

DevOps in a .NET World

Matteo Tumiatì

Senior Consultant @icubed

Windows Dev MVP

matteot@aspitalia.com | @xtumiox



Agenda

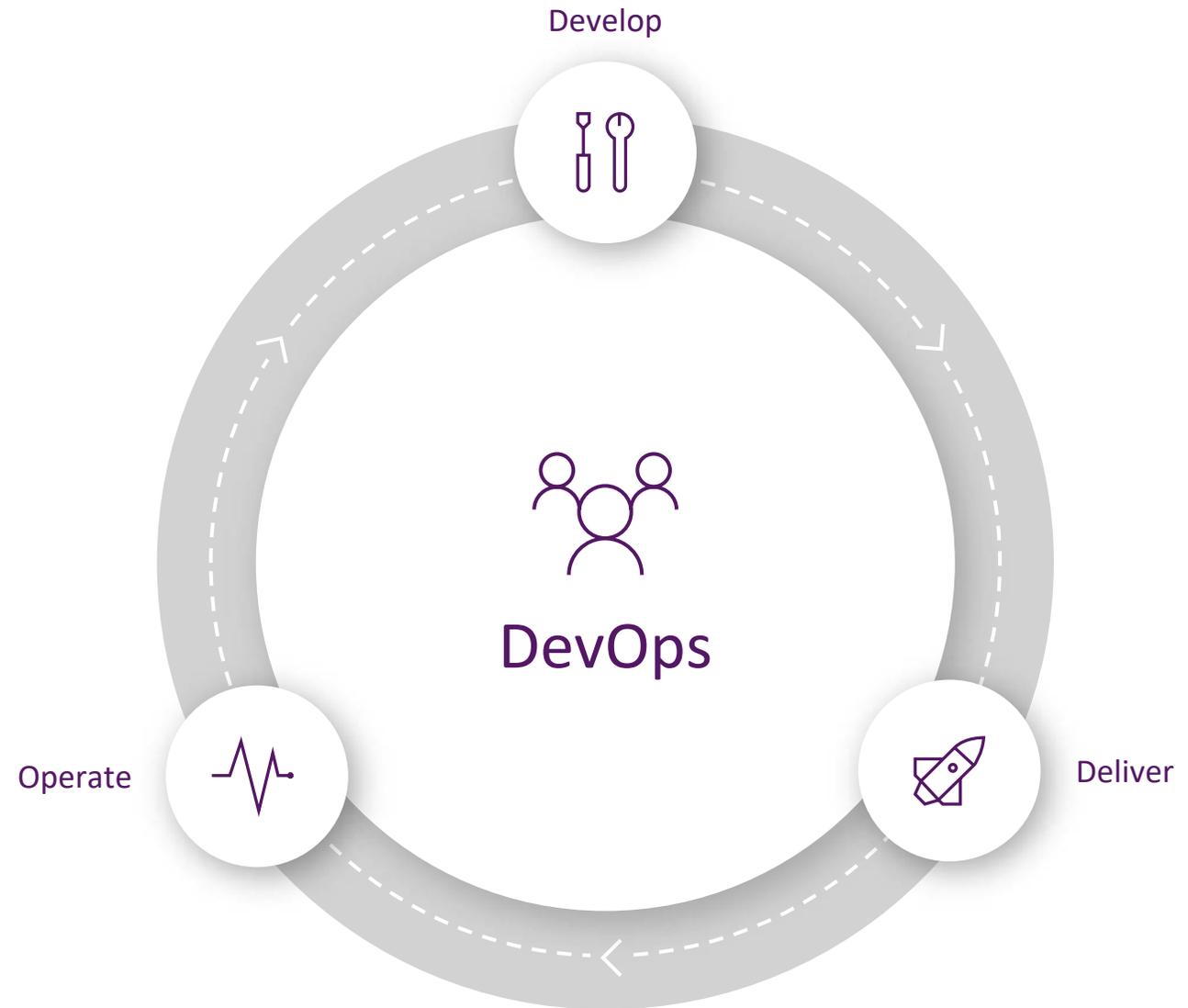
(Azure) DevOps



What is DevOps?



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



Common DevOps practices

Infrastructure as Code (IaC)

Continuous Integration

Automated Testing

Continuous Deployment

Release Management

App Performance Monitoring

Load Testing & Auto-Scale

Availability Monitoring

Capacity Management

Change/Configuration Management

Feature Flags

Automated Environment De-Provisioning

Self Service Environments

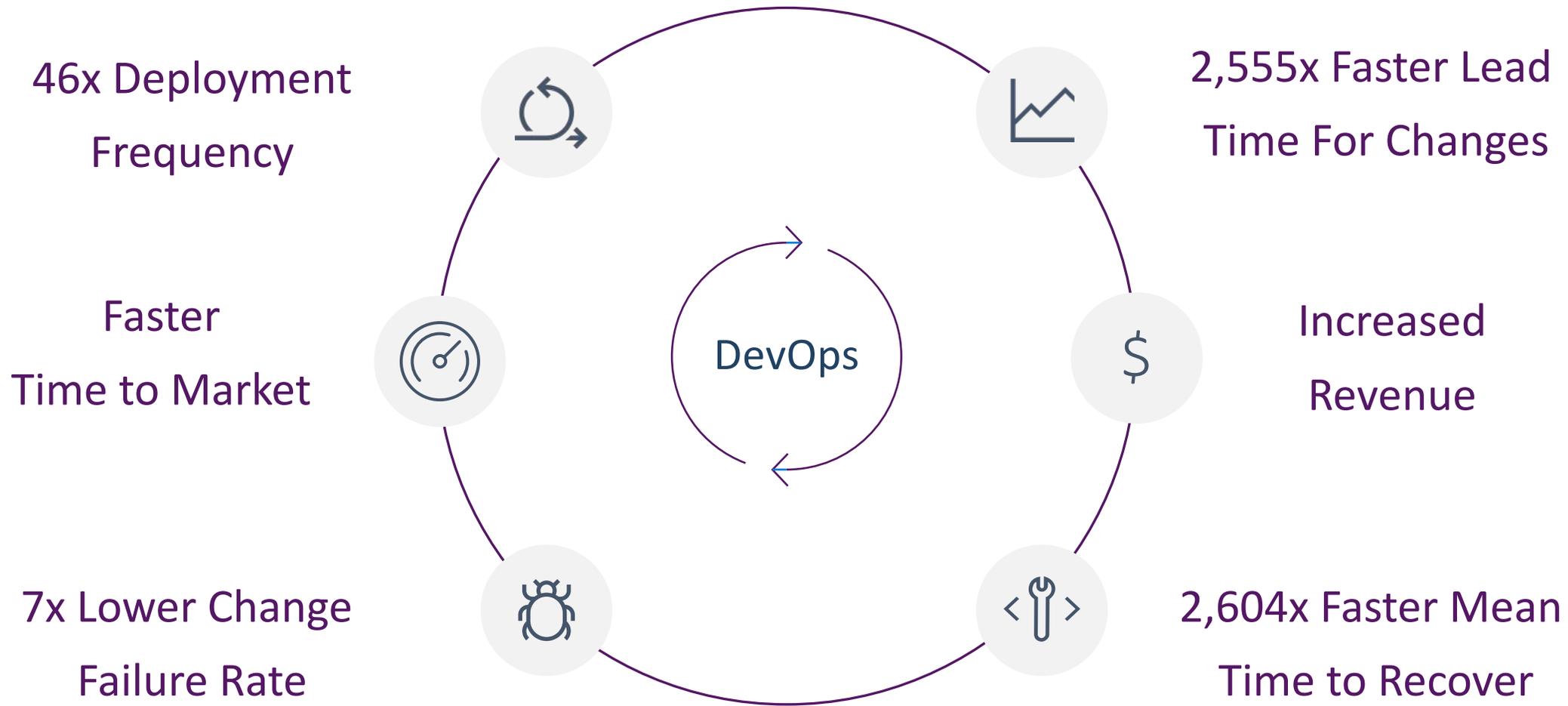
Automated Recovery (Rollback & Roll-Forward)

Hypothesis Driven Development

- Testing in Production

- Fault Injection

- Usage Monitoring / User Telemetry



Source: 2018 Accelerate: State of DevOps: Strategies for a New Economy." N. Forsgren, J. Humble, G. Kim. DevOps Research and Assessment (DORA)

DevOps at Microsoft

We have over 96,000 active engineering employees internally critically reliant on Azure DevOps & GitHub to ship Microsoft software & services

Microsoft experiments on themselves and then bring things to market

The innovations made in creating a cloud-scale engineering system provide value both to Microsoft and its customers

500m

Test executions per day

372k

Internal pull requests a month

86,000

Deployments per day

7,700

Employees contributing to open source

2m

Private Git commits per month

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>

Benefits of a cloud-native solution

Global availability

Hosted and maintained by Microsoft with 99.9% uptime guarantee and 24x7 support

Immediate access to latest features

Simplified deployment to Azure

Security

Strong integration with Active Directory (or AAD)

Also works on cloud with on-prem AD through AD Connect

Only members inside AD can join a VSTS account / team project (external guest access may be enabled inside AD)

Multi factor authentication

Conditional access

Access may be granted through conditional access policies (based on group membership, device, location and so on...)

Security

User provisioning and de-provisioning

- Provisioning is automatic thanks to AAD

- Users gain access almost immediately

Access management

- Permissions based on AD groups

Account and team project policies

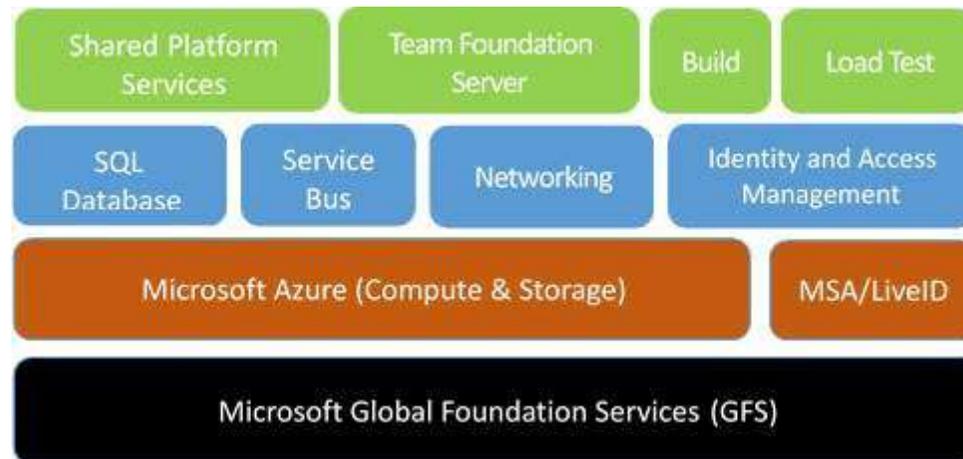
- Granular policies for branch, build, release, package and so on for each user...

Data availability

VSTS is built on Microsoft Azure platform

Data geo-replica in opposite DCs within the same region

Daily backup of the entire system



Azure DevOps pricing

Open Source Projects

Free

Unlimited users and build time

Azure Pipelines: 10 parallel jobs with unlimited minutes for CI/CD

Azure Boards: Work item tracking and Kanban boards

Azure Repos: Unlimited public Git repos

Small Teams

Free

Start free with up to 5 users

Azure Pipelines: Run 1 Microsoft-hosted job for 1,800 minutes per month and 1 self-hosted job for any amount of time

Azure Artifacts: package management

Unlimited stakeholders

Teams of any size

Starts at \$6

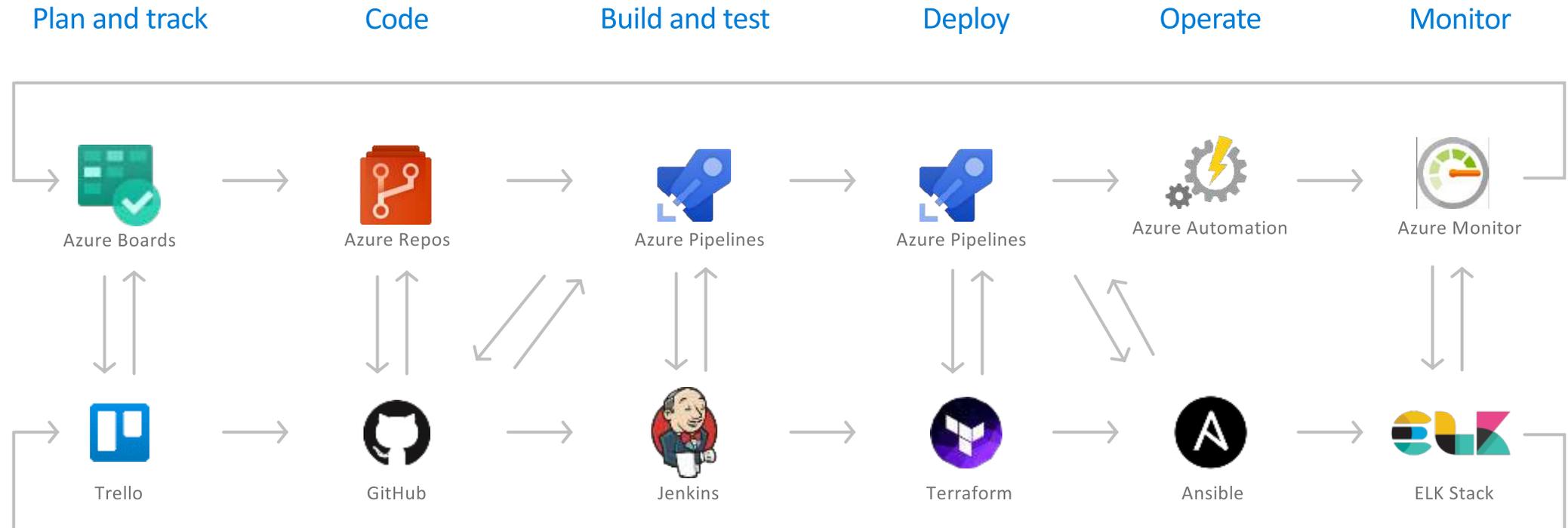
per user, per month for Boards & Repos*

Boards & Repos included for Visual Studio subscribers

<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.

Interoperability



Azure Boards



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.



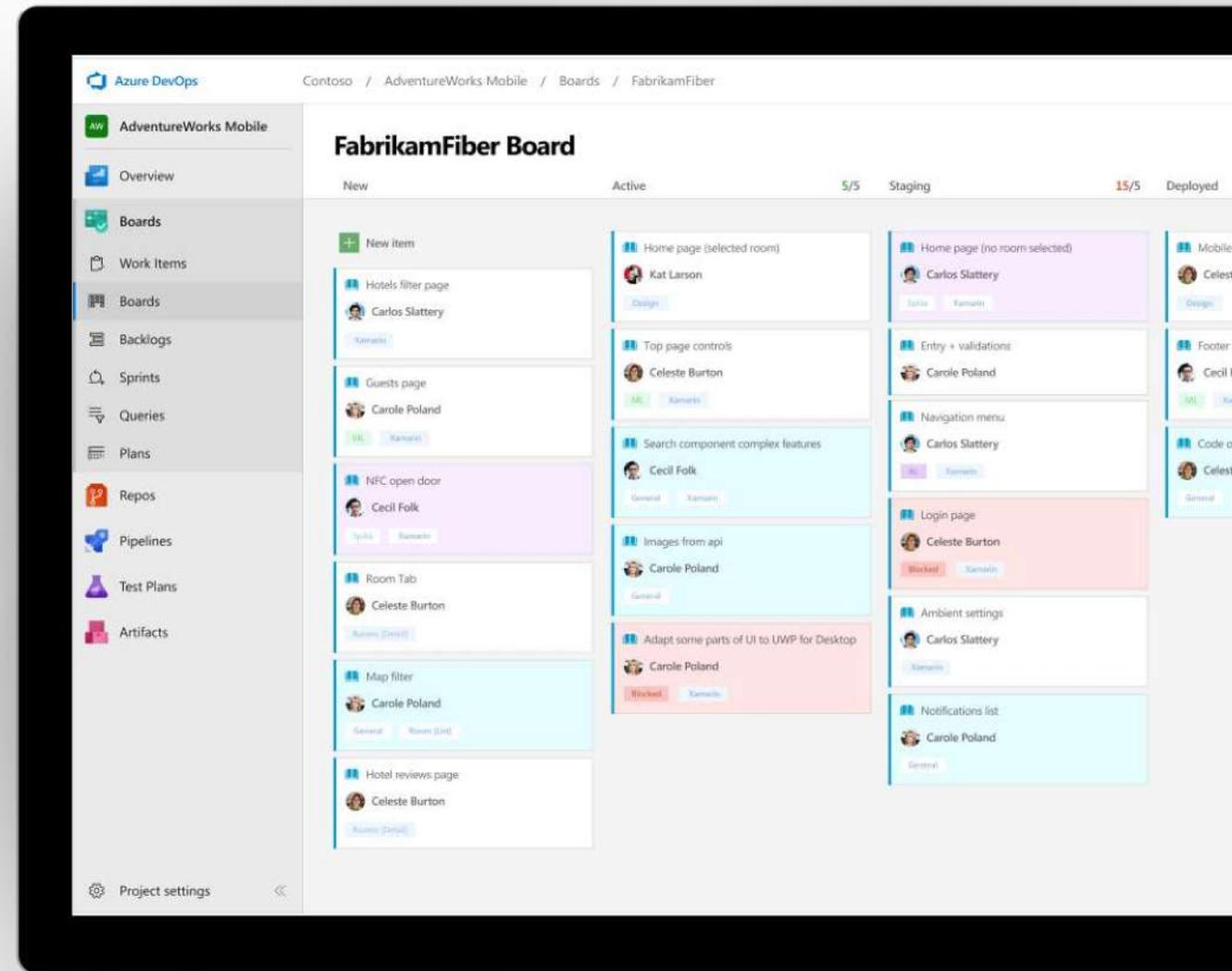
Scrum ready

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.



Backlog: validation rules

Work items cannot really contain only the title...

- Or a description that matches the title
- For bugs we need repro steps, build identification number and so on...
- Maybe it is due to missing refinement but...
- We need rules

Solution: Custom fields, validation rules

+ New rule

Empty activity <small>When no value is defined for Activity</small>	
Empty Repro Steps <small>When no value is defined for Repro Steps</small>	
Empty version number <small>When no value is defined for Found In</small>	
On Dev In Progress ... <small>When a work item state is 'Dev in progress'</small>	
On Ready for development <small>When a work item state is 'Ready for development'</small>	
On Ready for sprint <small>When a work item state is 'Ready for sprint'</small>	

Name:

Conditions ⓘ

When:

+ New condition

Actions ⓘ

Then:

And:

And:

+ New action

Save Cancel

Backlog: styling rules

It can become surprisingly difficult to identify high priority bugs or work items inside the backlog

Solution: styling rules



The screenshot shows the 'SETTINGS' dialog for 'Styles'. The 'Styles' section is active, showing a list of styling rules. The 'Stale tasks' rule is selected and enabled. The rule name is 'Stale tasks', and the color is set to red. The rule criteria are defined as follows:

Field	Operator	Value
Changed Date	>	@Today - 5
Work Item Type	=	Task

Buttons at the bottom include 'Save', 'Save and close', and 'Cancel'.

Query

It might be critical to understand what will be delivered in the next release

- So much time between releases
- Too many work item types

Involve the dev team in the refinement process...

Notify your managers with raw data...

Queries > Shared Queries > Reporting > Releases > Features to be released in 1.5

Results Editor Charts Run query New Save query Save as... Rename Revert changes Column options

Type of query Tree of work items

Filters for top level work items

And/Or	Field	Operator	Value
<input type="checkbox"/>	Work Item Type	=	Release
<input type="checkbox"/>	ID	=	30730

Filters for linked work items

And/Or	Field	Operator	Value
<input type="checkbox"/>	Work Item Type	In	Feature,Product Backlog Item,Bug,Task
<input type="checkbox"/>	State	<>	Removed

Filter options: Match top-level work items first | Type of tree: Parent/Child

ID	Work Item Type	Title	Assigned To	State
8	Epic	Wiki Social	Adithya Ramasubramanian	Active
9	Feature	Follow Wiki Pages	Amit Kumar	Closed
10	User Story	User should be able to follow a wiki page	Amit Kumar	Closed
11	User Story	User should get notified when a page is edited	Amit Kumar	Closed
12	User Story	User should be able to unfollow a followed wiki page	Amit Kumar	Closed
13	Feature	Comments in Wiki	Amit Kumar	Active
14	User Story	User should be able to add comments to a wiki page	Amit Kumar	Active
15	User Story	User should be able to edit a comment	Amit Kumar	New
16	User Story	User should be able to delete a comment	Amit Kumar	New

Azure Repos



Works with your Git client

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



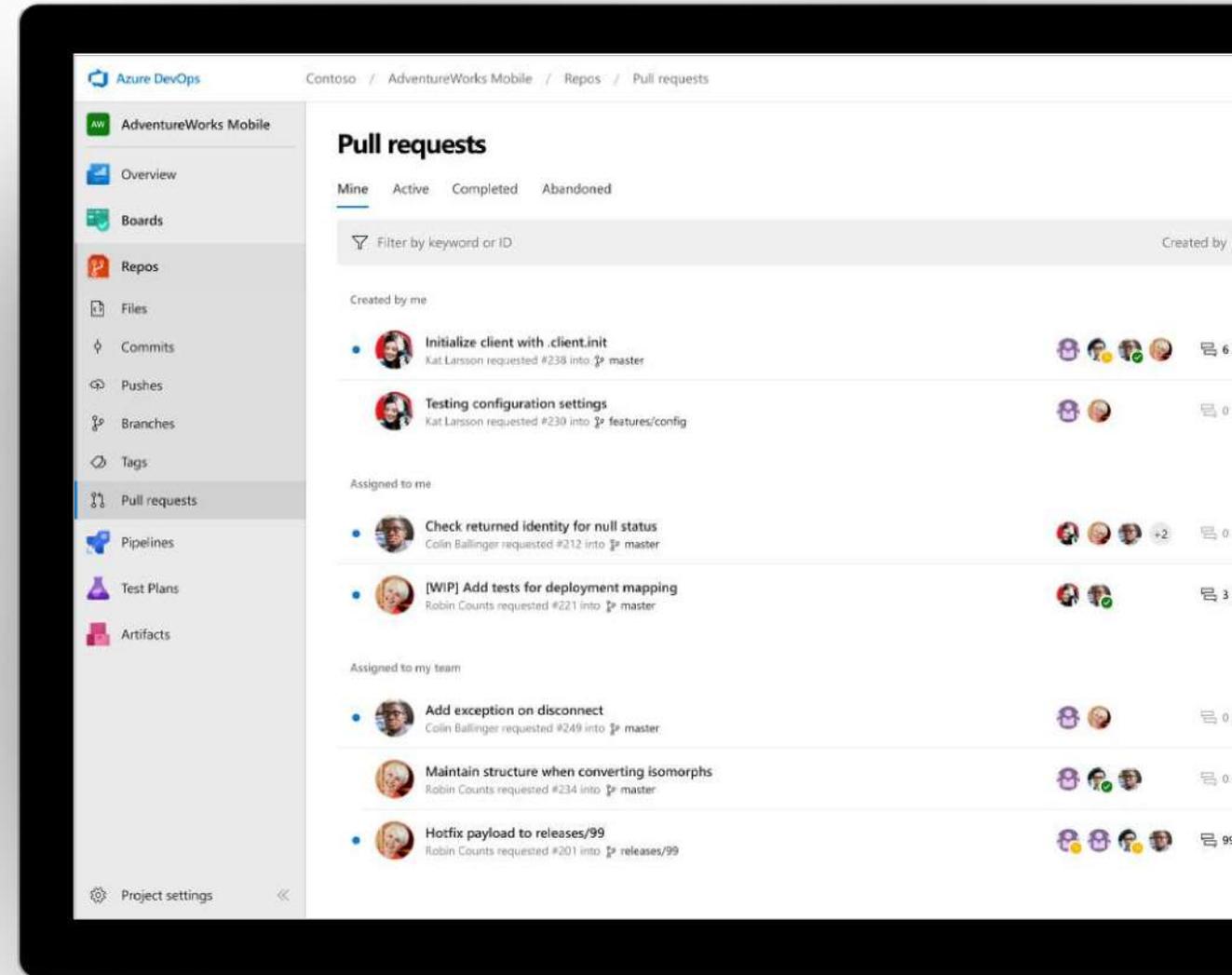
Web hooks and API integration

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.



Branch policies

Used to protect branches from unwanted commits

- «dangerous» commits in the middle of a new release

You *must* create a PR

- Unless you bypass the policies
- Higher quality standards and better engagement from the entire team

More traceability

Can be applied to any branch

The screenshot shows the GitHub 'Branches' page for user 'users/jamal'. It lists several branches: 'international-address-support', 'readme-fix', 'develop', and 'master'. The 'master' branch is marked as 'Default' and 'Compare'. A context menu is open over the 'master' branch, with 'Branch policies' highlighted. A red circle with the number '1' is around the menu trigger, and another red circle with the number '2' is around the 'Branch policies' option.

Branch	Commit	Author
international-address-support	4162b62f	Jamal Hartnett
readme-fix	e3b6ea83	Jamal Hartnett
develop	9bdd18e	Jamal Hartnett
master	4162b62f	Jamal Hartnett

Policies

Required

- ✓ 1 reviewer approved
- ✓ Required reviewers have approved
- ✓ Work items linked

Work Items

3 ReadMe.txt missing from project

Reviewers

Required

- Johnnie McLeod Approved

Optional

- Jamal Hartnett

Azure Test Plans



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.



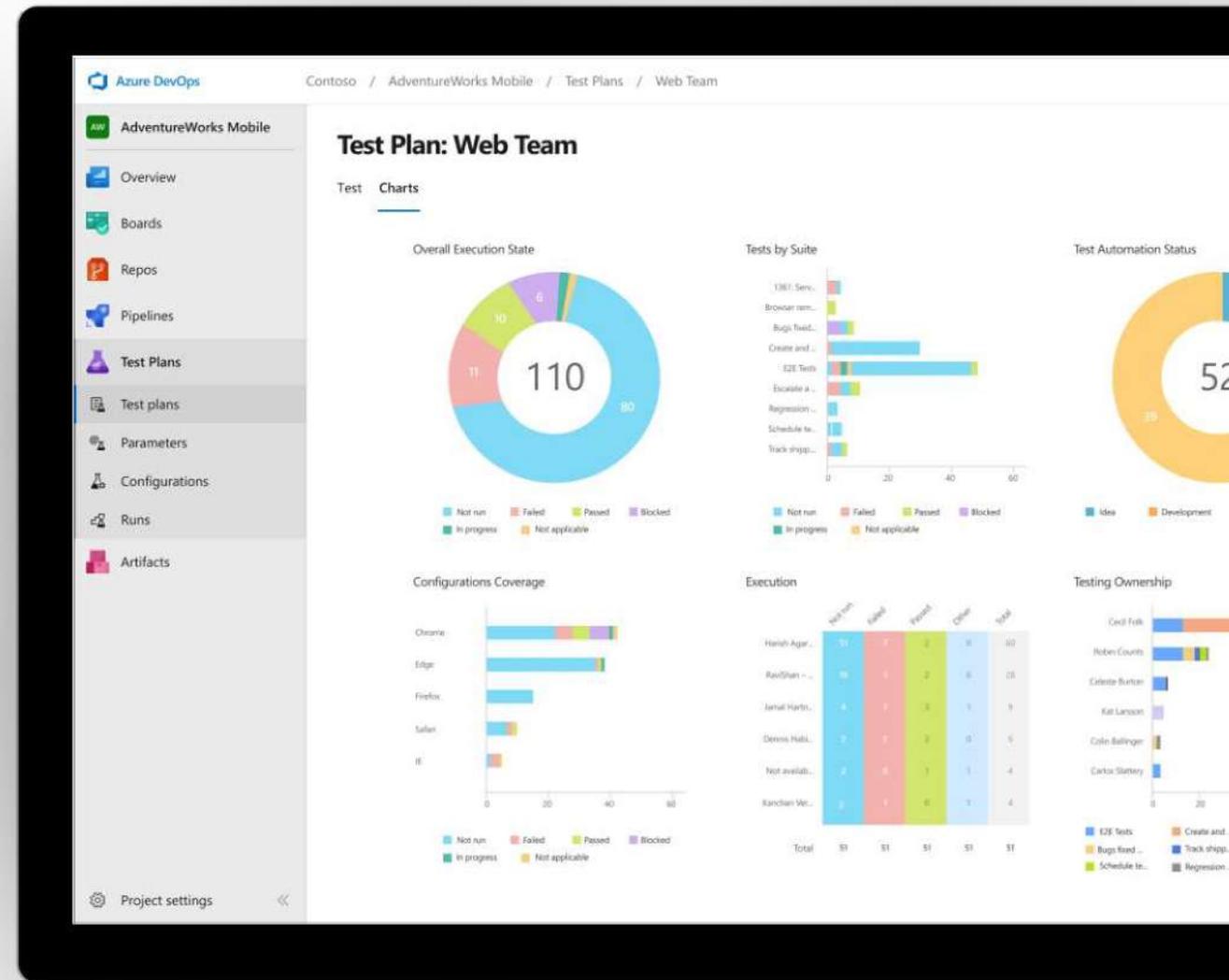
Test across web and desktop

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.



Azure Artifacts



Manage all package types

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



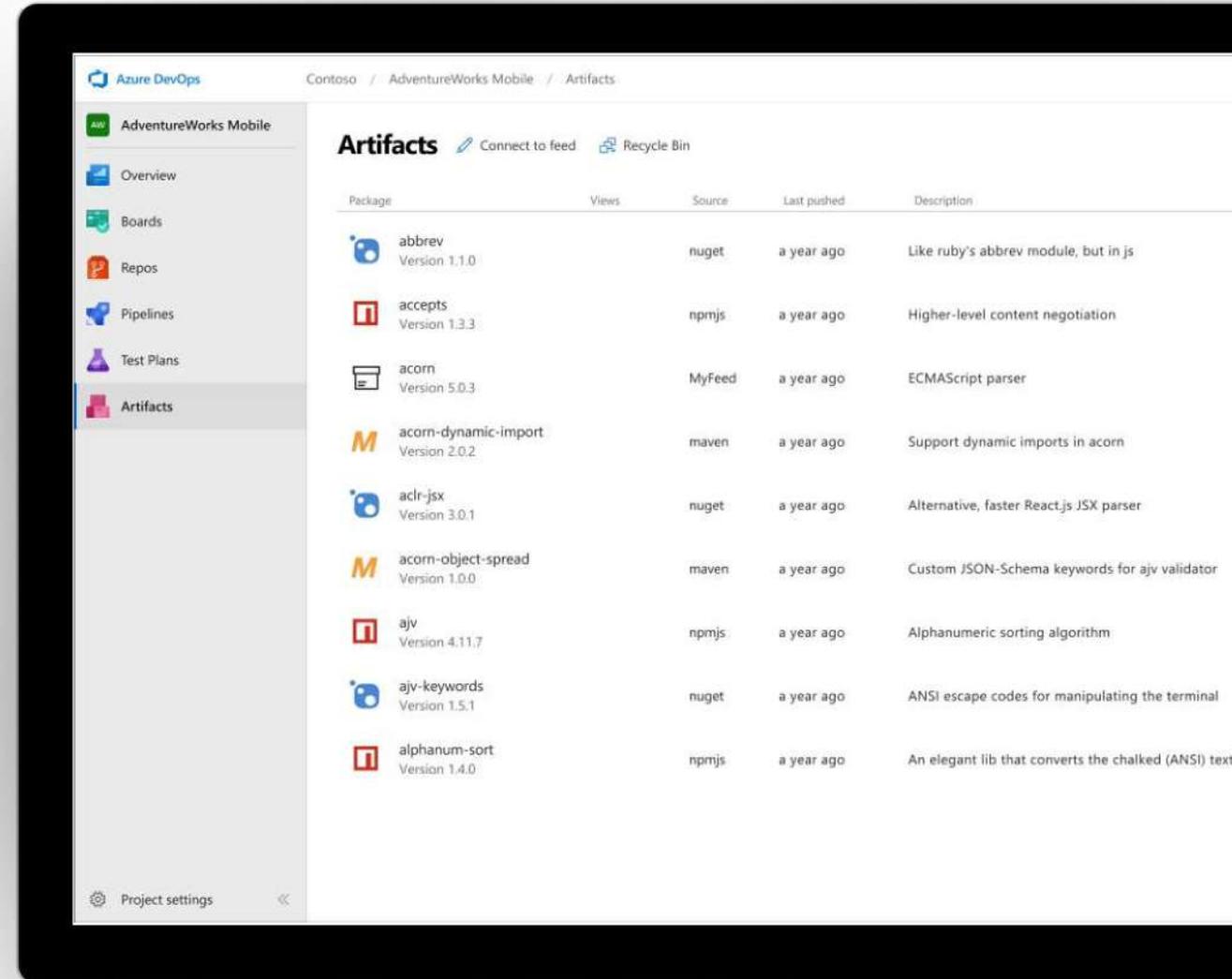
Add packages to any pipeline

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



Share code efficiently

Easily share code across small teams and large enterprises.



Azure Pipelines



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

Implement a wide range of community-built build, test, and deployment tasks. Support for YAML



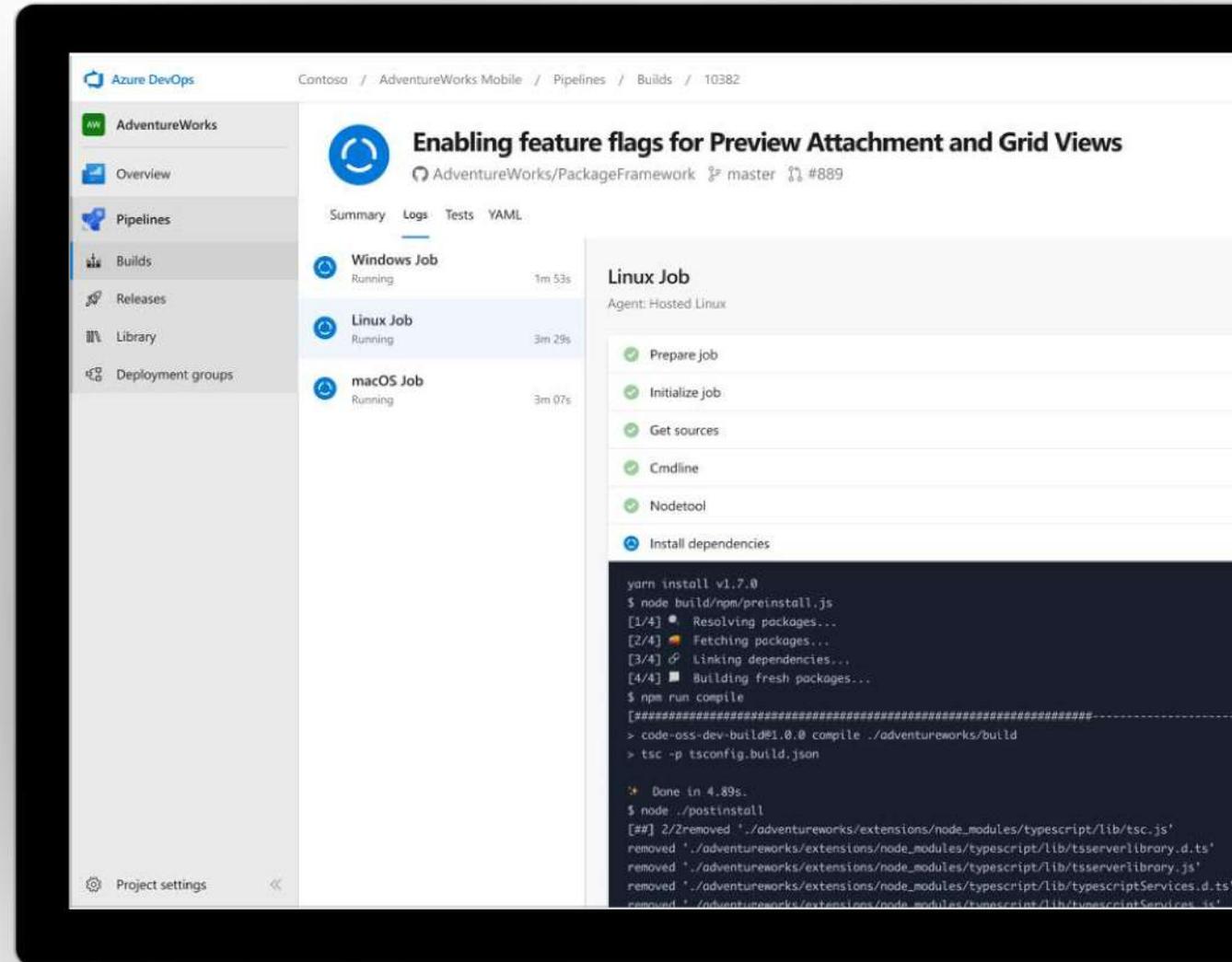
Containers and Kubernetes

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 CI/CD pipelines for every open source project. Get unlimited build minutes for all open source projects with up to 10 free parallel jobs



YAML Pipelines

The pipeline is defined as a set of YAML files

Pipeline can be:

- Versioned 😊
- Reused 100%

Changes are limited to the branch we're using and pipeline will follow the branch flow

Moving from classic templates is fairly easy

```
trigger:
- master

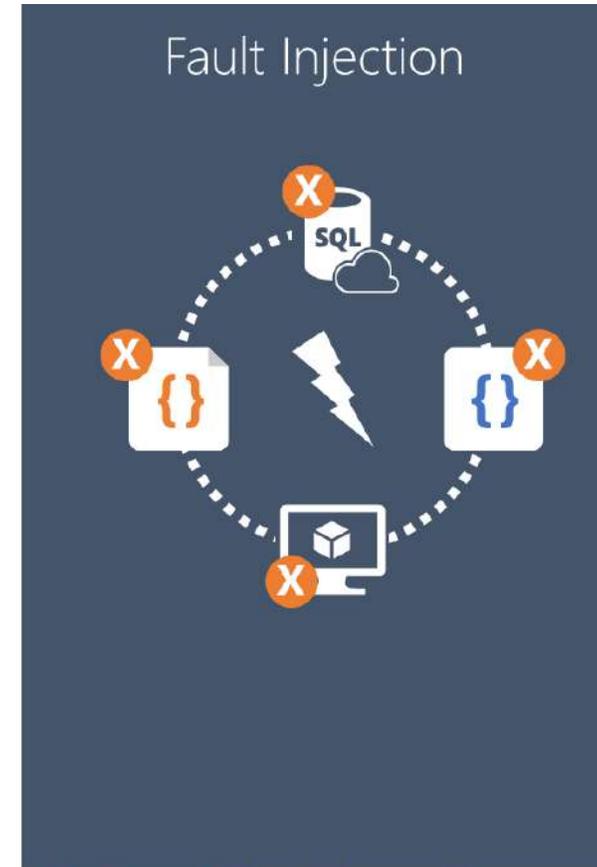
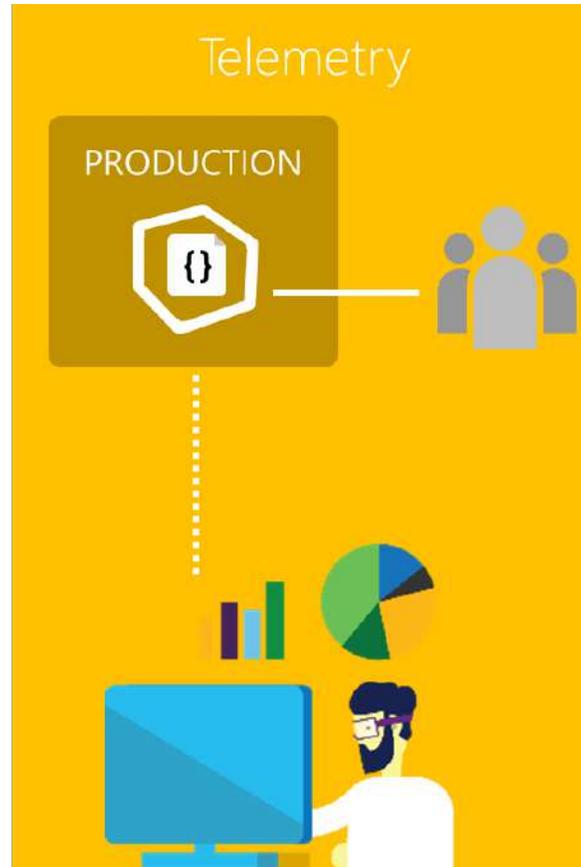
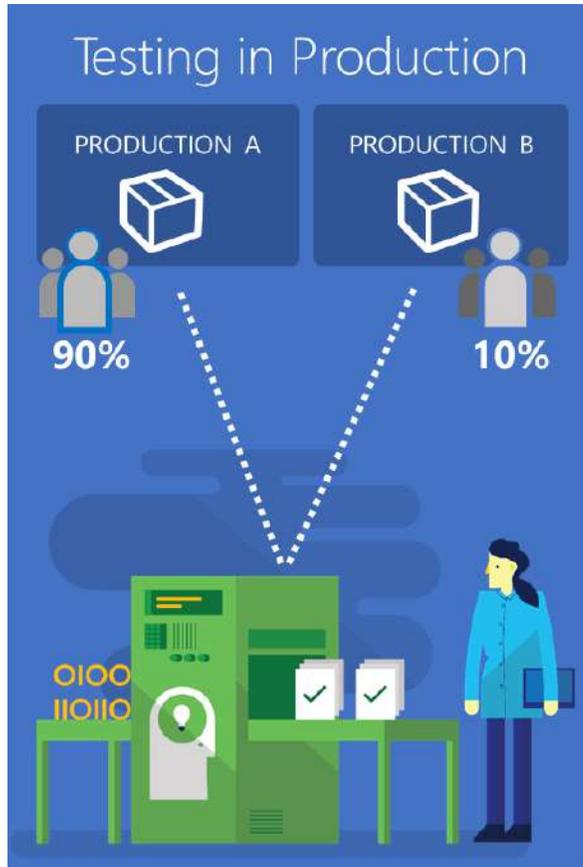
pool:
  vmImage: 'ubuntu-latest'

variables:
  buildConfiguration: 'Release'

steps:
- script: dotnet publish -c $(buildConfiguration)

- task: AzureRmWebAppDeployment@4
  inputs:
    ConnectionType: 'AzureRM'
    appType: 'webApp'
    WebAppName: 'aspitalia'
    packageForLinux: '**\*.zip'
```

Monitoring != telemetry



Testing in production

It's not what you think 😊

It means deployments shouldn't match with features to be released:

- Features in PROD are mostly incomplete when released
- «Minor» breaking changes 😊

Features enabled on-demand

A/B testing, dark launch, canary release

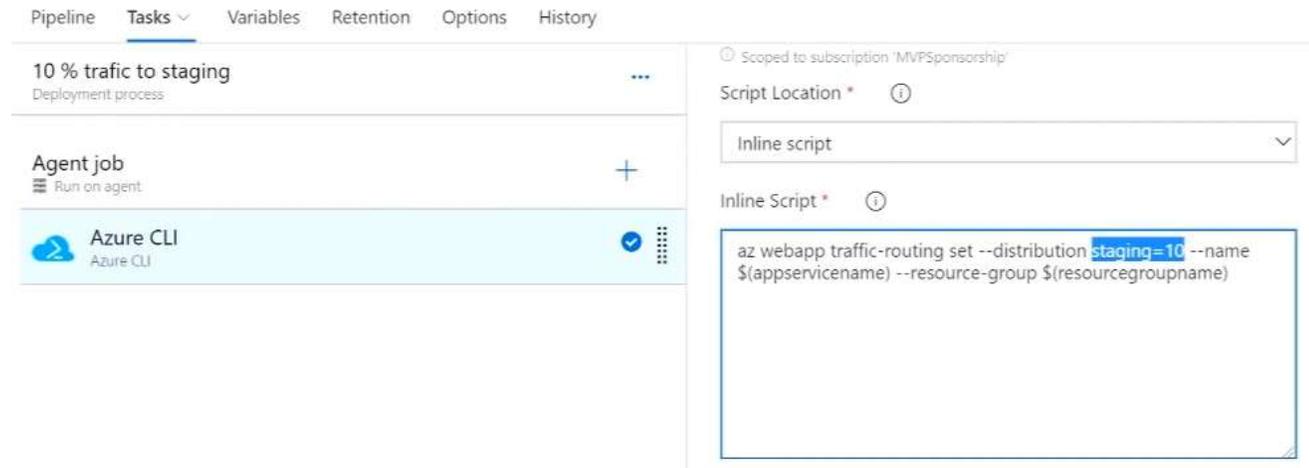
Partial rollout

It's not guaranteed that post-deployment everything will work

- No matter how much testing we did...
- We still have to test new features that have to be enabled!

With App Services you can create a progressive rollout

When something doesn't work the way you expect to, release gates can stop the rollout before reaching PROD



The screenshot shows the configuration for a task in an Azure DevOps pipeline. The task is named "10 % traffic to staging" and is a "Deployment process" type. It is configured to run on an "Agent job" using the "Azure CLI" tool. The "Script Location" is set to "Inline script". The "Inline Script" field contains the following command: `az webapp traffic-routing set --distribution staging=10 --name $(appservice) --resource-group $(resourcegroup)`. The task is scoped to the subscription "MVPSponsorship".

Power BI

Range

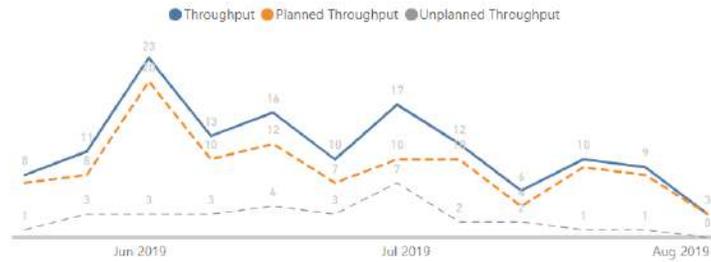
Last 12 Weeks

16/05/2019 - 07/08/2019

[Help & Support](#)

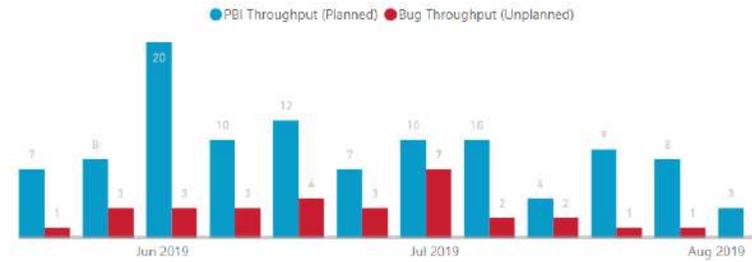
Planned Vs. Unplanned Throughput Rate

Shows the rate at which we are completed planned (PBI) and unplanned (Bug) work.



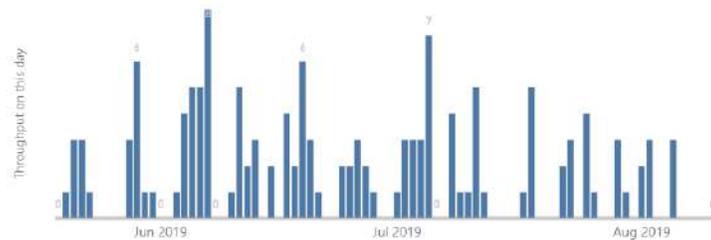
Completed PBIs Vs. Bugs Per Week

Shows the split of completed work (either PBIs or Bugs) on an individual week.



Throughput Per Day

Shows how many items (PBIs & Bugs) were completed on a respective date.



Started Vs. Completed Items Per Week

Compares across each week how many items were started against how many were completed.



Nuovi libri su .NET Core, C# 8, VB 2019

Nuovi libri su

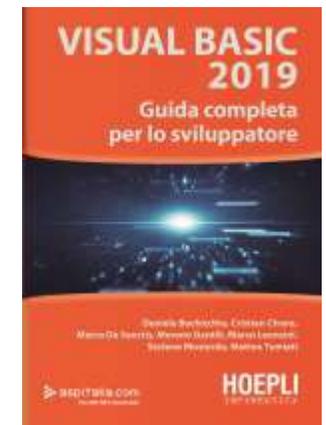
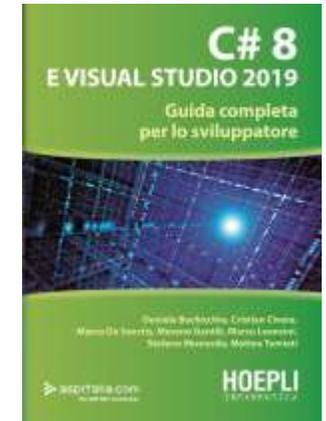
ASP.NET Core 2.1

C# 8 e .NET Core 3

VB 2019 e .NET Core 3

<http://books.aspitalia.com/>

<https://www.dopsitalia.com/articoli/DevOps/intro-azure-devops.aspx>



Grazie!

@xtumiox

matteot@aspitalia.com

Materiale su

<https://aspit.co/netconf-19>

