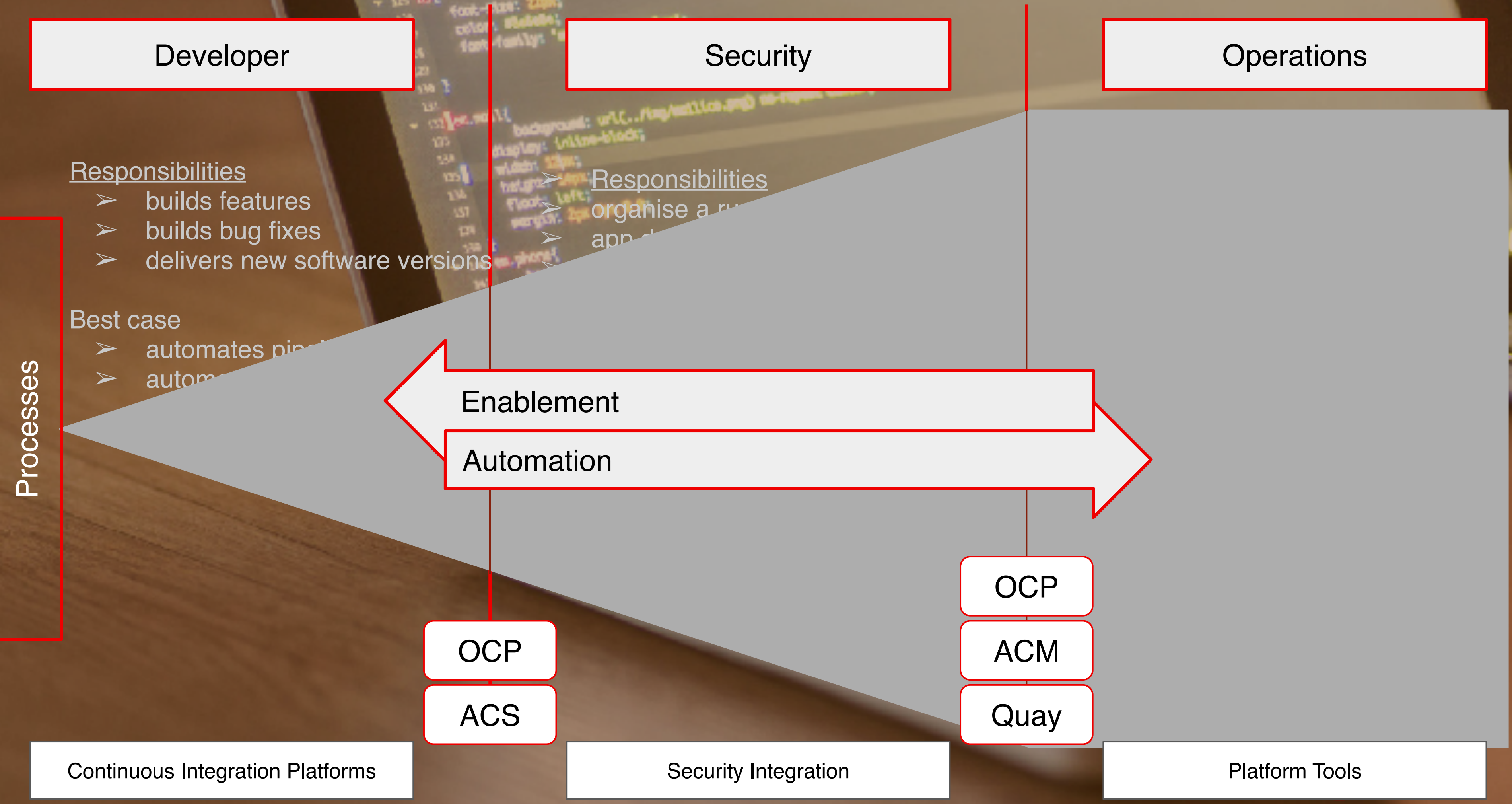
The Strategic Benefits of an Internal Development Platform Integrated with Golden Path

Driving Software Development Success

What's happening?

How did we end up here?

Why should I care?



Developer

Security

Operations

Responsibilities

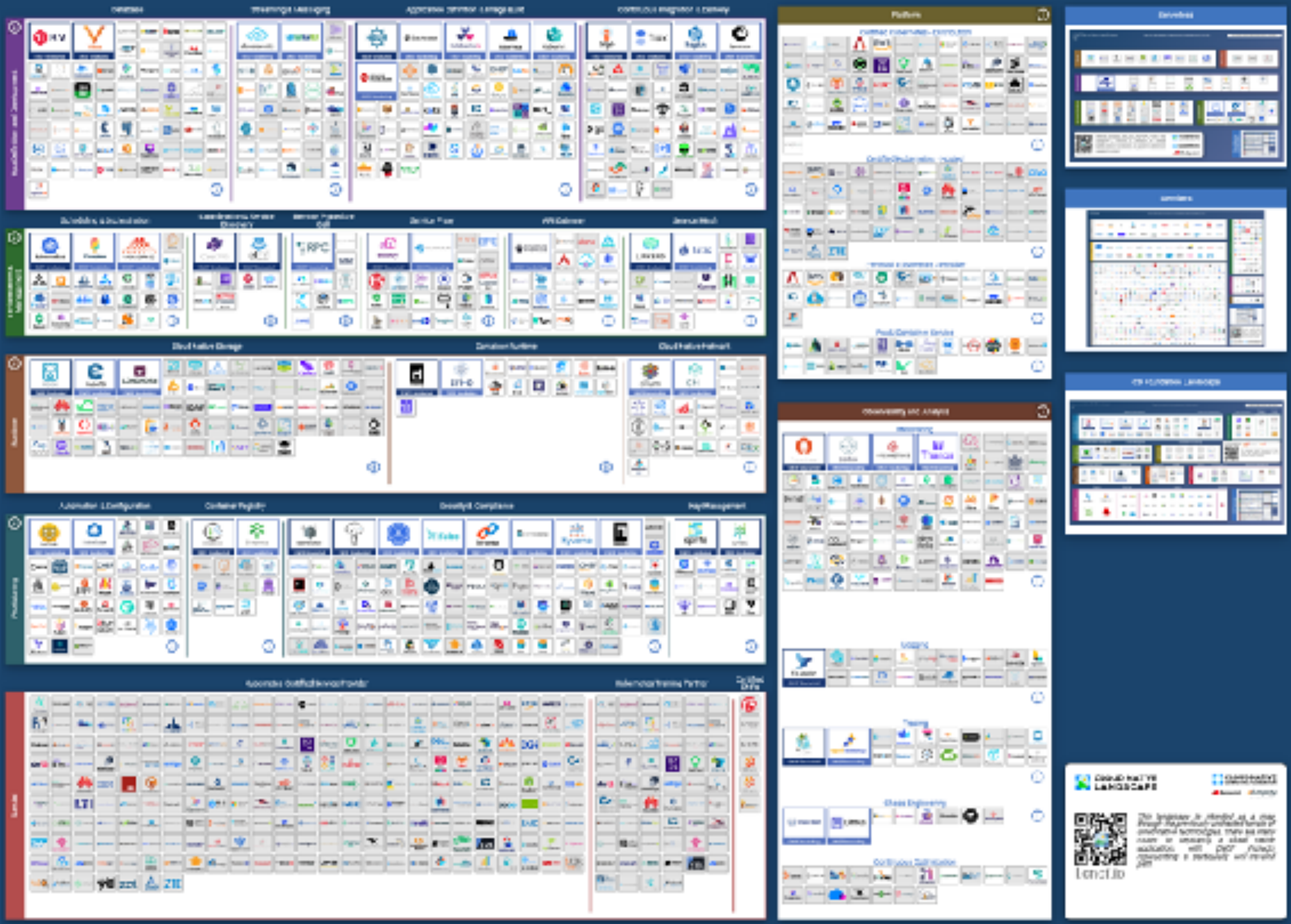
- ➢ builds features
- ➢ builds bug fixes
- ➢ delivers new software versions

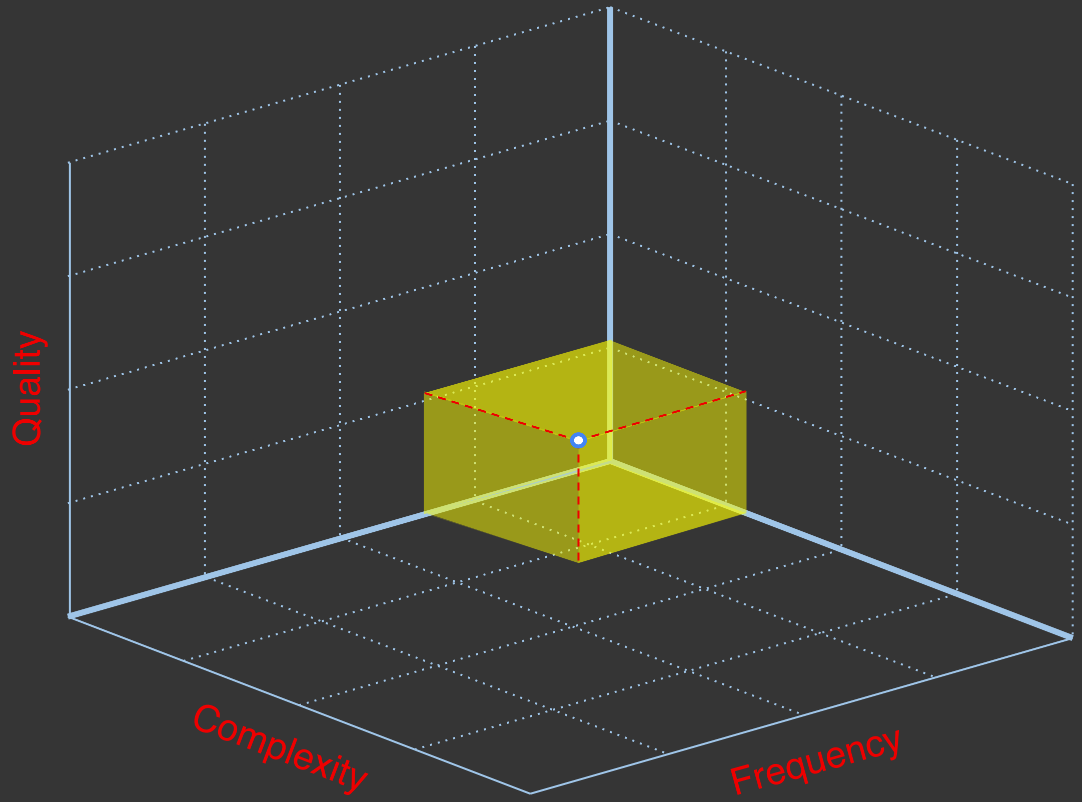
Best case

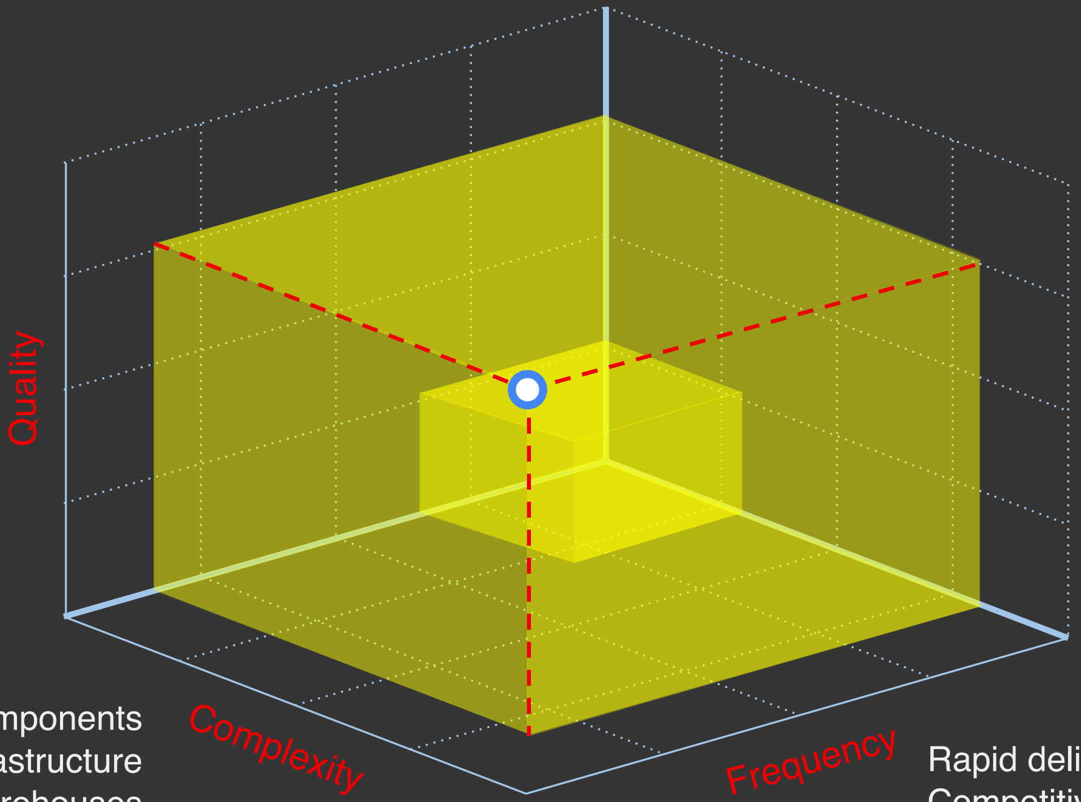
- ➢ automates pip
- ➢ autom

Responsibilities

- ➢ organise a ru
- ➢ app d


Database
Streaming & Messaging
Application Definition & Image Build
Continuous Integration & Delivery
App Definition and Development
Scheduling & Orchestration
Coordination & Service Discovery
Remote Procedure Call
Service Proxy
API Gateway
Service Mesh
Orchestration & Management
Cloud Native Storage
Container Runtime
Cloud Native Network
Runtime
Automation & Configuration
Container Registry
Security & Compliance
Key Management
Provisioning
Kubernetes Certified Service Provider
Kubernetes Training Partner
Special
Platform
Certified Kubernetes - Distribution
Certified Kubernetes - Hosted
Certified Kubernetes - Installer
PaaS/Container Service
Observability and Analysis
Monitoring
Logging
Tracing
Chaos Engineering
Continuous Optimization
Serverless
Members
CNCF Cloud Native Landscape
CLOUD NATIVE LANDSCAPE
l.cncf.io


Quality
Complexity
Frequency



New technologies
Regulations
Market conditions

System components
Cloud infrastructure
Data warehouses
IoT

Rapid delivery
Competitive edge
High standards

Agile

focuses on
processes

highlighting
change

while accelerating
delivery

CI/CD

focuses on
software-define life cycles

highlighting
tools

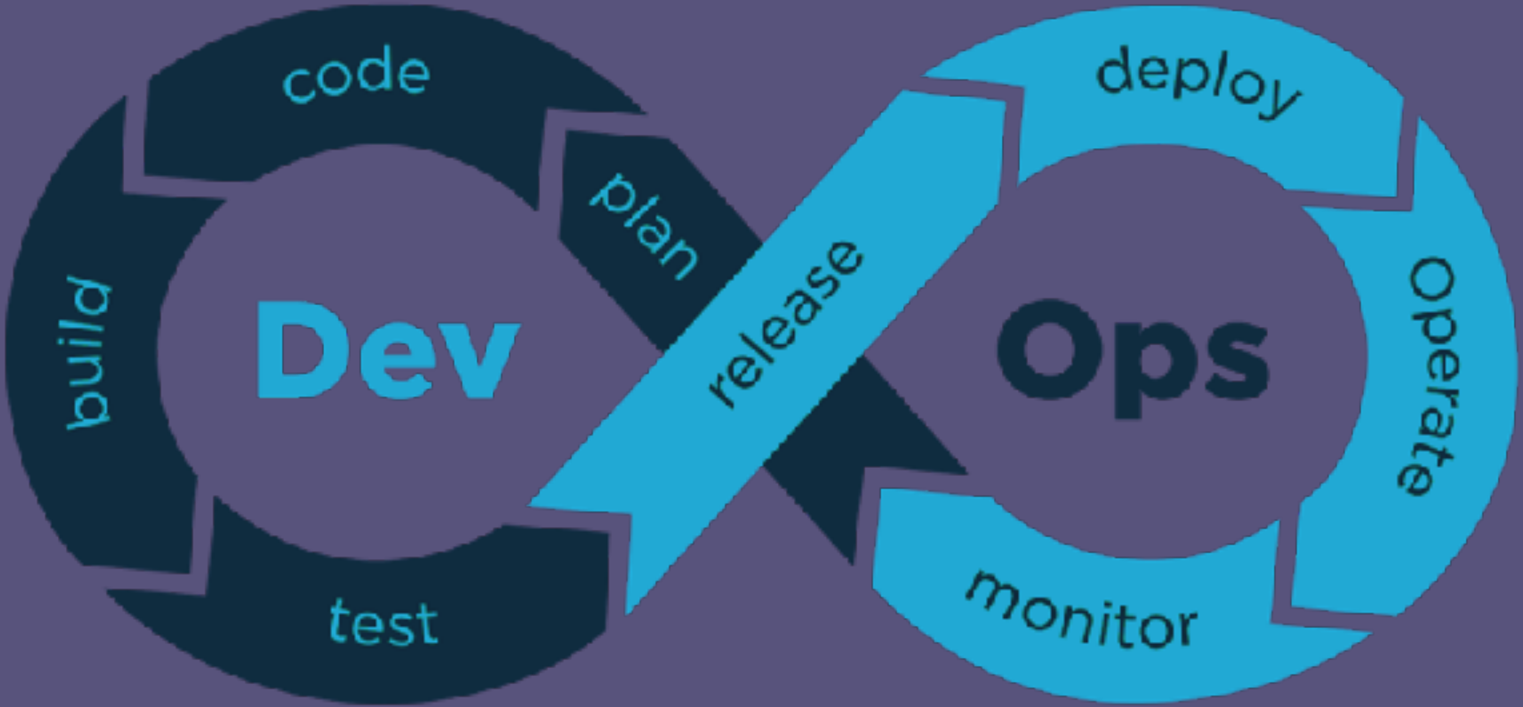
that emphasize
automation

DevOps

focuses on
culture

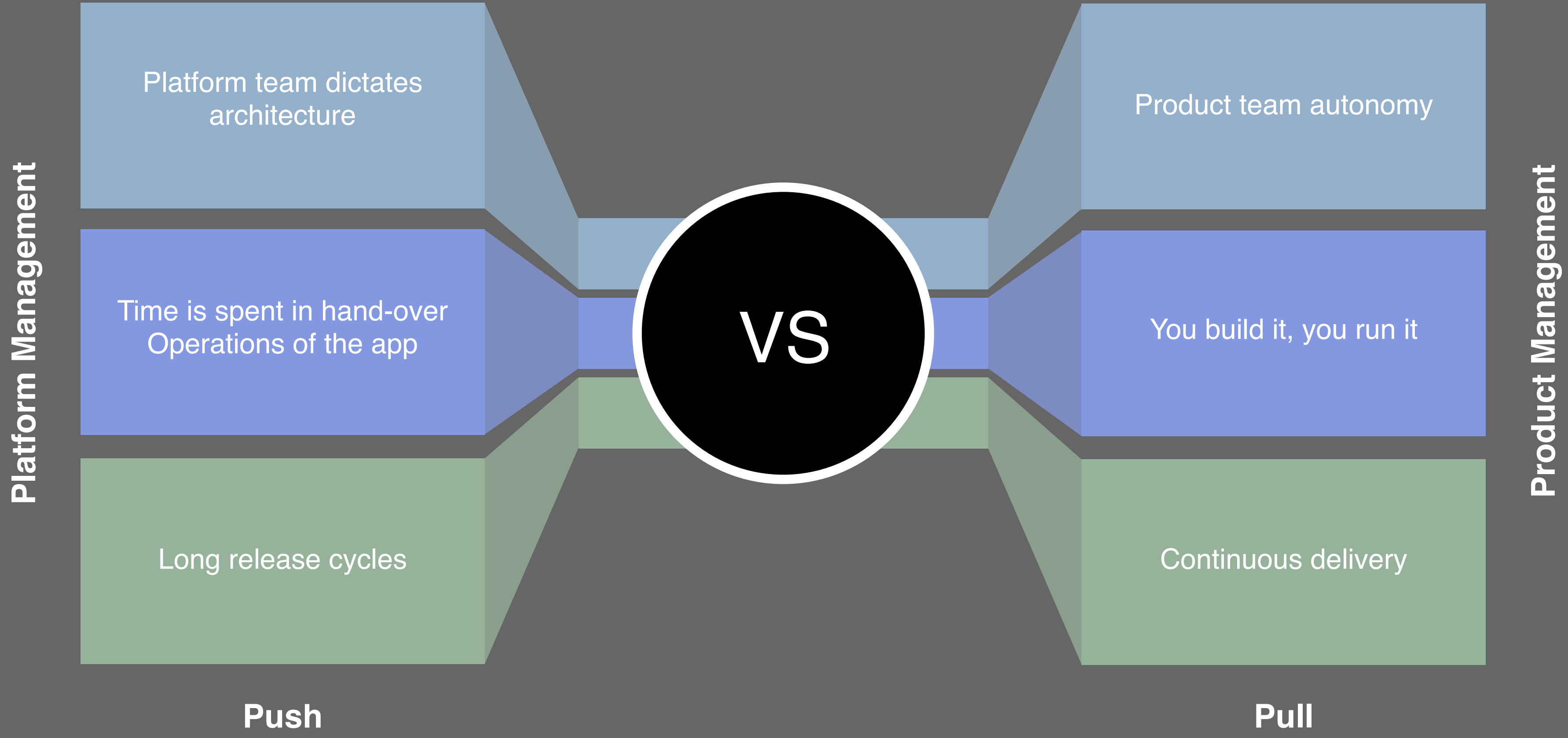
highlighting
roles

that emphasize
responsiveness


code
plan
build
Dev
test
release
deploy
Ops
Operate
monitor

REACHING GOALS

DevOps solves many of the team's problems. And there’s an inherent value of DevOps and CI/CD, which a container platform naturally entails



Platform Management

Platform team dictates architecture

Time is spent in hand-over Operations of the app

Long release cycles

Push

VS

Product team autonomy

You build it, you run it

Continuous delivery

Pull

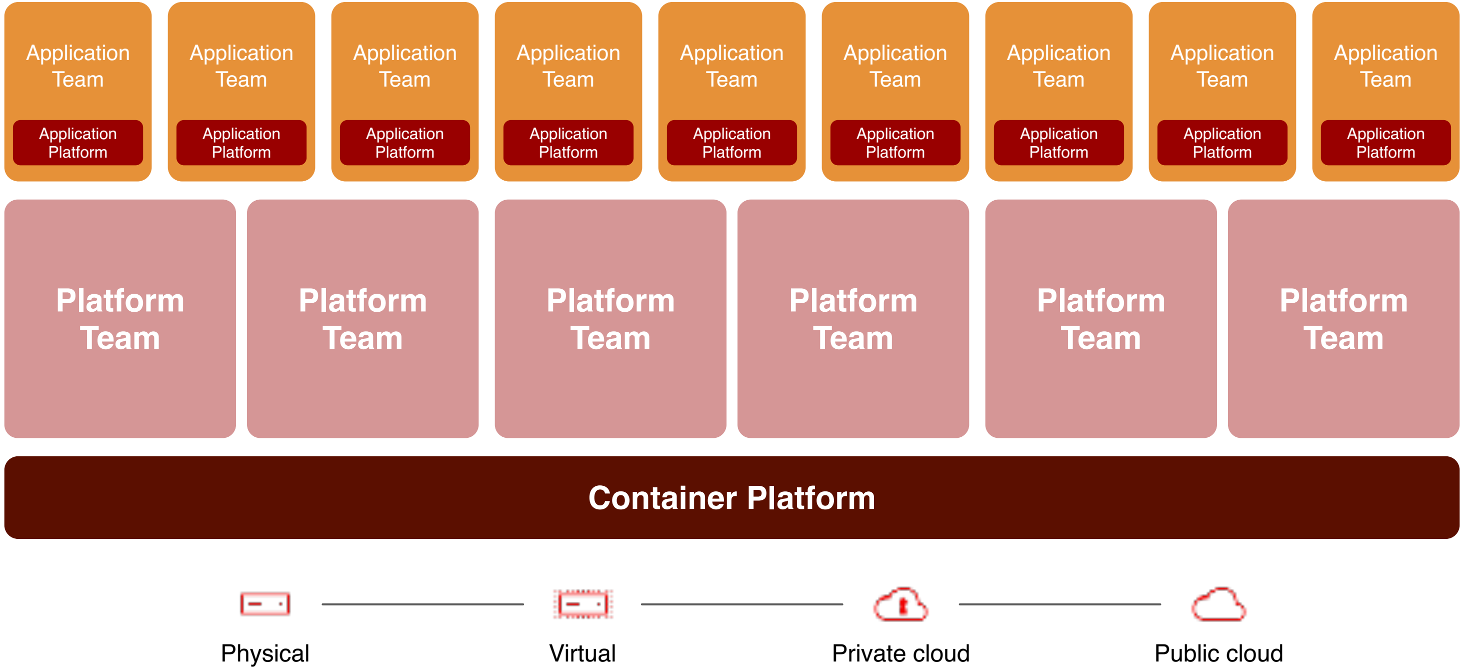
Product Management

I must do this for you,
please create a ticket

VS

Here, we’ve made it possible
for you to do it yourself

Application and Platform Model–Non Optimal Use



Application Team

Application Platform

Application Team

Application Platform

Application Team

Application Platform

Application Team

Application Platform

Application Team

Application Platform

Application Team

Application Platform

Application Team

Application Platform

Application Team

Application Platform

Application Team

Application Platform

Platform Team

Platform Team

Platform Team

Platform Team

Platform Team

Platform Team

Container Platform

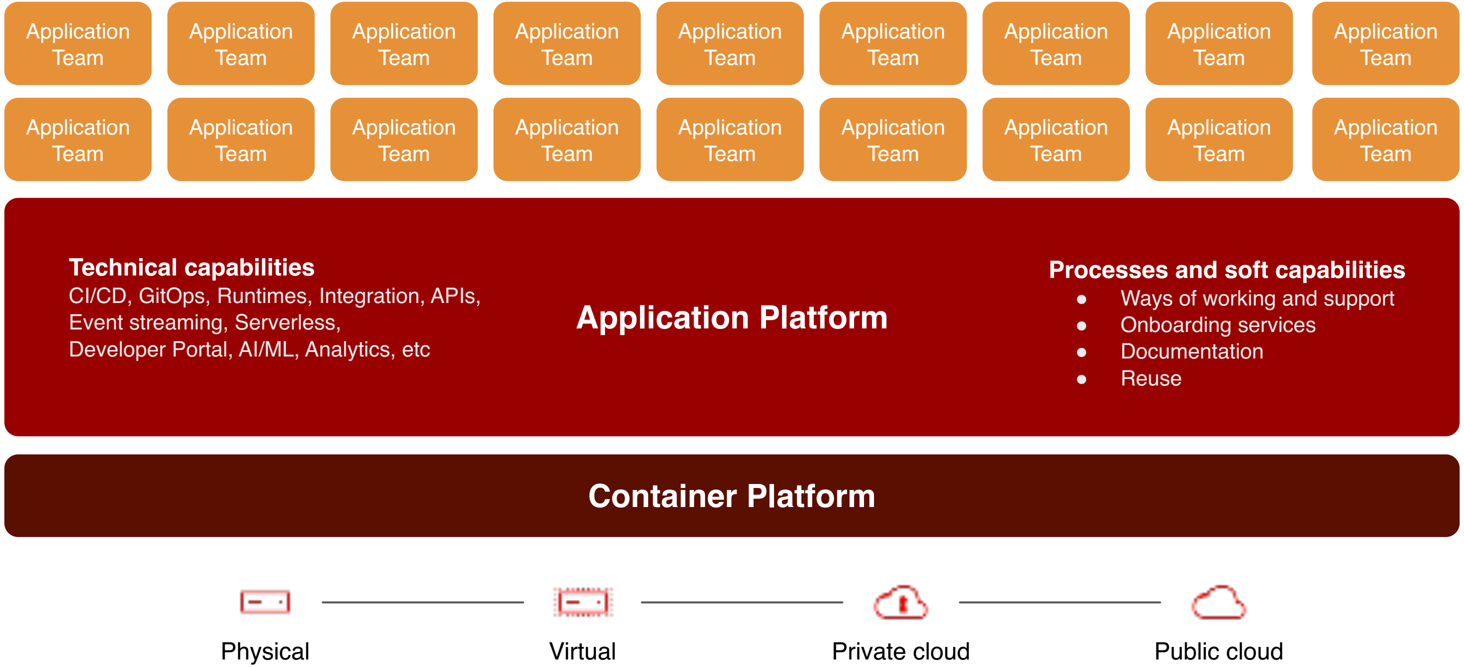
Physical

Virtual

Private cloud

Public cloud

Application and Platform Model



Application Team	Application Team	Application Team	Application Team	Application Team	Application Team	Application Team	Application Team	Application Team
Application Team	Application Team	Application Team	Application Team	Application Team	Application Team	Application Team	Application Team	Application Team

Technical capabilities
CI/CD, GitOps, Runtimes, Integration, APIs,
Event streaming, Serverless,
Developer Portal, AI/ML, Analytics, etc

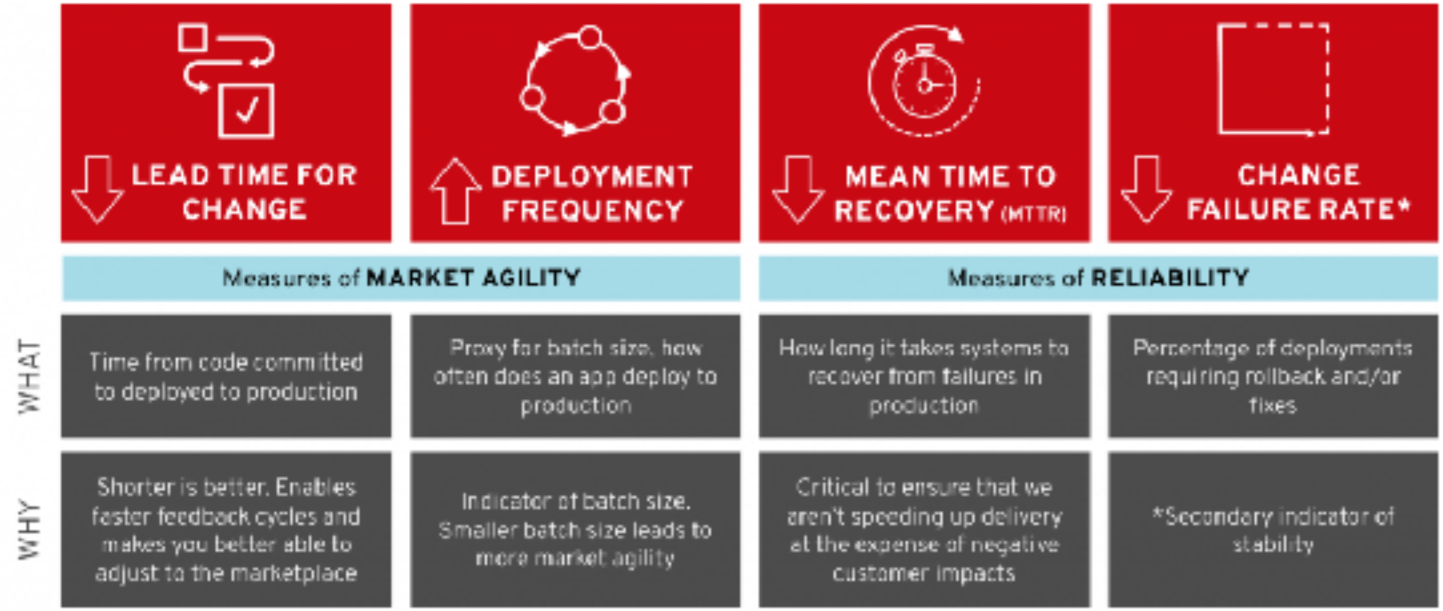
Application Platform

Processes and soft capabilities

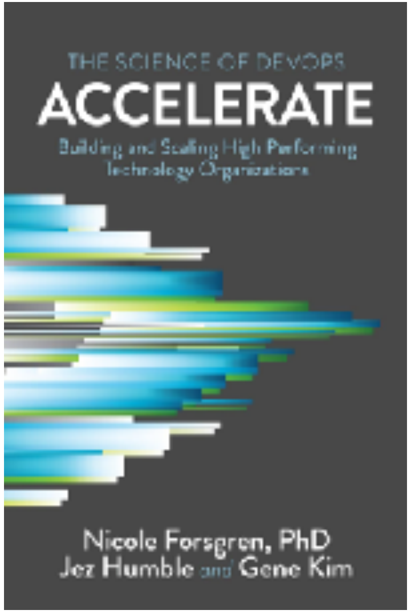
- Ways of working and support
- Onboarding services
- Documentation
- Reuse

Container Platform

A Metrics-Driven Approach to Transformation



	⇩ LEAD TIME FOR CHANGE	⇧ DEPLOYMENT FREQUENCY	⇩ MEAN TIME TO RECOVERY (MTTR)	⇩ CHANGE FAILURE RATE*
	Measures of **MARKET AGILITY**		Measures of **RELIABILITY**	
WHAT	Time from code committed to deployed to production	Proxy for batch size, how often does an app deploy to production	How long it takes systems to recover from failures in production	Percentage of deployments requiring rollback and/or fixes
WHY	Shorter is better. Enables faster feedback cycles and makes you better able to adjust to the marketplace	Indicator of batch size. Smaller batch size leads to more market agility	Critical to ensure that we aren't speeding up delivery at the expense of negative customer impacts	*Secondary indicator of stability

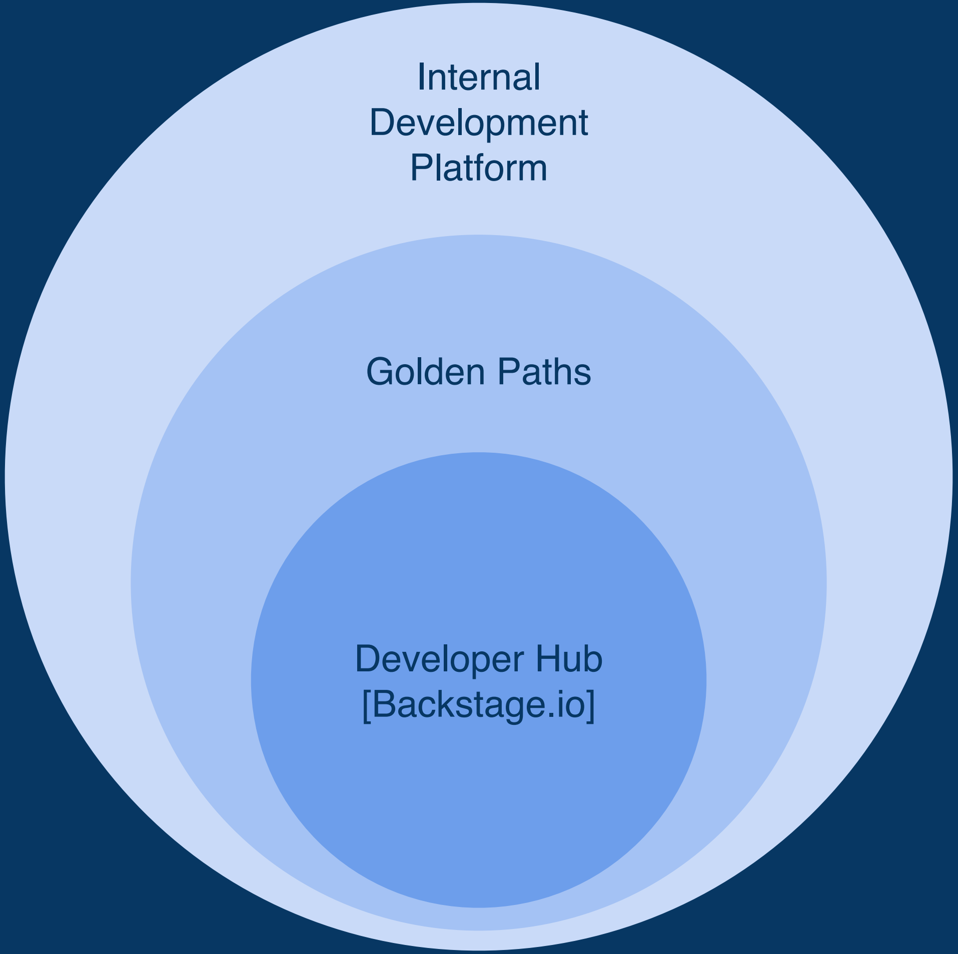


THE SCIENCE OF DEVOPS

ACCELERATE

Building and Scaling High Performing Technology Organizations

Nicole Forsgren, PhD
Jez Humble and Gene Kim



Provides a central place for teams to manage software development tools and processes, promoting collaboration and streamlining workflows.

Standard practices and processes for software development to improve quality and reduce errors.

A unified and open portal streamlining development, empowering teams to create high-quality apps. Built on Backstage, offering curated tools, self-service, security, and governance for productivity and simplicity.

IDP **for** Software Development

Let developers focus on what they do best *)

Standards

By defining best practices and standards for each step, IDP can help reduce manual work and errors that may arise due to lack of standardization.

Automation

By automating software development, testing, and deployment with tools such as Continuous Integration (CI) and Continuous Delivery (CD), IDP enables fast and continuous software delivery.

Processes

By standardizing practices and processes across the organization, IDP can create a common understanding and language around software development, improving collaboration and communication among teams.

*) Leading to much less activity in #aaargh Slack channel

Improve IDP

Promote collaboration, increased flexibility, and improving quality and speed in software delivery

Agile

Using best practices and procedures such as continuous integration, automated testing, and continuous delivery, teams can create a streamlined workflow that reduces the risk of errors and delays.

DevOps

Practices such as infrastructure automation, configuration management, and monitoring and logging are used to create a continuous delivery pipeline that is fast, reliable, and scalable.

Ways of Working

IDP is a flexible framework that supports a culture of continuous improvement and collaboration. Teams can constantly improve their process and adapt it to new requirements and changing circumstances.

IDP as a Product

Plan and manage known needs and wants, set goals and requirements, identify functional areas, and define sprint iterations.

Design a solution that considers project goals and requirements, define architecture, design interfaces, and create a prototype for testing.

Develop the solutions based on the design, using best practices to ensure scalability and maintainability.

Test the solution to ensure it works as intended and meets the goals and requirements, including tests.

Deploy the solution with the IDP and ensure it properly supports the needs.

Operate and monitor to ensure the solution continues to supports the organizational needs.

Identify the organization's specific needs and goals

Define the different steps in the process

Create standardized templates

Automate the process

Continuously update and improve the process.

Maximizing Your Success

The Importance of Onboarding, Support, and Training for Team Members

Onboarding

IDP must include onboarding support for new team members, including clear instructions on tools and processes, as well as training on how to use the platform effectively.

Support

Support for teams is important in IDP, including technical support, coaching, and mentoring to optimize the use of the platform. Feedback on performance and results can also help identify areas for improvement.

Training

By providing training and support to teams, the IDP can become more efficient and effective, leading to increased productivity and organizational success.

Red Hat Developer Hub

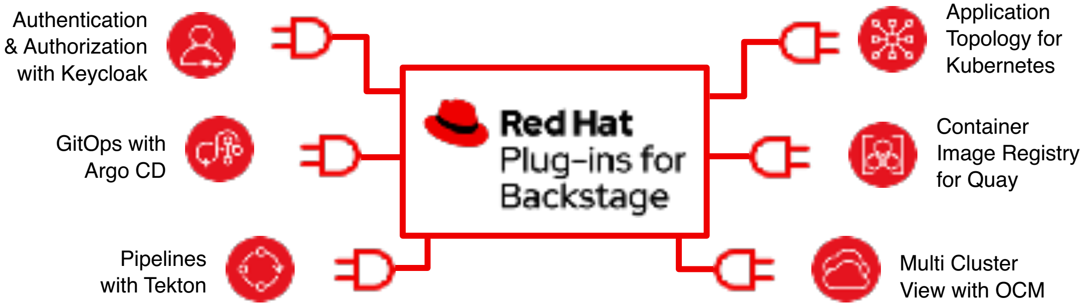
Navigating complexity to deliver business value

Streamline application and developer onboarding with Golden Paths

Increase developer productivity and product velocity, while lowering the cognitive load on developers

Simplify governance of technology choices with developer self-service

Enable technology agility and avoid vendor lock-in with solutions built on Open


Authentication & Authorization with Keycloak
GitOps with Argo CD
Pipelines with Tekton
Red Hat Plug-ins for Backstage
Application Topology for Kubernetes
Container Image Registry for Quay
Multi Cluster View with OCM

Product management together with IDP ensures a cohesive product vision and prioritizes requirements and features based on business needs and customer demands.

IDP can be used to implement agile principles and processes advocated by SAFe within the organization, providing a standardized approach to software development and delivery.

By using IDP as part of the SAFe framework, the organization can standardize its software process while implementing and following the agile principles necessary for SAFe success.

Culture

Leadership