SKILUP SM DAYS

by: DevOps Institute
ADVANCING THE HUMANS OF DEVOPS

The Three Basic Truths About Microservices

What you need to know to get started.

Tracy Ragan
CEO
DeployHub
@TracyRagan
Tracy@DeployHub.com

Agenda

Get your head around microservices

Know these three truths.

How to learn more.

Tracy Ragan
CEO
@TracyRagan

- CEO and Co-Founder – DeployHub, Inc.
- Founding Board member of the CD Foundation
- Founding Board Member Eclipse Foundation
- DevOps Institute Ambassador,
- 20+ DevOps Experience.

SKILUP DAYS

Wrap Your Head Around Microservices

The time is here.

We need to begin understanding cloud native architecture, Kubernetes and how our CD Pipelines will change.

SKILup DAYS

So Much to learn

The task at hand is to understand how cloud native changes pipeline automation, and the pitfalls that are in front of us.

Don’t try to learn everything about this platform – you will learn it all overtime.

SKILup DAYS

The Foundation

Microservices are loosely coupled and independently deployed functions that flow through the Continuous Delivery pipeline alone. They are the foundation of business agility.

Microservices create an endless cycle of changes moving out to K8s clusters continuously.

Microservices Vs. Monolithic

We are taking our static application and breaking it into smaller puzzle pieces.

Three Basic Truths You Must Understand

1) Shared Platform- To gain the full benefit of microservices, they should be shared across teams.

2) Independently Deployed - Microservices are independently deployed and can impact multiple ‘logical’ applications.

3) The build step -Microservices do not require a traditional ‘build’ step. Linking is not done at the CI build, it is done at runtime via APIs.

SKILUP DAYS

Microservices Architecture is About Sharing

Truth #1

Most microservices should be reused.

Microservice sprawl is a sign that teams are not sharing services. Sprawl is expensive and confusing. DevOps professionals need to help make sharing easy.

SKILUP DAYS

Truth #1 Challenge: Finding and Sharing is Hard

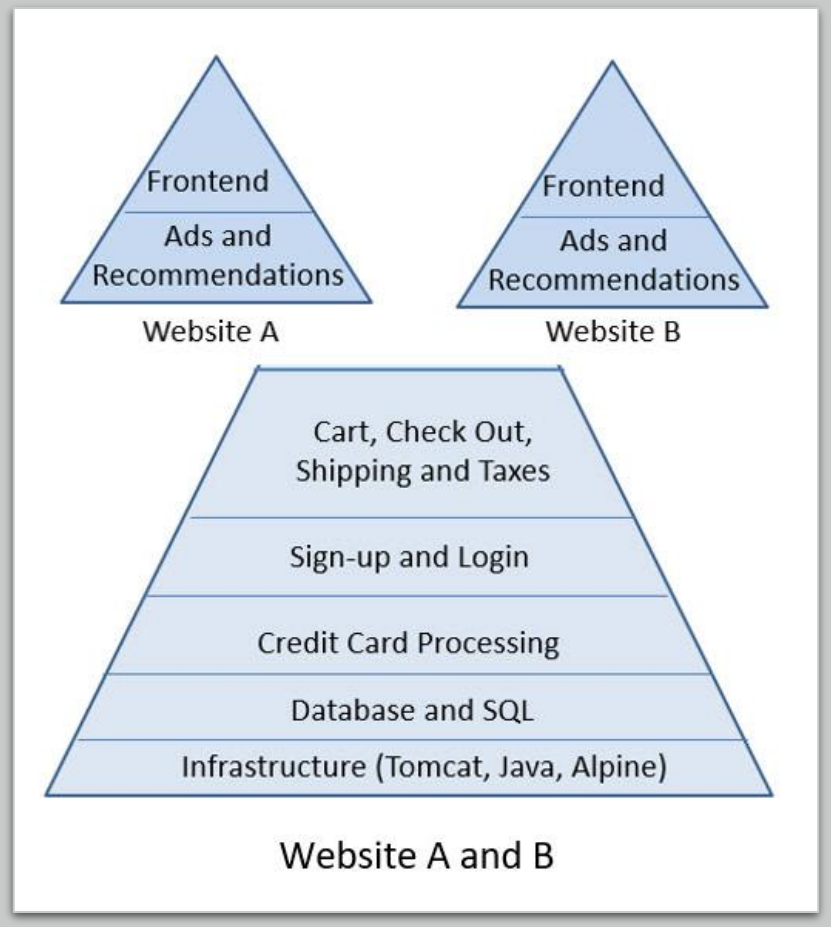
Microservice reuse must be facilitated as a function of DevOps.

Domain Driven Design

Organizing Your Microservices:

- Domain Driven Design is where you are managing an architecture based on the microservice ‘problem space.’

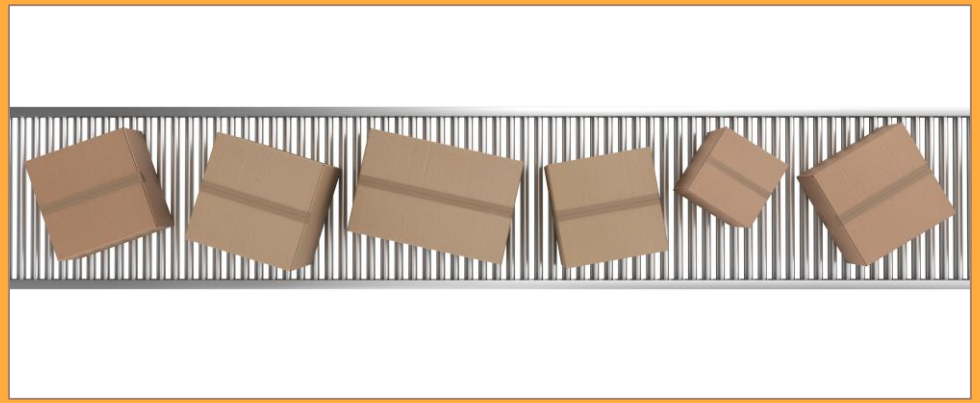
- Domains can be defined based on your organizational patterns. Start by decomposing a few applications and you will begin recognizing their commonality. What is common are potential domains.
  - Login routines
  - Database calls
  - Logging



Microservices are Independently Deployed

Truth #2

Moving from one big static package. . .

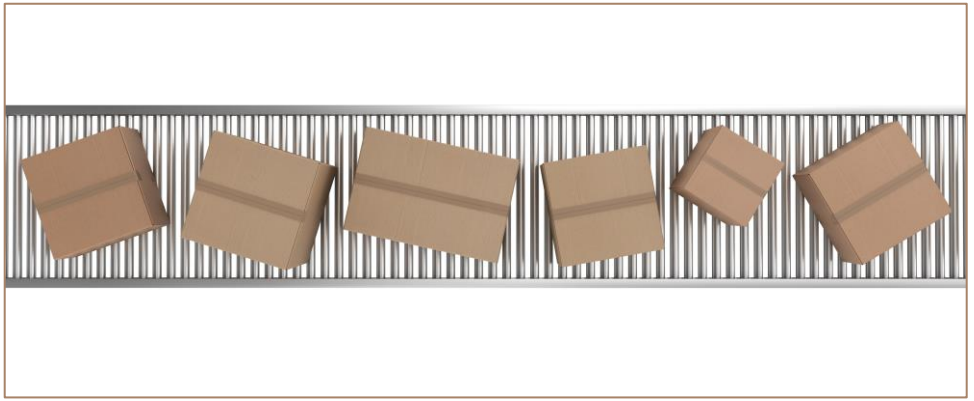


To many little dynamic packages.

SKILUP DAYS

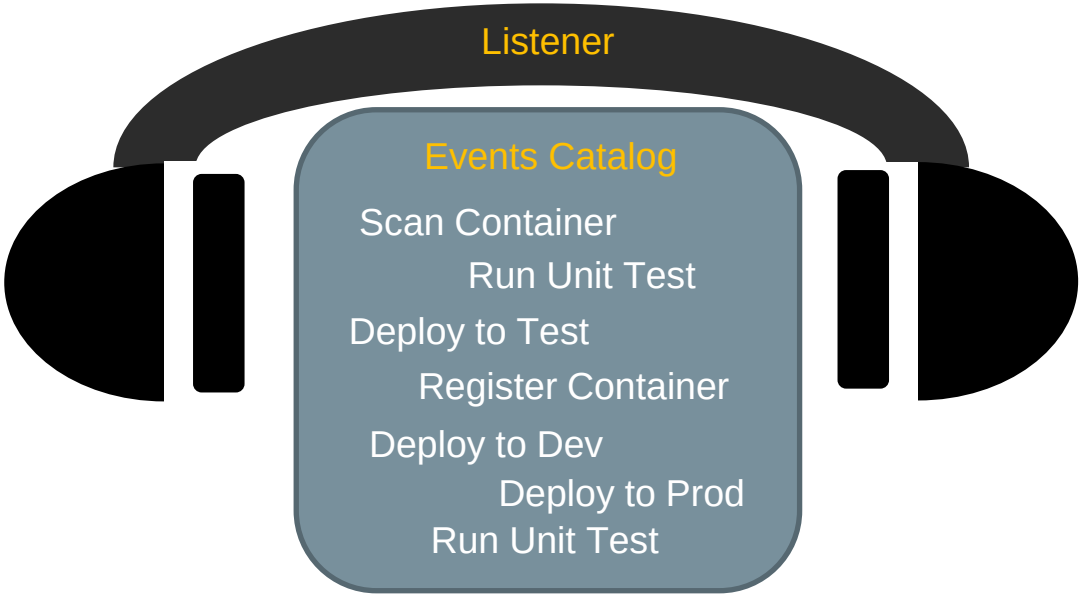
Truth #2 Challenge - DevOps at Scale

This requires **Unscripted** and **Automated** DevOps.



The number of pipeline workflows will increase and become difficult to manage.

Event Driven and Templated Pipelines



SKILUP DAYS

The Application Goes Away - Truth #3

Today's Application Pipeline

The CI Step:
- Pull request / Approval / Trigger
- Compile/Link the Application, create a new version, generate BOM, Diff and Impact Reports
- Deploy to Dev

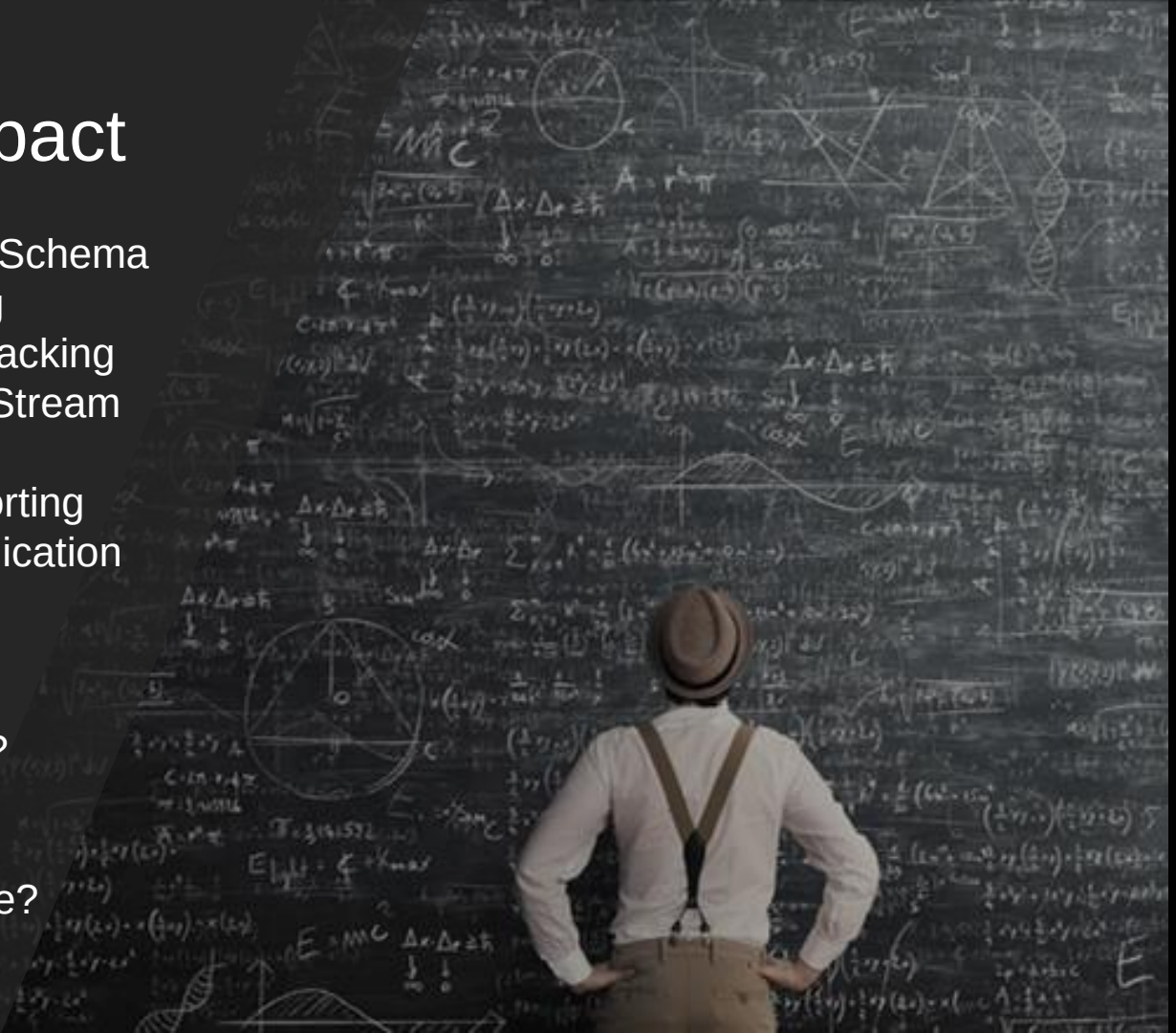
The Test Step:
- Deploy to Test
- Run Test
- Approve for Prod

The Prod Step:
- Approve
- Deploy to Prod

SKILUP DAYS

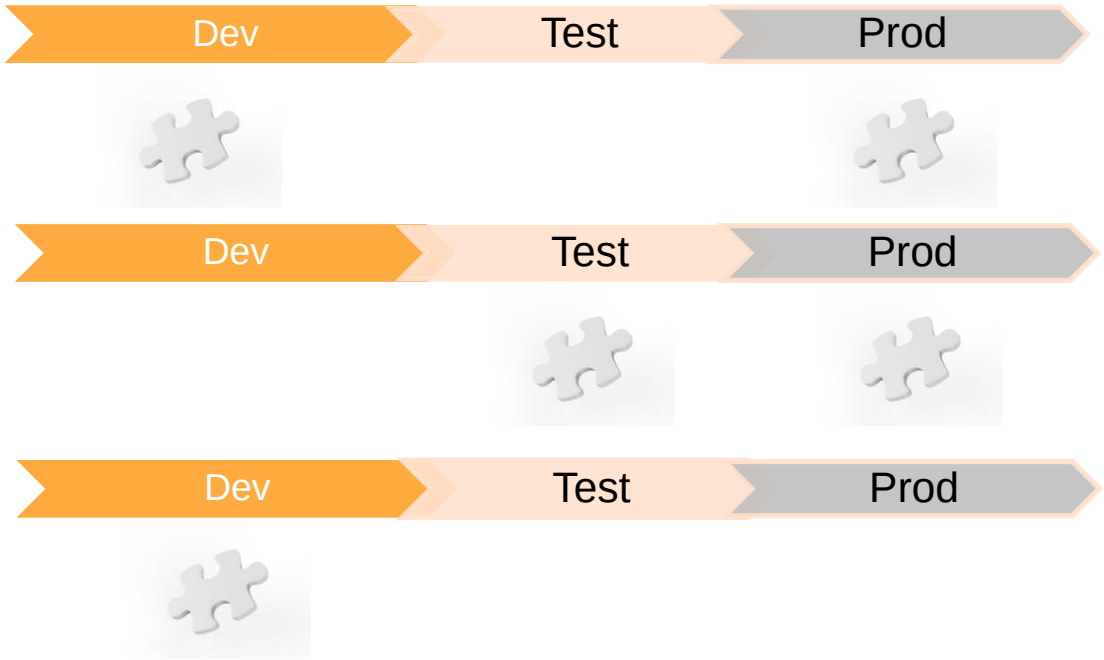
Truth #3 Impact

- Application Version Schema
  - Impacts Testing
  - Impacts Bug Tracking
  - Impacts Value Stream
- Bill of Material Reporting
  - How is the Application Configured?
- Difference Reports
  - What was new?
- Impact Analysis
  - Should I release?



Truth #3 Challenge - Configuration Management

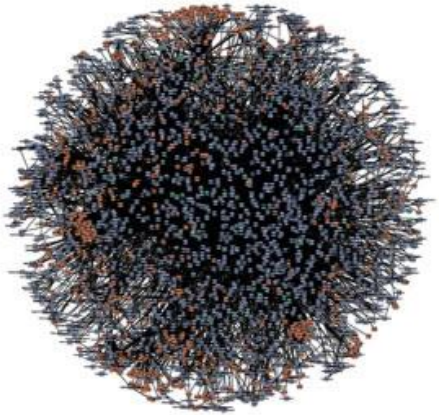
- Dev does not create an “application” much less tracks versions.
- You may not know when a new version of a service was released – you now have a new version of your application.



Visualizing the Logical View of the Application

Microservice Configuration Management

Navigating the Death Star



Critical Data for both Dev and Ops:

- Tracking what microservices your application consumes (Version and BOM).
- Knowing when a particular microservice is about to be updated or has been (Difference Reports).
- What cluster is the new service active in (Deployment Tracking).
- If I update a microservice who will I impact (Impact Analysis).

SKILUP DAYS

A Modern Pipeline

Provides BOM, Difference and Impact Analysis Reports.

Establishes Logical Application versions.

Track SHA, Git Commit and CR. Creates a new microservice version for sharing based on Domains.

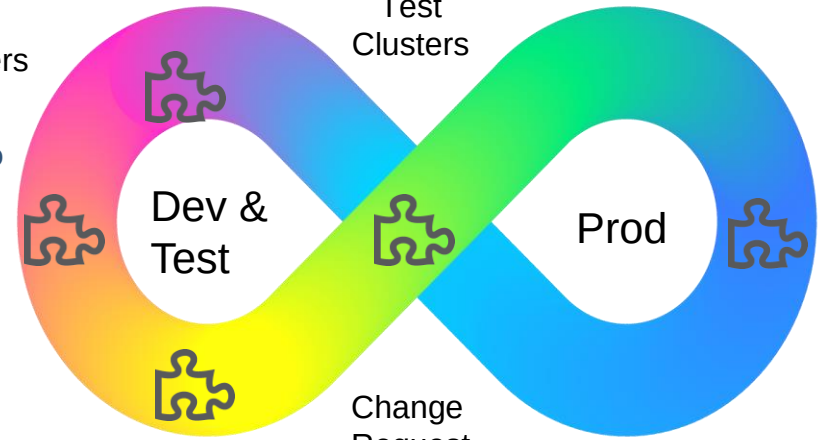
Track Service to Application Configuration

Create Container Image

Scan Container

Register Container

Run CI Build



Deploy to Dev Clusters

Run Contract Based Unit Test

Deploy to Test Clusters

Run Multiple Application Tests

Deploy to Prod Clusters

Security Scan

APM Observability

Change Request

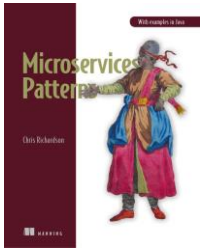
Code Complete

SKILup DAYS

Learn More

Training and Certifications

- devopsinstitute.com/certifications/
- training.linuxfoundation.org/
- cd.foundation/training/
- Microservices.io



SKILUP DAYS

Learn About New Open Source Solutions for CD

- Tekton - https://cloud.google.com/tekton
- Spinnaker - https://spinnaker.io/
- Jenkins Templating Engine - https://www.jenkins.io/blog/2019/05/09/templating-engine/
- Ortelius – Ortelius.io
- Helm – Helm.sh

Templating Engine

SKILUP DAYS

THANK YOU!

Meet Me in the Network Chat Lounge for Questions

LinkedIn: https://www.linkedin.com/in/tracy-ragan-oms/
Twitter: @TracyRagan
Calendar: https://drift.me/tracyragan/meeting/coffeechat
Email: Tracy@DeployHub.com
Dig In at: DeployHub.com or Ortelius.io

SKILUP DAYS