

# *Enabling Customer Obsession and Innovation through Large- Scale DevOps Practices at*



Marcilio Mendonca (marcilio@amazon.com)

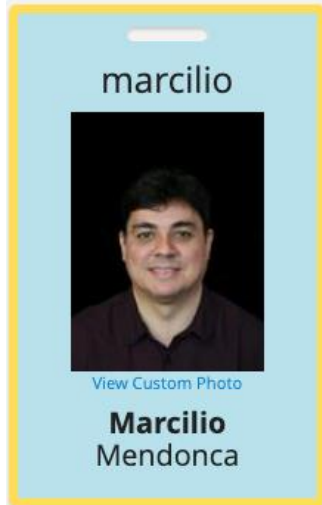
Sr. Solutions Architecture

AWS Global Solutions Prototyping

<https://www.linkedin.com/in/marcilio/>



# Who's your presenter today?



- Brazilian-Canadian living in the US 😊
- 7-year Amazonian playing multiple roles (SDE, consultant, SA)
- Spent 30 years in academia + industry (software engineering)
- BSc, MSc, PhD in Software Engineering
- Love writing AWS blogs, PR/FAQs, and scientific articles



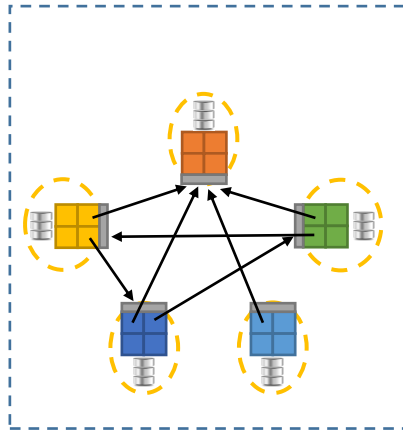
Some of my hobbies

Disclaimer: I must include soccer in order to retain my Brazilian citizenship

# Agenda

## Four Pillars of **amazon** 's DevOps Journey

Micro-services



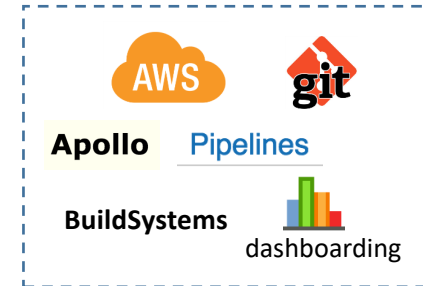
2-Pizza Teams



Agility



Tool Support





# Let's start at the Amazon warehouse...

Worldwide operation, millions of transactions, thousands of workers, hundreds of processes





How to set up an IT organization (processes/people/tools) where Customer Obsession, Agility, and Innovation are primary components of its culture?



# What type of organizations are doing this well?

Customer Obsession, Agility, Innovation



Question then becomes:

How can a large-scale organization  
operate like a start-up?



# Back to the Warehouse...

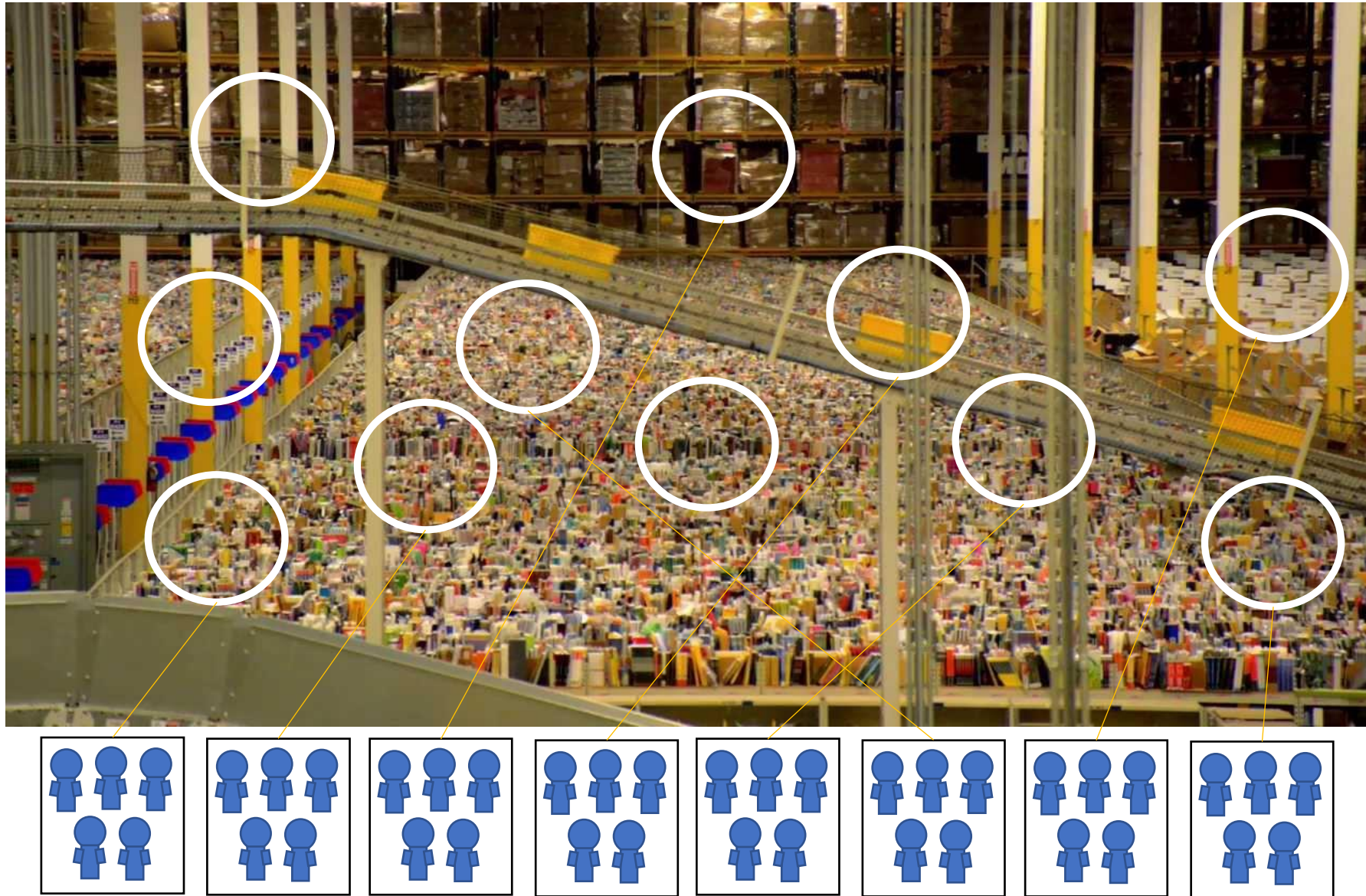
Identify and break warehouse processes into fine-grained units





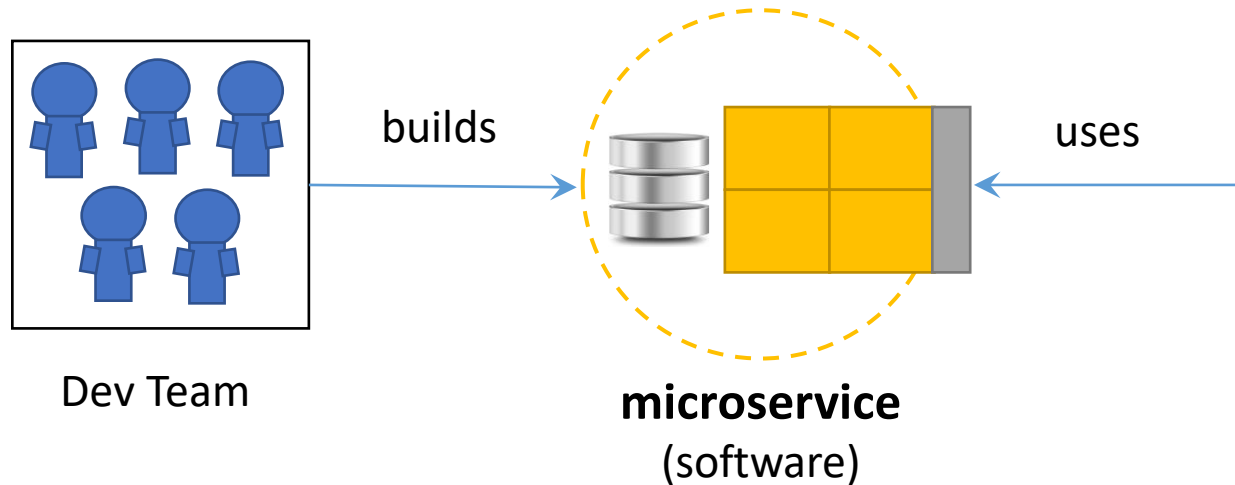
# Assign those units to small development teams

Amazon Fulfilment Technologies



# Now let each team operate as a start-up

Result: customer obsession, ownership, freedom to innovate, agility



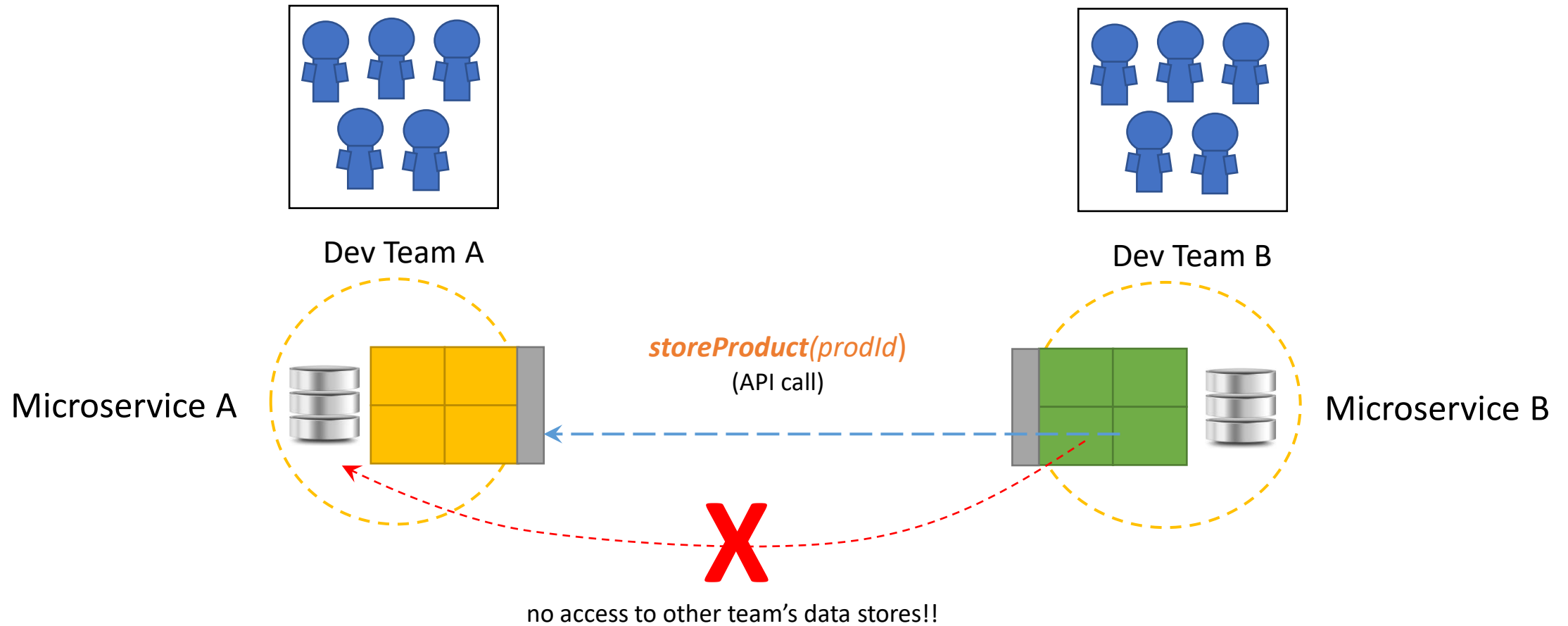
Process: Store products into slots



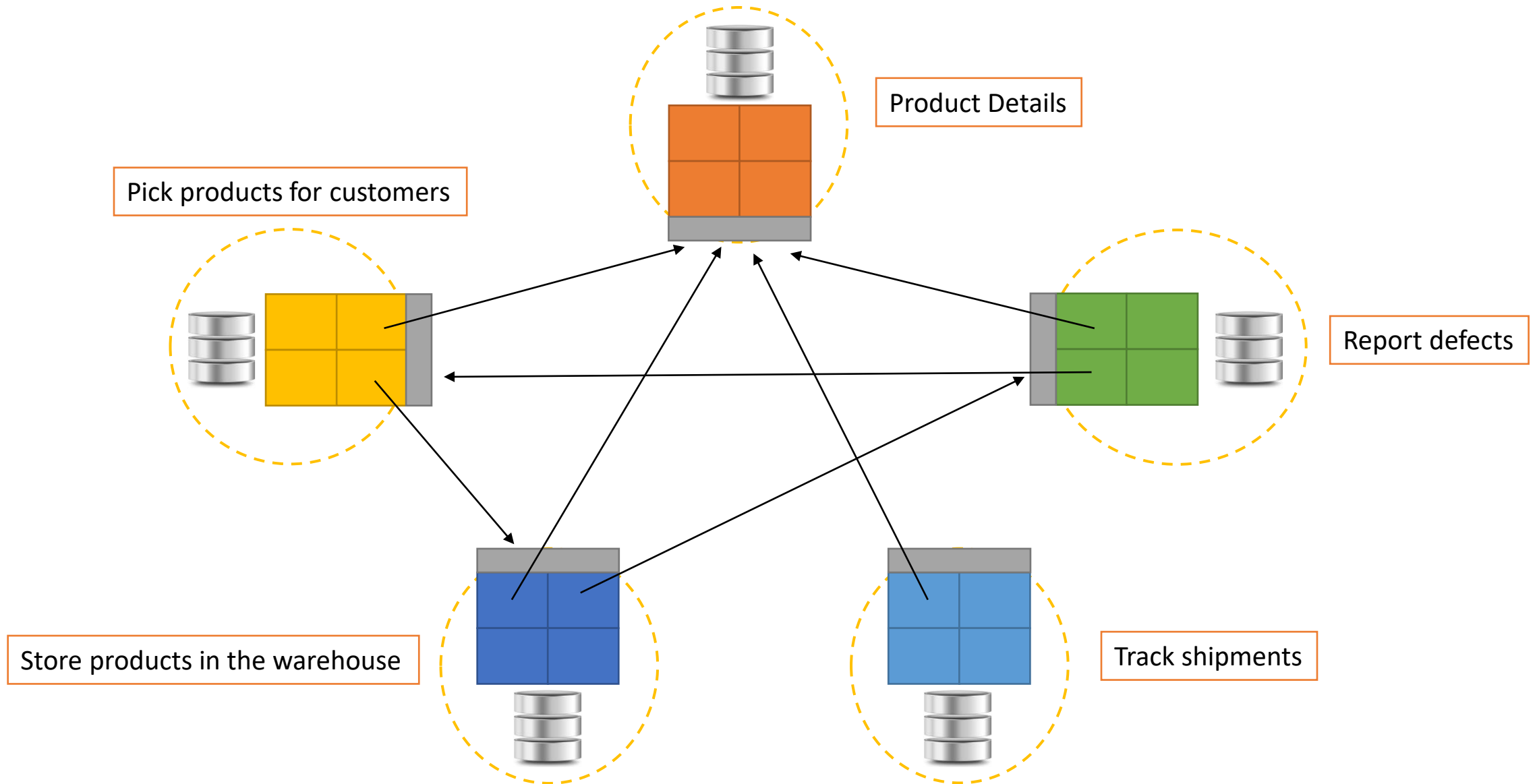


# Microservices can only be accessed via its public API

## Loosely-Coupled Software



# Web of loosely-coupled Microservices





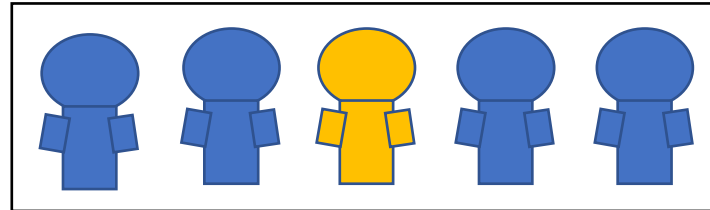
# Small Autonomous Teams treated as Start-ups

## 2-Pizza Teams



### Amazon 2-Pizza Development Team

few developers (SDEs) + manager (SDM)

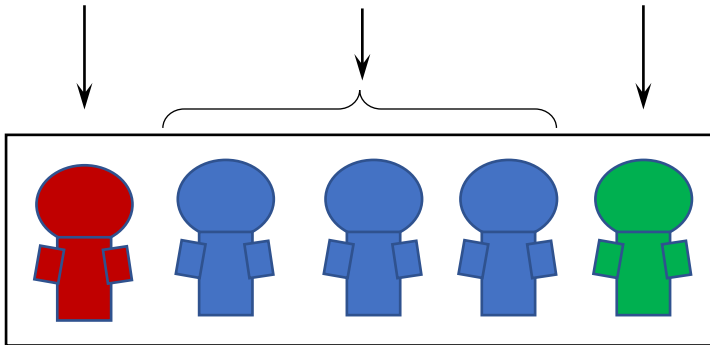


# Teams adopt an Agile methodology (eg, Scrum, Kanban, XP, etc.)

## Agile Teams

### Example of Scrum Roles

scrum master, developers, product owner

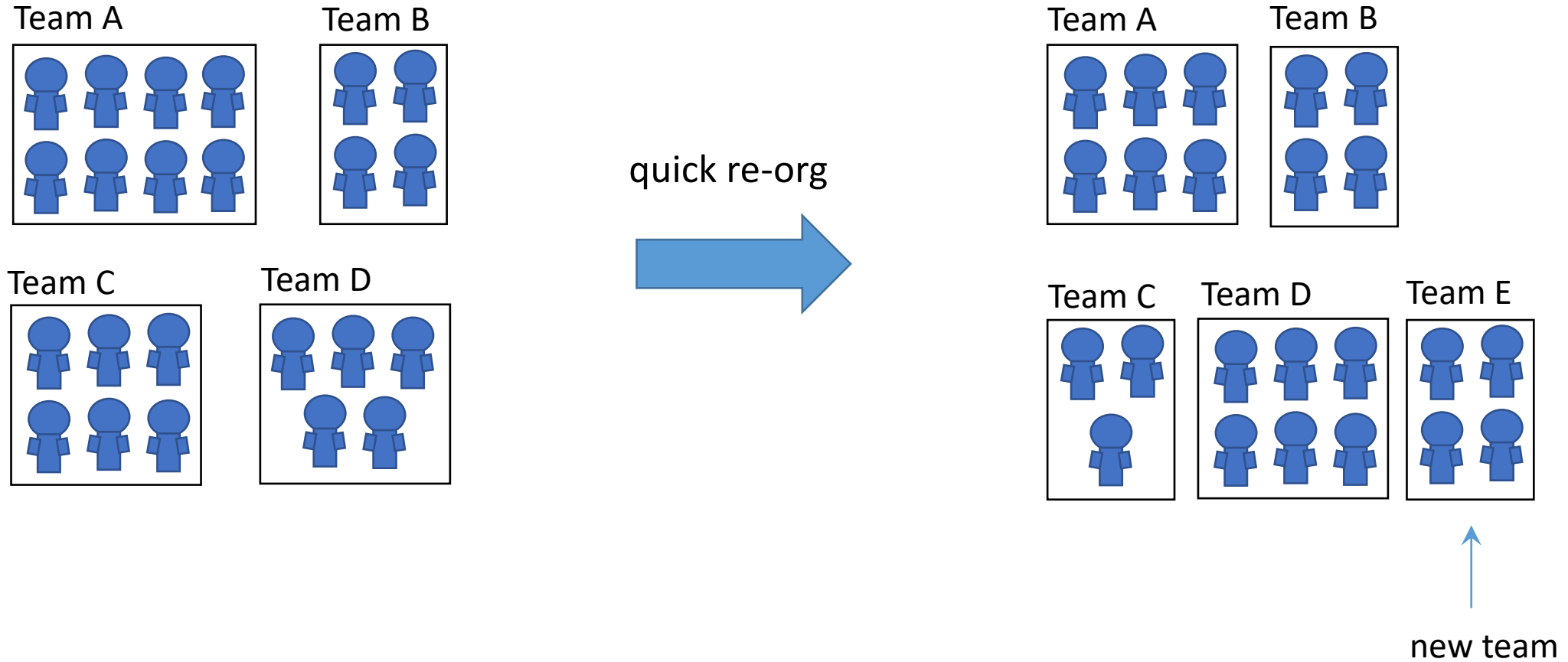


### Example of Scrum board

Stories	To Do		In Progress	Testing	Done
<div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text. Replace it with your own text.</div> <div>This is a sample text. Replace it with your own text.</div>
<div>This is a sample text. Replace it with your own text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text.</div> <div>This is a sample text.</div>	<div>This is a sample text. Replace it with your own text.</div>

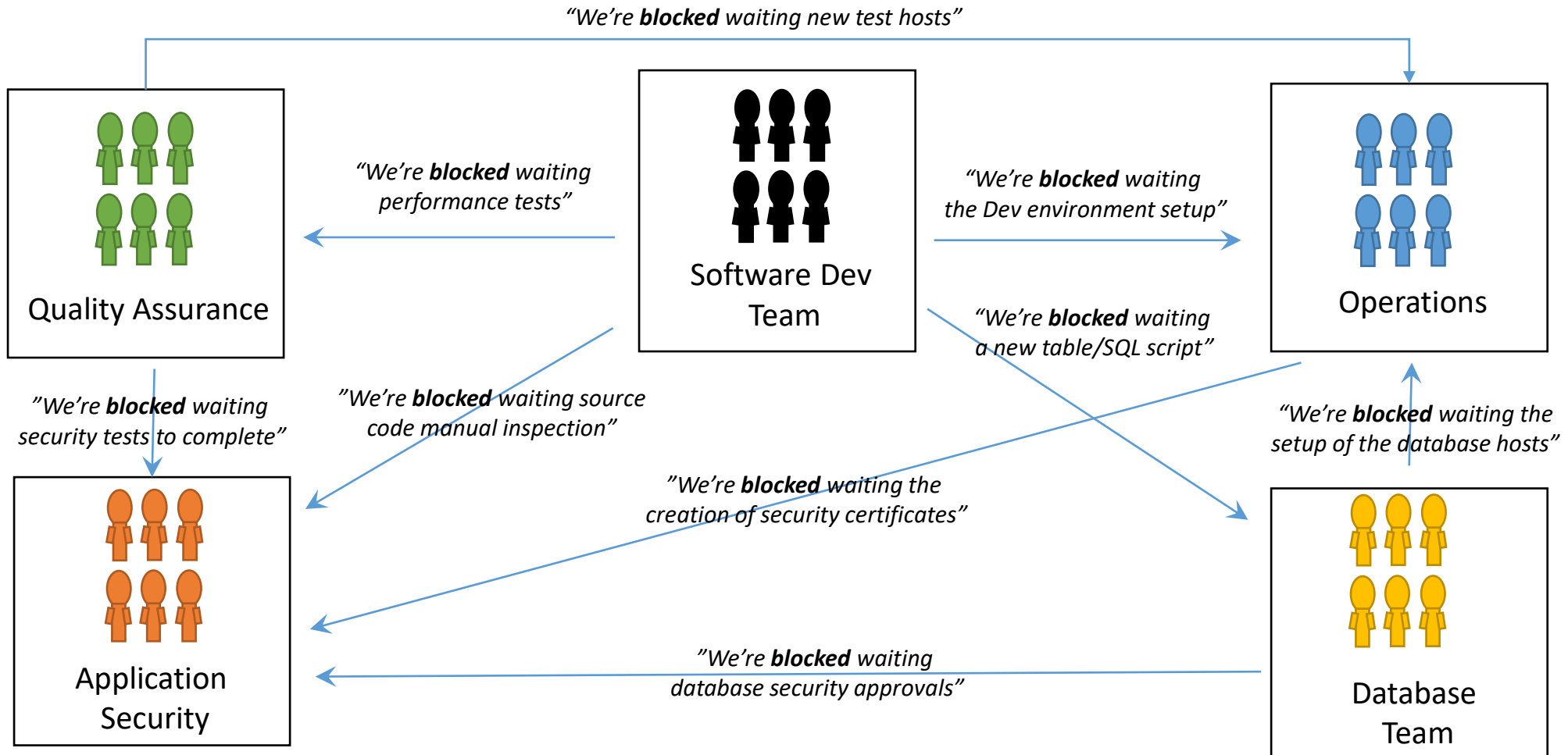


# Teams are flexible and can be quickly re-organized



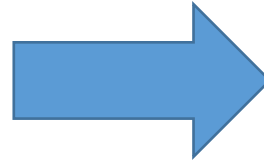
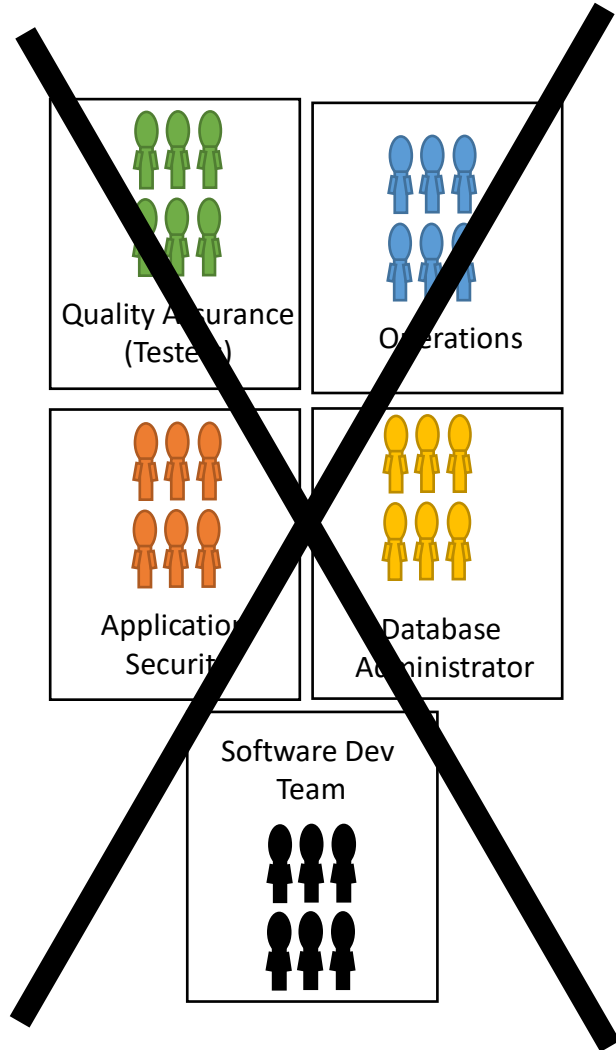
# Discourage “silos” in the organization

It leads to lack of Agility / Ownership

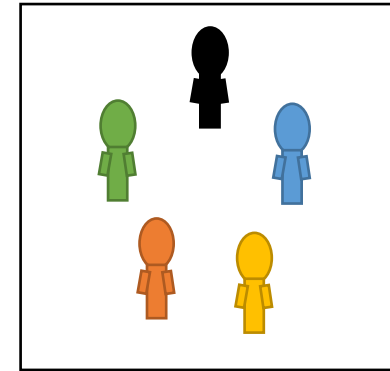


# Instead teams should operate in a DevOps setting

## Break the Silos!




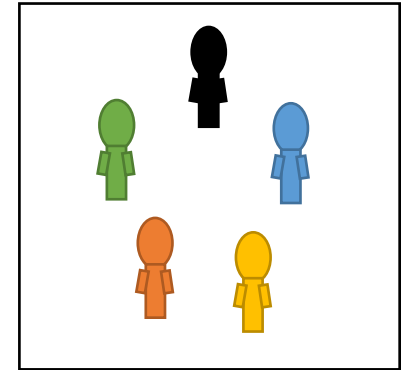
*“Working together”*





# Responsibilities of an Amazon 2-Pizza Team

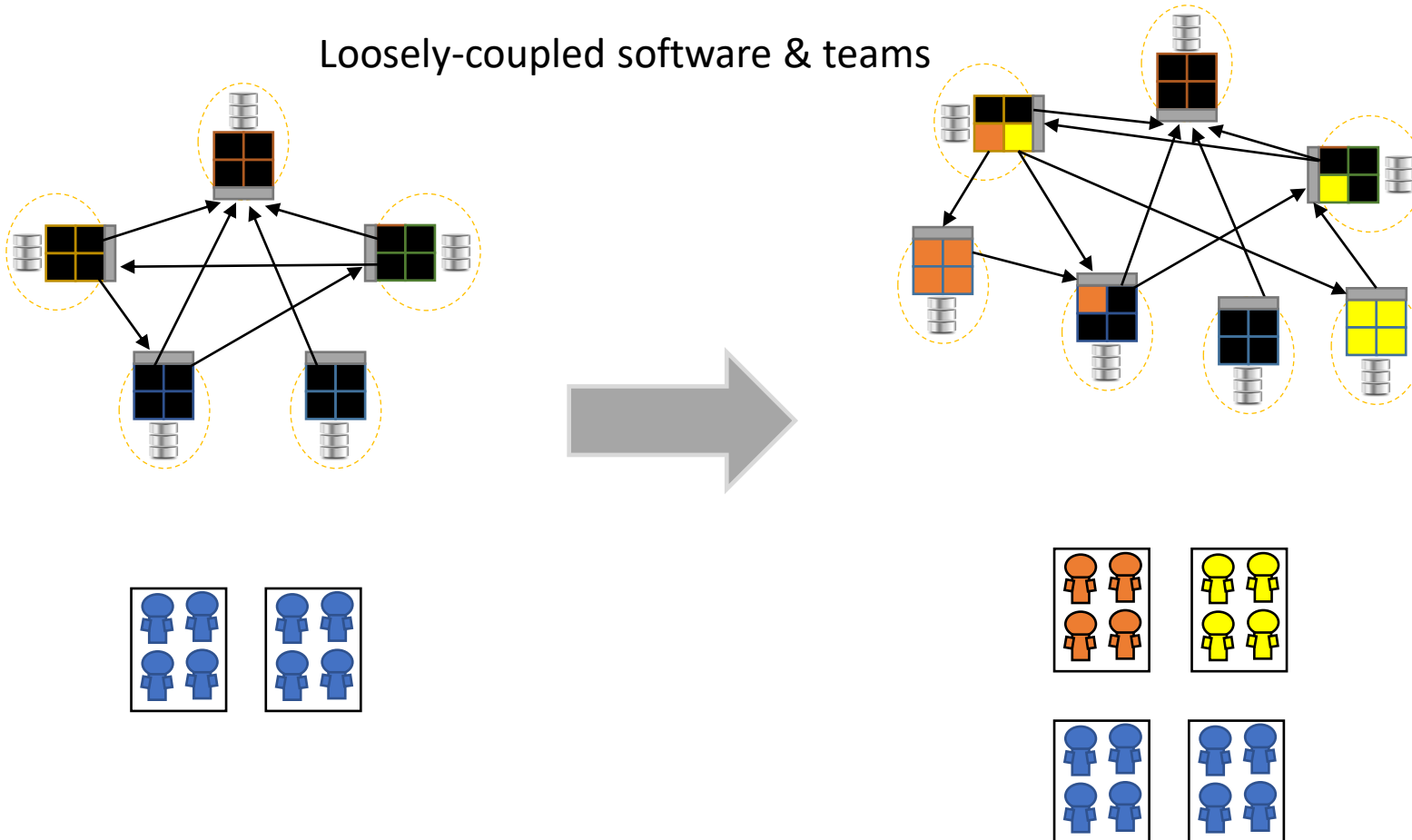
- Write application and infrastructure code
  - Review code of others
  - Write unit/integration/performance tests
  - Create database schemas/SQL queries
  - Create and automate CI/CD pipelines
  - Provision test and production hosts
  - Use secure APIs and apply security policies
  - Monitor production environments
-  Be on-call on a rotation-basis



# Continuous Innovation: Bananas, Robots & Agility

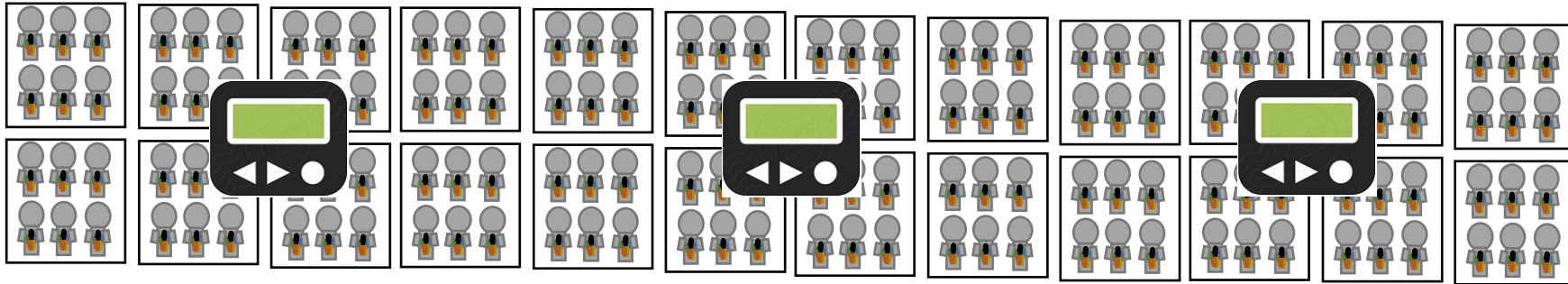
- Re-arrange and/or form new teams when needed
- Adjust and/or add new micro-services to address business requirements

Loosely-coupled software & teams



# amazon DevOps Teams

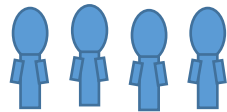
## Large-scale DevOps teams



Empower developers and let them OWN the system they build!

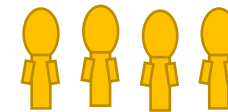


Tools, APIs, standards, best practices,  
documentation, training, reviews, etc.



Builder tools team

...



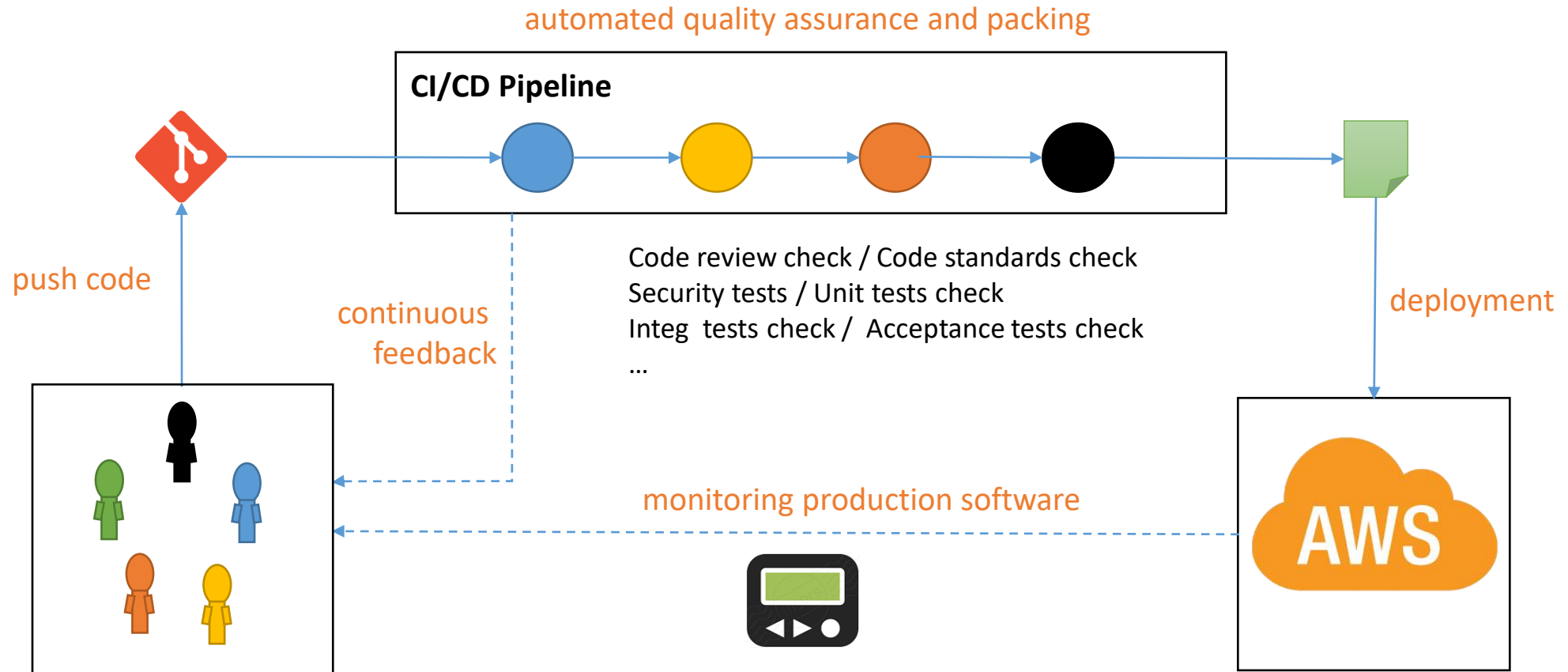
Application security team



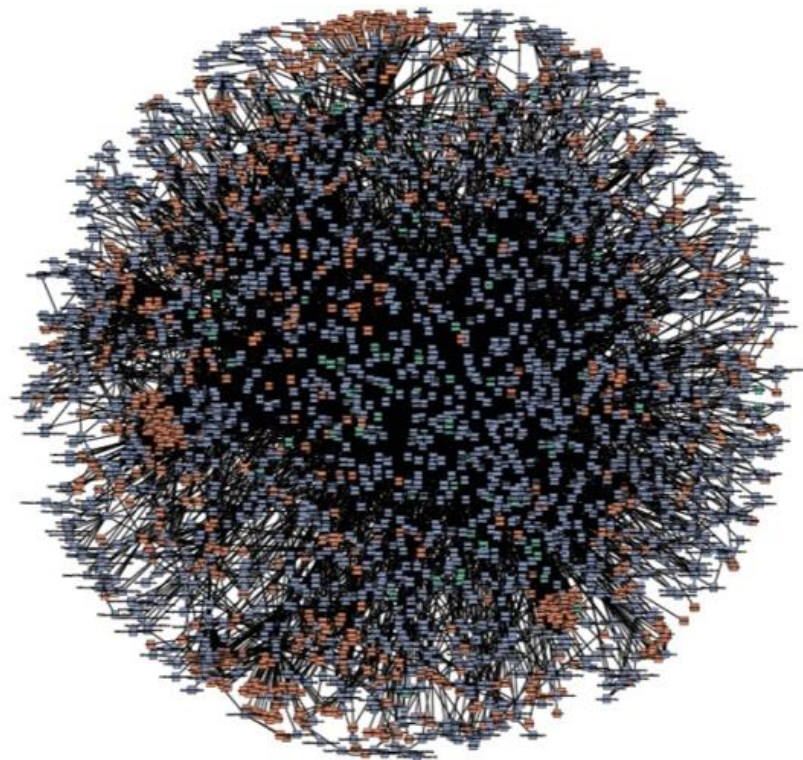
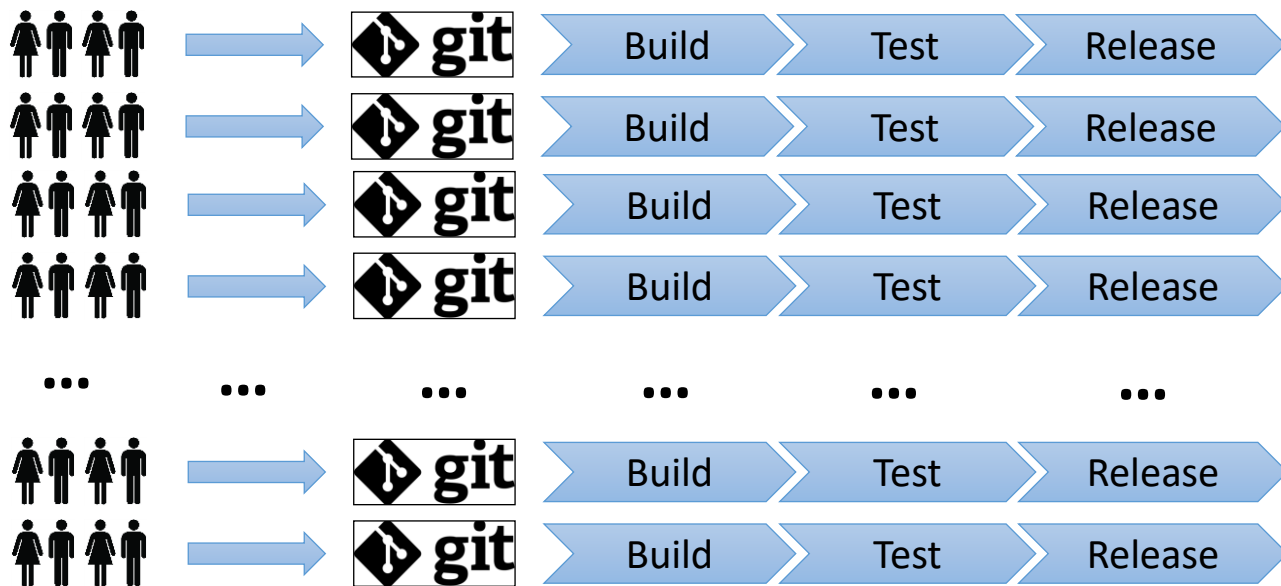
# Separation of Duties via CI/CD Pipelines

All team members contribute to the automation pipeline via code

Code pushed to production is subject to various automated checks



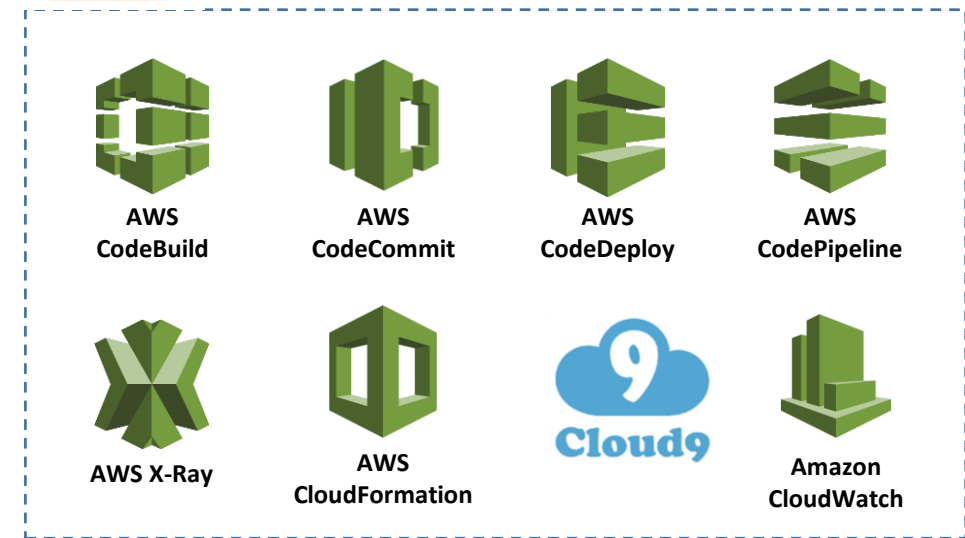
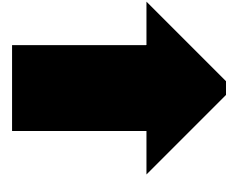
# Micro-services + 2-Pizza Teams + Tools + Agile = 50 million deployments a year (2014)



Gigantic Web of Micro-services

# Amazon Tools Inspired AWS DevOps Services

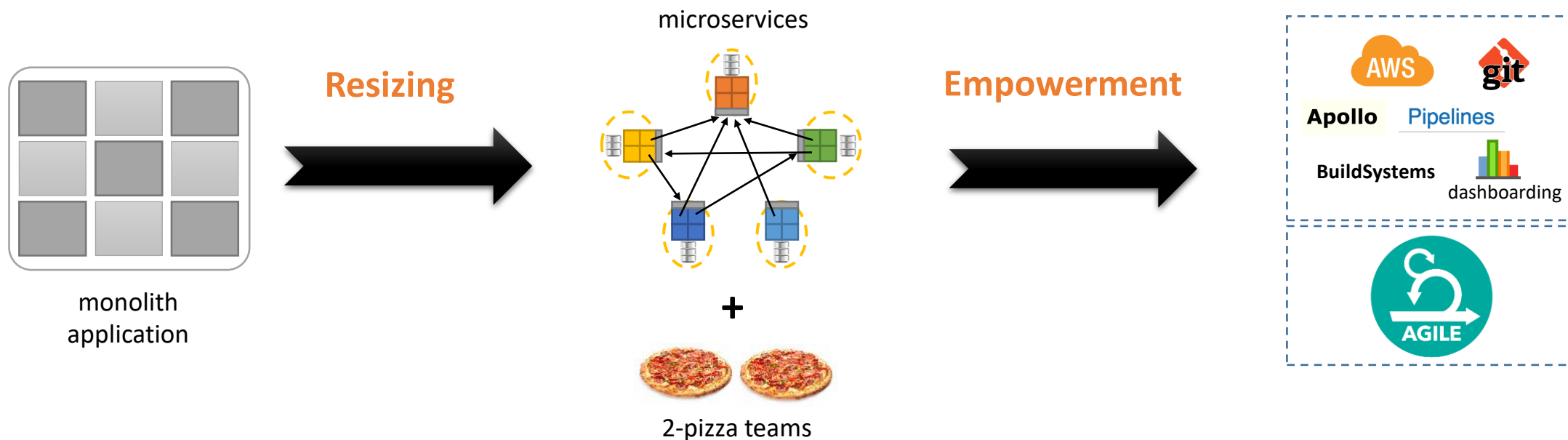
Amazon-style DevOps in the AWS Cloud





# Wrapping Up: Amazon's DevOps Journey

## Resizing + Empowerment



### Key Components of Amazon's DevOps Culture

- Loosely-coupled software via micro-service architectures
- Thousands of small, flexible, and autonomous teams (2-pizza start-ups)
- Company-wide adoption of standard internal DevOps tooling
- Company-wide adoption of Agile practices
- *Working Backwards* culture (start from customer requirements, write PR, manual, ...)

Thank you!

