Microsoft

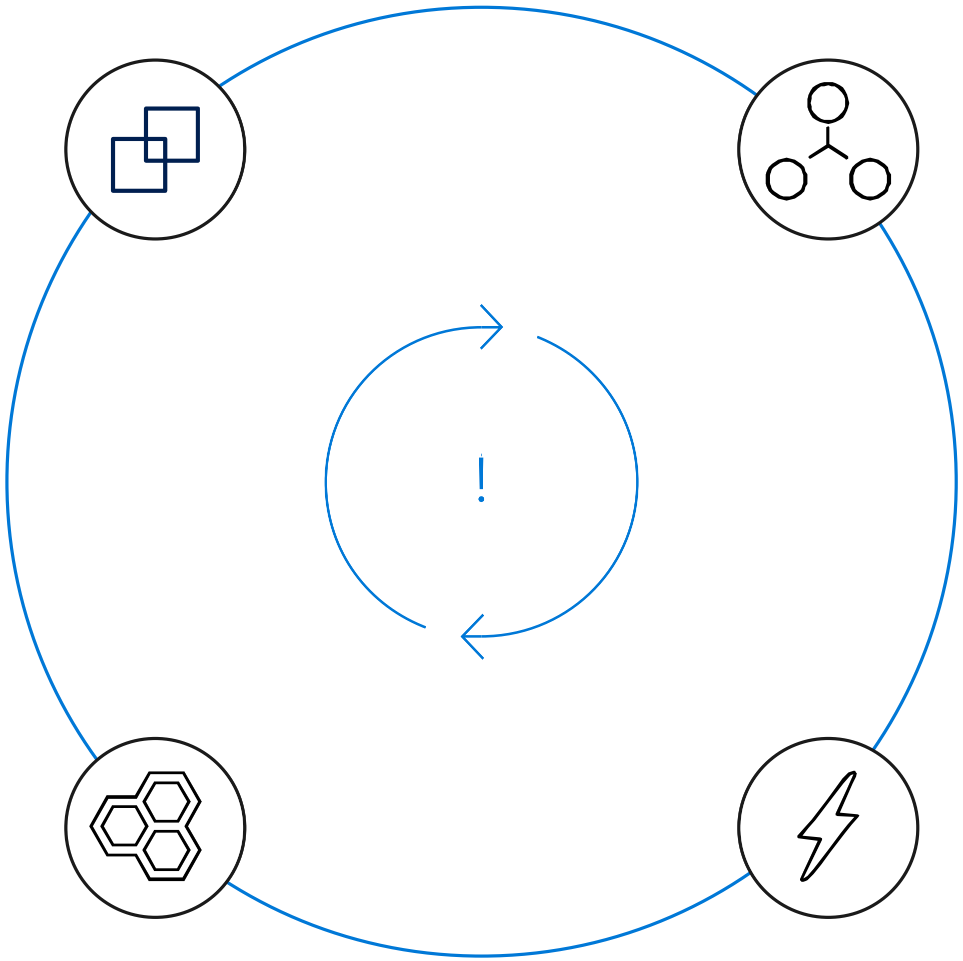
Azure DevOps

Andreas Mock

Agenda

- Theorie
  - What is DevOps?
  - DevOps Building blocks
  - DevOps on Azure
- Praxis / HandsOn
  - Let’s Hack ☺

Common software delivery challenges



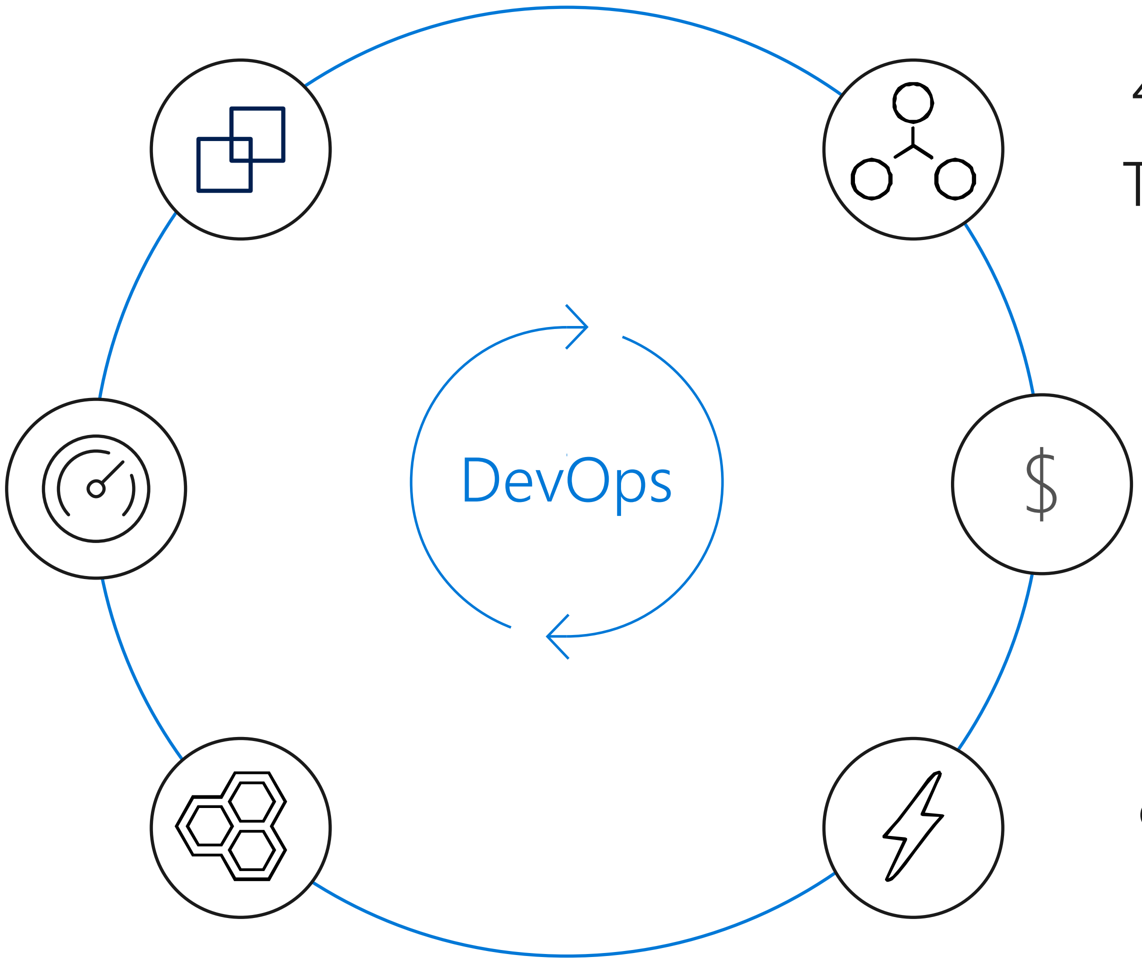
Low Deployment Frequency

High Lead Time For Changes

High Change Failure Rate

Long Mean Time to Recover

High Performance DevOps Companies Achieve...



46x Deployment Frequency

440x Faster Lead Time For Changes

20% Faster Time to Market

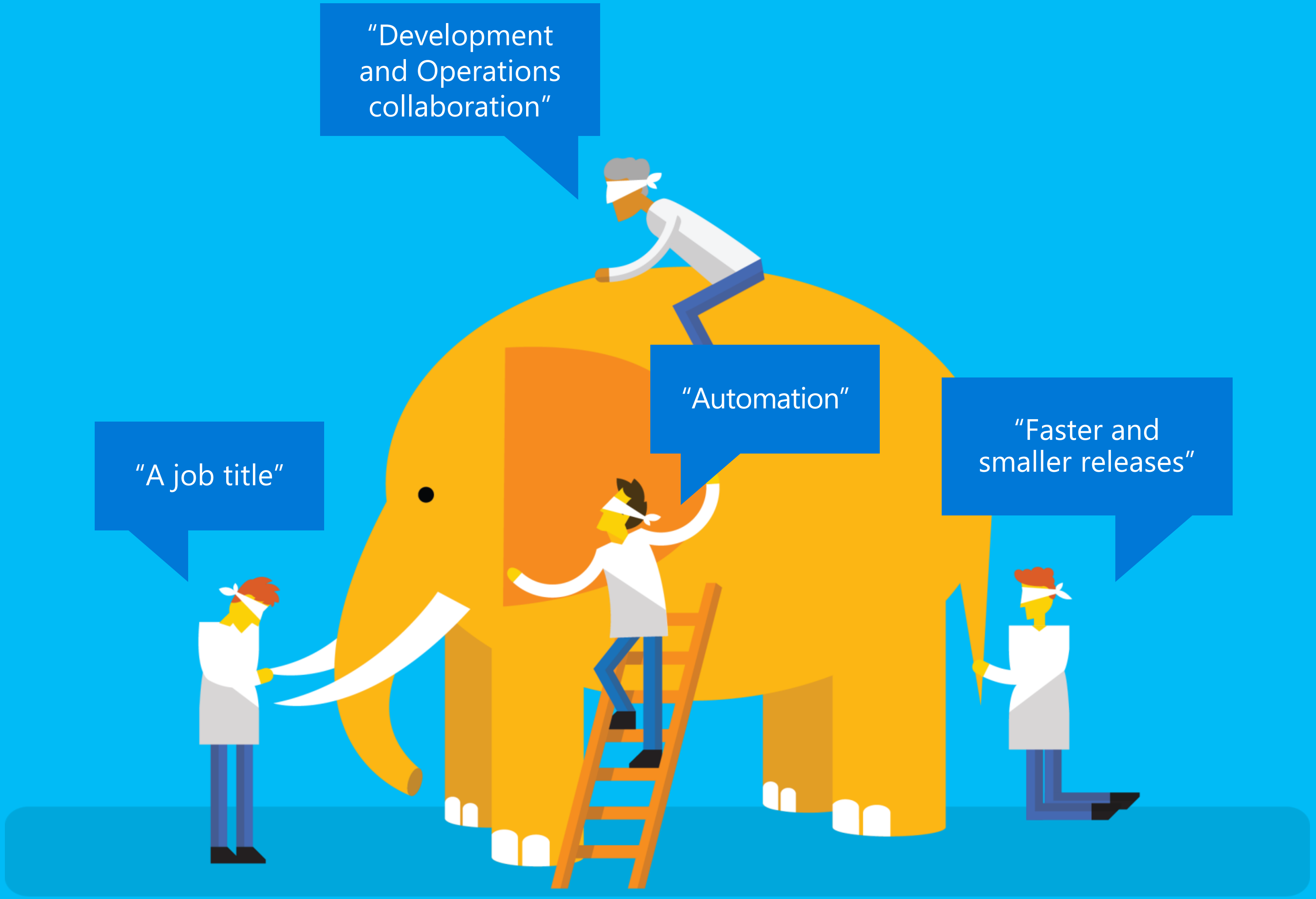
20% Increase Revenue

5x Lower Change Failure Rate

96x Faster Mean Time to Recover

Source: 2017 State of DevOps Report: Presented by Puppet and DORA

What is DevOps?


"Development and Operations collaboration"
"A job title"
"Automation"
"Faster and smaller releases"

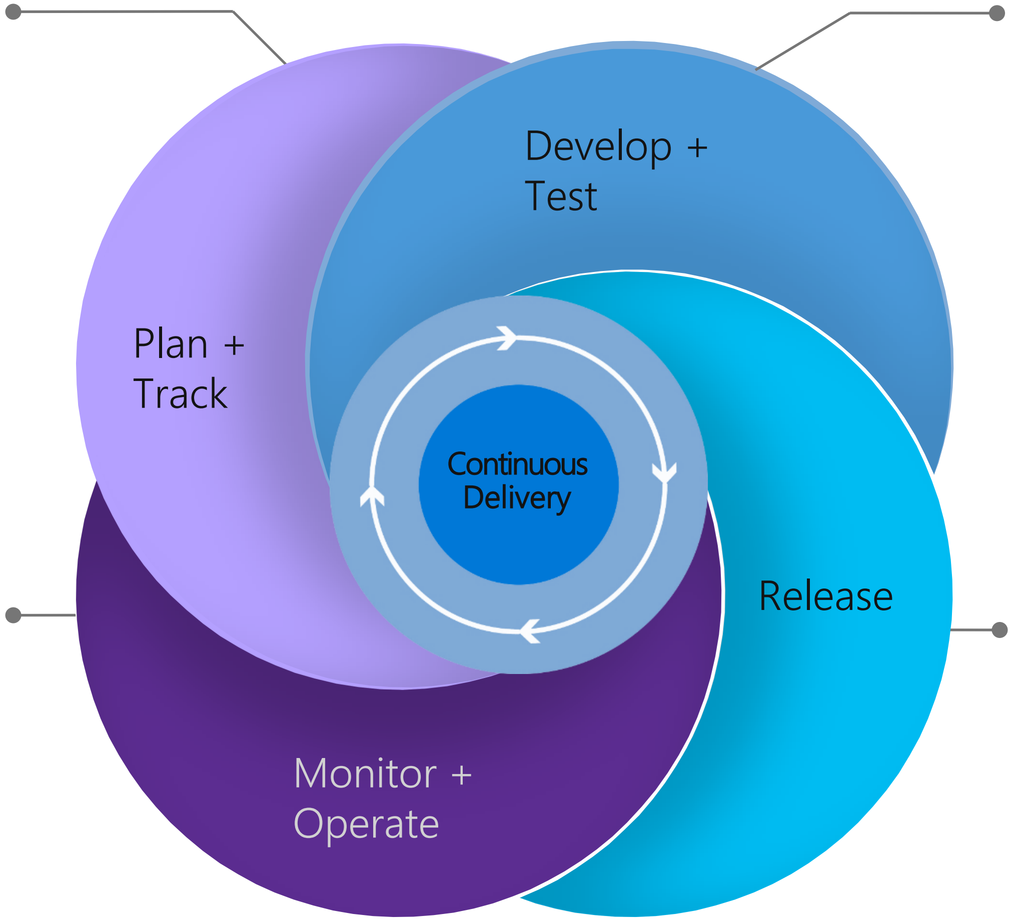
What is DevOps?

People. Process. Technology.

DevOps is the union of people, process, and technology to <u>enable continuous delivery of value to your end users</u>.

Plan & Track

Agile Planning
Dashboards
Kanban Boards
Taskboards



Build & Test

Git Source Control
Modern Code Workflow
Continuous Integration
Continuous Testing
Package Management
Open Source Compliance

Monitoring & Operations

Telemetry
Diagnostics
Analysis
Compliance

Deployment

Deployment of app and infrastructure
PaaS, IaaS and Containers

Deliver faster and more reliably.

Simple

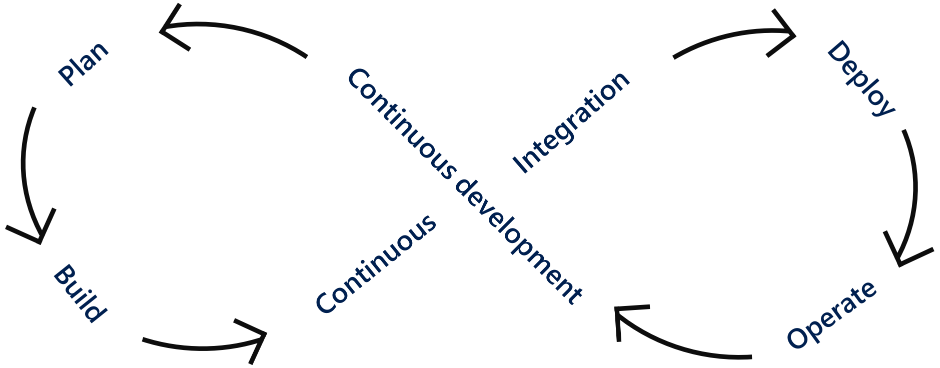
End-to-end DevOps Solutions

Reliable

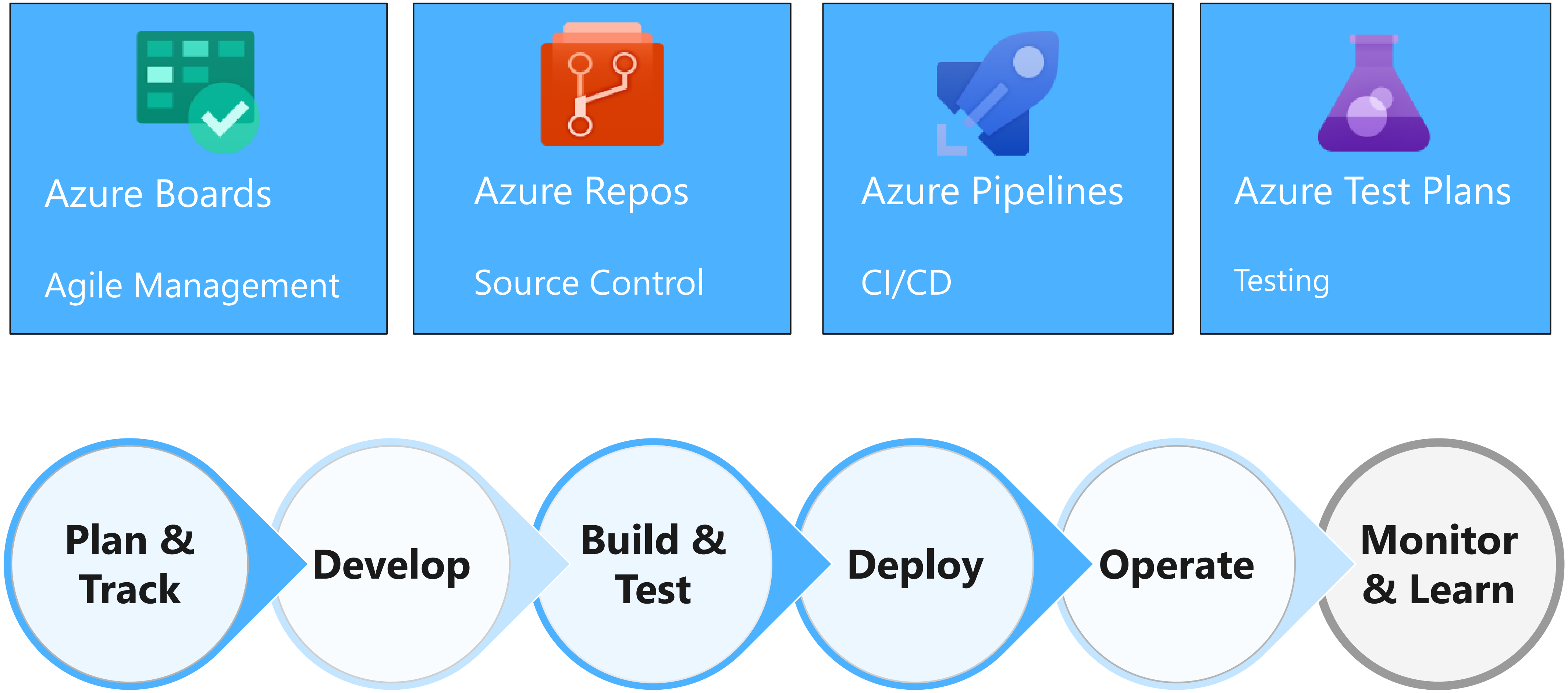
Enterprise-grade platforms & services

Open

Use the tools & languages you know



Azure DevOps



Azure Boards

Agile Management

Azure Repos

Source Control

Azure Pipelines

CI/CD

Azure Test Plans

Testing

Plan & Track

Develop

Build & Test

Deploy

Operate

Monitor & Learn

Azure Boards

Agile Management

Introduce an Agile Projectmanagement

Introduce a well defined agile process like Scrum, or

define your own agile process and visualize and control the value flow through Kanban.

But always use the basic ingredients of an agile Process

- Product vision statement
- Product Roadmap
- Release Plan
- Product Backlog
- Sprint Backlog
- Increment

Planing: Minimum Viable Product

A minimum viable product has just those core features sufficient to deploy the product, and no more. Developers typically deploy the product to a subset of possible customers - such as early adopters thought to be more forgiving, more likely to give feedback, and able to grasp a product vision from an early prototype or marketing information.

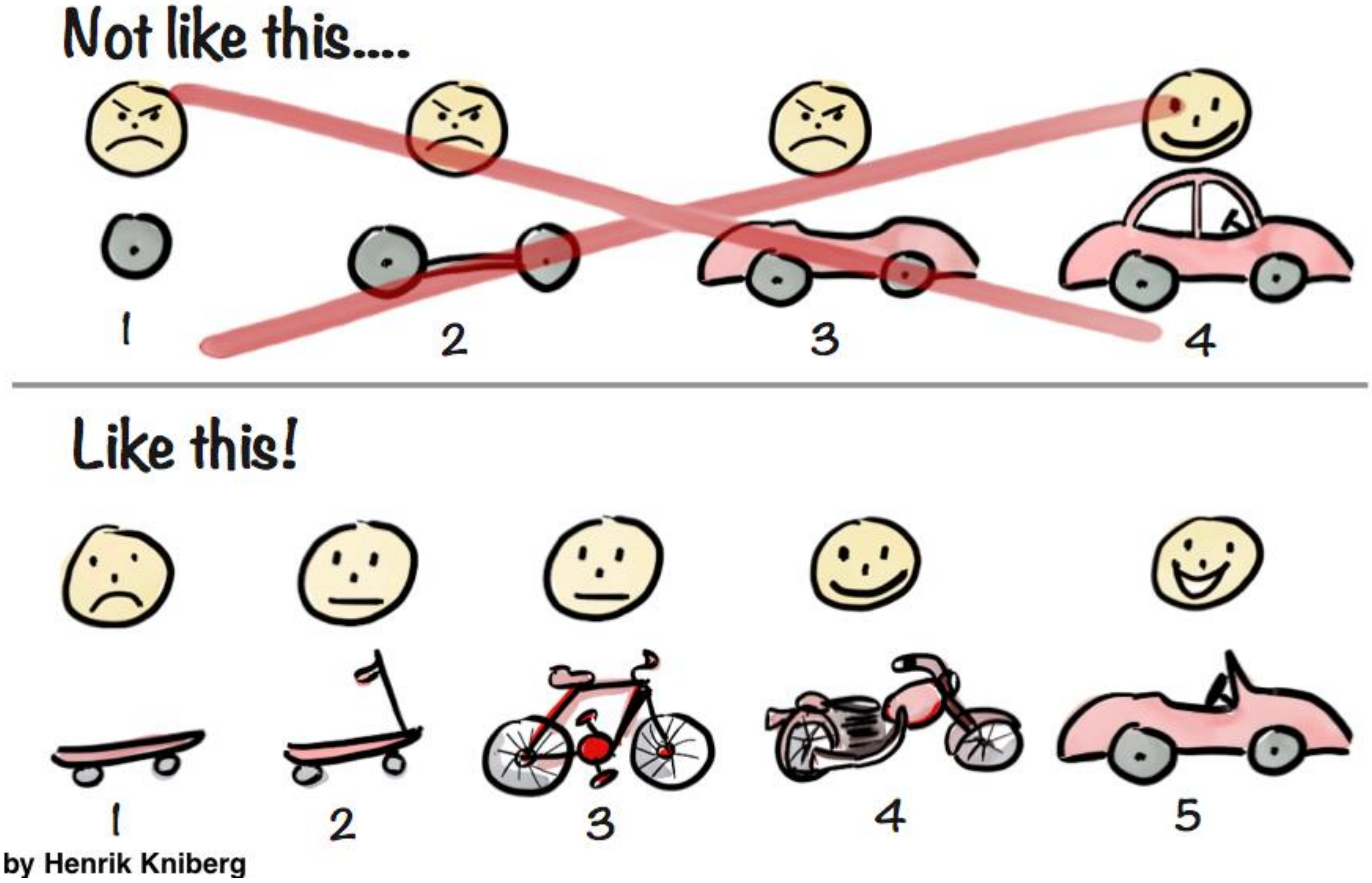
Steve Blank typically refers to minimum viable product as minimum feature set.

Purpose:

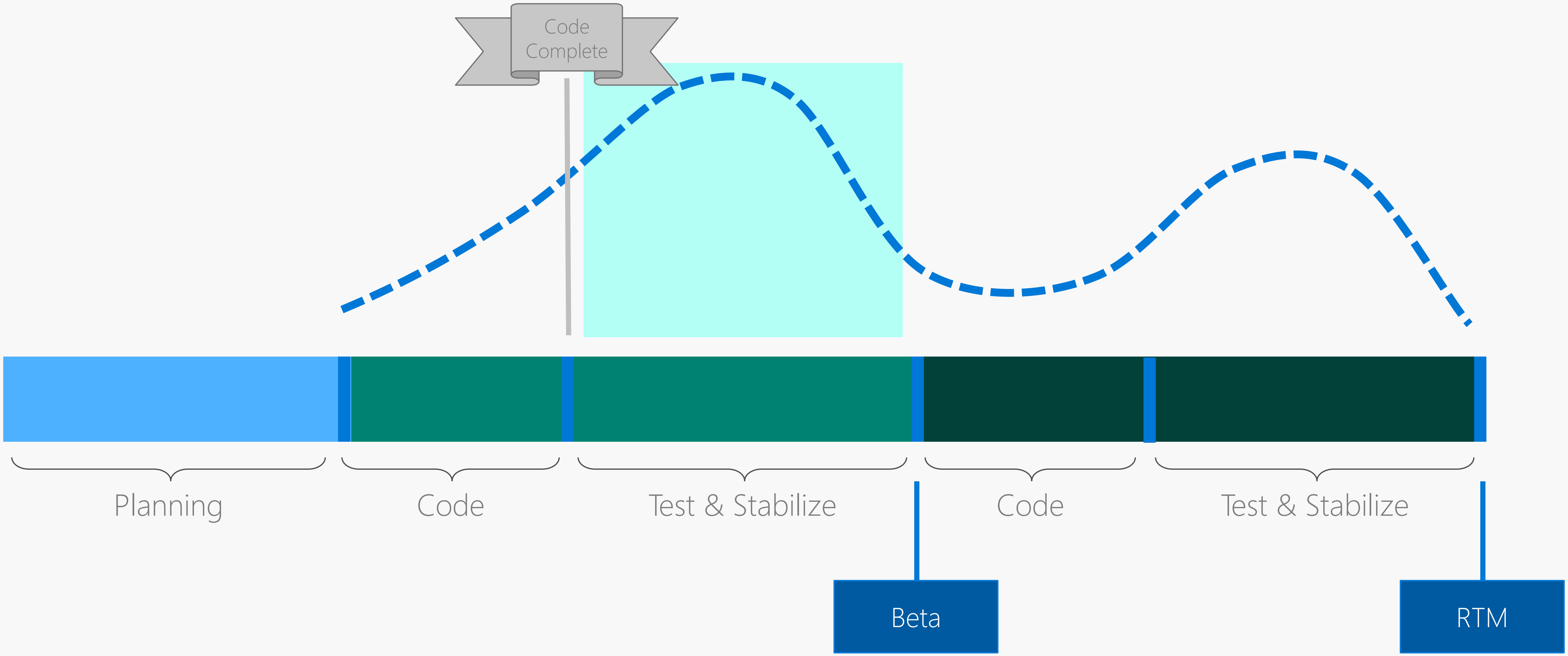
- Be able to test a product hypothesis with minimal resources
- Accelerate learning
- Reduce wasted engineering hours
- Get the product to early customers as soon as possible

https://en.wikipedia.org/wiki/Minimum_viable_product

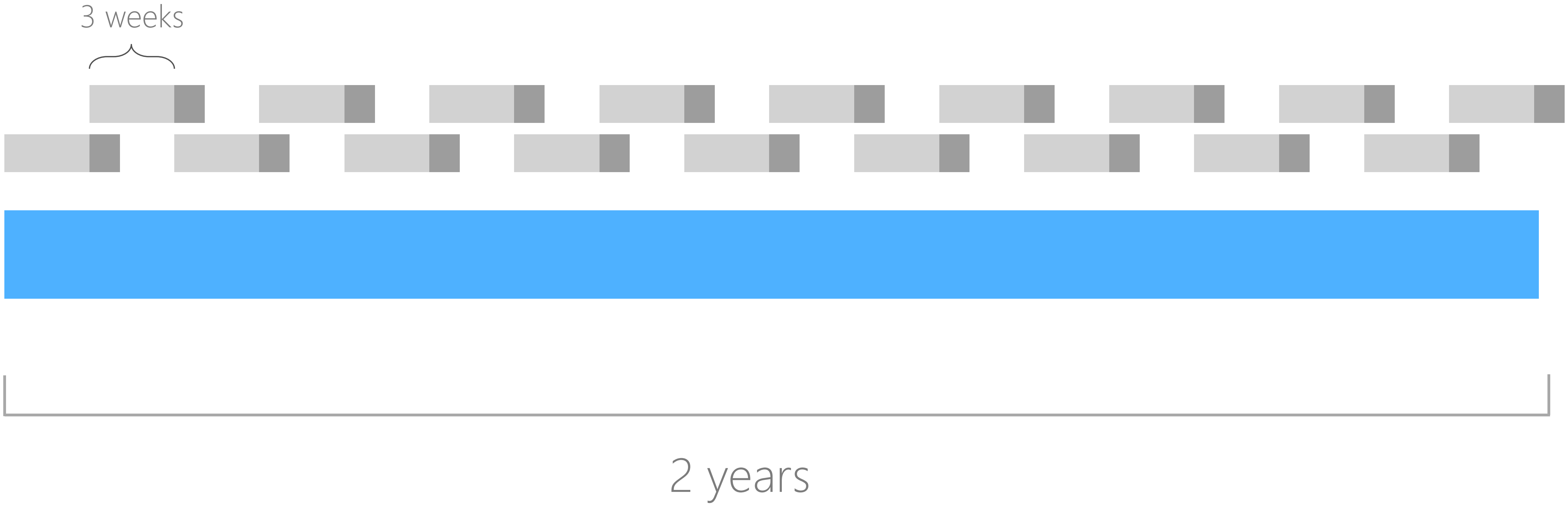
Planing: Minimum Viable Product



The way before Agile

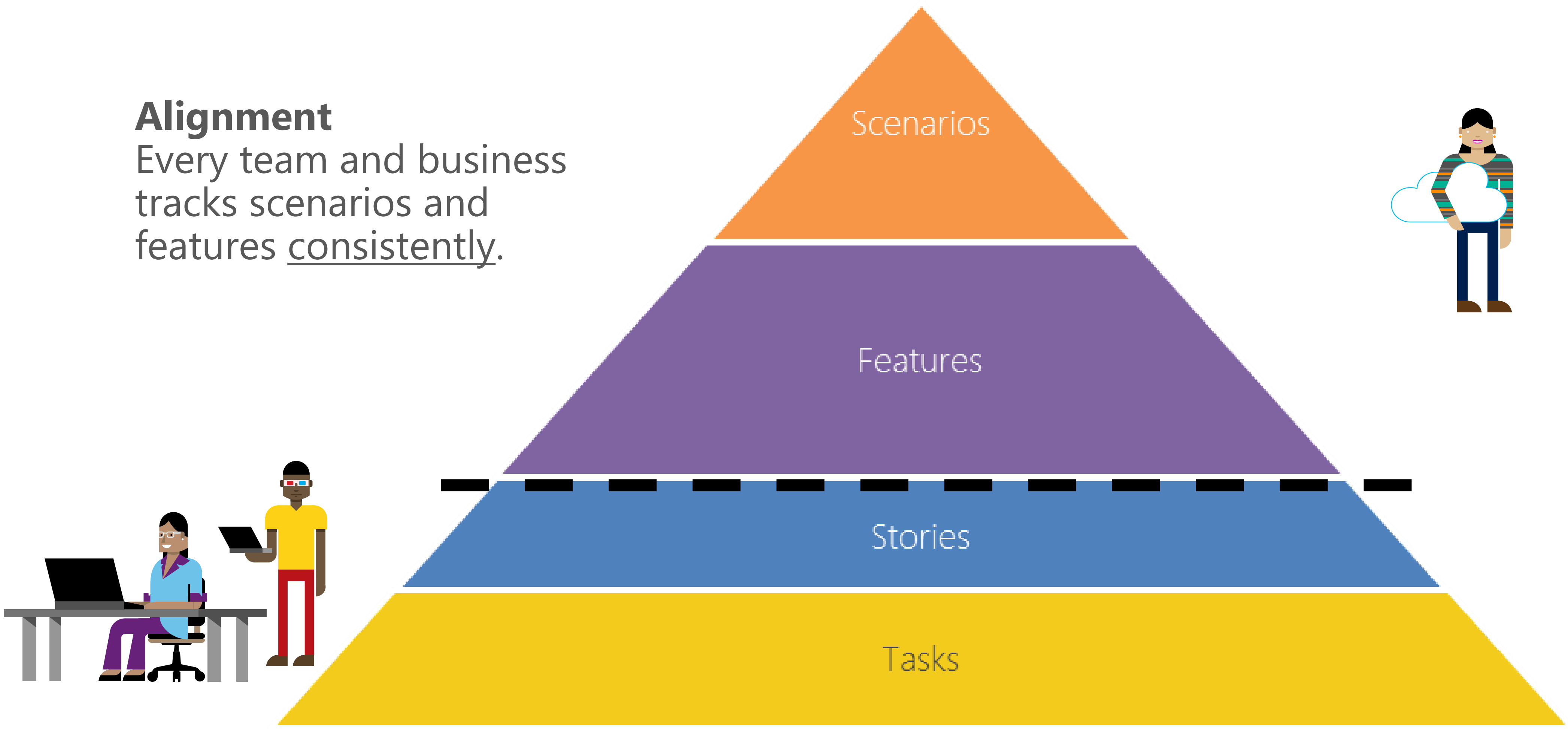


Now



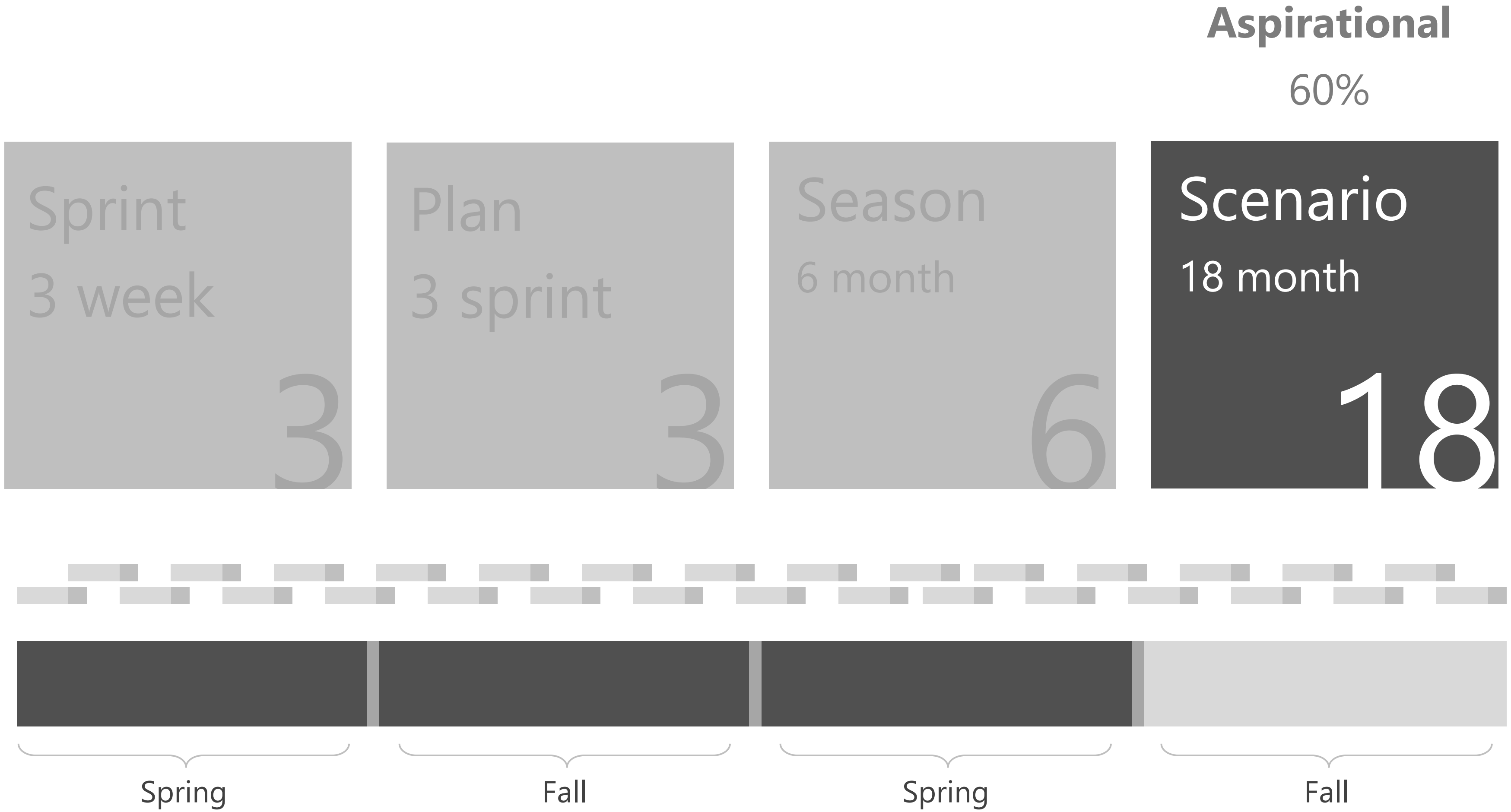
Planning: Staying Aligned

Alignment
Every team and business tracks scenarios and features <u>consistently</u>.



Team (Dev, Ops, QS)
Team <u>chooses</u> how to manage stories and/or tasks

Staying Aligned



Aspirational

60%

Sprint

3 week

3

Plan

3 sprint

3

Season

6 month

6

Scenario

18 month

18

Spring

Fall

Spring

Fall

Staying Aligned

Hopeful

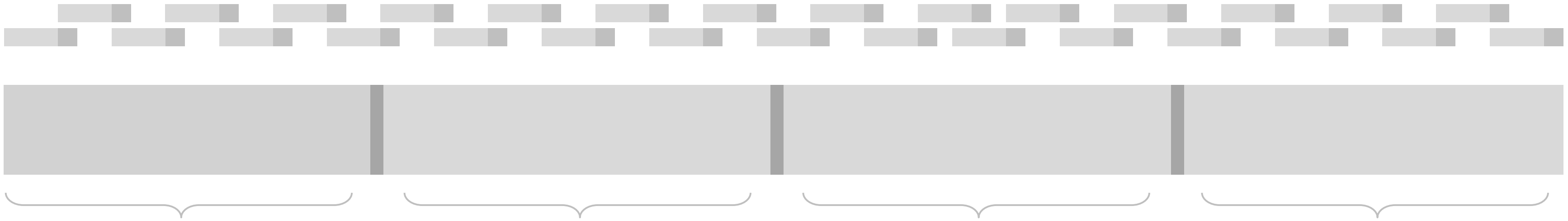
80%

Sprint
3 week
3

Plan
3 sprint
3

Season
6 month
6

Scenario
18 month
18



Staying Aligned

Thoughtful

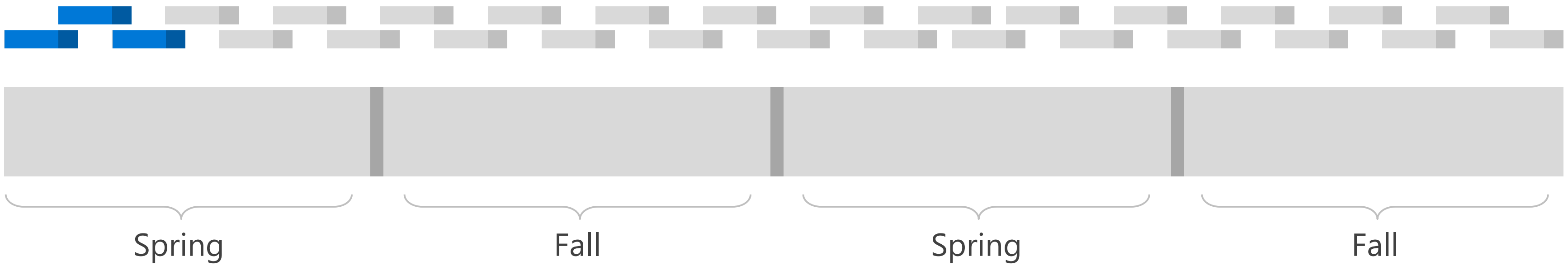
90%

Sprint
3 week
3

Plan
3 sprint
3

Season
6 month
6

Scenario
18 month
18



Staying Aligned

Confident

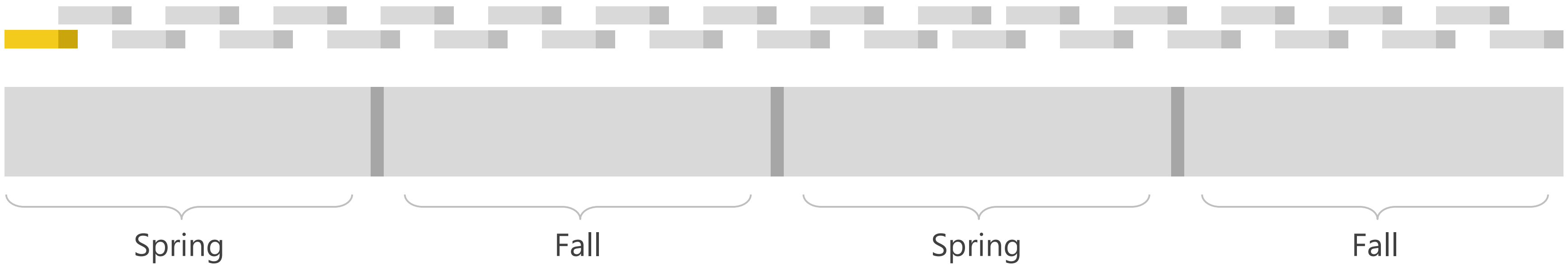
95%

Sprint
3 week
3

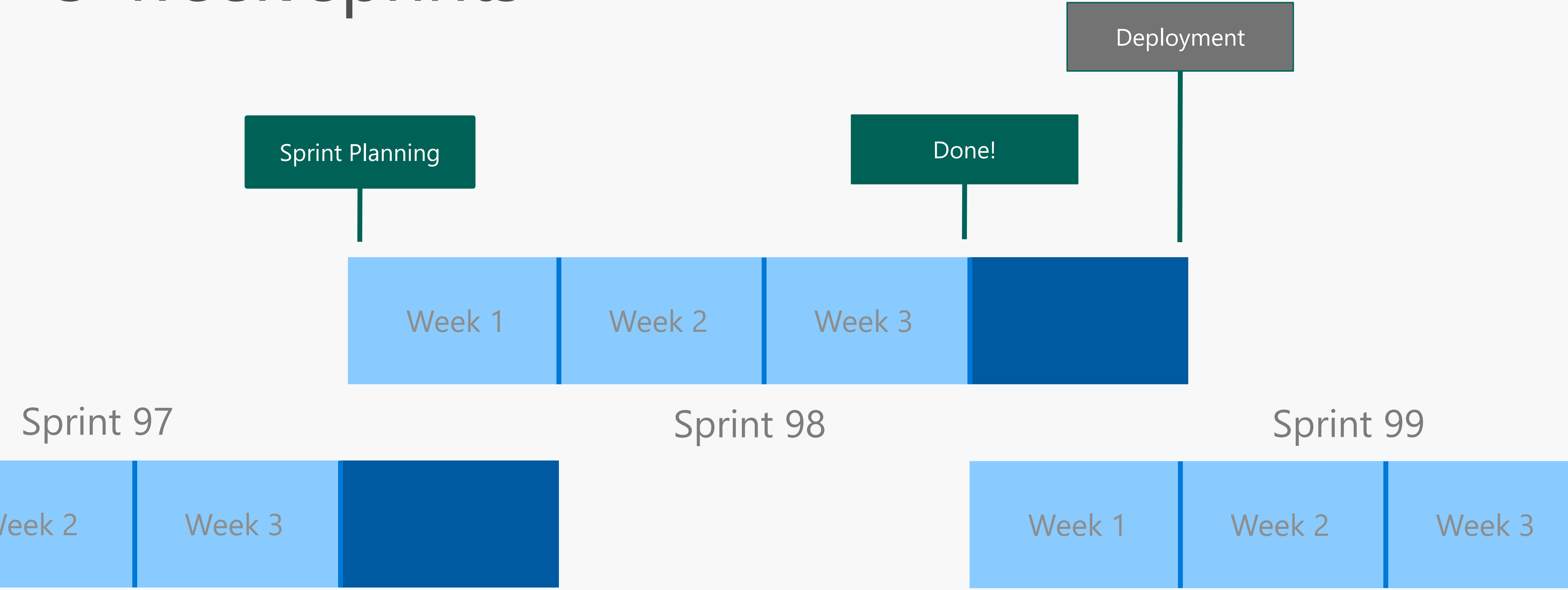
Plan
3 sprint
3

Season
6 month
6

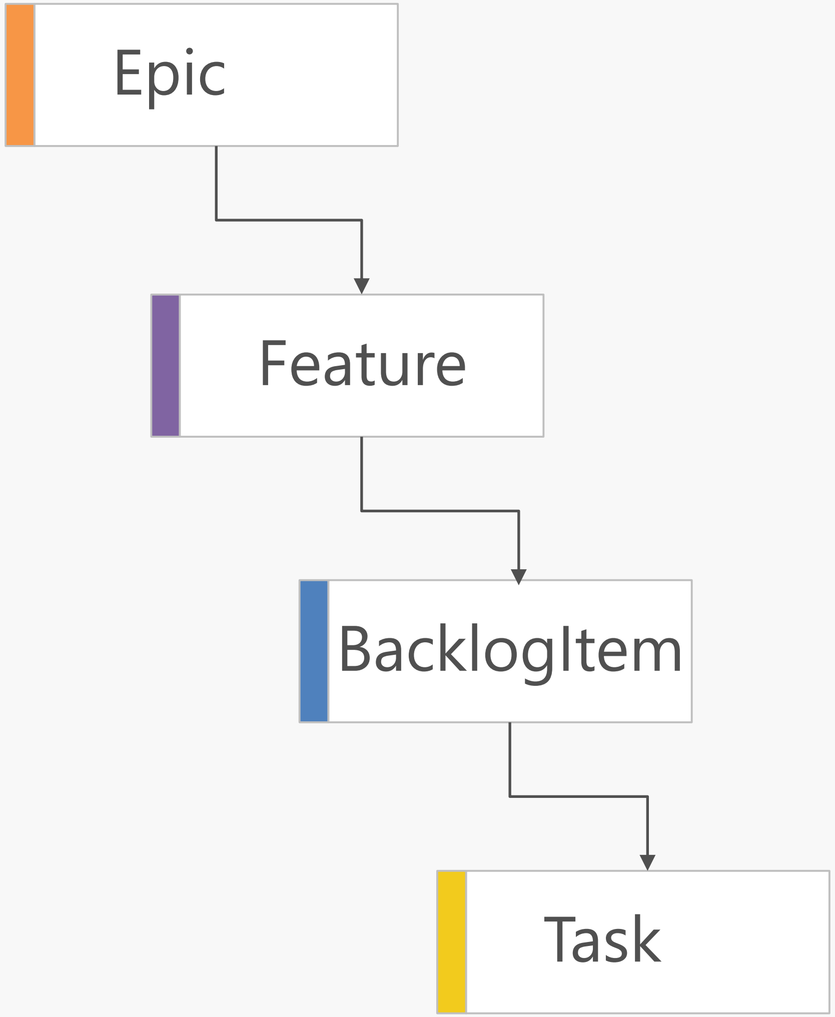
Scenario
18 month
18



3-week sprints



VSTS Workitems

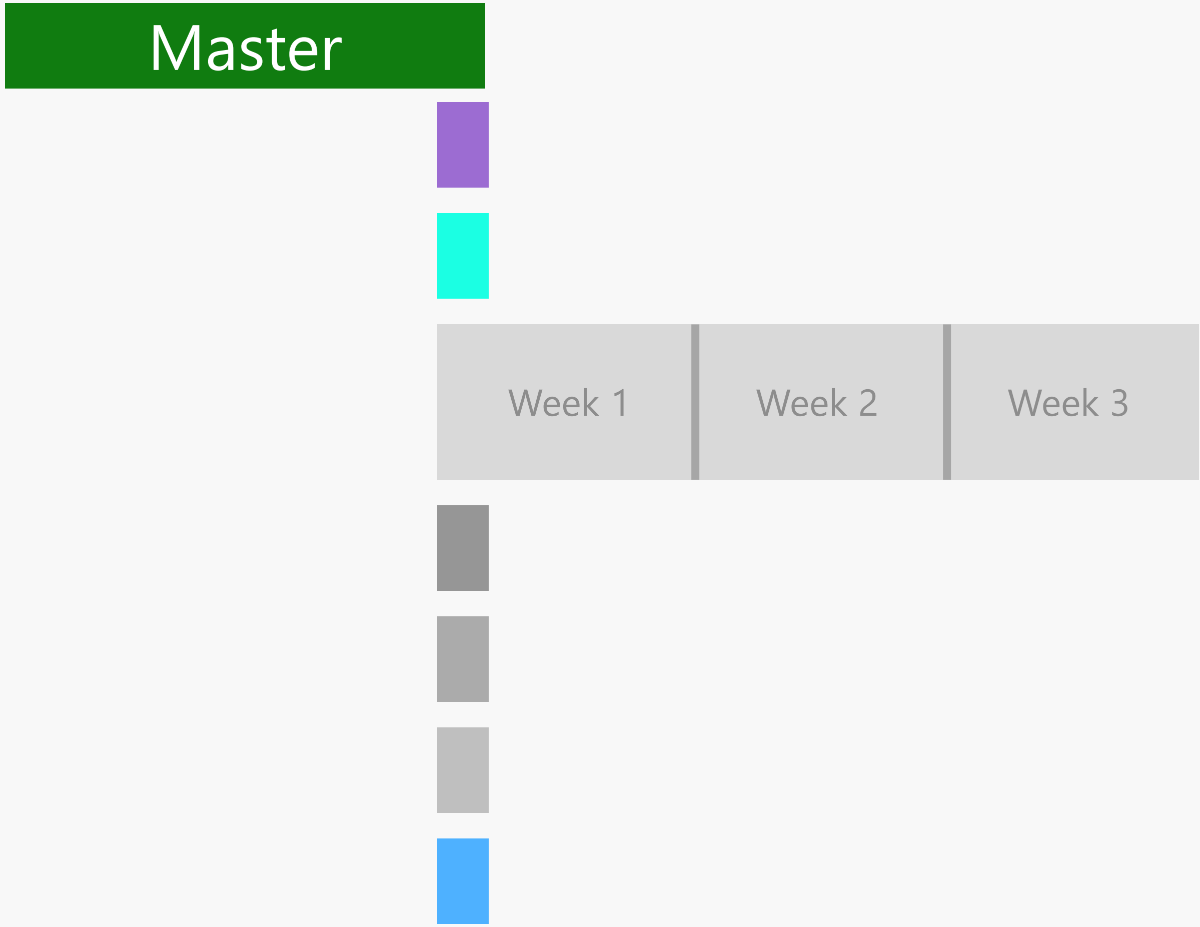


		Iteration Path
Epic	Used to define your Scenarios	*Scenario1*
Feature	Used to define your Seasons	*Scenario1/Season1*
BacklogItem	Used to define your Sprint	*Scenario1/Season1/Sprint1*
Task	Used to define your work	*Scenario1/Season1/Sprint1*

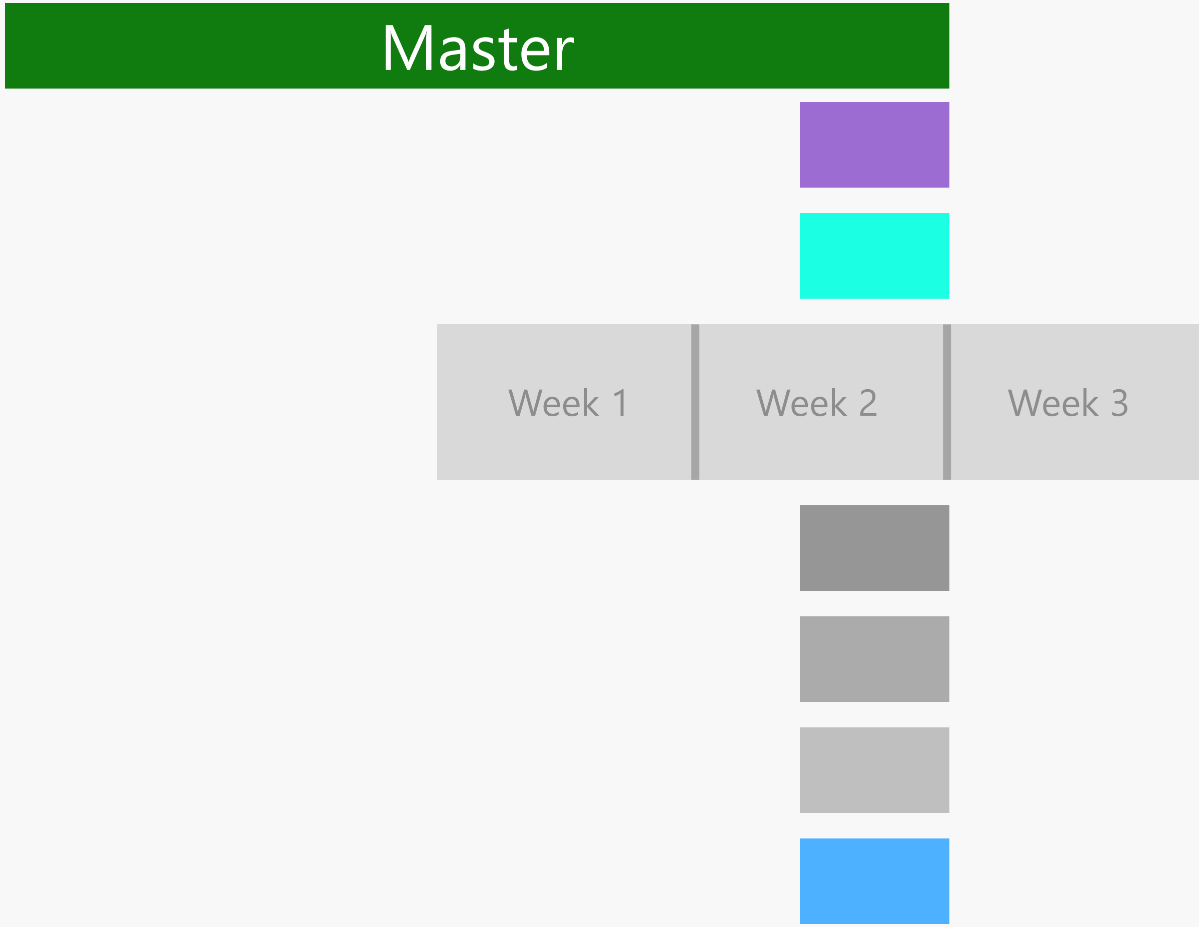
Azure Repos

Source Control

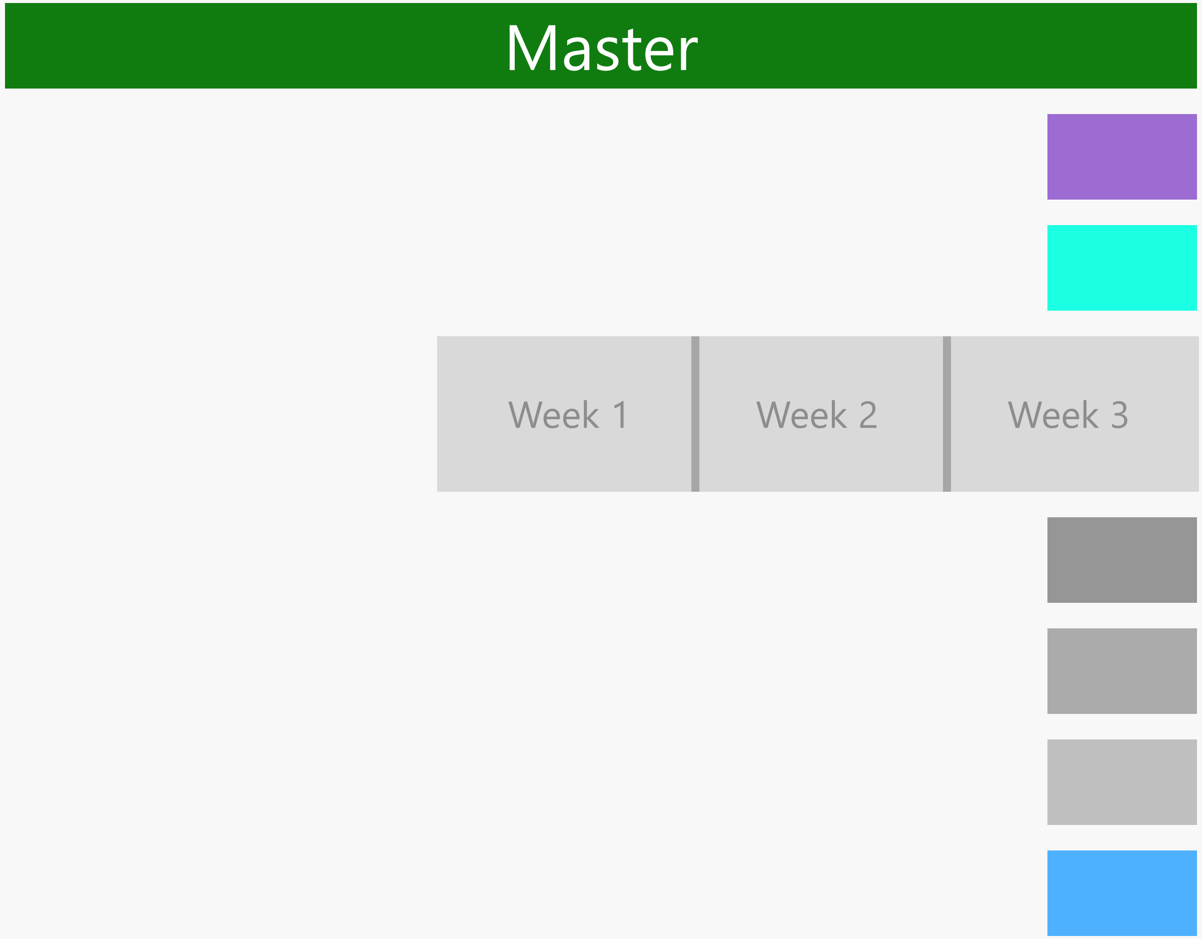
Everyone creates a branch…



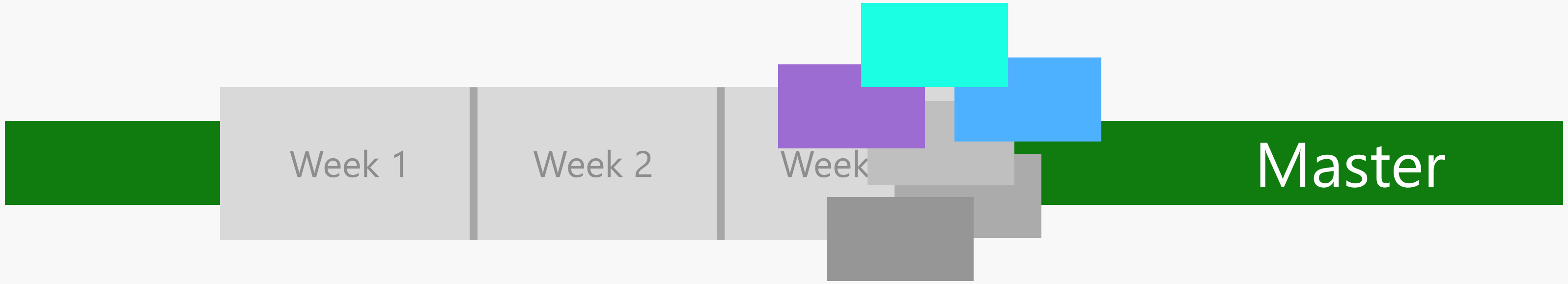
Writes a lot of code...



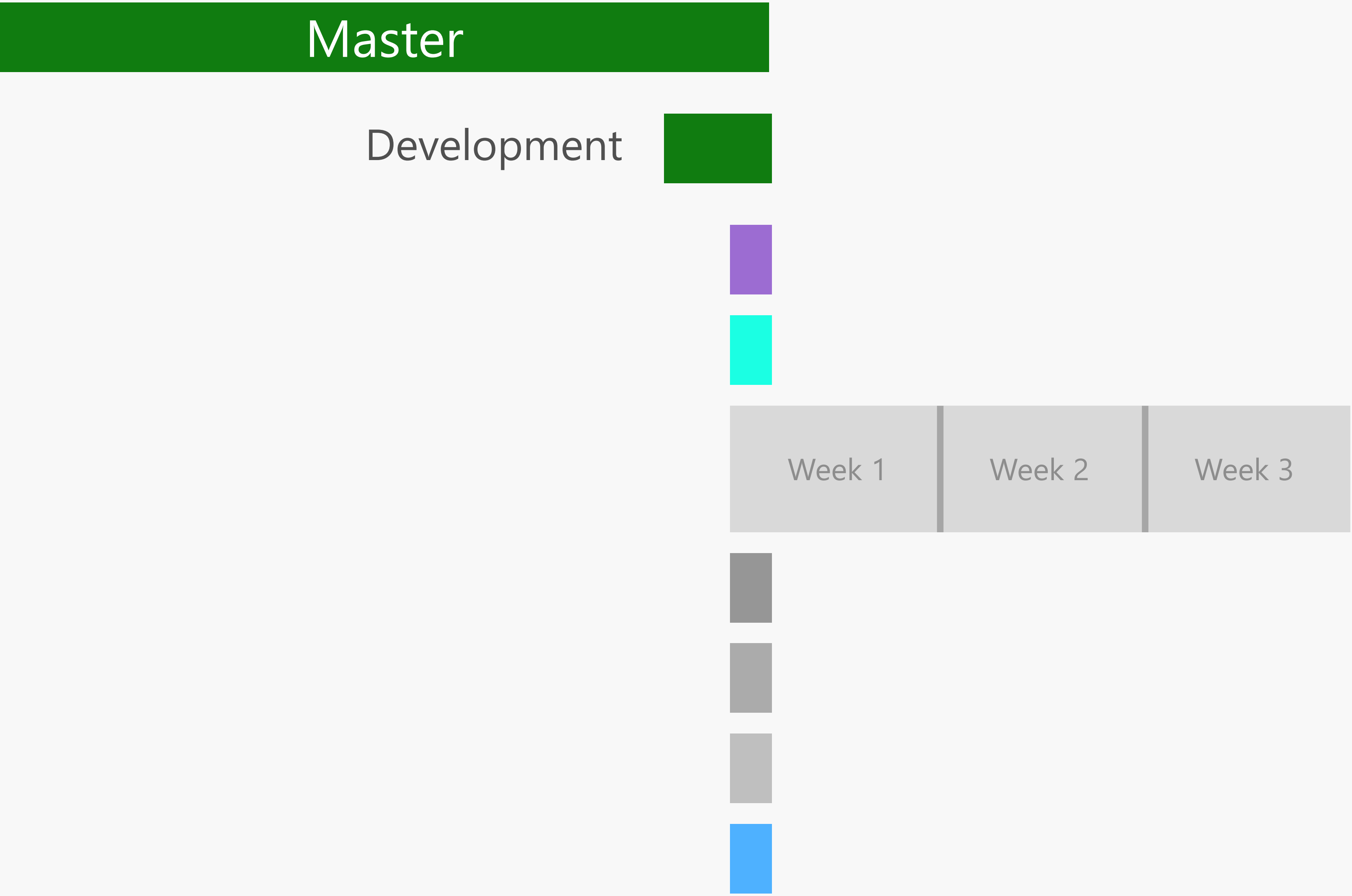
It needs to come together...



Merge Debt



Work with Development Branch…



Master

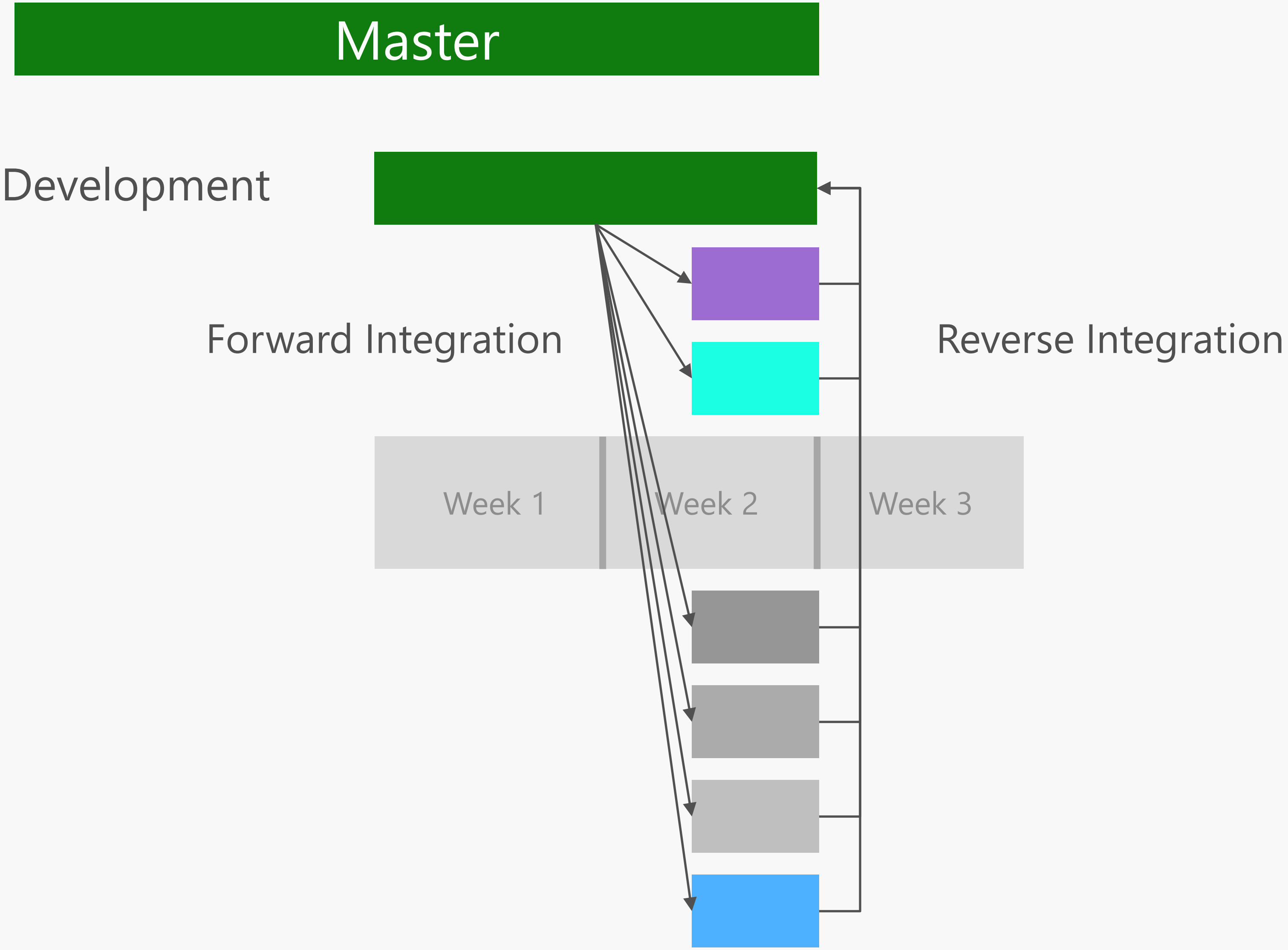
Development

Week 1

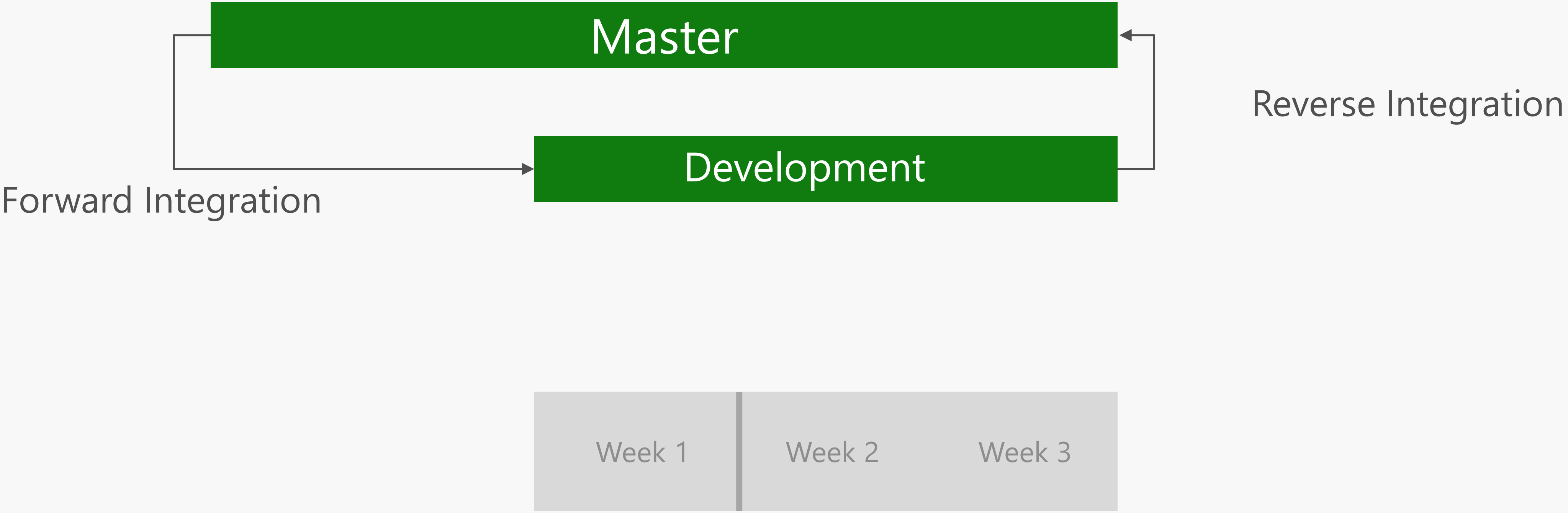
Week 2

Week 3

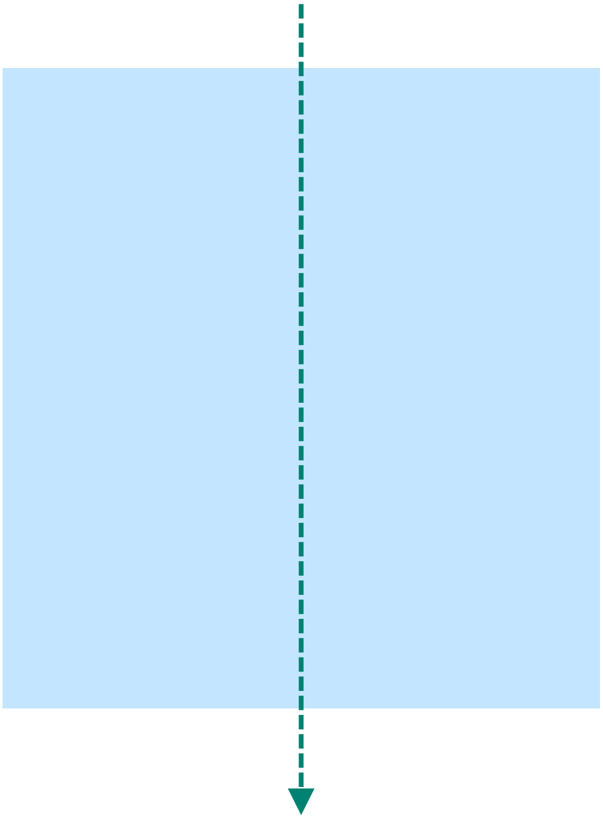
Integrate continuously...



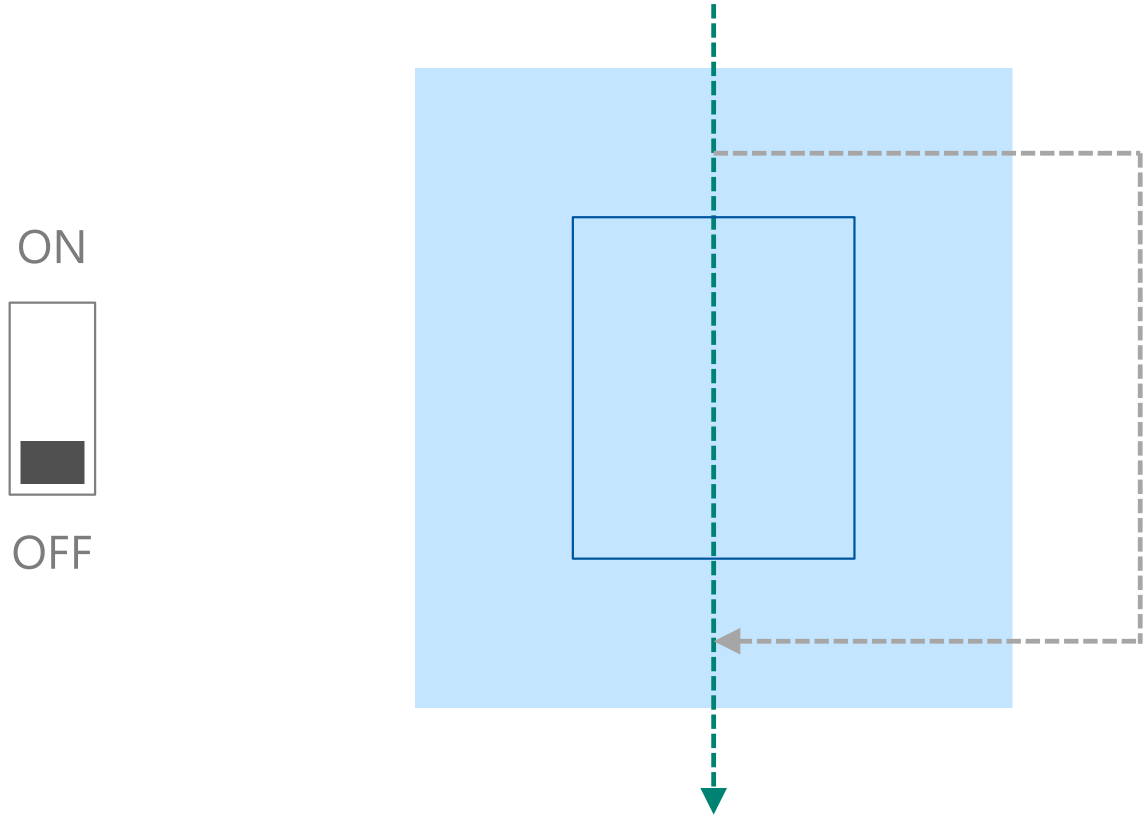
Integrate Development Branch...



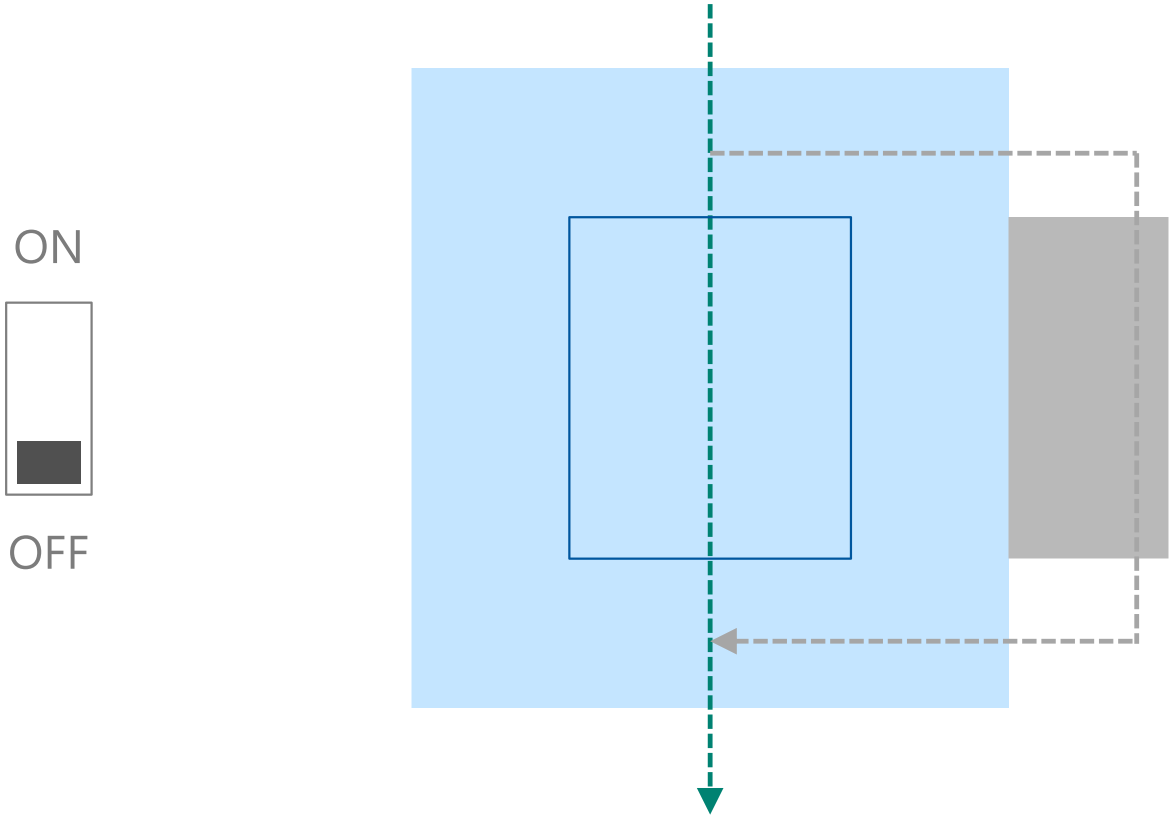
Feature Flags :Everything is going to production



Feature Flags



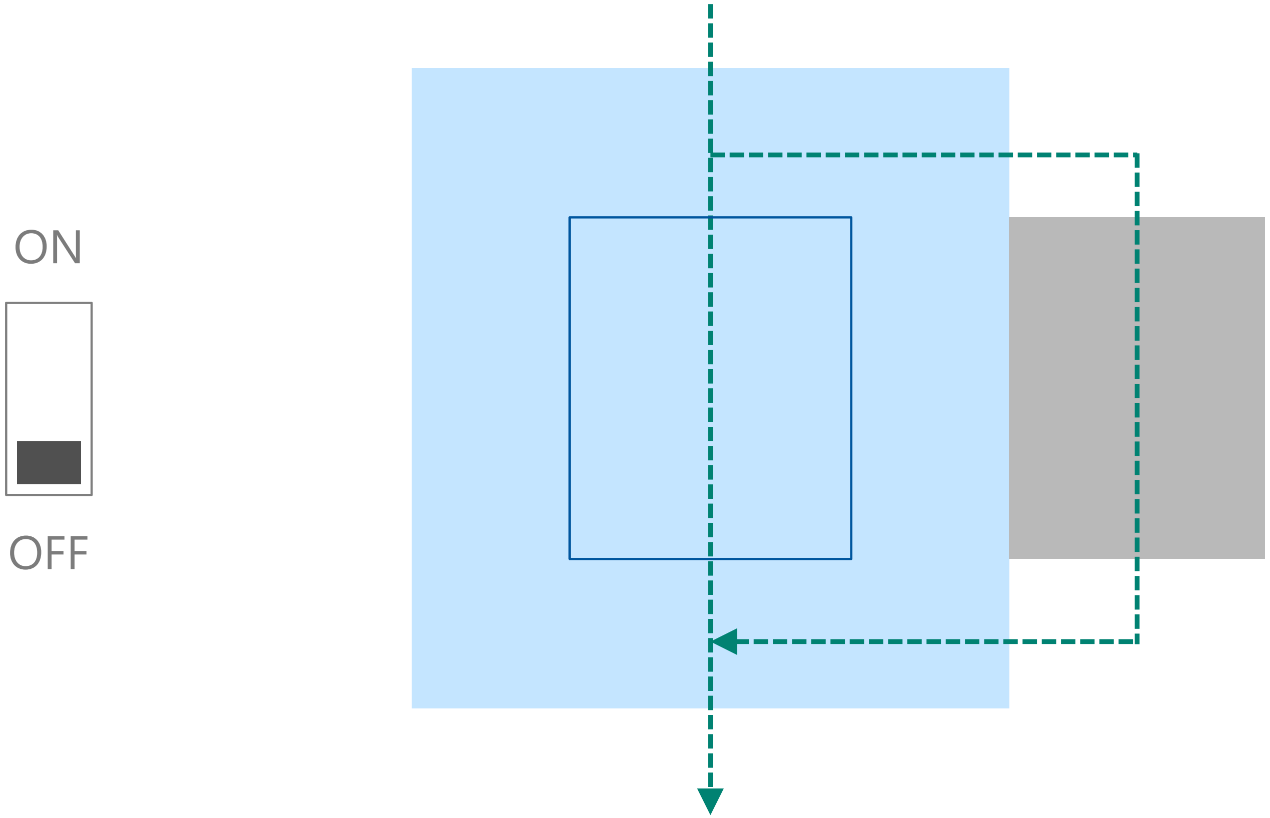
Feature Flags



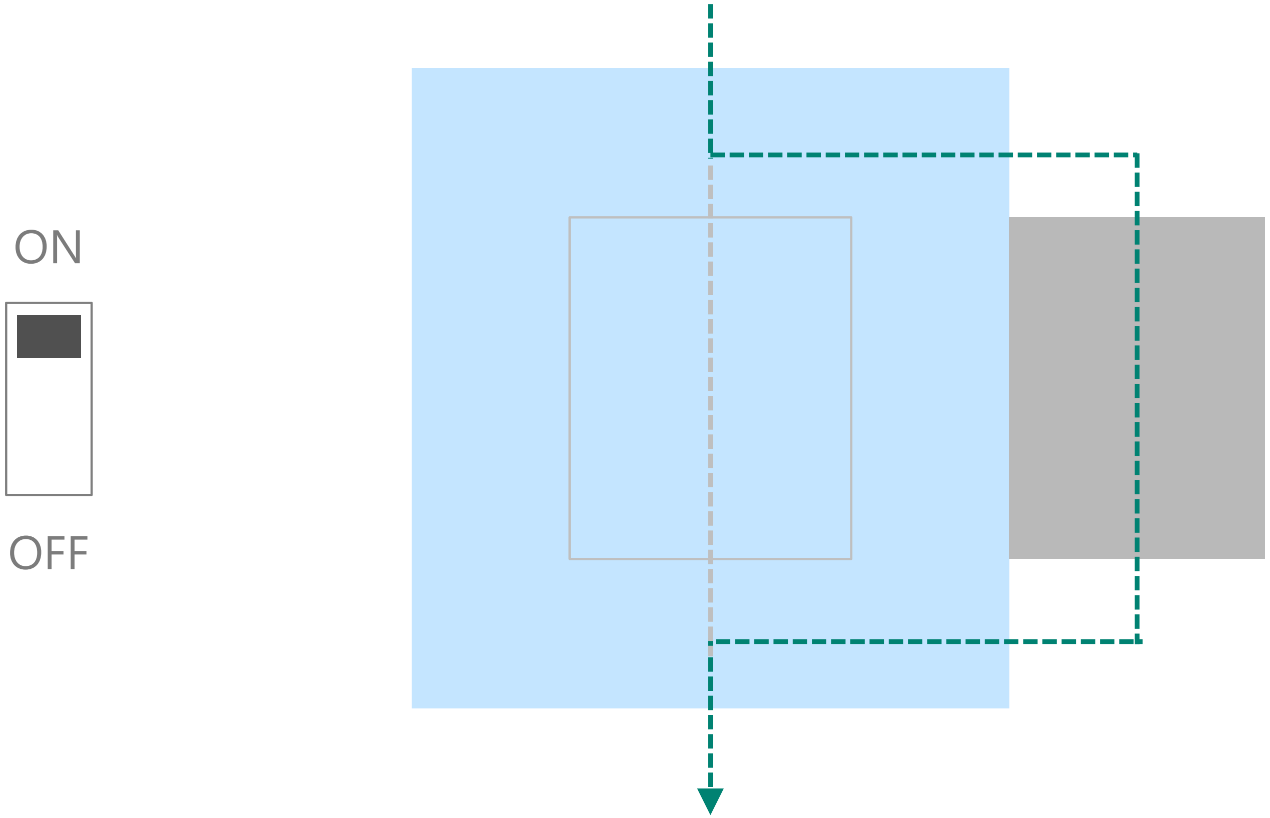
ON

OFF

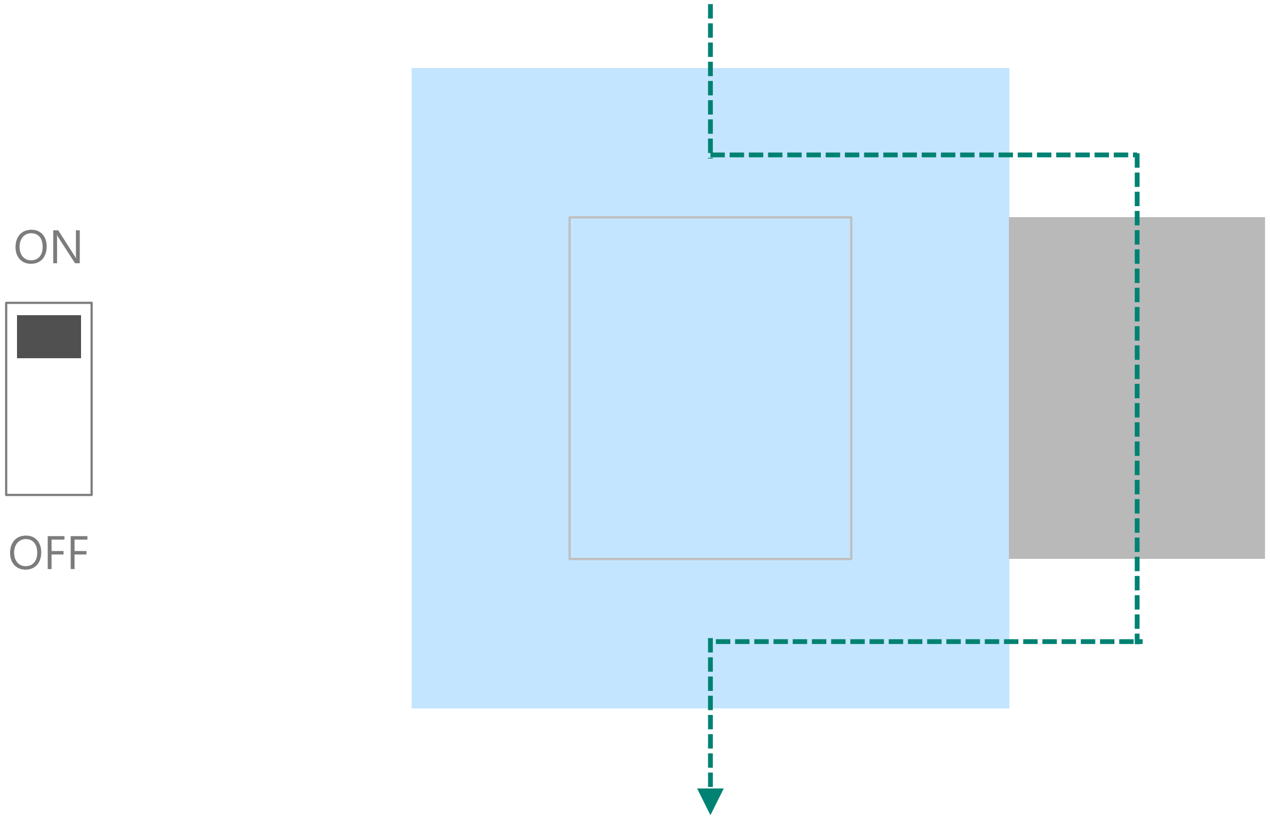
Feature Flags



Feature Flags



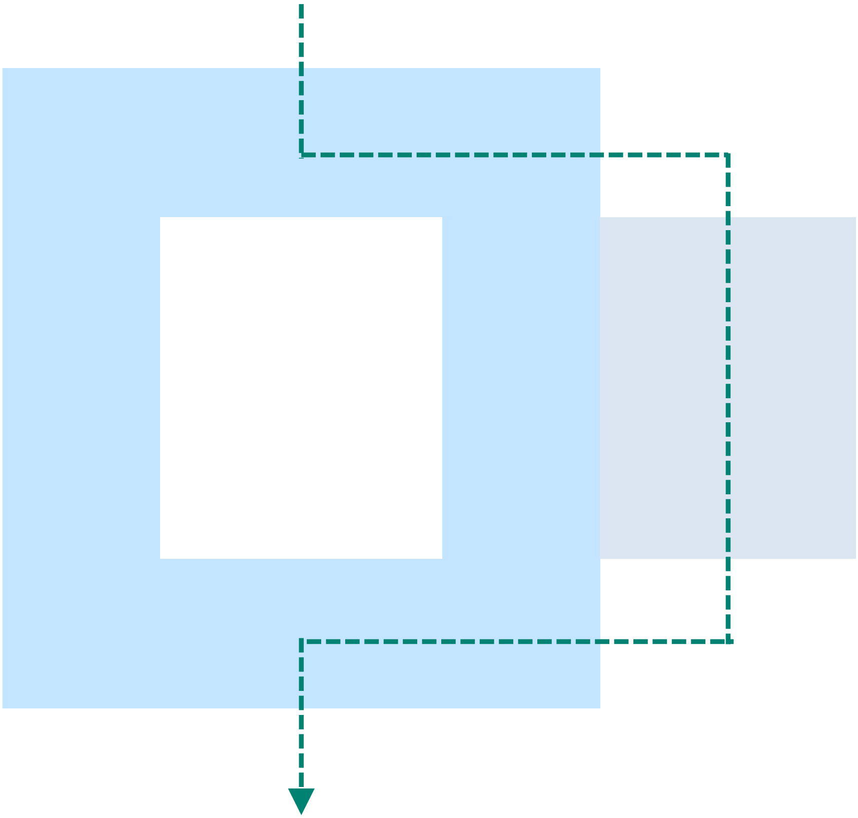
Feature Flags



ON

OFF

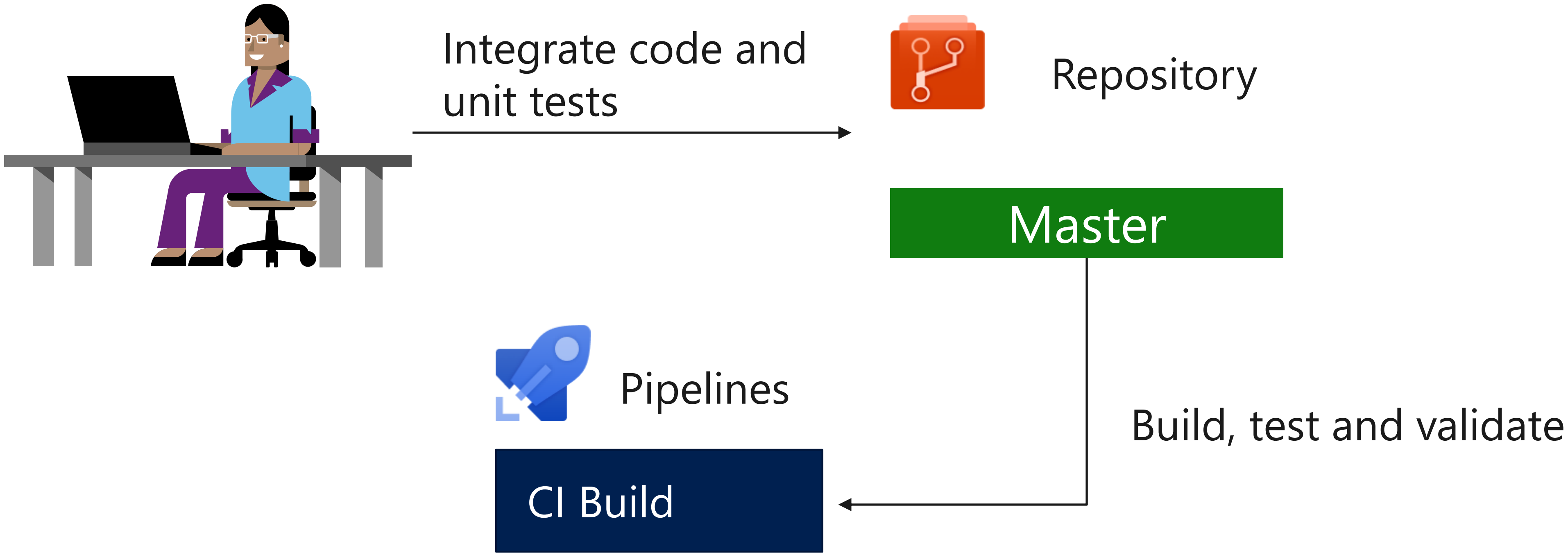
Feature Flags



Azure Pipelines

CI/CD

Continuous Integration (CI)



Integrate code and unit tests

Repository

Master

Build, test and validate

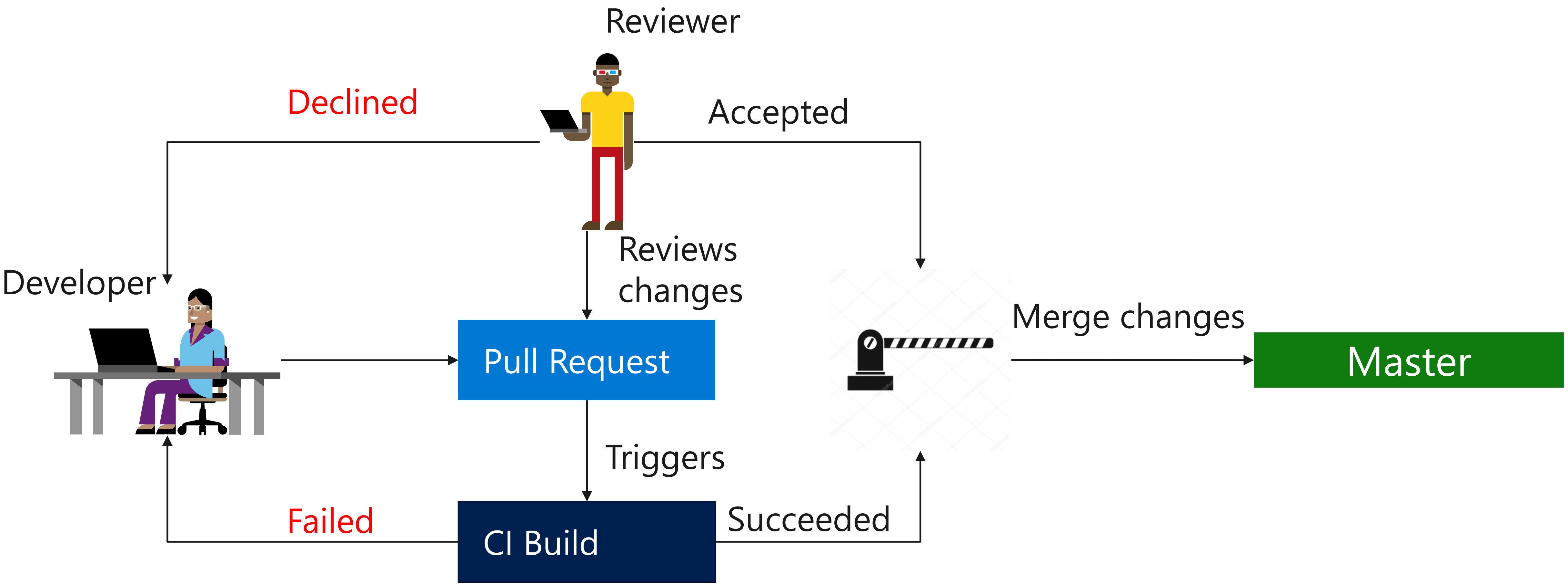
Pipelines

CI Build

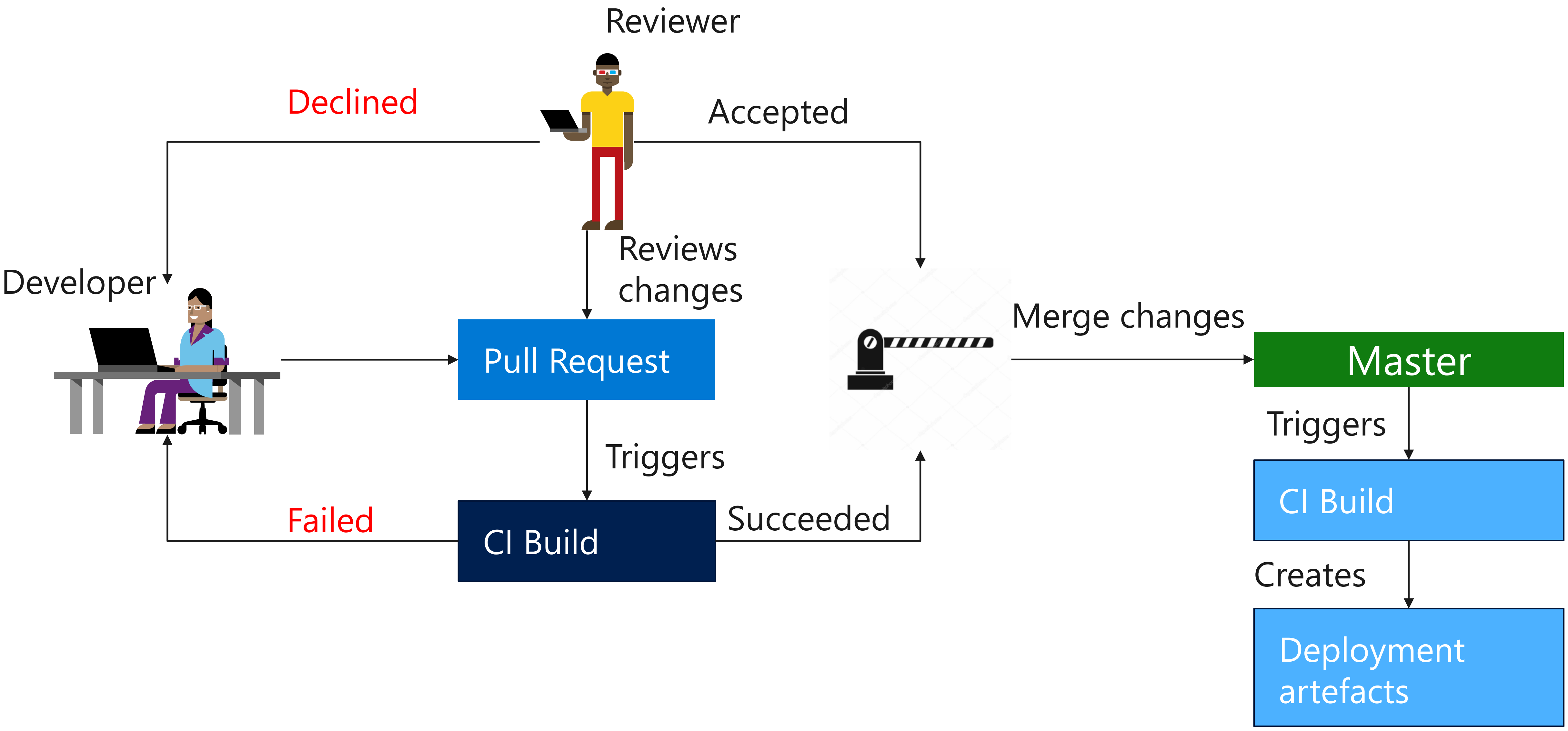
Pull Request

- CI keeps the master branch clean.
- Developer submits a “Pull Request” when a feature is completed and, on approval of the pull request, changes get merged to the master branch.
- A pull request is approved by reviewers
- A pull request is validated by a CI build

Pull request



CI Build to create deployment artefacts

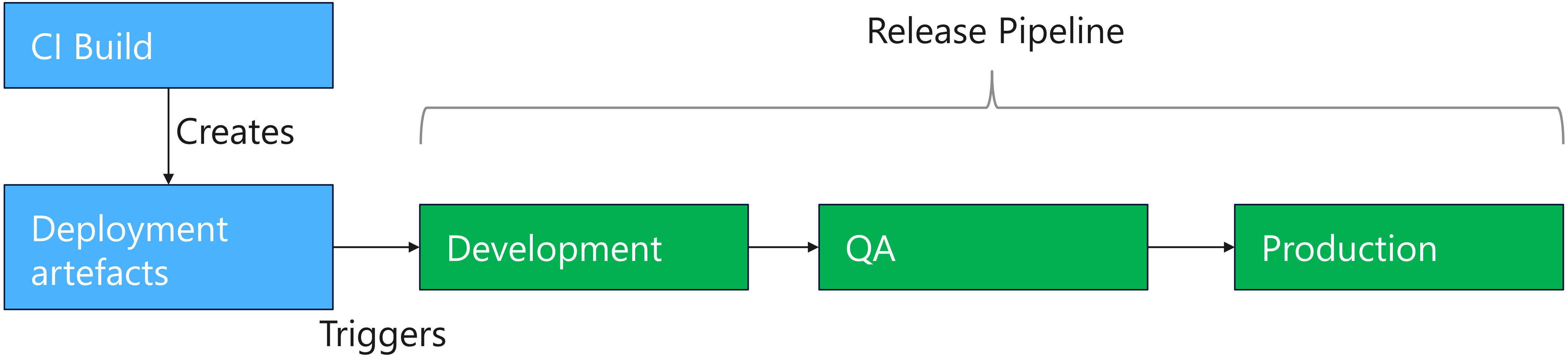


Continuous Deployment (CD)

Continuous Deployment is the process to build, test and deploy from a build to a production environment.

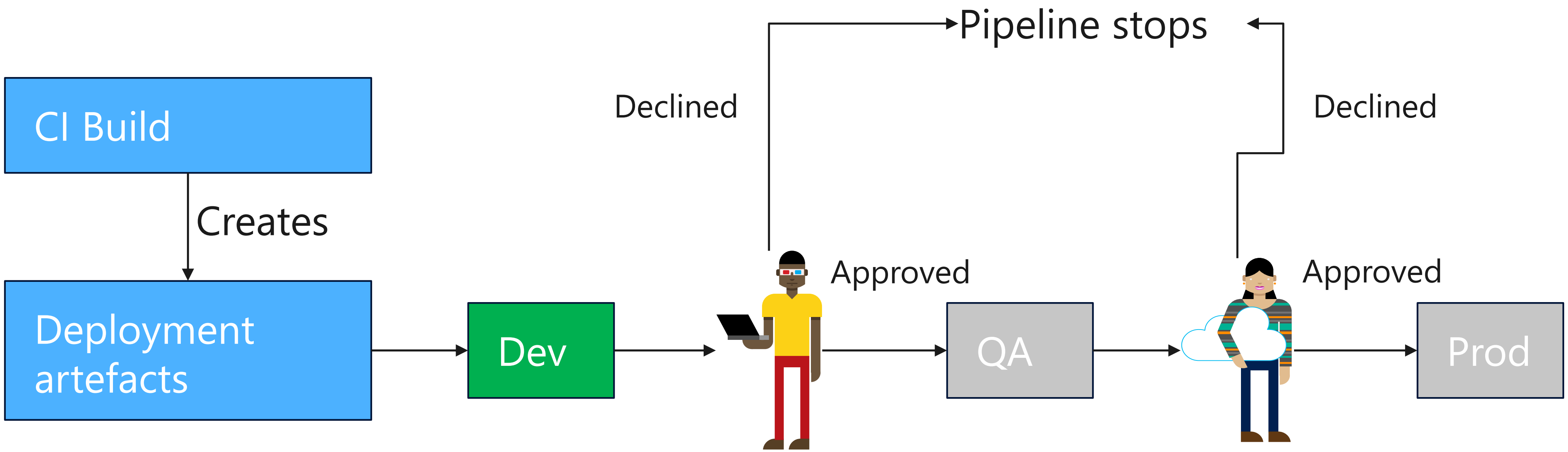
Multiple testing or staging environment create a *Release Pipeline* to automate the creation of infrastructure and deployment of applications.

Continuous Integration starts the CD process.

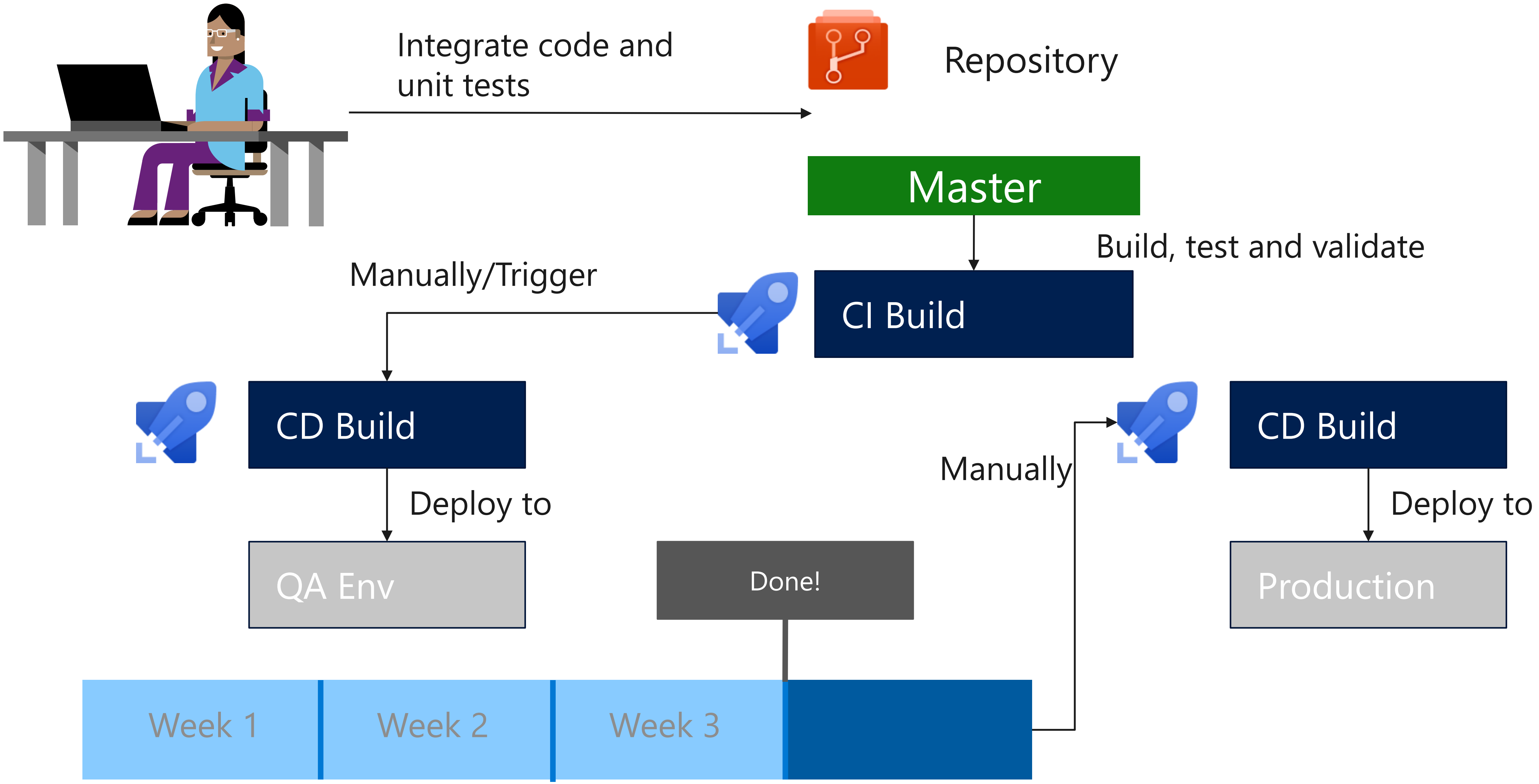


Approval Step

Before a deployment is rolled out to another environment in the pipeline an approval step can be used in which a decision maker signs off on the changes electronically.

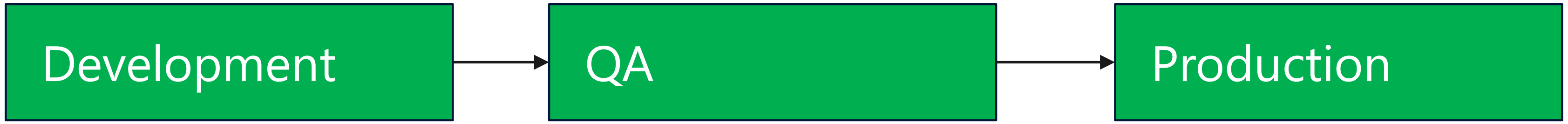


Continuous Delivery



CD Strategies

- Test A/B

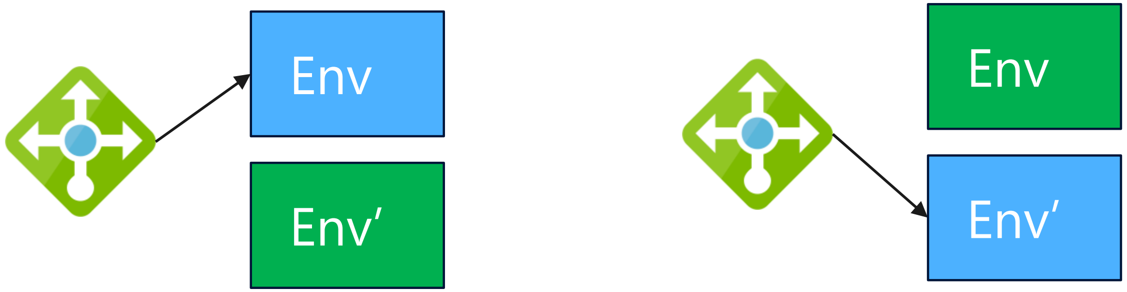


Development

QA

Production

- Blue/Green



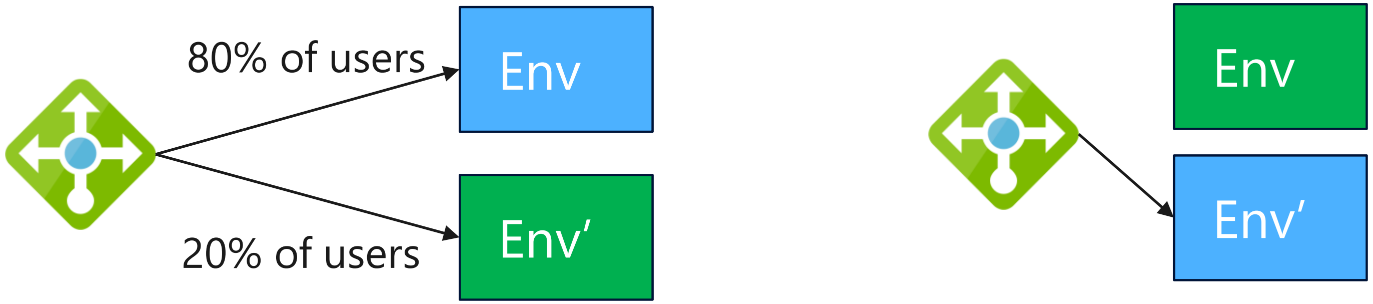
Env

Env′

Env

Env′

- Canary Release



80% of users

Env

20% of users

Env′

Env

Env′

DevOps on Azure

DevOps on Azure

First Party Tools

End to end solutions, enterprise grade

Azure DevOps
Team Foundation Server

Visual Studio App Center

Third Party Integrations

Integrations with industry leading tools

HashiCorp Terraform

Jenkins

CHEF

ANSIBLE

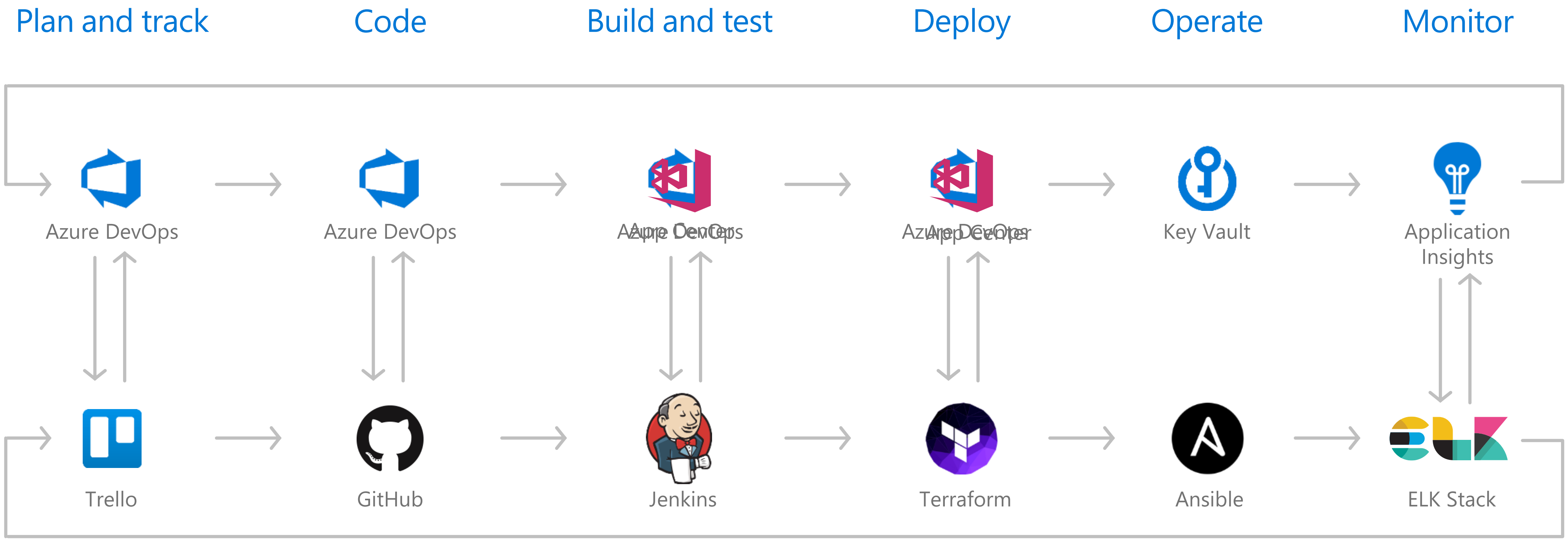
puppet

Azure Capabilities

Built into Azure, out-of-the-box capabilities

- ARM Templates
- Azure Monitor
- Application Insights
- Log Analytics
- Azure Key Vault

Azure DevOps framework



DevOps for all

First Party Tools

End to end solutions, enterprise grade

Third Party Integrations

Integrations with industry leading tools

Azure Capabilities

Built into Azure, out of the box

What is the right solution for my customer?

Customer Scenario: DevOps Beginner

Customer

"I know I need DevOps, but there are so many options out there and I don't know where to start"

"I want an enterprise grade, scalable, end-to-end solution"

"What's Microsoft's approach to implementing DevOps?"

First Party Tools

+

Azure Capabilities

Talking Points

Single End-To-End industry leading solution

Start small and grow

Best way to do Azure DevOps

Enterprise ready

This is how we do DevOps!

Customer Scenario: DevOps Partial Implementer

Customer

“I’m already implementing a few DevOps practices with a technology I evaluated and works for me”

“I have specific pain points I haven’t addressed yet”

“I don’t want to switch from technologies I started investing in but open to extend my process”

First Party Tools

+

Third Party Integrations

HashiCorp Terraform

Jenkins

ANSIBLE

+

Azure Capabilities

Talking Points

Keep your current investments!

VSTS can work as an orchestrator to your current chosen technologies

VSTS is an end-to-end solution but you can pick and choose what to use.

Rich and vast marketplace for VSTS with +500 extensions

Customer Scenario: DevOps Invested (non-Microsoft)

Customer

“I’m already implementing DevOps and have vast investments in my chosen solution”

“I just need to understand how to use my toolchain on Azure”

“I’m multi-cloud”

Third Party Integrations

HashiCorp Terraform

Jenkins

ANSIBLE

+

Azure Capabilities

Talking Points

Keep your current investments!

Microsoft’s Openness story

Our investments in 3rd party technologies

Azure Marketplace

Strong DevOps Partners

We love you just the way you are!

DevOps and other conversations

Azure Migration

Refactor/Rearchitect/Rebuild with DevOps practices and new processes

Rehost TFS in the cloud. Migrate to VSTS

Adjust to cloud speed

Reap the benefits of the cloud

Digital Transformation & Application Innovation

Empower employees and drive culture change with DevOps

Optimize operations

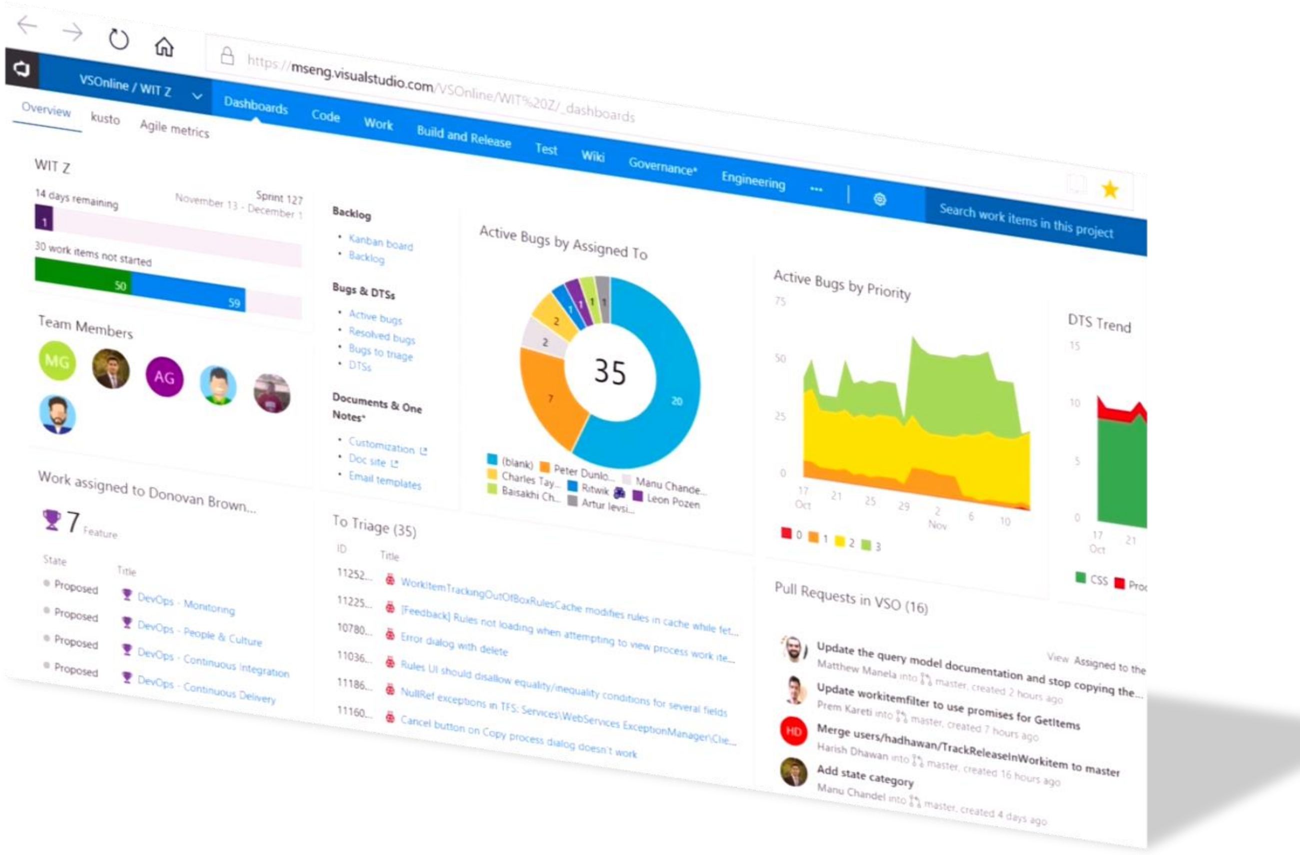
Continuously innovate with quick development cycles

Engage with customers and implement feedback loops

DevOps at Microsoft

DevOps at Scale

Visual Studio Team Services is the toolchain of choice for Microsoft engineering



Work assigned to Donovan Brown...

7 Feature

State	Title
Proposed	DevOps - Monitoring
Proposed	DevOps - People & Culture
Proposed	DevOps - Continuous Integration
Proposed	DevOps - Continuous Delivery

To Triage (35)

ID	Title
11252...	WorkItemTrackingOutOfBoxRulesCache modifies rules in cache while fet...
11225...	[Feedback] Rules not loading when attempting to view process work ite...
10780...	Error dialog with delete
11036...	Rules UI should disallow equality/inequality conditions for several fields
11186...	NullRef exceptions in TFS: Services\WebServices ExceptionManager\Clie...
11160...	Cancel button on Copy process dialog doesn't work

Pull Requests in VSO (16)

View Assigned to the

Update the query model documentation and stop copying the...
Matthew Manela into master, created 2 hours ago

Update workitemfilter to use promises for GetItems
Prem Kareti into master, created 7 hours ago

Merge users/hadhawan/TrackReleaseInWorkitem to master
Harish Dhawan into master, created 16 hours ago

Add state category
Manu Chandel into master, created 4 days ago

+85,000 Engineers at Microsoft are using VSTS

Windows Engineering Team

VSTS scales to host the world's largest Git repo

- → 300GB Source code
- → 3.5M Files
- → 4,000+ Devs

VSTS Engineering Team

Embracing DevOps practices = More value to customers

More features delivered to customers in 2016 than in previous 4 years combined

http://aka.ms/devopsatmicrosoft