Microsoft

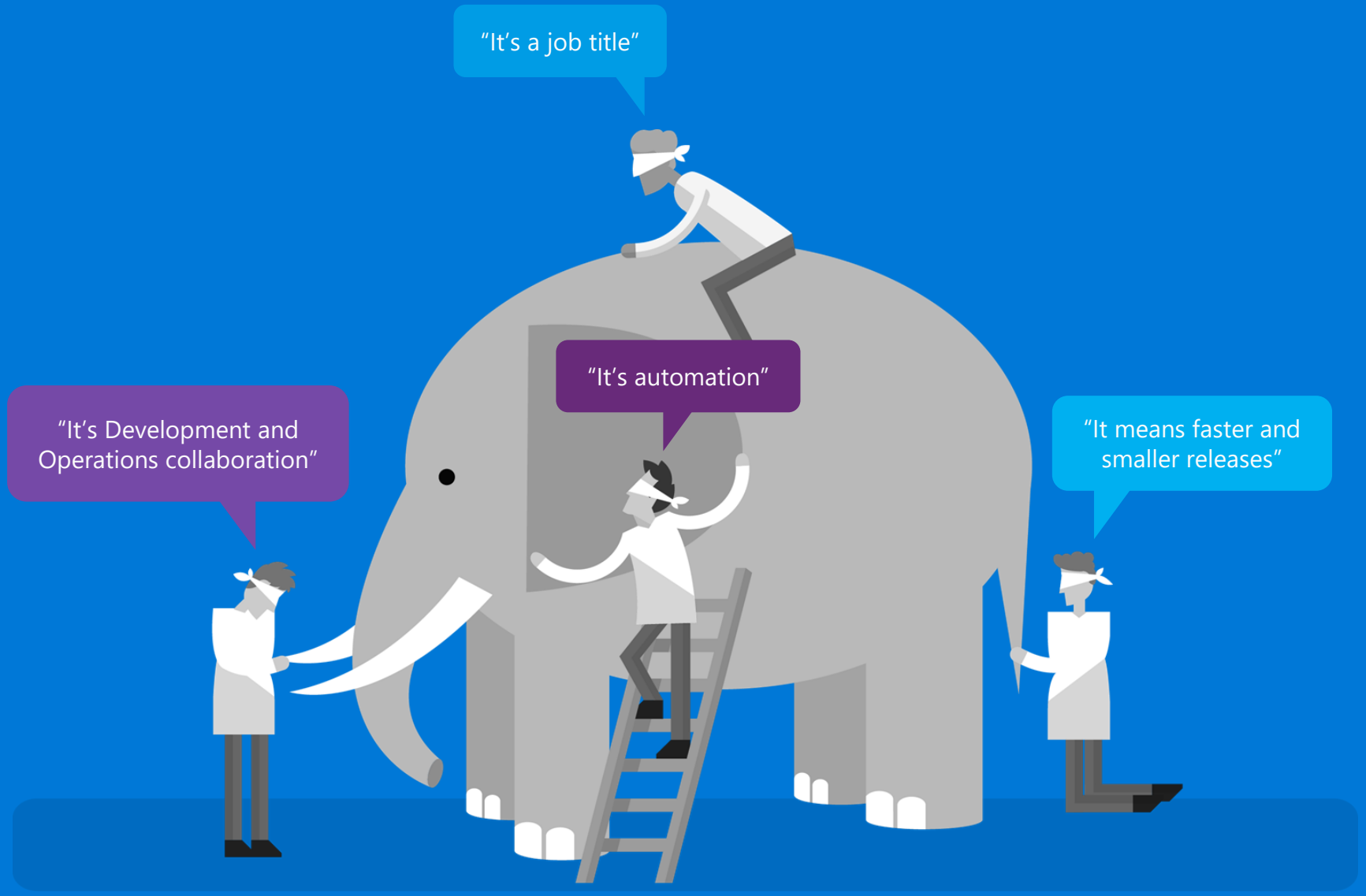
DevOps, Project Management, and VSTS

Crystal Tenn
Crystal.Tenn@microsoft.com

MICROSOFT CONFIDENTIAL

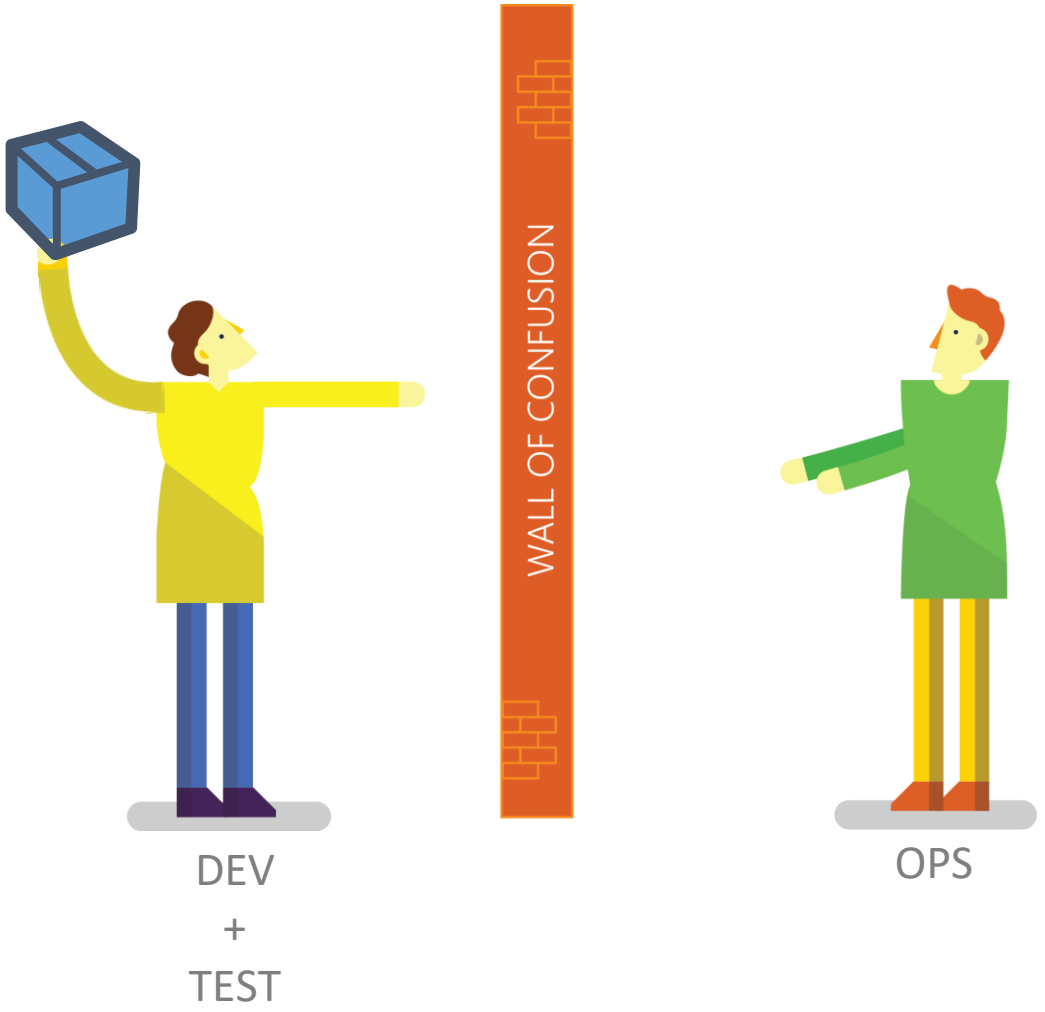
DevOps

What is DevOps?



MICROSOFT CONFIDENTIAL – INTERNAL ONLY

Traditional Development and Operations



Microsoft

What's the ~~problem~~ Opportunity?

60%

apps get to market late

Arriving to market too late

70%

apps go to market missing functionality

Poor quality

5

months on average to create a new app. Only 28% of companies worldwide create an app in less than 3 months

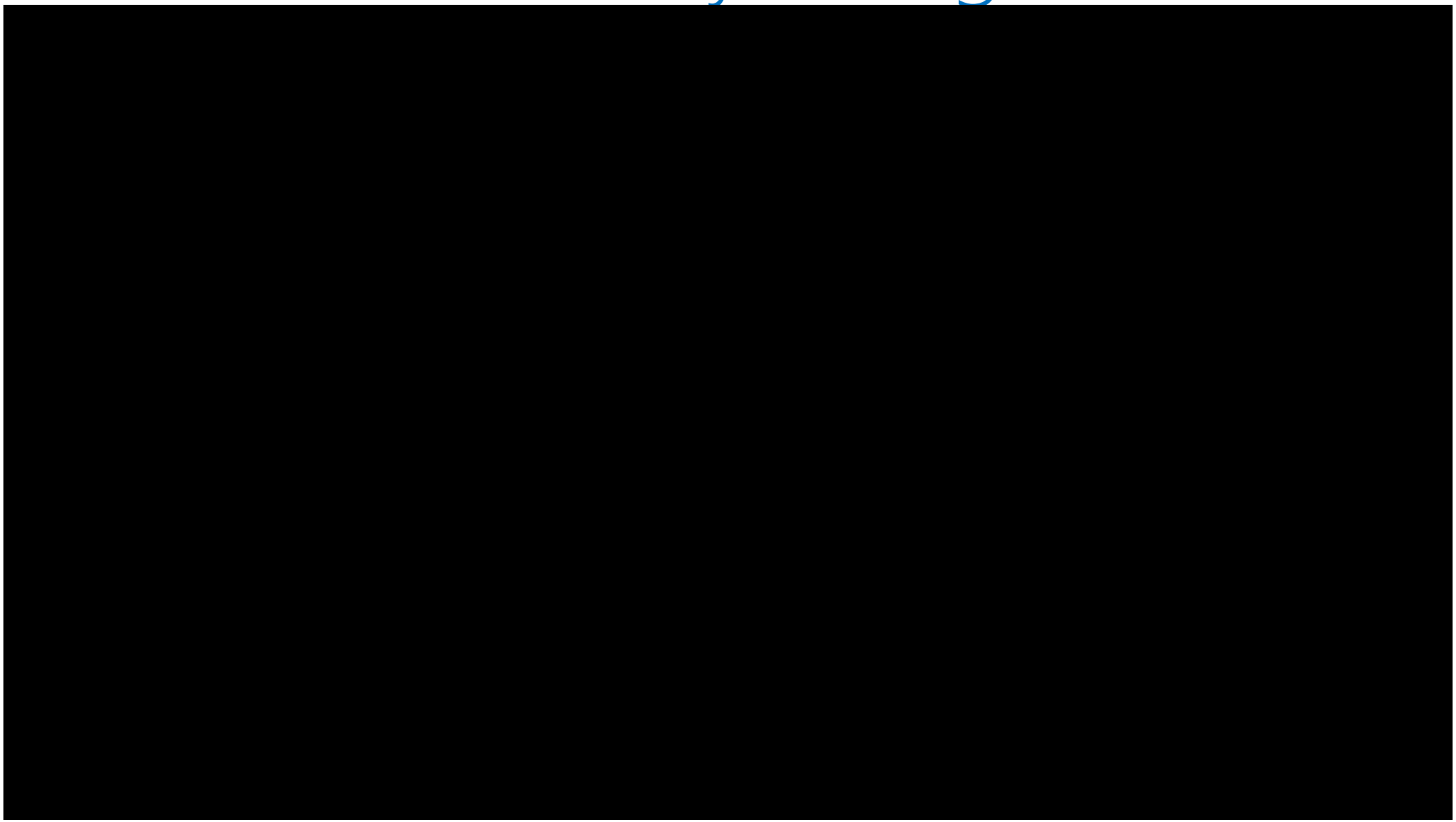
Lengthy lifecycles

Solution: Deliver Value Faster

Source: CA / Coleman Parkes Research 2012

Microsoft

Testimonial 1: Industry changes



Microsoft

Value delivery challenges

Outperforming teams are **54%** more likely to collaborate extensively with their counterparts

Collaboration blockers

26.7%
No executive support

56.7%
Cultural inhibitors

43.3%
Fragmented processes

41% ... of development budgets for software, IT staff and external professional services will be consumed by **poor requirements**

40% ... of implementations end up **getting reworked** because they don't meet the users' original requirements

80% failure rate ...
... for companies that try to adapt their existing tools for **DevOps practices**

High IT performance correlates with strong business performance, helps boost productivity, market share and profit

IT drives business success!

Developers

IT Ops

Business

70% of **CIOs** Would **increase risk** to **reduce IT costs** and **accelerate business agility**

1 in 6 **IT decision makers** is still unfamiliar with the term DevOps

DevOps was being initiated by more development teams than IT Ops teams by about a **40%** to **33%** margin

3/4 of teams **have adopted Agile methodologies**

Responding to ongoing needs for **efficiency and growth**

dual goals

Always keeping all systems **safe and secure**

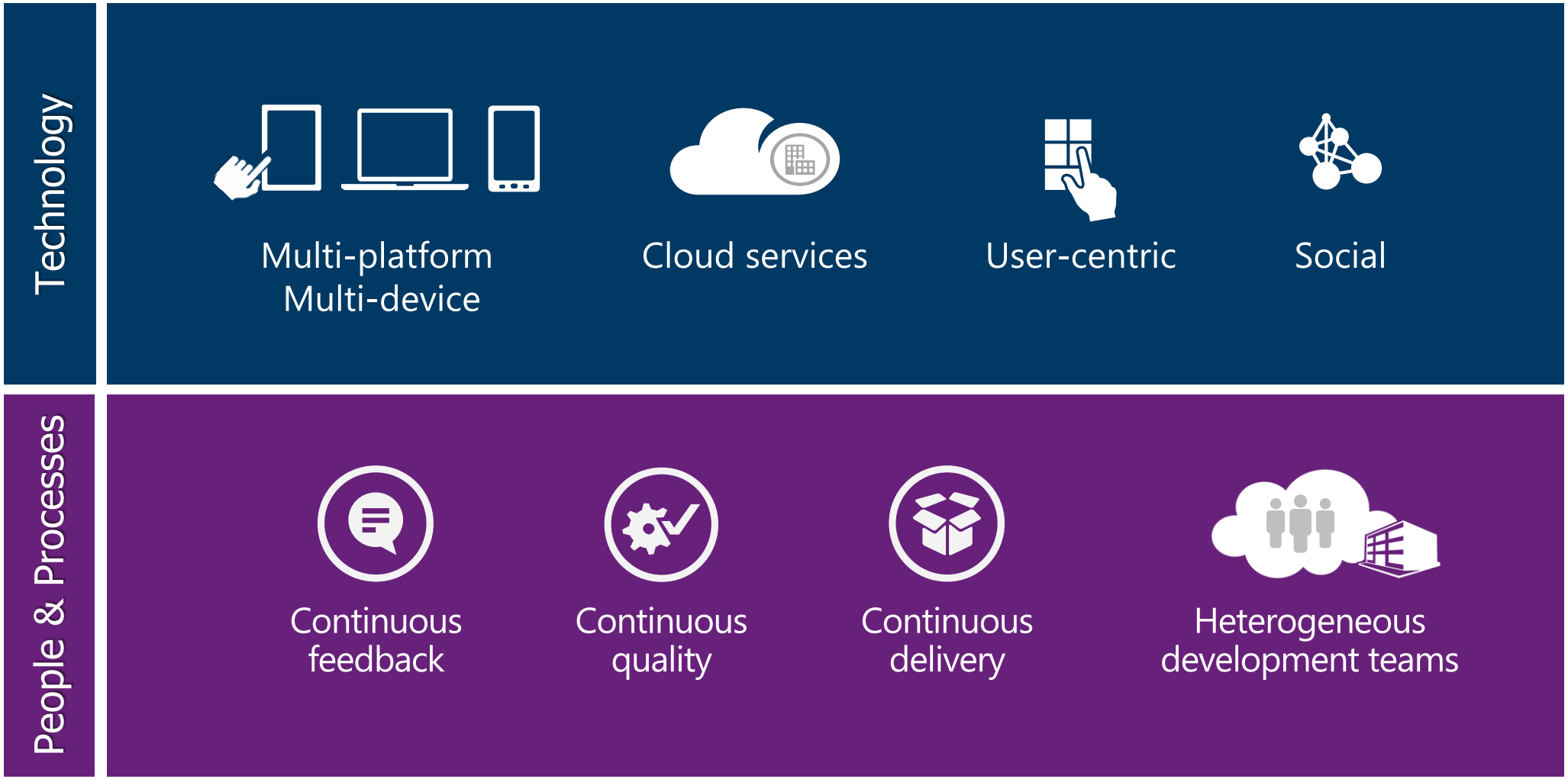
The average hourly cost of infrastructure failure is **$100,000** per hour

It takes on average **200 minutes** to diagnose and repair a production issue

A bug caught in production ends up costing **100x more** than if the same bug was found earlier in the development cycle

MICROSOFT CONFIDENTIAL – INTERNAL ONLY

Industry trends



Microsoft

DevOps

- DevOps is the union of people, process, and products to enable continuous delivery of value to our end users.

- The contraction of “Dev” and “Ops” refers to replacing siloed Development and Operations to create multidisciplinary teams that work together with shared and efficient practices and tools.

Why Implement DevOps?

- Deliver more secure, higher quality solutions faster and cheaper.
- Dynamic and reliable experience
- Teams must rapidly iterate on updates to their project
- Remove traditional bottlenecks

Why do DevOps?.

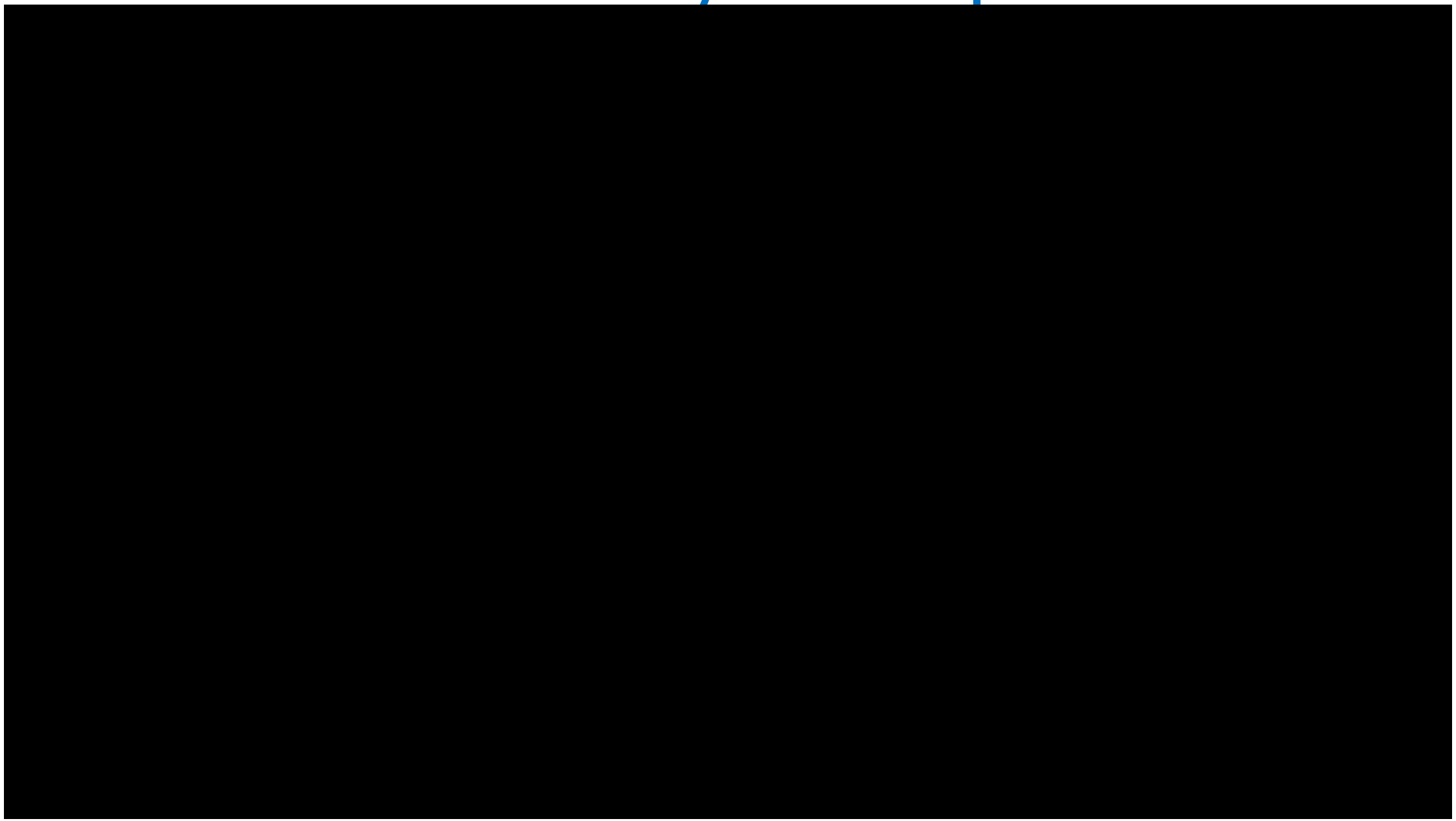
- Competitive Advantage – Productivity

The Phoenix Project

Company	Deploy Freq.	Lead Time	Reliability	Cust. Responsiveness
Amazon	23,000/ day	Minutes	High	High
Google	5,500/ day	Minutes	High	High
Netflix	500/ day	Minutes	High	High
Facebook	1/ day	Hours	High	High
Twitter	3/ wk	Hours	High	High
Traditional Enterprise	1/ 9 months	Months or Quarters	Low/ Med	Low/ Med

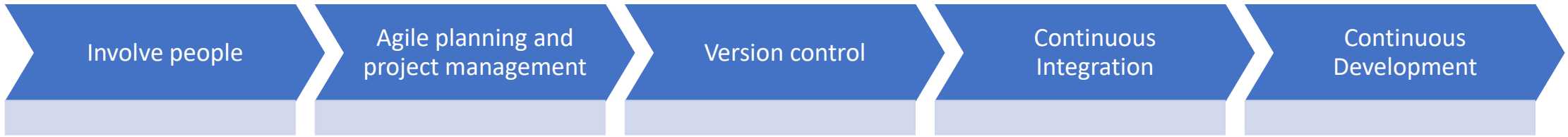
How much high performance DevOps organizations are outperforming the traditional enterprise counter parts.
*Note - from *The Phoenix Project*

Testimonial 2– Why DevOps

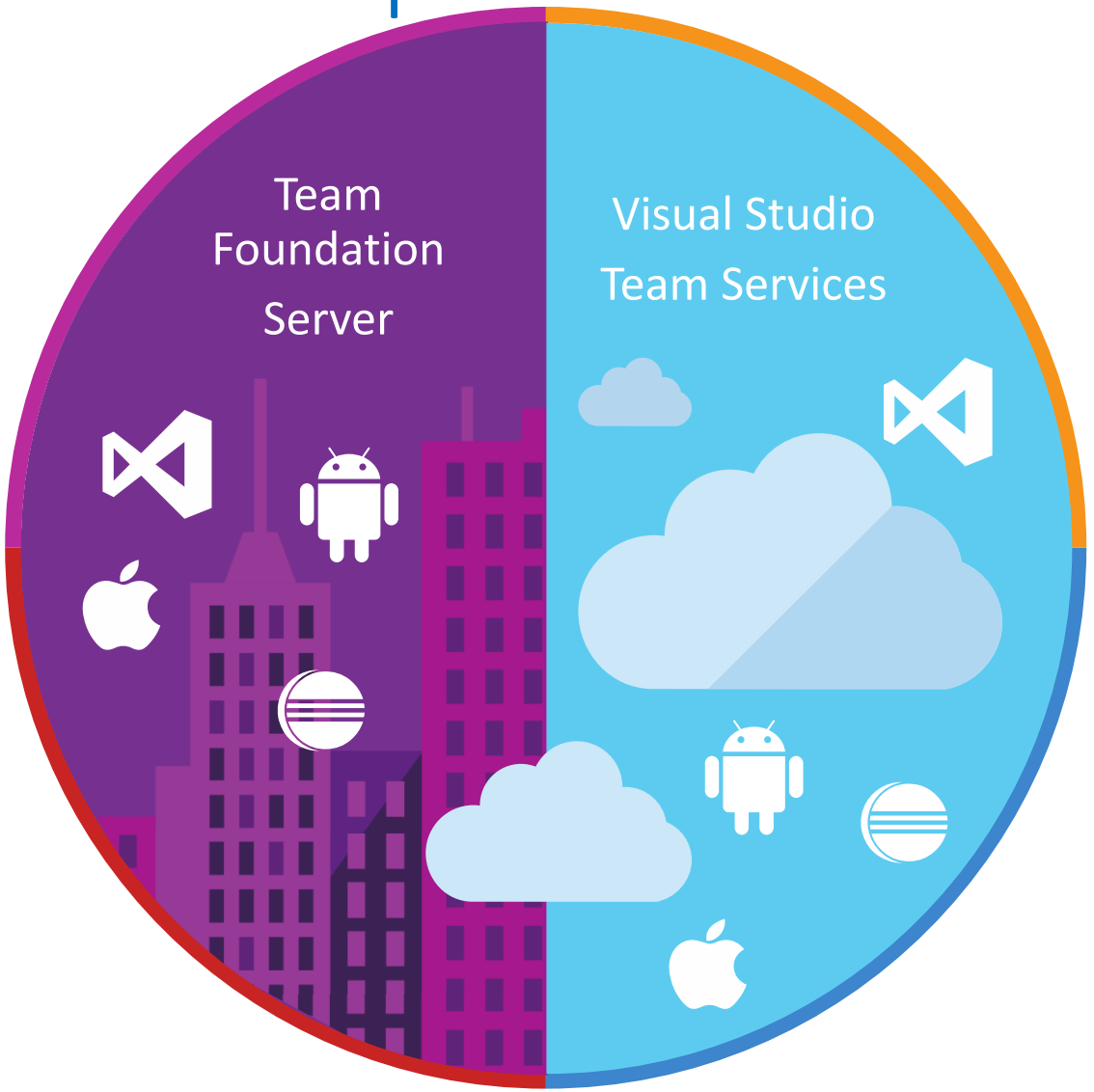


Microsoft

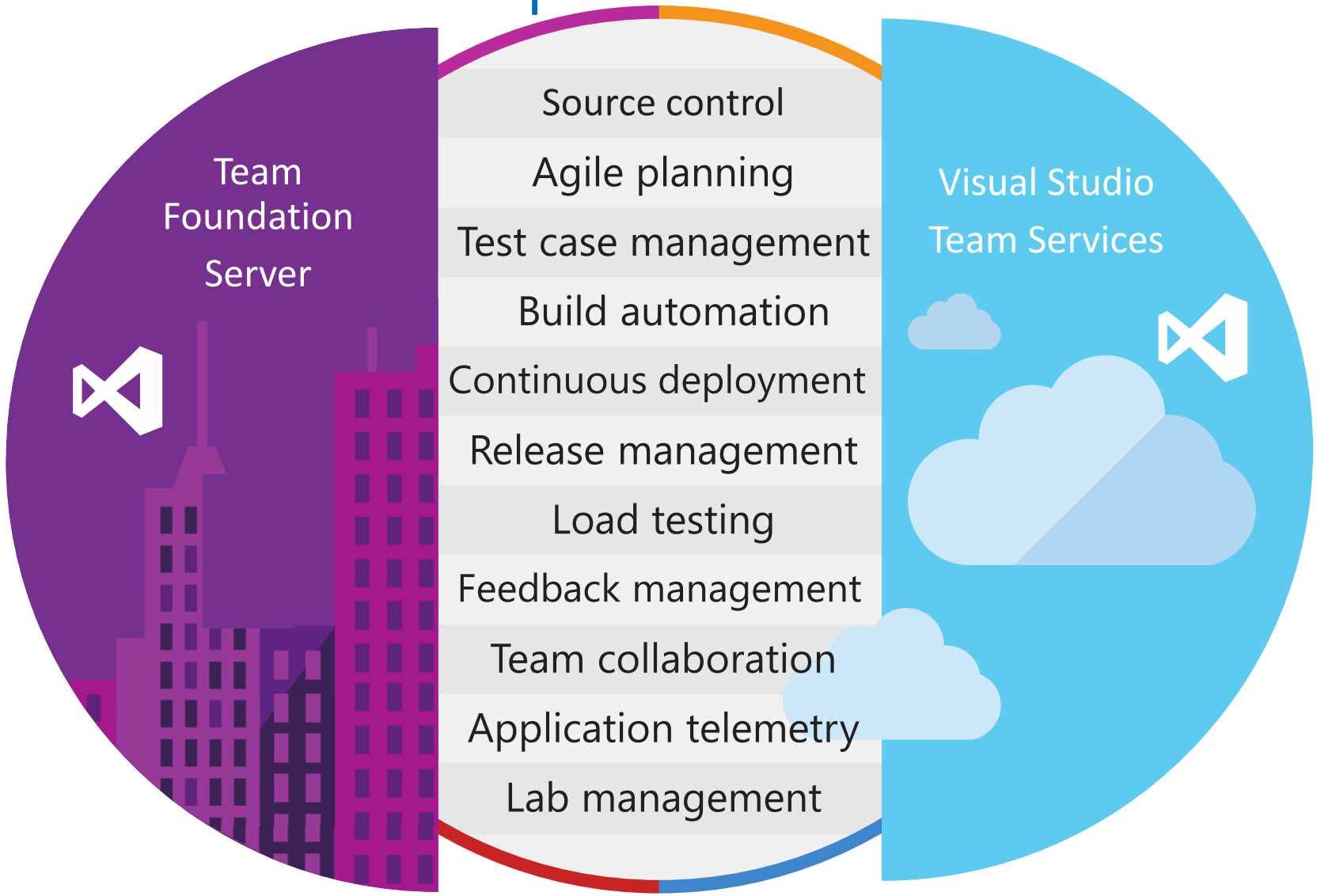
How to implement DevOps?



Microsoft ALM & DevOps Tools



Microsoft ALM & DevOps Tools



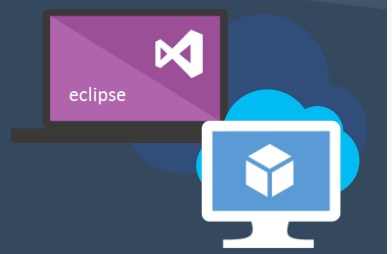
Microsoft Ecosystem

People | Process | Products

01

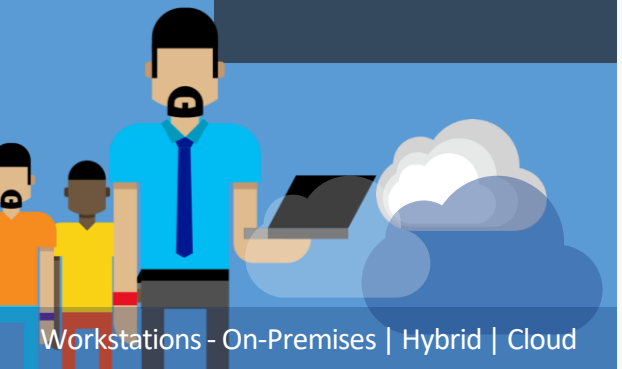
Develop

Developer Workstation



Team Collaboration

TFS / VSTS



Workstations - On-Premises | Hybrid | Cloud

02

Build & Test

Build/CI

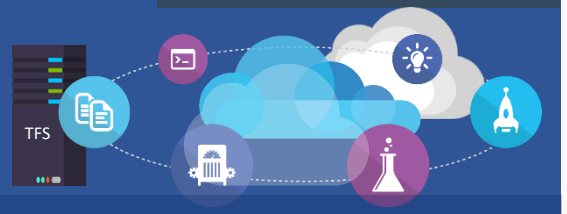
TFS / VSTS

Release Management for Visual Studio

Test

TFS / VSTS

Microsoft Test Manager



ALM Services - On-Premises | Hybrid | Cloud

03

Deploy

Release

Microsoft System Center

Release Management for Visual Studio

Automation Service

PowerShell | WAML

Azure Resource Management

Azure Command Line

Environments - On-Premises | Hybrid | Cloud

04

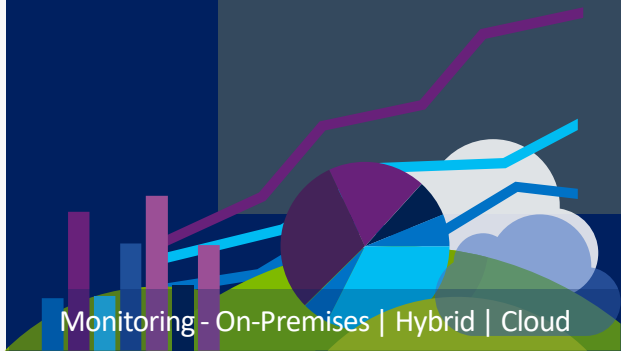
Monitor & Learn

Monitor

Microsoft System Center

VSTS

Application Insights



Monitoring - On-Premises | Hybrid | Cloud

Mixed Ecosystem

People | Process | Products

01 Develop

Developer Workstation

Team Collaboration

GitHub
CodePlex

02 Build & Test

Build/CI

gradle
GRUNT
Jenkins
Hudson

Test

gradle
GRUNT

03 Deploy

Configuration

puppet labs
CHEF

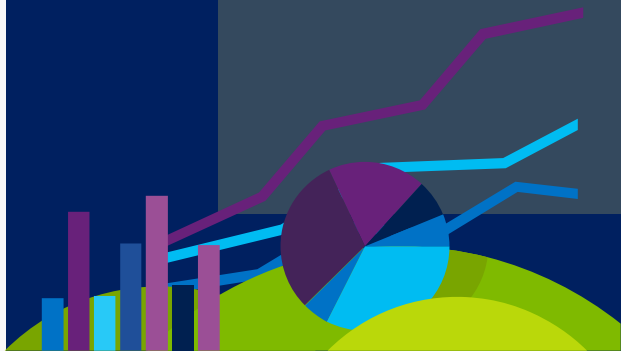
Release

gradle
GRUNT
Jenkins
Hudson
VAGRANT

04 Monitor & Learn

Monitor

Nagios
ZABBIX



This graphic shows OSS and partner products that are integrated with the Microsoft DevOps solution

Any Language, Any Platform

- Program in any language
  - .NET
  - Java
  - Python
  - Ruby
  - Nodejs
  - …
- Develop on any OS
  - Linux
  - Mac
  - Windows
- Deploy to any Platform
  - Android
  - iOS
  - Windows Phone
  - Docker
  - Azure
  - AWS
  - Linux
  - Mac
  - …

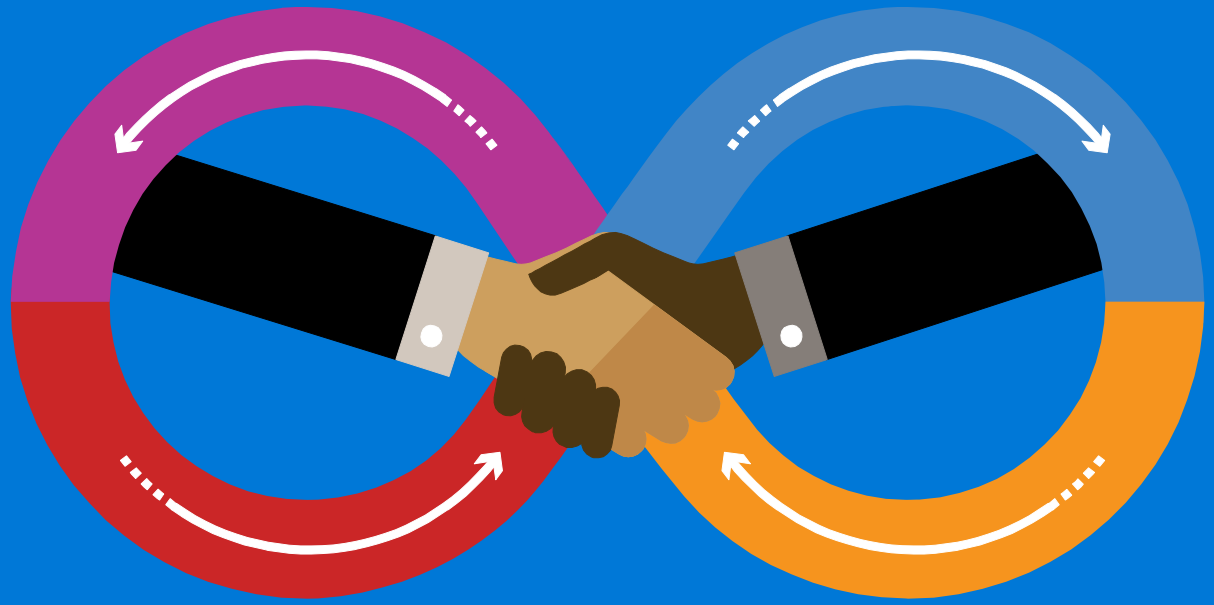
Microsoft

The converged DevOps lifecycle

Plan + Track

Release

Development



Production

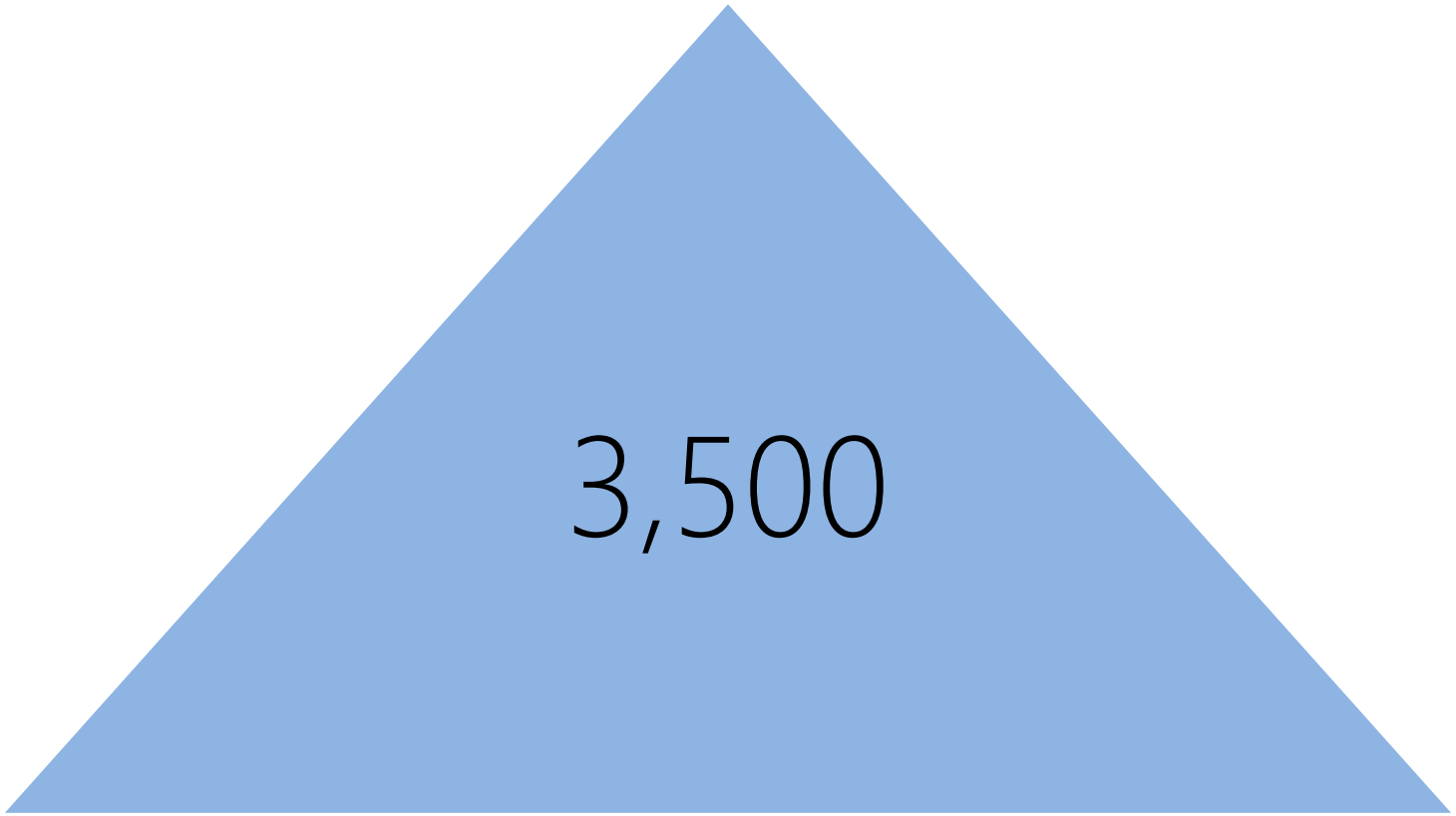
Develop + Test

Monitor + Learn

MICROSOFT CONFIDENTIAL – INTERNAL ONLY

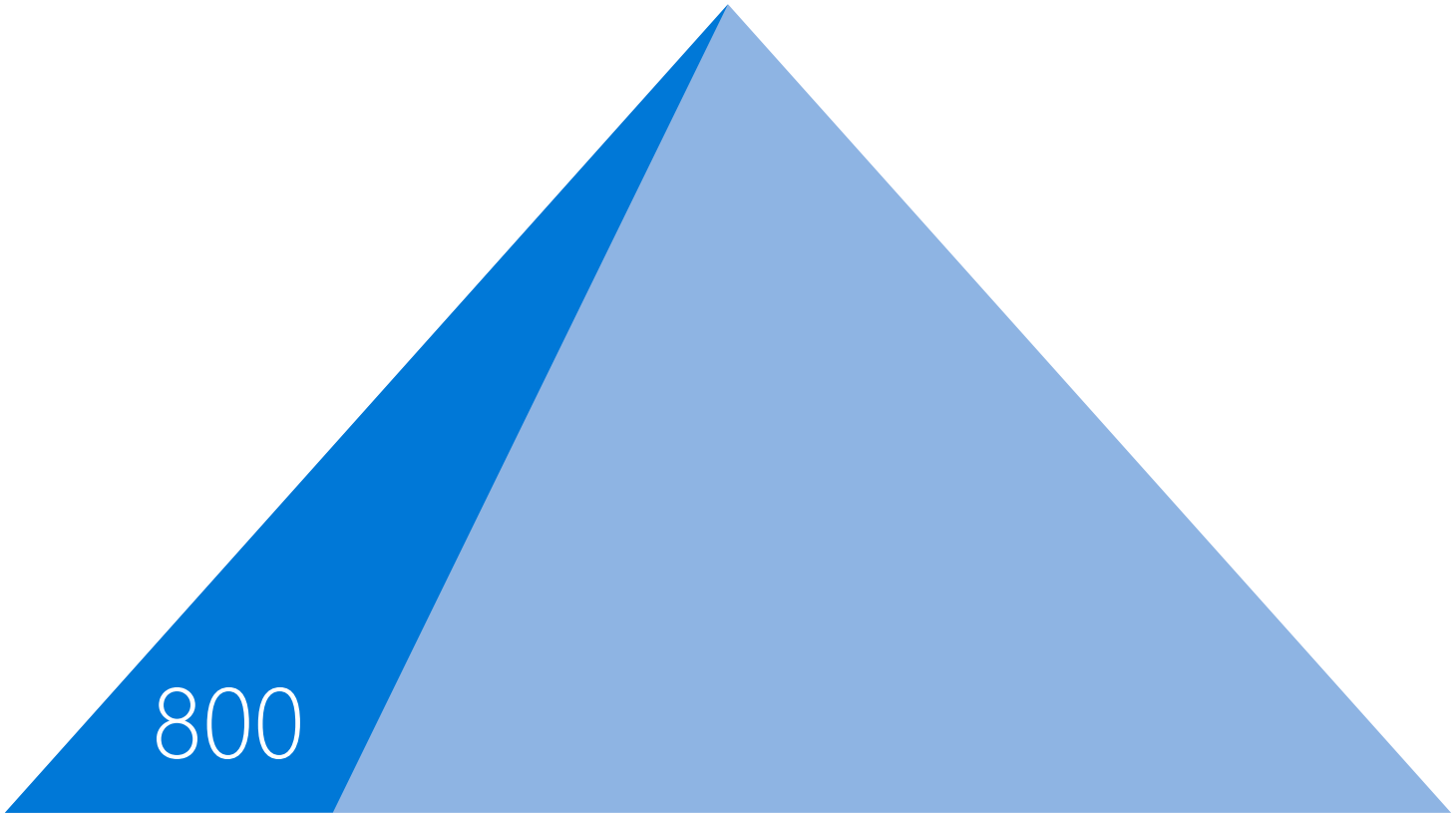
Case study:
Microsoft VSTS
DevOps Transformation

Microsoft



The Developer Division at Microsoft

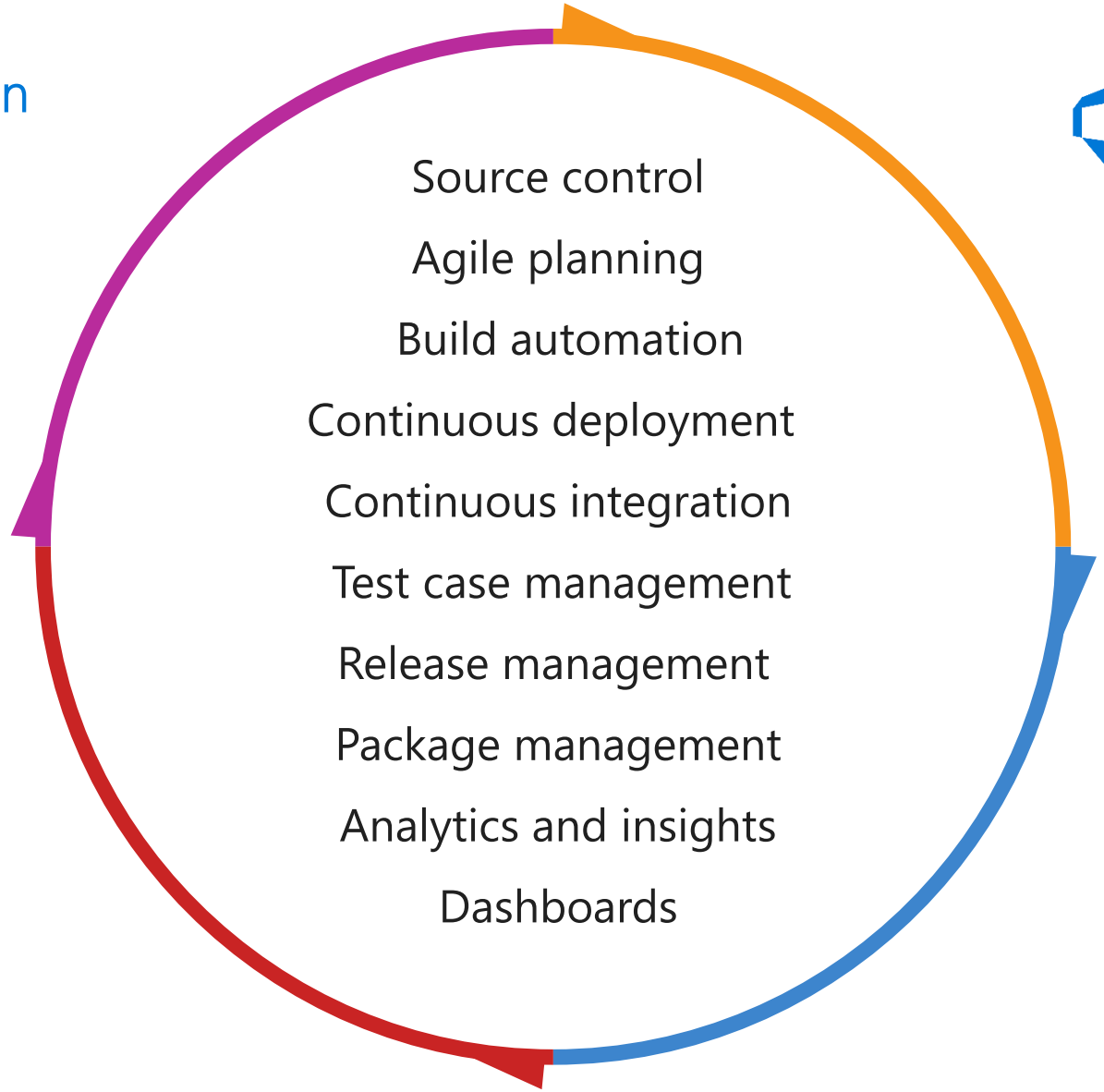
Microsoft



The VSTS team… spread out across 40 feature teams

Microsoft

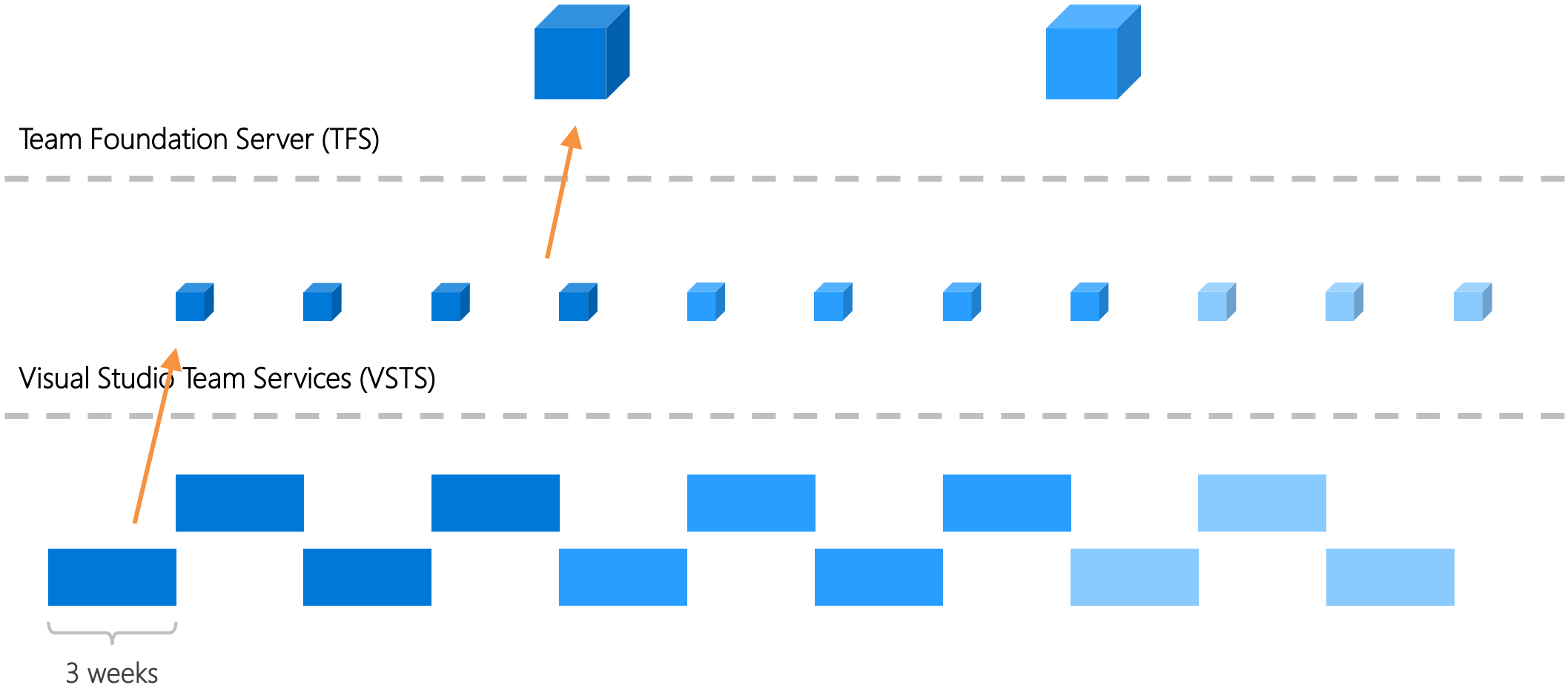
Team Foundation Server (TFS)



Visual Studio Team Services (VSTS)

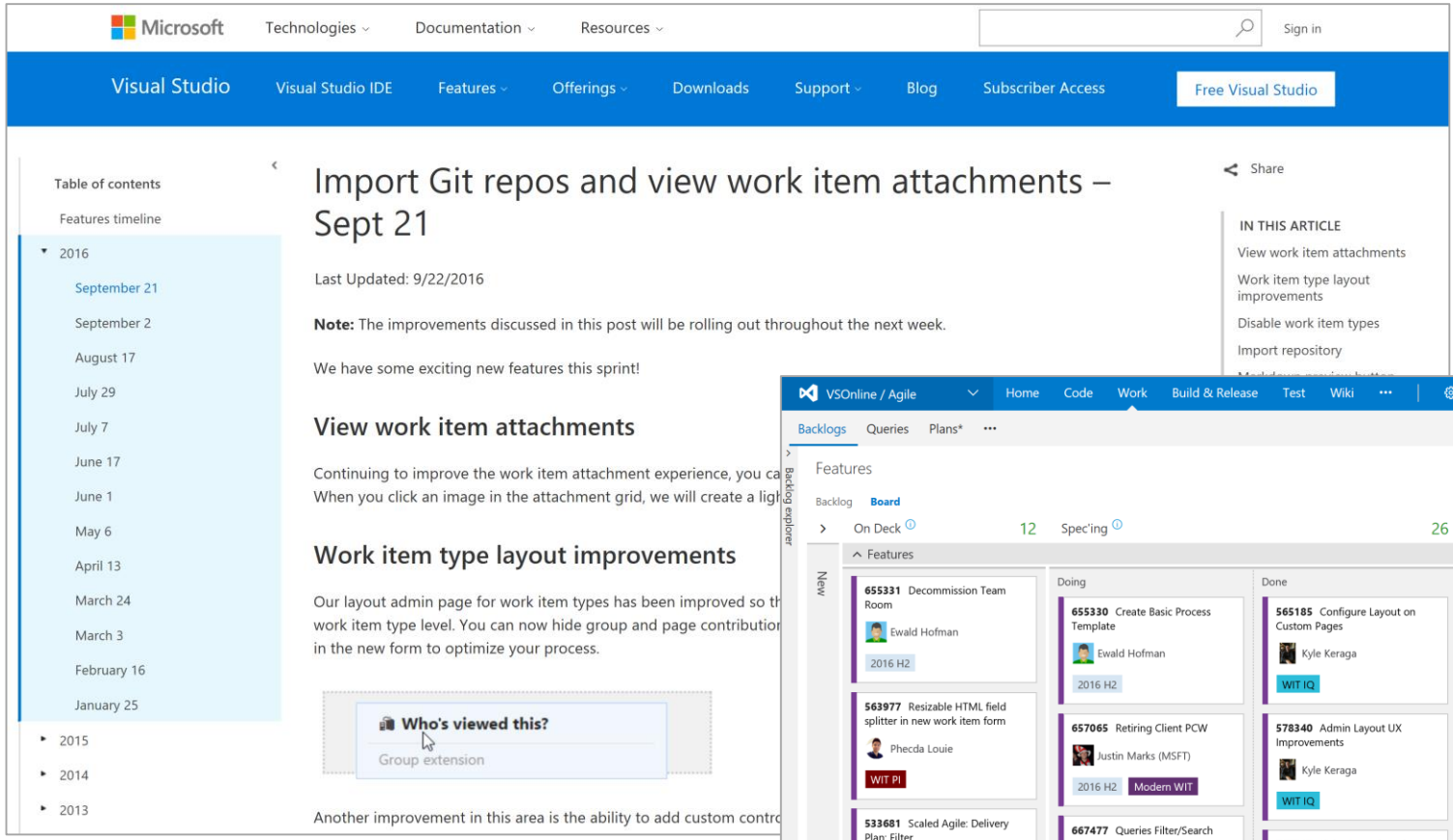
Microsoft

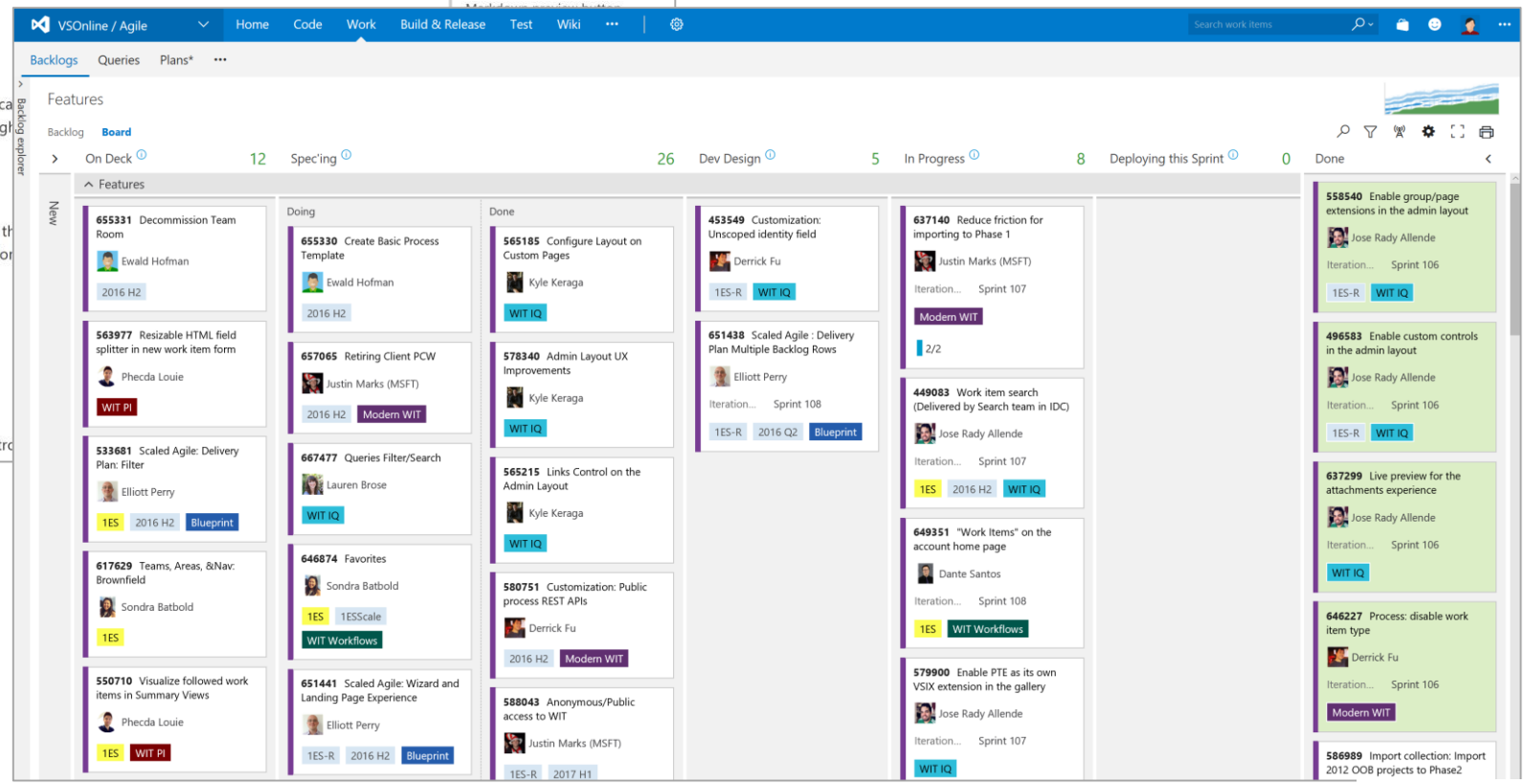
How do we work?



Microsoft

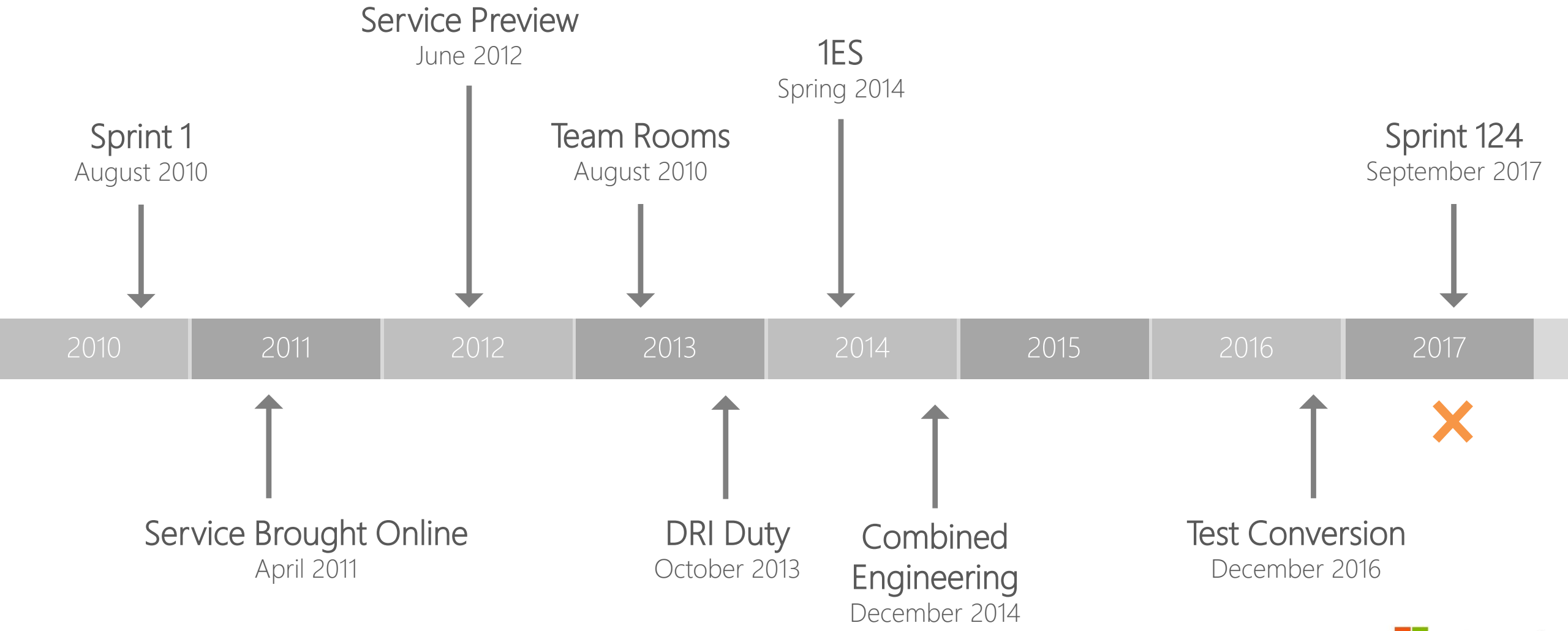
https://www.visualstudio.com/team-services/updates/





Microsoft

The Journey



Microsoft

What did it look like before?

Before

The OLD way

Microsoft

Before

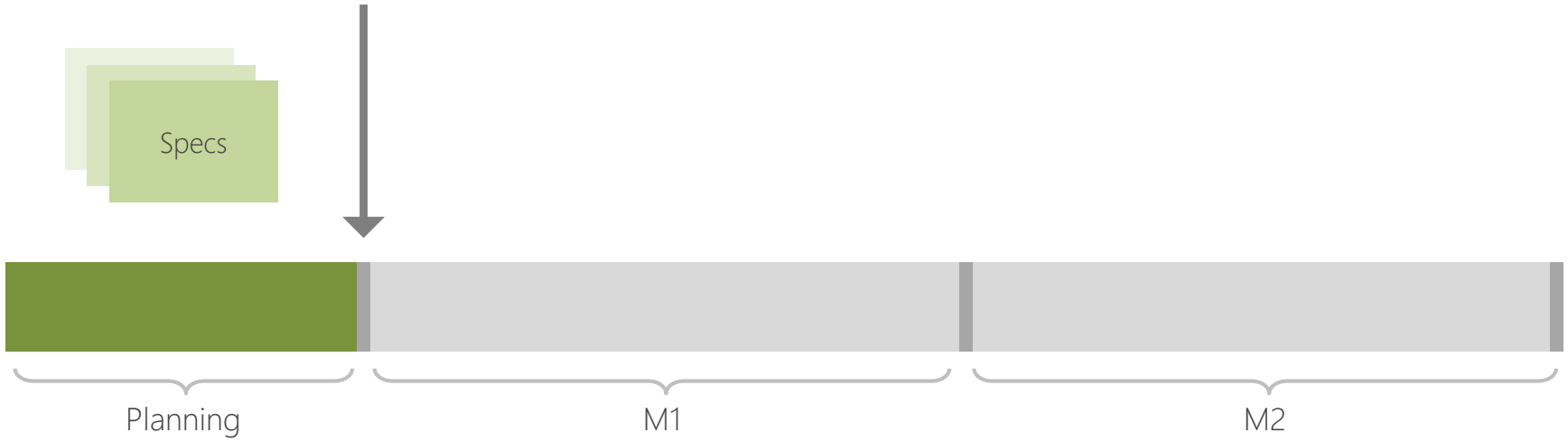
The OLD way

Microsoft

Before

The OLD way

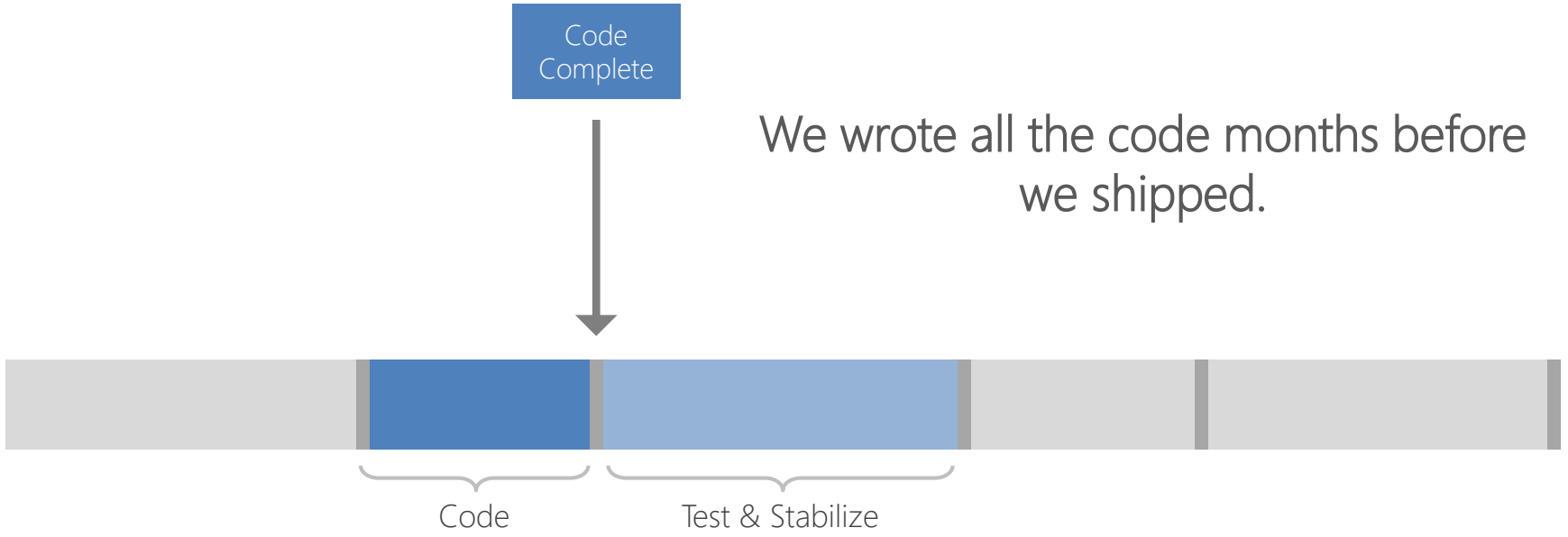
We knew exactly what to build…
and we knew it was right!



Microsoft

Before

The OLD way

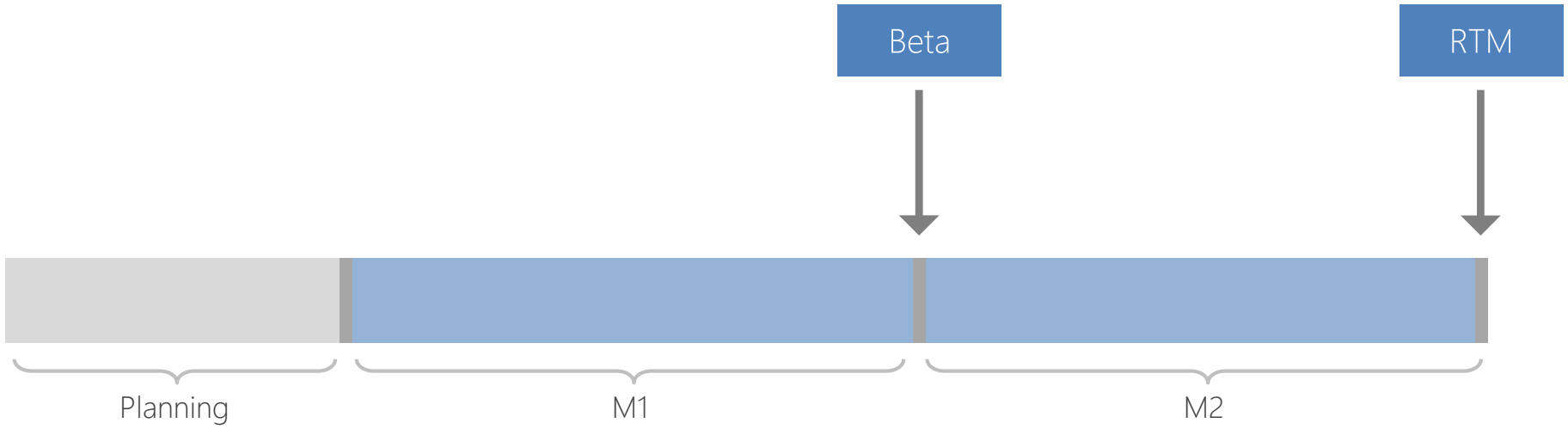


Microsoft

Before

The OLD way

We had a perfect schedule and knew exactly when it would be ready!

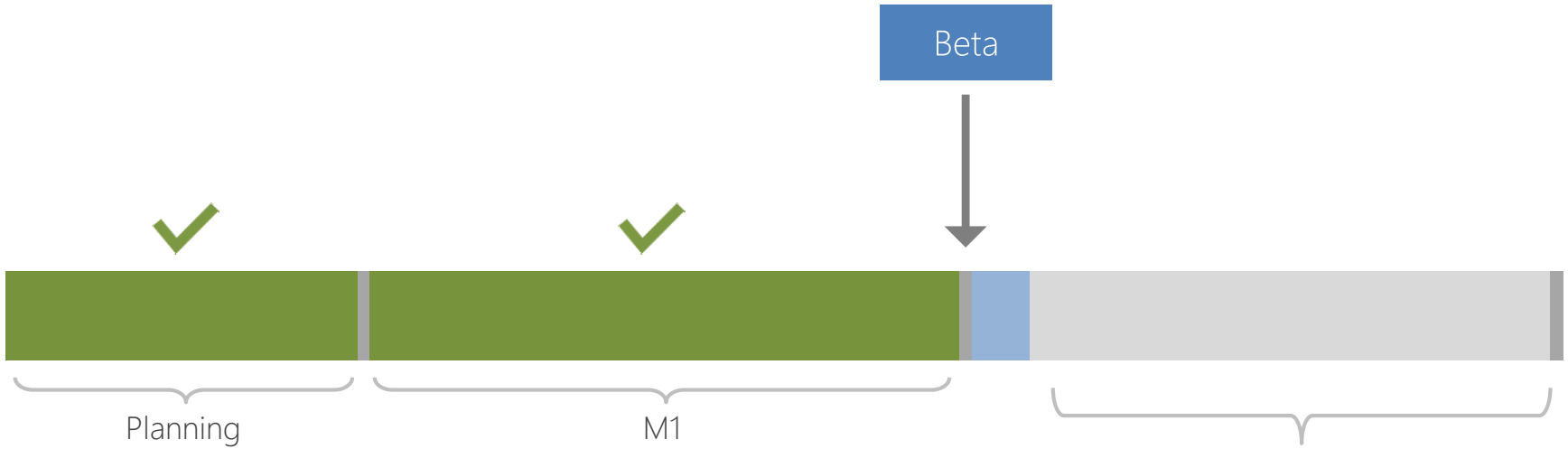


Microsoft

Before

The OLD way

Customer feedback – we should change the way a feature works. We didn't get it *quite* right...



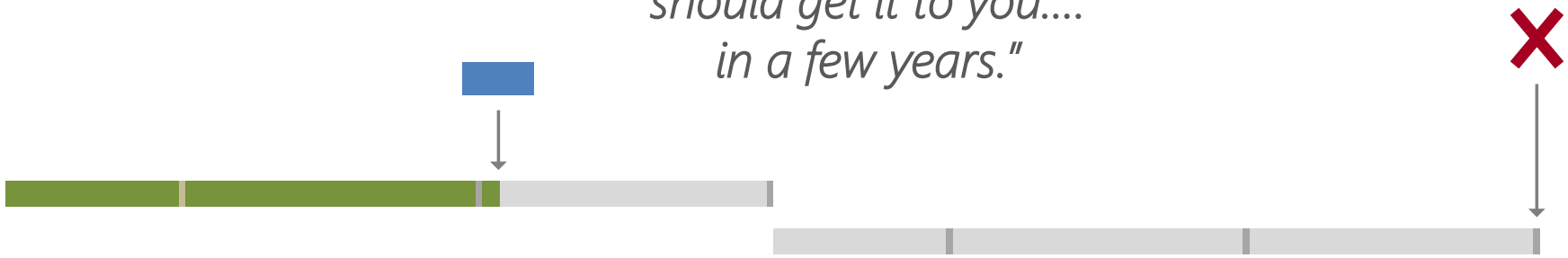
... but we're booked solid already.

Microsoft

Before

The OLD way

"Great feedback. Thanks! We'll take a look in planning for the next release. We should get it to you….
in a few years."



Microsoft

Q: How did it work?

A: Very well in the era in which it was born. But…

Times have changed!

“Firms today experience a much higher velocity of business change. Market opportunities appear or dissolve in months or weeks instead of years.”

Diego Lo Giudice and Dave West, Forrester
February 2011
Transforming Application Delivery

What changed?

Before

4-6 month milestones
Horizontal teams
Personal offices
Long planning cycles
PM, Dev, Test
Yearly customer engagement
Feature branches
20+ person teams
Secret roadmap
Bug debt
100 page spec documents
Private repositories
Deep organizational hierarchy
Success is a measure of install numbers
Features shipped once a year

After

3-week sprints
Vertical teams
Team rooms
Continual Planning & Learning
PM & Engineering
Continual customer engagement
Everyone in master
8-12 person teams
Publicly shared roadmap
Zero debt
Specs in PPT
Open source
Flattened organization hierarchy
User satisfaction determines success
Features shipped every sprint

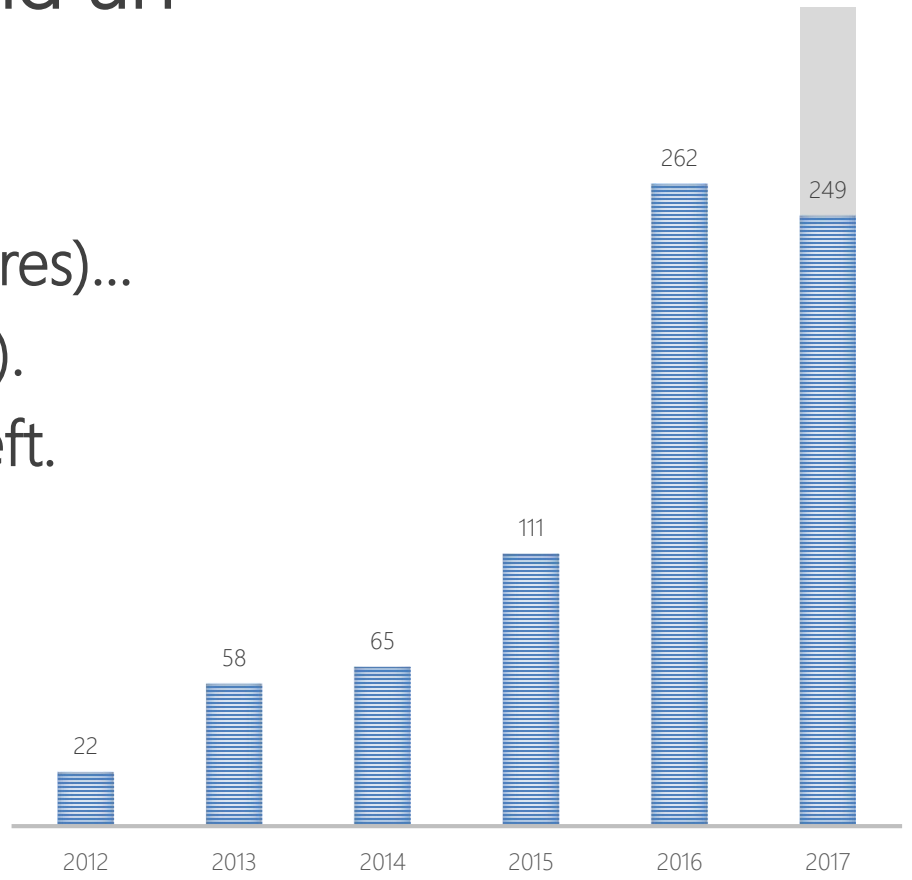
Microsoft

How do we know it's working?

Features Delivered per Year

We are delivering value to customers and an increased velocity.

- More features in the 2016 calendar year (262 features)…
- Than the previous 4 years combined (256 features).
- 249 features already in 2017… with three months left.



Microsoft

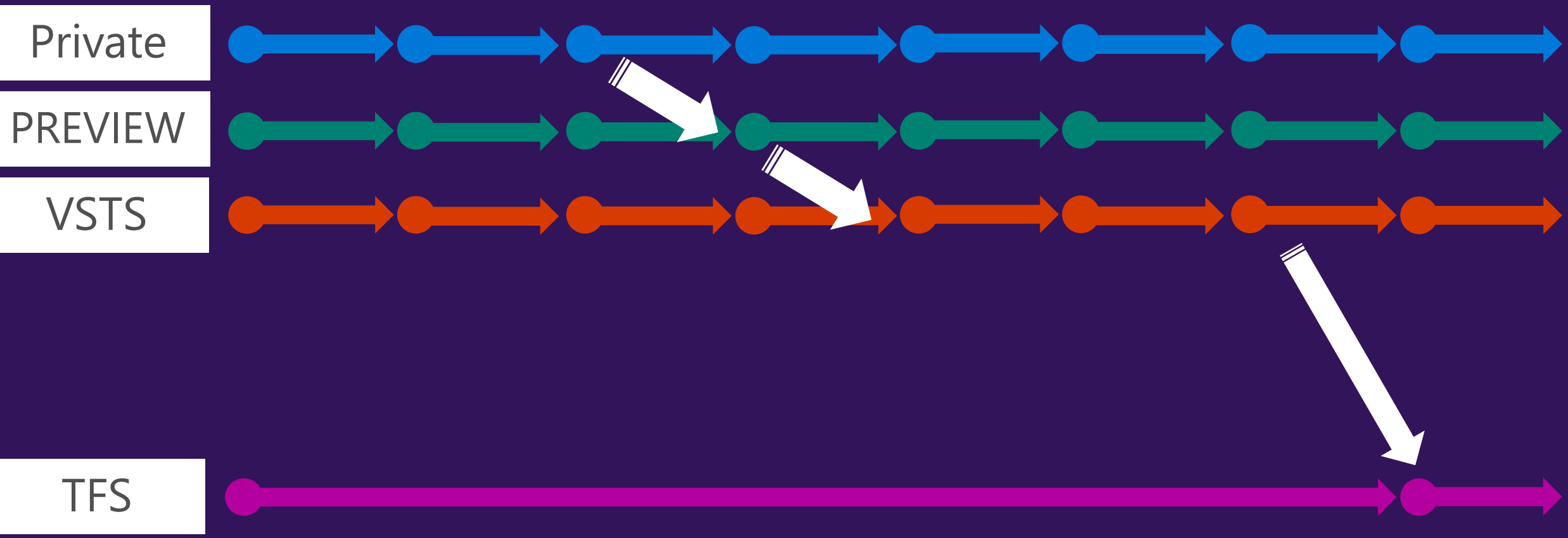
https://www.visualstudio.com/en-us/articles/news/features-timeline

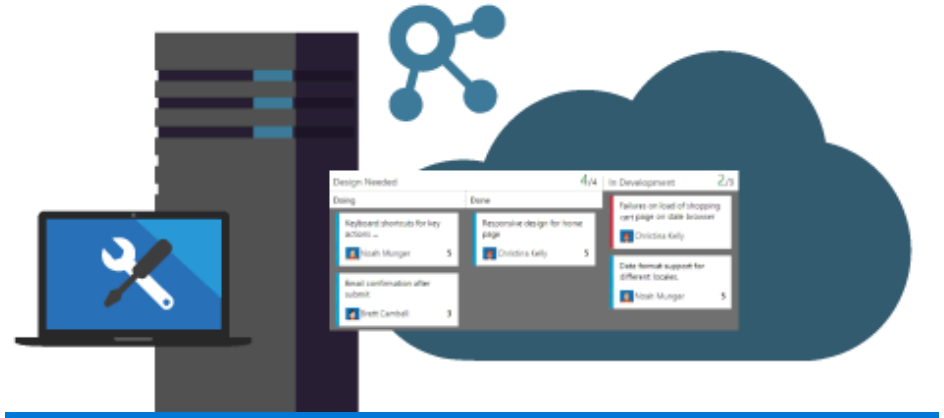
Module 0 – Business Value of DevOps

1. Get good at the science… but don’t be overly prescriptive.
2. Stop celebrating activity… start celebrating results.
3. Embrace the new normal.
4. You can’t cheat shipping.
5. Build the culture you want… and you’ll get the behavior you’re after.

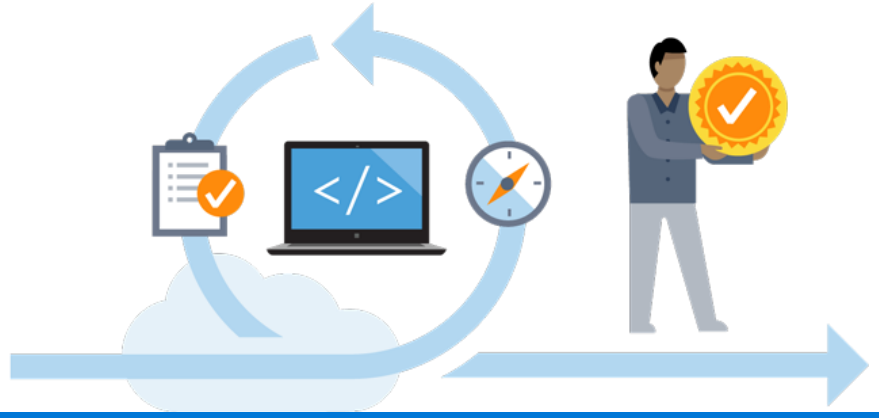
The Tools: Deliver better software, faster using VSTS

Frequent Updates

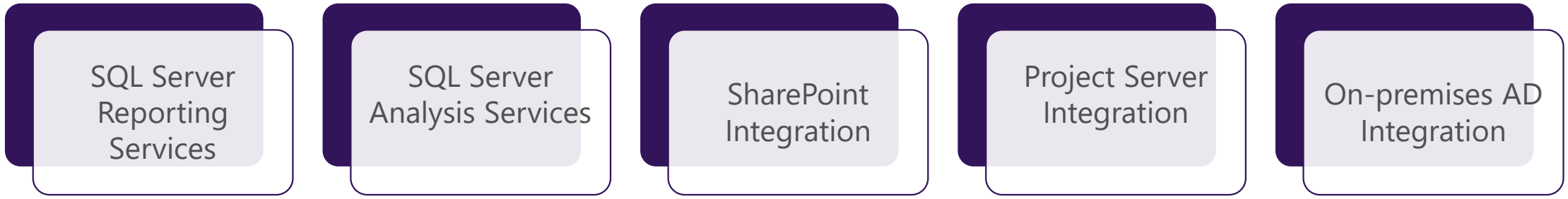




Team Foundation Server



Team Services



Services **not** available in VSTS

Services **not** available in TFS

PowerBI connector

Load testing

OData endpoint

Azure Active Directory Integration

What we will go over today in the Demo and Labs

- Setting up your projects and teams
- Adding epics, features, user stories, and bugs
- Details on editing a User Story (description, prioritization, child/dependent stories, attaching screenshots/videos, discussions)
- Bulk modification of user stories + adding templates
- Using the backlog and board
- Changing between different processes (Scrum vs Kanban)
- Watching user stories
- Customizing your Dashboard with Markdown and widgets (Reporting)
- Tagging
- Notification settings / email alerts

Demo + Lab Time!