

# BuzzOps Driven Development as a Service

July 21, 2022

# About Me

- Artem Vegera
- Lead Software Engineer, Exadel
- 11+ years in Java-development

# Buzzwords Everywhere



# What is Buzzword?

Is a word or phrase, new or already existing, that becomes popular for a period of time:

- often derives from technical terms yet
- often has much of the original technical meaning removed through fashionable use
- being simply used to impress others

# Buzzword Bingo

BUZZWORD				
B	I	N	G	O
Machine learning	Next-gen	Innovation	Mobile	Revenue Sharing
Apps	Resilient	SaaS	Blockchain	Efficiency
Data-mining	IoT		Gamification	Sensors
Pilot	Savings	Dashboard	Scalable	Disruption
SoLoMo	Engagement	Cloud	Big data	Smart

# Buzzword Explanation



Arvind Narayanan ✅

@random\_walker

Following



Tech buzzwords explained:  
AI—regression  
Big data—data  
Blockchain—database  
Algorithm—automated decision-making  
Cloud—Internet  
Crypto—cryptocurrency  
Dark web—Onion service  
Data science—statistics done by nonstatisticians  
Disruption—competition  
Viral—popular  
IoT—malware-ready device

8:02 AM - 22 Mar 2018

15,350 Retweets 31,331 Likes



# Senior Buzzword Engineer

## Software Development DevOps and Cloud 2022 Graph

<http://infoq.link/devops-and-cloud-trends-2022>

InfoQ



# What About This Talk?

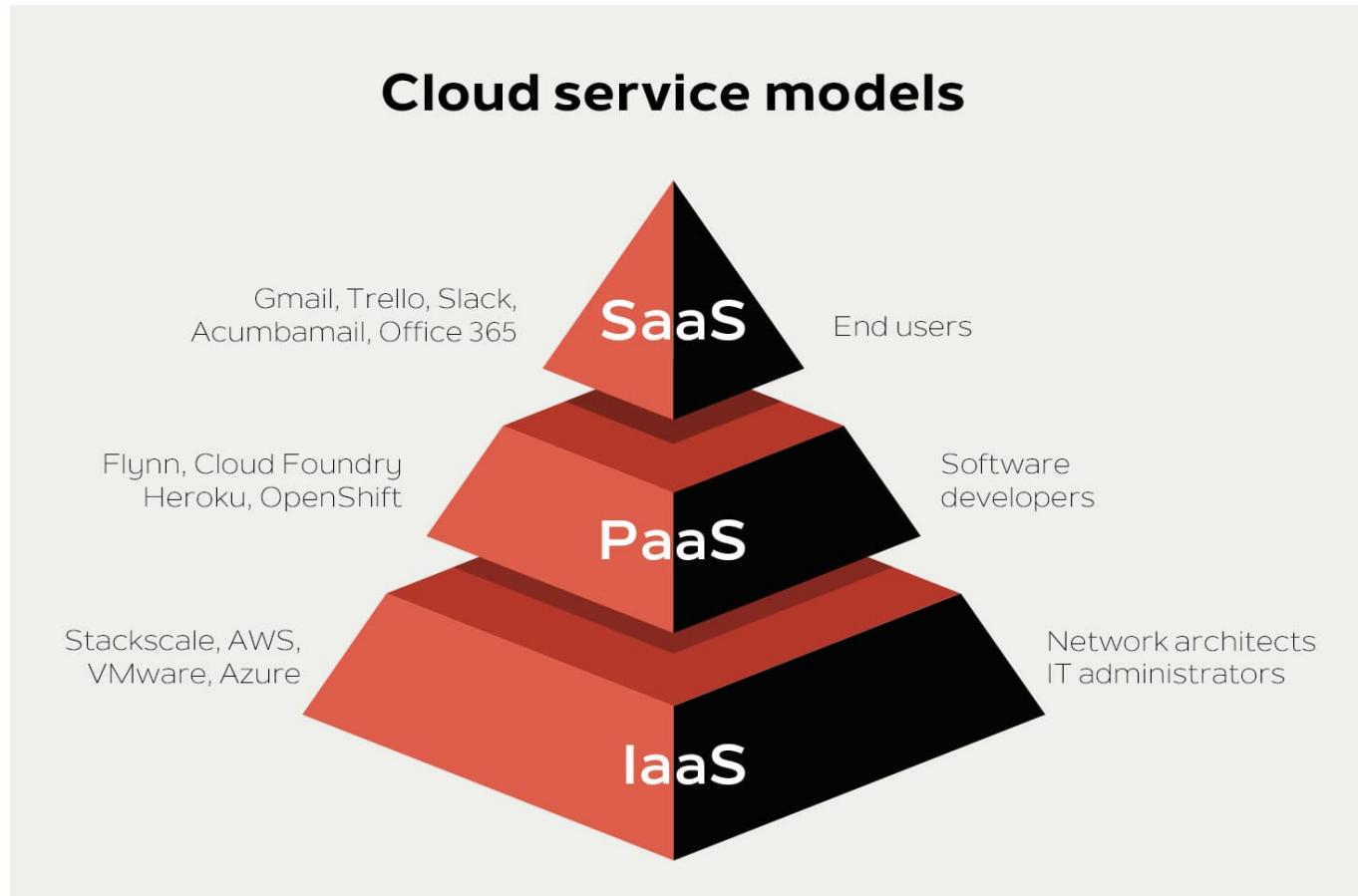
- Overview of **developer-related** buzzwords
- Organize them into one **big-picture** landscape

Disclaimer:

- It's **impossible** to be an expert in everything
- This is a **random** overview of some trends just for fun
- Calling them buzzwords **doesn't diminish** their value

# I. As a Service

# SaaS vs PaaS vs IaaS



# Everything as a Service

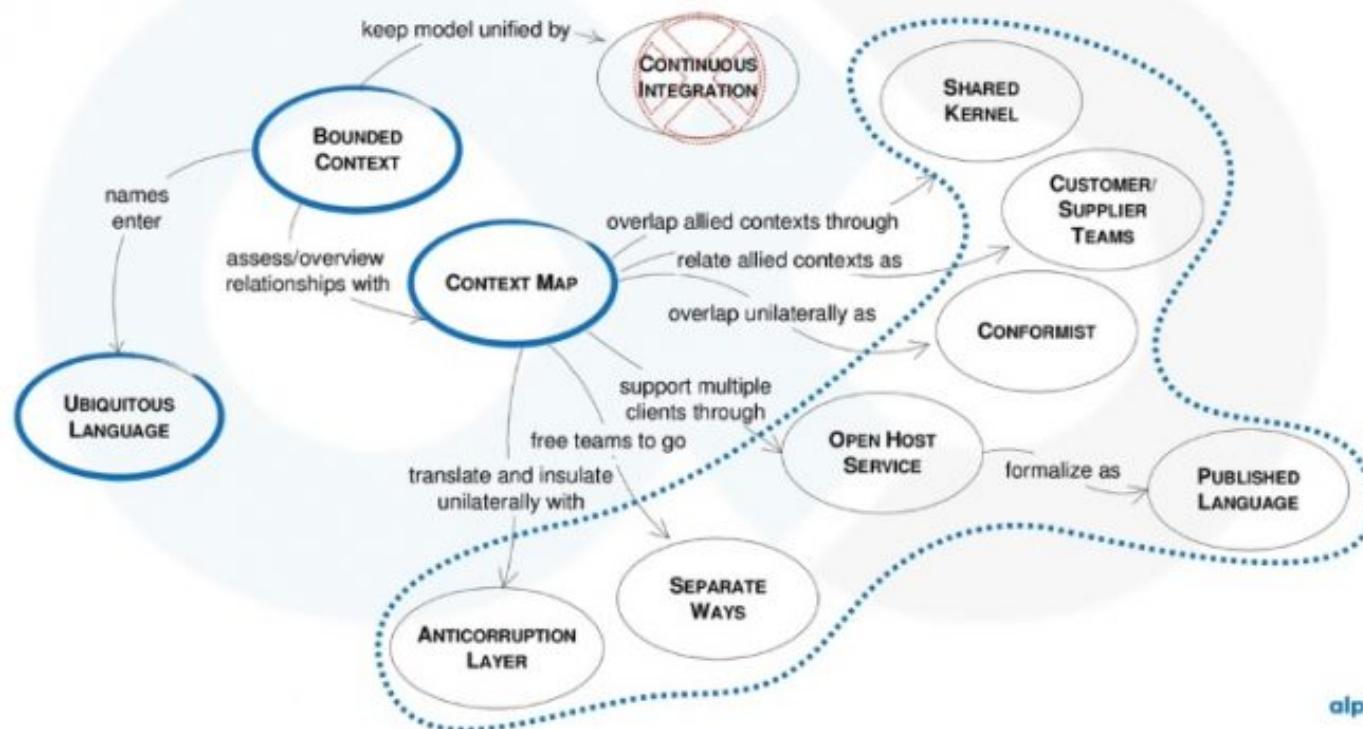
Service	Abbr.
Artificial intelligence as a service	AlaaS <sup>[3]</sup>
Banking as a service	BaaS
Blockchain as a service	
Content as a service	CaaS
Data as a service	DaaS
Desktop as a service	
Database as a service	DBaaS
Electric vehicle as a service <sup>[4][5]</sup>	EVaaS
Function as a service	FaaS
Games as a service	GaaS
Identity as a service	IDaaS <sup>[6]</sup>
Infrastructure as a service	IaaS
Integration platform as a service	iPaaS
IT as a service	ITaaS
Knowledge as a service	KaaS

Service	Abbr.
Logging as a service	LaaS
Lighting as a service	
Mobility as a service	MaaS
Monitoring as a service	
Mobile backend as a service	MBaaS
Network as a service	NaaS
Payments as a service	PaaS
Platform as a service	
Quality assurance as a service	QAaaS <sup>[7]</sup>
Recovery as a service	RaaS
Robot as a service	
Security as a service	SaaS
Software as a service	
Unified communications as a service	UCaaS

## II. Driven Design

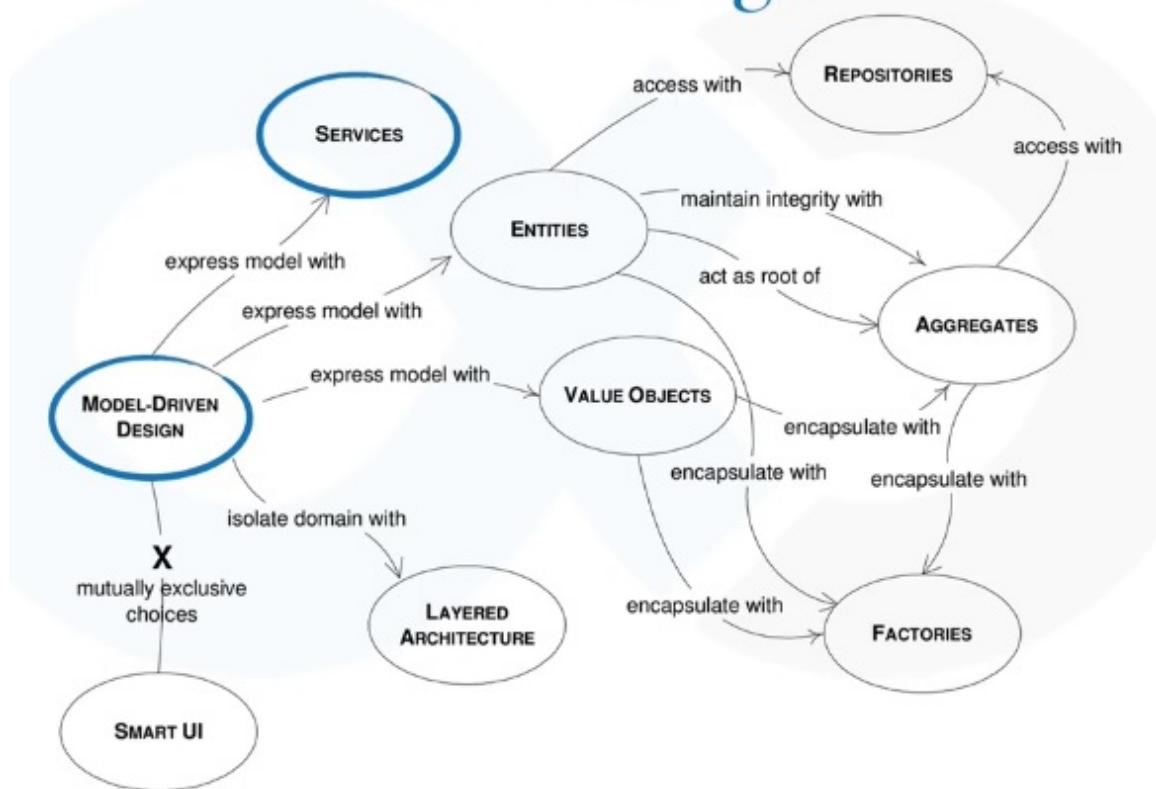
# Domain Driven Design

## Strategic Design Tools

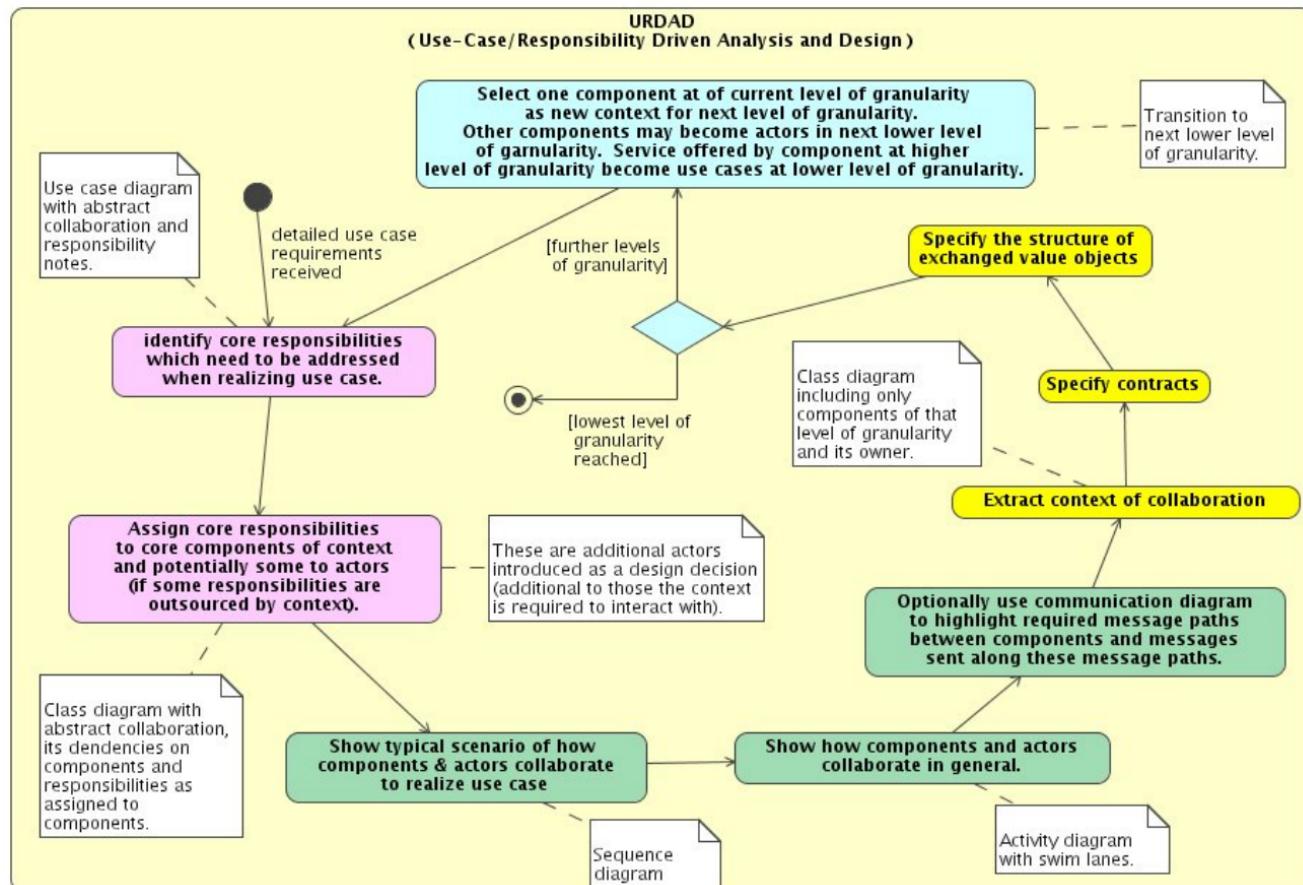


# Domain Driven Design

## Tactical Design



# Responsibility Driven Design



# Resumé Driven Design

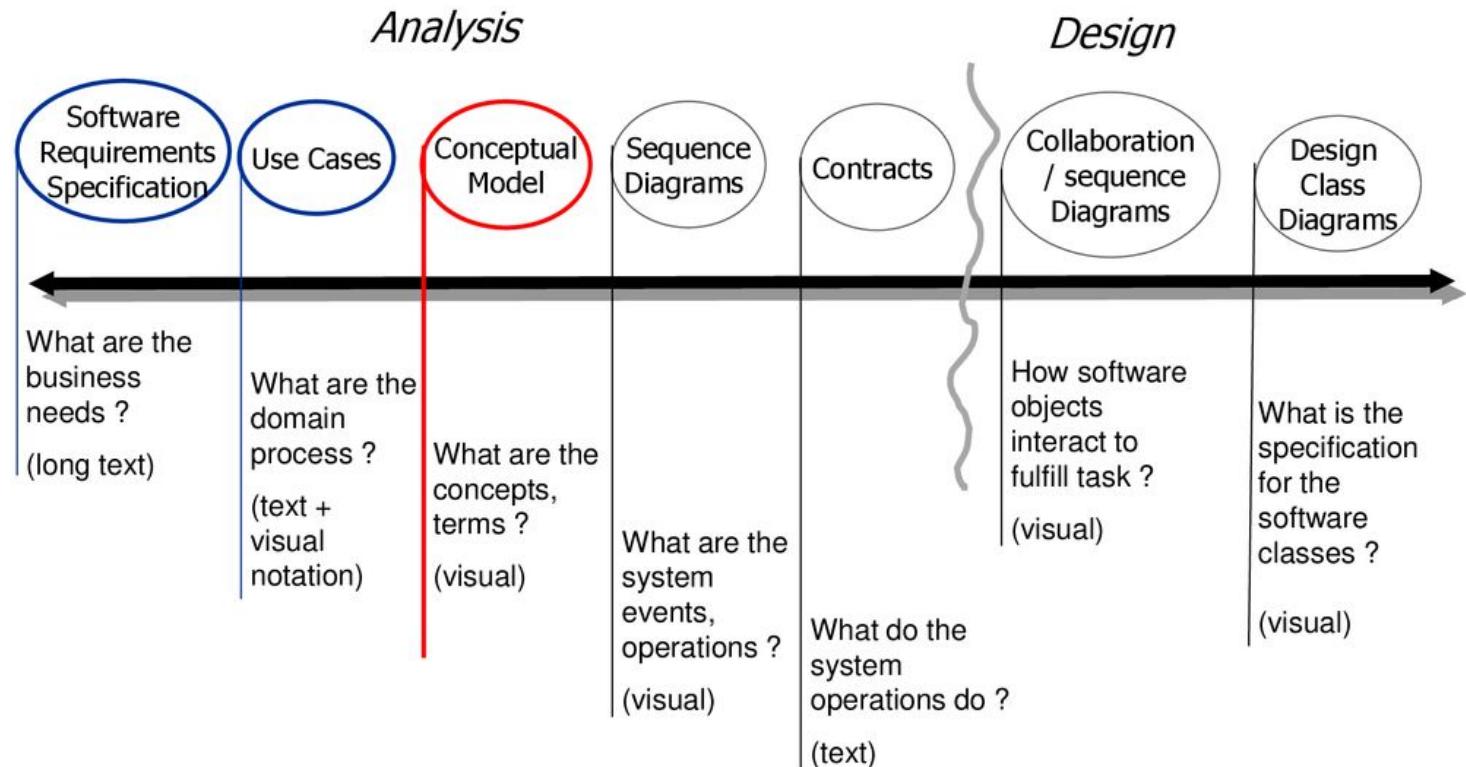


### III. Oriented Design

# Driven vs Oriented

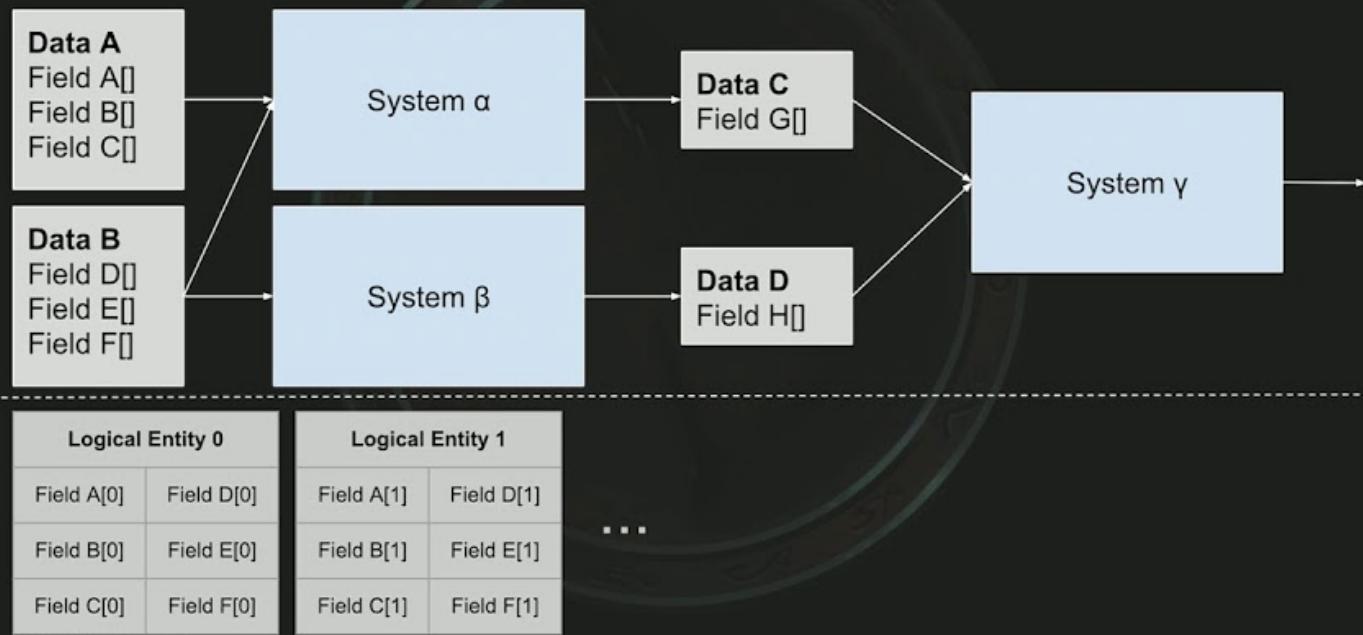
- **Driven** motivates you
- **Oriented** gives you a direction

# Object Oriented Design



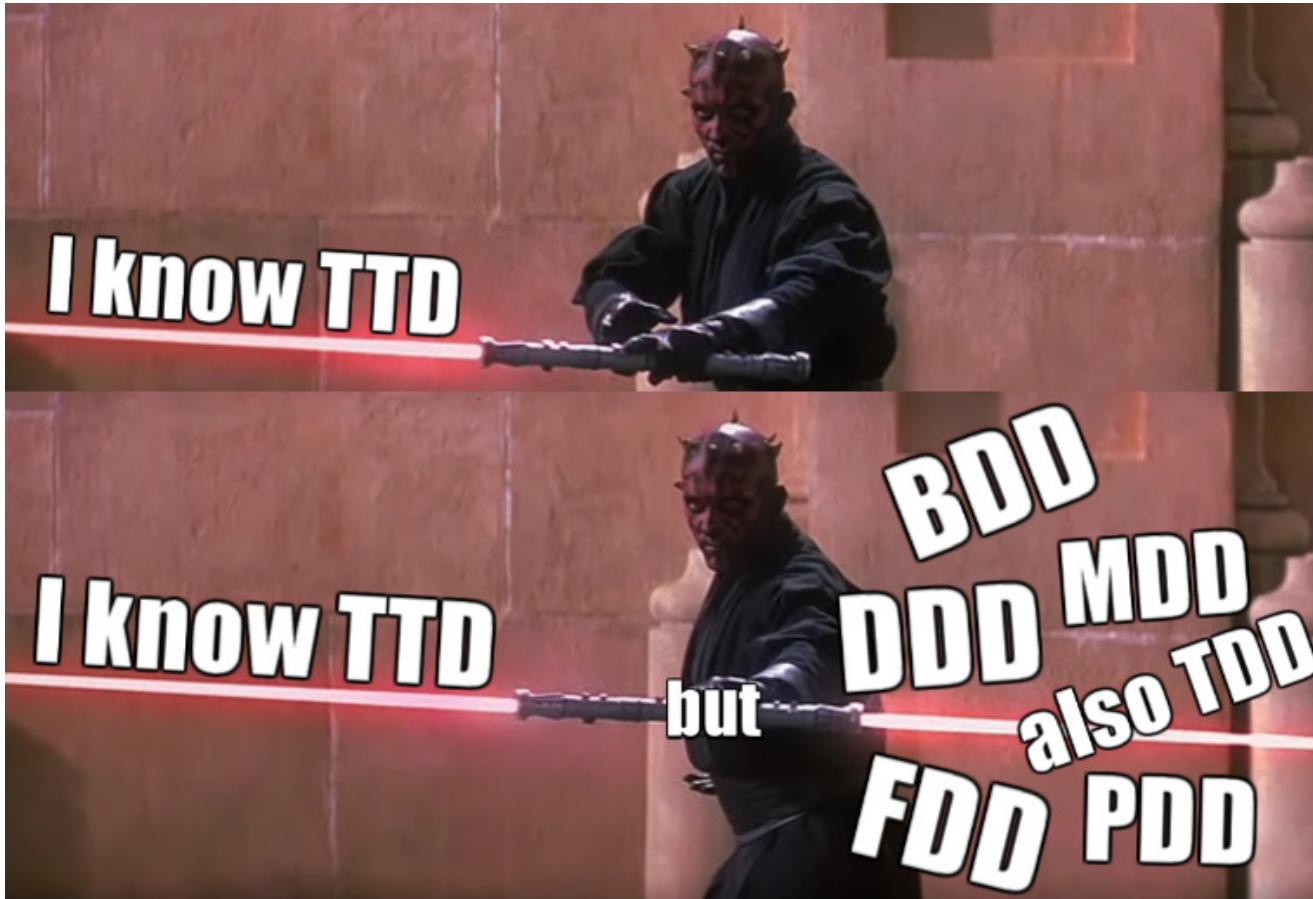
# Data Oriented Design

## Data-oriented design



# IV. Driven Development

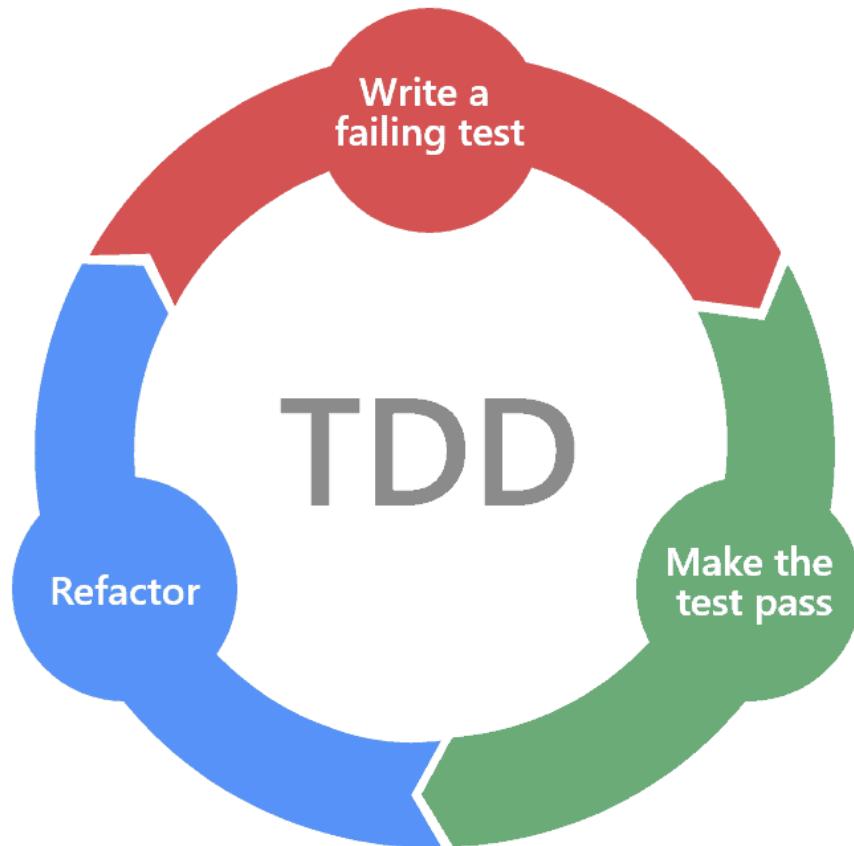
# X Driven Development



# Beer Driven Development



# Test Driven Development



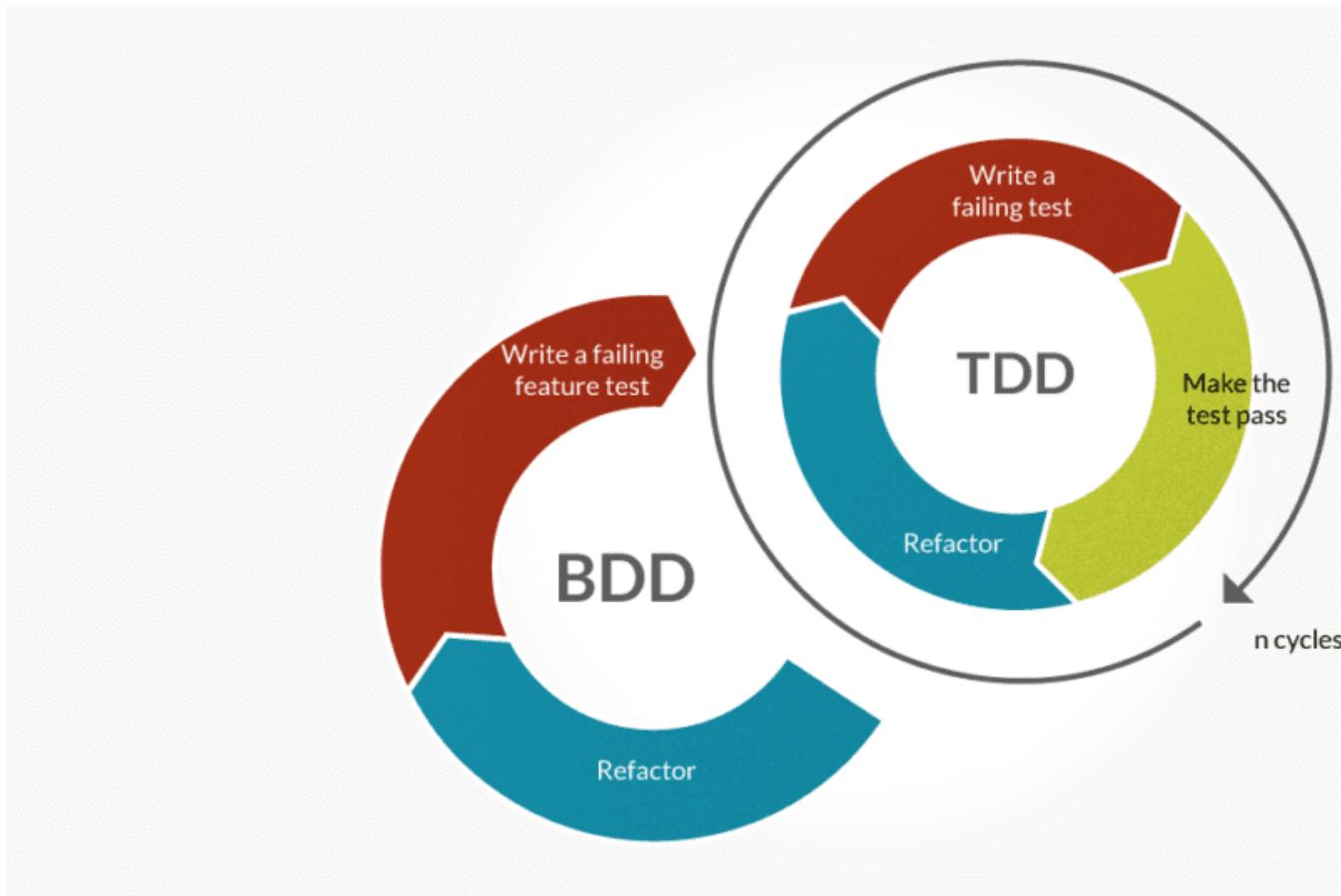
# Behavior Driven Development

## BDD Development Process

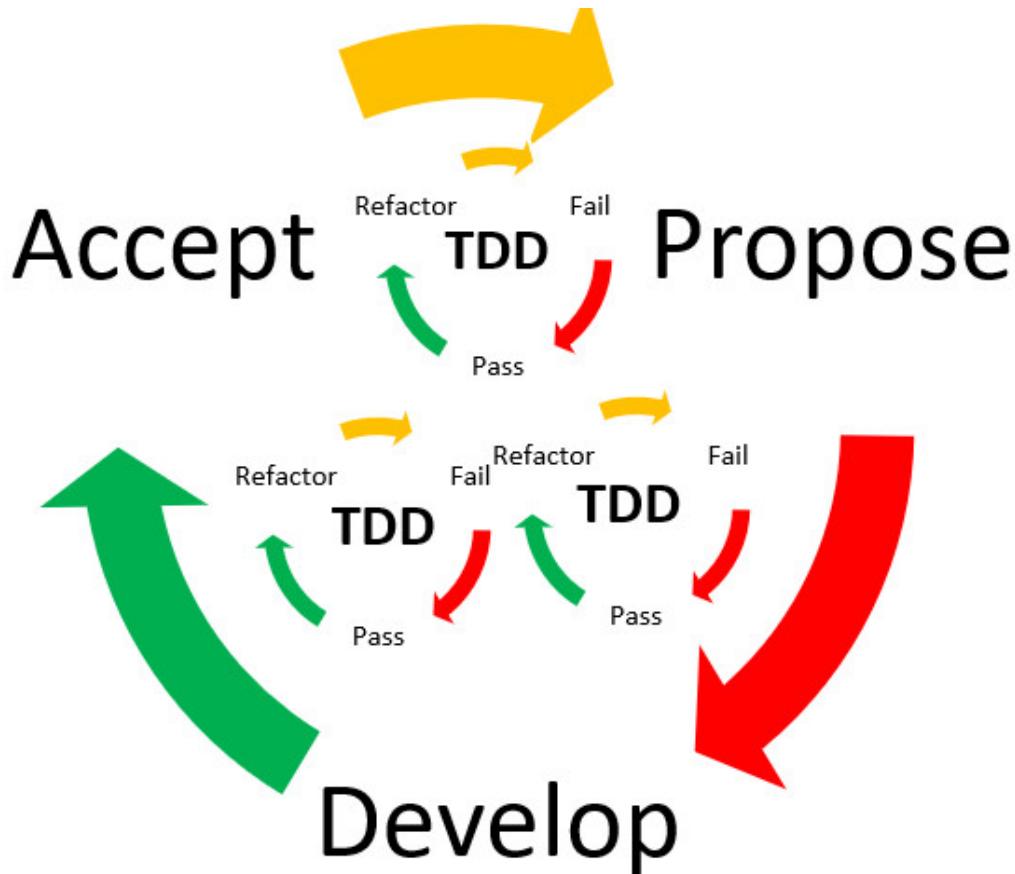


The tester uses these scenarios as the basis for their tests

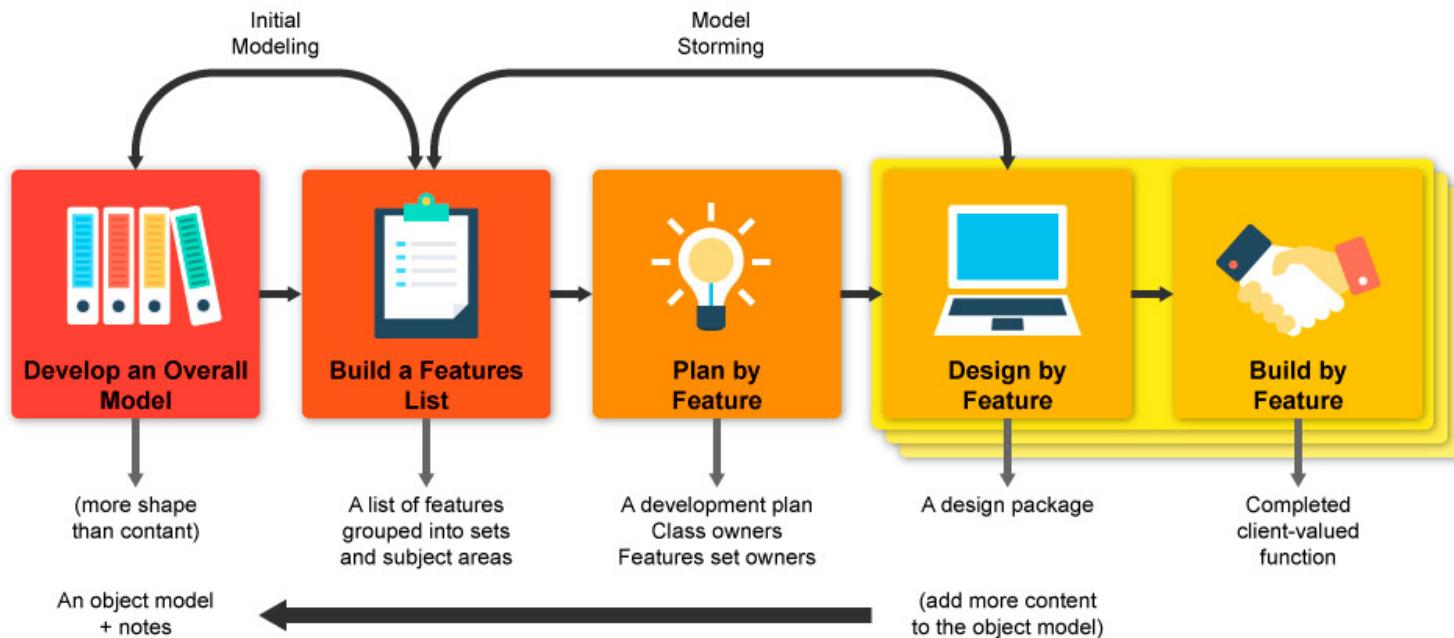
# TDD vs BDD



# Acceptance Test Driven Development



# Feature Driven Development



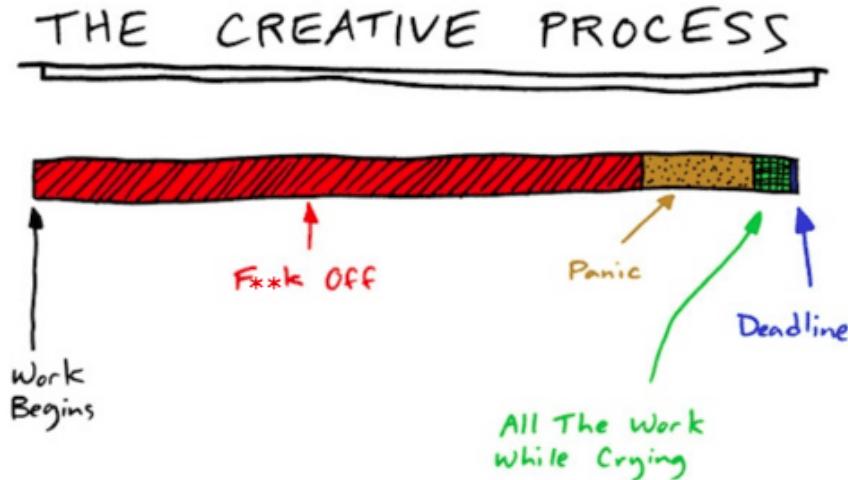
# Requirements Driven Development

	Requirements Driven Development	Feature Driven Development
<b>General Characteristics</b>	High-level, non-prescriptive	Varying degrees of detail
<b>Focus Area</b>	Product attributes	Product functionality
<b>Perspective</b>	Product or solution	End-user point of view
<b>Benefits / Advantages</b>	Derived from user perspective; May be developed without customer input; Potentially less cost/effort to develop	More likely to meet customer desired experience; Provides more opportunity for innovation
<b>Risks / Disadvantages</b>	Potential to build a product that does not support user objectives; Less freedom to innovate and/or experiment with new solutions	May require more time to develop due to customer collaboration;

# Job Safety Driven Development



# Panic Driven Development



# Resumé Driven Development

*The passionate, functional, micro-serviced approach*



*Expert*

Resumé Driven  
Development

O RLY?

@ThePracticalDev

# Hype Driven Development

---

*Looking for love in all the wrong frameworks*



Hype Driven  
Development

*Life on the Bandwagon*

O RLY?

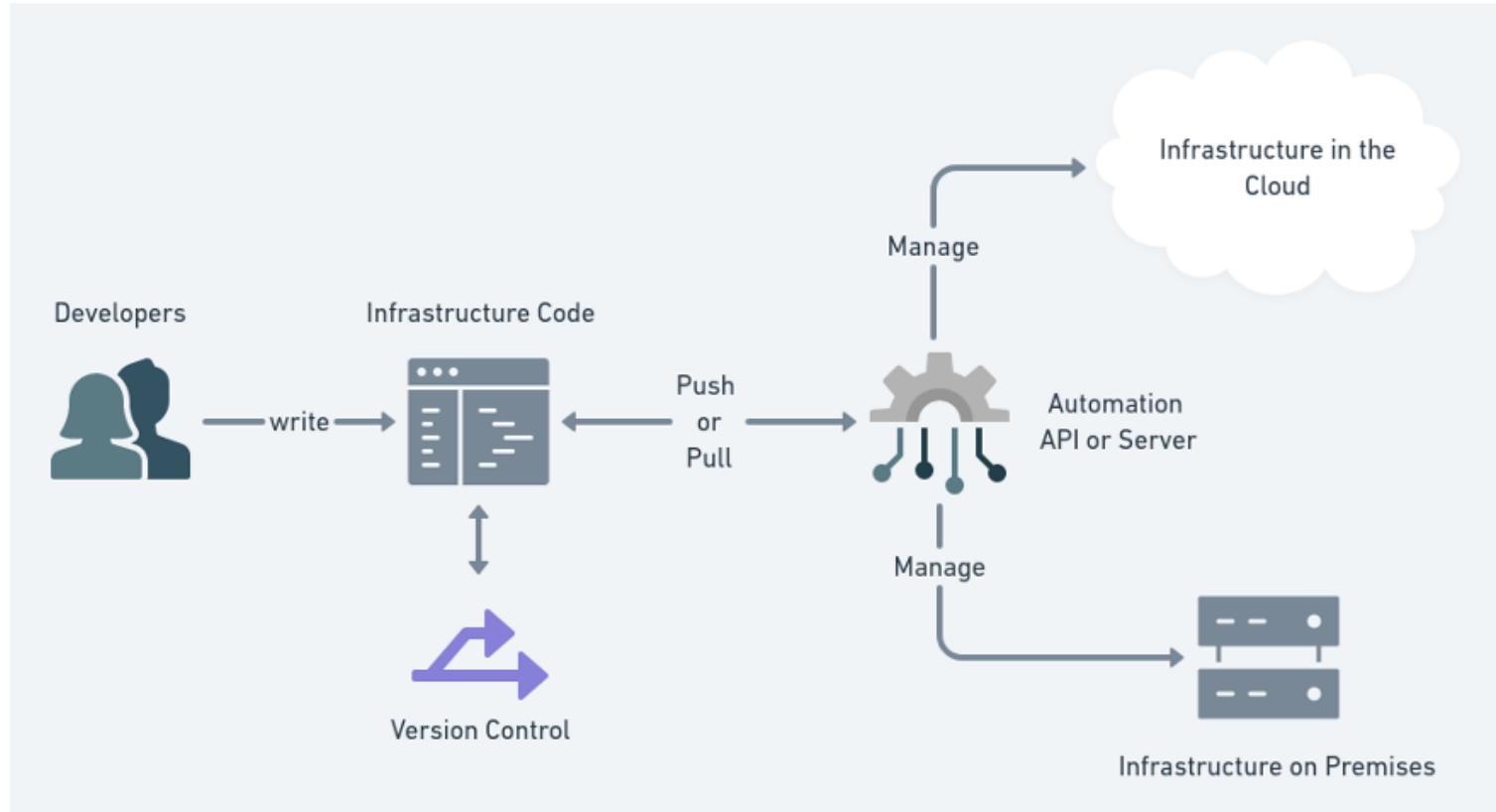
@ThePracticalDev

# And Others

- Business Driven Development
- Data Driven Development
- Model Driven Development
- Type Driven Development
- etc.

# V. as Code

# Infrastructure as Code



# Pipeline as Code

```
1  node {
2      stage("Checkout") {
3          git branch: "${params.BRANCH}", url: 'https://github.com/git-user/repo.git'
4      }
5      stage("Build") {
6          try {
7              withMaven(maven: "Maven363") {
8                  sh "mvn package"
9              }
10         } catch (error) {
11             currentBuild.result='UNSTABLE'
12         }
13     }
14     stage("Publish artifact") {
15         def server = Artifactory.server "$SERVER_ID"
16
17         def uploadSpec = """{
18             "files": [
19                 {
20                     "pattern": "target/repo*.jar",
21                     "target": "libs/com/user/pipelines/"
22                 }
23             ]
24         }"""
25
26         server.upload(uploadSpec)
27     }
28 }
```

# Everything As Code

The beginning of an Everything as Code (EaC) generation

---

## Infrastructure as Code



Simple workflows to auto-provision infrastructure in minutes

## Environments as Code



Single workflows to build and deploy virtual machine environments in minutes

## Configuration as Code



Simple, model-based workflows to scale app deployment and configuration management

## Data Pipelines as Code



Programmatically author, schedule and monitor data pipeline workflows as code.

## Security Configuration as Code



Detect and remediate build and production security misconfigurations at scale

## Encryption Management as Code



Programmatically secure, store and tightly control access across clouds and datacenter

@ethanjb

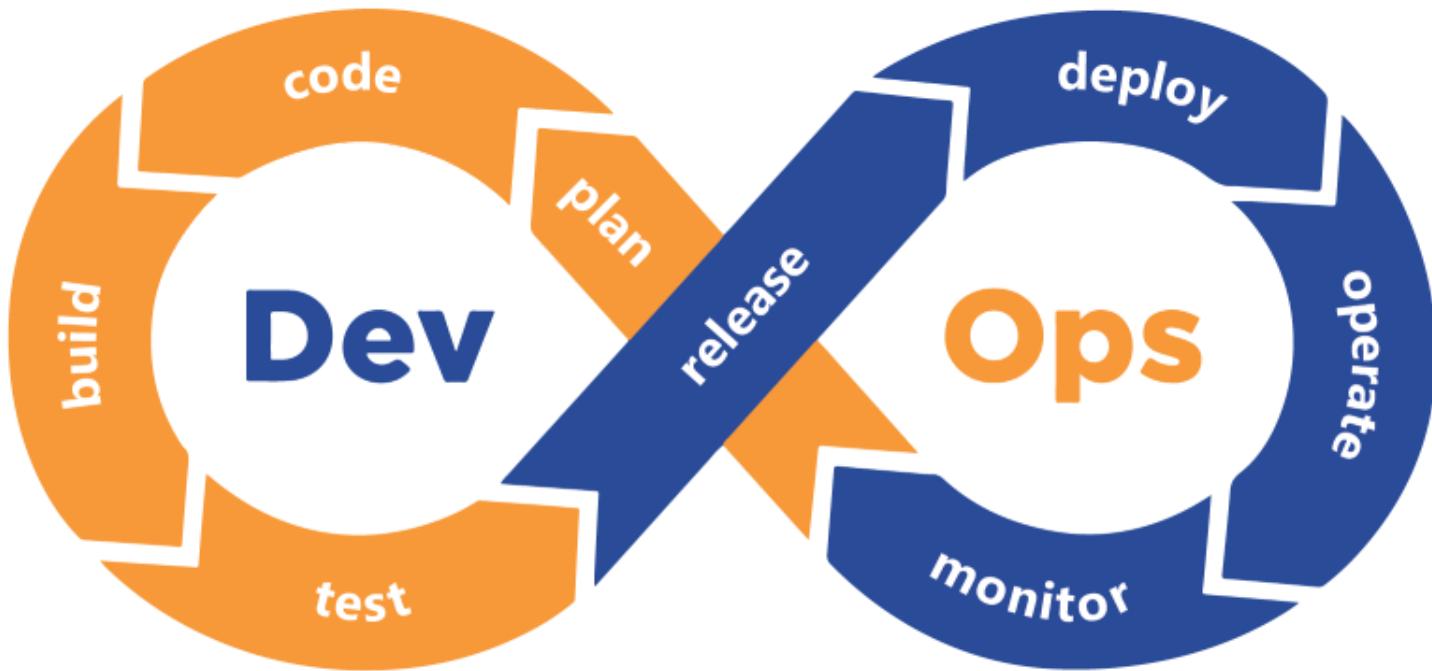


# And Others

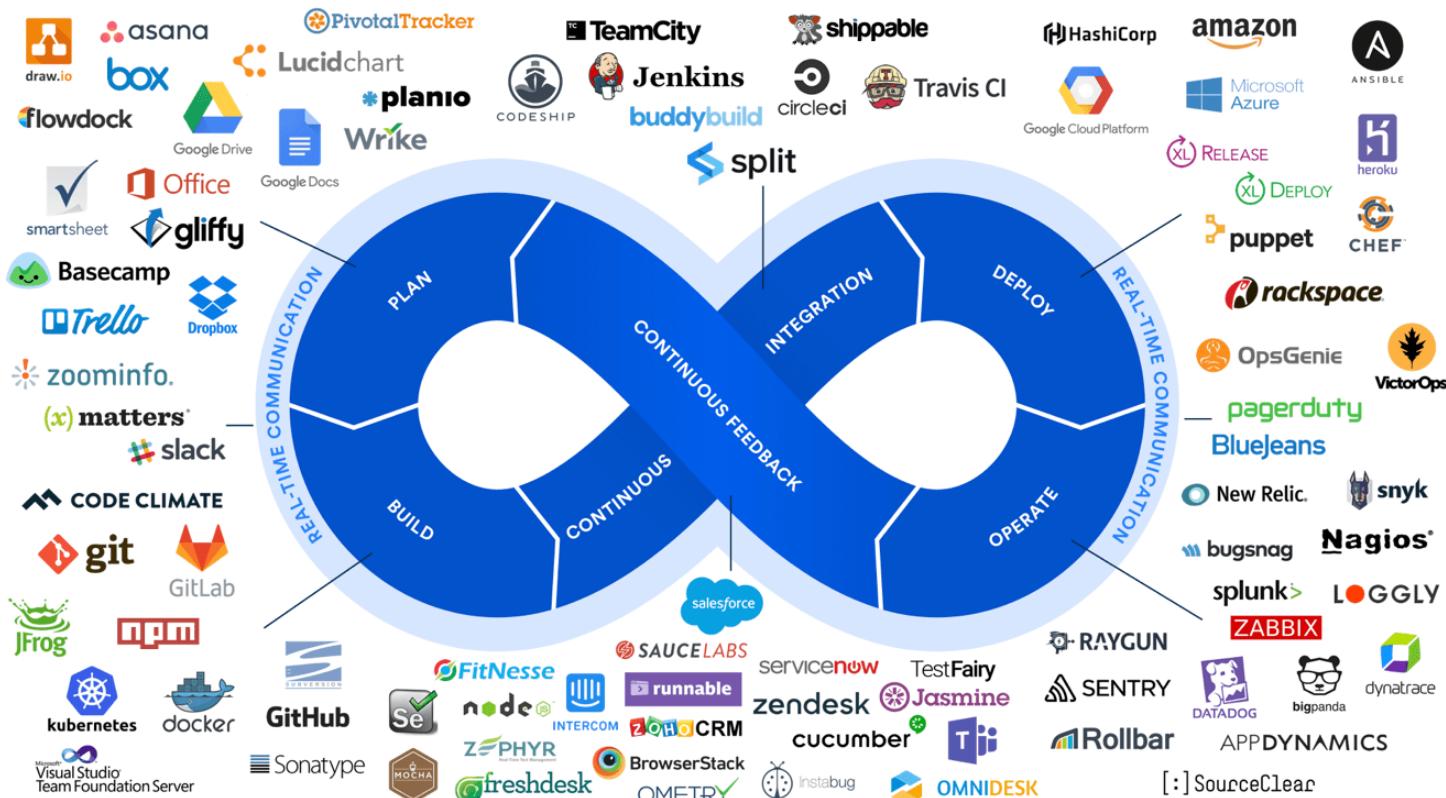
- **Dashboard** as code
- **Policy** as code
- **Docs** as code
- **Diagram** as code
- **Presentation** as code
- **Architecture Decisions** as code
- **Process Decisions** as code
- **Knowledge** as code
- ...

# VI. ...Ops

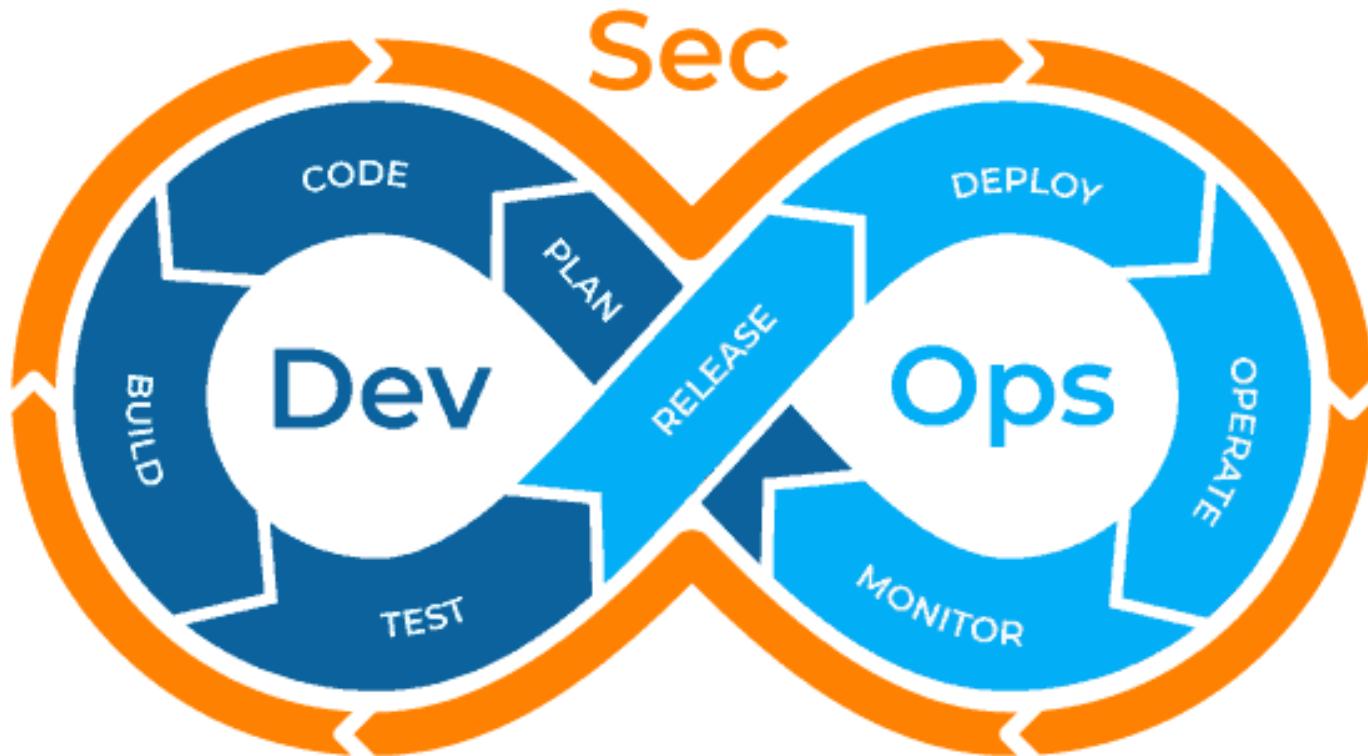
# DevOps



# DevOps



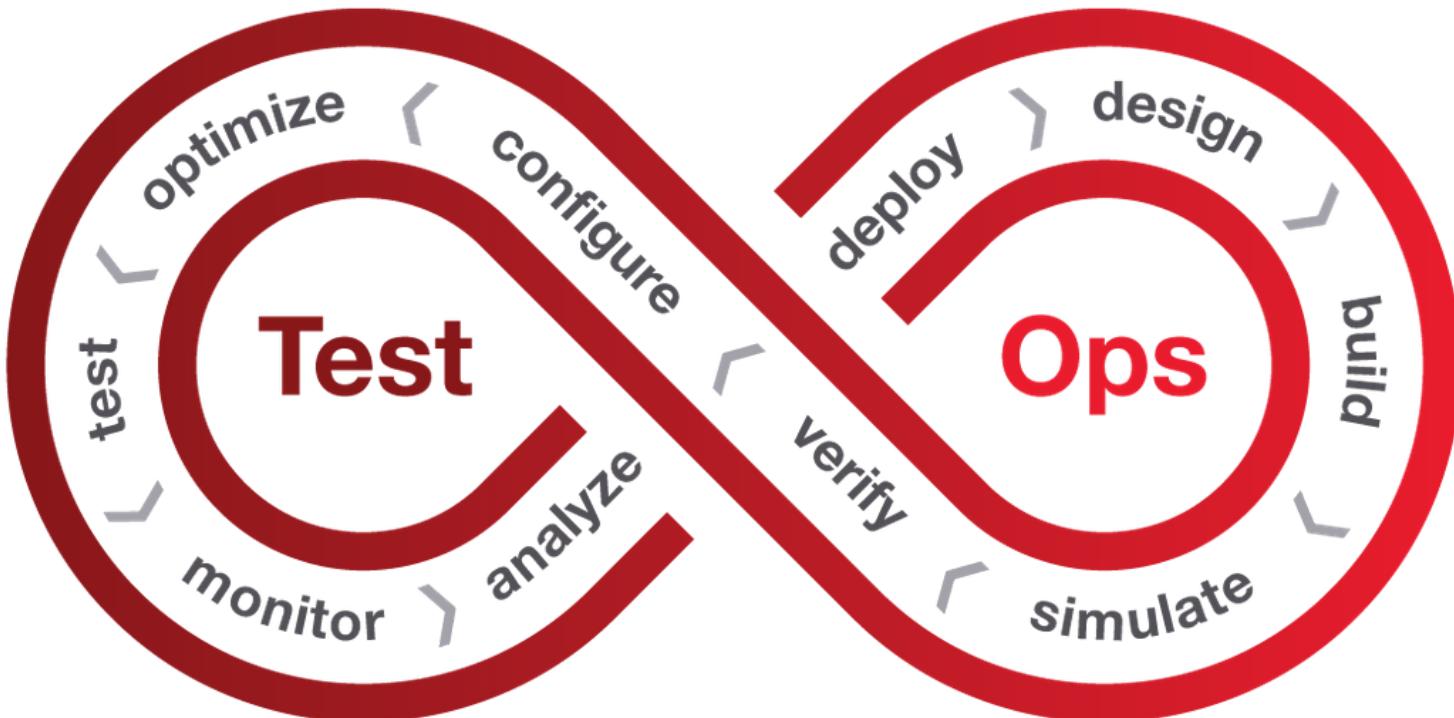
# DevSecOps



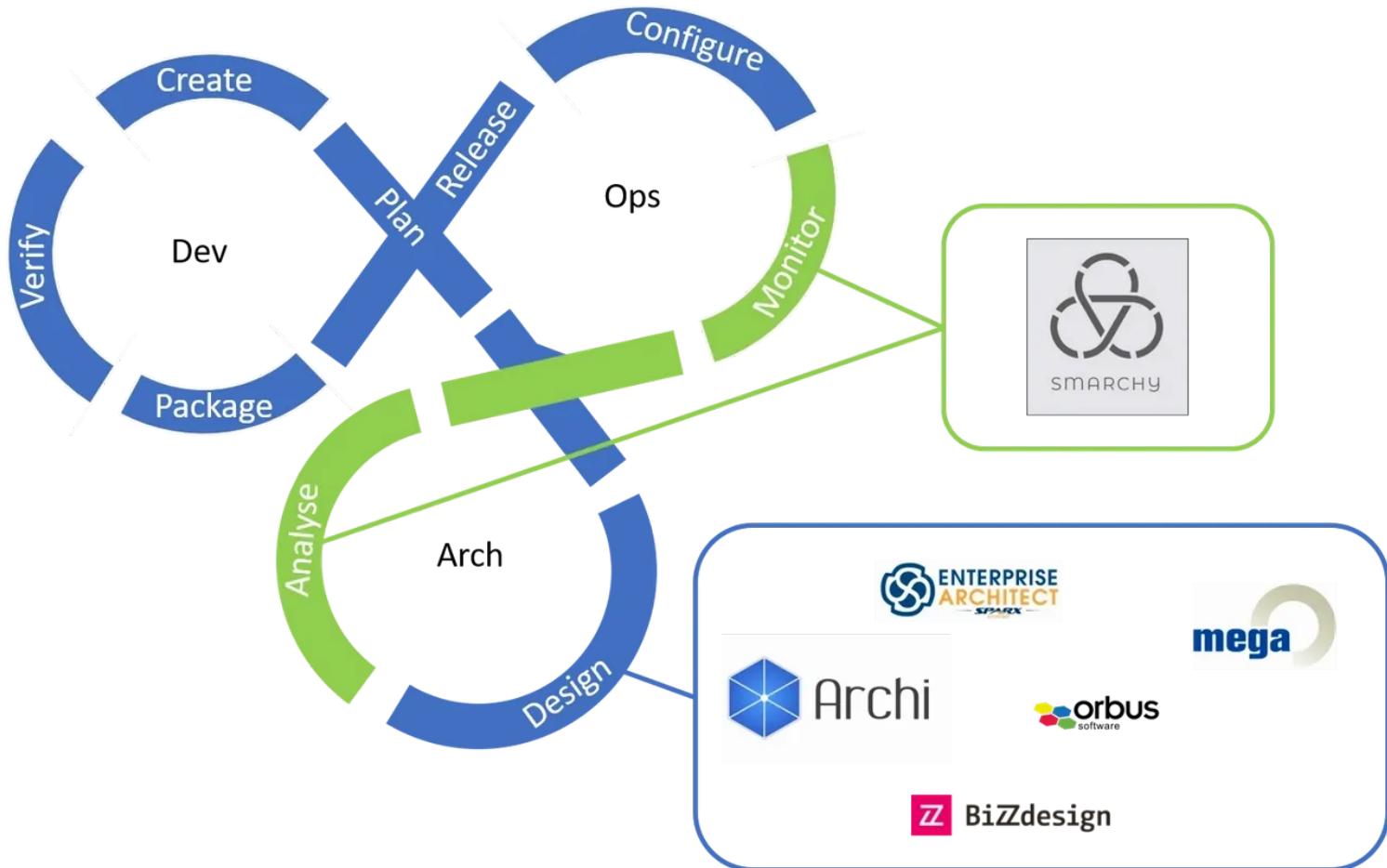
# DevSecOps



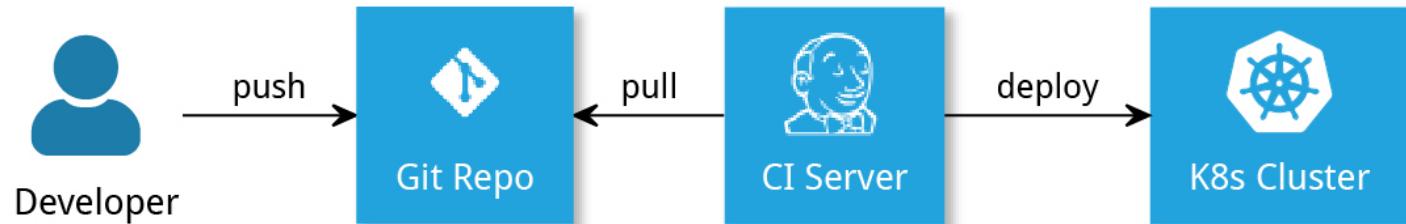
# TestOps



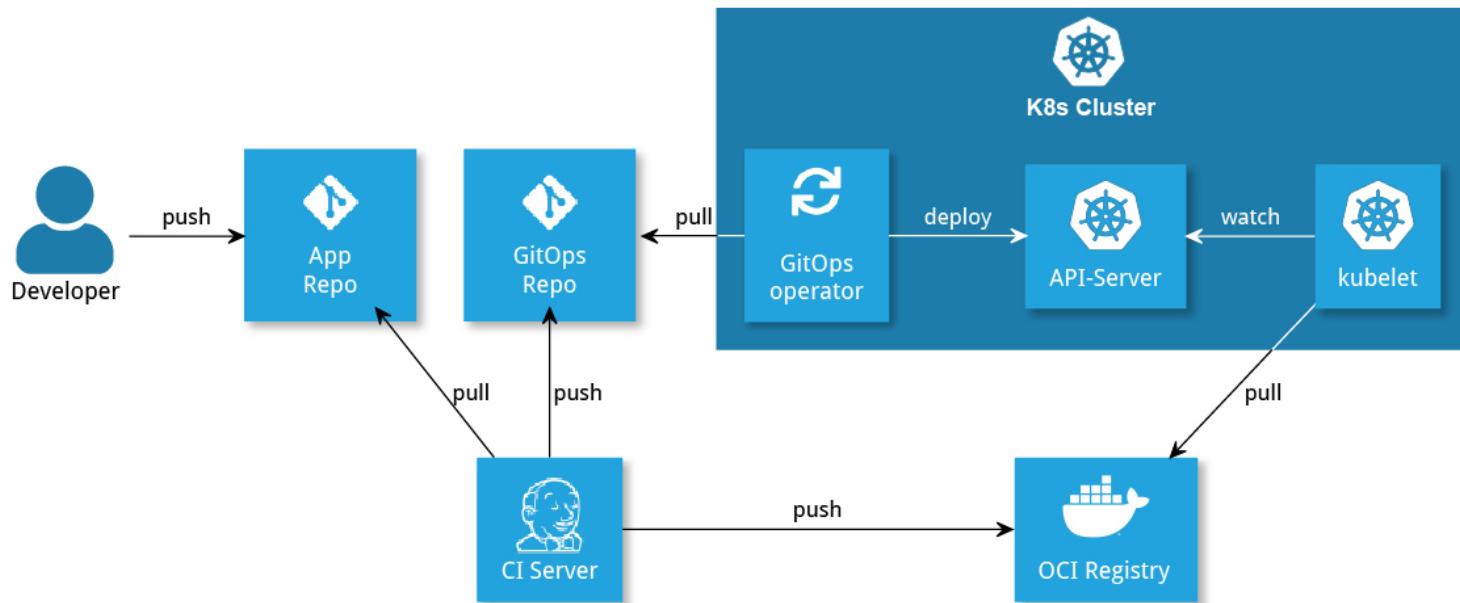
# ArchOps



# CIOps



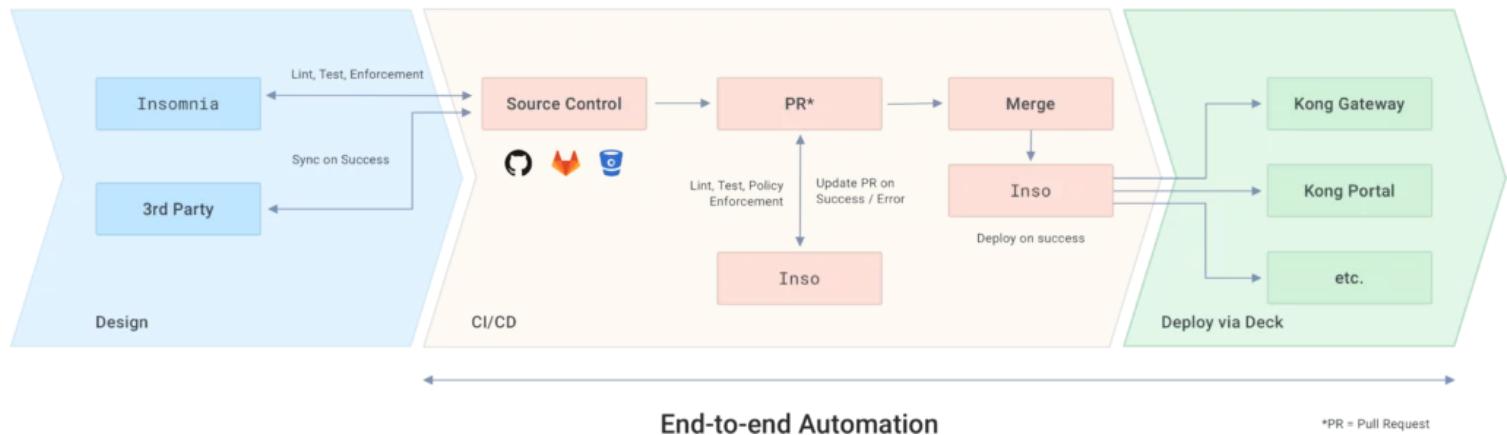
# GitOps



# ApiOps

## APIOps in action

Run unit tests, lint specification, enforce policies, generate config, and deploy configuration / specification



# xOps

- AIOps
- MLOps
- CloudOps
- ChainOps
- ChatOps
- CompOps
- CostOps
- DataOps
- DocOps
- LangOps
- MarketingOps
- PrivacyOps
- PeopleOps
- LifeOps
- NoOps
- SecretOps
- SalesOps
- WinOps
- etc.

# Conclusion

# There Are Also Prefixes

- Continues (integration, delivery, deployment, design, testing, learning)
- Evolutionary (architecture, design)
- Micro (services, frontends)
- Cloud (computing, native, storage)
- Clean (code, architecture)
- Lean
- Agile
- ...

# Summary

Engineering  
Process

