



# How to architect for continuous delivery?

Romano Roth | 30.11.2023 | Conf42 DevSecOps 2023

My passion is helping companies bringing people, processes and technology together so that they can deliver continuously value to their customers



### Romano Roth

Chief of DevOps & Partner

✉ [romano.roth@zuehlke.com](mailto:romano.roth@zuehlke.com)

🏠 <https://www.romanoroth.com/>

🐦 [@RomanoRoth](#)

linkedin [Romanoroth](#)

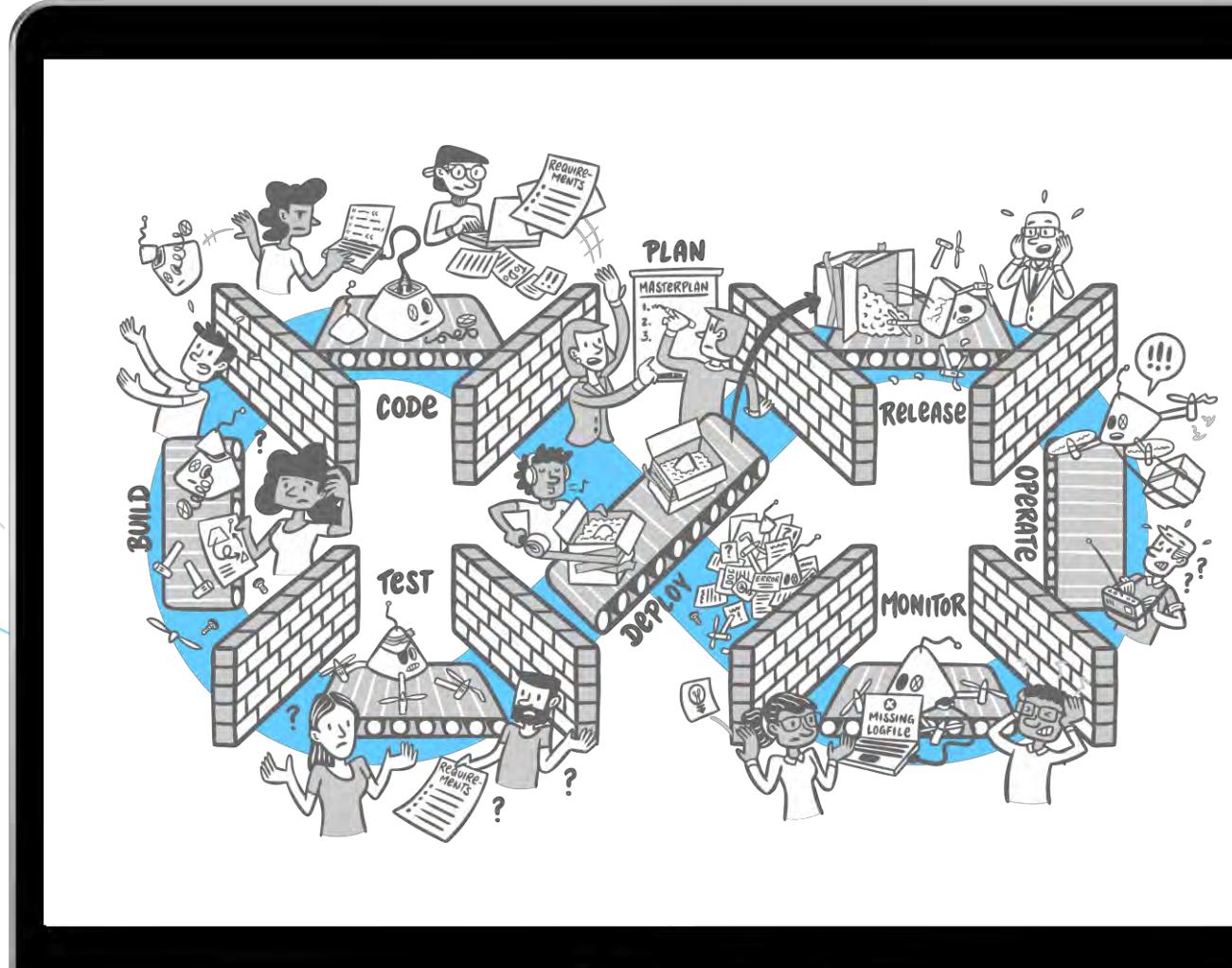
youtube [RomanoRoth](#)

meetup [DevOps Meetup Zürich](#)

event [DevOpsDays Zürich](#)

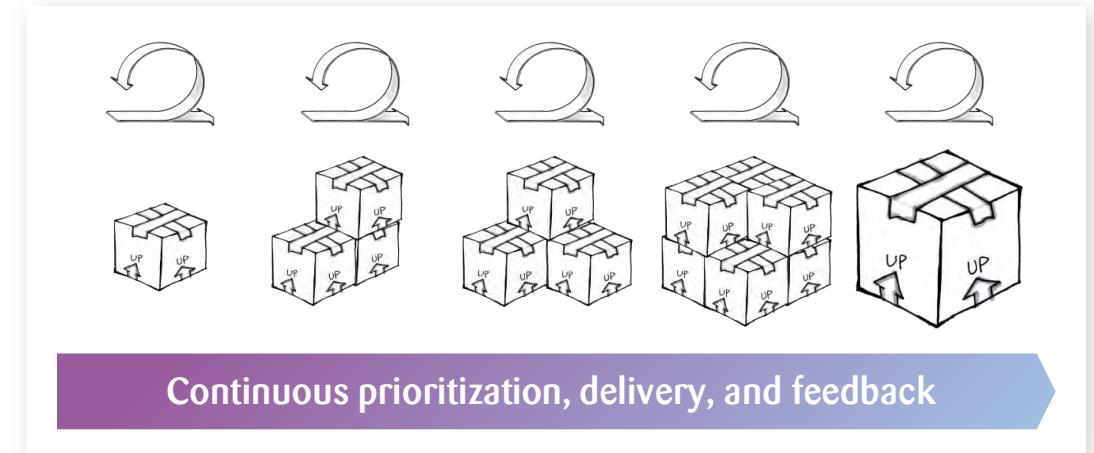
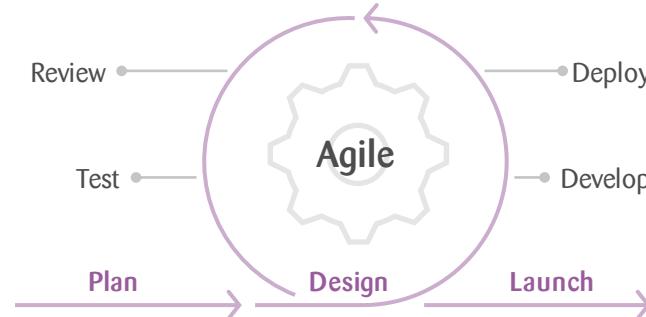
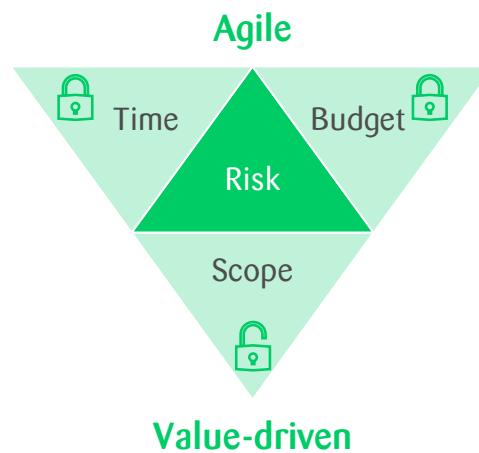
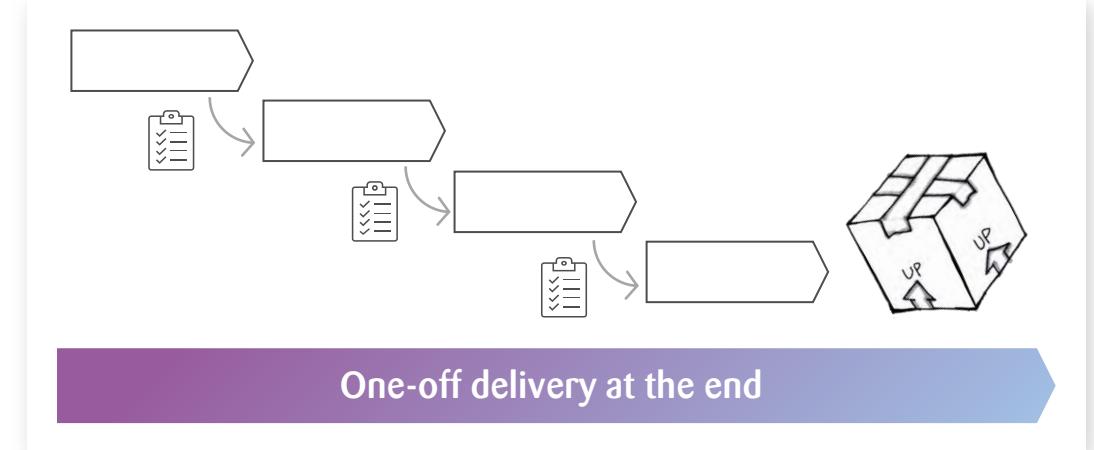
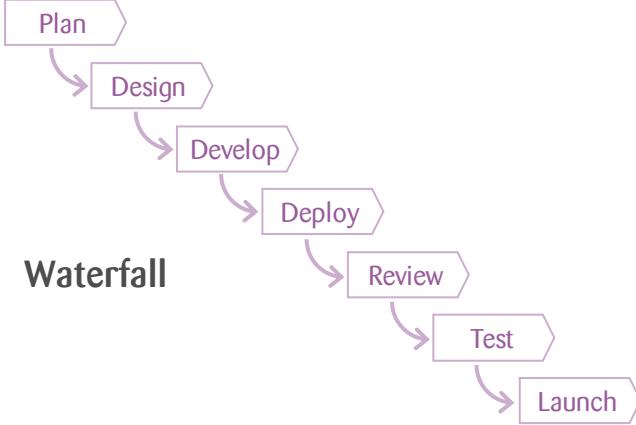
# Today's challenges

- Silo Organization
- Legacy system, technology
- Inflexible and slow processes
- No alignment
- Security and quality is an aftermath
- Cultural resistance
- Regulatory and Compliance



# Where do these challenges come from?

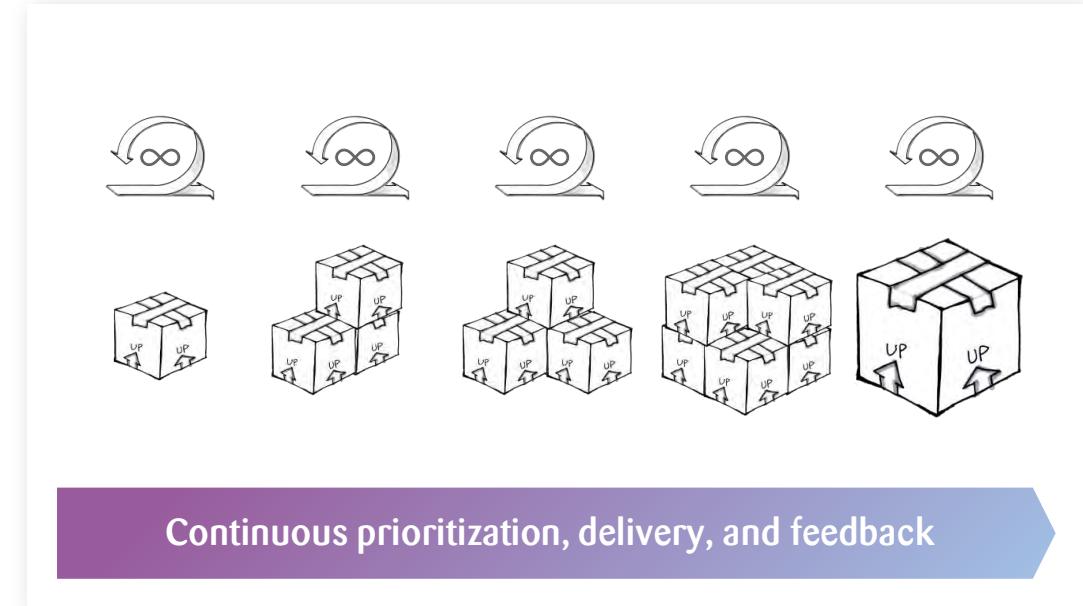
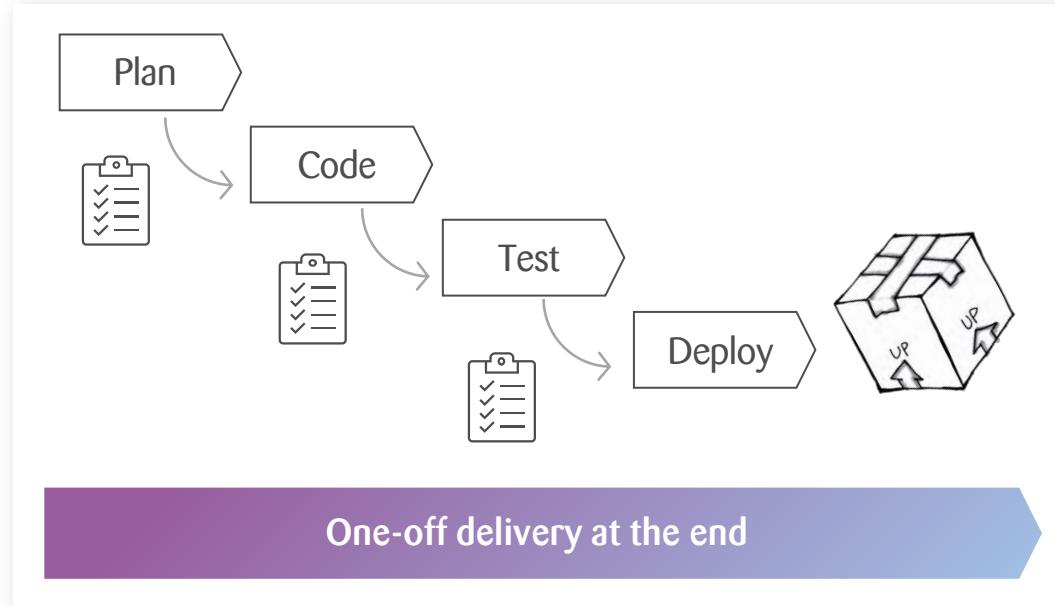
The move from projects to products



**It's about products!**

# What does that mean?

Focus on outcome and NOT output



## Project: Focus on output

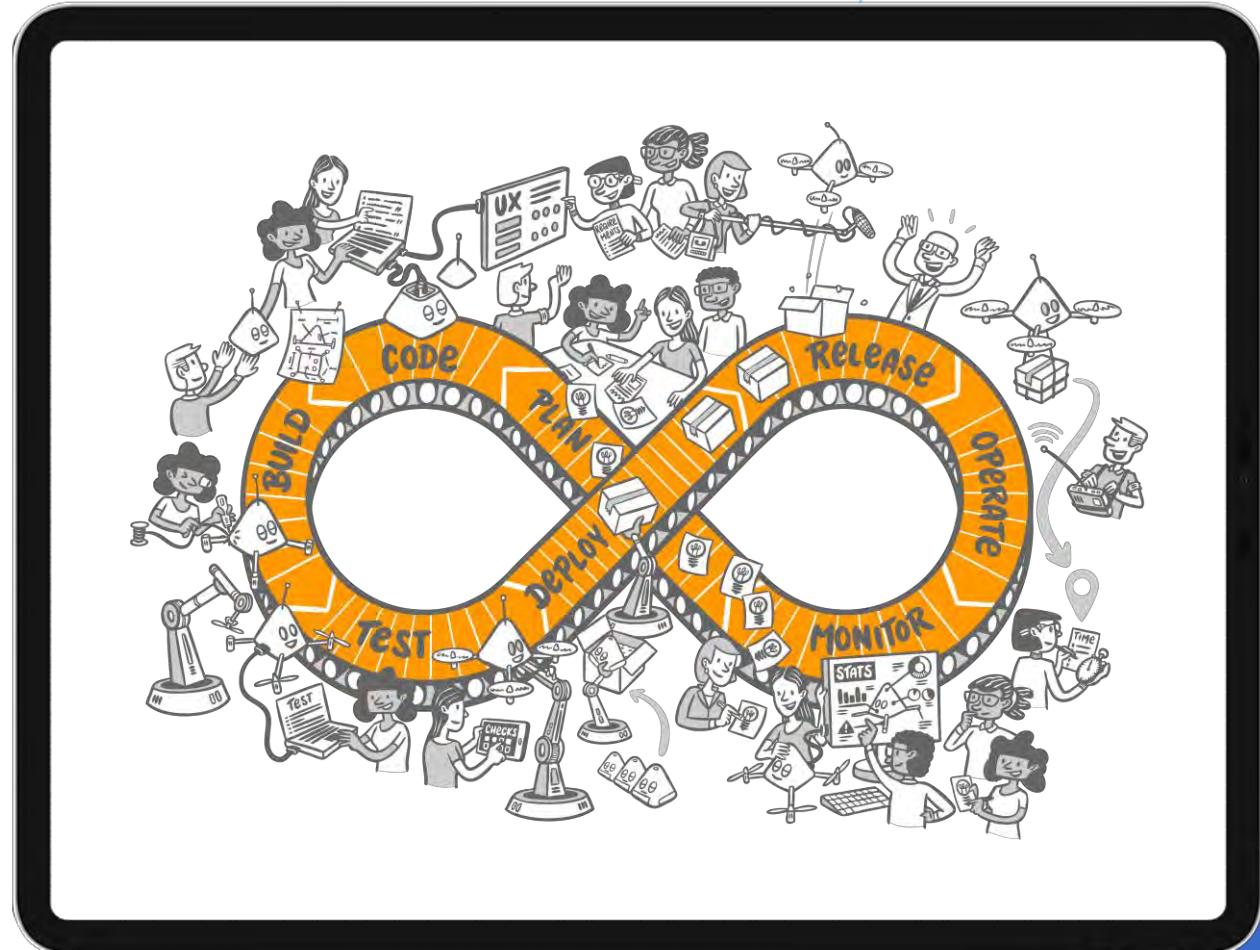
- Maximizing the number of “stuff” produced
- Feature/User stories/Tasks/Code

## Product: Focus on outcome

- Understand customer need
- Problem solved
- Customer adopt or change behaviour

# DevOps is here to help

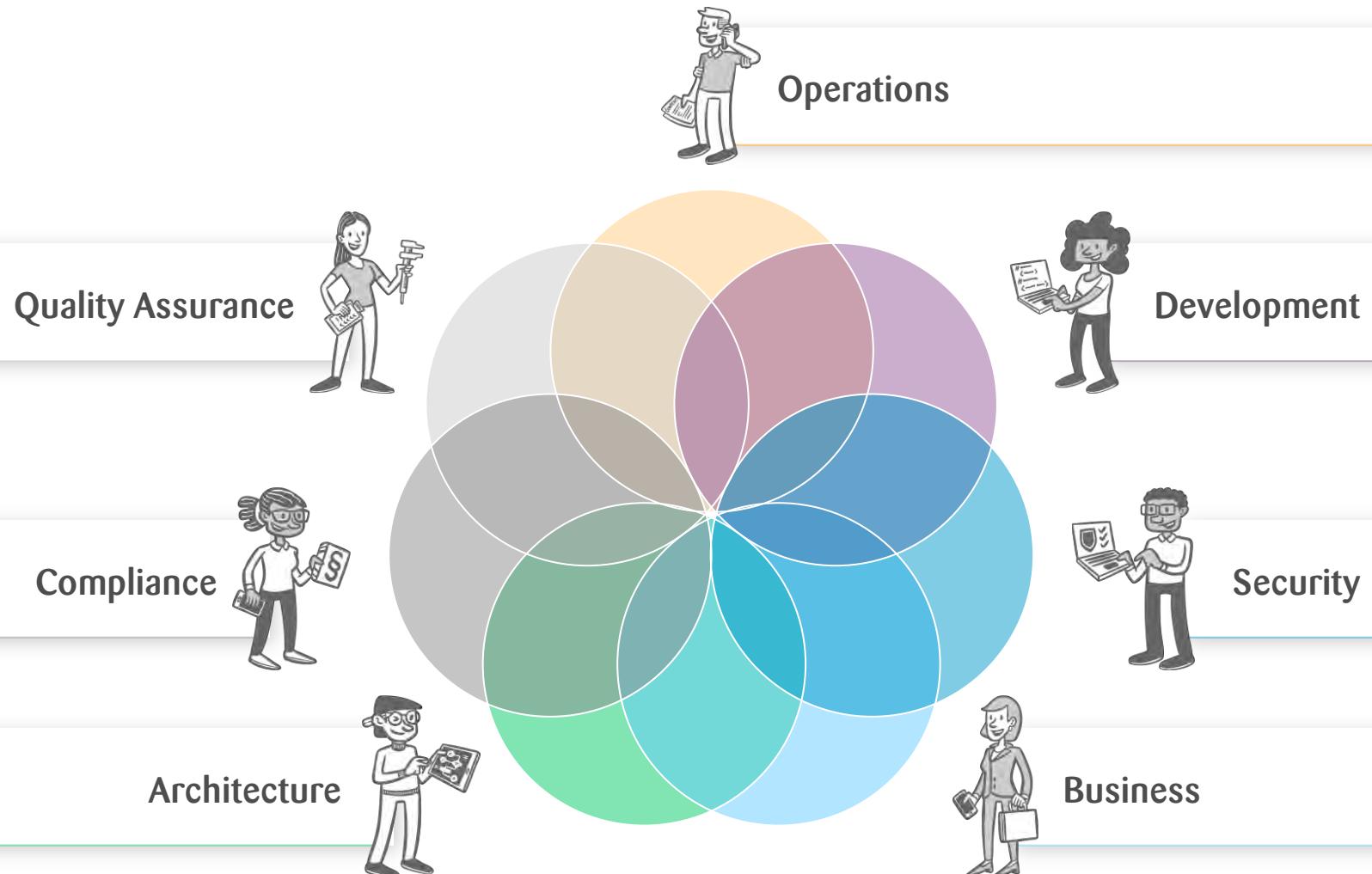
Continuously deliver value



DevOps is a **mindset**, a **culture**, and a set of technical **practices**. It provides communication, integration, automation, and close cooperation among all the people needed to plan, develop, test, deploy, release, and maintain a product.

# Who is DevOps

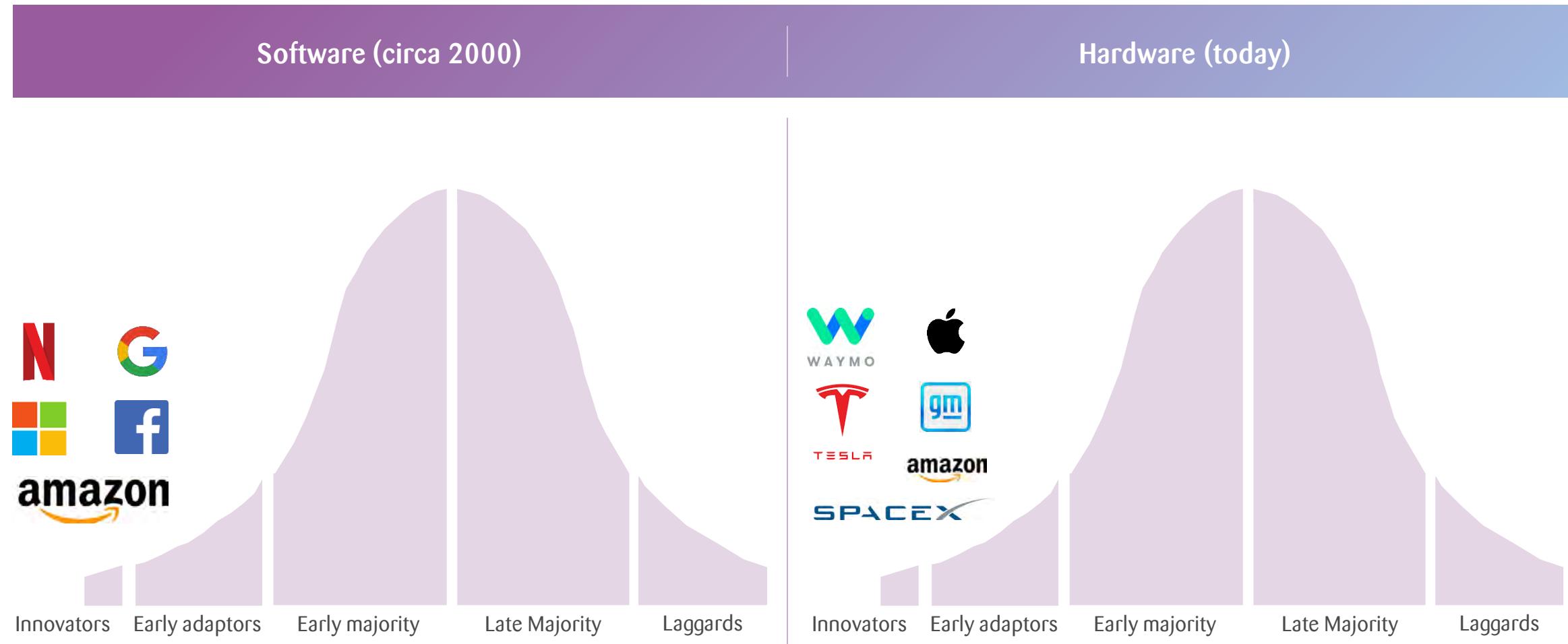
Everyone contributing to the value stream



DevOps:  
Bringing People, Process  
and Technology together to  
continuously deliver value!

# Why is this important for you?

Early adopters in software discovered their own ways of working and now dominate their respective markets.  
Organizations are now applying them to hardware



# Example?

Elon Musk @elonmusk · 7. Okt. 2021  
FSD Beta 10.2 rolls out Friday midnight to ~1000 owners with perfect 100/100 safety scores.  
Rollouts will hold for several days after that to see how it goes.  
If that looks good, beta will gradually begin rolling out to 99 scores & below.

20.338 replies 6.495 retweets 80.305 likes

Elon Musk @elonmusk · 9. Okt. 2021  
A few last minute concerns about this build. Release likely on Sunday or Monday. Sorry for the delay.

3.473 replies 1.341 retweets 24.404 likes

Elon Musk @elonmusk · 11. Okt. 2021  
Beta 10.2 now rolling out to cars with 100/100 safety score over 100 miles

2.597 replies 1.412 retweets 26.515 likes

Elon Musk @elonmusk  
Beta 10.3 releasing Friday next week to all cars with 99/100 safety score  
[Tweet übersetzen](#)  
5:18 vorm. · 15. Okt. 2021 · Twitter for iPhone

1.270 Retweets 244 Zitierte Tweets 25.127 „Gefällt mir“-Angaben

Elon Musk @elonmusk  
Seeing some issues with 10.3, so rolling back to 10.2 temporarily.

Please note, this is to be expected with beta software. It is impossible to test all hardware configs in all conditions with internal QA, hence public beta.

[Tweet übersetzen](#)  
8:44 nachm. · 24. Okt. 2021 · Twitter for iPhone

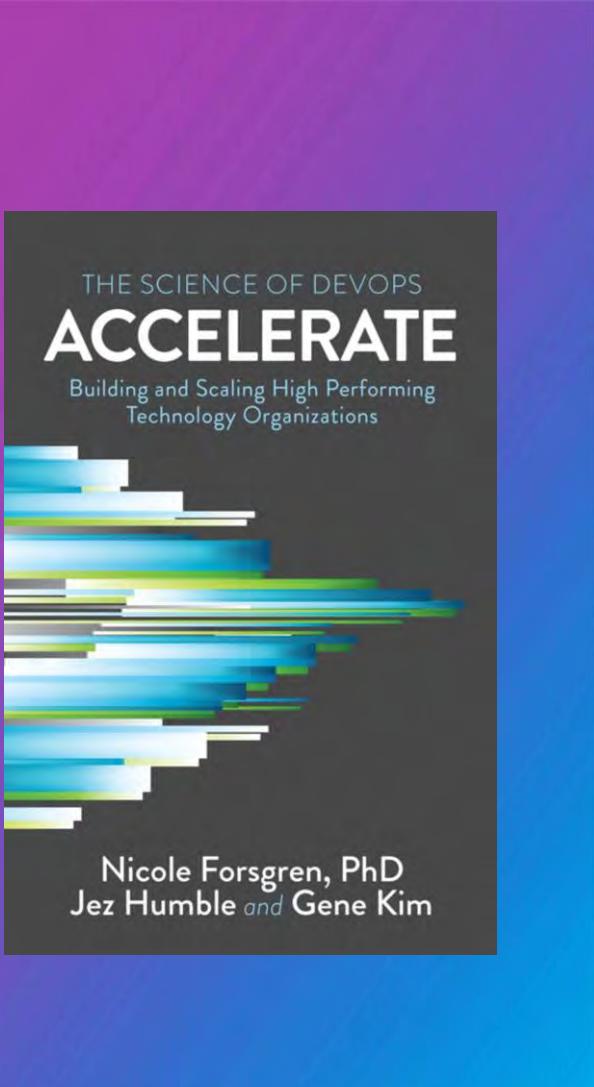
1.049 Retweets 420 Zitierte Tweets 23.502 „Gefällt mir“-Angaben

Elon Musk @elonmusk  
10.3.1 rolling out now  
[Tweet übersetzen](#)  
12:02 nachm. · 25. Okt. 2021 · Twitter for iPhone

958 Retweets 211 Zitierte Tweets 19.448 „Gefällt mir“-Angaben

# The 24 key capabilities that drive improvements in software delivery performance

The capabilities for how to architect for continuous delivery



## CONTINUOUS DELIVERY CAPABILITIES

- Version control
- Deployment automation
- Continuous integration
- Trunk-based development
- Test automation
- Test data management
- Shift left on security
- Continuous delivery (CD)

## ARCHITECTURE CAPABILITIES

- Loosely coupled architecture
- Empowered teams

## LEAN MANAGEMENT AND MONITORING CAPABILITIES

- No Change approval processes
- Monitoring
- Proactive notification
- WIP limits
- Visualizing work

## PRODUCT AND PROCESS CAPABILITIES

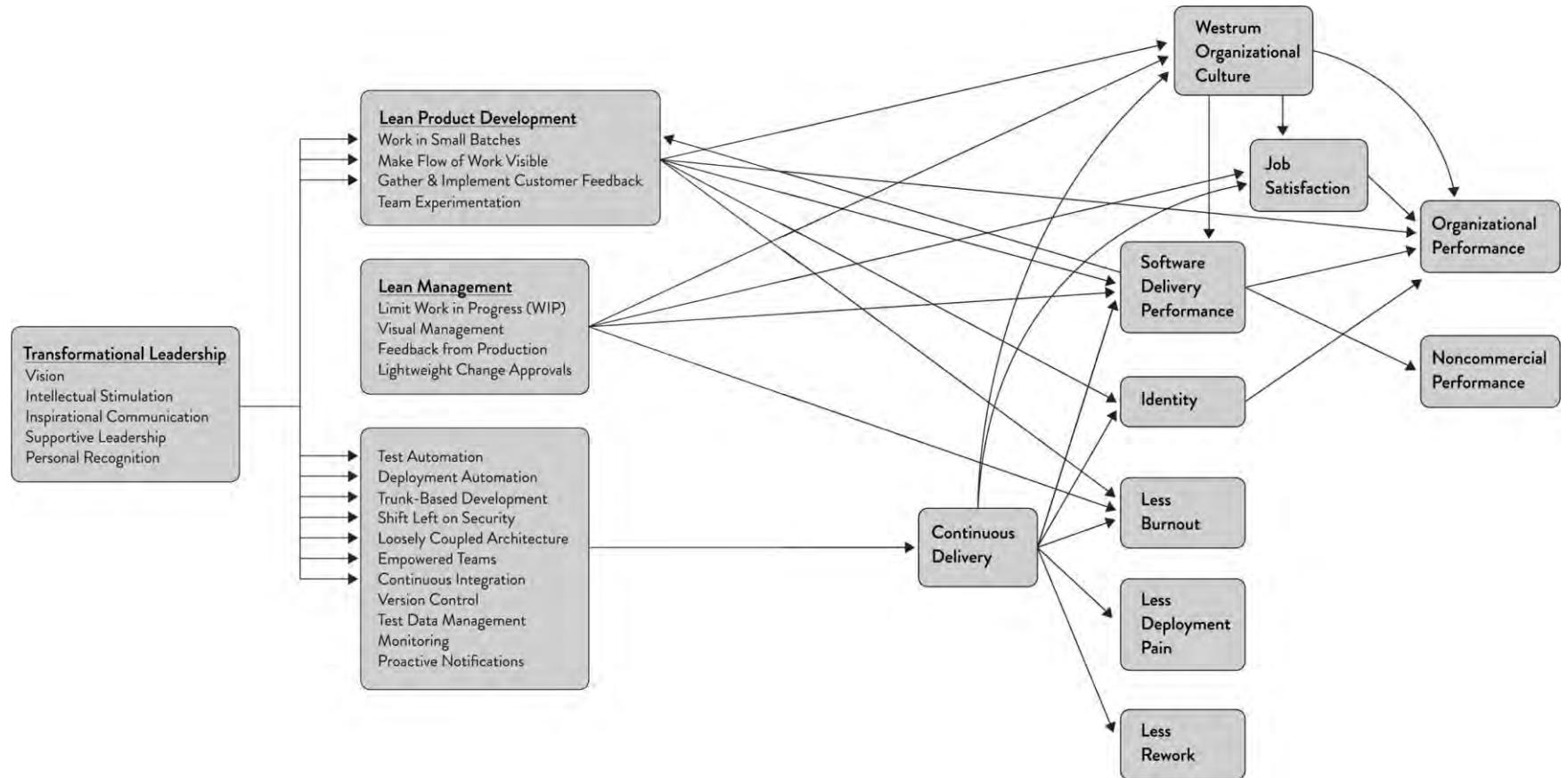
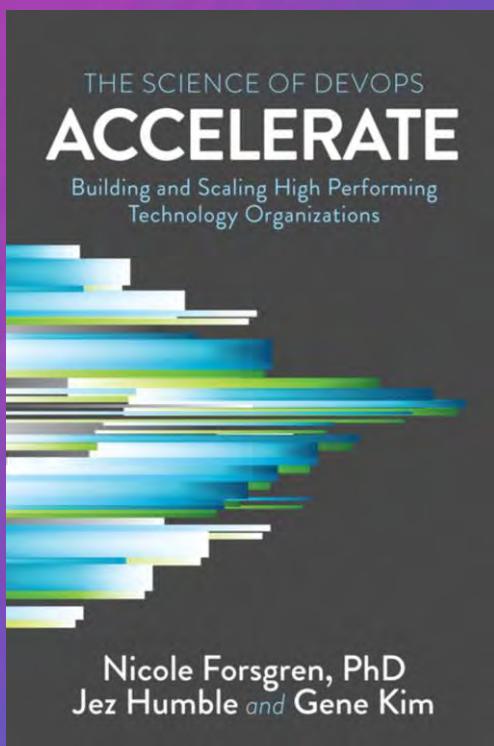
- Customer feedback
- Value stream
- Working in small batches
- Team experimentation

## CULTURAL CAPABILITIES

- Westrum organizational culture
- Supporting learning
- Collaboration among teams
- Job satisfaction
- Transformational leadership

# The science behind DevOps

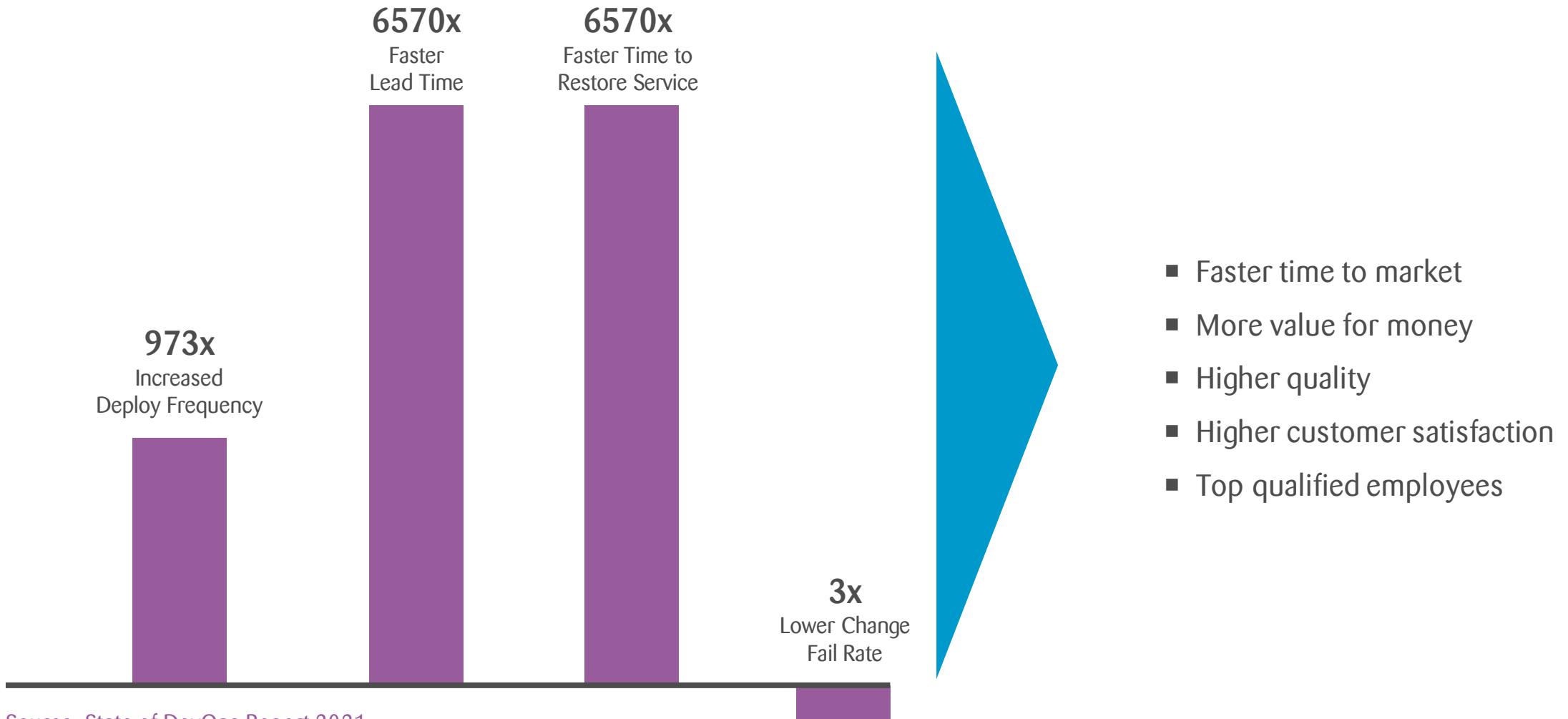
The capabilities for how to architect for continuous delivery



Source: Nicole Forsgren, Jez Humble, Gene Kim, *Accelerate: The Science of Lean Software and DevOps Building and Scaling High Performing Technology Organizations*, IT Revolution Press, March 2018

# The benefits of the DevOps

Comparing highest to lowest performers

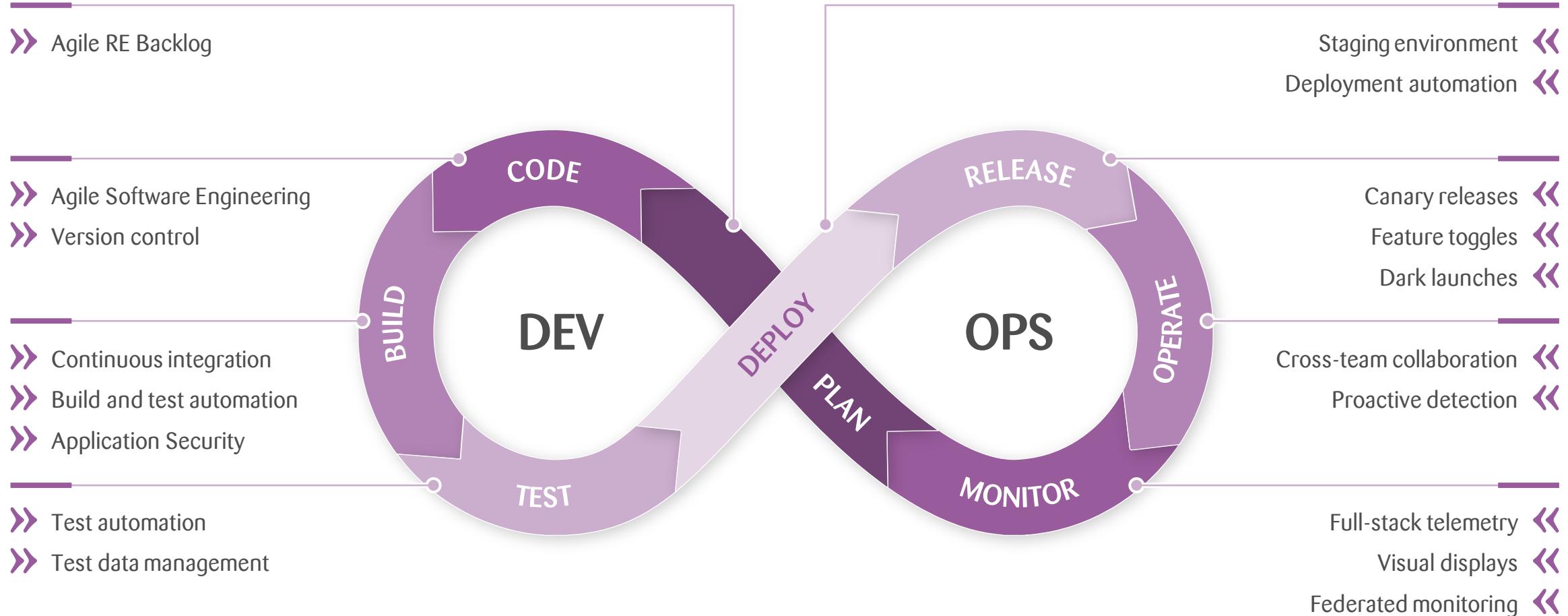


[Source: State of DevOps Report 2021](#)

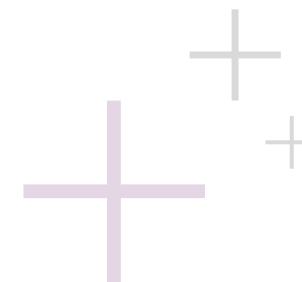
Source: Nicole Forsgren, Jez Humble, Gene Kim, Accelerate: The Science of Lean Software and DevOps Building and Scaling High Performing Technology Organizations, *IT Revolution Press*, March 2018

# Product Development

Modern Software Development is a continuous process across the value stream



» Built-in  
quality



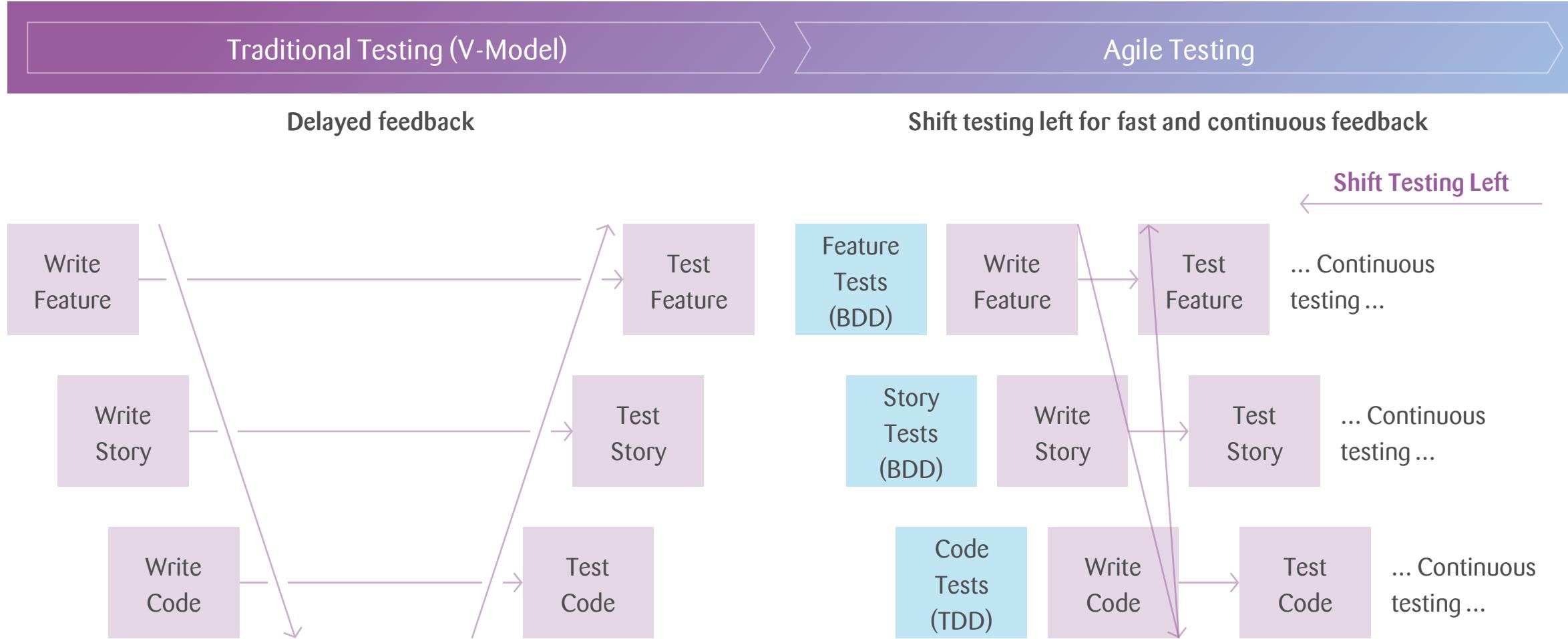
» 50% of the money a Musk company is investing in a new product is in automated testing.

Musk companies: SpaceX, Tesla, Neuralink, The Boring Company, xAI and X

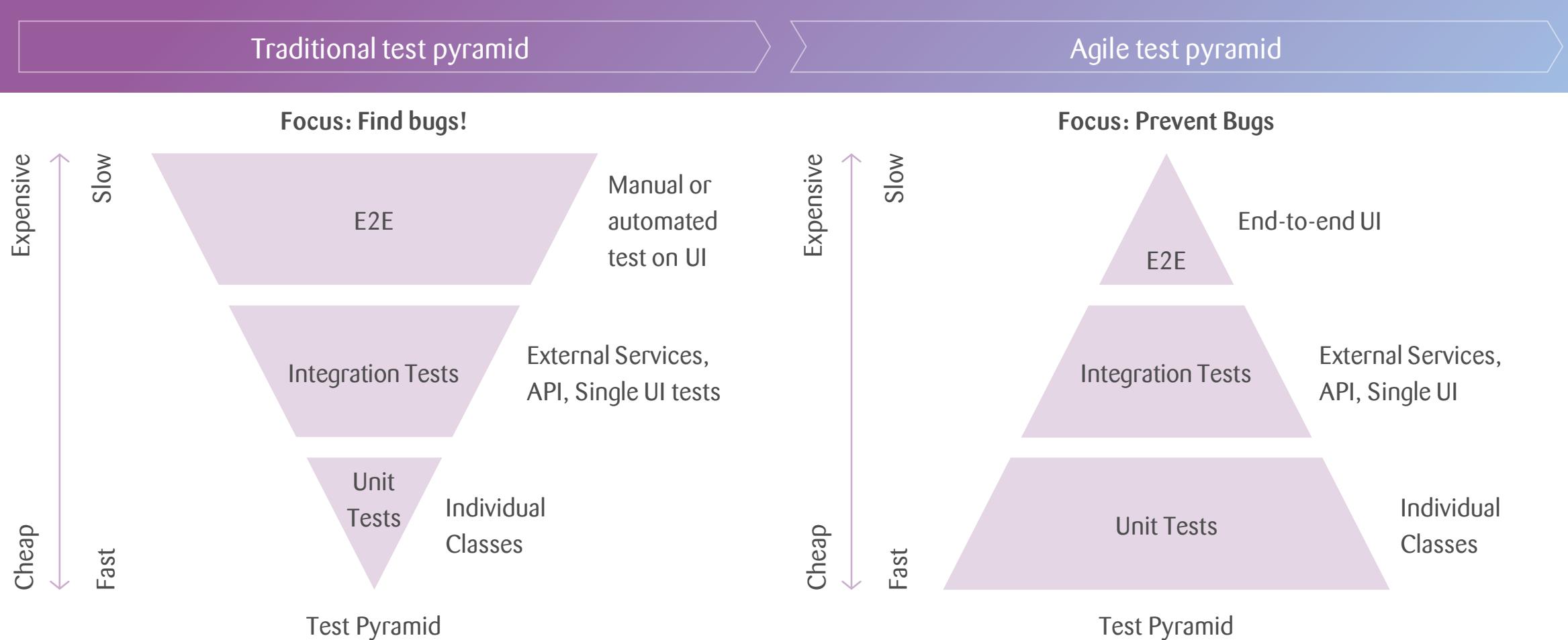


Joe Justice  
Agile Coach  
Ex. Tesla Employee

# Test fast for continuous feedback

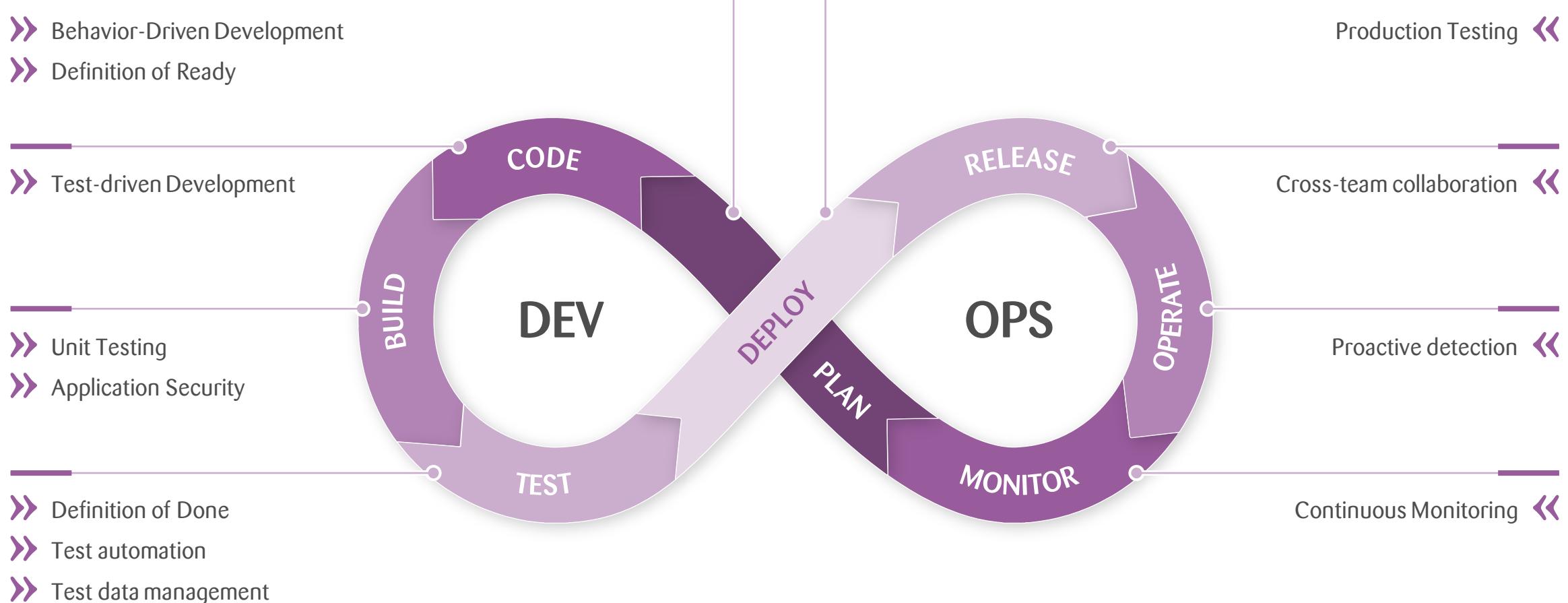


# The right balance of tests

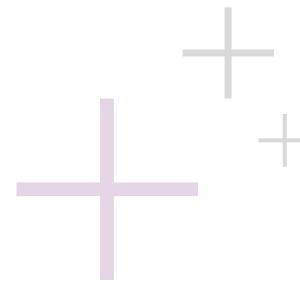


# Continuous Testing

Build quality into your product by testing early and often

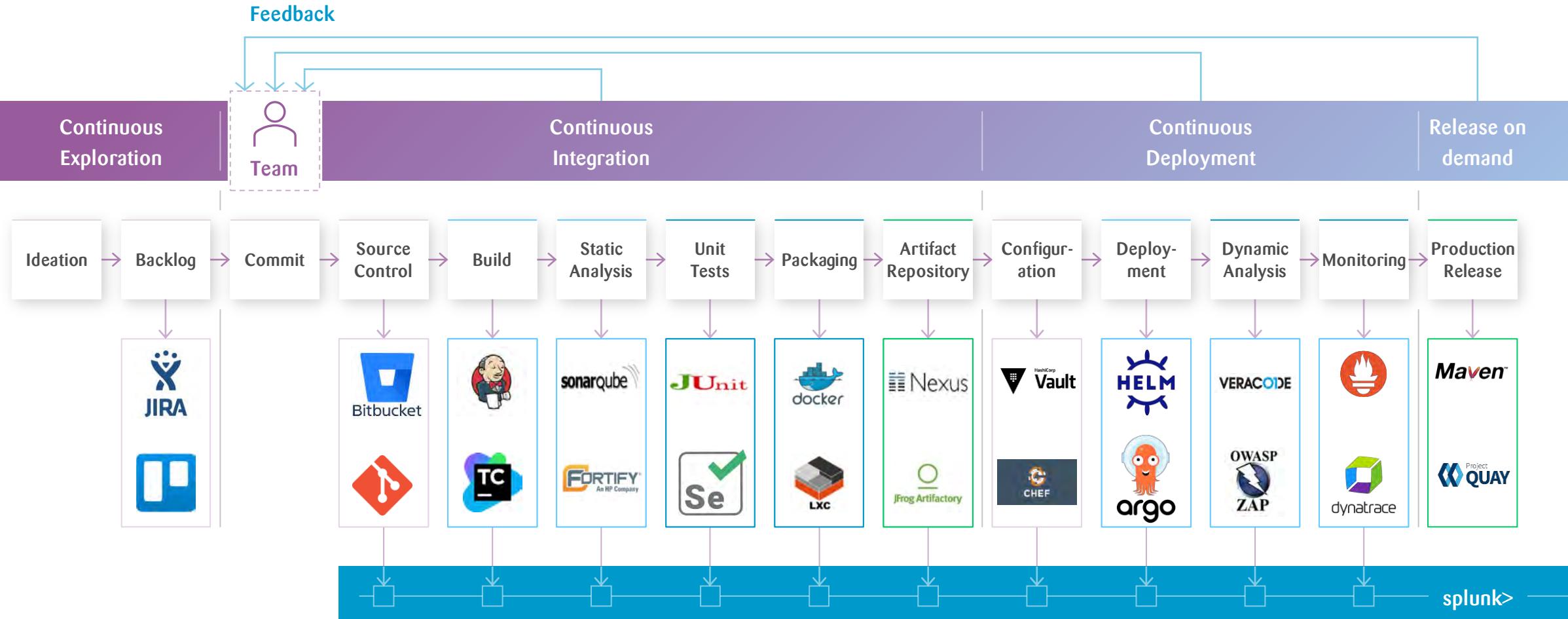


» Built-in  
security

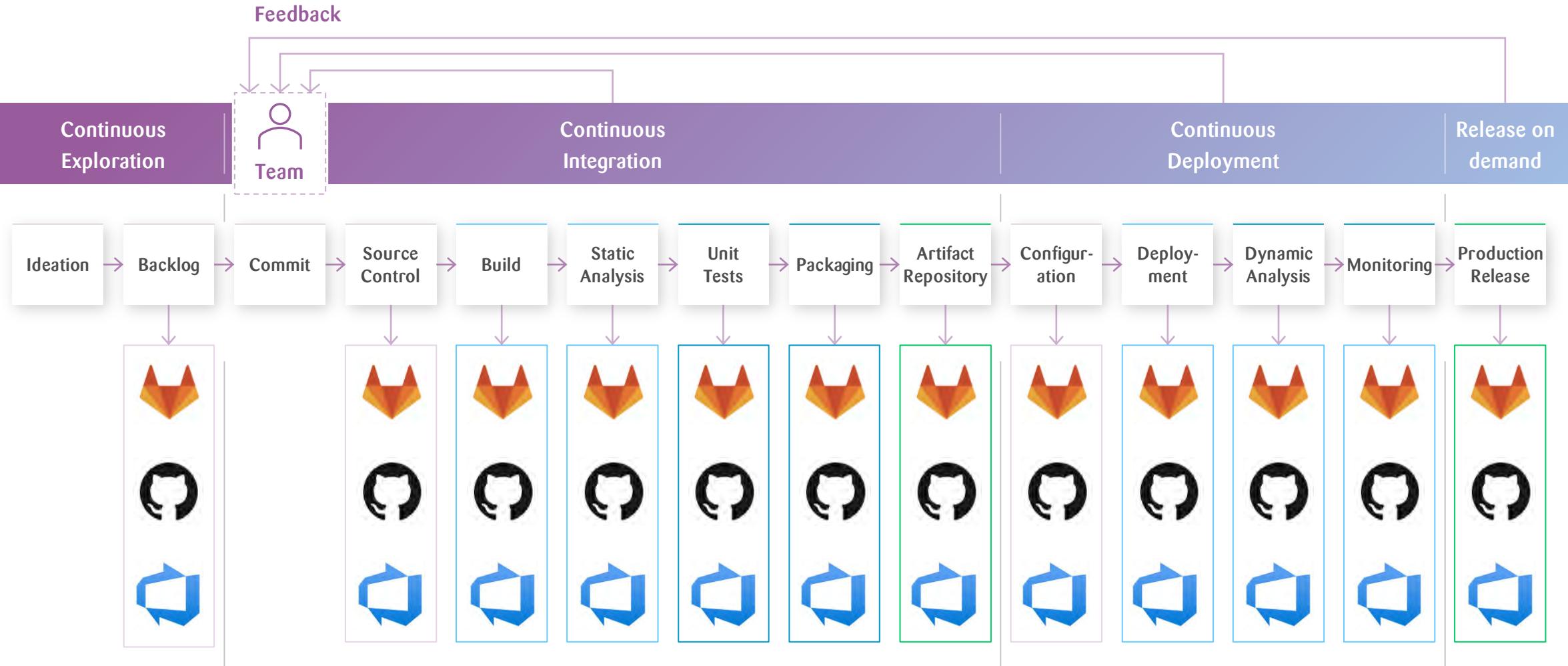


# The Continuous Delivery Pipeline

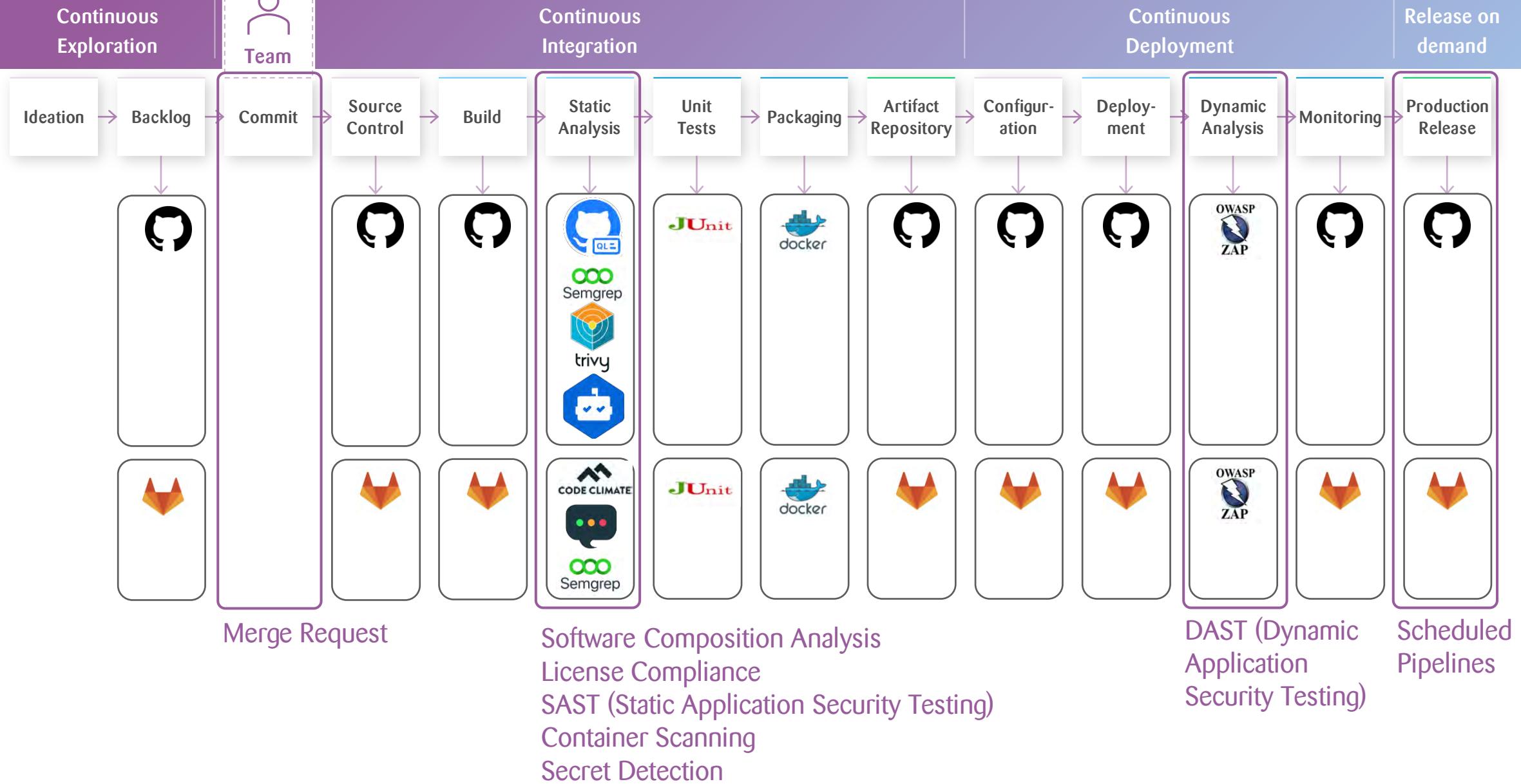
Example of a Continuous Delivery Pipeline with tools that can be used



# What the platform vendors promise....



# DevSecOps





## What is GitLab?

<https://youtu.be/sHK8uN5fBhs>



## Introduction to GitLab

<https://youtu.be/GQ3x9bkCK90>



## SCA

[https://youtu.be/l69W5Ym\\_M5o](https://youtu.be/l69W5Ym_M5o)



## License Compliance

[https://youtu.be/Kmbj\\_PCiHyk](https://youtu.be/Kmbj_PCiHyk)



## SAST

<https://youtu.be/owwIMUamdDc>



## Container Scanning

<https://youtu.be/1AUQ32K6D4>



## Secret Detection

<https://youtu.be/Qs28ONnj00s>



## DAST

<https://youtu.be/Jy1OiuPZrKs>



## Vulnerability Management

<https://youtu.be/XSrIVyv0H1c>



## Merge Request

<https://youtu.be/h4AN7S2gwug>



## Schedule Pipeline

<https://youtu.be/PqPW3zQeP94>



## Recommendations

<https://youtu.be/dphgw9xxjuw>



## What is GitHub?

[https://youtu.be/\\_m5KYEi1ThA](https://youtu.be/_m5KYEi1ThA)



## Introduction to GitHub

<https://youtu.be/6ZdxXDu8ZDA>



## SCA

<https://youtu.be/xM3elerxjYo>



## License Compliance

<https://youtu.be/l7IBh2xkDcQ>



## SAST

<https://youtu.be/p4xS2X5KsNk>



## Container Scanning

[https://youtu.be/\\_ZeKh3GcbgU](https://youtu.be/_ZeKh3GcbgU)



## Secret Detection

<https://youtu.be/k-uuPTLNxGM>



## DAST

[https://youtu.be/v\\_xo1kgNYsE](https://youtu.be/v_xo1kgNYsE)



## Vulnerability Management

<https://youtu.be/XSrIVyv0tbd>



## Pull Requests

<https://youtu.be/h4AN7S2gtbd>



## Schedule Pipeline

<https://youtu.be/PqPW3zQetbd>

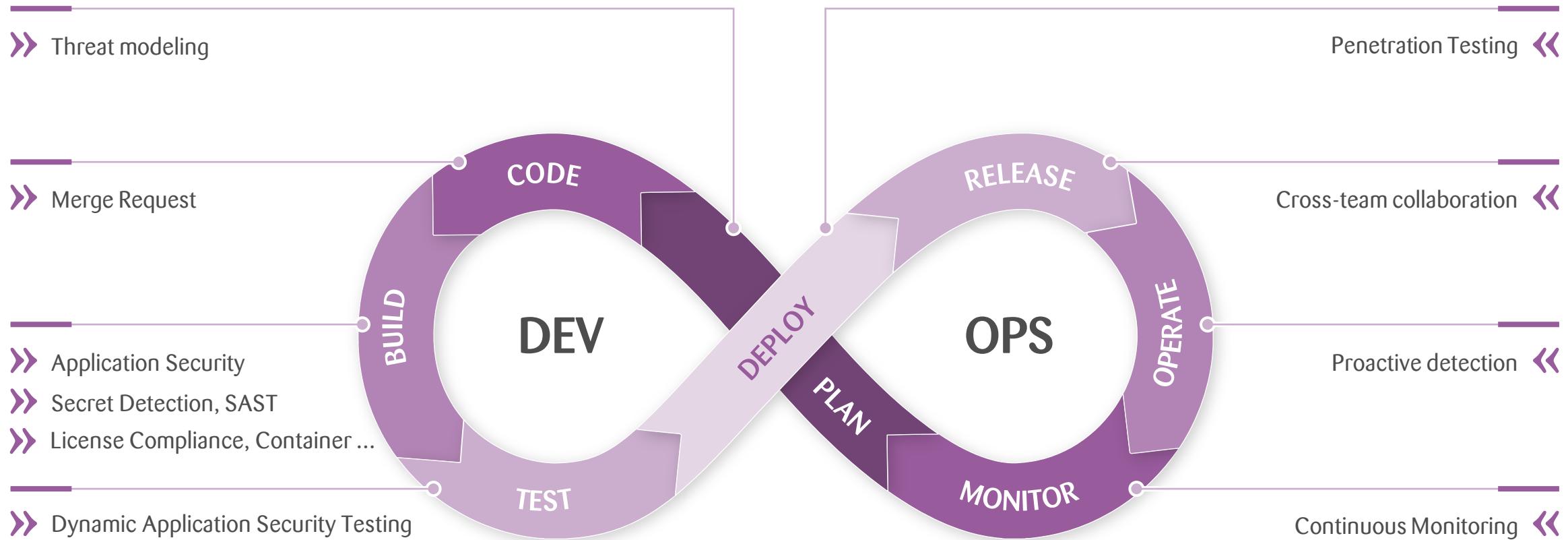


## Recommendations

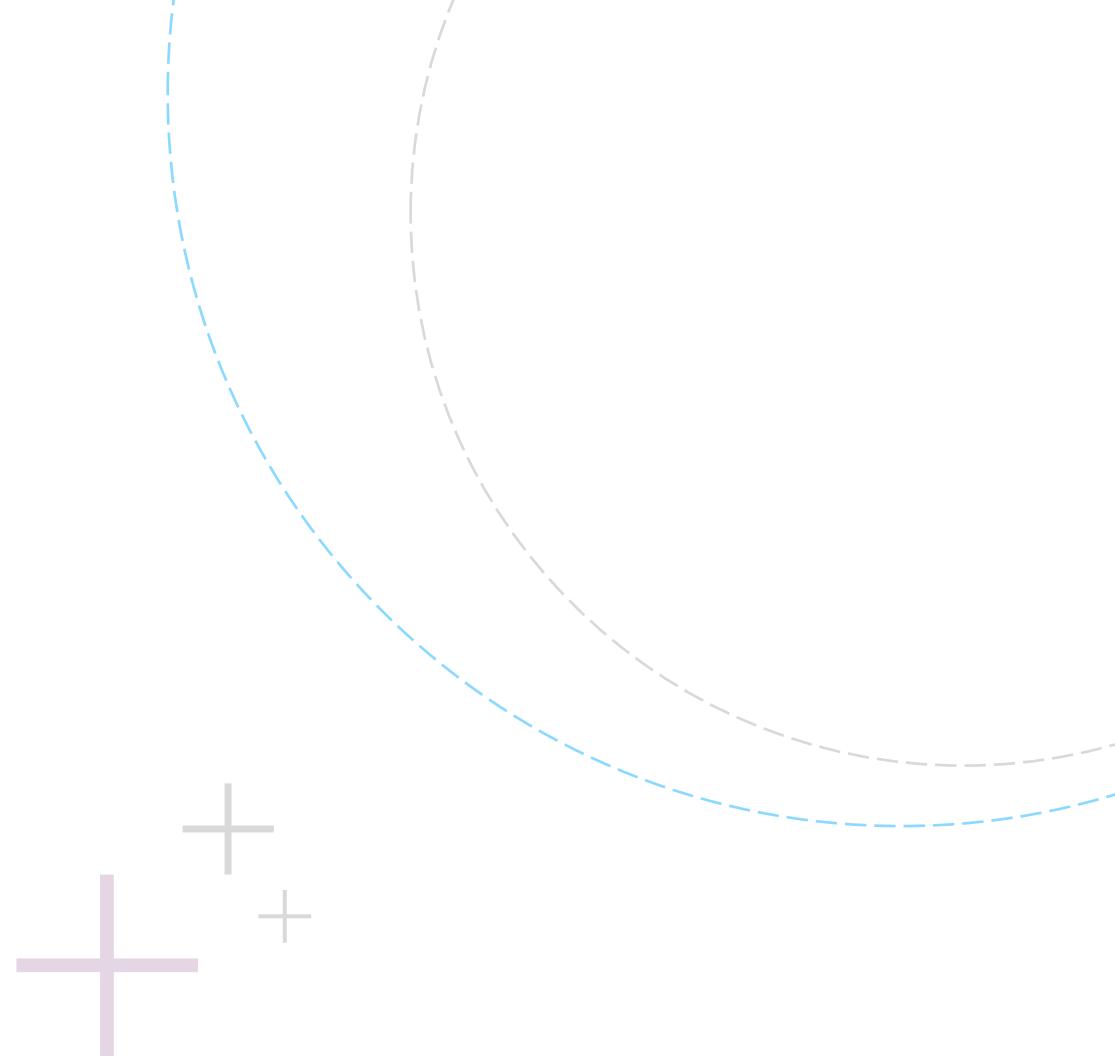
<https://youtu.be/dphgw9xxtbd>

# Continuous Security

Security affects every aspect of the continuous delivery cycle



» Built for  
operability



# You build it you run it!

## Proactive detection



The monitoring systems alert us to dangerous conditions based on tolerance thresholds

Practice disaster recovery (DR) procedures

Create a notification strategy

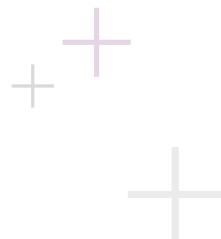


## Cross-team collaboration

Teams across the Value Stream work together to detect, resolve, and learn from production incidents

Responding to production failures is a cross-functional responsibility

Hold incident post-mortems

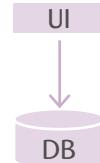


# Evolution of monitoring

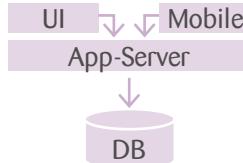


## Typical Systems

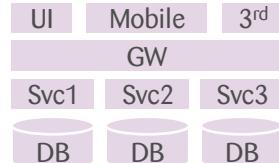
Database centric  
2-tier applications/  
mainframes



Application  
server centric  
3-Tier applications



Distributed  
service oriented  
applications



## Challenge

Is my holy cow – the DB – still running?

Numbers are not enough. The Logic is in the  
App-Server! What happens in the App-Server?

There is no single App-Server anymore! Are  
all parts on all levels contributing to requests  
working as intended?



## Solution

Metrics

Logs

Monitoring

Traces

Application-Monitoring

Infrastructure-Monitoring

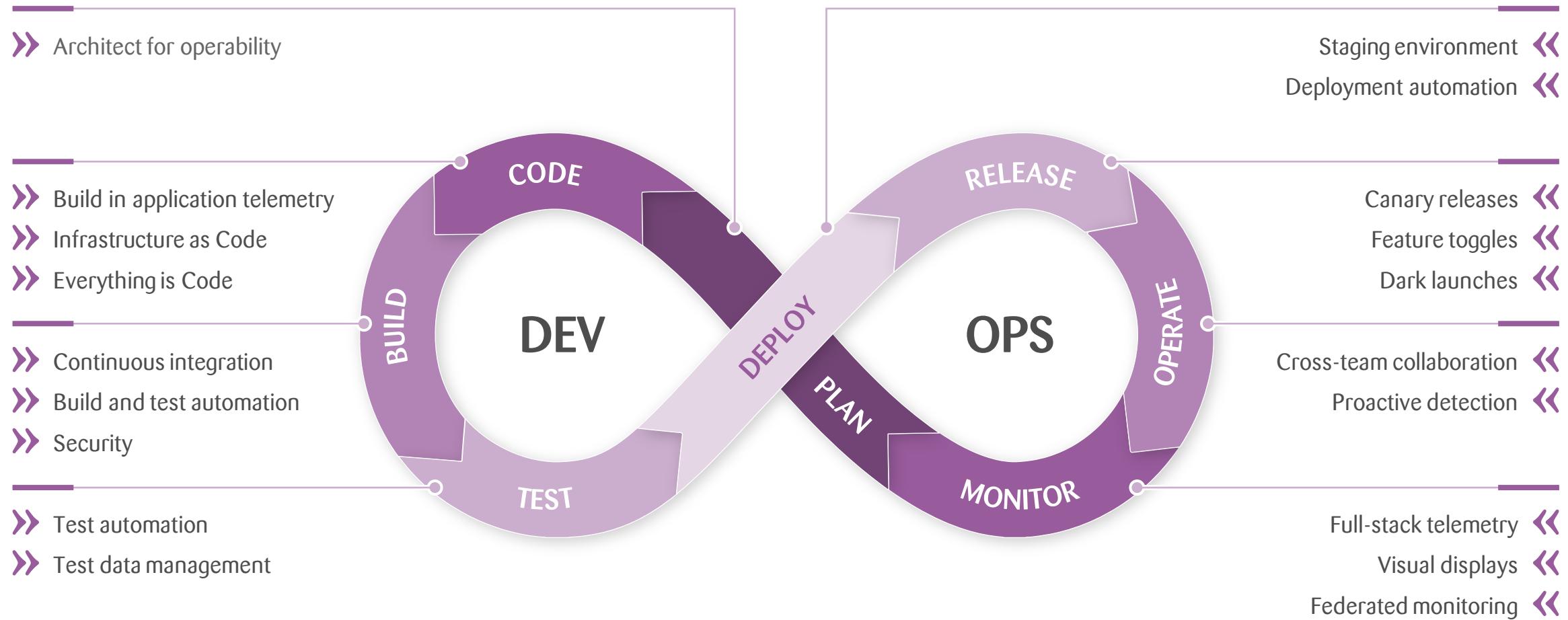
Observability

Gather metrics about Hardware and DB-Load  
and inspect logs in case of failure.

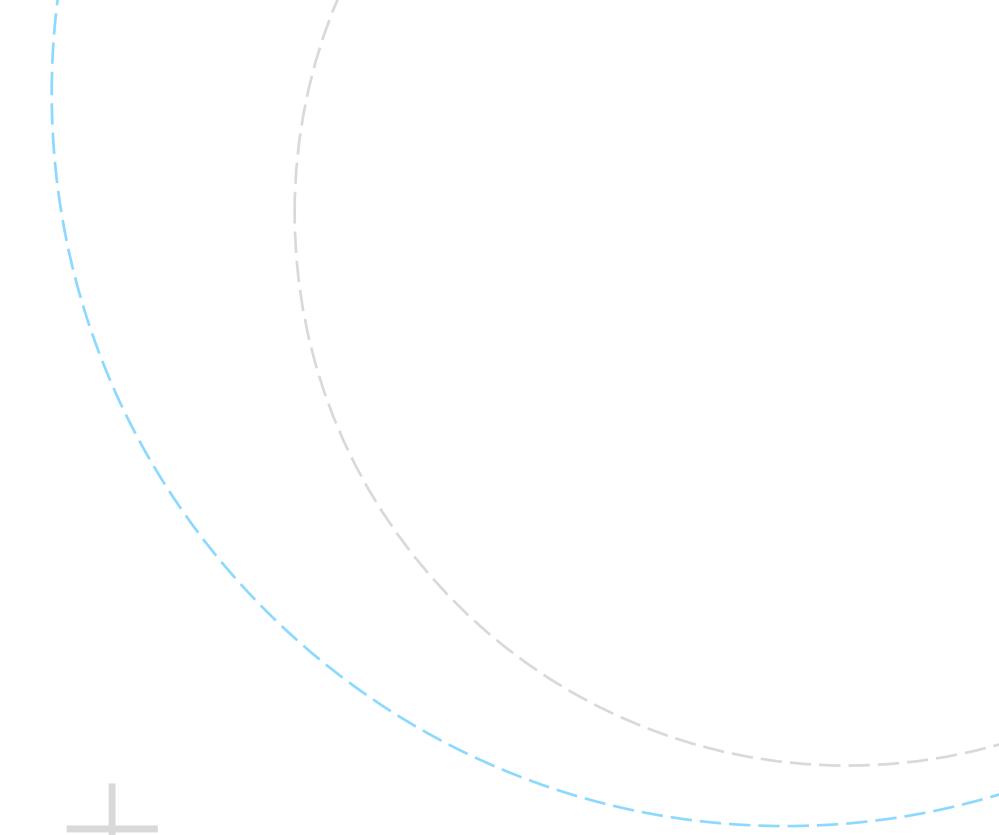
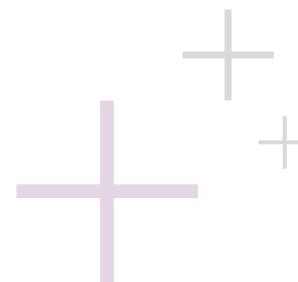
Looking into an application is something  
completely new, done by other people.  
Let's call it application monitoring in contrast  
to the old one, which we rename infra-  
structure monitoring.

Weird behavior could originate from all  
levels. Metrics, Logs and Traces need to  
be centralized and correlated – facilitating  
easy navigation through the huge amount  
of available data.

# We need to architect for operability

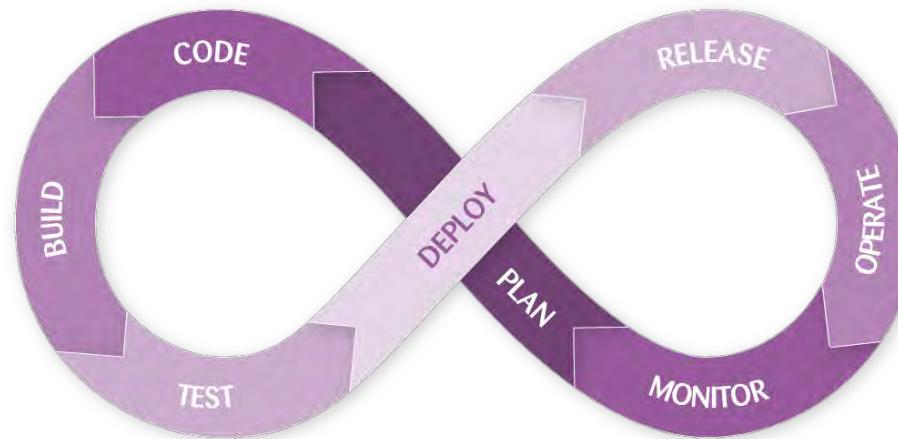


» Build a  
platform



# Modern Software Development is a continuous process across the value stream

DevOps is a mindset, a culture, and a set of **technical practices**.



Plan	Code	Build	Test	Deploy	Operate	Release	Monitor
<ul style="list-style-type: none"> <li>▪ Backlog management</li> <li>▪ Value Stream Management</li> <li>▪ UX</li> <li>▪ Lean Portfolio Management</li> <li>▪ BDD</li> <li>▪ Architecture</li> <li>▪ Quality Management</li> <li>▪ Agile SW Architecture</li> <li>▪ DoR</li> </ul>	<ul style="list-style-type: none"> <li>▪ Git</li> <li>▪ IDE</li> <li>▪ IaC</li> <li>▪ Code Review (PR)</li> <li>▪ Documentation</li> <li>▪ Secret Mgmt.</li> <li>▪ Agile SWE</li> <li>▪ TDD</li> <li>▪ Remote Development</li> <li>▪ Emergent SW Architecture</li> </ul>	<ul style="list-style-type: none"> <li>▪ Continuous Integration</li> <li>▪ Software Composition Analysis</li> <li>▪ License Compliance</li> <li>▪ SAST</li> <li>▪ Container Scanning</li> <li>▪ Secret Detection</li> <li>▪ Container Registry</li> </ul>	<ul style="list-style-type: none"> <li>▪ Test Automation</li> <li>▪ Test Data Management</li> <li>▪ Vulnerability Management</li> <li>▪ Compliance Management</li> <li>▪ SAST</li> <li>▪ Container Scanning</li> <li>▪ Secret Detection</li> <li>▪ Container Registry</li> </ul>	<ul style="list-style-type: none"> <li>▪ DAST</li> <li>▪ API Security</li> <li>▪ Scheduled Pipelines</li> <li>▪ Environment Management</li> <li>▪ Audit Security Policy</li> <li>▪ Logging</li> <li>▪ Tracing</li> <li>▪ DoD</li> </ul>	<ul style="list-style-type: none"> <li>▪ Cross-team collaboration</li> <li>▪ Proactive detection</li> <li>▪ Logging</li> <li>▪ Tracing</li> </ul>	<ul style="list-style-type: none"> <li>▪ Release Orchestration</li> <li>▪ Feature flags</li> <li>▪ Canary releases</li> <li>▪ Dark launches</li> </ul>	<ul style="list-style-type: none"> <li>▪ Continuous Monitoring</li> <li>▪ Full-stack telemetry</li> <li>▪ Visual displays</li> <li>▪ Federated monitoring</li> <li>▪ Observability</li> <li>▪ Metrics (DORA)</li> <li>▪ On-Call Schedule Mgmt.</li> <li>▪ Incident Mgmt.</li> <li>▪ Service desk</li> <li>▪ Software Bill of Material</li> </ul>

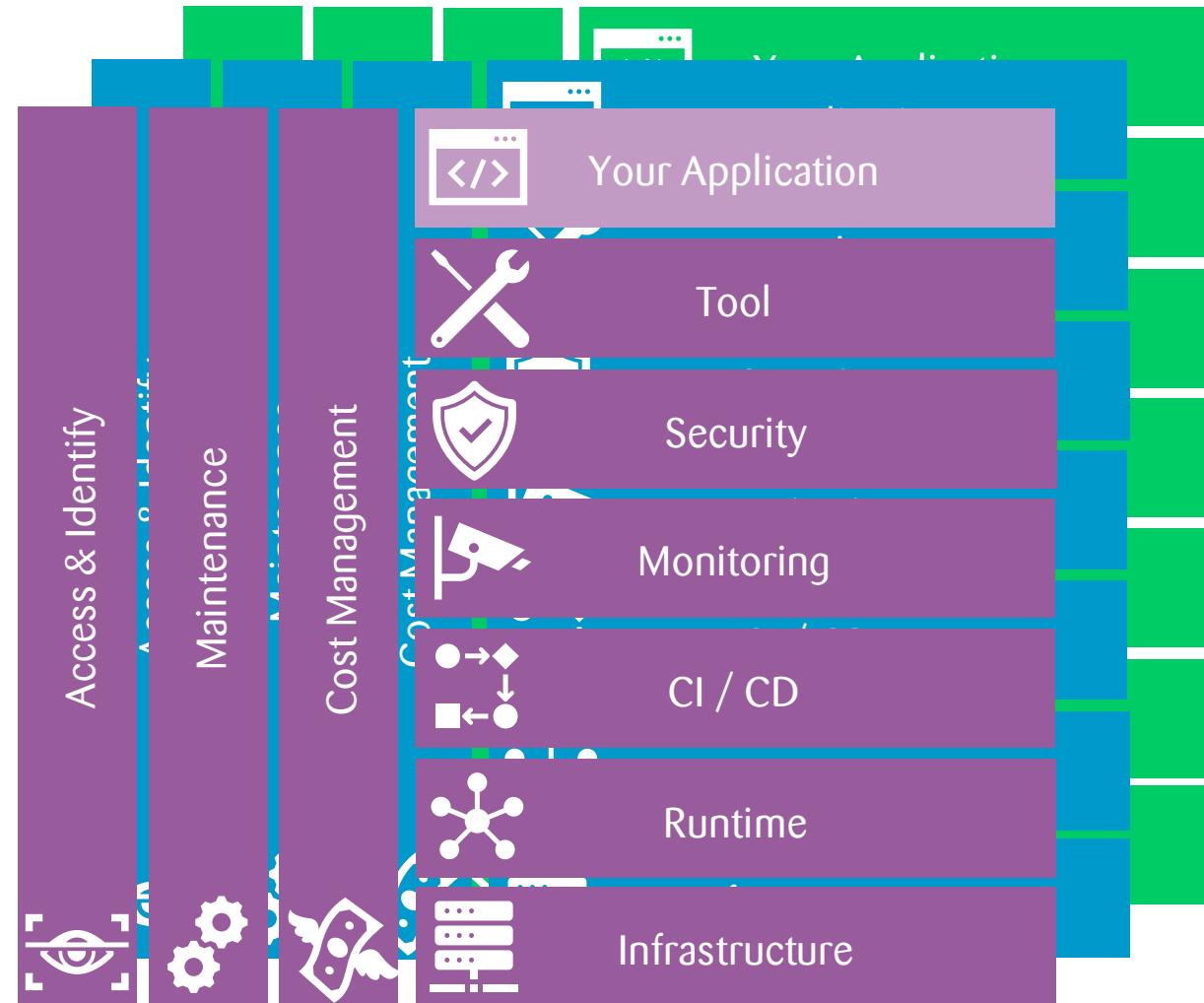
# You need to take care about the full stack

You build it you run it!



# Think about that on scale...

The cognitive load is too high!



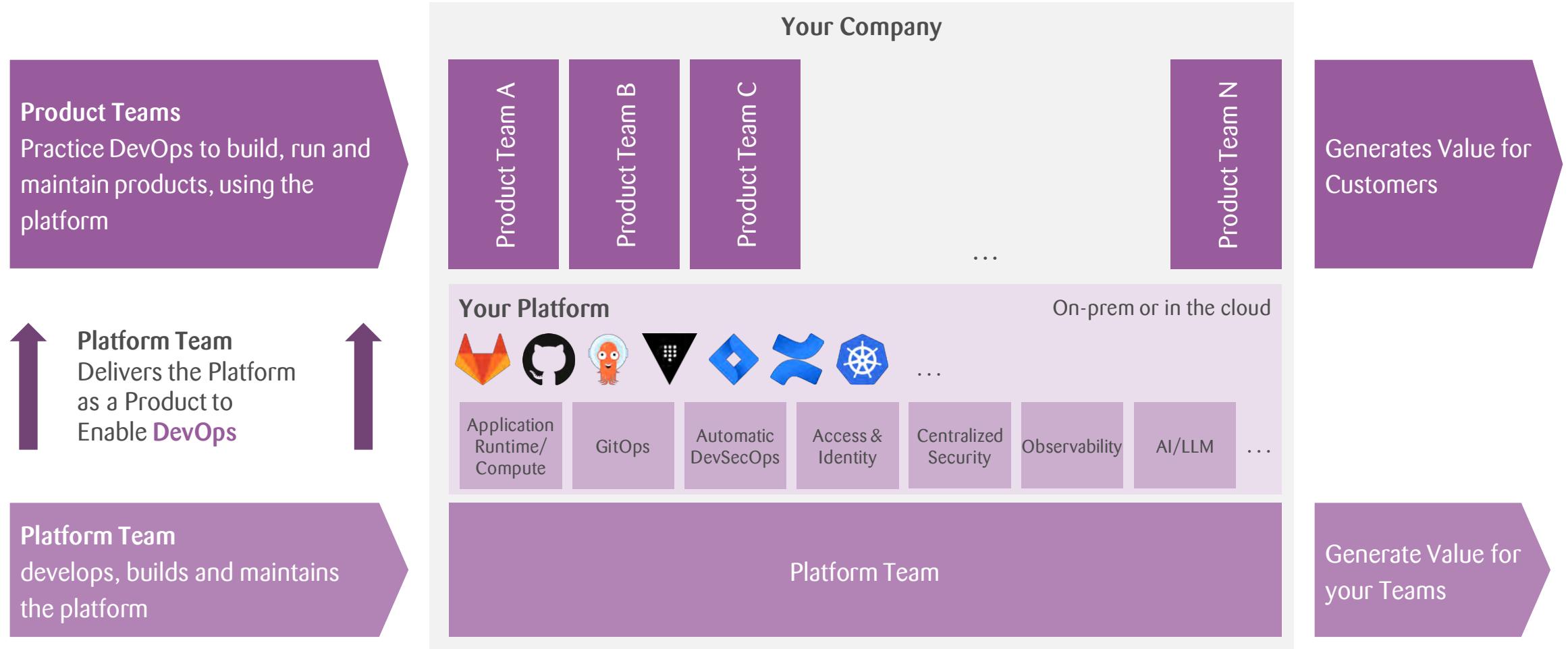
- Inconsistent and redundant platforms
- Every team reinvents the wheel
- Lack of operations experience in development
- No synergies
- It's not easy to move from one project / product to another
- Complexity in tools

👉 The cognitive load is too high

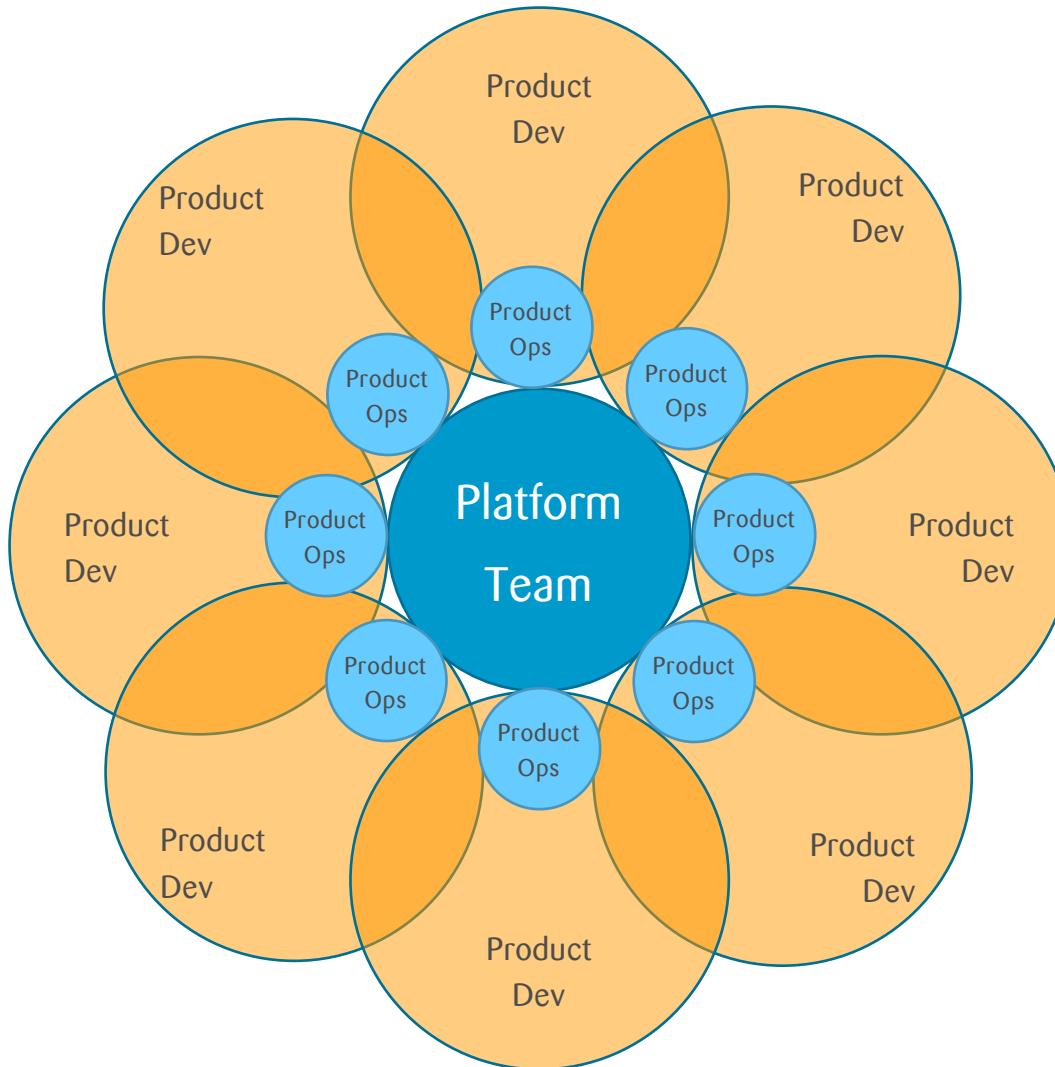


# Platform Engineering Enables DevOps in Product Teams

The platform team reduce the cognitive load



# Platform Engineering Scales the Platform to Multiple Product Teams

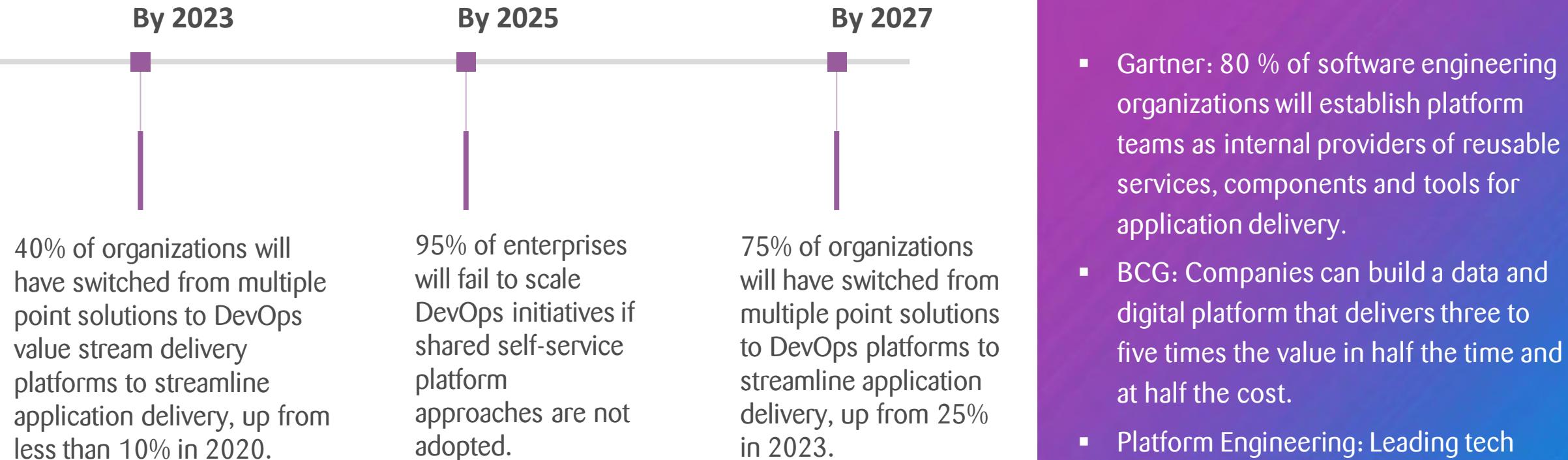


**Platform engineering implements reusable tools and self-service capabilities with automated infrastructure operations, improving the developer experience and productivity.**

**This technology approach utilizes reusable configurable application components and services.**

**The benefit to users is in standardized tools, components and automated processes.**

# Gartner and international consultancies see platform engineering as a key technology trend in the next 1-2 years



All Sources from 2023: Gartner – Technology trends, McKinsey and BCG studies about Platform Engineering and the association of Platform Engineering

# Summary

This is how to continuously build great products!

00

You're developing  
a products, not  
projects



01

DevOps: Bringing People,  
Process and Technology together  
to continuously deliver value



02

Apply continuous  
testing and build-in  
quality



03

Apply continuous  
security



04

Build a platform



# We are entering the age of industrialization of Software Development

Platform Engineering builds your Platform which enables teams to do DevSecOps.

This is the way how you can architect for continuous delivery!

