

DevOps with VSTS

Mathieu Benoit
April 2017



Agenda

DevOps

VSTS

VSTS – Plan

VSTS – Code

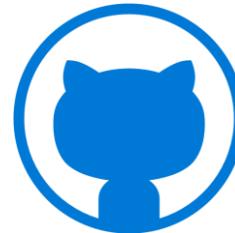
VSTS – Build

VSTS – Release

Conclusion & References

Trainer as Code

```
{  
    "firstName": "Mathieu",  
    "lastName": "Benoit",  
    "city": "Quebec",  
    "country": "Canada",  
    "company": "Microsoft",  
    "role": "Cloud Solution Architect",  
    "gitHub": "mathieu-benoit",  
    "blog": "aka.ms/mabenoit",  
    "misc": [  
        "software development",  
        "cloud + web + mobile",  
        "agile & devops enthusiast",  
        "continuous learning"  
    ]  
}
```



[GitHub account](#)



[Personal blog](#)

Do you have any experience with?

DevOps

VSTS

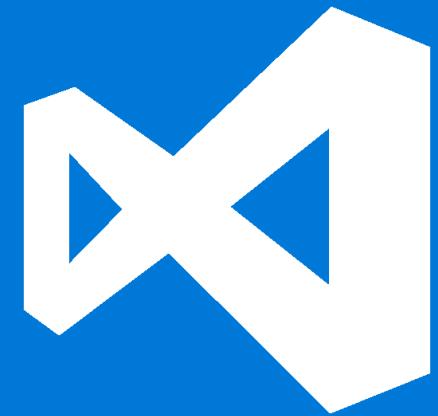
TFS

Git

Automate Testing

Azure

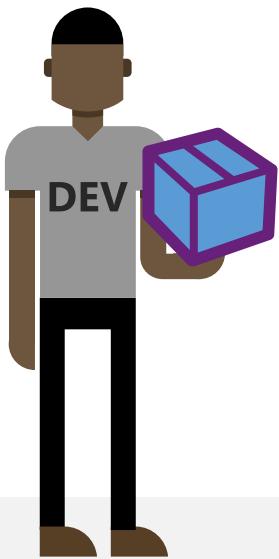
Which other ALM/CI/CD tools do you use?



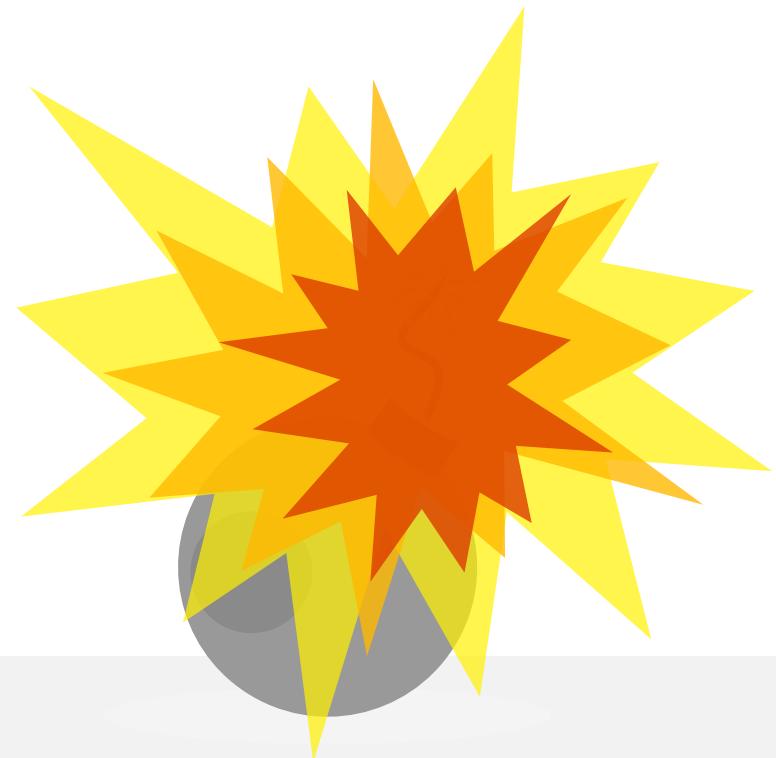
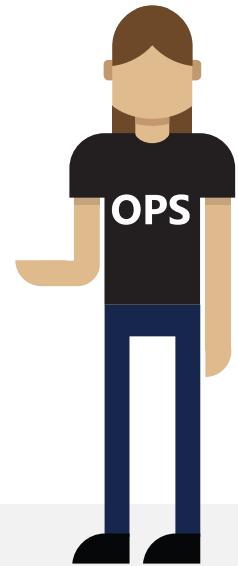
DevOps

Traditional Dev and Ops

Measured for abilities
to deliver CHANGE



Measured for abilities
to deliver RELIABILITY



“DevOps is
development
and operations
collaboration”

“DevOps
is using
automation”

“DevOps
is **small**
deployments”

“DevOps is
treating your
infrastructure
as code”

“DevOps
is feature
switches”

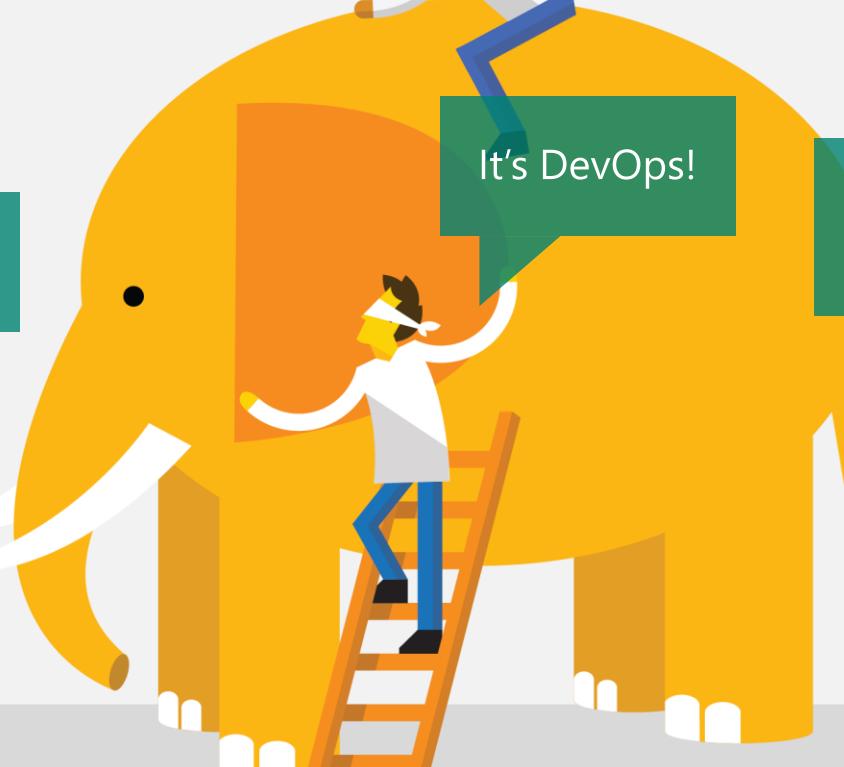
“Kanban
for Ops?”

It's DevOps!

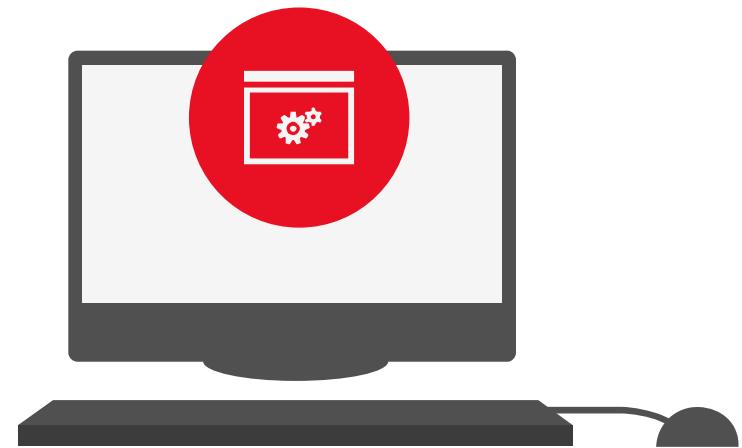
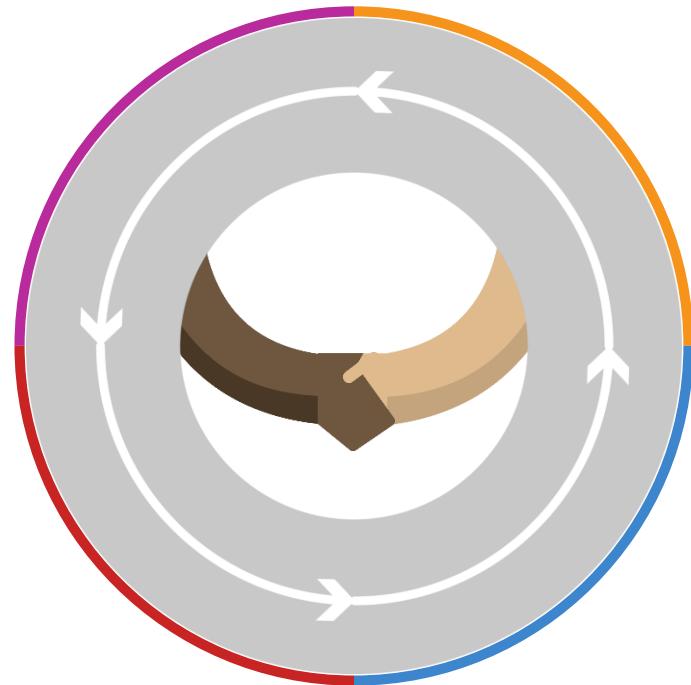
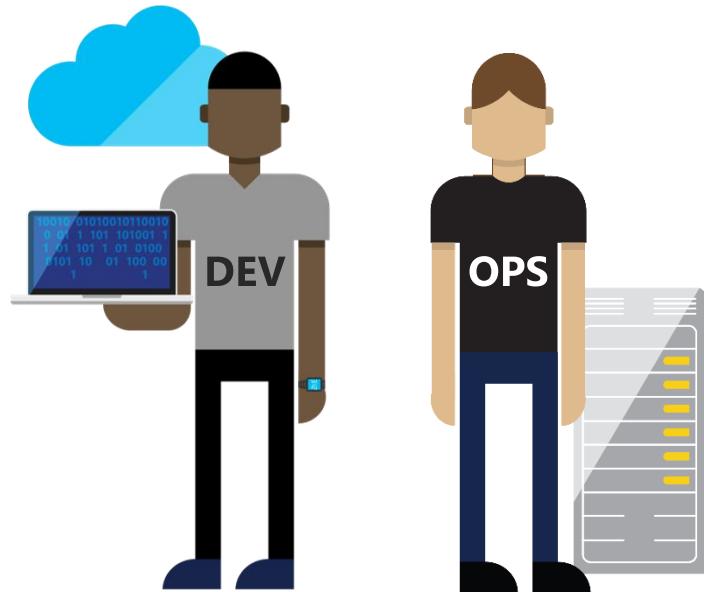
It's DevOps!

It's DevOps!

It's DevOps!



DevOps: the 3 stage conversation



1 | People

2 | Process

3 | Products

Assumptions about DevOps

- We all do « some parts » of DevOps!
- It's not magic
- Deliver often and fast with high value for end-user
- Evolution of current and existing practices
- Its' continuous learning
- It's not a methodology
- It's not only « Dev & Ops », many stakeholders!

5 pillars

Culture

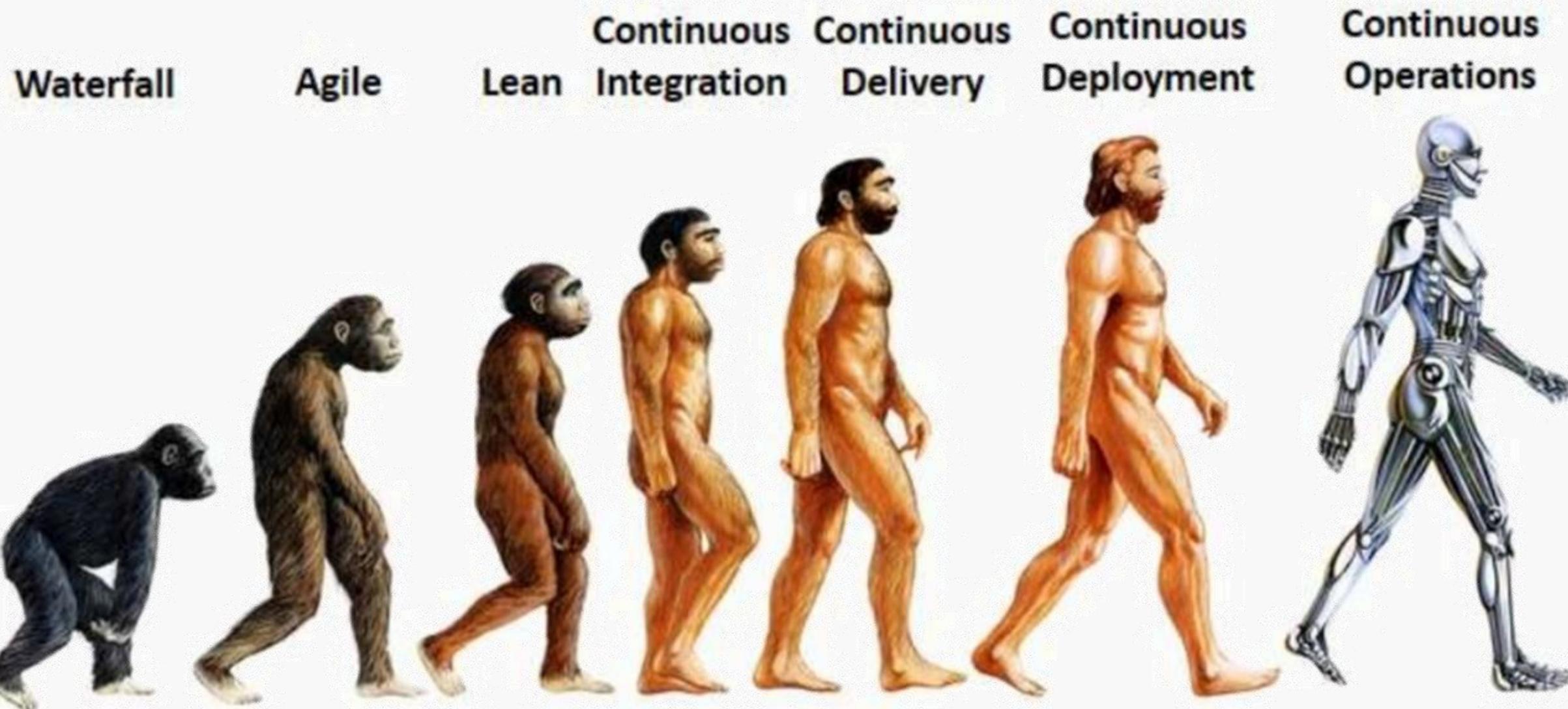
Automation

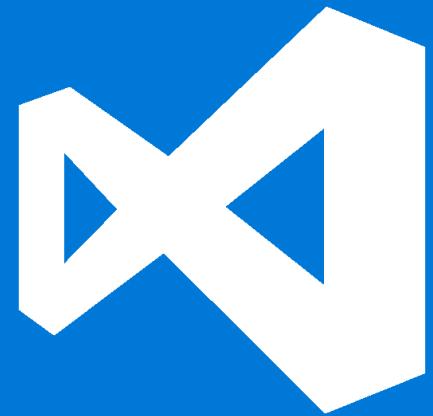
Lean

Metrics

Collaboration

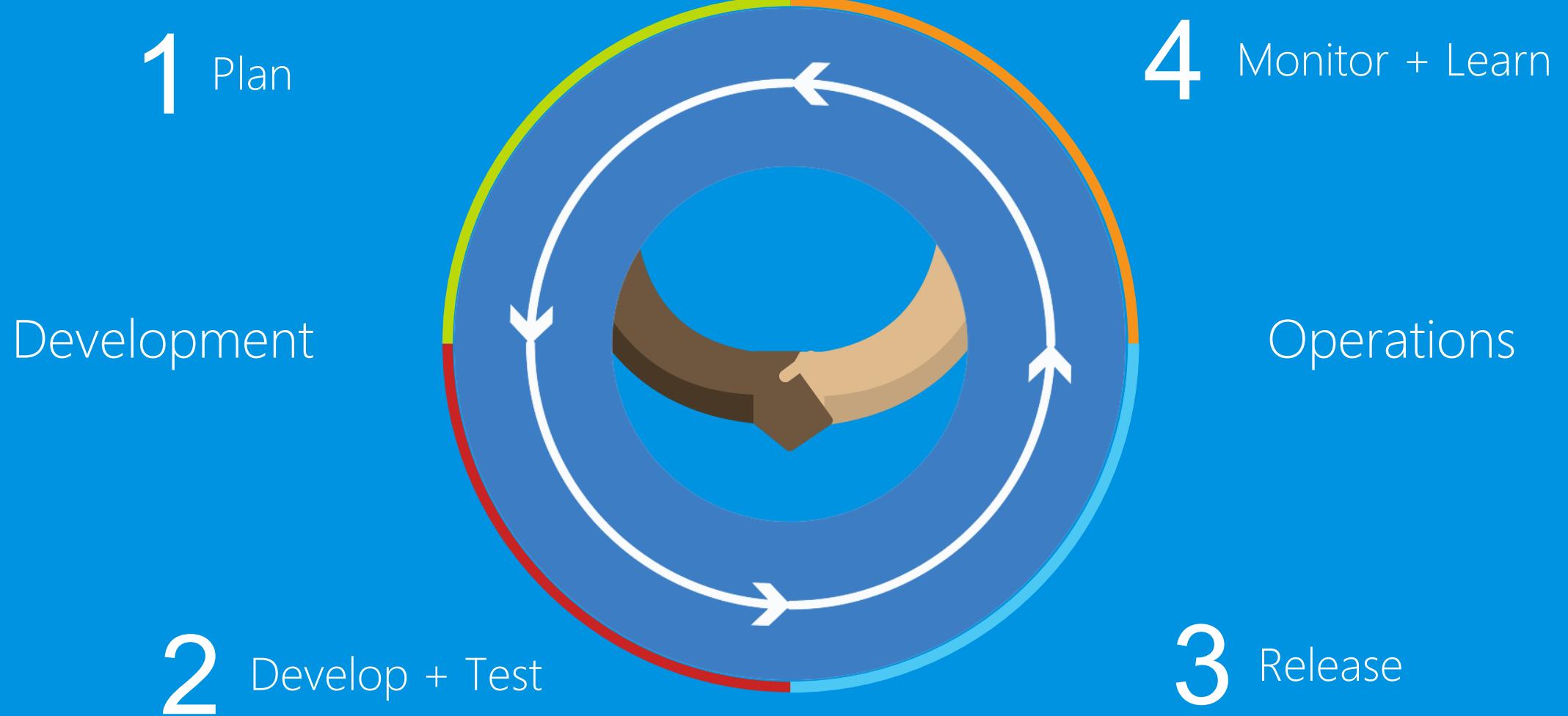
DevOps Movement



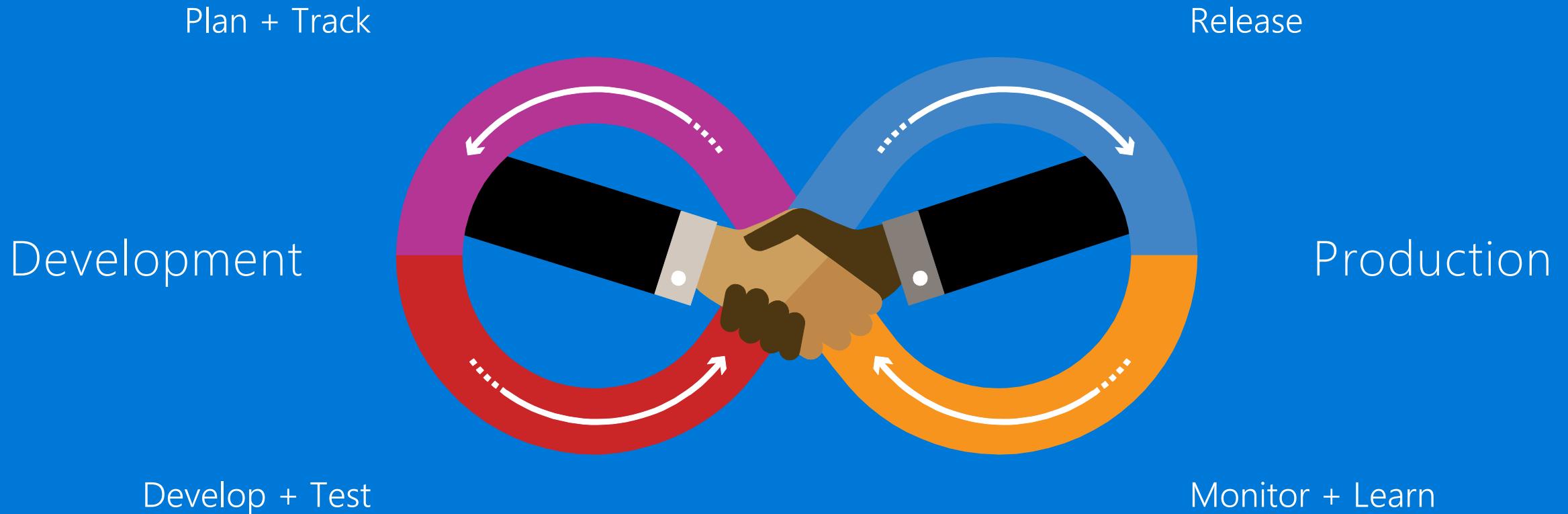


Your ALM/DevOps
with VSTS

DevOps is ALM first



The converged DevOps lifecycle

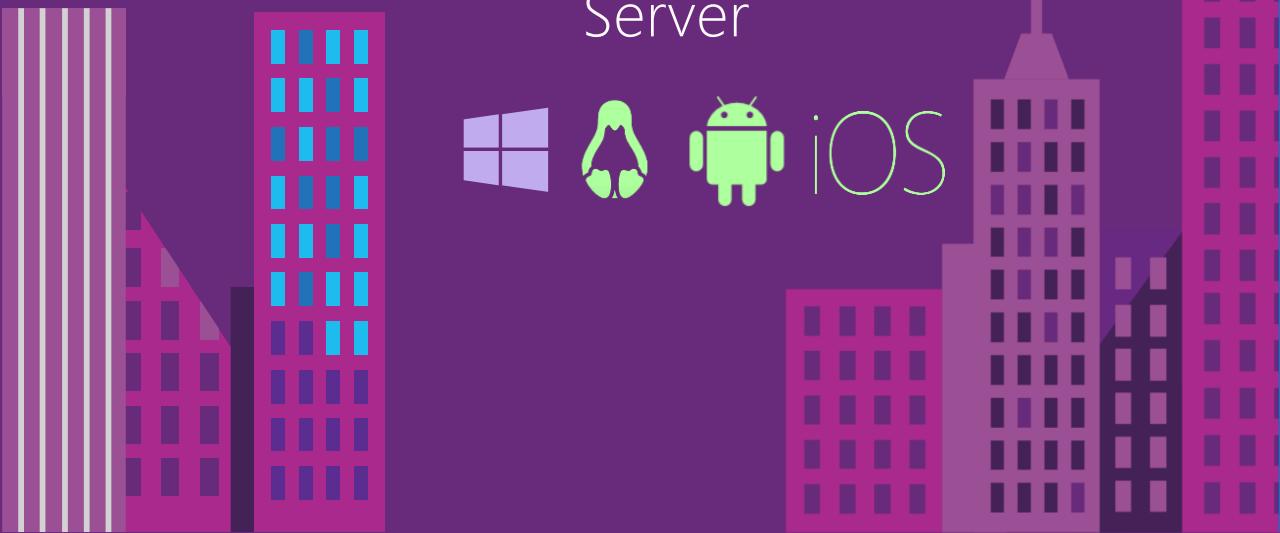




Visual Studio
Team Services



Team Foundation
Server



Plan & Track Work

Source Code Management

Package Management

Quality Management

Cross-platform Build

Continuous Deployment

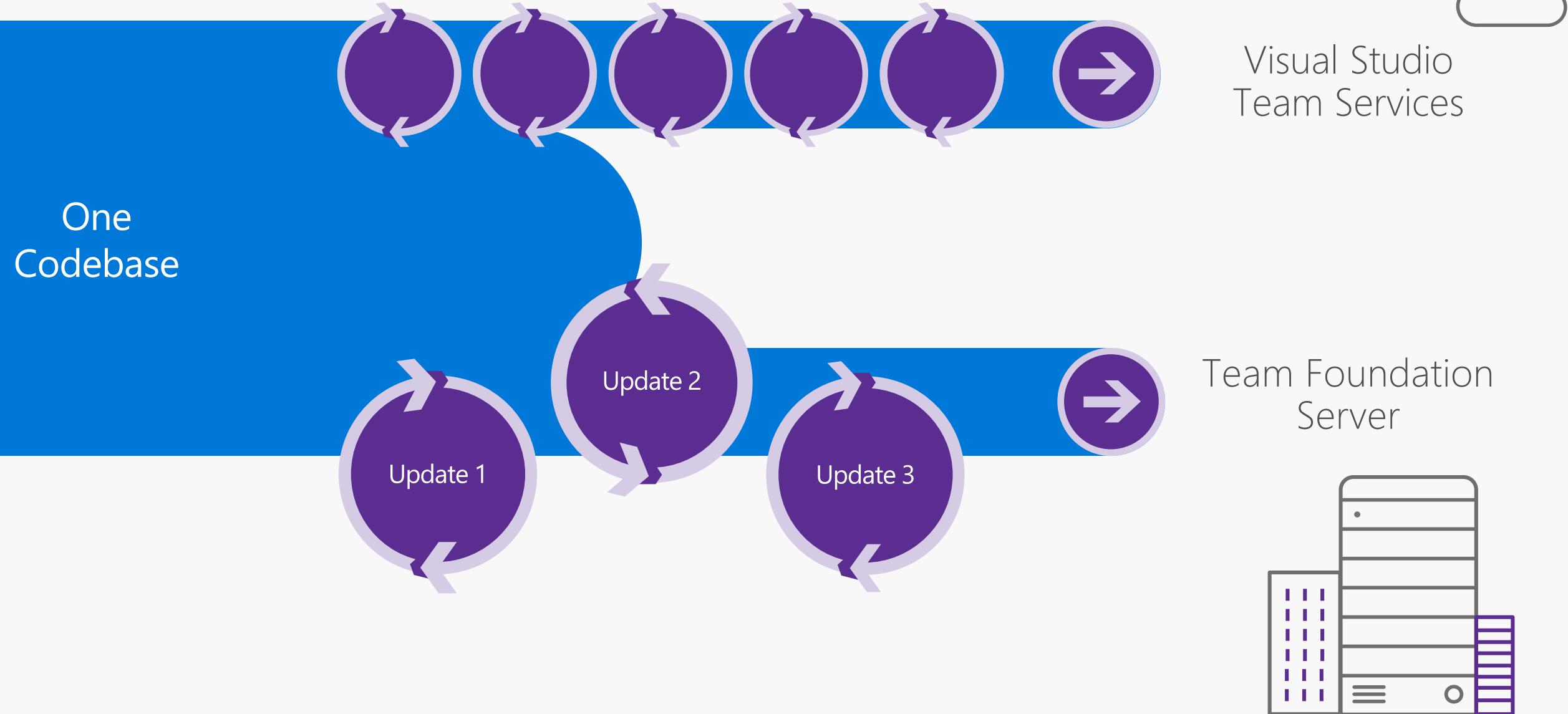
Release Management

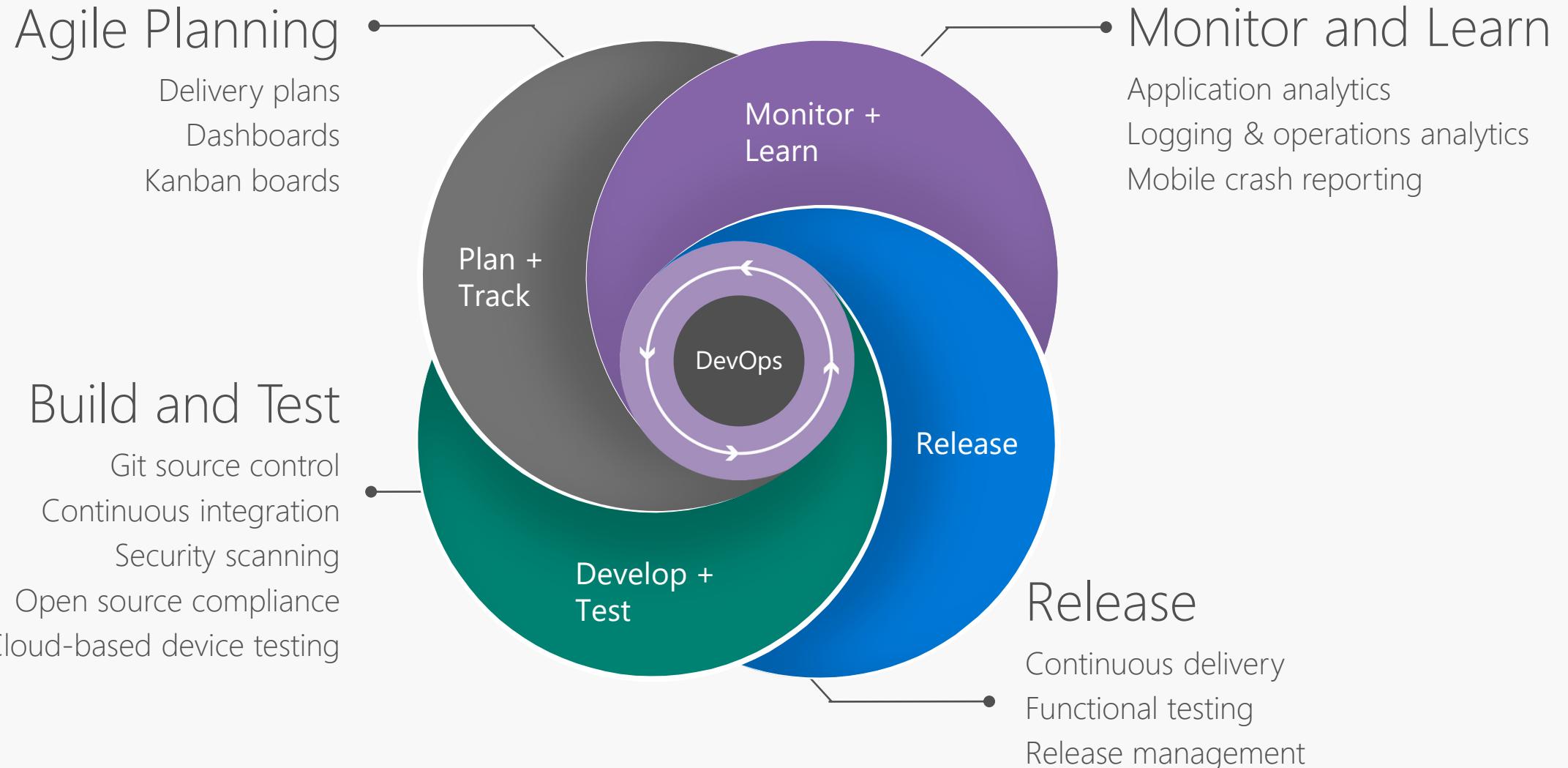
Feedback Management

Application Telemetry

Extend and Customize

One codebase, two delivery streams





Microsoft DevOps Solution

[Visual Studio Team Services](#)

[Team Foundation Server](#)

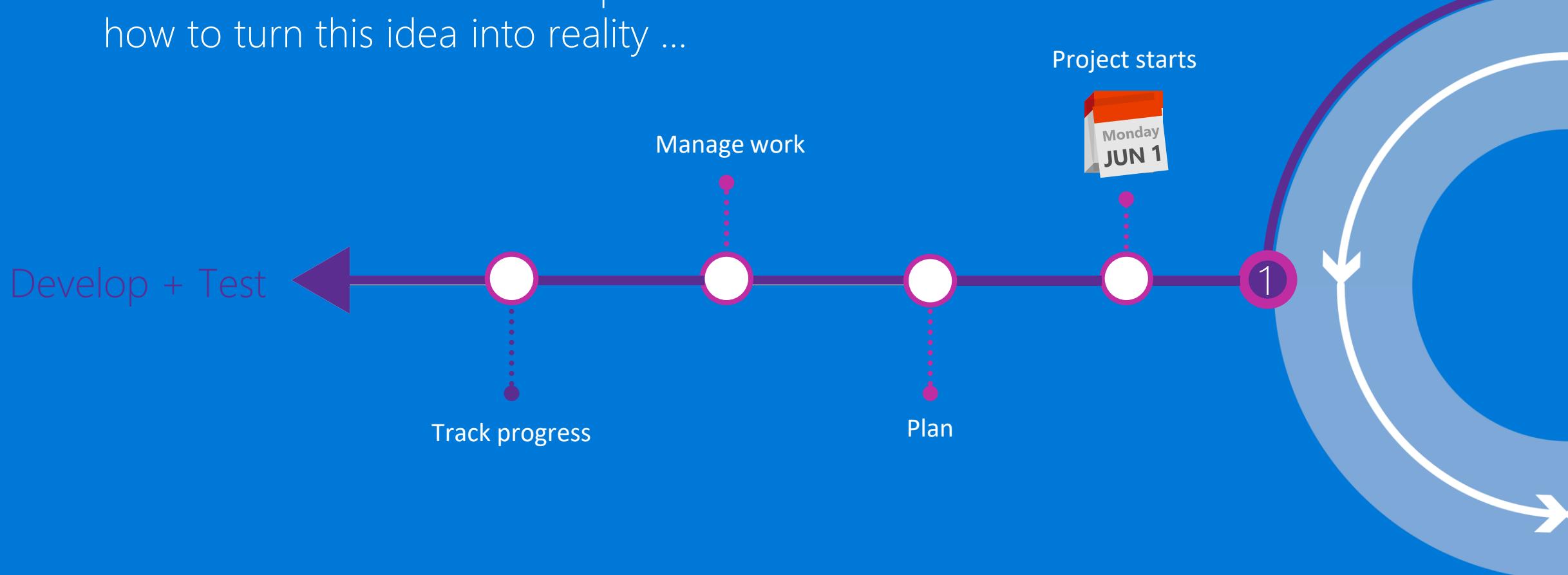
[Application Insights](#)

[Visual Studio Mobile Center](#)

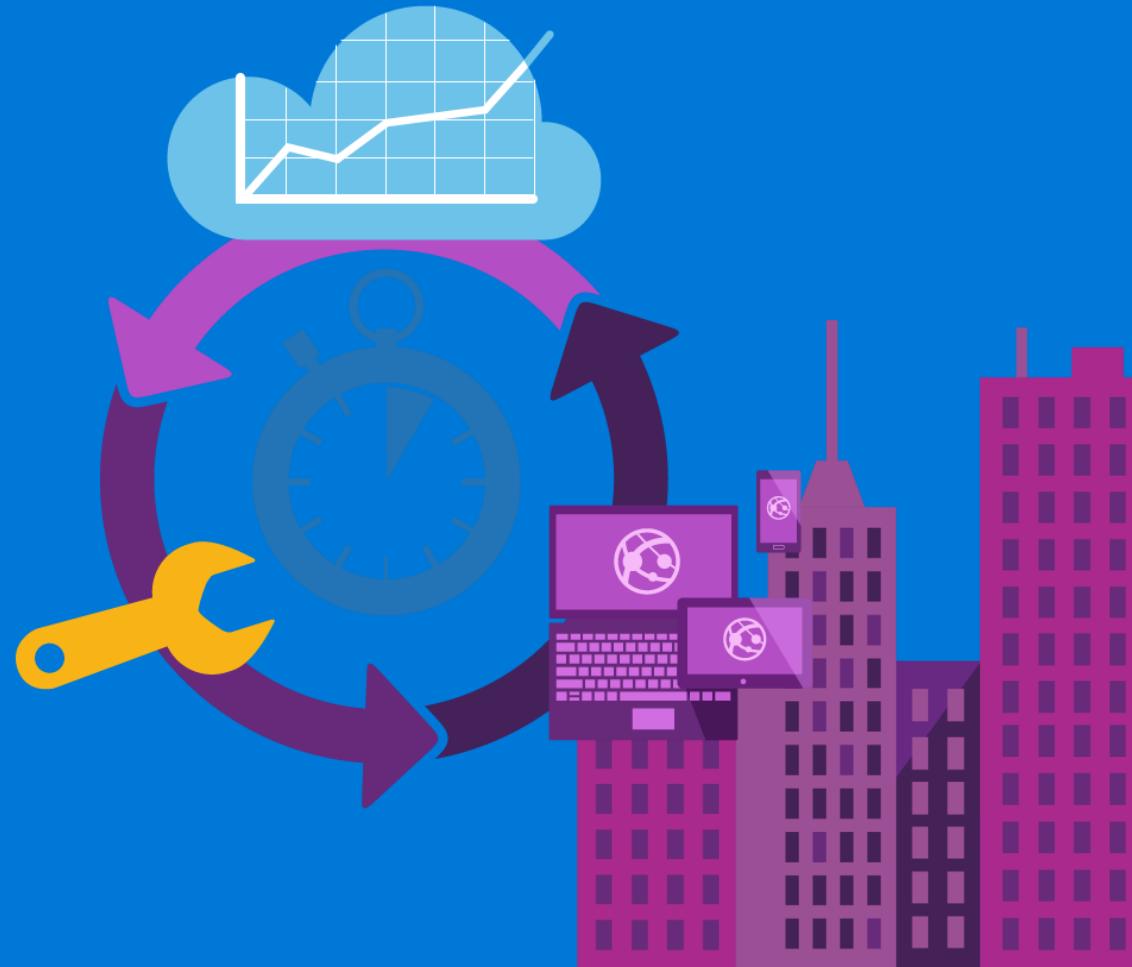


Plan

It starts with an idea – and a plan
how to turn this idea into reality ...



Demo: VSTS Plan



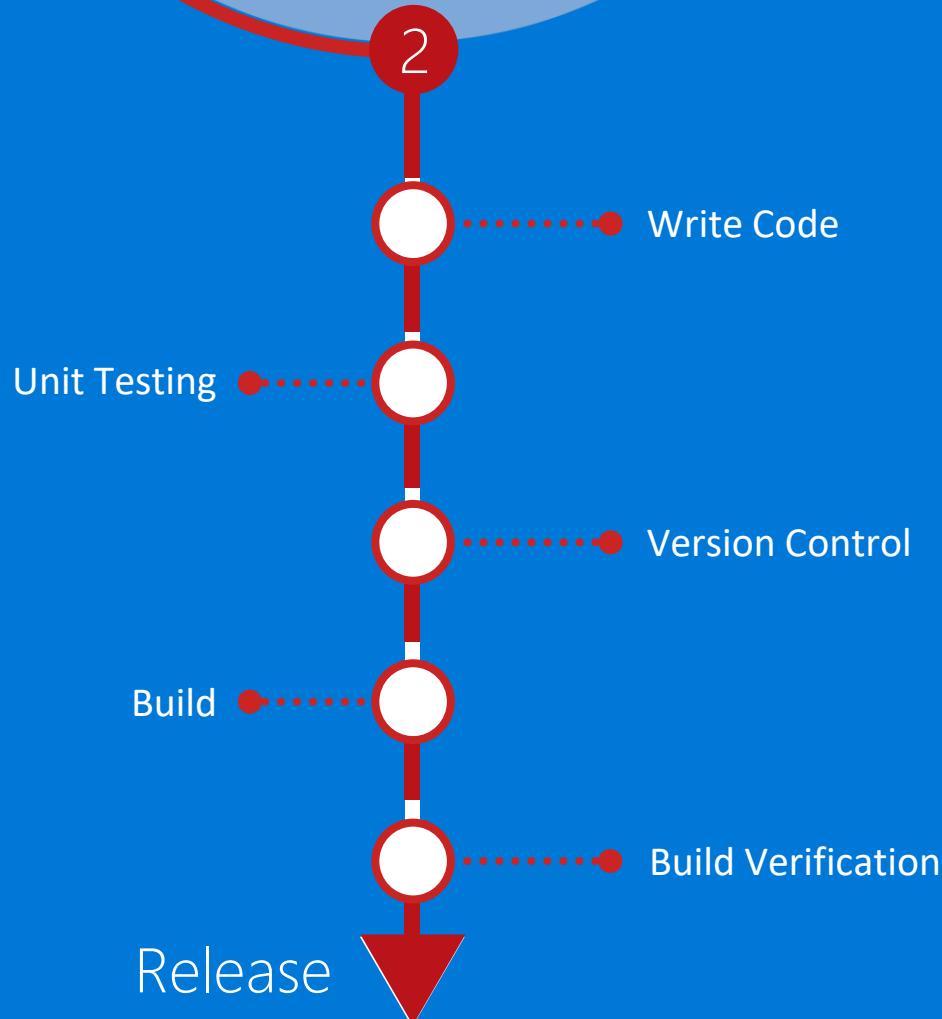
Demo: VSTS Plan

- Work items
- Related items links
- Definition of done
- Discussions (with @mention)
- Related work, development (links to commits)
- Visualize work item (graphical view of links and related items - work item visualization add-in from marketplace)
- Kanban board
- Styling rules for card (red if one task/work item is not taken within 2 days)
- Charts and Queries

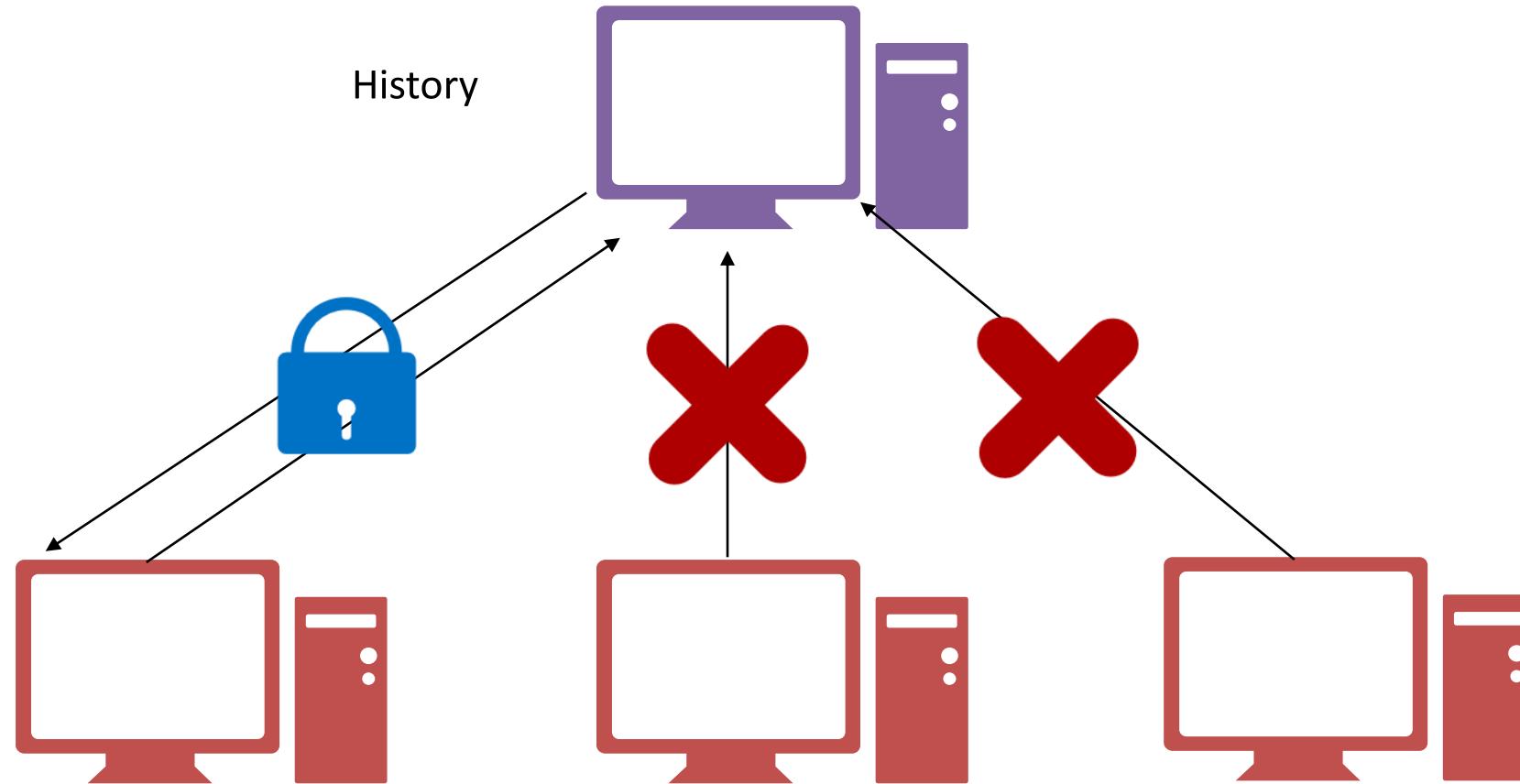


Develop, Build, Test

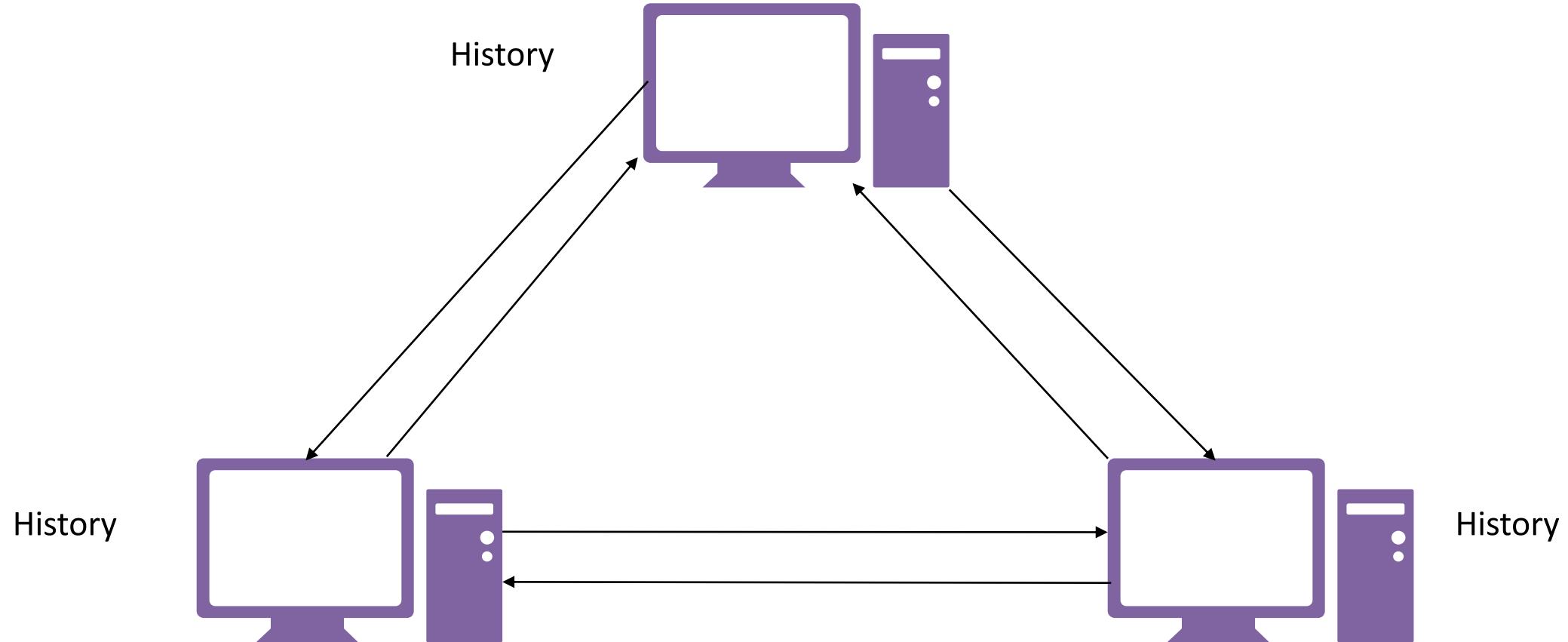
Once the iteration starts, developers turn great ideas into features ...



Centralized Version Control - Locking



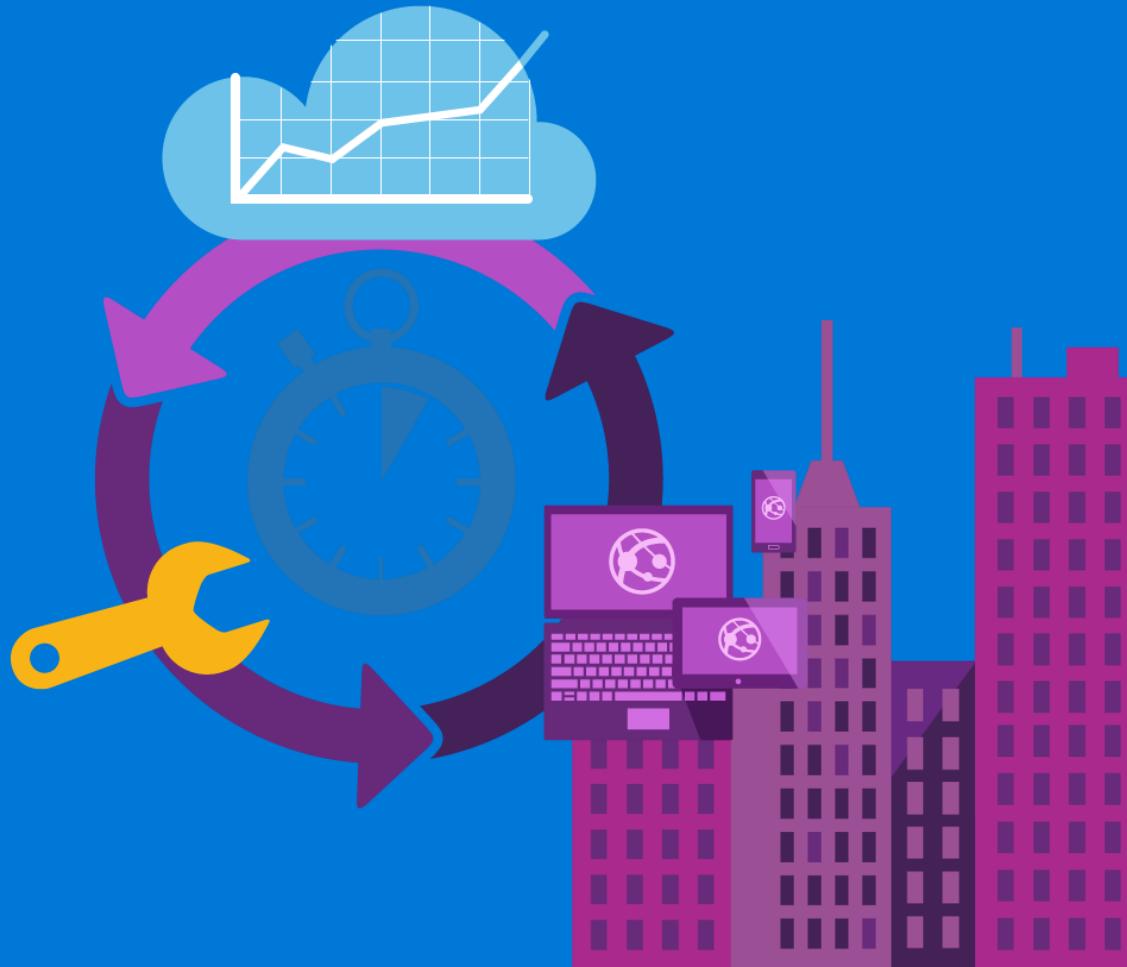
Distributed Version Control



Comparison: Centralized versus Distributed

	Centralized	Distributed
Hard drive space required for history?	No	Yes
Who has latest revision?	Central “master version”	Variable, need to make rules
Where is full history?	Central machine	Local machine
Work offline?	No*	Yes
How fast are operations?	Network-dependent	Fast, most are local
Branching and merging?	Reliable, use w/ caution	Reliable, used often
Learning curve	Relatively simple	Relatively hard

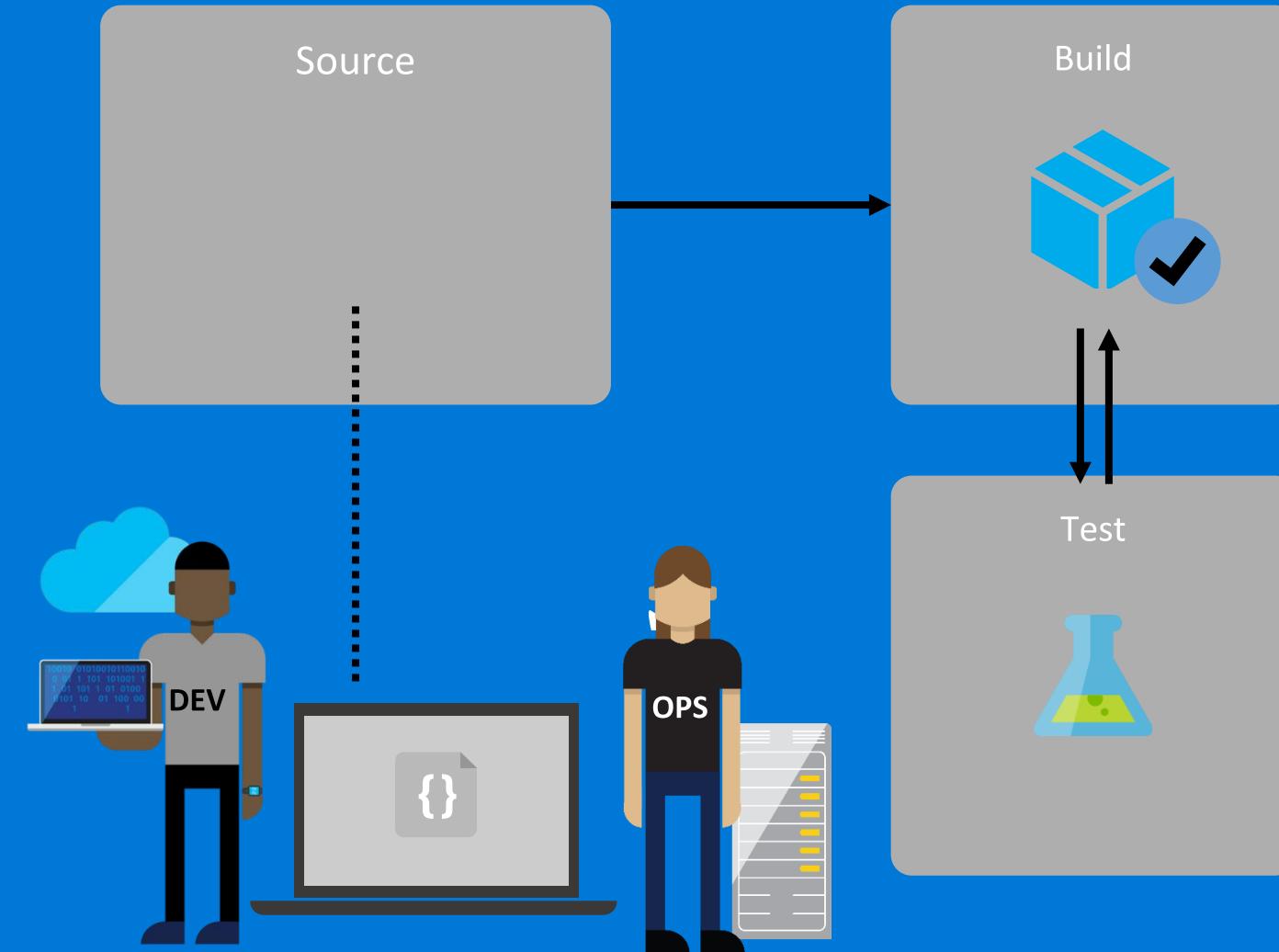
Demo: VSTS Code



Demo: VSTS Code

- TFVC or Git - centralized versus decentralized repo
- Multi repositories per project
- Online editor
- History
- Discussion on changings, branch,
- Pull Request for review, cherry-pick, revert, etc.
- Integration with Visual Studio, Eclipse, Android Studio

Continuous Integration



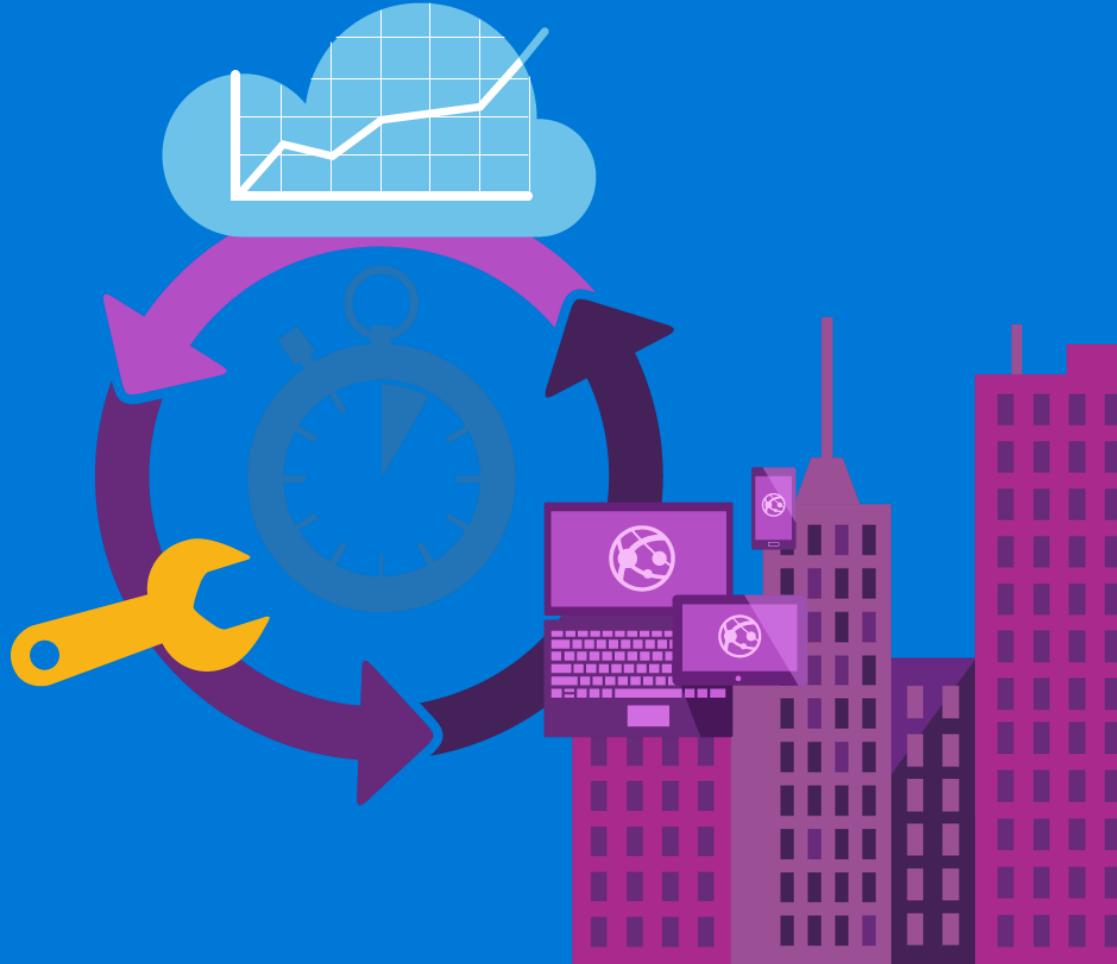
Value

- Accelerate Delivery
- Repeatability
- Optimized Resources

Measure

- More frequent releases
- MTTR
- MTTD

Demo: VSTS Build



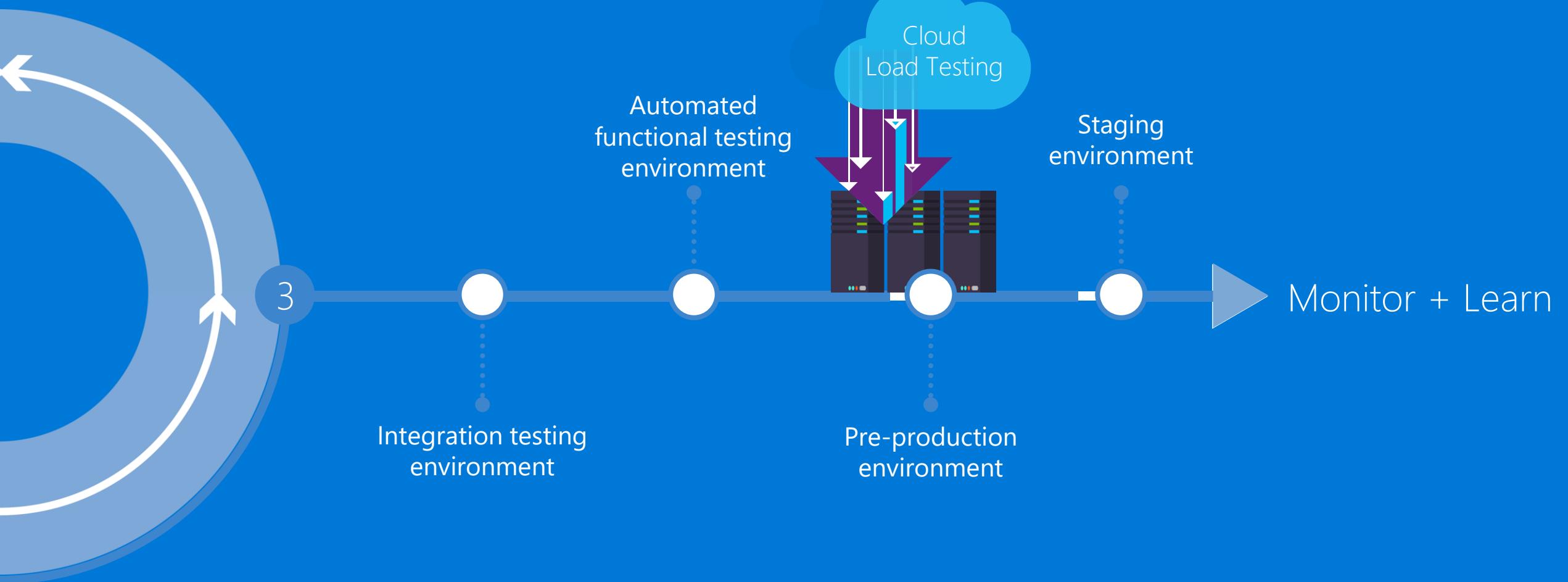
Demo: VSTS Build

- Build Definition – Continuous Integration
- Source code integration: VSTS, GitHub, Subversion, Remote Git
- Tasks
- Summary results, unit test integrated, associated commits, associated deployments,
- Console, logs,
- Marketplace (as devs you could build your own)
- History of the build definition modification by itself (json file)

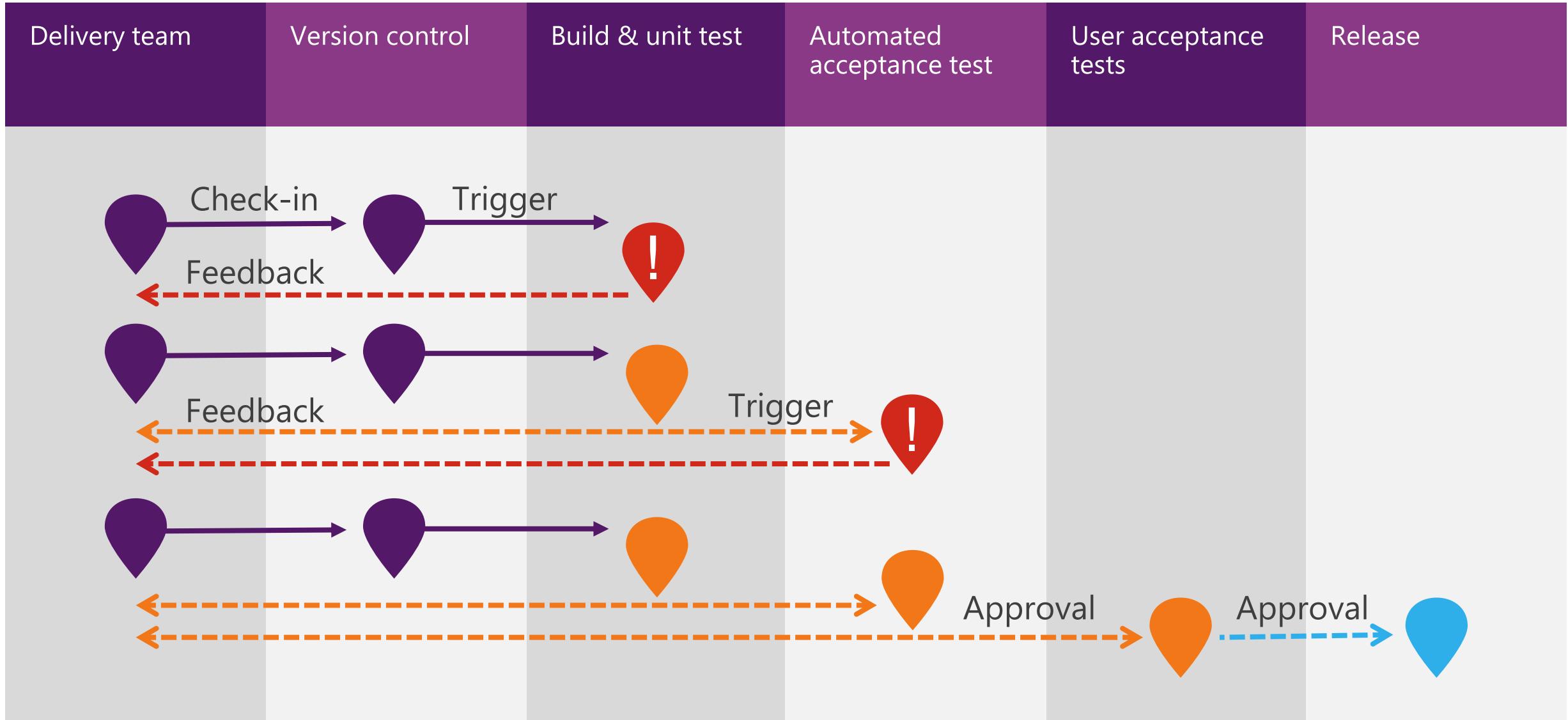


Release

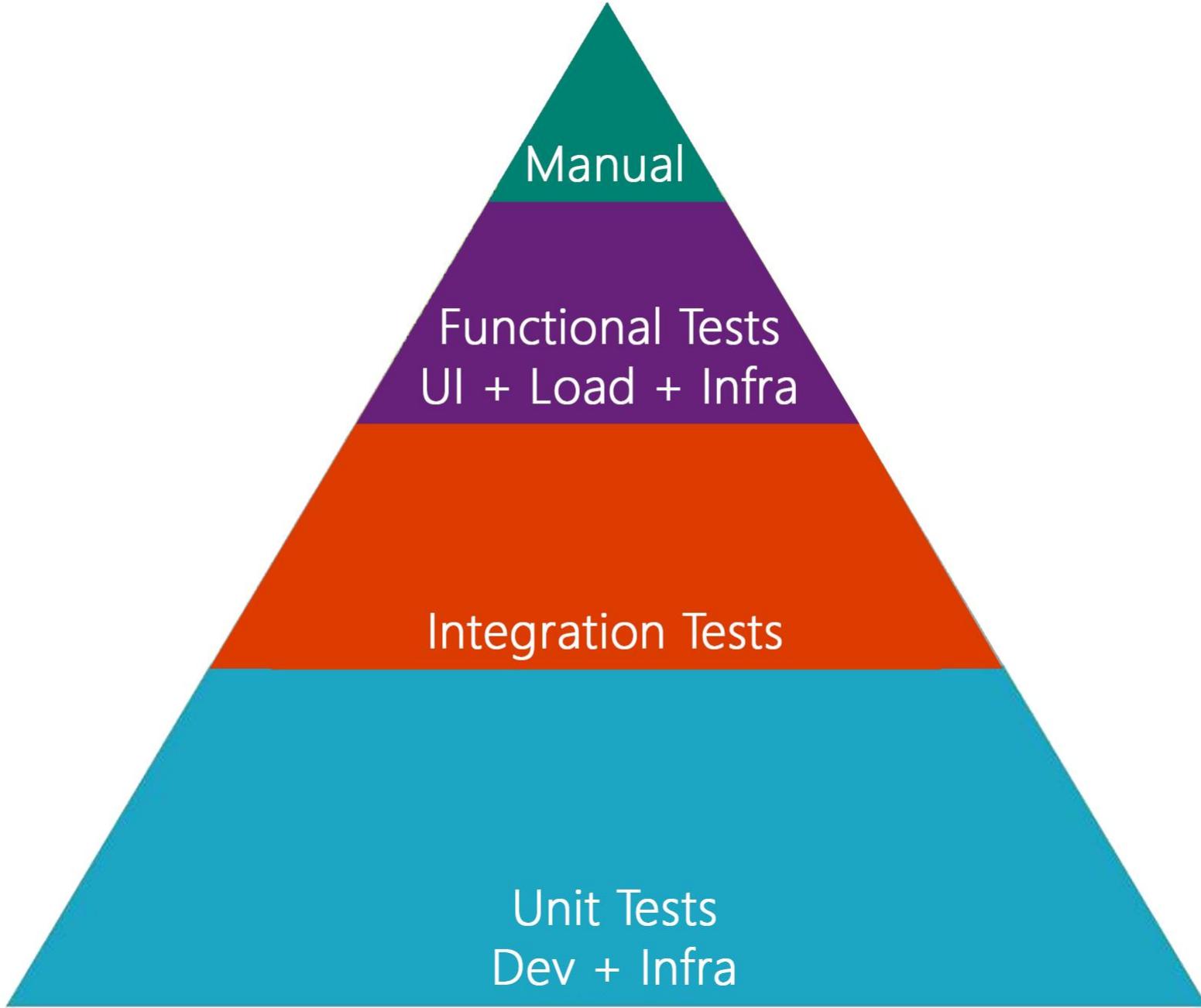
When unit tests pass, the build is deployed and tested for each stage in the release process



Continuous delivery process



Tests



Infrastructure as Code

- Automation
- Script as Code
- Script as Documentation
- Included in the “Definition of Done”
- Repeatable
- ARM Templates:
 - Control Access (with RBAC)
 - Store Secrets Securely (with KeyVault)
 - Enforce Policy (with Resource Policy)
 - Log, log, log (with Audit Logs and Deployment Operations)
 - Protect (with Resource Locks)

Xamarin Test Cloud

2,000+ devices

Automate your app testing and run them on over 2,000 different real devices. Test everything users do, as well as any performance problems with step-by-step memory and performance tracking.

Real devices, real quality

- Ensure higher quality by testing on real devices
- Automate app testing on over 2,000 real devices
- Use C#, Ruby, or Cucumber to build automated tests
- Integrates with any continuous integration system

The screenshot shows the Xamarin Test Cloud interface. At the top, it displays the URL "testcloud.xamarin.com" and the date "Aug 23, 2015 10:03:59 PM". Below this, the main area has a header "Xamarin test cloud" with a navigation path: "Xamarin CRM" > "master" > "Aug 23, 2015 10:03:59 PM". The main content area is titled "Overview" and shows a table of test results. A specific test case, "Then I tap 'Sales'", is highlighted with a blue background and a red lightning bolt icon, indicating it failed. The table includes columns for "TEST RESULTS" (with a dropdown arrow), "NAME", and "STATUS" (indicated by green checkmarks or red lightning bolts). To the right of the table, there is a grid of 12 smartphone icons, each showing a different screen of a mobile application. The devices listed are Apple iPhone 5C, Apple iPhone 5, Apple iPhone 5C, Apple iPhone 6, Apple iPhone 5S, Apple iPhone 6, Apple iPhone 5, Apple iPhone 5C, Apple iPhone 5, Apple iPhone 5S, Apple iPhone 5C, and Apple iPhone 5. Each device icon shows a different stage of the "Sales" test being executed.



for iOS

Bringing DevOps to mobile apps

Distribute your app

Collect crash reports

Gather user feedback

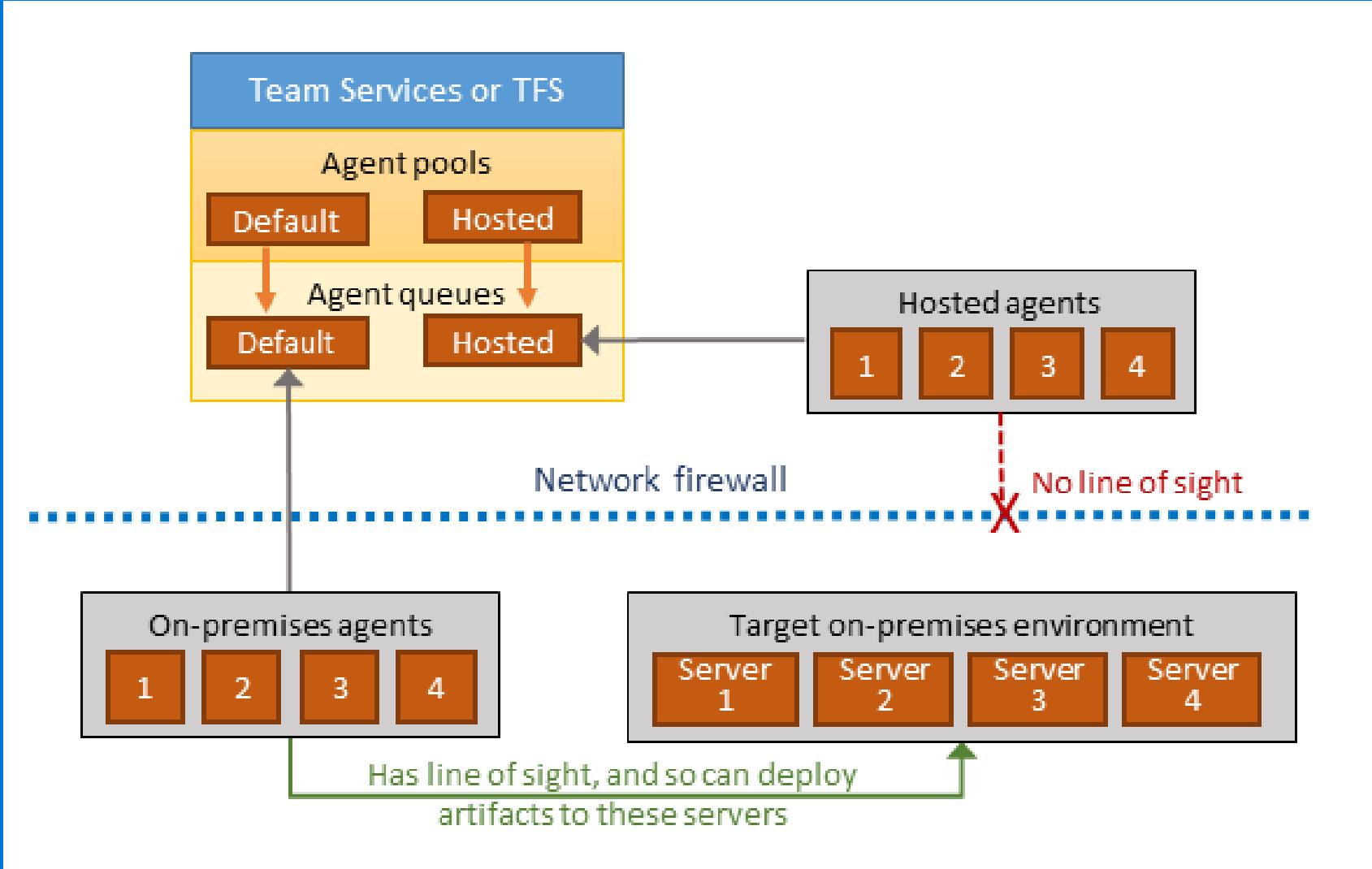
Understand app adoption

Track app usage

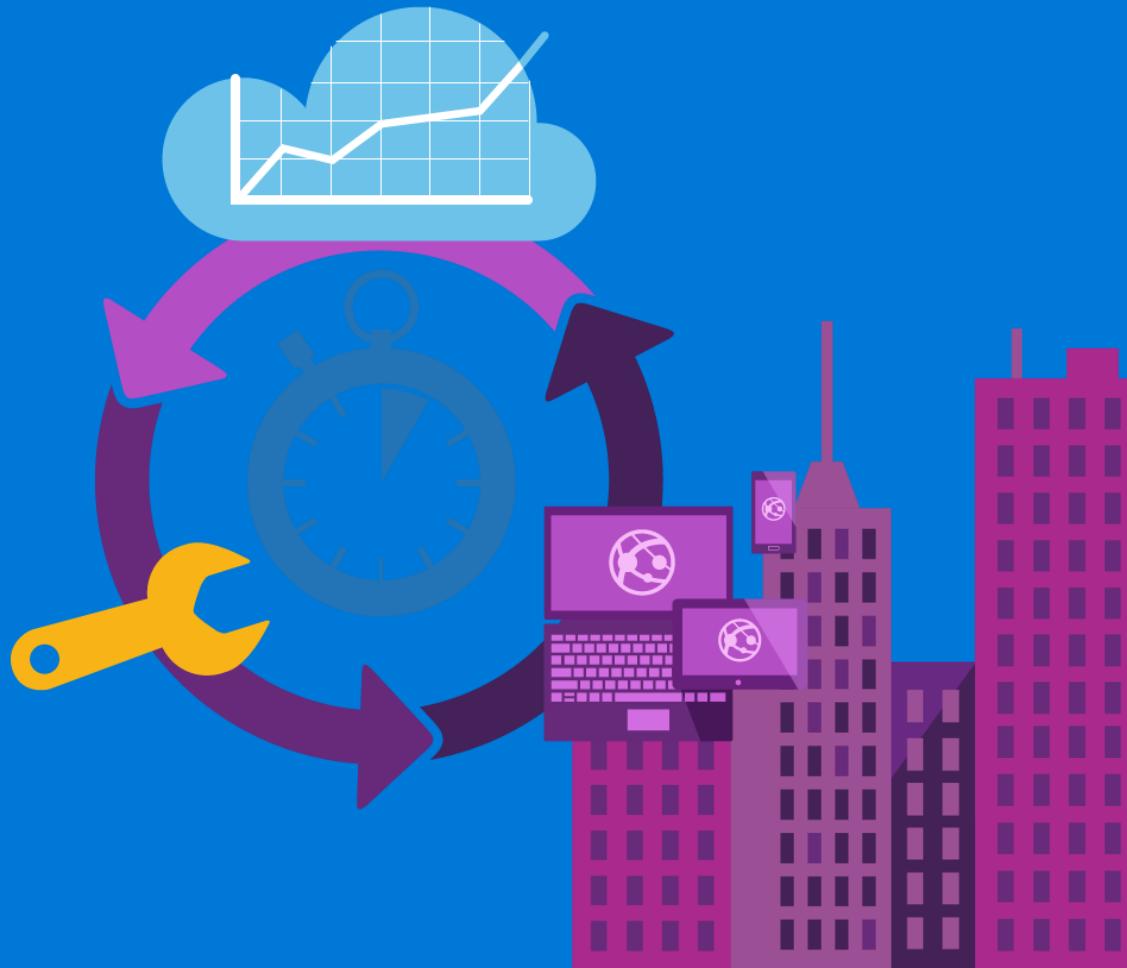
The screenshot shows the HockeyApp web interface at <https://rink.hockeyapp.net/manage/apps/>. The page displays a crash group for 'HockeyApp for Android' (Android | Live) in 'Version 2.5.1 (334)'. The 'Crash Group' tab is selected. A prominent error message is shown: **net.hockeyapp.client.HomeActivity\$3.onErrorResponse** in **HomeActivity.java**, line 267. The reason for the crash is listed as **java.lang.NullPointerException**. Below this, key statistics are displayed: **Number of Crashes: 132**, **Percentage for Version: 20.89%**, and **First Crash:** (partially visible). At the bottom, tabs for **Stacktrace**, **Histogram**, **Devices**, and **OS Versions** are shown. The **Stacktrace** tab is active, displaying the following exception backtrace:

```
1 net.hockeyapp.client.HomeActivity$3.onErrorResponse
2 com.android.volley.Request.deliverError
3 com.android.volley.ExecutorDelivery$ResponseDeliveryRunnable.run
4 android.os.Handler.handleCallback
5 android.os.Handler.dispatchMessage
6 android.os.Looper.loop
7 android.app.ActivityThread.main
8 java.lang.reflect.Method.invokeNative
```

Deploying to On-Premises Environments with Visual Studio Team Services or Team Foundation Server



Demo: VSTS Release



Demo: VSTS Release

Release

- Release Definition
- Build link (artifacts) – VSTS or Jenkins
- Tasks
- Environments
- Workflow, list of approvers, etc.

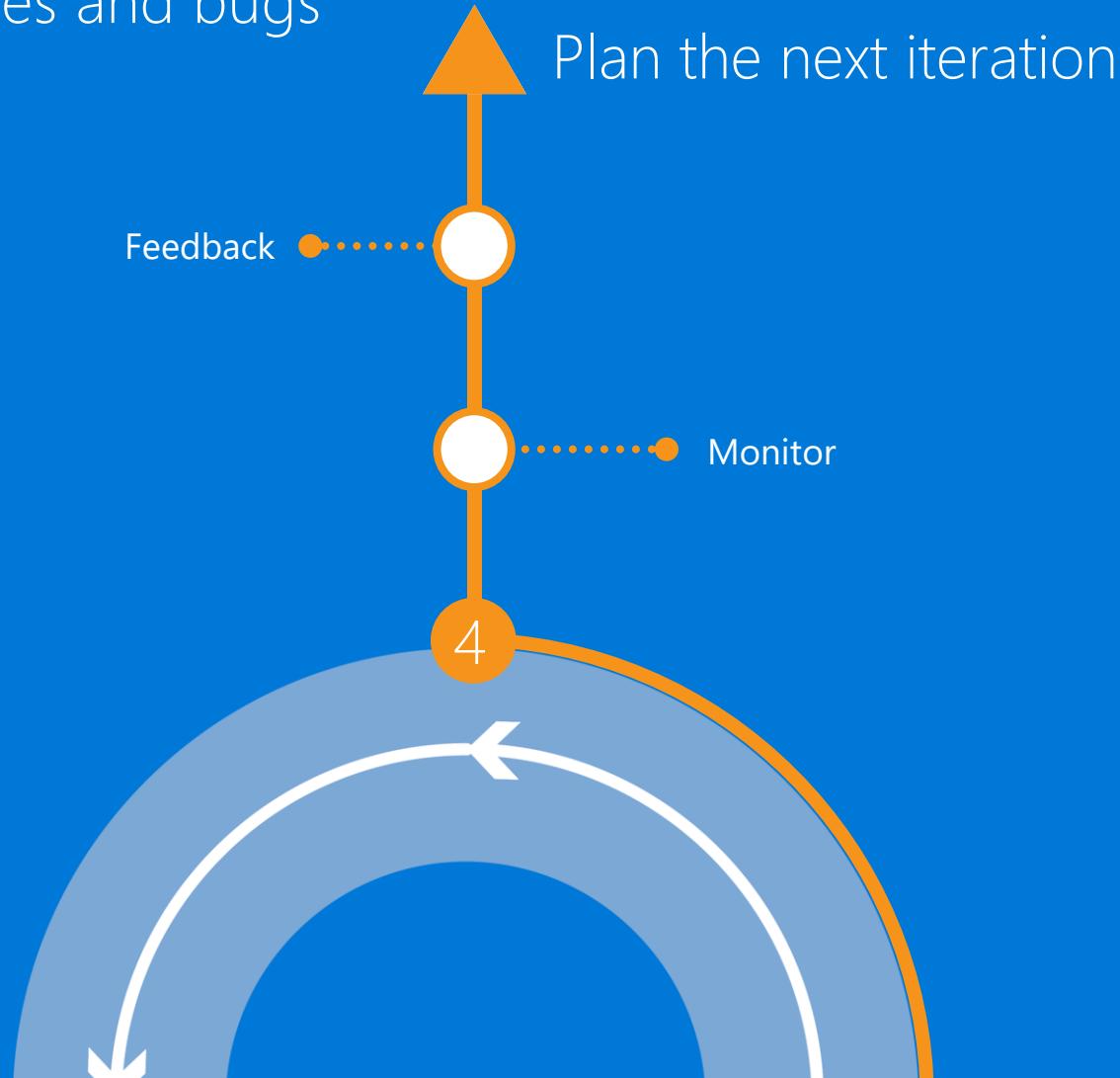
Test

- Test plan
- Load Tests

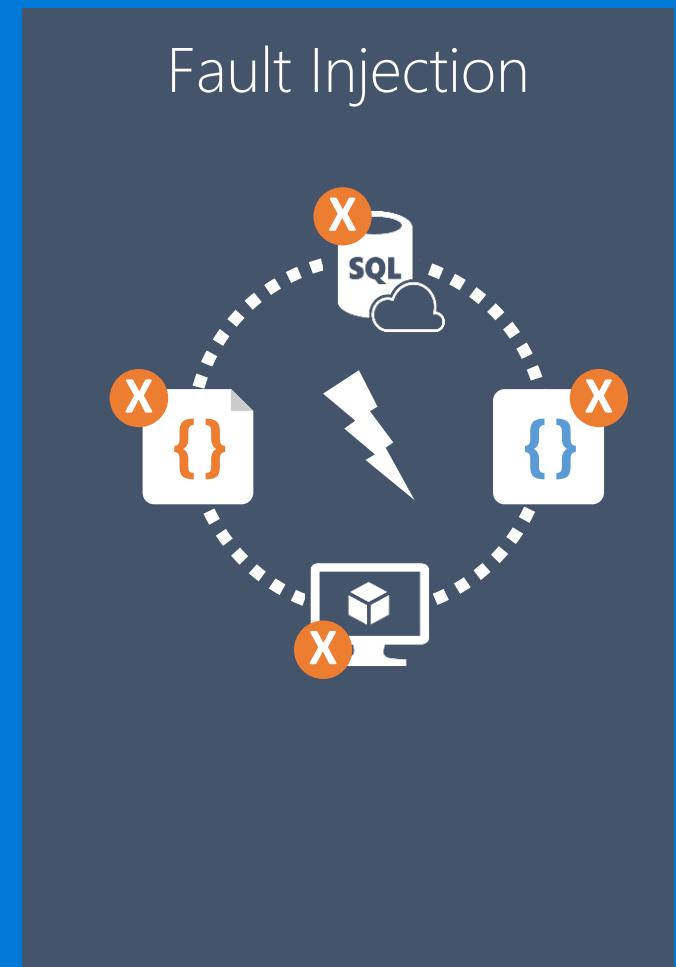
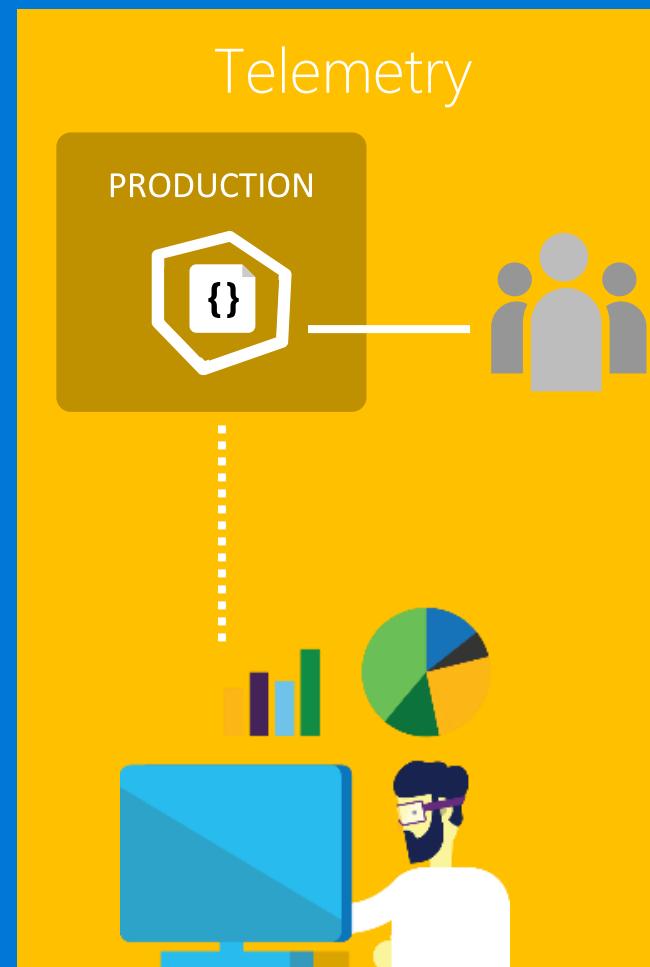
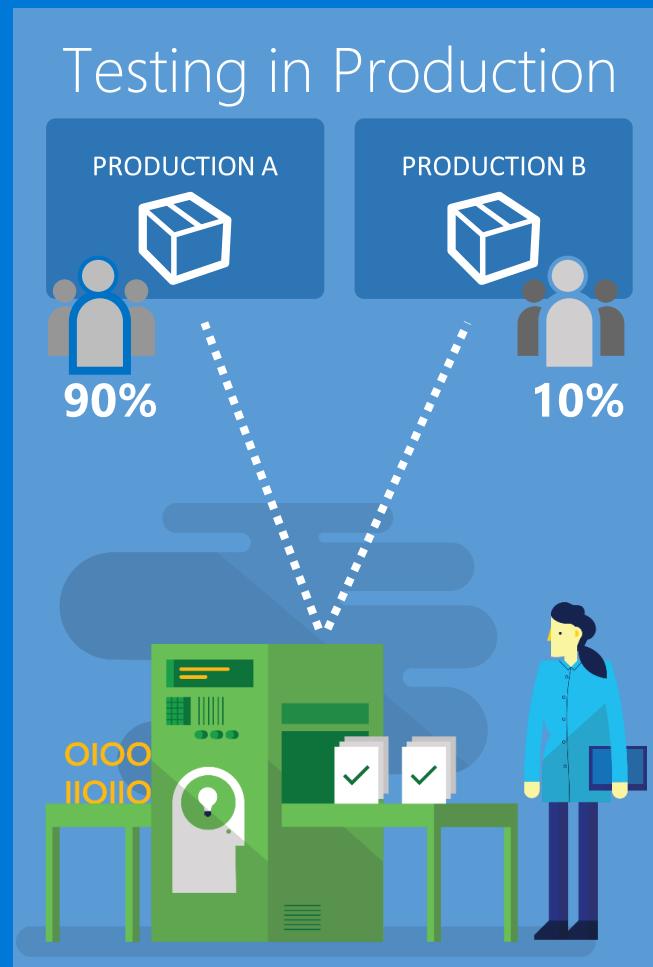


Monitor + Learn

Learn and understand how users use your app, how it reacts and quickly fix issues and bugs



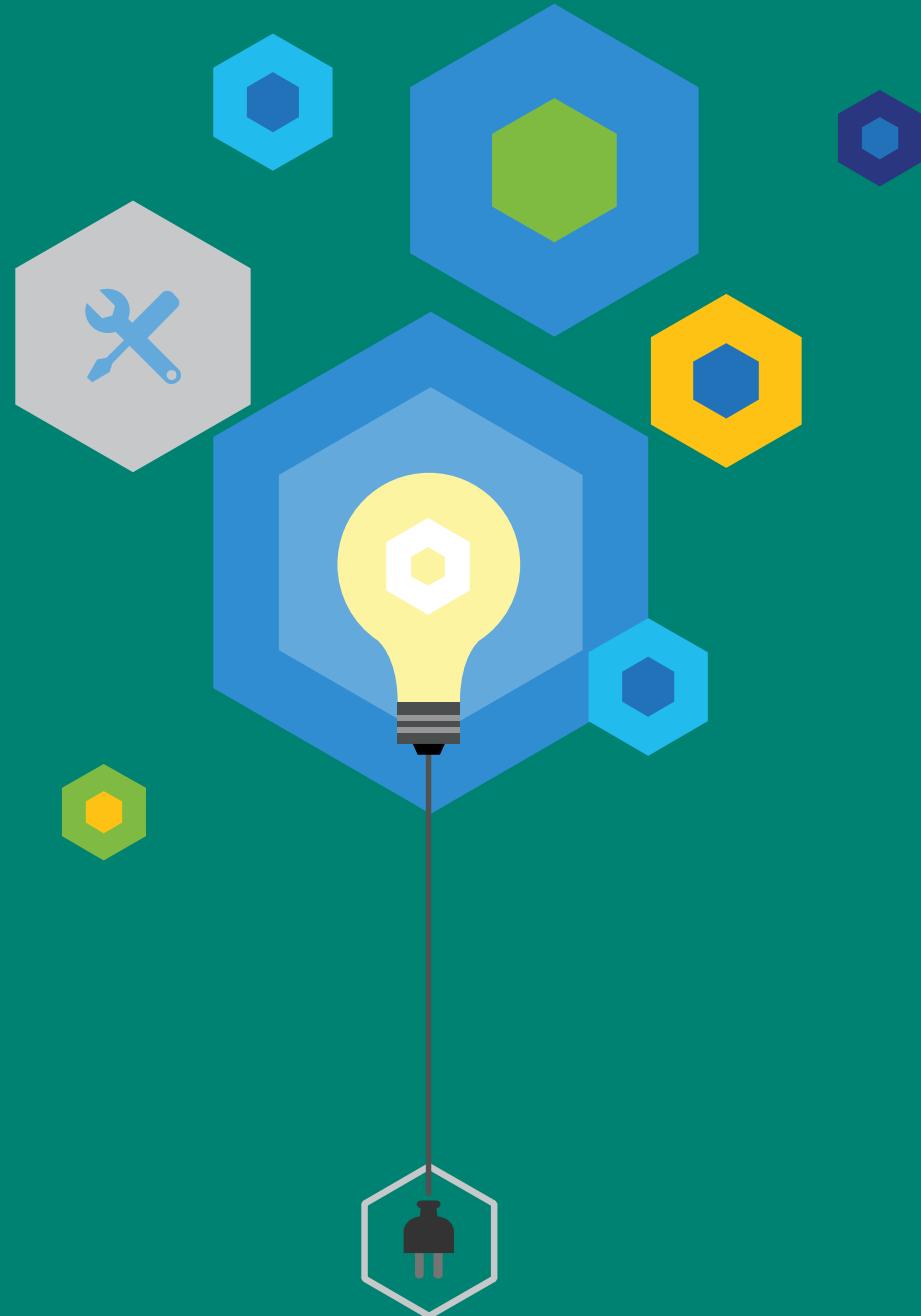
Monitor and Learn



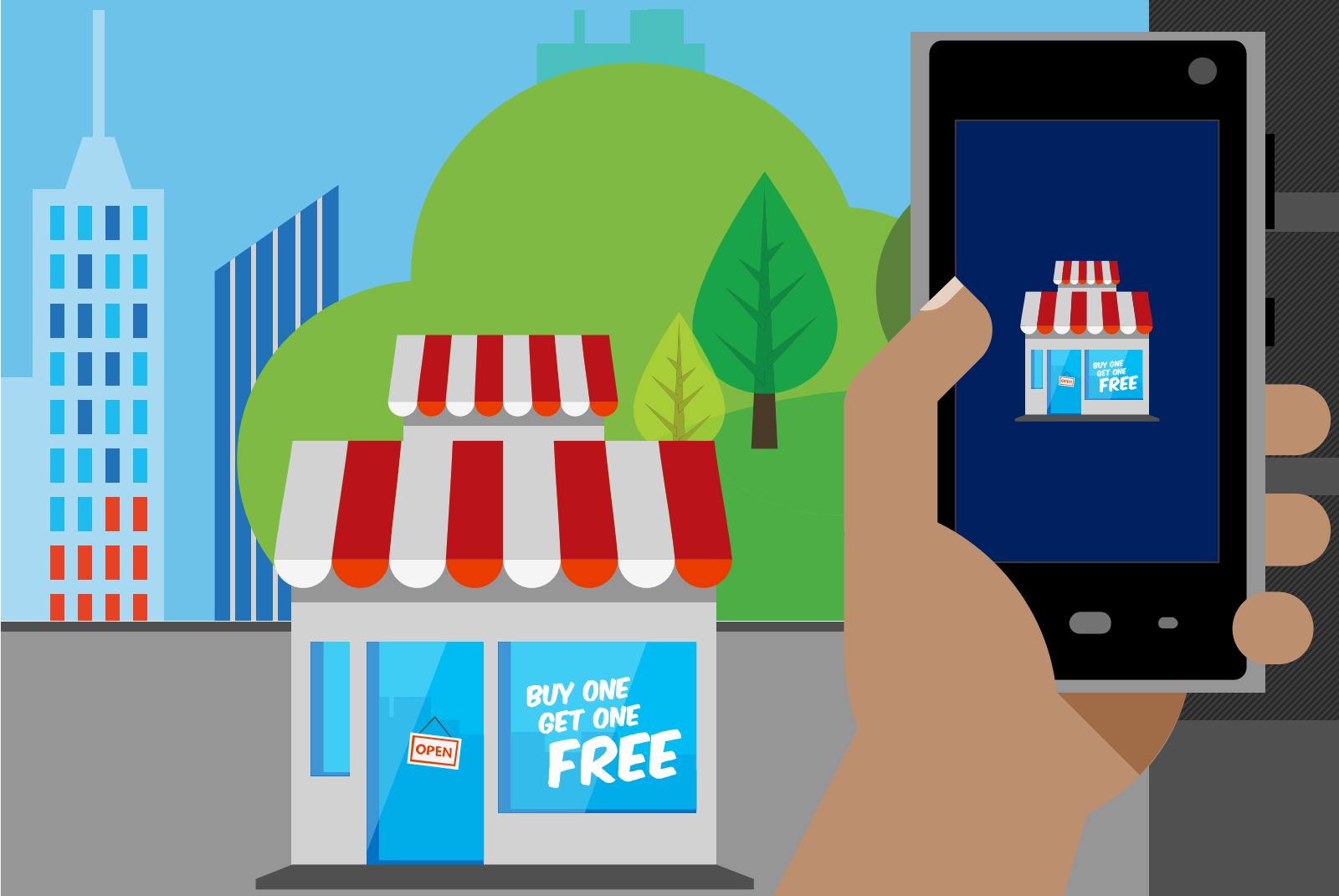
Monitoring

- Continuous feedback
 - App Insights
 - Nagios
- Proactivity
 - Tableau
 - Excel / PowerBI
 - OMS
- Metrics
 - Email
 - WebHook
- Analytics
- Alerts
- Logs
 - Metrics
 - MTTDiagnose
 - MTTRepair
 - MTTFailure
 - MTBetweenFailures

Conclusion and References



It's tough out there...



Business is faster

Competition is fiercer

Time-to-market is key

VSTS can help increase the flow of value and respond to the DevOps change



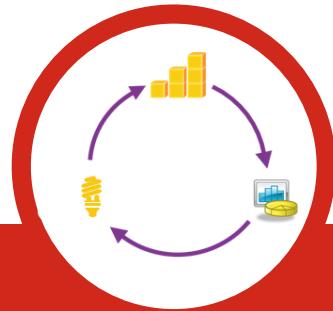
Shorten cycle times



Optimize resources



Improve quality and availability



Hypothesis-driven development & continuous learning

Hosted by Microsoft
Visual Studio Team Services



Cloud services for teams

Hosted in your data center
Team Foundation Server



Team collaboration server

Getting Visual Studio Team Services

Completely free for small teams

FREE: 5 users + unlimited stakeholders

FREE: 240 minutes cloud build time/month

FREE: 20k virtual user minutes for load testing/month

Pricing for additional team members

No. of users:	5	10	20	50	100	200	1000
Cost per month:	FREE	\$30	\$110	\$350	\$750	\$1,150	\$4,350

Each Team Services paid user also gets a Team Foundation Server CAL (Client Access License)

Free for MSDN and Visual Studio subscribers

Users with the following subscriptions can be added to an unlimited number of VSTS accounts for free, with all-feature access:

- Visual Studio Enterprise - annual
- Visual Studio Enterprise - monthly
- Visual Studio Enterprise with MSDN - including subscriptions offered through BizSpark and the Microsoft Partner Network
- Visual Studio Professional - annual
- Visual Studio Professional - monthly
- Visual Studio Professional with MSDN
- Visual Studio Test Professional with MSDN
- MSDN Platforms

Additional services per account

Cloud Build

FREE: 240 minutes/month on a hosted build agent with 30 mins max runtime for a single build

FREE: 1 private agent (new build system) or XAML controller (legacy build system)

THEREAFTER:

Hosted build agent for \$40/month* (you can upgrade the free agent or buy additional agents)

Private agent for \$15/month

XAML controller for \$15/month (ends September 2016)

Cloud Load Testing*

FREE: 20,000 virtual user minutes per month

THEREAFTER:

\$0.004/virtual user minute for 20,001-2M VUM

\$0.002/VUM for 2,000,001-10M VUM

\$0.001/VUM for usage above 10M VUM/month

* Requires Visual Studio Enterprise

- Services purchased via Microsoft Azure based on actual usage
- Additional services are accessible to all users on the account and shared among them
- Build time is actual computing time required to run the build, and does not include queuing time (if any)
- Load test runs specify the number of virtual users and duration in minutes
Virtual users x minutes = virtual user minutes (VUM)

Getting Team Foundation Server

Completely free for individuals and small teams

Team Foundation Server Express is free for individual developers and small teams of five or less and can easily be installed on a personal desktop or laptop without a dedicated server.

Pricing for additional team members

With **Team Foundation Server**, you license the server and the users connecting to the server. Users who only need access to work items are free, and other users need a user CAL (client access license) for access to most features in Team Foundation Server.

Free for MSDN and Visual Studio subscribers

Users with the following subscriptions can be added to an unlimited number of team projects for free, with all-feature access:

- Visual Studio Enterprise - annual
- Visual Studio Enterprise - monthly
- Visual Studio Enterprise with MSDN - including subscriptions offered through BizSpark and the Microsoft Partner Network
- Visual Studio Professional - annual
- Visual Studio Professional - monthly
- Visual Studio Professional with MSDN
- Visual Studio Test Professional with MSDN
- MSDN Platforms

Feature comparison

	Visual Studio Team Services	Team Foundation Server
Plan & Track Work	■	■
Source Code Management	■	■
Package Management	■	■
Test Management	■	■
Cross-Platform Build	■	■
Continuous Deployment	■	■
Release Management	■	■
Application Telemetry	Application Insights/HockeyApp	System Center/PreEmptive Analytics
Reporting	Limited	■
Extend, Customize & Integrate	■	■
Team Structure	1 Collection – Unlimited Projects/Teams	Unlimited Collections/Projects/Teams
Identity & Permissions	Azure Active Directory Microsoft Accounts (Live ID)	Active Directory Integration
Languages	English Only	Multilanguage Support
Data Location	US, Europe, Australia DCs - India (coming soon)	Data Stored On Premises
Backup/Data Migration	Limited support, TFS Import (coming soon)	Backup/Restore Tools Available
Support	Phone, Online and Forums, Engineering Excellence	Phone, Online, Forums, Microsoft Premier Support
Updates	Frequent Updates (~3 weeks)	Periodic Updates (~3 or 4 months)

Feature comparison

	Visual Studio Team Services	Team Foundation Server
Version Control		
Team Foundation Version Control	■	■
Distributed Version Control with Git	■	■
Agile Planning & Collaboration		
Scrum, Agile, CMMI	■	■
Custom Process Templates & WIT Customization	Limited	■
Work Item Tagging, Kanban Board, Feedback	■	■
SharePoint, Project Server, System Center Integration		■
Build		
Automated Builds On-Premises with CI	Support for hybrid scenarios	■
Hosted Build Service with Continuous Integration	■	
Application/Platform Support	Many Microsoft platforms and growing (Build Server Software)	Fully customizable
Continuous Deployment to Azure	■	■
Testing		
Test Case Management	■	■
Cloud-Based Load Testing Service	■	
Work Item Queries and Favorites		Work Item Queries and Favorites/Work Item

Some lessons learned

- Automate
- Start small
- Fail fast
- Solve a problem that really matters
- Don't think too much: Do & Learn - Validate & Learn.
- Stay safe. You will make errors. Things will break.
- Don't wait till you have enough time. You never will.
- Automate again

DevOps Resources

Optimize your DevOps practices and tools

aka.ms/DevOps

DevOps Dimension Show on Channel 9

aka.ms/DevOpsDimension

Technical resources for Practitioners:

aka.ms/DevOpsLearn

DevOps Maturity Self-Assessment

<http://devopsassessment.azurewebsites.net/>

Microsoft Visual Studio ALM Virtual Machines – to practice your DevOps with VSTS

<https://microsoft.github.io/almvm/labs/vsts/>

OpenEdX - Continuous Integration and Continuous Deployment

https://openedx.microsoft.com/courses/course-v1:Microsoft+DevOps200.3+2017_T1/about

OpenEdX - DevOps Testing

https://openedx.microsoft.com/courses/course-v1:Microsoft+DEVOPS200.5+2017_T1/about

Using Git with Visual Studio 2013

<https://mva.microsoft.com/en-us/training-courses/using-git-with-visual-studio-2013-jump-start-8306>

Microsoft Technical Case Studies – DevOps

<https://microsoft.github.io/techcasestudies/#technology=DevOps&technology=Mobile%20DevOps>

VSTS - Associated latest news and updates

The screenshot shows a web browser displaying the Visual Studio Team Services Features Timeline. The URL in the address bar is visualstudio.com/en-us/articles/news/features-timeline. The Microsoft logo is at the top left, followed by navigation links: Technologies, Documentation, and Resources. Below that is a main navigation bar with links: Visual Studio, Visual Studio Team Services, Features, Pricing, News, Support, and Subscriber Access. The main content area has a sidebar titled "Features timeline" with a list of years from 2012 to 2017, each with a corresponding date. A "Subscribe to Updates" link is also present. The main content area features a large title "Visual Studio Team Services Features Timeline" and a subtitle "Last Update: 4/10/2017". It includes a section titled "Features under development" with a descriptive paragraph and a table listing various features with their planned dates and server availability.

visualstudio.com/en-us/articles/news/features-timeline

Microsoft Technologies Documentation Resources

Visual Studio Visual Studio Team Services Features Pricing News Support Subscriber Access

Features timeline

2017

March 29

March 8

February 15

January 25

January 5

2016

2015

2014

2013

2012

Subscribe to Updates

Visual Studio Team Services Features Timeline

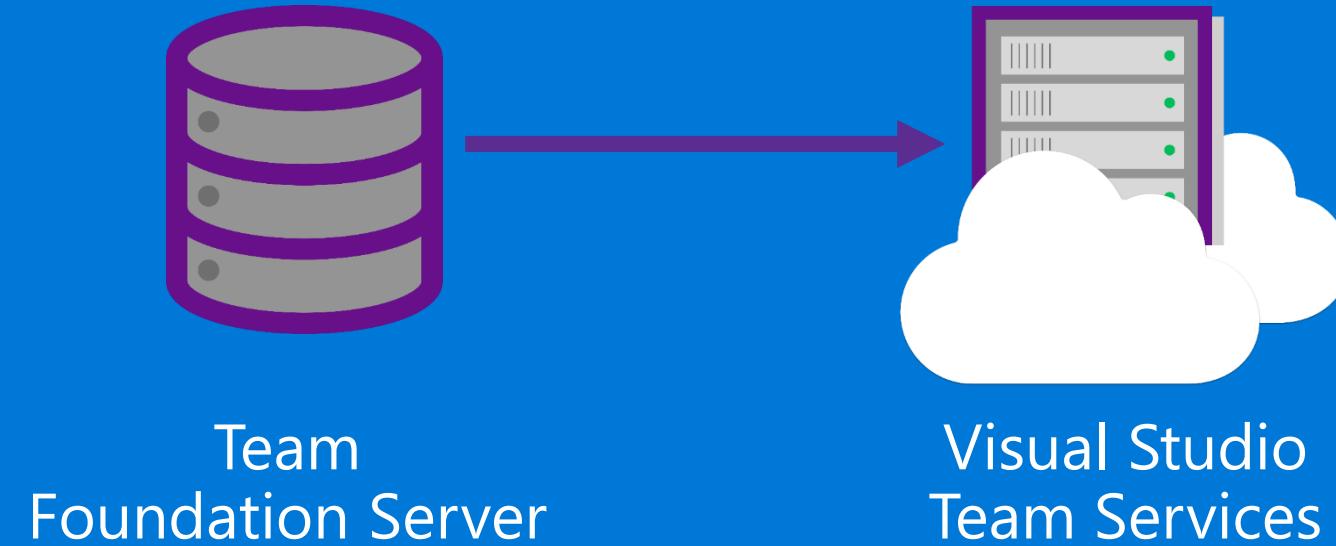
Last Update: 4/10/2017

Features under development

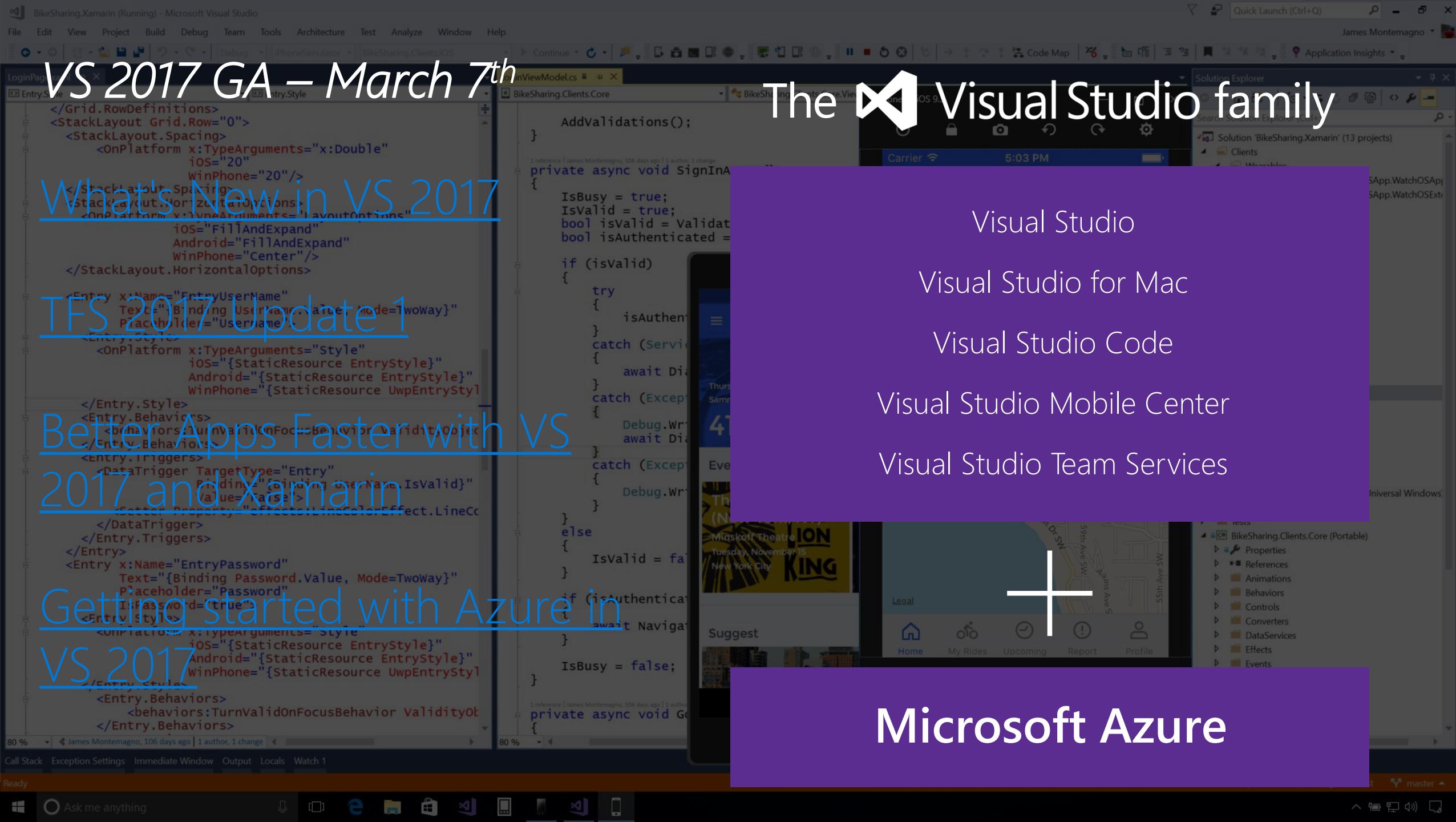
This feature list is a peek into our roadmap. It identifies some of the significant features we are currently working on and a rough timeframe for when you can expect to see them. It is not comprehensive but is intended to provide some visibility into key investments. Most features are linked to a blog post and/or UserVoice entry where you can learn more and comment. These features and dates are the current plans at this time and are subject to change. The "Planned Date" reflects when the feature will be available on VSTS and the "Server" column reflects when it will be available in TFS on-premises, if applicable.

Planned Date	Feature	Server
2017 Q1	Work Items – Delivery Plan	2017.2
2017 Q1	Work Items – Delivery Plan Field-level Filtering	2017.2
2017 Q1	Work Items – Mobile Web	2017.2
2017 Q1	TFVC support in IntelliJ	--
2017 Q1	Git – Tags view	2017.2
2017 Q1	Git – Commit graph	2017.2
2017 Q1	Package management support for Maven packages	TBD
2017 Q1	Release Management Improvements - Artifact defaults, Branch Conditions, Schedules	2017.2

TFS to VSTS migration



<https://aka.ms/tfsimportdata>



VS 2017 GA – March 7th

What's New in VS 2017

IFS 2017 Update 1

Better Apps Faster with VS 2017 and Xamarin

Getting started with Azure in VS 2017

The  Visual Studio family

Visual Studio

Visual Studio for Mac

Visual Studio Code

Visual Studio Mobile Center

Visual Studio Team Services

Microsoft Azure

Visual Studio 2017 - Enterprise
Announcing
Enterprise DevOps Accelerator

Our best tools including Visual Studio Enterprise
CI/CD and testing service
Discounted Azure pricing for dev/test
On-site expert DevOps coaching

<http://aka.ms/devopsnow>

Questions

Answers

