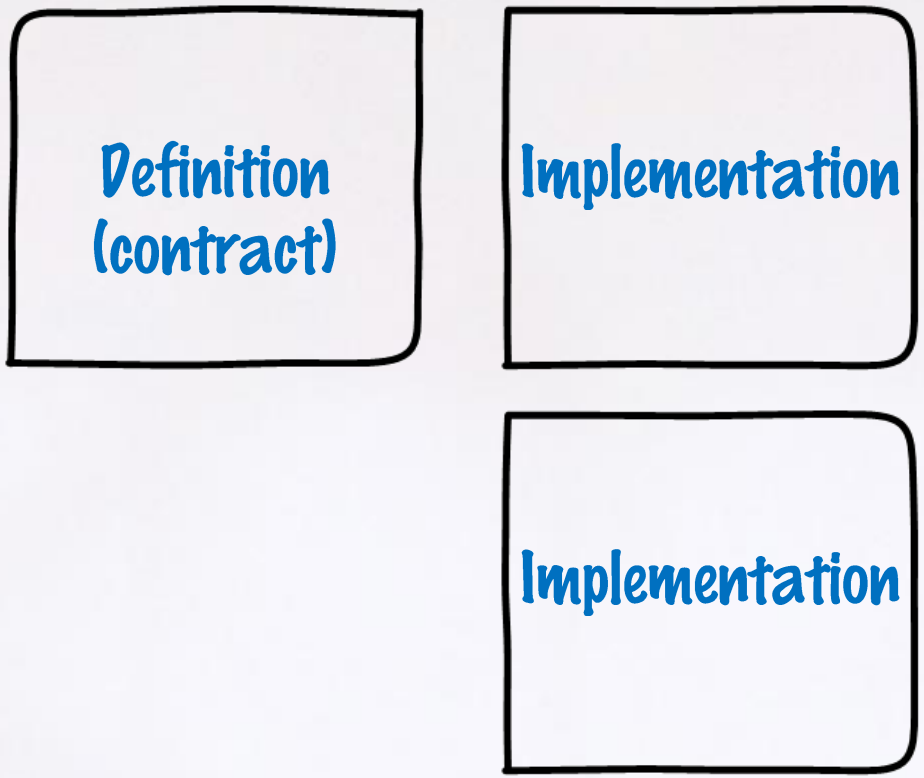


From "Characteristics of a Microservice" -
Martin Fowler
http://goo.gl/AZwucZ

MUSIC CITY CODE

Sidebar 1

What is a Component?

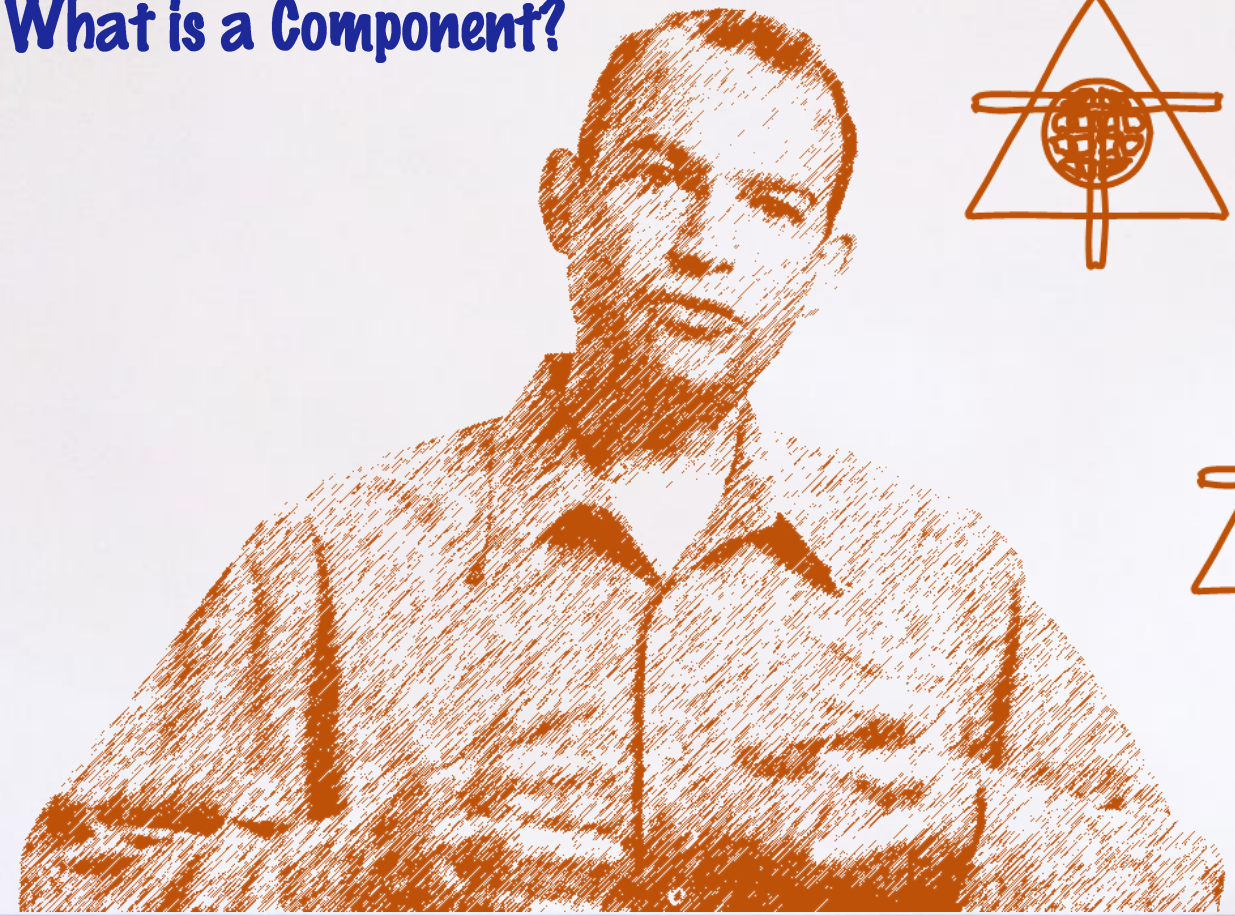


MUSIC CITY CODE

Microsoft .NET WCF

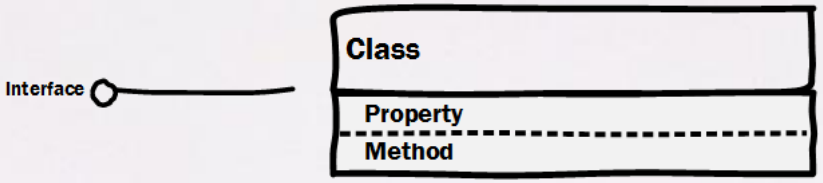
Sidebar 1

What is a Component?



Contract

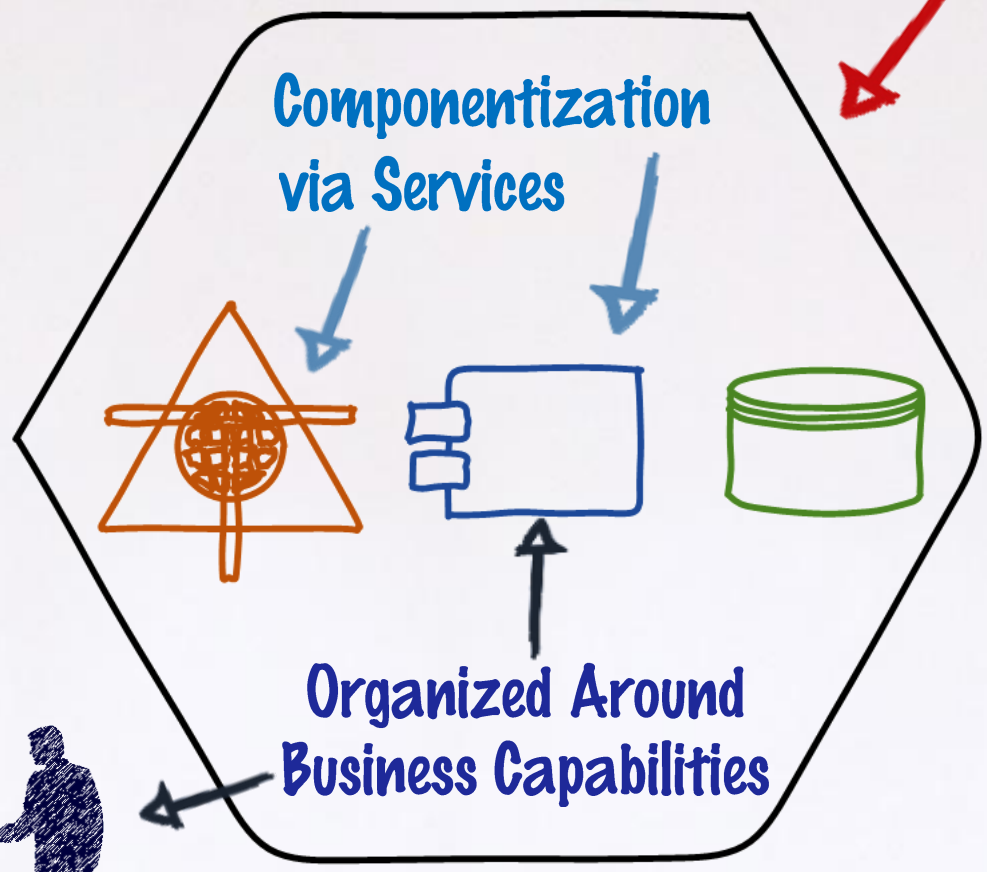
Implementation



MUSIC CITY CODE

What?

Characteristics

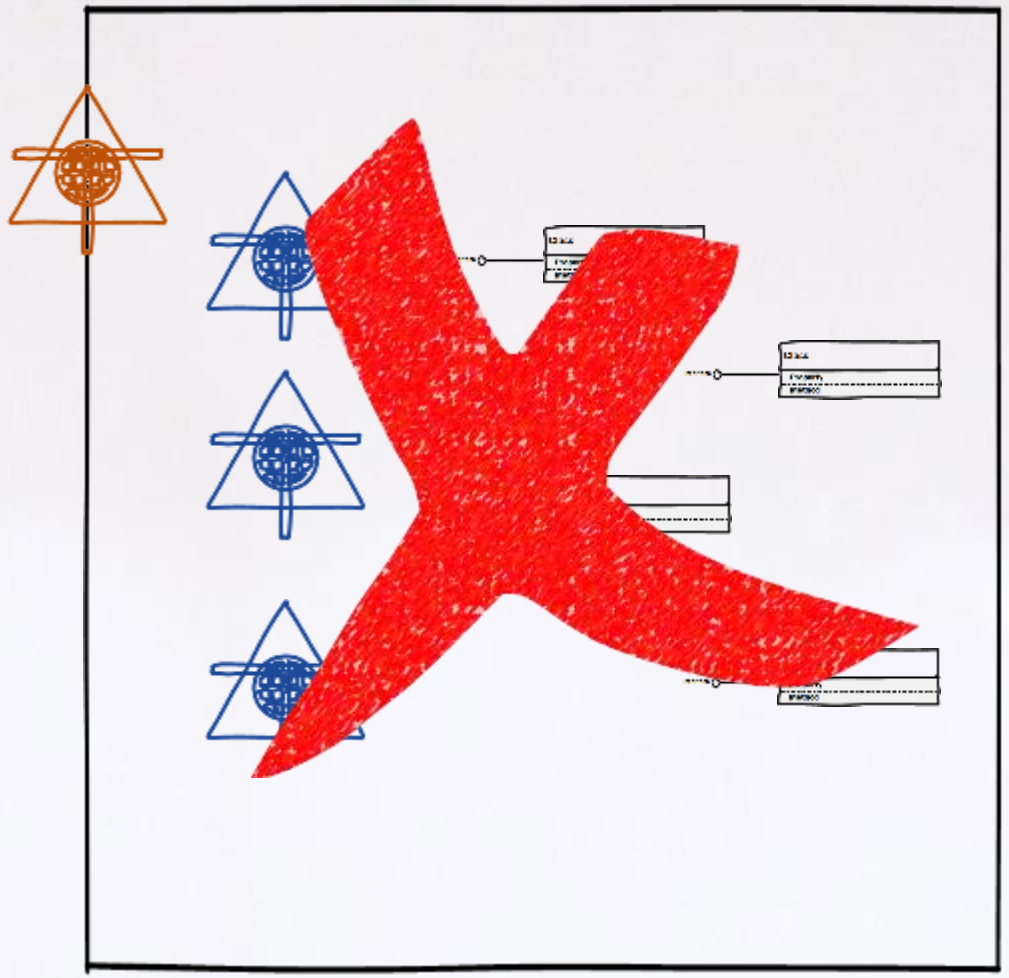


From "Characteristics of a Microservice"- Martin Fowler
http://goo.gl/AZwucZ

MUSIC CITY CODE

Sidebar 2

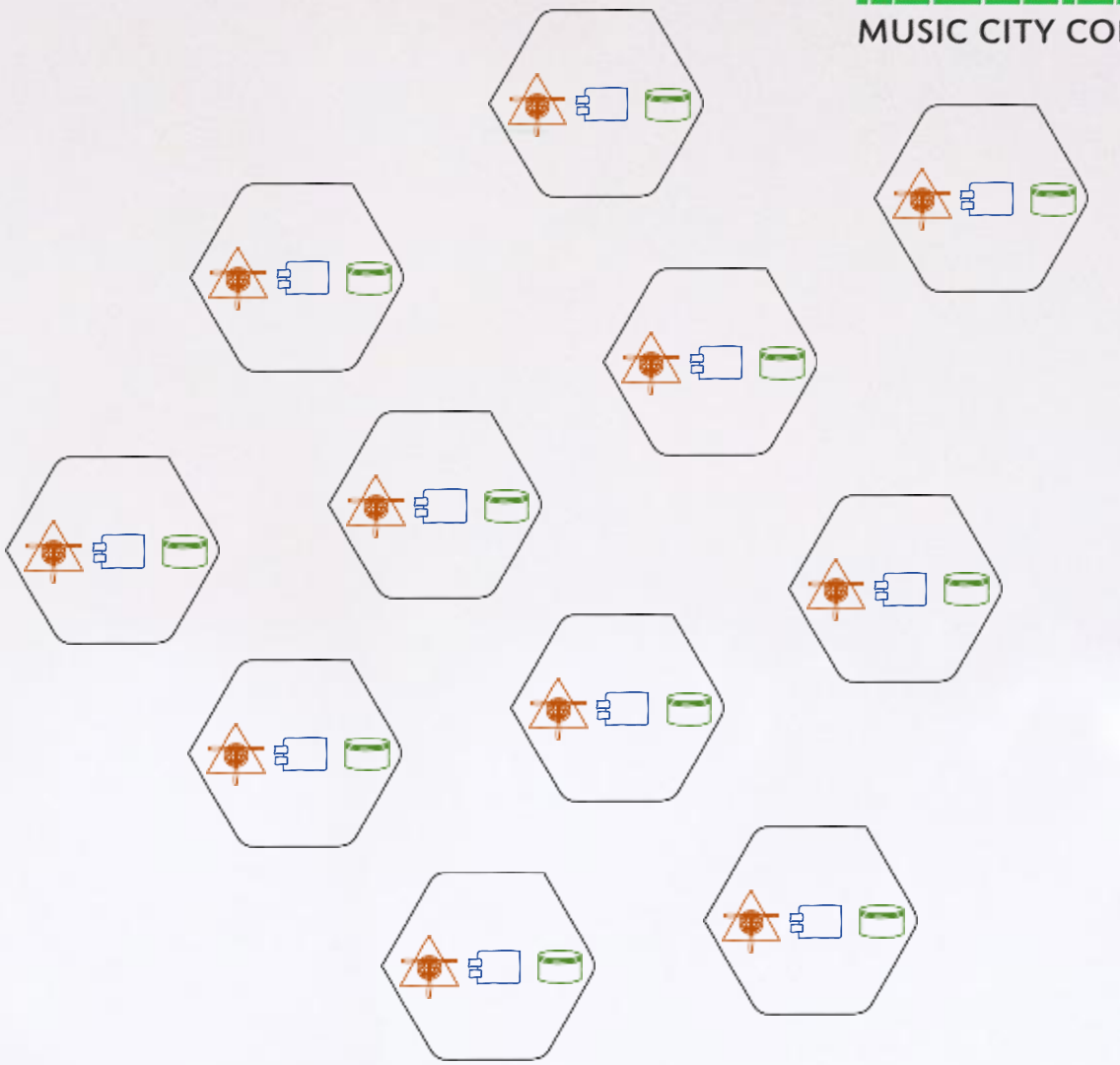
What is a Product?



MUSIC CITY CODE

Sidebar 2

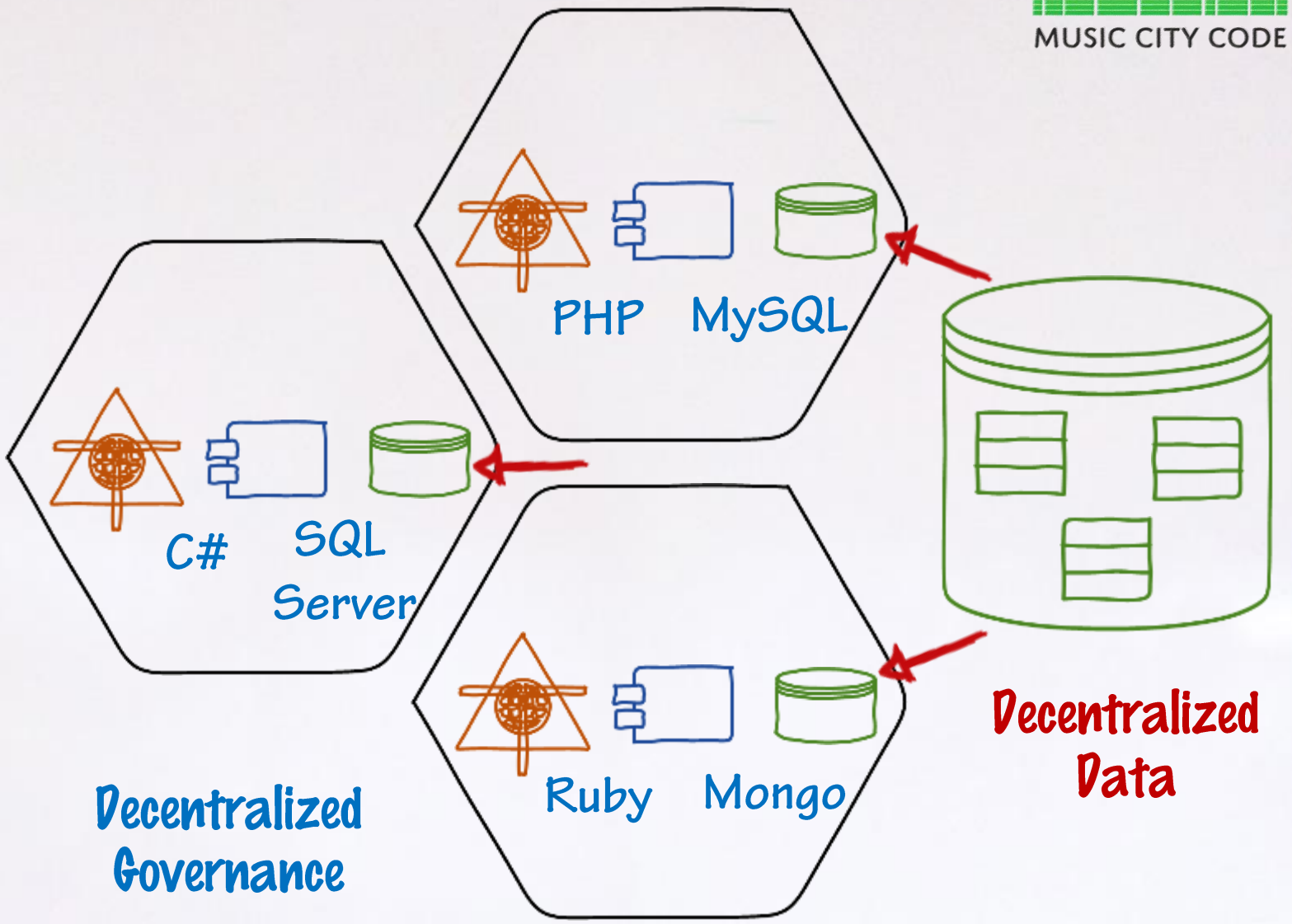
What is a Product?



MUSIC CITY CODE

What?

Characteristics

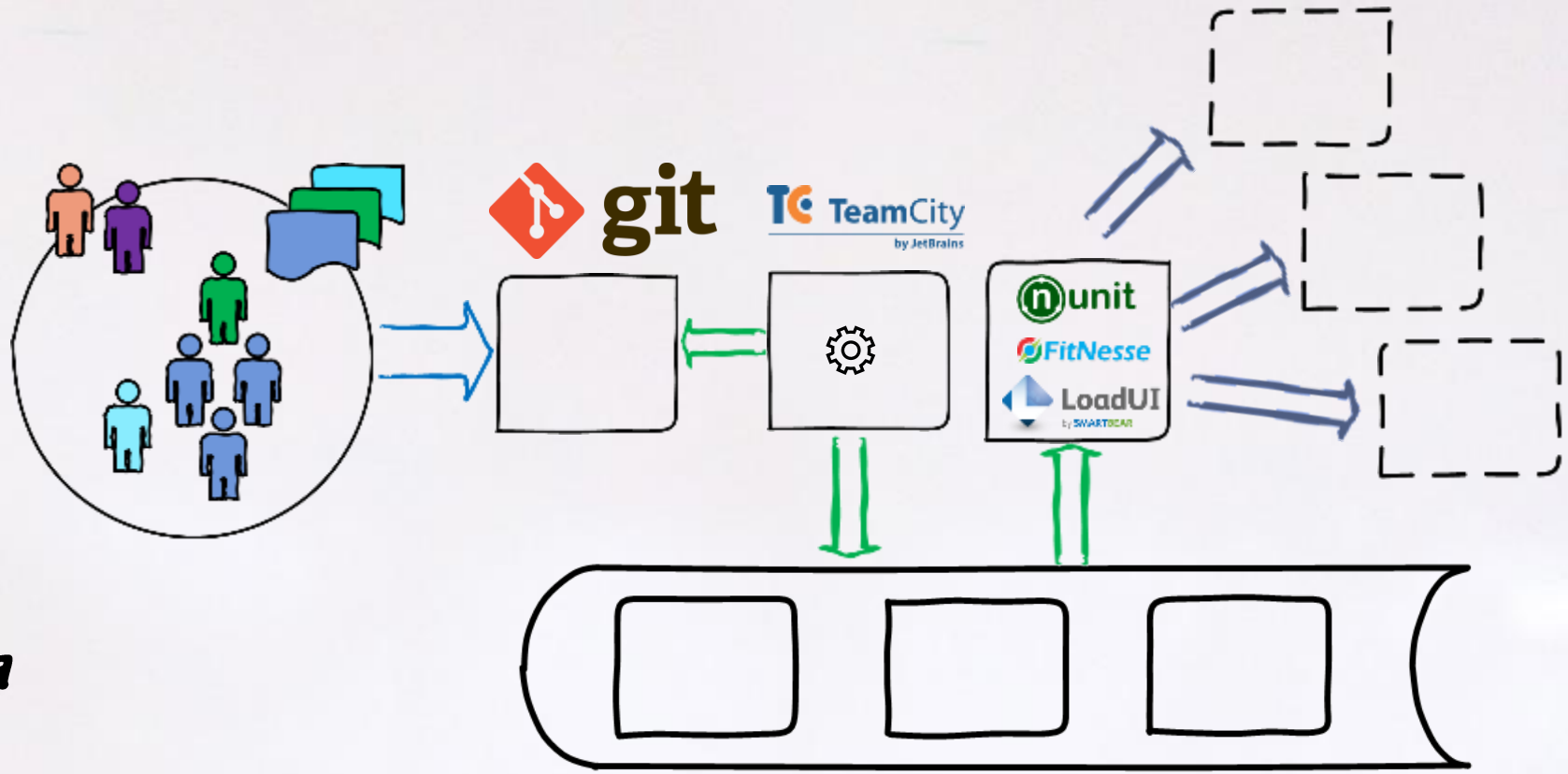


From "Characteristics of a Microservice"- Martin Fowler
http://goo.gl/AZwucZ

Infrastructure Automation

What?

Characteristics



From "Characteristics of a Microservice"-
Martin Fowler
http://goo.gl/AZwucZ

puppet

CHEF
CODE CAN

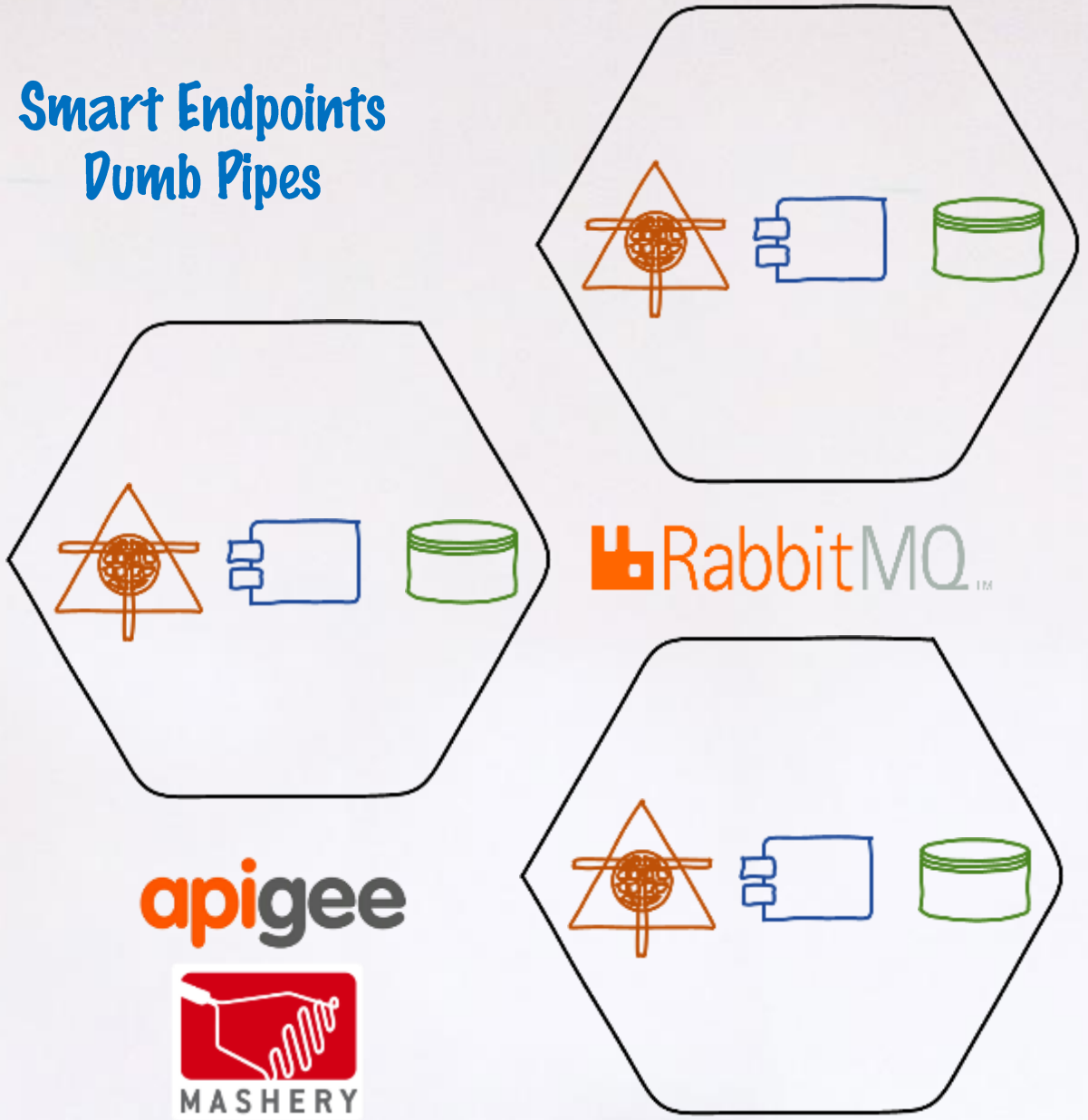
Bamboo

MUSIC CITY CODE

What?

Characteristics

Smart Endpoints
Dumb Pipes



RabbitMQ

apigee

MASHERY

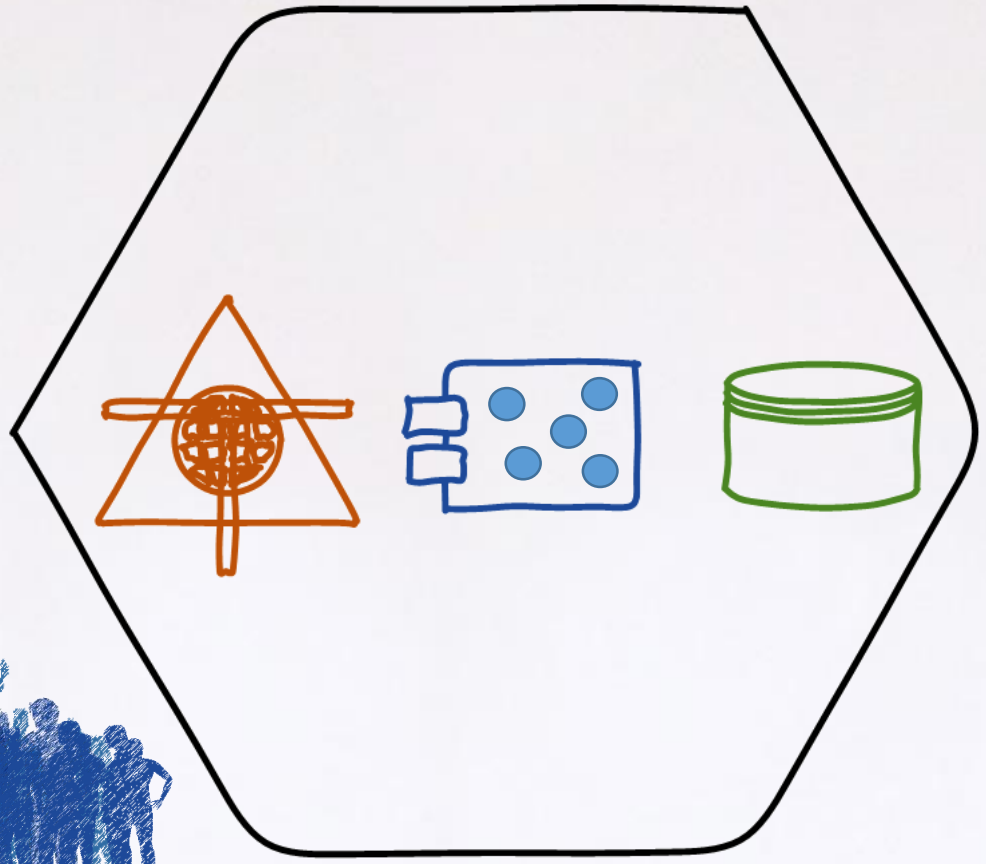
From "Characteristics of a Microservice"- Martin Fowler
http://goo.gl/AZwucZ

MUSIC CITY CODE

Evolutionary Design

What?

Characteristics



From "Characteristics of a Microservice"- Martin Fowler
http://goo.gl/AZwucZ

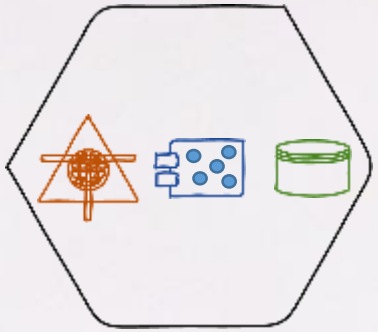
MUSIC CITY CODE

What?

Characteristics

Microservices

- Componentization via Services
- Organized around business capabilities



Agile Practices

- Products not Projects
- Decentralization (governance, data)
- Evolutionary Design

From "Characteristics of a Microservice"- Martin Fowler
http://goo.gl/AZwucZ

DevOps

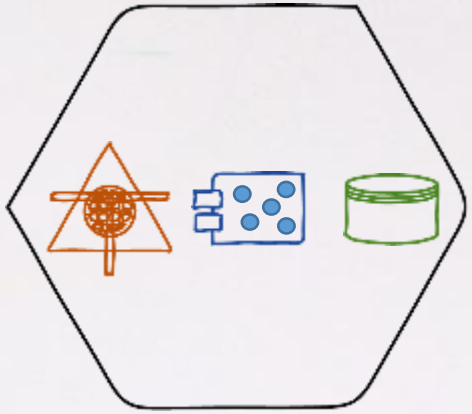
- Infrastructure Automation Smart Endpoints, Dumb Pipes

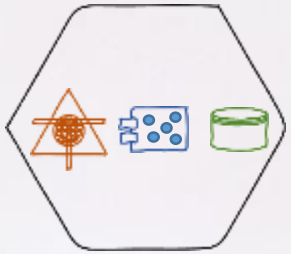
MUSIC CITY CODE

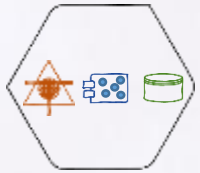
What?

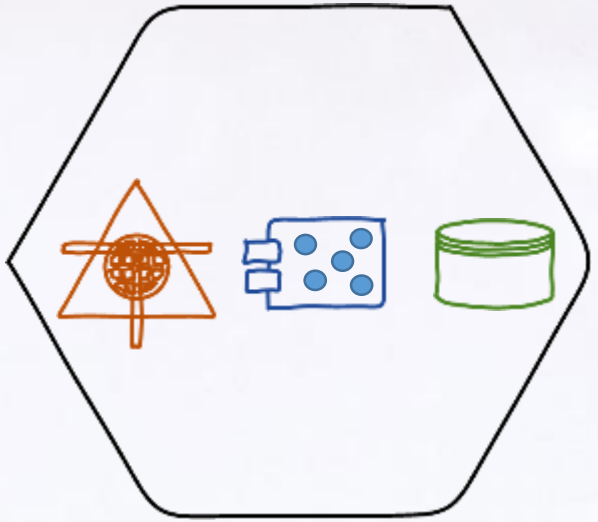
How Big is a Microservice?

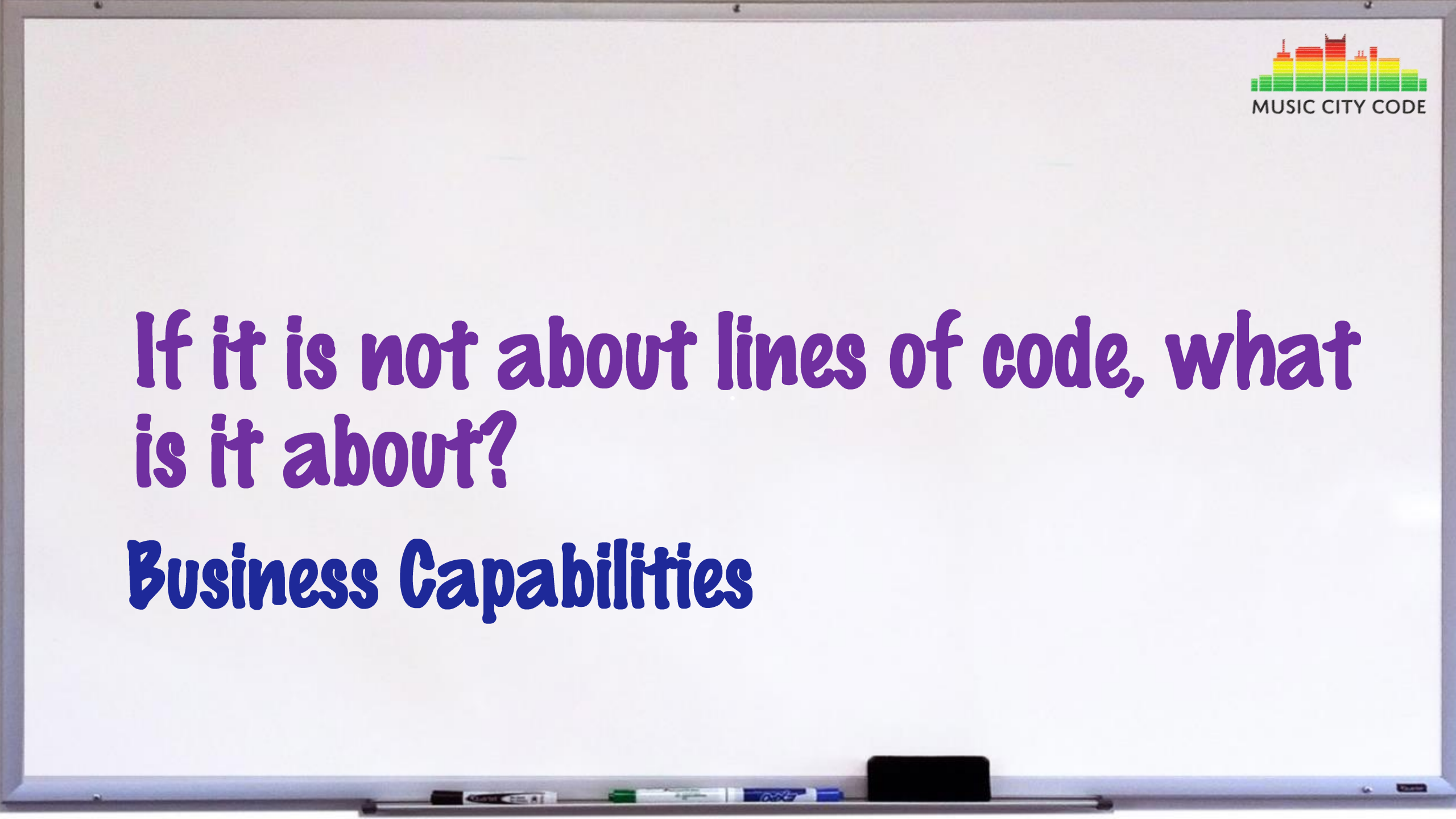
It's **not** about lines of code!










MUSIC CITY CODE
If it is not about lines of code, what is it about?
Business Capabilities


MUSIC CITY CODE
Why Use Microservices?
Benefits and breaking some myths about microservices

MUSIC CITY CODE

Why?

The Benefits of Microservices

- Better than Monoliths
- Easy to
  - Understand
  - Enhance
  - Test
  - Deploy
- Resilience
  - Low impact on other services
- Autonomy of Teams
  - Decentralized Governance

Independent Scale

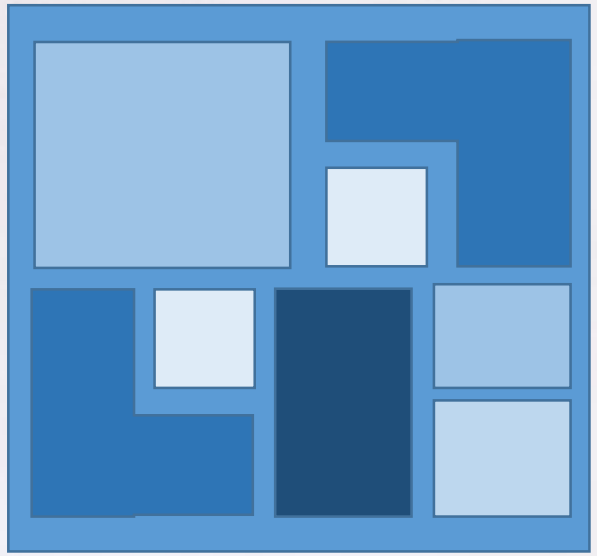
Flexibility

Loose Coupling
Tight Cohesion

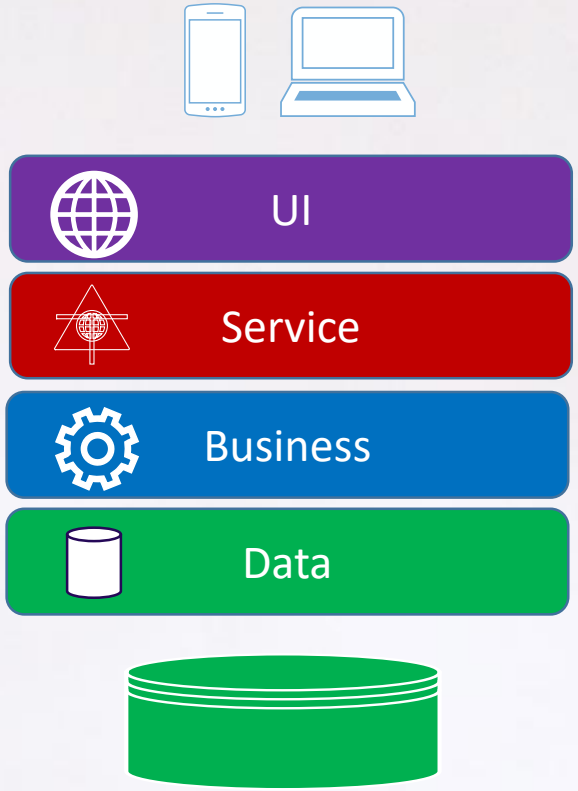
CITY CODE

Why?

Monolith versus Microservices

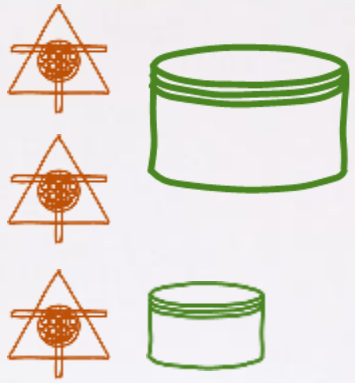


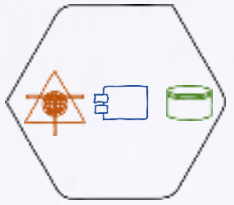
Monolithic Architecture



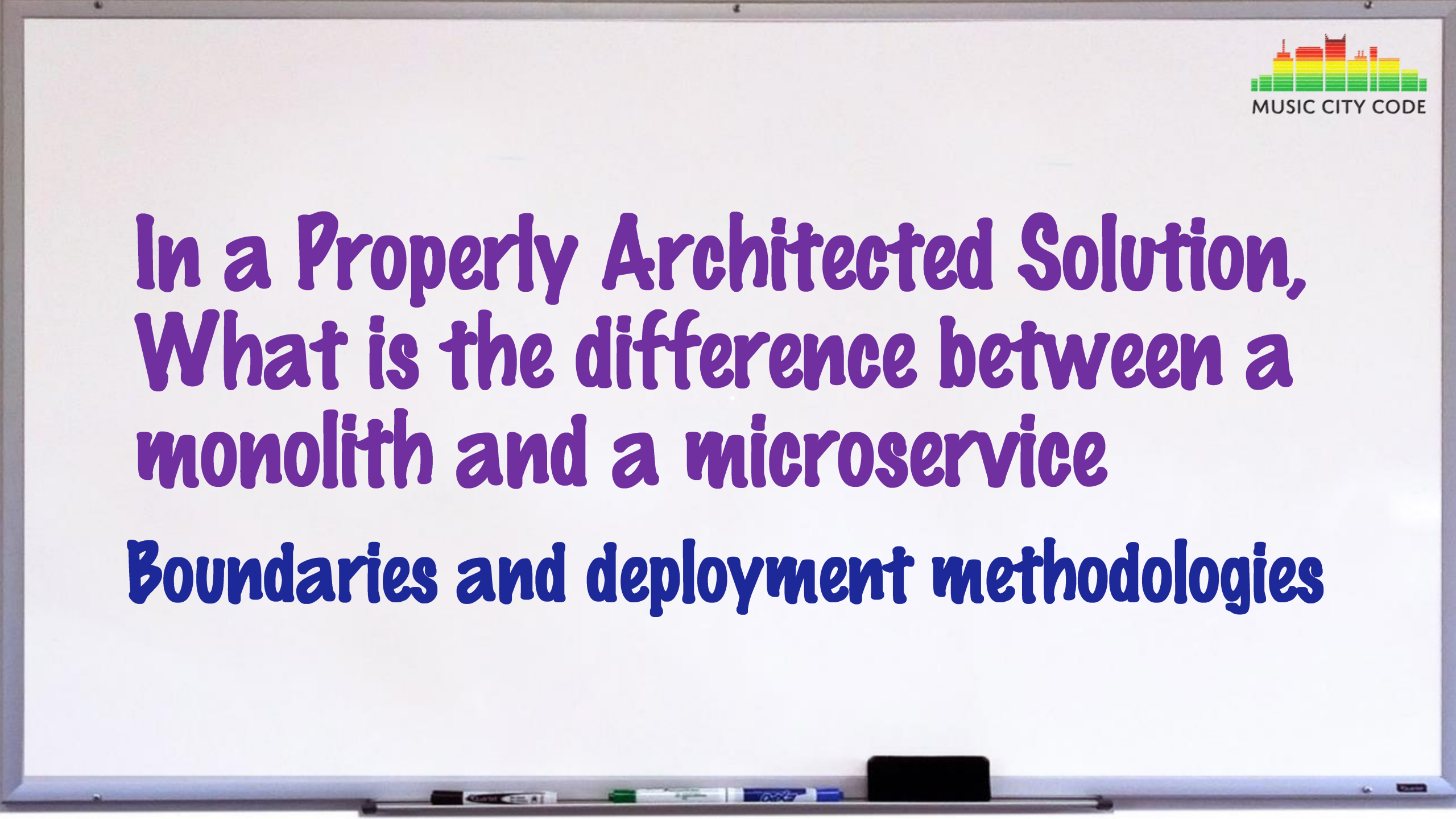
Layered Architecture

SOA





Microservice Architecture


MUSIC CITY CODE
In a Properly Architected Solution, What is the difference between a monolith and a microservice
Boundaries and deployment methodologies

MUSIC CITY CODE

Why?

Microservices are Easy

- Easy to Understand
- Easy to Enhance
- Easy to Test
- Easy to Deploy

MUSIC CITY CODE

Why?

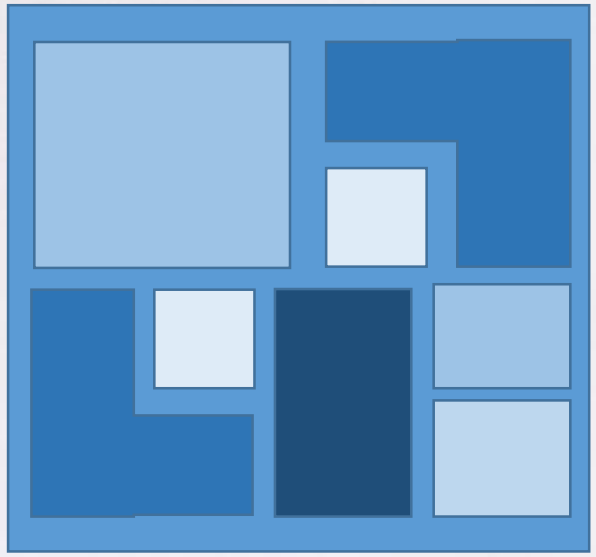
Microservices are Easy

Item	Microservice	Monolith
Lines of Code	500	1,000,000
Easier to understand	✓	A*
Easier to enhance	✓	
Easier to test	✓	
Easier to deploy	✓	

MUSIC CITY CODE

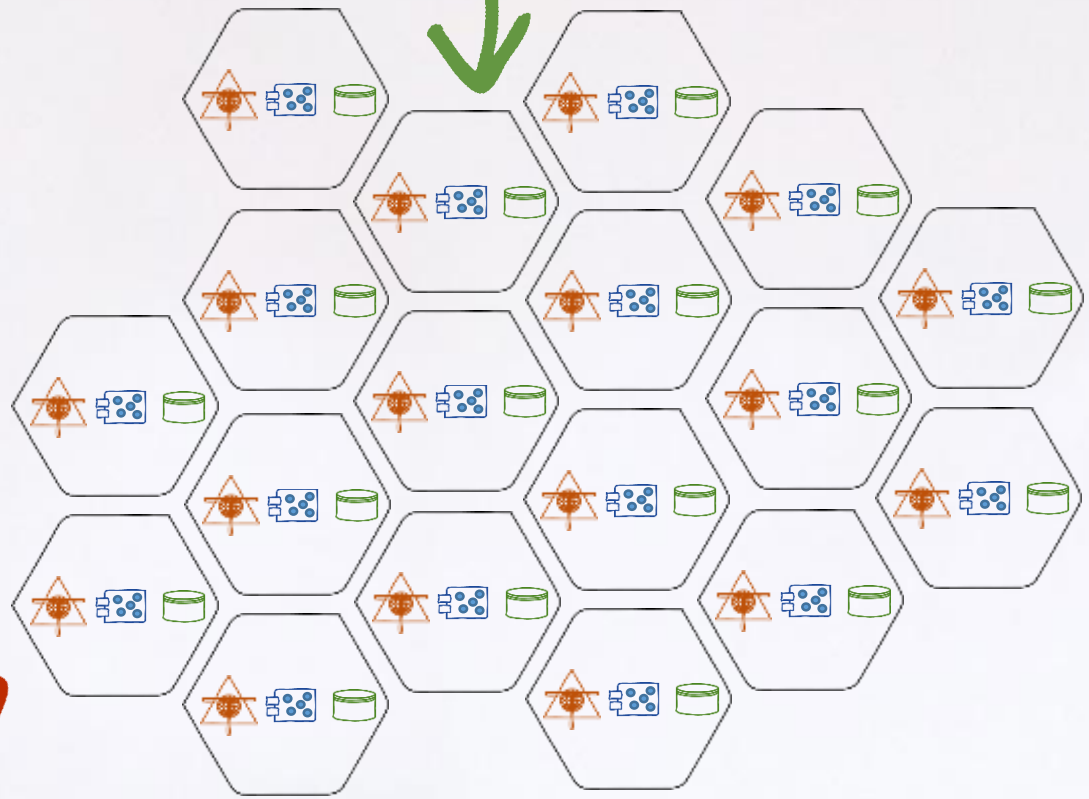
Why?

Microservices are Easy



This is not easy

This is easy



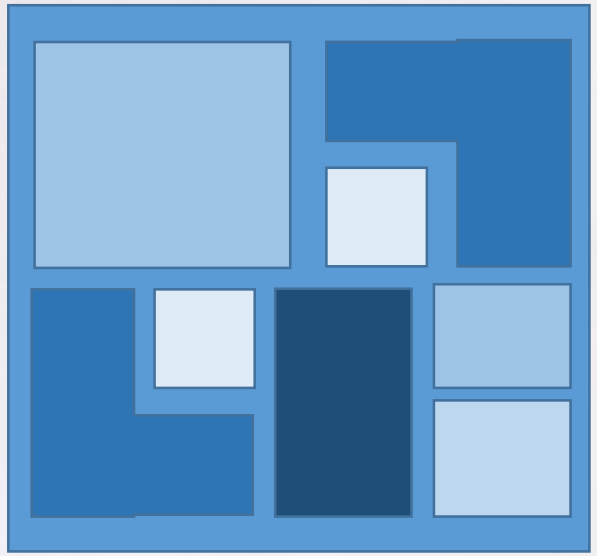
Inner Complexity

Outer Complexity

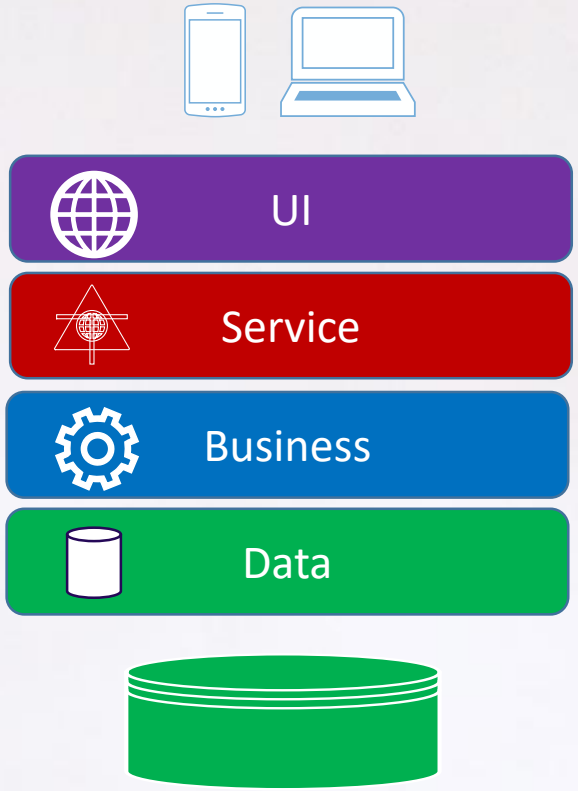
TY CODE

Sidebar

Complexity

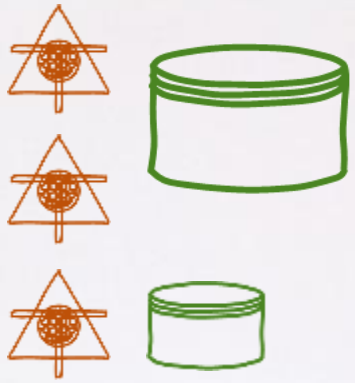


Monolithic Architecture

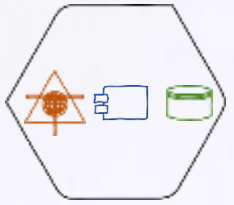


Layered Architecture

SOA



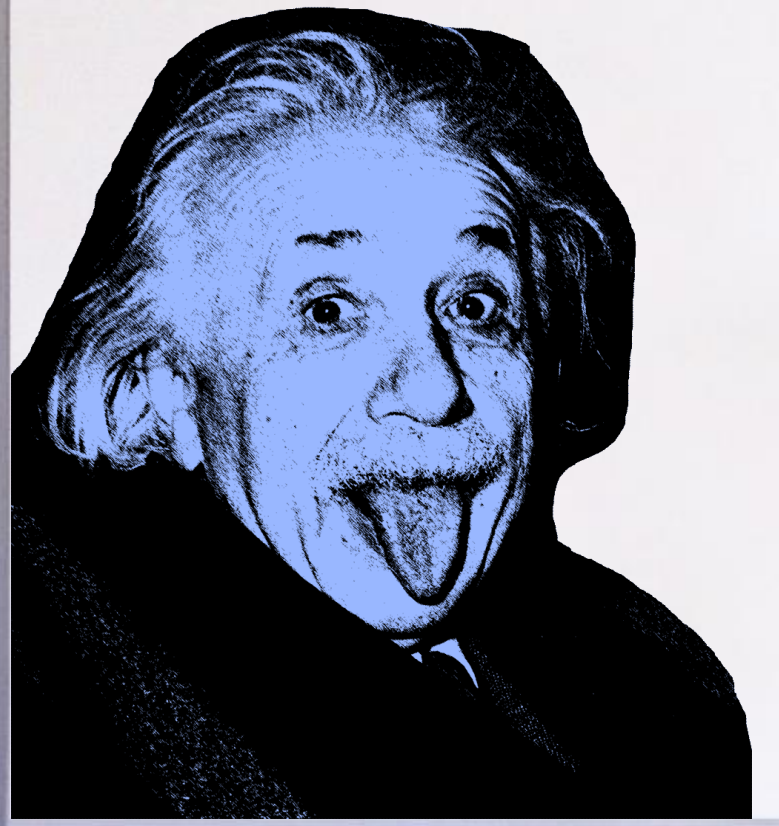
Microservice Architecture



MUSIC CITY CODE

Why?

Microservices are Easy



Energy cannot be created or destroyed, it can only be changed from one from to another.

– Albert Einstein

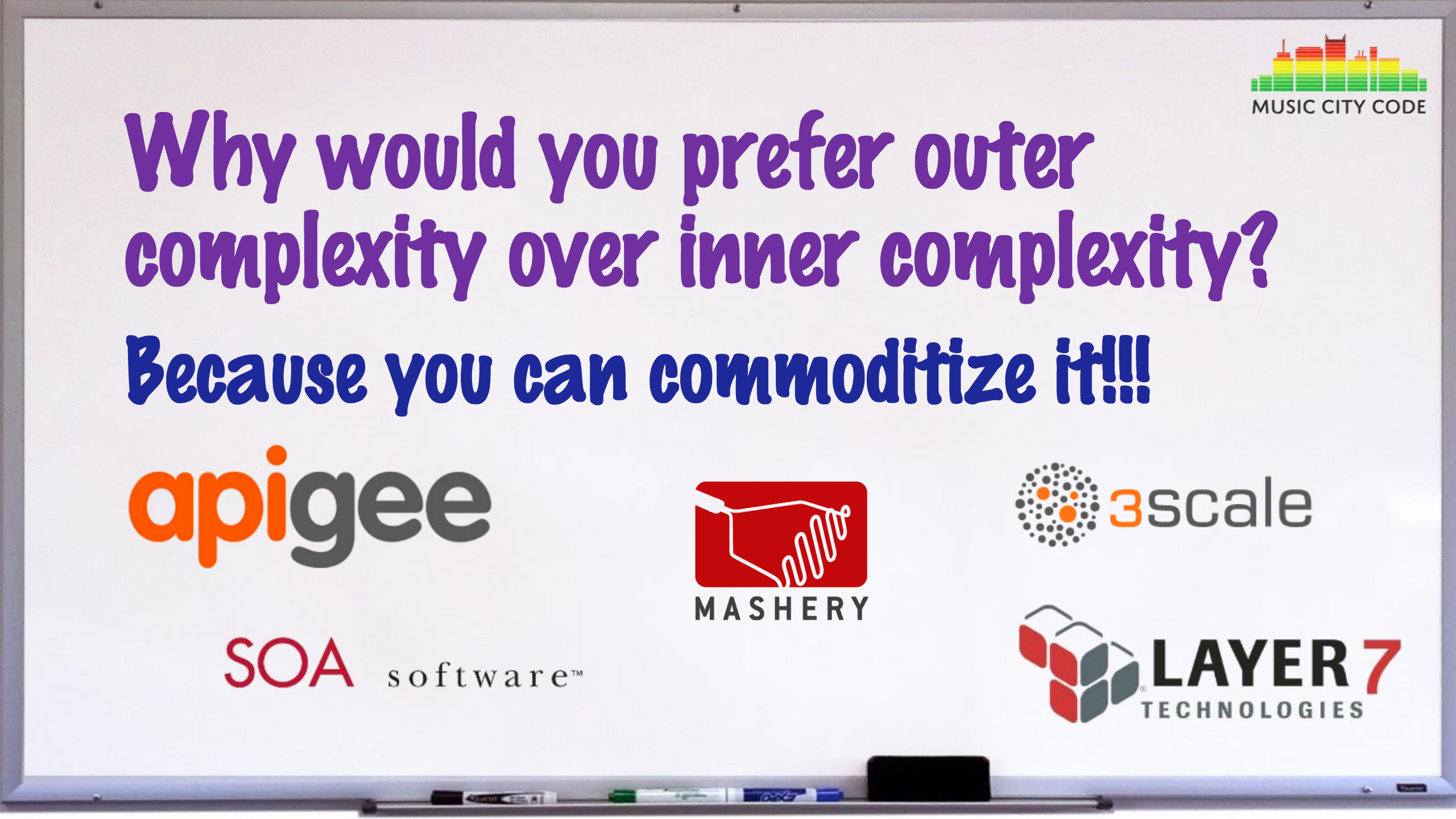
MUSIC CITY CODE

Why?

Microservices are Easy

Complexity cannot be destroyed, it can only be changed from one from to another.

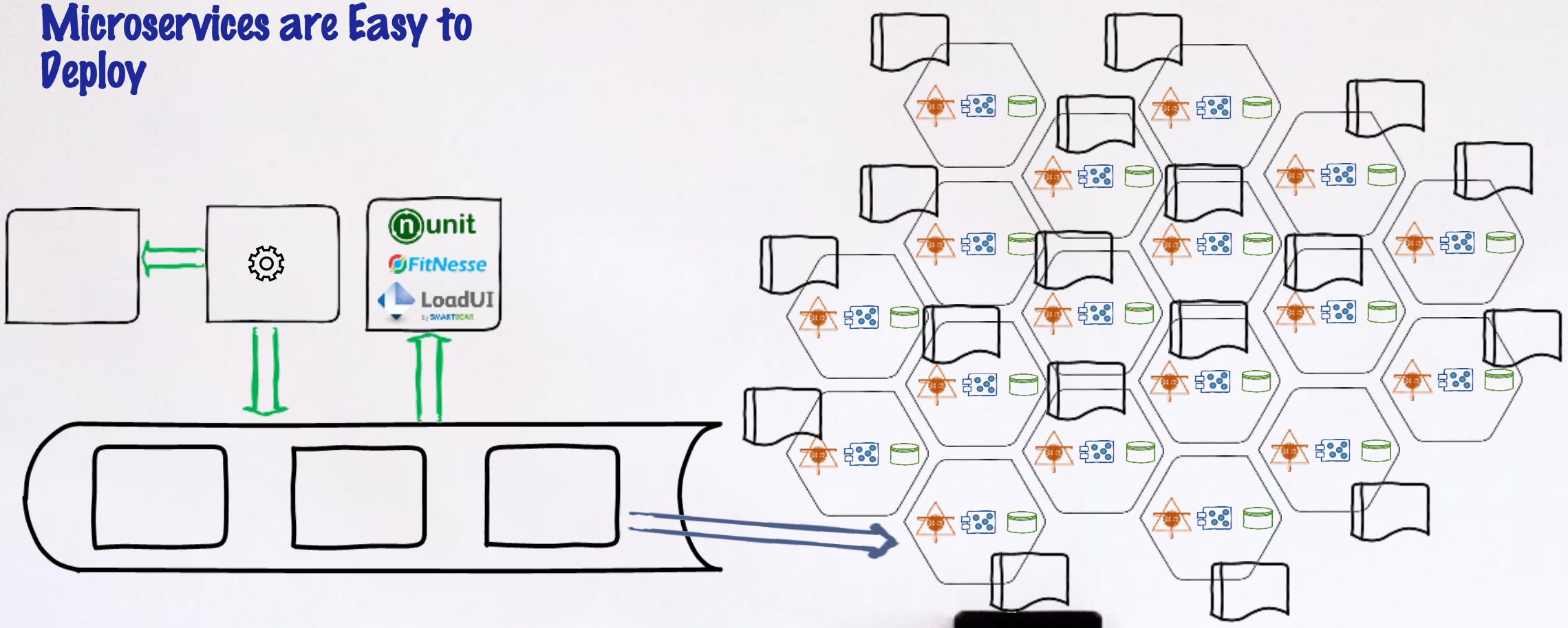
– Gregory A. Beamer


MUSIC CITY CODE
Why would you prefer outer complexity over inner complexity?
Because you can commoditize it!!!
apigee
MASHERY
3scale
SOA software™
LAYER 7 TECHNOLOGIES

MUSIC CITY CODE

Why?

Microservices are Easy to Deploy



MUSIC CITY CODE

Why spend all that time automating?

Because your ops people have better things to do

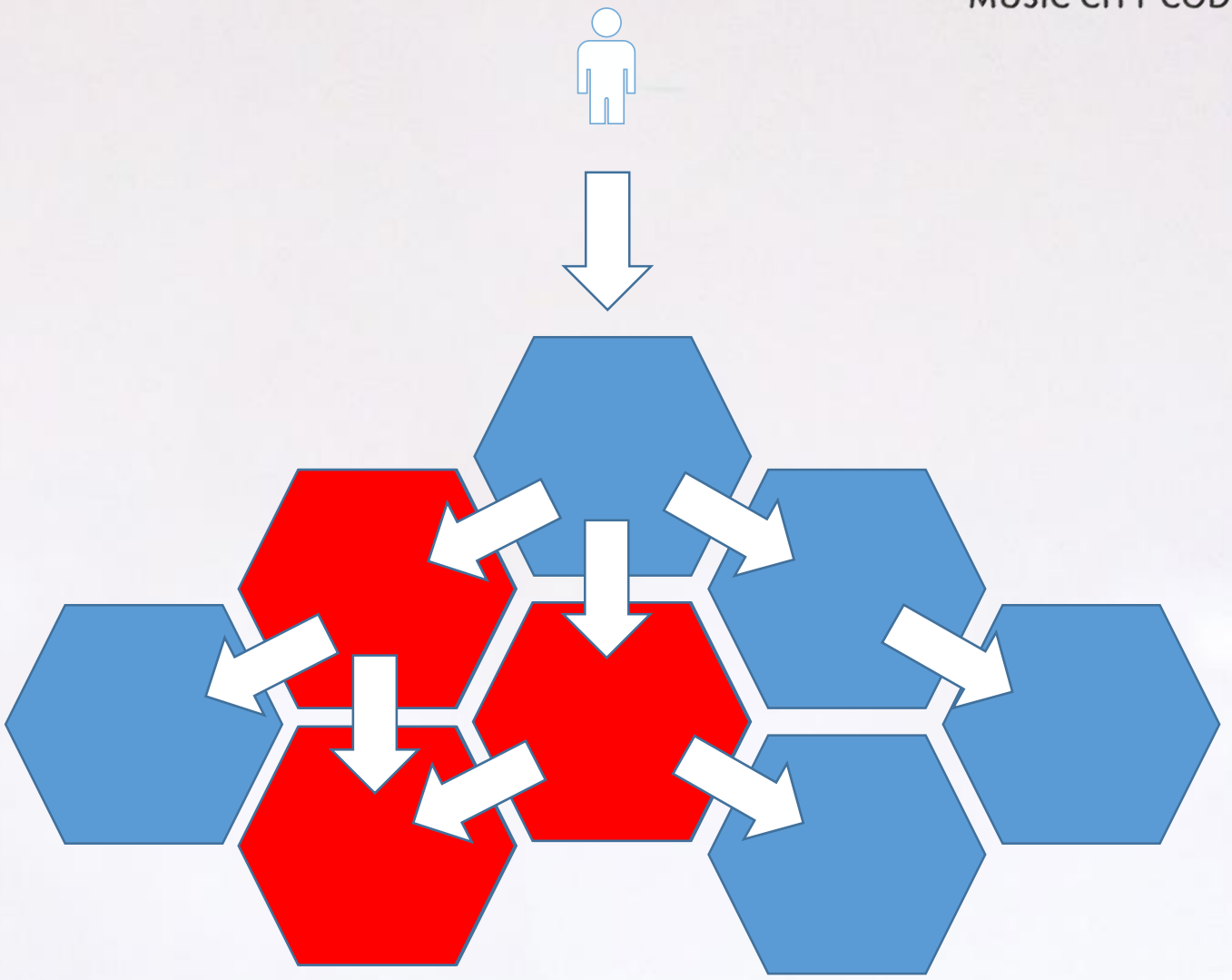
Because you can commoditize it!!!

MUSIC CITY CODE

Why?

Resilience

Low Impact on Other Services



MUSIC CITY CODE

How do you insure low impact on other services?

kubernetes by Google

docker

AppDynamics

apigee

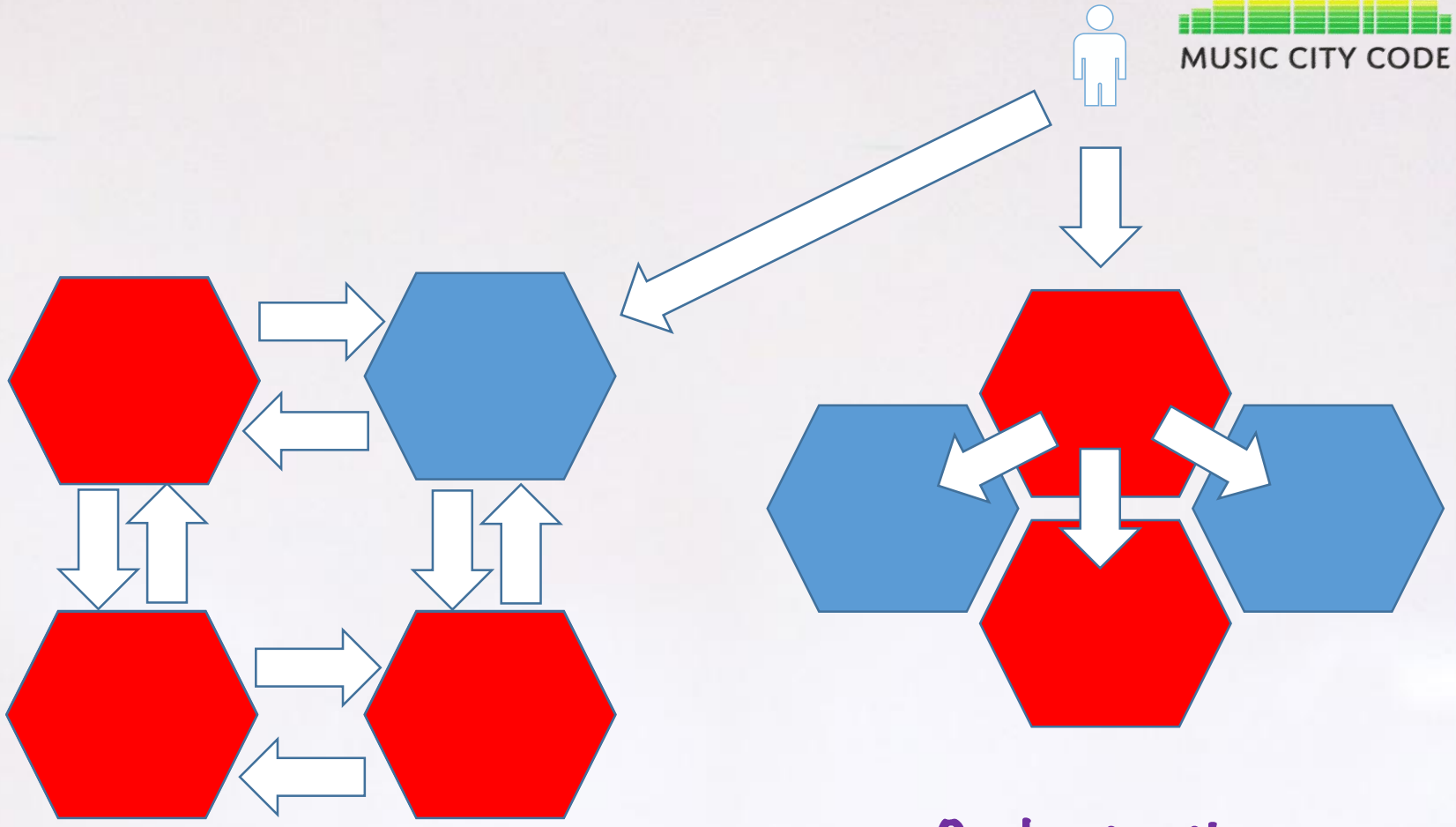
Because you can commoditize

... and WORLD PEACE!

MUSIC CITY CODE

Why?

Resilience



Choreography

Orchestration

MUSIC CITY CODE

Which is a better pattern: orchestration or choreography?

Which is better: A hammer or a screwdriver?

Hammers SUCK!!!

MUSIC CITY CODE

Why?

Autonomy of Teams


C#


SQL Server


Python


PHP


F#


Mongo DB


Oracle


Java


Couch

MUSIC CITY CODE

What do you call an organization with hundreds of microservices & every new technology known to man?

Bankrupt!

What?

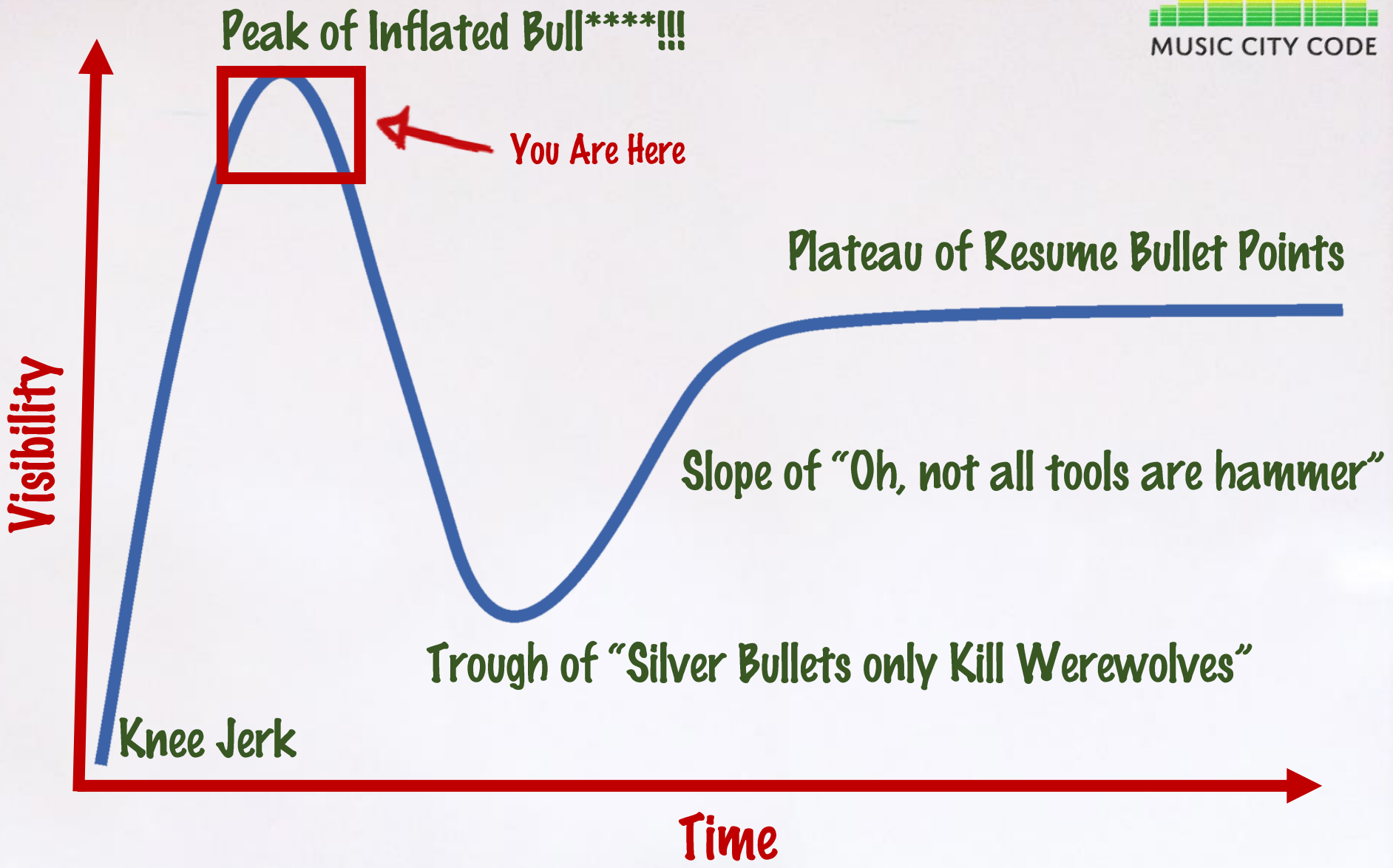
Silver Bullet or Hot Air?

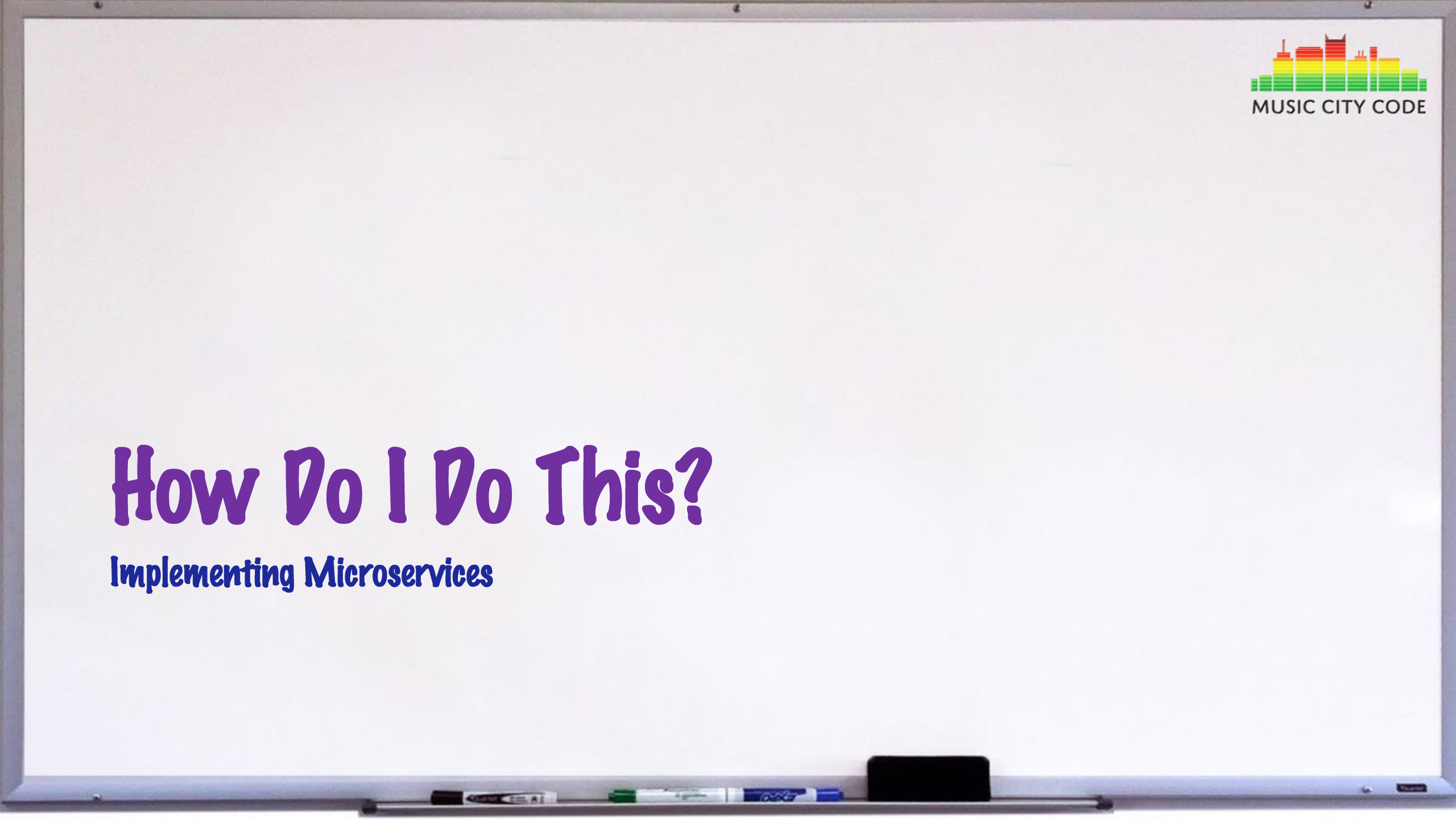
MUSIC CITY CODE

MUSIC CITY CODE

What?

Silver Bullet or Hot Air?




MUSIC CITY CODE
How Do I Do This?
Implementing Microservices

CASA Development Model

Core AS Application

MUSIC CITY CODE

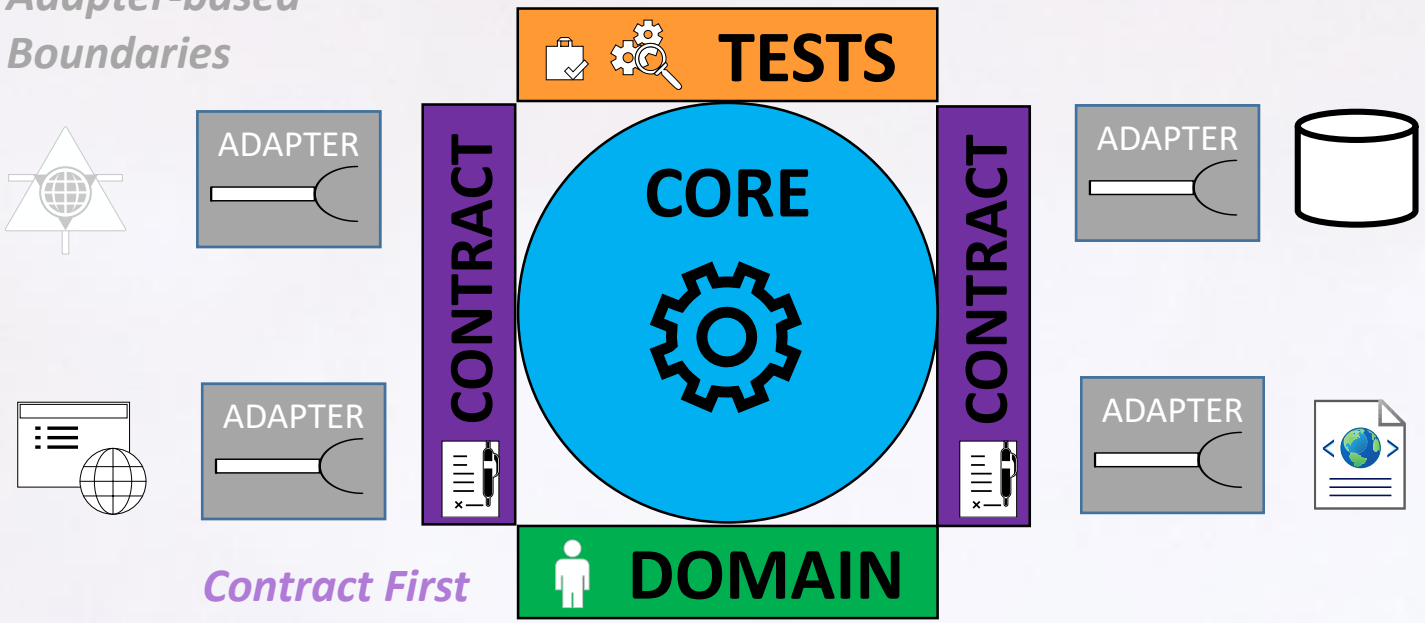
How?

Organize Your Code

Test Driven/ Test Early Approach

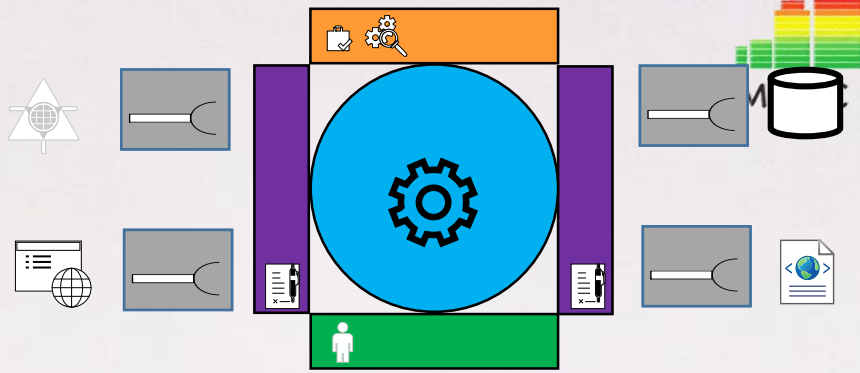
TDD *BDD* *ATDD*

Adapter-based Boundaries



Contract First Development

Domain Driven Design



CITY CODE

Solution Explorer

Search Solution Explorer (Ctrl+;)

Solution 'Microservices.Template' (13 projects)

- Core
  - Microservice.Template.Core
  - Microservice.Template.Core.Contracts
  - Core.txt
- Domain
  - Microservices.Template.Domain.Exceptions
  - **Microservices.Template.Domain.Models**
  - Domain.txt
- Framework
  - Framework.DependencyInjection
  - Framework.txt
- Persist
  - Microservice.Template.Persist.Contracts
  - Microservice.Template.Persist.Sql
  - Persist.txt
- Present
  - MicroserviceTemplate.Present.Web
  - Present.txt

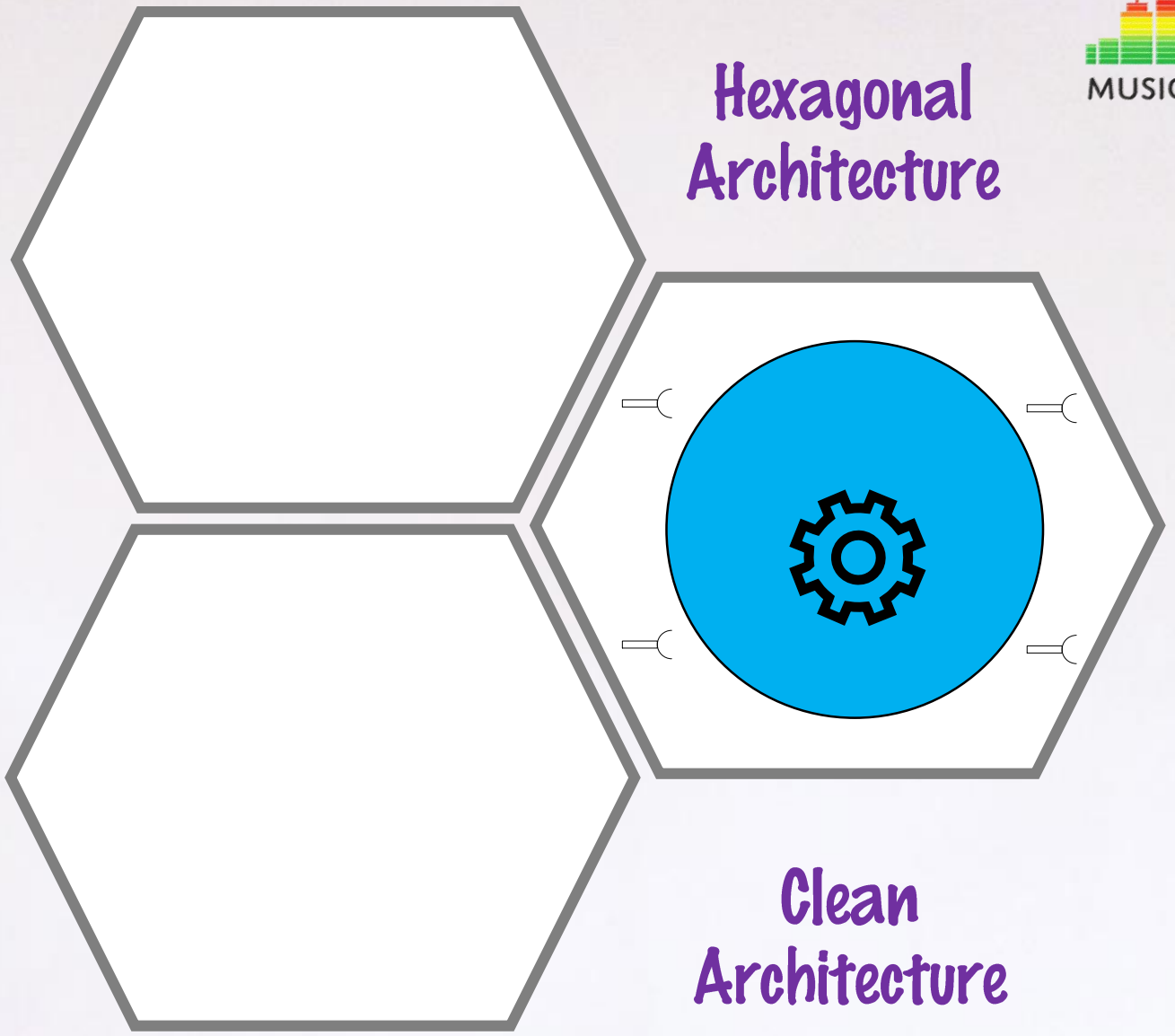
- Service
  - Microservice.Template.Service
  - Service.txt
- Test
  - Accept
  - Integ
    - Microservice.Template.Core.Test.Integ
    - Microservice.Template.Persist.Sql.Test.Integ
    - test.integ.txt
  - Spec
  - Unit
    - Microservice.Template.Core.Test.Unit
    - MicroserviceTemplate.Present.Web.Test.Unit
    - test.unit.txt
  - Test.txt

Solution Explorer | Team Explorer | Class View

MUSIC CITY CODE

How?

Organize Your Code



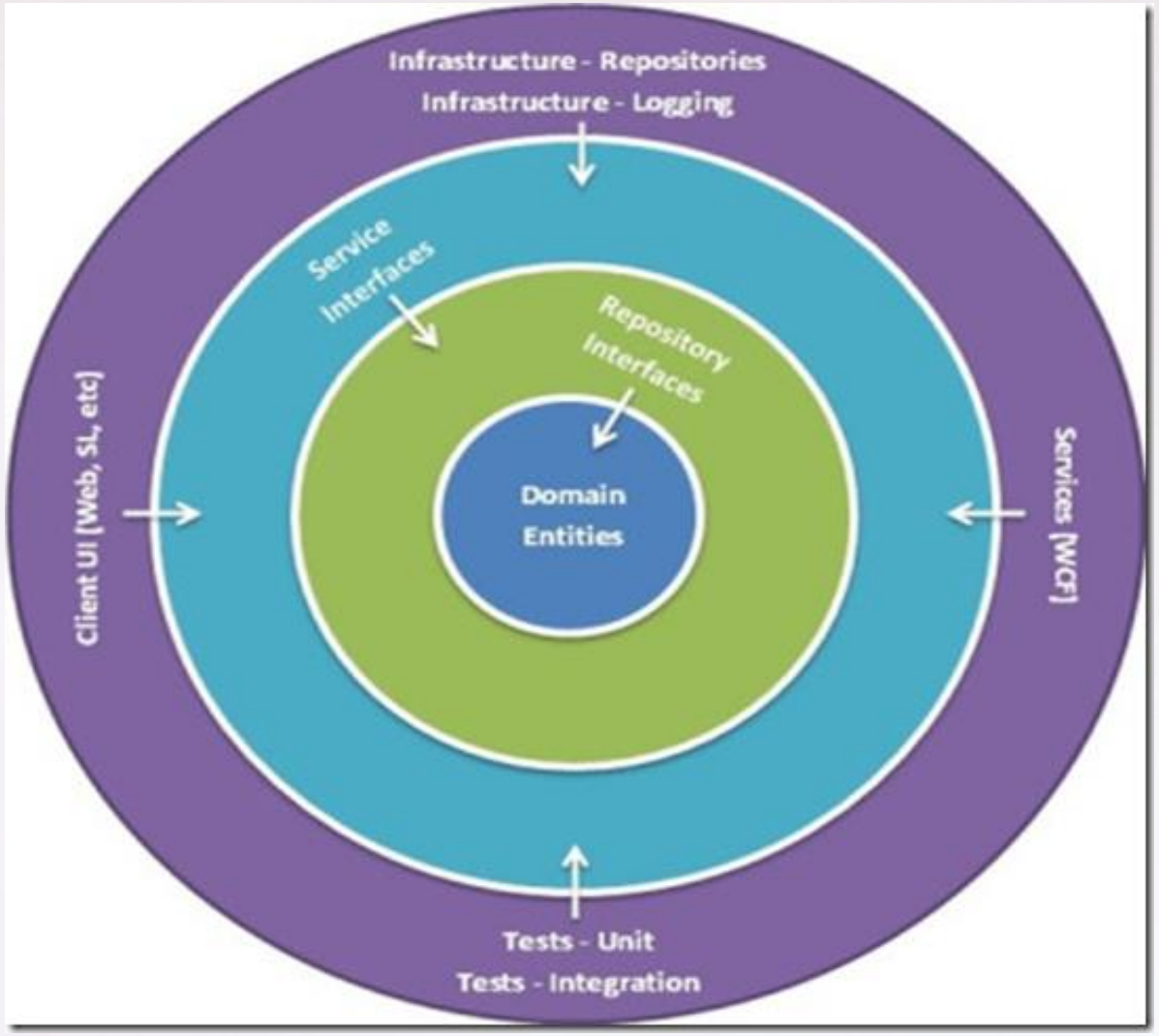
Hexagonal Architecture

Clean Architecture

MUSIC CITY CODE

How?

Organize Your Code



CASA Development Model

Core AS Application

MUSIC CITY CODE

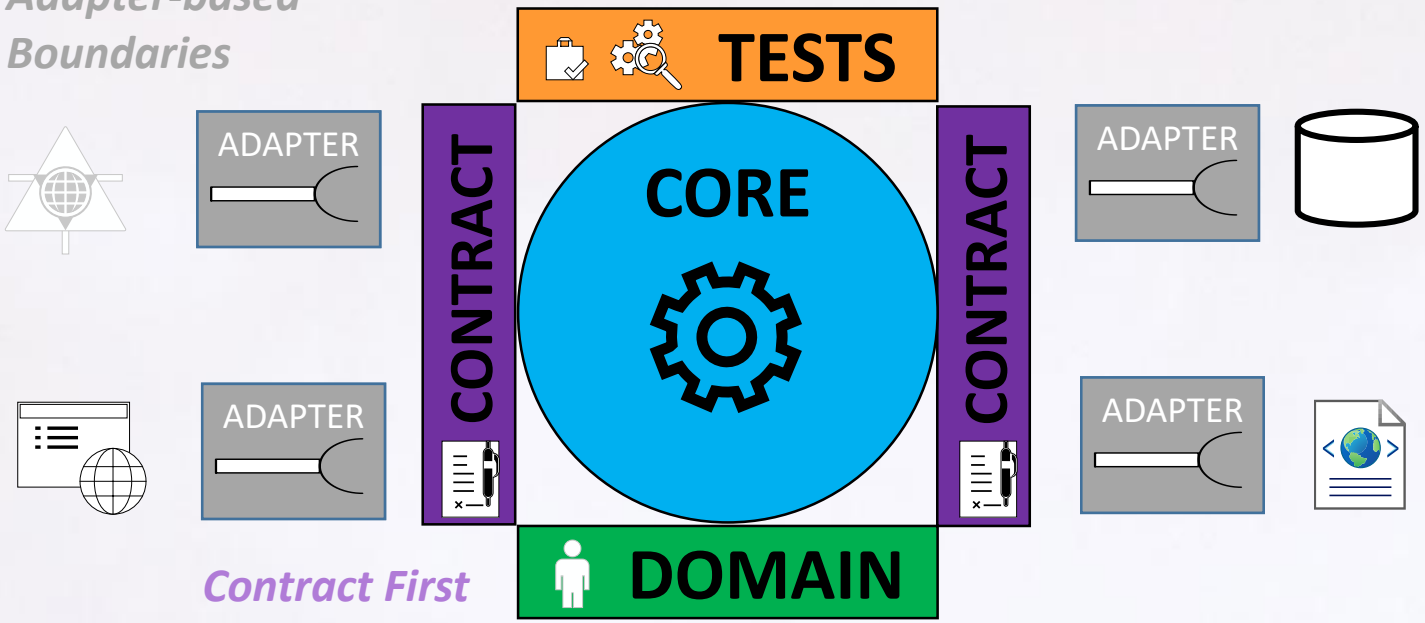
How?

Organize Your Code

Test Driven/ Test Early Approach

TDD *BDD* *ATDD*

Adapter-based Boundaries



Contract First Development

Domain Driven Design

MUSIC CITY CODE

How?

Best Practices & Patterns

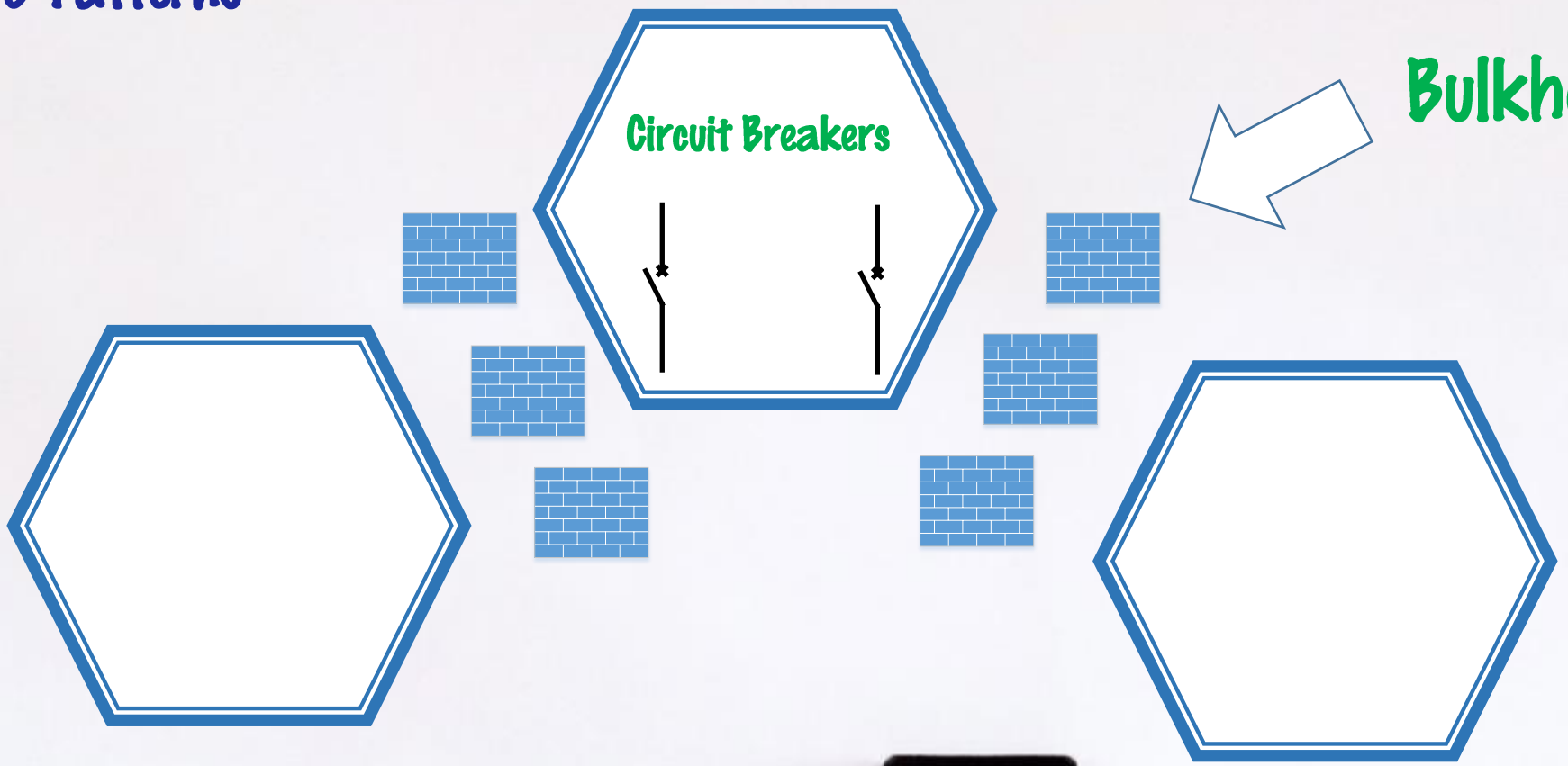
- SOLID Design Principles
  - You know these right?
- Domain Driven Design
  - State Objects
  - Core Business Logic
  - Other Concerns
- Contract First Development
  - Customer Driven Contracts

MUSIC CITY CODE

How?

Best Practices & Patterns

- Circuit Breakers
- Bulkheads



MUSIC CITY CODE

How?

Organize Around Capabilities

- What is a Capability?

Any business function that can be coded as a stand alone

Any business function that can easily be handed off to another team

Any business function that can be handled by another company

MUSIC CITY CODE

How?

Organize Around Capabilities

- Which of these are capabilities?

Shopping Cart?

Purchase Path?

Company Catalog?

User Access?

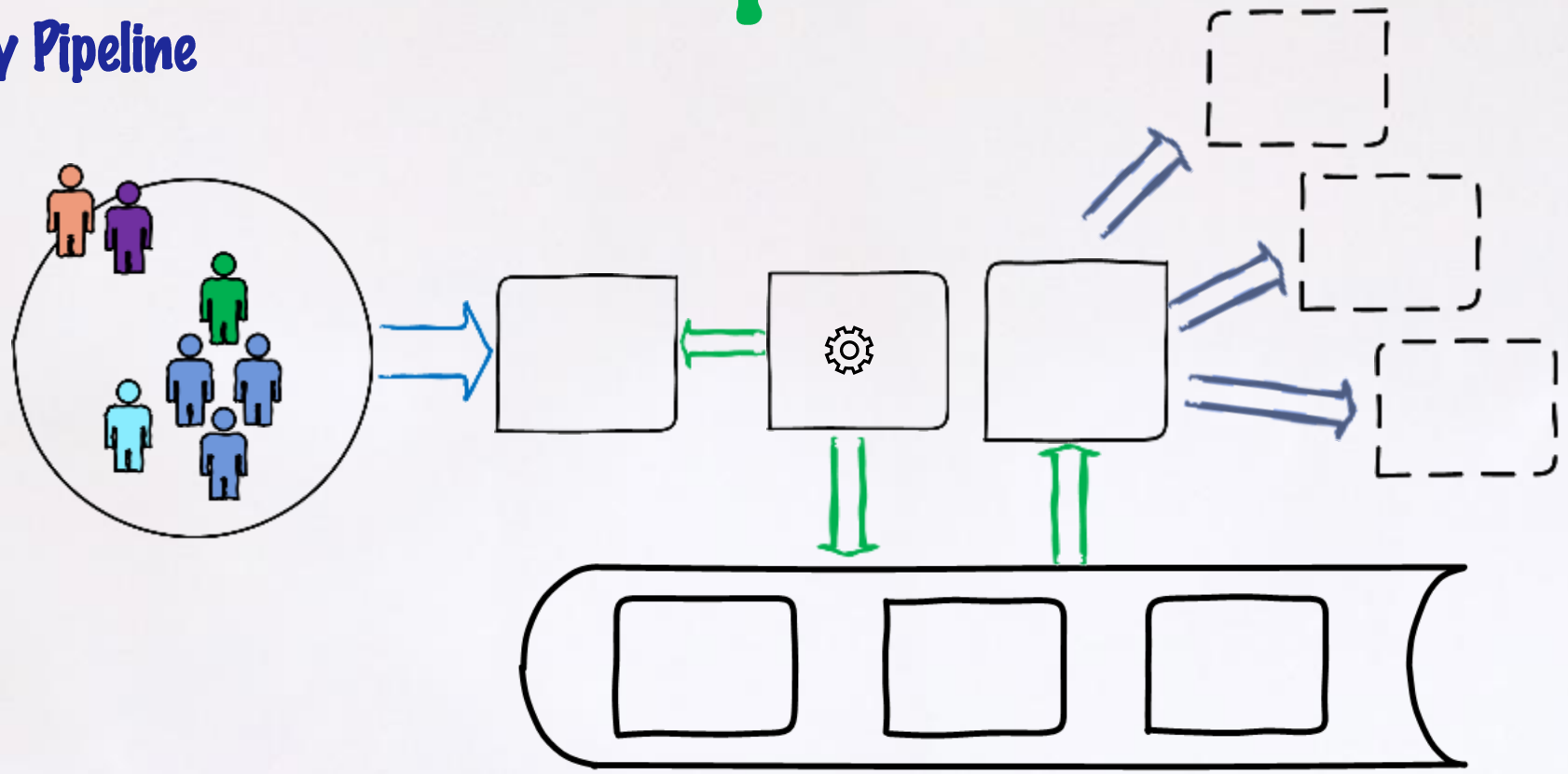
Database Access?

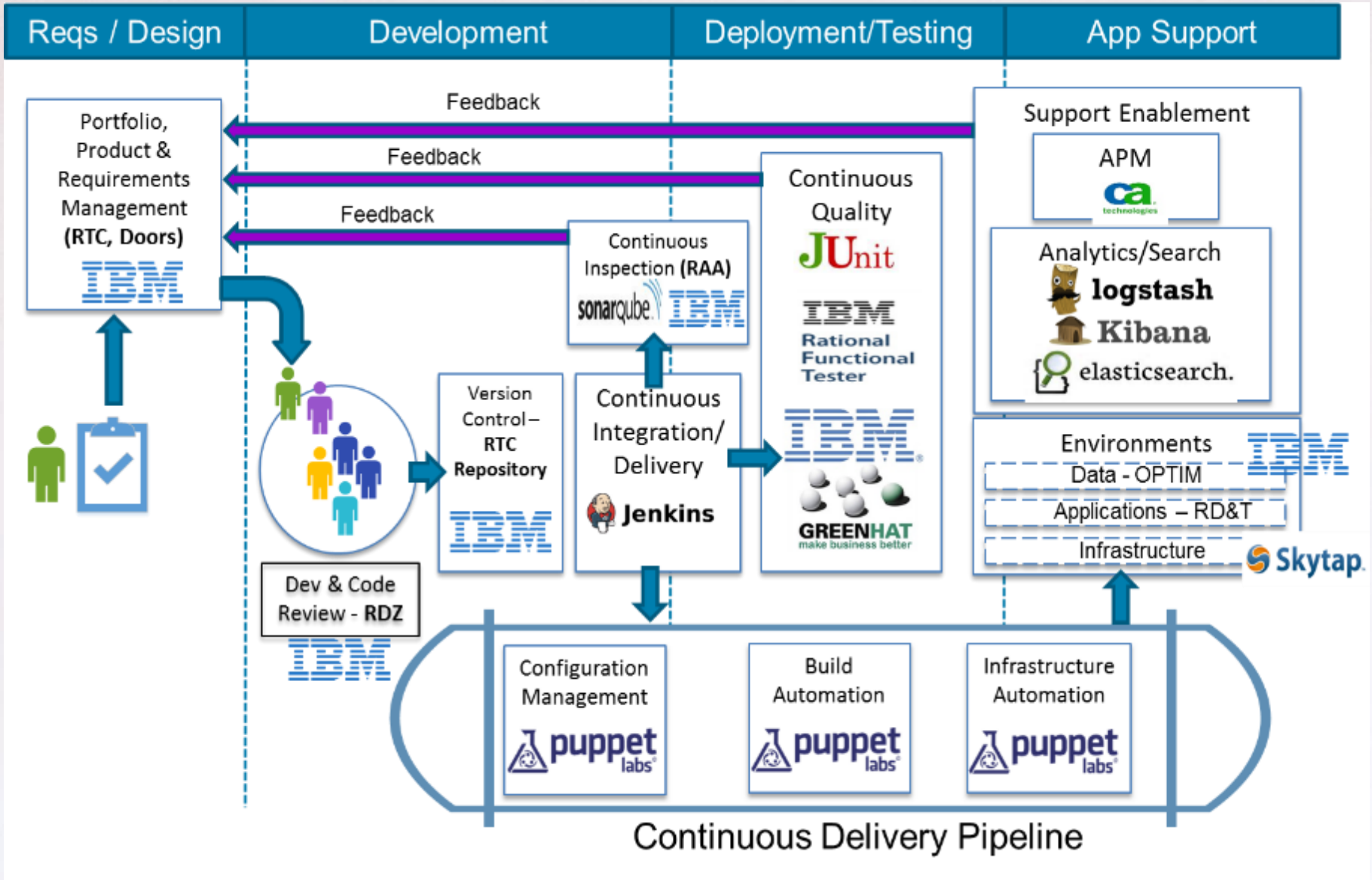
MUSIC CITY CODE

DevOps

How?

Automate the Delivery Pipeline

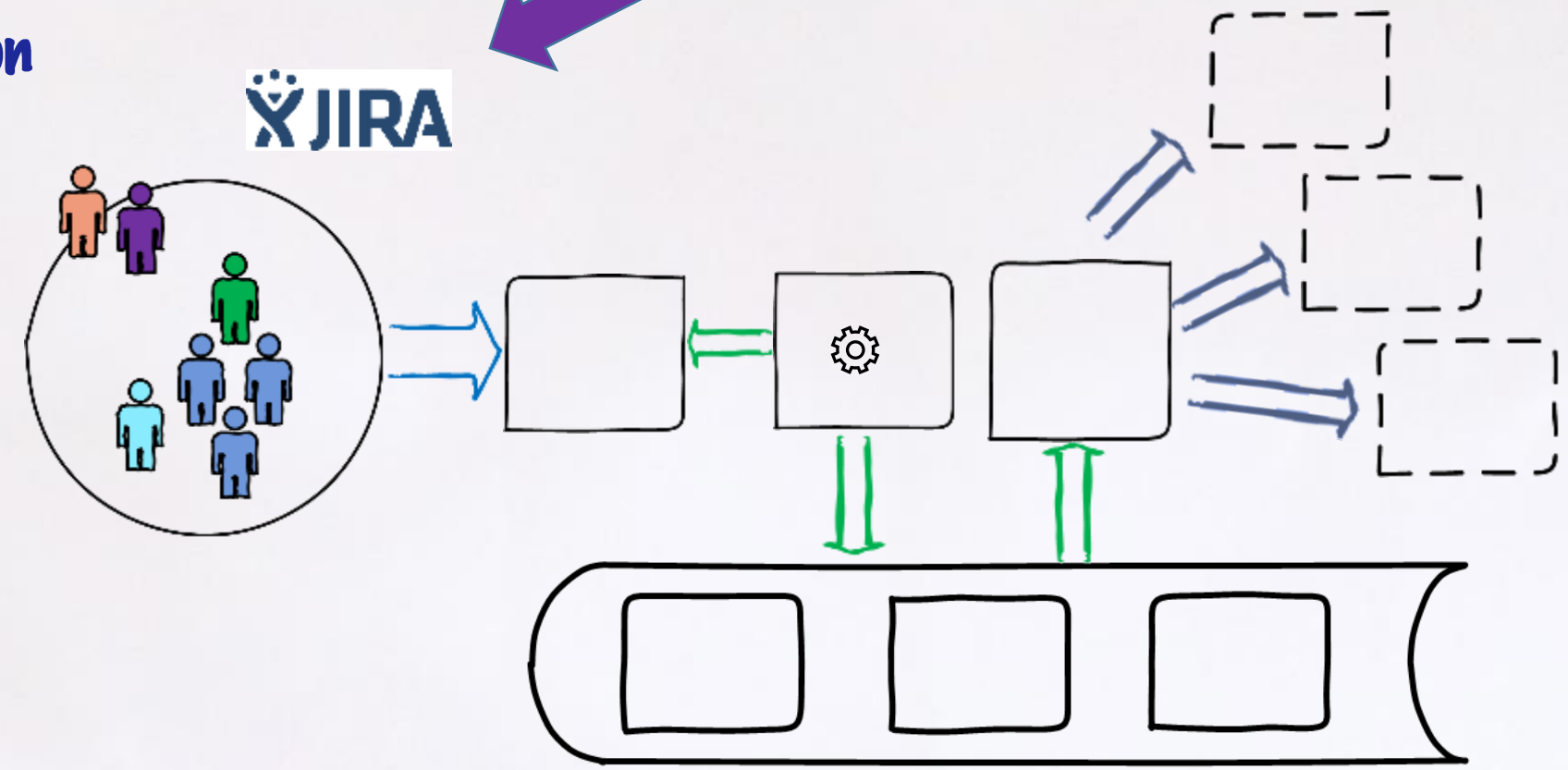




MUSIC CITY CODE

How?

Monitor Your Solution



MUSIC CITY CODE

How?

Abstracting Service Boundaries

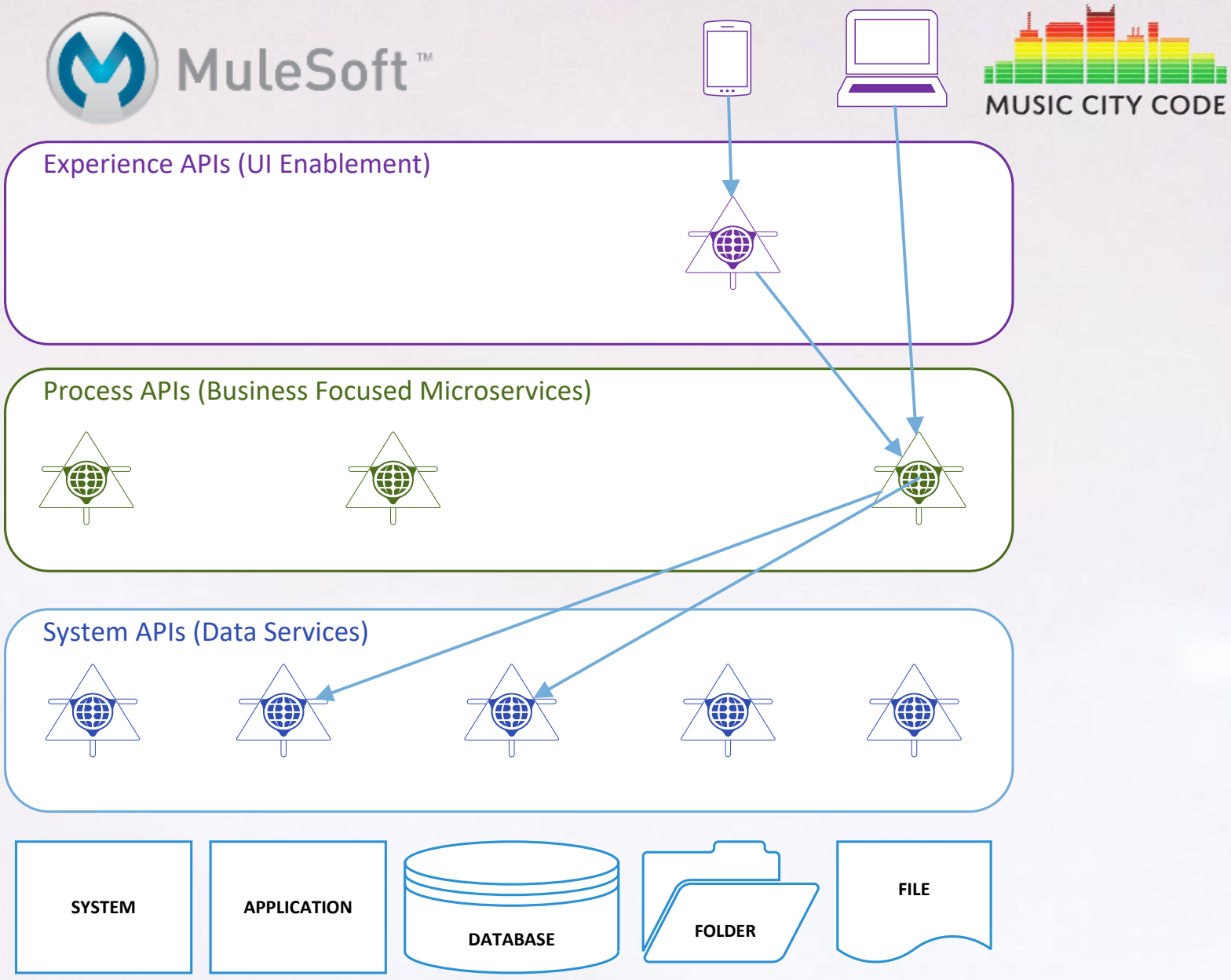
- How do we abstract?
  - API Management
  - Message Queueing

Stay in the room for Shawn's talk (NEXT) to learn more about abstraction with an event-driven approach

How?

Abstract

- No ESB, right?



MUSIC CITY CODE

Here in the Real World

Inspired by Real Events

MUSIC CITY CODE

Real World

Determining Services: Decomposition

Let's Make Asssumptions

- We are a board game company
- We recently completed a kickstarter
- The game is selling
- We need to go beyond the kickstarter and sell on our own

MUSIC CITY CODE

Shopping Cart

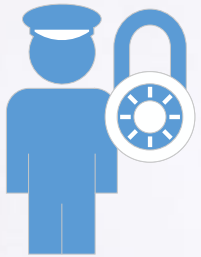
Real World

Determining Services: Decomposition

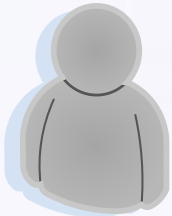
Shopping Cart

Item Lookup

Wish Lists



Security



User

Payment

MUSIC CITY CODE

Real World

Determining Services: Decomposition

What if we are green field?

- Same Process
  - Gather Requirements (stories?)
  - Find business capabilities

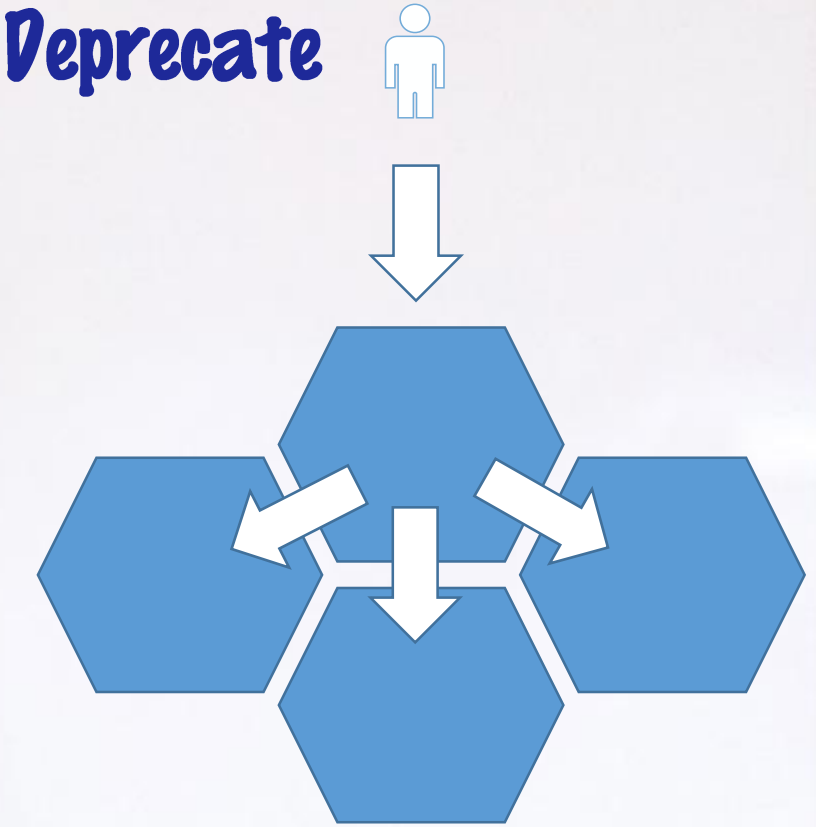
MUSIC CITY CODE

Real World

Determining Services: Decomposition

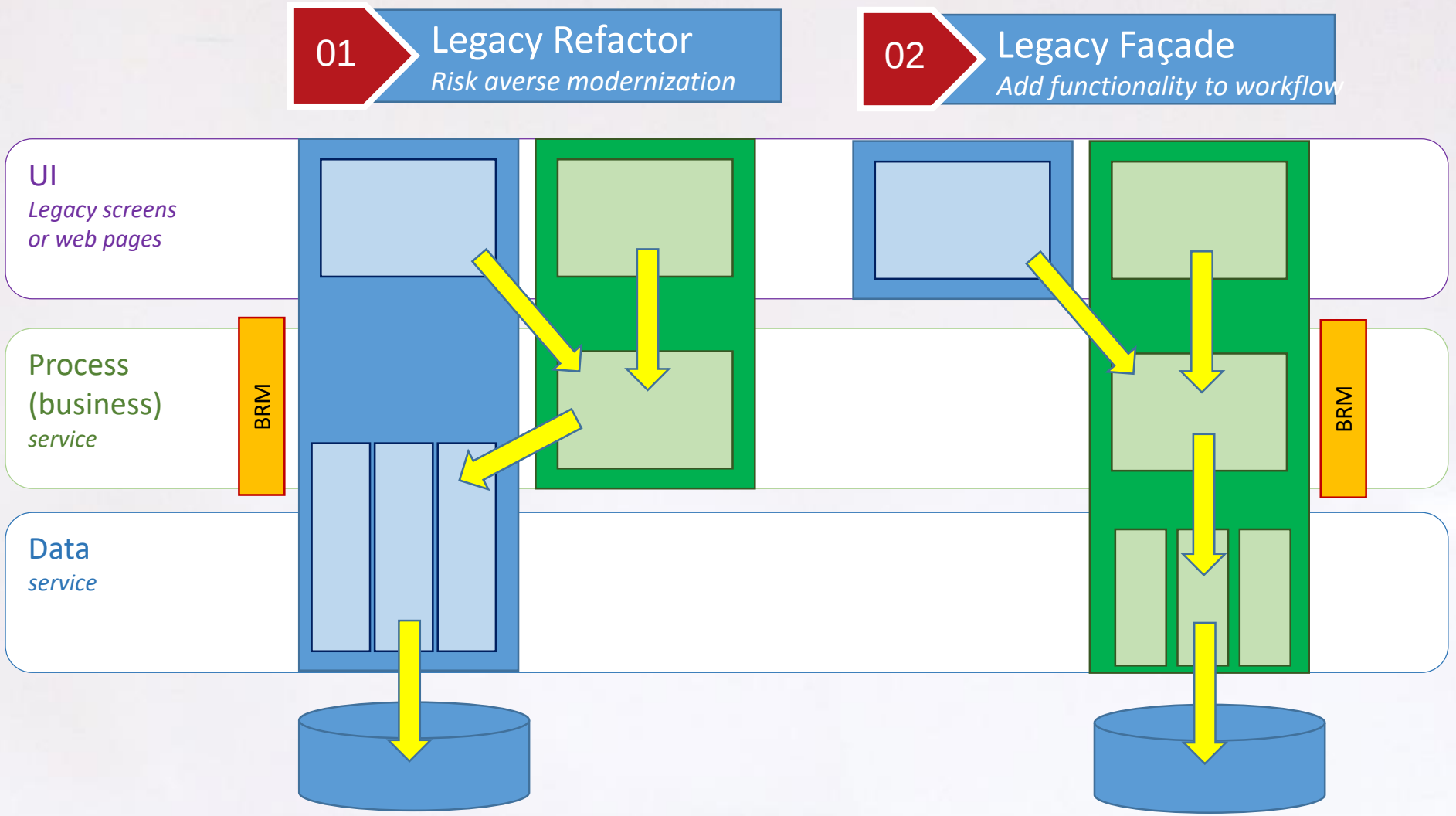
What if I have larger services?

- Orchestrate and Deprecate



MUSIC CITY CODE

Real World Legacy Modernization

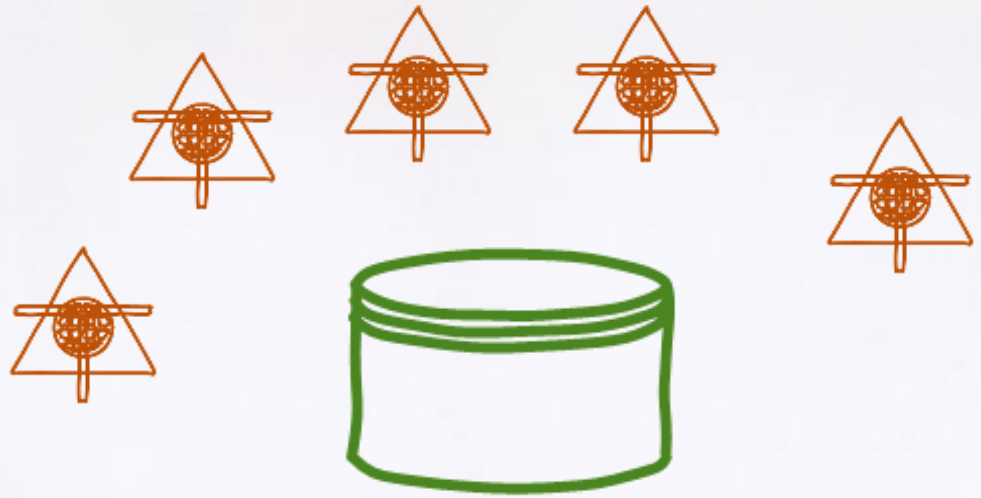


MUSIC CITY CODE

Real World

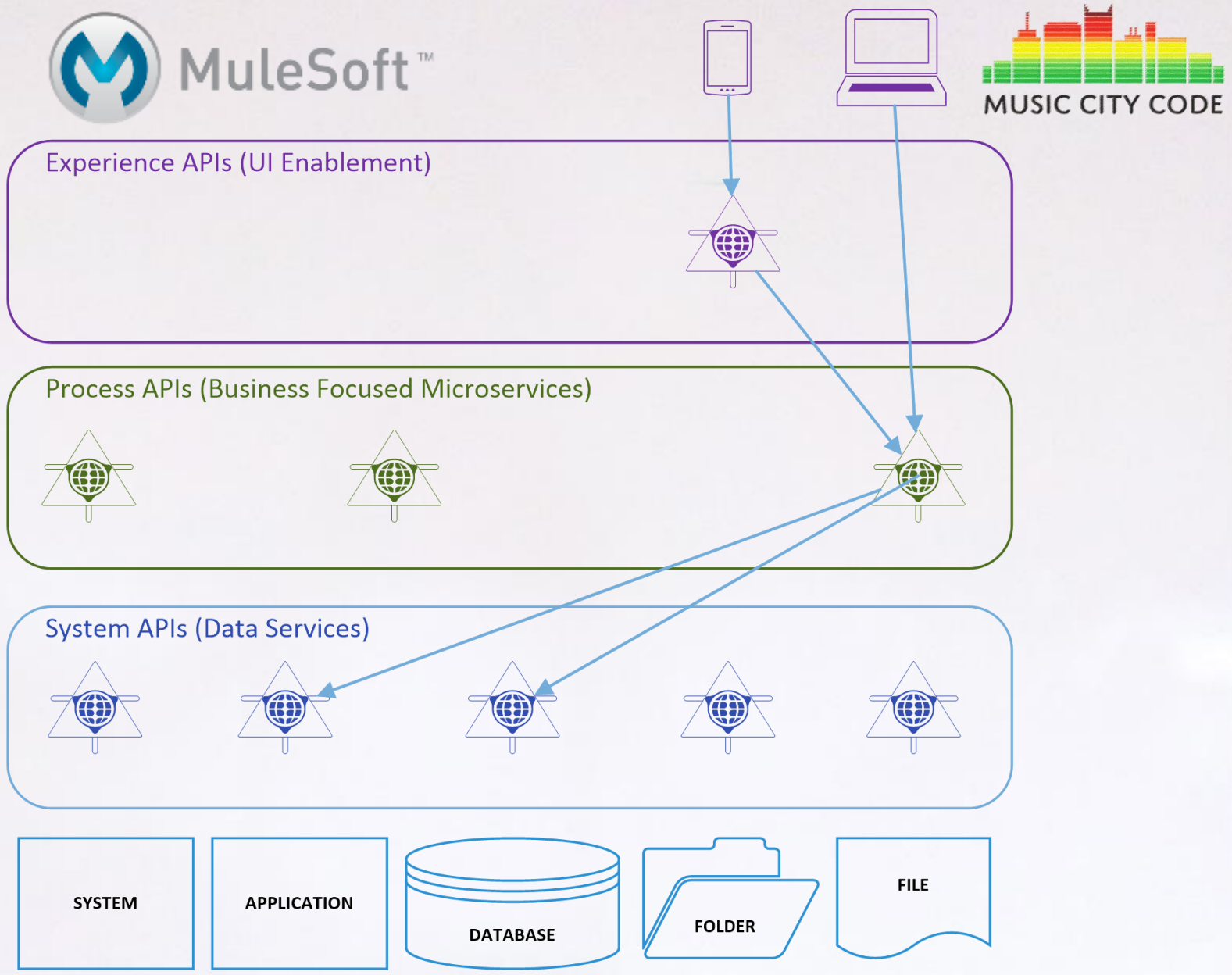
Data Focused Microservices

- Don't do this!!!



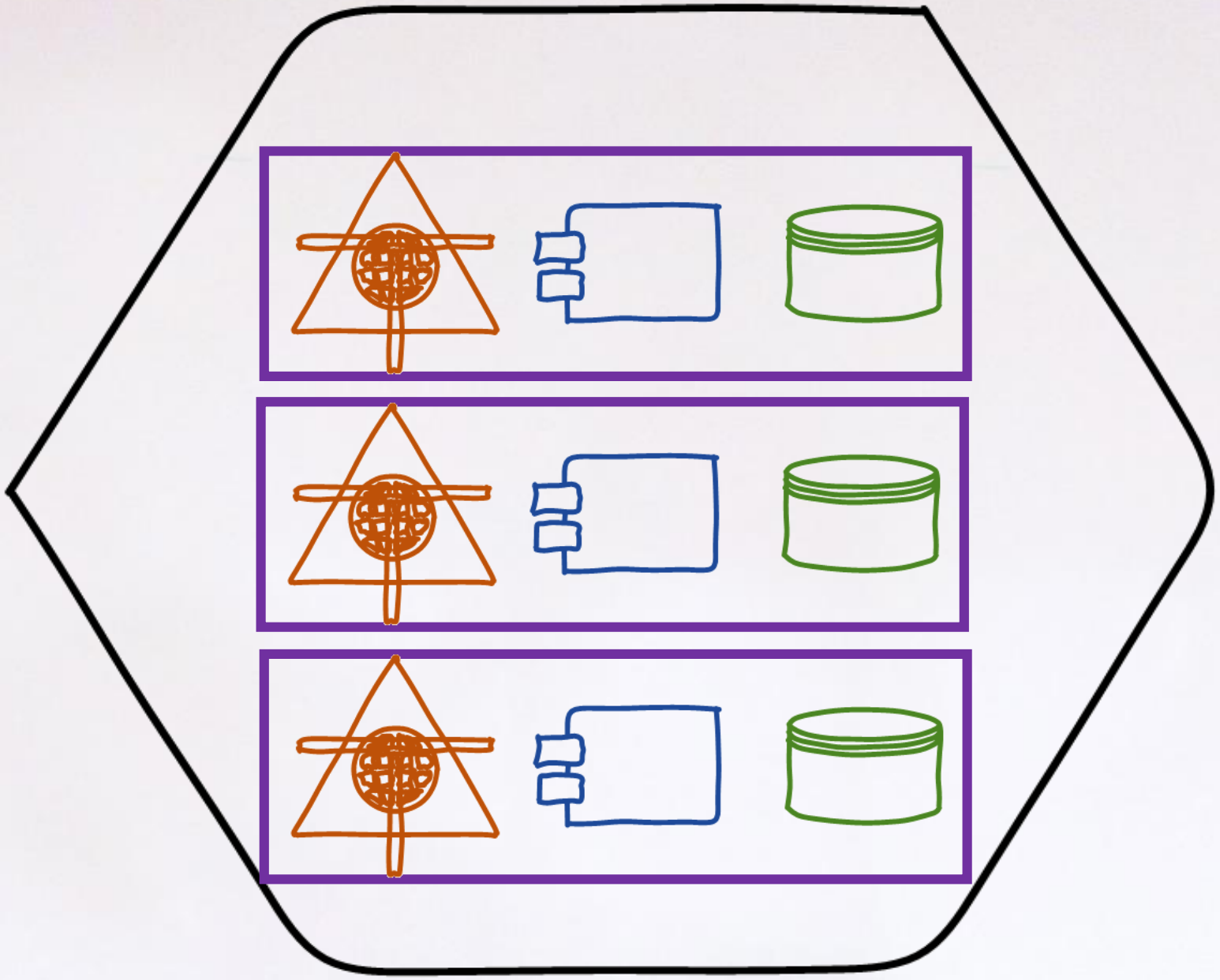
Real World

Data Focused Microservices



Real World

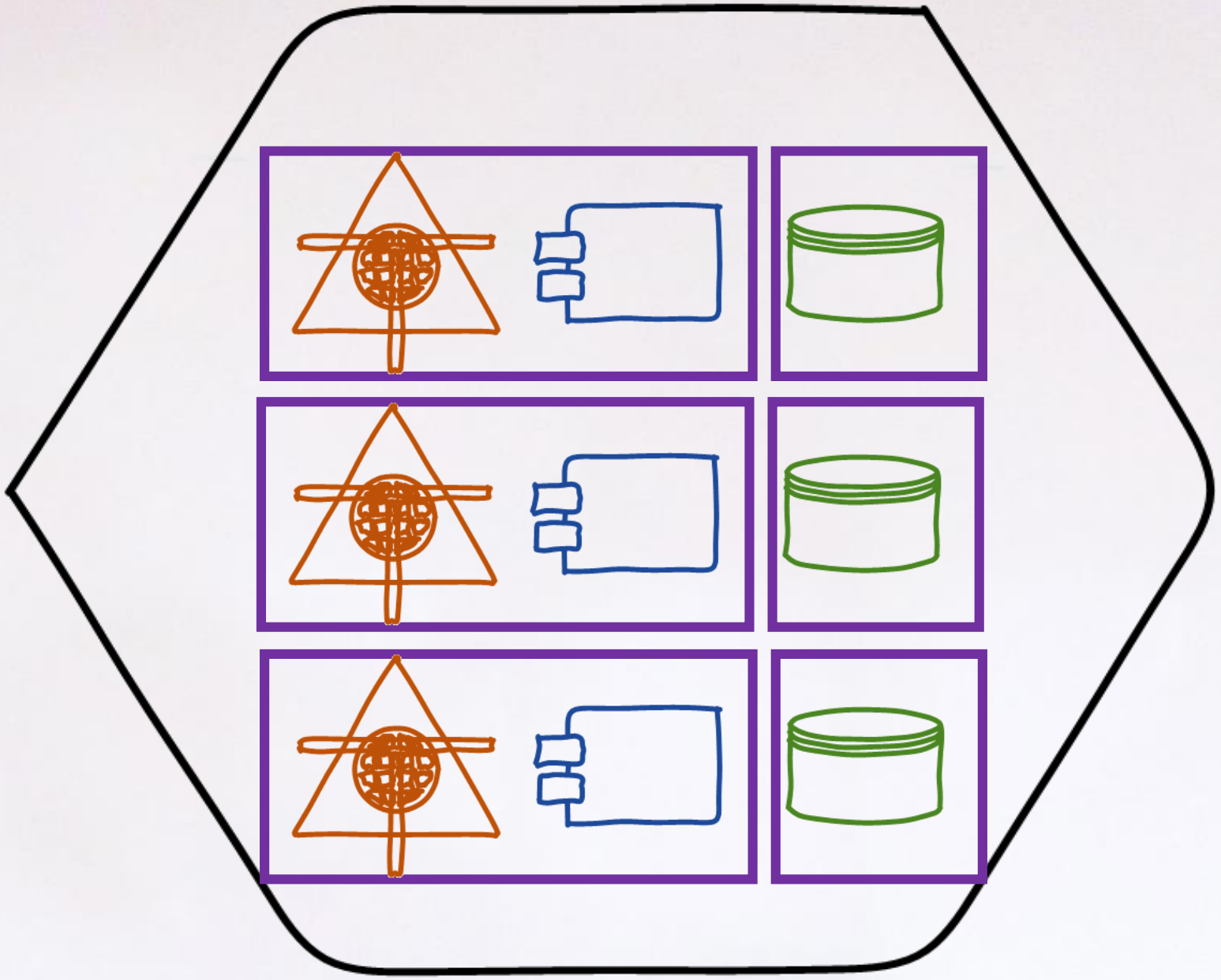
Scaling



MUSIC CITY CODE

Real World

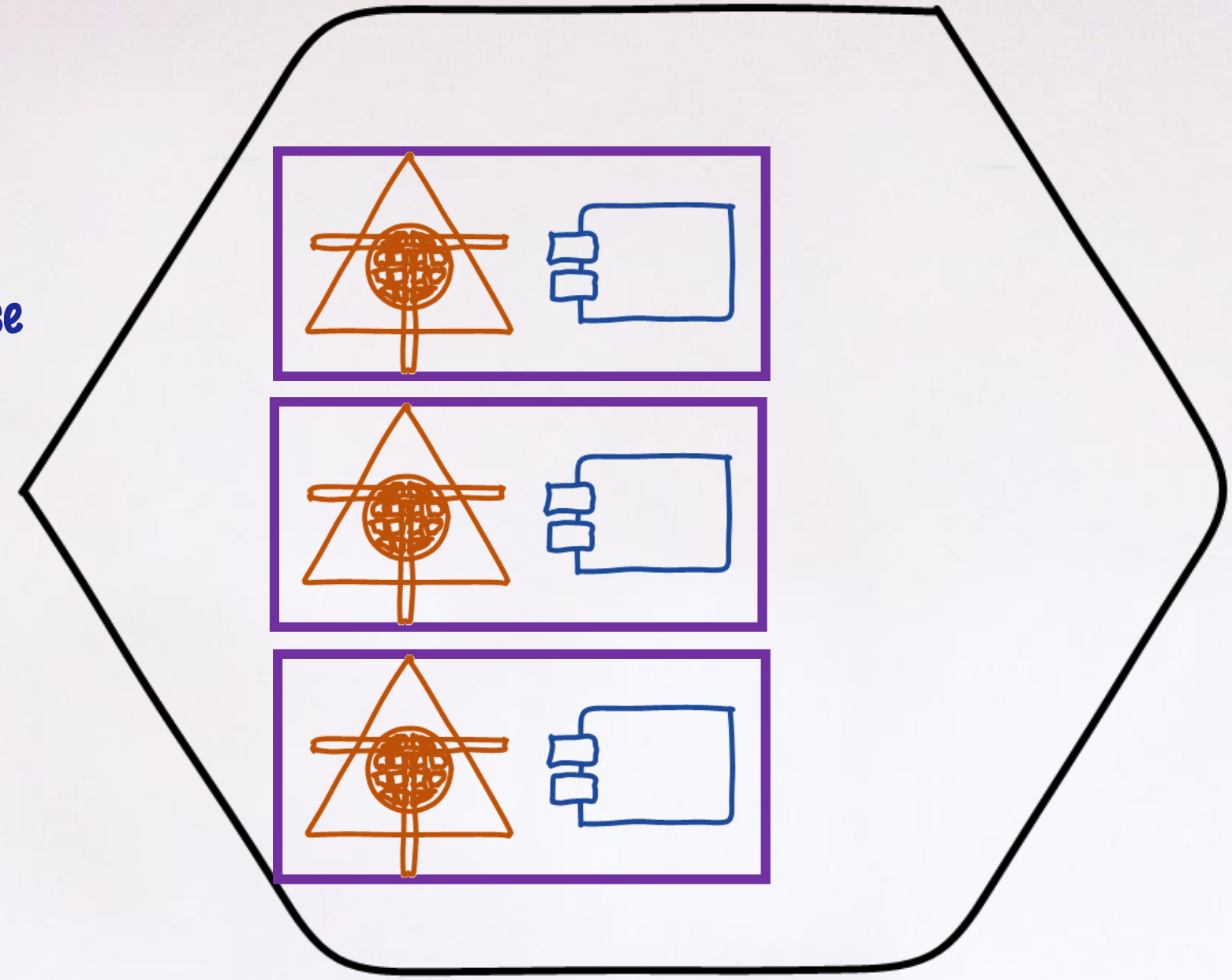
Scaling



MUSIC CITY CODE

Real World

Central Database



MUSIC CITY CODE

MUSIC CITY CODE

Real World

Central Database

Best Practices

- A single microservice owner for each table – if possible
- All access through the microservice

MUSIC CITY CODE

Real World

Other Notes

- In a purist environment, you will have to consolidate the data at some point
- Be prepared to introduce Master Data concepts
- How do you eat an elephant?
  - One bite at a time

MUSIC CITY CODE

Other Random Bull****!!!

MUSIC CITY CODE

Thanks to our Sponsors

Hall of Famers

INGRAM®

Availity®

Rockstars

iVision®

#buildwhatmatters
LIFEWAY LABS
NASHVILLE

TEKsystems
Our people make IT possible

Code Mash Edition

MUSIC CITY CODE

MICROSERVICES

Lessons from the Trenches

@gbworld