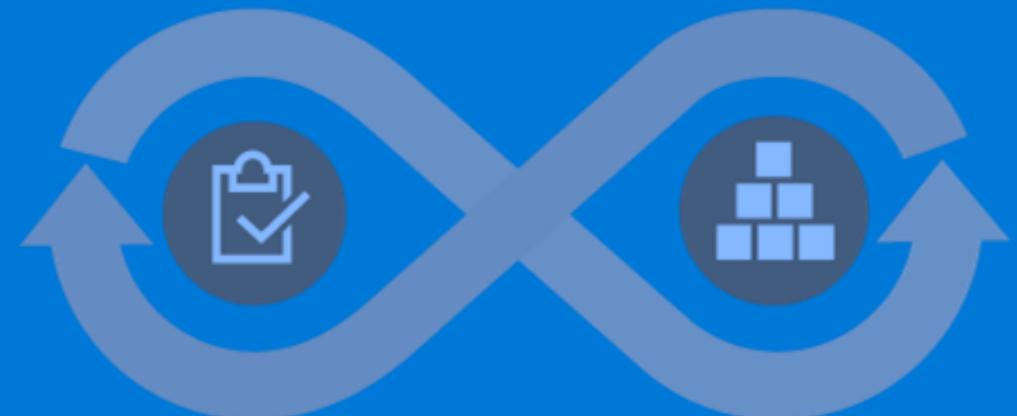


# DevOps

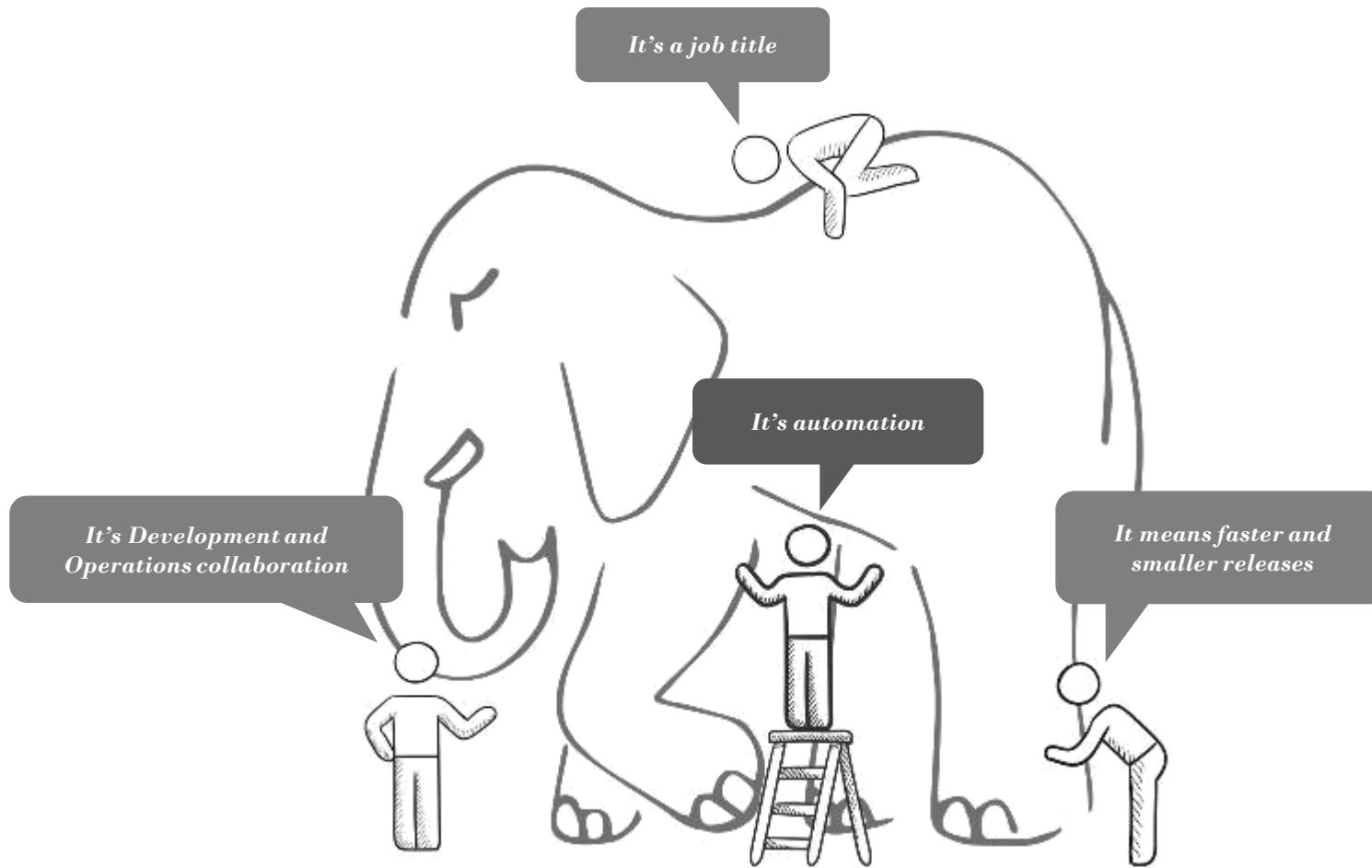
Alice Gibbons  
*Azure App Dev TSP*



# Agenda

- What is DevOps?
- DevOps at Microsoft
- Azure DevOps Overview &
  - Azure Boards
  - Azure Repos
  - Azure Pipelines
  - Azure Test Plans
  - Azure Artifacts
- Hands-On Work

# WHAT IS DEVOPS?



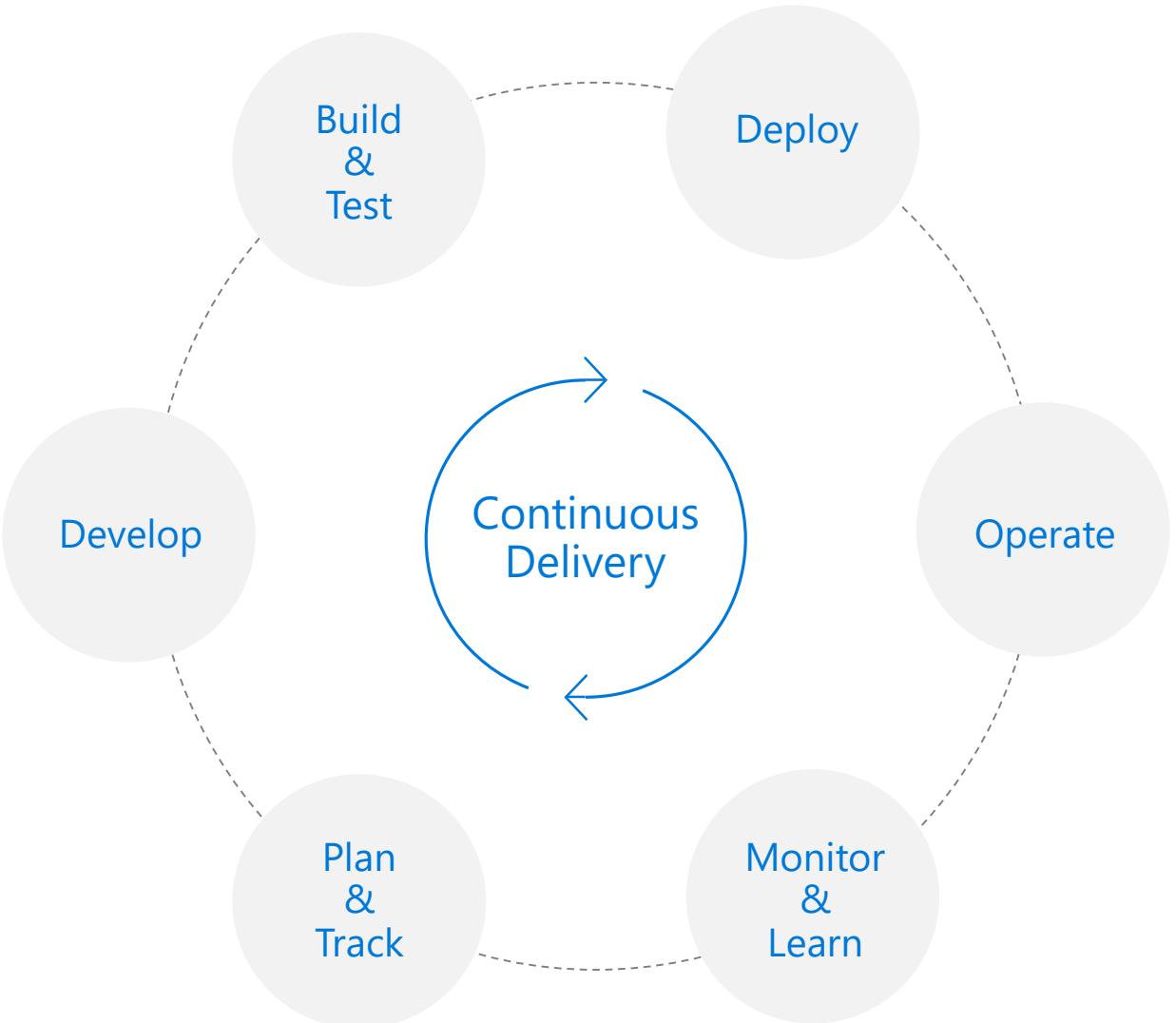
# What is DevOps?

People. Process. Products.

“

DevOps is the union of **people**, **process**, and **products** to enable continuous delivery of value to your end users.”

”



# One Engineering System for DevOps

There cannot be a more important thing for an engineer, for a product team, than to work on the systems that drive our productivity.

So I would, any day of the week, trade off features for our own productivity.

I want our best engineers to work on our engineering systems, so that we can later on come back and build all of the new concepts we want.

- Satya Nadella





# 1ES DevOps tools adoption at Microsoft

Of the 75K 1ES users,  
approximately 800 of them  
actually support our DevOps tools.



Company wide engaged users of VSTS in the Microsoft organization (Aug 2017)

# DevOps at Microsoft

Azure DevOps is the toolchain of choice for Microsoft engineering with over 90,000 internal users



<https://aka.ms/DevOpsAtMicrosoft>

**372k**

Pull Requests per month

**4.4m**

Builds per month

**5m**

Work items viewed per day

**2m**

Git commits per month

**500m**

Test executions per day

**500k**

Work items updated per day

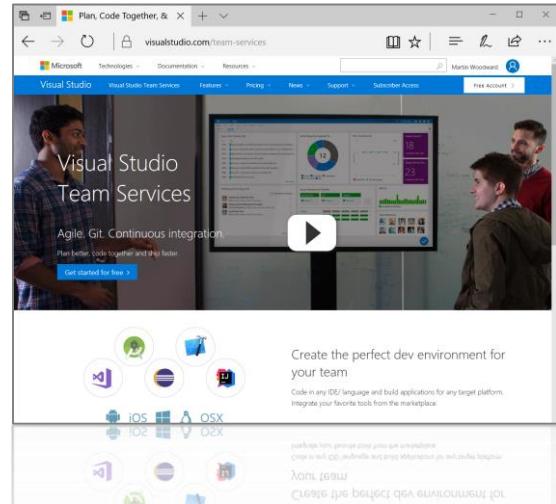
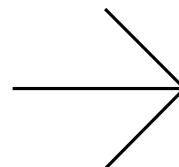
**78,000**

Deployments per day

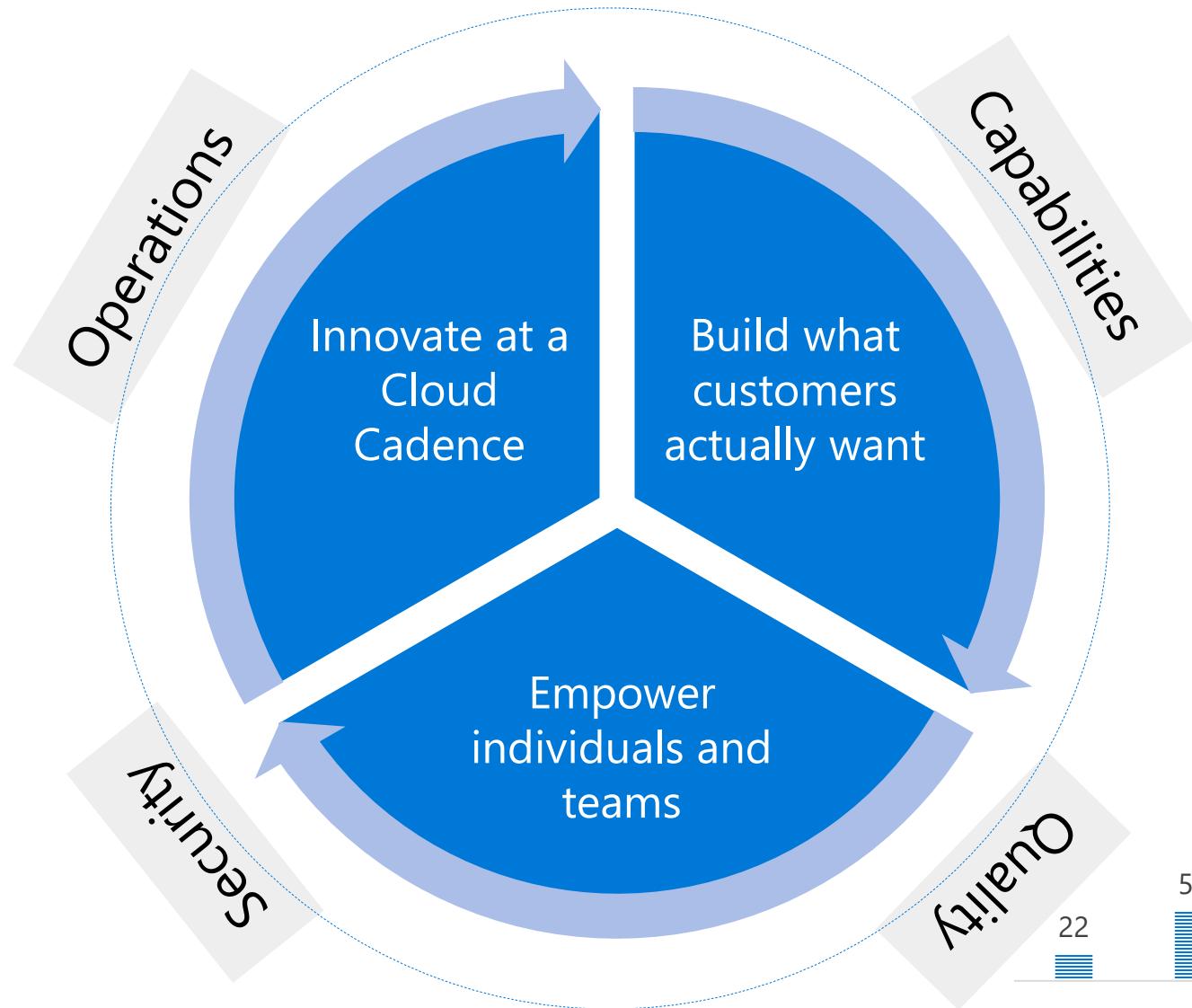
Data: Internal Microsoft engineering system activity, August 2018



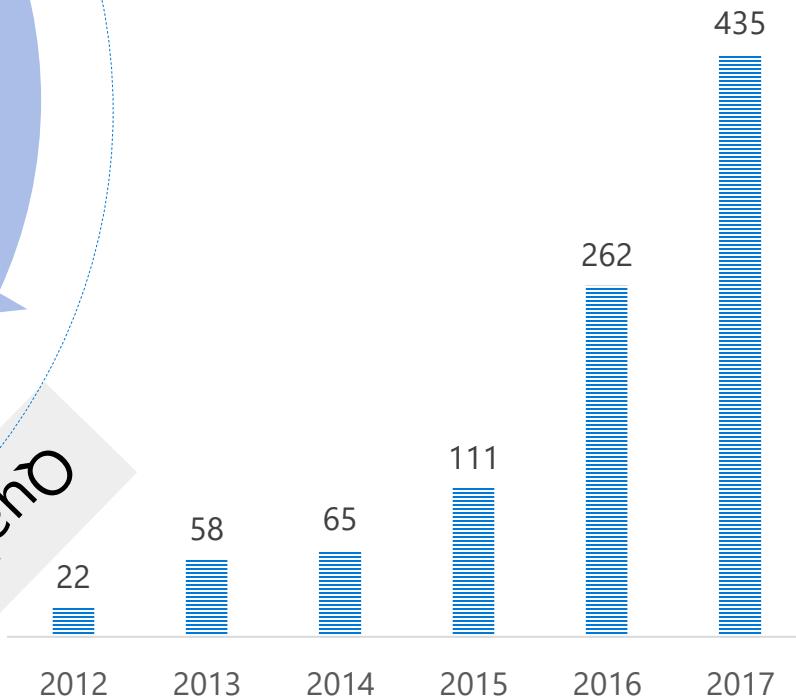
# The VSTS Journey to DevOps



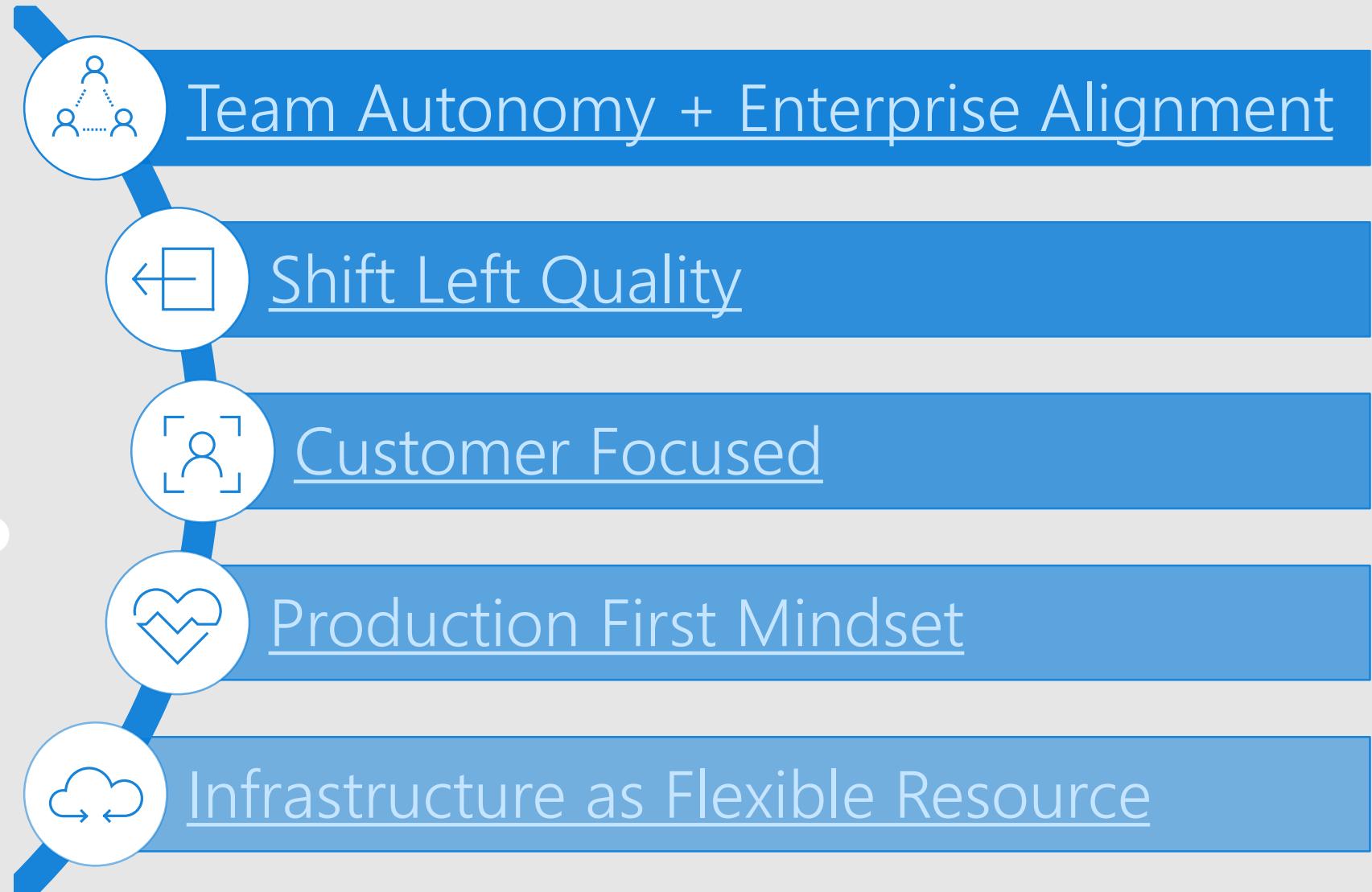
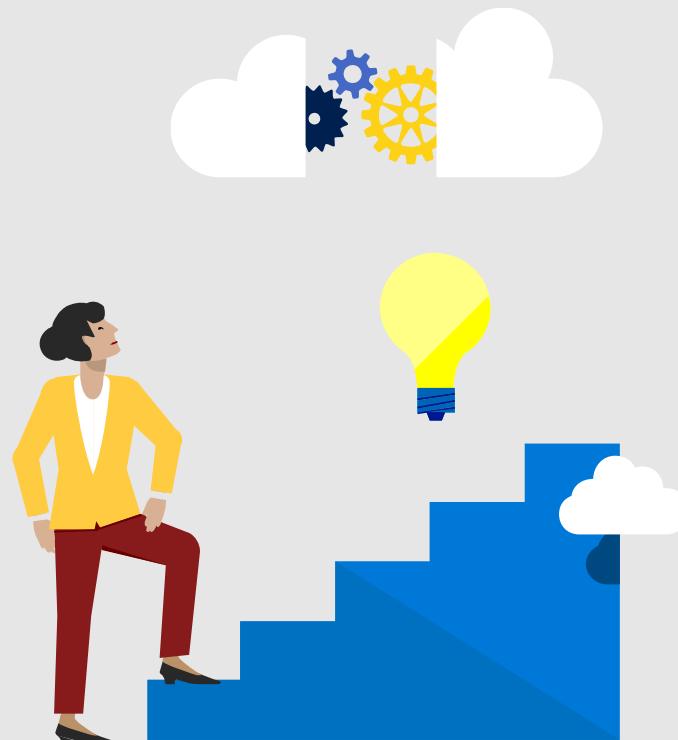
# Why even embark on the journey?



**More features in the 2016 calendar year (262 features) than the previous 4 years Combined (256 features). And the growth continues!**



# Five DevOps Habits we've learned...



# Automate completely

- No more “one time” commands run manually
- Every command goes in PowerShell scripts that are checked in
- Deployment to pre-production & canary is the same as deployment to production **every time**
- All orchestrated with Release Management in VSTS

VSOnline Home Code Work Build & Release Test Wiki Compliance \* |

Overview My Dashboard Favorites - CI Runs Calendar | Welcome

We've made big improvements to the navigation experience in Team Services. [Take the tour](#) or view the [release notes](#) for more details.

### Release Branch Runs - Trial Phase in S107

Environments

	Ring 0	Ring 1	PPE Binary	Prod Binary	Sps.SelfTest	Sps.SelfHost	Tfs.SelfHost Set 1	Tfs.SelfHost Set 2	Tfs.SelfTest	Tfs.Deploy	TfsOnPrem.SelfHost
Ring 0	✓ 100%	✓ 100%	✓ 100%	✓ 100%	✓ 100%	✓ 100%	✓ 100%	✓ 100%	✓ 100%	✓ 100%	✓ 100%
Ring 1	✓ 100%	✓ 100%	✓ 100%	✗ 100%	✓ 100%	✓ 100%	✓ 100%	✗ 99.68%	✓ 100%	✗ 98.7%	✓ 100%
PPE Binary											
Prod Binary											
Sps.SelfTest											
Sps.SelfHost											
Tfs.SelfHost Set 1											
Tfs.SelfHost Set 2											
Tfs.SelfTest											
Tfs.Deploy											
TfsOnPrem.SelfHost											

Branch: refs/heads/releases/M108      Latest build: [VSO.Release.CI\\_M108\\_20161101.25](#)

### TFS - Prod Update

Ring 0	Ring 1	Ring 2	Ring 3	+1
✓ TFS - Prod...	+1			

TFS - Prod Update 424 ✓ ✓ ✓ ✓ ✓  
TFS - Prod Update 423 ✓ ✓ ✓ ✓ ✓  
TFS - Prod Update 422 ✓ ✓ ✓ ✓ ✓  
TFS - Prod Update 421 ✗ ✗ ✗ ✗ ✗  
TFS - Prod Update 420 ✓ ✓ ✓ ✓ ✓

[View all releases for TFS - Prod Update release definition](#)

Master Branch Runs - -

Environments

	Tfs.SelfTest	Tfs.Deploy	Tfs.SelfHost Set 1	Tfs.SelfHost Set 2	TfsOnPrem.SelfTest	TfsOnPrem.SelfHost	Sps.SelfTest	Sps.SelfHost
Tfs.SelfTest	✓							
Tfs.Deploy		✓						
Tfs.SelfHost Set 1			✓					
Tfs.SelfHost Set 2				✓				
TfsOnPrem.SelfTest					✓			
TfsOnPrem.SelfHost						✓		
Sps.SelfTest							✓	
Sps.SelfHost								✓

Branch: refs/heads/master

VSO.Release.CI

11/1/2016

VSO.Package.RealSign

11/1/2016

# Tracking Deployments to Production (5 Rings)

mseng.visualstudio.com/VSOnline/Service%20Insights/\_apps/hub/ms.vss-releaseManagement-web.hub-explorer?definitionId=1208\_a=releases

VSOnline / Service Insights | Home Code Work Build & Release Test Wiki Compliance \* | ☰

Builds Releases Task Groups Packages Explorer

TFS - Prod Config Change | Edit

Overview Releases Deleted

Search release definitions...

1	2	3	4	5			
Lock	Title	Environments	Build	Branch	Created ↓	Created By	Description
	Draft-2501	...	VSO.Release.Cl_M...	refs/heads/releases...	just now	Hitesh Sajnani	
	TFS - Prod Config Change 2358	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	2 hours ago	Joe Zhang (VSCS)	Add tarpitting exception for sp...
	TFS - Prod Config Change 2357	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	7 hours ago	Prawal Agarwal	
	TFS - Prod Config Change 2356	✓ ✓ ✓ ✓ ✘	VSO.Release.Cl_M...	refs/heads/releases...	18 hours ago	Ritwik Saxena	
	Draft-2497	...			18 hours ago	Ritwik Saxena	
	TFS - Prod Config Change 2355	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	21 hours ago	Artur levsieiev	
	TFS - Prod Config Change 2354	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	21 hours ago	Navneet Gupta	Disabling tracepoint collection f...
	TFS - Prod Config Change 2353	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	22 hours ago	Artur levsieiev	
	TFS - Prod Config Change 2352	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	22 hours ago	Navneet Gupta	[tayoun] Enabling Tracepoint co...
	TFS - Prod Config Change 2351	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	23 hours ago	Jeff Buswell	Fix State Constants for DTS #68...
	TFS - Prod Config Change 2350	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	23 hours ago	Sam Sharifi	Enable limiting the number of t...
	TFS - Prod Config Change 2349	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Tuesday	Sanyogita Ranade	Queue a retention job on MSEng
	TFS - Prod Config Change 2348	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Tuesday	Tingluo Huang	2.107.1 to SBR1
	TFS - Prod Config Change 2347	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Chandru Ramakri...	Disable RLS on tbl_DatabasePar...
	TFS - Prod Config Change 2346	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Richard Mishaan	Config Change to set SmtpPass...
	TFS - Prod Config Change 2345	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Dan Hellem	polaristest - increase attachme...
	TFS - Prod Config Change 2344	✗	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Hitesh Sajnani	Enable AnalyticsTagDefinitionC...
	TFS - Prod Config Change 2343	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Hitesh Sajnani	Enabling AnalyticsWorkItemCh...
	TFS - Prod Config Change 2342	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Tingluo Huang	
	TFS - Prod Config Change 2341	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Dan Hellem	Walmart - process mapping for...
	TFS - Prod Config Change 2340	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Kavipriya Adhinar...	
	TFS - Prod Config Change 2339	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Kavipriya Adhinar...	
	TFS - Prod Config Change 2338	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	Monday	Pavan Kumar Ade...	Enabling the auto SPN across al...
	TFS - Prod Config Change 2337	✓ ✓ ✓ ✘	VSO.Release.Cl_M...	refs/heads/releases...	10/7/2016	Rogan Ferguson	Turn on for and not rules on im...
	TFS - Prod Config Change 2336	✓ ✓ ✓ ✓ ✓ ✓	VSO.Release.Cl_M...	refs/heads/releases...	10/7/2016	Patrick Carnahan	Patch prc_GetBuilds on SU0

1. Canary (internal users)
2. Smallest external data center
3. Largest external data center
4. International data centers
5. All the rest

# Our DevOps Transformation – the story so far

## 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

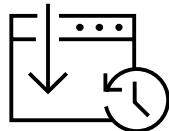
## After

- 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
- Mockups in PPT
- Inner source
- Flattened organization hierarchy
- User satisfaction determines success
- Features shipped every sprint

# Azure DevOps Services

# What technologies do I need to support DevOps?

DevOps brings together people, processes, and products, automating software delivery to provide continuous value to your users. Using Azure DevOps, you can deliver software faster and more reliably - no matter how big your IT department or what tools you're using.



## Continuous integration (CI)

- Improve software development quality and speed.
- When you use Azure Pipelines or Jenkins to build apps in the cloud and deploy to Azure, each time you commit code, it's automatically built and tested and bugs are detected faster.

101010  
010101  
101010

## Continuous Deployment (CD)

- By combining continuous integration and infrastructure as code (IaC), you'll achieve identical deployments and the confidence to deploy to production at any time.
- With continuous deployment, you can automate the entire process from code commit to production if your CI/CD tests are successful.



## Continuous Learning & Monitoring

- With Azure Application Insights you can identify how your applications are performing and test if the recent deployment made things better or worse.
- Using CI/CD practices, paired with monitoring tools, you'll be able to safely deliver features to your customers as soon as they're ready.

# Introducing Azure DevOps



## Azure Boards

Deliver value to your users faster using proven agile tools to plan, track, and discuss work across your teams.



## Azure Test Plans

Test and ship with confidence using manual and exploratory testing tools.



## Azure Pipelines

Build, test, and deploy with CI/CD that works with any language, platform, and cloud. Connect to GitHub or any other Git provider and deploy continuously.



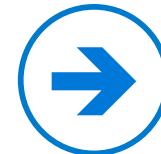
## Azure Artifacts

Create, host, and share packages with your team, and add artifacts to your CI/CD pipelines with a single click.



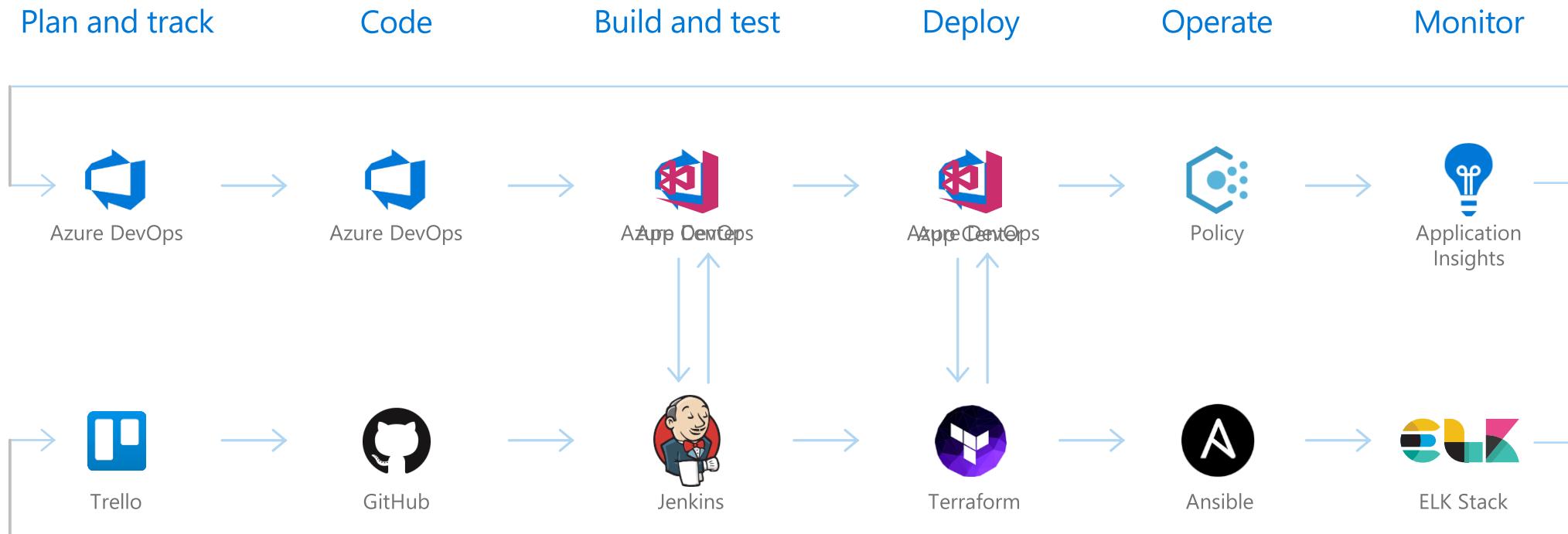
## Azure Repos

Get unlimited, cloud-hosted private Git repos and collaborate to build better code with pull requests and advanced file management.



<https://azure.com/devops>

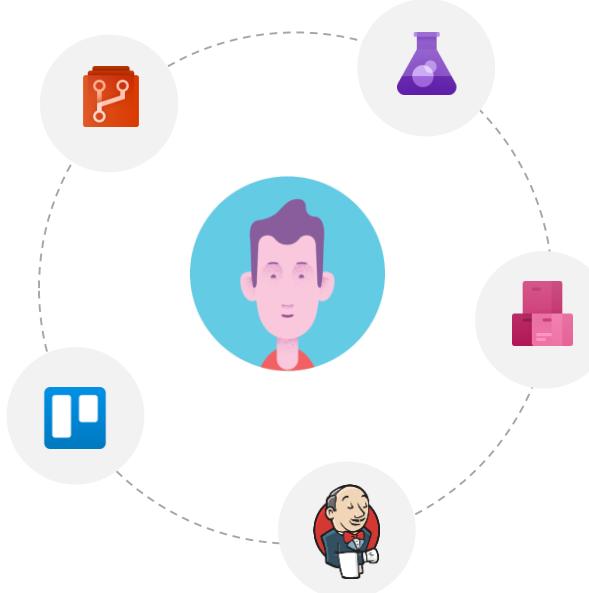
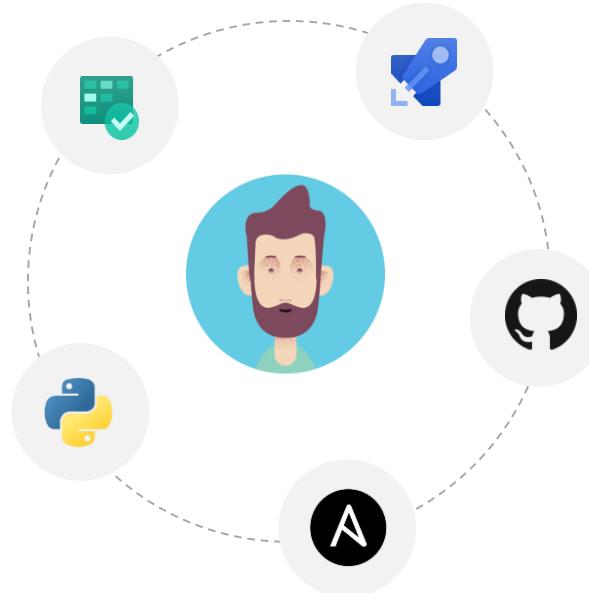
# Azure DevOps framework



# Azure DevOps: Choose what you love

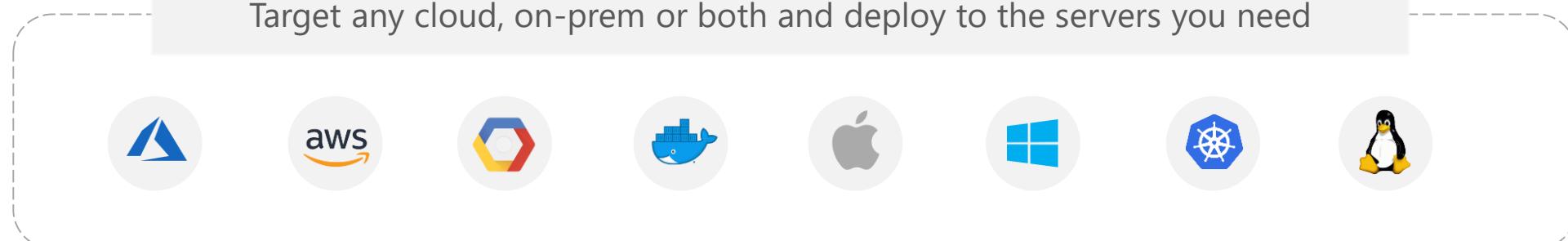
Any Language, Any Platform

Azure DevOps lets developers choose the tools and languages that are right for them



Mix and match to create workflows with tools from Microsoft, open source or your favorite 3rd party tools

Target any cloud, on-prem or both and deploy to the servers you need



# Self-Service Dev/Test Environments

## Azure Lab Services

- Simplify cloud environment management for developers and testers.
- Enforce policies and control costs with full visibility
- Use templates, custom images and formulas to reproduce environments.
- Orchestrate with Azure Pipelines or integrate using REST API

The screenshot shows the Microsoft Azure portal interface. On the left, the classic Azure navigation menu is visible. In the center, the 'Contoso DevTest Lab' configuration page is displayed. The 'Configuration and policies' section is selected. On the right, the 'Marketplace images' blade is open, showing a list of available images for the lab. The 'Allowed images' section includes a search bar and a checkbox for 'Select all'. Below this, a table lists various Red Hat Enterprise Linux images, each with a checkbox and a 'Standard terms' status indicator. Some images have specific notes in parentheses, such as 'SQL Server 2017 Enterprise on Red Hat Enterprise Linux 7.4 (RHEL)'.

NAME	OFFER STATUS
Free SQL Server License: SQL Server 2017 Developer on Red Hat Enterprise Linux 7.3 (RHEL)	Standard terms
Free SQL Server License: SQL Server 2017 Express on Red Hat Enterprise Linux 7.3 (RHEL)	Standard terms
Red Hat Enterprise Linux 6.7	Standard terms
Red Hat Enterprise Linux 6.7 for SAP HANA	Standard terms
Red Hat Enterprise Linux 6.8	Standard terms
Red Hat Enterprise Linux 6.8 for SAP Business Apps	Standard terms
Red Hat Enterprise Linux 6.9	Standard terms
Red Hat Enterprise Linux 7.2	Standard terms
Red Hat Enterprise Linux 7.2 for SAP HANA	Standard terms
Red Hat Enterprise Linux 7.3	Standard terms
Red Hat Enterprise Linux 7.3 for SAP Business Apps	Standard terms
Red Hat Enterprise Linux 7.4	Standard terms
SQL Server 2017 Enterprise on Red Hat Enterprise Linux 7.4 (RHEL)	Standard terms
SQL Server 2017 Standard on Red Hat Enterprise Linux 7.4 (RHEL)	Standard terms

# Infrastructure and Configuration as Code

## Azure Resource Manager, Automation & 3<sup>rd</sup> Party Integrations

→ Infrastructure as Code,  
built-in

→ Azure Config & Automation

→ Support for 3<sup>rd</sup> party and OSS  
tooling such as Terraform,  
Ansible, Chef, Puppet & SaltStack



The screenshot shows the Microsoft Azure portal interface. On the left, the navigation menu includes 'Create a resource', 'All services', and various Azure services like Dashboard, Resource groups, Virtual machines, and DevOps Projects. The main area displays a list of Resource groups: AustraliaSEDevelopment, AustraliaSEProduction (selected), autoShutdown, cloud-shell-storage-westus, DefaultResourceGroup-EUS, and securitydata. To the right, the 'AustraliaSEProduction - Automation script' blade is open. It shows an overview with 6 items, a template editor with JSON code, and sections for Parameters (9), Variables (0), and Resources (9). The Resources section lists several DNS zones with their parameters and values. The JSON code in the template editor is as follows:

```
$schema: "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#"
contentVersion: "1.0.0.0"
parameters:
  "dnszones_onazure.io_name": {
    "defaultValue": "onazure.io",
    "type": "String"
  },
  "NS_@_name": {
    "defaultValue": "@",
    "type": "String"
  },
  "SOA_@_name": {
    "defaultValue": "@",
    "type": "String"
  },
  "A_vote_name": {
    "defaultValue": "vote",
    "type": "String"
  },
  "A_draft_name": {
    "defaultValue": "draft",
    "type": "String"
  },
  "A_devops_name": {
    "defaultValue": "devops",
    "type": "String"
  },
  "A_*_draft_name": {
    "defaultValue": "*.draft",
    "type": "String"
  }
}
```

# DevOps Pipelines in Minutes

## Azure DevOps Projects

- Create a full DevOps pipeline with 3 easy steps from the Azure Portal
- Start with a Git repo and any source language
- Web apps, Kubernetes, soon VMs and more.
- Customize, extend and scale when needed.

The screenshot shows the Azure DevOps CI/CD Pipeline interface. On the left, there's a vertical flow diagram representing the pipeline stages: Code, Build, and Production. The 'Code' stage shows a commit from 'nodesampleproject' on the 'master' branch, with a commit hash '411c6bfa' and a message 'updated style'. The 'Build' stage shows a build named 'nodesampleprojectsite' that has succeeded. The 'Production' stage shows a deployment to 'nodesampleprojectsite' that is currently in progress. To the right of the pipeline, there's a section titled 'Azure resources' which lists an 'Application endpoint' at 'http://nodesampleprojectsite.azurewebsites.net' and an 'App Service' named 'nodesampleprojectsite' which is running. Below that is a section for 'Application Insights' with a status entry for 'nodesampleprojectsite'. At the bottom right, there are performance metrics: 'SERVER REQUEST' (0) and 'FAILED REQUEST' (0).

Demo

# Azure DevOps

# Changes for existing VSTS / TFS customers

The same functionality you know and love today, with greater openness, flexibility and focus

## Existing Accounts

- Existing <https://contoso.visualstudio.com> URL continues to work. <https://dev.azure.com/contoso> available for opt-in.
- New UI **opt-in** per user as preview feature. Will start advertising new UI once feedback from new accounts and early adopters has been incorporated.
- Can disable services on a per project basis for new UI
- New Azure branding in communications and documentation (emails, alerts etc).
- Websites and documentation will move from Visual Studio to Azure based locations (with redirects in place).
- Redirects available for some time.
- TFS will remain the on-premises brand until the next major version in 2019. The new UI will be enabled in that release.
- Existing TFS branded information and downloads remain in Visual Studio locations until next release.

## Pricing

- Public project usage is now free.
- The free tier for Pipelines now includes 1,800 minutes per month, up from 240.
- Pipelines can be used independently from Repos — so if you are only using Pipelines and your repos are hosted on GitHub you don't need to pay for Repos or Boards (Basic) users.

## New Accounts

- <https://dev.azure.com/contoso> based URL.
- New navigation & branding by default.

# Migrating from TFS to Azure DevOps

Move from Team Foundation Server to Azure DevOps and bring your data along

## Benefits of Cloud Hosted Azure DevOps Services

- Global availability
- Hosted and maintained by Microsoft with 99.9% uptime guarantee and 24x7 support
- Immediate access to latest features
- Simplified deployment to Azure

## TFS Import Service

- Fully supported high fidelity migration path
- Trusted by many large enterprises
- Now faster and easier to use

➡ <https://aka.ms/tfsimport>



# Azure DevOps Services Pricing

Open Source Projects	Small Teams	Teams of any size
<p><b>Free</b></p> <p>Unlimited users and build time</p> <ul style="list-style-type: none"><li>• <b>Azure Pipelines:</b> 10 parallel jobs with unlimited minutes for CI/CD</li><li>• <b>Azure Boards:</b> Work item tracking and Kanban boards</li><li>• <b>Azure Repos:</b> Unlimited public Git repos</li></ul>	<p><b>Free</b></p> <p>Start free with up to 5 users</p> <ul style="list-style-type: none"><li>• <b>Azure Pipelines:</b> Run 1 Microsoft-hosted job for 1,800 minutes per month and 1 self-hosted job for any amount of time</li><li>• <b>Azure Boards:</b> Work item tracking and Kanban boards</li><li>• <b>Azure Repos:</b> Unlimited public Git repos</li><li>• <b>Azure Artifacts:</b> package management</li><li>• Unlimited stakeholders</li></ul>	<p><b>Starts at \$6</b></p> <p>per user, per month for Boards &amp; Repos*</p> <p>Easy pricing that grows with your team</p> <ul style="list-style-type: none"><li>• <b>Azure Pipelines:</b> Run 1 Microsoft-hosted job for 1,800 minutes per month and 1 self-hosted job for any amount of time</li><li>• <b>Azure Boards:</b> Work item tracking and Kanban boards</li><li>• <b>Azure Repos:</b> Unlimited public Git repos</li><li>• <b>Azure Artifacts:</b> package management</li><li>• Unlimited stakeholders</li><li>• Boards &amp; Repos included for Visual Studio subscribers</li></ul>



<https://azure.com/pricing/details/devops/>

\* 5 Boards & Repos users and 5 Artifacts users free. Pipelines with unlimited minutes, Test Plans users and additional Artifacts users also available. Please see the Azure pricing calculator for details.

# Azure DevOps

## #AzureDevOps



<https://azure.com/devops>



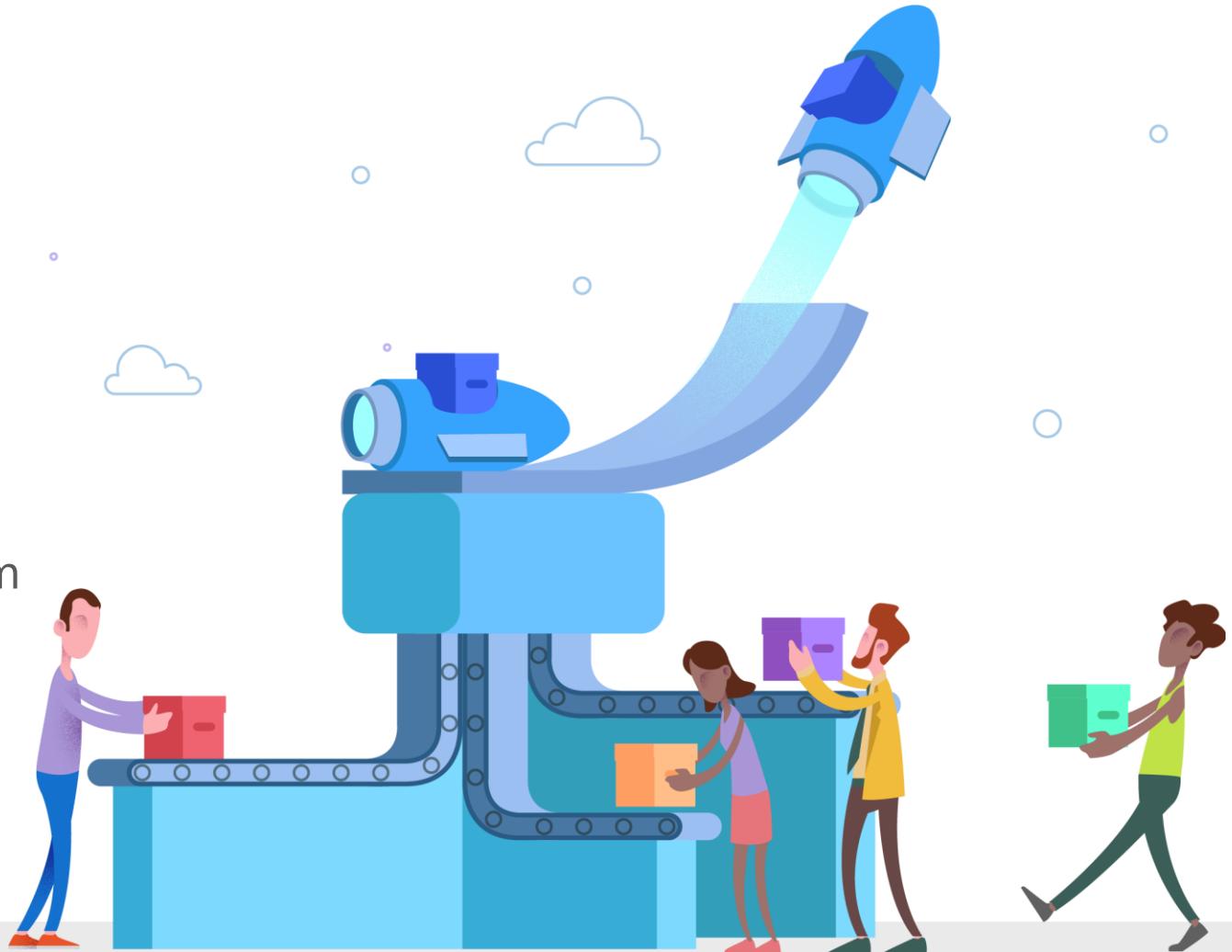
@AzureDevOps



<https://aka.ms/AzureDevOpsForum>



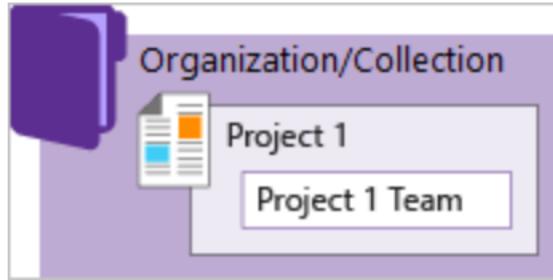
<https://aka.ms/DevOpsBlog/>



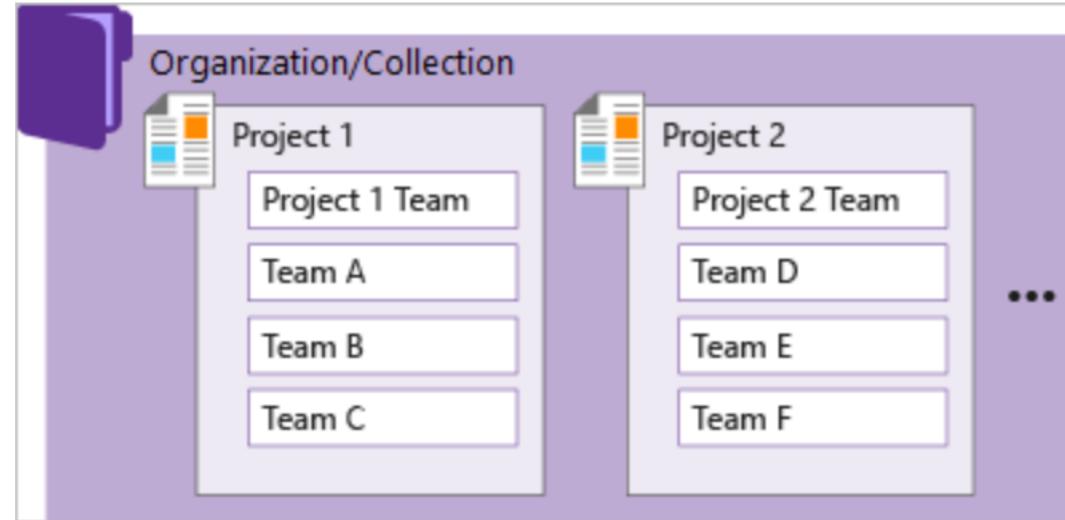
# Organizational Structure

---

Single project and team defined within an organization or collection

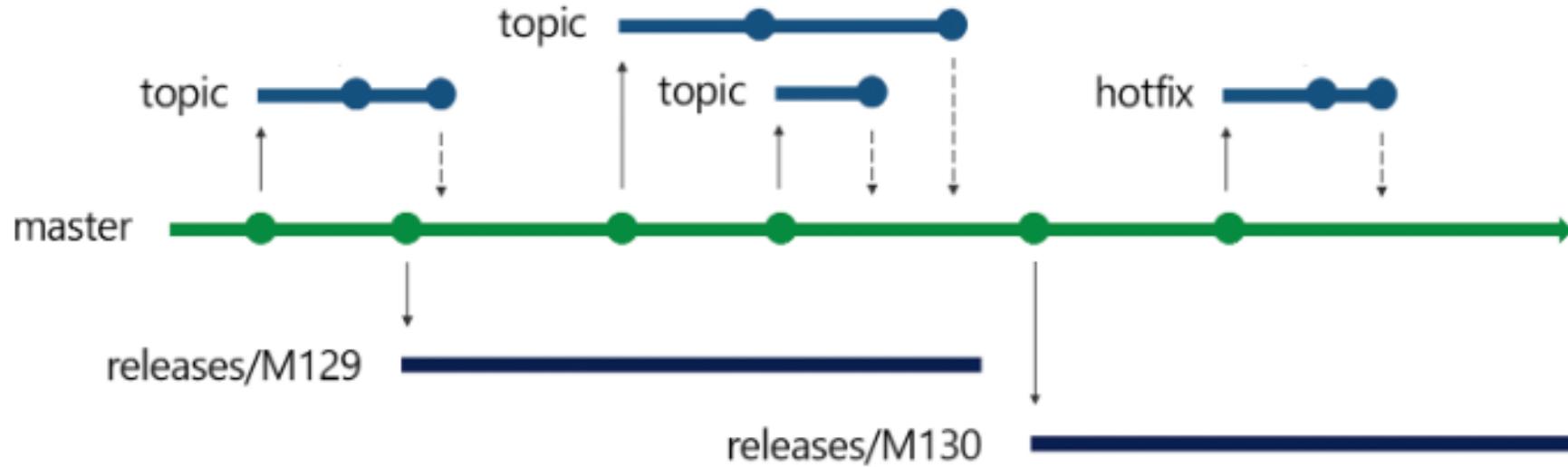


Multiple projects and teams defined within organization or collection



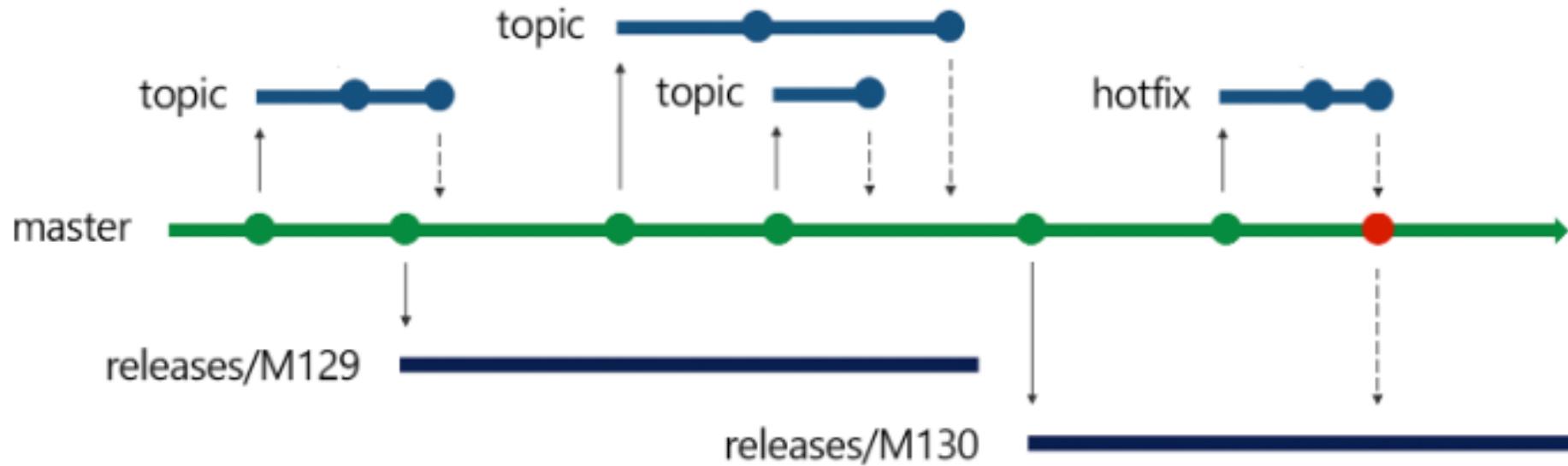
Collection → Project → Team

# Release Flow - Trunk Based Development



- One single master branch – the “trunk”
- No other long-running dev branches
- Enables CI/CD
- No commits to release branches
- PR workflow for commits to master
- Release branches cut from master and can be deleted after Sprint

# Hot fixes - Trunk Based Development



1. Create hotfix on new branch
2. Merge PR into master as usual (PR workflow)
3. Cherry-pick hotfix from master into release branch
4. Never hotfix from Production directly - "master first" policy

# Acquire from GitHub

Offering Azure Pipelines to any developer through the GitHub marketplace



Marketplace / Azure Pipelines

## Azure Pipelines

Fully managed CI/CD platform that works with any language, platform, and cloud.

Azure Pipelines offers cloud-hosted build agents running Linux, macOS, and Windows. Get 10 free concurrent jobs and unlimited minutes for public repositories, and 1 free job for private repositories.

[Read more...](#)

**Categories**

- Continuous integration
- Deployment

**Supported languages**

C++, Go, Java and 5 other languages supported

**Developer links**

Support  
Status  
Documentation  
Privacy Policy

The screenshot shows the Azure Pipelines interface with a build log for a job named "VSO.PR\_VSO.PR\_20180622.165". The log details the execution of various tasks, including initializing the job, killing orphan processes, getting sources, changing impacted files, initializing, pre-search, verifying Docker image exists, and pulling the Docker image. The log also shows the start time as 6/22/2018 11:32 AM and the end time as 11:32:19. The log output includes command-line logs for each task.



Azure Pipelines

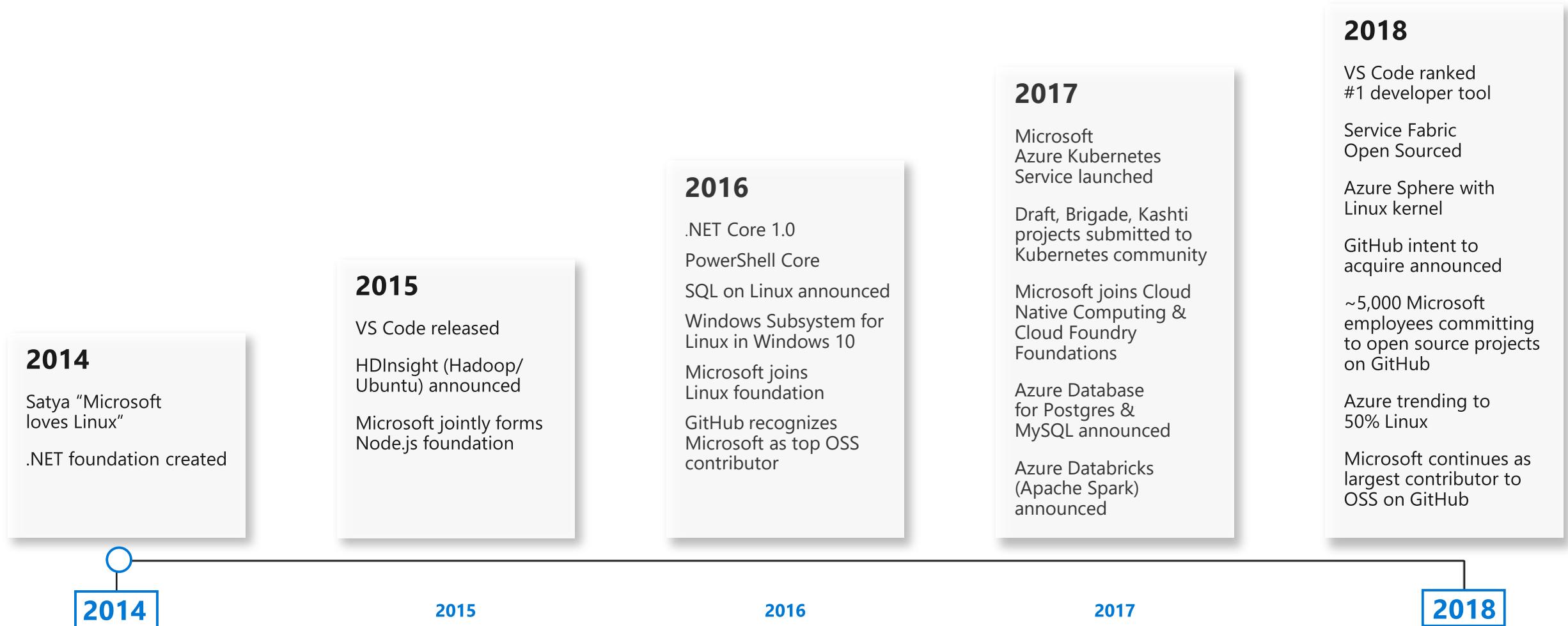
10 Free  
concurrent pipelines  
with **unlimited** build  
minutes for public  
projects.

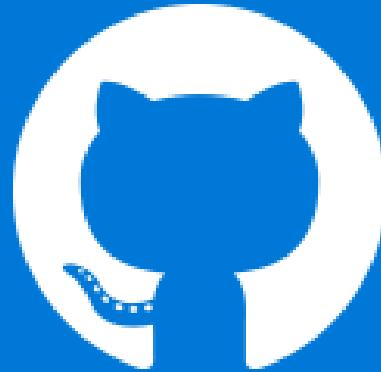
Microsoft



Open Source

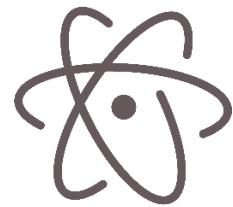
# Microsoft ❤️ Open Source



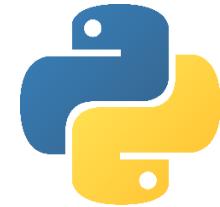


# Early adopters

GitHub projects already powered  
by Azure Pipelines for CI



atom/atom



python/cpython



dotnet/reactive



microsoft/vscode

# Azure Pipelines

Cloud-hosted pipelines for Linux, Windows and macOS, with unlimited minutes for open source



## Any language, any platform, any cloud

Build, test, and deploy Node.js, Python, Java, PHP, Ruby, C/C++, .NET, Android, and iOS apps. Run in parallel on Linux, macOS, and Windows. Deploy to Azure, AWS, GCP or on-premises



## Extensible

Explore and implement a wide range of community-built build, test, and deployment tasks, along with hundreds of extensions from Slack to SonarCloud. Support for YAML, reporting and more



## Containers and Kubernetes

Easily build and push images to container registries like Docker Hub and Azure Container Registry. Deploy containers to individual hosts or Kubernetes.



## Best-in-class for open source

Ensure fast continuous integration/continuous delivery (CI/CD) pipelines for every open source project. Get unlimited build minutes for all open source projects with up to 10 free parallel jobs across Linux, macOS and Windows

The screenshot shows the Azure DevOps Pipelines interface for the Contoso / AdventureWorks Mobile / Pipelines / Builds / 10382 pipeline. The pipeline is titled "Enabling feature flags for Preview Attachment and Grid Views". It includes three parallel jobs: Windows Job (Running, 1m 53s), Linux Job (Running, 3m 29s), and macOS Job (Running, 3m 07s). The Linux Job details pane is expanded, showing the following tasks and their status:

- Prepare job: Completed
- Initialize job: Completed
- Get sources: Completed
- Cmdline: Completed
- Nodetool: Completed
- Install dependencies: In progress

The log output for the Linux Job shows the following command sequence:

```
yarn install v1.7.0
$ node build/npm/preinstall.js
[1/4] Resolving packages...
[2/4] Fetching packages...
[3/4] Linking dependencies...
[4/4] Building fresh packages...
$ npm run compile
#####
> code-oss-dev-build@1.0.0 compile ./adventureworks/build
> tsc -p tsconfig.build.json

✖ Done in 4.89s.
$ node ./postinstall
[#=] 2/2 removed './adventureworks/extensions/node_modules/typescript/lib/tsc.js'
removed './adventureworks/extensions/node_modules/typescript/lib/tsserverlibrary.d.ts'
removed './adventureworks/extensions/node_modules/typescript/lib/tsserverlibrary.js'
removed './adventureworks/extensions/node_modules/typescript/lib/typescriptServices.d.ts'
removed './adventureworks/extensions/node_modules/typescript/lib/typescriptServices.js'
```



<https://azure.com/pipelines>

@DonovanBrown

Demo:  
VSCode in Azure Pipelines

# Azure Boards

Track work with Kanban boards, backlogs, team dashboards, and custom reporting



## Connected from idea to release

Track all your ideas at every development stage and keep your team aligned with all code changes linked directly to work items.



## Scrum ready

Use built-in scrum boards and planning tools to help your teams run sprints, stand-ups, and planning meetings.



## Project insights

Gain new insights into the health and status of your project with powerful analytics tools and dashboard widgets.

The screenshot shows the Azure DevOps Boards interface. At the top, it says "Azure DevOps" and "Contoso / AdventureWorks Mobile / Boards / FabrikamFiber". Below that is a navigation bar with links: "Overview", "Boards", "Work items", "Backlogs", "Sprints", "Queries", "Plans", "Repos", "Pipelines", "Test Plans", and "Artifacts". The main area is titled "FabrikamFiber Board". It has four columns: "New" (5/5), "Active" (15/15), "Staging" (0/0), and "Deployed" (0/0). The "Active" column contains several work items, each with a title, description, and assigned team members:

- Home page (selected room) - Kat Larson (Design)
- Guests page - Carole Poland (ML, Xamarin)
- NFC open door - Cecil Folk (Spike, Xamarin)
- Room Tab - Celeste Burton (Rooms [Detail])
- Map filter - Carole Poland (General, Room [List])
- Hotel reviews page - Celeste Burton (Rooms [Detail])
- Top page controls - Celeste Burton (ML, Xamarin)
- Search component complex features - Cecil Folk (General, Xamarin)
- Images from api - Carole Poland (General)
- Adapt some parts of UI to UWP for Desktop - Carole Poland (Blocked, Xamarin)
- Ambient settings - Carlos Slattery (Xamarin)
- Notifications list - Carole Poland (General)



<https://azure.com/devops>

@DonovanBrown

Demo

# Azure Boards

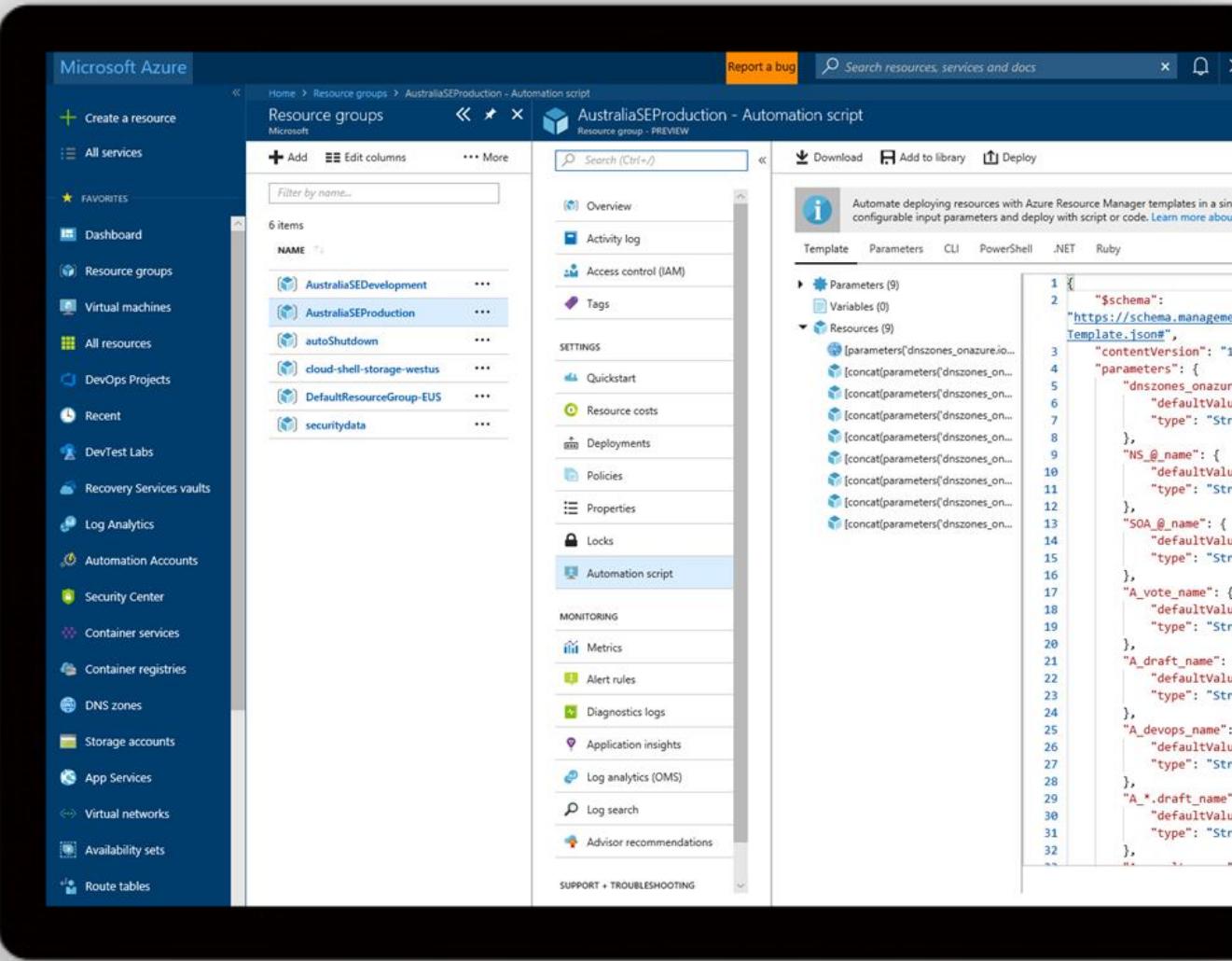
# Deploy Repeatedly & Reliably

# Azure Resource Manager & DevOps Tool Integrations

# Infrastructure as Code, built-in with Azure Resource Manager

# Use Azure Automation & Config to automate repetitive tasks

Support for DevOps tool integrations  
and OSS tooling such as Terraform,  
Ansible & Chef



Demo

# Azure Pipelines

# Azure Repos

Unlimited private Git repo hosting and support for TFVC that scales from a hobby project to the world's largest Git repositories



## Works with your Git client

Securely connect with and push code into your Git repos from any IDE, editor, or Git client.



## Web hooks and API integration

Add validations and extensions from the marketplace or build your own using web hooks and REST APIs.



## Semantic code search

Quickly find what you're looking for with code-aware search that understands classes and variables.

The screenshot shows the Azure DevOps interface for the Contoso / AdventureWorks Mobile project. The left sidebar includes links for Overview, Boards, Repos (selected), Files, Commits, Pushes, Branches, Tags, and Pull requests (also selected). The main content area is titled 'Pull requests' and shows a list of pull requests categorized by assignee: 'Mine', 'Active', 'Completed', and 'Abandoned'. The 'Mine' section contains four items:

- Initialize client with .client.init (Kat Larsson requested #238 into master)
- Testing configuration settings (Kat Larsson requested #230 into features/config)
- Check returned identity for null status (Colin Ballinger requested #212 into master)
- [WIP] Add tests for deployment mapping (Robin Counts requested #221 into master)

The 'Active' section contains two items:

- Add exception on disconnect (Colin Ballinger requested #249 into master)
- Maintain structure when converting isomorphs (Robin Counts requested #234 into master)

The 'Completed' section contains one item:

- Hotfix payload to releases/99 (Robin Counts requested #201 into releases/99)

Each pull request entry includes a small profile picture, the title, the requester, and the target branch. To the right of each entry are icons for users, comments, and other actions.



<https://azure.com/devops>

@DonovanBrown

Demo

# Azure Repos

# Azure Test Plans

Get end-to-end traceability. Run tests and log defects from your browser. Track and assess quality throughout your testing lifecycle.



## Capture rich data

Capture rich scenario data as you execute tests to make discovered defects actionable. Explore user stories without test cases or test steps. You can create test cases directly from your exploratory test sessions.



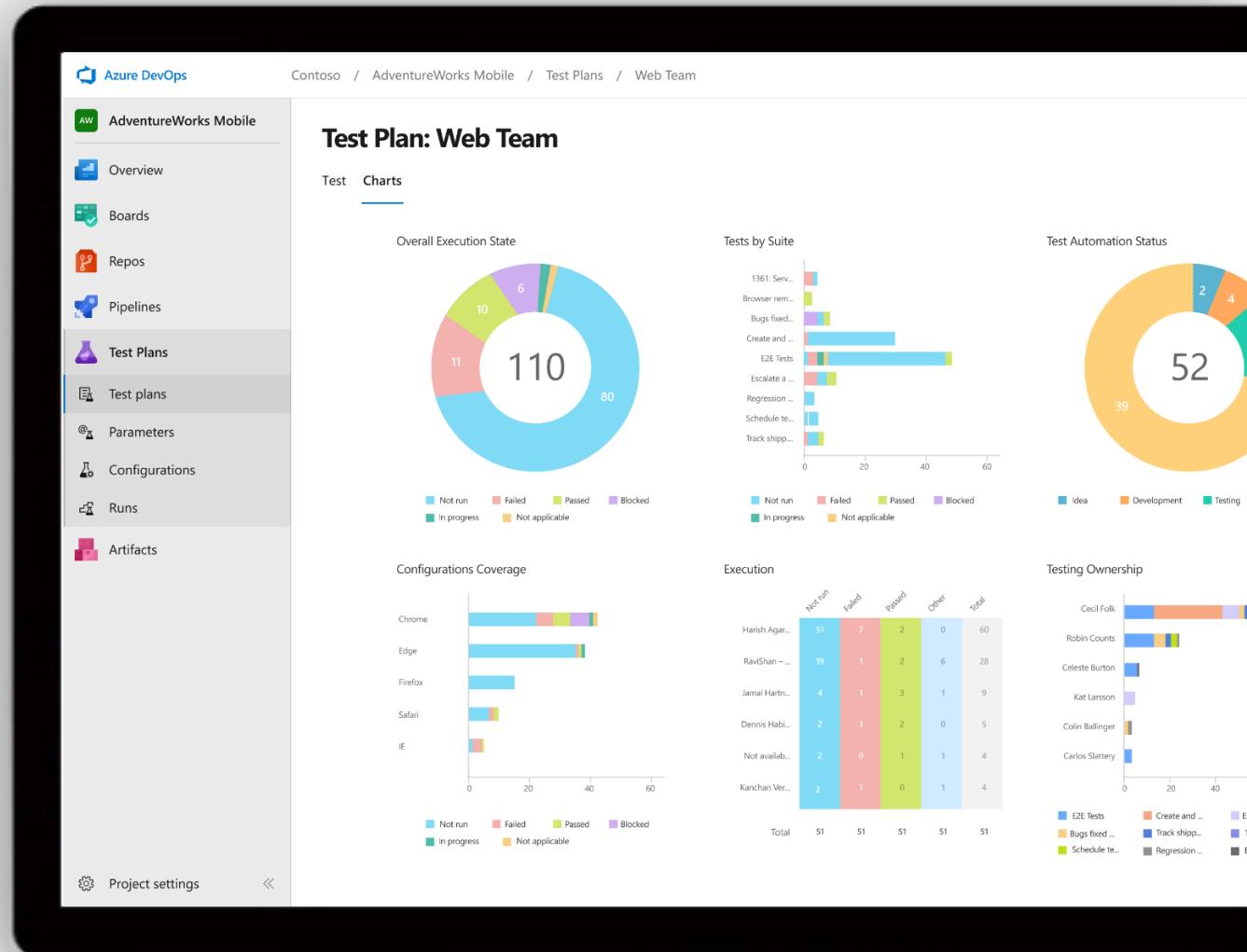
## Test across web and desktop

Test your application where it lives. Complete scripted tests across desktop or web scenarios. Test on-premises application from the cloud and vice-versa.



## Get end-to-end traceability

Leverage the same test tools across your engineers and user acceptance testing stakeholders. Pay for the tools only when you need them.



<https://azure.com/devops>

@DonovanBrown

Demo

# Azure Test Plans

# Testing: Shift Left from Integration to Unit

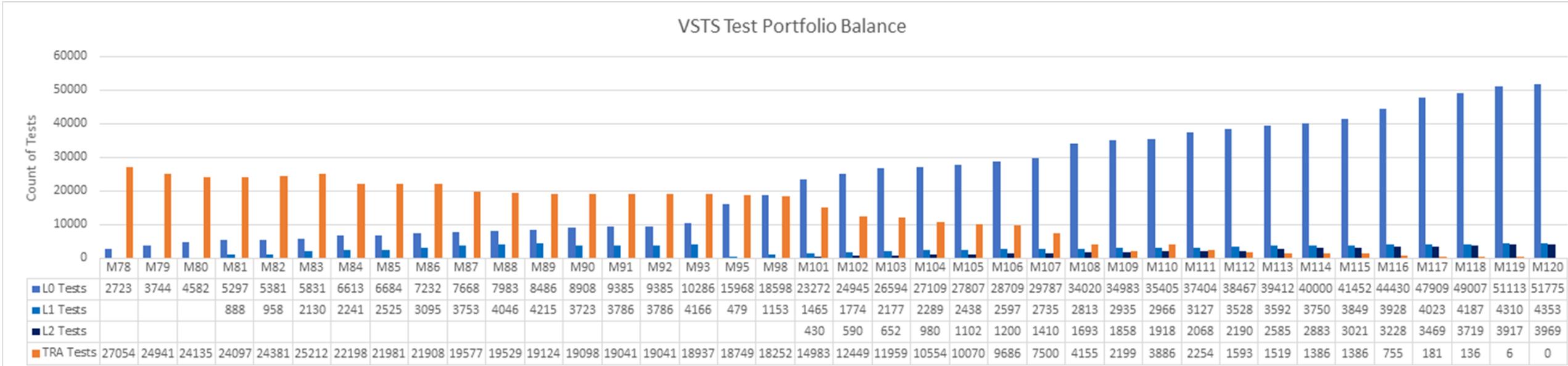
L0 – Requires only built binaries, no dependencies

L1 – Adds ability to use SQL and file system

Run L0 & L1 in the pull request builds

L2 – Test a service via REST APIs

L3 – Full environment to test end to end



# Azure Artifacts

Create and share Maven, npm, and NuGet package feeds from public and private sources – fully integrated into CI/CD pipelines



## Manage all package types

Get universal artifact management for Maven, npm, and NuGet.



## Add packages to any pipeline

Share packages, and use built-in CI/CD, versioning, and testing.



## Share code efficiently

Easily share code across small teams and large enterprises.

The screenshot shows the Azure DevOps interface for managing artifacts. On the left, there's a sidebar with icons for Overview, Boards, Repos, Pipelines, Test Plans, and Artifacts (which is selected). The main area is titled "Artifacts" and shows a list of packages. Each package entry includes the name, version, source (NuGet, npmjs, Maven), last pushed date, and a brief description. The packages listed are abbrev (Version 1.1.0), accepts (Version 1.3.3), acorn (Version 5.0.3), acorn-dynamic-import (Version 2.0.2), aclr-jsx (Version 3.0.1), acorn-object-spread (Version 1.0.0), ajv (Version 4.11.7), ajv-keywords (Version 1.5.1), and alphanum-sort (Version 1.4.0).

Package	Views	Source	Last pushed	Description
abbrev Version 1.1.0		nuget	a year ago	Like ruby's abbrev module, but in js
accepts Version 1.3.3		npmjs	a year ago	Higher-level content negotiation
acorn Version 5.0.3		MyFeed	a year ago	ECMAScript parser
acorn-dynamic-import Version 2.0.2		maven	a year ago	Support dynamic imports in acorn
aclr-jsx Version 3.0.1		nuget	a year ago	Alternative, faster React.js JSX parser
acorn-object-spread Version 1.0.0		maven	a year ago	Custom JSON-Schema keywords for ajv validator
ajv Version 4.11.7		npmjs	a year ago	Alphanumeric sorting algorithm
ajv-keywords Version 1.5.1		nuget	a year ago	ANSI escape codes for manipulating the terminal
alphanum-sort Version 1.4.0		npmjs	a year ago	An elegant lib that converts the chalked (ANSI) text to HTML



<https://azure.com/devops>

@DonovanBrown

Demo

# Azure Artifacts

# Q&A

# Thank You

ευχαριστώ Salamat Po متشرّم شكرًا Grazie

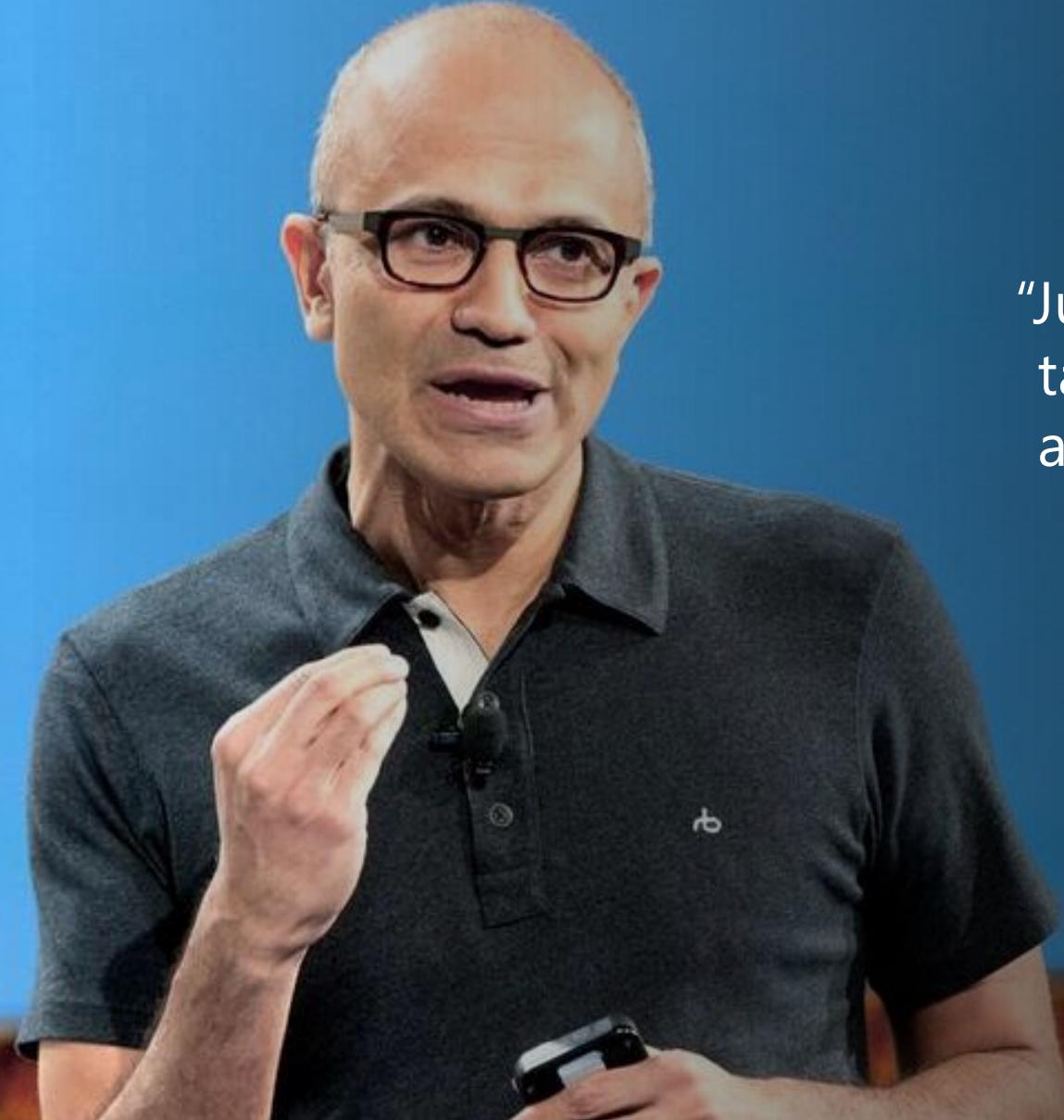
благодаря ありがとうございます Kiitos Teşekkürler 谢谢

ឧបម្ពុណមរ៉ា Obrigado شكريه Terima Kasih Dziękuję

Hvala Köszönöm Tak Dank u wel дякую Tack

Mulțumesc спасибо Danke Cám ơn Gracias

多謝晒 Ďakujem הַדּוֹל දේකුඩ් Děkuji 감사합니다



"Judge us by the actions we have taken in the recent past, our actions today and in the future"

—Satya Nadella, CEO  
Microsoft

2018

# Who are we?

## League of Extraordinary Cloud DevOps Advocates

Cloud Developer Advocates

- #LoECDA



@DonovanBrown

# Resources

## Channel 9

Azure Friday

Visual Studio Toolbox

DevOps Interviews

The DevOps Lab

## Twitter

@DonovanBrown

#LoECDA

## Web

Welcome to DevOps ([aka.ms/whatisdevops](http://aka.ms/whatisdevops))



Azure Friday  
Azure Container Registry Geo-replication



DevOps Interviews  
Interview with Munil Shah  
(Safe Deployment)



Visual Studio Toolbox  
VSTS Work Item Rules



The DevOps Lab  
The DevOps Lab Kickoff Show #1  
– Connect(); 2017 Recap

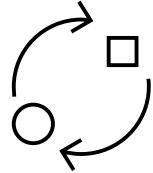
# Enterprise level compliance

## Azure policy



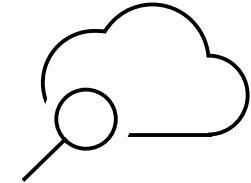
### Enforcement & compliance

- ❯ Turn on built-in policies or build custom ones for all resource types
- ❯ Real-time policy evaluation and enforcement
- ❯ Periodic & on-demand compliance evaluation



### Apply policies at scale

- ❯ Apply policies to a Management Group with control across your entire organization
- ❯ Apply multiple policies and & aggregate policy states with policy initiative
- ❯ Exclusion scope

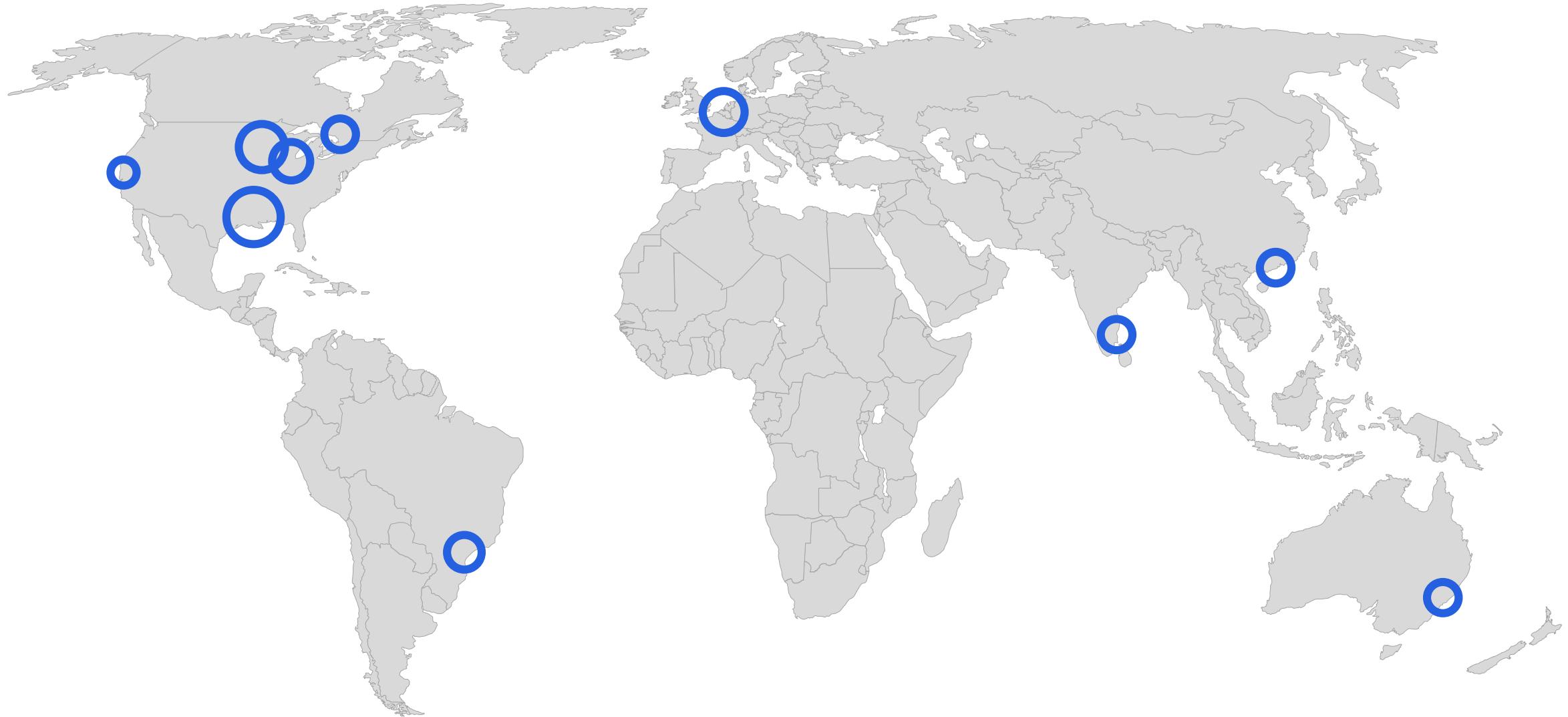


### Remediation

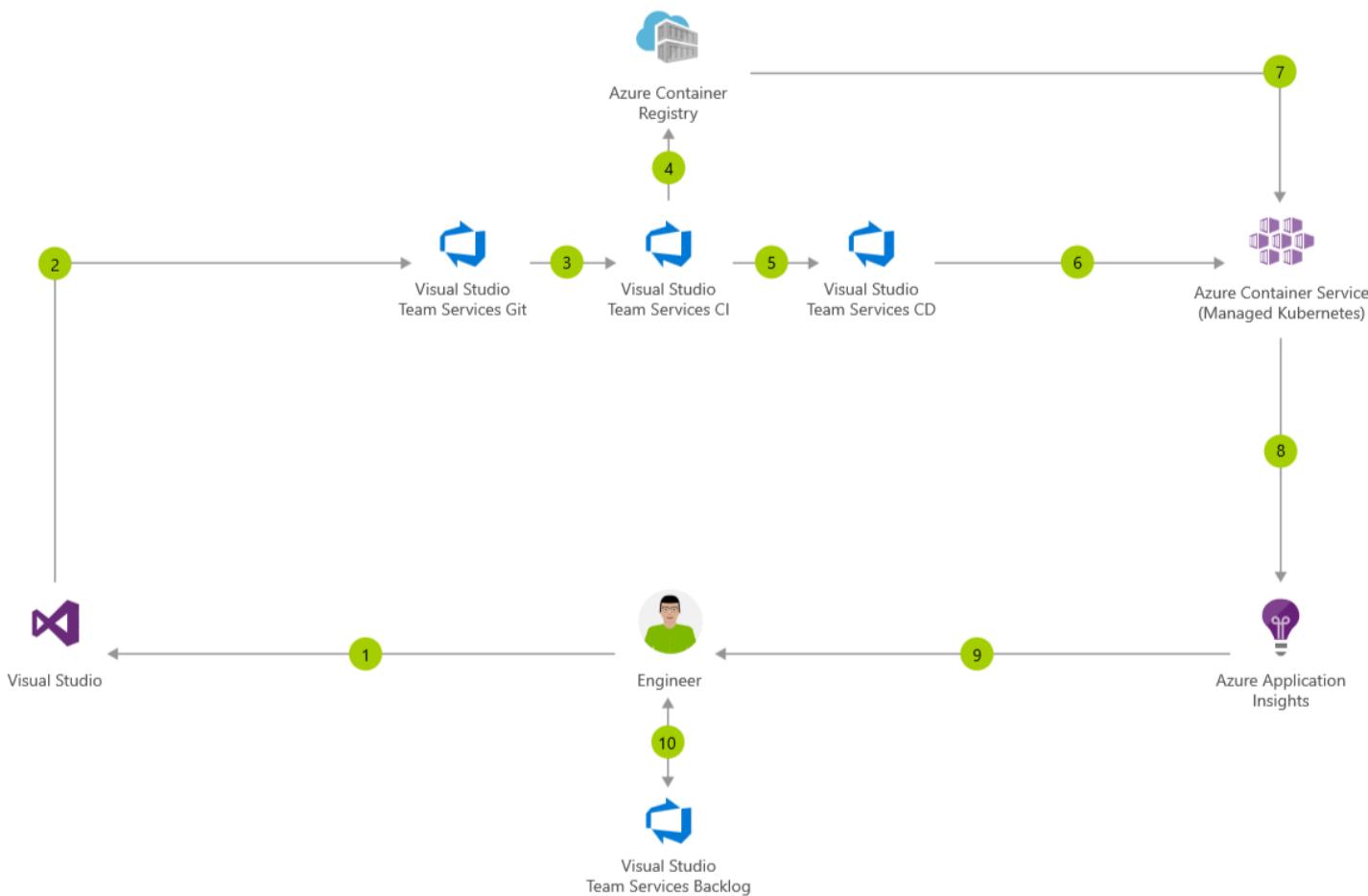
- ❯ Real time remediation
- ❯ Remediation on existing resources (Coming Soon)

# Global Reach

Azure DevOps has more global regions than any other cloud provider



# CI/CD for Containers



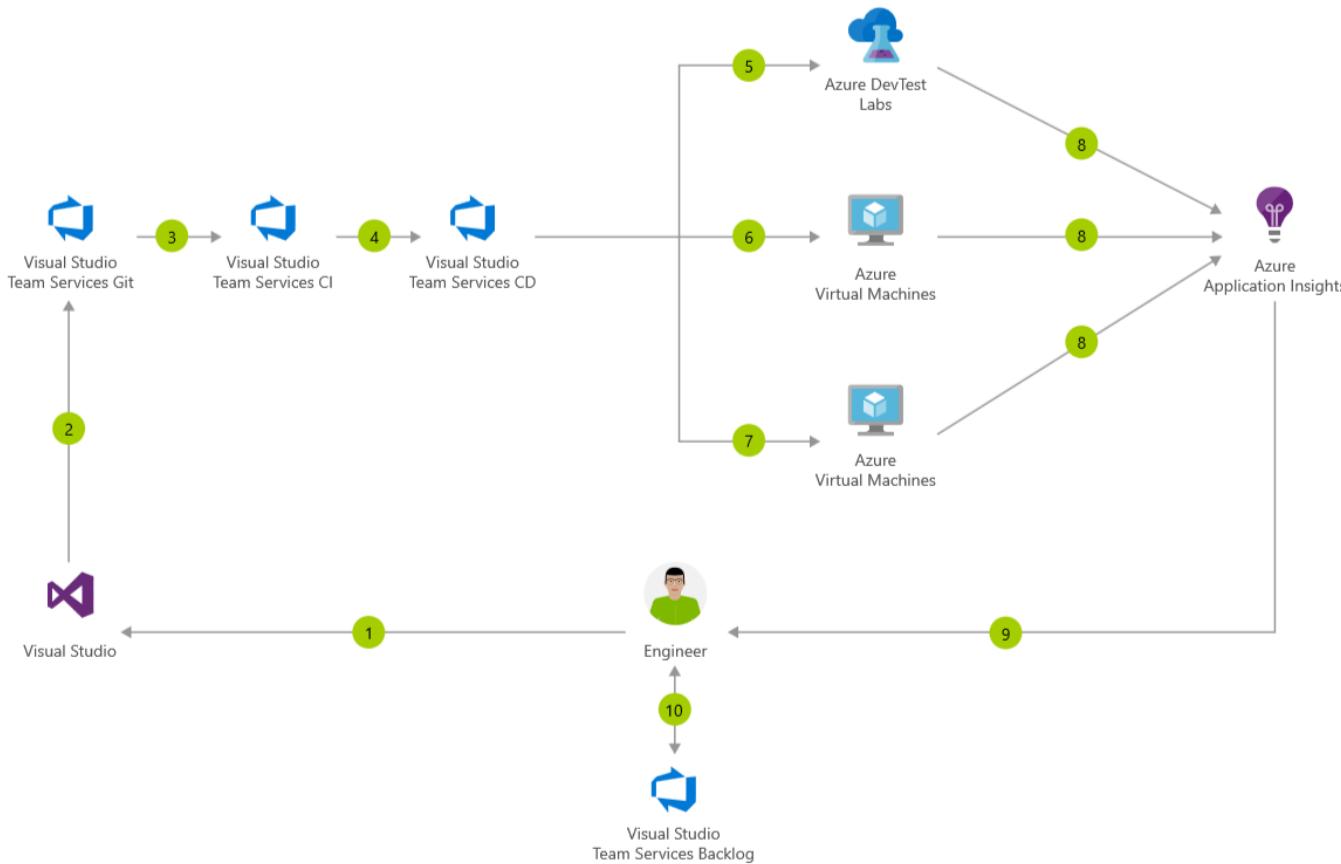
## Architecture overview

Containers make it easy for you to continuously build and deploy applications. By orchestrating the deployment of those containers using Kubernetes in Azure Container Service, you can achieve replicable, manageable clusters of containers.

By setting up a continuous build to produce your container images and orchestration, Visual Studio Team Services increases the speed and reliability of your deployment.

- 1 Change application source code.
- 2 Commit Application Code.
- 3 Continuous integration triggers application build, container image build, and unit tests.
- 4 Container image pushed to Azure Container Registry.
- 5 Continuous deployment trigger orchestrates deployment of application artifacts with environment-specific parameters.
- 6 Deployment to Azure Container Service.
- 7 Container is launched using Container Image from Azure Container Registry.
- 8 Azure Application Insights collects and analyzes health, performance, and usage data.
- 9 Review health, performance, and usage information.
- 10 Update backlog item.

# CI/CD for Azure Virtual Machines



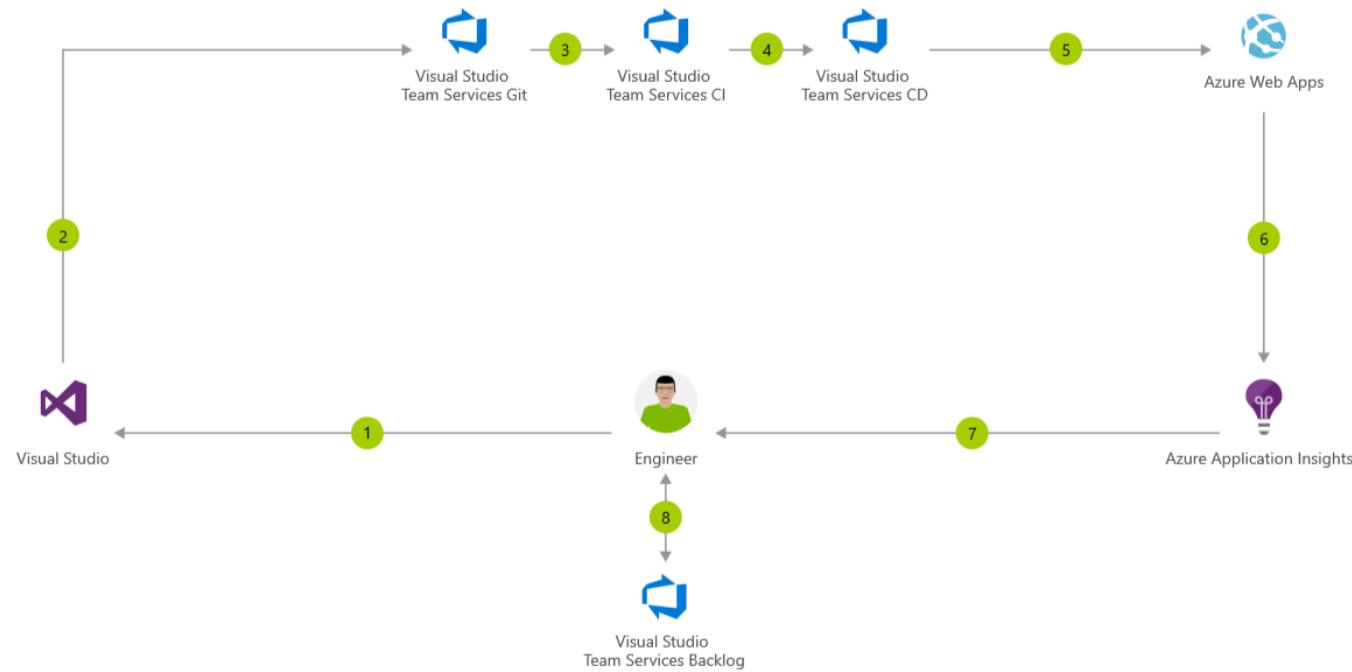
## Architecture overview

Azure is a world-class cloud for hosting virtual machines running Windows or Linux. Whether you use ASP.NET, Java, Node.js, or PHP to develop applications, you'll need a continuous integration and continuous deployment (CI/CD) pipeline to push changes to these virtual machines automatically.

Visual Studio Team Services provides the CI/CD pipeline, starting with a Git repository for managing your application source code and infrastructure code (Azure Resource Manager templates), a Build system for producing packages and other build artifacts, and a Release Management system for setting up a pipeline to deploy your changes through dev, test, and production environments. The pipeline uses Resource Manager templates to provision or update your infrastructure as necessary in each environment, and then deploys the updated build. You can also use Azure DevTest Labs to automatically tear down test resources that are not in use.

- 1 Change application source code.
- 2 Commit application code and Azure Resource Manager template.
- 3 Continuous integration triggers application build and unit tests.
- 4 Continuous deployment trigger orchestrates deployment of application artifacts with environment-specific parameters.
- 5 Deployment to QA environment.
- 6 Deployment to staging environment.
- 7 Deployment to production environment.
- 8 Azure Application Insights collects and analyzes health, performance, and usage data.
- 9 Review health, performance, and usage information.
- 10 Update backlog item.

# CI/CD for Azure Web Apps



## Architecture overview

Azure Web Apps is a fast and simple way to create web apps using ASP.NET, Java, Node.js, or PHP. Deliver value faster to your customers with a continuous integration and continuous deployment (CI/CD) pipeline that pushes each of your changes automatically to Web Apps.

- 1 Change application source code.
- 2 Commit application code and Web Apps web.config file.
- 3 Continuous integration triggers application build and unit tests.
- 4 Continuous deployment trigger orchestrates deployment of application artifacts with environment-specific parameters.
- 5 Deployment to Web Apps.
- 6 Azure Application Insights collects and analyzes health, performance, and usage data.
- 7 Review health, performance, and usage information.
- 8 Update backlog item.

# Tech use differentiation

# DevOps tool integrations

Your favorite toolchain, seamlessly integrated with Azure



## The Best Development Experience

Whether you're developing on Azure, on-premise, or another cloud, Visual Studio Code extensions from Microsoft and the community help accelerate development across Linux, macOS and Windows



## First-class Integration

Microsoft collaborates directly in open source projects with our partners, and the community, to bring native Azure integration. Many of these tools are also directly available in Azure Cloud Shell – try them out!



## Accelerated Customer Success

You can get clear guidance for integrating your favorite tools with Azure, with dedicated documentation hubs and example solution architecture. Get started, fast.



ANSIBLE



CHEF



Packer



Terraform

The screenshot shows a web browser displaying the Microsoft Azure documentation for Jenkins. The URL in the address bar is <https://docs.microsoft.com/en-us/azure/jenkins/>. The page title is "Jenkins on Azure". The left sidebar has a "Jenkins on Azure" section with links for Overview, Quickstarts (Create a Jenkins server, Tutorials), How-to (Secure Jenkins on Azure, Use the App Service plugin, Publish to Azure Storage, Use Jenkins with VSTS), Samples (Sample jobs and scripts, Resources), and a "Download PDF" button. The main content area has sections for "Jenkins on Azure" (describing building apps in the cloud and deploying to Azure), "5-Minute Quickstarts" (Create a Jenkins system in Azure and build an app), "Azure Portal" (link to the Azure Portal icon), "Step-by-Step Tutorials" (Set up continuous integration for your apps, run agents in the cloud, and use Azure services in your jobs), "Samples" (Sample scripts and stages for your pipelines, Jenkinsfile stages and snippets), "Reference" (link to the Reference icon), and "Plugins" (link to the Plugins icon). The top navigation bar includes links for Why Azure, Solutions, Products, Documentation, Pricing, Training, Marketplace, Partners, Blog, Resources, and Support. There are also sales contact links at the top right.

# Ansible



## Azure DevOps Tool Integrations

Bringing native Azure support for customers using Ansible

[Documentation Hub for Jenkins](#)

[Ansible in Azure Cloud Shell](#)

[Visual Studio Code Extension](#)

[Azure Modules](#)

[Azure Preview Modules](#)

[Azure Playbook Samples](#)

A screenshot of the Visual Studio Code interface. The title bar says "vm.yml - ansible-testapp - Visual Studio Code". The left sidebar shows a file tree with files like ".gitignore", "azur...-createvm.yml", "credentials.yml", "role.yml", "test.yml", "vm.retry", and "vm.yml". The main editor area shows an Ansible YAML file for creating an Azure VM. A context menu is open over the "vm.yml" file in the editor, listing options such as "Run Ansible Playbook in Docker", "Run Ansible Playbook in Local Ansible", and "Run Ansible Playbook in Cloud Shell".

```
---  
- name: Create Azure VM  
hosts: localhost  
connection: local  
vars:  
  resource_group: demo-451  
  vm_name: testvm  
  location: eastus  
tasks:  
  - name: Create a resource group  
    azure_rm_resourcegroup:  
      name: "{{ resource_group }}"  
      location: "{{ location }}"  
    create virtual network  
    #_virtualnetwork:  
    resource_group: "{{ resource_group }}"  
    #_vm_name }}"  
    #_ss_prefixes: "10.0.0.0/16"  
    #_subnet:  
    #_resource_group: "{{ resource_group }}"  
    #_vm_name }}"  
    #_ss_prefix: "10.0.1.0/24"  
    #_al_network: "{{ vm_name }}"  
    #_create public IP address  
    #_publicipaddress:  
    #_resource_group: "{{ resource_group }}"  
    #_name: "{{ vm_name }}"  
    #_location: "{{ location }}"  
    #_allocation_method: Static  
    #_name: "{{ vm_name }}"  
  - name: Create Network Security Group that allows SSH  
    azure_rm_securitygroup:  
      resource_group: "{{ resource_group }}"  
      name: "{{ vm_name }}"  
      rules:
```





## Azure DevOps Tool Integrations

Bringing native Azure support for customers using Chef

[Documentation Hub for Chef](#)

[Chef in Azure Cloud Shell](#)

[Chef VS Code Extension](#)

[Habitat VS Code Extension](#)

[InSpec VS Code Extension](#)



The screenshot shows a terminal window titled 'Bash' with a dark theme. The command entered is:

```
shell@Azure:~$ inspec exec virtual_machine_internal_vm.rb -t azure://$subscription_id
```



[docs.microsoft.com/azure/chef](https://docs.microsoft.com/azure/chef)

# Jenkins



## Azure DevOps Tool Integrations

Bringing native Azure support for customers using Jenkins

[Documentation Hub for Jenkins](#)

[Jenkins Solution Template](#)

Dynamic Build Agents:

[Virtual Machine](#)    [Container \(ACS/AKS & ACI\)](#)

Continuous Delivery:

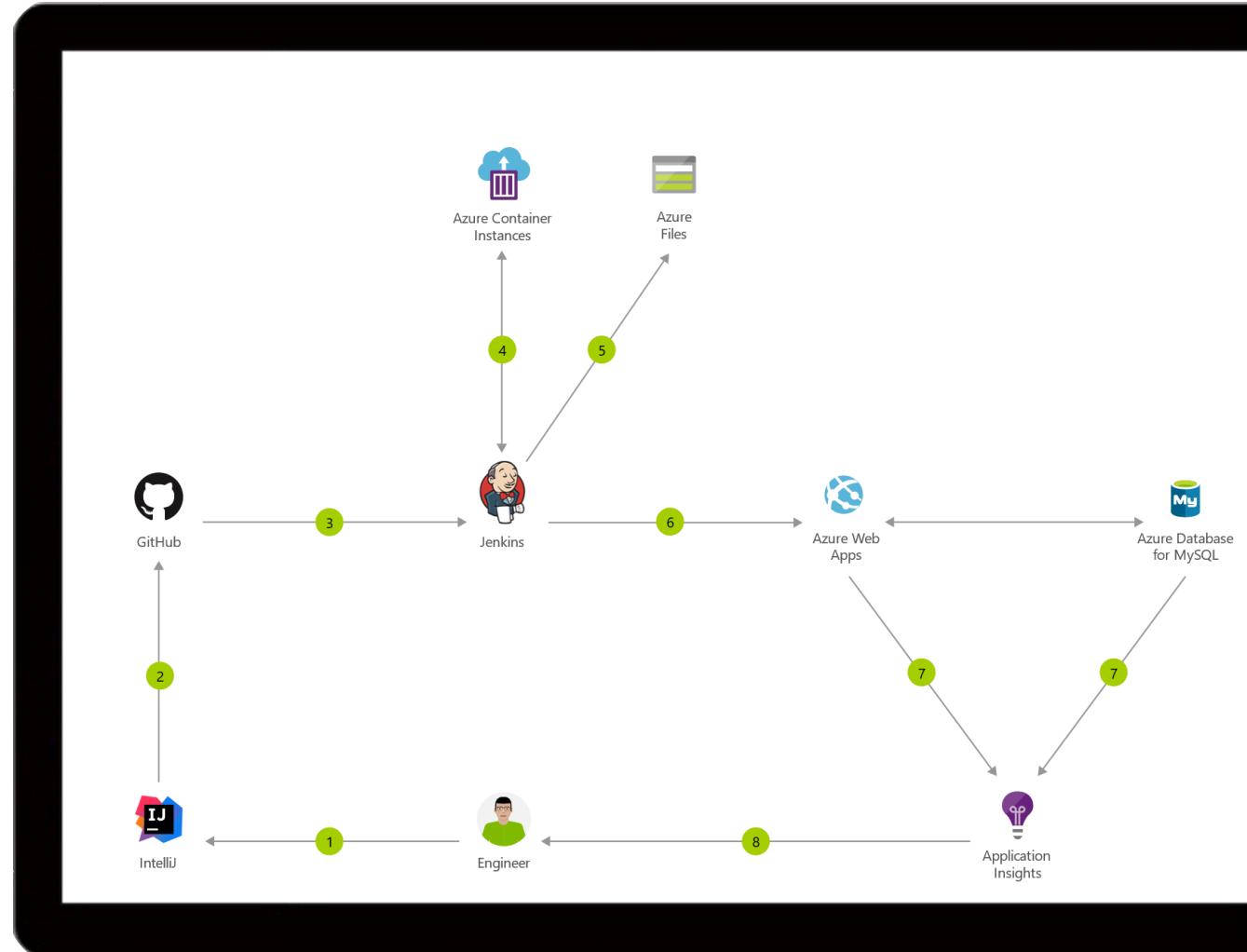
[App Service](#)

[VMSS](#)

[Service Fabric](#)

[ACS & AKS](#)

[Functions](#)





HashiCorp  
**Terraform**

# Terraform

## Azure DevOps Tool Integrations

Bringing native Azure support for customers using Terraform

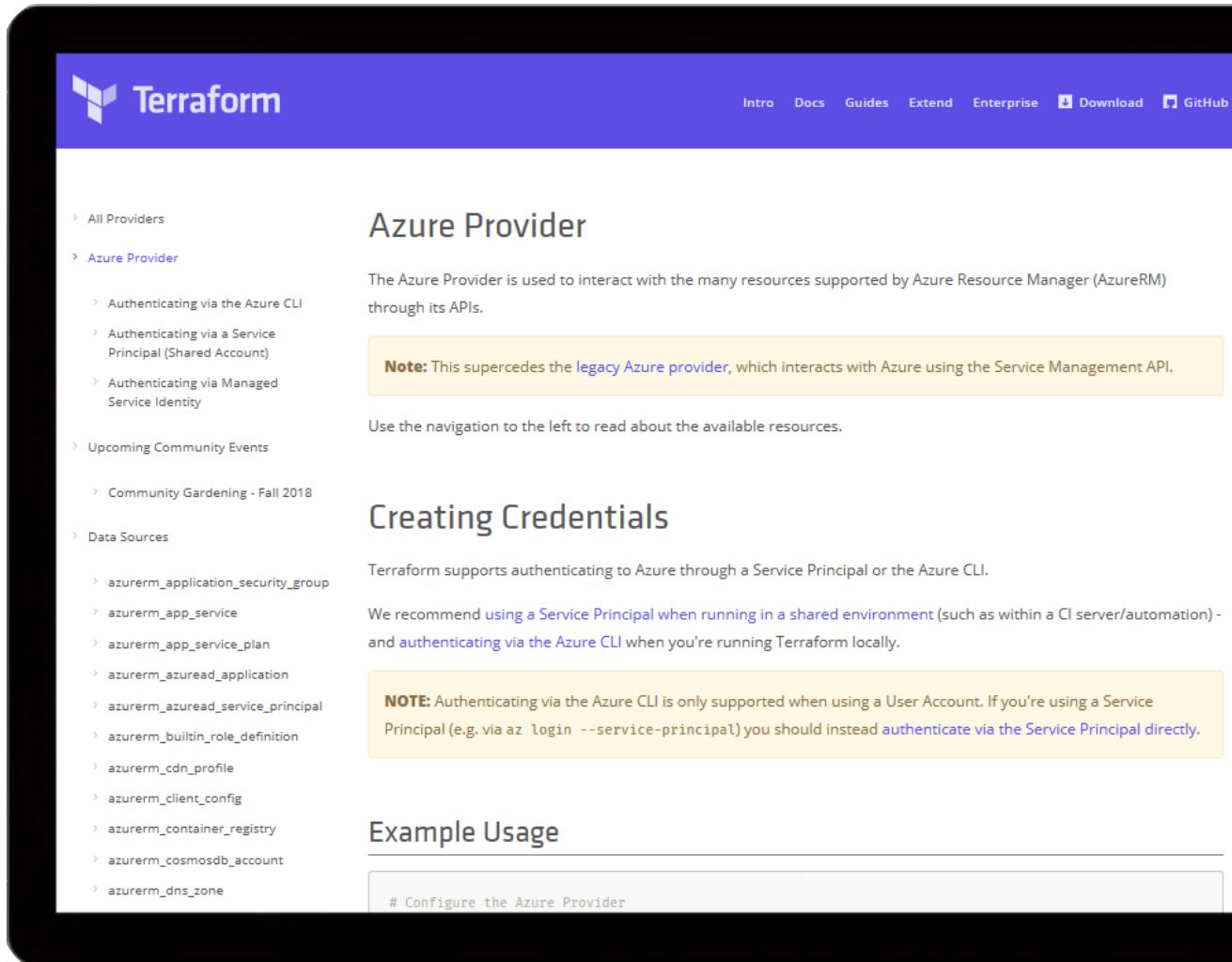
[Documentation Hub for Terraform](#)

[Terraform in Azure Cloud Shell](#)

[Azure Resource Provider](#)

[Azure Module Registry](#)

[Azure Cloud Shell Integration](#)



The screenshot shows the Terraform documentation website. The top navigation bar includes links for Intro, Docs, Guides, Extend, Enterprise, Download, and GitHub. The main content area is titled "Azure Provider". It describes the provider's use for interacting with Azure Resource Manager (ARM) through APIs. A note states it supercedes the legacy Azure provider. A sidebar on the left lists various providers and resources, including "azurerm\_application\_security\_group", "azurerm\_app\_service", "azurerm\_app\_service\_plan", "azurerm\_azuread\_application", "azurerm\_azuread\_service\_principal", "azurerm\_builtin\_role\_definition", "azurerm\_cdn\_profile", "azurerm\_client\_config", "azurerm\_container\_registry", "azurerm\_cosmosdb\_account", and "azurerm\_dns\_zone". Below the provider description is a section titled "Creating Credentials" which explains authentication methods. A note in this section states that authenticating via the Azure CLI is only supported for User Accounts, while Service Principals require a different approach. At the bottom, there is an "Example Usage" section with a code snippet.

**Azure Provider**

The Azure Provider is used to interact with the many resources supported by Azure Resource Manager (AzureRM) through its APIs.

**Note:** This supercedes the [legacy Azure provider](#), which interacts with Azure using the Service Management API.

Use the navigation to the left to read about the available resources.

**Creating Credentials**

Terraform supports authenticating to Azure through a Service Principal or the Azure CLI.

We recommend [using a Service Principal when running in a shared environment](#) (such as within a CI server/automation) - and [authenticating via the Azure CLI](#) when you're running Terraform locally.

**NOTE:** Authenticating via the Azure CLI is only supported when using a User Account. If you're using a Service Principal (e.g. via `az login --service-principal`) you should instead [authenticate via the Service Principal directly](#).

**Example Usage**

```
# Configure the Azure Provider
```



# Jet.com

## Customer need

"We realized that we simply did not have the resources to build and manage the kind of datacenters and development infrastructure to meet our growth strategy," says Mike Hanrahan, CTO at Jet, "So we quickly decided on a cloud model."

## Results

Jet.com was able to take advantage of Azure and Visual Studio to go from zero to a full-fledged e-commerce marketplace in just over 12 months.

"To be one of the best e-commerce destinations in the US, we will have to handle millions of customers, placing tens of thousands of orders a day. That requires a top-class e-commerce system built on a flexible, open cloud platform. That is exactly what we got with Azure," says Hanrahan.

Source: <https://customers.microsoft.com/en-gb/story/jetcustomerstory>





## Customer need

Challenged by increased customer appetite for 24/7 digital engagement, Geico needed to deliver new services and features to customers faster.

Traditionally new feature rollouts to their tier 1 business systems often results in downtime that could be devastating to their online business.

---

## Results

By adopting a DevOps development strategy and transitioning development to Microsoft Azure, GEICO is working to engage customers more personally and dynamically.

"With DevOps and Azure, we're able to reduce our new-feature release cycle down to one week, and we think we can even speed that up." – Fikri Larguet, Director of Cloud Services

Source: <https://customers.microsoft.com/en-us/story/geico>



# Ambit Energy

## Customer need

Ambit developers could move very fast on the software side but had to wait months for servers to be ordered and provisioned.

"We wanted our infrastructure velocity to match our software velocity," – Robert Rudduck, Director of Architecture and DevOps at Ambit Energy.

---

## Results

"Speeding up our software delivery engine has had a huge impact on our business," Phillips says. "It's enabled us to introduce new services faster, move into new markets, and respond to the everyday needs of the business."

"Electricity is a very competitive market, so the lower operational costs we achieve with Azure help us price our service more competitively and earn higher profits"



Source: <http://customers.microsoft.com/en-us/story/ambit-energy>

# Alaska Airlines

## Customer need

Alaska has requirements to build mobile apps for both their customers as well as their internal employees. Alaska needed an end-to-end platform to manage their development process, and build, test, deployment and monitoring across a variety of devices.

## Results

Used Azure DevOps, Xamarin and Visual Studio App Center (formerly Xamarin Test Cloud)

"With Xamarin, we deliver value quickly, integrating native experiences like Touch ID and push notifications, and using Xamarin Test Cloud automation to run our test suite on thousands of devices. We don't think about the tools anymore, we think about what experience we want to deliver to our users." – Mike Lorengo, Director IT Enterprise Architecture

Source: <https://customers.microsoft.com/en-us/story/visual-studio-alaska-airlines>



# RMS

## Customer need

Before adopting Azure and Azure DevTest Labs, getting hardware deployed for dev/test could take a month or more.

## Results

Now, the time between request and delivery of environments for app development is hours if not minutes. At RMS, IT is supporting the mission and helping the business get more done.

Source: <https://customers.microsoft.com/en-us/story/risk-management-solutions>

