

# Contents

- Get started
- Start using Azure DevOps
- What is Azure DevOps?
- Overview of services
- Azure DevOps hosted vs. on-premises
- Get started for end users
  - Sign in to the web or a client
  - Code with Git
  - Set up continuous integration and delivery
  - Plan and track work
  - Add and run manual tests
  - View permissions
  - Follow work and pull requests
  - Get started as a Stakeholder
- Get started for administrators
  - Sign up for Azure DevOps
  - Create an organization or project collection
- Manage your project
- Add users to a project or team
- Manage teams and configure team tools
- Change individual permissions
- Grant or restrict permissions to select tasks
- Key concepts
  - Plan your organizational structure
  - Source control
  - Clients and tools
  - Software development roles
- Troubleshooting
  - Troubleshoot connection

[TF31002: Unable to connect](#)

[Troubleshoot access and permissions](#)

[Allowed address lists and network connections](#)

[Configure a network adapter to adjust speed](#)

[Get support or provide feedback](#)

## Reference

[Navigate in Team Explorer](#)

[FAQs](#)

[Service limits](#)

[Features index](#)

[Azure CLI](#)

## Web portal navigation

[Navigation](#)

[Open a service, page, or setting](#)

[Add an artifact or team artifacts](#)

[Use breadcrumbs, selectors, and directories](#)

[Open another project or repo](#)

[Set favorites](#)

[Filter basics](#)

[Search your repo, work items, or wiki](#)

[Manage or enable features](#)

## Search

[Get started with search](#)

[Search code](#)

[Search work items](#)

## Migrate & import

[Migrate data to Azure DevOps Services](#)

[Migrate options](#)

[Import](#)

[Import large collections](#)

[Process templates](#)

[Post-import](#)

[Troubleshooting](#)

[FAQs, migration and process models](#)

[Permissions & access](#)

[Permissions and access \(Security\)](#)

[About access levels](#)

[Status & security](#)

[Service status](#)

[Data protection](#)

[Data location](#)

[Credential storage](#)

[IDE Client Resources](#)

[Visual Studio IDE](#)

[Visual Studio Code](#)

[Visual Studio for Mac](#)

[Eclipse](#)

[IntelliJ IDEA](#)

[Resources](#)

[Settings, security, & usage](#)

[Manage projects](#)

[Marketplace & extensibility](#)

[Search your Wiki](#)

[Code search video](#)

[Journey articles](#)

[Agile](#)

[DevOps](#)

[Git at Scale](#)

# What is Azure DevOps?

3/6/2021 • 3 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Azure DevOps provides developer services for support teams to plan work, collaborate on code development, and build and deploy applications. Azure DevOps supports a culture and set of processes that bring developers and project managers and contributors together to complete software development. It allows organizations to create and improve products at a faster pace than they can with traditional software development approaches.

You can work in the cloud using Azure DevOps Services or on-premises using Azure DevOps Server. For information on the differences between the cloud versus on-premises platforms, see [Azure DevOps Services and Azure DevOps Server](#).

Azure DevOps provides integrated features that you can access through your web browser or IDE client. You can use one or more of the following standalone services based on your business needs:

- **Azure Repos** provides Git repositories or Team Foundation Version Control (TFVC) for source control of your code. For more information about Azure Repos, see [What is Azure Repos?](#).
- **Azure Pipelines** provides build and release services to support continuous integration and delivery of your applications. For more information about Azure Pipelines, see [What is Azure Pipelines?](#).
- **Azure Boards** delivers a suite of Agile tools to support planning and tracking work, code defects, and issues using Kanban and Scrum methods. For more information about Azure Boards, see [What is Azure Boards?](#).
- **Azure Test Plans** provides several tools to test your apps, including manual/exploratory testing and continuous testing. For more information about Azure Test Plans, see [Overview of Azure Test Plans](#)
- **Azure Artifacts** allows teams to share packages such as Maven, npm, NuGet, and more from public and private sources and integrate package sharing into your pipelines. For more information about Azure Artifacts, see [Overview of Azure Artifacts](#).

You can also use the following collaboration tools:

- Customizable team dashboards with configurable widgets to share information, progress, and trends
- Built-in wikis for sharing information
- Configurable notifications

Azure DevOps supports adding extensions and integrating with other popular services, such as: Campfire, Slack, Trello, UserVoice, and more, and developing your own custom extensions.

Azure DevOps Services supports integration with GitHub.com and GitHub Enterprise Server repositories. Azure DevOps Server supports integration with GitHub Enterprise Server repositories. For more information, see the following video, [Using GitHub with Azure DevOps](#).

## Choose Azure DevOps Services

Choose Azure DevOps Services when you want the following outcomes:

- Quick set-up
- Maintenance-free operations
- Easy collaboration across domains
- Elastic scale

- Rock-solid security

To learn more about data protection in Azure DevOps Services, see [Data protection overview](#).

Azure DevOps Services also gives you access to cloud build and deployment servers, and application insights.

We've made it easy for you to start for free and try out our services. Sign up for free by creating an organization. Then, either upload your code to share or source control. Begin tracking your work using Scrum, Kanban, or a combination of methods.

You can use all the services included with Azure DevOps, or choose just what you need to complement your existing workflows.

- [Azure Boards](#). Plan, track, and discuss work across your teams.
- [Azure Pipelines](#). Continuously build, test, and deploy to any platform and cloud.
- [Azure Repos](#). Get unlimited, cloud-hosted private Git repositories for your project.

## Choose Azure DevOps Server

Choose on-premises Azure DevOps Server when:

- You need your data to stay within your network.
- Your work tracking customization requirements are met better with the on-premises XML process model over the inheritance process model. The on-premises model supports modification of XML definition files.

When you deploy Azure DevOps Server, you can also configure the following servers or integration points:

- **Build server** supports on-premises and cloud-hosted builds.
- **SQL Server and SQL Analysis Server** support SQL Server Reports and the ability to create Excel pivot charts based on the cube.

Start for free by downloading [Azure DevOps Server Express](#). Then, either upload your code to share or source control. Or, begin tracking your work using Scrum, Kanban, or a combination of methods.

To learn more about managing Azure DevOps Server, see the [Administrative tasks quick reference](#).

## Next steps

[Sign up for Azure DevOps Services](#) or [Install Azure DevOps Server](#)

## Related articles

- [A tour of services](#)
- [Client-server tools](#)
- [Software development roles](#)
- [Azure DevOps pricing](#)
- [Azure DevOps release notes](#)
- [Azure DevOps blog](#)

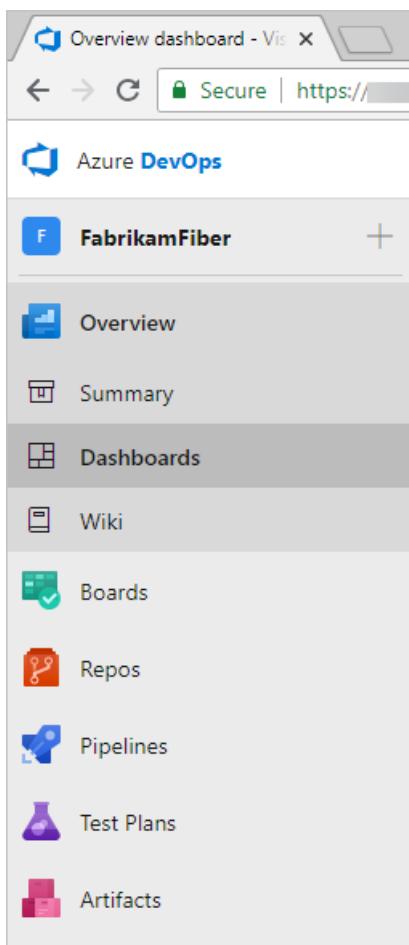
# What features and services do I get with Azure DevOps?

3/6/2021 • 8 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

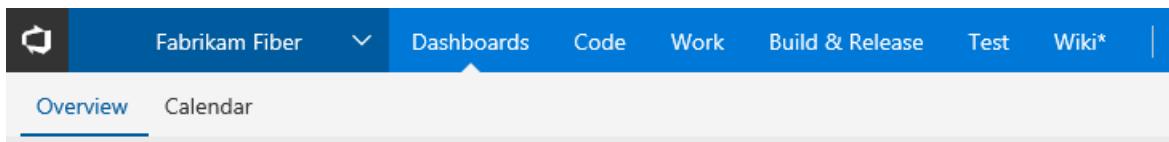
With Azure DevOps, you gain an integrated set of services and tools to manage your software projects, from planning and development through testing and deployment. Services are delivered through a client/server model. Many of them are delivered through an easy-to-use web interface that you can access from all major browsers. Some services, such as source control, build pipelines, and work tracking, can also be managed through a client.

You access Azure DevOps services through the left pane, as shown in the following image. To jump to information for each major service, see the associated articles.



- [Dashboards](#)
- [Wiki](#)
- [Boards](#)
- [Repos](#)
- [Pipelines](#)
- [Test Plans](#)
- [Artifacts](#)

You access Azure DevOps services through the top navigational bar, as shown in the following image. To jump to information for each major service, see the associated articles.

The screenshot shows the top navigation bar of the Azure DevOps interface. It features a dark blue header with the 'Fabrikam Fiber' organization name on the left. To the right of the organization name are several tabs: 'Dashboards', 'Code', 'Work', 'Build & Release', 'Test', and 'Wiki\*'. Below the main header, there is a secondary navigation bar with two items: 'Overview' and 'Calendar'. The 'Overview' item is underlined, indicating it is the active page.

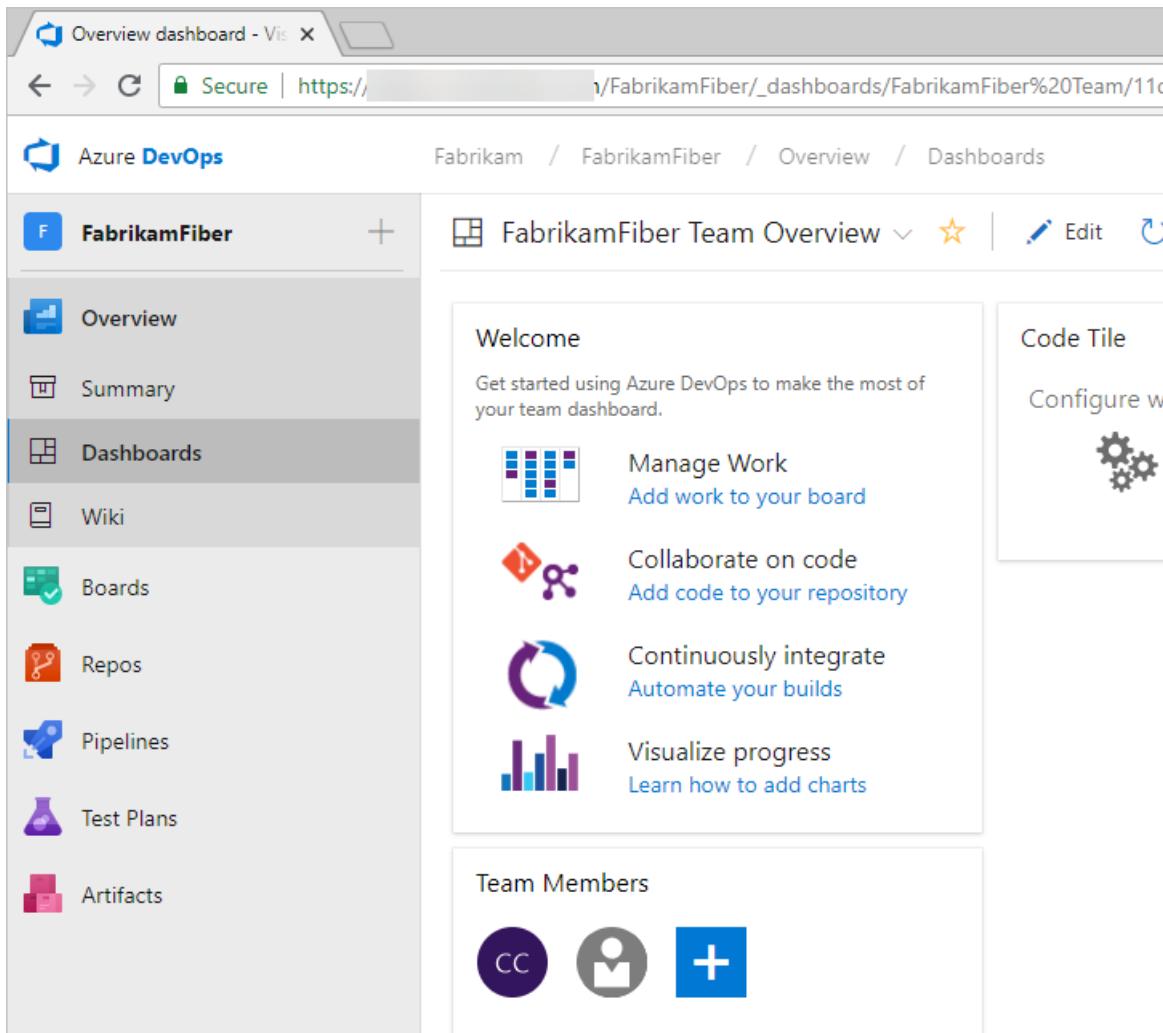
- [Dashboards](#)
- [Code](#)
- [Work](#)
- [Build & Release](#)
- [Test](#)
- [Wiki](#)

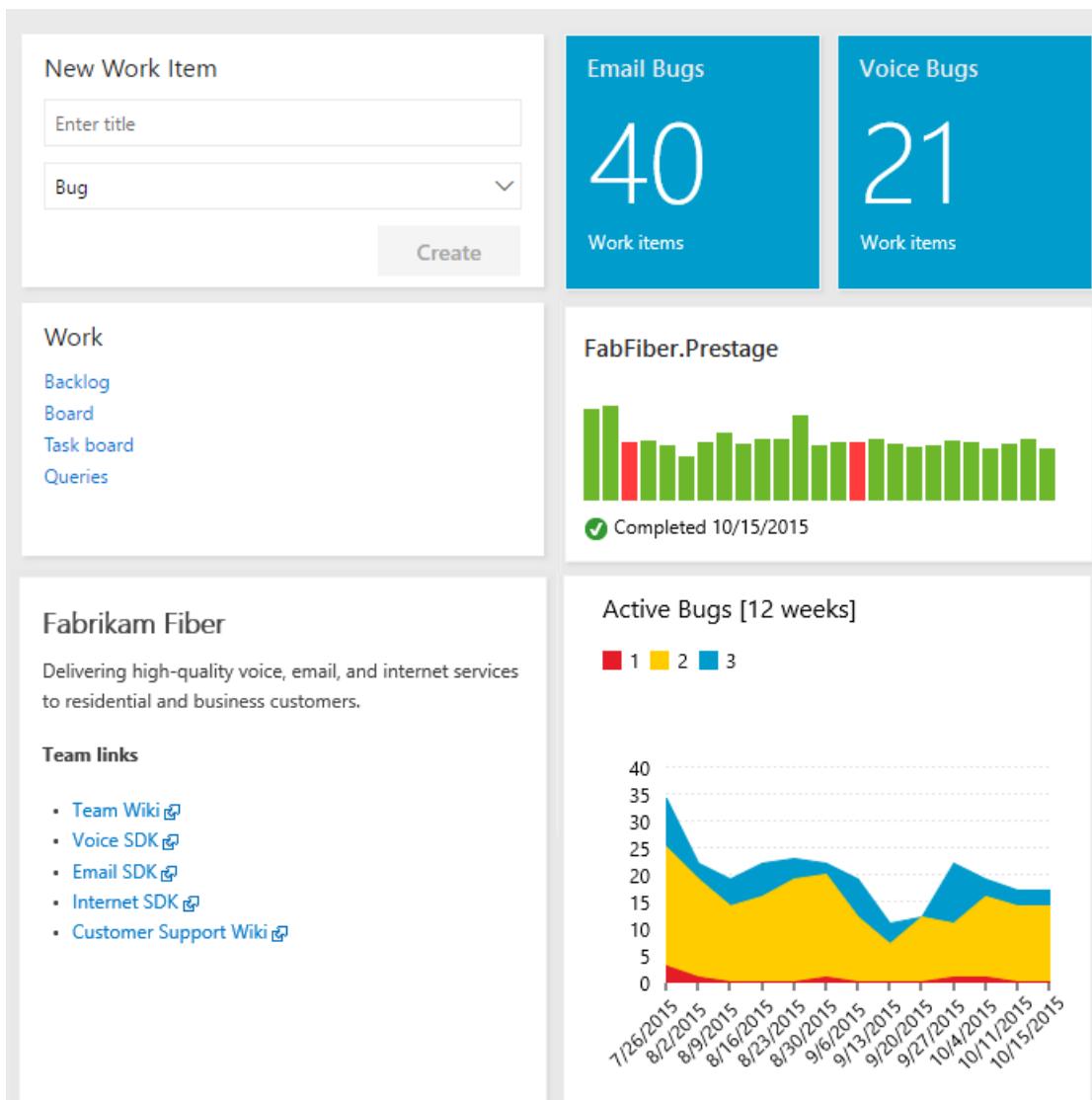
Many of our services are either free for small teams or available through a subscription model or per-use model. You can do a hybrid approach where you use an on-premises deployment to manage your code and work. Then, you purchase cloud build or testing services on an as-needed basis.

For information about client tools, see [Tools](#).

## Dashboards

From [Dashboards](#), you gain access to user-configurable dashboards.

The screenshot displays the 'FabrikamFiber Team Overview' dashboard. At the top, there is a navigation bar with the 'Azure DevOps' logo, the organization name 'Fabrikam', and the team name 'FabrikamFiber'. Below the navigation bar, the dashboard has a sidebar on the left containing links for 'Overview', 'Summary', 'Dashboards' (which is highlighted), 'Wiki', 'Boards', 'Repos', 'Pipelines', 'Test Plans', and 'Artifacts'. The main content area is titled 'FabrikamFiber Team Overview'. It includes a 'Welcome' section with a brief introduction and four call-to-action tiles: 'Manage Work', 'Collaborate on code', 'Continuously integrate', and 'Visualize progress'. At the bottom of the dashboard, there is a 'Team Members' section with icons for managing team members.



You can do the following tasks in **Dashboards**:

- Add, configure, and manage dashboards
- Configure widgets that you add to dashboards
- Quickly go to different areas of your project

To learn more, see [Dashboards](#).

## Source control

Source or version control systems allow developers to collaborate on code and track changes made to the code base. Source control is an essential tool for multi-developer projects.

Our systems support two types of source control: Git (distributed) or Team Foundation Version Control (TFVC), a centralized, client-server system. Both systems enable you to check in files and organize files within folders, branches, and repositories.

With Git, each developer has a copy on their dev machine of the source repository, including all branch and history information. Each developer works directly with their own local repository and changes are shared between repositories as a separate step.

Developers commit each set of changes and do version control operations like history and compare without a network connection. Branches are lightweight. When developers need to switch contexts, they create a private local branch and can switch from one branch to another to pivot among different variations of the codebase. Later, they merge, publish, or dispose of the branch.

#### NOTE

Git in Azure DevOps is standard Git. You can use Visual Studio with third-party Git services. You can also use third-party Git clients with Azure DevOps Server.

With TFVC, developers have only one version of each file on their dev machines. Historical data is maintained only on the server. Branches are path-based and created on the server.

From **Repos**, you gain access to your source control Git-based or Team Foundation Version Control (TFVC) repositories to support version control of your software projects. These repositories are private.

The screenshot shows the Azure DevOps interface for a repository named 'DotNetSample'. The left sidebar includes links for Overview, Boards, Repos (which is selected), Files, Commits, Pushes, Branches, Tags, Pull requests, Pipelines, Test Plans, and Artifacts. The main area displays the contents of the 'master' branch of the 'DotNetSample' repository. The contents include a 'docs' folder, a 'dotnetcore-sample' folder, a 'dotnetcore-tests' folder, a '.gitignore' file, a '.vsts-ci.acr.yml' file, a '.vsts-ci.docker.yml' file, a '.vsts-ci.yml' file, a 'Dockerfile', a 'dotnetcore-sample.sln' file, a 'LICENSE' file, a 'LICENSE-CODE' file, and a 'README.md' file. The 'Contents' tab is selected, and there is a 'History' tab available. The URL in the browser bar is https:///\_git/DotNetSample.

From **Code**, you gain access to your source control Git-based or TFVC repositories to support version control of your software projects. These repositories are private.

	Contents	History	README	+ New file	Upload file(s)	
	↑ Name		Last change		Commits	
M+ page-1.md	M+ page-1.md	10/15/2015	3458a6c7	Added file page-1.md		
M+ page-2.md	M+ page-2.md	10/15/2015	01a447ca	Added file page-2.md		
M+ page-3.md	M+ page-3.md	9/21/2016	68385e28	Added file page-3.md		
M+ README.md	M+ README.md	5/19/2017	fb9177d8	Merged PR 2: Updated		

From Azure Repos for Git, you can do the following tasks:

- Review, download, and edit files, and review the change history for a file
- Review and manage commits that have been pushed
- Review, create, approve, comment on, and complete pull requests
- Add and manage Git tags

To learn more, see the overviews for [Git](#) or [TFVC](#).

## Plan and track work

Software development projects require ways to easily share information and track the status of work, tasks, issues, or code defects. In the past, perhaps you used one or more tools. Microsoft Excel, Microsoft Project, a bug tracking system, or a combination of tools, for example. Now, many teams have adopted Agile methods and practices to support planning and development.

Our systems provide several types of work items that you use to track features, requirements, user stories, tasks, bugs, and issues. Each work item is associated with a work item type and a set of fields that can be updated, as progress is made.

For planning purposes, you have access to several types of backlogs and boards to support the main Agile methods—Scrum, Kanban, or Scrumban.

- Product backlog: Used to create and rank stories or requirements.
- Kanban: Used to visualize and manage the flow of work as it moves from beginning, to in-progress, to done.
- Sprint backlogs: Used to plan work to complete during a sprint cycle, a regular two to four-week cadence that teams use when implementing Scrum.
- Task board: Used during daily Scrum meetings to review work that's completed, remaining, or blocked.

Project managers and developers share information by tracking work items on the backlogs and boards. Useful charts and dashboards complete the picture and help teams monitor progress and trends.

From [Boards](#), you gain access to Agile tools to support planning and tracking work.

The screenshot shows the Azure DevOps Boards interface for the 'FabrikamFiber' team. The left sidebar has 'Boards' selected. The main area displays the 'FabrikamFiber Team' backlog. It includes a 'New' section with a 'New item' button and a search icon, and an 'Active' section showing five items:

- Technician can report busy/late on Windows Phone (3)
- Technician can see service tickets on Windows Phone (0/2, 1)
- Add an information form
- Welcome back page
- Secure sign in (Unassigned, 1)

From **Work**, you gain access to Agile tools to support planning and tracking work.

The screenshot shows the Azure DevOps Work interface. The top navigation bar has 'Work' selected. The left sidebar shows 'Backlogs' selected, with 'Stories' expanded, showing past and current sprints. The main area displays the 'Stories' backlog. It includes a 'Backlog' tab, a chart showing story points, and a table of stories:

	Order	State	Story Points	Title
	1	New	5	Add an information form
	2	New	3	Welcome back page
+	3	New	8	Interim save on long forms
	4	Active	5	> Secure Sign-in
	5	Active	5	Canadian addresses don't display

Specifically, you can do the following tasks:

- Add and update work items
- Define work item queries, and create status and trend charts based on those queries
- Manage your product backlog

- Plan sprints by using sprint backlogs
- Review sprint tasks and update tasks through the task boards
- Visualize the workflow and update the status by using Kanban boards
- Manage portfolios by grouping stories under features and grouping features under epics

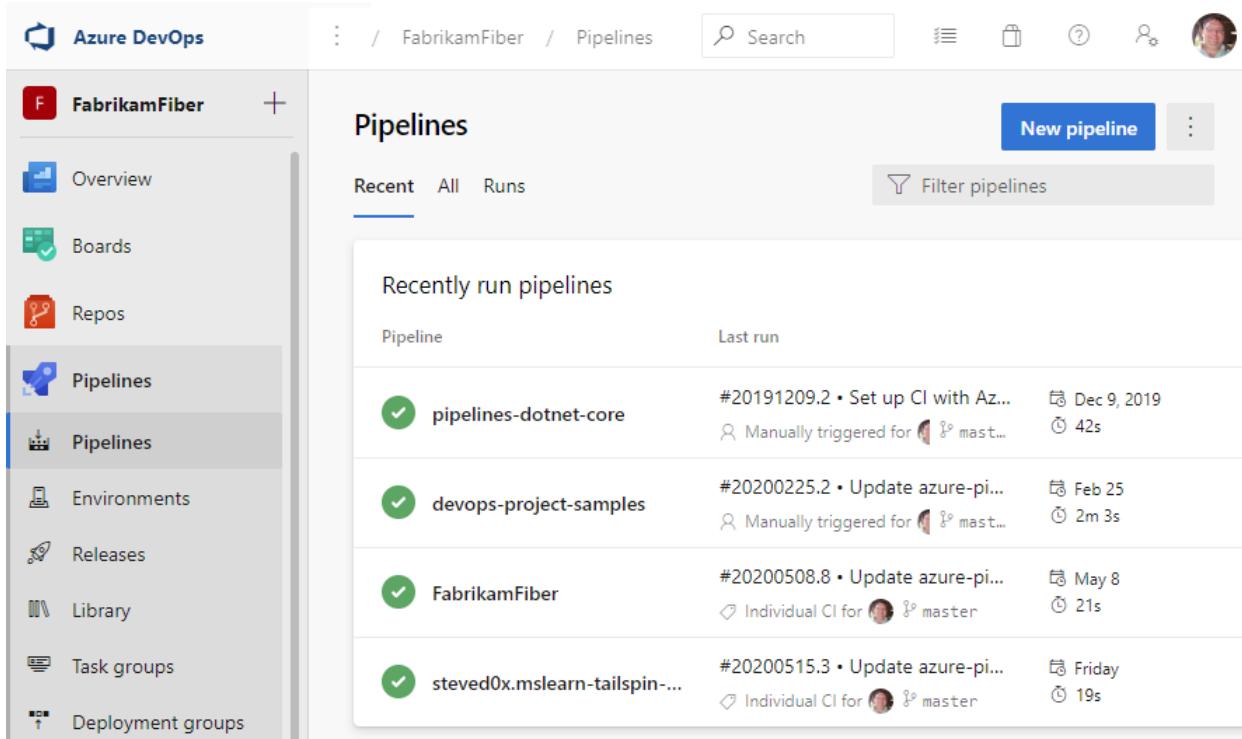
See [Backlogs, boards, and plans](#) for an overview of each.

## Continuous integration and deployment

The rapid and reliable release of software comes from automating as many processes as possible. Our systems support build, test, and release automation.

- You can define builds to automatically run whenever a team member checks in code changes.
- Your build pipelines can include instructions to run tests after the build runs.
- Release pipelines support managing deployment of your software builds to staging or production environments.

**Azure Pipelines** provides an integrated set of features to support building and deploying your applications.



The screenshot shows the Azure DevOps Pipelines interface for the 'FabrikamFiber' project. The left sidebar has 'Pipelines' selected. The main area displays a table of recently run pipelines:

Pipeline	Last run
pipelines-dotnet-core	#20191209.2 • Set up CI with Az... Manually triggered for master Dec 9, 2019 42s
devops-project-samples	#20200225.2 • Update azure-pi... Manually triggered for master Feb 25 2m 3s
FabrikamFiber	#20200508.8 • Update azure-pi... Individual CI for master May 8 21s
steved0x.mslearn-tailspin-...	#20200515.3 • Update azure-pi... Individual CI for master Friday 19s

**Azure Pipelines** provides an integrated set of features to support building and deploying your applications.

Build Definition	Status	Triggered by	7-day pass rate
Content.VS Build : #Content.VS Build_20160609.1 requested a year ago	passing	Updated the overview s... 80496e4 in 89 users/...	0% →
Content.VS.PR : #Content.VS.PR_20161019.14 requested 10 months ago	passing	Merge pull request 152... 2be71b1 in 152638	0% →
MSDN.GatedCheck.ALM-master : #20170313.2 requested 5 months ago	passing	Merge pull request 194... 8f7955d in 194899	0% →
MSDN.GatedCheck.VS-master : #20160725.1 requested a year ago	passing	Merge pull request 126... 2d56c79 in 126293	0% →

Use pipelines to implement continuous integration and continuous delivery.

- **Build automation:** Define the steps to take during build and the triggers that start a build.
- **Release management:** Supports a rapid release cadence and management of simultaneous releases. You can configure release pipelines that represent your environments from development to production. Run automation to deploy your app to each environment. Add approvers to confirm that the app has been successfully deployed in an environment. Create your release manually or automatically from a build. Then track your releases as they're deployed to various environments.

To learn more, see [Continuous integration on any platform](#).

## Manual and exploratory testing

Test features support manual and exploratory testing, and continuous testing.

**Test Plans** supports creating and managing manual tests.

Outcome	Order	ID ↑	Title
Active	1	368	Fabrikam Test
Active	2	369	Test sign in flow

**Test** supports creating and managing manual tests.

The screenshot shows the Microsoft DevOps Test Plans interface. At the top, there's a navigation bar with links for Dashboards, Code, Work, Build & Release, Test, Wiki\*, and a gear icon. Below the navigation bar, there's a sub-navigation menu with links for Test Plans, Parameters, Configurations, Runs, Machines, and Load test. The main area is titled "Test suite: 379 : Phone sign in (Suite ID: 477)". On the left, there's a sidebar with icons for creating new items and a dropdown menu for "Fabrikam Fiber: Fabrikam Fiber Team\_Sto...". The main content area displays a table of test cases. The columns are labeled: Tests, Charts, Outcome All, Tester All, Configuration All. There are buttons for "+ New", "Add existing", "Run", and other actions. The table has rows for two test cases, both labeled "Active" with IDs 474 and 478. The "Configuration" column shows "Windows 8" for both rows.

With test features, you gain access to the following features:

- Customization of workflows with test plan, test suite, and test case work items
- End-to-end traceability from requirements to test cases and bugs with requirement-based test suites
- Criteria-based test selection with query-based test suites
- Excel-like interface with the grid for easy creation of test cases
- Reusable test steps and test data with shared steps and shared parameters
- Sharable test plans, test suites, and test cases for reviewing with Stakeholders
- Browser-based test execution on any platform
- Real-time charts for tracking test activity

To learn more, see [Testing overview](#).

## Collaboration services

The following services work across the previously mentioned services to support:

- Team dashboards
- Project wiki
- Discussion within work item forms
- Linking of work items, commits, pull requests, and other artifacts to support traceability
- Alerts and change notifications managed per user, team, project, or organization
- Ability to request and manage feedback
- Analytics service, analytic views, and Power BI reporting
- Dashboards
- Project wiki
- Discussion within work item forms
- Linking of work items, commits, pull requests, and other artifacts to support traceability
- Alerts and change notifications managed per user, team, project, or project collection
- Ability to request and manage feedback
- SQL Server Reporting
- Dashboards
- Discussion within work item forms
- Linking of work items, commits, pull requests and other artifacts to support traceability
- Alerts and change notifications managed per user, team, project, or project collection
- Ability to request and manage feedback
- Team (chat) rooms

- SQL Server Reporting

**NOTE**

Team rooms are deprecated for TFS 2017.2. Instead, we recommend that you [use service hooks to integrate with Microsoft Teams](#).

- Dashboards
  - Linking of work items, commits, pull requests, and other artifacts to support traceability
  - Alerts and change notifications managed per user or for teams
  - Ability to request and manage feedback
  - Team (chat) rooms
  - SQL Server Reporting
- 
- Team home page
  - Linking of work items, commits, pull requests, and other artifacts to support traceability
  - Alerts and change notifications managed per user or for teams
  - Ability to request and manage feedback
  - Team (chat) rooms
  - SQL Server Reporting

## Service hooks

Service hooks enable you to complete tasks on other services when events happen within your project hosted on Azure DevOps. For example, you can send a push notification to your team's mobile devices when a build fails. You can also use service hooks in custom apps and services as a more efficient way to drive activities in your projects.

The following services are available as the target of service hooks. To learn about other apps and services that integrate with Azure DevOps, visit the [Visual Studio Marketplace](#), Azure DevOps tab.

For the latest set of supported services, see [Integrate with service hooks](#).

## Cloud-hosted services based on usage

The following services support your DevOps operations:

- Cloud-based, Microsoft-hosted build and deployment agents
- On-premises self-hosted agents to support build and deployment

To learn more, see [Pricing](#).

## Azure cloud-hosted services

Azure provides cloud-hosted services to support application development and deployment. You can make use of these services solely or in combination with Azure DevOps.

To browse the directory of integrated services, features, and bundled suites, see [Azure products](#).

For continuous delivery to Azure from Azure DevOps Services, see [Automatically build and deploy to Azure web apps or cloud services](#).

## Administrative services

There are features and tasks associated with administering a collaborative software development environment. You complete most of these tasks through the web portal. To learn more, see [About user, team, project, and organization-level settings](#).

The screenshot shows the 'Project Settings > Overview' page for the 'FabrikamFiberTest' project. The left sidebar lists project management areas: Overview, Boards, Repos, Pipelines, Test Plans, and Artifacts. A red box highlights the 'Project settings' link at the bottom of the sidebar. The main content area is titled 'Project details' and includes fields for Name (FabrikamFiberTest), Description (Fabrikam Fiber test project), Process (Scrum), and Visibility (Private). Below this is a 'Save' button. The 'Azure DevOps services' section lists 'Boards', 'Repos', 'Pipelines', and 'Packages'. The 'Boards' item has a green checkmark icon.

Project Settings > Overview

**Project details**

Name  
FabrikamFiberTest

Description  
Fabrikam Fiber test project

Process  
Scrum

Visibility  
Private

This determines who can view this project. [Learn more](#)

Save

**Azure DevOps services**

- Boards** Flexible agile planning with cards and checkmarks
- Repos** Repos, pull requests, and code reviews
- Pipelines** Build, manage, and scale CI/CD pipelines
- Packages**

Project settings

S Fabrikam Fiber Dashboards Code Work Build & Release Test Wiki\* | 

Overview Work Security Version Control Policies Agent queues Notifications Service Hooks

Project profile

Teams

New team | 

Team Name ↑	Members	Description
 Customer Service	7	
 <b>Fabrikam Fiber Team</b>	7	The default project team.
 Management team	1	
 Phone	1	
 Voice	1	
 Web	2	

Name  
**Fabrikam Fiber**

Process  
Scrum

Description  
Web, voice, and phone apps

## Related articles

- [Understand differences between Azure DevOps Services and Azure DevOps Server](#)
- [Client-server tools](#)
- [Software development roles](#)
- [Azure DevOps pricing](#)
- [Azure DevOps data protection overview](#)

# Compare Azure DevOps Services with Azure DevOps Server

3/6/2021 • 9 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

The **cloud offering**, Azure DevOps Services, provides a scalable, reliable, and globally available hosted service. It's backed by a 99.9% SLA, monitored by our 24/7 operations team, and available in local data centers around the world.

The **on-premises offering**, Azure DevOps Server, is built on a SQL Server back end. Customers usually choose the on-premises version when they need their data to stay within their network. Or, when they want access to SQL Server reporting services that integrate with Azure DevOps Server data and tools.

Although both offerings provide the same [essential services](#), compared with Azure DevOps Server, Azure DevOps Services offers the following added benefits:

- Simplified server management.
- Immediate access to the latest and greatest features
- Improved connectivity with remote sites.
- A transition from capital expenditures (servers and the like) to operational expenditures (subscriptions).

To determine which offering—cloud or on-premises—meets your needs, consider the following key differences.

## Fundamental differences between Azure DevOps Services and Azure DevOps Server

When you're choosing which platform you want, or if you're considering a move from on-premises to the cloud, consider the following areas:

- [Scope and scale data](#)
- [Authentication](#)
- [Users and groups](#)
- [Manage user access](#)
- [Security and data protection](#)

### Differences in specific feature areas

Although Azure DevOps Services is a hosted version of Azure DevOps Server, there are some differences between features. Some Azure DevOps Server features aren't supported in Azure DevOps Services. For example, Azure DevOps Services doesn't support integration with SQL Server Analysis Services to support reporting.

Two of the following other areas differ in their support:

- [Process customization](#)
- [Reporting](#)

Are you on Azure DevOps Server and considering moving? Read [Migration options](#) to understand your options.

## Scope and scale data

As your business grows, you may need to scale up your Azure DevOps instance.

### Azure DevOps Services scales by using organizations and projects

Azure DevOps Services differs slightly from Azure DevOps Server. There are currently only two options for scoping and scaling data: organizations and projects. Organizations in Azure DevOps Services get their own URLs (for example, <https://dev.azure.com/fabrikamfiber>), and they always have exactly one project collection. Organizations can have many projects within a collection.

We recommend that you create organizations in Azure DevOps Services wherever you would create collections in Azure DevOps Server. The following scenarios apply:

- You can purchase Azure DevOps Services users per organization - Paid users can access only the organization in which the payment is made. If you have users who need access to many organizations, Visual Studio subscriptions can be an attractive option. Visual Studio subscribers can be added to any number of organizations at no charge. We're also considering other ways to make access available to many organizations that are grouped into a single organization.
- You currently have to administer organizations one at a time. This process can be cumbersome when you have many organizations.

Learn more: [Plan your organizational structure in Azure DevOps](#).

### Azure DevOps Server scales by using deployments, project collections, and projects

Azure DevOps Server offers the following three options for scoping and scaling data: deployments, project collections, and projects. In the simplest case, deployments are just servers.

Deployments can be more complicated, however, which could include:

- Two-server deployment where SQL is split out on a separate machine
- High-availability farms with lots of servers

Project collections serve as containers for security and administration, and physical database boundaries. They're also used to group related projects.

Finally, projects are used to encapsulate the assets of individual software projects, including source code, work items, and so on.

Learn more: [Plan your organizational structure in Azure DevOps](#).

## Authentication

With Azure DevOps Services, you connect over the public internet (for example, <https://contoso.visualstudio.com>). You either authenticate with [Microsoft account](#) credentials or with [Azure AD](#) credentials, depending on your organization setup. You can also set up Azure AD to require features such as multi-factor-authentication, IP address restrictions, and so on.

We recommend that you configure your organizations to use Azure AD rather than Microsoft accounts. This method provides a better experience in many scenarios and more options for enhanced security.

Learn more: [About accessing Azure DevOps Services with Azure AD](#).

With Azure DevOps Server, you connect to an intranet server (for example, <https://tfs.corp.contoso.com:8080/tfs>). You authenticate with Windows Authentication and your Active Directory (AD) domain credentials. This process is transparent and you never see any kind of sign-in experience.

## Manage users and groups

In Azure DevOps Services, you can use a similar mechanism to [provide access to groups of users](#). You can add Azure AD groups to Azure DevOps Services groups. If you use Microsoft Accounts instead of Azure AD, you have to [add users](#) one at a time.

In Azure DevOps Server, you provide users access to deployments by adding Active Directory (AD) groups to various Azure DevOps groups (for example, the Contributors group for an individual project). The AD group memberships are kept in sync. As users are added and removed in AD, they also gain and lose access to Azure DevOps Server.

## Manage user access

In both Azure DevOps Services and Azure DevOps Server, you manage access to features by assigning users to an [access level](#). All users must be assigned to a single access level. In both the cloud and on-premises offerings, you can give free access to work item features to an unlimited number of Stakeholders. Also, an unlimited number of Visual Studio subscribers can have access to all Basic features at no additional charge. You pay only for other users who need access.

In Azure DevOps Services, you must [assign an access level](#) to each user in your organization. Azure DevOps Services validates Visual Studio subscribers as they sign in. You can assign Basic access for free to five users without Visual Studio subscriptions.

To give Basic access or higher to more users, [set up billing](#) for your organization and [pay for more users](#). Otherwise, all other users get Stakeholder access.

Azure AD groups give access to groups of users. Access levels are automatically assigned at first sign-in. For organizations that are configured to use Microsoft accounts for signing in, you must assign access levels to each user explicitly.

In Azure DevOps Server, all use is on the honor system. To set access levels for users based on their licenses, specify their [access levels](#) on the administration page. For example, assign unlicensed users Stakeholder access only.

Users with an Azure DevOps Server Client Access License (CAL) can have Basic access. Visual Studio subscribers can have either Basic or Advanced access, depending on their subscriptions. Azure DevOps Server doesn't attempt to verify these licenses or enforce compliance.

## Security and data protection

Many entities want to know more about data protection when they consider moving to the cloud. We're committed to ensuring that Azure DevOps Services projects stay safe and secure. We have technical features and business processes in place to deliver on this commitment. You can also take steps to secure your data. Learn more in our [Data Protection overview](#).

## Process customization

You can customize the work-tracking experience in two different ways, depending on the supported process model:

- Azure DevOps Services: you use the **Inheritance** process model, which supports WYSIWYG customization
- Azure DevOps Server: you can choose the **Inheritance** process model or the **On-premises XML** process model, which supports customization through import or export of XML definition files for work-tracking objects
- Azure DevOps Server 2018 and earlier versions: you only have access to the **On-premises XML** process model

Although the **On-premises XML** process model option is powerful, it can cause various issues. The main issue is that processes for existing projects aren't automatically updated.

Azure DevOps Server 2013, for example, introduced several new features that depended on new work-item types and other process template changes. When you upgrade from 2012 to 2013, each project collection gets new versions of each of the "in the box" process templates that include these changes. However, these changes aren't automatically incorporated into existing projects. Instead, after you finish upgrading, you have to include the changes in each project by using the [Configure features](#) wizard or a more manual process.

To help you avoid these issues in Azure DevOps Services, custom process templates and the **witadmin.exe** tool have always been disabled. This approach has enabled us to automatically update all projects with each Azure DevOps Services upgrade. Meanwhile, the product team is working hard to make customizing processes possible in ways that we can support easily and continuously. We recently introduced the first of these changes and more changes are on the way.

With the new process-customization capability, you can make changes directly within the web user interface (UI). If you want to customize your processes programmatically, you can do so through REST endpoints. When you customize projects this way, they're automatically updated when we release new versions of their base processes with Azure DevOps Services upgrades.

To learn more, see [Customize your work-tracking experience](#).

## Reporting

Azure DevOps Services and Azure DevOps Server offer a many tools that give you insight into the progress and quality of your software projects. Included are the following tools:

- **Dashboards** and lightweight **charts** that are available in both the cloud and on-premises platforms. These tools are easy to set up and use.

Azure DevOps Services and Azure DevOps Server 2019 also provide access to the following services:

- **The Analytics service** and **Analytics widgets**. The Analytics service is optimized for fast read-access and server-based aggregations.
- **Microsoft Power BI integration**, which supports getting Analytics data into Power BI reports and provides a combination of simplicity and power.
- **OData support**, which allows you to directly query the Analytics service from a supported browser, and then use the returned JSON data as you want. You can generate queries that span many projects or your entire organization.

To learn more about the Analytics service and future releases, see our [Reporting roadmap](#).

[SQL Server Reporting Services \(SSRS\) reports](#) are available from Azure DevOps Server when configured with SQL Server Analysis Services.

## Visual Studio Team Services is now Azure DevOps Services

Many of the featured services in VSTS are now offered as standalone services in both Azure DevOps Services and Azure DevOps Server 2019. You can get services separately or all together as Azure DevOps Services. If you're an Azure DevOps subscriber, you have access to all of the services already.

VSTS FEATURE NAME	AZURE DEVOPS SERVICE NAME	DESCRIPTION
-------------------	---------------------------	-------------

VSTS FEATURE NAME	AZURE DEVOPS SERVICE NAME	DESCRIPTION
Build & release	Azure Pipelines	Continuous integration and continuous delivery (CI/CD) that works with any language, platform, and cloud.
Code	Azure Repos	Unlimited cloud-hosted private Git and Team Foundation Version Control (TFVC) repositories for your project.
Work	Azure Boards	Work tracking with Kanban boards, backlogs, team dashboards, and custom reporting.
Test	Azure Test Plans	All-in-one planned and exploratory testing solution.
Packages (extension)	Azure Artifacts	Maven, npm, Python, Universal Package, and NuGet package feeds from public and private sources.

Both Azure DevOps Services and Azure DevOps Server 2019 use the new navigation user interface, with a vertical sidebar to go to the main service areas: **Boards**, **Repos**, **Pipelines**, and more. To learn more, see [Web portal navigation in Azure DevOps](#).

#### NOTE

You can disable select services from the user interface. For more information, see [Turn a service on or off](#).

You can still use `visualstudio.com` to access Azure DevOps Services. We've moved to the new `dev.azure.com` domain name as the primary URL for new organizations. That URL is

`https://dev.azure.com/{your organization}/{your project}`. If you want to change your URL to be based on `dev.azure.com` as the primary, an organization administrator can do so from the organization settings page.

## Related articles

- [Essential services](#)
- [Client-server tools](#)
- [Software development roles](#)
- [Azure DevOps Services - pricing](#)
- [Azure DevOps Server - pricing](#)

# Connect to a project in Azure DevOps

3/6/2021 • 7 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Learn how to connect to a project to share code, build apps, track work, and collaborate with team members.

You can use any of the following clients:

- [Web portal](#)
- [Visual Studio or Team Explorer](#)
- [Eclipse/Team Explorer Everywhere](#)
- [Android Studio with the Azure DevOps Services Plugin for Android Studio](#)
- [IntelliJ with the Azure DevOps Services Plugin for IntelliJ](#)
- [Visual Studio Code](#)

A project defines a process and data storage in which you manage your software projects from planning to deployment. When you connect to a project, you connect to an organization or project collection. One or more projects may be defined within a collection. There must be at least one project. For more information, see [About projects and scaling your organization](#).

## Prerequisites

- If you don't have a project yet, [create one](#).
- If you need to add a team, see [Add teams](#). If you don't have access to the project, [get invited to the team](#).
- From each of these clients, you can switch context to a different project and connect as a different user. If you work remotely, configure your client to [connect to an Azure DevOps Proxy Server](#).
- To get started with a code base, [set up Git](#) or [set up TFVC](#).

## Connect from the web portal

1. If you're not a member of a security group, ask your Project Administrator to add you.

2. Open a browser and enter a URL that uses the following form:

```
https://dev.azure.com/OrganizationName/ProjectName
```

```
http://ServerName/DefaultCollection/ProjectName
```

For example, to connect to the server named **FabrikamPrime**, type:

**http://FabrikamPrime/DefaultCollection**.

```
http://ServerName:8080/tfs/DefaultCollection/ProjectName
```

For example, to connect to the server named **FabrikamPrime**, type:

**http://FabrikamPrime:8080/tfs/DefaultCollection**.

The default Port is 8080. If you don't use default values, specify the port number and directory for your server.

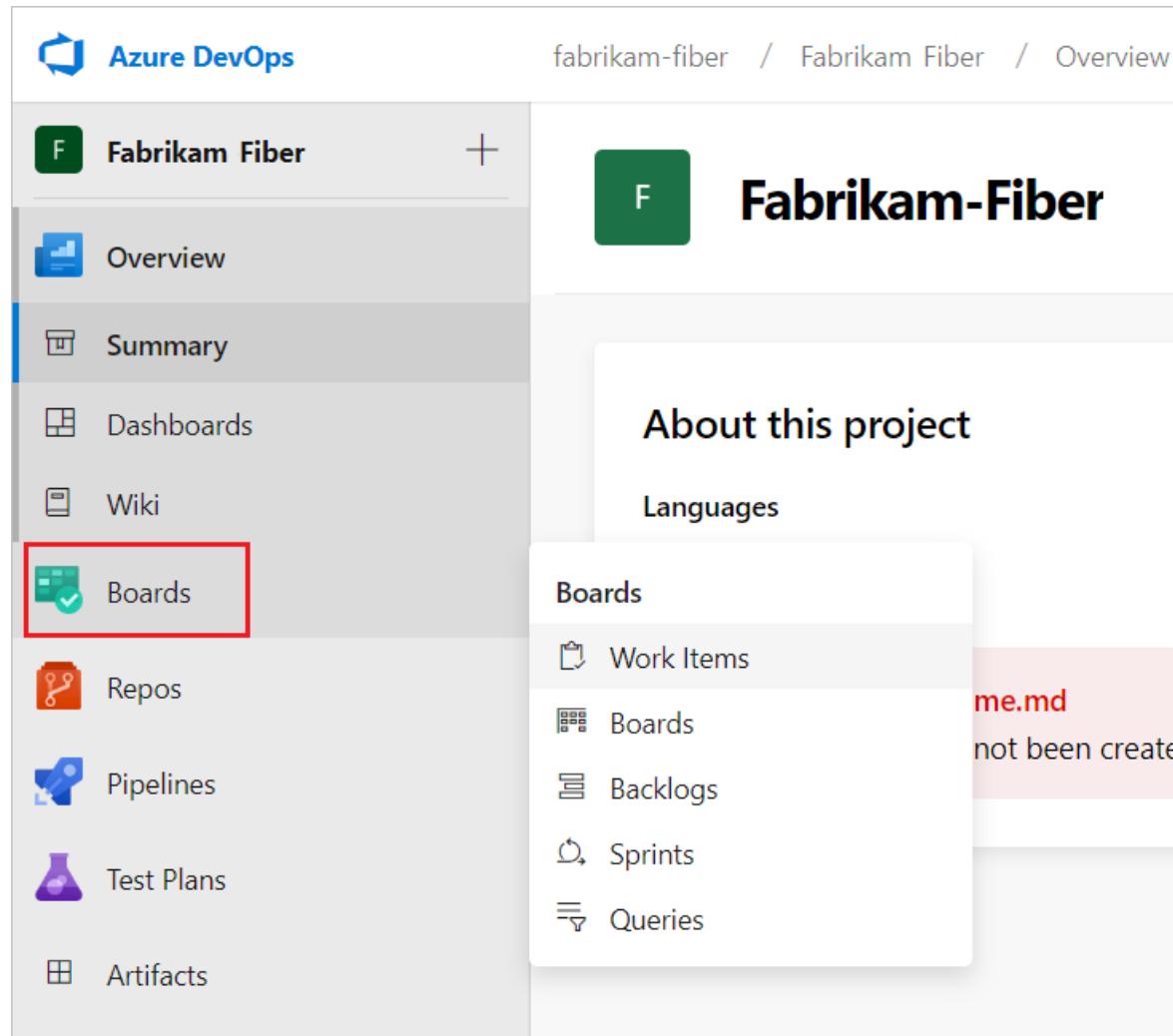
- When you access the server for the first time, a Windows Identity dialog box appears. Enter your credentials and choose OK.

**TIP**

If you select **Remember me**, you won't have to enter your credentials the next time you connect.

- Choose your project, team, or page of interest.

From the project summary page, hover over a service and then choose the page you want. To choose another project, choose **Azure DevOps**. 



The screenshot shows the Azure DevOps interface for the 'Fabrikam Fiber' project. On the left, there's a sidebar with various navigation links: Overview, Summary, Dashboards, Wiki, Boards, Repos, Pipelines, Test Plans, and Artifacts. The 'Boards' link is highlighted with a red box. To its right, the main content area displays the 'Fabrikam-Fiber' project summary. Below the project name, there are sections for 'About this project' and 'Languages'. A dropdown menu is open over the 'Boards' link in the sidebar, listing 'Work Items', 'Boards', 'Backlogs', 'Sprints', and 'Queries'. The 'Boards' option in the dropdown is also highlighted with a red box. The 'Boards' link in the sidebar has a green background, indicating it's the active selection.

From the project summary page, hover over a service and then choose the page you want. To choose another project, choose the  Azure DevOps logo.

The screenshot shows the 'Work' tab selected in the top navigation bar. A dropdown menu is open under 'Work' with options: 'Work Items', 'Backlogs', 'Queries', 'Plans', and 'New work item'. The main content area displays the 'Fabrikam Fiber' project details, including its logo (FF), name, and tagline 'Web, voice, and phone apps'. Below this is a 'New work item' section with a link to 'Fabrikam Fiber / README.md'. The README file content includes a minor modification note, a link to update it, and links to three pages related to Markdown.

Choose your project or team from the set of available links, or choose **Browse** to access all projects and teams.

The screenshot shows the 'Overview' tab selected in the top navigation bar. The main content area features a 'About Team Foundation Server' summary with four purple cards: 'Features' (What does Team Foundation Server have to offer?), 'Learn' (Access online help for Team Foundation Server), 'Get Visual Studio' (View all the download options), and 'Administer' (Manage projects, users, groups and permissions). Below this are sections for 'Recent projects & teams' and 'Recent team rooms'. The 'Recent projects & teams' section lists 'Fabrikam Fiber / Web Service' (2 minutes ago), 'Fabrikam Fiber' (21 hours ago), 'Fabrikam Fiber / Migrate' (5/27/2016), and 'Fabrikam Fiber / Fiber Suite' (2/3/2016). The 'Recent team rooms' section shows 'Fabrikam Fiber Team Room' with '0 users in room'.

To learn more about each page and the tasks you can do, see [Web portal navigation](#).

### Sign in with different credentials

1. Open Windows Security from the context menu associated with your name.

Visual Studio Team Foundation Server 2015 / Fabrikam Fiber

Raisa Pokrovskaya | ms | ?

HOME CODE WORK BUILD TEST RELEASE

Welcome | Overview +

Visual Studio

 Open in Visual Studio  
Requires Visual Studio 2013+

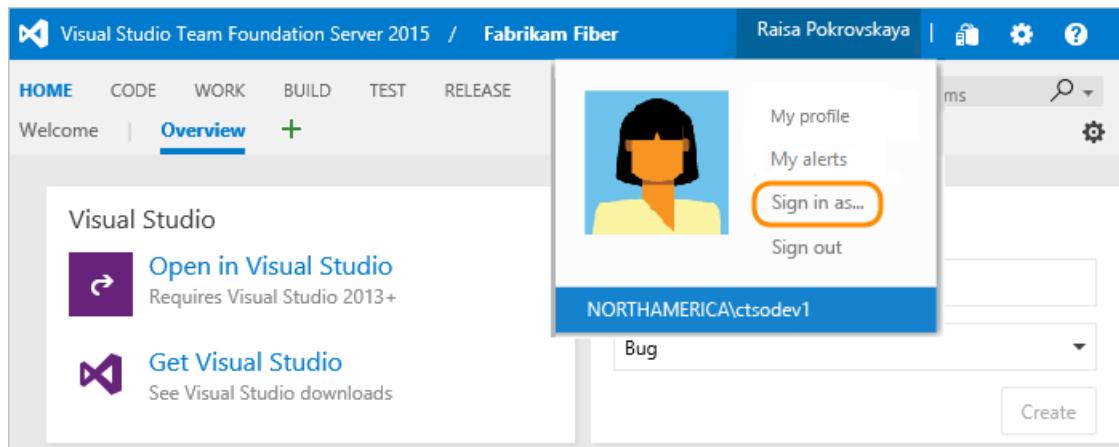
 Get Visual Studio  
See Visual Studio downloads

My profile  
My alerts  
**Sign in as...**  
Sign out

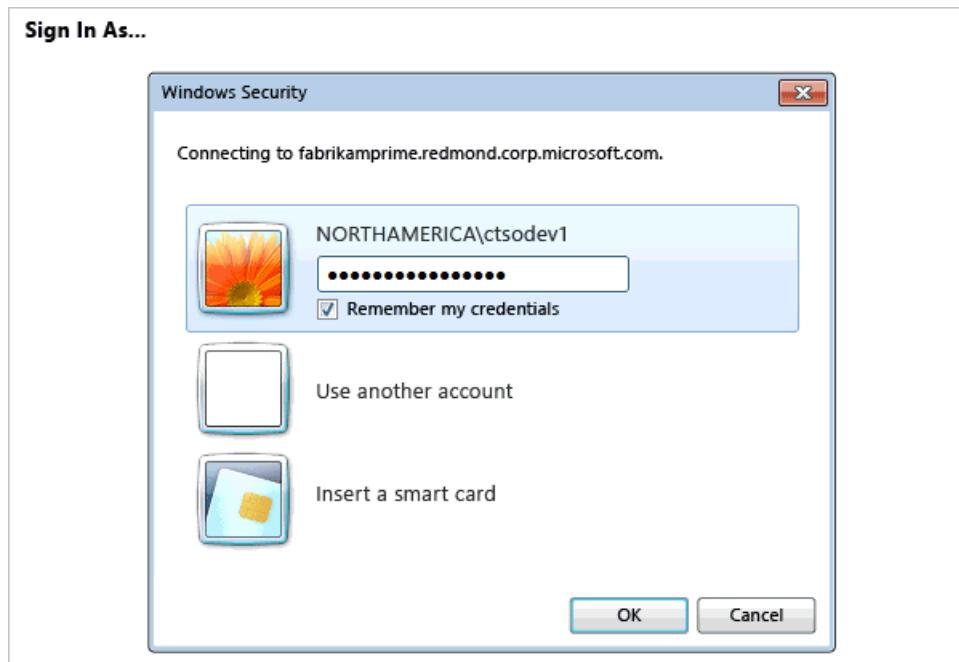
NORTHAMERICA\ctsodev1

Bug

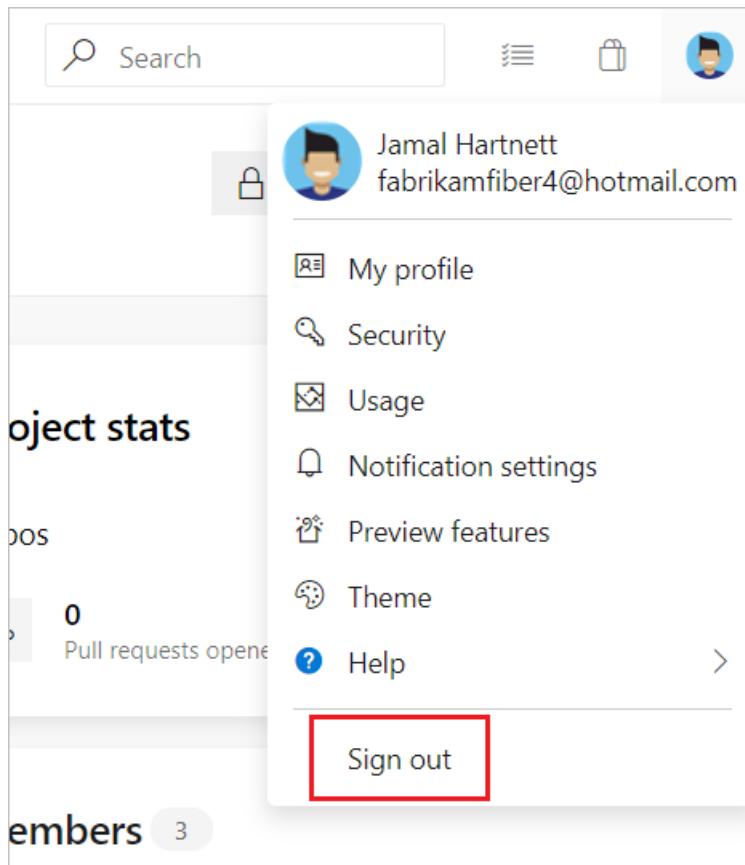
Create



2. Enter your credentials.



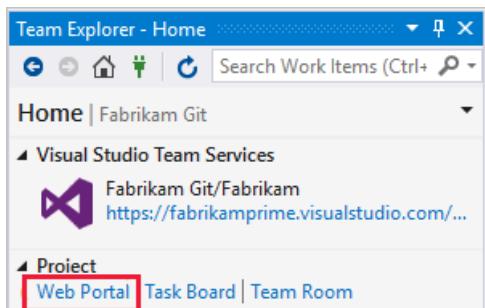
1. Open your profile menu and choose **Sign out**.



2. Choose **Sign in** and enter your credentials.

### Open the web portal from Team Explorer

Open the web portal from the home page.



### Connect from Visual Studio or Team Explorer

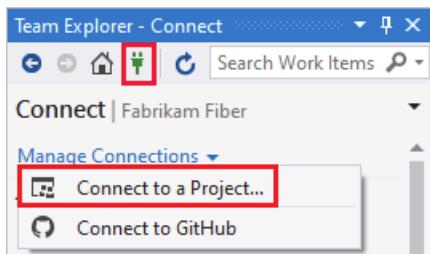
If you haven't already, [download and install a version of Visual Studio](#).

If you're not a member of an Azure DevOps security group, [get added to one](#). Check with a team member. You'll need the names of the server, project collection, and project to connect to.

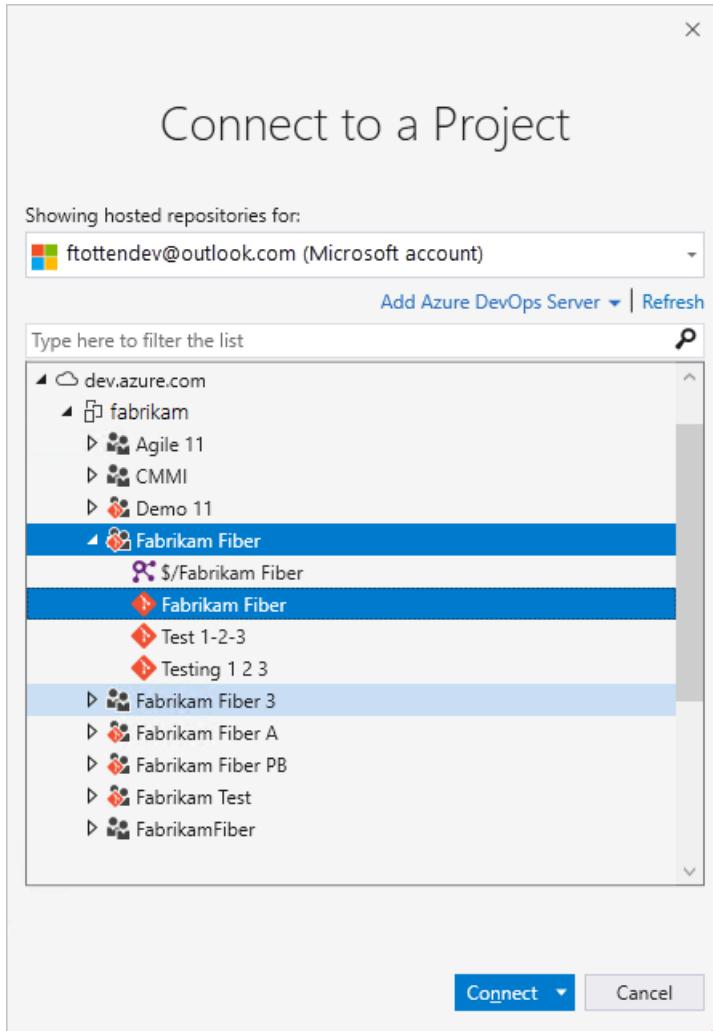
- [Visual Studio 2019](#)
- [Visual Studio 2017](#)
- [Visual Studio 2015](#)

#### Visual Studio 2019

1. Select the **Manage Connections** button in Team Explorer to open the **Connect** page. Choose **Connect to a Project** to select a project to connect to.



Connect to a Project shows the projects you can connect to, along with the repos in those projects.



2. Select **Add Azure DevOps Server** to connect to a project in Azure DevOps Services. Enter the URL to your server and select **Add**.



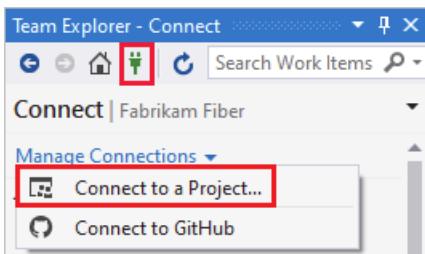
3. Select a project from the list and select **Connect**.

#### Change sign-in credentials

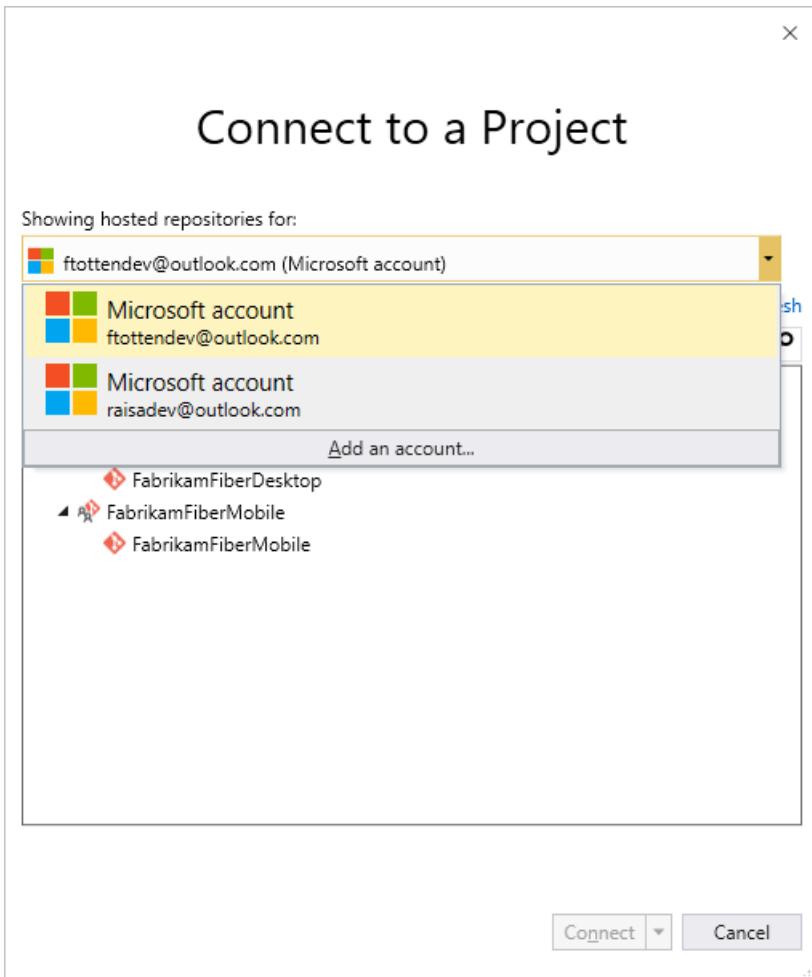
- [Visual Studio 2019](#)
- [Visual Studio 2017](#)
- [Visual Studio 2015](#)

#### Visual Studio 2019

1. From Connect, choose the **Connect to a Project** link to sign in with different credentials.



2. Select a different user or select **Add an account** to access a project using different credentials.

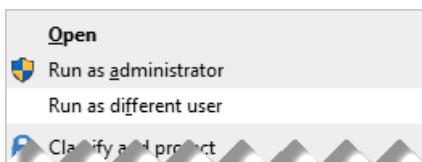


3. Sign in using an account that is associated with an Azure DevOps project, either a valid Microsoft account or GitHub account.

### Use different Visual Studio credentials

You can run Visual Studio with credentials different from your current Windows user account. Find *devenv.exe* under the *Program Files (86)* folder for your version of Visual Studio.

Select Shift and right-click *devenv.exe*, then select **Run as different user**.



### User accounts and licensing for Visual Studio

To connect to a project, you need your user account added to the project. The [organization owner \(Azure DevOps Services\)](#) or a [Project Administrator](#) usually adds user accounts.

Azure DevOps Services provides access to the first five account users free. After that, you need to [pay for more users](#).

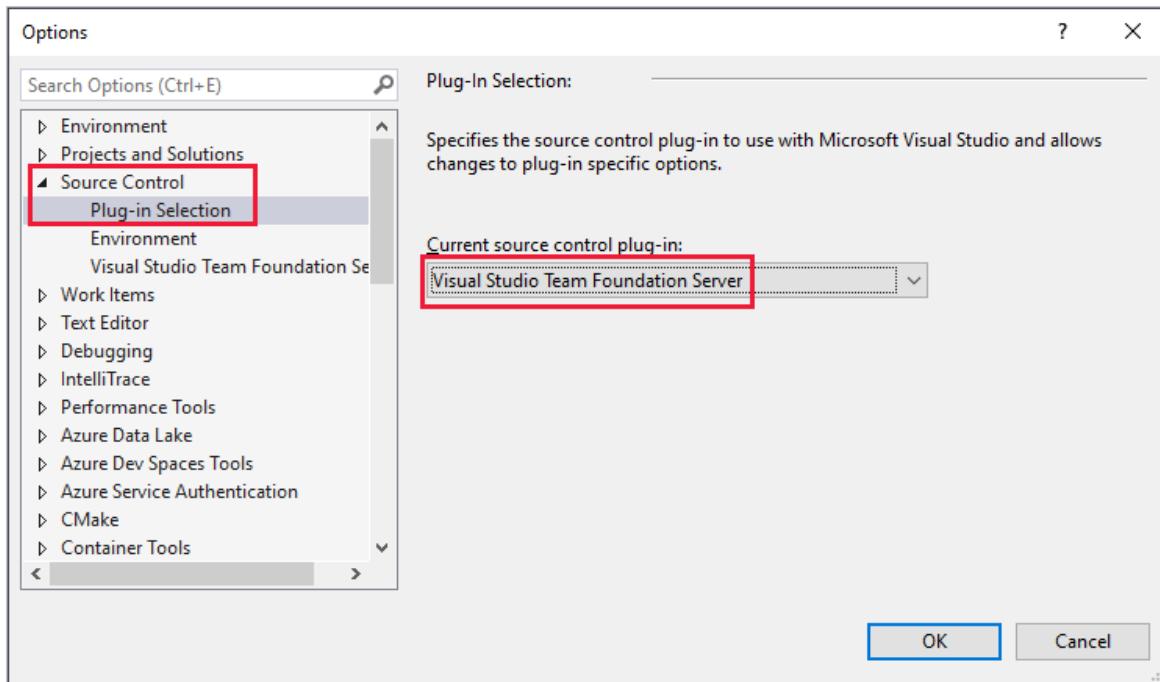
For on-premises TFS, each user account must have a TFS client access license (CAL). All Visual Studio subscriptions and paid Azure DevOps Services users include a TFS CAL. Find out more about licensing from the [Team Foundation Server pricing page](#).

You can also provide access to Stakeholders in your organization who have limited access to select features as described in [Work as a Stakeholder](#).

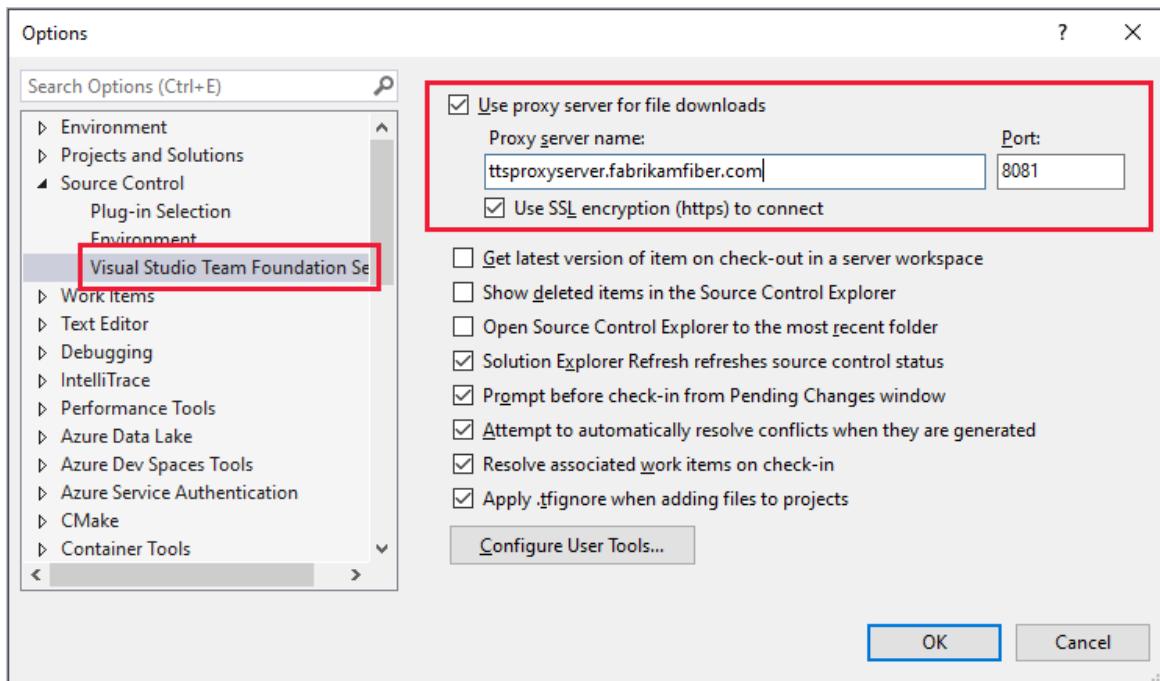
## Configure Visual Studio to connect to Azure DevOps Proxy Server

If your remote team uses a [Azure DevOps Proxy Server](#) to cache files, you can configure Visual Studio to connect through that proxy server and download files under Team Foundation version control.

1. First, make sure that you've connected to Azure DevOps Server as described [in the previous section](#).
2. From the Visual Studio Tools menu, select Options, then select Source Control > Plug-in Selection. Select Visual Studio Team Foundation Server.



3. For **Visual Studio Team Foundation Server**, enter the name and port number for the Azure DevOps Proxy Server. Select **Use SSL encryption (https) to connect**.



Make sure you specify the port number that your administrator assigned to TFS Proxy.

To associate a file type with a compare or merge tool, see [Associate a file type with a file-comparison tool](#) or [Associate a file type with a merge tool](#).

### What other clients support connection to Azure DevOps?

Besides connecting through a web browser, Visual Studio, Eclipse, Excel, and Project you can connect to a project from these clients:

- [Visual Studio Code](#)
- [Visual Studio Community](#)
- [Eclipse: Team Explorer Everywhere](#)
- [Azure Test Plans](#) (formerly Test Manager)
- [Microsoft Feedback Client](#)

### Requirements and client compatibility

Some tasks or features aren't available when you connect to a later version of Azure DevOps Server than your client supports. For more information, see [client compatibility](#).

### Determine your platform version

See [Feedback and support](#).

## Next steps

Learn more about how to:

- [Work in web portal](#)
- [Work in Team Explorer](#)
- [Work in Office Excel or Project](#)
- [Troubleshoot connection](#)

If all you need is a code repository and bug tracking solution, then start with the [Get Started with Azure Repos](#) and [Manage bugs](#).

To start planning and tracking work, see [Get started with Agile tools to plan and track work](#).

# Quickstart: Code with Git

3/18/2021 • 11 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

In this quickstart, learn how to share your code with others. After you create a new organization and project in Azure DevOps, you can begin coding with Git.

To work with a Git repo, you clone it to your computer. Cloning a repo creates a complete local copy of the repo for you to work with. Cloning also downloads all [commits](#) and [branches](#) in the repo, and sets up a named relationship with the repo on the server. Use this relationship to interact with the existing repo, pushing and pulling changes to share code with your team.

## Install Git command-line tools

Install one of the following Git command-line tools:

- To install Git for Windows, including Git Credential Manager, see [Install the Git Credential Manager](#).
- To install on macOS or Linux, check out the [Installing Git](#) chapter in the open-source *Pro Git* book. For macOS and Linux, we recommend [configuring SSH authentication](#)

## Get your code

To get a copy of the source code, you clone the Git repo that contains the code. Cloning creates both a local copy of the source code so you can work with it. Cloning also creates all the version control information so Git can manage the source code.

If you're just getting started with Azure Repos, your code might be in one of several places:

- [I just created my organization in Azure DevOps, so I don't have any code](#)
- [The code is in my \(or my organization's\) Azure Repos Git repo](#)
- [The code is in another Git repo such as GitHub or another Azure Repos Git repo](#)
- [The code is on my local computer and not yet in version control](#)

### I just created my organization in Azure DevOps, so I don't have any code

If you just signed up for Azure DevOps Services, by default you have a project named `MyFirstProject` and a Git repo named `MyFirstProject`. If you want to work in that repo, you can [clone it](#) and then add your code to that repo.

If you want to make a new repo, follow the steps in [Create a new Git repo in your project](#). Then, [clone](#) the new repo and add your code there.

### The code is in my (or my organization's) Azure Repos Git repo

If the code is in your (or your organization's) Azure Repo, you can clone the Git repo to your local computer and start working with it by jumping down to [Clone the repo](#).

### The code is in another Git repo

If the code is in another Git repo, such as a GitHub repo or a different Azure Repo instance, you can import it into a new or existing empty Git repo. Follow the steps in [Import a Git repo](#). Then, return to this article and jump down to [Clone the repo](#).

## The code is on my local computer and not yet in version control

If your code is not yet in version control, you have a couple of options:

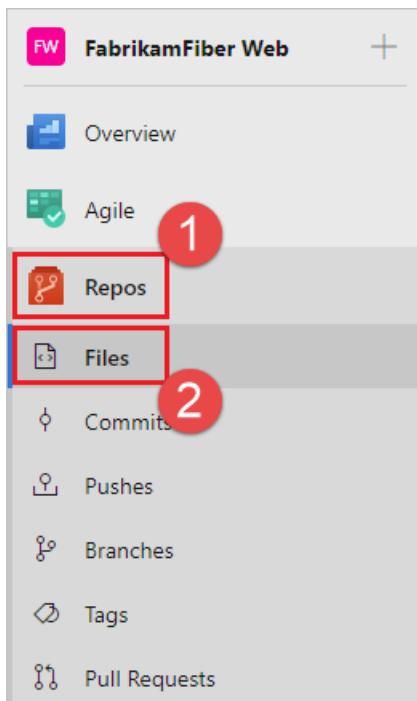
- Create a new repository and add your code there. To create a new repository and add your code there, follow the steps in [Create a new Git repo in your project](#). Then, come back to this article and jump down to [Clone the repo](#).
- Add your code to an existing repository. To do add your code to an existing repository, jump down to [Clone the repo](#).

After the repository is cloned, we'll show you how to add your existing code to the repo.

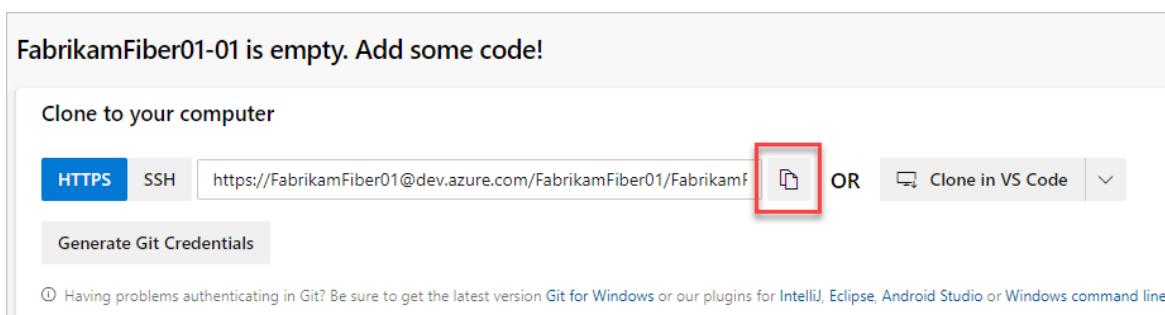
## Clone the repo to your computer

To work with a Git repo, you clone it to your computer. Cloning a repo creates a complete local copy of the repo for you to work with. Cloning also downloads all [commits](#) and [branches](#) in the repo and sets up a named relationship with the repo on the server. Use this relationship to interact with the existing repo, pushing and pulling changes to share code with your team.

1. From your web browser, open the team project for your organization and select **Repos > Files**. If you don't have a team project, [create one now](#).



2. Select **Clone** in the upper-right corner of the Code window and copy the URL.



3. Open the Git command window (Git Bash on Git for Windows). Go to the folder where you want the code from the repo stored on your computer, and run `git clone`, followed by the path copied from **Clone URL** in the previous step. See the following example:

```
git clone https://FabrikamFiber01@dev.azure.com/FabrikamFiber01/FabrikamFiber01-01/_git/FabrikamFiber01-01
```

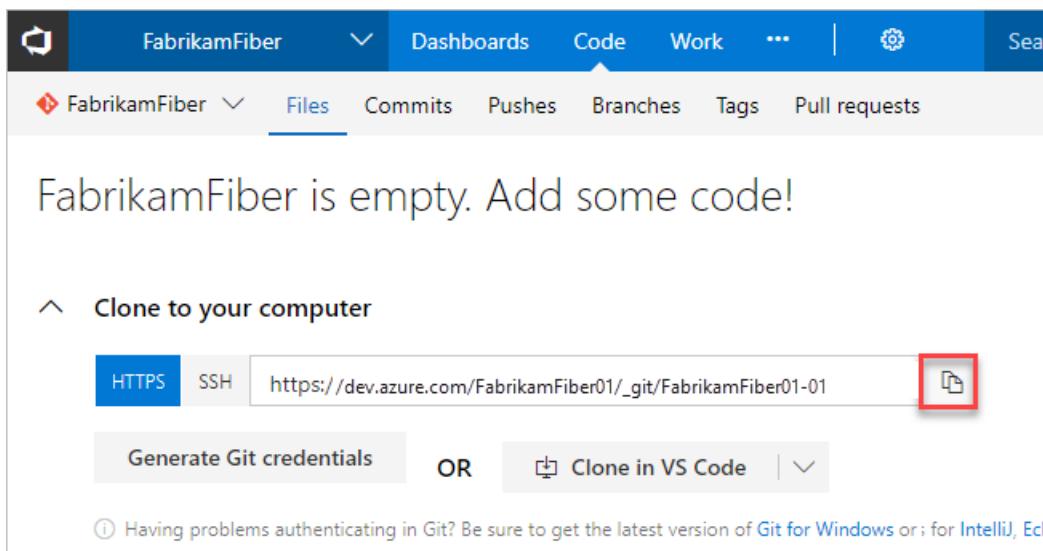
Git downloads a copy of the code, including all [commits](#), and [branches](#) from the repo, into a new folder for you to work with.

4. Switch your directory to the repository that you cloned.

```
cd fabrikam-web
```

Keep this command window open, as you'll use it in the following steps.

1. From your web browser, open the project for your organization, and select **Code**. If you don't have a project, [create one now](#).
2. Select **Clone** in the upper-right corner of the Code window, and copy the URL.



3. Open the Git command window (Git Bash on Git for Windows). Go to the folder where you want the code from the repo stored on your computer, and run `git clone`, followed by the path copied from **Clone URL** in the previous step. See the following example:

```
git clone https://contoso-ltd.visualstudio.com/MyFirstProject/_git/contoso-demo
```

Git downloads a copy of the code in a new folder for you to work with. The download includes all [commits](#) and [branches](#) from the repo.

4. Switch your directory to the repository that you cloned.

```
cd contoso-demo
```

Keep the command window open (use it in the following steps).

## Work in a branch

Git [branches](#) isolate your changes from other work being done in the project. The recommended [Git workflow](#) uses a new branch for every feature or fix that you work on.

Create branches by using the `branch` command. This command creates a reference in Git for the new branch. It

also creates a pointer back to the parent commit so Git can keep a history of changes as you add commits to the branch.

Git always adds new commits to the current local branch. Check what branch you're working on before you commit so that you don't commit changes to the wrong branch.

Switch between local branches by using the `checkout` command. Git will change the files on your computer to match the latest commit on the checked-out branch.

In this step, we'll create a working branch and make a change to the files on your computer in that branch.

Use the `branch` command to create the branch and `checkout` to switch to that branch. In the following example, the new branch is named `users/jamal/feature1`.

```
git branch users/jamal/feature1  
git checkout users/jamal/feature1
```

When you create a branch from the command line, the branch is based on the currently checked-out branch. When you clone the repository, the default branch (typically `main`) is checked out. Because you cloned, your local copy of `main` has the latest changes.

If you're working with a previously cloned repository, ensure that you've checked out the right branch (`git checkout main`) and that it's up to date (`git pull origin main`) before you create your new branch.

```
git checkout main  
git pull origin main  
git branch users/jamal/feature1  
git checkout users/jamal/feature1
```

You can replace the first three commands in the previous example with the following command, which creates a new branch named `users/jamal/feature1` based on the latest `main` branch.

```
git pull origin main:users/jamal/feature1
```

Switch back to the Git Bash window that you used in the previous section. Run the following commands to create and check out a new branch based on the main branch.

```
git pull origin main:users/jamal/feature1  
git checkout feature1
```

Browse to the location of the repository on your local computer, make an edit to one of the files, and save it. If you're adding code from your local computer to the repository, you can add it here by copying it to the folder where you cloned the repository.

## Work with the code

In the following steps, we make a change to the files on your computer, commit the changes locally, and push the commit to the repo stored on the server. We can then view the changes.

1. Browse to the folder on your computer where you cloned the repo, open the `README.md` file in your editor of choice, and make some changes. Then save and close the file.
2. In the Git command window, go to the `contoso-demo` directory by entering the following command:

```
cd contoso-demo
```

3. Commit your changes by entering the following commands in the Git command window:

```
git add .
git commit -m "My first commit"
```

The `git add .` command stages any new or changed files, and `git commit -m` creates a commit with the specified commit message.

4. Push your changes to the Git repo on the server. Enter the following command into the Git command window:

```
git push origin users/jamal/feature1
```

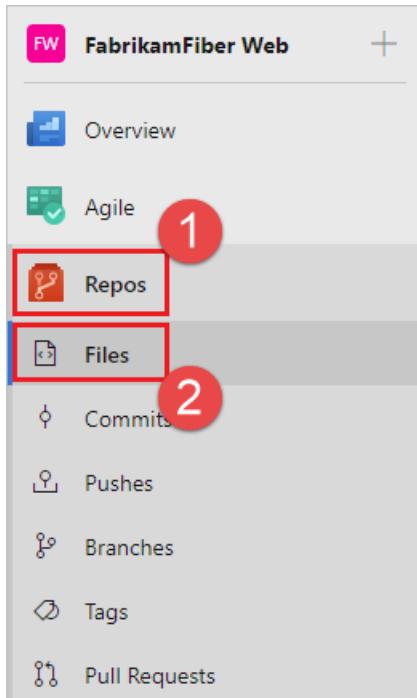
Your code is now shared to the remote repository, in a branch named `users/jamal/feature1`. To merge the code from your working branch into the `main` branch, use a pull request.

## Review and merge your changes with a pull request

Pull requests combine the review and merge of your code into a single collaborative process. After you're done fixing a bug or new feature in a branch, create a new pull request. Add the members of the team to the pull request so they can review and vote on your changes. Use pull requests to review works in progress and get early feedback on changes. There's no commitment to merge the changes because you can abandon the pull request at any time.

This example shows the basic steps of creating and completing a pull request.

1. From your web browser, open the team project for your organization and select **Repos** > **Files**. If you kept your browser open after getting the clone URL, you can just switch back to it.



2. Select **Create a pull request** in the upper-right corner of the **Files** window. If you don't see a message like **You updated users/jamal/feature1 just now**, refresh your browser.

Find a file or folder...

Set up build Fork Clone

(i) You updated [users/jamal/feature1](#) just now — Create a pull request X

Contents History README | + New ↘ Upload file(s) ↕ Download as Zip ↗

Name ↑	Last change	Commits
AzureEndpoint.png	5/2/2018	ae9e9911 Initial Commi...
gear.png	5/2/2018	ae9e9911 Initial Commi...
README.md	5/2/2018	ae9e9911 Initial Commi...

3. New pull requests are configured to merge your branch into the default branch, which in this example is `main`. The title and description are pre-populated with your commit message.

### New Pull Request

into `main`

Title \*

Add label

Description

My first commit

Markdown supported.

Aa **B** *I* @ #

My first commit

Reviewers

Search users and groups to add as reviewers

Work Items

Search work items by ID or title

**Create**

You can [add reviewers](#) and [link work items](#) to your pull request.

You can review the files included in the pull request at the bottom of the **New Pull Request** window.

The screenshot shows a pull request interface with a 'Create' button at the top right. Below it, 'Files (1)' and 'Commits (1)' tabs are selected. A message says 'Showing 1 file change: 1 edit'. The file 'README.md' is shown with a diff:

```
M README.md +1 -1 /README.md
...
13 13
14 14 -----
15 15
16 - Please follow below exercises inorder to deploy your application, :
16 + Please follow below exercises inorder to deploy your application:
17 17
18 18 ## Exercise 1: Endpoint Creation
19 19
```

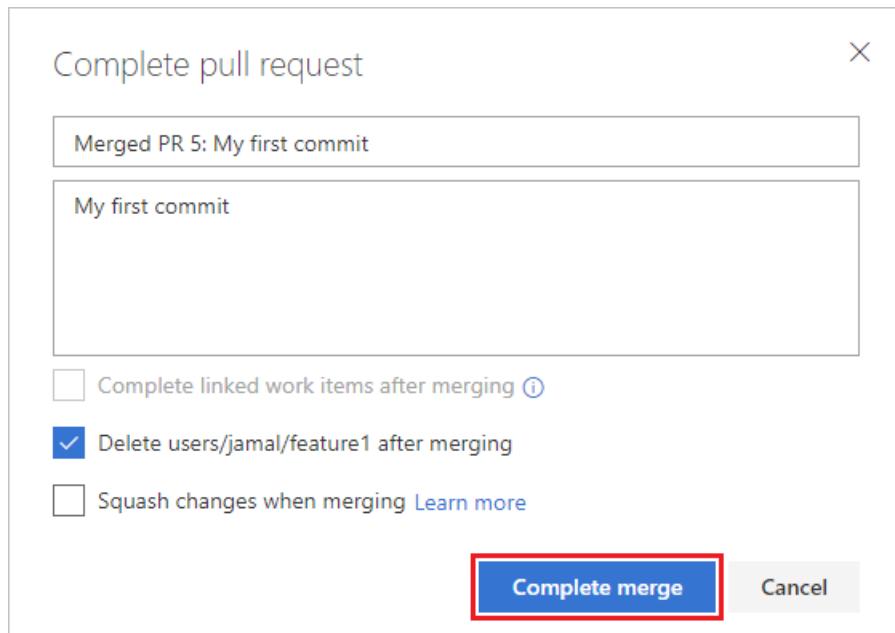
Select **Create** to create the pull request.

4. You can view the details of your pull request from the **Overview** tab. You can also view the changed files, updates, and commits in your pull request from the other tabs. Select **Complete** to begin the process of completing the pull request.

The screenshot shows the 'My first commit' pull request overview. The 'Overview' tab is selected and highlighted with a red box. The 'Complete' button in the top right is also highlighted with a red box. Other tabs include 'Files', 'Updates', and 'Commits'. The pull request details include:

- Description: My first commit
- Show everything dropdown
- Add a comment input field
- Created by: Jamal Hartnett just now
- Work Items: No related work items
- Reviewers: No reviewers
- Labels: Add label

5. Select **Complete merge** to complete the pull request and merge your code into the **main** branch.



#### NOTE

This example shows the basic steps of creating and completing a pull request. To learn more about pull requests, including voting and reviewing, commenting, autocomplete, and more, see [Create, view, and manage pull requests](#).

1. From your web browser, open the team project for your organization and select the **Code** page. If you don't have a team project, [create one now](#).
2. Select **Clone** in the upper-right corner of the **Code** page and copy the **Clone URL**.

3. Open the Git command window, for example Git Bash on Git for Windows, and browse to the folder where you want the code from the repo that is stored on your computer. Run `git clone` followed by the path copied from the **Clone URL** in the previous section, as shown in the following example.

```
git clone https://dev.azure.com/contoso-ltd/MyFirstProject/_git/contoso-demo
```

Git downloads a copy of the code into a new folder for you to work with. The download includes all [commits](#) and [branches](#) from the repo.

4. Switch your directory to the repository that you cloned.

```
cd fabrikam-web
```

Keep this command window open, because you'll use it in the following steps.

Your changes are now merged into the `main` branch, and your `users/jamal/feature1` branch is deleted on the remote repository. To delete your local copy of the branch, switch back to your Git Bash command prompt and

run the following commands.

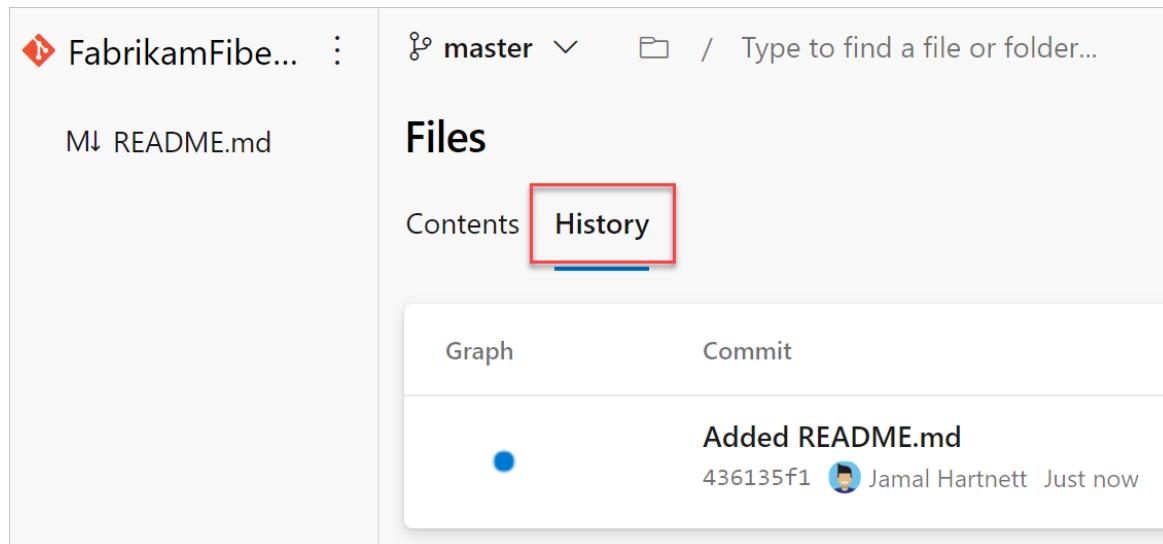
```
git checkout main  
git pull origin main  
git branch -d users/jamal/feature1
```

The `git checkout main` command switches you to the `main` branch. The `git pull origin main` command pulls down the latest version of the code in the main branch, including your changes and the fact that `users/jamal/feature1` was merged. The `git branch -d users/jamal/feature1` command deletes your local copy of that branch.

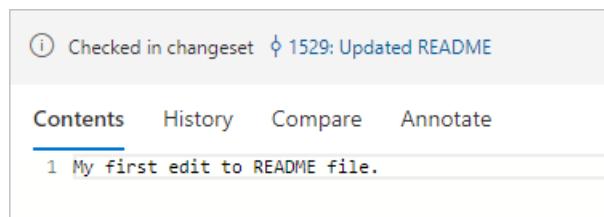
Now you're ready to create a new branch, write some code, and do it again.

## View history

1. Switch back to the web portal, and select **History** from the **Code** page to view your new commit.



2. Switch to the **Files** tab, and select the README file to view your changes.



1. Switch back to the web portal, and select **History** from the **Code** tab to view your new commit. Two commits appear: the first commit, where the README and .gitignore were added upon repo creation, and the commit you just made.

The screenshot shows the GitHub interface for the repository 'MyFirstProject'. The 'Code' tab is selected (1). The 'History' tab is also highlighted (2). The commit history displays two commits:

	Graph	Commit	Message
1	4b38f92b	My first commit	
2	fefb3a74	Added README.md, .gitignore (VisualStudio) files	

2. Switch to the Files tab, and select the README file to view your changes.

The screenshot shows the GitHub interface for the repository 'MyFirstProject'. The 'Files' tab is selected (1). The 'README.md' file is selected (2). The file content is displayed as follows:

```
Introduction
This is my first edit.

TODO: Give a short introduction of your project. Let the user know what the project does, who created it, etc.

Getting Started
TODO: Guide users through getting your code up and running.
```

## Next steps

Set up continuous integration & delivery or learn more about working with a Git repo.

# Create your first pipeline

5/21/2021 • 32 minutes to read • [Edit Online](#)

Azure Pipelines | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 | TFS 2017

This is a step-by-step guide to using Azure Pipelines to build a GitHub repository.

## Prerequisites - Azure DevOps

- A GitHub account, where you can create a repository. If you don't have one, you can [create one for free](#).
- An Azure DevOps organization. If you don't have one, you can [create one for free](#). (An Azure DevOps organization is different from your GitHub organization. Give them the same name if you want alignment between them.)

If your team already has one, then make sure you're an administrator of the Azure DevOps project that you want to use.

### NOTE

If you want to create a new pipeline by copying another pipeline, see [Clone or import a pipeline](#).

## Create your first pipeline

- [Java](#)
- [.NET](#)
- [Python](#)
- [JavaScript](#)
- [Azure CLI \(Java\)](#)

### Get the Java sample code

To get started, fork the following repository into your GitHub account.

```
https://github.com/MicrosoftDocs/pipelines-java
```

### Create your first Java pipeline

1. Sign in to your Azure DevOps organization and navigate to your project.
2. In your project, navigate to the **Pipelines** page. Then choose the action to create a new pipeline.
3. Walk through the steps of the wizard by first selecting **GitHub** as the location of your source code.
4. You might be redirected to GitHub to sign in. If so, enter your GitHub credentials.
5. When the list of repositories appears, select your desired sample app repository.
6. Azure Pipelines will analyze your repository and recommend a Maven pipeline template. Select **Save and run**, then select **Commit directly to the main branch**, and then choose **Save and run again**.
7. A new run is started. Wait for the run to finish.

Learn more about [working with Java](#) in your pipeline.

## Add a status badge to your repository

Many developers like to show that they're keeping their code quality high by displaying a status badge in their repo.



To copy the status badge to your clipboard:

1. In Azure Pipelines, go to the **Pipelines** page to view the list of pipelines. Select the pipeline you created in the previous section.
2. In the context menu for the pipeline, select **Status badge**.
3. Copy the sample Markdown from the status badge panel.

Now with the badge Markdown in your clipboard, take the following steps in GitHub:

1. Go to the list of files and select `Readme.md`. Select the pencil icon to edit.
2. Paste the status badge Markdown at the beginning of the file.
3. Commit the change to the `master` branch.
4. Notice that the status badge appears in the description of your repository.

To configure anonymous access to badges for private projects:

1. Navigate to **Project Settings**
2. Open the **Settings** tab under **Pipelines**
3. Toggle the **Disable anonymous access to badges** slider under **General**

### NOTE

Even in a private project, anonymous badge access is enabled by default. With anonymous badge access enabled, users outside your organization might be able to query information such as project names, branch names, job names, and build status through the badge status API.

Because you just changed the `Readme.md` file in this repository, Azure Pipelines automatically builds your code, according to the configuration in the `azure-pipelines.yml` file at the root of your repository. Back in Azure Pipelines, observe that a new run appears. Each time you make an edit, Azure Pipelines starts a new run.

## Manage your pipeline with Azure CLI

You can manage the pipelines in your organization using these `az pipelines` commands:

- [`az pipelines run`](#): Run an existing pipeline
- [`az pipelines update`](#): Update an existing pipeline
- [`az pipelines show`](#): Show the details of an existing pipeline

These commands require either the name or ID of the pipeline you want to manage. You can get the ID of a pipeline using the [`az pipelines list`](#) command.

## Run a pipeline

You can queue (run) an existing pipeline with the [az pipelines run](#) command. To get started, see [Get started with Azure DevOps CLI](#).

```
az pipelines run [--branch]
                  [--commit-id]
                  [--folder-path]
                  [--id]
                  [--name]
                  [--open]
                  [--org]
                  [--project]
                  [--variables]
```

### Parameters

- **branch**: Name of the branch on which the pipeline run is to be queued, for example, `refs/heads/main`.
- **commit-id**: Commit-id on which the pipeline run is to be queued.
- **folder-path**: Folder path of pipeline. Default is root level folder.
- **id**: Required if **name** is not supplied. ID of the pipeline to queue.
- **name**: Required if ID is not supplied, but ignored if ID is supplied. Name of the pipeline to queue.
- **open**: Open the pipeline results page in your web browser.
- **org**: Azure DevOps organization URL. You can configure the default organization using  
`az devops configure -d organization=ORG_URL`. Required if not configured as default or picked up using  
`git config`. Example: `--org https://dev.azure.com/MyOrganizationName/`.
- **project**: Name or ID of the project. You can configure the default project using  
`az devops configure -d project=NAME_OR_ID`. Required if not configured as default or picked up using  
`git config`.
- **variables**: Space separated "name=value" pairs for the variables you would like to set.

### Example

The following command runs the pipeline named `myGithubname.pipelines-java` in the branch `pipeline` and shows the result in table format.

```
az pipelines run --name myGithubname.pipelines-java --branch pipeline --output table

Run ID      Number      Status      Result      Pipeline ID      Pipeline Name          Source Branch
Queued Time           Reason
-----  -----
-----  -----
123        20200123.2  notStarted    12          myGithubname.pipelines-java  pipeline
2020-01-23 11:55:56.633450  manual
```

## Update a pipeline

You can update an existing pipeline with the [az pipelines update](#) command. To get started, see [Get started with Azure DevOps CLI](#).

```
az pipelines update [--branch]
                  [--description]
                  [--id]
                  [--name]
                  [--new-folder-path]
                  [--new-name]
                  [--org]
                  [--project]
                  [--queue-id]
                  [--yaml-path]
```

#### Parameters

- **branch**: Name of the branch on which the pipeline run is to be configured, for example, `refs/heads/main`.
- **description**: New description for the pipeline.
- **id**: Required if **name** is not supplied. ID of the pipeline to update.
- **name**: Required if ID is not supplied. Name of the pipeline to update.
- **new-folder-path**: New full path of the folder to which the pipeline is moved, for example, `user1/production_pipelines`.
- **new-name**: New updated name of the pipeline.
- **org**: Azure DevOps organization URL. You can configure the default organization using  
`az devops configure -d organization=ORG_URL`. Required if not configured as default or picked up using  
`git config .` Example: `--org https://dev.azure.com/MyOrganizationName/`.
- **project**: Name or ID of the project. You can configure the default project using  
`az devops configure -d project=NAME_OR_ID`. Required if not configured as default or picked up using  
`git config .`
- **queue-id**: Queue ID of the agent pool where the pipeline needs to run.
- **yaml-path**: Path of the pipeline's yaml file in the repo.

#### Example

The following command updates the pipeline with the ID of 12 with a new name and description and shows the result in table format.

```
az pipelines update --id 12 --description "rename pipeline" --new-name updatedname.pipelines-java --output table

ID      Name           Status     Default Queue
----  -----
12    updatedname.pipelines-java  enabled   Hosted Ubuntu 1604
```

## Show pipeline

You can view the details of an existing pipeline with the `az pipelines show` command. To get started, see [Get started with Azure DevOps CLI](#).

```
az pipelines show [--folder-path]
                  [--id]
                  [--name]
                  [--open]
                  [--org]
                  [--project]
```

#### Parameters

- **folder-path**: Folder path of pipeline. Default is root level folder.
- **id**: Required if **name** is not supplied. ID of the pipeline to show details.
- **name**: Required if **name** is not supplied, but ignored if ID is supplied. Name of the pipeline to show details.

- **open**: Open the pipeline summary page in your web browser.
- **org**: Azure DevOps organization URL. You can configure the default organization using `az devops configure -d organization=ORG_URL`. Required if not configured as default or picked up using `git config`. Example: `--org https://dev.azure.com/MyOrganizationName/`.
- **project**: Name or ID of the project. You can configure the default project using `az devops configure -d project=NAME_OR_ID`. Required if not configured as default or picked up using `git config`.

#### Example

The following command shows the details of the pipeline with the ID of 12 and returns the result in table format.

```
az pipelines show --id 12 --output table

ID      Name           Status    Default Queue
-----  -----          -----    -----
12      updatedname.pipelines-java  enabled   Hosted Ubuntu 1604
```

#### NOTE

In Microsoft Team Foundation Server (TFS) 2018 and previous versions, build and release *pipelines* are called *definitions*, *runs* are called *builds*, *service connections* are called *service endpoints*, *stages* are called *environments*, and *jobs* are called *phases*.

#### NOTE

This guidance applies to TFS version 2017.3 and newer.

We'll show you how to use the classic editor in Azure DevOps Server 2019 to create a build and release that prints "Hello world".

We'll show you how to use the classic editor in TFS to create a build and a release that prints "Hello world".

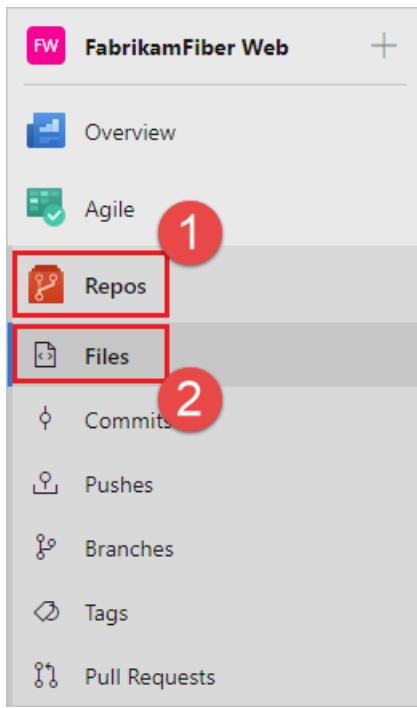
## Prerequisites

- A [self-hosted Windows agent](#).

## Initialize your repository

If you already have a repository in your project, you can skip to the next step: [Skip to adding a script to your repo](#)

1. Go to [Azure Repos](#). (The Code hub in the previous navigation)



2. If your project is empty, you will be greeted with a screen to help you add code to your repository. Choose the bottom choice to **initialize** your repo with a `readme` file:

FabrikamFiber Pipelines is empty. Add some code!

Clone to your computer

HTTPS SSH [https://regius.visualstudio.com/FabrikamFiber%20Pipelines/\\_git/Fabrika...](https://regius.visualstudio.com/FabrikamFiber%20Pipelines/_git/Fabrika...) OR Clone in VS Code

Generate Git credentials

Having problems authenticating in Git? Be sure to get the latest version of [Git for Windows](#) or our plugins for [IntelliJ](#), [Eclipse](#), [Android Studio](#) or [Windows command line](#).

or push an existing repository from command line

HTTPS SSH

```
git remote add origin https://regius.visualstudio.com/FabrikamFiber%20Pipelines/_git/FabrikamFiber%20Pipelines
git push -u origin --all
```

or import a repository

Import

or initialize with a README or .gitignore

Add a README  Initialize

1. Navigate to your repository by clicking **Code** in the top navigation.
2. If your project is empty, you will be greeted with a screen to help you add code to your repository. Choose the bottom choice to **initialize** your repo with a `readme` file:

## FabrikamFiber Pipelines is empty. Add some code!

### Clone to your computer

HTTPS SSH [https://regius.visualstudio.com/FabrikamFiber%20Pipelines/\\_git/FabrikamFiber%20Pipelines](https://regius.visualstudio.com/FabrikamFiber%20Pipelines/_git/FabrikamFiber%20Pipelines) OR [Clone in VS Code](#)

Generate Git credentials

Having problems authenticating in Git? Be sure to get the latest version of [Git for Windows](#) or our plugins for [IntelliJ](#), [Eclipse](#), [Android Studio](#) or [Windows command line](#).

### or push an existing repository from command line

HTTPS SSH

```
git remote add origin https://regius.visualstudio.com/FabrikamFiber%20Pipelines/_git/FabrikamFiber%20Pipelines  
git push -u origin --all
```

### or import a repository

Import

#### or initialize with a README or gitignore

Add a README

Add a .gitignore: None

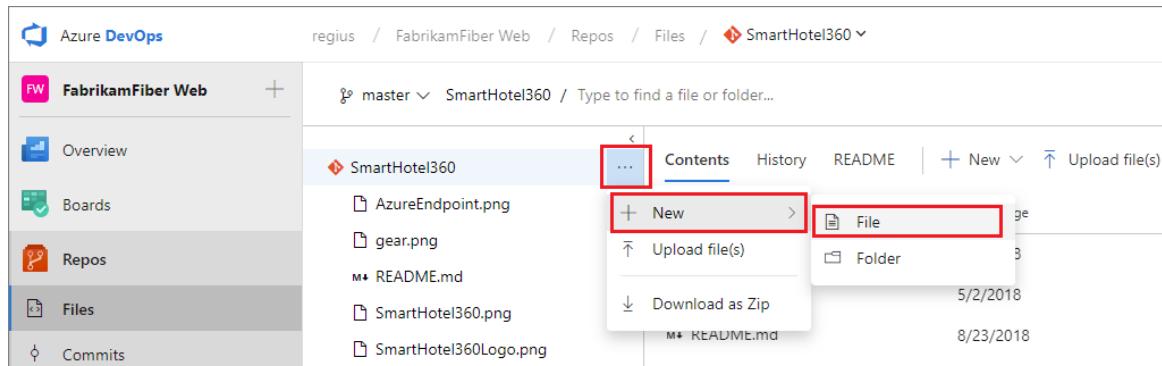
Initialize

## Add a script to your repository

Create a PowerShell script that prints `Hello world`.

1. Go to **Azure Repos**.

2. Add a file.



3. In the dialog box, name your new file and create it.

HelloWorld.ps1

4. Copy and paste this script.

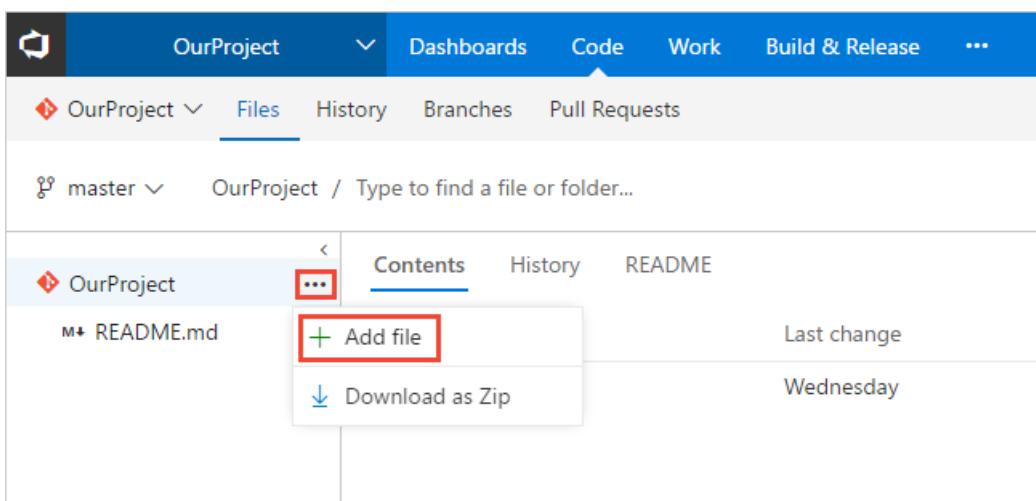
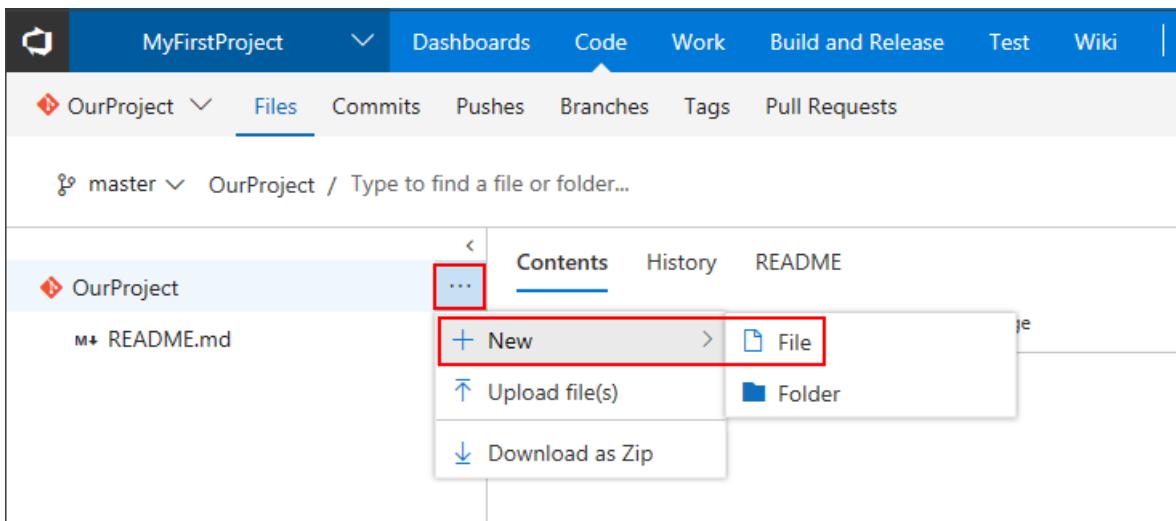
```
Write-Host "Hello world"
```

5. Commit (save) the file.

1. Go to the **Code hub**.

2. Add a file.

- TFS 2018.2
- TFS 2018 RTM



1. In the dialog box, name your new file and create it.

```
HelloWorld.ps1
```

2. Copy and paste this script.

```
Write-Host "Hello world"
```

3. Commit (save) the file.

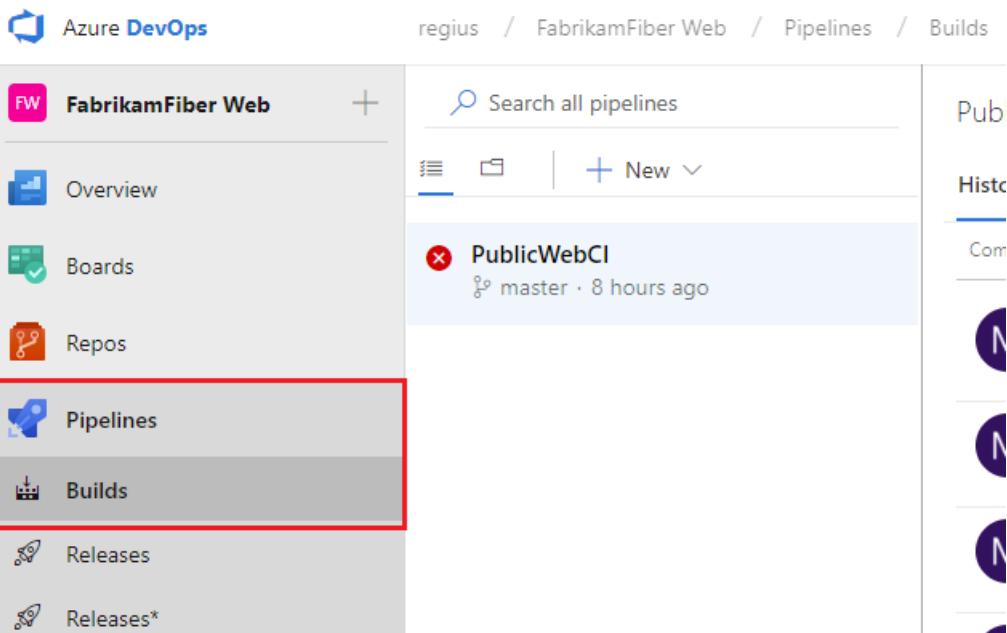
In this tutorial, our focus is on CI/CD, so we're keeping the code part simple. We're working in an Azure Repos Git repository directly in your web browser.

When you're ready to begin building and deploying a real app, you can use a wide range of version control clients and services with Azure Pipelines CI builds. [Learn more](#).

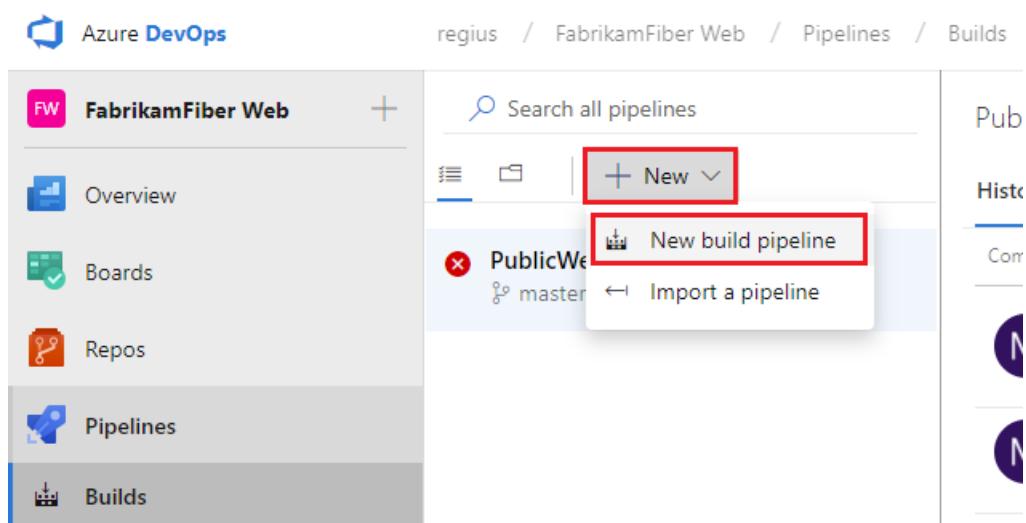
## Create a build pipeline

Create a build pipeline that prints "Hello world."

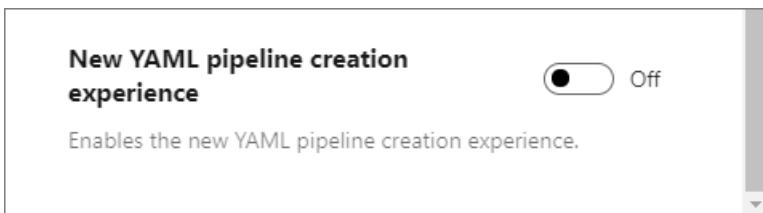
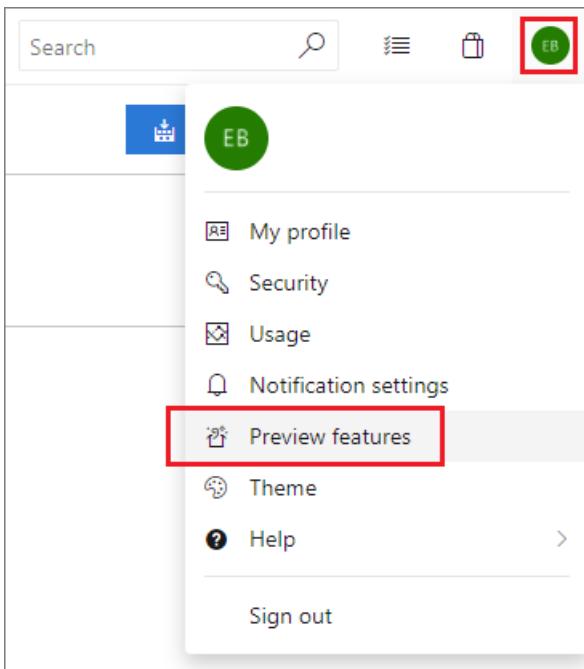
1. Select **Azure Pipelines**, it should automatically take you to the **Builds** page.



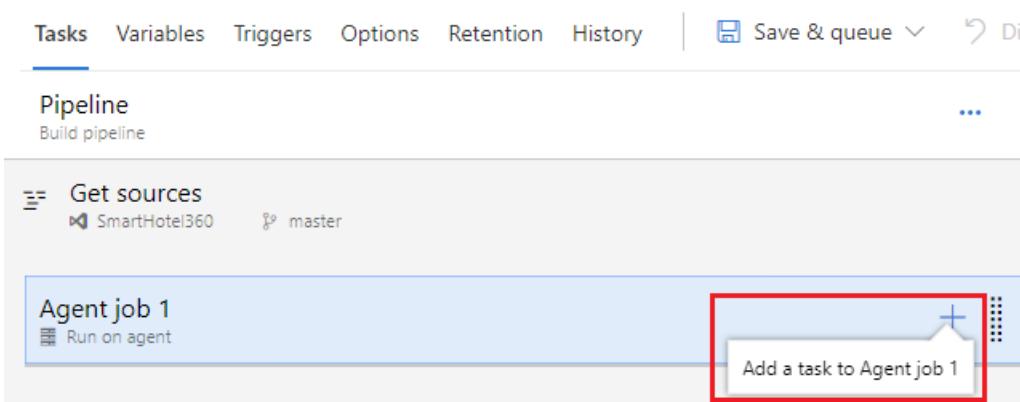
2. Create a new pipeline.



For new Azure DevOps users, this will automatically take you to the *YAML pipeline creation experience*. To get to the classic editor and complete this guide, you must turn off the **preview feature** for the *New YAML pipeline creation experience*.



3. Make sure that the **source**, **project**, **repository**, and default **branch** match the location in which you created the script.
4. Start with an **Empty job**.
5. On the left side, select **Pipeline** and specify whatever **Name** you want to use. For the **Agent pool**, select **Hosted VS2017**.
6. On the left side, select the plus sign ( + ) to add a task to **Job 1**. On the right side, select the **Utility** category, select the **PowerShell** task from the list, and then choose **Add**.



7. On the left side, select your new **PowerShell** script task.
8. For the **Script Path** argument, select the ... button to browse your repository and select the script you created.

The screenshot shows the 'Pipeline' tab in the Azure DevOps interface. A 'Get sources' task is at the top, followed by an 'Agent job 1' section containing a 'PowerShell Script' task. On the right, a configuration pane for the PowerShell task is open, showing the following details:

- PowerShell**: Version 2.\*
- Display name**: PowerShell Script
- Type**: File Path (radio button selected)
- Script Path**: HelloWorld.ps1 (with a red box around the ellipsis button)

9. Select **Save & queue**, and then select **Save**.

10. Select **Build and Release**, and then choose **Builds**.

The screenshot shows the 'Builds' page under the 'Build and Release' section. It lists two files: 'HelloWorld.ps1' and 'README.md'. The 'HelloWorld.ps1' file was last changed 9 minutes ago.

11. Create a new pipeline.

The screenshot shows the 'Build Definitions' page. It includes tabs for 'Builds', 'Releases', 'Library', 'Task Groups', and 'Deployment Groups'. A red box highlights the '+ New' button.

12. Start with an **empty pipeline**

13. Select **Pipeline** and specify whatever **Name** you want to use. For the **Agent pool**, select **Default**.

14. On the left side, select **+ Add Task** to add a task to the job, and then on the right side select the **Utility** category, select the **PowerShell** task, and then choose **Add**.

The screenshot shows an empty pipeline configuration screen. At the top, there are tabs for 'Tasks', 'Variables', 'Triggers', 'Options', 'Retention', and 'History'. Below these, a 'Process' section is shown with the sub-section 'Build process'. Under 'Build process', there is a 'Get sources' task and a red box highlighting the '+ Add Task' button.

15. On the left side, select your new **PowerShell** script task.

16. For the **Script Path** argument, select the **...** button to browse your repository and select the script you created.

17. Select **Save & queue**, and then select **Save**.

1. Select **Azure Pipelines**, and then the **Builds** tab.

2. Create a new pipeline.

3. Start with an **empty pipeline**.

4. Select **Pipeline** and specify whatever **Name** you want to use.

5. On the **Options** tab, select **Default** for the **Agent pool**, or select whichever pool you want to use that has Windows build agents.

6. On the **Tasks** tab, make sure that **Get sources** is set with the **Repository** and **Branch** in which you created the script.

7. On the left side select **Add Task**, and then on the right side select the **Utility** category, select the **PowerShell** task, and then select **Add**.

8. On the left side, select your new **PowerShell** script task.

9. For the **Script Path** argument, select the **...** button to browse your repository and select the script you created.

The screenshot shows the 'Tasks' tab selected in the build pipeline editor. A 'PowerShell Script' task is currently selected. In the configuration pane, the 'Script Path' field is set to 'HelloWorld.ps1', with a red box highlighting the ellipsis button at the end of the input field.

10. Select **Save & queue**, and then select **Save**.

A build pipeline is the entity through which you define your automated build pipeline. In the build pipeline, you compose a set of tasks, each of which perform a step in your build. The task catalog provides a rich set of tasks for you to get started. You can also add PowerShell or shell scripts to your build pipeline.

## Publish an artifact from your build

A typical build produces an artifact that can then be deployed to various stages in a release. Here to demonstrate the capability in a simple way, we'll simply publish the script as the artifact.

1. On the **Tasks** tab, select the plus sign ( + ) to add a task to **Job 1**.
2. Select the **Utility** category, select the **Publish Build Artifacts** task, and then select **Add**.

The screenshot shows the 'Tasks' tab selected in the build pipeline editor. A 'Publish Build Artifacts' task is currently selected. In the configuration pane, the 'Path to publish' field is set to 'HelloWorld.ps1', the 'Artifact name' field is set to 'drop', and the 'Artifact publish location' field is set to 'Visual Studio Team Services/TFS'. All three of these fields are highlighted with a red box.

**Path to publish:** Select the ... button to browse and select the script you created.

**Artifact name:** Enter `drop`.

**Artifact publish location:** Select **Azure Artifacts/TFS**.

1. On the **Tasks** tab, select **Add Task**.
2. Select the **Utility** category, select the **Publish Build Artifacts** task, and then select **Add**.

The screenshot shows the 'Tasks' tab of an Azure Pipeline. On the left, there's a list of tasks: 'Get sources' (OurProject, master), 'PowerShell Script' (PowerShell), and 'Publish Artifact: drop' (selected). The 'Publish Artifact: drop' task has a checkmark and three vertical dots. To the right, under 'Options', is a 'Publish Build Artifacts' section. It includes a dropdown for 'Version 1.\*', a 'Link settings' button, and a 'Display name' input field containing 'Publish Artifact: drop'. Below these are three fields: 'Path to Publish' (HelloWorld.ps1), 'Artifact Name' (drop), and 'Artifact Type' (Server). The 'Path to Publish' field is highlighted with a red box.

**Path to Publish:** Select the ... button to browse and select the script you created.

**Artifact Name:** Enter `drop`.

**Artifact Type:** Select `Server`.

Artifacts are the files that you want your build to produce. Artifacts can be nearly anything your team needs to test or deploy your app. For example, you've got a .DLL and .EXE executable files and .PDB symbols file of a C# or C++ .NET Windows app.

To enable you to produce artifacts, we provide tools such as copying with pattern matching, and a staging directory in which you can gather your artifacts before publishing them. See [Artifacts in Azure Pipelines](#).

## Enable continuous integration (CI)

1. Select the Triggers tab.
2. Enable Continuous integration.

A continuous integration trigger on a build pipeline indicates that the system should automatically queue a new build whenever a code change is committed. You can make the trigger more general or more specific, and also schedule your build (for example, on a nightly basis). See [Build triggers](#).

## Save and queue the build

Save and queue a build manually and test your build pipeline.

1. Select **Save & queue**, and then select **Save & queue**.
2. On the dialog box, select **Save & queue** once more.

This queues a new build on the Microsoft-hosted agent.

3. You see a link to the new build on the top of the page.

The screenshot shows the Azure Pipelines interface for the 'FabrikamFiber Web' project. A green notification bar at the top right indicates that 'Build #116 has been queued.' The pipeline name 'FabrikamFiber Web-CI' is visible above the queue status.

Choose the link to watch the new build as it happens. Once the agent is allocated, you'll start seeing the live logs of the build. Notice that the PowerShell script is run as part of the build, and that "Hello world" is printed to the console.

The screenshot shows the build summary for 'Agent job 1 Job'. The 'Logs' tab is selected, displaying the PowerShell script output:

```
PowerShell Script
1 2018-08-27T17:30:29.8976103Z ##[section]Starting: PowerShell Script
2 2018-08-27T17:30:29.8988064Z =====
3 2018-08-27T17:30:29.8988307Z Task : PowerShell
4 2018-08-27T17:30:29.8988513Z Description : Run a PowerShell script on Windows, macOS, or Linux.
5 2018-08-27T17:30:29.8988736Z Version : 2.136.0
6 2018-08-27T17:30:29.8988923Z Author : Microsoft Corporation
7 2018-08-27T17:30:29.8989137Z Help : [More Information](https://go.microsoft.com/fwlink/?LinkID=857000)
8 2018-08-27T17:30:32.6103320Z Generating script.
9 2018-08-27T17:30:32.8474601Z Formatted command: . 'D:\a\1\s\HelloWorld.ps1'
10 2018-08-27T17:30:32.8474601Z ##[command]"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe"
11 2018-08-27T17:30:33.0990683Z Hello world
12 2018-08-27T17:30:33.2699756Z ##[section]Finishing: PowerShell Script
13
14
```

The 'Artifacts' tab is also visible, showing a list of successful steps:

- ✓ Prepare job · succeeded
- ✓ Initialize Agent · succeeded
- ✓ Initialize Job · succeeded
- ✓ Checkout · succeeded
- ✓ PowerShell Script · succeeded
- ✓ Publish Artifact: drop · succeeded
- ✓ Post-job: Checkout · succeeded
- ✓ Report build status · succeeded

4. Go to the build summary. On the Artifacts tab of the build, notice that the script is published as an artifact.

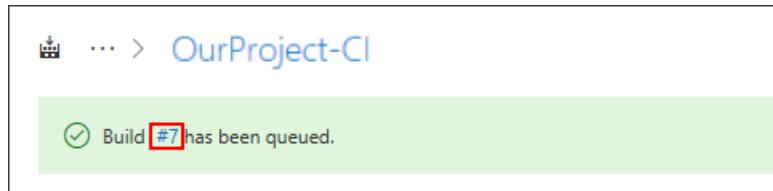
The screenshot shows the build summary for 'FabrikamFiber Web-CI 162'. The 'Summary' tab is selected. The 'Artifacts' section shows one artifact named 'drop' under the 'File container' category, which is highlighted with a red box.

1. Select **Save & queue**, and then select **Save & queue**.

2. On the dialog box, select **Save & queue** once more.

This queues a new build on the Microsoft-hosted agent.

3. You see a link to the new build on the top of the page.



Choose the link to watch the new build as it happens. Once the agent is allocated, you'll start seeing the live logs of the build. Notice that the PowerShell script is run as part of the build, and that "Hello world" is printed to the console.

- [TFS 2018.2](#)
- [TFS 2018 RTM](#)

A screenshot of the TFS 2018 Build summary page. The top navigation bar includes 'Builds', 'Releases', 'Packages', 'Library', 'Task Groups', 'Deployment Groups\*', and 'Explorer'. The 'Builds' tab is selected. On the left, a tree view shows 'Build 1722' expanded, with 'Build' selected. Under 'Build', several tasks are listed with green checkmarks: 'Initialize Agent', 'Initialize Job', 'Get Sources', 'PowerShell Script', 'Publish Artifact: drop', 'Post Job Cleanup', 'Finalize build', and 'Report build status'. To the right, the build details are shown. The title is 'Hello world / Build 1722 / Build'. A toggle switch is set to 'Build not retained'. Below the title are buttons for 'Edit build definition', 'Queue new build...', 'Download all logs as zip', and 'Release'. A large green banner at the top says 'Build succeeded'. Below it, a bar chart shows the build progress. The log output section displays the PowerShell commands and their execution results, including the 'Hello world' message.

4. Go to the build summary.

A screenshot of the TFS 2018 Build summary page. The interface is identical to the previous screenshot, but the build number 'Build 1722' in the tree view and the title 'Hello world / Build 1722 / Build' are highlighted with a red box. The rest of the page content, including the log output, remains the same.

5. On the **Artifacts** tab of the build, notice that the script is published as an artifact.

The screenshot shows the Azure Pipelines interface after a build has succeeded. The top navigation bar includes options like 'Edit build definition', 'Queue new build...', 'Download all logs as zip', and 'Release'. Below the navigation is a green header bar with the text 'Build succeeded'. Underneath, it says 'Build 1722' and 'Ran for 31 seconds (Hosted), completed 15.4 minutes ago'. There are tabs for 'Summary', 'Timeline', 'Artifacts' (which is highlighted with a red box), 'Code coverage\*', and 'Tests'. A search bar with 'Name ↑' and a dropdown menu showing 'drop' are also visible. At the bottom of the main area are 'Download' and 'Explore' buttons, with 'Explore' being highlighted with a red box. A modal window titled 'Artifacts Explorer' is open, showing a tree view with a folder 'drop' expanded, revealing a file 'HelloWorld.ps1'.

You can view a summary of all the builds or drill into the logs for each build at any time by navigating to the [Builds](#) tab in [Azure Pipelines](#). For each build, you can also view a list of commits that were built and the work items associated with each commit. You can also run tests in each build and analyze the test failures.

1. Select **Save & queue**, and then select **Save & queue**.
2. On the dialog box, select the **Queue** button.

This queues a new build on the agent. Once the agent is allocated, you'll start seeing the live logs of the build. Notice that the PowerShell script is run as part of the build, and that "Hello world" is printed to the console.

The screenshot shows the Azure Pipelines interface for Build 1722. The top navigation bar includes 'Builds', 'Releases', 'Packages', 'Library', 'Task Groups', 'Deployment Groups\*', and 'Explorer'. The 'Builds' tab is selected. The main area shows a list of build steps for 'Build 1722': 'Initialize Agent', 'Initialize Job', 'Get Sources', 'PowerShell Script', 'Publish Artifact: drop', 'Post Job Cleanup', 'Finalize build', and 'Report build status'. Each step has a green checkmark next to it. To the right of the build steps is a summary for 'Build 1722 / Build': 'Build succeeded', 'Build', 'Ran for 4 seconds (Hosted Agent), completed 1 seconds ago'. Below this are tabs for 'Console', 'Logs', 'Code coverage\*', and 'Tests'. The 'Console' tab is active, displaying the build logs. The logs show the execution of a PowerShell script named 'HelloWorld.ps1', which prints 'Hello world' to the console.

```

git checkout -b <new-branch-name>
HEAD is now at 0ab86c0... Updated HelloWorld.ps1
=====
Finishing: Get Sources
=====
Starting: PowerShell Script
=====
Task      : PowerShell
Description : Run a PowerShell script
Version   : 1.2.3
Author    : Microsoft Corporation
Help      : [More Information](https://go.microsoft.com/fwlink/?LinkId=613736)
=====
. 'd:\a\1\s\HelloWorld.ps1'
Hello world
  
```

3. Go to the build summary.

The screenshot shows the 'Builds' tab in the Azure DevOps interface. A build named 'Build 1722' is selected. The status bar at the bottom of the build card indicates 'Build succeeded'. Navigation links include 'Builds', 'Releases', 'Packages', 'Library', 'Task Groups', and 'Deployment Groups\*'. Action buttons for the selected build are 'Edit build definition' and 'Queue new build...'. A toggle switch is also present.

4. On the **Artifacts** tab of the build, notice that the script is published as an artifact.

The screenshot shows the 'Artifacts' tab for build 1722. The build status is 'Build succeeded'. The 'Artifacts' tab is selected. An 'Explore' button is highlighted with a red box. The 'drop' folder contains a file named 'HelloWorld.ps1'. Other tabs include 'Summary', 'Timeline', 'Code coverage\*', and 'Tests'. Action buttons include 'Edit build definition', 'Queue new build...', 'Download all logs as zip', and 'Release'.

You can view a summary of all the builds or drill into the logs for each build at any time by navigating to the **Builds** tab in **Build and Release**. For each build, you can also view a list of commits that were built and the work items associated with each commit. You can also run tests in each build and analyze the test failures.

## Add some variables and commit a change to your script

We'll pass some build variables to the script to make our pipeline a bit more interesting. Then we'll commit a change to a script and watch the CI pipeline run automatically to validate the change.

1. Edit your build pipeline.
2. On the **Tasks** tab, select the PowerShell script task.
3. Add these arguments.

The screenshot shows the Azure DevOps Pipelines interface for a 'FabrikamFiber Web-Cl' pipeline. The left sidebar lists various tasks: 'Get sources', 'Agent job 1', 'PowerShell Script' (which is selected and highlighted in blue), and 'Publish Artifact: drop'. The main pane displays the configuration for the selected 'PowerShell Script' task. The task is set to 'PowerShell' version '2.\*'. The 'Display name' is 'PowerShell Script'. The 'Type' is set to 'File Path'. The 'Script Path' is 'HelloWorld.ps1'. The 'Arguments' field contains the command: '-greeter "\$(Build.RequestedFor)" -trigger "\$(Build.Reason)"', which is highlighted with a red box.

- TFS 2018.2
- TFS 2018 RTM

The screenshot shows the Azure DevOps Pipelines interface for a 'Build process' pipeline. The left sidebar lists tasks: 'Get sources', 'Phase 1' (which is selected and highlighted in blue), and 'Publish Artifact: drop'. The main pane displays the configuration for the selected 'PowerShell Script' task. The task is set to 'PowerShell' version '1.\*'. The 'Display name' is 'PowerShell Script'. The 'Type' is set to 'File Path'. The 'Script Path' is 'HelloWorld.ps1'. The 'Arguments' field contains the command: '-greeter "\$(Build.RequestedFor)" -trigger "\$(Build.Reason)"', which is highlighted with a red box. Below the arguments, there are sections for 'Advanced' and 'Control Options'.

The screenshot shows the Azure DevOps build pipeline editor. On the left, under the 'Tasks' tab, there is a list of steps: 'Get sources' (branch: OurProject, branch: master), 'PowerShell Script' (selected, type: PowerShell, script path: HelloWorld.ps1), and 'Publish Artifact: drop' (Publish Build Artifacts). Below this is an 'Add Task' button. On the right, the 'PowerShell Script' task is detailed. It has a 'Display name' of 'PowerShell Script', a 'Type' of 'File Path' (selected), a 'Script Path' of 'HelloWorld.ps1', and an 'Arguments' section containing '-greeter "\$(Build.RequestedFor)" -trigger "\$(Build.Reason)"'. The 'Arguments' section is highlighted with a red box.

## Arguments

```
-greeter "$(Build.RequestedFor)" -trigger "$(Build.Reason)"
```

Finally, save the build pipeline.

Next you'll add the arguments to your script.

1. Go to your **Files** in Azure Repos (the **Code** hub in the previous navigation and TFS).
2. Select the **HelloWorld.ps1** file, and then **Edit** the file.
3. Change the script as follows:

```
Param(  
    [string]$greeter,  
    [string]$trigger  
)  
Write-Host "Hello world" from $greeter  
Write-Host Trigger: $trigger
```

4. **Commit** (save) the script.

Now you can see the results of your changes. Go to **Azure Pipelines** and select **Queued**. Notice under the **Queued or running** section that a build is automatically triggered by the change that you committed.

Now you can see the results of your changes. Go to the **Build and Release** page and select **Queued**. Notice under the **Queued or running** section that a build is automatically triggered by the change that you committed.

1. Select the new build that was created and view its log.
2. Notice that the person who changed the code has their name printed in the greeting message. You also see printed that this was a CI build.

```

1 2018-08-30T17:33:29.1723775Z ##[section]Starting: PowerShell Script
2 2018-08-30T17:33:29.1729508Z =====
3 2018-08-30T17:33:29.1729715Z Task : PowerShell
4 2018-08-30T17:33:29.1729878Z Description : Run a PowerShell script on Windows, macOS, or Linux.
5 2018-08-30T17:33:29.1730038Z Version : 2.136.0
6 2018-08-30T17:33:29.1730265Z Author : Microsoft Corporation
7 2018-08-30T17:33:29.1730373Z Help : [More Information](https://go.microsoft.com/fwlink/?LinkId=902752)
8 2018-08-30T17:33:29.1730567Z =====
9 2018-08-30T17:33:30.9668773Z Generating script.
10 2018-08-30T17:33:30.9756003Z Formatted command: . 'D:\a\1\s\HelloWorld.ps1' -greeter "Elijah Batkoski"
11 2018-08-30T17:33:31.0882200Z ##[command]"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -NoProfile -Command
12 2018-08-30T17:33:31.3101480Z Hello world from Elijah Batkoski
13 2018-08-30T17:33:31.3107687Z Trigger: Manual
14 2018-08-30T17:33:31.4515161Z ##[section]Finishing: PowerShell Script
15

```

**Build 1723**

- ✓ Build
- ✓ Initialize Agent
- ✓ Initialize Job
- ✓ Get Sources
- ✓ PowerShell Script
- ✗ Publish Artifact: drop
- ✓ Post Job Cleanup
- ✓ Finalize build
- ✓ Report build status

**Build succeeded**

**PowerShell Script**

Ran for 1 seconds (Hosted Agent), completed 74 seconds

**Logs**

```

1 2017-04-10T20:55:12.0502205Z ##[section]Starting: PowerShell Script
2 2017-04-10T20:55:12.0592196Z =====
3 2017-04-10T20:55:12.0602014Z Task : PowerShell
4 2017-04-10T20:55:12.0602014Z Description : Run a PowerShell script
5 2017-04-10T20:55:12.0602014Z Version : 1.2.3
6 2017-04-10T20:55:12.0602014Z Author : Microsoft Corporation
7 2017-04-10T20:55:12.0602014Z Help : [More Information](https://go.microsoft.com/fwlink/?LinkId=902752)
8 2017-04-10T20:55:12.0602014Z =====
9 2017-04-10T20:55:12.1292010Z ##[command]. 'd:\a\1\s\HelloWorld.ps1' -g
10 2017-04-10T20:55:12.8952061Z Hello world from Raisa Pokrovskaya
11 2017-04-10T20:55:12.8952061Z Trigger: IndividualCI
12 2017-04-10T20:55:12.9002073Z ##[section]Finishing: PowerShell Script

```

We just introduced the concept of build variables in these steps. We printed the value of a variable that is automatically predefined and initialized by the system. You can also define custom variables and use them either in arguments to your tasks, or as environment variables within your scripts. To learn more about variables, see [Build variables](#).

## You've got a build pipeline. What's next?

You've created a build pipeline that automatically builds and validates whatever code is checked in by your team. At this point, you can continue to the next section to learn about release pipelines. Or, if you prefer, you can [skip ahead](#) to create a build pipeline for your app.

## Create a release pipeline

Define the process for running the script in two stages.

1. Go to the **Pipelines** tab, and then select **Releases**.
2. Select the action to create a **New pipeline**. If a release pipeline is already created, select the plus sign (+) and then select **Create a release pipeline**.
3. Select the action to start with an **Empty job**.
4. Name the stage **QA**.
5. In the Artifacts panel, select + **Add** and specify a **Source (Build pipeline)**. Select **Add**.
6. Select the **Lightning bolt** to trigger continuous deployment and then enable the **Continuous deployment trigger** on the right.

Pipeline Tasks Variables Retention Options History

**Artifacts** | + Add

 \_FabrikamFiber Web-Cl

 Schedule not set

**Stages** | + Add ▾

 QA  
1 job, 0 task

**Continuous deployment trigger**  
Build: \_FabrikamFiber Web-Cl

 Enabled  
Creates a release every time a new build is available.

Build branch filters ⓘ  
No filters added.  
+ Add | ▾

**Pull request trigger**  
Build: \_FabrikamFiber Web-Cl

 Disabled

7. Select the **Tasks** tab and select your **QA** stage.
8. Select the plus sign (+) for the job to add a task to the job.
9. On the **Add tasks** dialog box, select **Utility**, locate the **PowerShell** task, and then select its **Add** button.
10. On the left side, select your new **PowerShell** script task.
11. For the **Script Path** argument, select the ... button to browse your artifacts and select the script you created.
12. Add these **Arguments**:

```
-greeter "$(Release.RequestedFor)" -trigger "$(Build.DefinitionName)"
```

13. On the **Pipeline** tab, select the **QA** stage and select **Clone**.

Pipeline Tasks Variables Retention Options History

**Artifacts** | + Add

 \_FabrikamFiber Web-Cl

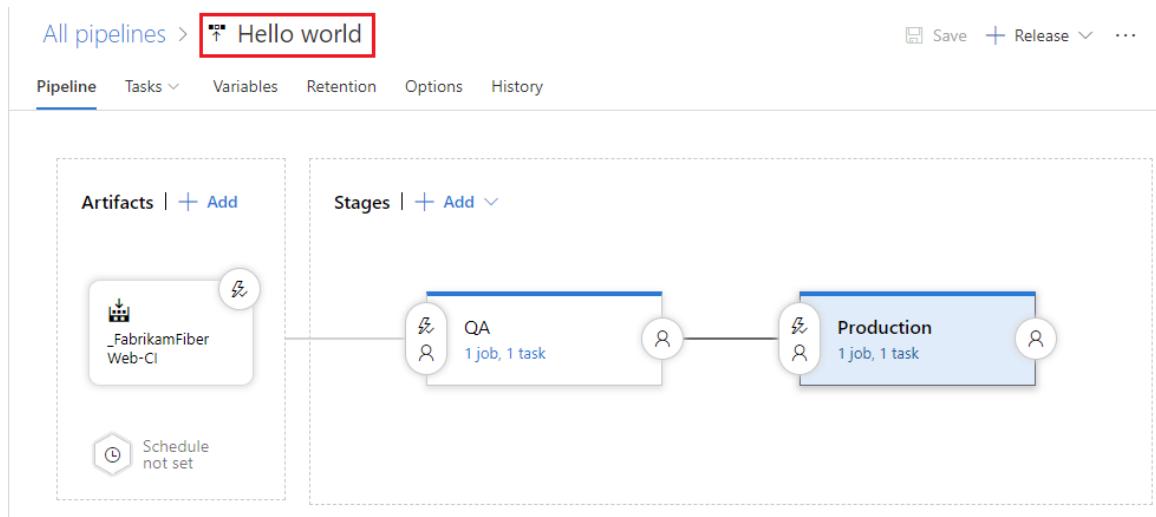
 Schedule not set

**Stages** | + Add ▾

 QA  
1 job, 1 task

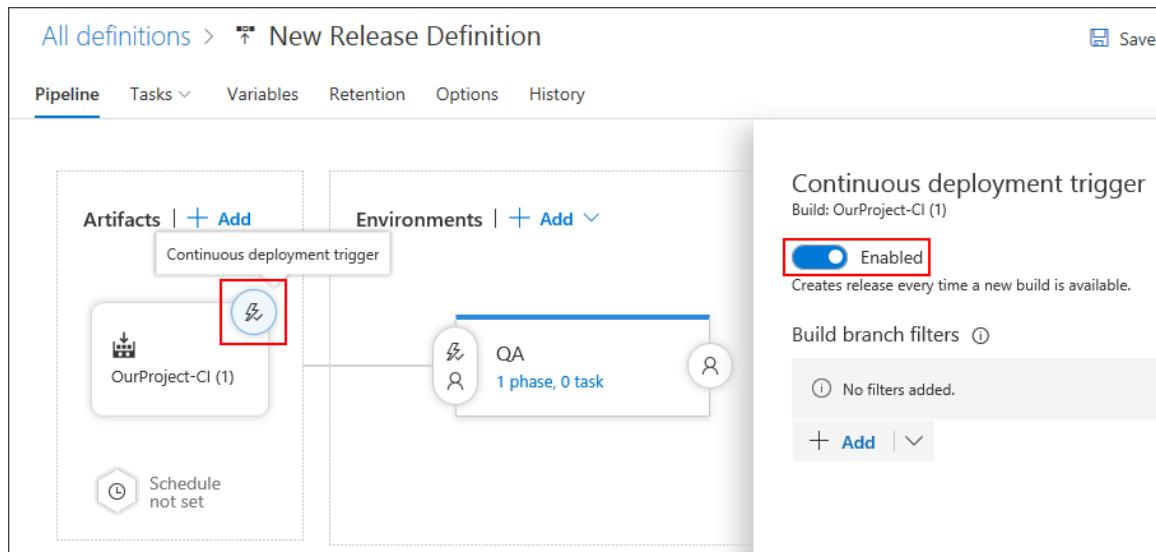
 +  Clone

14. Rename the cloned stage **Production**.
15. Rename the release pipeline **Hello world**.



16. Save the release pipeline.

1. Go to the **Build and Release** tab, and then select **Releases**.
2. Select the action to create a **New pipeline**. If a release pipeline is already created, select the plus sign (+) and then select **Create a release definition**.
3. Select the action to start with an **Empty definition**.
4. Name the stage **QA**.
5. In the Artifacts panel, select + **Add** and specify a **Source (Build pipeline)**. Select **Add**.
6. Select the **Lightning bolt** to trigger continuous deployment and then enable the **Continuous deployment trigger** on the right.
  - [TFS 2018.2](#)
  - [TFS 2018 RTM](#)



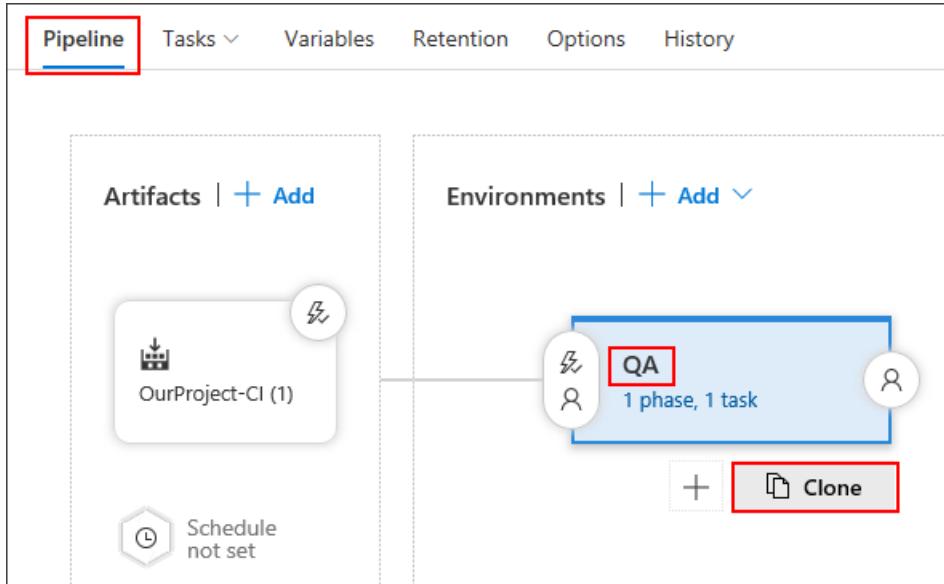
7. Select the **Tasks** tab and select your QA stage.
8. Select the plus sign (+) for the job to add a task to the job.
9. On the **Add tasks** dialog box, select **Utility**, locate the **PowerShell** task, and then select its **Add** button.
10. On the left side, select your new **PowerShell** script task.
11. For the **Script Path** argument, select the ... button to browse your artifacts and select the script you

created.

12. Add these Arguments:

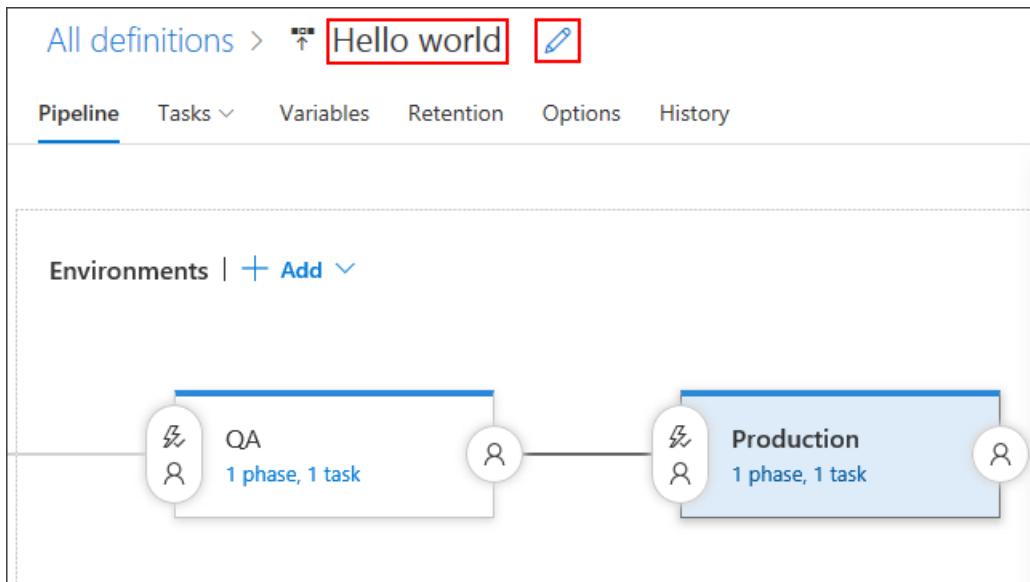
```
-greeter "$(Release.RequestedFor)" -trigger "$(Build.DefinitionName)"
```

13. On the Pipeline tab, select the QA stage and select Clone.



14. Rename the cloned stage Production.

15. Rename the release pipeline Hello world.



16. Save the release pipeline.

1. Go to **Azure Pipelines**, and then to the **Releases** tab.
2. Select the action to create a **New pipeline**.
3. On the dialog box, select the **Empty** template and select **Next**.
4. Make sure that your **Hello world** build pipeline that you created above is selected. Select **Continuous deployment**, and then select **Create**.
5. Select **Add tasks** in the stage.

6. On the **Task catalog** dialog box, select **Utility**, locate the **PowerShell** task, and then select its **Add** button. Select the **Close** button.
7. For the **Script Path** argument, select the ... button to browse your artifacts and select the script you created.
8. Add these **Arguments**:

```
-greeter "$(Release.RequestedFor)" -trigger "$(Build.DefinitionName)"
```

9. Rename the stage **QA**.

The screenshot shows the 'Environments' tab of a pipeline definition. A stage named 'QA' is selected. To the right, the 'PowerShell' task configuration is visible, showing the arguments: '-greeter "\$(Release.RequestedFor)" -trigger "\$(Build.DefinitionName)"'. The stage name 'QA' is highlighted with a red box.

10. Clone the QA stage.

The screenshot shows the 'Environments' tab of a pipeline definition. The 'QA' stage is selected and a context menu is open. The 'Clone environment' option is highlighted with a red box.

Leave **Automatically approve** and **Deploy automatically...** selected, and select **Create**.

11. Rename the new stage **Production**.
12. Rename the release pipeline **Hello world**.

The screenshot shows the Azure DevOps Releases interface. At the top, there are tabs for Builds, Releases, Packages, Library, Task Groups, Deployment Groups\*, and Explorer. The Releases tab is selected. In the center, there's a search bar for 'Search release definitions...' and a 'Release Definitions' section with a 'All release definitions' button. On the right, a release definition named 'Hello world' is being edited. The 'Environments' tab is active, showing two environments: 'QA' and 'Production'. Each environment has one task enabled. Below the environments, there's a 'Run on agent' section with a 'PowerShell Script' task. A red box highlights the 'Definition\*' input field containing 'Hello world'.

13. Save the release pipeline.

A release pipeline is a collection of stages to which the application build artifacts are deployed. It also defines the actual deployment pipeline for each stage, as well as how the artifacts are promoted from one stage to another.

Also, notice that we used some variables in our script arguments. In this case, we used [release variables](#) instead of the build variables we used for the build pipeline.

## Deploy a release

Run the script in each stage.

1. Create a new release.

The screenshot shows the Azure DevOps Pipeline interface for the 'Hello world' release. At the top, there are tabs for Pipeline, Tasks, Variables, Retention, Options, History, and a 'Save' button. To the right, there are buttons for '+ Release' (highlighted with a red box) and '+ Create a draft release'. The main area shows the pipeline structure. On the left, under 'Artifacts', there's a single artifact named '\_FabrikamFiber Web-Cl'. On the right, under 'Stages', there are two stages: 'QA' and 'Production'. The 'QA' stage has one job and one task. The 'Production' stage also has one job and one task. Arrows indicate the flow from QA to Production.

When **Create new release** appears, select **Create**.

2. Open the release that you created.

## All pipelines > Hello world

Release **Release1** has been created

Pipeline Tasks Variables Retention Options History

- View the logs to get real-time data about the release.

Hello world > Release-1

Pipeline Variables History | + Deploy | Cancel Refresh Release (old view)

**Release**

Manually triggered  
by Elijah Batkoski  
8/27/2018 6:16 PM

Artifacts  
 \_FabrikamFiber Web-Cl  
118  
master

**Stages**

QA  
✓ Succeeded  
on 8/27/2018 6:16 PM

Redeploy Logs

Production  
⌚ Not deployed

- Create a new release.

All definitions > Hello World

Save + Release ...

+ Create release + Create draft release

Pipeline Tasks Variables Retention Options History

**Artifacts** | + Add

OurProject-Cl (2)

Schedule not set

**Environments** | + Add

QA 1 phase, 1 task

Production 1 phase, 1 task

When **Create new release** appears, select **Create** (TFS 2018.2) or **Queue** (TFS 2018 RTM).

- Open the release that you created.

All definitions > Hello World

Release **Release-2** has been created

Pipeline Tasks Variables Retention Options History

6. View the logs to get real-time data about the release.

Hello World / Release-2

Summary Environments Artifacts Variables General Commits Work items Tests **Logs**

Deploy Save Abandon Download all logs as zip

Step	Action	Log Content
QA	...	Agent queue: Hosted VS2017   Starting: Initialize Job ***** Prepare release directory. ReleaseId=1, TeamProjectId=eb7 Release folder: d:\a\r1\qa Environment variables available
Pre-deployment approval	...	
Agent phase	...	
Production	...	

7. Create a new release.

Definition: Hello world | Releases

**Environments** Artifacts Variables Triggers General Retention

Save + Release

+ Add environment Create Release

**QA**  
1 / 1 tasks enabled

Create Draft Release Agent PowerShell Script PowerShell

8. Open the release that you created.

Hello world | Edit

Overview **Releases** Deleted

Release **Release-2** has been created.

Title Environments

Release-2

9. View the logs to get real-time data about the release.

The screenshot shows the 'Logs' tab in the Azure DevOps interface. At the top, there are tabs for Summary, Environments, Artifacts, Variables, General, Commits, Work items, Tests, and Logs, with 'Logs' being the active tab and highlighted with a red border. Below the tabs is a toolbar with icons for Refresh, Deploy, Save, Abandon, and Download all logs as zip. The main area displays deployment logs for a step named 'Run on agent'. The log output includes:

```

Agent: Hosted Agent
Starting: Initialize Job
*****
Prepare release directory.
ReleaseId=1, TeamProjectId=eb7...
Release folder: d:\a\r1\a
Environment variables available

```

You can track the progress of each release to see if it has been deployed to all the stages. You can track the commits that are part of each release, the associated work items, and the results of any test runs that you've added to the release pipeline.

## Change your code and watch it automatically deploy to production

We'll make one more change to the script. This time it will automatically build and then get deployed all the way to the production stage.

1. Go to the **Code hub**, **Files** tab, edit the **HelloWorld.ps1** file, and change it as follows:

```

Param(
[string]$greeter,
[string]$trigger
)
Write-Host "Hello world" from $greeter
Write-Host Trigger: $trigger
Write-Host "Now that you've got CI/CD, you can automatically deploy your app every time your team checks in code."

```

2. **Commit** (save) the script.
3. Select the **Builds** tab to see the build queued and run.
4. After the build is completed, select the **Releases** tab, open the new release, and then go to the **Logs**.

Your new code automatically is deployed in the **QA** stage, and then in the **Production** stage.

The screenshot shows the 'PowerShell Script' logs for a deployment. The logs are timestamped and show the execution of a PowerShell script. A specific line of text is highlighted with a red box:

```

1  2018-08-27T18:31:42.7222014Z ##[section]Starting: PowerShell Script
2  2018-08-27T18:31:42.7228660Z =====
3  2018-08-27T18:31:42.7228871Z Task      : PowerShell
4  2018-08-27T18:31:42.7229057Z Description : Run a PowerShell script on Windows, macOS, or Linux.
5  2018-08-27T18:31:42.7229253Z Version   : 2.136.0
6  2018-08-27T18:31:42.7229424Z Author    : Microsoft Corporation
7  2018-08-27T18:31:42.7229609Z Help     : [More Information](https://go.microsoft.com/fwlink/?LinkId=613736)
8  2018-08-27T18:31:42.7229827Z =====
9  2018-08-27T18:31:45.5150962Z Generating script.
10 2018-08-27T18:31:45.5196754Z Formatted command: . 'D:\a\r1\a\_FabrikamFiber Web-CI\drop\HelloWorld.ps1' -greeter "Elijah Batkoski"
11 2018-08-27T18:31:45.6901750Z ##[command]"C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -NoLogo -NoProfile -NonInteractive
12 2018-08-27T18:31:45.9197804Z Hello world from Elijah Batkoski
13 2018-08-27T18:31:45.9203491Z Trigger: FabrikamFiber Web-CI
14 2018-08-27T18:31:45.9208413Z Now that you've got CI/CD, you can automatically deploy your app every time your team checks in code.
15 2018-08-27T18:31:46.0566737Z ##[section]Finishing: PowerShell Script
16 |

```

Hello World / Release-2

Summary Environments Artifacts Variables General Commits Work items Tests Logs History View All Details pane On

Abandon | Download all logs as zip | Send Email

Step	Action	Logs
> QA	...	Agent queue: Hosted VS2017   Agent: Hosted Agent 1 2018-04-25T14:53:54.9742188Z ##[section]Starting: PowerShell Script 2 2018-04-25T14:53:54.974641Z ===== 3 2018-04-25T14:53:54.9746591Z Task : PowerShell 4 2018-04-25T14:53:54.9746697Z Description : Run a PowerShell scri 5 2018-04-25T14:53:54.9746796Z Version : 1.2.3 6 2018-04-25T14:53:54.9746897Z Author : Microsoft Corporation 7 2018-04-25T14:53:54.9747808Z Help : [More Information](ht 8 2018-04-25T14:53:54.9747921Z ===== 9 2018-04-25T14:53:55.0138848Z ##[command]. 'D:\a\r1\a\OurProject-C 10 2018-04-25T14:53:57.2485315Z Hello world from Raisa Pokrovskaya 11 2018-04-25T14:53:57.2486035Z Trigger: Hello world 12 2018-04-25T14:53:57.2486230Z Now that you've got CI/CD, you can a 13 2018-04-25T14:53:57.3058801Z ##[section]Finishing: PowerShell Scr 14
Production	...	
Pre-deployment approval	...	
Agent phase	...	
Initialize Agent	...	
Initialize Job	...	
Download artifact - OurProject-CI (2)	...	
PowerShell Script	...	
Post-deployment approval	...	

Hello world / Release-7

Summary Environments Artifacts Variables General Commits Work items Tests Logs History

Deploy | Save | Abandon | Download all logs as zip | Send Email

Step	Action	Logs
> QA	...	Agent: Hosted Agent 1 2017-04-11T12:54:58.5891186Z ##[section]Starting: PowerShell Script 2 2017-04-11T12:54:58.6047463Z ===== 3 2017-04-11T12:54:58.6047463Z Task : PowerShell 4 2017-04-11T12:54:58.6047463Z Description : Run a PowerShell script 5 2017-04-11T12:54:58.6047463Z Version : 1.2.3 6 2017-04-11T12:54:58.6047463Z Author : Microsoft Corporation 7 2017-04-11T12:54:58.6047463Z Help : [More Information](https://go.micro 8 2017-04-11T12:54:58.6047463Z ===== 9 2017-04-11T12:54:58.6672458Z ##[command]. 'd:\a\r1\Hello_world\drop\HelloWor 10 2017-04-11T12:54:59.3703946Z Hello world from Raisa Pokrovskaya 11 2017-04-11T12:54:59.3703946Z Trigger: Hello world 12 2017-04-11T12:54:59.3703946Z Now that you've got CI/CD, you can automatically d 13 2017-04-11T12:54:59.4641245Z ##[section]Finishing: PowerShell Script 14
Production	...	
Pre-deployment approval	...	
Run on agent	...	
Initialize Agent	...	
Initialize Job	...	
Download Artifacts	...	
PowerShell Script	...	
Post-deployment approval	...	

In many cases, you probably would want to edit the release pipeline so that the production deployment happens only after some testing and approvals are in place. See [Approvals and gates overview](#).

## Next steps

You've just learned how to create your first pipeline in Azure. Learn more about configuring pipelines in the language of your choice:

- [.NET Core](#)
- [Go](#)
- [Java](#)
- [Node.js](#)
- [Python](#)
- [Containers](#)

Or, you can proceed to [customize the pipeline](#) you just created.

To run your pipeline in a container, see [Container jobs](#).

For details about building GitHub repositories, see [Build GitHub repositories](#).

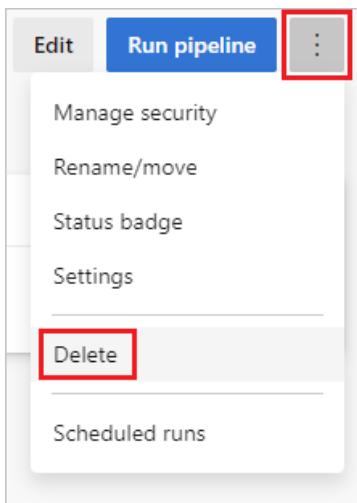
To learn what else you can do in YAML pipelines, see [YAML schema reference](#).

## Clean up

If you created any test pipelines, they are easy to delete when you are done with them.

- [Browser](#)
- [Azure DevOps CLI](#)

To delete a pipeline, navigate to the summary page for that pipeline, and choose **Delete** from the ... menu at the top-right of the page. Type the name of the pipeline to confirm, and choose **Delete**.



You've learned the basics of creating and running a pipeline. Now you're ready to configure your build pipeline for the programming language you're using. Go ahead and create a new build pipeline, and this time, use one of the following templates.

LANGUAGE	TEMPLATE TO USE
.NET	ASP.NET
.NET Core	ASP.NET Core
C++	.NET Desktop
Go	Go
Java	Gradle
JavaScript	Node.js
Xcode	Xcode

## FAQ

**Where can I read articles about DevOps and CI/CD?**

[What is Continuous Integration?](#)

[What is Continuous Delivery?](#)

[What is DevOps?](#)

## What kinds of version control can I use?

When you're ready to get going with CI/CD for your app, you can use the version control system of your choice:

- Clients

- Visual Studio Code for Windows, macOS, and Linux
- Visual Studio with Git for Windows or Visual Studio for Mac
- Eclipse
- Xcode
- IntelliJ
- Command line

- Services

- Azure Pipelines
- Git service providers such as GitHub and Bitbucket Cloud
- Subversion

- Clients

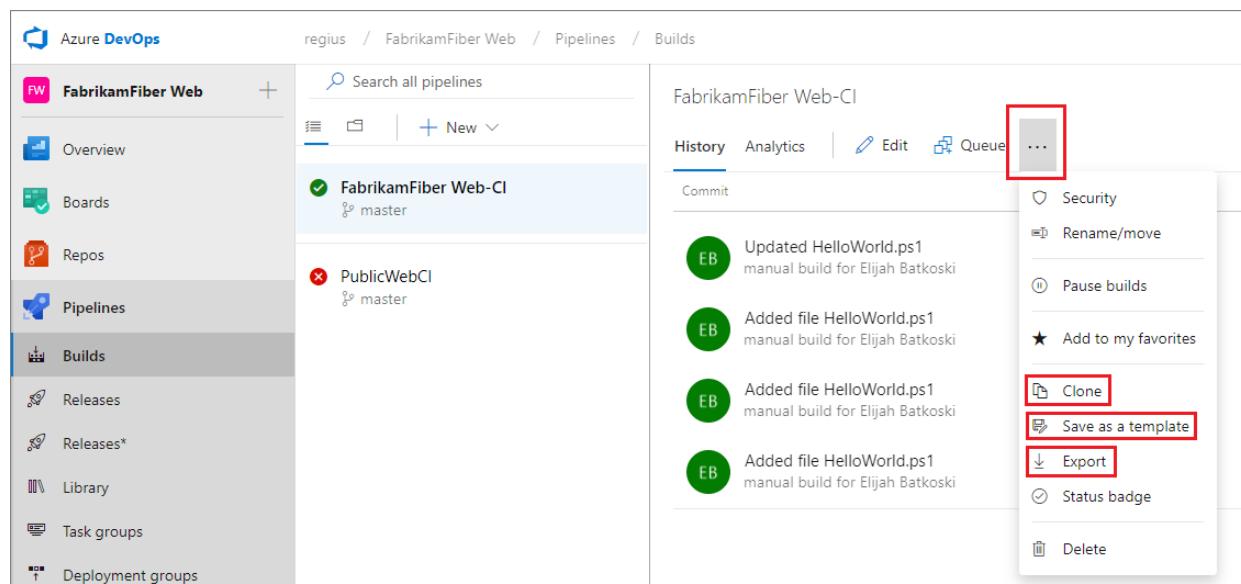
- Visual Studio Code for Windows, macOS, and Linux
- Visual Studio with Git for Windows or Visual Studio for Mac
- Visual Studio with TFVC
- Eclipse
- Xcode
- IntelliJ
- Command line

- Services

- Azure Pipelines
- Git service providers such as GitHub and Bitbucket Cloud
- Subversion

## How do I replicate a pipeline?

If your pipeline has a pattern that you want to replicate in other pipelines, clone it, export it, or save it as a template.



The screenshot shows the 'Build Definitions' section of the Azure DevOps interface. At the top, there are tabs for 'Builds', 'Releases', 'Packages', 'Library', 'Task Groups', and 'Deployment Groups\*'. Below these are buttons for 'Search all definitions', '+ New', 'Mine', 'All Definitions' (which is selected), 'Queued', and 'XAML'. A search bar and a 'Default branch summary' button are also present. The main area lists several pipeline definitions under a folder named 'HelloWorld-Cl'. One definition, 'OurProject-Cl', is selected and has a context menu open. The menu includes options like 'Queue new build...', 'Move definition', 'View definition summary', 'Edit...', 'Add to my favorites', 'Add to team favorites', 'Clone...' (which is highlighted with a red box), 'Export' (also highlighted with a red box), 'Rename...', 'Save as a template...' (highlighted with a red box), 'Delete definition', 'Security...', and 'Add to dashboard'.

After you clone a pipeline, you can make changes and then save it.

After you export a pipeline, you can import it from the All pipelines tab.

After you create a template, your team members can use it to follow the pattern in new pipelines.

#### TIP

If you're using the **New Build Editor**, then your custom templates are shown at the bottom of the list.

### How do I work with drafts?

If you're editing a build pipeline and you want to test some changes that are not yet ready for production, you can save it as a draft.

The screenshot shows the 'FabrikamFiber Web-ClTest' pipeline editor. At the top, there are tabs for 'Tasks', 'Variables', 'Triggers', 'Options', 'Retention', 'History', 'Save & queue' (with a dropdown arrow), 'Discard', and 'Summary'. The 'Save & queue' dropdown menu is open, showing 'Save & queue', 'Save', and 'Save as draft' (which is highlighted with a red box). On the right, there's a 'Name \*' field with 'FabrikamFiber W' typed into it. The pipeline itself shows a single step: 'Get sources' from 'SmartHotel360' on the 'master' branch.

The screenshot shows the 'Artifacts' tab of a pipeline configuration. A context menu is open over the 'Save & queue' button, with the 'Save as draft' option highlighted by a red box. Other options in the menu include 'Save' and 'Save & queue'. To the right of the menu, there's a panel for naming the artifact, with 'Name \*' set to 'Artifacts'.

You can edit and test your draft as needed.

The screenshot shows the pipeline list. On the left, under 'All build pipelines', the 'FabrikamFiber Web-CITest' pipeline is selected and highlighted with a red box. On the right, the pipeline details page for 'FabrikamFiber Web-CITest' is shown, with the 'Edit' button highlighted by a red box.

The screenshot shows the 'Build Definitions' list. Under the 'Artifacts' section, the 'Artifacts' definition is selected and highlighted with a red box. A context menu is open over the three-dot ellipsis button, with the 'Edit...' option highlighted by a red box. Other options in the menu include 'Queue new build...', 'Move definition', and 'View definition summary'.

When you're ready you can publish the draft to merge the changes into your build pipeline.

The screenshot shows the pipeline configuration page for 'FabrikamFiber Web-CITest'. At the top, the 'Publish draft' button is highlighted by a red box. Below it, the pipeline tasks are listed, including 'Get sources' from 'SmartHotel360' on the 'master' branch. To the right, there's a panel for naming the pipeline, with 'Name \*' set to 'FabrikamFiber Web-CITest'.

The screenshot shows the top navigation bar of the Azure DevOps interface. The tabs include 'Builds', 'Releases', 'Packages', 'Library', 'Task Groups', 'Deployment Groups\*', 'Explorer', 'Artifacts' (selected), 'Save draft & queue', 'Publish draft' (highlighted with a red box), 'Discard', 'Queue', and '...'. The 'Publish draft' button is located in the top right corner of the main content area.

Or, if you decide to discard the draft, you can delete it from the All Pipeline tab shown above.

### How can I delete a pipeline?

To delete a pipeline, navigate to the summary page for that pipeline, and choose **Delete** from the ... menu in the top-right of the page. Type the name of the pipeline to confirm, and choose **Delete**.

### What else can I do when I queue a build?

You can queue builds [automatically](#) or manually.

When you manually queue a build, you can, for a single run of the build:

- Specify the [pool](#) into which the build goes.
- Add and modify some [variables](#).
- Add [demands](#).
- In a Git repository
  - Build a [branch](#) or a [tag](#).
  - Build a [commit](#).
- In a TFVC repository
  - Specify the source version as a [label](#) or [changeset](#).
  - Run a private build of a [shelveset](#). (You can use this option on either a [Microsoft-hosted agent](#) or a [self-hosted agent](#).)

You can queue builds [automatically](#) or manually.

When you manually queue a build, you can, for a single run of the build:

- Specify the [pool](#) into which the build goes.
- Add and modify some [variables](#).
- Add [demands](#).
- In a Git repository
  - Build a [branch](#) or a [tag](#).
  - Build a [commit](#).

### Where can I learn more about build pipeline settings?

To learn more about build pipeline settings, see:

- [Getting sources](#)
- [Tasks](#)
- [Variables](#)
- [Triggers](#)
- [Options](#)
- [Retention](#)

- [History](#)

To learn more about build pipeline settings, see:

- [Getting sources](#)
- [Tasks](#)
- [Variables](#)
- [Triggers](#)
- [Retention](#)
- [History](#)

## How do I programmatically create a build pipeline?

[REST API Reference: Create a build pipeline](#)

### NOTE

You can also manage builds and build pipelines from the command line or scripts using the [Azure Pipelines CLI](#).

## Can I use a single command at the command line to run multiple pipelines in Azure DevOps Services?

Currently, the Azure CLI and Azure APIs don't offer commands that run multiple pipelines from the command line. You can use [Azure CLI commands](#) to list all pipelines and definitions and provide a *single* release or build ID as a parameter. All commands are designed to work for independent runs of independent pipelines, and they require unique ID requests that allow only one, unique value. To learn about pipeline triggers, see [Specify events that trigger pipelines](#).

# Plan and track work

4/12/2021 • 21 minutes to read • [Edit Online](#)

Azure Boards | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

You track your work by creating work items. This article walks you through creating issues and tasks using a Kanban board for the Basic process, or creating user stories and tasks using for the Agile process.

Choose one of the following four system processes—**Agile**, **Basic**, **Scrum**, or **Capability Maturity Model Integration (CMMI)**—for guidance depending on what process was selected for your project. For an overview of each of these processes, see [Choose a process](#).

## NOTE

The Basic process is available when you add a project to Azure DevOps Services or [Azure DevOps Server 2019 Update 1](#). For earlier on-premises deployments, choose Agile, Scrum, or CMMI process.

- [Agile process](#)
- [Basic process](#)
- [Scrum process](#)
- [CMMI process](#)

The Agile process provides several work item types—for example, user stories, tasks, bugs, features, and epics among others—to plan and track work. We recommend you start by adding user stories. If you need to group them into a hierarchy, you can define features. If you want to track additional details of work, you can add tasks to a user story.

WORK ITEM TYPES

BACKLOG HIERARCHY

WORK ITEM TYPES	BACKLOG HIERARCHY																																				
<p><b>Portfolio backlog</b></p> <pre> graph TD     Portfolio[Portfolio backlog] --&gt; Epic[Epic]     Portfolio --&gt; Feature1[Feature]     Feature1 --&gt; UserStory1[User Story]     UserStory1 --&gt; Task1[Task]     Product[Product backlog] --&gt; Feature1     Product --&gt; UserStory1     UserStory1 --&gt; Task1     Issues[Issue tracking] --&gt; Issue1[Issue]     Configurable[Configurable] --&gt; Bug1[Bug]     Configurable --&gt; Task2[Task]   </pre> <p><b>Product backlog</b></p> <p><b>Issue tracking</b></p> <p><b>Configurable</b></p>	<table> <thead> <tr> <th>State</th> <th>Title</th> </tr> </thead> <tbody> <tr> <td>● New</td> <td>Web site updates</td> </tr> <tr> <td>● New</td> <td>Web pages</td> </tr> <tr> <td>● New</td> <td>Cancel order form</td> </tr> <tr> <td>● Active</td> <td>Hello World web site</td> </tr> <tr> <td>● Active</td> <td>Change background color</td> </tr> <tr> <td>● New</td> <td>Change page layout</td> </tr> <tr> <td>● New</td> <td>Develop about page</td> </tr> <tr> <td>● New</td> <td>Slow response on form</td> </tr> <tr> <td>● Active</td> <td>Secure Sign-in</td> </tr> <tr> <td>● New</td> <td>Improve User Experience</td> </tr> <tr> <td>● Active</td> <td>Emoticon feedback enabled in client</td> </tr> <tr> <td>● New</td> <td>Service status</td> </tr> <tr> <td>● New</td> <td>Service support</td> </tr> <tr> <td>● New</td> <td>Lookup service outages</td> </tr> <tr> <td>● Resolved</td> <td>Canadian addresses don't display</td> </tr> <tr> <td>● Active</td> <td>Voicemail hang issue</td> </tr> <tr> <td>● Active</td> <td>Check issues with permissions</td> </tr> </tbody> </table>	State	Title	● New	Web site updates	● New	Web pages	● New	Cancel order form	● Active	Hello World web site	● Active	Change background color	● New	Change page layout	● New	Develop about page	● New	Slow response on form	● Active	Secure Sign-in	● New	Improve User Experience	● Active	Emoticon feedback enabled in client	● New	Service status	● New	Service support	● New	Lookup service outages	● Resolved	Canadian addresses don't display	● Active	Voicemail hang issue	● Active	Check issues with permissions
State	Title																																				
● New	Web site updates																																				
● New	Web pages																																				
● New	Cancel order form																																				
● Active	Hello World web site																																				
● Active	Change background color																																				
● New	Change page layout																																				
● New	Develop about page																																				
● New	Slow response on form																																				
● Active	Secure Sign-in																																				
● New	Improve User Experience																																				
● Active	Emoticon feedback enabled in client																																				
● New	Service status																																				
● New	Service support																																				
● New	Lookup service outages																																				
● Resolved	Canadian addresses don't display																																				
● Active	Voicemail hang issue																																				
● Active	Check issues with permissions																																				

Within each work item form, you can describe the work to be done, assign work to project contributors, track status, and collaborate with others through the Discussion section.

Here we show how to add user stories and child tasks from the web portal and add details to those work items .

## Prerequisites

- After you connect to a project, you can add work items. If you don't have a project yet, [create one in Azure DevOps](#).
- To add work items to a board, and use all other board features, you must be granted **Basic** access and have been added as a member of the Contributors or Project Administrators group.
- If you have been granted **Stakeholder** access for a private project and have been added as a member of the Contributors or Project Administrators group, you can view boards, open and modify work items, and add child tasks to a checklist. However, you can't reorder or reparent a backlog item using drag-and-drop, nor update a field on a card.
- If you have been granted **Stakeholder** access for a public project, and have been added as a member of the Contributors or Project Administrators group, you have full access to all Boards features.
- After you connect to a project, you can add work items. If you don't have a project yet, [create one in Azure DevOps](#).
- To add work items to a board, and use all other board features, you must be granted **Basic** access and have been added as a member of the Contributors or Project Administrators group.
- If you have been granted **Stakeholder** access and have been added as a member of the Contributors or Project Administrators group, you can view boards, open and modify work items, and add child tasks to a checklist. However, you can't reorder or reparent a backlog item using drag-and-drop, nor update a field on a card.

#### NOTE

The ability to drag-and-drop cards to different columns requires installation of Azure DevOps Server 2020.1 update. To learn more, see [Azure DevOps Server 2020 Update 1 RC1 Release Notes, Boards](#).

- After you connect to a project, you can add work items. If you don't have a project yet, [create one in Azure DevOps](#).
- To add work items to a board, and use all other board features, you must be granted **Basic** access and have been added as a member of the Contributors or Project Administrators group.
- If you have been granted **Stakeholder** access for a private project and have been added as a member of the Contributors or Project Administrators group, you can view boards, open and modify work items, and add child tasks to a checklist. However, you can't update the status of a backlog item or reorder or reparent a backlog item using drag-and-drop, nor update a field on a card.
- If you have been granted **Stakeholder** access for a public project, and have been added as a member of the Contributors or Project Administrators group, you have full access to all Boards features.

For details, see [Default permissions and access for Azure Boards](#)

#### NOTE

The images shown in this article correspond to the latest version of Azure Boards. While they may differ from those shown in earlier, on-premises versions of Azure DevOps, they are similar in the functions described unless otherwise noted.

## Open the Kanban board

A Kanban board is provisioned with the addition of each project and each team. You can only create or add Kanban boards to a project by adding another team. To learn more, see [About teams and Agile tools](#).

- [Agile process](#)
- [Basic process](#)
- [Scrum process](#)
- [CMMI process](#)

The User Stories Kanban board is the best tool for quickly adding user stories and child tasks. To open, choose **Boards > Boards**.

The screenshot shows the Azure DevOps interface with the following elements:

- Header:** Azure DevOps, fabrikam / Fabrikam Fiber (highlighted with red box 1).
- Sidebar:** Boards (highlighted with red box 2) is selected. Other options include Overview, Work Items, Boards, Backlogs, and Sprints.
- Team Selection:** Fabrikam Fiber Team (highlighted with red box 3).
- Board Options:** View as backlog, Stories.
- Columns:** Backlog, Active, Resolved.
- Backlog Column:** New item button, Secure sign-in (Raisa Pokrovskaya, Priority 1), Hello World web site (Jamal Hartnett, Priority 2, 0/4 completed).
- Active Column:** Cancel form (Christie Churchill, Priority 3).
- Resolved Column:** No items listed.

The Features Kanban board is the best tool for quickly adding features and user stories that are children of those features. To open the Features board from the Stories board, choose **Features** from the board selector.

A screenshot of the TFS Stories board interface. At the top, there's a header with the team name "Fabrikam Fiber Team" and some status indicators. Below the header, there are three columns: "Backlog", "Active", and "Resolved". In the "Active" column, there's a card for "Hello World web site" assigned to "Jamal Hartnett" with a progress of "5". In the "Resolved" column, there are cards for "Cancel order form" and "GPS locator". On the right side, there's a "Board selector" dropdown menu with options: "Stories" (selected), "Epics", "Features" (highlighted with a red box and arrow), and "Stories".

## Add work items to your board

- Agile process
- Basic process
- Scrum process
- CMMI process

1. From the Stories board, choose **New item** and start adding those stories you want to track.

A screenshot of the TFS Stories board interface, specifically the "New" section. It shows a "New item" button highlighted with a red box. Below it is a "Change initial view" link.

2. Enter return and the system assigns a work item ID to the user story.

A screenshot of the TFS Stories board interface, showing a user story card. The card has a blue icon, the number "1", the title "Change initial view", and the state "New".

3. To track the work you want to manage, add as many user stories that you need.

## Add details to a board item

Choose the issue or user story title to open it. Change one or more field values, add a description, or make a note in the **Discussion** section. You can also choose the **Attachments** tab and drag-and-drop a file to share the file with others.

### NOTE

The **Discussion** section is available with TFS 2017.2 and later versions.

- Agile process
- Basic process
- Scrum process
- CMMI process

For example, here we assign the story to Raisa Pokrovskaya and we add a discussion note, at-mentioning Raisa.

The screenshot shows a 'User Story' card with the following details:

- Title:** USER STORY 1\*
- Assignee:** Raisa Pokrovskaya
- Comments:** 0 comments
- Save & Close:** Save & Close button
- Follow:** Follow button
- Details:** Tab selected for 'Details'.
- State:** New
- Reason:** New
- Area:** Fabrikam Fiber
- Iteration:** Fabrikam Fiber
- Description:** Switch initial view to the updated design.
- Acceptance Criteria:** Click to add Acceptance Criteria
- Planning:**
  - Story Points
  - Priority: 2
  - Risk
- Classification:**
  - Value area: Business
- Development:**
- Related Work:**
- Discussion:** A comment from Raisa Pokrovskaya: '@Raisa Pokrovskaya - Can you make this happen in the next week?' with rich text editor controls.

Choose Save & Close when done.

## Field descriptions

### Field

### Usage

### Title

Enter a description of 255 characters or less. You can always modify the title later.

### Assigned To

Assign the work item to the team member responsible for performing the work. Depending on the context you are working in, the drop-down menu lists only team members or contributors to the project.

#### NOTE

You can only assign work to a single user. If you need to assign work to more than one user, add a work item for each user and distinguish the work to be done by title and description. The Assigned To field only accepts user accounts that have been [added to a project or team](#).

## State

When the work item is created, the State defaults to the first state in the workflow. As work progresses, update it to reflect the current status.

---

## Reason

Use the default first. Update it when you change state as need. Each State is associated with a default reason.

---

## Area (Path)

Choose the area path associated with the product or team, or leave blank until assigned during a planning meeting. To change the dropdown list of areas, see [Define area paths and assign to a team](#).

---

## Iteration (Path)

Choose the sprint or iteration in which the work is to be completed, or leave it blank and assign it later during a planning meeting. To change the drop-down list of iterations, see [Define iteration paths and configure team iterations](#).

---

## Description

Provide enough detail to create shared understanding of scope and support estimation efforts. Focus on the user, what they want to accomplish, and why. Don't describe how to develop the product. Do provide sufficient details so that your team can write tasks and test cases to implement the item.

---

## Acceptance Criteria

Provide the criteria to be met before the work item can be closed. Define what "Done" means by describing the criteria for the team to use to verify whether the backlog item or bug fix is fully implemented. Before work begins, describe the [criteria for customer acceptance](#) as clearly as possible. Have conversations between the team and customers to determine the acceptance criteria. These criteria help ensure a common understanding within the team to meet customers' expectations. Also, this information provides the basis for acceptance testing.

---

## Priority

A subjective rating of the issue or task it relates to the business. You can specify the following values:

- **1:** Product cannot ship without the successful resolution of the work item, and it should be addressed as soon as possible.
- **2:** Product cannot ship without the successful resolution of the work item, but it does not need to be addressed immediately.
- **3:** Resolution of the work item is optional based on resources, time, and risk.
- **4:** Resolution of the work item is not required.

---

## Value Area

A subjective rating of the issue or task it relates to the business. You can specify the following values:

- **Architectural:** Technical services to implement business features that deliver solution .
- **Business:** Services that fulfill customers or stakeholder needs that directly deliver customer value to support the business (Default).

---

## Effort, Story Points, Size

Provide a relative estimate of the amount of work required to complete an issue. Most Agile methods

recommend that you set estimates for backlog items based on relative size of work. Such methods include powers of 2 (1, 2, 4, 8) and the Fibonacci sequence (1, 2, 3, 5, 8, etc.). Use any numeric unit of measurement your team prefers.

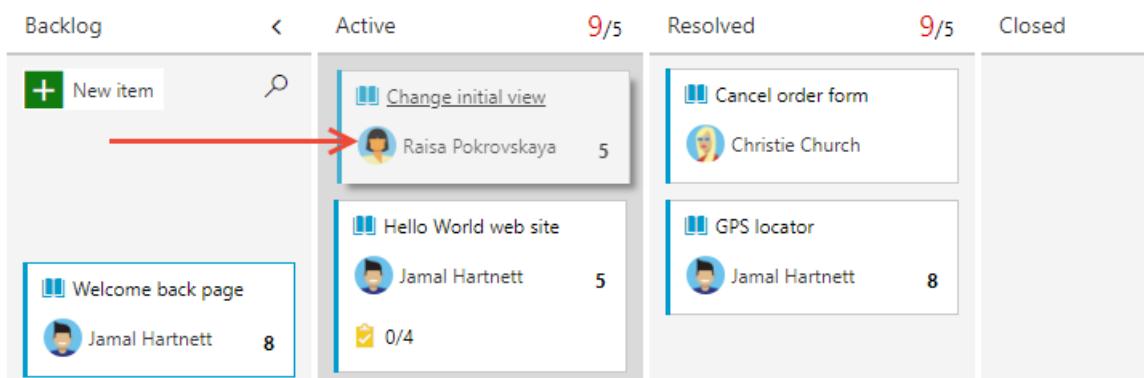
The estimates you set are used to calculate [team velocity](#) and [forecast sprints](#).

## Update status

The State field tracks the status of a work item. With the Kanban board, you can quickly update the status of backlog items by dragging and dropping them to a different column. This feature requires that you have Basic access or higher.

- [Agile process](#)
- [Basic process](#)
- [Scrum process](#)
- [CMMI process](#)

As work starts, drag the user story card from the **Backlog** column to the **Active** column. Once work is ready for review, move to the **Resolved** column. After it is reviewed and accepted, move to the **Closed** column.



You can add or rename columns as needed, see [Customize your board](#).

### TIP

You can add or rename columns as needed, see [Customize your board](#).

## Add tasks

Task checklists provide a quick and easy way to track elements of work which are important to support completing a backlog item. In addition, you can assign individual tasks to different team members.

### TIP

Tasks that you create from the Kanban board are automatically assigned to the sprint/iteration path of the parent work item under which you define them.

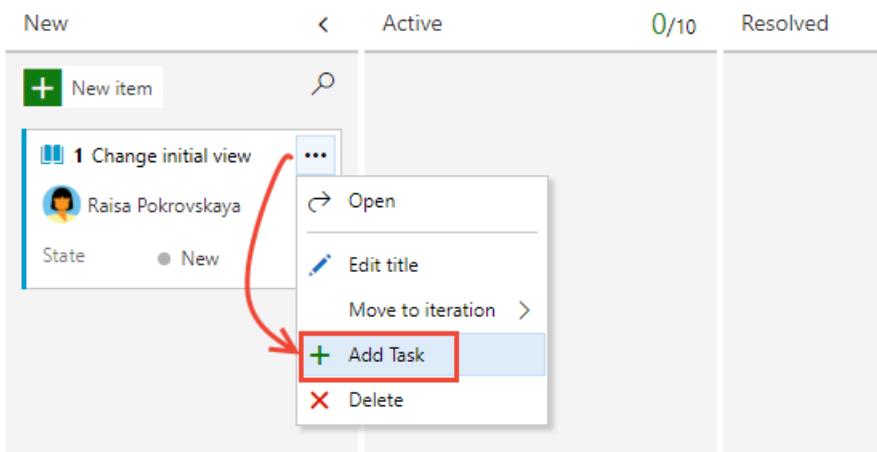
Tasks that you create from the Kanban board show up on your sprint taskboard. Also, tasks that you create from the [sprint backlog](#) or [taskboard](#) show up within tasks checklists on the Kanban board.

#### NOTE

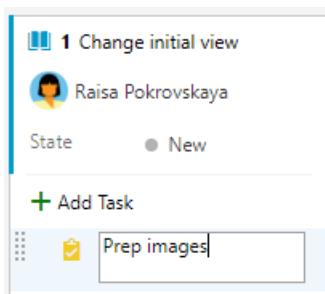
The Task checklists are available from with TFS 2015.1 and later versions.

- Agile process
- Basic process
- Scrum process
- CMMI process

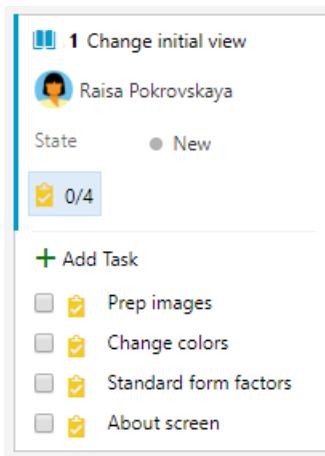
1. To start adding tasks, choose the **...** actions icon for the story and select the **+ Add Task** option.



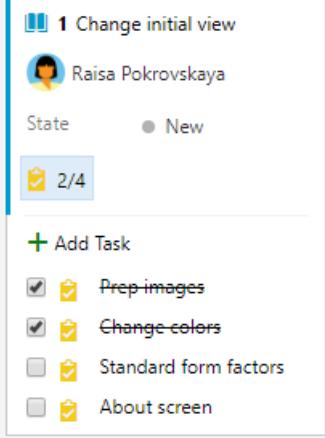
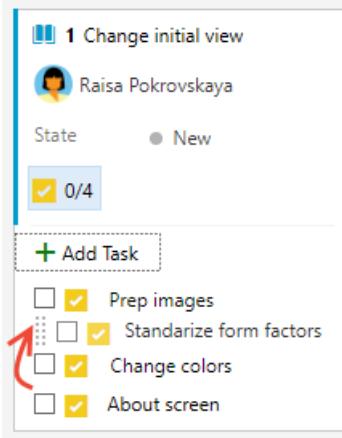
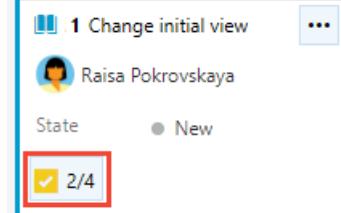
Enter a title for the task and type Enter when done.



2. If you have a number of tasks to add, simply keep typing your task titles and type Enter.



3. You can mark a task as done, expand or collapse the task checklist, or reorder and reparent tasks.

MARK A TASK AS DONE	REORDER AND REPARENT TASKS	EXPAND OR COLLAPSE THE CHECKLIST
To mark a task as complete, check the task checkbox. The task State changes to <b>Done</b> .  <p>The screenshot shows a task card for '1 Change initial view' assigned to Raisa Pokrovskaya. The state is 'New'. The checklist shows one item checked: 'Prep images'. The total count is 2/4.</p>	To reorder a task, drag it within the checklist. To reparent a task, drag it to another issue on the board.  <p>The screenshot shows the same task card as above, but the checklist is expanded. A red arrow points to the 'Standarize form factors' item, which is currently at the top of the list. Other items include 'Change colors' and 'About screen'.</p>	To expand or collapse a task checklist, simply choose the task annotation.  <p>The screenshot shows the task card with the checklist collapsed. The total count is now 0/4.</p>

## Add details to a task

If you have details you want to add about a task, choose the title, to open it. Change one or more field values, add a description, or make a note in the **Discussion** section. Choose **Save & Close** when done.

- [Agile process](#)
- [Basic process](#)
- [Scrum process](#)
- [CMMI process](#)

Here we assign the task to Christie Church.

The screenshot shows a task card in the Azure DevOps interface. At the top, it says "TASK 2\*" and has a "Save & Close" button. Below that, there are fields for "State" (New), "Area" (Fabrikam Fiber), "Reason" (New), and "Iteration" (Fabrikam Fiber). A "Details" tab is selected, showing a "Description" field containing "Prep new images for use on web site." Below the description is a rich text editor toolbar. To the right of the description are sections for "Planning" (Priority 2, Activity Design), "Effort (Hours)" (Original Estimate 8, Remaining 8, Completed 0), "Development", and "Related Work".

## Field descriptions

In addition to the fields you can define for a backlog item—user story, issue, product backlog item, or requirement—you can specify the following fields for a task to support capacity and time tracking.

### NOTE

There are no inherent time units associated with this field even though the taskboard always shows "h" for hours in relationship to Remaining Work. You can specify work in any unit of measurement your team chooses.

### Field

### Usage

### Activity

The type of activity that is required to perform a task. To learn more about how this field is used, see [Capacity planning](#). Allowed values are:

- Deployment
- Design
- Development
- Documentation
- Requirements
- Testing

### Discipline (CMMI process)

The type of activity that is required to perform a task. To learn more about how this field is used, see [Capacity](#)

[planning](#). Allowed values are:

- Analysis
- Development
- Test
- User Education
- User Experience

### Original Estimate

The amount of estimated work required to complete a task. Typically, this field doesn't change after it is assigned.

### Remaining Work

The amount of work that remains to finish a task. You can specify work in hours or in days. As work progresses, update this field. It's used to calculate [capacity charts](#) and the [sprint burndown chart](#).

If you divide a task into subtasks, specify Remaining Work for the subtasks only.

### Completed Work

The amount of work spent implementing a task. Enter a value for this field when you complete the task.

### Task Type (CMMI only)

Select the kind of task to implement from the allowed values:

- Corrective Action
- Mitigation Action
- Planned

## Capture comments in the Discussion section

Use the **Discussion** section to add and review comments made about the work being performed.



The rich text editor tool bar displays below the text entry area when you click your cursor within each text box that can be formatted.



#### NOTE

There is no Discussion work item field. To query work items with comments entered in the Discussion area, you filter on the [History](#) field. The full content of the text entered into the Discussion text box is added to the History field.

### Mention someone, a group, work item, or pull request (, , or )

Choose one of these icons —, , or — to open a menu of recent entries you've made to mention someone, link to a work item, or link to a pull request. Or, you can simply type @, #, or ! to open the same menu.



**NOTE**

This latest version of the rich text editor requires Azure DevOps Server 2019 Update 1 or later version.

Type a name, or enter a number and the menu list will filter to match your entry. Choose the entry you want to add. You can bring a group into the discussion by typing @ and the group name, such as a team or security group.

**Edit or delete a comment**

If you need to edit or delete any of your discussion comments, choose  **Edit** or choose the  actions icon and then choose **Delete**.

**NOTE**

The edit/delete feature requires Azure DevOps Server 2019 Update 1 or later version.

After updating the comment, choose **Update**. To delete the comment, you'll need to confirm that you want to delete it.

A full audit trail of all edited and deleted comments is maintained in the **History** tab on the work item form.

Use the **@mention control** to notify another team member about the discussion. Simply type @ and their name. To reference a work item, use the **#ID control**. Type # and a list of work items that you've recently referenced will appear from which you can select.

To reference a work item, use the **#ID control**. Type # and a list of work items that you've recently referenced will appear from which you can select.

Note that you can't edit or delete comments once they've been entered.

**IMPORTANT**

For on-premises Azure DevOps Server, you must configure an **SMTP server** in order for team members to receive notifications.

**Add a reaction to a comment**

You can add one or more reactions to any comment. Choose a smiley icon at the upper-right corner of any comment or choose from the icons at the bottom of a comment next to any existing reactions. To remove your reaction, click the reaction on the bottom of your comment. The following shows an example of the experience of adding a reaction, as well as the display of reactions on a comment.

## Try this next

[Customize your board](#)

## Related articles

- [Azure Boards FAQs](#)
- [Index to field descriptions](#)
- [Add tags to issues or tasks](#)

# Add, run, update inline tests

3/6/2021 • 3 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

Learn how to add, run, update, and expand and collapse inline tests in Azure DevOps.

To start manual testing, add the test to the user story or bug that you want to test. From the Kanban board, you can define inline tests or a set of manual tests for a backlog item. You also can run these tests and update their status. If you're new to working with the Kanban board, see the [Kanban quickstart](#).

Tests you create from the Kanban board are automatically linked to the user story or backlog item.

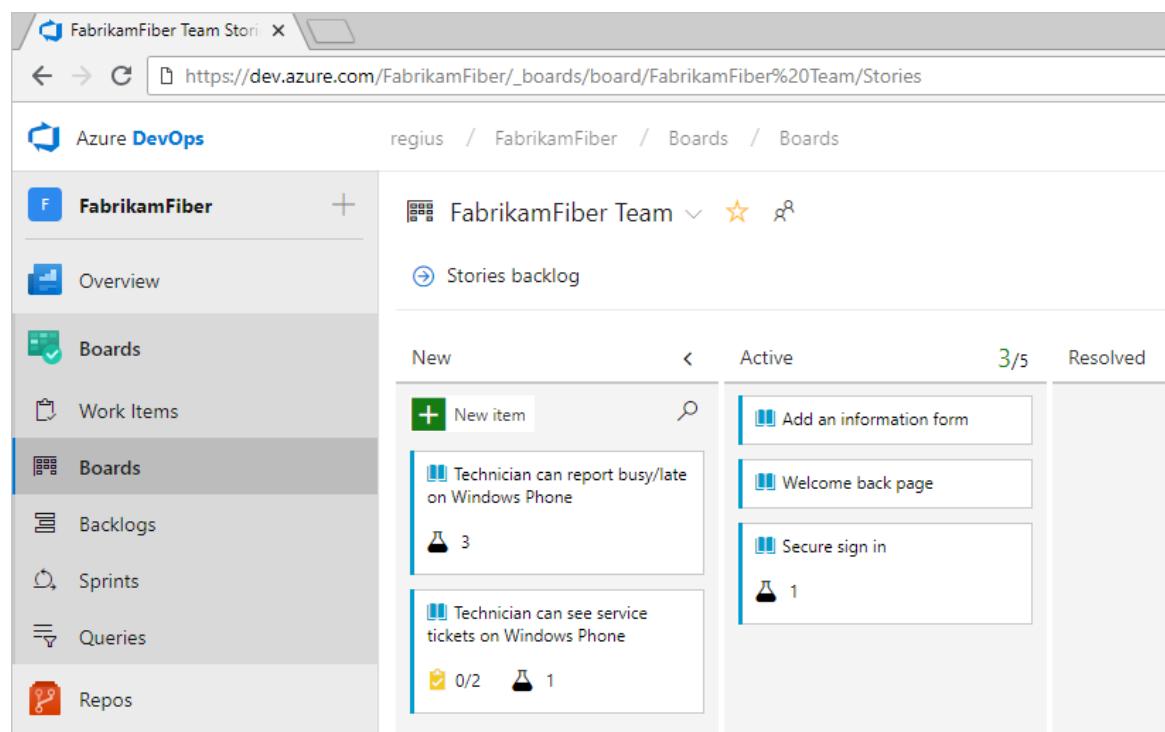
## Open your Kanban board

1. From your web browser, open the project for your organization and select **Azure Boards**. If you don't have a project, [create one now](#). If you haven't been added as a team member, [get invited now](#).

The URL follows this pattern: `https://dev.azure.com/fabrikamfiber/_boards/board`

If you don't see the team or project you want, select  [Azure DevOps](#) to [browse all projects and teams](#).

2. Select **Boards** to open the Kanban board.



1. From your web browser, open the project for your organization and select **Azure Boards**. If you don't have a project, [create one now](#). If you haven't been added as a team member, [get invited now](#).

The URL follows this pattern: `https://dev.azure.com/fabrikamfiber/_backlogs/board`

If you don't see the team or project you want, select  [Azure DevOps](#) to [browse all projects and teams](#).

2. Select **Board** to open the Kanban board.

The screenshot shows the 'Backlogs' tab selected in the navigation bar. On the left, a sidebar lists 'Features' and 'Stories' under 'Current'. Under 'Iteration 1', there is one item: 'Technician can report busy/late on Windows Phone' (status: 3). Under 'Iteration 2', there is one item: 'Technician can see service tickets on Windows Phone' (status: 0/2). The main area displays the 'Stories' backlog with two columns: 'New' and 'Active'. The 'Active' column contains three items: 'Add an information form', 'Welcome back page', and 'Secure sign in' (status: 1).

## Add tests

1. To add tests, open the menu for a work item.

The screenshot shows the 'Boards' tab selected in the navigation bar. On the left, a sidebar lists 'Overview', 'Work Items', 'Backlogs', 'Sprints', 'Queries', 'Repos', 'Pipelines', and 'Test Plans'. The 'Backlogs' section is expanded, showing 'FabrikamFiber Team Stories backlog'. A context menu is open over the first item in the backlog, 'Technician can report busy/late on Windows Phone'. The menu options are: Open, Edit title, Move to iteration, Add Task, Add Test (which is highlighted with a red box), Delete, New branch..., and Add to dashboard.

Inline tests are the same as test cases in a test suite. A default test plan and test suite automatically get created under which the manual test cases are grouped.

For example, a test suite is created for the following user story, and inline tests are added to that suite. User story 314 is highlighted. It has two manual tests defined with the IDs 337 and 341.

The screenshot shows the 'Test Plans' section of the Visual Studio Test Manager. A test suite named '314 : Technician can see service tickets on Windows Phone' is selected. This suite contains two manual tests with IDs 337 and 341, both marked as 'Active'. The interface includes standard navigation and filtering tools.

Outcome	Order	ID	Title
Active	1	337	Change colors on initial view
Active	2	341	Change initial page size

2. If you have a number of tests to add, enter each title and select **Enter**.

The screenshot shows the 'Add Test' dialog box. It lists two inline tests under the user story '314 : Technician can see service tickets on Windows Phone'. The first test is 'Change colors on initial view' and the second is 'Change initial page size'. There is also a placeholder for a third test.

To add details to the test case, open it. You can select the title, double-select the inline item, or open the context menu and choose **Open**.

The screenshot shows the details of a test case named "337 Change colors on initial view". The top navigation bar includes "TEST CASE 337", "Jamal Hartnett", "0 comments", "Add tag", "Save & Close", "Follow", and a three-dot menu. Below this, the "State" is "Design" and "Reason" is "New". The "Area" is "FabrikamFiber" and the "Iteration" is "FabrikamFiber\Release 1\Sprint 3". A tabs bar at the bottom includes "Steps" (selected), "Summary", "Associated Automation", a clock icon, a link icon with "(1)", and a clipboard icon.

**Steps**

Click or type here to add a step

**Development**

+ Add link  
Development hasn't started on this item.

**Related Work**

+ Add link ▾  
Tests  
314 Technician can see serv... Updated 2/28/2013, New

**Details**

Priority 2  
Automation status Not Automated

Parameter values

To learn more about how to define tests, see [Create manual tests](#).

Before you run the test, you must add details.

1. To add tests, open the menu for the work item.

The backlog board shows a user story titled "168 Hello World Web Site" by "Jamal Hartnett" with 1/4 completed. A context menu is open over another user story titled "169 Slow response on form". The menu items are: Open, Edit title, Add Task, Add Test (which is highlighted with a blue oval), Delete, Do exploratory testing, and New branch.

Inline tests are the same as test cases in a test suite. A default test plan and test suite automatically get created under which the manual test cases are grouped.

For example, a test suite gets created for each user story, and all inline tests are added to that suite. The following user story 152 is highlighted. It has three manual tests defined with the IDs 153, 155, and 161.

The screenshot shows the 'Test plan' section of a software application. At the top, there are tabs for HOME, CODE, WORK, BUILD, and TEST. Below the tabs, there are links for Test plan, Parameters, Runs, and Machines\*. A dropdown menu shows 'FabrikamFiber: FabrikamFiber Team\_Stories\_FabrikamFiber (Id: 157)'. On the left, a tree view shows 'FabrikamFiber Team\_Stories\_FabrikamFiber' expanded, with three items under it: '151 : Customer log in (2)', '152 : Customer welcome page (3)', and '169 : Slow response on form (1)'. On the right, a table titled 'Test suite: 152: Customer welcome pa...' shows three tests: 'Change colors on initial view' (Active, ID 153), 'Change initial page size' (Active, ID 155), and 'Log in with email' (Active, ID 161). There are buttons for 'New', 'Add existing', 'Delete', 'Edit', and 'Refresh'.

To learn more about test plans and test suites, see [Plan your tests](#).

2. If you have a number of tests to add, enter each title and select Enter.

The screenshot shows a modal dialog for a specific test case. The title is '152 Customer welcome page'. It displays a user profile for 'Raisa Pokrovskaya'. Below the profile are two progress bars: one yellow labeled '0/3' and one orange labeled '0/3'. A button 'Add Test' is present. A list of three test items is shown: 'Change colors on initial view', 'Change initial page size', and 'Log in with email'. Each item has a blue circular icon to its left.

To add details to the test case, open it. You can select the title, double-select the inline item, or open the context menu and choose **Open**.

TEST CASE 153

### 153 Change colors on initial view

Design Raisa Pokrovskaya 0

Area Iteration  
Fabrikam Fiber Fabrikam Fiber\Iteration 1

Add Tag Steps Summary Associated Automation (1)

**Steps**

Action Click or type here to add a step

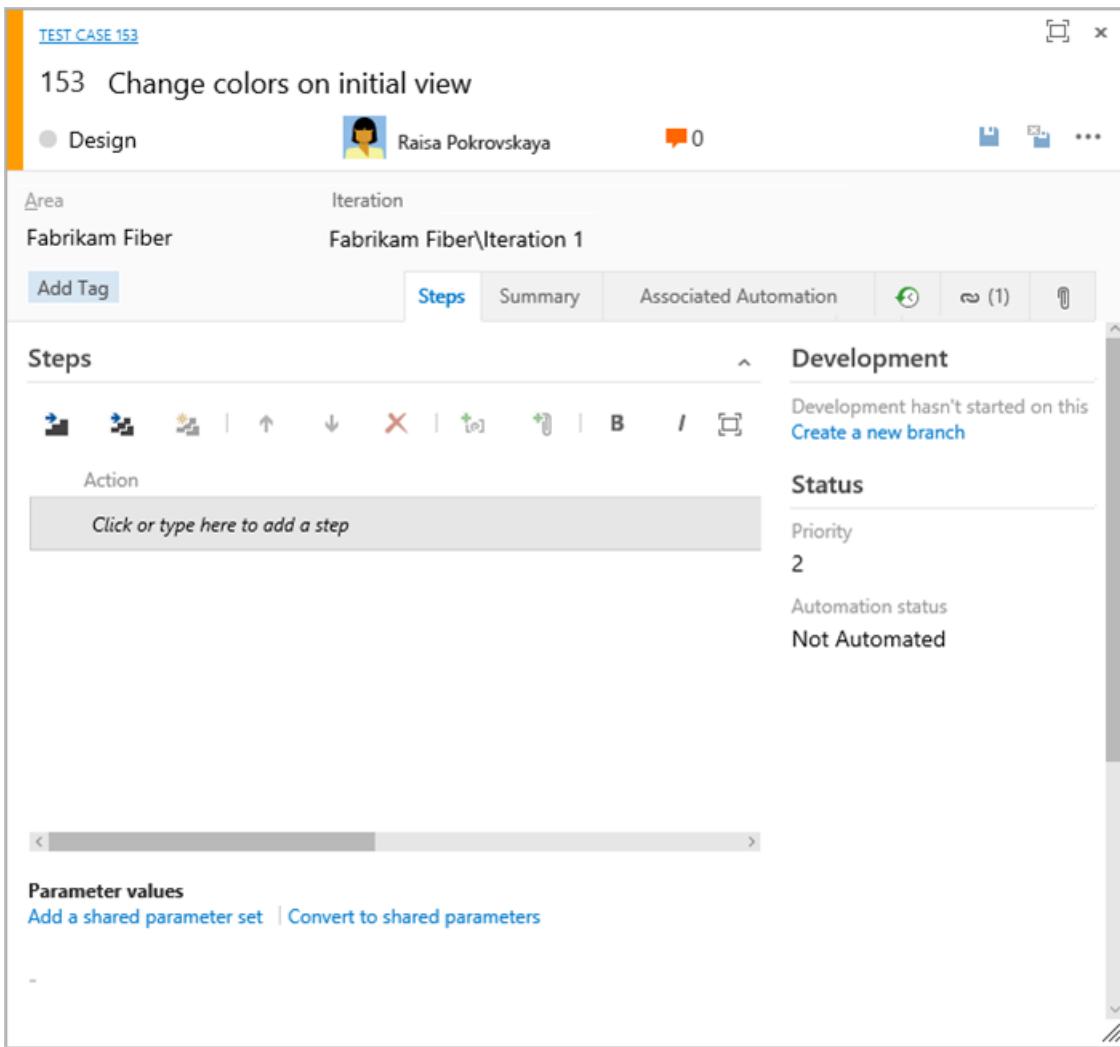
**Development**

Development hasn't started on this  
[Create a new branch](#)

**Status**

Priority 2  
Automation status Not Automated

Parameter values  
[Add a shared parameter set](#) | [Convert to shared parameters](#)

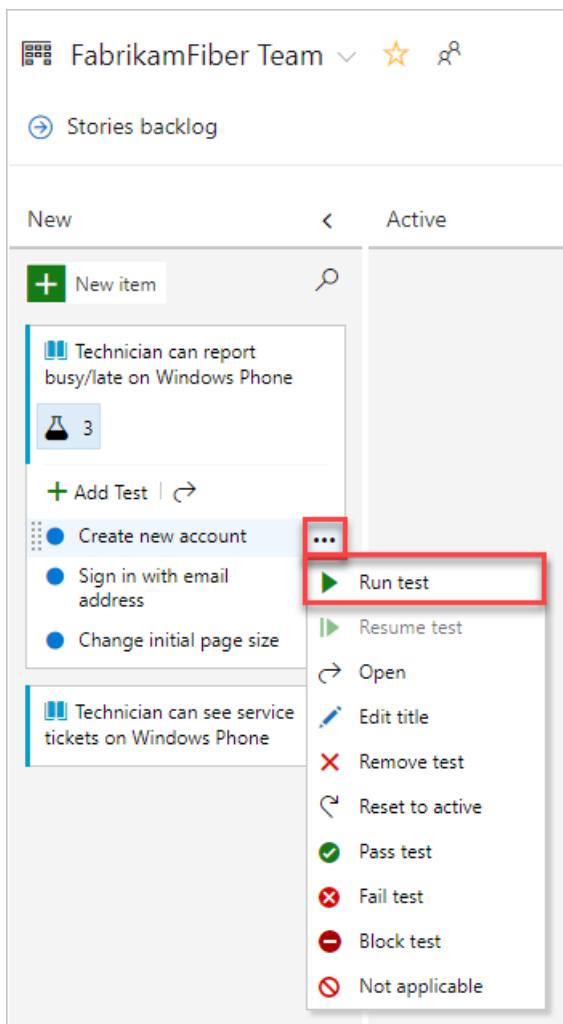
The screenshot shows a Microsoft Test Case management interface. At the top, there's a header with 'TEST CASE 153' and a title '153 Change colors on initial view'. Below the title, it shows 'Design' and the name 'Raisa Pokrovskaya' with a '0' notifications icon. There are buttons for 'Add Tag', 'Steps' (which is selected), 'Summary', 'Associated Automation', and three more icons. The main area is titled 'Steps' and has a placeholder 'Click or type here to add a step'. To the right, there's a sidebar with sections for 'Development' (not started, 'Create a new branch'), 'Status' (Priority 2, Not Automated), and 'Parameter values' (with links to add or convert shared parameter sets). A scroll bar is visible on the right side of the main content area.

To learn more about how to define tests, see [Create manual tests](#).

Before you run the test, you must add details.

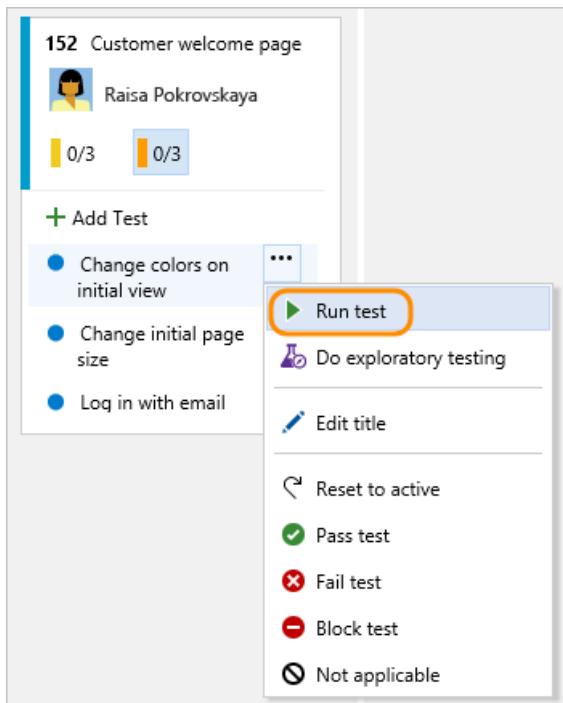
## Run a test

Run the test by selecting Run test from the actions menu for the inline test.



Microsoft Test Runner starts in a new browser instance. For information on how to run a test, see [Run manual tests](#).

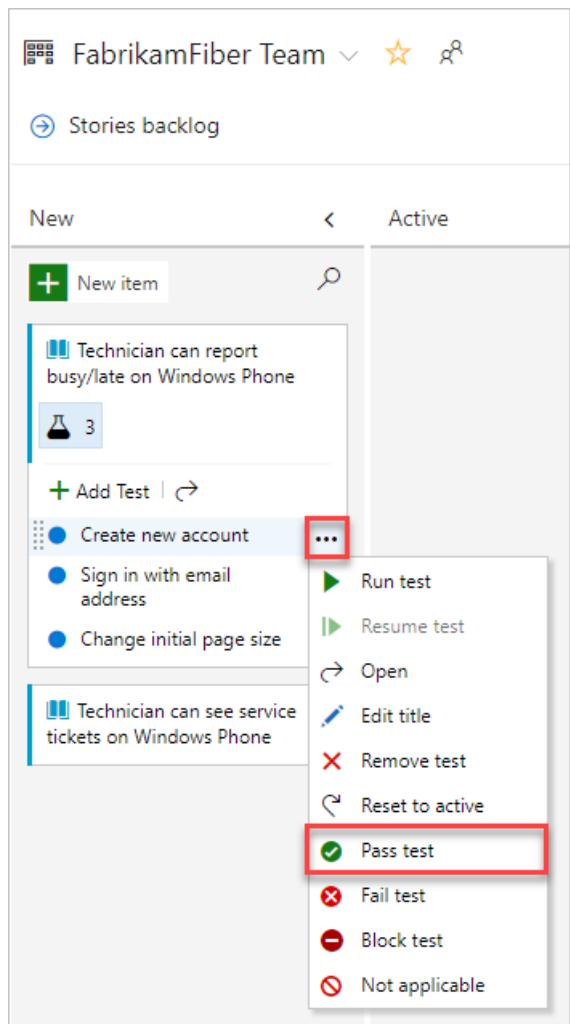
Run the test by selecting ► Run test from the ... actions menu for the inline test.



Microsoft Test Runner starts in a new browser instance. For information on how to run a test, see [Run manual tests](#).

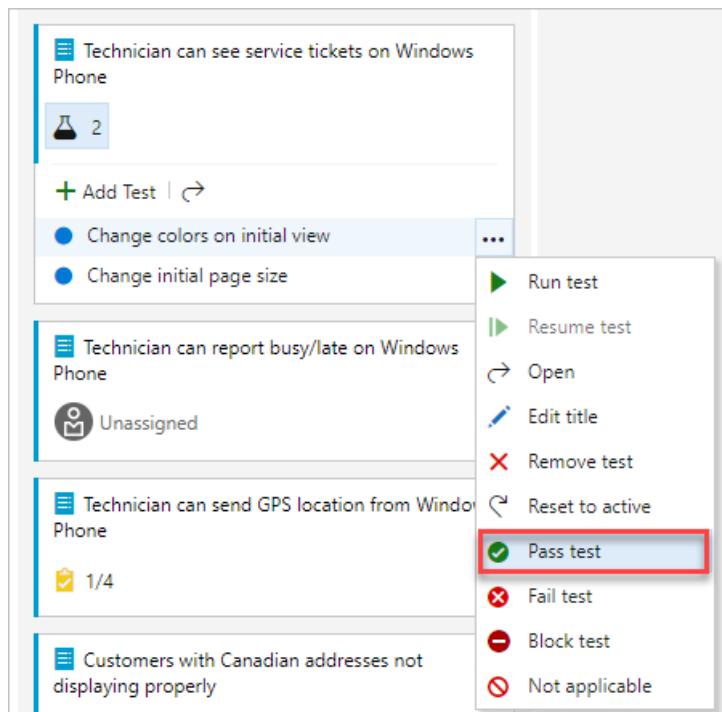
## Update the status of a test

You can update the status of the test from the \*\*\* actions menu.



When you update the status of tests, you can [track test results](#).

You can update the status of the test from the \*\*\* actions menu.



When you update the status of tests, you can [track test results](#).

## Expand or collapse inline tests

When you first open the Kanban board, you'll see an unexpanded view of checklists and tests.

The screenshot shows the Azure DevOps interface for the 'FabrikamFiber' team. The left sidebar has 'Boards' selected. The main area displays a backlog of stories. One specific story, 'Technician can see service tickets on Windows Phone', is highlighted. Its checklist is expanded, showing 2 items. A red arrow points from the collapsed state on the left to the expanded state on the right.

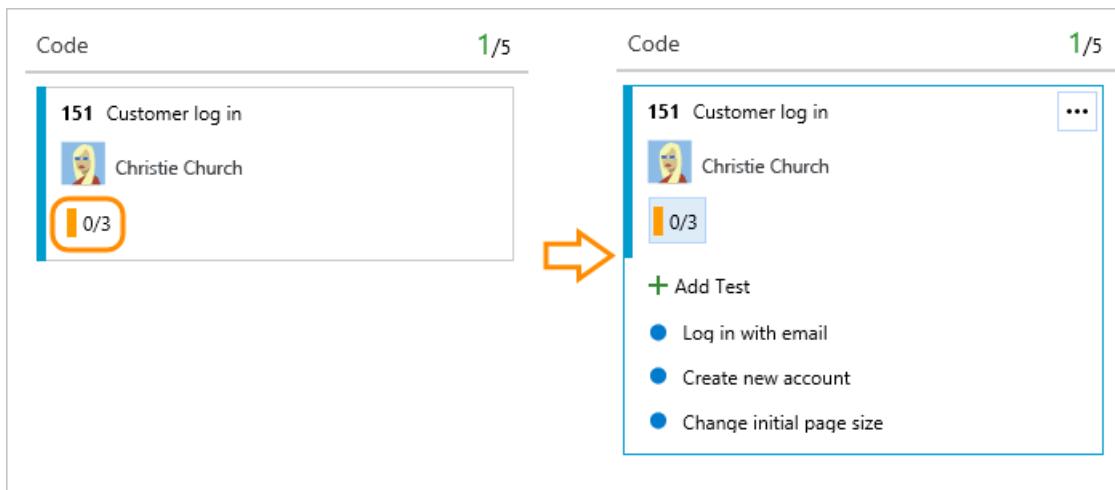
Select the inline test summary to expand a collapsed set of tests. Select the same summary to collapse an expanded list.

The screenshot shows the Azure DevOps interface for the 'FabrikamFiber' team. The left sidebar has 'Boards' selected. The main area displays a backlog of stories. One specific story, 'Technician can see service tickets on Windows Phone', is highlighted. Its checklist is expanded, showing 2 items. A red arrow points from the collapsed state on the left to the expanded state on the right.

When you first open the Kanban board, you'll see an unexpanded view of checklists.

The screenshot shows the Azure DevOps interface for the 'FabrikamFiber' team. The left sidebar has 'Boards' selected. The main area displays a backlog of stories across three columns: Analyze, Code, and Test. Stories 168, 151, and 169 have their first checklist item expanded, while others remain collapsed. A red arrow points from the collapsed state on the left to the expanded state on the right.

Select the inline test summary to expand a collapsed set of tests. Select the same summary to collapse an expanded list.



## Next steps

[Kanban quickstart](#)

## Related articles

- [Learn more about test case management](#)
- [Exploratory test your web app directly in your browser](#)
- [Essential services](#)
- [Client-server tools](#)
- [Software development roles](#)

# View permissions for yourself or others

4/28/2021 • 3 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Learn how to view your permissions or the permissions that are set for others in Azure DevOps. If you don't have a permission to access a feature or function, you can request it from the right resource.

Permissions are set at the collection, project, and object level as described in [Get started with permissions, access, and security groups](#). So to view the permissions you have, you need to open the permissions at the object, project, or collection level.

## NOTE

This article shows how to view permissions assigned to a user at the project-level or collection-level. However, the steps are similar when you work from the Security dialog of an object.

## Prerequisites

- You must have a project to connect to. If you don't have a project yet, [create one](#).
- You must be a member of the Project Valid Users Group or Project Collection Valid Users Group to view permissions.

## View project-level permissions

## NOTE

To enable the preview feature, for the new user interface for the [Project Permissions Settings Page](#), see [Enable preview features](#).

- [Preview page](#)
- [Current page](#)

1. Choose **Project Settings** and then **Permissions**.

The screenshot shows the 'Project Settings' page in Azure DevOps for the 'FabrikamFiber' project. On the left, there's a sidebar with links like Overview, Boards, Repos, Pipelines, and Artifacts. Below that is a 'Project settings' button, which is highlighted with a red box. The main content area is titled 'Project Settings' and contains sections for General, Boards, Repos, Pipelines, and Test. Under 'General', the 'Permissions' link is highlighted with a red box. To the right is a 'Permissions' panel with tabs for 'Groups' and 'Users'. The 'Groups' tab is selected, showing a list of groups with icons and names: Build Administrators (BA), Contributors (C), Fabrikam Fiber group (FG), Project Administrators (PA), Project Valid Users (PU), Readers (R), Release Administrators (RA), Email team (ET), and FabrikamFiber Team (FT). The 'Users' tab is also present but not selected.

2. Choose **Users**. To filter the list, enter a name into the *Search groups or users* box.

This screenshot shows the 'Permissions - Users' page. It has a 'Groups' tab and a 'Users' tab, with the 'Users' tab highlighted by a red box. A search bar at the top contains the name 'Jamal'. Below the search bar is a table with columns for 'Name' and 'Email'. The table lists several users: Azure Boards (AB), MyPublicProject Build Service (fabrikam) (MS), Demo 11 Build Service (fabrikam) (DS), Christie Church (fabrikamfiber1@hotmail.com), Chuck Reinhart (CR), and Jamal Hartnett (fabrikamfiber4@hotmail.com). The user 'Jamal Hartnett' is also shown in a dropdown menu below the search bar, along with an 'Edit' icon.

3. Choose the name you want. The project-level permissions for that user displays. These permissions are based on the groups the user belongs to or the permissions set specifically for the user's account.

The screenshot shows the 'Users' page in Microsoft Project. On the left, there's a sidebar with a back arrow labeled 'Users' and a search bar. The main area is titled 'Jamal Hartnett' and shows his profile picture, name, and email. Below this, there are two tabs: 'Permissions' (which is selected) and 'Member of'. The 'Permissions' tab is divided into several sections: 'General', 'Service Account', and 'Boards'. Each section contains a list of permissions with their current status (e.g., 'Not set', 'Allow').

Section	Permission	Status
General	Alter trace settings	Not set
	Create new projects	Allow
	Delete team project	Not set
	Edit instance-level information	Not set
	View audit log	Not set
Service Account	View instance-level information	Allow (inherited)
	Make requests on behalf of others	Not set
	Trigger events	Not set
Boards	View system synchronization information	Not set
	Create process	Not set
	Delete field from organization	Not set

4. Choose **Member of** to see which security groups and teams that the user belongs to.

Here we see that *Jamal Hartnett* belongs to several teams and the Project Collection Administrators group for several projects.

**Jamal Hartnett**

**Add member of**

**Permissions** **Member of** **Filter groups**

Name	User or scope
MT MyPublicProject Team	[MyPublicProject]
FS Fiber Suite	[Fabrikam Fiber A]
I Internet	[Fabrikam Fiber]
MT MyFirstProject Team	[MyFirstProject]
FT Fabrikam Team	[Fabrikam Fiber]
PA Project Administrators	[MyPublicProject]
PA Project Administrators	[MyFirstProject]

1. Open Project Settings. Choose the gear settings icon, and choose **Security**.

Fabrikam Fiber

Search

Overview Work **Security** Version Control Policies Agent queues Notifications Service Hooks

2. Begin entering the name into the *Filter users and groups* box. The system automatically shows the names that begin with the characters you enter.

Create group

Showing 1 result

Permissions	Members	Member of
Create tag definition	Allow (inherited)	
Create test runs	Allow (inherited)	
Delete and restore work items	Not set	
Delete team project	Not set	
Delete test runs	Allow (inherited)	
Edit project-level information	Not set	
Manage project properties	Not set	

3. Choose the name you want. The project-level permissions you have set are based on the groups you belong to or the permissions set for your account.

Create group < fabrikam > Jamal Hartnett

Permissions Member of	
Create tag definition	Allow (inherited)
Create test runs	Allow (inherited)
Delete and restore work items	Not set
Delete team project	Not set
Delete test runs	Allow (inherited)
Edit project-level information	Not set
Manage project properties	Not set
Manage test configurations	Allow (inherited)
Manage test environments	Allow (inherited)
Move work items out of this project	Not set
Permanently delete work items	Not set
Rename team project	Not set
View project-level information	Allow (inherited)
View test runs	Allow (inherited)

For a description of each permission, see [Permissions and groups reference](#).

4. Choose **Member of** to see which security groups the user belongs to.

Here we see that *Jamal Hartnett* belongs to several teams and the Project Collection Administrators group.

Create group < fabrikam > Jamal Hartnett

Permissions Member of		
<a href="#">+ Add...</a>   <a href="#">Edit</a>   <a href="#">Search</a>		
Display Name	Username Or Scope	
[Customer Service]	[Fabrikam Fiber]	<a href="#">Remove</a>
[Fabrikam Fiber Team]	[Fabrikam Fiber]	
[Web]	[Fabrikam Fiber]	
[Project Collection Administrators]	[fabrikam]	

For a description of each group, see [Permissions and groups reference](#).

## View organization or collection-level permissions

Open admin settings for the organization or a project collection.

1. Choose the  Azure DevOps logo to open **Projects**. Then choose **Organization settings**.

2. Choose **Permissions**, the **Project Collection Administrators** group, and then **Members**.

[fabrikam]\Project Collection Administrators

**Permissions** **Members** Member of

+ Add members

Filters users and groups

Name	Type	Username or scope
Christie Church fabrikamfiber1@hotmail.com	user	fabrikamfiber1@hotmail.com
PA Project Collection Service Accounts	group	[mseng]
Jamal Hartnett fabrikamfiber4@hotmail.com	user	fabrikamfiber4@hotmail.com
Raisa Pokrovskaya fabrikamfiber5@hotmail.com	user	fabrikamfiber5@hotmail.com
Helena Petersen fabrikamfiber8@hotmail.com	user	fabrikamfiber8@hotmail.com

3. Follow steps 2 through 4 in the procedure outlined previously for view project-level permissions.

1. Choose the  Azure DevOps logo to open **Projects**. Then choose **Admin settings**.



2. Choose **Security**, the **Project Collection Administrators** group, and then **Members**.

Create group

Filter users and groups

- ✓ Azure DevOps Groups
  - >  **Project Collection Administrators**
  - >  Project Collection Build Administrators
  - >  Project Collection Build Service Accounts
  - >  Project Collection Proxy Service Accounts
  - >  Project Collection Service Accounts
  - >  Project Collection Test Service Accounts
  - >  Project Collection Valid Users
  - >  Security Service Group

fabrikam > Project Collection Administrators

Permissions **Members** Member of

+ Add... |  | Search

Display Name	Username Or Scope
Project Collection Service ...	[REDACTED]
Christie Church	fabrikamfiber1@hotmail.com
Jamal Hartnett	fabrikamfiber4@hotmail.com
Raisa Pokrovskaya	fabrikamfiber5@hotmail.com

3. Follow steps 2 through 4 in the procedure outlined previously for view project-level permissions.

1. Choose the  settings icon and select **Organization settings** or **Collection settings**.



2. Choose **Security**, **Project Collection Administrators** group, and then **Members**.

The screenshot shows the Azure DevOps Security interface. The top navigation bar includes 'fabrikam' (dropdown), 'Projects', 'My favorites', 'My work items', 'My pull requests', '...', and a gear icon. Below the bar, tabs for 'Overview', 'Settings', 'Security' (highlighted with a red box), 'Users', 'Process', 'Build and Release', 'Agent Pools', and 'Notifications' are visible. On the left, a sidebar titled 'Create group' shows a list of 'Azure DevOps Groups': 'Project Collection Administrators' (selected and highlighted with a red box), 'Project Collection Build Administrators', 'Project Collection Build Service Accounts', 'Project Collection Proxy Service Accounts', 'Project Collection Service Accounts', 'Project Collection Test Service Accounts', 'Project Collection Valid Users', and 'Security Service Group'. The main content area displays 'fabrikam > Project Collection Administrators' with an 'Edit' link. It has tabs for 'Permissions' and 'Members' (highlighted with a red box). Below are buttons for '+ Add...', a refresh icon, and 'Search'. A table lists members with columns for 'Display Name' and 'Username Or Scope'. The table includes entries for 'Project Collection Service ...', 'Christie Church' (fabrikamfiber1@hotmail.com), 'Jamal Hartnett' (fabrikamfiber4@hotmail.com), and 'Raisa Pokrovskaya' (fabrikamfiber5@hotmail.com).

3. Follow steps 2 through 4 in the procedure outlined previously for view project-level permissions.

## View object-level permissions

You can define the security or permissions for a number of objects. You access them from the context menu of the object.

From the web portal, open the Security dialog for the object whose permissions you want to set. For specific instructions, see the following articles:

AREA	TASK
Wiki & Dashboard permissions	<ul style="list-style-type: none"><li>• <a href="#">README &amp; Wiki</a></li><li>• <a href="#">Dashboards</a></li></ul>
Azure Repos, Azure Pipelines/DevOps (code, build, test, release) permissions	<ul style="list-style-type: none"><li>• <a href="#">Git branch</a></li><li>• <a href="#">Git repository</a></li><li>• <a href="#">TFVC</a></li><li>• <a href="#">Builds</a></li><li>• <a href="#">Release pipeline security</a></li><li>• <a href="#">Approvals and approvers</a></li></ul>
Azure Boards/Work tracking permissions	<ul style="list-style-type: none"><li>• <a href="#">Area and iteration paths</a></li><li>• <a href="#">Work item query and folder</a></li><li>• <a href="#">Plan permissions</a></li></ul>

## Next steps

[Look up the organization owner or a Project Administrator](#)

## Related articles

- [Troubleshoot permissions](#)

- [Permissions and role lookup guide](#)

# Tutorial: Follow a user story, bug, issue, or other work item or pull request

3/6/2021 • 5 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

To get notified of changes made to a specific work item or a pull request, you can elect to follow them. The Follow feature provides an ad hoc way of getting notified on a case-by-case basis.

On the other hand, if you want to subscribe to receive notifications automatically based on changes that occur based on your targeted set of criteria, see [Manage personal notifications](#). For example, you can create a subscription to automatically get notified whenever a work item that you created or that was assigned to you is modified.

## NOTE

Notification subscriptions allow you to personalize the notifications you receive automatically based on additional criteria you specify for [yourself](#), a team, or a project. For example, you can create a subscription and add field criteria to receive changes based on one or more of the following templates.

Template

- A work item I created is changed
- A work item assigned to me is changed
- A work item is created
- A work item is assigned to me
- A work item is deleted
- A work item is restored

This article shows you how to:

- Follow a work item
- Follow a pull request
- Manage work items that you're following

You must configure an [SMTP server](#) in order for team members to receive notifications.

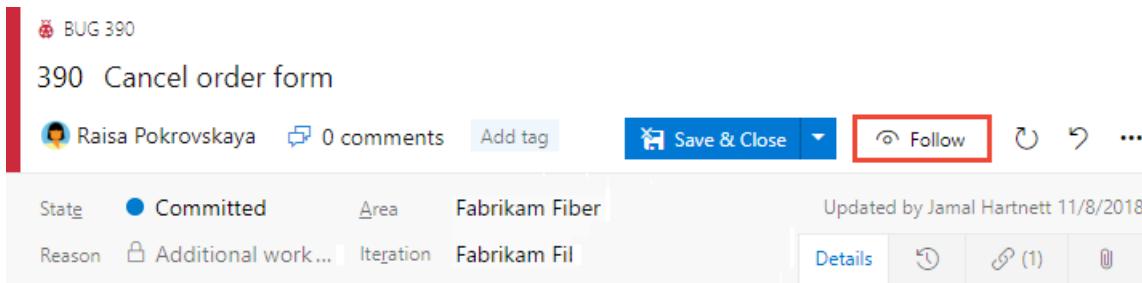
## Prerequisites

- You must connect to a project. If you don't have a project yet, [create one](#).
- You must be added to a project as a member of the **Contributors** or **Project Administrators** security group. To get added, [Add users to a project or team](#).
- To view or follow work items, you must be granted **Stakeholder** access or higher. For details, see [About access levels](#). Also, you must have your **View work items in this node** and **Edit work items in this node** permissions set to **Allow**. By default, the **Contributors** group has this permission set. To learn more, see [Set permissions and access for work tracking](#).

- To view or follow pull requests, you must have **Basic** access or higher.
- You must connect to a project. If you don't have a project yet, [create one](#).
- You must be added to a project as a member of the **Contributors** or **Project Administrators** security group. To get added, [Add users to a project or team](#).
- To view or follow work items, you must be granted **Stakeholder** access or higher. For details, see [About access levels](#). Also, you must have your **View work items in this node** and **Edit work items in this node** permissions set to **Allow**. By default, the **Contributors** group has this permission set. To learn more, see [Set permissions and access for work tracking](#).
- To view or follow pull requests, you must have **Basic** access or higher.

## Follow a work item

When you want to track the progress of a single work item, choose the  [Follow](#) follow icon. This signals the system to notify you when changes are made to the work item.



The screenshot shows a work item details page for a task titled "390 Cancel order form". The work item is assigned to "Raisa Pokrovskaya" and has 0 comments. The "Save & Close" button is visible, along with a "Follow" button which is highlighted with a red box. Below the title, there are fields for "State" (Committed), "Area" (Fabrikam Fiber), "Reason" (Additional work...), and "Iteration" (Fabrikam Fil). The last update was made by "Jamal Hartnett" on "11/8/2018". At the bottom right, there are buttons for "Details", "Edit", "Delete", and "Copy".

If you want to specify conditions on when you'll get notified of changes, choose the  gear icon and choose from the options provided.

### Notification Settings

**Not Subscribed**

Only receive notifications from this work item when you have been @mentioned.

**Subscribed**

Receive all notifications from this work item.

**Custom**

You will only be notified for the events selected from the list below or when you are @mentioned in this work item.

State Changed

Assigned To Changed

Iteration Changed

**OK**

Cancel

By default, you are **Subscribed** to receive a notification when any change is made to the work item. Choose **Not Subscribed** to receive notification only when you are @mentioned. Or choose **Custom** to receive notifications when one of the checked fields changes, **State**, **Assigned To**, or **Iteration Path**.

BUG 390

## 390 Cancel order form

Raisa Pokrovskaya 0 comments Add tag Save & Close Follow

State Committed Area Fabrikam Fiber Updated by Jamal Hartnett 11/8/2018

Reason Additional work... Iteration Fabrikam Fil Details (1) [Attachment]

### NOTE

The **Follow a work item** feature is available from TFS 2017 and later versions. The **Follow a pull request** feature is available from TFS 2017.1 and later versions. To update your on-premises Azure DevOps, visit the [Visual Studio downloads page for Team Foundation Server](#).

You'll only receive notifications when other members of your team modifies the work item, such as adding to the discussion, changing a field value, or adding an attachment.

Notifications are sent to your preferred email address, which [you can change from your user profile](#).

To stop following changes, choose the following icon.

## Follow a pull request

To track the progress of a single pull request, choose the actions icon for the pull request, and select the **Follow** option. This signals the system to notify you when changes are made to the PR.

Approve Complete ...

- Share Pull Request
- Save all comments
- Follow
- Restart merge
- Cherry-pick...

You'll only receive notifications when other members of your team modifies the PR, such as adding to the discussion or adding an attachment.

Notifications are sent to your preferred email address, which [you can change from your user profile](#).

To stop following changes, open the PR context menu and choose the Following icon.

## Manage work items that you're following

You can review and manage all the work items you've selected to follow.

Open **Boards>Queries**, choose **All**, and under **My Queries**, choose **Followed work items**.

FF Fabrikam Fiber

Overview Boards Work Items Boards Backlogs Sprints Queries Plans Code

Queries

Favorites All + New query

Title

My Queries

- Active bugs
- All Items
- Assigned to me
- Closed bugs
- Fabrikam Fiber Team - Backlog items
- Followed work items
- Following - my query

From this view, you can view all items you're following across all projects. Also, you can perform similar actions supported with a query results view, such as:

- Refresh the view
- Add or remove visible columns
- Sort the order of specific columns
- Filter results by text or tags
- Set work item pane
- Enter full screen mode.

You can also view and manage work that you're following from **Boards>Work Items** and pivot to **Following**.

FF Fabrikam Fiber

Overview Boards Work Items Boards Backlogs Sprints Queries

Work Items

Following + New Work Item Open in Queries Column Op

ID	Assigned To	State	Title
375	Jamal Hartnett	● Committed	Check service status
361	Christie Church	● Approved	Interim save on long form
384	Christie Church	● Committed	Secure sign-in
360	Raisa Pokrovskaya	● New	Change initial view
436	Jamal Hartnett	● Committed	Hello World Web Site

Open **Work>Queries** and choose **Followed work items**.

The screenshot shows the 'Followed work items' section within the 'Queries' tab of the Microsoft Team Services interface for the 'Fabrikam Fiber' project. On the left, a sidebar lists categories like 'Assigned to me', 'Followed work items' (which is selected and highlighted with a blue border), and 'Unsaved work items'. Below these are sections for 'My favorites' and 'Team favorites', each with a placeholder for dragging queries. To the right, the main area displays a table titled 'Results' with columns for 'ID', 'Work Item Type', 'Title', and 'State'. Three work items are listed:

ID	Work Item Type	Title	State
3	Bug	Slow response on form	Resolved
2	User Story	Cancel order form	Active
1	User Story	Welcome page	Active

From this view, you can view all items you're following across all projects. Also, you can perform similar actions supported with a query results view, such as:

- Refresh the view
- Add or remove visible columns
- Sort the order of specific columns
- Filter results by text or tags
- Set work item pane
- Enter full screen mode.

You can also view and manage work that you're following from your Project pages. To learn more, see [Work across projects](#).

## Query work items that you're following

You can use the `@Follows` macro in a query to filter a list based on work items you're following in addition to other query filters.

For example, the following query shows how to query across all projects for active work items that you're following. You use the ID field and the `In` operator with the `@Follows` macro.

The screenshot shows the 'My Queries' query editor. At the top, it displays '3 work items' with '1 selected'. Below this are tabs for 'Results', 'Editor', 'Charts', 'GANTT', and 'Export', with 'Editor' being the active tab. A 'Run query' button and a 'More' (three-dot) menu are also present. The main area is divided into sections: 'Type of query' (set to 'Flat list of work items'), 'Filters for top level work items', and 'Advanced filters'. The 'Filters for top level work items' section contains the following configuration:

And/Or	Field*	Operator	Value
+ X	Work Item Type	=	[Any]
+ X	State	=	Active
+ X	ID	In	@Follows

Below this, there's a link to 'Add new clause'.

## Try this next

[Add, update, and follow a work item](#)

## Related articles

- [Manage personal notifications](#)
- [View and update work items via the mobile work item form](#)

### **Q: Can I add someone else to follow a work item or PR?**

A: You can't add another team member to follow a work item or pull request at this time. You can subscribe them to get notified based on select criteria, such as when a work item is created or modified, or a pull request is created. For details, see [Manage team notifications](#).

# Get started as a Stakeholder

4/12/2021 • 21 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Stakeholders are users with free but limited access to Azure DevOps features and functions. With Stakeholder access, you can add and modify work items, manage build and release pipelines, and view dashboards. You can check project status and provide direction, feedback, feature ideas, and business alignment to a team. For a quick overview of the features available to Stakeholders, see the [Features and functions available to Stakeholders](#) later in this article.

## NOTE

For public projects, Stakeholder access gives users greater access to features. To learn more, see [Default roles and access for public projects](#). For information about working with pipelines, see these articles: [Build your GitHub repository](#) and [Build OSS repositories](#).

Stakeholders are users with free but limited access to Azure DevOps features and functions. With Stakeholder access, you can add and modify work items, view and approve pipelines, and view dashboards. You can check project status and provide direction, feedback, feature ideas, and business alignment to a team.

Stakeholders are users with free but limited access to Azure DevOps features and functions. With Stakeholder access, you can add and modify work items. You can check project status and provide direction, feedback, feature ideas, and business alignment to a team.

Stakeholder access is one of several supported access levels as described in [About access levels](#). To get access as a Stakeholder, ask your organization owner or Project Collection Administrator to [add you to a project with Stakeholder access](#).

Stakeholder access is one of several supported access levels as described in [About access levels](#). To get access as a Stakeholder, ask your server administrator to [add you to a security group that has Stakeholder access](#).

## NOTE

Azure Boards supports several Agile methods such as Kanban and Scrum. Depending on what methods your team uses, you'll want to become familiar with other tools that Azure Boards supports. This article focuses on getting familiar with work items and the Kanban board. For additional information, see [Related articles](#) at the end of this article.

Use this tutorial to learn how to do the following tasks:

- Sign in to a project
- Understand which work item types are available to your project
- Open the Kanban board and open a work item
- Add details, tags, or comments to a work item
- View the product backlog
- Find work assigned to you, or query for other work items
- Understand what features are and aren't available to users with Stakeholder access

# Connect to the web portal of a project

You must have been added to the Azure DevOps project and been granted Stakeholder or higher access level.

1. Choose the link provided in the email invitation you should have received. Or, open a browser window and enter the URL for the web portal.

`https://dev.azure.com/OrganizationName/ProjectName`

`http://ServerName:8080/tfs/DefaultCollection/ProjectName` For example, to connect to the server named *FabrikamPrime* and project named *Contoso*, enter  
`http://FabrikamPrime:8080/tfs/DefaultCollection/Contoso`.

2. Enter your credentials. If you can't sign in, ask the organization owner or Project Administrator to add you as a member of the project with Stakeholder access.

## Understand work items and work item types

Work items support planning and tracking work. Each work item represents an object stored in the work item data store. Each work item is based on a work item type and is assigned an identifier which is unique within an organization or project collection. Different work items are used to track different types of work as described in [About work items](#). The work item types available to you are based on the [process used when your project was created](#)—Agile, Basic, Scrum, or CMMI—as illustrated in the following images.

- [Agile process](#)
- [Basic process](#)
- [Scrum process](#)
- [CMMI process](#)

The following image shows the Agile process backlog work item hierarchy. User Stories and Tasks are used to track work, Bugs track code defects, and Epics and Features are used to group work under larger scenarios.



Each team can configure how they manage Bugs—at the same level as User Stories or Tasks—by configuring the [Working with bugs](#) setting. To learn more about using these work item types, see [Agile process](#).

## Open your Kanban board from the web portal

You can start viewing work items once you connect to a project.

1. Check that you selected the right project, and select **Boards > Boards**. Then select the correct team from the team selector menu.



To select another team's board, open the selector. Then select a different team, or select the `[[team]]` **Browse all team boards** option. Or, you can enter a keyword in the search box to filter the list of team backlogs for the project.



**TIP**

Select the  star icon to make a team board a favorite. Favorite artifacts () appear at the top of the team selector list.

2. Check that you selected **Stories** for Agile, **Issues** for Basic, **Backlog items** for Scrum, or **Requirements** for CMMI as the backlog level.



1. Check that you selected the right project, and select **Boards > Boards**. Then select the correct team from the team selector menu.



To select another team's board, open the selector. Then select a different team, or select the  **Browse all team boards** option. Or, you can enter a keyword in the search box to filter the list of team backlogs for the project.

**TIP**

Select the  star icon to make a team board a favorite. Favorite artifacts () appear at the top of the team selector list.

2. Check that you selected **Stories** for Agile, **Issues** for Basic, **Backlog items** for Scrum, or **Requirements** for CMMI as the backlog level. Here we have selected **Backlog Items** for the Scrum process.



1. To view your Kanban board, open your project from a web browser. Select **Work > Backlogs > Stories**, and then select **Board**.



If you don't see **Work**, your screen size might be reduced. Select the three dots  icon. Then select **Work > Backlogs > Board**.



2. To select another team, open the project and team selector. Select a different team, or select the **Browse** option.



Your Kanban board appears.



1. To view your Kanban board, open your project from a web browser. Select **Work > Backlogs > Stories**, and then select **Board**.



If you don't see **Work**, your screen size might be reduced. Select the three dots  icon. Then select **Work > Backlogs > Board**.

2. To select another team, open the project and team selector. Select a different team, or select the **Browse** option.



Your Kanban board appears.



## Add work items

From the Kanban board, you can add work items, open them, and modify them. To add work items, open the backlog by choosing the **Backlog** link. To add a work item, select the plus sign, enter a title, and then press Enter.

The screenshot shows a Kanban board with four columns: New, Active, Resolved, and Closed. The 'New' column has a header 'New' and a status '0/5'. Below it is a card with a green '+' button labeled 'New item'. The 'Active' column has a header 'Active' and a status '0/5'. It contains a card with the text 'Hello World'. The other two columns are empty.

Or, you can add work items to the bottom of the product backlog. Open the backlog by choosing the **Backlog** link.

From the Kanban board, you can't add work items, but you can open them and annotate them. To add work items, open the backlog by choosing the **Backlog** link. Also, you can't update the status of a work item by drag-and-drop to a different column or reorder cards within a column.

## Update status of work items

As work completes in one stage, update the status of an item by dragging it to a downstream stage.

The screenshot shows a Kanban board with five columns: Backlog, Analyze, Develop, Test, and another Develop column. The 'Backlog' column has a 'New item' button. The 'Analyze' column has a header 'Analyze' and a status '4/10'. It contains two cards: 'Interim save on long forms' (Raisa Pokrovsk... 8) and 'Resume' (Jamal Hartnett 2). The 'Develop' column has a header 'Develop' and a status '4/5'. It contains two cards: 'Customer can find the nearest Fabrikam Fiber location' (Jamal Hartnett 3) and 'Show response on welcome page' (Raisa Pokrovsk... 2). The 'Test' column has a header 'Test' and a status '4/5'. It contains two cards: 'Welcome back page' (Jamal Hartnett 3) and 'Autolocation persisting in state state between sessions' (Raisa Pokrovsk... 2). An orange arrow points from the 'Resume' card in the 'Analyze' column to the 'Show response on welcome page' card in the 'Develop' column.

### NOTE

The drag-and-drop feature to update the work item state requires installation of Azure DevOps Server 2020.1 update. To learn more, see [Azure DevOps Server 2020 Update 1 RC1 Release Notes, Boards](#).

## Add details to a work item

To add information to a work item, open it by double-clicking the title or by selecting it and then typing Enter. Change one or more field values, add a description, [add a tag](#), or add a comment in the **Discussion** section. You can also choose the **Attachments** tab and drag-and-drop or upload a file to share with others.

To add information to a work item, open it by double-clicking the title or by selecting it and then typing Enter. Add a description, change one or more field values, or [add a tag](#). You can also choose the **Attachments** tab and upload a file to the work item to share with others.

You can only assign work to a user who has been added to the project.

### NOTE

The work item form you see may differ from those shown in the following images. The basic functionality is the same, however, changes have been made with different versions of Azure DevOps.

- [Agile process](#)
- [Basic process](#)
- [Scrum process](#)
- [CMMI process](#)

For example, here we assign the story to Raisa Pokrovskaya and we add a discussion note, at-mentioning Raisa. Choose **Save & Close** when done.

The screenshot shows the 'USER STORY 1\*' work item details page. At the top, there's a header with the title, a 'Save & Close' button, and other navigation links. Below the header, there are sections for 'Description' and 'Planning'. The 'Description' section contains a note about switching the initial view. The 'Planning' section shows story points as 2, priority as 2, and risk as low. There are tabs for 'Details', 'Related Work items', and other work item types. The 'Discussion' tab is active, showing a comment from Raisa Pokrovskaya asking if something can happen in the next week. Below the discussion, there are rich text editing tools. On the right side, there are sections for 'Classification' (Value area: Business), 'Development', and 'Related Work'.

### Field descriptions

#### Field

## Usage

---

### Title

Enter a description of 255 characters or less. You can always modify the title later.

---

### Assigned To

Assign the work item to the team member responsible for performing the work. Depending on the context you are working in, the drop-down menu lists only team members or contributors to the project.

#### NOTE

You can only assign work to a single user. If you need to assign work to more than one user, add a work item for each user and distinguish the work to be done by title and description. The Assigned To field only accepts user accounts that have been [added to a project or team](#).

---

### State

When the work item is created, the State defaults to the first state in the workflow. As work progresses, update it to reflect the current status.

---

### Reason

Use the default first. Update it when you change state as need. Each State is associated with a default reason.

---

### Area (Path)

Choose the area path associated with the product or team, or leave blank until assigned during a planning meeting. To change the dropdown list of areas, see [Define area paths and assign to a team](#).

---

### Iteration (Path)

Choose the sprint or iteration in which the work is to be completed, or leave it blank and assign it later during a planning meeting. To change the drop-down list of iterations, see [Define iteration paths and configure team iterations](#).

---

### Description

Provide enough detail to create shared understanding of scope and support estimation efforts. Focus on the user, what they want to accomplish, and why. Don't describe how to develop the product. Do provide sufficient details so that your team can write tasks and test cases to implement the item.

---

### Acceptance Criteria

Provide the criteria to be met before the work item can be closed. Define what "Done" means by describing the criteria for the team to use to verify whether the backlog item or bug fix is fully implemented. Before work begins, describe the [criteria for customer acceptance](#) as clearly as possible. Have conversations between the team and customers to determine the acceptance criteria. These criteria help ensure a common understanding within the team to meet customers' expectations. Also, this information provides the basis for acceptance testing.

---

### Priority

A subjective rating of the issue or task it relates to the business. You can specify the following values:

- 1: Product cannot ship without the successful resolution of the work item, and it should be addressed as

soon as possible.

- 2: Product cannot ship without the successful resolution of the work item, but it does not need to be addressed immediately.
- 3: Resolution of the work item is optional based on resources, time, and risk.
- 4: Resolution of the work item is not required.

## Value Area

A subjective rating of the issue or task it relates to the business. You can specify the following values:

- **Architectural**: Technical services to implement business features that deliver solution .
- **Business**: Services that fulfill customers or stakeholder needs that directly deliver customer value to support the business (Default).

## Effort, Story Points, Size

Provide a relative estimate of the amount of work required to complete an issue. Most Agile methods recommend that you set estimates for backlog items based on relative size of work. Such methods include powers of 2 (1, 2, 4, 8) and the Fibonacci sequence (1, 2, 3, 5, 8, etc.). Use any numeric unit of measurement your team prefers.

The estimates you set are used to calculate [team velocity](#) and [forecast sprints](#).

## Add tags to a work item

Tags are useful for filtering backlogs, boards, and queries. As a Stakeholder, you can add existing tags to a work item, however, you can't add new tags.

From the web portal, open a work item and choose **Add tag** and type a keyword of an existing tag. Or, select from the list of previously assigned tags.

This screenshot shows a work item details page for a User Story. At the top right, there is a button labeled "Add tag" which is highlighted with a red box. Below the title, there are sections for State (New), Reason (New), Area (Fabrikam Fiber A), and Iteration (Fabrikam Fiber A). On the far left, there is a sidebar with a blue header bar containing the text "USER STORY 532".

This screenshot shows a Product Backlog Item details page for "Hello World Web site". At the top, there is a tag bar with the word "page" followed by an "Add..." button, which is highlighted with an orange box. Below the tag bar, the item title "Hello World Web Site" is displayed in a blue box. Further down, there are sections for Iteration (Fabrikam Fiber Website) and Area (Fabrikam Fiber Website).

Tags that appear in the tag bar are already assigned to the work item. To unassign a tag, choose the x on the tag, [Web](#) .

#### **NOTE**

By default, all Contributors and Stakeholders of public projects are granted permissions to add new and existing tags. Stakeholders in private projects can add tags that are already defined, but not add new tags. To grant or restrict permissions to create new tags, you set the permission **Create tag definition** at the project-level. To learn more, see [Add administrators, set permissions at the project-level or project collection-level](#).

## Capture comments in the Discussion section

Use the **Discussion** section to add and review comments made about the work being performed.



The rich text editor tool bar displays below the text entry area when you click your cursor within each text box that can be formatted.



#### **NOTE**

There is no Discussion work item field. To query work items with comments entered in the Discussion area, you filter on the **History** field. The full content of the text entered into the Discussion text box is added to the History field.

### Mention someone, a group, work item, or pull request (, , or )

Choose one of these icons —, , or — to open a menu of recent entries you've made to mention someone, link to a work item, or link to a pull request. Or, you can simply type @, #, or ! to open the same menu.



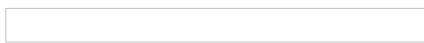
#### **NOTE**

This latest version of the rich text editor requires Azure DevOps Server 2019 Update 1 or later version.

Type a name, or enter a number and the menu list will filter to match your entry. Choose the entry you want to add. You can bring a group into the discussion by typing @ and the group name, such as a team or security group.

### Edit or delete a comment

If you need to edit or delete any of your discussion comments, choose **Edit** or choose the actions icon and then choose **Delete**.



#### **NOTE**

The edit/delete feature requires Azure DevOps Server 2019 Update 1 or later version.

After updating the comment, choose **Update**. To delete the comment, you'll need to confirm that you want to delete it.

A full audit trail of all edited and deleted comments is maintained in the **History** tab on the work item form.

Use the **@mention control** to notify another team member about the discussion. Simply type @ and their name. To reference a work item, use the **#ID control**. Type # and a list of work items that you've recently referenced will appear from which you can select.

To reference a work item, use the **#ID control**. Type # and a list of work items that you've recently referenced will appear from which you can select.

Note that you can't edit or delete comments once they've been entered.

#### IMPORTANT

For on-premises Azure DevOps Server, you must configure an SMTP server in order for team members to receive notifications.

## Add a reaction to a comment

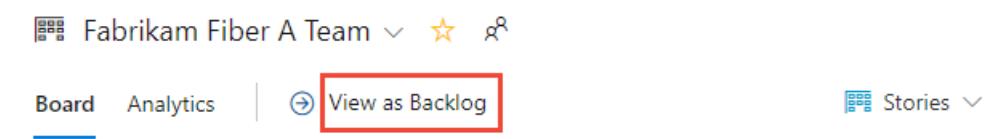
You can add one or more reactions to any comment. Choose a smiley icon at the upper-right corner of any comment or choose from the icons at the bottom of a comment next to any existing reactions. To remove your reaction, click the reaction on the bottom of your comment. The following shows an example of the experience of adding a reaction, as well as the display of reactions on a comment.



## Check the backlog and prioritized work

You can check the product backlog to see how the team has prioritized work. Backlog items appear in priority order. Work item types may include bugs depending on the settings made for the team.

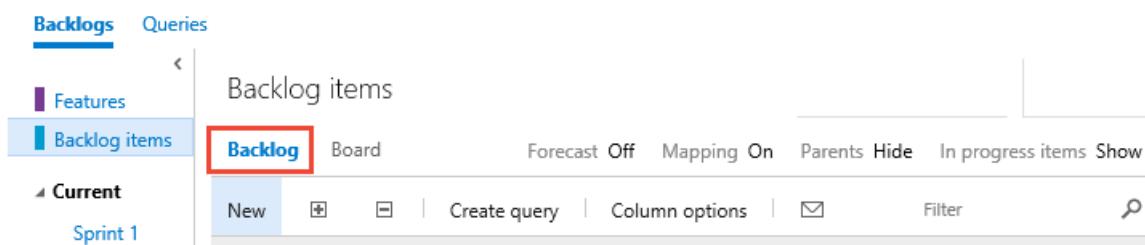
From the Kanban board, choose **View as backlog**.



From the Kanban board, choose **View as backlog**.



From the Kanban board, choose **Backlog**.



You should see the list of backlog items listed in priority order. You can add a backlog item which will be placed at the bottom of the list. With Stakeholder access, you can't re-prioritize work.

To view or edit a work item, select it and choose **Enter**.

## Find work assigned to you, or query for other work items

1. Choose **Boards > Work Items**, and then select **Assigned to me**.

Work Items

	with permissions	...	New	Fabrikam Fiber\Voice
Recently updated	hitecture changes		New	Fabrikam Fiber\Voice
Recently completed			In Progress	Fabrikam Fiber\Voice
Recently created				

You can focus on relevant items inside a project using one of the seven pivots as described next.

Additionally, you can filter and sort each pivot view. For details, see [View and add work items using the Work Items page](#).

2. To query for work items, see [View, run, or email a work item query](#).

1. Open **Work > Queries** and select **Assigned to me** to see the list of work items assigned to you.

ID	Work Item Type	Title	State
190	Bug	Simplify the search experience	New
191	Bug	Log-in button needs to be more prominent	New

**Bug 190: Simplify the search experience**

Simplify the search experience

2. Or, open any of the queries defined in the Shared Queries folder.

The screenshot shows the 'Work in progress' query results in Visual Studio Team Foundation Server 2015. The left sidebar has a 'Queries' tab selected. The main area displays a table of work items:

ID	Work Item Type	State	Remaining Work
164	Task	In Progress	8
165	Task	In Progress	8
166	Task	In Progress	6
167	Task	In Progress	2
168	Task	In Progress	2
169	Task	In Progress	1
170	Task	In Progress	4
173	Task	In Progress	2
174	Task	In Progress	1.5
181	Task	In Progress	1
186	Task	In Progress	1

3. And, you can create new queries or edit existing queries and save them under My Queries folder.

The screenshot shows the 'Work in Progress' query editor in Visual Studio Team Foundation Server 2015. The 'Editor' tab is selected. The filters section shows the following clauses:

- +  And/Or  Field Iteration Path  Operator Under  Value Fabrikam\Sprint 1
- +  And  Field Work Item Type  Operator In Group  Value Microsoft.TaskCategory
- +  And  Field State  Operator =  Value In Progress

[Add new clause](#)

## Features and functions available to Stakeholders

With Stakeholder access, users can create and modify work items and create and save queries. They have limited access to many of the Azure Boards features. They also can view and approve release pipelines and perform administrative tasks when granted administrative permissions or added to an administrative group.

### NOTE

Stakeholders that choose a feature that's not available to them may in some instances receive an error message indicating that they don't have permissions to complete a task.

### Public versus private feature access

Stakeholder access grants access to features differently depending on whether you're working from a private or

a public project. To learn more about public projects, see [What is a public project?](#).

SERVICE, APPLICATION, OR SETTING	PRIVATE PROJECT	PUBLIC PROJECT
Dashboards	Partial access	Full access
Wiki (Project wiki)	Partial access	Full access
Wiki (Code wiki)	No access	Full access
Azure Boards	Partial access	Full access
Azure Repos	No access	Full access
Azure Pipelines	Full access	Full access
Azure Test Plans	No access	No access
Azure Artifacts	Full access	Full access
Notifications	Full access	Full access
Semantic search	Full access	Full access
Project settings	Partial access	Partial access
Organization settings	Partial access	Partial access

### Features not available to users with Stakeholder access

If a Stakeholder needs access to one or more of the following features—which support the daily work of product owners, team leads, developers, testers, and project administrators—you need to provide them **Basic** access.

#### NOTE

Even if Stakeholders are explicitly granted permissions to some features, they are disallowed access to the feature due to their access level. Stakeholders that choose a feature that's not available to them receive an error message indicating that they don't have permissions to complete the task.

#### For Private projects:

- Change the priority of an item within a backlog or board
- Delete work items or move work items to another project
- Change fields on cards on a Kanban board or Taskboard, except for State field
- Drag-and-drop work items from a Backlog to the Mapping pane (parent a work item) or Planning pane (to assign to a sprint)
- Add new work item tags
- Create shared queries, view query charts, and modify the home page
- View Delivery Plans
- Access the full set of features under **Pipelines**, **Repos**, or **Test Plans**.

#### For Public projects:

- View Delivery Plans (a Marketplace extension)
- Access the full set of features under **Repos** or **Test Plans**.

- Change the priority of an item within a backlog or board
  - Delete work items or move work items to another project
  - Change fields on cards on a Kanban board or Taskboard, except for State field
  - Drag-and-drop work items from a Backlog to the Mapping pane (parent a work item) or Planning pane (to assign to a sprint)
  - Add new work item tags
  - Create shared queries, view query charts, and modify the home page
  - View Delivery Plans (a Marketplace extension)
  - Access the full set of features under **Pipelines, Repos, or Test Plans.**
- 
- Drag-and-drop work items from one column to another on a Kanban board or Taskboard to change the work item state
  - Change the priority of an item within a backlog or board
  - Delete work items or move work items to another project
  - Change fields on cards on a Kanban board or Taskboard
  - Drag-and-drop work items from a Backlog to the Mapping pane (parent a work item) or Planning pane (to assign to a sprint)
  - Add new work item tags
  - Create shared queries, view query charts, and modify the home page
  - View Delivery Plans (a Marketplace extension)
  - Access the full set of features under **Pipelines, Repos, or Test Plans.**
- 
- Drag-and-drop work items from one column to another on a Kanban board or Taskboard to change the work item state
  - Change the priority of an item within a backlog or board
  - Delete work items or move work items to another project
  - Change fields on cards on a Kanban board or Taskboard
  - Drag-and-drop work items from a Backlog to the Mapping pane (parent a work item) or Planning pane (to assign to a sprint)
  - Add new work item tags
  - Create shared queries, view query charts, and modify the home page
  - View Delivery Plans (a Marketplace extension)
  - Access the full set of features under **Code, Build and Release, or Test.**
- 
- Change the priority of an item within a backlog
  - Delete work items
  - Add work items, drag-and-drop work items, or change fields on cards on a Kanban board
  - Add new work item tags
  - Create shared queries, view charts, and modify dashboards
  - View Delivery Plans (a Marketplace extension)
  - Access the full set of features under **Code, Build and Release, or Test**
  - Participate in team rooms, which capture interactive, detailed conversations about the project.
- 
- Change the priority of an item within a backlog
  - Delete work items
  - Add work items, drag-and-drop work items, or change fields on cards on a Kanban board
  - Add new work item tags
  - Create shared queries, view charts, and modify the home page
  - Access the full set of features under **Code, Build and Release, or Test**

- Participate in team rooms, which capture interactive, detailed conversations about the project.

## Related articles

For a comparison chart of Stakeholder versus Basic access, see this [feature matrix](#). See also these quickstart guides:

- [Add work items](#)
- [Create your backlog](#)
- [Kanban quickstart](#)
- [Access levels](#)
- [Change access levels](#)

# Sign up, sign in to Azure DevOps

3/6/2021 • 4 minutes to read • [Edit Online](#)

## Azure DevOps Services

Learn how to sign up for Azure DevOps for free. Also, sign in with a Microsoft or GitHub account, create an organization or project, and invite your teammates.

Sign up for Azure DevOps to upload and share code in free, unlimited private Git repositories.

Then, connect to your favorite development tool like Eclipse, Xcode, Visual Studio, IntelliJ, or Android Studio to work on apps anytime, anywhere.

## Sign up with a personal Microsoft account

1. Select the sign-up link for [Azure DevOps](#).
2. Enter your email address, phone number, or Skype ID for your Microsoft account. If you're a Visual Studio subscriber and you get Azure DevOps as a benefit, use the Microsoft account associated with your subscription. Select **Next**.

If you don't have a Microsoft account, choose **Create one**. To learn more, see [create a Microsoft account](#).

3. Enter your password and select **Sign in**.

4. To get started with Azure DevOps, select **Continue**.

An organization is created based on the account you used to sign in. Sign in to your organization at any time, (  
<https://dev.azure.com/{yourorganization}> ).

You can rename and delete your organization, or change the organization location. To learn more, see the following articles:

- [Rename an organization](#)
- [Change the location of your organization](#)

If you signed in with an existing Microsoft account, your next step is to [Create a project](#). If you signed in with a newly created Microsoft account, then your project is automatically created and named after your account name. To learn more about managing projects, see [Manage projects](#).

## Sign up with a GitHub account

### IMPORTANT

If your GitHub email address is associated with an Azure AD-backed organization in Azure DevOps, you can't sign in with your GitHub account, rather you must sign in with your Azure AD account.

1. Select the sign-up link for [Azure DevOps](#), [Start free with GitHub](#). If you're already part of an Azure DevOps organization, select **Sign in to Azure DevOps**.

2. Select **Sign in with GitHub**.

If you have an account in session already, select **Use another account**. You're taken to GitHub sign-in where you can enter your GitHub user name or email address.

3. Enter your GitHub account credentials, and then select **Sign in**.

4. Select **Authorize Microsoft corporation**.

5. To get started with Azure DevOps, select **Continue**.

An organization is created based on the account you used to sign in. Sign in to your organization at any time, (<https://dev.azure.com/{yourorganization}>).

You can rename and delete your organization, or change the organization location. To learn more, see [Manage organizations](#).

#### **Enable GitHub invitations**

Creating a new Azure DevOps organization with your GitHub username turns on the Invite GitHub users policy by default. For existing organizations, your administrator can turn on this capability via [Organization settings](#) > [Policies](#) tab.

Once the setting is changed, sign out of Azure DevOps, and then from a fresh browser session, sign back in to the organization [dev.azure.com/{organizationName}](https://dev.azure.com/{organizationName}) or [organizationName.visualstudio.com](https://organizationName.visualstudio.com) with your GitHub credentials. You're now recognized as a GitHub user and the GitHub invitation experience is available to you.

For more information about GitHub authentication, see [FAQs](#).

## Create a project

If you signed up for Azure DevOps with a newly created Microsoft account (MSA), your project is automatically created and named based on your sign-in.

If you signed up for Azure DevOps with an existing MSA or GitHub identity, you're automatically prompted to create a project. You can create either a public or private project. To learn more about public projects, see [What is a public project?](#)

1. Enter information into the form provided, which includes a project name, description, visibility selection, initial source control type, and work item process.

See [choosing the right version control for your project](#) and [choose a process](#) for guidance.

2. When your project is complete, the welcome page appears.



## Invite team members

Give team members access to your organization by adding their email addresses or GitHub usernames to your organization. For GitHub user invitations, ensure you've [enabled the policy, \*Invite GitHub users\*](#) in [Organization settings > Policies](#) tab.

1. Sign in to your organization (<https://dev.azure.com/{yourorganization}>).

2. Select **Organization settings**.



3. Select **Users > Add new users**.



4. Enter the following information:

- **Users:** Enter the email addresses (Microsoft accounts) or [GitHub usernames](#) for the users. You can add several email addresses by separating them with a semicolon (;). An email address appears in red when it's accepted.
- **Access level:** Leave the access level as **Basic** for users who will contribute to the code base. To learn more, see [About access levels](#).
- **Add to project:** Select the project you want to add them to.
- **DevOps Groups:** Leave as **Project Contributors**, the default security group for users who will contribute to your project. To learn more, see [Default permissions and access assignments](#).

### NOTE

Add email addresses for [personal Microsoft accounts](#) and IDs for GitHub accounts unless you plan to use [Azure Active Directory \(Azure AD\)](#) to authenticate users and control organization access. If a user doesn't have a Microsoft or GitHub account, ask the user to [sign up](#) for a Microsoft account or a GitHub account.

5. When you're done, select **Add** to complete your invitation.

For more information about managing users and organization access, see [Add organization users for Azure DevOps](#).

## Choose your content version

This content supports a platform/version selector. Select from the Content version selector dropdown, located above the table of contents, to access the content that's specific to your version. The table of contents and content page refresh to show only that content specific to the selected version.

## Version

Azure DevOps Services ^

- Azure DevOps Services
- Azure DevOps Server 2020
- Azure DevOps Server 2019
- TFS 2018
- TFS 2017
- TFS 2015
- TFS 2013

## Next steps

[Add code to your Git repository](#)

[Plan and track work](#)

# Create an organization or project collection

5/22/2021 • 2 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Learn how to create an organization. An organization is used to connect groups of related projects, helping to scale up an enterprise. You can use a personal Microsoft account, GitHub account, or a work or school account. Use your work or school account to *automatically connect* your organization to your Azure Active Directory (Azure AD).

## NOTE

All organizations must be manually created via the web portal. We don't support automated creation of organizations. We do support automated organization configuration, project creation, and resource provisioning via REST API.

## Prerequisites

- Read and understand how to [Plan your organizational structure](#).
- Complete the following steps if you want to use only Microsoft accounts with your organization.

Without Azure AD, you're solely responsible for controlling organization access. And all users must sign in with their Microsoft account. [What are other differences?](#)

  - If you don't have a Microsoft account, you can create one when you sign up for Azure DevOps.
  - Use your Microsoft account if you don't need to authenticate users for an organization with [Azure AD](#). All users must sign in to your organization with a Microsoft account.
- Complete the following steps if you want to authenticate users and control organization access through your Azure AD.
  - You need a work or school account that's managed by your Azure AD. If you use Azure or Microsoft 365, you might have one already. If you don't, learn how to [sign up for Azure as an organization](#).
  - To use existing on-premises identities, see [use Azure AD Connect for integrating on-premises directories with Azure AD](#).
  - All users must be members in that directory to access your organization. To add users from other organizations, use [Azure AD B2B collaboration capabilities](#).
- Organization names must start with a letter or number, followed by letters, numbers or hyphens, and must end with a letter or number.

## IMPORTANT

Currently, you can only use letters from the English alphabet in your organization name.

## Create an organization

1. Sign in to [Azure DevOps](#).

2. Select **New organization**.

3. Confirm information, and then select **Continue**.

Congratulations, you're now an organization owner!

Sign in to your organization at any time, <https://dev.azure.com/{yourorganization}>.

## Create a project collection

A project collection is a container of projects. By grouping projects together, you can manage projects more efficiently and assign the same resources to those projects.

For more information about how to create a project collection, see [create a project collection](#).

## Next steps

[Create a project](#)

## Related articles

- [Rename your organization](#)
- [Change organization time-zone](#)
- [Change organization owner](#)
- [Delete your organization](#)
- [Resolve orphaned organization](#)

# Manage your project

4/21/2021 • 7 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

With most Azure DevOps Services, you can start using the service and configure resources as you go. No up-front work is required. Most settings define defaults.

As an organization Owner or a Project Administrator, there are a few tasks you might want to do to ensure a smooth operational experience. If you need to manage an organization with a large user base, consider additional tasks to structure your projects to support multiple teams or software development applications.

## Add users to your project

Ensure that all members of your organization or group are added to your organization and projects. For small groups, using [Microsoft Accounts](#) to add users to your organization and projects works fine.

Larger enterprises may want to consider using Azure Active Directory to manage permissions and user access. To learn more, see [About organization management](#).

Ensure that all members of your organization or group are added to your organization and project. Larger organizations may want to consider using Azure Active Directory to keep the maintenance of managing permissions and user access. Typically, you should install Azure Active Directory before installing TFS. To learn more, see the following articles.

- [Install Azure Active Directory Domain Services \(Level 100\)](#)
- [Step-By-Step: Setting up Azure Active Directory in Windows Server 2016](#)

To delegate the task of managing user access, add a user with Stakeholder or higher access to the [Project Collection Administrators group](#).

## Grant or restrict permissions

Access to features and functions is controlled by access-level assignments, permissions, and security groups. To quickly understand the defaults configured for your project, see [Default permissions and access](#).

### NOTE

If the **Project-Scoped Users well known group to hide settings** preview feature is enabled for the organization, users added to the **Project-Scoped Users** group won't be able to access projects that they haven't been added to. To learn more, see [About projects and scaling your organization](#), [Project-scoped Users group](#).

To delegate specific tasks to others, add them to a built-in or custom security group or add them to a specific role. To learn more, see the following articles.

- [Grant or restrict access to select features and functions](#)
- [Set permissions at the project level or project collection level](#)

To learn more about permissions and security, review the following articles:

- [About security and identity](#)

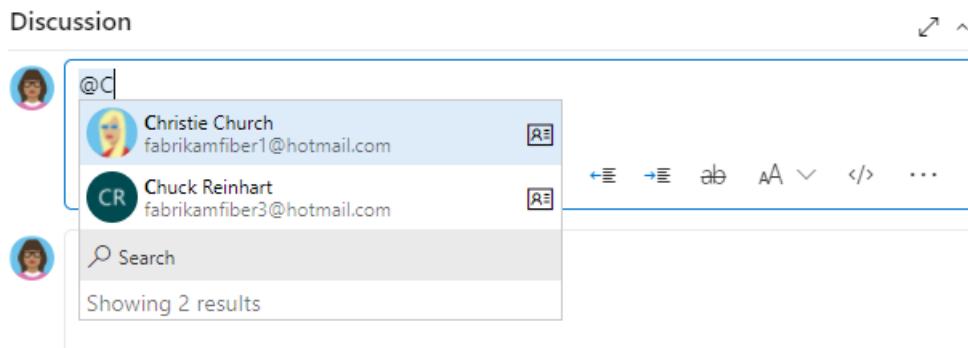
- [About permissions and groups](#)
- [About security roles](#)
- [About access levels](#)

## Limit identity search and selection

For organizations that manage their users and groups using Azure Active Directory (Azure AD), people pickers provide support for searching all users and groups added to Azure AD, not just those added to your project. People pickers support the following Azure DevOps functions:

- Selection of a user identity from a work tracking identity field such as **Assigned To**
- Selection of a user or group using **@mention** in a work item discussion or rich-text field, a pull request discussion, commit comments, or changeset or shelveset comments
- Selection of a user or group using **@mention** from a wiki page

As shown in the following image, you simply start typing into a people picker box until you find a match to a user name or security group.



To limit the identity selection to just those users and groups added to a project, perform the following procedure for your organization and projects.

1. Enable the **Limit user visibility for projects** preview feature for the organization. To learn how, see [Manage or enable features](#).
2. Add the users to your project(s) as described in [Add users to a project or team](#). Users added to a team are automatically added to the project and team group.
3. Open **Organizations Settings > Security > Permissions** and choose **Project-SScoped Users**. Choose the **Members** tab. Add all users and groups that you want to scope to the project(s) you've added them to. To learn more, see [Set permissions at the project- or collection-level](#). The **Project-SScoped Users** group only appears under the **Permissions > Groups** once **Limit user visibility for projects** preview feature is enabled.

## Share your project vision

Each project has a summary page that's useful for sharing information through README files. Or, redirect users to a project Wiki. For users who are new to your project, we recommend that you [set up your project summary page](#) or [provision a Wiki](#). Use these features to share established processes and procedures for your project.

Each project has a summary page that's useful for sharing information through README files. For users who are new to your project, we recommend that you [set up your project summary page](#). Use this feature to share established processes and procedures for your project.

## Remove unused services

To simplify the web portal user interface, you can disable select services. For example, if you use a project only to log bugs, then disable all services except for **Boards**.

This example shows that **Test Plans** is disabled:

The screenshot shows the 'Project Settings > Overview' page. On the left, a sidebar lists 'General' (selected), 'Teams', 'Security', 'Notifications', 'Service hooks', 'Dashboards', 'Boards' (selected), 'Project configuration', 'Team configuration', and 'GitHub connections'. On the right, under 'Azure DevOps services', there are six service items with toggle switches: 'Boards' (On), 'Repos' (On), 'Pipelines' (On), 'Artifacts' (On), and 'Test Plans' (Off). Each item has a brief description below it.

Azure DevOps Service	Status
Boards	On
Repos	On
Pipelines	On
Artifacts	On
Test Plans	Off

## Set DevOps policies

Set policies to support collaboration across your teams, secure your projects, and automatically remove obsolete files. To set policies, review the following articles:

- [Change application access policies for your organization](#)
- [Manage branch policies](#)
- [Add Team Foundation Version Control \(TFVC\) check-in policies](#)
- [Set build and release pipeline retention policies](#)
- [Set test retention policies](#)
- [Manage branch policies](#)
- [Add TFVC check-in policies](#)
- [Set build and release pipeline retention policies](#)
- [Set test retention policies](#)

## Define area and iteration paths to track work

If you support several products, you can assign work items by feature area by defining [area paths](#). To assign work items to specific time intervals, also known as sprints, you configure [iteration paths](#). To use the Scrum tools—sprint backlogs, taskboards, and team capacity—you need to configure several sprints. For an overview, see [About areas and iteration paths](#).

The screenshot shows two side-by-side sections of the Project Settings page for a project named 'Fabrikam Fiber'.

**Iterations Section:**

- Project Settings:** General, Overview, Teams, Permissions, Notifications, Service hooks, Dashboards.
- Boards:** Project configuration (selected), Team configuration, GitHub connections.
- Iterations:** A red box highlights the 'Iterations' tab under Boards. A tooltip says: "This project is currently using the Agile process. To customize your work item types, go to the process customization page." Below it, a table lists iterations for 'Fabrikam Fiber':

Iterations	Start Date	End Date
Release 1	...	
Sprint 1	1/6/2020	1/10/2020
Sprint 2	1/13/2020	1/17/2020
Sprint 3	1/20/2020	1/24/2020
Release 2		

**Areas Section:**

- Project Settings:** General, Overview, Teams, Permissions, Notifications, Service hooks, Dashboards.
- Boards:** Project configuration (selected), Team configuration, GitHub connections.
- Areas:** A red box highlights the 'Areas' tab under Boards. A tooltip says: "This project is currently using the Agile process. To customize your work item types, go to the process customization page." Below it, a table lists areas and teams:

Areas	Teams
Fabrikam Fiber	Customer Service
Phone	Customer Service
Voice	
Web	...

## Customize work-tracking processes

All work-tracking tools are available immediately after you create a project. Often, one or more users may want to customize the experience to meet one or more business needs. Processes are easily customized through the user interface. However, you may want to establish a methodology for who manages the updates and evaluates requests.

To learn more, see the following articles:

- [About process customization and inherited processes](#)
- [Customize a project](#)
- [Add and manage processes](#)

All work-tracking tools are available immediately after you create a project. Often, one or more users may want to customize the experience to meet one or more business needs. But, you may want to establish a methodology for who manages the updates and evaluates requests.

To learn more, see [On-premises XML process model](#).

## Review and update notifications

A number of notifications are predefined for each project you add. Notifications are based on subscription rules. Subscriptions arise from the following areas:

- [Out-of-the-box or default subscriptions](#).
- [Team, organization, and collection-level notifications](#), managed by a team administrator or member of the Project Collection Administrators group.
- [Project notifications](#), managed by a member of the Project Administrators group.

If users believe they're getting too many notifications, direct them to [opt out of a subscription](#).

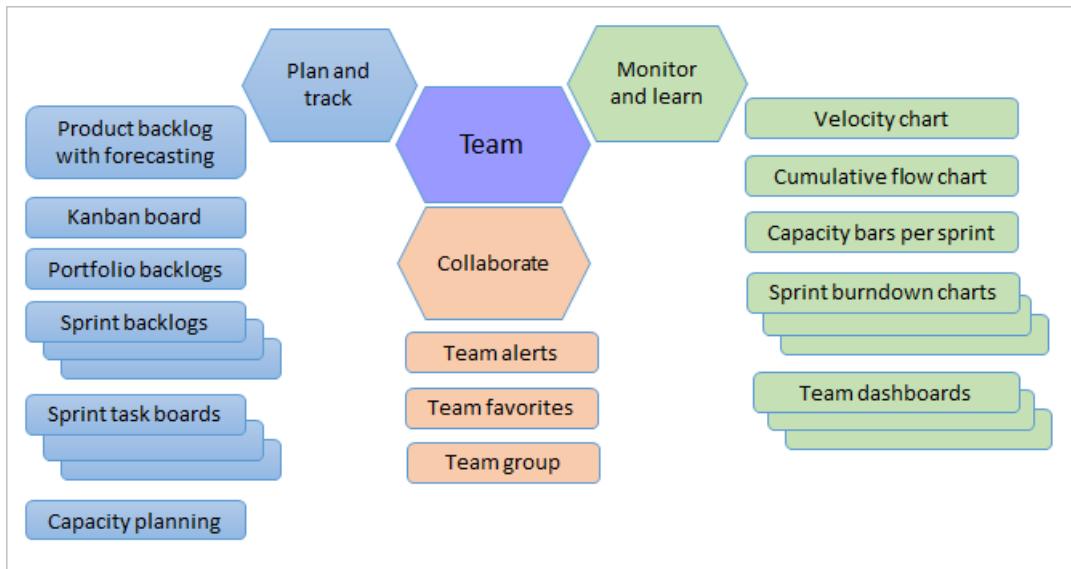
Description	Type	Notifies	State
<b>Build</b>			
 Build completes Notifies you when a build you queued or that was queued for you compl...	Build completed (any project)	 You	<input checked="" type="checkbox"/> On
<b>Code (Git)</b>			
 Pull request reviewers added or removed Notifies you when you are added to a pull request or when a user is add...	Pull request (any project)	 You	<input checked="" type="checkbox"/> On
 Pull request completion failures Notifies you when a pull request you created fails to complete	Pull request (any project)	 You	<input checked="" type="checkbox"/> On
 Pull request changes Notifies you when changes are made to a pull request you created or are...	Pull request (any project)	 You	<input checked="" type="checkbox"/> On
 A comment is left on a pull request Notifies you about comments made to a pull request you created or a di...	Pull request comment (any project)	 You	<input checked="" type="checkbox"/> On

## Configure an SMTP server

In order for team members to receive notifications, [you must configure an SMTP server](#).

## Add teams to scale your organization

We recommend that you add teams as your organization grows. Each team gets [access to their own set of customizable Agile tools](#).



To learn more, see the following articles:

- [About projects and scaling your organization](#)
- [Add a team, move from one default team to several teams](#)
- [Add a team administrator](#)

## Install and manage extensions

An extension is an installable unit that adds new capabilities to your projects. Azure DevOps extensions support the following functions:

- Planning and tracking of work items, sprints, scrums, and so on
- Build and release flows

- Code testing and tracking
- Collaboration among team members

For example, to support [code search](#), install the [Code Search extension](#).

You want to tell your users about extensions and that they can [request an extension](#). To install and manage extensions, you must be an organization Owner, a member of the Project Collection Administrators group. Or, you can get added to the [Manager role for extensions](#).

## Set up billing

All organizations can add up to five users with Basic access and unlimited users with Stakeholder access. If you need to add more users or pay for additional services or extensions, [set up billing](#).

## Next steps

[Share your project vision](#)

## Related articles

- [Project and team quick reference](#)
- [Security & identity](#)
- [Organization management](#)
- [About user, team, project, and organization-level settings](#)
- [Project and team quick reference](#)
- [Security & identity](#)
- [Organization management](#)
- [About user, team, project, and organization-level settings](#)
- [TFS administration](#)

# Add users to a project or team

4/17/2021 • 15 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Learn how to add users to a project or specific team. For anyone to access a project, they must be added to one of the default security groups or a custom group. Usually you add them to the Contributors group. For a quick look at what permissions are assigned to the default groups, see [Permissions and access](#).

The easiest way to add a number of users to a project is to add groups defined in [Azure Active Directory \(Azure AD\)](#) or [Active Directory \(AD\)](#).

## IMPORTANT

If you're adding users to an organization in Azure DevOps Services and you don't use Azure AD, then you need to first [add their "personal" Microsoft accounts to your organization or project](#).

Once you've added a user to an organization or project, their user identity becomes searchable from an identity field, such as the Assigned To field, or from the security and permission dialogs. After you've added them to one project or team, you can add them to additional projects or teams using the procedures provided in this article. For more information about managing users and organization access, see [About permissions, access, and security groups](#), [Active Directory and Azure Active Directory security groups](#).

## IMPORTANT

On-premises Azure DevOps instances automatically reference user identities defined in the Active Directory or Windows workgroup of the local network. You can add security groups defined in Active Directory or a workgroup to a collection. For more information, [About permissions, access, and security groups](#), [Active Directory and Azure Active Directory security groups](#).

Once you've added security groups to a collection or project, user identities defined with that group become searchable from an identity field, such as the Assigned To field, or from the security and permission dialogs. After you've added them to one project, you can add them to additional projects and teams using the procedures provided in this article.

## Prerequisites

You can add users to a project or team, add projects to organizations, and add teams to projects.

- You must have an organization and project. If you don't have a project yet, [create one](#).
- To add users to an organization, you must be a member of the [Project Collection Administrators group](#). Organization owners are automatically members of this group.
- To add users to a project, you must be a member of the [Project Administrators or Project Collection Administrators groups](#). Any new users added to a project are automatically added to the organization.
- To add users to a team, you must be a [Team Administrator](#), or you must be a member of one of the administrative groups. Any new users added to a team are automatically added to the organization.

#### **NOTE**

When the organization policy, **Allow team and project administrators to invite new users**, is disabled, Team and Project Administrators can't add users who are not already in the organization to a team or project. Project Collection Administrators can add users whether this policy is on or off. For more information, see [Restrict invitations from Project and Team Administrators](#). For an overview of the methods supported for adding users to an organization, see [About organization management, Add and manage user access](#).

You add users to a project or team. You add projects to project collections, and you add teams to projects.

- If you don't have a project yet, [create one](#).
- To add users to a project, you must be a member of the [Project Administrators or Project Collection Administrators groups](#).
- To add users to a team, you must be [added as a team administrator](#), or you must be a member of one of the administrative groups.

Once users have been added to a project, you can browse for their display name or user name (email alias). Also, you can [add them to a specific team](#). To add a team, see [Add a team](#).

## Add users to a project

If you're adding a user to Azure DevOps for the first time, see [Add account users for Azure DevOps](#).

#### **NOTE**

To enable the new user interface for the Project Permissions Settings Page, see [Enable preview features](#).

- [Preview page](#)
- [Current page](#)
- [Azure DevOps CLI](#)

1. Open the web portal and choose the project where you want to add users or groups. To choose another project, see [Switch project, repository, team](#).
2. Choose **Project settings**, and then **Permissions**.

The screenshot shows the Azure DevOps 'Permissions' page for the 'FabrikamFiber' project. The left sidebar has a tree view with 'General' expanded, showing 'Overview', 'Teams', 'Notifications', 'Service hooks', 'Dashboards', and 'Boards'. 'Teams' is expanded, and 'Permissions' is selected and highlighted with a red box. The main area has a 'Permissions' section with tabs for 'Groups' and 'Users'. Under 'Groups', there are two entries: 'Build Administrators' (BA) and 'Contributors' (C). Both entries have a 'Description' column.

Name	Description
BA Build Administrators	Member create, modify and delete build definitions and manage queued and completed builds.
C Contributors	Member add, modify, and delete items within the team project.

3. Under **Groups**, choose one of the following options:

- **Readers**: To add users who require read-only access to the project, choose.
- **Contributors**: To add users who contribute fully to this project or who have been granted Stakeholder access.
- **Project Administrators**: To add users who need to administrate the project. To learn more, see [Set permissions at the project-level or project collection-level](#).

Here we choose the **Contributors** group.

The screenshot shows the 'Permissions' page with the 'Groups' tab selected. It lists two groups: '[Fabrikam Fiber]\Build Administrators' and '[Fabrikam Fiber]\Contributors'. The '[Fabrikam Fiber]\Contributors' entry is highlighted with a red box.

Name	Description
[Fabrikam Fiber]\Build Administrators	Members of this group can create, modify and delete build definitions and manage queued and completed builds.
[Fabrikam Fiber]\Contributors	Members of this group can add, modify, and delete items within the team project.

4. Next, choose the **Members** tab.

The default team group, and any other teams you add to the project, get included as members of the **Contributors** group. Add a new user as a member of a team instead, and the user automatically inherits Contributor permissions.

**TIP**

Managing users is much easier [using groups](#), not individual users.

5. Choose **Add** to add a user or a user group.

**C** Contributors

Members of this group can add, modify, and delete items within the team project.

Permissions Members Member of Settings

Search users and groups

**Add**

Name	Type	Username or scope
P	Phone	[Fabrikam Fiber]

6. Enter the name of the user account into the text box. You can enter several identities into the text box, separated by commas. The system automatically searches for matches. Choose the match(es) that meets your requirements.

**Invite members to Contributors**

Search and add users and/or groups to your group

Add users and/or groups

Ch

c Ch

Christie Church  
fabrikamfiber1@hotmail.com

CR Chuck Reinhart  
fabrikamfiber3@hotmail.com

Cancel Save

**NOTE**

The first time you add a user or group to Azure DevOps, you can't browse to it or check the friendly name. After the identity has been added, you can just enter the friendly name.

Choose **Save** when done.

7. You may customize user permissions for other functionality in the project. For example, in [areas and iterations](#) or [shared queries](#).

**NOTE**

Users that have limited access, such as Stakeholders, won't be able to access select features even if granted permissions to those features. To learn more, see [Permissions and access](#).

Choose the **Current page** tab for information on adding a user to a project.

1. Open the web portal and choose the project where you want to add users or groups. To choose another project, see [Switch project, repository, team](#).
2. Choose **Project Settings** and then **Security**.

*To see the full image, select to expand.*

The screenshot shows the 'Project Settings' page for the 'Fabrikam Fiber' project. On the left, there's a sidebar with icons for Overview, Work, Code, Build and release, and Packages. The 'Project settings' icon at the bottom is highlighted with a red box and a '1'. The main area shows a 'General' section with options like Overview, Services, Teams, Security (which is highlighted with a red box and a '2'), Notifications, Service hooks, and Dashboards. Below this is a 'Boards' section, followed by 'Build and release', 'Code', 'Test', and 'Extensions'. A 'Create group' dialog is open on the right, titled 'Create group'. It has a 'Filter users and groups' input field. Under 'Teams', it lists: Customer Service, Email, Fabrikam Fiber Team, Management team, Phone, Voice, and Web. Under 'Azure DevOps Groups', it lists: Build Administrators, Contributors, Deployment Group Administrators, Disallow access group, Endpoint Administrators, Endpoint Creators, Project Administrators, Project Collection Valid Users, and Security Service Group.

3. Under **Groups**, choose one of the following options:

- **Readers**: To add users who require read-only access to the project, choose.
- **Contributors**: To add users who contribute fully to this project or who have been granted Stakeholder access.
- **Project Administrators**: To add users who need to administrate the project. To learn more, see [Set permissions at the project-level or project collection-level](#).

4. Next, choose the **Members** tab.

Here we choose the **Contributors** group.

The screenshot shows the 'Contributors' group members page. On the left, there's a sidebar titled 'Create group' with sections for 'Teams' and 'Azure DevOps Groups'. Under 'Teams', 'Contributors' is highlighted with a red box. The main area has tabs for 'Permissions', 'Members' (which is selected and highlighted with a red box), and 'Member of'. Below these are buttons for '+ Add...', a refresh icon, and 'Search'. A table lists members with columns for 'Display Name', 'Username Or Scope', and 'Remove'. The table includes entries for 'Customer Service', 'Fabrikam Fiber Team', 'Management team', 'Phone', 'Voice', 'Web', and 'Jia-hao Tseng'.

The default team group, and any other teams you add to the project, get included as members of the **Contributors** group. Add a new user as a member of a team instead, and the user automatically inherits Contributor permissions.

**TIP**

Managing users is much easier [using groups](#), not individual users.

5. Choose **+ Add** to add a user or a user group.
6. Enter the name of the user account into the text box. You can enter several identities into the text box, separated by commas. The system automatically searches for matches. choose the match(es) that meets your requirements.

The screenshot shows the 'Add users and groups' dialog box. It has a title bar with a close button. The main area is titled 'Add users and groups' and contains a sub-instruction: 'To add users or groups to this group, just type their sign-in addresses or group aliases'. Below this is a 'User or group' input field containing 'Chris'. A search result card for 'Christie Church' is displayed, showing a small profile picture, the name 'Christie Church', the email 'fabrikamfiber1@hotmail.com', and a copy icon. At the bottom of the search results, it says 'Showing 1 result'. At the very bottom of the dialog are 'Save changes' and 'Cancel' buttons.

#### NOTE

The first time you add a user or group to Azure DevOps, you can't browse to it or check the friendly name. After the identity has been added, you can just enter the friendly name.

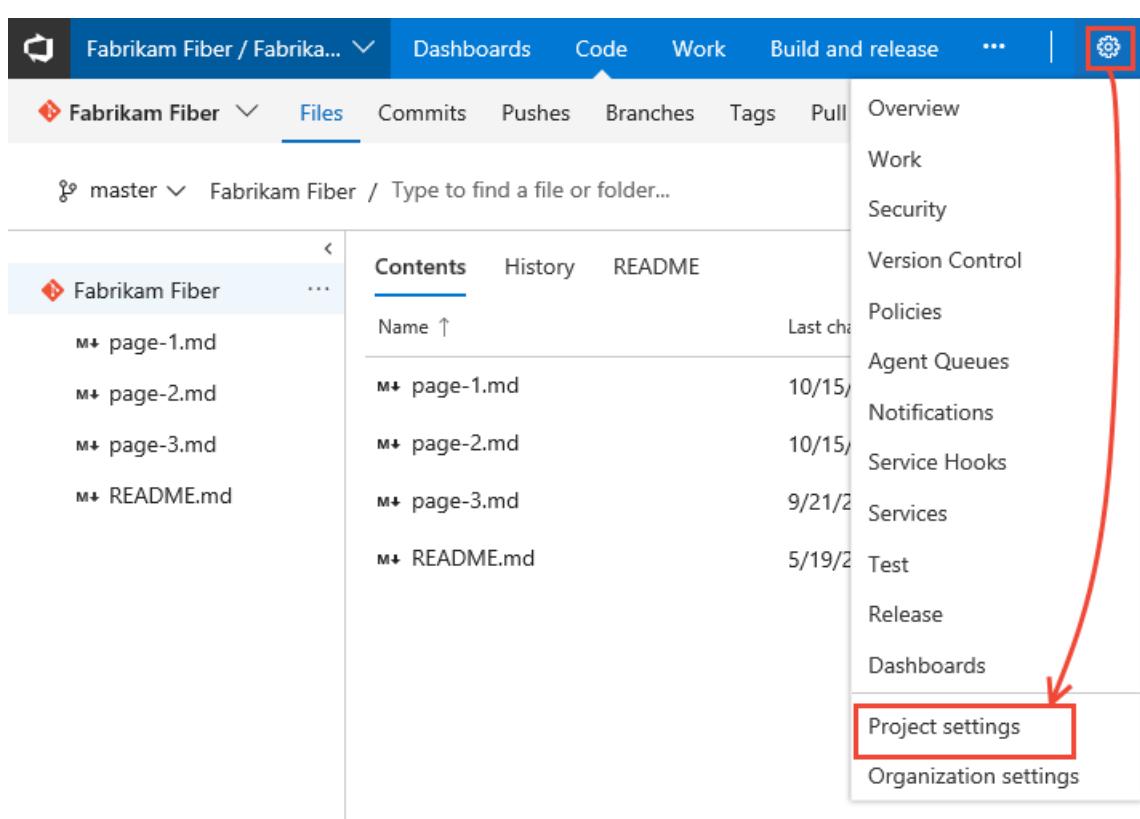
Choose **Save changes** when complete.

7. (Optional) You can customize a user's permission for other functionality in the project. For example, in [areas and iterations](#) or [shared queries](#).

#### NOTE

Users that have limited access, such as Stakeholders, won't be able to access select features even if granted permissions to those features. To learn more, see [Permissions and access](#).

1. Open the web portal and choose the project where you want to add users or groups. To choose another project, see [Switch project, repository, team](#).
2. Choose the gear icon to open the administrative context.



3. Choose **Security** and under **Groups**, choose one of the following options:

- **Readers**: To add users who require read-only access to the project, choose.
- **Contributors**: To add users who contribute fully to this project or who have been granted Stakeholder access.
- **Project Administrators**: To add users who need to administrate the project. To learn more, see [Set permissions at the project-level or project collection-level](#).

4. Next, choose the **Members** tab.

Here we choose the Contributors group.

The screenshot shows the 'Contributors' group page in the Azure DevOps interface. On the left, there's a sidebar with sections for 'Teams' (Customer Service, Fabrikam Fiber Team, Management team, Phone, Voice, Web) and 'Azure DevOps Groups' (Build Administrators, Contributors, Project Administrators, Project Valid Users, Readers). The 'Contributors' item under 'Azure DevOps Groups' is highlighted with a red box. The main area shows a table of members:

Display Name	Username Or Scope	
Customer Service	[Fabrikam Fiber]	Remove
Fabrikam Fiber Team	[Fabrikam Fiber]	
Management team	[Fabrikam Fiber]	
Phone	[Fabrikam Fiber]	
Voice	[Fabrikam Fiber]	
Web	[Fabrikam Fiber]	
Jia-hao Tseng	fabrikamfiber9@hotmail.com	

#### TIP

Managing users is much easier [using groups](#), not individual users.

By default, the default team group and any other teams you add to the project, are included as members of the **Contributors** group. Add a new user as a member of a team instead, and the user automatically inherits Contributor permissions.

5. Choose **+ Add** to add a user or a user group.
6. Enter the name of the user account into the text box. You can enter several identities into the text box, separated by commas. The system automatically searches for matches.

The screenshot shows the 'Add users and groups' dialog box. The 'User or group' input field contains 'Chris'. Below it, a search result for 'Christie Church' is displayed, showing her profile picture, name, and email address (fabrikamfiber1@hotmail.com). A small info icon is next to the email address. At the bottom of the dialog, there are 'Save changes' and 'Cancel' buttons.

#### NOTE

The first time you add a user or group to Azure DevOps, you can't browse to it or check the friendly name. After the identity has been added, you can just enter the friendly name.

7. (Optional) You can customize user permissions for other functionality within the project, such as [areas](#) and [iterations](#) or [shared queries](#).

#### NOTE

Users that have limited access, such as Stakeholders, won't be able to access select features even if granted permissions to those features. To learn more, see [Permissions and access](#).

## Add users to a team

Several Agile tools, like capacity planning, team alerts, and dashboard widgets are team-scoped. That is, they automatically reference the user accounts added as members of a team to support planning activities or sending alerts. To learn more, see [About teams and Agile tools](#).

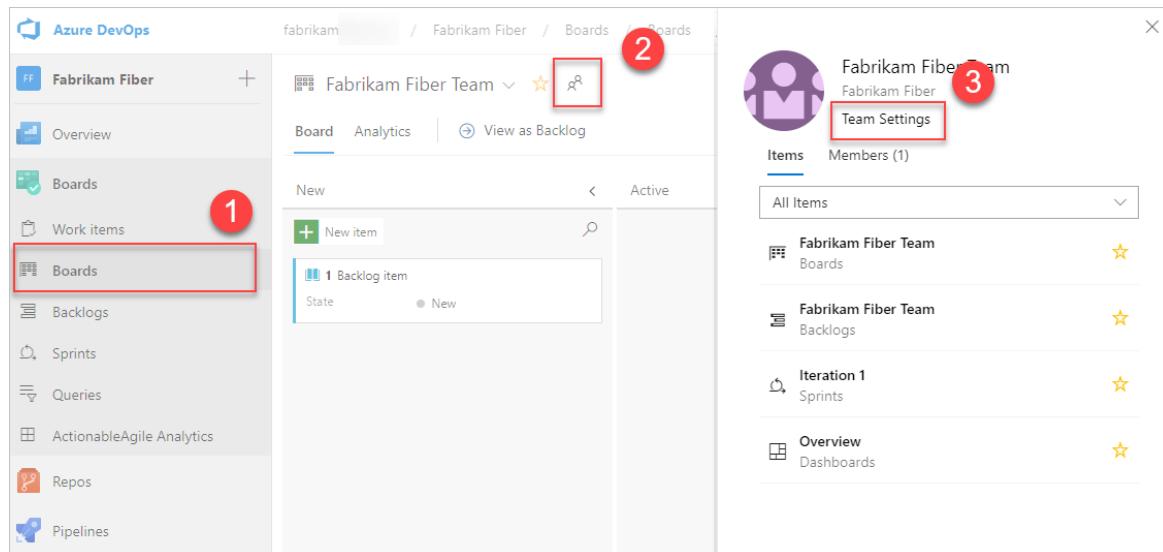
#### NOTE

To enable the preview feature, [New Teams Page](#), see [Enable preview features](#).

- [Preview page](#)
- [Current page](#)
- [Azure DevOps CLI](#)

1. Open a backlog or board for a team and choose the  team profile icon. Then choose **Team Settings**.

Here we open the Board for the Web team and from there the team profile.



2. If you need to switch the team context, use the team selector within the breadcrumbs.

The screenshot shows the Microsoft Teams settings interface. The top navigation bar has 'Settings' / 'Teams' / 'Fabrikam Fiber Team'. The 'Fabrikam Fiber Team' item is highlighted with a red box. Below the navigation is a 'Team Profile' section with a purple icon. To the right is a list of team members: 'Customer Service (Fabrikam Fiber)' with a yellow star icon, and 'Fabrikam Fiber Team (Fabrikam F...)' which is currently selected. Below this is a row with '+ Add...' and a refresh icon. A table follows with columns for 'Display Name' and 'Username Or Scope'. It shows 'Jamal Hartnett' with the email 'fabrikamfiber4@hotmail.com'. There are also 'Name' (Fabrikam Fiber Team) and 'Description' (The default project team.) fields.

3. Choose Add.

This screenshot shows the Microsoft Teams settings interface again. The top navigation bar has 'Settings' / 'Teams' / 'FabrikamFiber Team'. The 'FabrikamFiber Team' item is highlighted with a red box. Below the navigation is a 'Team Profile' section with a purple icon. To the right is a list of team members: 'FabrikamFiber Team' (highlighted with a red box). Below this is a row with '+ Add...' and a refresh icon. A table follows with columns for 'Display Name' and 'Username Or Scope'. It shows 'Jamal Hartnett' with the email 'fabrikamfiber4@hotmail.com'. There are also 'Name' (FabrikamFiber Team) and 'Description' (The default project team.) fields. On the left side of the screen, there are links for 'Administrators', '+ Add', 'Manage other settings for this team', 'Notifications', 'Dashboards', and 'Iterations and areas'.

4. Enter the sign-in addresses or display name for each account you want to add. Add them one at a time or all at the same time. You can enter several identities into the text box, separated by commas.

**Add users and groups**

To add users or groups to this group, just type their sign-in addresses or group aliases

User or group

 Christie Church  
fabrikamfiber1@hotmail.com

Showing 1 result

**Save changes** **Cancel**

**TIP**

You must enter user and group names one at a time. However, after entering a name, the account is added to the list, and you can enter another name in the Identities text box before choosing to save your changes.

You may need to choose the  refresh icon to see your updates.

5. To remove members, return to this page, highlight the user name and choose **Remove**.

FabrikamFiber Team

**Members**

**Add...** | 

Display Name	Username Or Scope	Action
Jamal Hartnett	fabrikamfiber4@hotmail.com	<a href="#">Remove</a>

**NOTE**

To remove a team administrator as a team member, you must first remove them as an administrator.

6. To add an account as a team administrator, choose **Add** located in the Team Profile page. For details, see [Add a team administrator](#)

Choose the **Current page** tab for information on adding a user to a team.

## Add users or groups to an access level

For on-premises deployments, you may need to set the access level for a user or group, particularly if those groups don't belong to the default access level. To learn more, see [Change access levels](#).

## Add users or groups to SQL Server Reports

If your on-premises deployment is integrated with SQL Server Reports, you need to manage membership for those products separately from their websites. See [Grant permissions to view or create SQL Server reports in Azure DevOps](#).

# Add users or groups to SharePoint or SQL Server Reports

If your on-premises deployment is integrated with a SharePoint product or SQL Server Reports, you need to manage membership for those products separately from their websites.

- [Set SharePoint site permissions](#)
- [Grant permissions to view or create SQL Server reports in Azure DevOps Server](#)

## Next steps

[Add administrators or set permissions at the project or collection level](#)

## Related articles

- To view permissions for yourself or another user, see [View permissions](#).
- [Set Git repository permissions](#)
- [Set TFVC repository permissions](#)
- [Set Git branch permissions](#)
- [Set build and release permissions](#)
- [Set permissions and access for work tracking](#)

# Manage and configure team tools

3/6/2021 • 7 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

As a team administrator, you can customize your backlogs and board to best meet how your team works. If you need to have a team created, request a member of your Project Administrators group do so. It only takes a minute to add a new team. Team settings are managed by the team administrator role. Users assigned as team administrator can configure and manage all team tools.

Team administrators should do the following tasks:

- [Add team members](#)
- [Add another team administrator](#)
- [Configure areas and iteration paths](#)
- [Configure backlogs, boards, and general settings](#)

Also, consider the following optional tasks:

- [Configure and manage team dashboards](#)
- [Configure team notifications](#)

## Prerequisites

- To perform any team configuration task, you need to be added as a team administrator for the team to be modified, or be a member of the [Project Administrator or Project Collection Administrators group](#).
- To add a team, you must be a member of the [Project Administrator or Project Collection Administrators group](#). For more information, see [Add teams](#).

### NOTE

For guidance on configuring and customizing your project and teams to support your business needs, review [Configuration and customization of Azure Boards](#).

## Open your team profile

Open your team profile to quickly access items defined for your team.

1. Sign in to your organization (<https://dev.azure.com/{yourorganization}>), and then open your project.
2. Select **Project settings > Teams > your team name**.

The screenshot shows the 'Project Settings' page for the 'Fabrikam Fiber' project in Azure DevOps. A red circle labeled '1' highlights the 'Project settings' button at the bottom left of the sidebar. A red circle labeled '2' highlights the 'Teams' section in the left-hand navigation menu. A red circle labeled '3' highlights the 'Fabrikam Fiber Team' card in the main 'Teams' list, which includes a 'Team Name' (Fabrikam Fiber Team), a 'Members' count (2), and a 'New team' button.

## Add users to a team

Several tools, such as capacity planning, team alerts, and dashboard widgets, are team-scoped. These tools automatically reference the users that are as members of a team to support planning activities or sending alerts.

To add users to a team, see [Add users to a project or specific team](#).

The screenshot shows the 'Fabrikam Fiber Team' settings page. On the left, there's a 'Team Profile' section with a purple icon, a 'Name' (Fabrikam Fiber Team), a 'Description' (The default project team.), and sections for 'Administrators' (Jamal Hartnett) and 'Members'. On the right, there's a 'Members' section with a table. A red box highlights the '+ Add...' button. The table lists members with their display names, user names, and email addresses.

Display Name	Username Or Scope
Security Service Group	[FabrikamFiber01]
Fabrikam Fiber Build Servic...	Build\99612ac2-9c5f-4191-9e8...
Jamal Hartnett	fabrikamfiber4@hotmail.com

All members of a team can favorite team artifacts and define work item templates. For more information, see:

- [Set personal or team favorites](#)

- Use templates to add and update work items.

If team members don't have access to all the features they want, make sure they have [the permissions needed for those features](#).

## Add an administrator

When you add a team to a project, a Project Administrator should [add one or more team administrators](#).

The screenshot shows the 'Fabrikam Fiber Team' settings page. On the left, there's a sidebar with 'Team Profile' information: Name (Fabrikam Fiber Team), Description (The default project team.), Administrators (Jamal Hartnett, with a red 'X' icon), and three buttons: '+ Add' (highlighted with a red box), 'Manage other settings for this team', 'Notifications', 'Dashboards', and 'Iterations and areas'. On the right, under 'Members', there's a table:

Display Name	Username Or Scope
Security Service Group	[FabrikamFiber01]
Fabrikam Fiber Build Servic...	Build\99612ac2-9c5f-4191-9e8...
Jamal Hartnett	fabrikamfiber4@hotmail.com

## Configure team areas and iterations

Many Agile tools depend on the area and iteration paths that are configured for the team. To learn more about configuring team areas and iterations, see [About teams and Agile tools](#).

Once project administrators [add area paths](#) and [iteration paths](#) for a project, team administrators can select the area and iteration paths associated with their team. These settings affect many Agile tools available to the team.

**Team Profile**



Name  
Fabrikam Fiber Team

Description  
The default project team.

Administrators  
Jamal Hartnett X

+ Add

Manage other settings for this team

**Iterations and areas**

Fabrikam Fiber Team

**Members**

+ Add... |  

Display Name	Username Or Scope
 Security Service Group	[FabrikamFiber01]
 Fabrikam Fiber Build Servic...	Build\99612ac2-9c5f-4191-9e8...
 Jamal Hartnett	fabrikamfiber4@hotmail.com

Settings include making the following associations for each team:

- **Select team Area Paths**  
Can select the default area path(s) associated with the team. These settings affect many Agile tools available to the team.
- **Select team Iteration Paths or sprints** Can select the default area path(s) associated with the team. These settings affect many Agile tools available to the team.

For more information, see [Define area paths and assign to a team](#) and [Define iteration paths and configure team iterations](#).

## Configure team backlogs, boards, and general settings

Team administrators can choose which backlog levels are active for a team. For example, a feature team may choose to show only the product backlog and a management team may choose to show only the feature and epic backlogs. Also, administrators can choose whether bugs are treated similar to user stories and requirements or as tasks.

Team administrators can also choose which days are non-working days for the team. Sprint planning and tracking tools automatically consider days off when calculating capacity and sprint burndown.

You can configure most of your team settings from the common configuration dialog.

### NOTE

The common configuration Settings dialog is available for TFS 2015.1 and later versions.

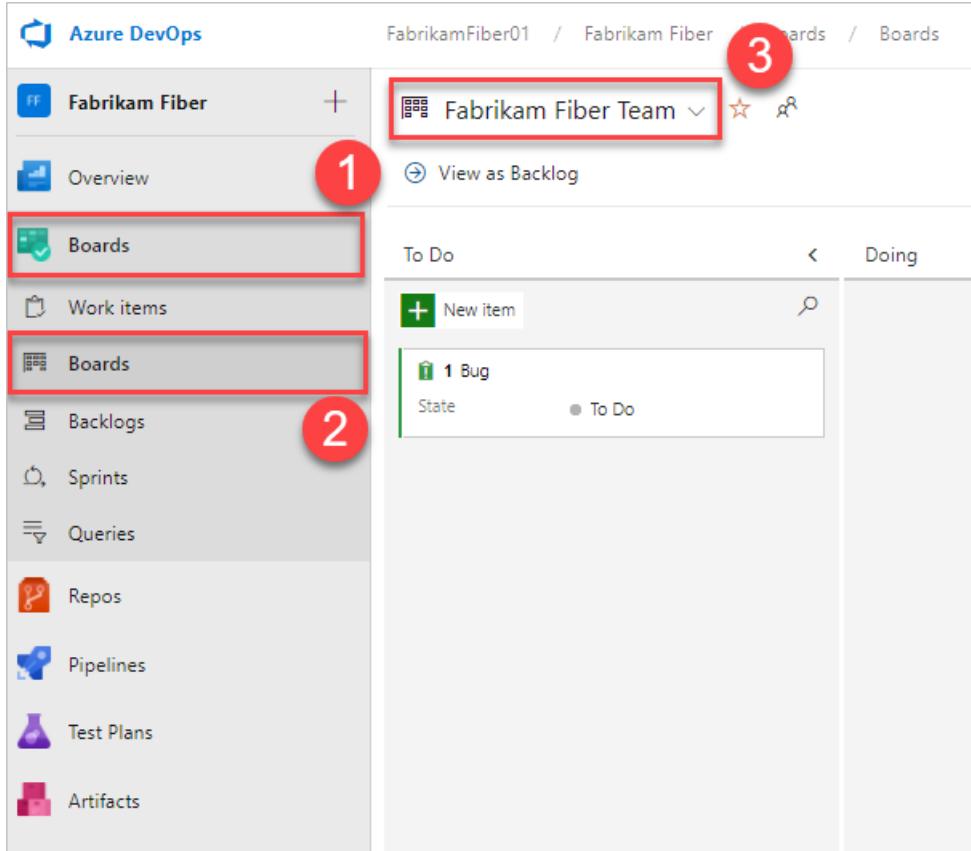
### NOTE

To understand the differences between backlogs, boards, taskboards, and Delivery plans, see [Backlogs, boards, and plans](#). If your backlog or board doesn't show the work items that you expect or want, see [Set up your backlogs and boards](#).

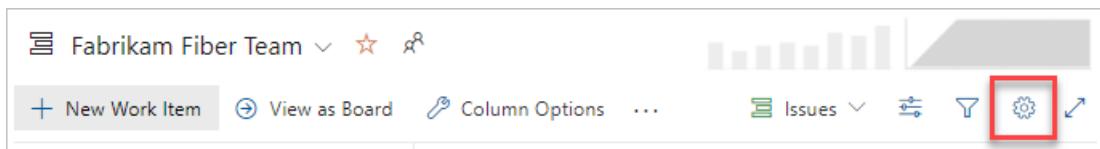
#### NOTE

To understand the differences between backlogs, boards, and taskboards, see [Backlogs, and boards](#). If your backlog or board doesn't show the work items that you expect or want, see [Set up your backlogs and boards](#).

1. Check that you selected the correct project, and then choose **Boards > Boards**, and select the correct team from the team selector dropdown menu. For more information, see [Use breadcrumbs and selectors to navigate and open artifacts](#).



2. Choose **Team settings** to configure the board and set general team settings.



3. Choose a tab under any of the sections—**Cards, Board, Charts, and General**—to configure the cards or boards, the cumulative flow chart, or other team settings. When you're done configuring the settings, select **Save and close**.

Settings

Cards

**Fields \***

- Styles
- Tag colors
- Annotations
- Tests

Board

- Columns
- Swimlanes
- Card reordering
- Status badge

Charts

- Cumulative Flow

General

- Backlogs
- Working days

Fields

Show the important information to your team. Fields are editable directly on the card.

**Issue**

**Core fields**

Show ID

Show Assigned To as:

Avatar and full name (default)

Show Effort

Show Tags

**Additional fields**

Add up to 10 fields in the order that you want them to appear on the card.

+ Field

State

Show empty fields

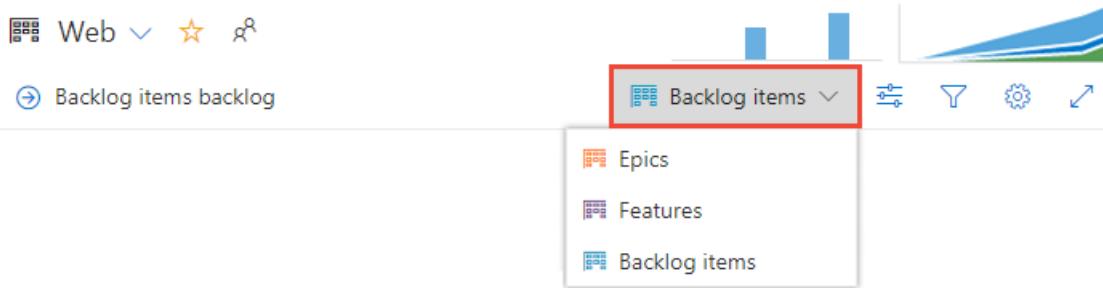
Check if you want to display fields, even when they are empty.

**Save and close**    **Cancel**

1. Check that you selected the right project, (2) choose Boards > Boards, and then (3) select the correct team from the team selector menu.

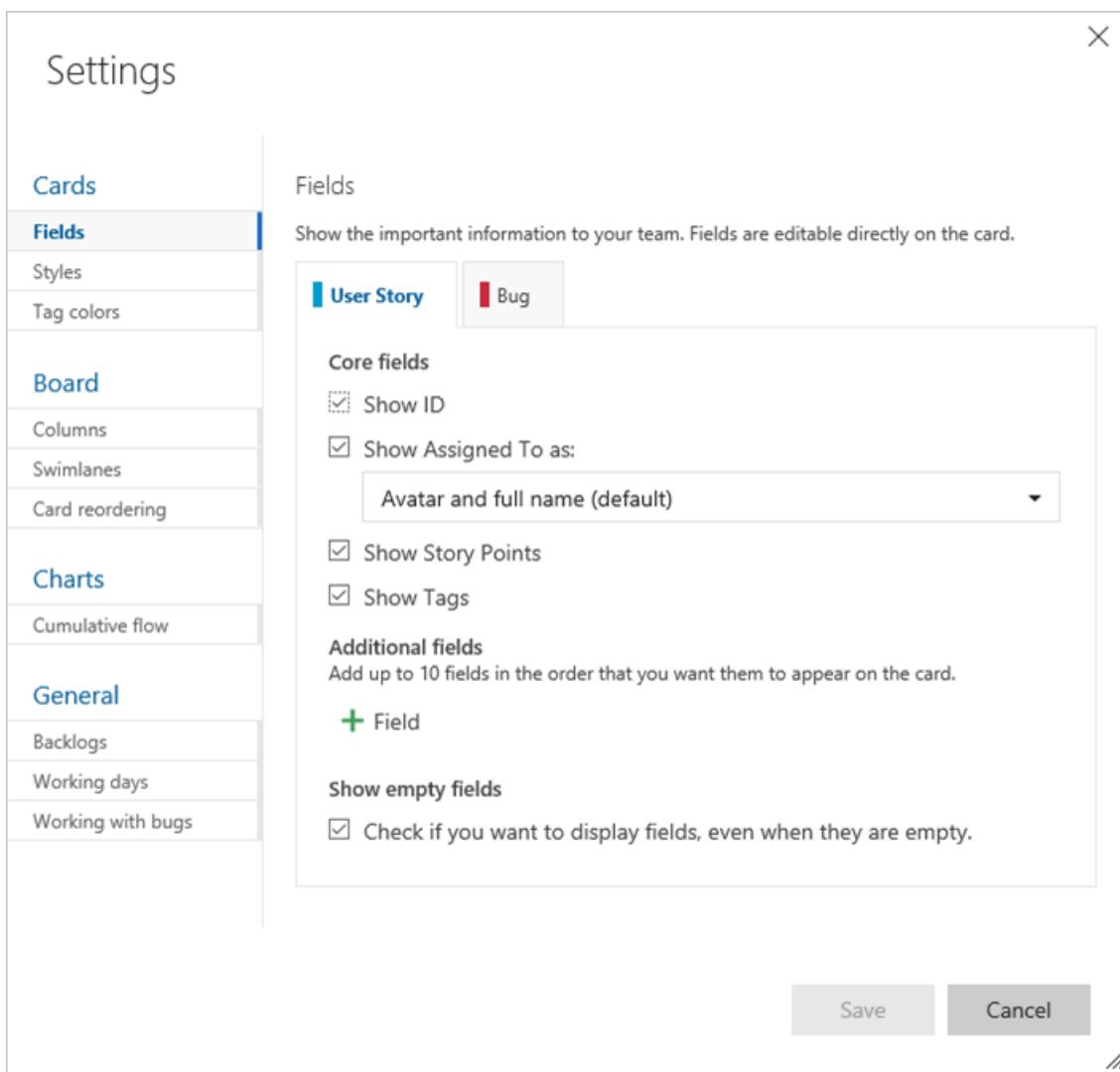
The screenshot shows the Azure DevOps interface. At the top left is the 'Azure DevOps' logo. Next to it is a dropdown menu 'fabrikam / Fabrikam Fiber' with a red box around it and a circled '1'. Below the header is a navigation bar with several items: 'Fabrikam Fiber' (with a plus sign), 'Overview', 'Boards' (which is highlighted with a red box and has a circled '2' over it), 'Work Items', 'Backlogs', and 'Sprints'. To the right of the navigation bar is a team selector dropdown 'Fabrikam Fiber Team' with a red box around it and a circled '3'. The main area displays a 'Backlog' board. It includes a 'New item' button, a search bar, and a list of backlog items. One item is highlighted: 'Add an information form' by 'Raisa Pokrovskaya' in 'Iteration ... Sprint 3'. To the right of the backlog is an 'Analyze' section showing '5/10' and a 'Develop' section showing 'Slow form' and 'Jam' status. A sidebar on the far right shows user activity: 'Johnnie McLeod' (iteration 3, 0/4), 'Raisa Pokrovskaya' (iteration 3, 0/2), and 'Slow form' (0/2).

2. Make sure that you select the team backlog or board that you want to configure using the team selector. To learn more, see [Use breadcrumbs and selectors to navigate and open artifacts](#).
3. Choose the product or portfolio backlog from the board-selection menu.



4. Choose **Team settings**  to configure the board and set general team settings.

5. Choose a tab under any of the sections—**Cards**, **Board**, **Charts**, and **General**—to configure the cards or boards, the cumulative flow chart, or other team settings.



1. Make sure that you select the team from the project/team selector. You can switch your team focus to one that you've recently viewed from the project/team selector. If you don't see the team or project you want, choose **Browse...** or choose **Azure DevOps**  to access the **Projects** page.

Fabrikam Fiber Home

Recent projects/teams

- Agile 11
- FabrikamFiber
- Fabrikam Fiber A
- Fabrikam Fiber PB

Browse...

New team

A README.md file is intended to quickly orient readers to what your project can do.  
Learn more about Markdown.

2. Open Work > Backlogs > Board.

Work Items\* Backlogs Queries Plans

Epics Features Stories

Past Current Sprint 5

Stories

Backlog Board

Backlog

Active 6/5 Resolved

+ New item

Slow response on form Christie Church 8

Add animated emoticons Christie Church 3

Welcome back page

Cancel order form Jamal Hartnett 13

Implement a factory abstracts Jamal Hartnett 0/1

Bug 6 Raisa Pokrovskaya 3

3. Choose the board you want to configure and then choose Team settings to configure the board and set general team settings.

For example, from the Kanban board ...

Stories

Backlog Board

Backlog Analyze Develop Test

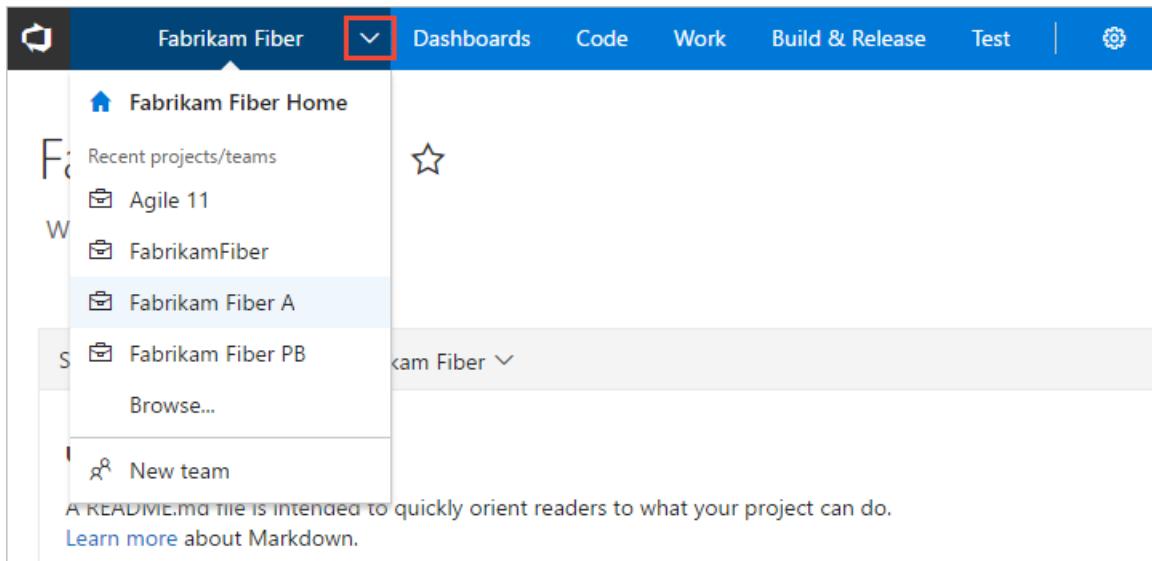
4/10 4/10 4/5 4/5

4. Choose a tab under Cards or Board to configure the cards and Kanban board columns and swimlanes.

![Common configuration dialog team settings]../../boards/boards/media/customize-cards/common-config-141.png)

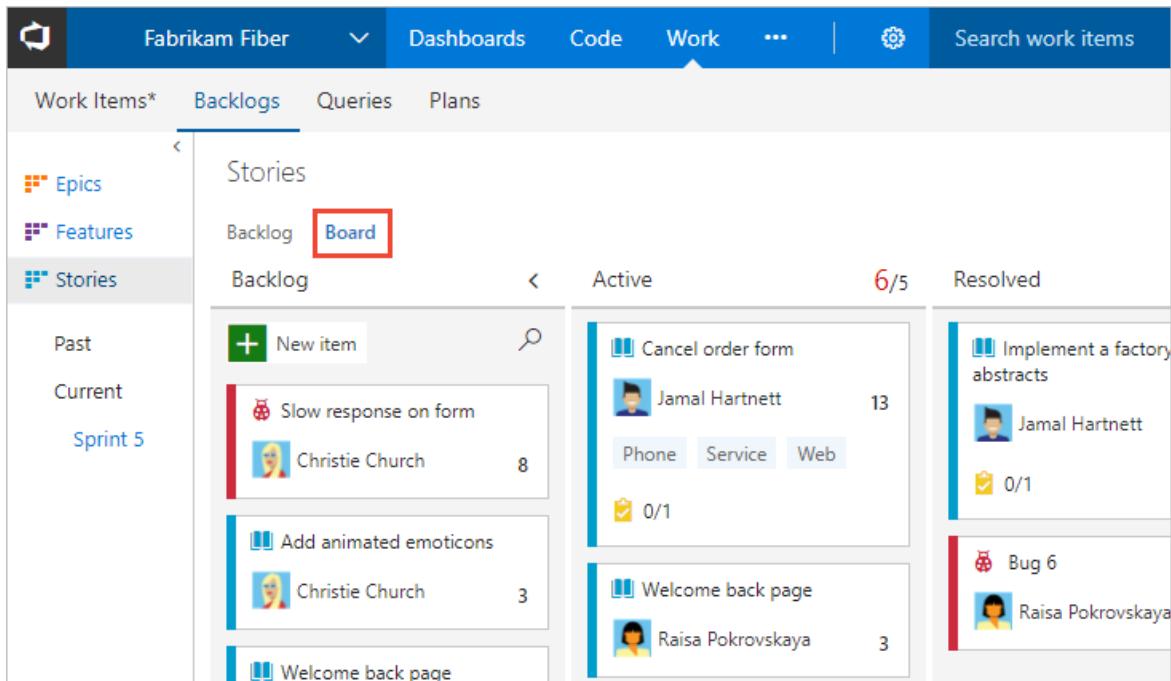
1. Make sure that you select the team from the project/team selector. You can switch your team focus to one that you've recently viewed from the project/team selector. If you don't see the team or project you want,

choose **Browse...** or choose **Settings**  to access the **Projects** page.



The screenshot shows the Microsoft Teams navigation bar with the 'Fabrikam Fiber' team selected. The 'Work' tab is highlighted with a red box. The left sidebar shows recent projects/teams including 'Agile 11', 'FabrikamFiber', 'Fabrikam Fiber A', 'Fabrikam Fiber PB', and 'New team'. A note at the bottom states: 'A README.md file is intended to quickly orient readers to what your project can do. Learn more about Markdown.'

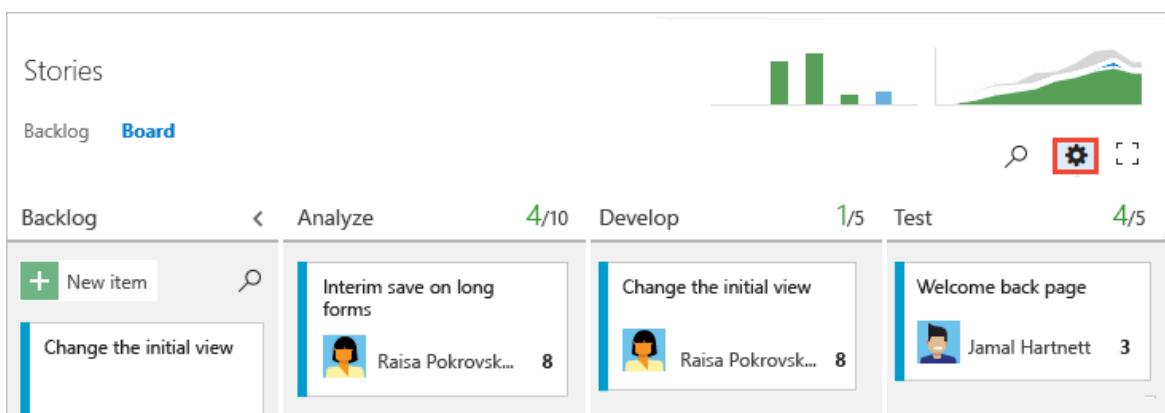
2. Open **Work > Backlogs > Board**.



The screenshot shows the 'Work' section with the 'Backlogs' tab selected. The left sidebar shows 'Stories' selected. The main area displays a Kanban board titled 'Stories' with columns 'Backlog' and 'Board'. The 'Board' column is highlighted with a red box. The 'Backlog' column shows a card for 'Slow response on form' assigned to 'Christie Church' with priority '8'. The 'Board' column shows cards for 'Cancel order form' (assigned to 'Jamal Hartnett', priority '13'), 'Welcome back page' (assigned to 'Raisa Pokrovskaya', priority '3'), and 'Add animated emoticons' (assigned to 'Christie Church', priority '3').

3. Choose the board you want to configure and then choose **Team settings**  to configure the board and set general team settings.

For example, from the Kanban board ...



The screenshot shows the Kanban board configuration interface. The top navigation bar has 'Backlog' and 'Board' tabs, with 'Board' selected. Below the navigation are four columns: 'Backlog', 'Analyze', 'Develop', and 'Test'. Each column has a green progress bar above it. The 'Backlog' column shows a card for 'Change the initial view'. The 'Analyze' column shows a card for 'Interim save on long forms' assigned to 'Raisa Pokrovsk...' with priority '8'. The 'Develop' column shows a card for 'Change the initial view' assigned to 'Raisa Pokrovsk...' with priority '8'. The 'Test' column shows a card for 'Welcome back page' assigned to 'Jamal Hartnett' with priority '3'. On the right side, there are icons for search, settings, and a refresh button.

4. Choose a tab under **Cards** or **Board** to configure the cards and Kanban board columns and swimlanes.

## Settings

**Cards**

**Fields**

Show the important information to your team. Fields are editable directly on the card.

**User Story**   **Bug**

**Core fields**

Show ID

Show Assigned To as:

Avatar and full name (default) ▾

Show Story Points

Show Tags

**Additional fields**

Add up to 10 fields in the order that you want them to appear on the card.

+ Field

**Show empty fields**

Check if you want to display fields, even when they are empty.

**Save**   **Cancel**

Team administrators can fully customize the team's Kanban boards associated with the product and portfolio backlogs. You configure a Kanban board by first defining the columns and WIP limits from the common configuration dialog. For guidance, see [Kanban basics](#).

For more information on each configuration option, see the following articles:

### General

- [Backlogs](#)
- [Working days](#)
- [Working with bugs](#)

### Cards

- [Add fields](#)
- [Define styles](#)
- [Add tag colors](#)
- [Enable annotations](#)
- [Configure inline tests](#)
  
- [Add fields](#)
- [Define styles](#)
- [Add tag colors](#)

### Boards

- [Add columns](#)

- Split columns
- WIP limits
- Definition of Done
- Add swimlanes
- Card reordering
- Configure status badges
  
- Add columns
- Split columns
- WIP limits
- Definition of Done
- Add swimlanes
- Card reordering
  
- Add columns
- WIP limits
- Definition of Done

## Chart

- Configure cumulative flow chart

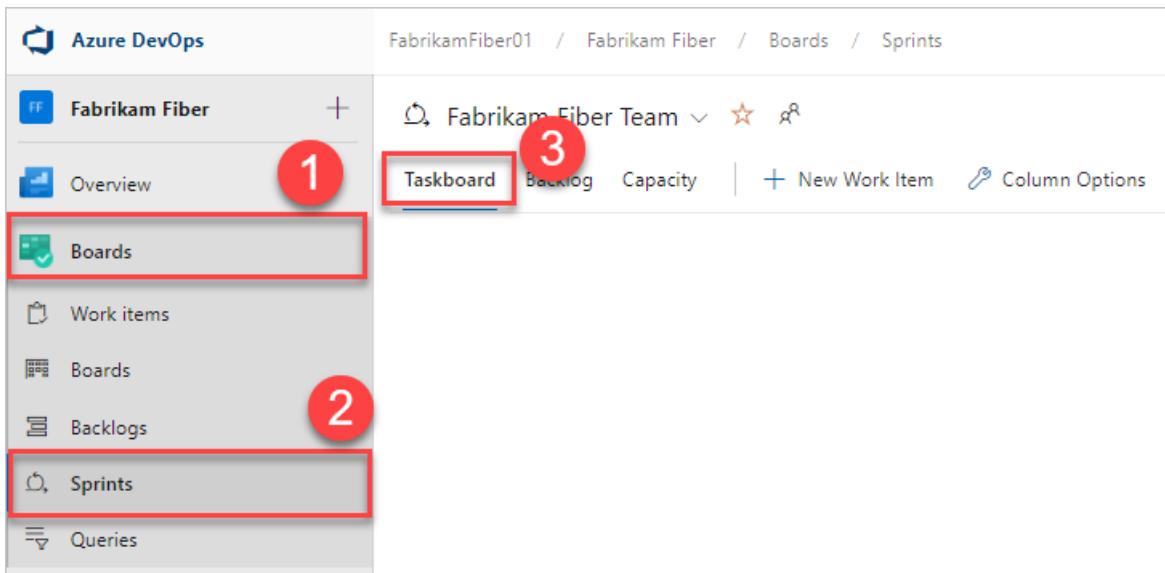
## Kanban board

Backlog	<	Active	2/5	Resolved	2/5	Closed
<a href="#">+ New item</a>  532 Hello World Web Site  Jamal Hartnett		 486 Welcome back page  Raisa Pokrovskaya 3	2/5	 344 Implement a factory which abstracts  Jamal Hartnett 8	2/5	
 398 Cancel order form  Jamal Hartnett 13 Phone Service Web  0/1		 346 Add animated emoticons  Christie Church 3		 405 GPS locator  Jamal Hartnett 8		

## Configure sprint Taskboards

Similar to Kanban boards, each sprint Taskboard can be customized to support information-rich, color-coded cards as well as addition of customized columns. For details, see [Customize sprint Taskboards](#).

Similar to Kanban boards, each sprint Taskboard can be customized to support information-rich, color-coded cards. For details, see [Customize sprint Taskboards](#).



## Add and manage team dashboards

By default, all team members can add and edit team dashboards. In addition, team administrators can manage permissions for team dashboards. For details, see [Add and manage dashboards](#).

Team administrators can add, configure, and manage permissions for team dashboards. For details, see [Add and manage dashboards](#).

Azure DevOps interface showing the Fabrikam Fiber team overview. The 'Dashboards' menu item (1) is highlighted with a red box. The 'Edit' button (2) in the top right corner is highlighted with a red box.

## Update team name, description, and image

Team settings also include the team name, description, and team profile image. To add a team picture, select the image icon. The maximum file size is 2.5 MB.

The screenshot shows the 'Project Settings' interface for 'Fabrikam Fiber'. On the left, a sidebar lists 'General', 'Overview', 'Teams' (which is selected), 'Permissions', 'Notifications', 'Service hooks', and 'Dashboards'. The main area is titled 'Team Profile' and contains a placeholder image icon (three stylized people) which is highlighted with a red box. Below the image are fields for 'Name' (set to 'Fabrikam Fiber Team') and 'Description' (set to 'The default project team.'), both of which are also highlighted with red boxes. At the bottom are 'Save' and 'Undo' buttons, with 'Save' also highlighted with a red box.

Team settings also include the team name, description, and team profile image. To add a team picture, select the image icon. The maximum file size is 2.5 MB.

The screenshot shows the 'Team' settings page. It features a large blue circular profile picture with 'TT' in white. The title 'Team' is at the top right. Below the title is a description: 'The default project team.' and 'Relevant links: Notifications | Dashboards | Iterations and Area Paths'. There are two tabs: 'Members' and 'Settings', with 'Settings' being the active tab. Under the 'Settings' tab, there's a section titled 'Team Image' with a smaller blue circular profile picture with 'TT'. Below it is a text input field with the placeholder 'Select an image file on your computer (2.5MB max)' and a red-bordered 'Upload image' button. A note below states: 'By uploading a file you certify that you have the right to distribute this picture and you agree to the Terms of Service, Privacy Statement, and Code of Conduct.'

Team settings also include the team name, description, and team profile image. To add a team picture. Open the Team Profile and choose the picture icon. The maximum file size is 4 MB.

## Manage notifications

Team administrators can add and modify alerts so that the team can receive email notifications as changes occur to work items, code reviews, source control files, and builds. Many alerts are defined for each team. For details, see [Manage team alerts](#).

## Team Profile



Name

Fabrikam Fiber Team

Description

The default project team.

Administrators

Jamal Hartnett

[+ Add](#)

Manage other settings for this team

[Notifications](#)[Dashboards](#)[Iterations and areas](#)

## Fabrikam Fiber Team

## Members

[+ Add...](#) | 

## Display Name

## Username Or Scope

Security Service Group	[FabrikamFiber01]
Fabrikam Fiber Build Servic...	Build\99612ac2-9c5f-4191-9e8...

Jamal Hartnett

fabrikamfiber4@hotmail.com

## Manage team rooms

Team administrators can add users and events to team rooms, and add team rooms. Team rooms are chat rooms limited to team members. For details, see [Collaborate in a team room](#).

**NOTE**

Team rooms are deprecated for TFS 2018 and later versions as described in [Deprecation of team rooms](#) blog post. Several good solutions are available that integrate well with TFS that support notifications and chat, such as [Microsoft Teams](#) and [Slack](#).

## Related articles

- [About projects and scaling your organization](#)
- [About teams and Agile tools](#)
- [Add teams](#)
- [Add a team administrator](#)

# Change individual or group permissions

4/28/2021 • 4 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

The standard way to set permissions is by adding them to one or more built-in security groups. However, sometimes you may want to grant additional permissions to select users, where not all permissions are assigned to the security group. For example, if you want to give some users the ability to add or edit area and iteration paths, but don't want them to have all permissions available to members of the Project Administrators group.

You can change individual permissions in one of the following three ways:

- Create a custom Azure DevOps security group, define permissions for that group, add the user account to the group
- For object-level permissions: Add the user account and set permissions
- For project or collection-level permissions: Search for the user account and selectively change their permission assignments

In this article you learn how to do the following tasks:

- Create a custom security group
- Set permissions for a custom security group
- Add members to a custom security group
- Change the permission assignments for an individual user

If you're new to managing permissions and groups, review [Get started with permissions, access, and security groups](#) to learn about permission states and inheritance.

## Prerequisites

- To manage permissions or groups at the project level, you must be a member of the Project Administrators Group or have your **Edit project-level information** set to Allow. If you created the project, you are automatically added as a member of this group.
- To manage permissions or groups at the collection or instance level, you must be a member of the Project Collection Administrators Group or have your **Edit instance-level information** set to Allow. If you created the organization or collection, you are automatically added as a member of this group.

### NOTE

The images you see from your web portal may differ from the images you see in this article. These differences result from updates made to Azure DevOps Services or your on-premises deployment. However, the basic functionality available to you remains the same unless explicitly mentioned.

## Create a custom security group

Create a custom security group at the project-level or the collection-level. The method for creating a custom security group is the same, no matter at what level you add it.

To create a project-level security group, open the web portal and choose the project where you want to add users or groups.

**NOTE**

To enable the new user interface for the Project Permissions Settings Page, see [Enable preview features](#).

- [Preview page](#)
- [Current page](#)

1. Choose **Project settings > Permissions**.

The screenshot shows the 'Project Settings' page for the 'FabrikamFiber' project in Azure DevOps. The left sidebar lists various project management sections: Overview, Boards, Repos, Pipelines, Test Plans, and Artifacts. The 'Project settings' button at the bottom of the sidebar is also highlighted with a red box. The main content area is titled 'Project Settings' and contains several sections: General (Overview, Teams, Permissions - which is highlighted with a red box), Boards (Project configuration, Team configuration, GitHub connections), Repos (Repositories, Cross-repo policies), Pipelines (Agent pools, Parallel jobs, Settings, Test management, Release retention, Service connections\*, XAML build services), and Test.

2. Choose **New group** to open the dialog for adding a group.

**Permissions**

**+ New Group**

Groups Users Search groups

Name	Description	Me...
BA Build ...	Members of this group can create, modify and delete build definitions and manage queued and completed builds.	G.. ↗ 0
C Contr...	Members of this group can add, modify, and delete items within the team project.	G.. ↗ 2
PA Projec...	Members of this group can perform all operations in the	G.. ↗ 1

3. Enter a name for the group, select users or groups for membership, optionally add a description, and then choose **Create**.

**Create a new group**



**Name**  
Fabrikam Fiber group

There is already an existing group with this name.

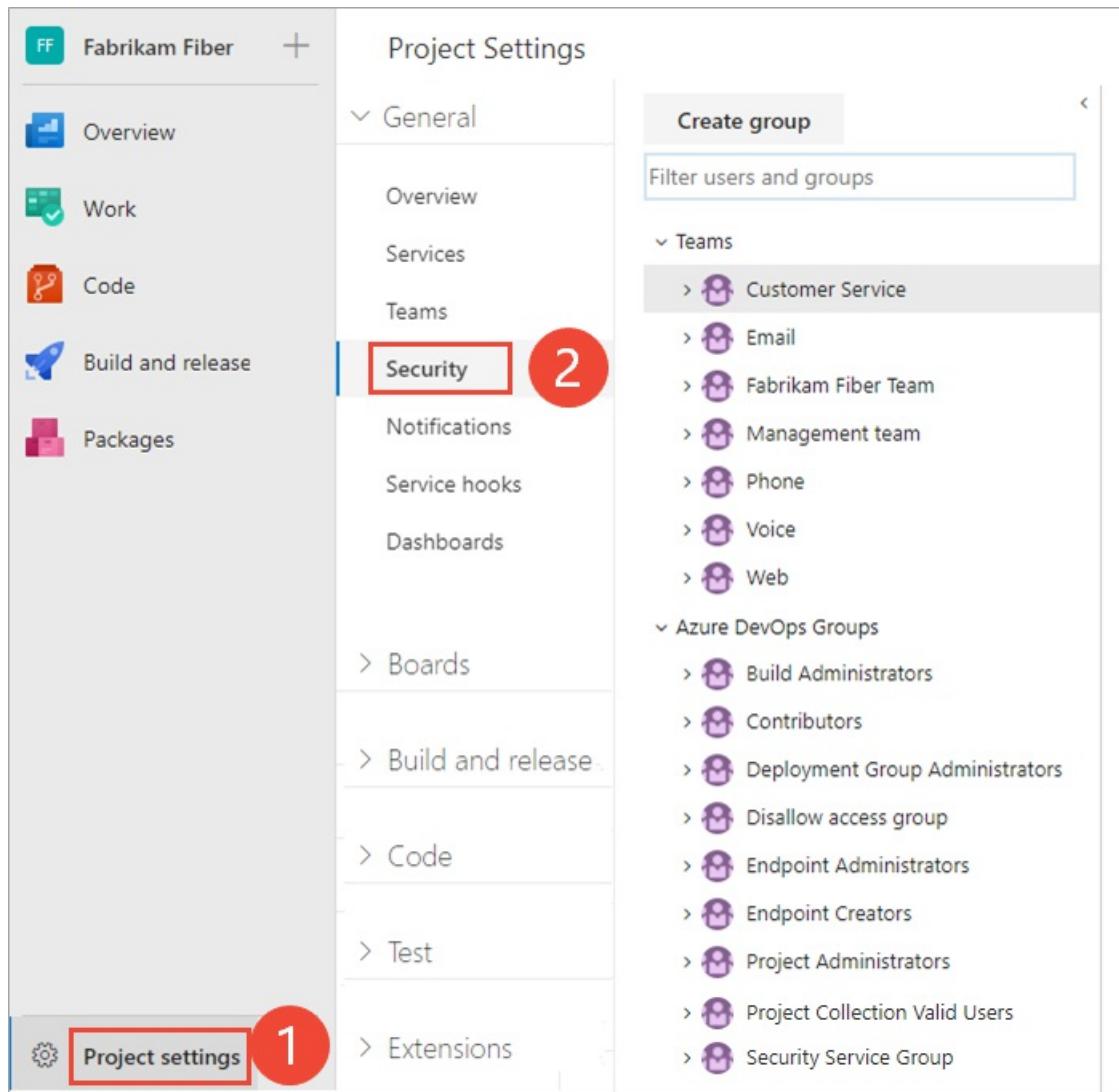
**Members**  
Jamal Hartnett  Add members

Group membership can be a combination of users, other groups, and Azure Active Directory groups.

**Description**  
Add a description to your group this will appear in the group page

1. Choose **Project settings > Security**.

*To see the full image, select to expand.*



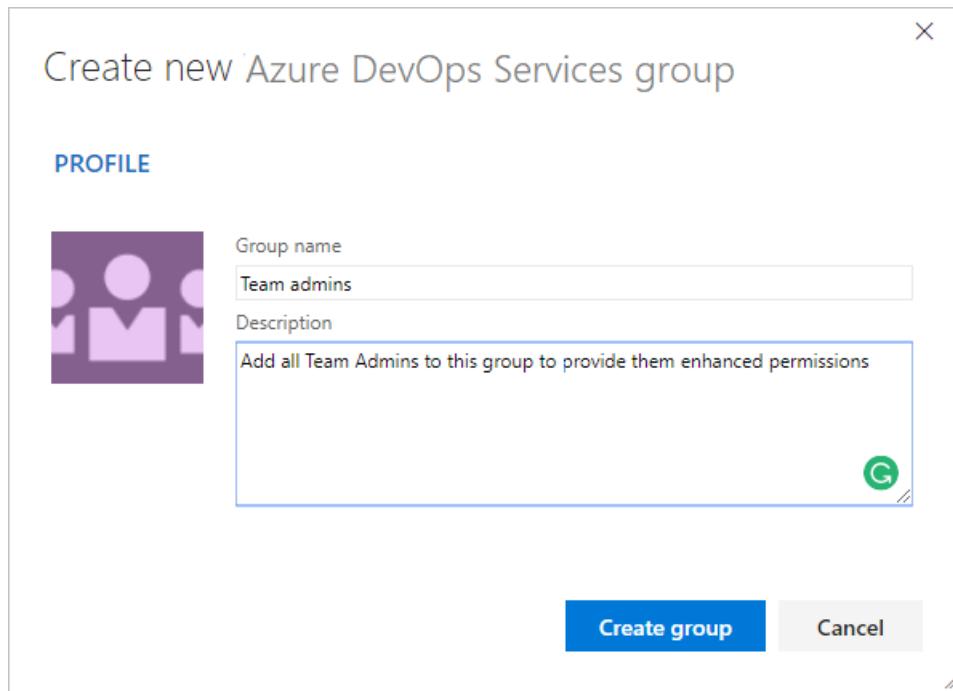
2. Choose **Create group** to open the dialog for adding a group.

The screenshot shows the 'Create group' dialog for the 'Customer Service' group. The 'Teams' section is expanded, showing the 'Customer Service' group selected. The 'Azure DevOps Groups' section is also visible. To the right, a table lists permissions for the 'Customer Service' group under the 'Fabrikam Fiber > Customer Service' scope. The table has columns for 'Permissions', 'Members', and 'Member of'. Most permissions are set to 'Not set' or 'Allow (inherited)'. Some specific permissions include:

Permissions	Members	Member of
Bypass rules on work item updates		Not set
Change process of team project.		Not set
Create tag definition		Allow (inherited)
Create test runs		Allow (inherited)
Delete and restore work items		Not set
Delete shared Analytics views		Allow (inherited)
Delete team project		Not set
Delete test runs		Allow (inherited)
Edit project-level information		Not set
Edit shared Analytics views		Allow (inherited)
Manage project properties		Not set
Manage test configurations		Allow (inherited)
Manage test environments		Allow (inherited)
Move work items out of this project		Not set
Permanently delete work items		Not set

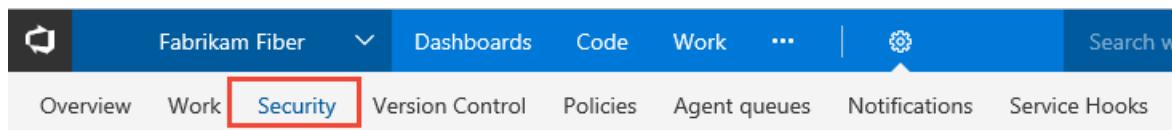
3. Enter a name for the group, and optionally a description.

For example, here we define a Team Admins group.



4. Choose **Create group**.

1. Open Project Settings. Choose the gear settings icon, and choose **Security**.

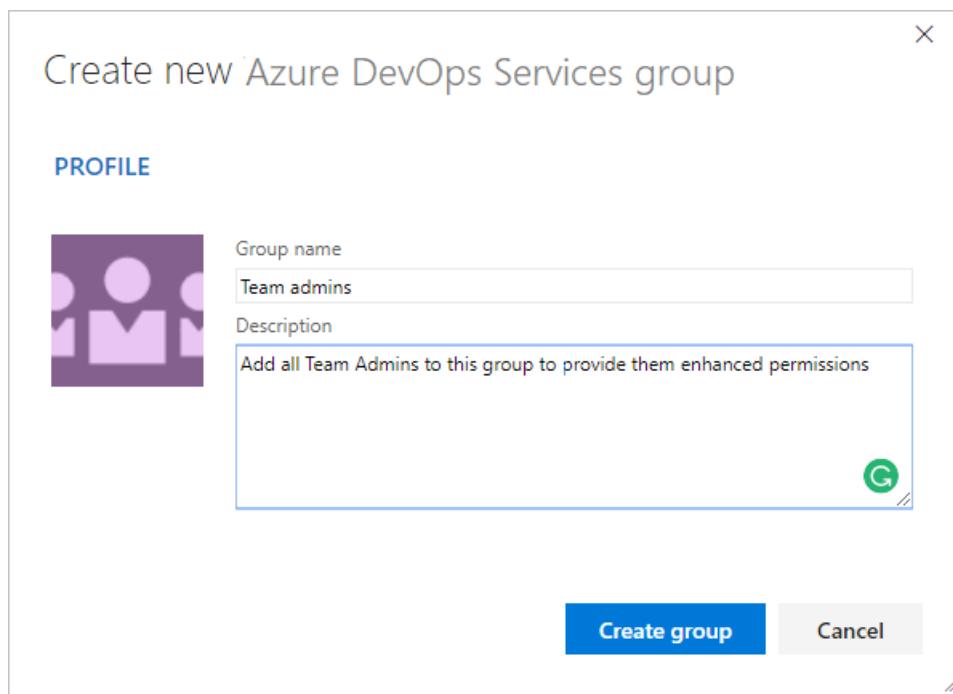


2. Choose **Create group** to open the dialog for adding a group.

Permissions	Members	Member of
Bypass rules on work item updates	Not set	
Change process of team project.	Not set	
Create tag definition	Allow (inherited)	
Create test runs	Allow (inherited)	
Delete and restore work items	Not set	
Delete shared Analytics views	Allow (inherited)	
Delete team project	Not set	
Delete test runs	Allow (inherited)	
Edit project-level information	Not set	
Edit shared Analytics views	Allow (inherited)	
Manage project properties	Not set	
Manage test configurations	Allow (inherited)	
Manage test environments	Allow (inherited)	
Move work items out of this project	Not set	
Permanently delete work items	Not set	

3. Enter a name for the group, and optionally a description.

For example, here we define a Team Admins group.



4. Choose **Create group**.

## Set permissions for a custom security group

1. To set permissions for the custom group you created, choose the group name and then set one or more permissions.

The screenshot shows the 'Security' tab selected in the top navigation bar. Under 'Fabrikam Fiber > Team Admins', there is an 'Edit...' button. The main area displays a table of permissions:

	Permissions	Members	Member of
Bypass rules on work item updates	Allow		
Create tag definition	Allow		
Create test runs	Allow		
Delete and restore work items	Allow		
Delete team project	Deny		
Delete test runs	Not set		
Edit project-level information	Not set		
Manage project properties	Deny		
Manage test configurations	Allow		
Manage test environments	Allow		
Move work items out of this project	Allow		
Permanently delete work items	Allow		
Rename team project	Deny		
Suppress notifications for work item updates	Not set		
View analytics	Allow (inherited)		
View project-level information	Allow		
View test runs	Allow		

At the bottom, there are 'Clear explicit permissions', 'Save changes', and 'Undo changes' buttons.

For a description of each permission, see [Permissions and groups reference, project-level permissions](#).

2. Choose **Save changes**.

## Add members to a custom security group

You add members to a custom security group in the same way you add users to a built-in group.

1. Choose the security group, choose **Members**, and then choose **Add**.

The screenshot shows the 'Create group' interface. On the left, there's a sidebar with options like 'Teams', 'Azure DevOps Groups', and 'Project Administrators'. The 'Team Admins' option is highlighted. The main area shows the 'Fabrikam Fiber > Team Admins' group. The 'Members' tab is selected, indicated by a red box around it. Below it, there's a search bar with '+ Add...', 'Search', and 'Edit...' buttons. The results table has columns for 'Display Name' and 'Username Or Scope'. A message says 'No identities found in current scope.'

2. Enter the user identity into the text box. You can enter several identities into the text box, separated by commas. The system automatically searches for matches. Choose the match(es) that meets your choice.

The screenshot shows the 'Add users and groups' dialog box. It has a search input field containing 'Chris'. Below it, a list shows one result: 'Christie Church' with the email 'fabrikamfiber1@hotmail.com'. At the bottom are 'Save changes' and 'Cancel' buttons.

**NOTE**

Users that have limited access, such as Stakeholders, won't be able to access select features even if granted permissions to those features. To learn more, see [Permissions and access](#).

## Change individual permission at the project-level

1. From the project-level Security page, enter the user identity in the Filter users and groups box. Then, select the account whose permissions you want to change.

The screenshot shows the 'Create group' interface again. The 'Members' tab is selected. The search input field contains 'Rais'. Below it, a list shows one result: 'Raisa Pokrovskaya' with the email 'fabrikamfiber5@hotmail.com'. At the bottom are 'Edit...', 'Members', and 'Member of' buttons, along with a 'Search' button.

2. Change the permission, setting a permission as Allow or Deny.

The screenshot shows the 'Create group' dialog. On the left, there's a sidebar with a search bar containing 'Rais' and a list of users, with 'Raisa Pokrovskaya' selected. The main area is titled 'fabrikam > Raisa Pokrovskaya'. It shows a table of permissions:

Permissions	Member of
Bypass rules on work item updates	Not set
Create tag definition	Allow (inherited)
Create test runs	Allow (inherited)
Delete and restore work items	Allow
Delete team project	Not set
Delete test runs	Allow (inherited)
Edit project-level information	Not set
Manage project properties	Allow
Manage test configurations	Allow (inherited)
Manage test environments	Allow (inherited)
Move work items out of this project	Allow
Permanently delete work items	Allow
Rename team project	Not set
Suppress notifications for work item updates	Not set
View analytics	Allow (inherited)
View project-level information	Allow (inherited)
View test runs	Allow (inherited)

At the bottom, there are buttons for 'Clear explicit permissions', 'Save changes', and 'Undo changes'. The 'Move work items out of this project' row has its 'Allow' button highlighted with a blue border.

For a description of each permission, see [Permissions and groups reference, project-level permissions](#).

3. Choose **Save changes**.

#### Change individual permission at the collection-level

1. Open the user-level or collection-level **Security** admin page and follow the instructions provided in the previous section for project-level permissions.

For a description of each collection-level permission, see [Permissions and groups reference, collection-level permissions](#).

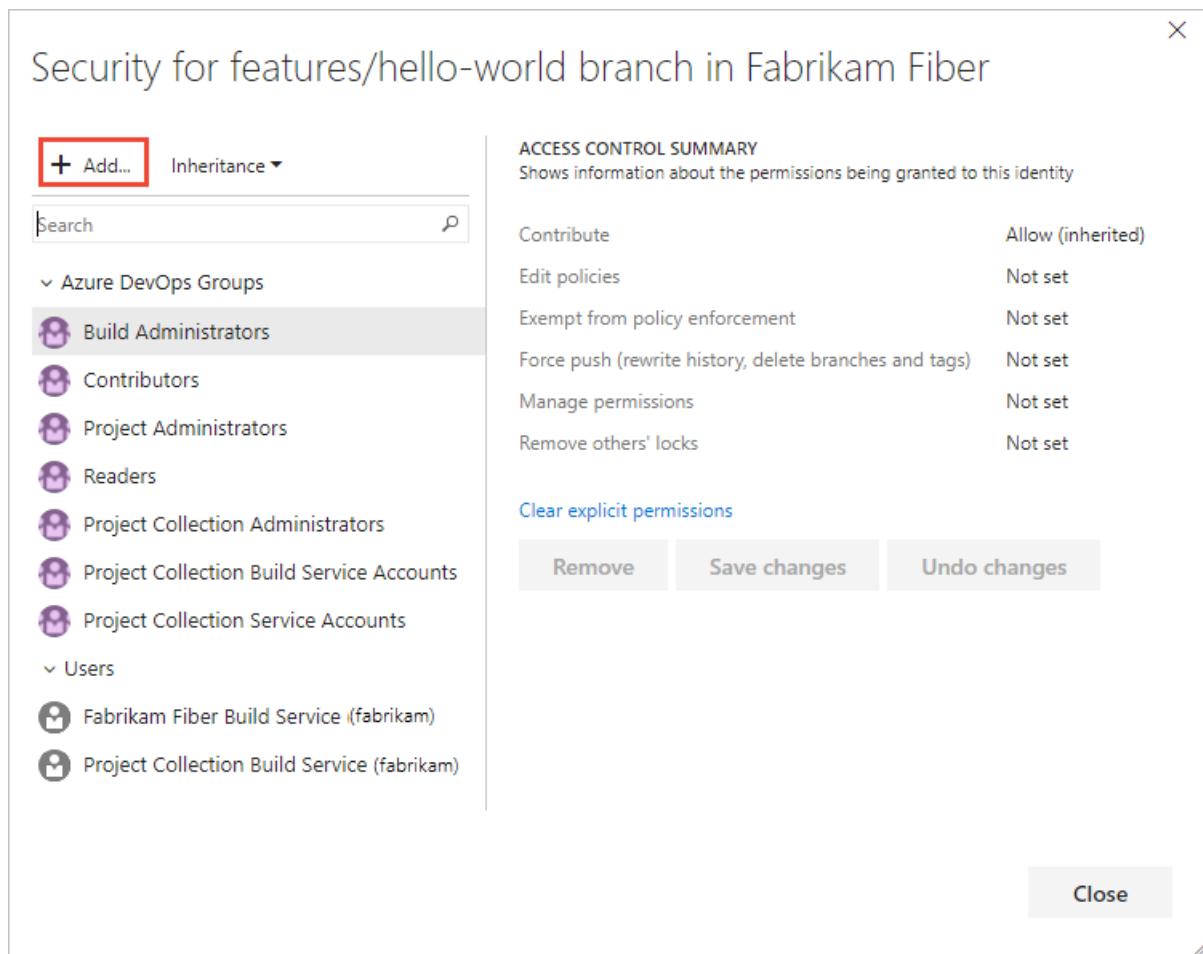
#### Change individual permission at an object-level

From the web portal, open the Security dialog for the object whose permissions you want to set. For specific instructions, see the following articles:

AREA	TASK
Wiki & Dashboard permissions	<ul style="list-style-type: none"><li>• <a href="#">README &amp; Wiki</a></li><li>• <a href="#">Dashboards</a></li></ul>

DevOps (code, build, test, release) permissions	<ul style="list-style-type: none"> <li>● <a href="#">Git branch</a></li> <li>● <a href="#">Git repository</a></li> <li>● <a href="#">TFVC</a></li> <li>● <a href="#">Builds</a></li> <li>● <a href="#">Release pipeline security</a></li> <li>● <a href="#">Approvals and approvers</a></li> </ul>
Work tracking permissions	<ul style="list-style-type: none"> <li>● <a href="#">Area and iteration paths</a></li> <li>● <a href="#">Work item query and folder</a></li> <li>● <a href="#">Plan permissions</a></li> </ul>

1. From the Security dialog, choose Add.



2. Enter the user ID, choose search, and then make your selection in the left pane.

3. Update the permission setting to **Allow** or **Deny** for specific permissions.

## Security for features/hello-world branch in Fabrikam Fiber

The screenshot shows the 'ACCESS CONTROL SUMMARY' for the identity 'Raisa Pokrovskaya'. The summary lists several permissions with their current status:

Permission	Status
Contribute	Allow (inherited)
Edit policies	Allow (inherited)
Exempt from policy enforcement	Allow
Force push (rewrite history, delete branches and tags)	Allow (inherited)
Manage permissions	Allow (inherited)
Remove others' locks	Allow (inherited)

Below the summary, there is a 'Clear explicit permissions' button and three action buttons: 'Remove', 'Save changes', and 'Undo changes'. The 'Save changes' button is highlighted with a grey background.

For a description of specific permissions, see [Permissions and groups reference](#).

4. Choose **Save changes**.

## Next steps

[Grant or restrict access to select features](#)

## Related articles

- [Permissions lookup guide](#)
- [Get started with permissions, access, and security groups](#)
- [Permissions and groups reference](#)
- [Set permissions at the project-level or project collection-level](#)
- [Troubleshoot permissions](#)

# Grant or restrict access using permissions

4/28/2021 • 10 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

You can grant or restrict access to resources that you manage in Azure DevOps. You may want to open up or close down access to a select set of features and for a select set of users. While the built-in security groups provide a standard set of permission assignments, you may need additional security requirements not met by these assignments.

If you're new to administrating permissions and groups, review [Get started with permissions, access, and security groups](#)to learn about permission states and inheritance.

In this article you learn how to do the following tasks:

- Recommended method for granting and restricting permissions
- Delegate tasks by assigning select permissions to specific roles
- Limit visibility to organization information
- Limit people picker to project users and groups
- Restrict access to view or modify objects
- Restrict modification of work items based on a user or group
- Recommended method for granting and restricting permissions
- Delegate tasks by assigning select permissions to specific roles
- Restrict access to view or modify objects
- Restrict modification of work items based on a user or group

## TIP

Because you set many permissions at an object-level, such as repositories and area paths, how you structure your project determines the areas you can open up or close down.

## Recommended method for granting and restricting permissions

For maintenance purposes, we recommend you use either the built-in security groups or [custom security groups to manage permissions](#).

You can't change the permission settings for the Project Administrators group or the Project Collection Administrators group, which is by design. However, for all other groups, you can change the permissions.

If you manage a small number of users, then you may find changing individual permissions a valid option. However, custom security groups allow you to better track roles and permissions assigned to those roles.

## Delegate tasks to specific roles

As an administrator or account owner, it's a good idea to delegate administrative tasks to those team members who lead or manage an area. Several of the main built-in roles that come with default permissions and role assignments are:

- Readers

- Contributors
- Team Administrator (role)
- Project Administrators
- Project Collection Administrators

For a summary of permissions for the above roles, see [Default permissions and access](#), or for the Project Collection Administrators, see [Add administrators](#)

To delegate tasks to other members within your organization, consider creating a custom security group and then granting permissions as indicated in the following table.

Role	Tasks to perform	Permissions to set to Allow
Development lead (Git)	Manage branch policies	Edit policies, Force push, and Manage permissions See <a href="#">Set branch permissions</a> .
Development lead (TFVC)	Manage repository and branches	Administer labels, Manage branch, and Manage permissions See <a href="#">Set TFVC repository permissions</a> .
Software architect (Git)	Manage repositories	Create repositories, Force push, and Manage permissions See <a href="#">Set Git repository permissions</a>
Team administrators	Add area paths for their team Add shared queries for their team	Create child nodes, Delete this node, Edit this node See <a href="#">Create child nodes, modify work items under an area path</a> Contribute, Delete, Manage permissions (for a query folder), See <a href="#">Set query permissions</a> .
Contributors	Add shared queries under a query folder, Contribute to dashboards	Contribute, Delete (for a query folder), See <a href="#">Set query permissions</a> View, Edit, and Manage dashboards, See <a href="#">Set dashboard permissions</a> .
Project or product manager	Add area paths, iteration paths, and shared queries Delete and restore work items, Move work items out of this project, Permanently delete work items	Edit project-level information, See <a href="#">Add administrators, set permissions at the project-level or project collection-level</a> .
Process template manager ( <a href="#">Inheritance process model</a> )	Work tracking customization	Administer process permissions, Create new projects, Create process, Delete field from account, Delete process, Delete project, Edit process See <a href="#">Add administrators, set permissions at the project-level or project collection-level</a> .
Process template manager ( <a href="#">Hosted XML process model</a> )	Work tracking customization	Edit collection-level information, See <a href="#">Add administrators, set permissions at the project-level or project collection-level</a> .
Project management ( <a href="#">On-premises XML process model</a> )	Work tracking customization	Edit project-level information, See <a href="#">Add administrators, set permissions at the project-level or project collection-level</a> .

Permissions manager	<p>Manage permissions for a project, account, or collection</p>	<p>For a project, Edit project-level information For an account or collection, Edit instance-level (or collection-level) information To understand the scope of these permissions, see <a href="#">Permission lookup guide</a>. To grant permissions, See <a href="#">Add administrators, set permissions at the project-level or project collection-level</a>.</p> <p>You can also grant permissions to manage permissions for the following objects:</p> <ul style="list-style-type: none"> <li>• <a href="#">Set Git repository permissions</a></li> <li>• <a href="#">Manage Git branch permissions</a></li> <li>• <a href="#">Set TFVC repository permissions</a></li> <li>• <a href="#">Administer build and release permissions</a></li> <li>• <a href="#">Manage Wiki permissions</a>.</li> </ul>
---------------------	---	---

## Limit visibility to organization and project information

By default, users added to an organization can view all organization and project information and settings. To restrict access to only those projects that you add users to, you can enable the **Limit user visibility for projects** preview feature for the organization. To enable this feature, see [Manage or enable features](#).

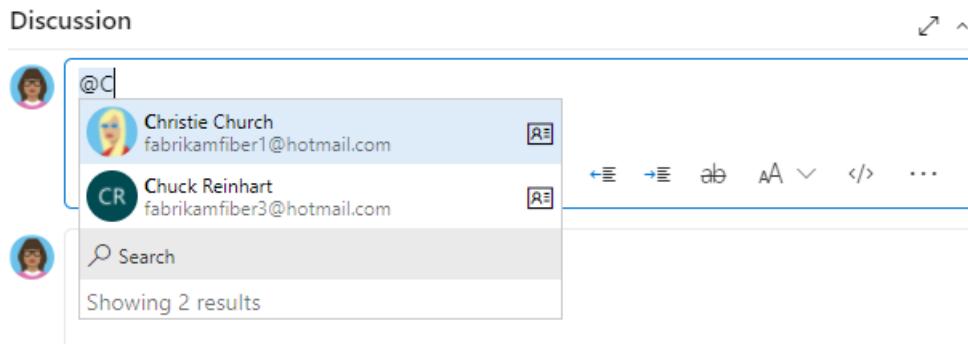
With this feature enabled, users added to the **Project-SScoped Users** group can't view most **Organization settings** and can only connect to those projects to which they've been added.

## Limit people picker to project users and groups

For organizations that manage their users and groups using Azure Active Directory (Azure AD), people pickers provide support for searching all users and groups added to Azure AD, not just those added to a project. People pickers support the following Azure DevOps functions:

- Selection of a user identity from a work tracking identity field such as **Assigned To**
- Selection of a user or group using **@mention** in a work item discussion or rich-text field, a pull request discussion, commit comments, or changeset or shelveset comments
- Selection of a user or group using **@mention** from a wiki page

As shown in the following image, you simply start typing into a people picker box until you find a match to a user name or security group.



Users and groups who are added to the **Project-SScoped Users** group can only see and select users and groups in the project they are connected to from a people picker. To scope people pickers for all project members, see [Manage your project, Limit identity search and selection](#).

## Restrict access to view or modify objects

Azure DevOps is designed to enable all valid users to view all objects defined in the system. You can restrict access to resources by setting the permission state to **Deny**. You can set permissions for members that belong to a custom security group or for an individual user. To learn more about how to set these types of permissions, see [Change individual permissions, grant select access to specific functions](#).

Area to restrict	Permissions to set to Deny
View or contribute to a repository	View, Contribute See <a href="#">Set Git repository permissions</a> or <a href="#">Set TFVC repository permissions</a> .
View, create, or modify work items within an area path	Edit work items in this node, View work items in this node See <a href="#">Set permissions and access for work tracking</a> , <a href="#">Modify work items under an area path</a> .
View or update select build and release pipelines	Edit build pipeline, View build pipeline Edit release pipeline, View release pipeline You set these permissions at the object level. See <a href="#">Set build and release permissions</a> .
Edit a dashboard	View dashboards See <a href="#">Set dashboard permissions</a> .

## Restrict modification of select fields based on a user or group

For the Inheritance process model, you can customize work item types to restrict who can modify a specific field for a work item type. You restrict modification by adding a custom rule to the work item type.

Using one of the following two conditions, you can make select fields required for a user of a security group or who are not a member of a security group.

- current user is a member of a group...
- current user is not a member of a group...

For example, you can make the Title or the State field Read-only for select users or groups.

For example, the Priority field, for the User Story work item type, becomes read-only for members of the Fabrikam Fiber\Voice group. When a user of this group opens a User Story, they are unable to change the value on the Priority field.

To learn more, see [Add a rule to a work item type \(Inheritance process\)](#).

### NOTE

For Azure DevOps Server 2019 and earlier versions, you can only restrict modification of work items based on a user or group with the On-premises XML process model.

For the Inheritance process model, you can customize work item types to restrict who can modify a specific field for a work item type. You restrict modification by adding a custom rule to the work item type.

Using one of the following two conditions, you can make select fields required for a user of a security group or who are not a member of a security group.

- current user is a member of a group...

- `current user is not a member of a group...`

For example, you can make the Title or the State field Read-only for select users or groups.

For example, the Priority field, for the User Story work item type, becomes read-only for members of the Fabrikam Fiber\Voice group. When a user of this group opens a User Story, they are unable to change the value on the Priority field.

To learn more, see [Add a rule to a work item type \(Inheritance process\)](#).

For the [On-premises XML process model](#), you can customize work item types to support these restriction requests:

- Restrict who can create or modify a work item
- Restrict who can create specific work item types, such as Epics or Features

For example, you can restrict modification of work items by adding a rule to the work item type, usually within the **WORKFLOW** section. To learn more, see [Add a rule to a work item type, Apply or ignore rules based on user or group](#).

You restrict access to work tracking objects in one of two ways:

- Set a [condition field rule](#), a [condition-based field rule](#) or a combination of the two that applies to a group. You can restrict changes from being made to a field by specifying a qualifying rule and making it apply for a specific group. Conditional rules can include **CANNOTLOSEVALUE**, **EMPTY**, **FROZEN**, **NOTSAMEAS**, **READONLY**, and **REQUIRED** elements.
- By [adding WITs to the Hidden Categories group](#), you can prevent the majority of project contributors from creating them. You [can create a hyperlink to a template](#) that opens the work item form and share that link with those team members who do want to create them.

## Restrict modification of closed work items

Depending on your business processes, you may want to prevent users from continuing to modify or update work items that have been closed or completed. You can add rules to work item types to prevent users from re-opening closed work items.

For the Inherited process, you can add a rule that restricts state transition. For example, the following rule restricts transitioning from closed to the other two States, New and Active.

### NOTE

The `A work item state moved from ...` condition is only available for Azure DevOps Services and only to those participating in the Private Preview. For details, see [State transition restriction rules \(private preview\)](#).

For more information on applying rules to a workflow, see [Apply rules to workflow states \(Inheritance process\)](#).

### NOTE

Depending on the rule action you specify, either the **Save** button on the work item form may be disabled, or an error message displays when a restricted user attempts to modify the work item.

Depending on your business processes, you may want to prevent users from continuing to modify or update

work items that have been closed or completed. You can add rules to work item types to prevent users from re-opening closed work items.

For on-premises deployments, you can add rules to a work item type to prevent re-opening after a work item has been closed. For example, the following workflow transition rules allow Testers to reopen a work item, but not members of the Developers group.

```
<TRANSITION from="Closed" to="New"
  for="[Project]\Testers"
  not="[Project]\Developers">
  . .
</TRANSITION>
<TRANSITION from="Closed" to="Active"
  for="[Project]\Testers"
  not="[Project]\Developers">
  . .
</TRANSITION>
```

To learn more, see [Apply a field rule](#).

## Next steps

[Remove user accounts](#)

## Related articles

- [Troubleshoot permissions](#)
- [Default permissions and access](#)
- [Permission lookup guide](#)
- [Get started with permissions, access, and security groups](#)
- [Permissions and groups reference](#)
- [Set permissions at the project-level or project collection-level](#)

# Plan your organizational structure

5/21/2021 • 15 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Your business structure should act as a guide for the number of organizations, projects, and teams that you create in Azure DevOps. This article helps you plan for different structures and scenarios for Azure DevOps.

Consider the following structures for your business or collaborative work in Azure DevOps:

- [Quantity of organizations](#)
- [Quantity of projects under an organization](#)

You also may want to plan for the following scenarios:

- [Mapping your organizations and projects](#) in Azure DevOps to your enterprise, business unit, and team structure
- [Structuring your repositories \(repos\)](#)
- [Structuring your teams](#)- it can either help or hinder teams to be Agile and autonomous
- [Managing access to data](#) - who needs to have access and who doesn't?
- [Reporting needs](#)
- Promoting common practices - learn more about [foundational elements you need to create an agile mindset and culture](#)

You need to have at least one organization, which may represent your company, your larger collection of code projects, or even multiple related business units.

## What is an organization?

An organization in Azure DevOps is a mechanism for organizing and connecting groups of related projects. Examples include business divisions, regional divisions, or other enterprise structures. You can choose one organization for your entire company, one organization just for you, or separate organizations for specific business units.

Each organization gets its own *free tier* of services (up to five users for each service type) as follows. You can use all the services, or choose just what you need to complement your existing workflows.

- [Azure Pipelines](#): One hosted job with 1,800 minutes per month for CI/CD and one self-hosted job
- [Azure Boards](#): Work item tracking and Kanban boards
- [Azure Repos](#): Unlimited private Git repos
- [Azure Artifacts](#): Package management
- Unlimited Stakeholders
  - Five Azure DevOps users (Basic)
  - Free tier of Microsoft-hosted CI/CD (one concurrent job, up to 30 hours per month)
  - 2 GiB of Azure Artifacts storage
  - One self-hosted CI/CD concurrent job

### Caution

The cloud-based load testing service is deprecated. More information about the deprecation, the service availability, and alternative services can be found [here](#).

# How many organizations do you need?

Start with just one organization in Azure DevOps. Then, you can add additional organizations—which may require different security models—later. A single code repo or project only needs one organization. If you have separate teams that need to work on code or other projects in isolation, consider creating separate organizations for those teams. They'll have different URLs. Add projects, teams, and repos, as necessary, before you add another organization.

Take some time to review your work structure and the different business groups and participants to be managed. To learn more, see [Mapping your projects to business units](#) and [Structure considerations](#).

## What is a team?

A team is a unit that supports many [team-configurable tools](#). These tools help you plan and manage work, and make collaboration easier.

### Creating a team for each distinct product or feature team

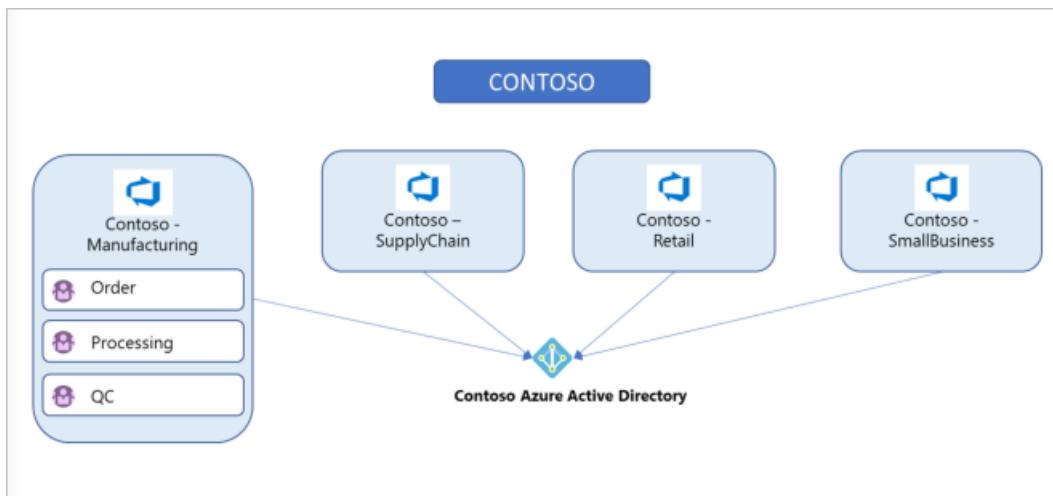
Every team owns their own backlog, to create a new backlog you create a new team. By [configuring teams and backlogs into a hierarchical structure](#), program owners can more easily track progress across teams, manage portfolios, and generate rollup data. A team group is created when you create a team. You can use this group in queries or to set permissions for your team.

## What is a project?

A project in Azure DevOps contains the following set of features:

- Boards and backlogs for agile planning
  - Pipelines for continuous integration and deployment
  - Repos for version control and management of source code and artifacts
  - Continuous test integration throughout the project life cycle
- Each organization contains one or more projects

In the following image, the fictitious Contoso company has four projects within their Contoso-Manufacturing organization.



## How many projects do you need?

You need to have at least one project to start using an Azure DevOps service, such as Azure Boards, Azure Repos, or Azure Pipelines. When you create your organization, a default project gets created for you. In your default project, there's a code repo to start working in, backlog to track work, and at least one pipeline to begin automating build and release.

Within an organization, you can do either of the following approaches:

- Create a single project that contains many repos and teams
- Create many projects, each with its own set of teams, repos, builds, work items, and other elements

Even if you have many teams working on hundreds of different applications and software projects, you can manage them within a single project in Azure DevOps. However, if you want to manage more granular security between your software projects and their teams, consider using many projects. At the highest level of isolation is an organization, where each organization is connected to a single Azure AD tenant. A single Azure AD tenant, however, can be connected to many Azure DevOps organizations.

#### NOTE

If the **Project-Scope Users well known group to hide settings** preview feature is enabled for the organization, users added to the **Project-Scope Users** group won't be able to access projects that they haven't been added to. To learn more, see [About projects and scaling your organization](#), [Project-scope Users group](#).

### Single project

A single project puts all of the work at the same "portfolio" level for the entire organization. Your work has the same set of repos and iteration paths. A single project allows teams to share source repos, build definitions, release definitions, reports, and package feeds. You might have a large product or service that's managed by many teams. Those teams have tight inter-dependencies on each other across the product life cycle. You create a project and divide the work using teams and area paths. This setup gives your teams visibility into each other's work, so the organization stays aligned. Your teams use the same taxonomy for work item tracking, making it easier to communicate and stay consistent.

#### TIP

When multiple teams work on the same product, having all teams on the same iteration schedule helps keep your teams aligned and delivering value on the same cadence. For example, the organization in Azure DevOps has over 40 feature teams and 500 users within a single project - this works well because we're all working on a common product set with common goals and a common release schedule.

A high volume of queries and boards can make it hard to find what you're looking for. Depending on the architecture of your product, this difficulty can bleed into other areas such as builds, releases, and repos. Make sure to use good naming conventions and a simple folder structure. When you add a repo to your project, consider your strategy and determine whether that repo could be placed into its own project.

### Many projects

Project structure is best determined by how you ship the product. Having several projects shifts the administration burden and gives your teams more autonomy to manage the project as the team decides. It also provides greater control of security and access to assets across the different projects. Having team independence with many projects creates some alignment challenges, however. If each project is using a different process or iteration schedule, it can make communication and collaboration difficult if the taxonomies aren't the same.

#### TIP

If you use the same process and iteration schedules across all your projects, your ability to roll-up data and report across teams is improved.

Azure DevOps provides cross-project experiences when it comes to managing work.

You may want to add another project because of the following scenarios:

- To prohibit or manage access to the information within a project
- To support custom work tracking processes for specific business units within your organization
- To support entirely separate business units that have their own administrative policies and administrators
- To support testing customization activities or adding extensions before rolling out changes to the working project

When you're considering many projects, keep in mind that Git repo portability makes it easy to migrate repos (including full history) between projects. Other history can't be migrated between projects. Examples are push and pull request history.

When you map projects to business units, your company gets a single organization and sets up many projects with one or more projects representing a business unit. All Azure DevOps assets of the company are contained within this organization and located within a given region (for example, Western Europe). Consider the following guidance for mapping your projects to business units:

	ONE PROJECT, MANY TEAMS	ONE ORGANIZATION, MANY PROJECTS, AND TEAMS	MANY ORGANIZATIONS
<b>General guidance</b>	Best for smaller organizations or larger organizations with highly aligned teams.	Good when different efforts require different processes.	Useful as part of TFS legacy migrations and for hard security boundaries between organizations. Used with multiple projects and teams within each organization.
<b>Scale</b>	Supports tens of thousands of users and hundreds of teams, but best at this scale if all teams are working on related efforts.	Same as with one project, but many projects may be easier.	
<b>Process</b>	Aligned processes across teams; team flexibility to customize boards, dashboards, and so on.	Independent processes for each project. For example, different work item types, custom fields, and so on.	Same as many projects.
<b>Collaboration</b>	Highest default visibility and reuse between work and assets of different teams.	Good visibility and reuse are possible, but it's easier to hide assets between projects whether intentional.	Poor visibility, collaboration, and reuse between organizations.
<b>Roll-up reporting and portfolio management</b>	Best ability to roll up across teams and coordinate between teams.	Good reporting possible across projects. More difficult for cross-project roll-up and team coordination.	No roll-up or coordination between organizations.
<b>Security/isolation</b>	Can lock down assets at a team level, but default is open visibility and collaboration.	Better ability to lock down between projects. By default, provides good visibility within projects and good isolation across projects.	Hard boundaries across organizations; excellent isolation and minimal ability to share across organizations.

	ONE PROJECT, MANY TEAMS	ONE ORGANIZATION, MANY PROJECTS, AND TEAMS	MANY ORGANIZATIONS
<b>Context switching</b>	Easiest for teams to work together and for users to switch between efforts.	Relatively easy for users to work together and switch contexts between efforts.	More difficult for users having to work across different organizations.
<b>Information overload</b>	By default, all assets are visible to users who make use of "favorites" and similar mechanisms to avoid "information overload."	Reduced risk of information overload; most project assets hidden across project boundaries.	Assets across organizations are isolated, reducing risk of information overload.
<b>Administrative overhead</b>	Much administration is delegated down to individual teams. Easiest for user licensing and org-level administration. Additional work may be needed if alignment is required between efforts.	Additional administration at the project level. Additional overhead, but can be useful when projects have different administrative needs.	As with additional projects, there's additional administrative overhead, which enables additional flexibility between orgs.

## Structure repos and version control within a project

Consider the specific strategic work scoped to one of the organizations you created previously and who should have access. Use this information to name and [create a project](#). This project has a URL defined under the organization you created it in and can be accessed at <https://dev.azure.com/{organization-name}/{project-name}>.

Configure your project by visiting its URL and selecting **Project settings** at the lower left portion of the page.

The screenshot shows the Azure DevOps Project Overview page for 'FabrikamFiber'. The left sidebar contains a list of project management tools: Overview, Summary, Dashboards, Wiki, Boards, Repos, Pipelines, Test Plans, and Artifacts. The 'Project settings' link is highlighted with a red box. The main content area features a cartoon illustration of a person working at a desk with a dog. Below the illustration, the text 'Welcome to the project!' is displayed, followed by a list of service options: Boards, Repos, Pipelines, Test Plans, and Artifacts. A blue 'Invite' button is located in the top right corner.

To learn more about managing projects, see [Manage projects in Azure DevOps](#). You can move a project to a different organization by migrating the data. To learn more about migrating your project, see [Migration options](#).

## Managing version control

In projects where the Azure Repos service is enabled, version control repos can store and revise code. Consider the following options when you're configuring repos.

### Git vs. Team Foundation Version Control (TFVC)

Azure Repos offers the following version control systems for teams to choose from:

- Git and TFVC. Projects can have repos of each type. By default, new projects have an empty Git repo. Git enables a great amount of flexibility in developer workflows and integrates with nearly every relevant tool in the developer ecosystem. Any project can use Git repos. There's no limit on the number of Git repos that can be added to a project.

TFVC is a centralized version control system that is also available. Unlike Git, only one TFVC repository is allowed for a project. But, within that repo, folders, and branches are used to organize code for multiple products and services, if wanted. Projects can use both TFVC and Git, if appropriate.

### One vs. many repos

Do you need to set up multiple repos within a single project or have a repo set up per project? The following guidance relates to the planning and administration functions across those repos.

One project containing multiple repos works well if the products/services are working on a coordinated release schedule. If developers are frequently working with multiple repos, keep them in a single project to ensure the processes remain shared and consistent. It's easier to manage repo access within a single project, as access controls and options like case enforcement and max file size are set at the project level. You can manage the access controls and settings individually, even if your repos are in a single project.

If the products stored in multiple repos work on independent schedules or processes, you can split them into multiple projects. Git repo portability makes it easy to move a repo between projects and still keep full-fidelity commit history. Other history, such as pull requests or build history, aren't easily migrated.

Your decision for one versus many repos should be largely based on code dependencies and architecture. A good first rule to apply is to put each independently deploy-able product or service in its own repo. Don't separate a codebase into many repos if you expect to make coordinated code changes across those repos, as there are no tools to assist in coordinating those changes. If your codebase is already a monolith, keep it in one repo. For more information about monolithic repos, see [How Microsoft develops modern software with DevOps](#) articles. If you have many disconnected services, one repo per service is a good strategy.

#### NOTE

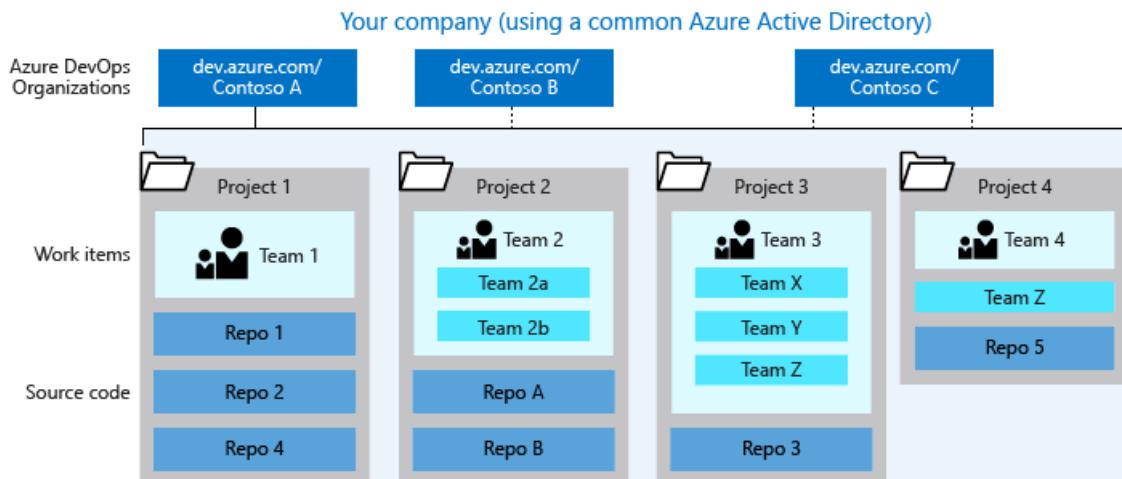
Consider [managing your permissions](#) so not everyone in your organization can [create a repo](#). A challenge growing teams or companies face is the rapid proliferation of repos. If you have too many repos, it's hard to keep track of who owns which code or other content stored in those repos.

### Shared repo vs. forked repos

We recommend using a shared repo within a trusted organization. Developers use branches to maintain isolation of their changes from one another. Used with a good branching and release strategy, a single repo can scale to support concurrent development for more than a thousand developers. For more information about branching and release strategy, see [Adopt a Git branching strategy and Release Flow: Our Branching Strategy](#).

Forks can be useful when you're working with vendor teams that shouldn't have direct access to update the main repository. Forks can also be useful in scenarios where many developers contribute infrequently, such as in an open-source project. When you're working with forks, you may want to maintain a separate project to isolate the forked repos from the main repo. There may be added administrative overhead, but it keeps the main project cleaner. For more information, see the [Forks article](#).

The following image displays a sample of how "your company" could structure its organizations, projects, work items, teams, and repos.



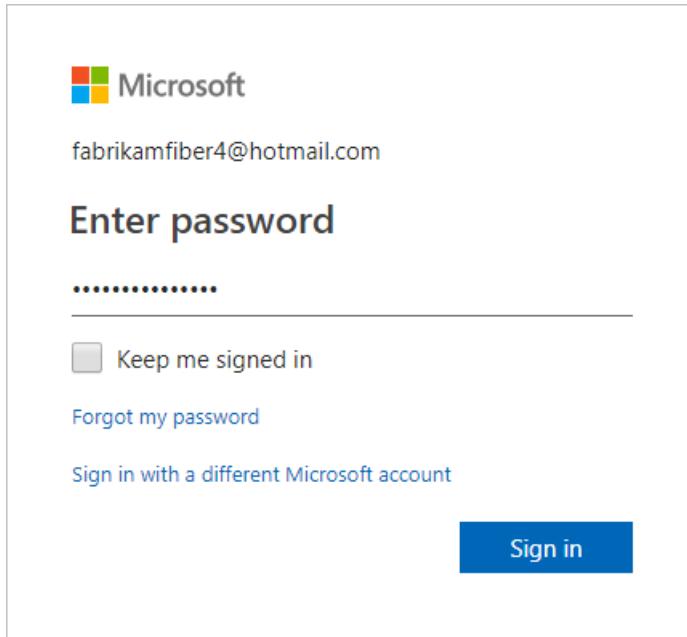
### More about organizational structure

#### Choosing your organization administrator account type

When you create an organization, the credentials that you sign in with define which identity provider your organization uses. Create your organization with a Microsoft account or Azure AD instance. Use those credentials to sign in as an administrator to your new organization at <https://dev.azure.com/{YourOrganization}>.

#### Using your Microsoft account

Use your Microsoft account if you don't need to authenticate users for an organization with Azure AD. All users must sign in to your organization with a Microsoft account. If you don't have one, you can [create a Microsoft account](#) now.



If you don't have an Azure AD instance, create one for free from the [Azure portal](#) or use your Microsoft account to create an organization. Then, you can [connect your organization to Azure AD](#).

#### Using your Azure AD account

You might have an Azure AD account already if you use Azure or Microsoft 365. If you work for a company that uses Azure AD to manage user permissions, you probably have an Azure AD account.

If you don't have an Azure AD account, learn how to [sign up for Azure AD](#) to automatically connect your organization to your Azure AD. All users must be members in that directory to access your organization. To add users from other organizations, use [Azure AD B2B collaboration](#).

Azure DevOps authenticates users through your Azure AD, so that only users who are members in that directory have access to your organization. When you remove users from that directory, they can no longer access your organization. Only specific [Azure AD administrators](#) manage users in your directory, so administrators control who accesses your organization.

For more information on managing users, see [Manage users](#).

#### Mapping organizations to business units

Each business unit within your company gets its own organization in Azure DevOps, along with its own Azure AD tenant. You can [set up projects](#) within those individual organizations, as required, based on teams or ongoing work.

For a larger company, you can create multiple organizations using different user accounts (most likely Azure AD accounts). Consider what groups and users share strategies and work, and group them into specific organizations. For example, the (fictional) Fabrikam company might create three organizations: Fabrikam-Marketing, Fabrikam-Engineering, and Fabrikam-Sales. Each organization has a separate URL, such as <https://dev.azure.com/Fabrikam-Marketing>, <https://dev.azure.com/Fabrikam-Engineering>, and <https://dev.azure.com/Fabrikam-Sales>. The organizations are all for the same company, but are mostly isolated from each other. You don't need to have anything separated, however you should only create boundaries when it

makes sense to your business. You can more easily partition an existing organization with projects, than combine different organizations.

## Related articles

- [Create an organization](#)
- [Create a project](#)
- [Connect your organization to Azure AD](#)
- [Set up billing](#)
- [Set user preferences](#)

# What is source control?

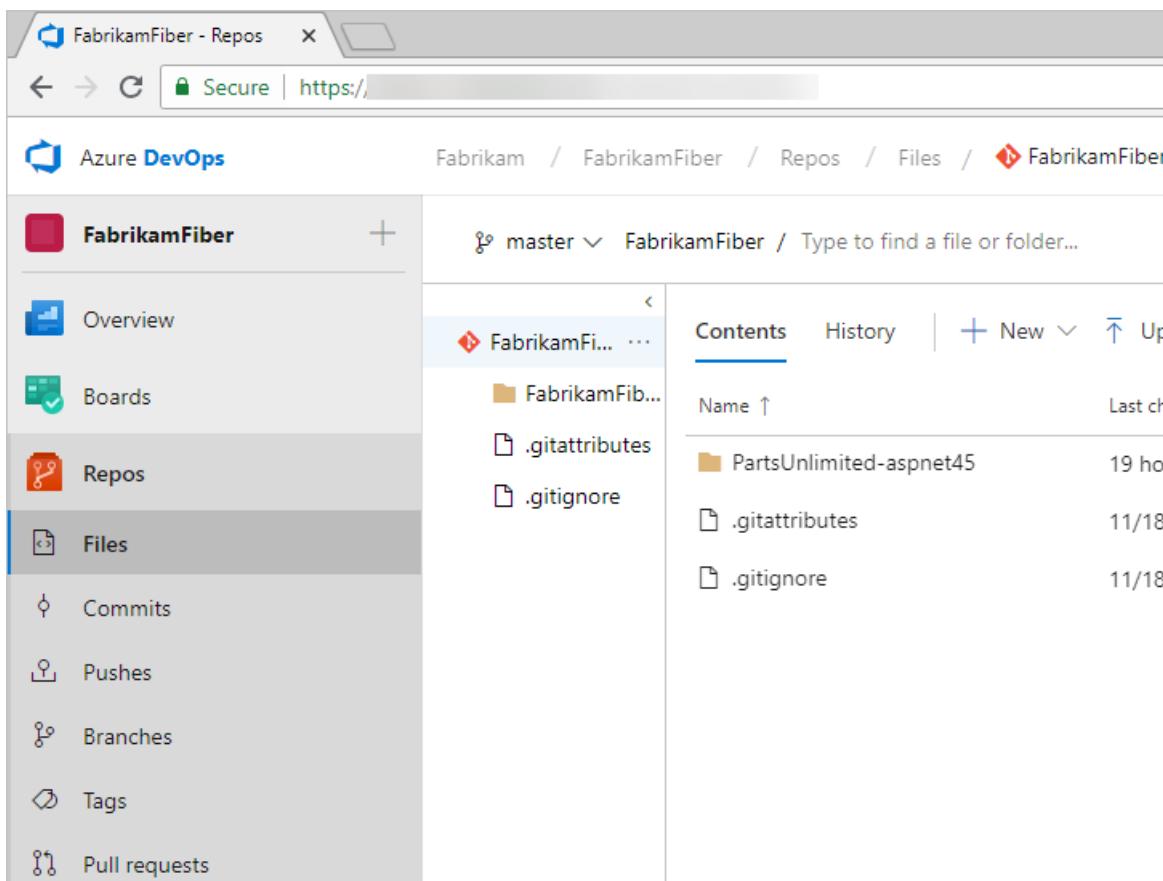
3/6/2021 • 2 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

A source control system, also called a *version control* system, allows developers to collaborate on code and track changes. Source control is an essential tool for multi-developer projects.

Our systems support two types of source control: Git (distributed) and Team Foundation Version Control (TFVC). TFVC is a centralized, client-server system. In both Git and TFVC, you can check in files and organize files in folders, branches, and repositories.

Manage your repos, branches, and other code development operations from Azure Repos.



The screenshot shows the Azure DevOps interface for the 'FabrikamFiber' repository. The left sidebar includes links for Overview, Boards, Repos (selected), Files, Commits, Pushes, Branches, Tags, and Pull requests. The main content area displays the 'master' branch of the 'FabrikamFiber' repository. It shows a list of files: '.gitattributes' and '.gitignore'. The interface includes standard navigation and search features, such as a back button, a search bar, and a 'New' button for creating new files or folders.

With Git, each developer has a copy of the source repository on their dev machine. The source repo includes all branch and history information. Each developer works directly with their local repository. Changes are shared between repositories as a separate step.

Developers can commit each set of changes and perform version control operations, such as history and compare without a network connection. Branches are lightweight. When developers need to switch contexts, they create a private local branch. Developers can quickly switch from one branch to another to pivot among different variations of the code base. Later, developers can merge, publish, or dispose of the branch.

#### **NOTE**

Git in Visual Studio and Azure DevOps is standard Git. You can use Visual Studio with third-party Git services. You can also use third-party Git clients with Azure DevOps Server.

With TFVC, developers have only one version of each file on their dev machines. Historical data is maintained only on the server. Branches are path-based and are created on the server.

## Next steps

Start sharing your code or get your code by using source control.

[Code with Git](#)

## Related articles

- [Azure Repos documentation](#)
- [Git repositories documentation](#)

# Tools and clients that connect to Azure DevOps

3/17/2021 • 6 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Our platform of software development tools began more than 20 years ago. We released Visual Basic and Visual Studio as an integrated development environment (IDE). Visual Studio supports many plug-ins that extend its functionality. In particular, the Team Explorer plug-in allows the Visual Studio client to connect to Azure DevOps to support source control, work tracking, build, and test operations.

## Desktop client developer tools

Developers have access to many tools through these versions of Visual Studio and plug-ins. To download any version of Visual Studio, go to the [Visual Studio Downloads page](#). To understand what features you get with the Visual Studio versions, see [Compare Visual Studio offerings](#).

- **Visual Studio Community:** A fully featured and extensible IDE for creating modern applications for Android, iOS, and Windows, including web applications and cloud services. (Replaces Visual Studio Express.)
- **Visual Studio Professional:** Development tools and services to support individual developers or small teams.
- **Visual Studio Enterprise:** Integrated, end-to-end development tools and solutions for teams of any size, and with a need to scale. It supports designing, building, and managing complex enterprise applications.
- **Visual Studio Test Professional:** Provides access to Microsoft Test and development tools to support quality and collaboration throughout the development process.
- **Visual Studio Team Explorer:** Free solution for non-developers to interact with Azure DevOps.
- **Eclipse/Team Explorer Everywhere:** Free plug in to support teams running Eclipse on Linux, macOS, or Windows that connects to Azure DevOps.
- **Android Studio with the Azure DevOps Services Plug-in for Android Studio:** Free plug in to support Android developers and connect to Git repositories on Azure DevOps.
- **IntelliJ with the Azure DevOps Services Plugin for IntelliJ:** Free plug in to support developers who use IntelliJ IDEA or Android Studio to connect to Git repositories on Azure DevOps.
- **Visual Studio Code:** Free, open-source code editor with a free extension to support connecting to Git repositories on Azure DevOps.

To get started with client libraries, see [Client library samples](#).

### Team Explorer plug-in

Team Explorer, a plug-in to all Visual Studio versions, connects Visual Studio to projects defined in Azure DevOps. You can manage source code, work items, and builds. To learn more, see [Work in Team Explorer](#).

## HOME PAGE WITH GIT

The screenshot shows the Team Explorer - Home interface for a Git repository named 'Fabrikam Fiber'. The top navigation bar includes links for Home, Search Work Items, and a refresh icon. Below the navigation, the 'Team Foundation Server' section displays the repository name and URL. The 'Project' section contains links for Web Portal, Task Board, and Team Room. A sidebar on the left lists various project management tasks: Changes, Branches, Pull Requests, Sync, Work Items, Builds, Team Members, and Settings. At the bottom, there's a 'Solutions' section with 'New...' and 'Open...' options, and a note stating 'There were no solutions found.'

## HOME PAGE WITH TFVC

The screenshot shows the Team Explorer - Home interface for a TFVC repository named 'Fabrikam Fiber'. The layout is similar to the Git version, with a top navigation bar, 'Team Foundation Server' section, and 'Project' section with links for Web Portal, Task Board, and Team Room. The sidebar on the left lists: My Work, Pending Changes, Source Control Explorer, Work Items, Builds, Team Members, and Settings. The 'Solutions' section at the bottom indicates 'There were no solutions found.'

## Office integration tools

You can integrate the following Microsoft Office tools with Azure DevOps.

- [Excel](#): Use Excel to add and bulk modify work items.
- [Project](#): By using Project, you can plan projects, schedule tasks, assign resources, and track changes. You have access to more features, such as a project calendar, Gantt charts, and resource views.

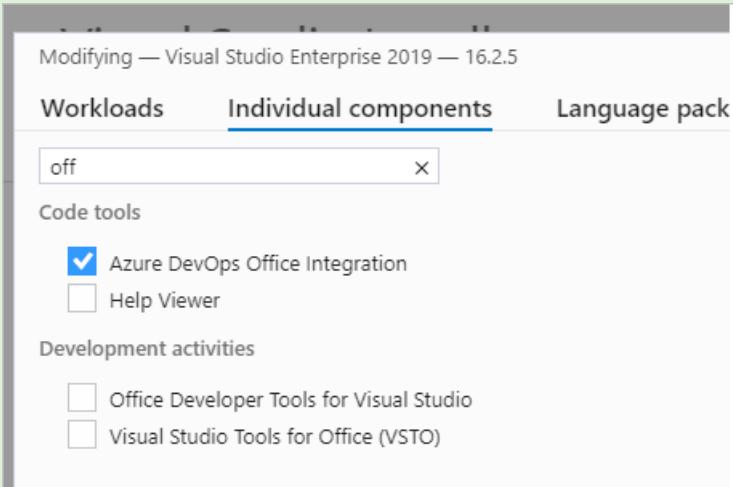
### IMPORTANT

Starting with Visual Studio 2019, the Team Foundation plug-in for Office is deprecating support for Microsoft Project. Project integration and the TFSFieldMapping command is not supported for Azure DevOps Server 2019 nor for Azure DevOps Services. However, you can continue to use Microsoft Excel.

- [Excel](#): Use Excel to add and bulk modify work items.
- [Project](#): By using Project, you can plan projects, schedule tasks, assign resources, and track changes. You have access to more features, such as a project calendar, Gantt charts, and resource views.
- [PowerPoint Storyboarding](#): Illustrate user stories and requirements by using PowerPoint.

#### TIP

Check to make sure the Azure DevOps Office Integration component is selected in the Visual Studio Installer, per the



following example.

When you install any edition of Visual Studio or [Team Foundation Server Standalone Office Integration 2015 \(free\)](#), the Team Foundation plug-in integrates work item tracking with select Office clients. The Team Foundation plug-in installs to your existing Office client. The plug-in supports Office 2007, Office 2010, or Office 2013 versions.

- [Excel](#): Use Excel to add and bulk modify work items.
- [Project](#): By using Project, you can plan projects, schedule tasks, assign resources, and track changes. You have access to features that TFS doesn't support, such as a project calendar, Gantt charts, and resource views.
- [PowerPoint Storyboarding](#): Illustrate user stories and requirements by using PowerPoint. The Team Foundation plug-in installs to your existing PowerPoint client.
- [Project Professional](#): With Project Professional and the Team Foundation Server Extensions for Project Server, you can manage projects that synchronize data that exists in both TFS and Project Server. Project managers and software development teams can use the tools that they prefer, work at the level of precision that supports their needs, and easily share information.

#### IMPORTANT

Support for integrating TFS with Project Server is deprecated for TFS 2017. However, synchronization support is provided by a Microsoft partner. See [Synchronize TFS with Project Server](#) for details.

#### Task-specific clients

The following clients support specific tasks, such as managing testing efforts, providing feedback, or modifying work items:

- [Azure Test Plans](#): Manage your test efforts, create and run manual tests, and create and track bugs that are found during test efforts.
- [Test & Feedback extension \(previously called the Exploratory Testing extension\)](#): This extension provides a lightweight plug-in to a web browser. Stakeholders can respond to feedback requests for user stories and features created in Azure DevOps. This extension is free to Stakeholders.
- [Microsoft Feedback Client](#): Your Stakeholders can use this client to record feedback for your application as video, audio, or type-written comments. This client is installed with all versions of Visual Studio, or it can be installed from the free download. All feedback is stored in the work item data store and requires Stakeholders to have permissions.

## IMPORTANT

Test Manager is deprecated for TFS 2017.

# Browser-based web tools

## Web portal

The collaboration tools supported through the web portal are summarized under [Essential services](#). New features are deployed every three weeks for Azure DevOps Services, and quarterly for Azure DevOps Server. For release notes, see [Azure DevOps Services Features Timeline](#).

You can use the following browsers to access the web portal:

VERSION	EDGE	INTERNET EXPLORER	SAFARI (MAC)	FIREFOX	CHROME
Azure DevOps Services Azure DevOps Server 2020 Azure DevOps Server 2019 TFS 2018 TFS 2017	Most recent	11 and later	9.1 and later	Most recent	Most recent
TFS 2015	Most recent	9 and later	5 and later	Most recent	Most recent
TFS 2013		9 and later	5 and later	Most recent	Most recent

Microsoft Edge, Firefox, and Chrome automatically update themselves, so Azure DevOps supports the most recent version.

To learn more, see [Web portal navigation](#).

## Browser-based extensions

The following extensions are available and are built and maintained by the Azure DevOps Services product team:

- [Code search](#): Increase cross-team collaboration and code sharing. Enables developers to quickly locate relevant information within the code base of all projects that are hosted within an organization or collection. You can discover implementation examples, browsing definitions, and error text.
- [Work item search](#): To quickly find relevant work items, search across all work item fields over all projects in an organization. Do full-text searches across all fields to efficiently locate relevant work items. Use inline search filters, on any work item field, to quickly narrow down a list of work items.

Find more extensions in Azure DevOps [Organization settings > Extensions > Browse marketplace](#).

# Command-line tools

You can do many code development and administrative tasks by using the following command-line tools:

- [Git commands](#)
- [TFVC commands](#)
- [TFSConfig](#)
- [TFSDeleteProject](#)

- [TFSSecurity](#)
- [TFSServiceControl](#)
- [witadmin \(work item tracking\)](#)

## Marketplace extensions

Visual Studio and Azure DevOps provide a wealth of features and functionality. They also provide a means to extend and share that functionality.

Extensions are simple add-ons that you can use to customize and extend your DevOps and work tracking experiences. They're written with standard technologies—HTML, JavaScript, and CSS. You can develop your own extensions by using your preferred dev tools.

You build extensions by using our RESTful API library. Publish your extensions to the Azure DevOps Marketplace. You can privately maintain or share them with millions of developers who use Visual Studio and Azure DevOps.

To learn more, visit the [Azure DevOps Marketplace](#) and see [Overview of extensions](#).

## REST APIs

The Azure DevOps APIs are based on REST, OAuth, JSON, and service hooks—all standard web technologies broadly supported in the industry.

REST APIs are provided to support building extensions to Azure DevOps. To learn more, see [REST API overview](#).

## Related articles

- [A tour of services](#)
- [Software development roles](#)
- [Pricing](#)
- [Azure DevOps data protection overview](#)

# Software development roles supported by Azure DevOps

3/6/2021 • 4 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

If you're a sole developer or work in a small setting, you track issues, plan features, code, test, build, and deploy.

If you work in a large setting, you may be more focused on a specific set of tasks that aligns with specific roles. These specific roles could be software development, project management, or DevOps.

This article describes the features and tasks available to you, based on your role.

## Contributor roles

Team members are contributors who have access to the following areas and more:

- code base
- work item tracking
- Agile tools
- build pipelines
- test tools

If you need to lock down specific areas to a select set of contributors, see [permission management](#).

### Software developers

Developers use Visual Studio or other [tools](#) to develop their applications. They then check in their changes to a Git or Team Foundation Version Control (TFVC) repository hosted in Azure DevOps. From the web portal or a supported IDE, they can view repositories, check history, and more.

- To get started with using Git, see one of the following resources:
  - [Share your code with Git and Visual Studio](#)
  - [Share your code in Git by using Eclipse](#)
  - [Share your code in Git by using Xcode](#)
  - [Share your code in Git by using IntelliJ](#)
  - [Get started with using Git and Azure DevOps Services](#)
- To get started with using TFVC, see one of the following resources:
  - [Develop and share your code in TFVC by using Visual Studio](#)
  - [Share your code in TFVC by using Eclipse](#)
  - [Share your code in TFVC by using Xcode](#)

### Project managers

Project managers (PMs) typically plan the feature set to deliver, set priorities, and track the status of work, code defects, and customer issues. The suite of web-based Agile tools provides PMs with the views and features that they need to do these tasks. All work is captured within a work item. Each work item represents a specific type such as a user story, task, or bug.

- Use the product backlog to quickly define and prioritize user stories, features, and other work items
- Use the sprint backlog and task board to implement Scrum practices

- Use the Kanban board to work with Kanban methods
- Use queries to list and update work items, create status and trend charts, and post charts to dashboards
- Use dashboards to share information, status, and trends with your team or organization

For more information about getting started, see [About Azure Boards and Agile tools](#).

You can integrate Microsoft Excel and Microsoft Project with Azure DevOps to plan and track your work. For more information, see [Bulk modify by using Excel](#) and [Create your backlog and tasks by using Project](#).

### **DevOps: builders, testers, and release managers**

An advantage of working with Azure DevOps is the suite of tools and integrated functionality that support build, testing, and deploying software applications. See the following general DevOps-associated tasks that Azure DevOps supports.

- Define builds
- Unit test your code
- Run tests with your builds
- Perform exploratory tests
- Define, manage, track, and approve releases
- Deploy applications to Azure, a virtual machine, Docker containers, and more

To get started, see the overviews in [Azure Pipelines](#) and [Azure Test Plans](#).

### **Stakeholders**

With Stakeholder access, anyone in your organization can check project status and provide feedback.

Stakeholders can track project priorities and provide direction, feature ideas, and business alignment to a team.

Stakeholders also contribute to plans by adding and modifying work items. They can't, however, contribute to the code base or exercise test tools.

Stakeholder access essentially provides free access to a limited set of feature to project sponsors and supporters. To learn more, see [Work as a Stakeholder](#).

## **Administrator roles**

A distinct advantage to working in Azure DevOps Services is the reduced overhead of server maintenance. But there are several administrative tasks required to support a collaborative, integrated software development environment.

The main tasks are grouped as follows by membership in a security group or role.

### **Team administrators**

Responsible for configuring team settings, which include:

- Backlog and board settings
- Team areas and iterations (sprints)
- Team members
- Team dashboards
- Team work item templates
- Team alerts

To get started, see [Manage teams and configure team tools](#).

### **Project administrators**

Responsible for configuring project-level resources, including:

- [Area paths](#) and [iteration paths](#)
  - [Project permissions](#) and [repository security](#)
  - [Build agents, pools, and service connections](#)
  - [Test](#) and [release](#) retention policies
- 
- [Area paths](#) and [iteration paths](#)
  - [Project permissions](#) and [repository security](#)
  - [Customizing work tracking objects](#)
  - [Build agents, pools, and service connections](#)
  - [Test](#) and [release](#) retention policies

## Organization Owners and Project Collection Administrators

Responsible for configuring organization-level resources, including the following tasks:

- Manage billing
- Add and manage projects
- Manage collection-level permissions
- Customize work tracking processes
- Install and manage extensions

To get started, see [Manage organizations](#) and [Settings](#).

## Project Collection Administrators

Responsible for configuring collection-level resources. These tasks include:

- Add and manage projects
- Manage collection-level permissions
- Install and manage extensions

To get started, see [Settings](#).

## Azure DevOps Server administrators

Responsible for installing, upgrading, and maintaining an on-premises Azure DevOps Server deployment, including the:

- Install Azure DevOps Server
- Update servers running Azure DevOps Server
- Manage database backups
- Manage server administrative settings and permissions
- Build retention policies
- Add and manage project collections

To get started, see [Server Administration \(Azure DevOps Server\)](#).

## Related articles

- [A tour of services](#)
- [Plan your organizational structure in Azure DevOps](#)

# Troubleshoot connecting to a project

3/6/2021 • 5 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

## Troubleshoot connectivity

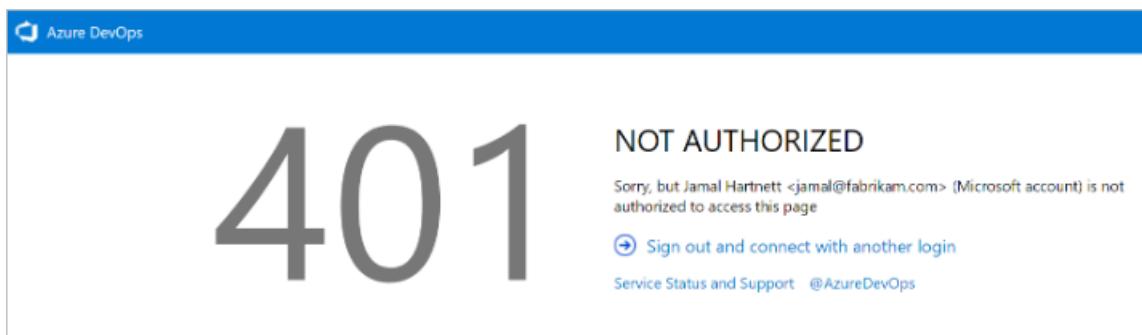
As a first step in resolving connectivity issues with Azure DevOps, complete the following steps:

1. Sign out of your browser. To do so, select the [Visual Studio sign out](#) link.
2. Delete the cookies in your browser. To delete cookies in most browsers, select Ctrl+Shift+Del.
3. Open Internet Explorer and delete the browser cookies. The Visual Studio IDE uses Internet Explorer cookies.
4. Close all browsers and close the Visual Studio IDE.
5. Use a private browser session to retry the connection. If the issue is with the Visual Studio IDE, remove the connection, and then readd it.

## Troubleshoot signing in

Two types of identities can sign in: Microsoft accounts and Azure Active Directory (Azure AD) accounts. Depending on your account, you might experience one of the following errors.

401 - Not Authorized



The most common error page is the *401 Not Authorized* error, which occurs when your identity doesn't have permissions to enter an organization. See the following common reasons for the error:

- Your identity isn't a member of the organization.
- Your identity has an invalid or missing license assignment.
- Your identity doesn't have enough memberships to access the resource. For example, membership to the Reader/Contributors group.
- Your identity is a B2B guest in the tenant, and the invitation hasn't been accepted.

If you think you're a member of the organization, but are blocked by this error page, [contact Support](#).

### Scenario 1

Your work or school Azure AD account doesn't have access, but your personal Microsoft account does.

The screenshot shows a large '401' error code. Below it, the text 'NOT AUTHORIZED' is displayed. A note states: 'Jamal@fabrikam.com has multiple accounts associated with it. Your work or school account does not have access to dev.azure.com/Fabrikam, but your personal account does have access.' It includes two links: 'Sign in with your personal account' and 'Sign out and connect with another login'. At the bottom, there's a link for 'Service Status and Support @AzureDevOps'.

A highly specific 401 error case. In this case, both a personal Microsoft account and a work or school account (Azure AD) that have the same sign-in address exist. You've signed in with your work or school account, but your personal account is the identity with access to the organization.

### Mitigation

In some cases, you might not know you have two identities with the same sign-in address. The work or school Azure AD account might have been created by an administrator when you were added to Office365 or Azure AD.

To sign out of your current work or school Azure AD account, select **Sign in with your personal MSA account**, and then sign in by using your personal Microsoft account. After authentication, you should have access to the organization.

- If you can't access to the organization, make sure that your Azure Active Directory still exists and that your work or school account is in the Azure AD tenant.

### TIP

To avoid seeing this prompt, you can rename your Microsoft account. Then, only one identity, your work or school account, or Azure AD account, uses your sign-in address.

### Scenario 2

Your personal Microsoft account doesn't have access, but your Azure AD account does. This scenario is an opposite version of the 401 error page. In this case, the personal account (Microsoft account identity) doesn't have access to the organization and the work or school account (Azure AD identity) does. The same guidance from Scenario 1 applies, but in reverse.

The screenshot shows a large '401' error code. Below it, the text 'NOT AUTHORIZED' is displayed. A note states: 'jamal@fabrikam.com has multiple accounts associated with it. Your personal account does not have access to dev.azure.com/Fabrikam, but your work or school account does have access.' It includes two links: 'Sign in with your work or school account' and 'Sign out and connect with another login'. At the bottom, there's a link for 'Service Status and Support @AzureDevOps'.

### Mitigation

When you get redirected back to the original sign-in page, we recommend that you clear all cookies, and then reattempt to sign in. If that doesn't fix the issue, [contact Support](#).

## Troubleshoot Azure DevOps Server connectivity

Here's a list of the most frequently reported connection problems and what to do about them. Complete the list in the order indicated.

1. Verify that you have the required permissions.

If the errors that you receive indicate read-only or blocked actions, you might not have permissions to act on the data.

2. Verify that your computer is connected to the network and that it can access network resources.

3. Verify that Azure DevOps Server hasn't been taken offline. Talk with your Azure DevOps Server administrator.

4. Check whether your project has been moved to another project collection in Azure DevOps Server. If it has been moved, you must create a connection to the new server name.

For additional troubleshooting tips, see [TF31002: Unable to connect to this Azure DevOps Server](#).

## Switch organizations

When you use two or more organizations that are linked to Azure AD, the sign-out function might not work as expected. For example, you can't switch between different organizations to connect to multiple organizations that are linked to directory tenants.

When this problem occurs, a blank screen flashes several times. Then, one of the following error messages appears after you connect to or add a new connection in the **Connect to Azure DevOps Server** dialog box:

TF31003: Either you have not entered the necessary credentials, or your user account does not have permission to connect to the Azure DevOps Server

TF31002: Unable to connect to this Azure DevOps Server

To resolve this issue, apply Visual Studio 2013.2 or install a later version from the [Visual Studio download website](#).

Another solution is to delete your browser cookies. For more information, see the support article [You can't switch between different organizations in Visual Studio Codespaces](#).

## Connect to Azure DevOps Server with Secure Sockets Layer

If you connect to an Azure DevOps Server instance that has Secure Sockets Layer (SSL) configured, install a certificate and clear the client cache. For details, see [Set up HTTPS with Secure Sockets Layer \(SSL\) for Azure DevOps Server - Configuring client computers](#).

## Clear the cache on client computers

When the on-premises Azure DevOps Server configuration changes, such as when you move or split a project collection, clear the cache.

1. Sign in to your client computer for Azure DevOps Server by using the credentials of the user whose cache you want to clear.

2. Close any open instances of Visual Studio.
3. Open a browser and go to one of the following folders, depending on the operating system your computer runs on:
  - **Windows 10** *Drive:\Users<username>\AppData\Local\Microsoft\Team Foundation\6.0\Cache*
  - **Windows 8** *Drive:\Users<username>\AppData\Local\Microsoft\Team Foundation\4.0\Cache*
  - **Windows 7 or Windows Vista** *Drive:\Users<username>\AppData\Local\Microsoft\Team Foundation\2.0\Cache*
4. Delete the contents of the Cache directory, including all subfolders.

# TF31002: Unable to connect

3/6/2021 • 5 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

You might receive this error when you try to connect to Azure DevOps Services or an on-premises Azure DevOps Server from Visual Studio.

## You receive this error when you try to connect to Azure DevOps Services

PROBLEM	RESOLUTION
You don't have an active account or license.	Check with your administrator that you're a member of the account and have an active, valid license. See <a href="#">Assign licenses to users</a> for details.
Your Azure DevOps Services organization is connected to the Azure Active Directory.	When your Azure DevOps Services organization is connected to a directory that is associated with a Microsoft 365 or Microsoft Azure subscription, only members in the directory can access the account.  Check with your directory administrator to have them <a href="#">create an organizational account for you or add your account to the directory as external member</a> .
You can't switch between different organizational accounts.	If you work with several organizations that connect to different directories, such as accounts created from the Microsoft Azure Portal, the sign-out function might not work as expected. For example, you can't switch between different organizational accounts to connect to multiple accounts that are linked to directory tenants.  When this problem occurs, you see a flashing blank sign in dialog box several times. Then, you receive either TF31002 or TF31003 error after you connect to or add a new connection in "Connect to Team Foundation Server" dialog box.  To resolve this problem, apply the most recent <a href="#">Visual Studio update</a> .  To learn more, see <a href="#">You can't switch between different organizational accounts in Visual Studio Online</a> .
You want to sign in to Azure DevOps Services from Visual Studio using different credentials.	See <a href="#">Connect to projects, Sign in with different credentials</a> .

## When you try to connect to an on-premises Azure DevOps Server from your client computer

If you determine that you're receiving this error from one computer but not others, or others aren't receiving

this error, then check the problem resolutions that are outlined below.

PROBLEM	RESOLUTION
Your password has expired.	Verify that you entered your user ID and password correctly, and that your password hasn't expired.
You've entered an incorrect server URL.	Verify that you've entered the server URL correctly including the server name, port number, and protocol (http/https). See <a href="#">Connect to projects</a> to learn more.
The TFS configuration has changed.	If the configuration for the on-premises Azure DevOps Server has changed, you must create a new connection. You might also need to <a href="#">clear the client cache</a> .
You work remotely and need to connect to a TFS Proxy server to check in files to Team Foundation version control.	<a href="#">Configure Visual Studio to connect to TFS Proxy</a> .
You're connecting to a later version of TFS than your Visual Studio client version.	Your version of Visual Studio or Team Explorer might be incompatible with Team Foundation Server. You might need to install one or more GDR packs. See <a href="#">Requirements and compatibility</a> for details.
Your firewall is blocking TFS services.	See <a href="#">Allow a program to communicate through Windows Firewall</a> .
Visual Studio stops responding when you run a query in Visual Studio.	Your computer might be configured to bypass the proxy server. Verify the configuration of the BypassProxyOnLocal setting on your computer. For more information, see <a href="#">BypassProxyOnLocal Configuration</a> .

## Several users can't connect to an on-premises Azure DevOps Server

If the problem occurs on more than one computer, contact your administrator to confirm whether the server is operational and available on the network.

As an administrator, check the event logs for the application-tier server to try to pinpoint the problem. Also, you can use the following table to determine whether the server is misconfigured. In the table, problems that are more likely to occur appear first. Try the resolutions in the order in which they appear, which increases the chance that you can solve the problem quickly.

PROBLEM	RESOLUTION
The <i>TFSService</i> account password has expired or is incorrect.	Many services for Team Foundation Server will stop running when the service account for Team Foundation has expired. For more information, see <a href="#">Change the service account or password for Team Foundation Server</a> .
The application-tier server for Team Foundation is unavailable.	Verify whether each required service is running. If a required service isn't running, you must restart it. If necessary, set it to start automatically. For more information, see <a href="#">Stop and start services, application pools, and websites</a> .
The network is unavailable.	Verify whether your network is operational.

PROBLEM	RESOLUTION
A website identity for Team Foundation is configured incorrectly.	Verify or correct the server binding assignments that are made to websites for Team Foundation.
Access to a website for Team Foundation has been restricted.	Verify or correct restrictions that are made to those websites that are based on IP addresses and domain names.
The firewall or ports are configured incorrectly.	Verify or correct port binding assignments for websites and port assignments for the firewall. First, you should open the administration console for Team Foundation, display the Application Tier page, and review the URL assignments. If necessary, you can click <b>Change URL</b> to modify the URL of a website. Next, you should verify the port assignments for Internet Information Services (IIS) and the ports that are allowed through the firewall. For more information, see <a href="#">Review Server Status and Settings</a> and <a href="#">Verify or Correct Port Assignments</a> .
Trust relationships between domains aren't configured correctly.	If a group of users can't access Team Foundation Server, you might have trust issues between domains.
When users connect to different versions of TFS from Visual Studio, for example, they connect to TFS 2012 and then TFS 2008, they can get the TF31002 error.	<p>This error can occur because the GUIDs for the TFS 2012 collection are the same as TFS 2008. The local client cache gets confused because it tries to maintain the same GUID-based local cache for both the 2008 server and the new Project Collection in 2012.</p> <p>To fix, run the <b>TFSCconfig ChangeServerID</b> command. See <a href="#">TFSCconfig ChangeServerID command</a>.</p>

# Troubleshoot access and permission issues

4/12/2021 • 10 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Azure DevOps security and permission structure is extensive, so you may find yourself needing to investigate why you or a project member doesn't have the access to a project, service, or feature that they expect to have. Find step-by-step guidance to understand and address problems a project member may be having in connecting to a project or accessing an Azure DevOps service or feature.

Before using this guide, we recommend that you're familiar with the following content:

- [Get started with permissions, access, and security groups](#)
- [Default permissions and access quick reference](#).
- [Quick reference index to Azure DevOps security](#)

## TIP

When you're creating an Azure DevOps security group, label it in a way that is easy to discern if it's created to limit access.

Permissions get set at one of the following levels:

- object level
- project level
- organization or project collection level
- security role
- team administrator role

## Common access and permission issues

See the following most common reasons a project member can't access a project, service, or feature:

ISSUE	TROUBLESHOOTING ACTION
Their access level doesn't support access to the service or feature.	To discover if this is the cause, you want to <a href="#">determine the user's access level and subscription status</a> .
Their membership within a security group doesn't support access to a feature or they have been explicitly denied permission to a feature.	To discover if this is the cause, <a href="#">trace a permission</a> .
The user has been recently granted permission, however a refresh is required for their client to recognize the changes.	Have the user <a href="#">refresh or re-evaluate their permissions</a> .
The user's trying to exercise a feature granted only to a team administrator for a specific team, however they haven't been granted that role.	To add them to the role, see <a href="#">Add, remove team administrator</a> .

ISSUE	TROUBLESHOOTING ACTION
The user hasn't enabled a preview feature.	Have the user open the Preview features and determine the on/off status for the specific feature. For more information, see <a href="#">Manage preview features</a> .
Project member has been added to a limited scope security group, such as the Project-Spaced Users group.	To discover if this is a cause, <a href="#">look up the user's security group memberships</a> .

## Less common access and permission issues

Less common reasons for limited access are when one of the following events has occurred:

ISSUE	TROUBLESHOOTING ACTION
A project administrator disabled a service. In this case, no one has access to the disabled service.	To determine whether a service is disabled, see <a href="#">Turn an Azure DevOps service on or off</a> .
A Project Collection Administrator disabled a preview feature, which disables it for all project members in the organization.	See <a href="#">Manage preview features</a> .
Group rules governing the user's access level or project membership are restricting access.	See <a href="#">Determine a user's access level and subscription status</a> .
Custom rules have been defined to a work item type's workflow.	see <a href="#">Rules applied to a work item type that restrict select operation</a> .

## Determine a user's access level and subscription status

You can assign users or groups of users to one of the following access levels:

- Stakeholder
- Basic
- Basic + Test Plans
- Visual Studio subscription

For more information about access level restriction in Azure DevOps, see [Supported access levels](#).

To use Azure DevOps features, users must be added to a security group with the appropriate permissions. Users also need access to the web portal. Limitations to select features get based on the access level and security group to which a user is assigned.

Users can lose access for the following reasons:

REASON FOR LOSS OF ACCESS	TROUBLESHOOTING ACTION
The user's Visual Studio subscription has expired.	Meanwhile, this user can <a href="#">work as a Stakeholder</a> , or you can give the user Basic access until the user renews their subscription. After the user signs in, Azure DevOps restores access automatically.
The Azure subscription used for billing is no longer active.	All purchases made with this subscription are affected, including Visual Studio subscriptions. To fix this issue, visit the <a href="#">Azure account portal</a> .

REASON FOR LOSS OF ACCESS	TROUBLESHOOTING ACTION
The Azure subscription used for billing was removed from your organization.	Learn more about <a href="#">linking your organization</a>

Otherwise, on the first day of the calendar month, users who haven't signed in to your organization for the longest time lose access first. If your organization has users who don't need access anymore, [remove them from your organization](#).

For more information about permissions, see [Permissions and groups](#) and the [Permissions lookup guide](#).

## Trace a permission

Use permission tracing to determine why a user's permissions aren't allowing them access to a specific feature or function. Learn how a user or an administrator can investigate the inheritance of permissions. To trace a permission from the web portal, open the permission or security page for the corresponding level. For more information, see [Change individual permissions](#).

If a user's having permissions issues and you use default security groups or custom groups for permissions, you can investigate where those permissions are coming from by using our permissions tracing. Permissions issues could be because of delayed changes. It can take up to 1 hour for Azure AD group memberships or permissions changes to propagate throughout Azure DevOps. If a user's having issues that don't resolve immediately, wait a day to see if they resolve. For more information about user and access management, see [Manage users and access in Azure DevOps](#).

If a user's having permissions issues and you use default security groups or custom groups for permissions, you can investigate where those permissions are coming from by using our permissions tracing. Permissions issues could be because the user doesn't have the necessary access level.

Users can receive their effective permissions either directly or via groups.

Complete the following steps so administrators can understand where exactly those permissions are coming from and adjust them, as needed.

1. Select **Project settings > Permissions > Users**, and then select the user.

The screenshot shows the 'Permissions' section of the Azure DevOps settings for the 'FabrikamFiber' project. The left sidebar lists various project settings like Overview, Boards, Repos, Pipelines, Artifacts, and Compliance. A red box labeled '1' highlights the 'Project settings' button at the bottom of the sidebar. The main content area shows 'Project Settings' for 'FabrikamFiber' under 'General', 'Teams', and 'Permissions'. A red box labeled '2' highlights the 'Permissions' link in the 'Teams' section. The 'Permissions' section on the right is titled 'Permissions' with a red box labeled '3' and a red notification badge showing '3'. It has tabs for 'Groups' and 'Users', with 'Users' selected. It shows a total of 6 users: Customer service Build Service (fabrikamfiber), Jamal Hartnett (fabrikamfiber4@hotmail.com), Project Collection Build Service (fabrikamfiber), Management team Build Service (fabrikamfiber), Fabrikam Test Build Service (fabrikamfiberorg), and FabrikamFiber Build Service (fabrikamfiberorg).

You should now have a user-specific view that shows what permissions they have.

2. To trace why a user does or doesn't have any of the listed permissions, select the information icon next to the permission in question.

JH Jamal Hartnett

Permissions Member of

**General**

Delete team project	Allow (inherited)	(i)
Edit project-level information	Allow (inherited)	(i) 
Manage project properties	Allow (inherited)	(i)
Rename team project	Allow (inherited)	(i)
Suppress notifications for work item updates	Allow (inherited)	(i)
Update project visibility	Allow (inherited)	(i)
View project-level information	Allow (inherited)	(i)

The permission value is being inherited through your direct or indirect membership in these groups:  
[FabrikamFiber]\Project Administrators

**Boards**

Bypass rules on work item updates	Allow (inherited)	(i)
Change process of team project.	Allow (inherited)	(i)
Create tag definition	Allow (inherited)	(i)
Delete and restore work items	Allow (inherited)	(i)
Move work items out of this project	Allow (inherited)	(i)
Permanently delete work items	Allow (inherited)	(i)

**Analytics**

Delete shared Analytics views	Allow (inherited)	(i)
Edit shared Analytics views	Allow (inherited)	(i)
View analytics	Allow (inherited)	(i)

The resulting trace lets you know how they're inheriting the listed permission. You can then adjust the user's permissions by adjusting the permissions that are provided to the groups that they're in.

1. Select **Project settings > Security**, and then enter the user name into the filter box.

The screenshot shows the Azure DevOps Settings - Security page for the Fabrikam Fiber project. The left sidebar lists various project management sections like Overview, Boards, Repos, Pipelines, Test Plans, and Artifacts. The main content area is titled 'Project Settings' and contains a list of security-related options: General, Overview, Teams, Security (which is highlighted with a red box and number 2), Notifications, Service hooks, Dashboards, Boards, Project configuration, Team configuration, GitHub connections, Pipelines, Agent pools, Retention, Release retention, Service connections, Repos, Repositories, Policies, Test, and Retention. In the top right corner, there is a 'Create group' button with a red box and number 3, and below it is a 'Filter users and groups' search bar. A red box also highlights the 'Filter users and groups' search bar.

You should now have a user-specific view that shows what permissions they have.

2. Trace why a user does or doesn't have any of the listed permissions. Hover over the permission, and then choose **Why**.

**Permissions** Member of

Bypass rules on work item updates	Not set
Change process of team project.	Not set
Create tag definition	Allow (inherited)
Create test runs	Allow (inherited)
Delete and restore work items	Allow (inherited)
Delete shared Analytics views	Allow (inherited)
Delete team project	Not set
Delete test runs	Allow (inherited)
Edit project-level information	Not set
Edit shared Analytics views	Allow (inherited)
Manage project properties	Not set
Manage test configurations	Allow (inherited)
Manage test environments	Allow (inherited)
Move work items out of this project	Not set
Permanently delete work items	Not set
Rename team project	Not set
SUPPRESS notifications for work item updates	Not set
View analytics	Allow (inherited)
View project-level information	Allow (inherited) <a href="#">Why?</a>
View test runs	Allow (inherited)

[Clear explicit permissions](#)

[Save changes](#)

[Undo changes](#)

The resulting trace lets you know how they're inheriting the listed permission. You can then adjust the user's permissions by adjusting the permissions that are provided to the groups they're in.

**TRACE INHERITANCE RESULTS**

Permission	View project-level information
Effective value	Allow (Inherited)
Identity	Christie Church

**GROUP MEMBER INHERITANCE**

The permission value is being inherited through your direct or indirect membership in these groups

[Fabrikam Fiber]\Contributors	Allow
[Fabrikam Fiber]\Fabrikam Fiber Team	Allow

**Close**

1. Go to the Security page for the project that the user is having access problems.
2. Enter their name into the box in the upper left-hand corner.

**Permissions**

Bypass rules on work item updates	Allow (inherited)
Create tag definition	Allow (inherited)
Create test runs	Allow (inherited)
Delete and restore work items	Allow (inherited)
Delete team project	Allow (inherited)

You should have a user-specific view that shows what permissions they have.

3. Trace why a user does or doesn't have any of the listed permissions. Hover over the permission, and then choose **Why**.

The screenshot shows the Azure DevOps interface for managing group permissions. On the left, there's a sidebar with 'Create group' and a list of groups including 'Christie Church'. The main area shows 'FabrikamFiber > Christie Church' with a 'Permissions' section. This section lists various permissions with their inheritance status (e.g., 'Allow (inherited)'). A red box highlights the 'Why?' link next to the 'View project-level information' permission. At the bottom, there are buttons for 'Save changes' and 'Undo changes'.

Permission	Status
Bypass rules on work item updates	Allow (inherited)
Create tag definition	Allow (inherited)
Create test runs	Allow (inherited)
Delete and restore work items	Allow (inherited)
Delete team project	Allow (inherited)
Delete test runs	Allow (inherited)
Edit project-level information	Allow (inherited)
Manage project properties	Allow (inherited)
Manage test configurations	Allow (inherited)
Manage test environments	Allow (inherited)
Move work items out of this project	Allow (inherited)
Permanently delete work items	Allow (inherited)
Rename team project	Allow (inherited)
Suppress notifications for work item updates	Allow (inherited)
Update project visibility	Allow (inherited)
View analytics	Allow (inherited)
View project-level information	Allow (inherited) <a href="#">Why?</a>
View test runs	Allow (inherited)

The resulting trace lets you know how they're inheriting the listed permission. You can then adjust the user's permissions by adjusting those permissions provided to the groups they're in.

For more information, see [Grant or restrict access to select features and functions](#) or [Change individual permissions](#).

## Refresh or reevaluate permissions

See the following scenario where refreshing or reevaluating permissions may be necessary.

### Problem

Users get added to an Azure DevOps or Azure AD group. This action grants inherited access to an organization or project. But, they don't get access immediately. Users must either wait or sign out, close their browser, and then sign back in to get their permissions refreshed.

Users get added to an Azure DevOps group. This action grants inherited access to an organization or project. But, they don't get access immediately. Users must either wait or sign out, close their browser, and then sign back in to get their permissions refreshed.

### Solution

Within **User settings**, on the **Permissions** page, you can select **Reevaluate permissions**. This function reevaluates your group memberships and permissions, and then any recent changes take effect immediately.

The screenshot shows the 'User settings' page in Azure DevOps. On the left, there's a sidebar with options like Account, Profile, Time and Locale, Permissions (which is highlighted with a red box), Preferences, Notifications, and Theme. On the right, under the 'Permissions' heading, there's a button labeled 'Re-evaluate permissions' with a red box around it. Below the button, there's a note: 'This tool will re-evaluate your group memberships and permissions; any recent changes will take effect immediately. Normally, changes to group memberships and permissions can take up to one hour to reflect in Azure DevOps.'

## Rules applied to a work item type that restrict select operations

Before you customize a process, we recommend that you review [Configure and customize Azure Boards](#), which provides guidance on how to customize Azure Boards to meet your business needs.

For more information about work item type rules that apply toward restricting operations, see:

- [Apply rules to workflow states \(Inheritance process\)](#)
- [Restrict modification of select fields based on a user group](#)
- [Restrict modification of closed work items](#)
- [Define area paths and assign to a team](#)

## Hide organization settings from users

If a user's limited to seeing only their projects, or from seeing the organization settings, the following information may explain why. To restrict users from accessing organization settings, you can enable the **Limit user visibility for projects** preview feature.

Examples of restricted users include Stakeholders, Azure Active Directory (Azure AD) guest users, or members of a security group. Once enabled, any user or group added to the Project-Spaced Users group gets restricted from accessing the Organization Settings pages, except for Overview and Projects. They're restricted to accessing only those projects to which they've been added.

Examples of restricted users include Stakeholders, or members of a security group. Once enabled, any user or group added to the Project-Spaced Users group gets restricted from accessing the Organization Settings pages, except for Overview and Projects. They're restricted to accessing only those projects to which they've been added.

For more information about hiding organization settings from users, see [About projects](#), [Project-scoped User group](#).

## View, add, and manage permissions with CLI

You can view, add, and manage permissions at a more granular level with the `az devops security permission` commands. For more information, see [Manage permissions with command line tool](#).

### Use tools to fix permission

You can use the following tools to fix a user's permission issue.

- **TFSSecurity.exe** - TFSSecurity is a command-line tool that can be used to view and update and delete permissions or groups.

Example usage:

```
tfssecurity /a+ Identity "81e4e4b5-bde0-4f2c-a7a5-4d25c2e8a81f\" Read "Project Collection Valid Users"  
ALLOW /collection:{collectionUrl}  
tfssecurity /a- Identity "3c7a0a47-27b4-4def-8d42-aab9b405fc8a\" Write n:"[Project1]\Contributors"  
DENY /collection:{collectionUrl}
```

- Use the public sproc

Example usage: Use `prc_pSetAccessControlEntry` or `prc_pRemoveAccessControlEntries` to add or remove ACEs directly from the security tables if TFS Security doesn't work for you.

For more information, see [Use TFSSecurity to manage groups and permissions for Azure DevOps](#).

## Group rules with lesser permissions

Group rule types get ranked in the following order: Subscriber > Basic + Test Plans > Basic > Stakeholder. Users always get the best access level between all the group rules, including Visual Studio (VS) subscription.

See the following examples, showing how subscriber detection factors into group rules.

### Example 1: Group rule gives me more access

If I have a VS Pro subscription and I'm in a group rule that gives me Basic + Test Plans – what happens?

Expected: I get Basic + Test Plans because what the group rule gives me is greater than my subscription. Group rule assignment always provides the greater access, rather than limiting access.

### Example 2: Group rule gives me the same access

I have a Visual Studio Test Pro subscription and I'm in a group rule that gives me Basic + Test Plans – what happens?

Expected: I get detected as a Visual Studio Test Pro subscriber, because the access is the same as the group rule. I'm already paying for the Visual Studio Test Pro, so I don't want to pay again.

## Work with GitHub

See the following troubleshooting information for when you're trying to deploy code in Azure DevOps with GitHub.

### Problem

You can't bring the rest of your team into the organization and project, despite adding them as organization and project members. They receive emails but when signing in they receive an error 401.

### Solution

You're likely signed into Azure DevOps with an incorrect identity. Complete the following steps.

1. Close all browsers, including browsers that aren't running Azure DevOps.
2. Open a private or incognito browsing session.
3. Go to the following URL: <https://aka.ms/vssignout>.

A message displays that says, "Sign out in progress." After you sign out, you're redirected to [dev.azure.microsoft.com](https://dev.azure.microsoft.com).

4. Sign in to [Azure DevOps](#) again. Select your other identity.

## Other areas where permissions might be applied

- [Area path permissions](#)

- [Work item tags](#)
- [Moved work items out of a project](#)
- [Deleted work items](#)
- [Azure Boards Team Administrator permissions and access](#)
- [Custom rules](#)
- [Custom fields](#)
- [Custom backlogs and boards](#)
- [Custom controls](#)

## Related articles

- [Manage permissions with the command line tool](#)
- [Change individual or group permissions](#)
- [Security and permission management tools](#)
- [Add users to an organization \(Azure DevOps Services\)](#)
- [Add users to a team or a project](#)
- [Add users to an administrator role](#)

# Allowed address lists and network connections

4/2/2021 • 5 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2015

If your organization is secured with a firewall or proxy server, you need to add certain IP addresses and domain URLs to the **allowlist**. Adding them to the allowlist helps to ensure that you have the best experiences with Azure DevOps.

For the best experience with Visual Studio and Azure Services, you open select ports and protocols. For more information, see [Install and use Visual Studio behind a firewall or proxy server](#), [Use Visual Studio and Azure Services](#).

## Domain URLs to allow

Network connection issues could occur because of your security appliances, which may be blocking connections - Visual Studio uses TLS 1.2. When you're using NuGet or connecting from Visual Studio 2015 and later, update the security appliances to support TLS 1.2 for the following connections.

### Azure DevOps domains to allow

To ensure your organization works with any existing firewall or IP restrictions, ensure that `dev.azure.com` and `*.dev.azure.com` are open.

### URLs to support sign in and licensing connections

- `management.core.windows.net`
- `login.microsoftonline.com`
- `login.live.com`
- `go.microsoft.com`
- `graph.microsoft.com`
- `app.vssps.dev.azure.com`
- `app.vssps.visualstudio.com`
- `aadcdn.msauth.net`
- `aadcdn.msftauth.net`

### Additional URLs for signing into Azure DevOps and Azure

- `amcdn.msftauth.net`
- `windows.net`
- `microsoftonline.com`
- `visualstudio.com`
- `microsoft.com`
- `live.com`
- `dev.azure.com`
- `azure.microsoft.com`
- `management.azure.com`
- `azurecomcdn.azureedge.net`
- `amp.azure.net`

- `aexprodea1.vsaex.visualstudio.com`
- `management.core.windows.net`
- `aex.dev.azure.com`
- `app.vssps.dev.azure.com`
- `app.vssps.visualstudio.com`
- `vstsagentpackage.azureedge.net`
- `cdn.vsassets.io` (hosts Azure DevOps Content Delivery Networks (CDNs) content)
- `gallerycdn.vsassets.io` (hosts Azure DevOps extensions)
- `static2.sharepointonline.com` (hosts some resources that Azure DevOps uses in "office fabric" UI kit for fonts, and so on)
- `*.vstmrblob.vsassets.io` (hosts Azure DevOps TCM log data)
- `vsrm.dev.azure.com` (package feed)

## Additional domains

Azure DevOps uses CDNs to serve static content. Ensure the following CDNs are allowed.

- `*.vsassets.io`
- `*.gallerycdn.vsassets.io` (Marketplace)

Users in China should also add the following domains to an allowlist:

- `*.vsassetcdn.azure.cn`
- `*.gallerycdn.azure.cn` (Marketplace)

We recommend you open port 443 to all traffic on these IP addresses and domains. We also recommend you open port 22 to a smaller subset of targeted IP addresses.

### Azure Artifacts

- `*.blob.core.windows.net`
- `*.visualstudio.com`
- all IP addresses in the "name": "Storage.{your region}" section of this file (updated weekly): [Azure IP ranges and Service Tags - Public Cloud](#)

## NuGet connections

- `azurewebsites.net`
- `nuget.org`

### NOTE

Privately owned NuGet server URLs may not be included in the previous list. You can check the NuGet servers you're using by opening up `%APPData%\Nuget\NuGet.Config`.

## IP addresses and range restrictions

### Outbound connections

*Outbound connections* are those that originate from inside your organization and that target Azure DevOps or other dependent sites. Examples of such connections include:

- Browsers connecting to Azure DevOps website as users go to and use features of Azure DevOps

- Azure Pipelines agents installed on your organization's network connecting to Azure DevOps to poll for pending jobs
- CI events being sent from a source code repository hosted within your organization's network to Azure DevOps

Ensure the following IP addresses are allowed for outbound connection, so your organization works with any existing firewall or IP restrictions. The endpoint data in the following chart lists requirements for connectivity from a machine in your organization to Azure DevOps Services.

IP V4 RANGES	IP V6 RANGES
13.107.6.0/24	2620:1ec:4::/48
13.107.9.0/24	2620:1ec:a92::/48
13.107.42.0/24	2620:1ec:21::/48
13.107.43.0/24	2620:1ec:22::/48

If you're currently allowing the `13.107.6.183` and `13.107.9.183` IP addresses, leave them in place, as you don't need to remove them.

**NOTE**

Azure Service Tags are not supported for *outbound* connection.

## Inbound connections

*Inbound connections* are those that originate from Azure DevOps and that target resources within your organization's network. Examples of such connections include:

- Azure DevOps Services connecting to endpoints for [Service Hooks](#)
- Azure DevOps Services connecting to customer-controlled SQL Azure VMs for [Data Import](#)
- Azure Pipelines connecting to on-premises source code repositories such as [GitHub Enterprise](#) or [BitBucket Server](#)
- Azure DevOps Services [Audit Streaming](#) connecting to on-premises or cloud-based Splunk

Ensure the following IP addresses are allowed for inbound connection, so your organization works with any existing firewall or IP restrictions. The endpoint data in the following chart lists requirements for connectivity from Azure DevOps Services to your on-premises or other cloud services.

REGION	IP V4 RANGES
Australia East	20.37.194.0/24
Australia South East	20.42.226.0/24
Brazil South	191.235.226.0/24
Central Canada	52.228.82.0/24
Asia Pacific (Hong Kong)	20.189.107.0/24

REGION	IP V4 RANGES
South India	20.41.194.0/24
Central United States	20.37.158.0/23
West Central United States	52.150.138.0/24
East United States	20.42.5.0/24
East 2 United States	20.41.6.0/23
North United States	40.80.187.0/24
South United States	40.119.10.0/24
West United States	40.82.252.0/24
West 2 United States	20.42.134.0/23
Western Europe	40.74.28.0/23
United Kingdom South	51.104.26.0/24

Azure Service Tags are supported for *inbound* connection. Instead of allowing the previously listed IP ranges, you may use the [AzureDevOps](#) service tag for Azure Firewall and Network Security Group (NSG) or on-premises firewall via a JSON file download.

#### NOTE

The Service Tag or previously mentioned inbound IP addresses do not apply to Microsoft Hosted Agents. Customers are still required to allow the [entire geography for the Microsoft Hosted Agents](#). If allowing the entire geography is a concern, we recommend using the [Azure Virtual Machine Scale Set Agents](#). The Scale Set Agents are a form of self-hosted agents that can be auto-scaled to meet your demands.

#### Azure DevOps ExpressRoute connections

If your organization uses ExpressRoute, ensure the following addresses are allowed for both outbound and inbound connections.

IP V4 RANGES	IP V6 RANGES
13.107.6.175/32	2620:1ec:a92::175/128
13.107.6.176/32	2620:1ec:a92::176/128
13.107.6.183/32	2620:1ec:a92::183/128
13.107.9.175/32	2620:1ec:4::175/128
13.107.9.176/32	2620:1ec:4::176/128

IP V4 RANGES	IP V6 RANGES
13.107.9.183/32	2620:1ec:4::183/128
13.107.42.18/32	2620:1ec:21::18/128
13.107.42.19/32	2620:1ec:21::19/128
13.107.42.20/32	2620:1ec:21::20/128
13.107.43.18/32	2620:1ec:22::18/128
13.107.43.19/32	2620:1ec:22::19/128
13.107.43.20/32	2620:1ec:22::20/128

For more information about Azure DevOps and ExpressRoute, see [ExpressRoute for Azure DevOps](#).

## Other IP addresses

- 40.82.190.38
- 52.108.0.0/14
- 52.237.19.6
- 52.238.106.116/32
- 52.244.37.168/32
- 52.244.203.72/32
- 52.244.207.172/32
- 52.244.223.198/32
- 52.247.150.191/32

For more information, see [Worldwide endpoints](#) and [Adding IP address rules](#).

## SSH connections

If you need to connect to Git repositories on Azure DevOps with SSH, you need to allow requests to port 22 for the following IP addresses:

- ssh.dev.azure.com
- vs-ssh.visualstudio.com
- all IP addresses in the "name": "AzureDevOps" section of [this downloadable file](#) (updated weekly) named: Azure IP ranges and Service Tags - Public Cloud

## Azure Pipelines Agents

If you use Microsoft-hosted agent to run your jobs and you need the information about what IP addresses are used, see [Microsoft-hosted agents Agent IP ranges](#).

If you're running a firewall and your code is in Azure Repos, see [Self-hosted Windows agents FAQs](#). This article has information about which URLs and IP addresses your private agent needs to communicate with.

For more information about hosted Windows and Linux agents, see [Microsoft-hosted Agent IP ranges](#).

Currently, we don't publish hosted Mac IP address ranges.

## Azure DevOps import service

During the import process, we highly recommend that you restrict access to your VM to only IPs from Azure DevOps. To restrict access, allow only connections from the set of Azure DevOps IPs, which were involved in the collection database import process. For information about identifying the correct IPs, see [\(Optional\) Restrict access to Azure DevOps Services IPs only](#).

## Related articles

- [Microsoft-hosted agents Agent IP ranges](#)
- [Self-hosted Windows agents FAQs](#)
- [Install and use Visual Studio behind a firewall or proxy server](#)

# Configure a network adapter to automatically adjust speed

11/2/2020 • 2 minutes to read • [Edit Online](#)

## TFS 2013

When a client computer is not configured to automatically adjust the link speed of its network adapter, some functions might take a long time to finish. Such functions include creating projects, saving work items, or merging code changes. In some cases, these functions might never finish, and an error message might appear that contains the phrase "The underlying connection was closed."

The speed with which a function finishes depends, in part, on the speed of the computer network. The configuration of network switches and your computers' network adapters can affect the network throughput. For example, "autosense mode" or "auto-negotiation" might be turned on, and information might be transmitted in either full-duplex mode or half-duplex mode.

To minimize the time required for a function to finish, you should confirm that these settings are set appropriately for maximum throughput. For more information about how full-duplex mode differs from half-duplex mode, contact your network administrator.

## Required Permissions

To perform these procedures, you must be a member of the **Administrators** security group on your local computer.

### To configure a computer to automatically adjust the link speed of its network adapter in Windows Server 2008

1. On the Start menu, click Control Panel.
2. Click Network and Internet.
3. Click Network and Sharing Center, and then click Manage network connections.

The Network Connections folder opens.

4. Right-click the relevant network connection (by default, Local Area Connection), and then click Properties.

The Local Area Connection Properties dialog box opens.

5. Click Configure.

The properties dialog box for the adapter opens.

6. Click the Advanced tab.
7. In the Property list, click the property that corresponds to the connection type, such as **Connection Type**, **Duplex Mode**, **Media**, **Media Type**, or **Link Speed & Duplex**, depending on the adapter's attributes.
8. In the Value list, click **Autosense**.
9. Click OK.

# Get support and provide feedback

3/6/2021 • 3 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Share your feedback and ideas with us, or join our communities. We're always working to improve Azure DevOps, and we want you to be part of the process!

Do you need to do any of the following?

- **Get advice** Visit StackOverflow for [Azure DevOps Services](#) or [Azure DevOps Server](#).
- **Report a bug** Submit it through our Developer Community for [Azure DevOps Services](#) or [Azure DevOps Server](#).
- **Suggest a feature or a fix** Submit your idea or issue through our Developer Community for [Azure DevOps Services](#) or [Azure DevOps Server](#).
- **Find out what's new in Azure DevOps** Check out the [current Azure DevOps Release Notes](#). These notes are updated every three weeks.

## Get live help

We offer a [live chat](#) (English only) support option. Choose from **Technical Support**, **Sales Support**, **Visual Studio (For your Company)**, and **Account, Subscription, and Billing Support**. Select your country from the dropdown menu, and then select **Live Chat (English)**.

## Documentation feedback

All docs on docs.microsoft.com have a ratings tool in the lower right-hand corner of the page. It asks "Is this content helpful?" Answer **Yes** or **No** depending on your experience.

Add more detailed feedback by selecting the **Tell us more** link after selecting **Yes** or **No**. Check an appropriate box and enter what we can do to improve the content for you! Although we can't reply back, we collect and review this feedback regularly, and use your sentiments in our content planning.

## Tips for effective feedback

If you just want to vent about the product or the docs, that's okay. It helps us a lot to know when you're happy or unhappy with an experience. For the most impact, though, provide details so we can better understand what we're doing right or wrong.

- Provide a little context. What problem were you trying to solve? At what point did it go wrong?
- What's your role? We don't need personal or professional details. Are you a dev? A manager? A business owner? When we understand our audience, we can come up with better solutions for you and other customers doing similar work.
- What version of the product were you using? What other products were you using with it?

The best feedback we get is clear and precise. For example:

- Product feedback: "I'm a project manager for a small start-up. I'm using Azure DevOps. I'm trying to create work item templates through the UI, but my changes don't seem to persist. It's not clear what I'm doing wrong."

- Doc feedback: "I'm a dev in a large organization that works on Java apps. I tried to use Maven with your build system in Azure DevOps Server 2017 Update 1 (15.112.26307.0), but I couldn't get the configuration shown in the docs to work."

The more details, the better!

## What platform/version am I using?

You can tell what platform you use from the URL you use to connect to Azure DevOps Services or Azure DevOps Server.

### Azure DevOps Services

An Azure DevOps URL consists of an organization name and dev.azure.com, for example:

`https://dev.azure.com/{yourorganization}`.

To learn the version number, enter the following address in a web browser:

`https://dev.azure.com/{yourorganization}/_home/About`

A page similar to the following example opens showing the version number.



### About Azure DevOps Service

Version Dev17.M149.1 (build: AzureDevOps\_M149\_20190326.1)

© Microsoft Corporation. All rights reserved.

### Azure DevOps Server

An on-premises URL consists of a server name, port number, and collection name, for example:

`https://ServerName:8080/tfs/CollectionName`

To learn the version number, enter the following address in a web browser:

`https://ServerName:8080/tfs/_home/About`

A page similar to the following example opens showing the version number.



About Microsoft Visual Studio Team Foundation Server

### Microsoft Visual Studio Team Foundation Server

Version 14.102.25423.0

© Microsoft Corporation. All rights reserved.

ON-PREMISES RELEASE	UPDATE	VERSION NUMBER
Azure DevOps Server 2019	2019.1	17.153.29522.3
	RTW	17.143.28511.3

ON-PREMISES RELEASE	UPDATE	VERSION NUMBER
<b>Azure DevOps Server 2018</b>	2018.3	16.131.28106.2
	2018.2	16.131.27701.1
	2018.1	16.122.27409.2
	RTW	16.122.27102.1
	RC2	16.122.26918.3
	RC1	16.121.26818.0
<b>Azure DevOps Server 2017</b>	Update 3	15.117.27024.0
	Update 3 RC	15.117.26912.0
	Update 2	15.117.26714.0
	Update 1	15.112.26307.0
	RTW	15.105.25910.0
	RC1	15.103.25603.0
<b>Azure DevOps Server 2015</b>	Update 3	14.102.25423.0
	Update 2.1	14.95.25229.0
	Update 2	14.95.25122.0
	Update 2 RC 2	14.95.25029.0
	Update 2 RC 1	14.95.25005.0
	Update 1	14.0.24712.0
	Update 1 RC 2	14.0.24626.0
	Update 1 RC 1	14.0.24606.0
	RTM	14.0.23128.0
	RC2	14.0.23102.0
	RC	14.0.22824.0
	CTP	14.0.22604.0
<b>Azure DevOps Server 2013</b>	Update 5	12.0.40629.0

ON-PREMISES RELEASE	UPDATE	VERSION NUMBER
	Update 4	12.0.31101.0
	Update 4 RC	12.0.31010.0
	Update 3	12.0.30723.0
	Update 3 RC	12.0.30626.0
	Update 2	12.0.30324.0
	RTM	12.0.21005.1
	RC	12.0.20827.3
Azure DevOps Server 2012	Update 4	11.0.61030.0
	Update 3	11.0.60610.1
	Update 2	11.0.60315.1
	CU 1	11.0.60123.100
	Update 1	11.0.51106.1
	RTM	11.0.50727.1
Azure DevOps Server 2010	CU 2	10.0.40219.371
	SP1	10.0.40219.1
	RTM	10.0.30319.1
Azure DevOps Server 2008	SP1	9.0.30729.1
	RTM	9.0.21022.8
Azure DevOps Server 2005	SP1	8.0.50727.762
	RTM	8.0.50727.147

## Related articles

- [Azure DevOps features timeline](#)
- [Report a problem with Visual Studio](#)

# Navigate in Visual Studio Team Explorer

4/17/2021 • 7 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

[Visual Studio 2019](#) | [Visual Studio 2017](#) | [Visual Studio 2015](#)

You use Team Explorer to coordinate your code efforts with other team members to develop a software project. In addition, you can manage work and that is assigned to you, your team, or your projects. Team Explorer is a plug-in that installs with Visual Studio and Team Explorer Everywhere is a plug-in that installs with Eclipse. Developers can effectively collaborate using Team Explorer connected to projects hosted on Azure DevOps Services or an on-premises Azure DevOps Server (previously named Team Foundation Server (TFS)).

## TIP

You can install the latest version of Visual Studio clients from the [Visual Studio downloads page](#).

Additional options for connecting to Azure DevOps Services or TFS include:

- [Team Explorer Everywhere](#)
- [Azure DevOps Plugin for Android Studio](#)
- [Azure DevOps Plugin for IntelliJ](#)
- [Visual Studio Code](#)

For information about compatibility among client and server versions, see [Requirements and compatibility](#).

If you don't need Visual Studio, but want to connect to a project in Azure DevOps, you can install the free [Visual Studio Community](#).

## Prerequisites

- You must have a project in Azure DevOps. If you need to add a project, see [Create a project](#).
- You must be a member of the project you connect to. To get added, see [Add users to a project or team](#).

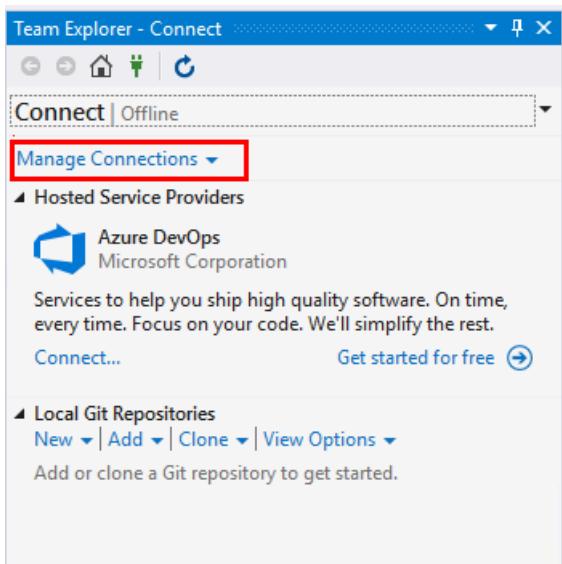
## Connect to a project or repository

Team Explorer connects Visual Studio to projects in Azure DevOps. You can manage source code, work items, and builds. The operations available to you depend on which source control option—Git or Team Foundation version control (TFVC)—was selected to manage source code when the project was created.

## TIP

If you open Visual Studio and the Team Explorer pane doesn't appear, choose the [View>Team Explorer](#) menu option from the tool bar.

From the [Connect](#) page, you can select the projects you want to connect to and quickly switch connection to a different project and or repository. For details, see [Connect to a project](#).



The Git and TFVC repos support different pages and functions. For a comparison of the two version control systems, see [Choosing the right version control for your project](#).

## Git version control and repository

The following images show the pages available when you connect to a Git repository from Team Explorer.

### NOTE

Visual Studio 2019 now includes a new Git tool that provides an improved experience when connecting to a Git repository. When you enable this tool, the Team Explorer tool is effectively disabled when connected to a Git repository. You can acquire the new tool by downloading [Visual Studio 2019 version 16.6](#). To enable and use the new tool, see [Git experience in Visual Studio \(Preview\)](#).

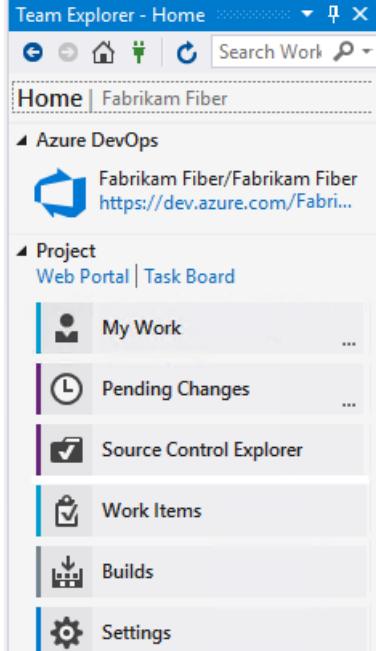
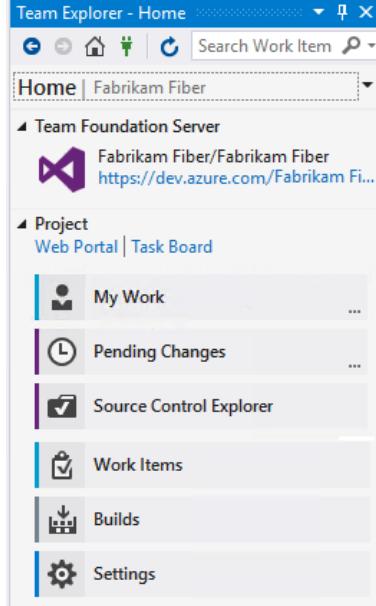
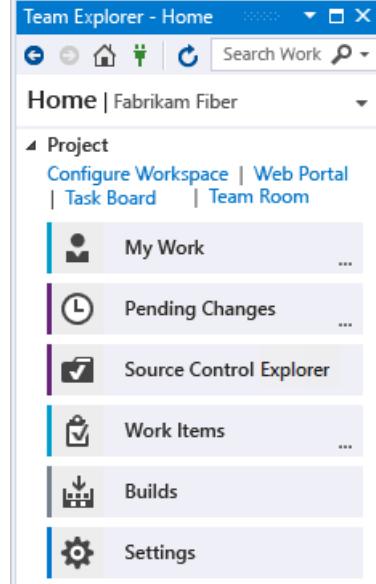
VISUAL STUDIO 2019	VISUAL STUDIO 2017	VISUAL STUDIO 2015
The screenshot shows the 'Team Explorer - Home' interface for Visual Studio 2019. It features a top navigation bar with icons for back, forward, home, and refresh, and a search bar labeled 'Search Work'. Below the bar, the 'Home' tab is selected, showing 'Fabrikam Fiber'. Under the 'Project' section, 'Azure DevOps' is listed with a link to 'Fabrikam Fiber/Fabrikam Fiber https://dev.azure.com/Fabri...'. The main pane contains large buttons for 'Changes', 'Branches', 'Pull Requests', 'Sync', 'Tags', 'Work Items', 'Builds', and 'Settings'.	The screenshot shows the 'Team Explorer - Home' interface for Visual Studio 2017. It has a similar layout to 2019, with a top bar and a 'Home' tab selected for 'Fabrikam Git'. The 'Project' section lists 'Visual Studio Team Services' with a link to 'Fabrikam Git/Fabrikam... https://fabrikamprime....'. The main pane includes buttons for 'Changes', 'Branches', 'Pull Requests', 'Sync', 'Tags', 'Work Items', 'Builds', and 'Settings'.	The screenshot shows the 'Team Explorer - Home' interface for Visual Studio 2015. It displays a 'Project' section with 'Clone Repository', 'Web Portal', and 'Task Board' options. The main pane is divided into several sections: 'Changes', 'Branches', 'Pull Requests', 'Sync', 'Tags', 'Work Items', 'Builds', and 'Settings'.

To learn more about each page, see the following articles.

HOME & BUILDS	GIT VERSION CONTROL	WORK ITEMS
<p><b>Home</b></p> <ul style="list-style-type: none"><li>• <a href="#">Web portal</a></li><li>• <a href="#">Task Board</a></li><li>• <a href="#">Team Room</a></li></ul> <p><b>Builds</b></p> <ul style="list-style-type: none"><li>• Create build pipelines</li><li>• View and manage builds</li><li>• Manage the build queue</li><li>• Install Continuous Delivery (CD) Tools for Visual Studio</li><li>• Configure and execute Continuous Delivery (CD) for your app</li></ul>	<ul style="list-style-type: none"><li>• <a href="#">Create a new repo</a></li><li>• <a href="#">Clone an existing repo</a></li><li>• <b>Changes:</b> <a href="#">Save work with commits</a></li><li>• <b>Branches:</b> <a href="#">Create work in branches</a></li><li>• <b>Pull Requests:</b> <a href="#">Review code with pull requests</a></li><li>• <b>Sync:</b> <a href="#">Update code with fetch and pull</a></li><li>• <b>Tags:</b> <a href="#">Work with Git tags</a></li><li>• <a href="#">Git preferences</a></li><li>• <a href="#">Git command reference</a></li></ul>	<p><b>Default experience</b> (Visual Studio 2019 only)</p> <ul style="list-style-type: none"><li>• <a href="#">View and add work items</a></li><li>• <a href="#">Set the Work Items experience in Visual Studio</a></li></ul> <p><b>Legacy experience</b> (All versions of Visual Studio)</p> <ul style="list-style-type: none"><li>• <a href="#">Add work items</a></li><li>• <a href="#">Query editor</a></li><li>• <a href="#">Query folders</a></li><li>• <a href="#">Query permissions</a></li><li>• <a href="#">Open query in Excel</a></li><li>• <a href="#">Open query in Project</a></li><li>• <a href="#">Email query results using Outlook</a></li><li>• <a href="#">Create reports from query in Excel (TFS only)</a></li></ul>

## Team Foundation version control

The following images show the pages available when you connect to a TFVC repository from Team Explorer.

VISUAL STUDIO 2019	VISUAL STUDIO 2017	VISUAL STUDIO 2015
		

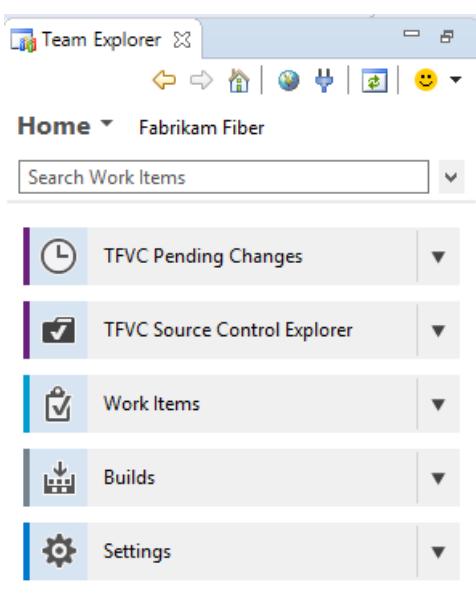
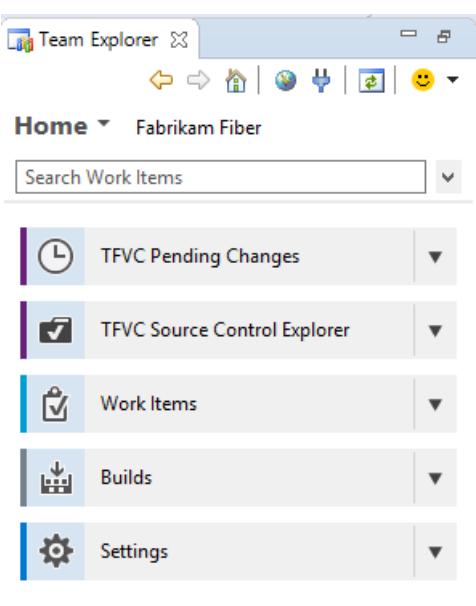
To learn more about each page, see the following articles.

HOME & BUILDS	TFVC	WORK ITEMS
---------------	------	------------

<p><b>Home</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Web portal</a></li> <li>• <a href="#">Task Board</a></li> <li>• <a href="#">Team Room</a></li> </ul> <p><b>Builds</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Create build pipelines</a></li> <li>• <a href="#">View and manage builds</a></li> <li>• <a href="#">Manage the build queue</a></li> <li>• <a href="#">Install Continuous Delivery (CD) Tools for Visual Studio</a></li> <li>• <a href="#">Configure and execute Continuous Delivery (CD) for your app</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">Configure workspace</a></li> <li>• <a href="#">My Work: Suspend/resume work   Code review</a></li> <li>• <a href="#">Pending Changes: Manage pending changes   Find shelvesets   Resolve conflicts</a></li> <li>• <a href="#">Source Control Explorer: Add/view files and folders</a></li> <li>• <a href="#">Add Check-In Policies</a></li> <li>• <a href="#">Version control commands</a></li> </ul>	<p><b>Default experience (Visual Studio 2019 only)</b></p> <ul style="list-style-type: none"> <li>• <a href="#">View and add work items</a></li> <li>• <a href="#">Set the Work Items experience in Visual Studio</a></li> </ul> <p><b>Legacy experience (All versions of Visual Studio)</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Add work items</a></li> <li>• <a href="#">Query editor</a></li> <li>• <a href="#">Query folders</a></li> <li>• <a href="#">Query permissions</a></li> <li>• <a href="#">Open query in Excel</a></li> <li>• <a href="#">Open query in Project</a></li> <li>• <a href="#">Email query results using Outlook</a></li> <li>• <a href="#">Create reports from query in Excel (TFS only)</a></li> </ul>
---	--	---

## Team Explorer plug-in for Eclipse

If you work in Eclipse or on a non-Windows platform, you can [install the Team Explorer plug-in for Eclipse](#). Once installed, you can share your Eclipse projects by adding them to Azure DevOps Services or TFS using [Git](#) or [TFVC](#).

HOME PAGE WITH GIT (ECLIPSE)	HOME PAGE WITH TFVC (ECLIPSE)
 <p>The screenshot shows the Eclipse Team Explorer interface with the following elements:</p> <ul style="list-style-type: none"> <li><b>Toolbar:</b> Includes icons for back, forward, home, search, and other navigation.</li> <li><b>Header:</b> Displays "Team Explorer" and the project name "Fabrikam Fiber".</li> <li><b>Search Bar:</b> Labeled "Search Work Items".</li> <li><b>Work Items:</b> A list of items including "TFVC Pending Changes", "TFVC Source Control Explorer", "Work Items", "Builds", and "Settings".</li> </ul>	 <p>The screenshot shows the Eclipse Team Explorer interface with the following elements:</p> <ul style="list-style-type: none"> <li><b>Toolbar:</b> Includes icons for back, forward, home, search, and other navigation.</li> <li><b>Header:</b> Displays "Team Explorer" and the project name "Fabrikam Fiber".</li> <li><b>Search Bar:</b> Labeled "Search Work Items".</li> <li><b>Work Items:</b> A list of items including "TFVC Pending Changes", "TFVC Source Control Explorer", "Work Items", "Builds", and "Settings".</li> </ul>

To learn more about each page, see the following articles.

HOME & BUILDS	VERSION CONTROL	WORK ITEMS
---------------	-----------------	------------

<p><b>Home</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Web portal</a></li> </ul> <p><b>Builds</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Create build pipelines</a></li> <li>• <a href="#">View and manage builds</a></li> <li>• <a href="#">Manage the build queue</a></li> <li>• <a href="#">Install Continuous Delivery (CD) Tools for Visual Studio</a></li> <li>• <a href="#">Configure and execute Continuous Delivery (CD) for your app</a></li> </ul>	<p><b>Git repo</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Share your code</a></li> <li>• <a href="#">Git preferences</a></li> <li>• <a href="#">Git command reference</a></li> </ul> <p><b>TFVC repo</b></p> <ul style="list-style-type: none"> <li>• <a href="#">Share your code</a></li> <li>• <a href="#">Pending changes</a></li> <li>• <a href="#">Source Control Explorer</a></li> <li>• <a href="#">Add Check-In Policies</a></li> <li>• <a href="#">Version control commands</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">Add work items</a></li> <li>• <a href="#">Query editor</a></li> <li>• <a href="#">Query folders</a></li> <li>• <a href="#">Query permissions</a></li> </ul>
--	---	--

## Reports

**NOTE**

Some pages, such as **Reports**, only appear when an on-premises TFS is configured with the required resources, such as SQL Server Reporting Services and SharePoint.

The **Reports** page opens the [Reporting Services report site](#). This page appears only when your project has been configured with SQL Server Analysis Services and Reporting Services. Also, the option to **Create Report in Microsoft Excel** appears only when reporting has been configured for the project.

If your project is missing one or more pages, you may be able to [add functionality to your on premises TFS deployment](#).

## Reports and Documents

**NOTE**

Some pages, such as **Reports and Documents**, only appear when an on-premises TFS is configured with the required resources, such as SQL Server Reporting Services and SharePoint.

The **Reports** page opens the [Reporting Services report site](#). This page appears only when your project has been configured with SQL Server Analysis Services and Reporting Services. Also, the option to **Create Report in Microsoft Excel** appears only when reporting has been configured for the project.

From the **Documents** page, you can [open project portal](#) and [manage documents and document libraries](#). This page appears only if your project has been configured with a SharePoint Products portal.

If your project is missing one or more pages, you may be able to [add functionality to your on premises TFS deployment](#).

## Settings

From the **Settings** page, you can configure administrative features for either a project or project collection. To learn more about each page, see the following articles. Most of the links open to a web portal administration page. Not all settings are available from the Team Explorer plug-in for Eclipse.

PROJECT	PROJECT COLLECTION	OTHER
---------	--------------------	-------

<ul style="list-style-type: none"> <li>• <a href="#">Security, Group Membership</a></li> <li>• <a href="#">Security, Source Control (TFVC)</a></li> <li>• <a href="#">Work Item Areas</a></li> <li>• <a href="#">Work Item Iterations</a></li> <li>• <a href="#">Portal Settings</a></li> <li>• <a href="#">Project Alerts</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">Security, Group Membership</a></li> <li>• <a href="#">Source Control (TFVC)</a></li> <li>• <a href="#">Process Template Manager</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">Git Global Settings</a></li> <li>• <a href="#">Git Repository Settings</a></li> </ul>
---	---	--

To learn more about settings, see [About team, project, and organizational-level settings](#).

## Refresh Team Explorer or Team Explorer Everywhere

If data doesn't appear as expected, the first thing to try is to refresh your client. Refreshing your client updates the local cache with changes that were made in another client or in TFS. To refresh Team Explorer, do one of the following actions:

- To refresh a page that you are currently viewing, choose  **Refresh** in the menu bar (or choose F5).
- To refresh the project you currently have selected, choose  **Home**, and then choose  **Refresh** (or choose F5).
- To refresh the set of teams defined for the project that you currently have selected, choose **Connect**, and then choose  **Refresh** (or choose F5).

To avoid potential errors, you should refresh your client application under the following circumstances:

- Process changes are made.
- Work item type definitions are added, removed, renamed, or updated.
- Area or iteration paths are added, removed, renamed, or updated.
- Users are added to or removed from security groups, or permissions are updated.
- A team member adds a new shared query or changes the name of a shared query.
- A build pipeline is added or deleted.
- A team or project is added or deleted.

### Resolve images that don't display in Team Explorer

If an inline image isn't displayed in a work item form that you view in Visual Studio Team Explorer, but the image is displayed in the web portal, your credentials might have expired. You can resolve this by completing the following steps:

1. In Visual Studio, select **View > Other Windows > Web Browser** (or use the shortcut **Ctrl+Alt+R**).
2. In the web browser, locate your organization.
3. Sign in with your credentials.
4. Refresh your work item in Team Explorer.

## Related articles

- [Troubleshoot connection](#)
- [Create a project](#)

### Additional tools provided with TFS Power Tools

By installing [TFS Power Tools](#), you gain access to these additional tools through the Team Explorer plug-in for Visual Studio:

- Process Template Editor
- Additional check-in policies for Team Foundation Version Control

- Team Explorer enhancements including Team Members
- Team Foundation Power Tool Command Line
- Test Attachment Cleaner
- Work Item Templates

Additional requirements may apply.

# FAQs about signing up and getting started

3/6/2021 • 3 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Signing up for Azure DevOps is now easier than ever - it's a two-minute process. See the following FAQs, which contain links for getting started.

## How do I sign up for the cloud?

- [Sign up and get started in Azure DevOps Services](#) - a two-minute process.
- You can also sign up and get started with only a single service in Azure DevOps:
  - [Azure Pipelines](#)
  - [Azure Repos](#)
  - [Azure Boards](#)

## How do I get started on-premises?

- Download and install [Azure DevOps Server](#)
- To get started with an on-premises instance, download and install the [latest version of TFS](#).
- [Configure the installation](#), which creates a default collection.
- If you need to create a project, [create one on-premises](#).
- If you don't have access to the project, [get invited to the team](#).
- If it's your first time connecting to a project, see [Connect to a project](#).

## How do I connect with a client tool?

Go to one of the following pages to download a version of Visual Studio or client tool plug-in that supports connecting to a project:

- [Visual Studio](#)
- [Eclipse/Team Explorer Everywhere](#)
- [Android Studio with the Azure DevOps Services Plugin for Android Studio](#)
- [IntelliJ with the Azure DevOps Services Plugin for IntelliJ](#)
- [Visual Studio Code](#)

## How much does Azure DevOps cost?

See the following links for pricing:

- [Azure DevOps Services pricing](#)
- [Azure DevOps on-premises pricing](#)
- [Azure Pipelines only pricing](#)

## How do I share code?

See about [sharing code](#).

## How do I track work?

See [Plan and track work](#).

## What do I do as an admin?

See [Administrator roles](#).

## What client-server compatibility issue are there?

See [Requirements and compatibility](#).

## Can Stakeholders who don't use Visual Studio participate on our team?

Yes. You can provide access to Stakeholders who have no client access license for the following activities:

- **Stakeholder access:** This view allows anyone on your team to check project status and provide feedback. Stakeholders can [track project priorities and provide direction, feature ideas, and business alignment to a team](#).

To grant Stakeholders access, add them to the [Stakeholder access group](#).

- **Provide feedback:** To allow your Stakeholders to provide feedback, you must [grant them specific permissions](#).

## Are there other clients that connect to Azure DevOps? Are there other tools I can use?

Yes. You can connect to a project from one of the following clients:

- [Excel](#) (requires Team Foundation add-in)
- [Project](#) (requires Team Foundation add-in)
- [Project Professional](#)
- [PowerPoint Storyboarding](#) (requires Team Foundation add-in)
- [Azure Test Plans](#)
- [Test & Feedback extension \(previously called the Exploratory Testing extension\)](#)
- [Microsoft Feedback Client](#)

### NOTE

Native support for integrating TFS with Project Server is deprecated for TFS 2017. However, synchronization support is provided by a third party. See [Synchronize TFS with Project Server](#) for details.

Test Manager is deprecated for TFS 2017.

You can also find several open-source clients that have been added to [Marketplace extensions](#). For example, you can install extensions to Visual Studio that support additional features:

- For TFS 2017 and later versions, you can [install the TFS Process Template editor from the Visual Studio Marketplace](#). You can use this version of the Process Editor to modify the old-style work item forms. You can't use it to edit forms associated with the [new web forms](#).
- For TFS 2015 and earlier versions, you can install [TFS Power Tools](#). TFS Power Tools provide enhancements, tools, and command-line utilities that support increased productivity.

**NOTE**

Team Foundation Server Power Tools is deprecated for TFS 2017 and later versions.

## Why can't I connect to a project?

See [Troubleshoot connection](#).

### Related articles

- [Essential services](#)
- [Client-server tools](#)
- [Software development roles](#)
- [Azure DevOps Support](#)
- [Live chat](#) (English only)

# Service limits and rate limits

11/2/2020 • 2 minutes to read • [Edit Online](#)

Learn which service limits and rate limits that all projects and organizations are subject to.

## Work items

- A long text field can contain 1M characters.
- You can't assign more than 100 tags to a work item.
- You can't add more than 1,000 links to a work item.
- You can't add more than 100 attachments to a work item.
- You can't add an attachment size larger than 60 MB to a work item.
- You can have up to 1,000 tasks on a task board
- You can have up to 10,000 work items on a backlog
- You are limited to 5,000 teams in a project
- You can't create more than 150,000 tag definitions per project

## Queries

- The execution time limit for queries is 30 seconds. See [Optimization best practices](#) to improve query performance.
- Query results are limited to 20,000
- Queries are limited in length to 32,000 characters

## Process customization

When customizing the work item types (WITs) defined in the Inheritance or Hosted XML process models, be aware of the limits placed on objects defined in [Work tracking, process, and project limits](#).

## Wiki

Wikis defined for a project are limited to 1 GB per git repository.

### TIP

To derive the size of a wiki/git repository, download the repo to your local computer, unzip the file, and then open the [Properties](#) for the corresponding folder.

## Rate limiting

Azure DevOps Services, like many Software-as-a-Service solutions, uses multi-tenancy to reduce costs and to enhance scalability and performance. This leaves users vulnerable to performance issues and even outages when other users of their shared resources have spikes in their consumption. To combat these problems, Azure DevOps Services limits the resources individuals can consume and the number of requests they can make to certain commands. When these limits are exceeded, subsequent requests may be either delayed or blocked.

See [Rate limits documentation](#) for details

## Data import

- Limited to 300 projects per collection
- See [data import documentation](#) for details

## Next steps

- [Work tracking, process, and project limits](#)

# What are the features in Azure DevOps?

4/21/2021 • 54 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Learn about all the features available to help you plan and track your projects and code, build, test, and release your software applications in Azure DevOps.

If you're new to Azure DevOps, see our overview articles that are designed to give beginners an understanding of the server-client structure and tools supported. For a description of the core services supported through the web portal, see [Essential services](#).

## NOTE

Some features are platform-dependent, based on the following two platforms:

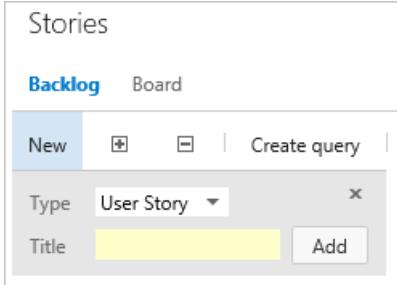
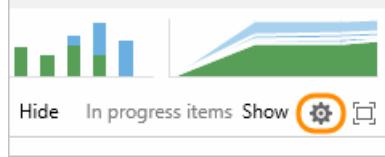
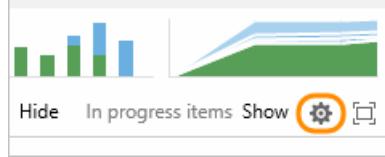
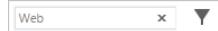
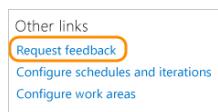
- **Azure DevOps Services** - cloud service
- **Azure DevOps Server** - on-premises

## Access and supported clients

Browsers	Manage users and groups	Access levels
<p><a href="#">Connect to the web portal</a> from the latest versions of these supported browsers:</p> <ul style="list-style-type: none"><li>- Chrome</li><li>- Microsoft Edge</li><li>- Firefox</li><li>- Internet Explorer</li><li>- Safari (Mac)</li></ul>	<p>Add members to your project adds them to the Contributor group. When managing a large group of users, use <a href="#">built-in groups to manage users and their permissions</a>.</p> <p><b>Add team members</b></p> <p>To share and contribute to your project, add users to <a href="#">Azure DevOps Services</a> or your <a href="#">Azure DevOps Server</a>.</p> 	<p>All users that you add to your Azure DevOps organization or to your Azure DevOps Server project have access to Basic features by default, except <b>Stakeholders</b> who have access to a limited set of features, or those added to the Advanced access level in Azure DevOps Server.</p> <ul style="list-style-type: none"><li>- <a href="#">Manage users (Azure DevOps Services)</a> - <a href="#">Change access levels (Azure DevOps Server)</a></li></ul>
<p><b>Integrated Development Environments (IDE)</b></p> <p>Track work and integrate with your code, build, and test environments from the following clients:</p> <ul style="list-style-type: none"><li>- <a href="#">Eclipse (Team Explorer Everywhere)</a></li><li>- <a href="#">Visual Studio</a></li><li>- <a href="#">Android Studio</a></li><li>- <a href="#">IntelliJ</a></li><li>- <a href="#">Visual Studio Code</a></li></ul> <p>To learn how to connect, see <a href="#">Connect to a project</a>.</p>	<p><b>Azure Active Directory (Azure AD) (Azure DevOps Services)</b></p> <p>Control who can access your team's critical resources and key business assets by <a href="#">managing access with Azure Active Directory groups</a>.</p>	<p><b>Permissions</b></p> <p>Control access to specific features by setting permissions for a user or group.</p> <ul style="list-style-type: none"><li>• <b>Area and iteration paths</b><ul style="list-style-type: none"><li>- <a href="#">Build &amp; Release</a></li><li>- <a href="#">Git</a></li><li>- <a href="#">TFVC</a></li><li>- <a href="#">Dashboards</a></li><li>- <a href="#">Queries</a></li><li>- <a href="#">Manage teams and configure team tools</a></li><li>- <a href="#">Test</a></li><li>- <a href="#">Work item tags</a></li></ul></li></ul>
<p><b>Office integration clients</b></p> <p>Use features supported by these familiar clients to manage your project and illustrate your requirements.</p> <ul style="list-style-type: none"><li>- <a href="#">Excel</a></li><li>- <a href="#">Project</a></li><li>- <a href="#">PowerPoint - Storyboarding</a></li></ul>		

# Agile tools to plan and track work

## Backlogs

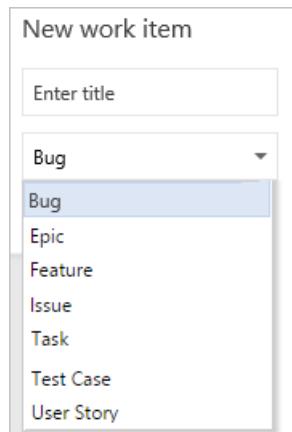
<p><b>Create your backlog</b></p> <p>Plan your project by <a href="#">adding a work item for each user story or requirement</a> you plan to develop.</p> 	<p><b>Move work item to a different project (Azure DevOps Services)</b></p> <p>Choose  <a href="#">Change project</a>, <a href="#">Actions</a> menu in a work item form to <a href="#">move the work item to a different project</a>.</p> <p><b>Full screen mode</b></p> <p>Choose  or  to enter or exit full screen mode.</p> <p><b>Backlog and board settings</b></p> <p>Choose  to configure team backlogs and boards, including <a href="#">show bugs on backlogs and boards</a> and <a href="#">set team backlog levels</a>.</p> 	<p><b>Change work item type (Azure DevOps Services)</b></p> <p>If you added a task instead of a bug and want to change the work item type to bug, you can. Choose  <a href="#">Change type</a> <a href="#">Actions</a> menu in a work item form to <a href="#">change the work item type</a>.</p>
<p><b>Organize your backlog</b></p> <p><a href="#">Group items into a hierarchical list using portfolio backlogs</a> and quickly reorder and re-parent items to effectively manage your deliverables.</p> <p><b>Forecast</b></p> <p>Use the <a href="#">forecast</a> tool to estimate work to be completed in future sprints.</p> <p><b>Storyboard</b></p> <p>Visualize your ideas and user stories and support greater understanding of them by <a href="#">storyboarding them with PowerPoint</a>, also link your storyboards to your backlog work items.</p>	<p><b>View portfolio backlog hierarchy</b></p> <p>Use <a href="#">Parents Show/Hide</a> to drill down into the backlog hierarchy.</p> <p><b>Multi-team backlog ownership</b></p> <p>Easily view and track items <a href="#">owned by other teams</a> and quickly reorder and re-parent items to effectively manage your backlog.</p> 	<p><b>Filter your backlog</b></p> <p>Use <a href="#">Show/Hide in progress</a> to only show or hide items which have moved from the new or proposed state to active or in progress state.</p> <p>Additionally, you can list a subset of items based on keywords <a href="#">keywords</a> or <a href="#">tags</a>.</p> 
	<p><b>Request feedback</b></p> <p><a href="#">Request feedback on working software</a> and easily track responses that capture interaction with video, verbal, or type-written comments.</p> 	<p><b>Feedback client</b></p>

Provide the free Microsoft feedback client to capture their responses to your feedback requests.

## Bug, task, and issue tracking

### Track issues and other types of work

Different types of work items [track different types of work](#) - such as bugs, test cases, risks, issues, and more.

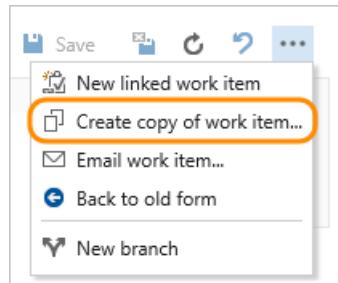


### Bulk modify

Quickly change one or more fields in several work items using [bulk modify in the web portal](#) or [bulk modify using Excel](#).

### Copy or clone a work item

[Copy an existing work item](#) or bulk copy several using [Excel](#).



### Follow a work item

Choose [Follow](#) / [Following](#) [Follow/Following](#) to quickly [start or stop tracking changes made to a work item](#).



### Estimates and time tracking

Track [estimated, completed, and remaining work](#) for tasks and other work items. Several reports and dashboards provide charts that display data based on team capacity and remaining work.

### New work item experience

The [new work item experience](#) provides access to a more modern form, additional features, and the ability to add fields and apply other customizations to the work item type.

### Manage bugs

[Capture and triage bugs](#) using different kinds of tools.

### Choose how you want to track bugs

Each team can [choose to manage bugs on their backlog or along with tasks](#).

### Share plans and information

Share information using work items and [generate summary lists with links to backlogs or queries](#).

### Remove or delete a work item

Remove work items from the backlog by changing their State to Removed. Or, [move them to the recycle bin or permanently delete them](#).



### Tags

[Add tags to work items](#) to filter backlogs and queries. [Bulk update work items](#) or [use work item templates](#) to add or remove tags.

### Discussion

[Add or review comments](#) added to a work item. Start by choosing [discussion](#).

### Integrate Git development with work tracking

Drive Git development and stay in sync as a team to complete backlog items and tasks using the [Git Development section](#). Add branches, create pull requests, and view all development done to support the specific work item.

Development	
	Added file Created 33 minutes ago, <span style="color: green;">Completed</span>
	features/cancel-order-form Updated 35 minutes ago <a href="#">Create a pull request</a>
	Added file Created 35 minutes ago, 4ba415

### Verify a bug, rerun test case

Choose [Verify](#) from the bug work item form context menu to launch the relevant test case in the web runner. For more information, see [Run tests for web apps](#).

### Link work items

Track related work, dependencies, and changes made over time by [linking work items](#).

Links	
ID	Title
Child (3)	
346	Add animated emoticons
347	Implement a service that receiv
348	As a <user>, I can select an em

### Add or modify a field

[Add a custom field \(Azure DevOps](#)

## Rich text comments

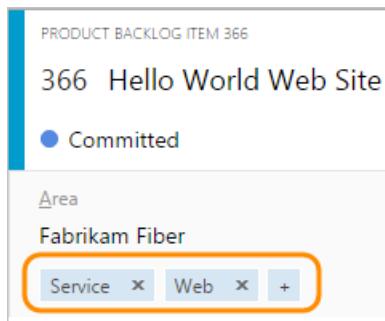
Describe and comment on work using [formatted text, hyperlinks, and inline images](#). Choose or to expand or contract the viewing area.

## Clear HTML formatting

Use or [CTRL+Spacebar](#) to remove formatting from highlighted text.

## Attachments

To support collaboration of work in progress, [add emails, documents, images, log files, or other file types](#) to work items.



## Work item templates

Quickly add new work items based on templates [with pre-populate values for your team's commonly used fields](#).

## History & auditing

Review and query [work item change history](#) to learn of past decisions and support future ones.

[Services | Azure DevOps Server](#) to support tracking additional data requirements or modify an existing field to apply optional rules.

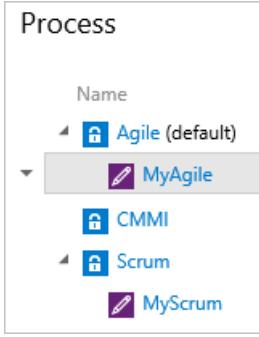
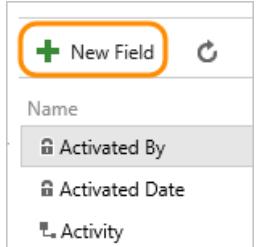
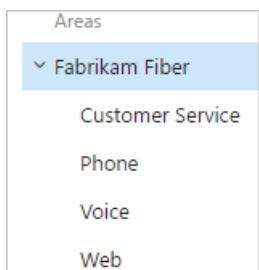
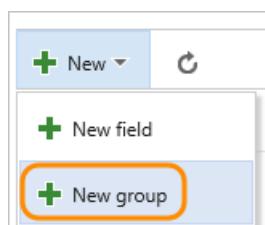
## Restrict access

Limit who can create or modify work items or a work item field based on area path, work item type, or based on your specific conditions.

## Field index

Find descriptions and usage information for each field from the [work item field index](#).

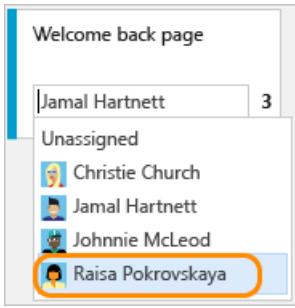
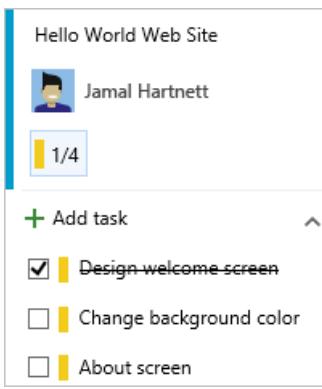
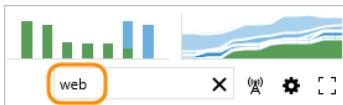
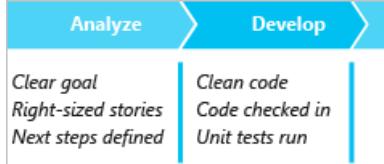
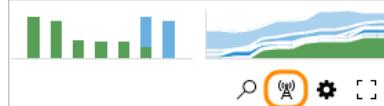
## Customize (Azure DevOps Services)

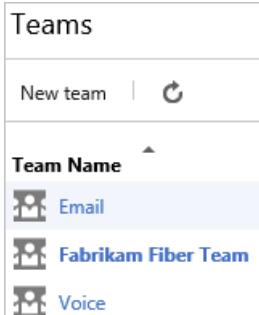
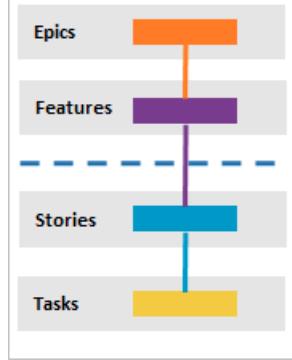
<p><b>Create an inherited process</b></p> <p>The first step in customizing a project is to <a href="#">create an inherited process</a>. You can only customize inherited processes.</p>  <p><b>New work item experience</b></p> <p>The <a href="#">new work item experience</a> provides access to a more modern form, additional features, and the ability to add fields and apply other customizations to the work item type.</p> <p><b>Customize a process</b></p> <p>Customizations you make to an inherited process automatically update all team projects that reference that process. You can customize your project as follows:</p> <ul style="list-style-type: none"> <li>• <a href="#">Add and modify fields</a></li> <li>• <a href="#">Modify the web form layout</a></li> <li>• <a href="#">Modify the workflow states</a></li> <li>• <a href="#">Add a custom work item type</a></li> <li>• <a href="#">Add a custom control</a></li> </ul> <p><b>Change the process used by a project</b></p> <p>To apply customizations to one or more team projects, you <a href="#">change the process they reference to a customized inherited process</a>.</p> <p><b>Enable/disable a process</b></p> <p>To make sure no one creates a project from a process that you don't want used, <a href="#">you can disable it</a>.</p>	<p><b>Add or modify a field</b></p> <p><a href="#">Add a custom field</a> to support tracking additional data requirements or modify an existing field to apply optional rules.</p>  <p><b>Remove a field from a form</b></p> <p>You can <a href="#">remove a custom field</a> and <a href="#">select inherited fields from a work item form</a>. You can also <a href="#">relabel the fields</a> that appear on the form.</p> <p><b>Area path pick lists</b></p> <p>Change the <a href="#">pick list of area paths</a> to support grouping work items by team, product, or feature area.</p>  <p><b>Sprint/iteration pick lists</b></p> <p>Change the <a href="#">pick list of iteration paths</a> to support grouping work into sprints, milestones, or other event-specific or time-related period. Activate sprints for each team.</p> 	<p><b>Review fields</b></p> <p>You can <a href="#">review the list of fields</a> defined for a process, their data type, and the WITs which reference them. For descriptions and usage of each field, see <a href="#">Work item field index</a>.</p> <p><b>Delete a field from the collection</b></p> <p>You can <a href="#">delete a custom field</a> if you find it's no longer required.</p> <p><b>Customize the web form</b></p> <p>For each work item type, you can <a href="#">add custom pages to group additional custom fields</a> and you can organize your forms by placing logically related groups and HTML fields on separate pages within a form.</p>  <p><b>Add a custom work item type</b></p> <p>You can <a href="#">add and modify a custom work item type</a>.</p> <p><b>Customize the workflow</b></p> <p>For each work item type, you can <a href="#">add custom workflow states to support your business tracking needs</a>.</p> <p><b>Delete a process</b></p> <p>Delete those inherited processes that you no longer want used. Choose <b>Delete</b>.</p> <p><b>Set process permissions</b></p> <p>To customize a process, add custom fields, or change the layout of a work item form, you must be a member of the Project Collection Administrators group or be <a href="#">granted explicit permissions to edit a specific process</a>.</p>
---	---	---

## Customize (Azure DevOps Server)

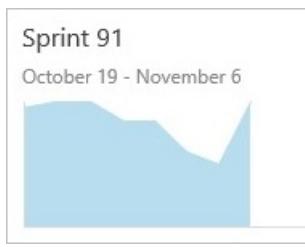
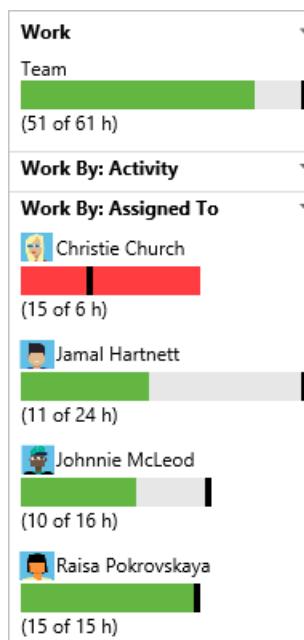
<p><b>Add or modify a field</b></p> <p><a href="#">Add or modify a field</a> to support work tracking and reporting by editing the WIT definition.</p> <p><b>Add rules to a field</b></p> <p>Apply <a href="#">various rules to custom fields</a> to qualify the value it can have, to copy a value, to specify a default, to restrict who can modify it, to enforce pattern matching, or to enforce conditional values.</p> <p><b>Remove a field</b></p> <p><a href="#">Stop tracking a field by removing the field</a> from the work item form of select work item types.</p>	<p><b>Area path pick lists</b></p> <p>Change the <a href="#">pick list of area paths</a> to support grouping work items by team, product, or feature area.</p> <p><b>Sprint/iteration pick lists</b></p> <p>Change the <a href="#">pick list of iteration paths</a> to support grouping work into sprints, milestones, or other event-specific or time-related period.</p> <p><b>Custom pick lists</b></p> <p><a href="#">Define or modify pick list values</a> by editing the work item type definition.</p>	<p><b>Modify the workflow</b></p> <p><a href="#">Design your custom workflow</a> by adding states, transitions, reasons, and optional actions.</p> <p><b>Change the work item form</b></p> <p><a href="#">Change the layout of your work item form</a> by adding fields, custom controls, or tabs.</p> <p><b>Add a custom work item type</b></p> <p><a href="#">Add a custom work item type</a> to track different data requirements.</p>
---	---	---

## Kanban

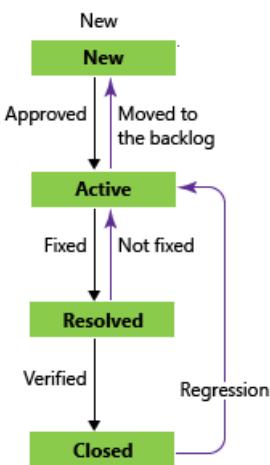
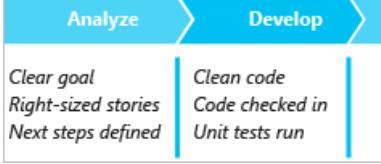
<p><b>Kanban basics</b></p> <p>Use your Kanban board to <b>visualize and track the flow of work</b> from idea to completion as well as quickly update work item fields</p>  <p><b>Drag-n-drop</b></p> <p><b>Drag and drop items</b> on the Kanban board to update status and to reorder and reparent items.</p> <p><b>Add task checklists</b></p> <p>Add and mark tasks as done with <b>lightweight tasks checklists</b>.</p>  <p><b>Filter</b></p> <p>Use <b>key words</b> to filter and find items on the Kanban board.</p> 	<p><b>Set WIP limits</b></p> <p>Set constraints on the amount of work your team undertakes at <b>each work stage</b> to gain access to sprint backlogs and task boards.</p> <p><b>Split columns</b></p> <p>Turn on split columns to <b>track the lag between when items are done in one state and work actually starts in a new state</b>.</p> <p><b>Map your workflow</b></p> <p>Customize columns to support your team's <b>workflow</b> and track work from start to finish.</p>  <p><b>Expedite work with swimlanes</b></p> <p>Use <b>swimlanes</b> to track work at different service-level classes.</p> <p><b>Definition of done</b></p> <p>Support your team to be in sync by <b>specifying requirements to fulfill</b> prior to handoff of items to a downstream work stage.</p> <p><b>Filter by field values or parent work items</b></p> <p>Choose  <b>field filter</b> to <b>filter the board based on assignment, iteration, work item type, or tags</b>.</p>  <p><b>Cumulative Flow Diagram</b></p> <p>With the CFD, you can <b>monitor the count of work items as they progressively move through various states which you define</b>.</p>	<p><b>Customize cards</b></p> <p>Add <b>fields to cards</b> that you can edit directly on your Kanban and task boards.</p>  <p><b>Live updates</b></p> <p>Enable <b>live updates</b> to automatically refresh your Kanban board when changes are made by others or to the board settings.</p>  <p><b>Add inline tests</b></p> <p>Add, run, and update tests with inline test on your Kanban board.</p> <p><b>Add checklists to features and epics</b></p> <p>Add and mark user stories and other work items as done from your <b>Kanban features or epics boards</b>.</p> <p><b>Set team's card reorder preference</b></p> <p>You can preserve the backlog priority when you move a card to a new column by setting your team's <b>Kanban board card reordering setting</b>.</p> <p><b>Enable/disable card annotations</b></p> <p>Turn on or off <b>task checklists or inline tests</b> for your Kanban board.</p> <p><b>Configure inline tests</b></p> <p>Configure how new inline tests are added to the Kanban board: <b>create a new test plan/test suite or choose an existing test plan</b>.</p>
--	--	--

<h3>Add another team</h3> <p>Add and structure teams and organize work to support team autonomy and organizational alignment. Teams can manage their work independently of one another while the organization gains visibility across all teams.</p>  <p><b>Set team defaults</b></p> <p>Several Agile tools reference the team's default area path, iteration path, and activated sprints to automatically filter the set of work items they display. <a href="#">Understand how defaults are used</a>.</p>	<h3>Set up a team hierarchy</h3> <p>By configuring your teams and backlogs into an hierarchical structure, program owners can more easily track progress across teams, manage portfolios, and generate rollup data.</p> <p><b>Autonomy and alignment</b></p> <p>As your organization grows, your tools can grow to support a culture of team autonomy and organizational alignment.</p> <p><b>Scale your tools and practices</b></p> <p>Incrementally adopt practices that scale to create greater rhythm and flow within your organization, engage customers, improve project visibility, and develop a productive workforce.</p>	<h3>Portfolio management</h3> <p>Manage a portfolio of backlogs and gain insight into each team's progress as well as the progress of all programs.</p>  <p><b>Scaled Agile Framework</b></p> <p>Structure team projects to support epics, release trains, and multiple backlogs to support the Scaled Agile Framework.</p>
---	--	--

## Scrum

<p><b>Define sprints</b></p> <p>Schedule and activate your team's sprints to gain access to sprint backlogs and task boards.</p> <p><b>Select team sprints, set team defaults</b></p> <p>Several tools reference the team's default and active iteration paths or sprints. For the Agile tools to work best, each team needs to <a href="#">set their team area path(s)</a> and <a href="#">iteration paths</a> to support their work tracking activities.</p> <p><b>Plan sprints</b></p> <p>Build your sprint backlog, add tasks, and load balance work across your team as you <a href="#">plan your sprint</a>.</p> <p><b>Track work on your task board</b></p> <p>Use your <a href="#">task board</a> during your daily Scrum meetings to view and update progress.</p>	<p><b>Velocity &amp; forecasting</b></p> <p>Use velocity charts and forecast tools to estimate work that can be completed in future sprints.</p>  <p><b>Sprint burndown charts</b></p> <p>Monitor progress and review team patterns from <a href="#">sprint burndown charts</a>.</p> 	<p><b>Manage resources</b></p> <p>Use capacity planning tools to track individual, team, and activity over and under capacity for a sprint.</p> 
---	---	---

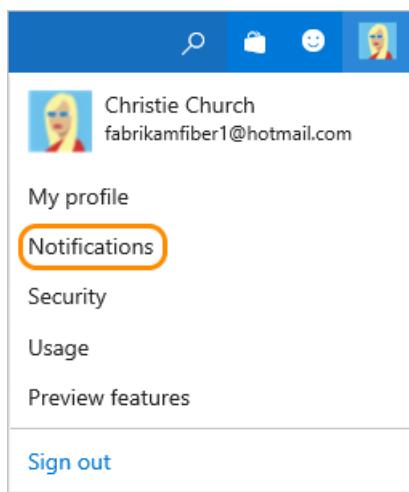
## Workflow

<p><b>What is workflow?</b></p> <p>You use workflow to track the progress of work as it moves from new, active, to complete or closed. Each workflow consists of a set of states, the valid transitions between the states, and the reasons for transitioning the work item to the selected state.</p>  <p><b>Default workflows</b></p> <p>Each process <a href="#">defines the workflow</a> for each work item type to track progress from newly defined, to in progress, to completed or closed.</p>	<p><b>Kanban workflow</b></p> <p>You can fully customize your Kanban board to map the workflow your team uses by <a href="#">adding and renaming columns</a></p>  <p><b>Customize the workflow</b></p> <p>For Azure DevOps Services: <a href="#">add custom workflow states to support your business tracking needs</a>. For Azure DevOps Server: <a href="#">Design your custom workflow</a> by adding states, transitions, reasons, and optional actions.</p> <p><b>States</b></p> <p>States allow you to <a href="#">track the status of work</a>. For example, a bug moves from <b>Active</b>, <b>Resolved</b>, and <b>Closed</b> to correspond to when it's defined, fixed, and verified as fixed.</p> <p><b>Transitions</b></p> <p>Transitions specify the <a href="#">valid progressions and regressions from state to state</a> for a work item type.</p> <p><b>Reasons</b></p> <p>Each transition <a href="#">specifies a default reason as well as optional reasons</a> for tracking the change in state.</p>	<p><b>Update fields during workflow changes (Azure DevOps Server)</b></p> <p>You can <a href="#">define rules that change a field value</a> whenever you change the state, perform a transition, or select a reason.</p> <p><b>Apply workflow conditional field rules (Azure DevOps Server)</b></p> <p>You can define rules that <a href="#">change a field value based on the contents of other fields</a> during workflow changes.</p> <p><b>Restrict who can make changes during workflow transitions (Azure DevOps Server)</b></p> <p>Set a condition field rule that applies to a group to <a href="#">restrict who can make changes to a workflow or a field</a>.</p> <p><b>Event-generated workflow changes or field assignments (Azure DevOps Server)</b></p> <p><a href="#">Add an action</a> to a custom workflow definition to automatically transition work items or specify a field value based on an internal Azure DevOps Server event or external event.</p> <p><b>Visual workflow design tool (Azure DevOps Server)</b></p> <p>You can change the workflow or view the workflow state diagram by using the <a href="#">Process Editor</a>, a power tool for Visual Studio.</p>
---	--	---

## Alerts and notifications

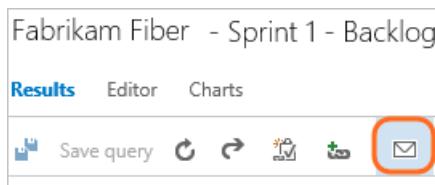
## Personal and team notifications or alerts

Get notified as changes occur to work items, code reviews, source control files, and builds by setting personal notifications or team notifications.



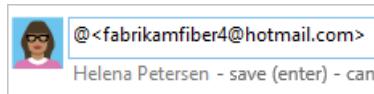
## Share queries and sprint plans

Email a query or [sprint plan](#).



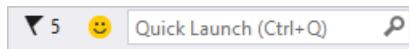
## Quick alerts to team members

Use the **@mention** control to send email to team members to bring them into a discussion around work changes, pull requests, or other items.



## Client feature flag updates

Alert flag within the IDE automatically notifies you of the latest client changes.



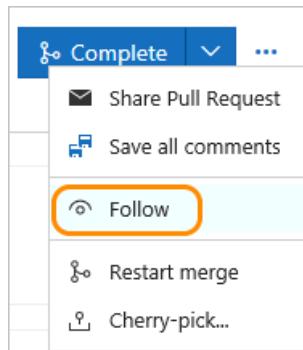
## Follow a work item

Choose [Follow](#) / [Following](#) to quickly start or stop tracking changes made to a work item.



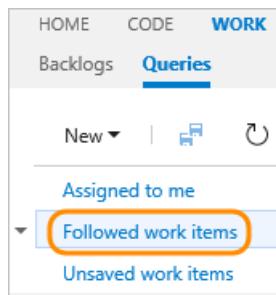
## Follow a pull request

To [track the progress of a single pull request](#), choose [Follow](#) from the menu.



## Manage work items you follow

From the **Work > Queries** page you can view the list of work items that you're following.



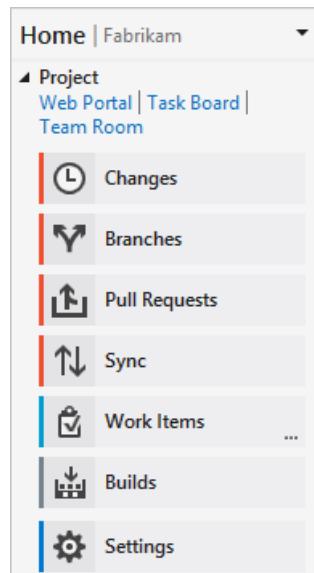
## Frequent on-line feature updates

Check the [News](#) for product updates, or read about them by accessing the News link in your web portal.

**Code: Git**

## Get started with Git in Visual Studio

To get started working with Git, clone a repository, add code, and create branches in Azure DevOps Services or Visual Studio. Learn how to commit, publish, and conduct a pull request of your changes.

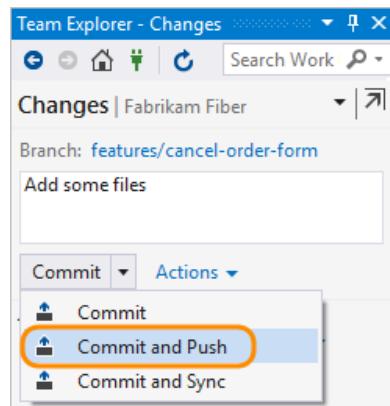


### Clone repositories

To work locally, you [clone a repository](#).

### Commit changes

Enter commit messages and [quickly push your local changes to the shared repo](#).



### Pull requests

Use [pull requests](#) to review and merge branch code to a main branch.

### Sync

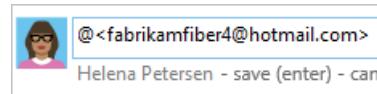
Quickly [sync](#) your local branch with a shared repo.

## Get started using Eclipse

[Work with Git repositories](#) using the Team Explorer Everywhere IDE for Eclipse.

### Add reviewers to get feedback

Use the [@mention control](#) to add [reviewers](#) to your pull request to get their feedback about your changes.



### Resolve Git merge conflicts

Merge conflicts occur when commits have changes to the same files as other newer commits in the branch history. Learn how to [prevent and resolve merge conflicts](#).

### Code search

Maximize cross-team collaboration and code sharing by finding code across all the projects to which you have access. Narrow down your results and focus in on code by using [filters](#), [preview code](#), [view history](#), [compare versions](#), and more



### Get notified about pull requests

Subscribe to email alerts to get notified about [new pull requests](#), [changes](#), [approvals](#), and [rejections](#).

### Set branch policies

To improve code quality, [set branch policies](#) to require code reviews or automatically add reviewers.

### Automatically build pull requests

Set a branch policy to automatically generate a build for a pull request to selected branches.

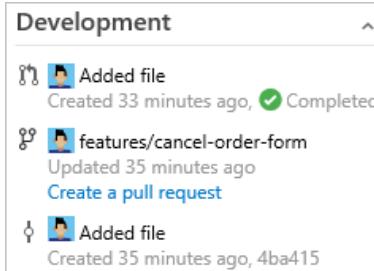
### Create Git repositories

When you create a project with Git as your version control system, you automatically create a Git repo. You can [Create additional Git repos](#) from the admin context.

### Rename a Git repository

## Integrate Git development with work tracking

Drive Git development and stay in sync as a team to complete backlog items and tasks using the [Git Development section](#). Add branches, create pull requests, and view all development performed to support the specific work item.



### Quickly link work items to pull requests

Use the [#ID control](#) to [link work items](#) to your pull request to support tracking work.

### Get started using Xcode

[Work with Git repositories](#) using the Xcode IDE.

### Git commands

Use [Git command line tools](#) when you need to perform select manual tasks or to automate work using a script.

### Bypass a branch policy

Grant an [Exempt from policy enforcement permission](#) to a user or group.

### Rebase a branch

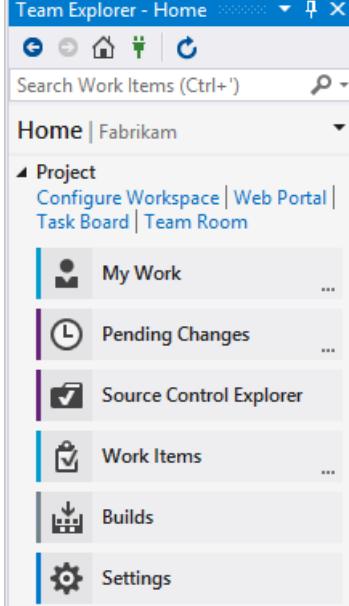
Before merging a branch into main, you may choose to first [rebase your branch onto the latest commit in main](#).

### Git permissions

Set permissions on a [Git project](#), [repository](#), or [branch](#) from the context menu or from the web portal administration page.

	Rename Git repos from the admin context.	
--	--	--

## Code: TFVC

<p><b>Get started with TFVC in Visual Studio</b></p> <p><a href="#">Develop and share your code.</a></p> <p>Learn how to configure your workspace, check-in your code, compare file changes, and view file history.</p>	<p><b>Track changesets</b></p> <p>Find information about which <a href="#">branches have received a particular set of changes and when those changes were merged</a>.</p>	<p><b>Code search</b></p> <p>Find code across all the projects to which you have access. Narrow down your results and focus in on code by using <a href="#">filters</a>, <a href="#">preview code</a>, <a href="#">view history</a>, <a href="#">compare versions</a>, and more</p>
	<p><b>Request code review</b></p> <p>Increase overall code quality and reduce the risk of creating bugs by <a href="#">requesting a code review when you check-in code</a>.</p> <p><b>Review history of a file</b></p> <p>Get detailed information about <a href="#">what changes have been made to your files</a>.</p> <p><b>Suspend work</b></p> <p>Use <a href="#">shelvesets</a> when you need to set aside some or all of your work in progress.</p> <p><b>Manage branches, isolate risk</b></p> <p>Use branches and locks to <a href="#">isolate risk introduced by work done by different teams</a>.</p> <p><b>Merge branches</b></p> <p>Integrate work completed in <a href="#">different branches</a> during certain phases of your project.</p> <p><b>Set check-in and check-out policies</b></p> <p>Enforce practices that lead to better code and more efficient group development by <a href="#">setting check-in/check-out rules</a>.</p>	<p><b>Search code</b></p> <p><b>Subscribe to alerts when check-ins occur</b></p> <p>Get notified when someone checks in code to your TFVC project by <a href="#">subscribing to receive email alerts</a>.</p> <p><b>Version control locks</b></p> <p>Lock files or folders when you need to prevent them from being checked out or modified.</p> <p><b>Download files from the server</b></p> <p>Get the latest files from the server on a regular basis so that the code you develop is compatible with the code developed by others on your team.</p> <p><b>TFVC permissions</b></p> <p>Set permissions on <a href="#">select code management tasks</a> from the context menu for TFVC files or folders or the admin context for the project.</p>
<p><b>Set up local or server workspaces</b></p> <p>Create a local workspace that maps to the code base of interest.</p> <p><b>Resolve conflicts</b></p> <p>Support for <a href="#">Resolve conflicts</a> that arise when several people work concurrently on a file.</p> <p><b>Compare files and folders</b></p> <p>Compare server folders and local folders to each other, and view the differences between the contents of each folder.</p>		

## Azure Artifacts (Azure DevOps Services)

<p><b>What is Azure Artifacts?</b></p> <p>Azure Artifacts is the new name for what was previously Package Management. Azure Artifacts helps you <a href="#">manage code sharing by automating common tasks for discovering, consuming, and sharing components</a>.</p> <p><b>Create feeds</b></p> <p><a href="#">Create feeds</a> to share code through packages.</p> <p><b>Move existing file shares to the cloud</b></p> <p>Eliminate dependencies on on-premises file shares and hosted instances of NuGet.Server by <a href="#">moving your packages to Azure DevOps Services</a>.</p>	<p><b>Discover and consume packages</b></p> <p><a href="#">Consume packages</a> by connecting to a feed.</p> <p><b>Publish packages to feeds</b></p> <p><a href="#">Publish packages</a> to share code with your team and your organization.</p> <p><b>Add identities to your feeds</b></p> <p><a href="#">Give teams and service identities</a> access to your feeds.</p>	<p><b>Bootstrap the developer environment</b></p> <p>Increase your team's velocity and decrease the amount of code duplication across your organization. Access a set of tools and conventions for integrating Azure DevOps Services NuGet into your workflow by <a href="#">getting the NuGet VSS.PackageManagement.Bootstrap package</a>.</p> <p><b>Remove a NuGet package from a feed</b></p> <p>Unlist or remove a package <a href="#">Delete packages and recover deleted packages from the recycle bin in Azure Artifacts</a> you no longer want users to discover.</p> <p><b>Secure feeds</b></p> <p>Control who can <a href="#">contribute to or consume from a feed</a>.</p>
--	--	---

## Continuous delivery

### Build

#### Define builds

Start from a build template and customize your build from there. Build for [Windows](#), [iOS](#), Android, Java (Ant, Maven, or Gradle), or Linux using the same domain-specific languages you use every day on your dev machine. [Build Xamarin apps](#) for both iOS and Android and run tests on the Xamarin Test Cloud as part of the build.

#### Customize build process using scripts

[Use a script](#) to add your team's business logic to your build process.

#### Build agents and agent pools

At least one [agent](#) is required to build your code. As you scale your system with more code, people, and builds, you'll need more build agents organized within [agent pools](#). You can use both on-premises or Microsoft-hosted agent pools.

#### Gated check-in (TFVC, Azure DevOps Services)

Use [gated check-in](#) to protect against breaking changes when checking code into TFVC.

#### Branch policies (Git)

Improve code quality by [setting branch policies](#) to ensure builds are never broken or getting the right people to review changes.

#### Specify your build steps

Add steps to specify what you [want to build](#), the [tests to run](#), and [all the other steps](#) needed to complete the process.

 Build an Android app using Gradle

 Sign and align Android APK files

 Build with Apache Ant

 Build using a Gradle wrapper script

 Grunt: The JavaScript Task Runner

 Gulp: Node.js task-based build system

 Index source code and publish symbols

 Build with Apache Maven

 Build with MSbuild

 SonarQube for MSbuild

 Visual Studio and MSbuild

 Build an Android app with Xamarin

 Build an iOS app with Xamarin on macOS

## Build variables

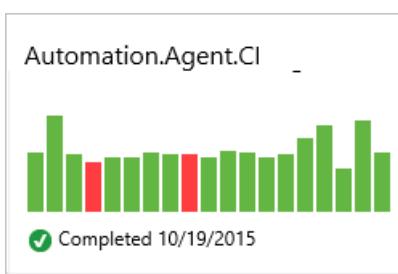
Use [predefined variables](#) or add your custom variables when configuring your build definition or your build scripts.

## Continuous integration builds

Define a CI build that compiles and tests your solutions whenever your team checks in code.

## Build summary charts

View real-time build status and [add build summary charts to your dashboards](#).



## Code coverage charts

From the Code Coverage tab on a Build summary page, you can view percentage of code coverage as well as upload code coverage data in Jacoco or Cobertura formats.

## Audit changes

Determine who [changed what in the build definition and when they did it](#).

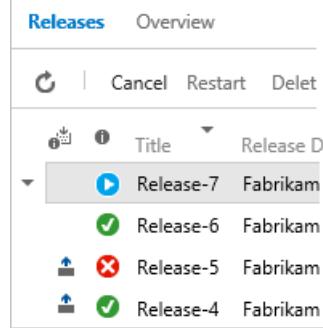
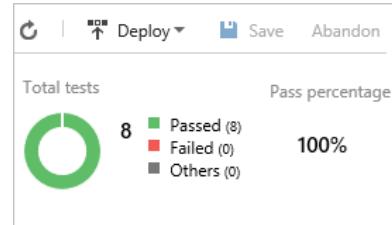
## Build retention policies

Define policies to automatically delete old completed builds to minimize clutter.

## Build permissions

Determine who can [define, delete, and manage builds](#).

## Release

<p><b>Automate deployments</b></p> <p>Reduce time-to-market and respond to customer feedback with greater agility by <a href="#">automating your release process</a>. Deploy applications across platforms to all environments of the pipeline with just one selection.</p> 	<p><b>Works for any app</b></p> <p>Deploy <a href="#">any type of application across multiple platforms</a> including Windows and Linux, whether on-premises or in the cloud.</p> <p><b>Approval workflows</b></p> <p>Streamline your application release workflow by <a href="#">routing pre- and post-deployment approvals</a> to multiple approvers or teams.</p> <p><b>Release notifications</b></p> <p>Receive email messages as releases occur. Approvers receive notifications automatically when a release is waiting for approval.</p> <p><b>Full traceability</b></p> <p>Monitor the status of your release pipelines and track every deployment in each of the environments. Retain full audit history of all activities performed on a release with detailed release logs and approval tracking.</p> <p><b>Release logs</b></p> <p>View or download log files as zip files. Log files contain the status for each step or task of a release, for each of the environments in the release definition. Each completed release--succeeded, failed, or abandoned--<a href="#">includes a live log file, details, and history for each step or task</a>.</p> <p><b>Triggers</b></p> <p>Automate release deployment by <a href="#">defining the events that trigger a release</a>.</p> <p><b>Variables</b></p> <p>Lookup the description for all <a href="#">release system, global, and agent variables</a>.</p>	<p><b>Release names</b></p> <p>Specify the <a href="#">naming and numbering scheme</a> you want used when adding releases.</p> <p><b>Global configuration properties</b></p> <p>Simplify management of custom values that you use to configure multiple releases by <a href="#">specifying custom values for any of the tasks in any of the environments of a release definition</a>.</p> <p><b>View test results</b></p> <p>Open the <b>Tests</b> tab to view a summary of the test results, including pass/fail percentages and run duration. Sort the test results into groups or filter the results to show just passed, failed, or other results.</p> 
<p><b>When to use Azure Pipelines or Build &amp; Release in Azure DevOps Server?</b></p> <p>Evaluate how Azure Pipelines and Build &amp; Release in Azure DevOps Server can help you in <a href="#">your development and deployment efforts</a>.</p> <p><b>Release definitions</b></p> <p>Add a release definition by <a href="#">choosing the build version, target release environments, and tasks</a>.</p> <p><b>Release environments</b></p> <p>Define and clone release environments, logical entities that represent where you want to deploy a release, such as a collection of servers, a cloud, multiple clouds, or an app store.</p> <p><b>Artifacts</b></p> <p>A release is fundamentally defined by <a href="#">versioned artifacts that make up the release</a>. As you deploy the release to various environments, you deploy and validate the same artifacts on all environments.</p> <p><b>Tasks</b></p>		<p><b>Add release summary to dashboard (Azure DevOps Services)</b></p> <p>Add a <a href="#">release summary chart</a> to a team dashboard.</p> <p><b>Extend and customize</b></p> <p>Create workflows tailored to your process by customizing our tasks, or extend with your own custom tasks.</p>

Automate release deployment by defining the events that trigger a release.

### Agents and agent pools

Agent pools are the execution containers that specify the security context and runtime environment for the agents that run when you deploy a release.

 Azure SQL Database Deployment  
Deploy Azure SQL DB using DACPAC

 Azure Web App Deployment  
Publish a Visual Studio Web project to Azure Web App using Web Deploy

 Chef  
Deploy to Chef environments by editing environment attributes

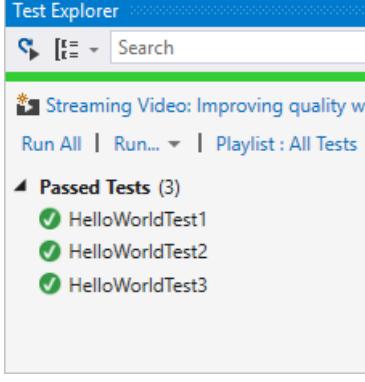
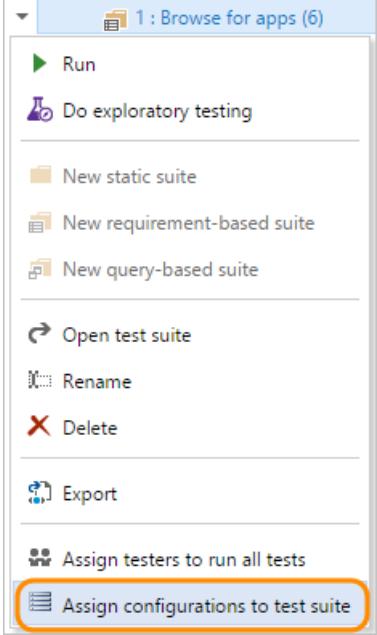
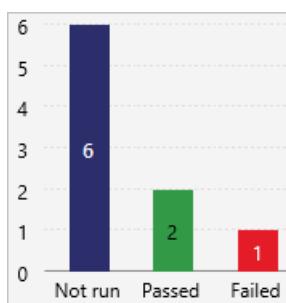
 Chef Knife  
Run Scripts with knife commands on your workstation

 Docker  
Deploy a docker image to a remote machine

### Manage permissions

Grant or deny permissions to manage release definitions, environments approvers, or release permissions. Set permissions for users, groups, or per release definition.

## Test

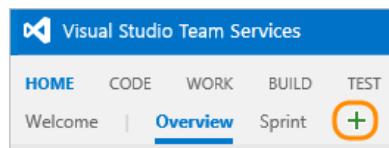
<p><b>Comprehensive testing</b></p> <p>Perform exploratory, manual, system, and user acceptance tests for any app, in any language. Using Visual Studio or 3rd-party test frameworks, you can include automated tests with builds and releases for continuous integration and deployment.</p> <p><b>Unit testing with Git</b></p> <p>Create <a href="#">unit tests</a> and run them frequently to make sure your code is working properly.</p>  <p><b>Manual test plans and test cases</b></p> <p>Get started by <a href="#">creating test plans and test cases</a> to track manual testing for sprints or milestones.</p> <p><b>Shared steps and shared parameters</b></p> <p>Create <a href="#">shared steps</a> to include often repeated sequence of steps in your manual test cases, such as logging in. Repeat manual tests with different data using <a href="#">shared parameters</a>.</p>	<p><b>Coded UI testing</b></p> <p>Use Visual Studio to create <a href="#">coded UI tests</a> to test your application's user interface.</p> <p><b>Run test with your builds for continuous integration</b></p> <p>Use continuous integration builds to <a href="#">run tests automatically</a>.</p> <p><b>Review automated test results after a build</b></p> <p><a href="#">Review your test results</a> to analyze any problems that were found.</p> <p><b>Quickly assign configurations to test plan, test suite, or test case</b></p> <p>From the context menu of a test plan, test suite, or test case, you can assign a configuration.</p> 	<p><b>Exploratory testing</b></p> <p>Explore user stories without test cases or test steps using <a href="#">Azure Test Plans and exploratory testing</a>.</p>  <p>Or, <a href="#">download and install the Test &amp; Feedback extension</a>. Capture screenshots, annotate them, and submit bugs while you explore your web app - all directly from your Chrome browser.</p> <p><b>Record and play back manual tests</b></p> <p>With Azure Test Plans, you can <a href="#">record your keystrokes and gestures while you test an application</a>. The next time you run the test, you can play back your actions quickly and accurately.</p> <p><b>Track test status and test results</b></p> <p>Quickly <a href="#">view the status</a> of your testing using lightweight charts.</p>  <p><b>Test environments</b></p> <p>Specify a combination of hardware and software that represents a user or machine environment in which your app runs.</p> <p><b>Test permissions</b></p> <p>Set permissions on who can <a href="#">manage test configurations, test environments, and publish and delete test results</a>.</p>
---	---	--

## Dashboards and reports

## Charts and dashboards

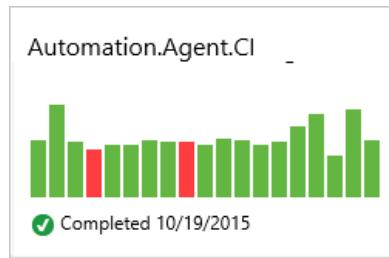
### Multiple team dashboards

Each team can create several [team dashboards](#) to help keep both the team and Stakeholders in sync. Each dashboard tile provides quick access to the progress of builds, status of work items, or latest code changes.



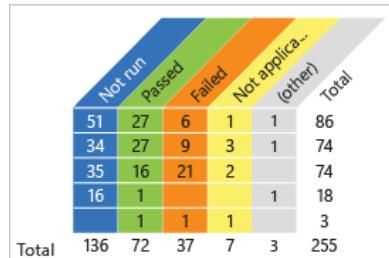
### Build history charts

Add [build history charts](#) to your dashboards.



### Test charts

Track the status of your [test progress](#) and [test runs](#). Optionally add these charts to a dashboard.



### Test quality trend charts

Add [failure and duration charts](#) for tests run as part of your build to your team dashboard.



Restrict or allow team members to manage dashboards ([Azure DevOps Services](#))

Set permissions to [restrict or allow team members to manage dashboards](#).

### Capacity planning and tracking

Easily track how much work your team has completed and has left to do in a sprint by adding the [sprint capacity chart widget](#) to your dashboard.



### Share dashboards with Stakeholders

Grant non-licensed users access as Stakeholders ([Azure DevOps Services](#) | [Azure DevOps Server](#)) so they can view progress, run queries, and contribute ideas.

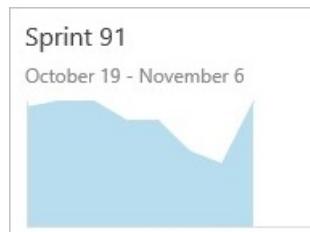
### Velocity charts

[Team velocity](#) tracks the total estimated effort (story points or size) of backlog items (user stories or requirements) completed or still in progress within each sprint.



### Sprint burndown charts

Monitor progress and review team patterns from [sprint burndown charts](#)



### Add release summary to dashboard ([Azure DevOps Services](#))

### Edit dashboard mode

Add, remove, move, and configure widgets by [choosing Edit dashboard](#). Select the checkmark to exit.



### Auto-refresh dashboards

You can [enable auto-refresh](#) for any team dashboard, and it automatically updates every five minutes. This is a useful feature for when your dashboard serves as a team wallboard.

### Widget catalog

Add [widgets](#) to your dashboard to provide information and monitor the data your team needs.



### Work item query charts

View the status of work in progress by [charting the results of a flat-list query](#). You can create several types of charts—such as pie, column, or trend—for the same query. Optionally add these charts to a dashboard.

### Drag-n-drop layout

Configure the layout to your specifications by [dragging tiles into the sequence you want](#).

### Cumulative flow diagrams

Track the progress of work on your backlogs through the [CFD charts](#).

### Power BI dashboards ([Azure DevOps Services](#))

You can create dashboards, individual reports, or explore data collected for your Azure DevOps organization once you [connect to Power BI](#).

	Add a release summary chart to a team dashboard.
--	--

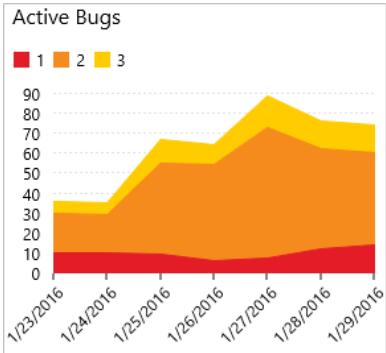
## Power BI dashboards and reports (Azure DevOps Services)

<p><b>Basic Power BI concepts</b></p> <p>The 3 major building blocks of Power BI are <a href="#">dashboards</a>, <a href="#">reports</a>, and <a href="#">datasets</a>.</p> <p><b>Get started</b></p> <p>You can <a href="#">create dashboards, individual reports, or explore data</a> collected for your organization once you connect to Power BI.</p>	<p><b>Connect to Power BI</b></p> <p>Steps required to authorize Power BI to access your organization.</p> <p><b>Available data</b></p> <p>The <a href="#">Power BI Data Connector</a> supports building reports to track status and trend work items.</p>
---	--

## SQL Server Reports (Azure DevOps Server)

<p><b>Reporting Services reports</b></p> <p>You can <a href="#">analyze the progress and quality of your project by using the out-of-the-box reports in SQL Server Reporting Services</a>. These reports aggregate metrics from work items, version control, test results, and builds. They are uploaded when you create a project based on the process - <a href="#">Agile, Scrum, or CMMI</a> - that you choose.</p> <p><b>Add Reporting Services reports</b></p> <p>If you need to add reporting services to a project or on-premises Azure DevOps Server after you've created your team projects, you can by <a href="#">adding a report server and uploading reports</a>.</p> <p><b>Manage the data warehouse</b></p> <p>The reporting warehouse is a traditional data warehouse that consists of a <a href="#">relational database and an Analysis Services database</a>. You manage it through the following activities:</p> <ul style="list-style-type: none"> <li>• <a href="#">Manually process the data warehouse</a> <ul style="list-style-type: none"> <li>- Rebuild the data warehouse</li> <li>- Resolve schema conflicts</li> <li>- Change a process control setting</li> </ul> </li> </ul>	<p><b>Build reports</b></p> <p>Build reports track the quality of software under development. By defining tests to run automatically as part of each build definition and instrumenting tests to gather code coverage data, you can gain insight about the quality of the builds, tests, and code.</p> <ul style="list-style-type: none"> <li>• <a href="#">Build Quality Indicators</a> (Agile &amp; CMMI)             <ul style="list-style-type: none"> <li>- <a href="#">Build Success Over Time</a></li> <li>- <a href="#">Build Summary</a></li> </ul> </li> </ul> <p><b>Test and bug reports</b></p> <p>Test planning reports support monitoring the test progress and coverage of backlog items or user stories. Bug tracking reports illustrate the team's capacity to find and resolve bugs.</p> <ul style="list-style-type: none"> <li>• <a href="#">Test Case Readiness</a> <ul style="list-style-type: none"> <li>- <a href="#">Test Plan Progress</a></li> <li>- <a href="#">Bug Status</a> (Agile &amp; CMMI)</li> <li>- <a href="#">Bug Trends</a> (Agile &amp; CMMI)</li> <li>- <a href="#">Reactivations</a> (Agile &amp; CMMI)</li> </ul> </li> </ul> <p><b>Required team activities to generate useful reports</b></p> <p>To gain useful, actionable information from your reports, <b>team members must perform certain activities</b>.</p>	<p><b>Project management</b></p> <p>Project management reports provide insight into how much work the team is tackling within a sprint or release, and the rate of their progress. By linking work items and updating specific fields as work is performed, you can track the progress of individual stories and be able to more accurately estimate future activities.</p> <p><i>Scrum reports</i></p> <ul style="list-style-type: none"> <li>- <a href="#">Backlog Overview</a></li> <li>- <a href="#">Release Burndown</a></li> <li>- <a href="#">Sprint Burndown</a></li> <li>- <a href="#">Velocity</a></li> </ul> <p><i>Agile and CMMI</i></p> <ul style="list-style-type: none"> <li>• <a href="#">Burndown and Burn Rate</a> <ul style="list-style-type: none"> <li>- <a href="#">Remaining Work</a></li> <li>- <a href="#">Requirements Overview</a> (CMMI)</li> <li>- <a href="#">Requirements Progress</a> (CMMI)</li> <li>- <a href="#">Status of All Iterations</a> (similar to Velocity)</li> <li>- <a href="#">Stories Overview</a> (Agile)</li> <li>- <a href="#">Stories Progress</a> (Agile)</li> <li>- <a href="#">Unplanned Work</a></li> </ul> </li> </ul> <p><b>Set permissions to view or create reports</b></p> <p>Enable members of your team to <a href="#">view or manage Reporting Services reports</a>. To create or modify reports, you need to grant them access to read databases.</p>
---	--	--

## Widgets

<p><b>What is a widget?</b></p> <p>You build your dashboards by adding information tiles or widgets. The <a href="#">widget catalog</a> provides a number of predefined widgets.</p> <p><b>Drag-and-drop widgets</b></p> <p>Drag widgets, tiles, or charts anywhere on a dashboard to <a href="#">configure the layout you want</a>.</p> <p><b>Informational content and other links</b></p> <p><b>Markdown widget</b></p> <p>Adds a configurable tile to your dashboard to <a href="#">display any type of information, guidance, or links</a> that you want using markdown syntax.</p> <div data-bbox="171 781 557 932"><p>Team Updates</p><ul style="list-style-type: none"><li>New - User Story 2830 created</li><li>New - Bug 9530 created</li><li>Update - Today's design meeting has been canceled</li><li>Update - Spec is ready for User Story 4295</li><li>Reminder - Work on P0 bugs before features</li></ul></div> <p><b>Team member</b></p> <p>Opens the team's quick dialog to <a href="#">add or remove team members</a>.</p> <div data-bbox="171 1118 557 1327"><p>Members</p></div> <p><b>Team rooms</b></p> <p>Provides <a href="#">status and access to a team room</a>, an archived space to discuss work in progress, ask questions, share status, and clarify issues that arise.</p> <p><b>Visual Studio widget</b></p> <p>Provides <a href="#">links to open or download Visual Studio</a>. The Visual Studio IDE client comes with the Team Explorer plug-in which provides quick access to several features (some of which aren't available through the web portal).</p> <p><b>Welcome widget</b></p> <p>Provides quick access to <a href="#">getting started info on how to track work, code, build, and test</a>.</p>	<p><b>Plan and track work</b></p> <p><b>Assigned to me widget</b></p> <p>Provides quick access to <a href="#">work items assigned to the logged in user</a>.</p> <p><b>Chart for work items</b></p> <p>Adds a configurable tile to display the <a href="#">chart for a shared query</a>.</p> <div data-bbox="599 489 985 842"><p>Active Bugs</p><table border="1"><thead><tr><th>Date</th><th>1</th><th>2</th><th>3</th><th>Total</th></tr></thead><tbody><tr><td>1/23/2016</td><td>10</td><td>25</td><td>10</td><td>45</td></tr><tr><td>1/24/2016</td><td>10</td><td>25</td><td>10</td><td>45</td></tr><tr><td>1/25/2016</td><td>10</td><td>25</td><td>10</td><td>45</td></tr><tr><td>1/26/2016</td><td>10</td><td>25</td><td>10</td><td>45</td></tr><tr><td>1/27/2016</td><td>10</td><td>25</td><td>10</td><td>45</td></tr><tr><td>1/28/2016</td><td>10</td><td>25</td><td>10</td><td>45</td></tr><tr><td>1/29/2016</td><td>10</td><td>25</td><td>10</td><td>45</td></tr></tbody></table></div> <p><b>New work item</b></p> <p>Add <a href="#">work items</a> pre-scooped to your team's default area and iteration paths.</p> <div data-bbox="599 1057 885 1484"><p>New work item</p><p>Enter title</p><p>Bug</p><ul style="list-style-type: none"><li>Bug</li><li>Epic</li><li>Feature</li><li>Issue</li><li>Task</li><li>Test Case</li><li>User Story</li></ul></div> <p><b>Other links widget</b></p> <p>Provides quick access links from a team dashboard to <a href="#">request feedback, define sprints, and modify your team's area paths</a>.</p> <div data-bbox="599 1724 985 1904"><p>Other links</p><ul style="list-style-type: none"><li>Request feedback</li><li>Configure schedules and iterations</li><li>Configure work areas</li></ul></div> <p><b>Query tile</b></p> <p>Configurable tile to display the <a href="#">results and link to a shared query</a>.</p>	Date	1	2	3	Total	1/23/2016	10	25	10	45	1/24/2016	10	25	10	45	1/25/2016	10	25	10	45	1/26/2016	10	25	10	45	1/27/2016	10	25	10	45	1/28/2016	10	25	10	45	1/29/2016	10	25	10	45	<p><b>Plan and track work (continued)</b></p> <p><b>Sprint burndown</b></p> <p>Adds a <a href="#">burndown chart</a> for tracking a team's Scrum progress for the current sprint.</p> <p><b>Sprint capacity</b></p> <p>Adds a <a href="#">chart for tracking remaining capacity</a> when tracking a team's Scrum progress for the current sprint.</p> <div data-bbox="1022 548 1366 795"><p>Sprint 91 October 19 - November 6 Total story count: 40% completed</p><table border="1"><thead><tr><th>Work days remaining</th><th>Planned</th><th>Completed</th><th>Remaining</th></tr></thead><tbody><tr><td>8</td><td>42</td><td>21</td><td>41</td></tr></tbody></table><p>8 Work days remaining</p></div> <p><b>Sprint overview</b></p> <p>Displays a visual overview of the <a href="#">current sprint progress</a> for tracking a team's Scrum progress for the current sprint, indicating the number of backlog items in progress, completed, or not started.</p> <p><b>Work links</b></p> <p>Provides quick access links from a team dashboard to open the <a href="#">team backlog, Kanban board, task board, and queries</a>.</p> <p><b>Build and test widgets</b></p> <p><b>Chart for build history</b></p> <p>Configurable tile to display the <a href="#">histogram for a specific build definition</a>.</p> <p><b>Deployment status (Azure DevOps Services)</b></p> <p>Configurable tile that shows you a consolidated view of the deployment status and test pass rate across multiple environments for a recent set of builds.</p> <p><b>Release definition overview</b></p> <p>Configurable tile to view and track the status of a release definition. The widget <a href="#">shows the release as a series of environments, with the name of the release and the date or time it was started</a>.</p> <p><b>Test trend results</b></p> <p>Provides <a href="#">trend of test results</a>, such</p>	Work days remaining	Planned	Completed	Remaining	8	42	21	41
Date	1	2	3	Total																																														
1/23/2016	10	25	10	45																																														
1/24/2016	10	25	10	45																																														
1/25/2016	10	25	10	45																																														
1/26/2016	10	25	10	45																																														
1/27/2016	10	25	10	45																																														
1/28/2016	10	25	10	45																																														
1/29/2016	10	25	10	45																																														
Work days remaining	Planned	Completed	Remaining																																															
8	42	21	41																																															

	Manage Work <a href="#">Add work to your board</a>
	Collaborate on code <a href="#">Add code to your repository</a>
	Continuously integrate <a href="#">Automate your builds</a>
	Visualize progress <a href="#">Learn how to add charts</a>

#### Code widgets

##### Code tile

Configurable tile to display status and links to a Git or TFVC code repository, branch, or folder.

##### Pull request

Adds a configurable tile to display active pull requests requested by the team, or assigned to or requested by the person logged in. You select the Git repository for the pull requests of interest.

Pull Request in Fabrikam (2)

	Updated ProjectController.cs Dan Paul into <a href="#">`master`</a> features/VirtualParameters, created 13 minutes ago
	Fixed layout issues, bug #8730 John Smith into <a href="#">`master`</a> , created 13 minutes ago

Backlogs Bugs  
42  
Work items

##### Query results

Adds a configurable [query results](#) list to a team dashboard.

##### Requirements quality

Displays a configurable widget that you can use to [track quality continuously from a build or release definition](#).

as passed or failed tests, for a selected build definition.



Test results trend  
Shows the trend of test results for a selected build definition.

#### Extensibility

##### Marketplace widgets

You can find additional widgets by browsing the [Marketplace](#)

##### Dashboard widget SDK

[Create a dashboard widget](#) using the REST API service.

## Extensibility

### Marketplace

**Feature availability:** You can add Marketplace extensions from the web portal for Azure DevOps or for Visual Studio or Visual Studio Code.

## What is the Marketplace?

From the [Marketplace](#), you can extend the functionality available to you by installing free extensions or purchasing a subscription or paid extension. Extensions support adding new capabilities to Visual Studio, Visual Studio Code, and Azure DevOps.

### Featured



#### Exploratory Testing

Microsoft

Explore your app, find and submit bugs directly from your browser

[PREVIEW](#)



#### Test Manager

Microsoft

Integrated test management system for all your manual, exploratory and user

[PAID](#)

## Subscriptions

[Visual Studio subscriptions](#) are a way for you to get the Visual Studio IDE, team collaboration benefits like Azure DevOps, and subscriber benefits like dev/test use of Windows, Windows Server, and SQL Server.

### Extensions

You can [get and quickly install extensions](#) to add functionality to Visual Studio, Visual Studio Code, or Azure DevOps Services.

#### Try Azure Test Plans for free

You can [start a trial for Azure Test Plans for free](#).

#### Get extensions for...

- [Azure DevOps Services](#)
- [Visual Studio](#)
- [Visual Studio Code](#)

### Get cloud subscriptions

Buy [cloud subscriptions](#) in the Marketplace.

## REST APIs

### Get started with REST APIs

Learn the basic patterns for [using the REST APIs](#) for Azure DevOps.

### Authorization

Get authorization from your customers to access Azure DevOps Services resources using [OAuth 2.0](#).

### REST API reference

Use the [REST APIs](#) to work with Azure DevOps resources.

### .NET client libraries

For .NET developers building Windows apps and services that integrate with Azure DevOps, [client libraries](#) are available for integrating with work item tracking, version control, build, and other services are now available. These packages replace the traditional TFS Client OM installer and make it easy to acquire and redistribute the libraries needed by your app or service.

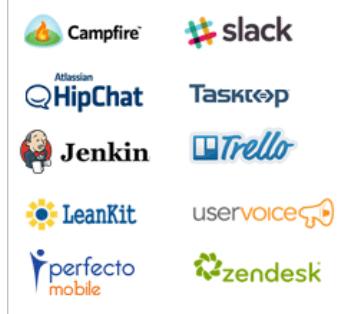
### REST API samples

Here are a number of [samples](#) that work with the REST APIs directly.

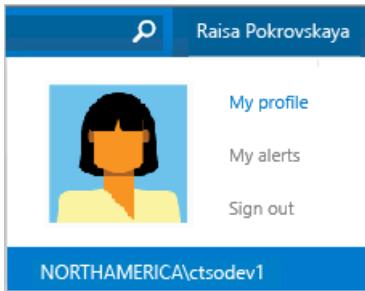
### C# client library samples

Here are a few quick [samples](#) to help you get started with the client libraries.

## Service hooks

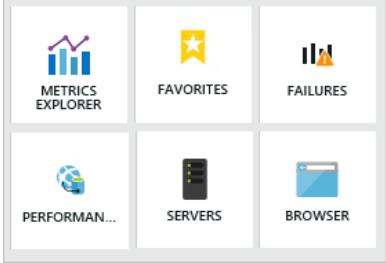
<p><b>Integrate with service hooks</b></p> <p>Service hooks enable you to perform tasks on other services when events happen in your Azure DevOps Projects</p> <p><b>Create integrations</b></p> <p>Integrate other services like <a href="#">HipChat</a>, <a href="#">Slack</a>, and <a href="#">UserVoice</a> with Azure DevOps using <a href="#">service hooks</a>.</p>		<p><b>Authorize</b></p> <p>Authorize other services to access your organization using the industry standard OAuth 2.0. OAuth 2.0 provides safe, secure access to your resources like work items, source code and build results by those other services.</p>
--	---	---

## Global

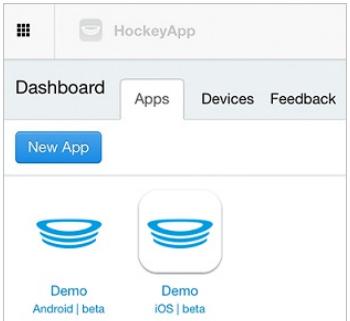
<p><b>Web portal preferences</b></p> <p>Choose your name to access <a href="#">your profile settings</a> and set your web portal preferences which include language (currently only English is supported for Azure DevOps), date and time pattern, and time zone.</p>  <p><b>Language Interface Packs (LIPs)</b></p> <p>By using a <a href="#">Windows Language Interface Pack (LIP)</a>, you can install a language version of Windows, and then install various User Interface Language Packs. Language packs switch your English Visual Studio Professional user interface into any of these languages and have a majority of the user interface localized.</p>	<p><b>Localized content</b></p> <p>Most content that supports Azure DevOps is localized into the following 14 languages.</p> <ul style="list-style-type: none"> <li>• English</li> <li>• Chinese Simplified</li> <li>• Chinese Traditional</li> <li>• Czech</li> <li>• German</li> <li>• French</li> <li>• Italian</li> <li>• Japanese</li> <li>• Korean</li> <li>• Polish</li> <li>• Portuguese (Brazil)</li> <li>• Russian</li> <li>• Spanish</li> <li>• Turkish</li> </ul> <div data-bbox="595 1572 968 1628" style="border: 1px solid #ccc; padding: 5px; margin-top: 20px;"> <p>Currently, the visualstudio.com content is only available in English.</p> </div>	<p><b>Visual Studio language pack</b></p> <p>Install the <a href="#">language pack</a> to switch the UI display to different languages. Visual Studio provides localized UI support for these 14 languages.</p> <ul style="list-style-type: none"> <li>• English</li> <li>• Chinese Simplified</li> <li>• Chinese Traditional</li> <li>• Czech</li> <li>• German</li> <li>• French</li> <li>• Italian</li> <li>• Japanese</li> <li>• Korean</li> <li>• Polish</li> <li>• Portuguese (Brazil)</li> <li>• Russian</li> <li>• Spanish</li> <li>• Turkish</li> </ul> <p><b>Eclipse plug-in language support</b></p> <p><a href="#">Install Team Explorer Everywhere</a>, which includes language support for English, French, German, Japanese, and Simplified Chinese.</p>
---	---	---

## Monitor

### Application Insights (Preview)

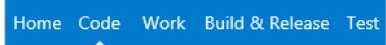
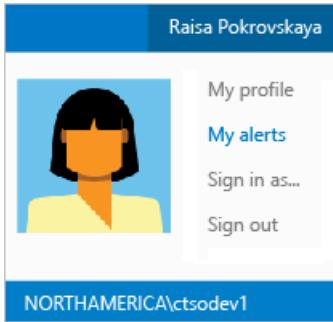
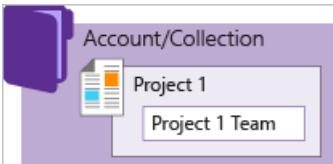
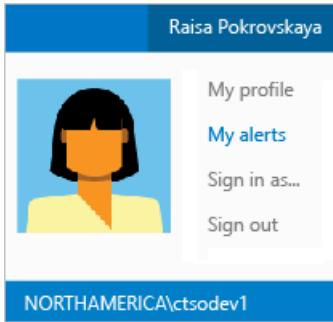
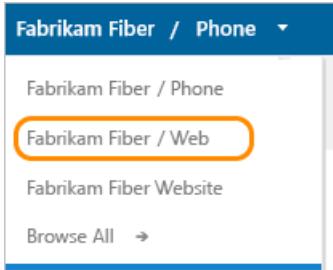
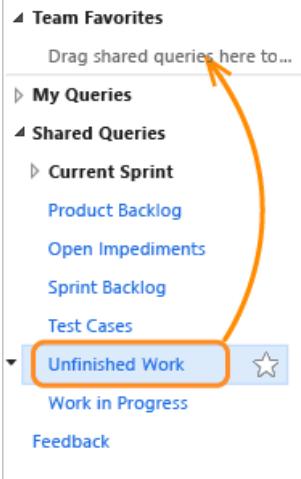
<p><b>What is Application Insights</b></p> <p>Application Insights, an extensible analytics service that monitors your live web application, supports developers to continuously improve the performance and usability of apps. With it you can <b>detect and diagnose performance issues, and understand what users actually do with your app.</b></p> <p><b>Web site availability monitoring</b></p> <p>Know when your site or service goes down by <b>setting up tests and performance thresholds to monitor both uptime and responsiveness.</b></p> <p><b>Web site performance &amp; usage</b></p> <p>Open the Performance blade to see <b>request, response time, dependency and other data.</b></p> <p><b>Power BI integration</b></p> <p>Get even more flexible views of <b>your telemetry</b>, and present your web app telemetry alongside data from devices and other business sources.</p>	<p><b>Dashboard</b></p> <p>Get the full picture with <b>customizable dashboards that track application health alongside usage metrics and app crashes.</b> Within the dashboard, you can filter, search, and drill down to an event instance for more detail or to segment data.</p>  <p><b>Diagnose failures and exceptions</b></p> <p>Quickly diagnose causes and <b>correlate failed requests with exceptions and other events at both the client and server.</b></p>	<p><b>Usage analysis</b></p> <p>Gain a clear view of where your <b>users are coming from and how they use your app.</b> Add custom instrumentation to determine usage patterns and next version investment areas.</p> <p><b>Diagnose dependency issues</b></p> <p>See how long your application <b>waits for dependencies and how often a dependency call fails.</b> Dependencies are external components that your app calls such as an HTTP service, database, or file system.</p> <p><b>Custom data collectors</b></p> <p>Add custom data collectors to your app using the <b>Application Insights API</b> to customize your telemetry data.</p> <p><b>Continuous data export</b></p> <p>Perform custom analysis on your telemetry through <b>continuous export of your data.</b></p>
---	---	--

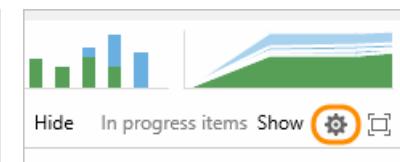
## HockeyApp

<p><b>Get HockeyApp for mobile app development</b></p> <p>Distribute mobile apps for testing, collect user metrics and feedback, and respond to crashes more easily by <b>adding HockeyApp to your Agile, continuous integration, and continuous delivery workflows.</b></p> <p><b>Simplified distribution</b></p> <p>Manage distribution of development and production versions of your apps and use independent bundle identifiers that can run in parallel on the same device.</p> <p><b>Integrate with Azure DevOps</b></p> <p>Integrate HockeyApp directly in <b>Azure DevOps</b> to upload your Android, iOS, or Windows builds.</p>	<p><b>Comprehensive dashboard</b></p> <p>Manage all your apps, users, and devices from a single dashboard. Monitor crashes and feedback as well. As an admin, you'll have full control over which user can see and install which app.</p> 	<p><b>Invite or recruit testers</b></p> <p>Invite beta testers and distribute your beta versions through the dashboard.</p> <p><b>Usage</b></p> <p>Get advanced metrics to understand the testing performed on your app. See which devices were tested, which testers used the app for how long, and which language was tested.</p> <p><b>Crash reports</b></p> <p>Get the information you need to analyze and respond to crashes by getting <b>symbolicated stack traces and environment details.</b></p> <p><b>Webhooks</b></p> <p>Use webhooks to receive notifications about new versions, crash groups, and feedback.</p>
--	---	--

# Navigation

## Web portal

<p><b>Operational hubs</b></p> <p>Each hub—<a href="#">Home</a>, <a href="#">Code</a>, <a href="#">Work</a>, <a href="#">build</a>, and <a href="#">Test</a>—supports specialized functions to share information, view and create dashboards, collaborate on code, plan and track work, build and test your applications, plus much, much more.</p> 	<p><b>Home</b></p> <p>Provide team guidance through <a href="#">Welcome</a> (Markdown format) pages and add team <a href="#">dashboards</a> to monitor progress and trends.</p> <p><b>Code</b></p> <p>Manage source code using distributed <a href="#">Git repositories</a> or <a href="#">Team Foundation version control</a>.</p> <p><b>Work</b></p> <p>Plan and track work by <a href="#">creating a product backlog</a>, and managing work using <a href="#">Kanban</a> or <a href="#">Scrum</a> processes. Find work items you want to review or update by <a href="#">creating queries</a>, or visualize progress by <a href="#">creating query-based charts</a></p> <p><b>Build</b></p> <p><a href="#">Define and monitor builds</a> and set up continuous builds to improve the quality of your app.</p> <p><b>Test</b></p> <p><a href="#">Create and run manual tests</a> for your app.</p> <p><b>Package (Azure DevOps Services, Preview)</b></p> <p>Share code as binary assets and control dependencies by <a href="#">subscribing to and working with Azure Artifacts feeds</a>.</p> <p><b>Release (Azure DevOps Services, Preview)</b></p> <p>Manage the release of your app by <a href="#">deploying it to a specific environment for each separate release step</a>, and by controlling the process through approvals for each step.</p> <p><b>Code search</b></p> <p>Search within your code branches (<a href="#">TFVC</a>) and repositories (<a href="#">Git</a>) to find files, commits, and more using powerful filters to obtain rich results.</p> 	<p><b>Collection-project-team structure</b></p> <p>The <a href="#">collection-project-team structure</a> provides teams a high-level of autonomy to configure their tools in ways that work for them. It also supports administrative tasks to occur at the appropriate level.</p> 
<p><b>Project page</b></p> <p>To view and quickly go to teams, team projects, branches, work items, pull requests and other objects that are relevant to you, use your <a href="#">Project page</a>.</p> <p><b>Your profile and preferences</b></p> <p>Choose your name to access <a href="#">your profile settings</a>, set preferences, create personal access tokens (<a href="#">Azure DevOps Services</a>), set alerts, and log-in or out.</p>  <p><b>Switch team context</b></p> <p>Go to a different team or project from the top row.</p>  <p><b>Change team settings</b></p> <p>Customize features to meet your team needs by <a href="#">configuring your team assets</a>.</p>	<p><b>My favorites</b></p> <p>From any context, you can drag folders, queries, or builds to <a href="#">My favorites</a> when working in the Code, Work, or Build hubs to provide quick access to those items.</p> <p><b>Team favorites</b></p> <p>From your team context, drag shared queries, builds, and folders to <a href="#">Team favorites</a> to provide quick access to those items.</p>  <p><b>Project admin context</b></p> <p>Open the admin context to <a href="#">add teams</a> and <a href="#">manage permissions</a>. From any project hub, choose  to open the admin context.</p>  <p><b>Project collection admin context</b></p>	



#### Keyboard shortcuts

Increase your productivity by working with [hot keys](#) and [shortcuts](#).

#### Find work items

When in the Work hub, [enter IDs or keywords to start a query](#) to find work items that you want to review, triage, or update.

🔍

From the collection admin context, you can [manage collection-level permissions](#), and set build policies, and [manage extensions](#). Choose to open the admin context, and then choose DefaultCollection.

## Search, queries, and filters

#### Quick work item search

Find work items based on [ID](#), [assignment](#), [changed date](#), or [keyword](#).

🔍

#### Code search

Find code based on [keywords](#) and [semantic search filters](#) across your Git repositories.

🔍

#### CodeLens search

Find references and changes to your code, linked bugs, work items, code reviews, and unit tests.

#### Work item queries

Open shared queries or create your own query using the query editor [to list work items or show hierarchical or dependent items](#).

#### >Manage risks and dependencies

Link work items to [track related work, dependencies, and changes made over time](#).

#### History & auditing

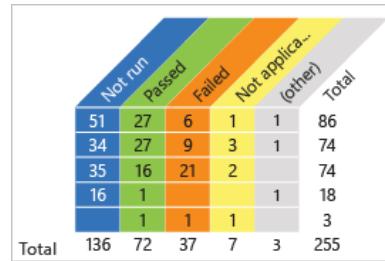
Review and query [work item change history](#) to learn of past decisions and support future ones.

#### Bulk add or modify using Excel

[Bulk add items to track or modify multiple field values](#) using Excel.

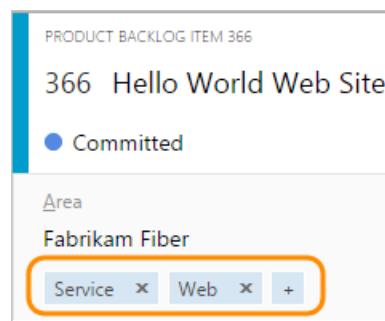
#### Charts

Turn your [queries](#) into a [status](#) or [trend chart](#) and share them with your team, organization, and Stakeholders.



#### Tags

Add [tags](#) to work items to filter backlogs and queries. Bulk update work items to add or remove tags: [Azure DevOps Services | Azure DevOps Server](#).



<p

#### Bulk modify

Edit or update multiple work items from any backlog or query result. Supported tasks include:

- Modify field values
- Add or remove tags
- Reassign
- Move to an iteration
- Delete
- Link to a new or existing work item
- Change work item type
- Move to another project
- Create a new Git branch

#### Query by date or current iteration

List work items based on [when changes occurred](#) or if they belong to the team's current sprint.

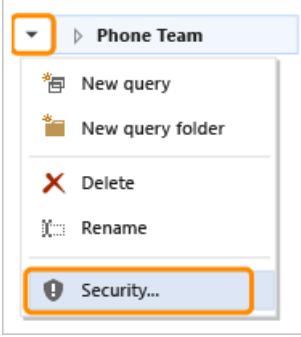
#### Query by workflow

Find and list work items based on their [current state](#), such as new, in progress, resolved, done, or closed.

#### Query by Kanban board change

Track status and trends of work items based on [changes made to the Kanban board](#).

# Security

<p><b>Manage users and groups</b></p> <p>Add users to built-in groups to grant them access to your project. Optionally, create groups to customize access based on your business requirements.</p> <p><b>Permission states</b></p> <p>Understand how <a href="#">Allow</a>, <a href="#">Deny</a>, <a href="#">Not set</a> and other permissions states control access to features and objects.</p> <table border="1" data-bbox="171 608 557 961"><thead><tr><th>Permissions</th><th>Members</th><th>Member of</th></tr></thead><tbody><tr><td>Members of this group can add, modify, and delete items within the team project.</td><td></td><td></td></tr><tr><td>Create tag definition</td><td>Inherited allow</td><td></td></tr><tr><td>Create test runs</td><td>Allow</td><td></td></tr><tr><td>Delete team project</td><td>Deny</td><td></td></tr><tr><td>Delete test runs</td><td>Allow</td><td></td></tr><tr><td>Edit project-level information</td><td>Not set</td><td></td></tr><tr><td>Manage test configurations</td><td>Allow</td><td></td></tr></tbody></table> <p><b>Manage work access (Azure DevOps Services)</b></p> <p>Control user access with a <a href="#">directory</a> to enforce policies about accessing company resources.</p> <p><b>Azure Active Directory (Azure DevOps Services)</b></p> <p>Easily control access to your team's critical resources and key business assets with <a href="#">Azure Active Directory groups</a>.</p> <p><b>Set up groups (Azure DevOps Server)</b></p> <p>Create Windows or Active Directory groups to manage access to your team projects and collections.</p> <p><b>Built-in groups</b></p> <p>Understand the <a href="#">permissions granted to built-in groups</a> and use them to manage access to your team projects and collections.</p>	Permissions	Members	Member of	Members of this group can add, modify, and delete items within the team project.			Create tag definition	Inherited allow		Create test runs	Allow		Delete team project	Deny		Delete test runs	Allow		Edit project-level information	Not set		Manage test configurations	Allow		<p><b>DevOps permissions</b></p> <p>Grant or restrict access to:</p> <ul style="list-style-type: none"><li>• <a href="#">Git repositories</a></li><li>• <a href="#">Git branches</a></li><li>• <a href="#">TFVC source code and folders</a></li><li>• <a href="#">Build</a></li><li>• <a href="#">Test</a>)</li><li>• <a href="#">Release</a></li></ul> <p><b>Work item tracking permissions</b></p> <p>Control access to specific features by setting permissions for a user or group.</p> <ul style="list-style-type: none"><li>• <a href="#">Area and iteration paths</a></li><li>• <a href="#">Query permissions</a></li><li>• <a href="#">Work item tags</a></li><li>• <a href="#">Move work items to another project</a></li><li>• <a href="#">Permanently delete work items</a></li><li>• <a href="#">Provide feedback through the Microsoft Feedback client</a></li></ul> <p><b>Team admin role and permissions</b></p> <p>Add users as team administrators to enable them to <a href="#">configure team settings and manage team assets</a>.</p> <p><b>Manage administrative permissions</b></p> <p>Add users to one of the following built-in groups to provide them permissions assigned to that group:</p> <ul style="list-style-type: none"><li>• <a href="#">Project Administrators</a>, who manage shared features for a project</li><li>• <a href="#">Project Collection Administrators</a>, who manage collection-level features</li><li>• <a href="#">Azure DevOps Server Administrators</a>, who manage on-premises application servers</li></ul> <p><b>Restrict access</b></p> <p>You can restrict access to several features and tasks by setting the permission state to Deny to individual users or a security group.</p>	<p><b>Stakeholder access</b></p> <p>Grant Stakeholders, non-licensed users, limited access to contribute ideas and access team dashboards.</p> <p><b>Query permissions</b></p> <p>Grant permissions to <a href="#">create shared queries and query folders</a>.</p>  <p><b>Process permissions</b></p> <p>To customize a process, add custom fields, or change the layout of a work item form, you must be a member of the Project Collection Administrators group or be <a href="#">granted explicit permissions to edit a specific process</a>.</p> <p><b>Valid users</b></p> <p>Understand how <a href="#">valid user groups are populated</a> and the permissions they're granted.</p> <p><b>Permission reference</b></p> <p>Provide or restrict access for practically any feature, function, or object at the collection or project level.</p> <p><b>SharePoint permissions (Azure DevOps Server)</b></p> <p>Grant permissions to <a href="#">view and contribute to SharePoint project portals</a>.</p> <p><b>SQL Server reporting permissions (Azure DevOps Server)</b></p> <p>Grant permissions to <a href="#">view and author Excel and SQL Server reports</a>.</p>
Permissions	Members	Member of																								
Members of this group can add, modify, and delete items within the team project.																										
Create tag definition	Inherited allow																									
Create test runs	Allow																									
Delete team project	Deny																									
Delete test runs	Allow																									
Edit project-level information	Not set																									
Manage test configurations	Allow																									

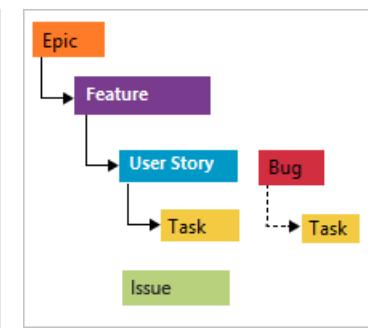
# Set up and installation

<p><b>Free developer offers</b></p> <p>To get started, <a href="#">download and install Visual Studio</a> an integrated development environment (IDE) that works with Azure DevOps.</p> <p><b>Migrate from on-premises to hosted</b></p> <p>You can <a href="#">migrate source code and work items</a> from an on-premises Azure DevOps Server to the cloud.</p>	<p><b>Sign up for Azure DevOps Services</b></p> <p>Store your code, tests, and test results in the cloud with <a href="#">Azure DevOps Services</a>, as well as plan your project and track progress.</p> <p><b>Install Azure DevOps Server</b></p> <p>Download and install the latest version of <a href="#">Azure DevOps Server</a>. Azure DevOps Server provides the collaboration hub to support your teams DevOps tasks. at the center of the Microsoft devops solution.</p>	<p><b>Email configuration (Azure DevOps Server)</b></p> <p>For feedback requests, alerts, and other special controls to work, you must <a href="#">configure an SMTP server</a> for your on-premises Azure DevOps.</p> <p><b>Automated, scheduled backups (Azure DevOps Server)</b></p> <p>Reduce the risk of lost data by <a href="#">scheduling automated backups of the data store</a>.</p> <p><b>Built-in SQL Server database (Azure DevOps Server)</b></p> <p>For small teams, you can install <a href="#">Azure DevOps Server using SQL Server Express</a> which installs with Azure DevOps Server.</p>
--	---	---

# Teams, team projects, and processes

## Processes and process guidance

<p><b>What is a process?</b></p> <p>A <a href="#">process defines the building blocks</a> of the work item tracking system as well as other subsystems you access through your project.</p> <p><b>Compare and choose a process</b></p> <p>Compare the three core system processes--<a href="#">Agile</a>, <a href="#">Scrum</a>, <a href="#">CMMI</a>--before you choose one to create a project.</p> <p><b>Agile process</b></p> <p>Choose <a href="#">Agile</a> when your team uses Agile planning methods, including Scrum, and tracks development and test activities separately. With Agile, you can track user stories and bugs on the Kanban board, or track bugs and tasks on the task board.</p>	<p><b>Kanban process tools</b></p> <p>You can use the Kanban board with any process--Agile, Scrum, CMMI--or project that you select or create. Agile Kanban tools support working with the <a href="#">Kanban board</a>, adding task checklists, setting WIP limits, custom columns, split columns, custom swimlanes, and <a href="#">customizing cards</a>.</p> <p><b>Scrum process</b></p> <p>Choose <a href="#">Scrum</a> when your team practices Scrum and you want to track product backlog items (PBIs) and bugs on the Kanban board, or break PBIs and bugs down into tasks on the task board.</p> <pre>graph TD; Epic[Epics] --&gt; Feature[Features]; Feature --&gt; PBI[Product backlog item]; PBI --&gt; Task[Task]; Bug[Bug] --- PBI; Impediment[Impediment]</pre> <p><b>Scrum work items and workflow process guidance</b></p>	<p><b>Scrum process tools</b></p> <p>Scrum processes can be used with any process--Agile, Scrum, CMMI--or project that you select or create. Agile Scrum tools support <a href="#">sprint planning</a>, <a href="#">capacity planning</a>, <a href="#">task boards</a>, and <a href="#">burndown charts</a>.</p> <p><b>Manage processes (Azure DevOps Services)</b></p> <p>Add users to <a href="#">built-in groups</a> to grant them access to your project. Optionally, create groups to customize access based on your business requirements.</p> <p><b>CMMI process</b></p> <p>Choose <a href="#">CMMI</a> when your team follows more formal project methods that require a framework for process improvement and an auditable record of decisions. CMMI supports tracking requirements, change requests, risks, and reviews.</p>
---	--	--



Plan and track your work using the [work item types and workflow supported by the Scrum process](#).

#### Agile work items and workflow process guidance

Plan and track your work using the [work item types and workflow supported by the Agile process](#).

#### Work item field index

For descriptions and usage of each field used by the core and inherited processes, see [Work item field index](#).

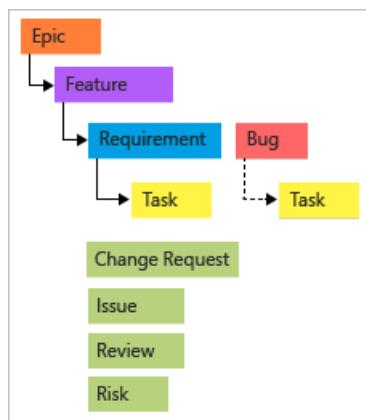
#### Customize a process (Azure DevOps Services)

Customizations you make to an inherited process automatically update all team projects that reference that process. You can customize your project as follows:

- [Add and modify fields](#)
- [Modify the web form layout](#)
- [Modify the workflow states](#)
- [Add a custom work item type](#)

#### Manage processes (Azure DevOps Services)

Create [inherited processes](#) and [migrate team projects to use them](#). Set the default process and enable, disable, or delete processes you no longer want to use.



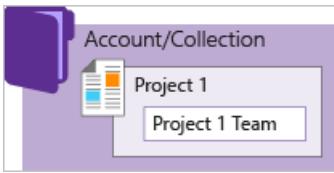
#### CMMI work items and workflow process guidance

Plan and track your work using the [work item types and workflow supported by the CMMI process](#).

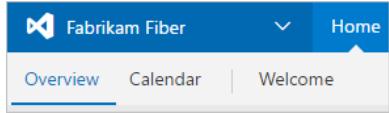
## Process templates (Azure DevOps Server)

What is a process template?	Process template files	Changes made to process templates
<p>A process template is the forerunner and on-premises version of a process. It provides the building blocks of the work item tracking system as well as other sub-systems you access through your project. Process templates support full <a href="#">customization of all its objects</a>.</p> <p><b>Manage process templates</b></p> <p><a href="#">Download and upload process templates</a> to support customization and upgrade of your work tracking experience and team projects.</p>	<p><b>Process template files</b></p> <p>You customize the initial configuration of team projects by <a href="#">customizing one or more process template files</a>. By customizing these files, you can define the initial configuration of all team projects that are created from the process template.</p> <p><b>Configure Features Wizard</b></p> <p>Use the Configure Features Wizard to <a href="#">configure team projects after an Azure DevOps Server upgrade</a> to access new features.</p>	<p><b>Changes made to process templates</b></p> <p>For a catalog of changes, see <a href="#">Changes made to process templates</a>.</p> <p><b>Customize the Microsoft Project field mapping file</b></p> <p>You can <a href="#">customize how work item fields</a> that are defined in Team Foundation map to fields in Microsoft Project. And, you can change how specific fields are published.</p>

## Team projects

<p><b>What is a project?</b></p> <p>A <a href="#">project</a> provides a repository for source code and a place for a group of developers to plan, track progress, and collaborate on building software solutions. A project lives within a project collection. You can grant permissions to and customize a project to support your business needs.</p> <p><b>Create a project</b></p> <p>You can <a href="#">create a project hosted in the cloud (Azure DevOps Services)</a>, avoiding maintenance and administrative overhead, or <a href="#">create a project on an on-premises Azure DevOps Server</a>.</p> <p><b>Rename a project</b></p> <p><a href="#">Rename a project</a> as needed to reflect changes that occur within your org.</p> <p><b>Delete a project</b></p> <p>Simplify the navigation to team projects that are in use by <a href="#">deleting team projects you no longer use</a>.</p>	<p><b>Collection-project-team structure</b></p> <p>The <a href="#">collection-project-team structure</a> provides teams a high-level of autonomy to configure their tools in ways that work for them. It also supports administrative tasks to occur at the appropriate level.</p>  <p><b>Change the process (Azure DevOps Services)</b></p> <p>You <a href="#">change the process of a project</a> to apply customizations you've made to an inherited process. You can <a href="#">add and modify fields and modify the layout of each work item type</a> defined for that process.</p>	<p><b>View your work across teams and team projects</b></p> <p>From your <a href="#">Project page</a>, you can view and quickly go to teams, team projects, branches, work items, pull requests and other objects that are relevant to you and that are stored in different team projects within the organization or collection.</p> <p><b>Customize a project (Azure DevOps Server)</b></p> <p>You customize a project defined on an on-premises Azure DevOps Server by <a href="#">modifying definition files for work item types or process configuration, or changing field attributes</a>.</p> <p><b>Update a project after an upgrade (Azure DevOps Server)</b></p> <p>Some features added when you upgrade your on-premises application server may require you to <a href="#">configure features to access them</a>.</p> <p><b>Upload reports (Azure DevOps Server)</b></p> <p><a href="#">Upload the latest reports provided for your process</a> or add reports after you've already created a project by adding SQL Server Reporting Services.</p>
---	--	--

## Teams

<p><b>What is a team?</b></p> <p>A team is an organizing unit used to support a number of <a href="#">team-configurable tools</a> to plan and manage work and facilitate collaboration.</p> <p><b>Add team members</b></p> <p>Add organizations—<a href="#">Azure DevOps Services</a>   <a href="#">Azure DevOps Server</a>—to a team to enable users to share code, plan and track work, and access other team assets and resources.</p>  <p><b>Add a team</b></p> <p>As your organization grows, consider moving from your <a href="#">default team of one to two or more teams</a> to support feature-focused groups within your org.</p> <p><b>Add a team admin</b></p> <p>Add users to the team admin role to enable them to <a href="#">Manage teams and configure team tools</a>. Team settings can only be configured by a team or project admin.</p> <p><b>Support Stakeholders</b></p> <p>Members within your org who don't have a license or contribute to developing the code base <a href="#">can track project priorities and provide direction, feature ideas, and business alignment to a team</a>.</p>	<p><b>Team dashboards</b></p> <p>Share progress, status, and guidance with your team using configurable team dashboards.</p>  <p><b>Team welcome page</b></p> <p>Provide in-project guidance through the <a href="#">Welcome page</a> and <a href="#">other pages you format using Markdown</a>.</p> <p><b>Setup a team hierarchy</b></p> <p>By <a href="#">configuring your teams and backlogs into an hierarchical structure</a>, program owners can more easily track progress across teams, manage portfolios, and generate rollup data.</p> <p><b>Set team defaults</b></p> <p>Several Agile tools reference the team's default area path, iteration path, and activated sprints to automatically filter the set of work items they display. Understand how defaults are used (../organizations/settings/about-teams-and-settings.md).</p> <p><b>Select team sprints</b></p> <p><a href="#">Select your team's sprints</a> to gain access to sprint backlogs and task boards.</p> <table border="1" data-bbox="596 1372 985 1596"> <thead> <tr> <th colspan="3">+ Select iteration(s)   X Remove   New New child</th> </tr> <tr> <th>Iteration</th> <th>Start Date</th> <th>End Date</th> </tr> </thead> <tbody> <tr> <td>Fabrikam Fiber\Iteration 1</td> <td>10/3/2016</td> <td>10/21/2016</td> </tr> <tr> <td>Fabrikam Fiber\Iteration 2</td> <td>10/24/2016</td> <td>11/11/2016</td> </tr> <tr> <td>Fabrikam Fiber\Iteration 3</td> <td>11/14/2016</td> <td>12/2/2016</td> </tr> </tbody> </table>	+ Select iteration(s)   X Remove   New New child			Iteration	Start Date	End Date	Fabrikam Fiber\Iteration 1	10/3/2016	10/21/2016	Fabrikam Fiber\Iteration 2	10/24/2016	11/11/2016	Fabrikam Fiber\Iteration 3	11/14/2016	12/2/2016	<p><b>Configure team settings</b></p> <p>Configure, customize, and manage all <a href="#">team-related activities</a></p> <p><b>Team alerts</b></p> <p>As changes occur to work items, code reviews, source control files, and builds, your team can automatically <a href="#">receive email notifications for alerts</a> that you define.</p> <p><b>Team rooms</b></p> <p>Team rooms, like chat rooms, provide teams with a <a href="#">space to discuss work in progress, ask questions, share status, and clarify issues</a> that arise. Use team rooms to foster and capture communication among team members, both near and far.</p> <p><b>Team groups</b></p> <p>A <a href="#">team group is created</a> when you create a team. Use this group in queries or to set permissions for your team.</p>
+ Select iteration(s)   X Remove   New New child																	
Iteration	Start Date	End Date															
Fabrikam Fiber\Iteration 1	10/3/2016	10/21/2016															
Fabrikam Fiber\Iteration 2	10/24/2016	11/11/2016															
Fabrikam Fiber\Iteration 3	11/14/2016	12/2/2016															

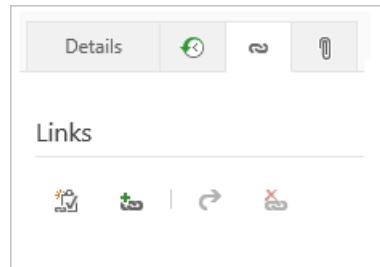
## Traceability

## Work item history & auditing

Review and query [work item change history](#) to learn of past decisions and support future ones.

## Manage risks and dependencies

Link work items to [track related work, dependencies, and changes made over time](#). Create queries based on link type to monitor dependencies.

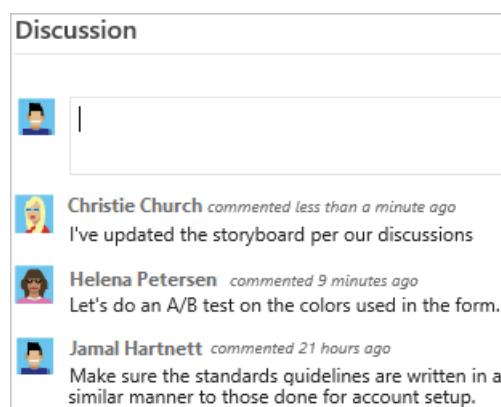


## Rich text comments

Describe and comment on work to perform using [formatted text, hyperlinks, and inline images](#).

## Discussion (Azure DevOps Services)

Add or review comments added to a work item. Start by choosing



## Storyboard

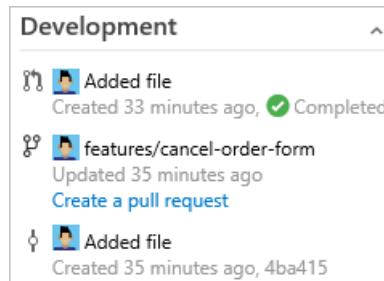
[Link your storyboards to your backlog work items](#).

## Git code changes

Get detailed information about what changes have been made to your local and centralized branches and [repositories](#), compare files and folders, review history of commits and file changes.

## Integrate Git development with work tracking (Azure DevOps Services)

Drive Git development and stay in sync as a team to complete backlog items and tasks using the [Git Development section](#). Add branches, create pull requests, and view all development performed to support the specific work item.



## TFVC code changes

Get detailed information about what changes have been made to your files, compare files and folders, view where and when changesets have been merged, and view file changes using [annotate](#).

## Build changes

Determine who [changed what in the build definition and when they did it](#).

## Release audit history

Retain full audit history of all activities performed on a release with detailed release logs and approval tracking.

## Release logs

View or download log files as zip files. Log files contain the status for each step or task of a release, for each of the environments in the release definition. Each completed release--succeeded, failed, or abandoned--[includes a live log file, details, and history for each step or task](#).

# Related articles

We add new features frequently. We'll work to keep this list up-to-date. Other resources you might want to bookmark:

- [Azure DevOps Services - Features update](#)
- [Azure DevOps Blog](#)

Get started today using our cloud offering, [Azure DevOps Services](#), or our [on-premises Azure DevOps Server](#).

# Web portal navigation in Azure DevOps

4/21/2021 • 6 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

The web portal for Azure DevOps is organized around a set of services, as well as administrative pages and several task-specific features such as the search box. The service labels differ depending on whether you work from Azure DevOps Services or Azure DevOps on-premises and its version.

## IMPORTANT

To view the content available for your platform, make sure that you select the correct version of this article from the version selector which is located above the table of contents. Feature support differs depending on whether you are working from Azure DevOps Services or an on-premises version of Azure DevOps Server, renamed from Team Foundation Server (TFS).

To learn which on-premises version you are using, see [What platform/version am I using?](#)

Each service provides you with one or more pages which support a number of features and functional tasks. Within a page, you may then have a choice of options to select a specific artifact or add an artifact.

The web portal for Azure DevOps Server is organized around a set of services—such as, **Overview**, **Boards**, **Repos**, **Pipelines**, **Test Plans**, and **Artifacts**—as well as administrative pages and several task-specific features such as the search box. Each service provides you with one or more pages which support a number of features and functional tasks. Within a page, you may then have a choice of options to select a specific artifact or add an artifact.

Each service provides you with one or more pages which support a number of features and functional tasks. Within a page, you may then have a choice of options to select a specific artifact or add an artifact.

The web portal for Team Foundation Server (TFS) is organized around a set of applications—such as, **Dashboards**, **Code**, **Work**, **Build and Release**—as well as administrative pages and several task-specific features such as the search box. Each service provides you with one or more pages which support a number of features and functional tasks. Within a page, you may then have a choice of options to select a specific artifact or add an artifact.

Here's what you need to know to get up and running using the web portal.

- **Open a service, page, or settings:** use to switch to a different [service or functional area](#)
- **Add an artifact or team:** use to quickly add a work item, Git repo, build or release pipelines, or a new team
- **Open another project or repo:** use to switch to a different project or access work items and pull requests defined in different projects, or items you've favorited
- **Open team artifacts, use breadcrumbs, selectors and directories:** use to navigate within a service, to open other artifacts or return to a root function
- **Work with favorites:** favorite artifacts to support quick navigation
- **Search box:** use to find code, work items, or wiki content
- **Your profile menu:** use to set personal preferences, notifications, and enable preview features
- **Settings:** use to add teams, manage security, and configure other project and organization-level resources.

- **Open a service, page, or settings:** use to switch to a different service or functional area
- **Add an artifact or team:** use to quickly add a work item, Git repo, build or release pipelines, or a new team
- **Open another project or repo, or switch to a different team:** use to switch to a different project or browse teams
- **Work across projects:** use to quickly open work assigned to you, your active pull requests, or items you've favorited
- **Open team artifacts, use breadcrumbs & selectors:** use to navigate within a service, to open other artifacts or return to a root function
- **Work with favorites:** favorite artifacts to support quick navigation
- **Search box:** use to find code, work items, or wiki content
- **Your profile menu:** use to set personal preferences, notifications, and enable preview features
- **Settings:** use to add teams, manage security, and configure other project and organization-level resources.

#### NOTE

Only those services that are enabled will appear in the user interface. For example, if **Boards** is disabled, then **Boards** or **Work** and all pages associated with that service won't appear. To enable or disable a service, see [Turn an Azure DevOps service on or off](#).

You select services—such as **Boards**, **Repos**, and **Pipelines**—from the sidebar and pages within those services.

The screenshot shows the Azure DevOps interface with the 'Dashboards' service selected in the sidebar. The main area displays a 'Web Overview' card for 'Assigned to me' with 56 work items, and a 'Work assigned to Jamal Hartnett (15)' card with 7 tasks, 4 backlog items, and 4 other items. The sidebar also lists 'Overview', 'Summary', 'Analytics views', 'Wiki backlogs', 'Boards', 'Queries', 'Repos', 'Pipelines', and 'Test Plans'.

You select a service—such as **Code**, **Work**, and **Build and Release**—from the horizontal bar and pages within those services.

All items

56 Work items

All bugs

0 Work items

Work assigned to Jamal Hartnett (15)

ID	State	Title
390	● Commit...	Cancel order form
492	● New	Build Settings Experience
375	● Commit...	Check service status
543	● To Do	Develop form
372	● To Do	Auto-save
539	● In Progr...	Standardize

Now that you have an understanding of how the user interface is structured, it's time to get started using it. As you can see, there are a lot of features and functionality.

If all you need is a code repository and bug tracking solution, then start with the [Get started with Git](#) and [Manage bugs](#).

To start planning and tracking work, see [About Agile tools](#).

## Connect to the web portal, user accounts and licensing

You connect to the web portal through a supported web browser—such as the latest versions of Edge, Chrome, Safari, or Firefox. Only users who have been [added to a project](#) can connect. This is typically done by the organization owner.

Five account users are free as are Visual Studio subscribers and stakeholders. After that, you need to [pay for more users](#). Find out more about licensing from [Azure DevOps pricing](#).

Limited access is available to an unlimited number of stakeholders for free. For details, see [Work as a Stakeholder](#).

You connect to the web portal through a supported web browser—such as the latest versions of Edge, Chrome, Safari, or Firefox. Only users who have been [added to a project](#) can connect. This is typically done by a member of the Project Administrators group.

Limited access is available to an unlimited number of stakeholders for free. For details, see [Work as a Stakeholder](#). Most regular contributors must have a TFS client access license (CAL). All Visual Studio subscriptions include a TFS CAL. Find out more about licensing from [TFS pricing](#).

## Refresh the web portal

If data doesn't appear as expected, the first thing to try is to refresh your web browser. Refreshing your client updates the local cache with changes that were made in another client or the server. To refresh the page or object you're currently viewing, refresh the page or choose the Refresh icon if available.

To avoid potential errors, you should refresh your client application under the following circumstances:

- Process changes are made
- Work item type definitions are added, removed, renamed or updated
- Area or iteration paths are added, removed, renamed or updated
- Users are added to or removed from security groups or permissions are updated
- A team member adds a new shared query or changes the name of a shared query
- A build definition is added or deleted
- A team or project is added or deleted

## Differences between the web portal and Visual Studio

Although you can access source code, work items, and builds from both clients, some task-specific tools are only supported in the web browser or an IDE, but not in both. Supported tasks differ depending on whether you connect to a Git or TFVC repository from Team Explorer.

### Web portal

### Visual Studio

- [Product backlog](#), [Portfolio backlogs](#), [Sprint backlogs](#), [Taskboards](#), [Capacity planning](#)
- [Kanban boards](#)
- [Dashboards](#), [Widgets](#), [Charts](#)
- [Request feedback](#)
- [Web-based Test Management](#)
- [Administration pages to administer accounts, team projects, and teams](#)
- Git: [Changes](#), [Branches](#), [Pull Requests](#), [Sync](#), [Work Items](#), [Builds](#)
- TFVC: [My Work](#), [Pending Changes](#) | [Source Control Explorer](#), [Work Items](#) | [Builds](#)
- Greater integration with work items and Office-integration clients. You can open a work item or query result in an office supported client.

#### NOTE

Visual Studio 2019 now includes a new Git tool that provides an improved experience when connecting to a Git repository. When you enable this tool, the Team Explorer tool is effectively disabled when connected to a Git repository. You can acquire the new tool by downloading [Visual Studio 2019 version 16.6](#). To enable and use the new tool, see [Git experience in Visual Studio \(Preview\)](#).

## Resources

- [Manage projects](#)
- [Project & Organizational Settings](#)

# Open a service, page, or settings

11/2/2020 • 4 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

The web portal for Azure DevOps provides support for software development teams to collaborate through the planning, development, and release cycles. You can manage source code, plan and track work, define builds, run tests, and manage releases.

This article shows you how to navigate to functional and administrative tasks available from the web portal. There are three levels of administrative tasks: team, project, and organization.

If you don't have a project yet, [create one](#). If you don't have access to the project, [get invited to the team](#).

This article shows you how to navigate to functional and administrative tasks available from the web portal. There are four levels of administrative tasks: team, project, collection, and server.

If you don't have a project yet, [create one](#). If you don't have access to the project, [get invited to the team](#).

## Open a service or functional task page

Services support getting work done—managing code, planning and tracking work, defining and managing pipelines, creating and running tests, and so on.

### NOTE

Only those services that are enabled will appear in the user interface. For example, if **Boards** is disabled, then **Boards** or **Work** and all pages associated with that service won't appear. To enable or disable a service, see [Turn an Azure DevOps service on or off](#).

You open a service by choosing the service from the sidebar and then selecting from the available pages.

For example, here we select **Boards**>**Backlogs**.



Order	Work item Type	Title
1	Product Backlog	> Hello World Web Site
2	Bug	> Slow response on information form
3	Product Backlog	> Change initial view
4	Product Backlog	> Interim save on long form
5	Bug	> Canadian addresses don't display correctly
6	Product Backlog	> Hello World Web Site
7	Product Backlog	> GSP locator interface
8	Product Backlog	> Request support

Within the page you may select a specific view or artifact, such as a team backlog or choose another page.

You open a service by choosing it from the horizontal blue bar. Then, select from the available pages.

For example, here we select **Work>Work Items**.

Fabrikam

Work

Work Items\*

## Open team settings

Select configurations are made to teams through the team settings pages. For an overview of all team settings, see [About user, team, project, and organization-level settings](#).

1. Choose **Project Settings**.

Summary - Overview   X

← → C https://dev.azure.com/fabrikam/FabrikamFiber

Azure DevOps   Fabrikam / FabrikamFiber / Overview / Summary

FabrikamFiber   Private   Invite

Overview   Summary   Dashboards   Wiki   Boards   Repos   Pipelines   Test Plans   Artifacts

Project settings

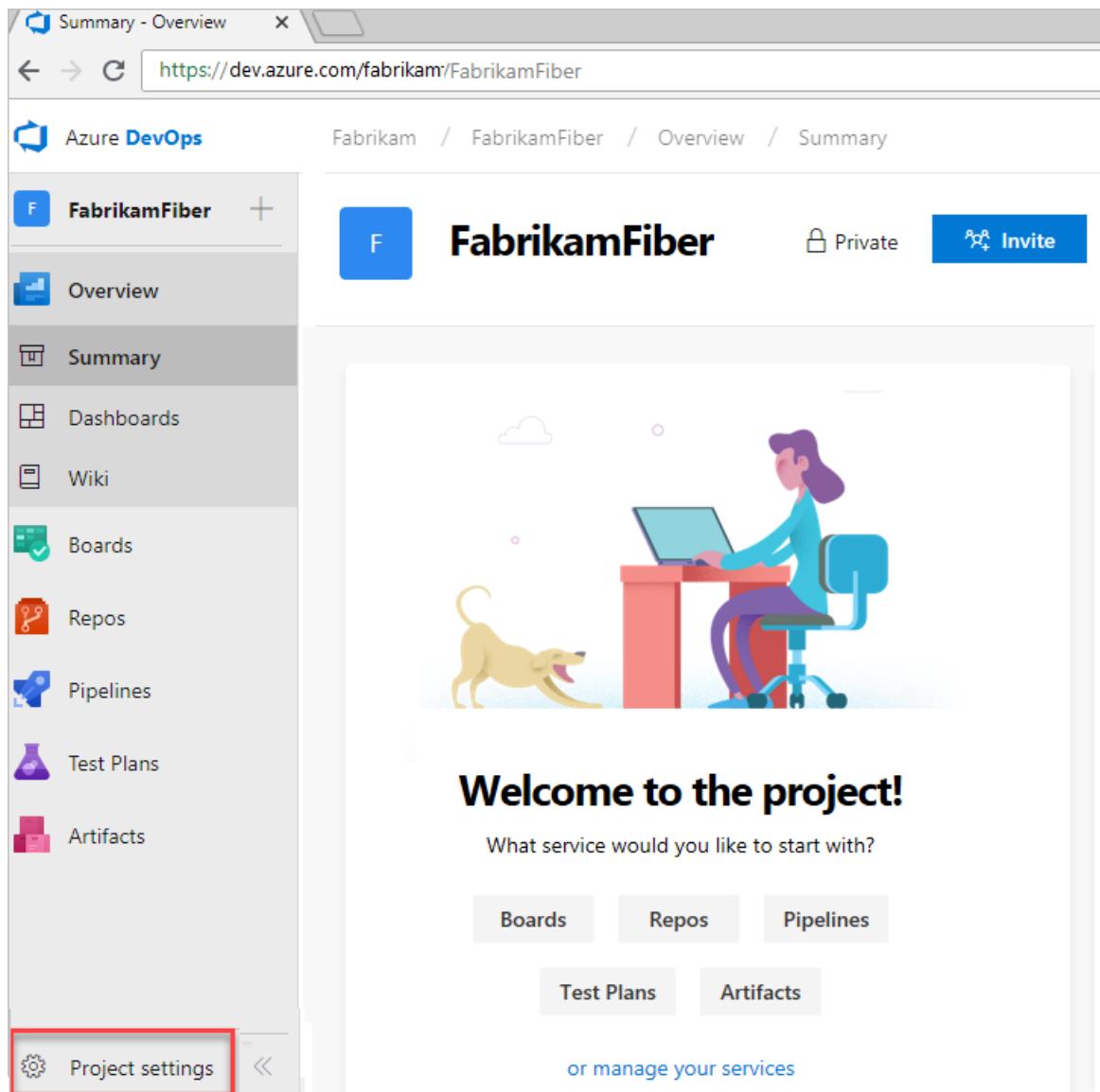


Welcome to the project!

What service would you like to start with?

Boards   Repos   Pipelines   Test Plans   Artifacts

or manage your services



2. Expand Boards and choose Team configuration.

**General**

Overview

Services

Teams

Security

Notifications

Service hooks

Dashboards

**Boards**

Project configuration

**Team configuration**

> Build and release

> Code

**Backlogs**

See only the backlogs your team manages.

**Backlog navigation levels**

- Epics
- Features
- Backlog items

**Working days**

Capacity and burndown are based on the days your team works.

**Select days**

- Monday
- Tuesday
- Wednesday
- Thursday
- Friday
- Saturday
- Sunday

3. Choose one of the pages **General**, **Iterations**, **Areas**, or **Templates** to configure settings for the team.  
To learn more, see [Manage teams](#).

4. If you need to switch to a different team, use the team selector within the breadcrumbs.

/ Settings / Teams / **Fabrikam Fiber Team** ▾

**Team Profile**

Name: **Fabrikam Fiber Team**

Description: The default project team.

**Customer Service (Fabrikam Fiber)**

**Fabrikam Fiber Team (Fabrikam F...)**

**+ Add...** |

Display Name	Username Or Scope
Jamal Hartnett	fabrikamfiber4@hotmail.com

5. To add a team administrator, add team members, or change the team profile, choose **Teams** from the vertical sidebar, and then choose the name of the team you want to configure.

You open team settings from the top navigation bar. Select the team you want and then choose the gear icon. To learn more about switching your team focus, see [Switch project, repository, team](#).

The screenshot shows the Azure DevOps interface for the 'Fabrikam Fiber' project. At the top, there's a navigation bar with links for Overview, Work (which is selected), Security, Version Control, Policies, Agent Queues, Notifications, and Service Hooks. Below this is a secondary navigation bar with General, Iterations, Areas, and Templates. The main content area is titled 'Work' and shows sections for General, Iterations, Areas, and Templates. A red box highlights the 'Project settings' link at the bottom left of the sidebar.

1. Choose one of the pages **General**, **Iterations**, **Areas**, or **Templates** to configure settings for the team. To learn more, see [Manage teams](#).

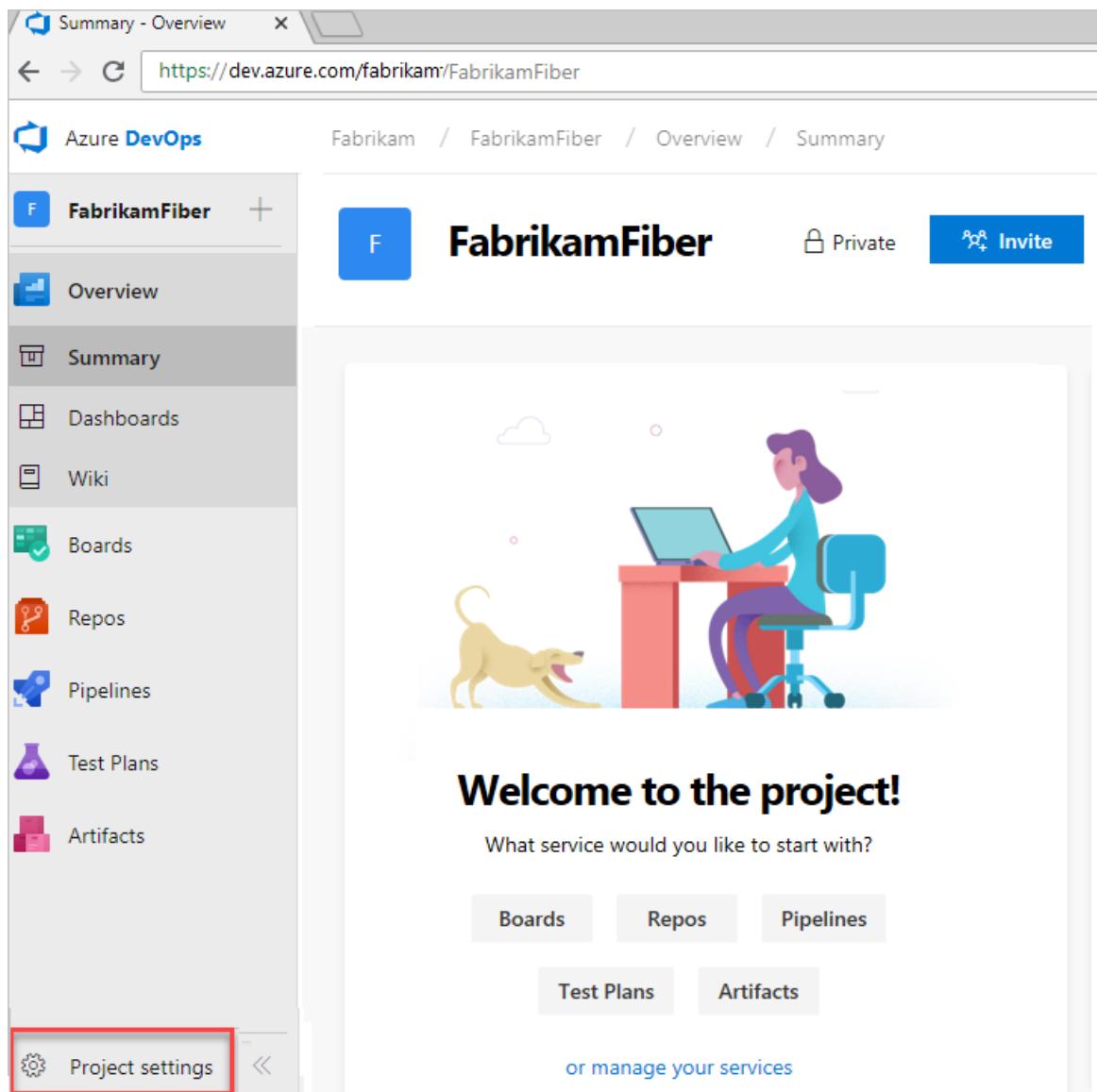
2. To add a team administrator, add team members, or change the team profile, choose **Overview**.

3.

## Open project settings

Administrators configure resources for a project and manage project-level permissions from the **Project settings** pages. Tasks performed in this context can impact the project and team functions. For an overview of all project settings, see [Project administrator role and managing projects](#).

1. Choose **Project Settings**.



2. From there, you can choose a page from the list. Settings are organized based on the service they

support. Expand or collapse the major sections such as **Boards**, **Build and release**, **Code**, **Test**, and **Extensions** to select from the list.

Project Settings

General

- Overview
- Services
- Teams
- Security
- Notifications
- Service hooks
- Dashboards

> Boards

> Build and release

> Code

> Test

> Extensions

Project details

Changes made here will affect all members and URLs associated with this project.

FF

Name

Fabrikam Fiber [Rename](#)

Description

Web, voice, and phone apps

Visibility

Private [Edit](#)

Process

MyScrum

From a user context, open **Project settings** by choosing the gear icon.

Open any admin page by choosing its name. Choose or hover over the gear icon to access other administrative options. Note that you can choose any of the user-context areas—**Dashboards**, **Code**, **Work**—to return to the user context.

The screenshot shows the Azure DevOps interface for a project named 'Fabrikam Fiber'. The top navigation bar includes links for Dashboards, Code, Work, Build and release, and a gear icon. Below the navigation is a secondary menu with options like Overview, Work, Security, Version Control, Policies, Agent Queues, Notifications, Service Hooks, Services, Test, Release, Dashboards, Project settings (which is highlighted with a red box), and Organization settings. The main content area displays a list of files within a folder structure.

Open any admin page by choosing its name. Choose or hover over the gear icon to access other administrative options. Note that you can choose any of the user-context areas—**Home** or **Dashboards**, **Code**, **Work**—to return to the user context.

## TFS 2017.2

The screenshot shows the TFS 2017.2 interface for a project named 'Fabrikam Fiber'. The top navigation bar includes links for Home, Code, Work, Build & Release, Test, and a gear icon. Below the navigation is a secondary menu with links for Overview, Work, Security, Version Control, Agent queues, Notifications, Service Hooks, Services, and Test. The 'Overview' link is highlighted with a red box.

## TFS 2017.1

The screenshot shows the TFS 2017.1 interface for a project named 'Fabrikam Fiber'. The top navigation bar includes links for Dashboards, Code, Work, Build & Release, Test, and a gear icon. Below the navigation is a secondary menu with links for Overview, Work, Security, Version Control, Agent Queues, Notifications, Service Hooks, Services, and Test. The 'Overview' link is highlighted with a red box.

## TFS 2017

The screenshot shows the TFS 2017 interface for a project named 'Fabrikam Fiber'. The top navigation bar includes links for Home, Code, Work, Build & Release, Test, and a gear icon. Below the navigation is a secondary menu with links for Overview, Work, Security, Alerts, Version Control, Agent Queues, Service Hooks, Services, and Test. The 'Overview' link is highlighted with a red box.

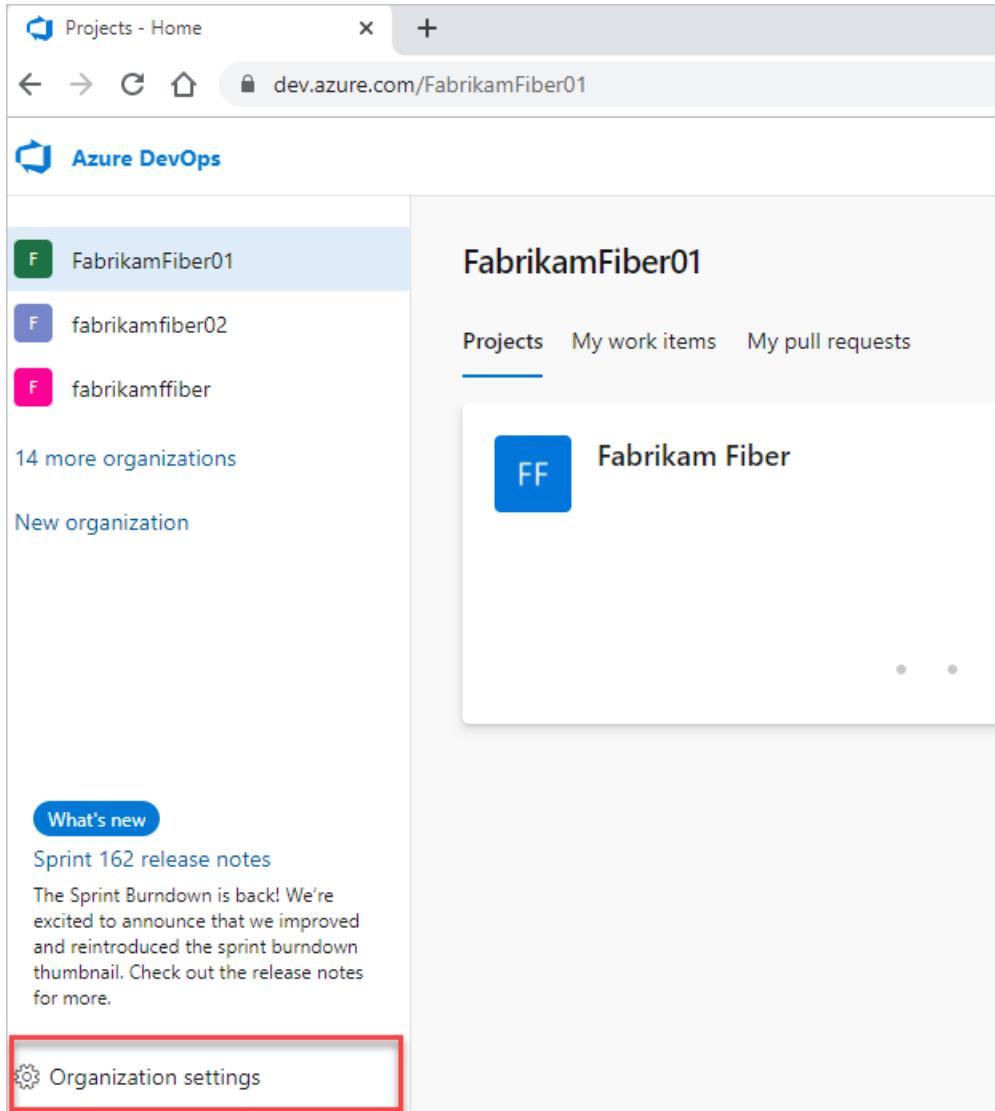
## Open Organization settings

Organization owners and members of the Project Collection Administrators group configure resources for all projects or the entire organization, including adding users, from the Organization settings pages. This includes managing permissions at the organization-level. For an overview of all organization settings, see [Project collection administrator role and managing collections of projects](#).

## Open Collection settings

Members of the Project Collection Administrators group configure resources for all projects or the entire project collection from the Collection settings pages. This includes managing permissions at the collection-level. For an overview of all collection-level settings, see [Project collection administrator role and managing collections of projects](#).

1. Choose the  Azure DevOps logo to open Projects. Then choose **Admin settings**.



The screenshot shows the Azure DevOps Projects - Home page for the organization 'FabrikamFiber01'. On the left sidebar, there's a list of organizations: 'FabrikamFiber01' (selected), 'fabrikamfiber02', 'fabrikamffiber', '14 more organizations', and 'New organization'. Below this is a 'What's new' section with a link to 'Sprint 162 release notes'. At the bottom of the sidebar, a red box highlights the 'Organization settings' option under a gear icon.

2. From there, you can choose a page from the list of settings. Settings are organized based on the service they support. Expand or collapse the major sections such as **Boards** and **Build and release** to select a page from the list.

## Organization Settings > Projects

General

Overview

Projects

Policy

Users

Security

Notifications

Extensions

Usage

> Boards

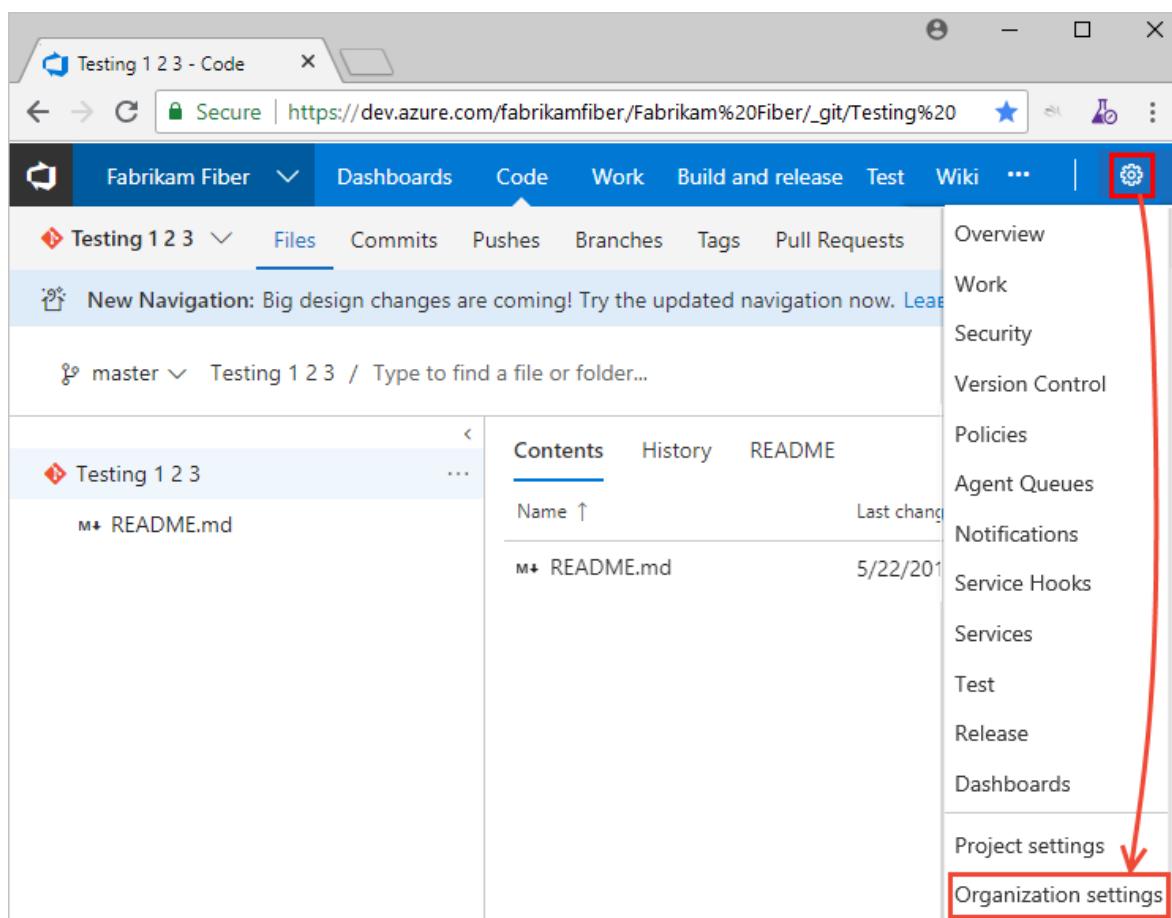
> Build and release

### Projects

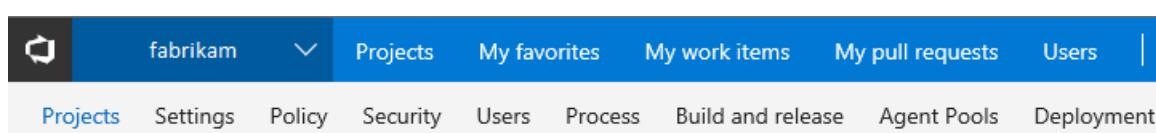
New team project... |

Project name	Process	Status	Description
Agile 11	MyAgile 2	Online	New agile project
Fabrikam Fiber	MyScrum	Online	Web, voice, and phone apps
Fabrikam Test	MyAgile Test	Online	MyAgile process customizations
Test Agile Repo	MyAgile	Online	
Visual Studio Code	Agile	Online	

1. Choose the gear icon to open Organization settings or Collection settings.



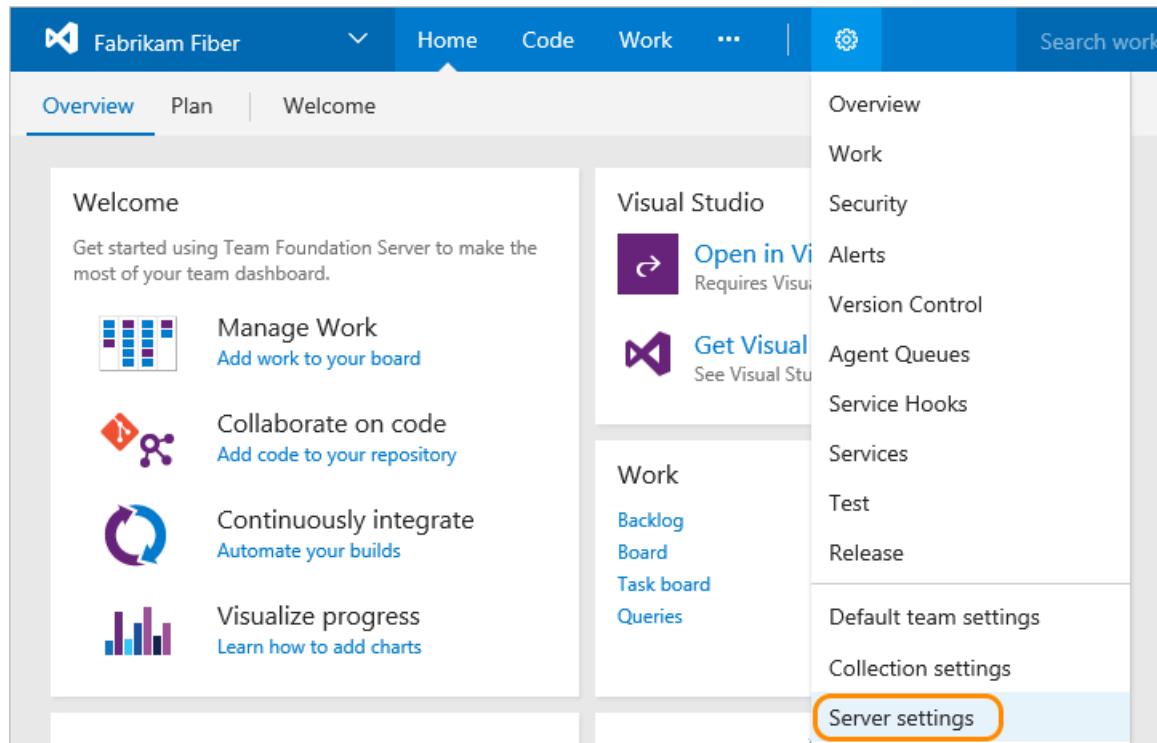
2. From there, you can choose a page. Settings are organized based on the service they support.



# Open Server settings

Members of the Team Foundation Server Administrators group configure resources for the server instance from the Server settings pages.

- From the web portal home page for a project, choose or hover over the gear icon and select **Server settings**.



- Choose **Access levels**, to set access levels for a member or group. For details, see [Change access levels](#).

If you don't see **Access levels**, you aren't a TFS administrator and don't have permission. [Here's how to get permissions](#).

## Related articles

- [Manage projects](#)
- [About team, project, and admin settings](#)

# Add an artifact or team artifacts

3/11/2021 • 3 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

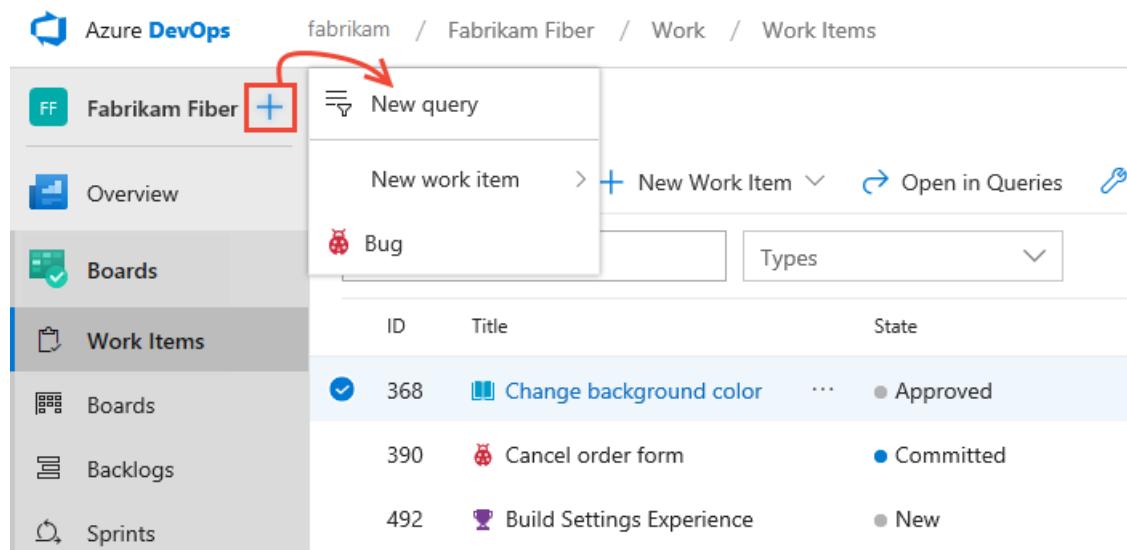
Select the service of interest to get started adding new artifacts or objects. For example, to add work items, choose **Boards** or **Work**. Some artifacts—such as a product backlog, Kanban board, portfolio backlogs—are added when you add a team.

Prior to adding an artifact, make sure that you've [selected the project and repository](#) that you want to work in.

## Add work items, queries, or other work tracking artifacts

You can quickly add a query or work item when working from a **Boards** or **Work** page.

Choose a **Boards** page—such as **Work Items**, **Boards**, or **Backlogs**. Then choose the **+** plus icon and select from the menu of options.



The screenshot shows the Azure DevOps interface for the 'Work Items' page. On the left, there's a sidebar with links for 'Overview', 'Boards', 'Work Items' (which is selected and highlighted in blue), 'Boards', 'Backlogs', and 'Sprints'. At the top right, there's a breadcrumb trail: 'fabrikam / Fabrikam Fiber / Work / Work Items'. Below the sidebar, there's a red box around a blue '+' icon. A red arrow points from this icon to a dropdown menu that says 'New query'. The main area shows a table of work items with columns for 'ID', 'Title', and 'State'. There are three items listed: one 'Bug' item titled 'Change background color' with state 'Approved', and two other items with titles 'Cancel order form' and 'Build Settings Experience' both in 'Committed' state.

ID	Title	State
368	Change background color	Approved
390	Cancel order form	Committed
492	Build Settings Experience	New

From a **Work** page, you can add a work item from the menu of options as shown in the following image.

Fabrikam Fiber ☆

Tracking web, voice, and other app development

Showing README.md in \$/Fabrikam Fiber A

Fabrikam Fiber Voice team

The Voice team supports all voice services, including VOIP, voicemail, instant messaging, and the frameworks, security, authentication, and file services associated with these services.

New Work Item >

Bug

User Story

Or, you can open one of the pages—**Boards**, **Backlogs**, **Queries**, or **Plans**—to add an artifact specific to each of these functional pages.

To add other work tracking artifacts, see one of the following articles:

- To add a board, backlog, or sprint backlog, first [add a team](#) which will be associated with those artifacts
- [Add a delivery plan](#)
- [Add a managed work item query](#)
- [Add work items](#).

## Add a pull request or Git repository

You can quickly add a pull request, Git repository, or work item using the **Add** menu when working from **Code**.

Expand the **Repos** service and choose **Files**, **Commits**, or **Pull Requests** (Git repos) or **Files**, **Changesets**, or **Shelvesets** (TFVC). Then, choose the **+** plus icon and select from the menu of options.

Azure DevOps fabrikam / Fabrikam Fiber / Code / Files / Testing 1 2 3

+ Fabrikam Fiber

Overview

Boards

Repos

Files

Commits

New pull request

New repository

New work item >

Bug

Contents History README

Name ↑

README.md

For details on adding a Git repository, see [Git repository](#).

From **Code**, open the context menu for the current repository and choose **+** **New repository**. For details on adding a Git repository, see [Git repository](#)

The screenshot shows the Azure DevOps interface with the 'Fabrikam Fiber' project selected. The top navigation bar includes 'Dashboards', 'Code', 'Work', 'Build and release', 'Test', and 'Wiki'. Under 'Code', the 'Files' tab is active. On the left, there's a sidebar with options like 'Testing 1 2 3', 'Fabrikam Fiber', '\$/Fabrikam Fiber', 'Test 1-2-3', 'Testing 1 2 3', '+ New repository' (which is highlighted with a red box), 'Import repository', and 'Manage repositories'. The main area shows a list of files: 'README.md' (modified, 5/22/2017). At the bottom of the sidebar, there are links for 'Contents', 'History', and 'README'.

From one of the other **Code** pages, you can add files or folders, a new branch, or a new pull request.

Note that you can only add one TFVC repository per project, but an unlimited number of Git repositories. To learn more about Git artifacts, see one of the following articles:

- [Git repository](#)
- [Git branch](#)
- [Git pull request](#)
- [Add work items](#)

## Add build and release pipelines

Expand **Pipelines** and choose **Builds** or **Releases**. Then choose the **+** plus icon and select from the menu of options.

The screenshot shows the Azure DevOps interface with the 'Fabrikam Fiber' project selected. The top navigation bar includes 'Dashboards', 'Code', 'Work', 'Build and release', 'Test', and 'Wiki'. Under 'Build and release', the 'Builds' tab is active. On the left, there's a sidebar with options like 'Overview', 'Boards', 'Repos', 'Pipelines' (which is highlighted with a red box), 'Builds', and 'Releases'. The main area shows a list of pipelines: 'New build pipeline', 'New release pipeline' (Queued, XAML), 'New work item' (Bug), and 'Fabrikam Fiber-Cl'. There's also a search bar for 'Build ID or build number'.

From **Build and Release**, choose **Builds**, **Releases**, or other page to add an artifact associated with that page.

The screenshot shows the Azure DevOps interface with the 'Fabrikam Fiber' project selected. The top navigation bar includes 'Dashboards', 'Code', 'Work', 'Build and release', 'Test', and 'Wiki'. Under 'Build and release', the 'Builds' tab is active. The main area shows a navigation bar with 'Builds' (selected), 'Releases', 'Library', 'Task Groups', 'Deployment Groups\*', and 'Build Tags'. There's also a search bar for 'Search work items in this'.

To learn more about adding other pipeline related artifacts, see the following articles:

- Deployment groups
- Task groups
- Variable groups
- Secure files

## Add a team

Agile tools and dashboards are typically associated with teams. You add teams to a project. To learn more about teams, see [About teams and Agile tools](#). To add a team, see [Add a team and team members](#).

## View teams already defined

To view the set of defined teams, open **Project settings**, and choose **Overview**.

The screenshot shows the Azure DevOps interface for a project named "Fabrikam Fiber". The left sidebar has a red box around the "Project settings" button. The main area shows "Project Settings > Teams". A red box highlights the "Teams" tab under the "General" category. On the right, there's a table of existing teams:

Team Name	Members	Description
Customer Service	7	
Fabrikam Fiber Team	7	
Management team	1	
Phone	1	
Voice	1	
Web	2	

To view the set of defined teams, open the admin context for the project, and choose **Overview**.

**Project profile**

Name: **Fabrikam Fiber**

Process: Scrum | [Change process](#)

Description: Web, voice, and phone apps

**Teams**

Team Name ↑	Members	Description
Customer Service	7	
<b>Fabrikam Fiber Team</b>	7	The default project team.
Management team	1	
Phone	1	
Voice	1	
Web	2	

## Add a dashboard

Dashboards are associated with a team or a project. Each team can create and configure a number of dashboards. And, any team member can create one or more project dashboards. To learn how, see [Add a dashboard](#).

Dashboards are associated with a team. Each team can create and configure a number of dashboards. To learn how, see [Add a dashboard](#).

## Add a wiki

If you don't have a wiki yet, you can add one. Once added, you can add and update pages to that wiki.

- [Create a wiki](#)
- [Add and edit wiki pages](#)
- [Publish a Git repository to a wiki](#)
  
- [Create a wiki](#)
- [Add and edit wiki pages](#)

## Related articles

- [Azure Artifacts](#)
- [Exploratory & Manual Testing](#)

# Use breadcrumbs, selectors, and directories to navigate and open artifacts

11/2/2020 • 3 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

To quickly navigate to a feature or artifact—such as a dashboard, repository, product backlog, Kanban board, build pipeline—you can use breadcrumbs, selectors, and directories.

## Organization and project breadcrumbs

To navigate to the project summary page, choose the project link within the breadcrumbs. To navigate to the organization page with all projects defined for the organization, choose the organization link.

**Organization / Project / Service / Page**

The screenshot shows the 'Work Items' page in Azure DevOps. At the top, there's a breadcrumb navigation bar: 'fabrikam / Fabrikam Fiber / Work / Work Items'. A red arrow points from the text 'Organization / Project / Service / Page' to the first segment of the breadcrumb ('fabrikam'). Below the breadcrumb is a left sidebar with links: 'Fabrikam Fiber' (selected), 'Overview', 'Boards', 'Work Items' (selected), 'Boards', and 'Backlogs'. The main area displays a table of work items:

ID	Title	State
390	Cancel order form	Committed
492	Build Settings Experience	New
375	Check service status	Committed
543	Develop form	To Do

Horizontal navigation doesn't provide a breadcrumb structure for the organization and project levels. Instead, you can select a recent team or project from the project/team selector.

The screenshot shows the project selector dropdown in Azure DevOps. The dropdown is open, showing a list of recent projects and teams: 'Fabrikam Fiber Home', 'Recent projects/teams', 'Agile 11', 'FabrikamFiber', 'Fabrikam Fiber A' (selected), 'Fabrikam Fiber PB', 'Browse...', and 'New team'. A red box highlights the dropdown icon. The main content area shows a placeholder for a dashboard or project page.

Choosing **Browse...** opens the **projects** page.

# Selectors

Selectors are used to select an artifact within the current page. Most Agile tools are defined for a team and therefore require selection of the team artifact or tool.

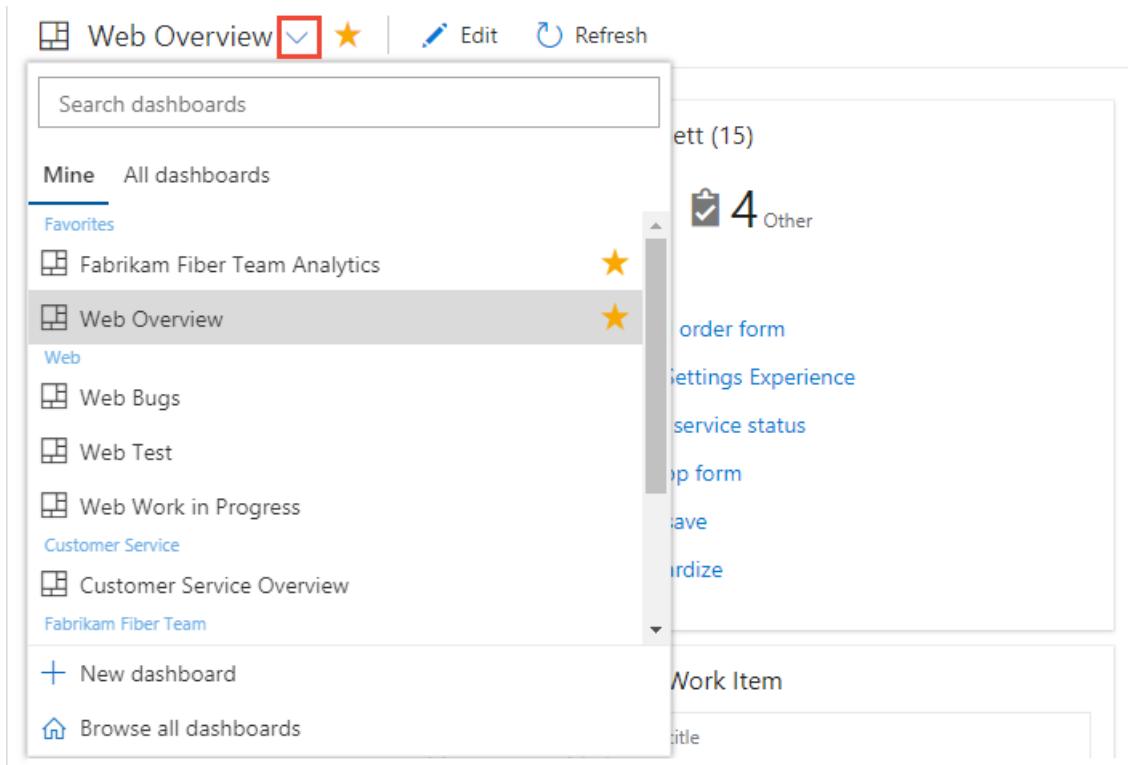
Selectors are used to select an artifact within the current page. Most Agile tools are defined for a team and therefore require selection of the team as well as the specific page.

## NOTE

When you navigate to a specific page or artifact, the system remembers your selection. You use selectors to choose a different artifact within the current page.

### Example: Dashboard selector

Within Dashboards, you open a specific dashboard from the selector.



This particular selector features these navigational elements:

- Search box for filtering dashboards based on a team name or keyword
- Two pages you can choose from:
  - Mine (dashboards you created) which are organized by team
  - All (dashboards created by everyone) which are listed alphabetically
- Dashboards you've favorited will appear at the top of the selector
- Add new dashboard feature
- Browse all dashboards - opens Dashboards>All

Within Dashboards, you select the team whose dashboards you want to view.

The screenshot shows the 'Fabrikam Fiber' application interface. At the top, there's a navigation bar with links for Dashboards, Code, Work, Build & Release, Test, and a gear icon. Below the navigation bar is a sidebar menu with sections for 'Recent projects/teams', 'Agile 11', 'FabrikamFiber', 'Fabrikam Fiber A' (which is selected and highlighted in blue), 'Fabrikam Fiber PB', 'Browse...', 'New team', and a note about a README.md file. The main content area displays a placeholder message: 'A README.md file is intended to quickly orient readers to what your project can do. Learn more about Markdown.'

Then, choose the name of the dashboard to view it.

For example, here we open the *Work in Progress* dashboard.

The screenshot shows the 'Fabrikam Fiber / Web' application interface. The top navigation bar includes links for Dashboards, Code, Work, Build and Release, and a gear icon. Below the navigation bar, a secondary navigation bar has tabs for Overview, Bugs, Work in Progress (which is selected and highlighted in red), and Test. The main content area is currently empty.

### Example: Backlogs

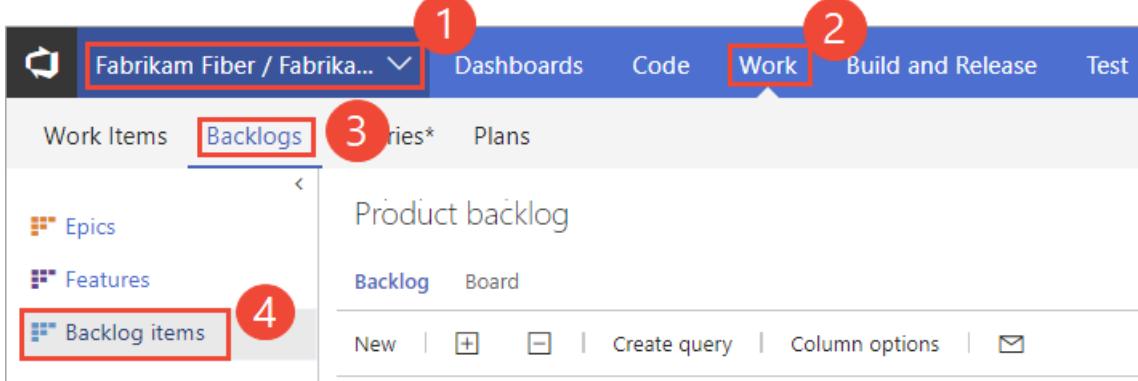
From the **Boards>Backlogs** page, you use the selector to switch to another team's backlog. Again, favorited backlogs appear towards the top of the menu. You can also filter the list based on a team name or keyword.

The screenshot shows the 'Boards>Backlogs' page. On the left, there's a sidebar with a project selector (set to 'Web') and a search bar ('Search team backlogs'). Below the search bar is a list of teams: Phone, Web (which is selected and highlighted in grey), Customer Service, Fabrikam Fiber Team, Management team, and Voice. At the bottom of the sidebar is a link to 'Browse all team backlogs'. The main content area shows a table of backlog items for the 'Web' team. The table has columns for 'Index', 'Title', 'State', and 'Last updated'. The items listed are:

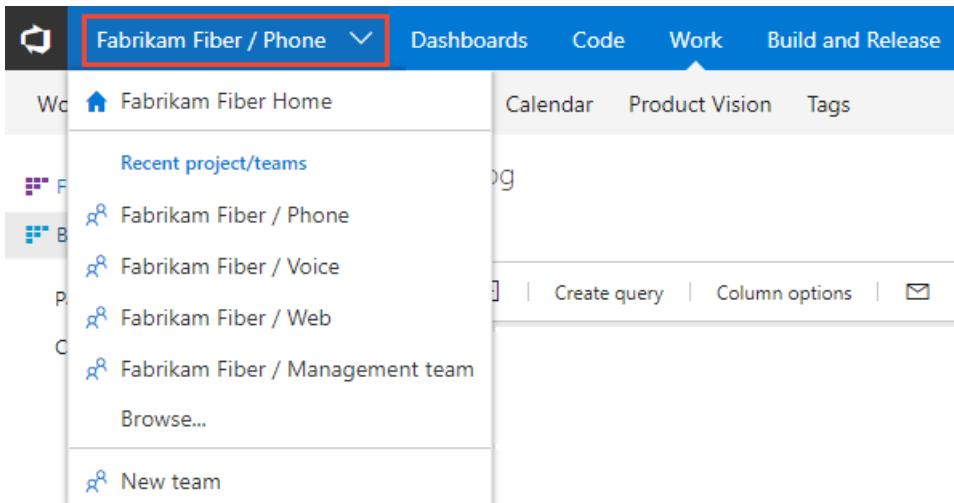
Index	Title	State	Last updated
8	Product Backl... > Request support	● Committed	2023-01-15
9	Product Backl... > Cancel order form	● Committed	2023-01-15

Or, choose **Browse all team backlogs** to open the **Backlogs>All** page.

(1) Select the team from the project/team selector, choose (2) **Work**, (3) **Backlogs**, and then (4) the product backlog, which is **Backlog items** (for Scrum), **Stories** (for Agile), or **Requirements** (for CMMI).



To choose another team, open the project/team selector and select a different team or choose the **Browse** option.



## Artifact breadcrumbs and selectors

Within select pages, breadcrumbs are provided to support navigating within the page or opening an artifact.

### Example: Queries folders and breadcrumbs

For example, when working in the **Queries** pages, you can navigate to a subfolder, folder, or page.

The screenshot shows the Azure DevOps **Queries** page with the following breadcrumb navigation path:

**Queries page / Queries folder / Queries subfolder / query**

The page title is "All Queries > Shared Queries > Current Sprint > Work in Progress".

The left sidebar shows navigation links for Overview, Boards, Work Items, Boards, Backlogs, Sprints, and **Queries** (which is highlighted).

The main content area displays a table of queries with columns: ID, State, Assigned To, Remaining Work, and Title. The first query in the list is highlighted with a blue background.

ID	State	Assigned To	Remaining Work	Title
399	In Progress	Jamal Hartnett	6	<input checked="" type="checkbox"/> Standardize on form
539	In Progress	Jamal Hartnett	8	<input checked="" type="checkbox"/> Standardize
538	In Progress	Johnnie McLeod	8	<input checked="" type="checkbox"/> Design welcome screen
371	In Progress	Johnnie McLeod	8	<input checked="" type="checkbox"/> Auto-complete user's name in form
388	In Progress	Raisa Pokrovskaya	6	<input checked="" type="checkbox"/> Code form

Also, you can choose a query that you've favorited from the selector menu, Or, you can choose to browse all queries which returns you to the **All Queries** page.

All Queries > Shared Queries > Current Sprint > Work in Progress

Results Editor Charts | Run query +

ID	State	Assigned To	Remaining Work
399	In Progress	Jamal Hartnett	6
539	In Progress	Jamal Hartnett	8
538	In Progress	Johnnie McLeod	8
371	In Progress	Johnnie McLeod	8
388	In Progress	Raisa Pokrovskaya	6

All items in a tree query ★

Work in Progress ★

Browse all queries

Standardize

Design welcome screen

Auto-complete user's name in form

Code form

### Example: Pipeline folders and breadcrumbs

Breadcrumb-and-selector navigation elements are used within most services that support defining and organizing artifacts within folders. This includes **Pipelines** or **Build and Release** applications pages.

Build Definitions / RedTeam / RedCreator

Summary History Deleted

Details

Repository vtbastmatt

Default queue

Queue status Enabled

Last updated by [redacted] | Tuesday, April 3, 2018 6:30 AM

Queued & running

No builds queued or running at the moment

Recently completed

#41 succeeded master

Choose the **Deployment** breadcrumb link to return to the *Deployment* folder.

Fabrikam Fiber

Builds Releases Releases\* Library Task groups Deployment Groups Build Tags

... > Deployment > Fabrikam Fiber-CI

## Directories

Directories provide a filterable list of all artifacts defined for a service area. Often when you navigate to an application, it will open the application's directory.

For example, here is the **Boards**>**Boards** directory.

Name	Team
Fabrikam Team boards	Fabrikam Team
My favorite boards (1)	
Fabrikam Team boards	Fabrikam Team
My team boards (8)	
Account Management boards	Account Management
Customer Profile boards	Customer Profile
Fabrikam Team boards	Fabrikam Team
Phone boards	Phone
Service Delivery boards	Service Delivery
Service Status boards	Service Status
Shopping Cart boards	Shopping Cart
TV boards	TV

It lists boards in the following order:

- Your last visited board
- Your favorited boards
- All boards of teams that you belong to
- All boards defined for the project in alphabetical order.

Choose the filter icon to filter the list as described in [Filter basics](#).

From a specific page, you can open the directory from the breadcrumbs or a selector. For example, choose **Browse all boards** from the Boards selector.

[Open from breadcrumb](#)

[Open from selector](#)

The screenshot shows the 'Browsing boards' interface in Azure DevOps. At the top, there's a search bar labeled 'Search team boards'. Below it, under 'Favorites', is a board titled 'Fabrikam Team' with a yellow star icon. Under 'My Team Boards', there are several boards: 'Account Management', 'Customer Profile', 'Fabrikam Team' (with a yellow star), 'Phone', and 'Service Delivery'. At the bottom of this list is a button labeled 'Browse all boards', which is highlighted with a red box.

fabrikam / Fabrikam Fiber / Boards / **Boards**

## Team profiles

- Open a team profile to quickly access items defined for a team. The team profile is available from the Overview > Dashboards, Boards > Boards, Boards > Backlogs, and Boards > Sprints pages.

The screenshot shows the 'Service Delivery' team profile in Azure DevOps. At the top, the team name 'Service Delivery' is displayed with a yellow star icon and a 'g8' link, both of which are highlighted with red boxes. Below the header, there are navigation links for 'Overview', 'New Work Item', 'View as Board', 'Column Options', and more. The main content area displays the team's items, starting with a purple circular icon containing two people, followed by the team name 'Service Delivery', its parent organization 'Fabrikam Fiber', and a 'Team Settings' link. A blue underline indicates the 'Items' tab is selected. Below this, a dropdown menu shows 'All Items' with a red box around it. The list of items includes 'Service Delivery Boards' (with a yellow star), 'Service Delivery Backlogs' (with a yellow star), 'Sprint 2 Sprints' (with a yellow star), and 'Overview Dashboards' (with a yellow star).

A panel opens that shows all items defined for the team.

This screenshot shows the expanded 'Service Delivery' team profile. It features a purple circular icon with two people, the team name 'Service Delivery', its parent organization 'Fabrikam Fiber', and a 'Team Settings' link. The 'Items' tab is selected, showing a dropdown menu with 'All Items'. Below this, a list of items is displayed with yellow star icons: 'Service Delivery Boards', 'Service Delivery Backlogs', 'Sprint 2 Sprints', and 'Overview Dashboards'.

- You can filter the list to show only Dashboards, Boards, Backlogs, or Sprints by choosing from the menu.

The screenshot shows the 'Service Delivery' team settings page. At the top, there's a purple icon with two people, followed by the team name 'Service Delivery', the project name 'Fabrikam Fiber', and the link 'Team Settings'. Below this, there are two tabs: 'Items' (which is underlined in blue) and 'Members (7)'. A red box highlights the 'Items' tab. A dropdown menu is open, showing five options: 'All Items' (which is highlighted with a light blue background), 'Dashboards', 'Boards', 'Backlogs', and 'Sprints'. There is also a small '▼' icon at the bottom right of the dropdown.

- To view the team admins and members of the team, choose **Members**.

The screenshot shows the 'Service Delivery' team settings page. At the top, there's a purple icon with two people, followed by the team name 'Service Delivery', the project name 'Fabrikam Fiber', and the link 'Team Settings'. Below this, there are two tabs: 'Items' and 'Members (7)', with 'Members (7)' being underlined in blue and highlighted with a red box. A red box also highlights the 'Members' tab. The main area shows the heading 'Admins' followed by a list item 'Cristina Potra' with a profile picture and initials 'CP'. Below this, the heading 'Members' is shown, followed by four more list items: 'Christie Church' (profile picture and initials 'CC'), 'Chuck Reinhart' (profile picture and initials 'CR'), 'Jamal Hartnett' (profile picture and initials 'JH'), and 'Johnnie McLeod' (profile picture and initials 'JM').

- To view or change the team configuration, choose **Team Settings**.  
You can then add [team members](#), [team admins](#), or navigate to [team notifications](#), or team [iterations](#) and [area paths](#).

See also [Manage and configure team tools](#).

## Related articles

- About teams and Agile tools
- Add an artifact or team
- Set favorites
- Open a service or page
- Filter basics

# Switch project, repository, team

3/6/2021 • 3 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

Several features depend on the project, repository, or team that you have selected. For example, dashboards, backlogs, and board views will change depending on the project and team you select.

Also, when you add a work item, the system references the default area and iteration paths defined for the team context. Work items you add from the team dashboard (new work item widget) and queries page are assigned the team default iteration. Work items you add from a team backlog or board, are assigned the team default backlog iteration. To learn more, see [About teams and Agile tools](#).

## Prerequisites

- You must be added to a project as a member of the **Contributors** or administrator security group. To get added, [Add users to a project or team](#).

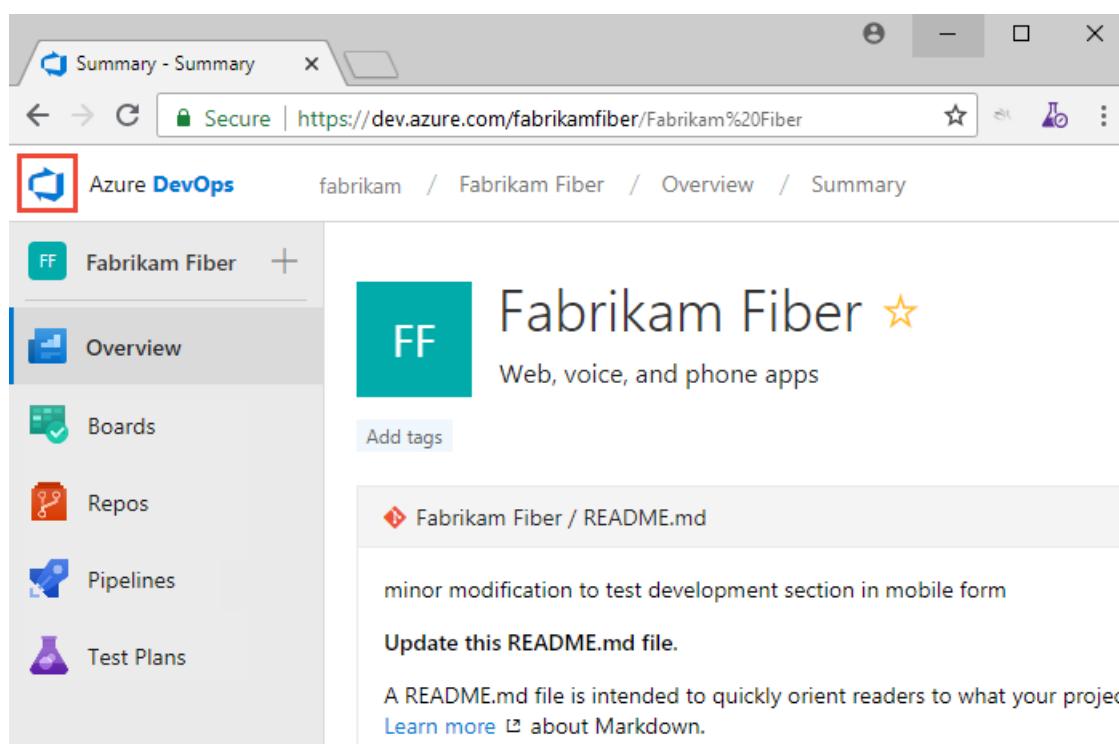
### NOTE

If the **Project-Spaced Users well known group to hide settings** preview feature is enabled for the organization, users added to the **Project-Spaced Users** group won't be able to access projects that they haven't been added to. To learn more, see [About projects and scaling your organization](#), [Project-scoped Users group](#).

## View and open a project

From the **Projects** page you can quickly navigate to a project that you have permissions to view.

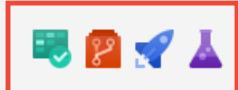
1. Choose the  Azure DevOps logo to open **Projects**.



The projects you most recently viewed are displayed, followed by a list of all projects in alphabetic order.

2. Hover over the dots and you can open the service of interest for that project.

**Projects**  [+ Create project](#)

<b>FF</b>	<b>Fabrikam Fiber</b> Web, voice, and phone apps	<b>M</b>	<b>MyFirstProject</b>
...			
<a href="#">All projects</a>			
<b>A1</b>	<b>Agile 11</b> New agile project	...	
<b>D1</b>	<b>Demo 11</b> Agile team project		
<b>FF</b>	<b>Fabrikam Fiber</b> Web, voice, and phone apps	...	
<b>M</b>	<b>MyFirstProject</b>	...	

3. You can filter the project and team list using the *Filter projects* search box. Simply type a keyword contained within the name of a project or team. Here we type **Fabrikam** to find all projects or teams with *Fabrikam* in their name.

**Projects**  [X](#) [+ Create project](#)

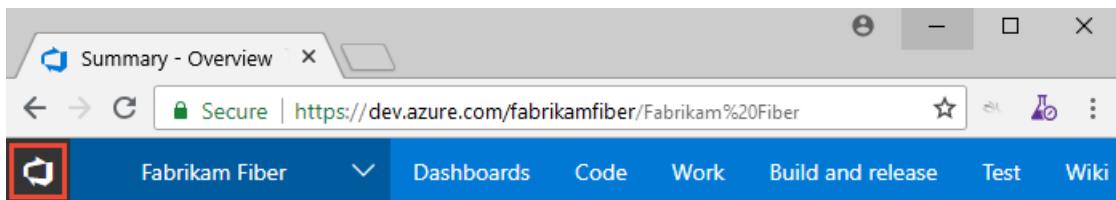
<b>FF</b>	<b>Fabrikam Fiber</b> Web, voice, and phone apps	...
<b>FT</b>	<b>Fabrikam Test</b> Project used to verify MyAgile process customizations	...
<b>F</b>	<b>FabrikamFiber</b> Customer-focused apps under development based on Agile process.	...

4. Choose **Create Project** to add a project. You must be an account administrator or a member of the Project Collection Administrators group to [add a project](#).

The screenshot shows the Azure DevOps 'Projects' page. On the left, there's a sidebar titled 'My accounts' with three entries: 'fabrikamfib' (dark blue), 'FabrikamFiber' (light blue), and 'fabrikam-fiber' (green). The main content area is titled 'Projects' and features a search bar 'Filter projects'. A prominent red box highlights the '+ Create project' button. Below the button are two project cards: 'Fabrikam Fiber' (Web, voice, and phone apps) and 'MyFirstProject'. Each card has a small icon (teal for Fabrikam Fiber, magenta for MyFirstProject) and a horizontal ellipsis below it.

From the **Projects** page you can quickly navigate to a project or a team that you've accessed or worked in previously. Projects and teams are listed in the order you've last accessed, with the most recent five projects accessed appearing first. All projects you've accessed are listed within the **All** section.

1. Choose the  Azure DevOps logo to open **Projects**.



The projects you most recently viewed are displayed, followed by a list of all projects in alphabetic order.

The screenshot shows the 'Projects' page with a navigation bar at the top. The 'Projects' tab is selected. Below the navigation bar is a 'Recent' section listing three projects: 'Fabrikam Fiber', 'Fabrikam Fiber / Web' (which has a yellow star icon to its right), and 'FabrikamFiber'. There's also a 'Filter projects and teams' search bar and a 'New Project' button.

2. As you hover over a project or team, you can choose one of the links to go to **Home** or **Dashboards**, **Code**, **Work**, **Build and Release**, **Test**, or **Wiki** pages. Choose the  star icon to mark the project as a favorite.

The screenshot shows the 'Projects' page with the 'Fabrikam\_Fiber' project selected. The navigation bar now includes a yellow star icon next to the project name 'Fabrikam\_Fiber'.

3. You can filter the project and team list using the *Filter projects and teams* search box. Simply type a keyword contained within the name of a project or team. Here we type **Fabrikam** to find all projects or teams with *Fabrikam* in their name.

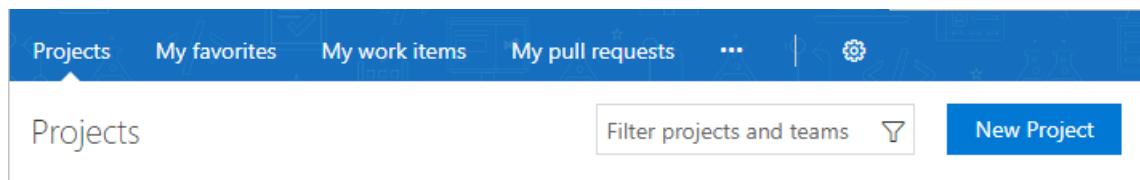
## Projects

Fabrikam 

### Results

-  Fabrikam Fiber
-  Fabrikam Fiber / Customer Service ★
-  Fabrikam Fiber / Management team ★
-  Fabrikam Fiber / Phone ★
-  Fabrikam Fiber / Voice ★
-  Fabrikam Fiber / Web ★
-  Fabrikam Test
-  FabrikamFiber

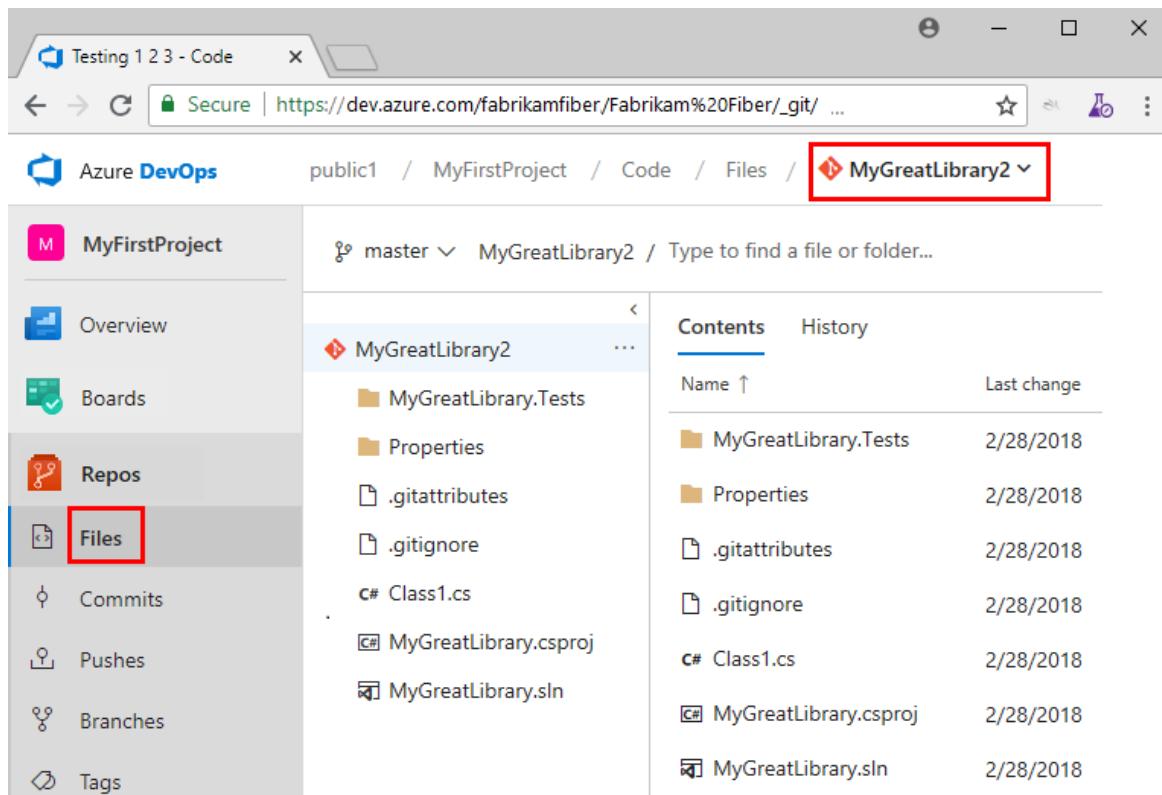
4. Choose **New Project** to add a project. You must be an account administrator or a member of the Project Collection Administrators group to [add a project](#).



The screenshot shows the Azure DevOps interface with the 'Projects' tab selected in the top navigation bar. Below the navigation, there's a search bar labeled 'Filter projects and teams' and a blue 'New Project' button.

## View and open a repository

1. Choose **Repos > Files**.



The screenshot shows the Azure DevOps repository browser. On the left, a sidebar menu has 'Repos' selected and 'Files' highlighted with a red box. The main area shows a repository structure for 'MyFirstProject' under 'MyGreatLibrary2'. The 'Contents' tab is selected, displaying a list of files and folders:

Name	Last change
MyGreatLibrary.Tests	2/28/2018
Properties	2/28/2018
.gitattributes	2/28/2018
.gitignore	2/28/2018
Class1.cs	2/28/2018
MyGreatLibrary.csproj	2/28/2018
MyGreatLibrary.sln	2/28/2018

2. Select the repository of interest from the repository selector.

The screenshot shows the GitHub interface for a user named 'public1'. The user has selected the 'MyFirstProject' organization. In the 'Code' section, the 'Files' tab is active. A dropdown menu titled 'MyGreatLibrary2' is open, listing several repositories: 'breadth-of-trees', 'Compose', 'GVFS', 'MyGreatLibrary', 'MyGreatLibrary2', 'RedBot', 'Specs', 'VSCode', 'New repository', 'Import repository', and 'Manage repositories'. The 'MyGreatLibrary2' item is highlighted with a red box and a yellow star icon. A red arrow points from the 'MyGreatLibrary2' dropdown to the 'MyGreatLibrary2' item in the list.

1. Choose **Code**.

The screenshot shows the main project page for 'MyFirstProject'. The top navigation bar is blue with tabs for 'MyFirstProject', 'Dashboards', 'Code' (which is highlighted), 'Work', 'Build and Release', and 'Wiki'. The project name 'MyFirstProject' is displayed prominently with a large pink 'M' icon. Below it, the text 'No description provided' is shown. A 'TypeScript' badge is present. A file list on the right includes 'VSCode / README.md'.

2. Select the repository from the selector.

The screenshot shows the 'Files' tab of the 'MyGreatLibrary' repository. A red box highlights the repository selector at the top left, and a red circle with the number '1' is placed over it. Another red box highlights the 'MyGreatLibrary' repository in the list below, and a red circle with the number '2' is placed over it. The list includes 'breadth-of-trees', 'Compose', 'GVFS', 'MyGreatLibrary' (highlighted), 'MyGreatLibrary2', 'RedBot', 'Specs', and 'VSCode'. To the right, a table shows commit history for files like 'MyGreatLibrary.Tests', 'MyGreatLibrary.csproj', and 'MyGreatLibrary.sln'. The commits are all marked as 'succeeded'.

File	Last change	Commits
MyGreatLibrary.Tests	2/28/2018	2fb94e4 create library
MyGreatLibrary.csproj	2/28/2018	2fb94e4 create library
MyGreatLibrary.sln	2/28/2018	2fb94e4 create library

## Switch to a different team

From a user page, one under—**Boards**, **Repos**, **Pipelines**, or **Test Plans**—you can't switch to a different team, you can [only select team artifacts](#).

From a **Project Settings>Work>Team configuration** page, you select a team from the team selector breadcrumb.

The screenshot shows the 'Project Settings' interface with 'Team configuration' selected. A dropdown menu is open under the breadcrumb, showing a list of teams: Phone (Fabrikam Fiber), Voice (Fabrikam Fiber), Web (Fabrikam Fiber), Customer Service (Fabrikam Fiber), Fabrikam Fiber Team (Fabrikam Fib...), Management team (Fabrikam Fiber), Epics, Features, Backlogs, and More teams. The 'Fabrikam Fiber Team (Fabrikam Fib...)' option is highlighted. On the left sidebar, 'General' and 'Boards' sections are visible, with 'Team configuration' currently selected.

You can switch your team focus to one that you've recently viewed from the project/team selector. If you don't see the team or project you want, choose **Browse...** or choose the Azure DevOps logo to [access the Projects page](#).

The screenshot shows the Azure DevOps navigation bar with the 'Fabrikam Fiber' project selected. A dropdown menu is open next to the project name, listing recent projects and teams: Fabrikam Fiber Home, Recent projects/teams (Agile 11, FabrikamFiber, Fabrikam Fiber A, Fabrikam Fiber PB), and a 'Browse...' option. The 'Fabrikam Fiber PB' option is highlighted. The 'Azure DevOps logo' is visible at the top right of the page.

### TFS 2017.1 To switch your team focus to a project or team you've recently viewed, hover over the  Azure DevOps logo and choose from the drop-down menu of options. If you don't see the team or project you want, choose \*\*Browse...\*\* to [browse all projects and teams](work-across-projects.md).

The screenshot shows the TFS 2017 web interface. At the top, there's a navigation bar with links for Dashboards, Code, Work, Build & Release, Test, and a gear icon. Below this is a sidebar with a dropdown menu titled "Recent projects/teams". The menu lists "Fabrikam Fiber / Web", "Fabrikam Fiber", "Fabrikam Fiber / Fabrikam Fiber Team", "Fabrikam Fiber Scrum", and "Browse...". To the right of the sidebar, the main content area has a large "new project!" button. Further down, there are sections for "My Projects", "My Favorites", "My Work items", "My Pull requests", and "More". At the bottom of the sidebar, there are buttons for "New team" and "New project".

## TFS 2017

Open the project/team drop-down menu and select the project/team that you've recently visited. If you don't see the team or project you want, choose **Browse all** to browse all projects and teams.

This screenshot shows the TFS 2017 web interface with a similar layout to the previous one. The "Fabrikam Fiber" project is selected in the dropdown menu. The main content area now displays specific options for "Visual Studio" and "Work". Under "Visual Studio", there are links to "Open in Visual Studio" (Requires Visual Studio 2013+) and "Get Visual Studio" (See Visual Studio downloads). Under "Work", there are links to "Backlog", "Board", "Task board", and "Queries". The sidebar and other navigation elements are consistent with the first screenshot.

## Related articles

- [Work across projects](#)
- [Add teams](#)

# Tutorial: Set personal or team favorites

11/2/2020 • 6 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

Favorite  those views that you frequently access. You can favorite all sorts of Azure DevOps features and tools—such as a project, repository, build pipeline, dashboard, backlog, board, or query. You can set favorites for yourself or your team.

As your code base, work tracking efforts, developer operations, and organization grows, you'll want to be able to quickly navigate to those view of interest to you and your team. Setting favorites allows you to do just that.

Team favorites are a quick way for members of your team to quickly access shared resources of interest. You favorite an item for yourself by choosing the  star icon. The favorited item will then show up easily from one or more directory lists. You set favorites for a team through the context menu for the definition, view, or artifact.

In this tutorial you'll learn how to view your personal favorites and to favorite or unfavorite the following views:

- Project or team
- Dashboard
- Team backlog, board, shared query, or other Azure Boards view
- Repository
- Build and release definition
- Test plans
  
- Project
- Shared query
- Repository
- Build and release definition
- Test plans

## Prerequisites

- You must connect to a project through the web portal. If you don't have a project yet, [create one](#). To connect to the web portal, see [Connect to a project](#).
- You must be a member of the **Contributors** or an administrators security group of the project. To get added, [Add users to a project or team](#).
- To favorite projects, backlogs, boards, queries, dashboards, or pipeline views, you must have **Stakeholder** access or higher.
- To favorite repositories, or delivery plans, you must have **Basic** access or higher.
- To favorite test plans, you must have **Basic + Test Plans** access level or equivalent.
  
- You must connect to a project through the web portal. If you don't have a project yet, [create one](#). To connect to the web portal, see [Connect to a project](#).
- You must be a member of the **Contributors** or an administrators security group of the project. To get added, [Add users to a project or team](#).
- To favorite projects, backlogs, boards, queries, dashboards, or pipeline views, you must have **Stakeholder** access or higher.
- To favorite repositories, or delivery plans, you must have **Basic** access or higher.

- To favorite test plans, you must have **Basic + Test Plans** access level or equivalent.

For details about the different access levels, see [About access levels](#).

## View personal favorites

Access views that you have favorited by choosing the inbox icon, and then choosing **Favorites**.

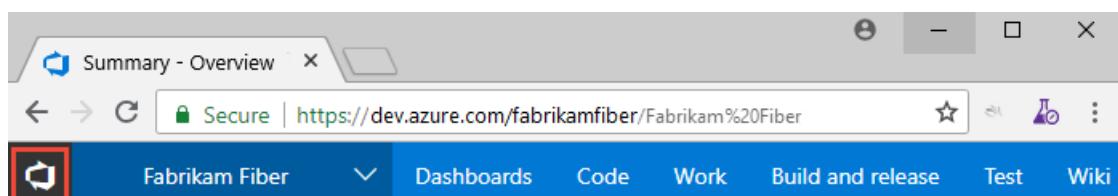
The screenshot shows the 'Favorites' section of the Azure DevOps interface. At the top, there's a navigation bar with icons for search, inbox (highlighted with a red box), library, and profile. Below the navigation bar, the 'Favorites' tab is selected, indicated by a blue underline and a red box around it. The main area is organized into sections: Projects, Teams, Dashboards, Plans, and Queries. Each section lists items with a star icon indicating they are favorited. The 'Projects' section includes 'Fabrikam Fiber'. The 'Teams' section includes 'Phone', 'Voice', and 'Web'. The 'Dashboards' section includes 'Fabrikam Fiber Team Analytics'. The 'Plans' section includes 'Backlog team plans' and 'Fabrikam Fiber Feature plans'. The 'Queries' section includes 'All items', 'All items on all projects', and 'Assigned to me'.

Category	Item	Status
Projects	Fabrikam Fiber	★
Teams	Phone	★
	Voice	★
	Web	★
Dashboards	Fabrikam Fiber Team Analytics	★
Plans	Backlog team plans	★
	Fabrikam Fiber Feature plans	★
Queries	All items	★
	All items on all projects	★
	Assigned to me	★

### NOTE

If a service is disabled, then you can't favorite an artifact or view of that service. For example, if **Boards** is disabled, then the favorite groups—Plans, Boards, Backlogs, Analytics views, Sprints, and Queries and all Analytics widgets—are disabled. To re-enable a service, see [Turn an Azure DevOps service on or off](#).

1. Access views that you have favorited by choosing the Azure DevOps logo to open **Projects**.



- Choose **My Favorites** to quickly access any view or item that you've marked as a favorite.

Favorites

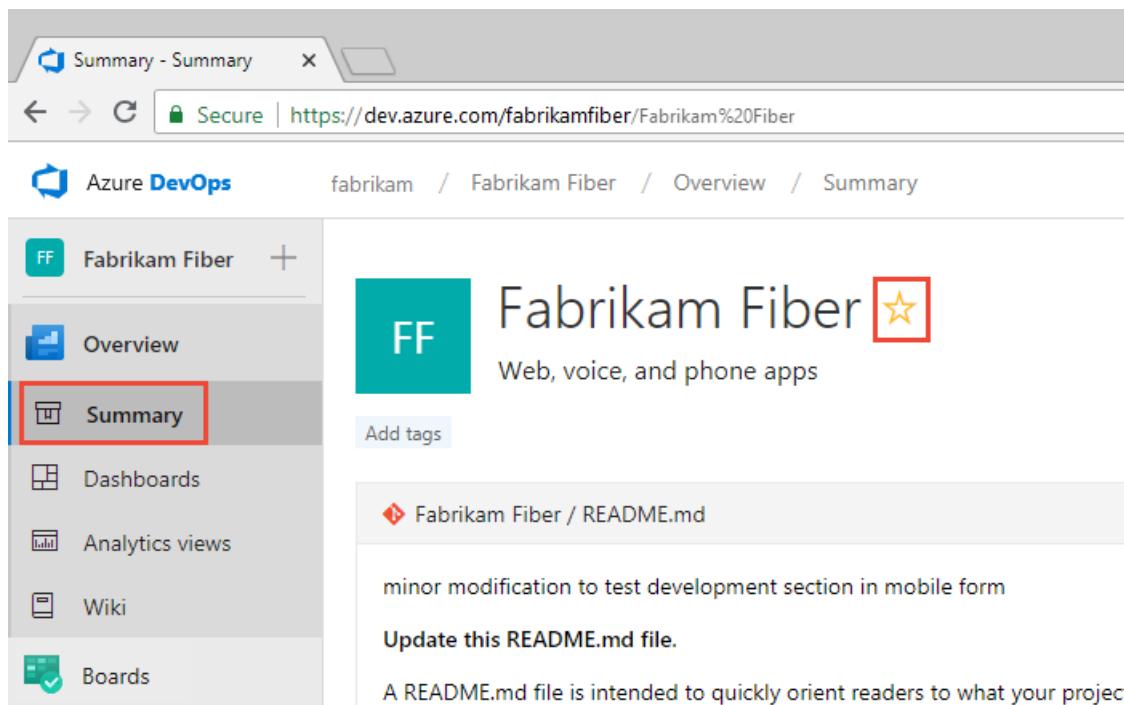
Filter favorites 

Queries

 Bug Triage	Fabrikam Fiber	.../Shared Queries/Current Iteration	
 My Bugs	Contoso	Shared Queries	
 Open User Stories	Contoso	.../Shared Queries/Current Iteration	
 Product Planning	Fabrikam Fiber	Shared Queries	
 Product Planning	Contoso	Shared Queries	

## Favorite a project or team

- To favorite a project, open the project **Summary** page and choose the  star icon.



The screenshot shows the Azure DevOps interface for the 'Fabrikam Fiber' project. The browser address bar shows the URL: <https://dev.azure.com/fabrikamfiber/Fabrikam%20Fiber>. The page title is 'Summary - Summary'. The left sidebar has a 'Fabrikam Fiber' section with tabs for Overview, Summary (which is selected and highlighted with a red box), Dashboards, Analytics views, Wiki, and Boards. The main content area displays the project logo 'FF', the project name 'Fabrikam Fiber' with a yellow star icon, and a brief description: 'Web, voice, and phone apps'. Below this, there's a 'Add tags' button and a 'Fabrikam Fiber / README.md' section containing a note about modifying the README file.

- To favorite a team artifact, open **Boards > Boards** or **Boards > Backlogs**. Select the team you want to favorite from the team selector and choose the  star icon.



The screenshot shows the 'Backlog items backlog' section of the Azure DevOps Boards > Backlogs page. At the top, there's a team selector showing 'Phone' and a star icon. Below it, the backlog items backlog is displayed with a blue progress bar at the top. The backlog items are listed with their titles and descriptions.

- To favorite other team artifacts, choose the  team icon, and then choose the  star icon next to one of the listed artifacts.

The screenshot shows the 'Phone' team settings page. At the top, there's a purple circular icon with three stylized human figures. Below it, the team name 'Phone' and owner 'Fabrikam Fiber' are displayed, along with a 'Team Settings' link. A navigation bar at the bottom has 'Items' (underlined) and 'Members (1)' tabs. A dropdown menu shows 'All Items'. The main list contains three items: 'Phone Boards', 'Phone Backlogs', and 'Phone Sprints', each with a yellow star icon.

Item	Sub-Category	Star Icon
Phone Boards	Boards	Yellow Star
Phone Backlogs	Backlogs	Yellow Star
Phone Sprints	Sprints	Yellow Star

## Favorite a project

To favorite a project, open the project **Summary** page and choose the star icon.

The screenshot shows the 'FabrikamFiber' project summary page. The title 'FabrikamFiber' is at the top left, with a yellow star icon to its right. Below it, a sub-header says 'Customer-focused apps under development based on Agile process.' To the right, there's a sidebar with sections for 'Members' (K), 'Activity' (empty), 'Code' (empty), and 'Build & Rel' (No builds yet). A central callout box is open, showing a large download icon and the text 'Use continuous integration'. It explains that code quality is improved by detecting breaking changes as soon as they happen. A 'Setup Build' button and a 'Learn more about continuous integration' link are also present.

Or, you can favorite a project from the **Projects** page by choosing the star icon next to the project.

## Favorite a dashboard

- From **Overview > Dashboards**, open the selector and choose the **Browse all dashboards** option.

The screenshot shows the Microsoft Power BI 'Mine' page. On the left, there's a sidebar with a search bar at the top. Below it, under 'Favorites', are 'Fabrikam Team Analytics' and 'Account Management Overview'. Under 'Customer Profile', is 'Customer Profile Overview'. Under 'Fabrikam Team', is 'Fabrikam Team Analytics'. At the bottom of the sidebar is a red-bordered button labeled 'Browse all dashboards'. The main area displays two cards: a green one for 'Fabrikam Items' with the number 6 and a purple one for 'Fabrikam Fiber ...' with the number 0. Below these cards is a section titled 'Commits by State'.

2. The **Mine** page shows your favorited dashboards, and all dashboards of teams that you belong to. The **All** page (shown below) lists all dashboards defined for the project in alphabetical order. You can filter the list by team or by keyword.

The screenshot shows the Microsoft Power BI 'All' page. At the top, there are tabs for 'Mine' and 'All', with 'All' being selected and highlighted with a red box. There's also a '+ New dashboard' button and a search/filter icon. Below the tabs is a 'Filter dashboards' button and a 'Filter by team' dropdown menu with a search bar and a list of teams: Account Management, Customer Profile, Email, Fabrikam Team, Internet, Phone, Service Delivery, and Service Status. A 'Clear' button is at the bottom of the filter menu. The main table lists various dashboards with columns for Name, Team, and a star icon indicating favoriting. The 'Team Guidance' dashboard is currently selected.

Name	Team
Analytics	Fabrikam Team
Bug status	Fabrikam Team
Bugs	Internet
Overview	Account Management
Overview	Customer Profile
Overview	Email
Overview	Fabrikam Team
Overview	Internet
Overview	Phone
Overview	Service Delivery
Overview	Service Status
Team Guidance	Fabrikam Team
Work in Progress	Internet

**TIP**

You can change the sort order of the list by choosing the column label.

3. To favorite a dashboard, hover over the dashboard and choose the star icon.

The screenshot shows the Azure DevOps interface with the 'Analytics' dashboard selected. At the top, there is a 'Name ↑' sorting header. Below it, a dark button labeled 'Add to favorites' is highlighted with a red box. To its right is a 'Team' link. Further down, a yellow star icon with a red border is also highlighted with a red box, indicating it's the 'Add to favorites' button. The 'Fabrikam Team' name is visible next to the star icon.

Favoriting a dashboard will cause it to appear on your **Favorites** page and towards the top in the **Dashboards** selection menu.

## Favorite a team's backlog, Kanban board, or other view

You can favorite several Agile tools for a team from a **Boards** page.

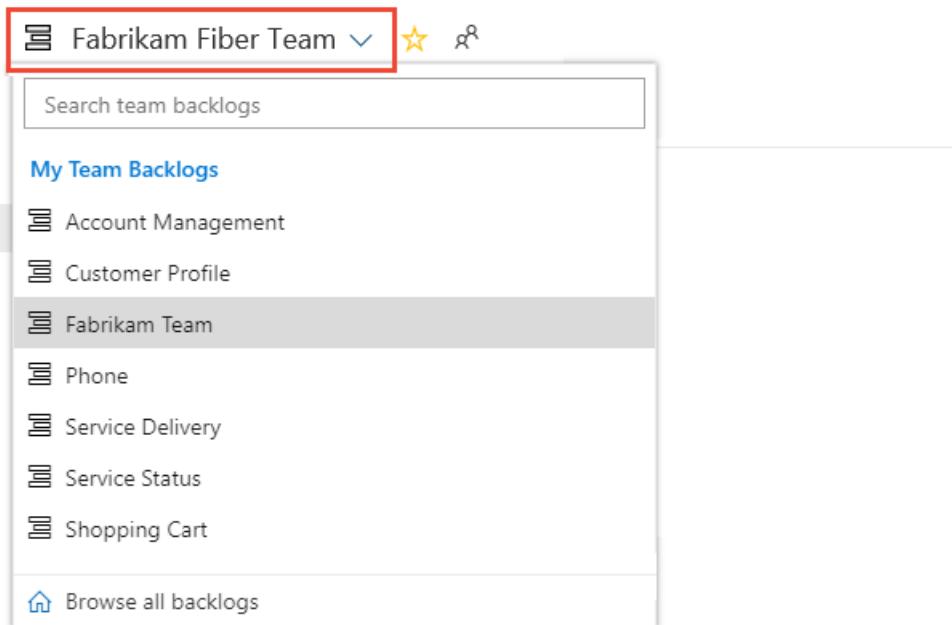
1. Choose **Boards**, and then choose the page of interest, such as **Boards**, **Backlogs**, or **Sprints**.

For example, here we choose (1) **Work** and then (2) **Backlogs**.

The screenshot shows the Azure DevOps 'Boards' page for the 'Fabrikam Fiber' project. On the left, a sidebar lists navigation options: Overview, Boards, Work Items, Boards, Backlogs (which is highlighted with a red box), Sprints, and Queries. The main area displays the 'Fabrikam Fiber Team' backlog. At the top of the backlog list is a yellow star icon with a red border, indicating it is favorited. Below the star are buttons for 'New Work Item', 'Backlog items Board', and 'Backlog items'. The backlog table has columns for Order, Assigned To, State, and Title. Seven items are listed:

Order	Assigned To	State	Title
1	Jamal Hartnett	Committed	Hello World Web Site
2	Jamal Hartnett	Committed	Slow response on informa
3	Raisa Pokrovskaya	New	Add an information form
4	Raisa Pokrovskaya	New	Change initial view
5	Christie Church	Committed	Secure sign-in
6	Johnnie McLeod	Approved	Welcome back page
7	Christie Church	Committed	Cancel order form

To choose a specific team backlog, open the selector and select a different team or choose the **Browse all team backlogs** option. Or, you can enter a keyword in the search box to filter the list of team backlogs for the project.



2. Choose the star icon to favorite a team backlog. Favorited artifacts ( favorited icon) appear on your **Favorites** page and towards the top of the team backlog selector menu.

## Favorite a shared query

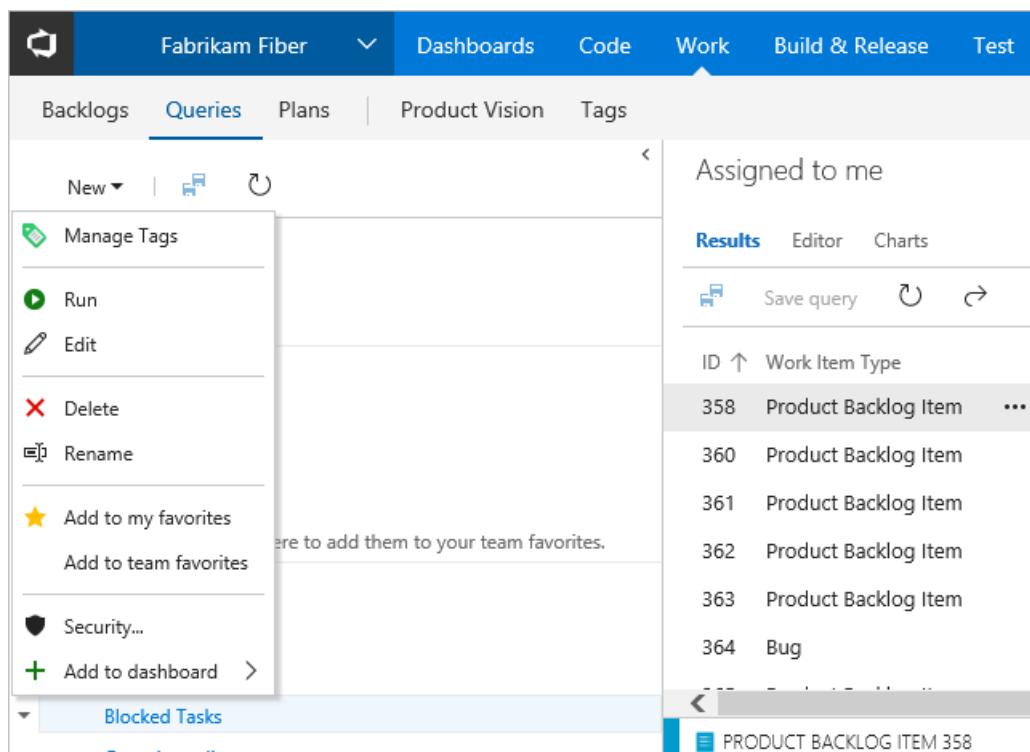
Open **Boards > Queries** and choose the **All** page. Expand a folder as needed. Choose the star icon next to the query you want to favorite.

Or, open the context menu of the query, and then select **Add to Team Favorites**, and then select from the list of teams.

The screenshot shows the 'Queries' page with the 'All' tab selected. A context menu is open over a query titled 'Blocked Tasks'. The 'Add to Team Favorites' option is highlighted with a red box. A second red box highlights the 'Phone' team selection in the dropdown menu.

You can also set a query as a personal favorite by opening the query and choosing the star icon.

Open **Work>Queries**. Next, open the \*\*\* actions icon menu of the shared query you want to favorite, and then select **Add to my favorites** or **Add to team favorites**.



The screenshot shows the Microsoft Teams interface with the 'Queries' tab selected. A context menu is open on the right side, specifically on a 'Product Backlog Item' card. The menu includes options like 'Run', 'Edit', 'Delete', 'Rename', and 'Add to my favorites' (which is highlighted with a yellow star icon). Other options like 'Add to team favorites', 'Security...', and 'Add to dashboard' are also visible.

## Favorite a delivery plan

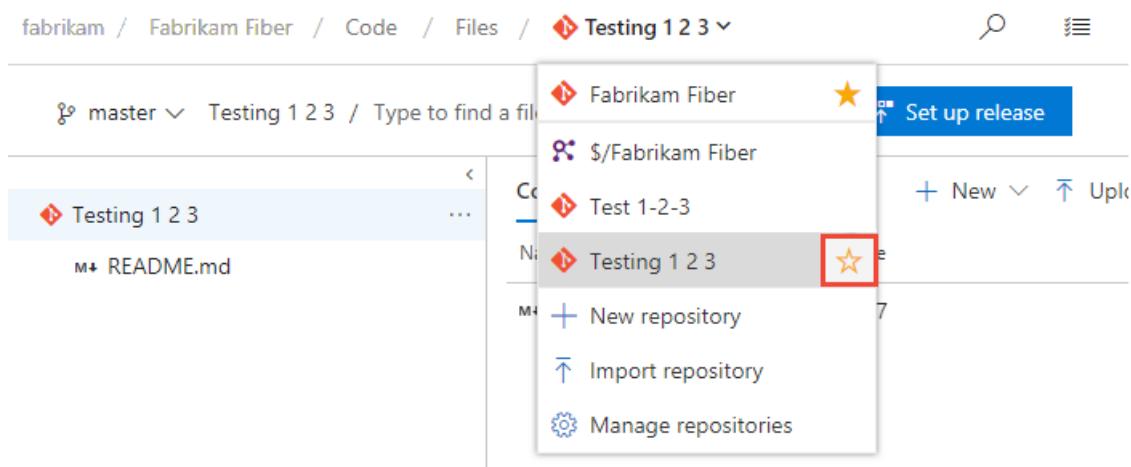
To learn more about delivery plans, see [Review team Delivery Plans](#).

To mark a delivery plan as a favorite, open the **Boards>Plans** page and choose the  star icon next to the Delivery Plan.

To mark a delivery plan as a favorite, open the **Work>Plans** page and choose the  star icon next to the Delivery Plan.

## Favorite a repository

From any **Repos** page, open the repository selector and choose the  star icon for the repository you want to favorite.



The screenshot shows the Microsoft Teams interface with the 'Repos' page selected. A context menu is open on the right side, specifically on a repository card named 'Testing 1 2 3'. The menu includes options like 'Fabrikam Fiber', '\$/Fabrikam Fiber', 'Test 1-2-3', and 'Testing 1 2 3' (which is highlighted with a red box around its star icon). Other options like 'New repository', 'Import repository', and 'Manage repositories' are also visible.

From any **Code** page, open the repository selector and choose the star icon next to the repository you want to favorite.

The screenshot shows the Azure DevOps 'Code' interface. At the top, there's a navigation bar with 'Fabrikam Fiber' selected. Below it, a sidebar on the left lists repositories: 'Favorites', 'All repositories' (which is selected), 'Filter repositories', '\$/Fabrikam Fiber', 'Fabrikam Fiber' (with a yellow star icon highlighted by a red box), 'Testing 1 2 3', '+ New repository', 'Import repository', and 'Manage repositories'. The main area shows a list of files under 'Fabrikam Fiber': 'page-1.md' (modified, 10/15/2015, commit 3458a6c7), 'page-2.md' (modified, 10/15/2015, commit 01a447ca), 'page-3.md' (modified, 9/21/2016, commit 68385e28), and 'README.md' (modified, 5/19/2017, commit fb9177d8). The 'Contents' tab is selected.

## Favorite a build pipeline

Open **Pipelines>Builds** and choose either **Mine** or **Definitions** page. Choose the star icon next to the build definition you want to favorite. Or, open the context menu of the build definition, and then select **Add to my favorites** or **Add to team favorites**.

The screenshot shows the Azure DevOps 'Pipelines > Builds' interface. At the top, there are tabs for 'Build Definitions', 'Build ID or build number' search, '+ New', and '+ Import'. Below, there are tabs for 'Mine', 'Definitions', 'Queued', and 'XAML'. The 'Mine' tab is selected. The main area shows two build definitions: 'fabrikam build' and 'Fabrikam Fiber-CI'. A context menu is open over the 'fabrikam build' definition, with the 'Add to my favorites' option highlighted by a red box. Other options in the menu include 'Queue new build...', 'Edit definition', 'Pause', 'View builds', 'Add to team favorites >', 'Clone...', 'Export', 'Rename...', 'Save as a template...', 'Delete definition', 'Security...', and '+ Add to dashboard >'.

Open **Build and Release>Builds** and choose either **Mine** or **Definitions** page. Choose the star icon next to the build definition you want to favorite. Or, open the context menu of the build definition, and then select

Add to my favorites or Add to team favorites.

The screenshot shows the 'Build Definitions' page in the Azure DevOps interface. At the top, there are tabs for 'Mine', 'All Definitions', 'Queued', and 'XAML'. A search bar at the top right contains the placeholder 'Build ID or build number' with a magnifying glass icon. Below the tabs, a table lists builds under the columns 'Recently built', 'Status', and 'Triggered by'. One row is selected, showing a checkmark icon, a user profile picture, the name 'fabrikam build', and three dots for more options. A context menu is open next to the three dots, listing various actions: 'Queue new build...', 'Edit...', 'View definition summary', 'Add to my favorites' (which is highlighted with a red box), 'Add to team favorites', 'Clone...', 'Export', 'Rename...', 'Save as a template...', 'Delete definition', and 'Security...'.

## Favorite a test plan

To learn more about test plans, see [Create a test plan and test suite](#).

To mark a test plan as a favorite, open **Test Plans > Test Plans** and choose the star icon next to a test plan from the menu that shows All test plans.

To mark a test plan as a favorite, open the **Test > Test Plans** page and choose the star icon next to a test plan from the menu that shows All test plans.

## Unfavorite a view you've favorited

You can unfavorite an artifact from your **Favorites** page. Choose the inbox icon, and then choose **Favorites**. Choose the favorited icon of a currently favorited artifact.

The screenshot shows the Microsoft Teams interface with the 'Favorites' tab selected. The top navigation bar includes a search icon, a red-highlighted 'Favorites' icon, a file icon, and a user profile icon. Below the navigation, there are sections for 'Projects', 'Teams', 'Dashboards', 'Plans', and 'Queries'. Each section lists artifacts with their icons and favorited status (indicated by a yellow star). The 'Favorites' tab is highlighted with a red box.

Section	Artifact	Favorited
Projects	Fabrikam Fiber	★
	Phone	★
	Voice	★
Teams	Web	★
	Fabrikam Fiber Team Analytics	★
	Backlog team plans	★
Plans	Fabrikam Fiber Feature plans	★
	All items	★
	All items on all projects	★
Queries	Assigned to me	★

Similarly, you can unfavorite an artifact from the same page where you favorited it.

You can unfavorite an artifact from the **Projects > Favorites** page and choose the ★ favorited icon of a currently favorited artifact.

Similarly, you can unfavorite an artifact from the same page where you favorited it.

## Try this next

[Follow a user story, bug, issue, or other work item or pull request](#)

## Related articles

- [Manage personal notifications](#)
- [Set your preferences](#)

# Filter lists, boards, and directories

3/6/2021 • 2 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

Several applications and pages support filtering, which is very useful when a large number of artifacts or items have been defined. Most directory views provide one or more filter functions.

You can filter most items using keywords or a user name for an author of an item or where work is assigned to them. You can filter lists and boards in the following areas:

- Git repositories: Branches, Commits, Commit history, Pull Requests, Pushes, and Repositories
  - Work tracking: Work Items, Kanban boards, Backlogs, Sprint Backlogs, and Taskboards
  - Directories: Dashboards, Boards, Backlogs, Sprints, Queries, Builds, Releases
- 
- Git repositories: Branches, Commits, Commit history, Pull Requests, Pushes, and Repositories
  - Work tracking: Work Items, Kanban boards, Backlogs, Sprint Backlogs, and Taskboards

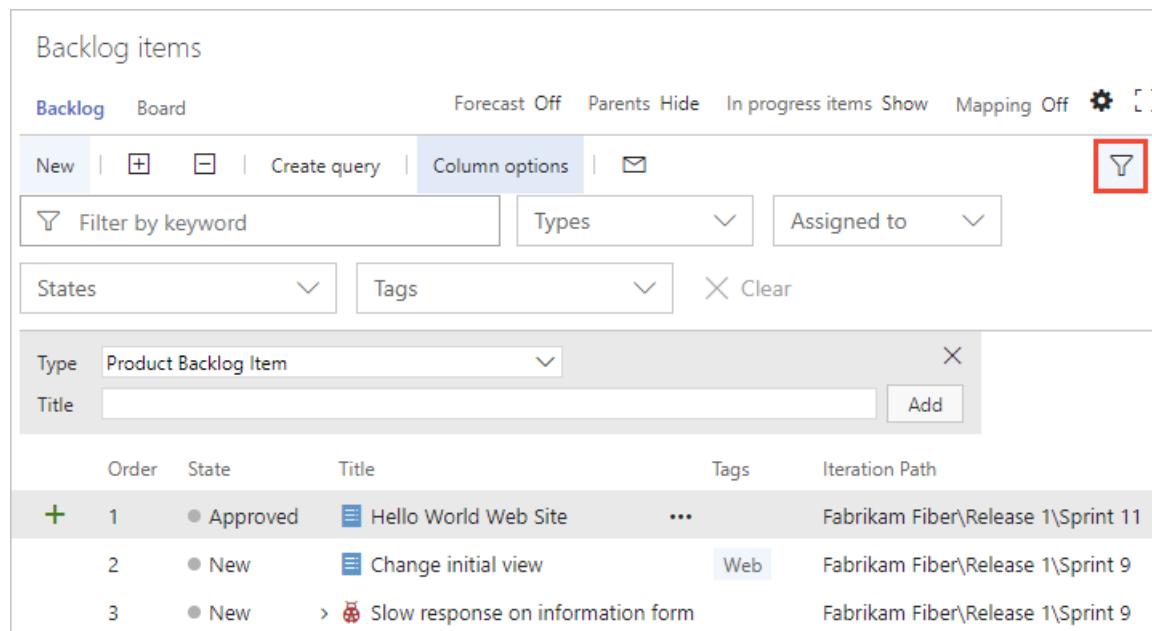
## NOTE

You may have fewer or additional filter options based on the [features you've enabled](#) or the platform and version that you are working from.

## Filter based on keywords, tags, or fields

To turn filtering on, choose the  filter icon.

You can filter work items by typing a keyword or using one or more of the fields provided, such as work item type, assigned to, state, and tags. Based on the keyword that you enter, the filter function will list work items based on any visible/displayed column or field, including tags. Also, you can enter a value for an ID, whether or not the ID field is visible.



The screenshot shows the 'Backlog items' board in the Azure DevOps interface. At the top, there's a toolbar with various buttons like 'Backlog', 'Board', 'Forecast Off', 'Parents Hide', 'In progress items Show', 'Mapping Off', and a gear icon. Below the toolbar, there's a search bar labeled 'Filter by keyword' with a magnifying glass icon, and dropdowns for 'Types' and 'Assigned to'. Further down, there are dropdowns for 'States' and 'Tags', and a 'Clear' button. A modal window is open, showing a dropdown for 'Type' set to 'Product Backlog Item' and a 'Title' input field with a 'Add' button. The main board area displays a table with columns: Order, State, Title, Tags, and Iteration Path. There are three backlog items listed:

Order	State	Title	Tags	Iteration Path
1	Approved	Hello World Web Site	...	Fabrikam Fiber\Release 1\Sprint 11
2	New	Change initial view	Web	Fabrikam Fiber\Release 1\Sprint 9
3	New	Slow response on information form		Fabrikam Fiber\Release 1\Sprint 9

The filtered set is always a flat list, even if you've selected to show parents.

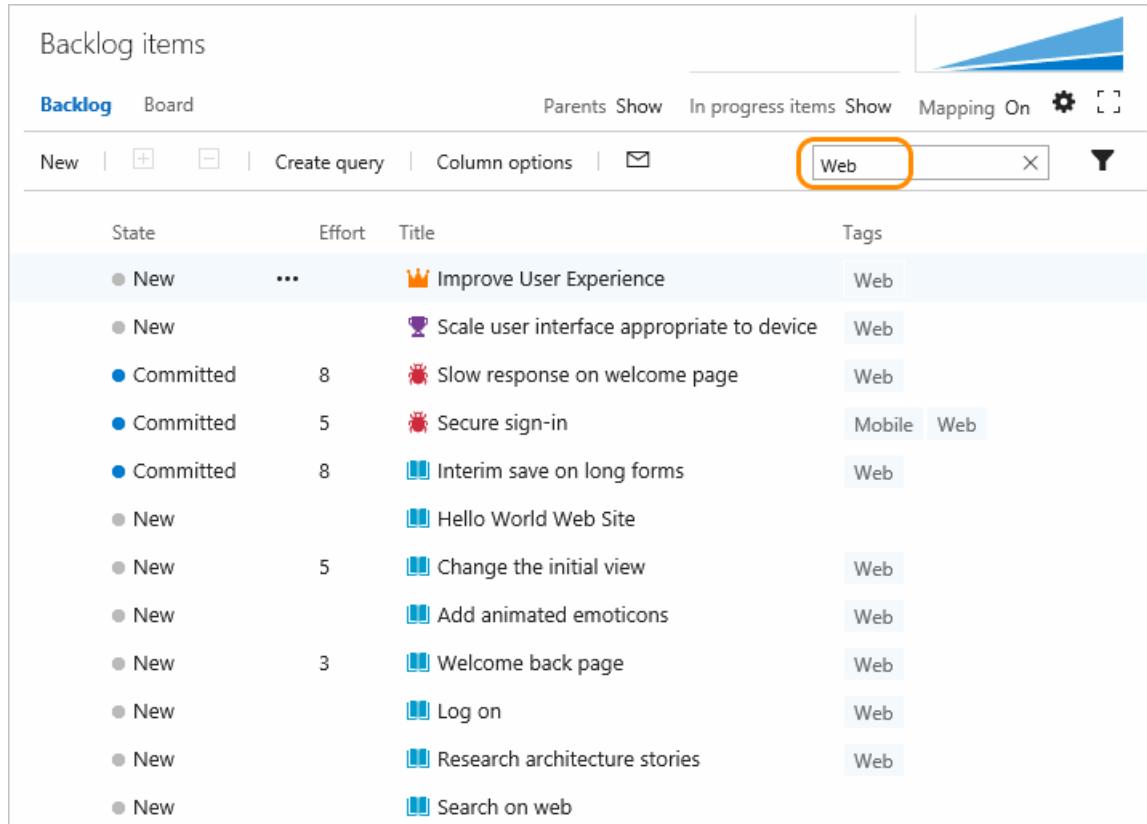
### Characters ignored by keyword filter criteria

The filter criteria ignores the following characters: `,` (comma), `.` (period), `/` (forward slash), and `\` (back slash).

## Filter work items based on keywords

You can use keywords to filter your backlogs or queries. The filter function lists those work items based on any visible/displayed column or field, including tags, based on the keyword that you enter. Also, you can enter a value for an ID, whether or not the ID field is visible.

Here, we filter the backlog to only show items that include 'Web' in any one of the displayed column fields.



The screenshot shows a Microsoft Backlog items board titled "Backlog items". At the top, there are navigation tabs for "Backlog" (which is selected) and "Board". Below the tabs are buttons for "New", "Create query", "Column options", and a search bar containing the word "Web", which is highlighted with a red oval. The main area displays a list of work items with columns for "State", "Effort", "Title", and "Tags". The "Tags" column shows that several items are associated with the "Web" tag. The items listed are:

State	Effort	Title	Tags
New	---	👑 Improve User Experience	Web
New		🏆 Scale user interface appropriate to device	Web
Committed	8	🐞 Slow response on welcome page	Web
Committed	5	🐞 Secure sign-in	Mobile Web
Committed	8	📘 Interim save on long forms	Web
New		📘 Hello World Web Site	
New	5	📘 Change the initial view	Web
New		📘 Add animated emoticons	Web
New	3	📘 Welcome back page	Web
New		📘 Log on	Web
New		📘 Research architecture stories	Web
New		📘 Search on web	

The filtered set is always a flat list, even if you've selected to show parents.

### Characters ignored by keyword filter criteria

The filter criteria ignores the following characters when the field value starts with the character:

`{, (, [, !, @, #, $, %, ^, &, *, ~, `,, ',, "`.

## Filter work tracking backlogs and queries based on tags

If you've [added tags to your work items](#), you can filter your backlogs, Kanban boards, and query results using the  tag filter. For backlogs and query results, add Tags as a column option prior to filtering on tags.

To learn more about filtering using tags, see [Add tags to work items to categorize and filter lists and boards](#), [Filter lists using tags](#)

## Filter directories

Choose the  filter icon to filter a directory list by keyword, team, or other supported field. Keywords apply to titles, descriptions, and team names.

For example, here we turn filtering on for the dashboard directory.

The screenshot shows the 'Dashboards' page in the Dynamics 365 interface. At the top right, there is a red box highlighting the filter icon (a magnifying glass) and the '+ New Dashboard' button. Below the header, there is a search bar labeled 'Filter by text' and a dropdown menu set to 'Team'. A modal window is open, titled 'Team', with a search bar containing a placeholder 'Search...'. The modal lists several items with checkboxes:

- Account Management
- Customer Profile
- Fabrikam Team
- Phone
- Service Delivery
- Service Status
- Shopping Cart
- TV

At the bottom right of the modal is a 'Clear' button.

The main content area displays a list of dashboards categorized by team:

Name	Team
Continue where you left off	
Analytics	Fabrikam Team
My favorite dashboards (2)	
Overview	Account Management
Team Guidance	Fabrikam Team
Account Management (1)	
Overview	Account Management
Customer Profile (1)	
Overview	Customer Profile
Fabrikam Team (5)	
Analytics	Fabrikam Team

## Related articles

- [Commit history](#)
- [Working with Git tags](#)
- [Filter backlogs and queries](#)
- [Filter your Kanban board](#)
- [Add tags to work items](#)

# Get started with semantic search

4/21/2021 • 8 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

The Search function for Azure DevOps enables you to easily search across all the projects, teams, and repositories to which you have access.

With semantic search, you can quickly find work items, code files, wiki pages, or packages based on a keyword, wildcards, and other supported semantic search filters.

With semantic search, you can quickly find work items and code files based on a keyword, wildcards, and other supported semantic search filters.

You can find an at-a-glance look at all of the [semantic search features](#) further in this article.

## Prerequisites

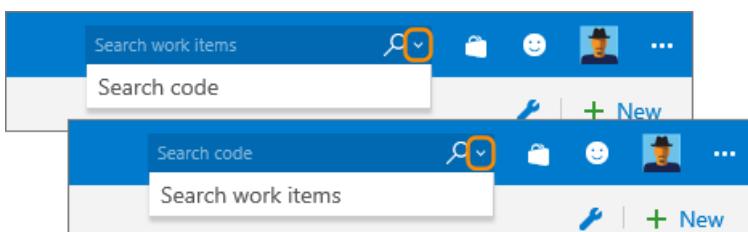
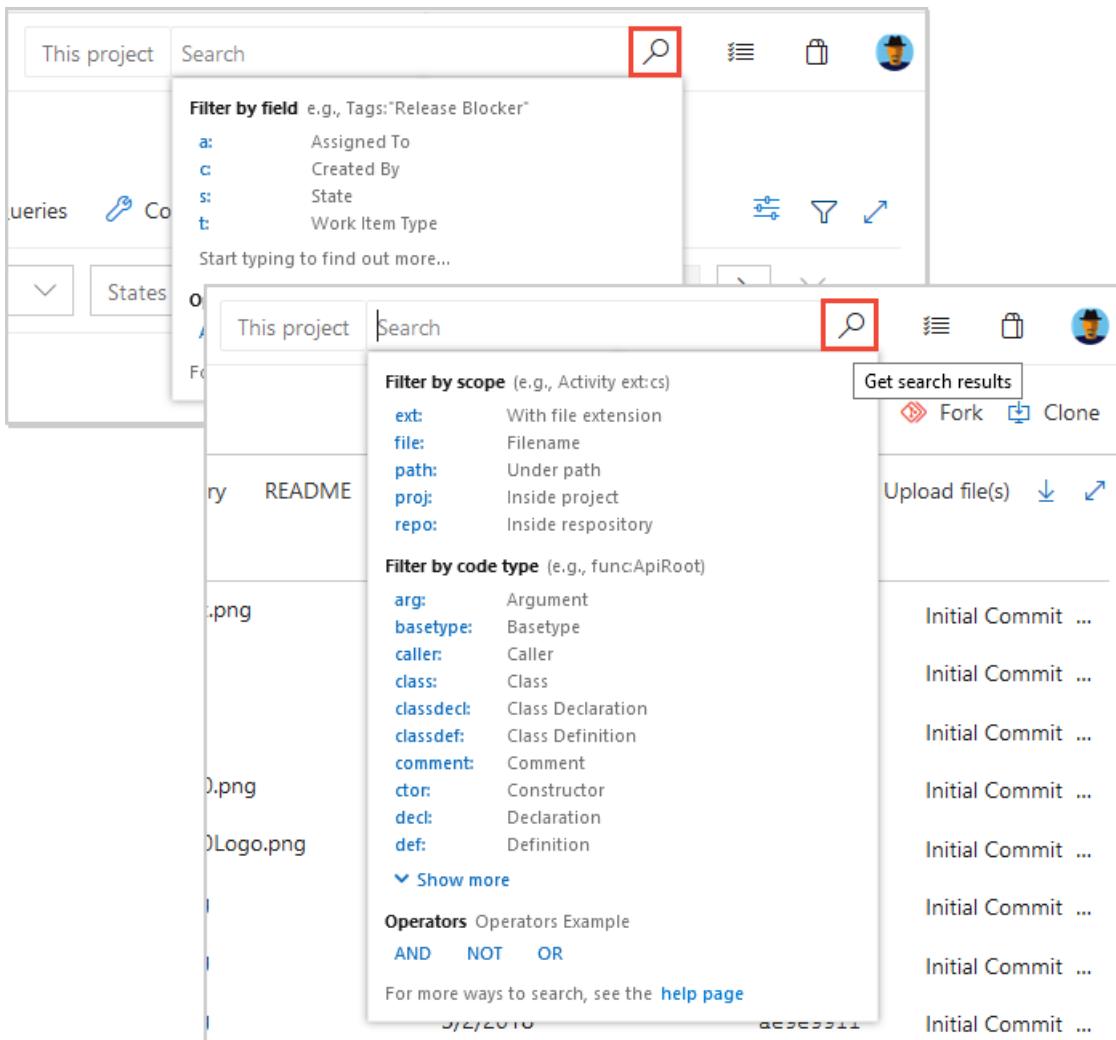
- Every project member can use the semantic search functions, including project members granted Stakeholder, Basic, and higher levels of access.
- When searching across the organization or collection, only results for which a project member has access are listed.
- Stakeholder wiki search results are limited to provisioned wikis. Because published wikis require access to regular repositories, which Stakeholders don't have access to, results for published wikis don't appear in the search results. Similarly, Code search results don't appear for Stakeholders.

### NOTE

For Code search, a Collection Administrator must [install the Code Search extension](#).

## Start your search with a keyword

Start your search using a keyword. You can then apply other options, as needed, to broaden or narrow your search results.



#### TIP

- If you get no results matching the input, try removing filters and retry the search. Broadening the search and after you view the search results, you can apply appropriate filters again and search again for relevant results.
- Check for the spelling of your search terms. Currently Work item search doesn't support ignoring of users' spelling mistakes.
- If there are lots of hits when you're using a wildcard search, such as when you're using a simple wildcard search string, you may see a message that no matching files are found. In this case, narrow your search to reduce the number of matches. Specify more characters of the word or words that you want to find, or add a condition or filter to limit the number of possible matches. Searches aren't case-sensitive.

## Semantic search features, usage, and examples

The following features apply to all searches, including work items, code, wikis, and packages.

The following features apply to all searches, including work items, code, and packages.

### Search feature

## Usage

### Example

---

#### Keyword

Search based on one or more keywords.

`validate` finds instances that contain the word *validate*.

---

#### Exact match

Search based on an exact match, enclosed in double-quotes.

`"Client not found"` finds instances that contain the exact phrase match *Client not found*.

---

#### Wildcard

- Add wildcard characters, `*` and `?`, to keywords to extend the search criteria.
- Add `*` at the end of a keyword to find items that start with the keyword.
- Add `?` in the middle to represent any alphanumeric character.
- Use wildcard characters anywhere in your search string except as a prefix. You can use prefix wildcards with the other search filter functions.
- You can use more than one wildcard to match more than one character.
- `alpha?version` finds instances of alpha1version and alphaXversion.
- `Browser*` finds instances of BrowserEdge, BrowserIE, and BrowserFirefox.
- `CodeSenseHttp*` finds files containing words that start with *CodeSenseHttp*, such as CodeSenseHttpClient and CodeSenseHttpClientTest.

---

#### Boolean operators

- Find two or more keywords using Boolean operators: `AND`, `OR`, and `NOT` (must be uppercase).
- Add parenthesis to clauses to support logical groupings.
- Because `AND` is the default operator, an entry of two keywords with no operator is the same as an `AND` search.
- `Validate AND revisit` finds files that contain both the words *validate* and *revisit*.
- `Validate OR revisit` finds files that contain either of the words *validate* or *revisit*.
- `Validate NOT revisit` finds files that contain the word *validate* but not the word *revisit*.
- `(Validate NOT revisit) OR "release delayed"` finds files that contain the word *validate* but not the word *revisit* or files that contain the phrase *release delayed*.

---

#### Proximity

- Search for files based on vicinity using proximity operators: NEAR, BEFORE, and AFTER (must be uppercase).
- By default, proximity search looks for terms within five tokens distance.
- `term1 BEFORE term2` returns all files where term1 occurs BEFORE term2 within a distance of five tokens between them.
- `term1 AFTER term2` returns the same results as term2 BEFORE term1.
- `term1 NEAR term2` returns all files where term1 is within five token distance from term2 in any direction.  
`term1 NEAR term2` returns the same results as `term1 BEFORE term2 OR term2 BEFORE term1`.

## Special characters

- Escape the special characters `(`, `)`, `[`, `]`, `:`, `*`, and `?` by enclosing them in a phrase delimited with double-quotes.
- Include special characters in a search string, or search specifically for special characters, according to the following rules:
  - CodeA23?R finds files containing words that start with CodeA23
  - Have any alphanumeric character next, and end with R. For example, CodeA234R and CodeA23QR.
  - Search for any special character that isn't a part of the query language.
- `"flatten()"` finds the literal string *flatten()*. Search for a literal occurrence of the double-quote character " by preceding it with the escape character `\` and enclosing the search string in double-quotes.
- `"\"react-redux\""` finds the literal string "react-redux".

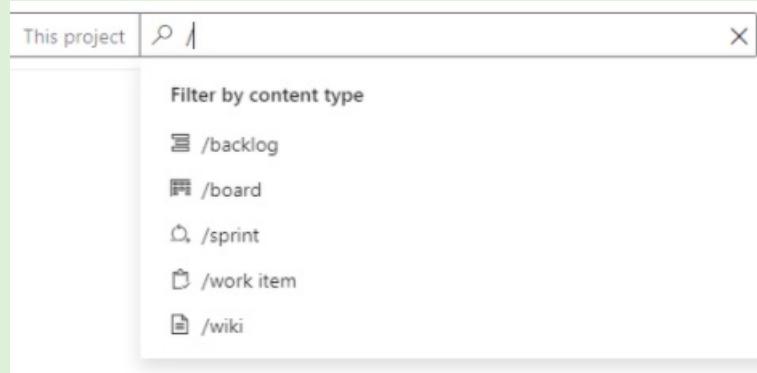
## Choose your semantic search starting page

You can start a search from one of the following pages:

- [Projects](#) page for the organization, starts a search across all projects.
- [Project](#) overview page, automatically applies a filter to search within the selected project.
- [Boards](#) page for a project, automatically displays recent work items and backlogs accessed by the user.
- [Repos](#), [Pipelines](#), [Test Plans](#), or an [Artifacts](#) page for a project, automatically displays functional filters for code searches.
- [Wiki](#) page for a project, automatically access recent wiki pages access by the user.

### TIP

Use the content type filter to access a page that you recently opened.



## Start your search from the Projects page

From your organization's Overview page, enter a keyword or phrase in the semantic search, and then select **Enter** or choose start search.

This project  Elasticsearch 

Work items

/ork Items 

- Elasticsearch stops functioning intermittently #1398199
- Load Balanced Elasticsearch for Azure DevOps Server 2019 Update 1 #1680710
- DTS: TFS 2017.2 - Code search indexing service is not working properly since the password... #121
- [On-Prem][Dave Zimmerman] Elastic search not returning results. #1100178
- [Customer Issue] Code and work item searches failing in TFS 2018.2 after upgrade from TF... #127

Assigned To

...  Anirudh  View code files (1)

 Deborah  View wiki pages (1)

 Unass  View packages (0)

## Start your search from the Project-Overview page

From your project's Overview page, enter a keyword or phrase in the semantic search, and then select **Enter** or choose  start search.

This project  Elasticsearch 

Wiki Pages

 Elasticsearch

rod through jit  
reDevOps.Conf

rogress?release  
ed on July 24th

 View all 184 wiki pages

 View code files (50+)

 View work items (50+)

 View packages (30)

## Start your search from a Boards page

Start searching across all your work items over all your projects with a keyword or phrase. Work item search includes all work item types, including test-related and custom work item types.

1. Choose any **Boards** page, enter a keyword or phrase in the semantic search, and select **Enter** or choose  start search.

This project  update wiki 

2. Search results display in a snippet view where the matches found are shown in bold.

This screenshot shows the search results for the term "check". The search bar at the top contains "This project check". Below the search bar, there are filters for "Area: Fabrikam Fiber", "Types: All", and "States: All". The results section displays 6 work item results:

- 549 Check performance** (Raisa Pokrovskaya, New)
- 375 Check service status** (Jamal Hartnett, Done)
- 559 Check out** (Unassigned, Done)
- 487 Check issues with permissions** (Raisa Pokrovskaya, Committed)
- 538 Design welcome screen** (Jamal Hartnett, In Progress)
 

Description: Review UI, simplify, check that it meets all standards.
- 578 559 : Check out**

The right side of the interface shows a detailed view of the first result, "549 Check performance". It includes fields for State (New), Reason (New task), Area (Fabrika), Iteration (Fabrika), and buttons for Details and Relate. A "Description" section with a placeholder "Click to add Description" is also present.

This full text search uses simple search strings for words or phrases. Work item search matches derived forms of your search terms; for example, a search for "check" also finds instances of the word "checked" and "checking".

3. Select a snippet of a work item to display it in the window on the right side of your screen.
4. Open the search results in a new browser tab from the semantic search: Select **Ctrl + Enter** or hold **Ctrl** and select start search. In Google Chrome, select **Ctrl + Shift + Enter** to switch the focus to the new browser tab.

1. In the semantic search, check that the text says *Search work items*. If it doesn't, use the selector to select it.

The search bar at the top contains "Search work items". The "Relevance" dropdown menu is highlighted with a red box.

2. Enter a search string in the text box, and select **Enter** or start search.
3. Search results display in a snippet view where the matches found are shown in bold.

The search bar at the top contains "login". The results section shows 3 work item results:

- 62 Login page** (Resolved)
 

Description: Login page
- 119 Login behaviour for booking** (Closed)
 

Description: Login behaviour for booking
- 97 Login and logout behaviours** (Closed)
 

Description: Login and logout behaviours

The right side of the interface shows a detailed view of the first result, "62 Login page". It includes a "Sort by" dropdown menu with options like Relevance, Assigned To, Changed Date, Created Date, ID, State, Tags, Title, and Work Item Type. The "Relevance" option is selected and highlighted with a red box.

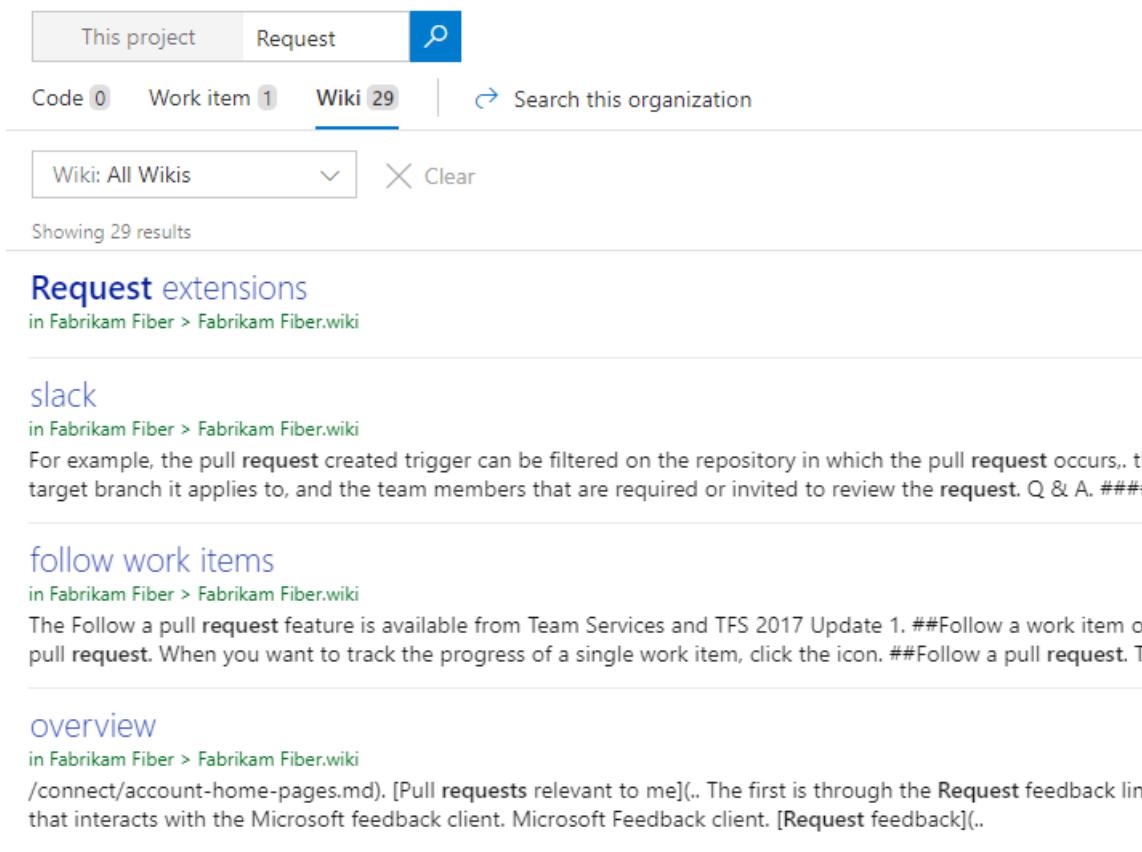
This full text search uses simple search strings for words or phrases. Work item search matches derived forms of your search terms. For example, a search for "updating" also finds instances of the word "updated" and "update". Searches aren't case-sensitive.

4. Select a snippet of a work item to display it in the right window.
5. Open the search results in a new browser tab from semantic search. Select **Ctrl + Enter** or hold **Ctrl** and select  start search. In Google Chrome, select **Ctrl + Shift + Enter** to switch the focus to the new browser tab.

For more information about searching and filtering in Azure Boards, see [Filter backlogs, boards, and plans](#).

## Start your search from a Wiki page

When you search from Wiki, you automatically navigate to wiki search results. Text search across the wiki is supported by the search platform.



The screenshot shows the Azure Boards search interface. At the top, there are tabs for "This project", "Request", and a magnifying glass icon. Below the tabs, there are counts for "Code 0", "Work item 1", and "Wiki 29", followed by a link to "Search this organization". A dropdown menu shows "Wiki: All Wikis" with a "Clear" button. Below this, it says "Showing 29 results". The results list includes items like "Request extensions" (in Fabrikam Fiber > Fabrikam Fiber.wiki), "slack" (in Fabrikam Fiber > Fabrikam Fiber.wiki), and "follow work items" (in Fabrikam Fiber > Fabrikam Fiber.wiki). Each result has a brief description and a link to the full article.

For more information about searching wikis, see [Search Wiki](#) and [Provisioned vs. published wiki](#).

### WARNING

#### No results found for ...

If there's a large number of hits when using a wildcard search, such as when using a very simple wildcard search string, you may see a message that no matching files were found. In this case, narrow your search to reduce the number of matches. For example, specify more characters of the word(s) you want to find, or add a condition or filter to limit the number of possible matches.

## Additional search functions

If you want to search for various settings, users, projects, and more, see the following table to find non-semantic search tasks and corresponding actions.

---

## Non-semantic search task

### Action

---

#### Find an organization setting

Go to your organization and select **Organization settings**.

---

#### Find a project setting

Go to your project and select **Project settings**.

---

#### Find a user setting

Go to your **User settings page**.

---

#### Find a user

Go to your organization and select **Organization settings > Users**, and then enter the name in the filter box.

---

#### Find an organization

Scroll through the left side of your screen, which lists all organizations.

---

#### Find a project

Go to your organization, and then enter the project name in the Filter projects box.

---

#### View file history and compare versions

Go to **Repos > Files**, highlight your file, and then select **History**.

---

#### Find wiki content

Go to your wiki and enter your semantic search.

---

#### NOTE

The organization settings search function finds all settings, both organization and project.

## Search re-index requirements

Search for Azure DevOps Server has the following limitation:

- If you do a disaster recovery (DR) operation and move your server back to an earlier snapshot of your SQL database, [reindex all your collections](#).

## Marketplace extensions

- [Code Search](#) - Extends semantic search with fast, flexible, and precise search results across all your code. Required for searching repositories.
- [Azure Paths Search](#) - Adds a special search hub to Boards for searching within iterations and area paths without having to create and maintain custom queries.

**NOTE**

Some extensions aren't supported features of Azure DevOps and therefore aren't supported by the product team. For questions, suggestions, or issues you have when using these extensions, visit their corresponding extension page on the [Visual Studio Marketplace](#).

## Next steps

[Functional work item search](#) or [Functional code search](#) or [Functional artifact or package search](#)

## Related articles

- [Code Search blog posts](#)
- [Work item search blog posts](#)
- [Search wiki](#)

# Manage or enable features

5/8/2021 • 4 minutes to read • [Edit Online](#)

## Azure DevOps Services | Azure DevOps Server 2020

As new features are introduced, you can turn them on or off. That way, you can try them out, provide feedback, and work with those features that meet your requirements.

Some preview features provide access to entire new functionality. Others, such as the New Wiki experience, reflect a change to the user interface, but little or no change in functionality.

### NOTE

It may take up to three weeks after a release to Azure DevOps Services for the preview feature to appear in your organization. The [latest release notes](#) usually provide information on new preview features. You can turn on or off select features for Azure DevOps Services. Preview features become available first on Azure DevOps Services and then become standard features with an update to Azure DevOps Server. At some point, the preview feature moves out of preview status and becomes a regular feature of the web portal.

There are a few features you or an administrator can enable or disable. Some features provide access to entire new functionality, while others provide a change to the user interface.

The following table indicates which preview features can be enabled per user or team member, and those that can be enabled for the organization. You must be a member of the Project Collection Administrators group to change a preview feature at the organization-level.

### Preview features

#### Per user

#### Per organization

##### [Analytics Views](#)

[Enable group by tags for work item chart widget on dashboard](#)

- ✓
- ✓
- ✓
- ✓

##### [Experimental themes](#)

[Full Access to Azure Pipelines for Stakeholders](#)

[Git Forks](#)

[Historical graph for agent pools](#)

[New account manager](#)

- ✓
- ✓
- ✓

- 

## New boards reports

## New release progress views

## New Repos pull request experience

## New service connections experience

- ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓

## New Settings Search in the organization settings panel

## New Teams page

## New Test Plans Page

## New TFVC pages

## New Wiki experience

- ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓

Organization Permissions Settings Page v2

## Project Permissions Settings page

## Limit user visibility for projects

## Task Insights for Failed Pipeline Runs

- ✓ ✓ ✓ ✓ ✓ ✓ ✓

The following table indicates those features that you can enable as a user, project administrator, or project collection administrator.

administrator.

---

## Feature

User

Project

Collection

---

[New service connections experience](#)

[Selective artifacts download feature for collection/project](#)



---

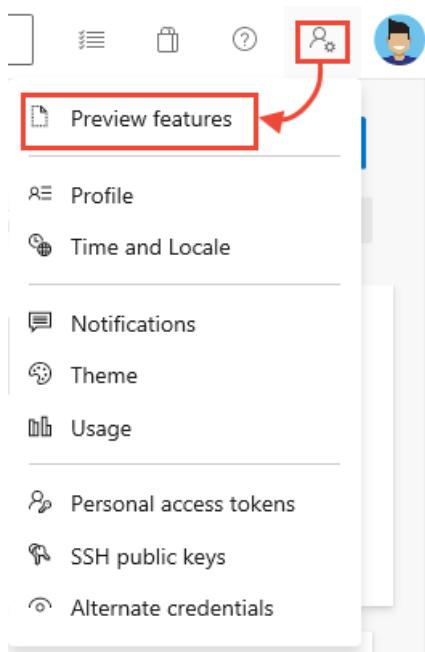
## Enable features for your use

From time to time, a new feature is introduced in Preview mode, which allows you to turn it on or off.

To access the Preview features options, open your profile menu. The profile menu appears as shown below based on whether the **New Account Manager** feature has been enabled or not.

- [New Account Manager enabled](#)
- [New Account Manager not enabled](#)

Choose the profile icon, and then choose **Preview features**.



To enable or disable a feature, choose the slider.

## Preview features

X

The following preview features are available for your evaluation. Help us make them better!

for me [Jamal Hartnett]

### Analytics Views

On

Enable a hub for creating data sets to use for custom reporting.

### Dependency Tracker Preview Features

Off

Enable Dependency Tracker preview features

### Experimental Themes

On

Adds an early preview of various themes to the Theme management panel.

### New account manager

On

Enables the new account manager which allows users to switch accounts easily and have quicker access to help menu options from the top navigation bar

### New boards reports

On

Turn on interactive reports in boards pages, replacing CFD, Velocity and Burndown charts in boards headers.

### New release progress views

On

Turn on the new release views to visualize the progress of your deployment pipelines. [Learn more](#)

### New Repos pull request experience

On

New pull request, commit details, and branch comparison pages within Repos. Modern, fast, and mobile-friendly. [Learn more](#)

### New Repos settings experience

On

New settings, policies and permissions pages within Repos.

### New service connections experience

On

Turns on the new service connections experience.

### New Settings Search in the organization settings panel

On

Enables settings search

### New Teams Page

On

Enables new teams page

### New Test Plans Page

On

Lights up the new Test Plans page which offers streamlined views, new UX and additional capabilities.

### New TFVC pages

On

New web pages for TFVC repositories. Modern, fast, and mobile-friendly.

### New Wiki Experience

On

Enables new wiki experience in the wiki hub.

### Organization Permissions Settings Page v2

On

Lights up version 2 of the permissions tab

### Project Permissions Settings page

On

Lights up new project permissions hub

### Task Insights for Failed Pipeline Runs

On

Enables insights toast for the failed tasks in Pipeline Run.

For information on other user settings and preferences, see [Set user preferences](#).

Enable features at the organization level

When you enable a feature at the organization level, you essentially turn it on for all users of your account. Each user can then disable the feature if they so choose. If you disable a feature at the organization level, user settings are not changed. Users can enable or disable the feature on their own.

#### TIP

If you don't see the **for this account** menu option, then you aren't a member of the Project Collection Administrators group. To get added as one, see [Add administrators, set permissions at the team project or collection level](#).

## Preview features



The following preview features are available for your evaluation. Help us make them better!

**for this account [fabrikam]**

### Analytics Views

On

Enable a hub for creating data sets to use for custom reporting.

### Enable group by tags for work item chart widget on dashboard

On

Enable group by tags for work item chart widget on dashboard

### Experimental Themes

Off

Adds an early preview of various themes to the Theme management panel.

### Full access to Azure Pipelines for Stakeholders

On

Gives users with the Stakeholder license full access to Azure Pipelines for private projects. Limit what they can do by using security groups and permissions. Turning on this feature doesn't affect public projects, where Stakeholders always have full access. [Learn more](#)

### Git Forks

On

Enable git repositories to be forked. [Learn more](#)

### Historical graph for agent pools

Off

Turn on the new historical graph for pools feature. [Learn more](#)

### New account manager

On

Enables the new account manager which allows users to switch accounts easily and have quicker access to help menu options from the top navigation bar

### New boards reports

On

Turn on interactive reports in boards pages, replacing CFD, Velocity and Burndown charts in boards headers.

### New Delivery Plans Experience

Off

Preview the new delivery plans experience in the work items hub.

### New release progress views

On

Turn on the new release views to visualize the progress of your deployment pipelines. [Learn more](#)

### New Repos pull request experience

On

New pull request, commit details, and branch comparison pages within Repos. Modern, fast, and mobile-friendly. [Learn more](#)

### New service connections experience

On

Turns on the new service connections experience.

### New Settings Search

Off

Enables settings search

### New Teams Page

On

Enables new teams page

### New TFVC pages

On

New web pages for TFVC repositories. Modern, fast, and mobile-friendly.

### New Wiki Experience

On

Enables new wiki experience in the wiki hub.

### Organization Permissions Settings Page v2

On

Lights up version 2 of the permissions tab

### Project Permissions Settings page

On

Lights up new project permissions hub

### Limit user visibility for projects

On

Enables well known group Project-Scope Users

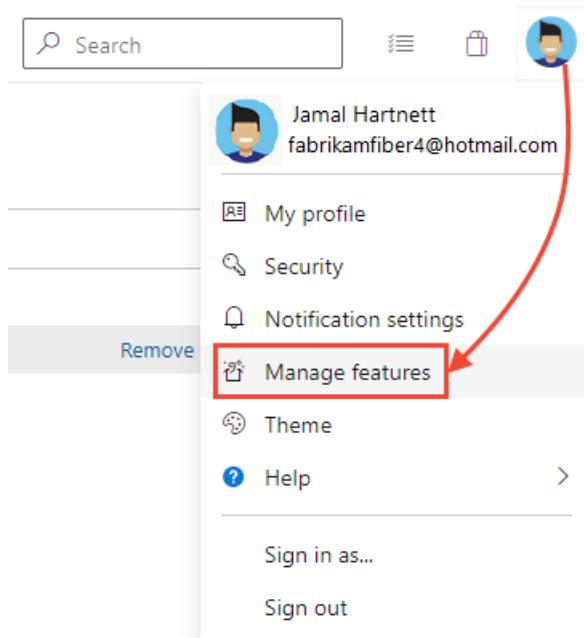
### Task Insights for Failed Pipeline Runs

On

Enables insights toast for the failed tasks in Pipeline Run.

## Enable or disable a feature

1. Open your profile menu by choosing your image icon and select **Manage features**.



2. Select the level from the menu provided.

**TIP**

If you don't see the **for this project** or **for this collection** menu options, then you aren't an administrator. To get added as one, see [Add administrators, set permissions at the team project or collection level](#).

3. To enable or disable a feature, choose the slider.

**User-level**

**Manage features** X

The following features are optional and can be turned on or off.

for me [Jamal.Hartnett] ▼

New service connections experience  On

Turns on the new service connections experience.

**Project-level**

**Manage features** X

The following features are optional and can be turned on or off.

for this project [FabrikamBasic] ▼

Selective artifacts download feature for collection/project  Off

Turns on selective artifacts download feature in release pipelines.  
This feature isn't supported if you have agents behind proxy in your release pipelines.

**Collection-level**

## Manage features



The following features are optional and can be turned on or off.

for this collection [DefaultCollection] ▼

### New service connections experience

On

Turns on the new service connections experience.

### Selective artifacts download feature for collection/project

Off

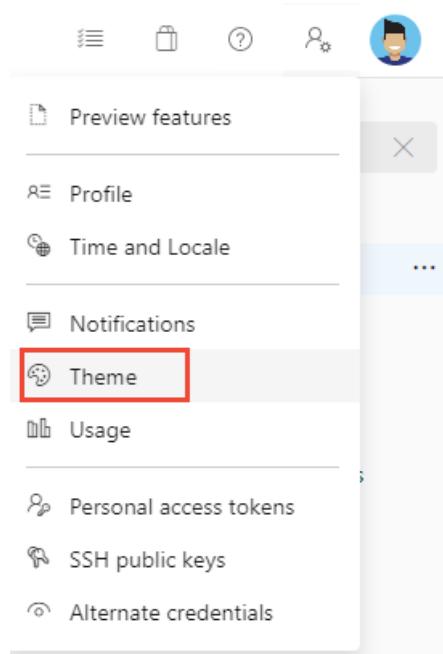
Turns on selective artifacts download feature in release pipelines.

This feature isn't supported if you have agents behind proxy in your release pipelines.

When you enable a feature at the project or collection-level, you essentially turn it on for all users. If you disable a feature at the project or collection-level, user settings are not changed. Users can enable or disable the feature on their own.

## Experimental themes

When you select **Theme** from the Profile menu you can select between **Dark** and **Light** themes for the display of Azure DevOps web portal.

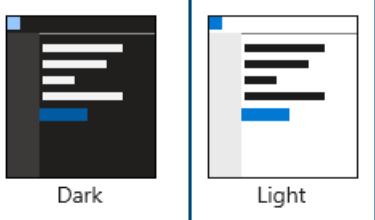


With **Experimental themes** enabled, you can select among a number of additional themes.

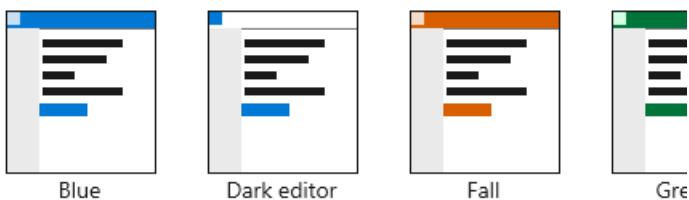
## Choose your theme

X

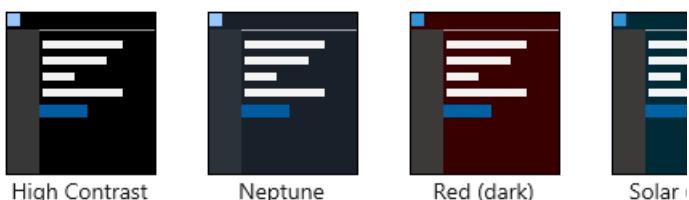
### Default themes



### Custom themes



### Preview themes



## Features now enabled for all Azure DevOps Services

### General

- [New user hub](#)
- [New PAT experience](#)
- [New Navigation](#)

### Azure Pipelines

- [Pipeline decorators](#)
- [Multi-stage pipelines](#)
- [Test tab in new web platform](#)
- [Test analytics in new web platform](#)
- [New builds hub](#)
- [Build with multiple queues](#)
- [New Releases Hub](#)
- [Approval gates in releases - New Release Definition Editor](#)
- [Symbol server](#)
- [Task tool installers](#)

### Azure Boards

- [New Delivery Plans Experience](#)
- [New Rich Text Editor](#)
- [New Queries Experience](#)
- [New Work Items](#)

### Azure Repos

- New Repos settings experience
- New Repos landing pages
- Pull Request Status Policy

## Azure Artifacts

- NuGet.org upstream sources
- Updated package experience

## Azure Test Plans

- New Test Plan Experience

## Dashboards and Analytics

- Analytics Views
- New Dashboards Experience

## Social tools

- Wiki
- Combine email recipients
- New experience in Code, Work Item, & Wiki search
- Out of the box notifications
- Team expansion for notifications

## Organization, project, and billing management

- Streamlined User Management

## Related articles

- Set user preferences
- Azure DevOps Feature Timeline

# Get started with semantic search

4/21/2021 • 8 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

The Search function for Azure DevOps enables you to easily search across all the projects, teams, and repositories to which you have access.

With semantic search, you can quickly find work items, code files, wiki pages, or packages based on a keyword, wildcards, and other supported semantic search filters.

With semantic search, you can quickly find work items and code files based on a keyword, wildcards, and other supported semantic search filters.

You can find an at-a-glance look at all of the [semantic search features](#) further in this article.

## Prerequisites

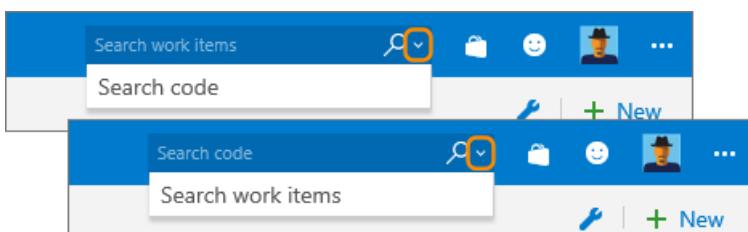
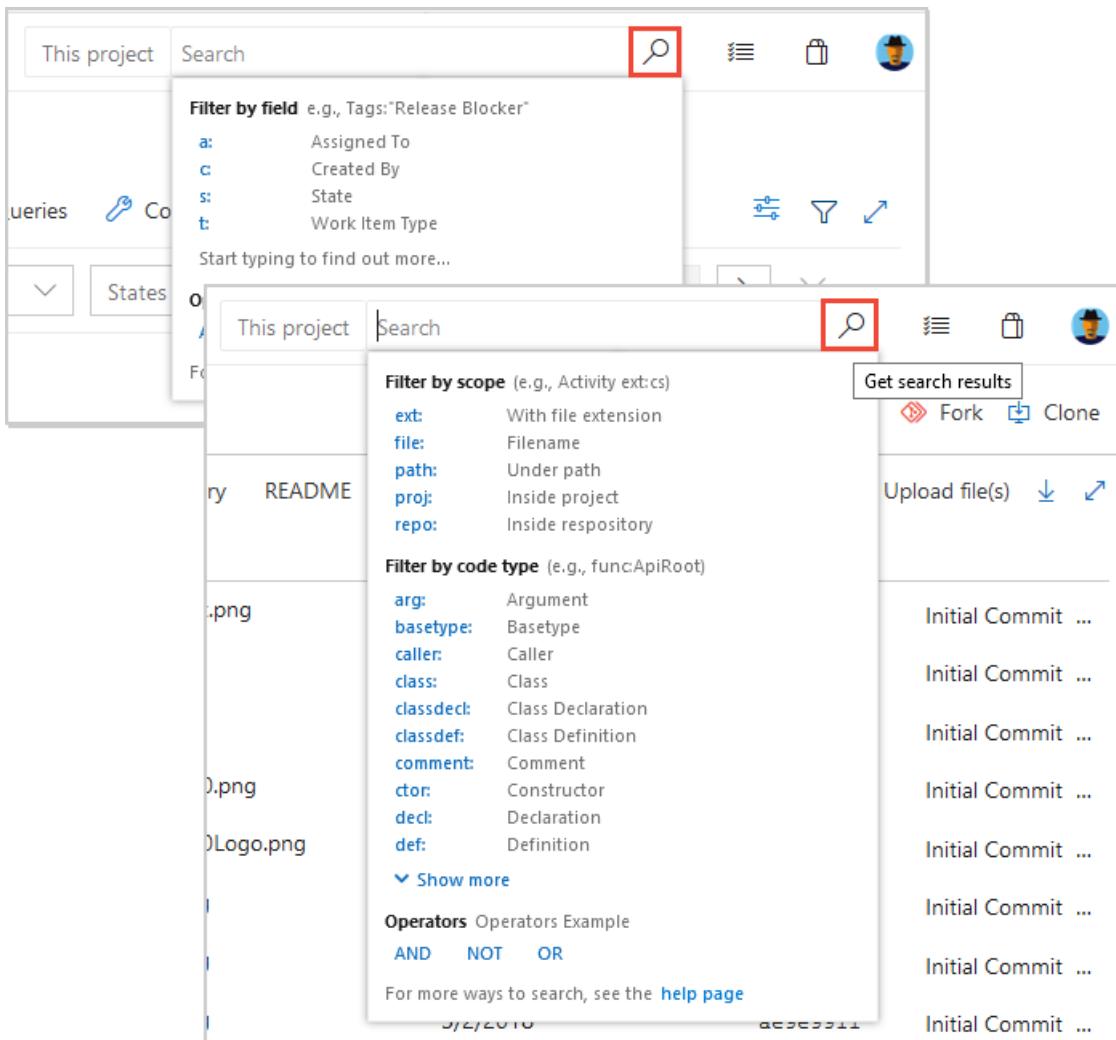
- Every project member can use the semantic search functions, including project members granted Stakeholder, Basic, and higher levels of access.
- When searching across the organization or collection, only results for which a project member has access are listed.
- Stakeholder wiki search results are limited to provisioned wikis. Because published wikis require access to regular repositories, which Stakeholders don't have access to, results for published wikis don't appear in the search results. Similarly, Code search results don't appear for Stakeholders.

### NOTE

For Code search, a Collection Administrator must [install the Code Search extension](#).

## Start your search with a keyword

Start your search using a keyword. You can then apply other options, as needed, to broaden or narrow your search results.



#### TIP

- If you get no results matching the input, try removing filters and retry the search. Broadening the search and after you view the search results, you can apply appropriate filters again and search again for relevant results.
- Check for the spelling of your search terms. Currently Work item search doesn't support ignoring of users' spelling mistakes.
- If there are lots of hits when you're using a wildcard search, such as when you're using a simple wildcard search string, you may see a message that no matching files are found. In this case, narrow your search to reduce the number of matches. Specify more characters of the word or words that you want to find, or add a condition or filter to limit the number of possible matches. Searches aren't case-sensitive.

## Semantic search features, usage, and examples

The following features apply to all searches, including work items, code, wikis, and packages.

The following features apply to all searches, including work items, code, and packages.

### Search feature

## Usage

### Example

---

#### Keyword

Search based on one or more keywords.

`validate` finds instances that contain the word *validate*.

---

#### Exact match

Search based on an exact match, enclosed in double-quotes.

`"Client not found"` finds instances that contain the exact phrase match *Client not found*.

---

#### Wildcard

- Add wildcard characters, `*` and `?`, to keywords to extend the search criteria.
  - Add `*` at the end of a keyword to find items that start with the keyword.
  - Add `?` in the middle to represent any alphanumeric character.
  - Use wildcard characters anywhere in your search string except as a prefix. You can use prefix wildcards with the other search filter functions.
  - You can use more than one wildcard to match more than one character.
  - `alpha?version` finds instances of alpha1version and alphaXversion.
  - `Browser*` finds instances of BrowserEdge, BrowserIE, and BrowserFirefox.
  - `CodeSenseHttp*` finds files containing words that start with *CodeSenseHttp*, such as `CodeSenseHttpClient` and `CodeSenseHttpClientTest`.
- 

#### Boolean operators

- Find two or more keywords using Boolean operators: `AND`, `OR`, and `NOT` (must be uppercase).
  - Add parenthesis to clauses to support logical groupings.
  - Because `AND` is the default operator, an entry of two keywords with no operator is the same as an `AND` search.
  - `Validate AND revisit` finds files that contain both the words *validate* and *revisit*.
  - `Validate OR revisit` finds files that contain either of the words *validate* or *revisit*.
  - `Validate NOT revisit` finds files that contain the word *validate* but not the word *revisit*.
  - `(Validate NOT revisit) OR "release delayed"` finds files that contain the word *validate* but not the word *revisit* or files that contain the phrase *release delayed*.
- 

#### Proximity

- Search for files based on vicinity using proximity operators: NEAR, BEFORE, and AFTER (must be uppercase).
  - By default, proximity search looks for terms within five tokens distance.
  - `term1 BEFORE term2` returns all files where term1 occurs BEFORE term2 within a distance of five tokens between them.
  - `term1 AFTER term2` returns the same results as `term2 BEFORE term1`.
  - `term1 NEAR term2` returns all files where term1 is within five token distance from term2 in any direction.  
`term1 NEAR term2` returns the same results as `term1 BEFORE term2 OR term2 BEFORE term1`.
-

## Special characters

- Escape the special characters `(`, `)`, `[`, `]`, `:`, `*`, and `?` by enclosing them in a phrase delimited with double-quotes.
- Include special characters in a search string, or search specifically for special characters, according to the following rules:
  - CodeA23?R finds files containing words that start with CodeA23
  - Have any alphanumeric character next, and end with R. For example, CodeA234R and CodeA23QR.
  - Search for any special character that isn't a part of the query language.
- `"flatten()"` finds the literal string *flatten()*. Search for a literal occurrence of the double-quote character " by preceding it with the escape character `\` and enclosing the search string in double-quotes.
- `"\"react-redux\""` finds the literal string "react-redux".

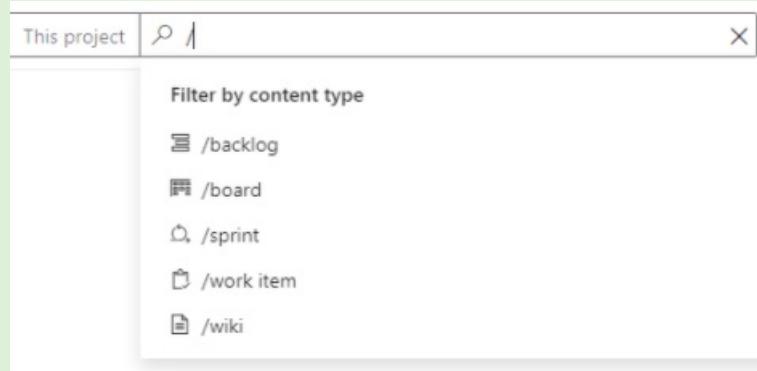
## Choose your semantic search starting page

You can start a search from one of the following pages:

- [Projects](#) page for the organization, starts a search across all projects.
- [Project](#) overview page, automatically applies a filter to search within the selected project.
- [Boards](#) page for a project, automatically displays recent work items and backlogs accessed by the user.
- [Repos](#), [Pipelines](#), [Test Plans](#), or an [Artifacts](#) page for a project, automatically displays functional filters for code searches.
- [Wiki](#) page for a project, automatically access recent wiki pages access by the user.

### TIP

Use the content type filter to access a page that you recently opened.



## Start your search from the Projects page

From your organization's Overview page, enter a keyword or phrase in the semantic search, and then select **Enter** or choose start search.

This project  Elasticsearch 

Work items

/ork Items 

- Elasticsearch stops functioning intermittently #1398199
- Load Balanced Elasticsearch for Azure DevOps Server 2019 Update 1 #1680710
- DTS: TFS 2017.2 - Code search indexing service is not working properly since the password... #121
- [On-Prem][Dave Zimmerman] Elastic search not returning results. #1100178
- [Customer Issue] Code and work item searches failing in TFS 2018.2 after upgrade from TF... #127

Assigned To

...  Anirudh  View code files (1)

 Deborah  View wiki pages (1)

 Unass  View packages (0)

## Start your search from the Project-Overview page

From your project's Overview page, enter a keyword or phrase in the semantic search, and then select **Enter** or choose  start search.

This project  Elasticsearch 

Wiki Pages

 Elasticsearch

rod through jit  
reDevOps.Conf  
rogress?release  
ed on July 24th

-  ElasticSearch Vulnerabilities
-  Managing ElasticSearch Cluster
-  Local Elasticsearch deployment for ALM Search
-  Disaster Recovery for ElasticSearch in Search-Service
-  OrgSearch Performance Metrics Elasticsearch
-  View all 184 wiki pages
-  View code files (50+)
-  View work items (50+)
-  View packages (30)

## Start your search from a Boards page

Start searching across all your work items over all your projects with a keyword or phrase. Work item search includes all work item types, including test-related and custom work item types.

1. Choose any **Boards** page, enter a keyword or phrase in the semantic search, and select **Enter** or choose  start search.

This project  update wiki 

2. Search results display in a snippet view where the matches found are shown in bold.

This screenshot shows the search results for the term "check". The search bar at the top contains "This project check". Below the search bar, there are filters for "Area: Fabrikam Fiber", "Types: All", and "States: All". The results section displays 6 work item results:

- 549 Check performance** (Raisa Pokrovskaya, New)
- 375 Check service status** (Jamal Hartnett, Done)
- 559 Check out** (Unassigned, Done)
- 487 Check issues with permissions** (Raisa Pokrovskaya, Committed)
- 538 Design welcome screen** (Jamal Hartnett, In Progress)
 

Description: Review UI, simplify, check that it meets all standards.
- 578 559 : Check out**

The right side of the screenshot shows a detailed view of the first result, "549 Check performance". It includes fields for State (New), Reason (New task), Area (Fabrika), Iteration (Fabrika), and Buttons for Details and Relate. The Description field is present but empty.

This full text search uses simple search strings for words or phrases. Work item search matches derived forms of your search terms; for example, a search for "check" also finds instances of the word "checked" and "checking".

3. Select a snippet of a work item to display it in the window on the right side of your screen.
4. Open the search results in a new browser tab from the semantic search: Select **Ctrl + Enter** or hold **Ctrl** and select start search. In Google Chrome, select **Ctrl + Shift + Enter** to switch the focus to the new browser tab.

1. In the semantic search, check that the text says *Search work items*. If it doesn't, use the selector to select it.



2. Enter a search string in the text box, and select **Enter** or start search.
3. Search results display in a snippet view where the matches found are shown in bold.

This screenshot shows the search results for the term "login". The search bar at the top contains "This project login". Below the search bar, there are filters for "Area: FabrikamFiber Web", "Types: All", and "Sort by: Relevance" (which is highlighted with a red box). The results section displays 3 work item results:

- 62 Login page** (Resolved)
 

Description: Login page
- 119 Login behaviour for booking** (Closed)
 

Description: Login behaviour for booking
- 97 Login and logout behaviours** (Closed)
 

Description: Login and logout behaviours

The right side of the screenshot shows a detailed view of the first result, "62 Login page". It includes fields for Assigned To, Changed Date, Created Date, ID, and a dropdown menu for Sort by. The dropdown menu lists various sorting options: Relevance, State, Tags, Title, and Work Item Type.

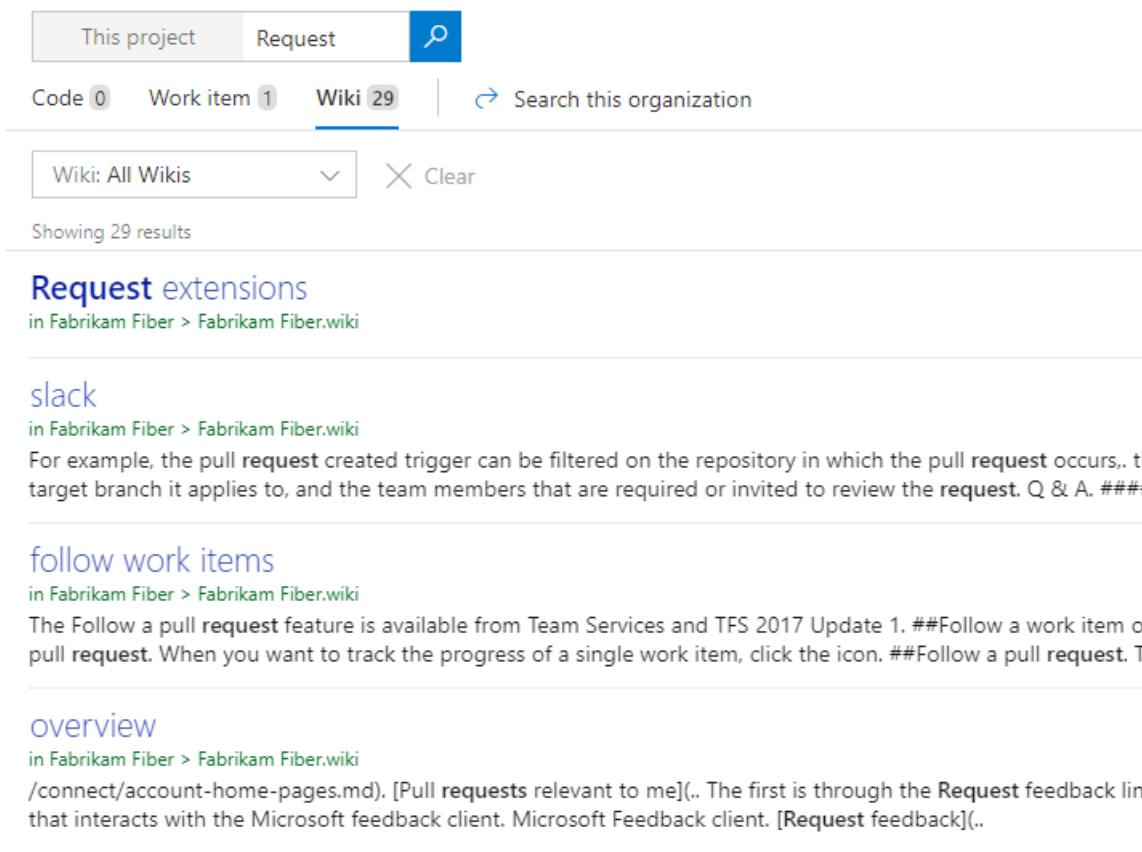
This full text search uses simple search strings for words or phrases. Work item search matches derived forms of your search terms. For example, a search for "updating" also finds instances of the word "updated" and "update". Searches aren't case-sensitive.

4. Select a snippet of a work item to display it in the right window.
5. Open the search results in a new browser tab from semantic search. Select **Ctrl + Enter** or hold **Ctrl** and select  start search. In Google Chrome, select **Ctrl + Shift + Enter** to switch the focus to the new browser tab.

For more information about searching and filtering in Azure Boards, see [Filter backlogs, boards, and plans](#).

## Start your search from a Wiki page

When you search from Wiki, you automatically navigate to wiki search results. Text search across the wiki is supported by the search platform.



The screenshot shows the Azure Boards search interface. At the top, there are tabs for "This project", "Request", and a magnifying glass icon. Below the tabs, there are counts for "Code 0", "Work item 1", and "Wiki 29", followed by a link to "Search this organization". A dropdown menu shows "Wiki: All Wikis" with a "Clear" button. Below this, it says "Showing 29 results". The results list includes items like "Request extensions" (in Fabrikam Fiber > Fabrikam Fiber.wiki), "slack" (in Fabrikam Fiber > Fabrikam Fiber.wiki), and "follow work items" (in Fabrikam Fiber > Fabrikam Fiber.wiki). Each result has a brief description and a "More" link at the end.

For more information about searching wikis, see [Search Wiki](#) and [Provisioned vs. published wiki](#).

### WARNING

#### No results found for ...

If there's a large number of hits when using a wildcard search, such as when using a very simple wildcard search string, you may see a message that no matching files were found. In this case, narrow your search to reduce the number of matches. For example, specify more characters of the word(s) you want to find, or add a condition or filter to limit the number of possible matches.

## Additional search functions

If you want to search for various settings, users, projects, and more, see the following table to find non-semantic search tasks and corresponding actions.

---

## Non-semantic search task

### Action

---

#### Find an organization setting

Go to your organization and select **Organization settings**.

---

#### Find a project setting

Go to your project and select **Project settings**.

---

#### Find a user setting

Go to your **User settings page**.

---

#### Find a user

Go to your organization and select **Organization settings > Users**, and then enter the name in the filter box.

---

#### Find an organization

Scroll through the left side of your screen, which lists all organizations.

---

#### Find a project

Go to your organization, and then enter the project name in the Filter projects box.

---

#### View file history and compare versions

Go to **Repos > Files**, highlight your file, and then select **History**.

---

#### Find wiki content

Go to your wiki and enter your semantic search.

---

#### NOTE

The organization settings search function finds all settings, both organization and project.

## Search re-index requirements

Search for Azure DevOps Server has the following limitation:

- If you do a disaster recovery (DR) operation and move your server back to an earlier snapshot of your SQL database, [reindex all your collections](#).

## Marketplace extensions

- [Code Search](#) - Extends semantic search with fast, flexible, and precise search results across all your code. Required for searching repositories.
- [Azure Paths Search](#) - Adds a special search hub to Boards for searching within iterations and area paths without having to create and maintain custom queries.

**NOTE**

Some extensions aren't supported features of Azure DevOps and therefore aren't supported by the product team. For questions, suggestions, or issues you have when using these extensions, visit their corresponding extension page on the [Visual Studio Marketplace](#).

## Next steps

[Functional work item search](#) or [Functional code search](#) or [Functional artifact or package search](#)

## Related articles

- [Code Search blog posts](#)
- [Work item search blog posts](#)
- [Search wiki](#)

# Functional code search

4/27/2021 • 7 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

Functional code search command filters extend your ability to refine your search across repositories beyond what is documented in [Get started with semantic search](#). To perform code searches, the [Code Search](#) Marketplace extension must be installed for your organization or collection.

## Prerequisites

- You must install [Code Search](#)

For more information, see [Install and configure search](#).

- To use Code Search, you must have at least a Basic access.
- Users granted Stakeholder access don't have access to code, and so don't have access to Code Search.
- Users granted Stakeholder access for a private project can perform code searches, as they have [Full Access to the code](#).
- When you're searching across the organization or collection, only results for which a project member has access are listed.

## Code search best practices

- Get the results you want even faster by starting with a higher-level search. You can narrow your search by using project, repository, path, file name, and other filter operators.
- Ensure that you get to the results you want even when you're not sure of the exact term you're looking for. [Use wildcards to widen your search](#) and [Boolean operators to fine-tune it](#).
- Find more information about an item of interest faster and with minimal efforts. When you find an item of interest, place the cursor on it and use the shortcut menu to quickly search for that text across all your projects and files.
- Easily trace how your code works by using the shortcut menu to search for related items such as definitions and references – directly from inside a file or from the search results.
- Go quickly to the implementation of, for example, an API your code might be taking dependency on by narrowing down your results to exact code type matches. Use code type filters to search for specific kinds of code such as definitions, references, functions, comments, strings, namespaces, and more.

### NOTE

You can't search code in forked repositories.

## Functions to find specific types of code

As you enter your semantic search, select functions and keywords from the drop-down list to quickly create your query. Use the **Show more** link to display all the available functions and keywords. Mix and match the functions as required.

You can also select one or a combination of filters from the list in the left column. Again, the **Show more** link displays all the available functions and keywords.

Instead, you can enter the functions and parameters directly into the semantic search. The following table shows the full list of functions for selecting specific types or members in your C#, C, C++, Java, and Visual Basic.NET code.

TO FIND CODE WHERE <i>FINDTHIS</i> APPEARS AS A ...	... SEARCH FOR ARGUMENT ARG: <i>FINDTHIS</i>
Argument	<b>arg:</b> <i>findThis</i> Deprecated in July 2019
Base type	<b>basetype:</b> <i>findThis</i>
Calling function	<b>caller:</b> <i>findThis</i> Deprecated in July 2019
Class definition or declaration	<b>class:</b> <i>findThis</i>
Class declaration	<b>classdecl:</b> <i>findThis</i> Merged with class:
Class definition	<b>classdef:</b> <i>findThis</i> Merged with class:
Comment	<b>comment:</b> <i>findThis</i>
Constructor	<b>ctor:</b> <i>findThis</i> Merged with method:
Declaration	<b>decl:</b> <i>findThis</i>
Definition	<b>def:</b> <i>findThis</i>
Destructor	<b>dtor:</b> <i>findThis</i> Merged with method:
Enumerator	<b>enum:</b> <i>findThis</i>
Extern	<b>extern:</b> <i>findThis</i> Deprecated in July 2019
Field	<b>field:</b> <i>findThis</i>
Friend function	<b>friend:</b> <i>findThis</i> Deprecated in July 2019
Function	<b>func:</b> <i>findThis</i> Merged with method:
Function declaration	<b>funcdecl:</b> <i>findThis</i> Merged with method:
Function definition	<b>funcdef:</b> <i>findThis</i> Merged with method:
Global	<b>global:</b> <i>findThis</i> Deprecated in July 2019
Header	<b>header:</b> <i>findThis</i> Deprecated in July 2019
Interface	<b>interface:</b> <i>findThis</i>
Macro	<b>macro:</b> <i>findThis</i>

TO FIND CODE WHERE <i>FINDTHIS</i> APPEARS AS A ...	... SEARCH FOR ARGUMENT ARG: <i>FINDTHIS</i>
Macro definition	<b>macrodef:</b> <i>findThis</i> Merged with macro:
Macro reference	<b>macroref:</b> <i>findThis</i> Merged with macro:
Method	<b>method:</b> <i>findThis</i>
Method declaration	<b>methoddecl:</b> <i>findThis</i> Merged with method:
Method definition	<b>methoddef:</b> <i>findThis</i> Merged with method:
Namespace	<b>namespace:</b> <i>findThis</i>
Property	<b>prop:</b> <i>findThis</i>
Reference	<b>ref:</b> <i>findThis</i>
String literal	<b>strlit:</b> <i>findThis</i>
Struct	<b>struct:</b> <i>findThis</i> Merged with type:
Struct declaration	<b>structdecl:</b> <i>findThis</i> Merged with type:
Struct definition	<b>structdef:</b> <i>findThis</i> Merged with type:
Template argument	<b>tmplarg:</b> <i>findThis</i> Deprecated in July 2019
Template specification	<b>tplspec:</b> <i>findThis</i> Deprecated in July 2019
Type	<b>type:</b> <i>findThis</i>
Typedef	<b>typedef:</b> <i>findThis</i> Merged with type:
Union	<b>union:</b> <i>findThis</i> Deprecated in July 2019

## Functions to select projects, repositories, paths, and files

Functions make it easy to narrow the search to specified locations, specific types of files within these locations, or specified filenames. Narrow the search to a specific location using the `proj`, `repo`, or `path` filters. Mix and match the functions as required.

USAGE	EXAMPLE
Find all occurrences of the word <i>QueueJobsNow</i> in the Fabrikam project.	<code>QueueJobsNow proj:Fabrikam</code>
Find all occurrences of the word <i>QueueJobsNow</i> in the Contoso repository.	<code>QueueJobsNow repo:Contoso</code>

USAGE	EXAMPLE
Find all occurrences of the word <i>QueueJobsNow</i> in the path <i>VisualStudio/Services/Framework</i> and its subpaths.	<code>QueueJobsNow path:VisualStudio/Services/Framework</code>
Enclose the argument to the filter in double-quotes if it contains a space.	<code>QueueJobsNow path:"VisualStudio/Windows Phones and Devices/Services"</code>
Find all occurrences of the word <i>QueueJobsNow</i> in all files where the filename starts with <i>queueRegister</i> .	<code>QueueJobsNow file:queueRegister*</code>
Find all files with the name <i>QueueRegister</i> without an extension. Use quotes to find files without extensions.	<code>file:"queueRegister"</code>
Find all occurrences of the word <i>QueueJobsNow</i> in only C# source files. A plain text search string that doesn't include file type functions also finds files where the string matches part of the filename.	<code>QueueJobsNow ext:cs</code>

## Find related items or other terms

One of the powerful features of Code Search is the capability to expand your search interactively, based on the results of previous searches. For example, you can easily broaden your search to related files when tracing or debugging code.

Place the insertion point on a term in the file and open the shortcut menu (mouse: right-click) to start a new search for other files containing the selected term. You can search for it as text, for the definition if you select an object name, or for references to a selected object.

For more information about the following search functions, see [Get started with semantic search](#).

- Keyword
- Exact match
- Wildcard
- Boolean operators
- Proximity

## Additional code search operations

See the following examples of even more code search functions. You can use the code type search functions with files written in C#, C, C++, Java, and Visual Basic.NET. Open the search results in a new browser tab from the semantic search and select **Ctrl + Enter**. In Google Chrome, select **Ctrl + Shift + Enter** to switch the focus to the new browser tab.

USAGE	EXAMPLE
Find all instances of "ToDo" comments in your code	Select <code>comment:</code> and enter <code>todo</code>
Search in specific locations, such as within a particular path	Use a search string such as <code>Driver path:MyShuttle/Server</code>

## USAGE

Search for files by name or just by file extension

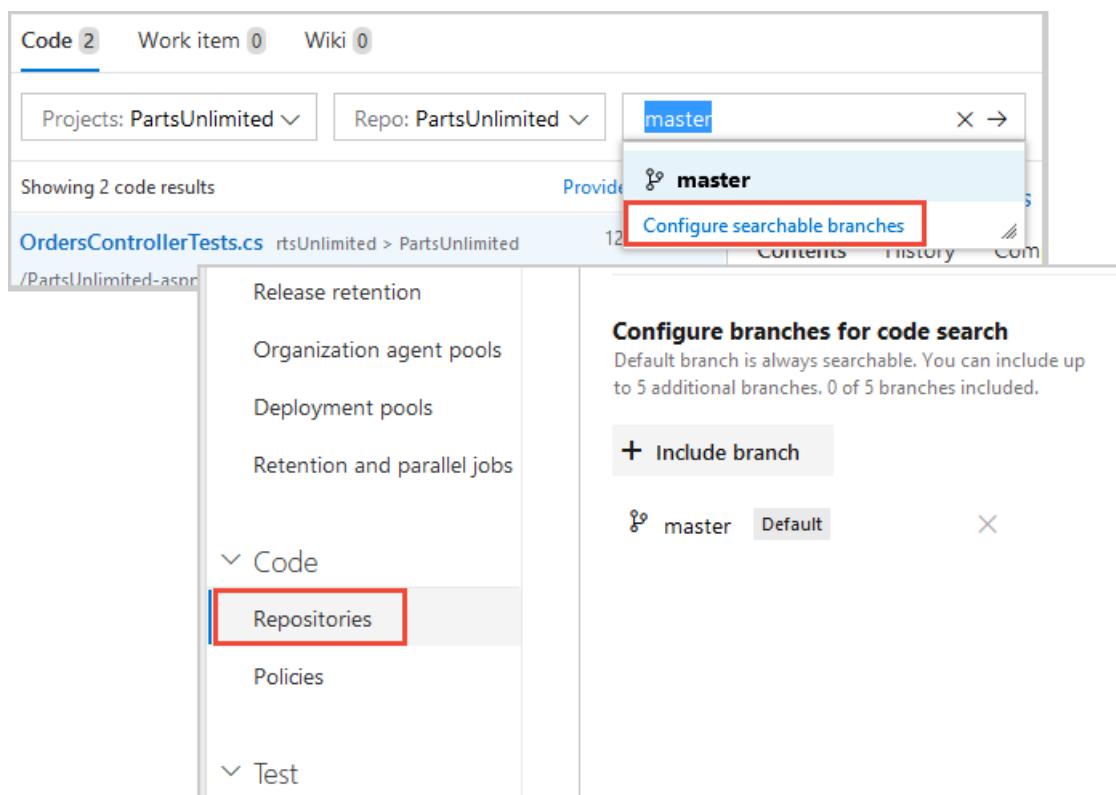
## EXAMPLE

Driver file:GreenCabs.cs . The search string error ext:resx could be useful if you want to review all error strings in your code. Even if your plain text search string matches part of a filename, the file appears in the list of found files. This search works without matching specific file type functions.

## Search Git projects and repositories

In a Git project, you see a list of the repositories that it contains. Use the project and repository checkboxes to widen your search. You can search more or all projects, or narrow your search to fewer projects and repositories. If there are more than a few projects or repositories, use the **Show more** link to see them all.

Code Search can index multiple branches in a Git repository. By default it indexes files in only the default branch of your Git repositories. Your default branch is usually the **main** branch. Specify the branches for each repository, indexing in the Options tab of the **Repositories** section, [project settings page](#).



## Search TFVC projects

In a TFVC project, you see a list of folder paths in that project for which you have read access - you won't see any projects and folders for which you don't have read permission. Select paths in the folder tree to narrow your search if necessary.

### TIP

Code Search remembers your last settings, such as the project and repository or path that you searched in. Clear the checkboxes to search across all projects easily with the **Clear all** links when you want to search in a different scope. In the results pane, Code Search highlights up to the first 100 hits or matches found in the target files.

## Search code with REST API

You can use APIs to extend or supplement the capabilities listed in this article. For information about Code Search with REST API, see [Fetch Code Search Results](#).

## Next steps

[Search work items](#)

## Related articles

- [Get started with Search](#)
- [Search artifacts and packages](#)
- [Search work items](#)
- [Search wiki](#)
- [Search FAQs](#)

# Functional work item search

4/27/2021 • 8 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2017

Functional work item search command filters extend your ability to refine your search of work items based on assignment, work item type, specific fields, and more. This is in addition to the filter functions documented in [Get started with semantic search](#). Work item search is a built-in feature available to all Azure DevOps users.

You can use Work Item Search by default without any installation when the Boards service is installed and enabled in Azure DevOps Services.

By using Work Item Search, you can do the following tasks and more.

SEARCH TASK	DESCRIPTION
<a href="#">Search over all your projects</a>	Search in your own and your partner teams' backlog. Use cross-project searches over all the work items to search across your enterprise's entire work items. Narrow your search by using project and area path filters.
<a href="#">Search across all work item fields</a>	Quickly and easily find relevant work items by searching across all work item fields, including custom fields. Use a full text search across all fields to efficiently locate relevant work items. The snippet view indicates where matches were found.
<a href="#">Search in specific fields</a>	Use the quick in-line search filters to narrow down to a list of work items in seconds. Use the filters on any work item field. The list of suggestions helps complete your search faster. For example, a search such as <b>AssignedTo:Chris</b> <b>WorkItemType:Bug State:Active</b> finds all active bugs assigned to a user named Chris.
Search across test	Search across Test Plans, Test Suites, and other test work item types.
<a href="#">Take advantage of integration with work item tracking</a>	The Work Item Search interface integrates with familiar controls for managing your work items; letting you view, edit, comment, share, and more.

## Prerequisites

- All users can use work item search.

### Search by work item ID

Enter the work item ID in the Azure DevOps title bar to quickly go to it. Searching for a work item ID opens the work item in a modal dialog, providing quick access to read and edit work items.

This project **119**

USER STORY 119

119 Login behaviour for booking

General

State: Closed Area: FabrikamFiber Web  
Reason: Acceptance te ... Iteration: FabrikamFiber Web\Iteration 2

Description

Login behaviour for booking

TEST SUITE 2 **272541**

FabrikamTest

Mateo Escobedo 0 Add tag

In Progress State: In Progress Area: Fabrikam  
New test suite Reason: New test suite Iteration: Fabrikam\Sprint 1

Description

## Full text search across all fields

You can easily search across all work item fields, including custom fields, which enables more natural searches. The snippet view indicates where matches were found.

## Search across all fields

Easily search across all work item fields.  
The snippet indicates where the matches were found.

- Use simple search strings for words or phrases. Work item search matches derived forms of your search terms; for example, a search for "updating" also finds instances of the word "updated" and "update". Searches aren't case-sensitive.
- When you search from inside a project, the default is to search only within that project.
- While searching from inside a team, the default is to search only within the default area path of that team.
  - When you have one project selected, you see a list of area paths in that project for which you have read access - you won't see any projects and area paths for which you don't have read permission
  - Select area paths in the tree to narrow your search if necessary.
- The selected projects are always at the top of the list. Notice that hit counts are also shown for projects that aren't selected.
- Open the search results in a new browser tab from either the semantic search or by selecting **Ctrl + Shift + Enter**.

## Work item search best practices

- Use a text search across all fields to efficiently locate relevant work items. Text search is useful when you're trying to, for example, search for all work items that had similar exception trace.
- Use the quick in-line search filters on any work item field to narrow down to a list of work items in seconds. The list of suggestions helps complete your search faster.

## Semantic search vs. managed work item queries

You have two ways to find and list work items: managed queries and semantic searches. If you're looking for a single work item, use the semantic search. If you want to generate a list of work items to triage, update, chart, or share with others, use a managed query.

### NOTE

With semantic search, you search against a more fully indexed set of fields than that of managed queries.

### Use a managed query

### Use a semantic search

- List items to perform bulk updates to fields.

- Review work that's in progress or recently closed.
- Triage work: set priority, review, update.
- Create a chart and add it to a dashboard.
- Create a chart to get a count of items or sum a field.
- Create a chart that shows a burndown or burnup over time.
- View a tree of parent-child related work items.
- List work items with link relationships.
- List work items for a single project, multiple projects, or across all projects.
  
- Find a specific work item using its ID or a keyword.
- Find one or more work items across all projects in a fast, flexible manner.
- Perform full text search across all work item fields.
- Review work items assigned to a specific team member.
- Search against specific work item fields to quickly narrow down a list of work items.
- Determine what key words will support a managed search.
- List work items for a single project, multiple projects, or across all projects.

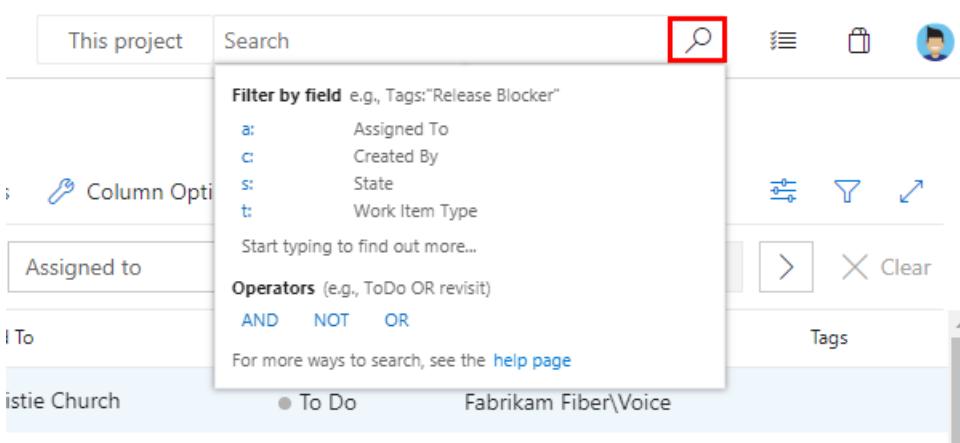
To get started, see the following articles:

- [View and run a query](#)
- [Use semantic search](#)
- [Define a query](#)

For specific managed query examples, see [Query quick reference](#), [Example queries](#).

## Apply supported functions to work item search

1. Fine-tune your search by specifying the fields to search. Enter `a:` and a user name to search for all items assigned to that user.



See the following quick filters that you can use:

- `a:` for **Assigned to**:
- `c:` for **Created by**:
- `s:` for **State**
- `t:` for **Work item type**

2. Start entering the name of a field in your work items; for example, enter `ta`.

The screenshot shows a search interface with a search bar containing 'ta'. Below the search bar is a dropdown menu titled 'Filter by field' with suggestions: 'Tags', 'Target', 'Target Date', 'Target Resolve Date', 'Task Due Date', and 'Task Type'. Below this is a section for 'Operators' with examples: 'e.g., a:@me (t:Task OR t:Bug)' and 'AND NOT OR'. At the bottom, it says 'For more ways to search, see the [help page](#)'.

The dropdown list shows work item field name suggestions that match user input. These suggestions help you complete the search faster. For example, a search such as `tags:Critical` finds all work items tagged 'Critical'.

3. Add more filters to further narrow your search, and use Boolean operators to combine terms if necessary. For example, `a: Chris t: Bug s: Active` finds all active bugs assigned to a user named Chris.
4. Narrow your search to specific types and states, by using the selector lists at the top of the results page.
5. Widen your search across all projects, or narrow it to specific types and states. Use the filter to show the selector lists.

The screenshot shows the search interface with the 'Search this organization' button highlighted with a red box. Other visible elements include the search bar ('login'), a 'Show filter panel' button, and a 'View' dropdown.

6. Select the criteria you want in the drop-down selector lists, or search across the entire organization.

The screenshot shows the search interface with expanded filter panels. The 'Work item' type filter shows 'FabrikamFiber Web' selected, with 'Team1' and 'Team2' listed below. The 'Types' filter shows 'All Types' selected, with 'User Story' listed. The 'States' filter shows 'All States' selected, with 'Closed' and 'Resolved' listed.

7. Sort the results as you need using the drop-down list of field names, work item types, or by relevance.

This project login

Code 0 Work item 3 Wiki 0 Search this organization

Area: FabrikamFiber Web Types: All

Showing 3 work item results

**Provide feedback**

- 62 Login page (Resolved) Description: Login page
- 119 Login behaviour for booking (Closed) Description: Login behaviour for booking
- 97 Login and logout behaviours (Closed) Description: Login and logout behaviours

Sort by

- Assigned To
- Changed Date
- Created Date
- ID
- Relevance
- State
- Tags
- Title
- Work Item Type

1. Fine-tune your search by specifying the fields to search. Enter `a:` and a user name to search for all items assigned to that user.

Filter by field e.g., Tags:"Release Blocker"

<code>a:</code>	Assigned To
<code>c:</code>	Created By
<code>s:</code>	State
<code>t:</code>	Work Item Type

Start typing to find out more...

Operators (e.g., ToDo OR revisit)

AND   NOT   OR

For more ways to search, see the [help page](#)

See the following quick filters that you can use:

- `a:` for **Assigned to**:
- `c:` for **Created by**:
- `s:` for **State**
- `t:` for **Work item type**

2. Start entering the name of a field in your work items; for example, enter `ta`.

The screenshot shows a search interface with a search bar containing 'ta'. Below the search bar is a dropdown menu titled 'Filter by field' with examples like 'e.g., Tags:"Release Blocker"'. The menu lists several work item fields: Tags, Target, Target Date, Target Resolve Date, Task Due Date, and Task Type. Below this, there's a section for 'Operators' with examples like 'a:@me (t:Task OR t:Bug)' and Boolean operators AND, NOT, OR. At the bottom, it says 'For more ways to search, see the [help page](#)'.

The dropdown list shows work item field name suggestions that match user input. These suggestions help you complete the search faster. For example, a search such as **tags:Critical** finds all work items tagged 'Critical'.

3. Add more filters to further narrow your search, and use Boolean operators to combine terms if necessary. For example, **a: Chris t: Bug s: Active** finds all active bugs assigned to a user named Chris.
4. Narrow your search to specific types and states, by using the drop-down selector lists at the top of the results page.
5. Widen your search across all projects, or narrow it to specific types and states. Use the filter to show the selector lists.

This screenshot shows the search results page. The search bar contains 'login'. Below the search bar are links for 'item 3', 'Wiki 0', and a 'Search this organization' button. To the right are dropdowns for 'Relevance' and 'View', and a 'Show filter panel' button. A red box highlights the 'Search this organization' button.

6. Select the criteria you want in the drop-down selector lists, or search across the entire organization.

This screenshot shows the search results page with the 'Work item' dropdown selected. The dropdown shows 'FabrikamFiber Web' with a red box around the 'x' icon. Below it is a list of area paths: 'Team1' and 'Team2'. To the right of the dropdown are two filter panels: 'Types: All' and 'States: All', both with red boxes around them. The 'Types' panel shows 'All Types' (checked) and 'User Story'. The 'States' panel shows 'All States' (checked) and 'Closed', 'Resolved'.

7. Sort the results as you need using the drop-down list of field names, work item types, or by relevance.

This screenshot shows the Microsoft Azure DevOps interface for searching work items. At the top, there are navigation links for 'This project', 'login', and a search bar. Below that, a header bar includes 'Code 0', 'Work item 3', 'Wiki 0', and a link to 'Search this organization'. On the right side of the header, there is a 'Sort by' dropdown menu with 'Relevance' selected, indicated by a red box.

The main area displays search results for 'Login page' under the 'FabrikamFiber Web' area. The results show three work items:

- ID 62 Login page**: Resolved, Description: Login page
- ID 119 Login behaviour for booking**: Closed, Description: Login behaviour for booking
- ID 97 Login and logout behaviours**: Closed, Description: Login and logout behaviours

The 'Sort by' dropdown menu lists various fields for sorting work items, including 'Assigned To', 'Changed Date', 'Created Date', 'ID', 'Relevance' (which is checked), 'State', 'Reason', 'Tags', 'Title', and 'Work Item Type'.

## Quick filters for matching in specific fields

Quick inline search filters let you refine work items in seconds. The dropdown list of suggestions helps complete your search faster. Mix and match the functions to create quick powerful searches.

USAGE	EXAMPLE
Scope your search terms to match in any work item field including custom fields. Enter the field name followed by the search terms.	<code>tags:Critical</code> finds work items having a field 'tags' containing the term 'Critical'.
Use multiple inline search filters to scope your search by any work item field, including custom fields.	<code>t: Bug path:"project\search"</code> finds all bugs in the area path "project\search".
Use the operators <code>&gt;</code> , <code>&gt;=</code> , <code>&lt;</code> , <code>&lt;=</code> , <code>=</code> , and <code>!=</code> for date, integer, and float fields.	<code>t: Bug CreatedDate &gt; @Today -7</code> finds all bugs created in the last week.
For the search query that contains multiple terms and users looking for exact match, embed the search term inside <code>" "</code>	<code>BuildPath: "tools.demoproject.com"</code> finds all work items that necessarily contain the path "tools.demoproject.com".

## Quick Filters

Quick in-line search filters lets you refine work items by specific criteria on any work item field, in seconds!

🔍

## Scope projects and area and iteration paths using filters

Filters make it easy to narrow the search to specified projects and area paths.

Narrow the search to a specific location using the `proj`, `area`, `iteration`, `path`, and `comment` filters:

USAGE	EXAMPLE
Finds all occurrences of the word <b>Wiki</b> in the <b>Fabrikam</b> project.	<code>Wiki proj:Fabrikam</code>
Finds all occurrences of the word <b>Wiki</b> in the area path <b>Contoso/Mobile</b> and its subpaths.	<code>Wiki area:Contoso/Mobile</code>
Finds all occurrences of the word <b>Wiki</b> in the iteration path <b>Contoso/Sprint101</b> and its subpaths.	<code>Wiki iteration:Contoso/Sprint101</code>
Enclose the argument to the filter in double-quotes if it contains a space.	<code>Wiki path:"Contoso/Windows Phones and Devices/Services"</code>
Finds backlog comments	<code>comment:todo</code>

## See more of the work item

You can quickly get a full screen view of the selected work item using  **expand** and  **shrink** in the toolbar. However, another way to see more of the work item, while you can still select work items from the list of matching results, is to hide the left column filter pane by choosing < at the top left of the column. Use > to restore the filter pane.

If you're using a portrait orientation screen, use the **Preview pane: Right** link at the top right of the window to display the code below the search results list.

### TIP

Search remembers the state of the filter pane, configuration of the work item view pane, and its position between sessions as part of your user preferences.

## Search Work Items with REST API

You can use APIs to extend or supplement the capabilities listed in this article. For information about Work Item Search with REST API, see [Fetch Work Item Search Results](#).

## Next steps

[Supported filter functions and more for work items](#)

## Related articles

- [Get started with Search](#)
- [Search code](#)
- [Search artifacts and packages](#)
- [Search wiki](#)
- [Search FAQs](#)

# Migrate data from Azure DevOps Server to Azure DevOps Services

4/5/2021 • 3 minutes to read • [Edit Online](#)

[Azure DevOps Services](#) | [Azure DevOps Server](#) | [TFS](#)

The data migration tool for Azure DevOps provides a high fidelity way to migrate collection databases from Azure DevOps Server to Azure DevOps Services. It's recommended that you download the [migration guide and tool](#) if you're looking to use this service to import your collection(s). The guide serves as a walk through of the different steps involved in an import. Providing best practices, checklists, and helpful tips to make your import as easy as possible. The guide should be used in conjunction with the more technical documentation referenced below to successfully import to Azure DevOps Services.

## Supported Azure DevOps Server versions for import

### IMPORTANT

It can take up to 2-3 weeks after a new RTW version of Azure DevOps Server is released for import support to come online for that version. It's important to take this into consideration when choosing to upgrade shortly after a new RTW Azure DevOps Server release.

The data migration tool for Azure DevOps supports the two latest releases of Azure DevOps Server at a given time. Releases include updates and major releases. Currently the following versions of Azure DevOps Server are supported for import:

- Azure DevOps Server 2020.0.1
- Azure DevOps Server 2020

### NOTE

The data migration tool doesn't support imports from Azure DevOps Server release candidates (RC). If you're planning on importing your collection database to Azure DevOps Services using this service, it's important that you don't upgrade your production database to an RC release. If you do upgrade, then you will need to wait and upgrade to the release to web (RTW) version when it's available or restore a backup copy of your database from a previous Azure DevOps Server version to import.

Normal release cadence for new Azure DevOps Server versions is once every three-to-four months. Meaning that support for a given version of Azure DevOps Server for migration to Azure DevOps Services should last for anywhere between six-to-eight months. It's important to ensure that your planning accounts for this support window to avoid having to suddenly upgrade to migrate.

## Preview features

#### **NOTE**

If you're not including preview features when running the migration tool, then you will need to re-run the migration tool prepare to generate a new import.json to queue an import. You DO NOT need to include preview features when you re-generate your import.json.

If you had previously been including preview features then you DO NOT need to take any additional actions after Monday, April 23, 2020.

The following features can be included with your import, but are currently in a preview state.

- [Analytics](#) - Note this is only supported for Azure DevOps Server 2019 and later.

When queueing an import you can elect to include preview features with your import. If you do, data related to these features will be copied into your new organization along with all your other data. Should you choose to not include these features then their data will not be copied.

For a list of items not included with an import, see the [migration guide and tool](#).

## Data migration tool for Azure DevOps resources

In general you should use the [Migration guide and tool](#) when going through an import. When it's required, the guide links back to the following articles. These articles offer deeper technical knowledge on various import topics.

### **Import process**

- [Validate a collection for import](#)
- [Prepare a collection for import](#)
- [Prepare for import](#)
  - [Prepare large collections for import](#)
- [Run an import](#)
- [Post import steps](#)

### **Troubleshooting**

- [Troubleshooting validation errors](#)
- [Troubleshooting process errors](#)
- [Troubleshooting import errors](#)

## Q & A

### **Q: Will my Personal Access Tokens also migrate when I migrate from on-premises to Azure DevOps Services?**

A: No. Your tokens will not migrate and you will need to [regenerate your Personal Access Tokens](#) on Azure DevOps Services.

### **Q: If I have feedback or additional questions is there somewhere I can reach out?**

A: Yes. You can contact [AzureDevOpsImport@microsoft.com](mailto:AzureDevOpsImport@microsoft.com). Please note that this alias is for general questions. If you need assistance with a failed import please contact Azure DevOps [customer support](#).

## Videos

## Related articles

- [Migration and process model FAQs](#)

# Migration options

4/2/2021 • 3 minutes to read • [Edit Online](#)

## Azure DevOps Services | Azure DevOps Server | TFS

When you decide to make the move from Azure DevOps Server to Azure DevOps Services, you might start fresh with an empty organization. Often, however, you will have existing code, work items, and other assets that you want to move. There are many approaches to doing this which vary in both the fidelity of the data transfer and the complexity of the process.

Prior to migrating data, review the differences that exist between [Azure DevOps Server and Azure DevOps Services](#).

### Option 1: Copy the most important assets manually

By far the easiest option for moving data into Azure DevOps Services is to manually copy your most important assets and start relatively fresh. This can be difficult when you are in the middle of a large project, but you can make it easier if you do some advance planning and schedule your move when it makes sense for your team.

For example, when the Azure DevOps team chose to move from Azure DevOps Server to Azure DevOps Services, we also decided to move from Team Foundation Version Control (TFVC) to Git. This required a fair bit of planning, but when we actually performed our migration, we created a new Git repo using the "tip" version of our TF VC sources, and left our history behind in Azure DevOps Server. We also moved our active work items, and left behind all our old bugs, completed user stories and tasks, and so on.

Here's the general process:

1. Identify the most important assets that you need to migrate - typically source code, work items, or both. Other assets in Azure DevOps Server - build pipelines, test plans, and so forth - are harder to manually migrate.
2. Identify a good time to make the transition.
3. Prepare your target organizations. Create the organizations and team projects that you need, provision users, and so on.
4. Migrate your data.
5. Consider making the source Azure DevOps Server deployments read-only.

### Option 2: High fidelity database migration.

The Azure DevOps Server & Azure DevOps Services product team provides a high fidelity data migration tool. A downloadable Migration Guide is available at <https://aka.ms/AzureDevOpsImport>.



Because the data migration tool operates at a database level, it can provide a very high fidelity migration. If you want to move your existing Azure DevOps Server data into Azure DevOps Services, we strongly recommend using this option.

## Option 3: Using public API-based tools for higher fidelity migration

If for some reason you cannot use the data migration tool but still want a higher fidelity migration than Option 1, you can choose from a variety of tools that use public APIs to move data. Generally these tools can provide a higher fidelity migration than a manual copy of "tip" data, but they are still relatively low fidelity. For example:

- None of them will preserve the dates of TF VC changesets.
- Many of them will not preserve the changed dates of work item revisions.
- None of them will migrate all Azure DevOps Server artifacts.

In general, we only recommend this approach if the extra fidelity beyond a manual copy is critical. If you decide to take this approach, you might consider hiring a consultant who has experience with one or more of the tools. You should definitely consider doing a test migration before doing your final migration.

Many organizations need a very high fidelity migration for only a subset of their work. New work could potentially start directly in Azure DevOps Services. Other work, with less stringent fidelity requirements, could be migrated using one of the other approaches. You will have to weigh the pros and cons of the various approaches against your motivations for moving into Azure DevOps Services and decide for yourself what is the right strategy.

## Related articles

- [About Azure DevOps Services and Azure DevOps Server](#)
- [Pricing, Azure DevOps Services](#)
- [Pricing, Azure DevOps Server](#)

# Validation and import processes

4/2/2021 • 31 minutes to read • [Edit Online](#)

## Azure DevOps Services | Azure DevOps Server | TFS

This article walks you through the preparation that's required to get an import to Azure DevOps Services ready to run. If you encounter errors during the process, see [Troubleshoot import and migration errors](#).

### NOTE

- Visual Studio Team Services (VSTS) is now [Azure DevOps Services](#).
- With the release of Azure DevOps Server 2019, the TFS Database Import Service has been rebranded as the data migration tool for Azure DevOps. This change includes TfsMigrator (Migrator) becoming the data migration tool. This service works exactly the same as the former import service. If you're running an older version of on-premises Azure DevOps Server with the TFS branding, you can still use this feature to migrate to Azure DevOps as long as you've upgraded to one of the supported server versions.
- Before you begin the import tasks, check to ensure that you're running a [supported version of Azure DevOps Server](#).

We recommend that you use the [Step-by-step migration guide](#) to progress through your import. The guide links to technical documentation, tools, and best practices.

## Validate a collection

After you've confirmed that you're running the latest version of Azure DevOps Server, your next step is to validate each collection that you want to migrate to Azure DevOps Services.

The validation step examines various aspects of your collection, including, but not limited to, size, collation, identity, and processes.

You run the validation by using the data migration tool. To start, [download the tool](#), copy the zip file to one of your Azure DevOps Server application tiers, and then unzip it. You can also run the tool from a different machine without Azure DevOps Server installed as long as the machine can connect to the configuration database of the Azure DevOps Server instance. An example is shown here.

1. Open a Command Prompt window on the server, and enter a cd command to change to the directory where the data migration tool is stored. Take a few moments to review the help content that's provided with the tool.

a. To view the top-level help and guidance, run the following command:

```
Migrator /help
```

b. View the help text for the command:

```
Migrator validate /help
```

2. Because this is your first time validating a collection, let's keep it simple. Your command should have the following structure:

```
Migrator validate /collection:{collection URL}
```

For example, to run against the default collection the command would look like:

```
Migrator validate /collection:http://localhost:8080/DefaultCollection
```

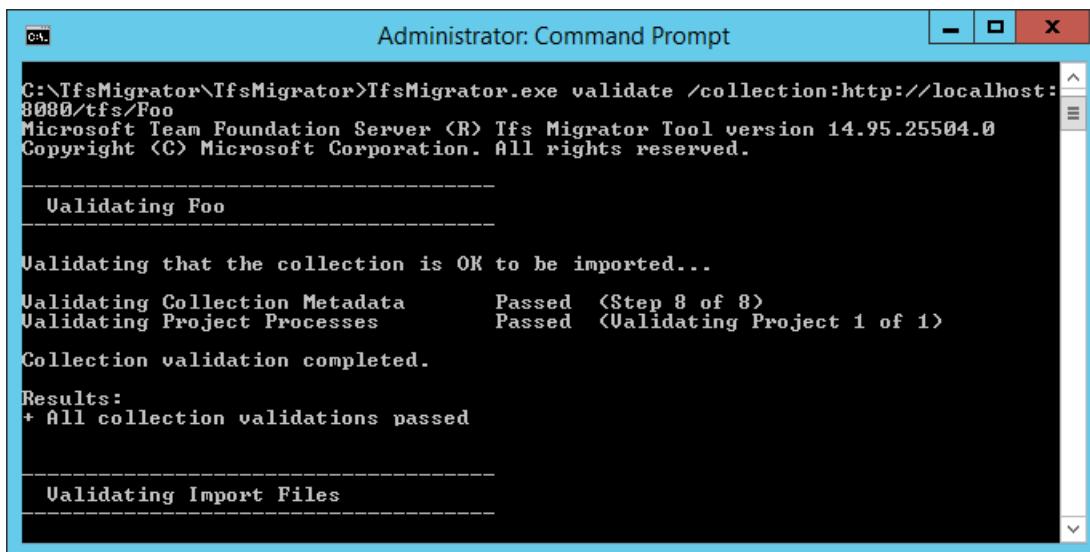
3. To run the tool from a machine other than the Azure DevOps Server, you need the **/connectionString** parameter. The connection string parameter points to your Azure DevOps Server configuration database. As an example, if the validate command is being run by the Fabrikam corporation, the command would look like:

```
Migrator validate /collection:http://fabrikam:8080/DefaultCollection  
/tenantDomainName:fabrikam.OnMicrosoft.com /connectionString:"Data Source=fabrikam;Initial  
Catalog=Configuration;Integrated Security=True"
```

#### IMPORTANT

The data migration tool *does not* edit any data or structures in the collection. It reads the collection only to identify issues.

4. After the validation is complete, you can view the log files and results.



The screenshot shows an Administrator Command Prompt window with the title "Administrator: Command Prompt". The command entered was `C:\TfsMigrator\TfsMigrator>TfsMigrator.exe validate /collection:http://localhost:8080/tfs/Foo`. The output indicates that the Microsoft Team Foundation Server (R) Tfs Migrator Tool version 14.95.25504.0 Copyright (C) Microsoft Corporation. All rights reserved. The validation process for the collection "Foo" is shown, including validating metadata, project processes, and import files. The results show that all collection validations passed. The final step is "Validating Import Files".

After all the validations pass, you can move to the next step of the import process. If the data migration tool flags any errors, you need to correct them before you proceed. For guidance on correcting validation errors, see [Troubleshoot import and migration errors](#).

#### Import log files

When you open the log directory, you'll notice several logging files.

The main log file is named `DataMigrationTool.log`. It contains details about everything that was run. To make it easier for you to focus on specific areas, a log is generated for each major validation operation.

For example, if TfsMigrator reports an error in the "Validating Project Processes" step, you can open the `ProjectProcessMap.log` file to view everything that was run for that step instead of having to scroll through the entire log.

You should review the `TryMatchOobProcesses.log` file only if you're trying to import your project processes to use the [inherited model](#). If you don't want to use the inherited model, you can ignore these errors, because they

won't prevent you from importing to Azure DevOps Services.

## Generate import files

By now, you've run the data migration tool validation against the collection, and it's returning a result of "All collection validations passed." Before you take a collection offline to migrate it, you need to generate the import files. When you run the `prepare` command, you generate two import files:

- *IdentityMapLog.csv*: Outlines your identity map between Active Directory and Azure Active Directory (Azure AD).
- *import.json*: Requires you to fill out the import specification you want to use to kick off your migration.

### The `prepare` command

The `prepare` command assists with generating the required import files. Essentially, this command scans the collection to find a list of all users to populate the identity map log, *IdentityMapLog.csv*, and then tries to connect to Azure AD to find each identity's match. To do this, your company needs to use the [Azure Active Directory Connect tool](#) (formerly known as the Directory Synchronization tool, Directory Sync tool, or DirSync.exe tool).

If directory synchronization is set up, the data migration tool should be able to find the matching identities and mark them as *Active*. If it doesn't find a match, the identity is marked as *Historical* in the identity map log, and you'll need to investigate why the user isn't included in your directory sync. The import specification file, *import.json*, should be filled out prior to the import.

Unlike the `validate` command, `prepare` *does* require an internet connection, because it needs to connect to Azure AD to populate the identity map log file. If your Azure DevOps Server instance doesn't have internet access, you need to run the tool from a machine that does. As long as you can find a machine with an intranet connection to your Azure DevOps Server instance and an internet connection, you can run this command. For help with the `prepare` command, run the following command:

```
Migrator prepare /help
```

Included in the help documentation are instructions and examples for running Migrator from the Azure DevOps Server instance itself and a remote machine. If you're running the command from one of the Azure DevOps Server instance's application tiers, your command should have the following structure:

```
Migrator prepare /collection:{collection URL} /tenantDomainName:{name} /region:{region}
```

```
Migrator prepare /collection:{collection URL} /tenantDomainName:{name} /region:{region}  
/connectionString:"Data Source={sqlserver};Initial Catalog=Configuration;Integrated Security=True"
```

The `ConnectionString` parameter is a pointer to the configuration database of your Azure DevOps Server instance. As an example, if the `prepare` command is being run by the Fabrikam corporation, the command would look like:

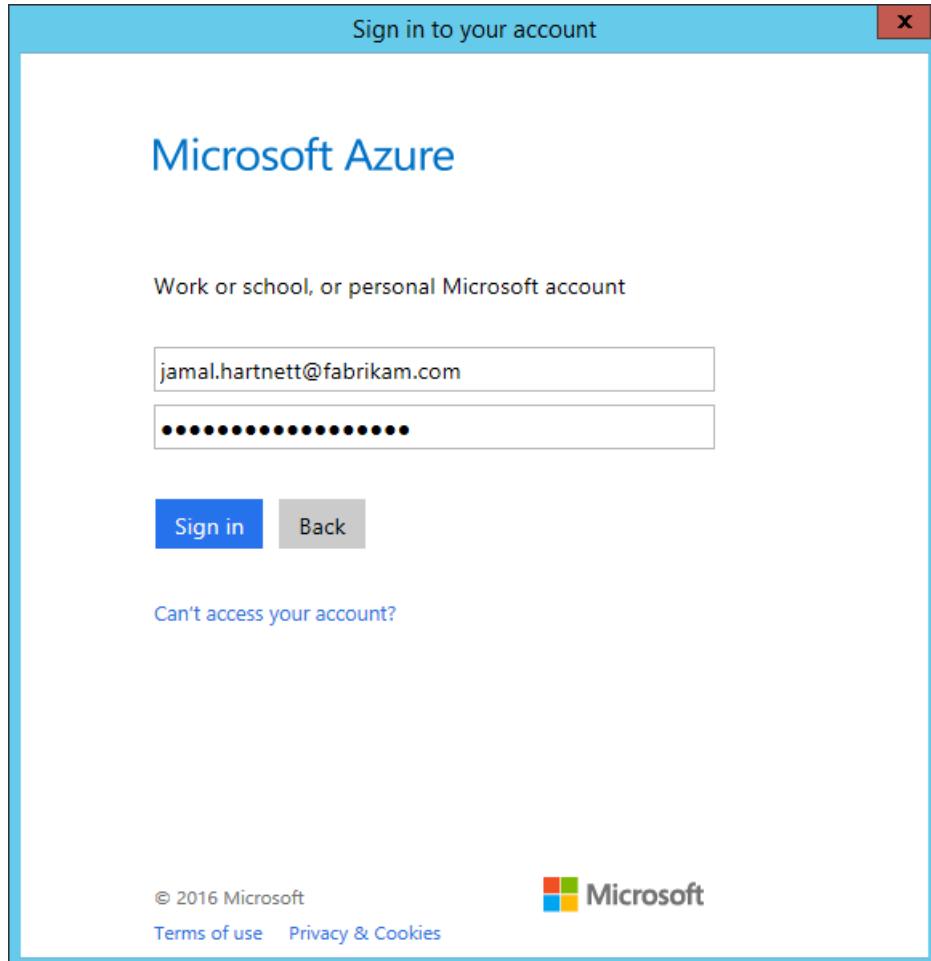
```
Migrator prepare /collection:http://fabrikam:8080/DefaultCollection  
/tenantDomainName:fabrikam.OnMicrosoft.com /region:{region} /connectionString:"Data Source=fabrikam;Initial  
Catalog=Configuration;Integrated Security=True"
```

When the data migration tool runs the `prepare` command, it runs a complete validation to ensure that nothing has changed with your collection since the last full validation. If any new issues are detected, no import files are generated.

Shortly after the command has started running, an Azure AD sign-in window is displayed. You need to sign in with an identity that belongs to the tenant domain that's specified in the command. Make sure that the specified Azure AD tenant is the one you want your future organization to be backed with. In our Fabrikam example, a user would enter credentials that are similar to what's shown in the following screenshot.

**IMPORTANT**

Do *not* use a test Azure AD tenant for a test import and your production Azure AD tenant for the production run. Using a test Azure AD tenant can result in identity import issues when you begin your production run with your organization's production Azure AD tenant.



When you run the `prepare` command successfully in the data migration tool, the results window displays a set of logs and two import files. In the log directory, you'll find a logs folder and two files:

- *import.json* is the import specification file. We recommend that you take time to fill it out.
- *IdentityMapLog.csv* contains the generated mapping of Active Directory to Azure AD identities. Review it for completeness before you kick off an import.

The two files are described in greater detail in the next sections.

### The import specification file

The import specification, *import.json*, is a JSON file that provides import settings. It includes the desired organization name, storage account information, and other information. Most of the fields are autopopulated, and some fields require your input before you attempt an import.

```

1  [
2      "Source": {
3          "Location": "<Provide the SASKey to the Azure storage container with the collection and import files.>",
4          "Files": {
5              "Dacpac": "Tfs_DefaultCollection.dacpac"
6          }
7      },
8      "Target": {
9          "Name": "<Provide a name for the account that will be created during the import.>"
10     },
11     "Properties": {
12         "ImportType": "<Provide the Type of Import: DryRun, ProductionRun>"
13     },
14     "ValidationData": {
15         "TfsMigratorVersion": "16.255.65000.0",
16         "SourceCollectionId": "8b245d37-d41d-4188-a6f1-b5bb397860ba",
17         "DataImportCollectionId": "ca970402-9b06-4720-9407-ba32684e9499",
18         "DatabaseCollation": "SQL_Latin1_General_CI_AS",
19         "CommandExecutionCount": 0,
20         "CommandExecutionTime": 0.0,
21         "TfsVersion": "Dev15.M117",
22         "DatabaseTotalSize": 181,
23         "DatabaseBlobSize": 0,
24         "DatabaseTableSize": 181,
25         "DatabaseLargestTableSize": 8,
26         "ActiveUserCount": 8,
27         "TenantId": "72f988bf-86f1-41af-91ab-2d7cd011db47",
28         "Region": "CUS",
29         "ValidationChecksumVersion": 1,
30         "ValidationChecksum": "66516G8u850KY6XKJm6MM5Ty3krNjhUFFCh4zyZMXqm7ZDLVpFpiIi0zDnJcoZmjHgDzvoCNS/9PwGm28hBgPg=="
31     },
32     "Identities": [
33         "S-1-5-21-1374400868-3601225936-2087002269-500",
34         "S-1-5-21-2127521184-1604012920-1887927527-11008431",
35         "S-1-5-21-2127521184-1604012920-1887927527-15795496"
36     ]
37 ]

```

0 0 0 json | import.json Ln 1, Col 1 Spaces: 2 UTF-8 CRLF JSON 😊

The *import.json* file's displayed fields and required actions are described in the following table:

FIELD	DESCRIPTION	REQUIRED ACTION
Source	Information about the location and names of the source data files that are used for import.	No action required. Review information for the subfield actions to follow.
Location	The shared access signature key to the Azure storage account that hosts the data-tier application package (DACPAC).	No action required. This field will be covered in a later step.
Files	The names of the files containing import data.	No action required. Review information for the subfield actions to follow.

FIELD	DESCRIPTION	REQUIRED ACTION
DACPAC	A DACPAC file that packages the collection database to be used to bring in the data during the import.	No action required. In a later step, you'll generate this file by using your collection and then upload it to an Azure storage account. You'll need to update the file based on the name you use when you generate it later in this process.
Target	Properties of the new organization to import into.	No action required. Review information for the subfield actions to follow.
Name	The name of the organization to be created during the import.	Provide a name. The name can be quickly changed later after the import has completed. <b>Note:</b> Do <i>not</i> create an organization with this name before you run the import. The organization will be created as part of the import process.
ImportType	The type of import that you want to run.	No action required. In a later step, you'll select the type of import to run.
Validation Data	Information that's needed to help drive your import experience.	The "ValidationData" section is generated by the data migration tool. It contains information that's needed to help drive your import experience. Do <i>not</i> edit the values in this section, or your import could fail to start.

After you complete the preceding process, you should have a file that looks like the following:

```
import.json - Untitled (Workspace) - Visual Studio Code
File Edit Selection View Go Debug Tasks Help
{} importjson x
1 [
2   "Source": {
3     "Location": "<Provide the SASKey to the Azure storage container with the collection and import files.>",
4     "Files": {
5       "Dacpac": "Tfs_DefaultCollection.dacpac"
6     }
7   },
8   "Target": {
9     "Name": "fabrikam-import"
10 },
11   "Properties": {
12     "ImportType": "<Provide the Type of Import: DryRun, ProductionRun>"
13   },
14   "ValidationData": {
15     "TfsMigratorVersion": "16.255.65000.0",
16     "SourceCollectionId": "8b245d37-d41d-4188-a6f1-b5bb397860ba",
17     "DataImportCollectionId": "ca970402-9b06-4720-9407-ba32684e9499",
18     "DatabaseCollation": "SQL_Latin1_General_CI_AS",
19     "CommandExecutionCount": 0,
20     "CommandExecutionTime": 0.0,
21     "TfsVersion": "Dev15.M117",
22     "DatabaseTotalSize": 181,
23     "DatabaseBlobSize": 0,
24     "DatabaseTableSize": 181,
25     "DatabaseLargestTableSize": 8,
26     "ActiveUserCount": 8,
27     "TenantId": "72f988bf-86f1-41af-91ab-2d7cd011db47",
28     "Region": "CUS",
29     "ValidationChecksumVersion": 1,
30     "ValidationChecksum": "66516G8u850KY6XKJmMM5Ty3krNjhUFFCh4zyZMXqm7ZDLVpFpiIi0zDnJcoZmjHgDzvoCNS/9PwGm28hBgPg=="
31   },
32   "Identities": [
33     "S-1-5-21-1374400868-3601225936-2087002269-500",
34     "S-1-5-21-2127521184-1604012920-1887927527-11008431",
35     "S-1-5-21-2127521184-1604012920-1887927527-15795496"
36   ]
37 ]
```

0 1 1 json | import.json Ln 1, Col 1 Spaces: 2 UTF-8 CRLF JSON

In the preceding image, note that the planner of the Fabrikam import added the organization name *fabrikam-import* and selected CUS (Central United States) as the region for import. Other values were left as is to be modified just before the planner took the collection offline for the migration.

#### NOTE

Dry-run imports have a '-dryrun' automatically appended to the end of the organization name. This can be changed after the import.

## Supported Azure regions for import

Azure DevOps Services is available in several [Azure regions](#). However, not all regions where Azure DevOps Services is available are supported for import. The following table lists the Azure regions that you can select for import. Also included is the value that you need to place in the import specification file to target that region for import.

GEOGRAPHIC REGION	AZURE REGION	IMPORT SPECIFICATION VALUE
United States	Central United States	CUS

GEOGRAPHIC REGION	AZURE REGION	IMPORT SPECIFICATION VALUE
Europe	Western Europe	WEU
United Kingdom	United Kingdom South	UKS
Australia	Australia East	EAU
South America	Brazil South	SBR
Asia Pacific	South India	MA
Asia Pacific	Asia Pacific (Hong Kong)	EA
Canada	Central Canada	CC

## The identity map log

The identity map log is of equal importance to the actual data that you'll be migrating to Azure DevOps Services. As you're reviewing the file, it's important to understand how identity import operates and what the potential results could entail. When you import an identity, it can become either *active* or *historical*. Active identities can sign in to Azure DevOps Services, but historical identities cannot.

### Active identities

Active identities refer to identities that will be users in Azure DevOps Services post-import. In Azure DevOps Services, these identities are licensed and are displayed as users in the organization. The identities are marked as *active* in the **Expected Import Status** column in the identity map log file.

### Historical identities

Historical identities are mapped as such in the **Expected Import Status** column in the identity map log file. Identities without a line entry in the file also become historical. An example of an identity without a line entry might be an employee who no longer works at a company.

Unlike active identities, historical identities:

- *Don't* have access to an organization after migration.
- *Don't* have licenses.
- *Don't* show up as users in the organization. All that persists is the notion of that identity's name in the organization, so that its history can be searched later. We recommend that you use historical identities for users who no longer work at the company or who won't need further access to the organization.

#### NOTE

After an identity is imported as historical, it *can't* become active.

## Understand the identity map log file

The identity map log file is similar to the example shown here:

AD: User(TFS)	AD: Security Identifier	AAD: Expected Import User(VSTS)	Expected Import Status	Validation Date
FABRIKAM\Jamal Hartnett	S-1-5-21-983578539-230207283-3682864982-500	No Match Found (Check AAD Sync)	Historical	2017-10-31T21:15:44Z
FABRIKAM\Mateo Escobedo	S-1-5-21-4100298327-4227319834-4140607669-500	No Match Found (Check AAD Sync)	Historical	2017-10-31T21:15:44Z
FABRIKAM\Helena Petersen	S-1-5-21-124525095-708259637-1543119021-1419599	helena.petersen@fabrikam.com	Active	2017-10-31T21:15:44Z
FABRIKAM\Raisa Pokrovskaya	S-1-5-21-2127521184-1604012920-1887927527-406986	raisa.pokrovskaya@fabrikam.com	Active	2017-10-31T21:15:44Z

The columns in the identity map log file are described in the following table:

**NOTE**

You and your Azure AD admin will need to investigate users that are marked as *No Match Found (Check Azure AD Sync)* to understand why they aren't part of your Azure AD Connect sync.

COLUMN	DESCRIPTION
Active Directory: User (Azure DevOps Server)	The friendly display name used by the identity in Azure DevOps Server. This name makes it easier to identify which user the line in the map is referencing.
Active Directory: Security Identifier	The unique identifier for the on-premises Active Directory identity in Azure DevOps Server. This column is used to identify users in the collection.
Azure Active Directory: Expected Import User (Azure DevOps Services)	Either the expected sign-in address of the matched soon-to-be-active user or <i>No Match Found (Check Azure AD Sync)</i> , indicating that the identity wasn't found during the Azure Active Directory sync and it will be imported as historical.
Expected Import Status	The expected user import status: either <i>Active</i> if there's a match between your Active Directory and Azure Active Directory, or <i>Historical</i> if there isn't a match.
Validation Date	The last time the identity map log was validated.

As you read through the file, notice whether the value in the **Expected Import Status** column is *Active* or *Historical*. *Active* indicates that it's expected that the identity on this row will map correctly on import and will become active. *Historical* means that the identities will become historical on import. It's important to review the generated mapping file for completeness and correctness.

**IMPORTANT**

Your import will fail if major changes occur to your Azure AD Connect security ID sync between import attempts. You can add new users between dry runs, and you can make corrections to ensure that previously imported historical identities become active. However, changing an existing user that was previously imported as active isn't supported at this time. Doing so will cause your import to fail. An example of a change might be completing a dry-run import, deleting an identity from your Azure AD that was imported actively, re-creating a new user in Azure AD for that same identity, and then attempting another import. In this case, an active identity import will be attempted between the Active Directory and newly created Azure AD identity, but it will cause an import failure.

1. Start by reviewing the correctly matched identities. Are all the expected identities present? Are the users mapped to the correct Azure AD identity?

If any values are incorrectly mapped or need to be changed, contact your Azure AD administrator to verify that the on-premises Active Directory identity is part of the sync to Azure AD and has been set up correctly. For more information, see [Integrate your on-premises identities with Azure Active Directory](#).

2. Next, review the identities that are labeled as *historical*. This labeling implies that a matching Azure AD identity couldn't be found, for any of the following reasons:
  - The identity hasn't been set up for sync between on-premises Active Directory and Azure AD.
  - The identity hasn't been populated in your Azure AD yet (for example, there's a new employee).

- The identity doesn't exist in your Azure AD instance.
- The user who owns that identity no longer works at the company.

To address the first three reasons, you need to set up the intended on-premises Active Directory identity to sync with Azure AD. For more information, see [Integrate your on-premises identities with Azure Active Directory](#). You must set up and run Azure AD Connect for identities to be imported as *active* in Azure DevOps Services.

You can ignore the fourth reason, because employees who are no longer at the company should be imported as *historical*.

#### **Historical identities (small teams)**

##### **NOTE**

The identity import strategy proposed in this section should be considered by small teams only.

If Azure AD Connect hasn't been configured, you'll notice that all users in the identity map log file are marked as *historical*. Running an import this way results in all users being imported as *historical*. We strongly recommended that you configure [Azure AD Connect](#) to ensure that your users are imported as *active*.

Running an import with all historical identities has consequences that need to be considered carefully. It should be considered only by teams with a small number of users and for which the cost of setting up Azure AD Connect is deemed too high.

To import all identities as historical, follow the steps outlined in later sections. When you queue an import, the identity that's used to queue the import is bootstrapped into the organization as the organization owner. All other users are imported as historical. Organization owners can then [add the users back in](#) by using their Azure AD identity. The added users are treated as new users. They do *not* own any of their history, and there's no way to re-parent this history to the Azure AD identity. However, users can still look up their pre-import history by searching for their <domain><Active Directory username>.

The data migration tool displays a warning if it detects the complete historical identities scenario. If you decide to go down this migration path, you'll need to consent in the tool to the limitations.

#### **Visual Studio subscriptions**

The data migration tool can't detect Visual Studio subscriptions (formerly known as MSDN benefits) when it generates the identity map log file. Instead, we recommend that you apply the auto license upgrade feature after the import. As long as users' work accounts are [linked](#) correctly, Azure DevOps Services automatically applies their Visual Studio subscription benefits at their first sign-in after the import. You're never charged for licenses that are assigned during the import, so this can be safely handled afterward.

You don't need to repeat a dry-run import if users' Visual Studio subscriptions aren't automatically upgraded in Azure DevOps Services. Visual Studio subscription linking happens outside the scope of an import. As long as their work account is linked correctly before or after the import, users' licenses are automatically upgraded on their next sign-in. After their licenses have been upgraded successfully, the next time you run an import, the users are upgraded automatically on their first sign-in to the organization.

## **Prepare for import**

By now, you have everything ready to execute on your import. You need to schedule downtime with your team to take the collection offline for the migration. When you've agreed upon a time to run the import, you need to upload to Azure both the required assets you've generated and a copy of the database. This process has five steps:

Step 1: [Take the collection offline and detach it](#).

Step 2: [Generate a DACPAC file from the collection you're going to import](#).

Step 3: Upload the DACPAC file and import files to an Azure storage account.

Step 4: Generate an SAS key to the storage account.

Step 5: Complete the import specification.

#### NOTE

Before you perform a production import, we *strongly* recommend that you complete a dry-run import. With a dry run, you can validate that the import process works for your collection and that there are no unique data shapes present that might cause a production import failure.

## Step 1: Detach your collection

Detaching the collection is a crucial step in the import process. Identity data for the collection resides in the Azure DevOps Server instance's configuration database while the collection is attached and online. When a collection is detached from the Azure DevOps Server instance, it takes a copy of that identity data and packages it with the collection for transport. Without this data, the identity portion of the import *can't* be executed. We recommend that you keep the collection detached until the import has been completed, because there isn't a way to import the changes that occurred during the import.

If you're doing a dry run (test) import, we recommend that you reattach your collection after you back it up for import, because you won't be concerned about having the latest data for this type of import. To avoid offline time altogether, you can also choose to employ an [offline detach](#) for dry runs.

It's important to weigh the cost of choosing to incur zero downtime for a dry run. It requires taking backups of the collection and configuration database, restoring them on a SQL instance, and then creating a detached backup. A cost analysis could prove that taking just a few hours of downtime to directly take the detached backup is better in the long run.

## Step 2: Generate a DACPAC file

DACPACs offer a fast and relatively easy method for moving collections into Azure DevOps Services. However, after a collection database size exceeds a certain threshold, the benefits of using a DACPAC start to diminish.

#### NOTE

If the data migration tool displays a warning that you can't use the DACPAC method, you have to perform the import by using the SQL Azure virtual machine (VM) method provided in [Import large collections](#).

If the data migration tool doesn't display a warning, use the DACPAC method described in this step.

**DACPAC** is a feature of SQL server that allows database changes to be packaged into a single file and deployed to other instances of SQL. A DACPAC file can also be restored directly to Azure DevOps Services, so you can use it as the packaging method for getting your collection's data in the cloud. You use the SqlPackage.exe tool to generate the DACPAC file. The tool is included as part of [SQL Server Data Tools \(SSDT\)](#).

Multiple versions of the SqlPackage.exe tool are installed with SSDT. The versions are stored in folders with names such as 120, 130, and 140. When you use SqlPackage.exe, it's important to use the right version to prepare the DACPAC.

- TFS 2018 imports need to use the SqlPackage.exe version from the 140 folder or higher.

If you installed SSDT for Visual Studio, you'll find your SqlPackage.exe version in one of the following folder paths:

- If you installed SSDT and integrated it with an existing installation of Visual Studio, your SqlPackage.exe folder path is similar to

```
C:\Program Files (x86)\Microsoft Visual Studio 14.0\Common7\IDE\Extensions\Microsoft\SQLDB\DAC\130\ .
```

- If you installed SSDT as a standalone installation, your SqlPackage.exe folder path is similar to  
`C:\Program Files (x86)\Microsoft Visual Studio\2017\SQL\Common7\IDE\Extensions\Microsoft\SQLDB\DAC\130\ .`
- If you already have an installation of SQL Server, SqlPackage.exe might already be present, and your folder path is similar to `%PROGRAMFILES%\Microsoft SQL Server\130\DAC\bin\ .`

Both versions of SSDT that you can download from [SQL Server Data Tools](#) include both the 130 and 140 folders and their SqlPackage.exe versions.

When you generate a DACPAC, keep two considerations in mind: the disk that the DACPAC will be saved on and the disk space on the machine that's generating the DACPAC. You want to ensure that you have enough disk space to complete the operation.

As it creates the package, SqlPackage.exe temporarily stores data from your collection in the temp directory on drive C of the machine you're initiating the packaging request from.

You might find that your drive C is too small to support creating a DACPAC. You can estimate the amount of space you'll need by looking for the largest table in your collection database. DACPACs are created one table at a time. The maximum space requirement to run the generation is roughly equivalent to the size of the largest table in the collection's database. If you're saving the generated DACPAC to drive C, you also need to take into account the size of the collection database as reported in the *DataMigrationTool.log* file from a validation run.

The *DataMigrationTool.log* file provides a list of the largest tables in the collection each time the validate command is run. For an example of table sizes for a collection, see the following output. Compare the size of the largest table with the free space on the drive that hosts your temporary directory.

#### IMPORTANT

Before you proceed with generating a DACPAC file, ensure that your collection is [detached](#).

[Info @08:23:59.539] Table name	Size in MB
[Info @08:23:59.539] dbo.tbl_Content	38984
[Info @08:23:59.539] dbo.tbl_LocalVersion	1935
[Info @08:23:59.539] dbo.tbl_Version	238
[Info @08:23:59.539] dbo.tbl_FileReference	85
[Info @08:23:59.539] dbo.Rules	68
[Info @08:23:59.539] dbo.tbl_MetaData	61

Ensure that the drive that hosts your temporary directory has at least as much free space. If it doesn't, you need to redirect the temp directory by setting an environment variable.

```
SET TEMP={location on disk}
```

Another consideration is where the DACPAC data is saved. Pointing the save location to a far-off remote drive could result in much longer generation times. If a fast drive such as a solid-state drive (SSD) is available locally, we recommend that you target the drive as the DACPAC save location. Otherwise, it's always faster to use a disk that's on the machine where the collection database resides rather than a remote drive.

Now that you've identified the target location for the DACPAC and ensured that you have enough space, it's time to generate the DACPAC file.

Open a Command Prompt window and go to the SqlPackage.exe location. To generate the DACPAC, replace the placeholder values with the required values, and then run the following command:

```
SqlPackage.exe /sourceconnectionstring:"Data Source={database server name};Initial Catalog={Database Name};Integrated Security=True" /targetFile:{Location & File name} /action:extract /p:ExtractAllTableData=true /p:IgnoreUserLoginMappings=true /p:IgnorePermissions=true /p:Storage=Memory
```

- **Data Source:** The SQL Server instance that hosts your Azure DevOps Server collection database.
- **Initial Catalog:** The name of the collection database.
- **targetFile:** The location on the disk and the DACPAC file name.

A DACPAC generation command that's running on the Azure DevOps Server data tier itself is shown in the following example:

```
SqlPackage.exe /sourceconnectionstring:"Data Source=localhost;Initial Catalog=Foo;Integrated Security=True" /targetFile:C:\DACPAC\Foo.dacpac /action:extract /p:ExtractAllTableData=true /p:IgnoreUserLoginMappings=true /p:IgnorePermissions=true /p:Storage=Memory
```

The output of the command is a DACPAC file that's generated from the collection database *Foo* called *Foo.dacpac*.

#### Configure your collection for import

After your collection database has been restored on your Azure VM, configure a SQL login to allow Azure DevOps Services to connect to the database to import the data. This login allows only *read* access to a single database.

To start, open SQL Server Management Studio on the VM, and then open a new query window against the database to be imported.

Set the database's recovery to simple:

```
ALTER DATABASE [<Database name>] SET RECOVERY SIMPLE;
```

Create a SQL login for the database, and assign that login the 'TFSEXECROLE':

```
USE [<database name>]
CREATE LOGIN <pick a username> WITH PASSWORD = '<pick a password>'
CREATE USER <username> FOR LOGIN <username> WITH DEFAULT_SCHEMA=[dbo]
EXEC sp_addrolemember @rolename='TFSEXECROLE', @membername='<username>'
```

Following our Fabrikam example, the two SQL commands would look like the following:

```
ALTER DATABASE [Foo] SET RECOVERY SIMPLE;

USE [Foo]
CREATE LOGIN fabrikam WITH PASSWORD = 'fabrikamimport1!'
CREATE USER fabrikam FOR LOGIN fabrikam WITH DEFAULT_SCHEMA=[dbo]
EXEC sp_addrolemember @rolename='TFSEXECROLE', @membername='fabrikam'
```

#### NOTE

Be sure to enable [SQL Server and Windows authentication mode](#) in SQL Server Management Studio on the VM. If you don't enable authentication mode, the import will fail.

#### Configure the import specification file to target the VM

Update the import specification file to include information about how to connect to the SQL Server instance. Open your import specification file and make the following updates:

1. Remove the DACPAC parameter from the source files object.

The import specification before the change is shown in the following code:

```
"Source": {  
    "Location": "<Provide the SASKey to the Azure storage container with the collection and  
import files.>",  
    "Files": {  
        "Dacpac": "Tfs_DefaultCollection.dacpac"  
    }  
},
```

The import specification after the change is shown in the following code:

```
"Source": {  
    "Properties": {  
        "ConnectionString": "Data Source=8.8.8.8;Initial Catalog=Tfs_Foo;Integrated Security=False;  
User ID=fabrikam;Password=fabrikam1!;Encrypt=True;TrustServerCertificate=True"  
    }  
},
```

2. Fill out the required parameters and add the following properties object within your source object in the specification file.

```
"Properties":  
{  
    "ConnectionString": "Data Source={SQL Azure VM Public IP};Initial Catalog={Database  
Name};Integrated Security=False;User ID={SQL Login Username};Password={SQL Login  
Password};Encrypt=True;TrustServerCertificate=True"  
}
```

Following the Fabrikam example, after you apply the changes, the import specification would look like the following:

import.json - Untitled (Workspace) - Visual Studio Code

File Edit Selection View Go Debug Tasks Help

{} import.json x

```
1  [
2      "Source": {
3          "Properties": {
4              "ConnectionString": "Data Source=8.8.8.8;Initial Catalog=Tfs_Foo;Integrated Security=False;
5                  User ID=fabrikam;Password=fabrikam1!;Encrypt=True;TrustServerCertificate=True"
6          }
7      },
8      "Target": {
9          "Name": "fabrikam-import"
10     },
11     "Properties": {
12         "ImportType": "DryRun"
13     },
14     "ValidationData": {
15         "TfsMigratorVersion": "16.255.65000.0",
16         "SourceCollectionId": "8b245d37-d41d-4188-a6f1-b5bb397860ba",
17         "DataImportCollectionId": "ca970402-9b06-4720-9407-ba32684e9499",
18         "DatabaseCollation": "SQL_Latin1_General_CI_AS",
19         "CommandExecutionCount": 0,
20         "CommandExecutionTime": 0.0,
21         "TfsVersion": "Dev15.M117",
22         "DatabaseTotalSize": 181,
23         "DatabaseBlobSize": 0,
24         "DatabaseTableSize": 181,
25         "DatabaseLargestTableSize": 8,
26         "ActiveUserCount": 8,
27         "TenantId": "72f988bf-86f1-41af-91ab-2d7cd011db47",
28         "Region": "CUS",
29         "ValidationChecksumVersion": 1,
30         "ValidationChecksum": "66516G8u850KY6XKJm6MM5Ty3krNjhUFFCh4zyZMXqm7ZDLVpFpiIi0zDnJcoZmjHgDzvoCNS/9PwGm28hBgPg=="
31     },
32     "Identities": [
33         "S-1-5-21-1374400868-3601225936-2087002269-500",
34         "S-1-5-21-2127521184-1604012920-1887927527-11008431",
35         "S-1-5-21-2127521184-1604012920-1887927527-15795496"
36     ]
37 ]
```

Your import specification is now configured to use a SQL Azure VM for import. Proceed with the rest of preparation steps to import to Azure DevOps Services. After the import has finished, be sure to delete the SQL login or rotate the password. Microsoft does not retain the login information after the import has finished.

### **Step 3: Upload the DACPAC file**

## NOTE

If you're using the SQL Azure VM method, you need to provide only the connection string. You don't have to upload any files, and you can skip this step.

Your DACPAC must be placed in an Azure storage container. This can be an existing container or one created specifically for your migration effort. It's important to ensure that your container is created in the right region.

Azure DevOps Services is available in multiple [regions](#). When you're importing to these regions, it's critical to place your data in the correct region to ensure that the import can start successfully. Your data must be placed in the same region that you'll be importing to. Placing the data anywhere else will result in the import being unable to start. The following table lists the acceptable regions for creating your storage account and uploading your data.

DESIRED IMPORT REGION	STORAGE ACCOUNT REGION
Central United States	Central United States
Western Europe	Western Europe
Australia East	Australia East
Brazil South	Brazil South
India South	India South
Canada Central	Canada Central
Asia Pacific (Hong Kong)	Asia Pacific (Hong Kong)

Although Azure DevOps Services is available in multiple regions in the US, only the Central United States region accepts new Azure DevOps Services. You can't import your data into other US Azure regions at this time.

You can [create a blob container](#) from the Azure portal. After you've created the container, you need to upload the Collection DACPAC file.

After the import has finished, you can delete the blob container and accompanying storage account. To do so, you can use tools such as [AzCopy](#) or any other Azure storage explorer tool, such as [Azure Storage Explorer](#).

#### NOTE

If your DACPAC file is larger than 10 GB, we recommend that you use AzCopy. AzCopy has multithreaded upload support for faster uploads.

### Step 4: Generate an SAS key

A [shared access signature \(SAS\) key](#) provides delegated access to resources in a storage account. The key allows you to give Microsoft the lowest level of privilege that's required to access your data for executing the import.

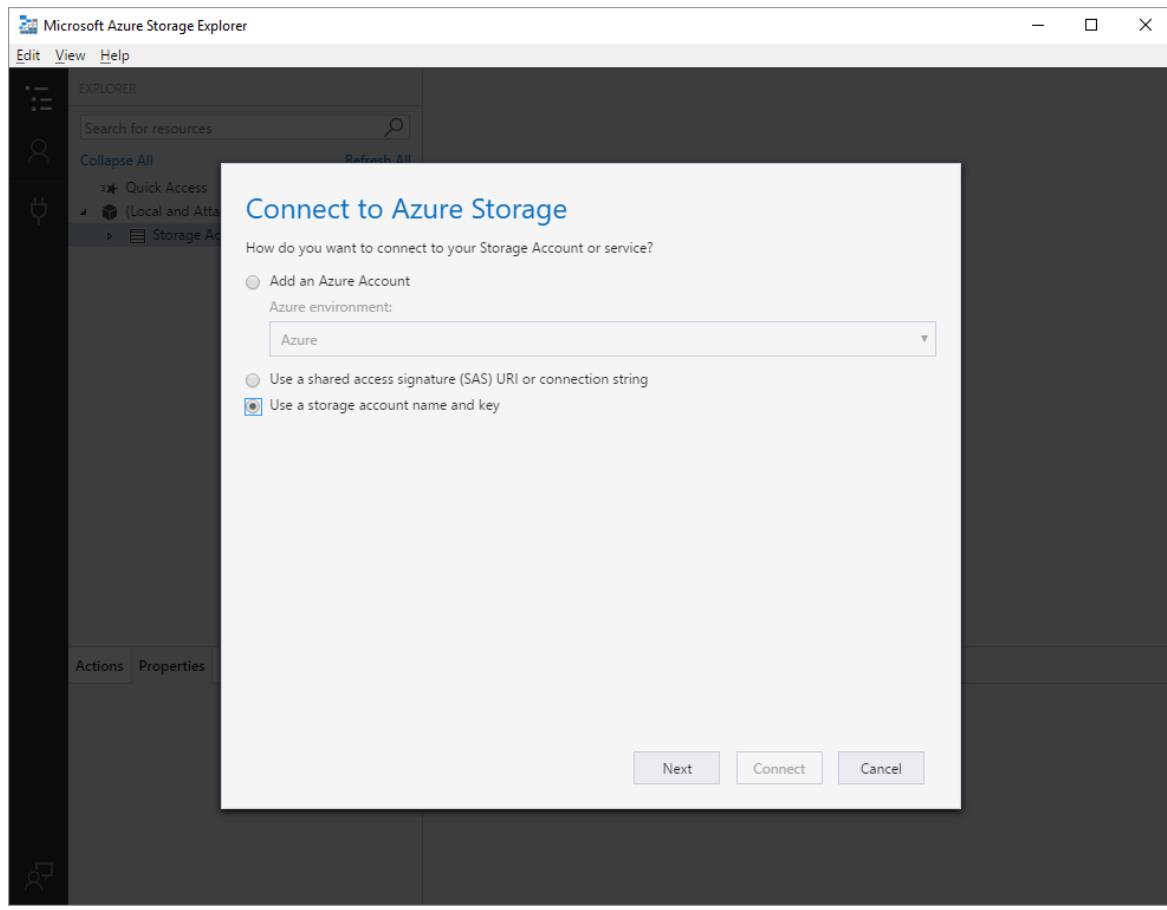
The recommended way to generate an SAS key is to use [Azure Storage Explorer](#). With Storage Explorer, you can easily create container-level SAS keys. This is essential, because the data migration tool does *not* support account-level SAS keys.

#### NOTE

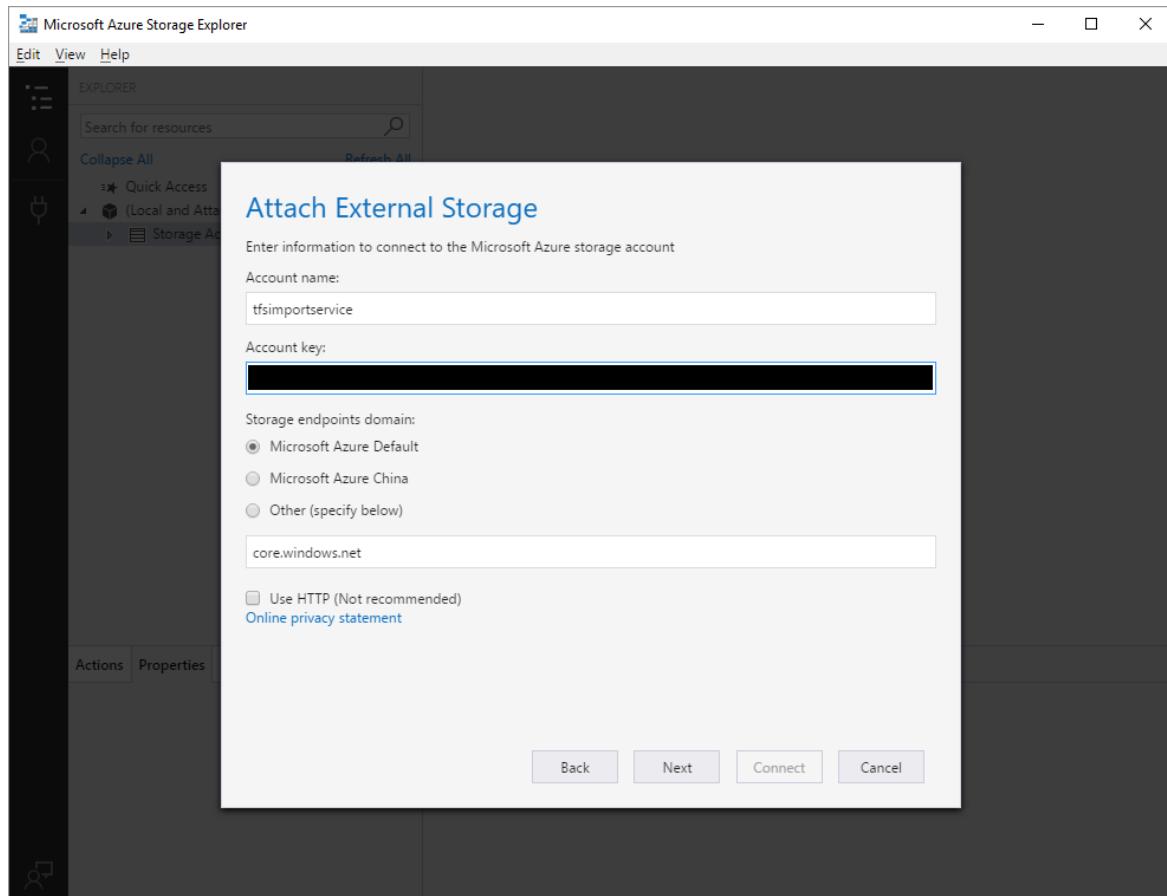
Do *not* generate an SAS key from the Azure portal. Azure portal-generated SAS keys are account scoped and don't work with the data migration tool.

After you install Storage Explorer, you can generate an SAS key by doing the following:

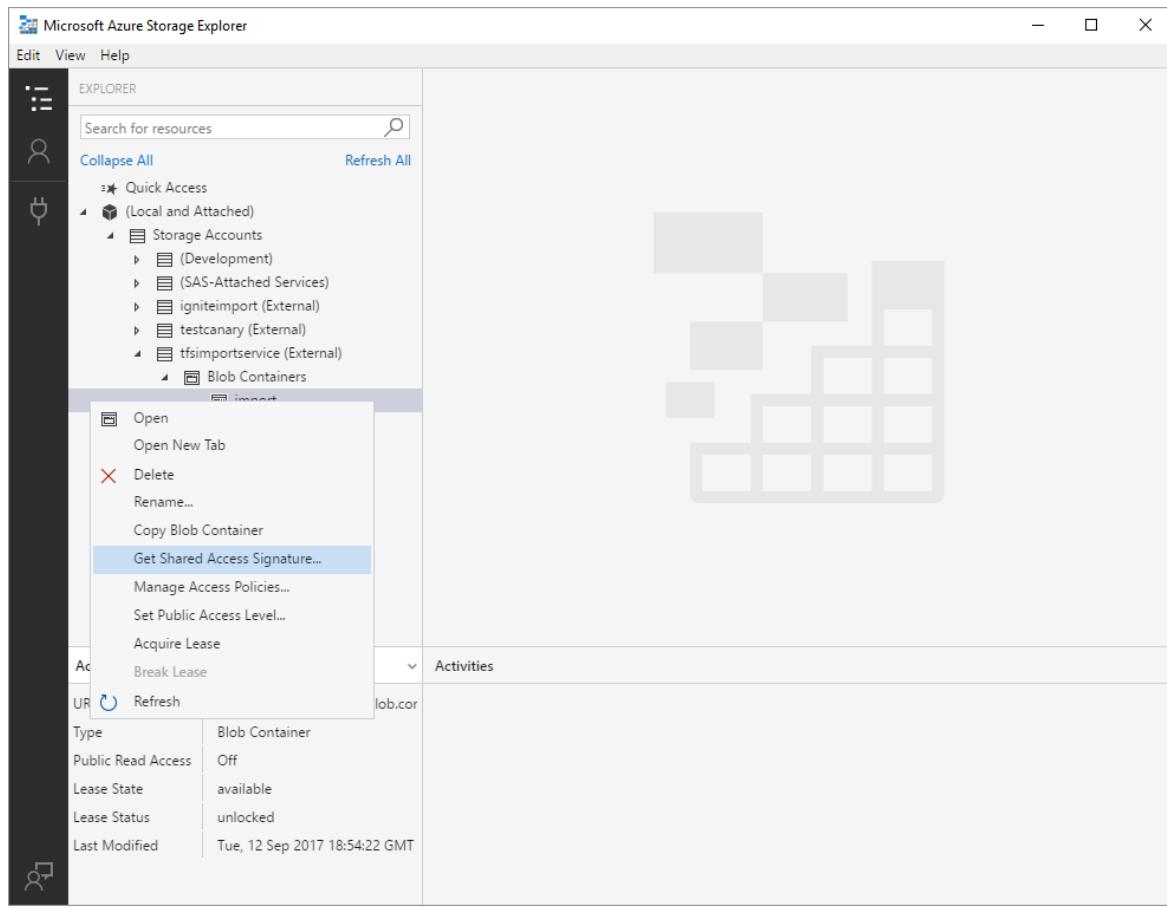
1. Open Storage Explorer.
2. Add an account.
3. Select **Use a storage account name and key**, and then select **Connect**.



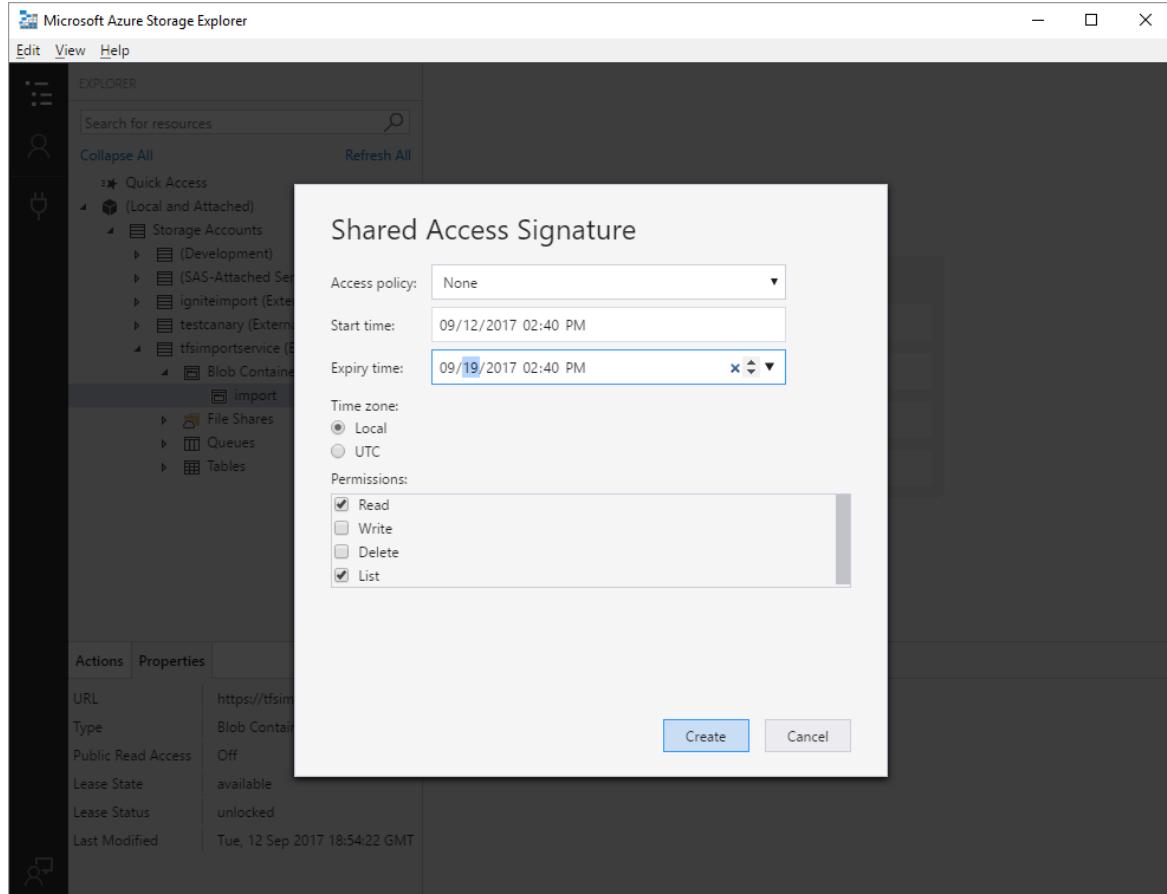
4. On the **Attach External Storage** pane, enter your storage account name, provide one of your two **primary access keys**, and then select **Connect**.



5. On the left pane, expand **Blob Containers**, right-click the container that stores your import files, and then select **Get Shared Access Signature**.



6. For **Expiry time**, set the expiration date for seven days in the future.



7. Under **Permissions** for your SAS key, select the **Read** and **List** check boxes. Write and delete permissions aren't required.

## NOTE

- Copy and store this SAS key to place in your import specification file in the next step.
  - Treat this SAS key as a secret. It provides access to your files in the storage container.

## Step 5: Complete the import specification

Earlier in the process you partially filled out the import specification file generally known as `import.json`. At this point, you have enough information to complete all the remaining fields except for the import type. The import type will be covered later, in the import section.

In the *import.json* specification file, under **Source**, complete the following fields:

- **Location:** Paste the SAS key you generated from the script and then copied in the preceding step.
  - **Dacpac:** Ensure that the file, including the `.dacpac` file extension, has the same name as the DACPAC file you uploaded to the storage account.

Using the Fabrikam example, the final import specification file should look like the following:

The screenshot shows the Visual Studio Code interface with the title bar "import.json - Untitled (Workspace) - Visual Studio Code". The menu bar includes File, Edit, Selection, View, Go, Debug, Tasks, and Help. On the left is a sidebar with icons for file operations like Open, Save, Find, and Refresh. The main editor area displays a JSON configuration file named "import.json". The file content is as follows:

```
{  
  "Source": {  
    "Location": "https://fabrikam.blob.core.windows.net/fabrikam?st=2017-08-28T17%3A15%3A00Z&se=2017-08-29T17%3A15%3A00Z&sp=rl&sv=2015-04-05&sr=c&sig=FSPF9sgifCsmAWyZ0icd009cxrmG9VP5pknFN16MgpY%3D",  
    "Files": [  
      "Dacpac": "Tfs_DefaultCollection.dacpac"  
    ]  
  },  
  "Target": {  
    "Name": "fabrikam-import"  
  },  
  "Properties": {  
    "ImportType": "<Provide the Type of Import: DryRun, ProductionRun>"  
  },  
  "ValidationData": {  
    "TfsMigratorVersion": "16.255.65000.0",  
    "SourceCollectionId": "8b245d37-d41d-4188-a6f1-b5bb397860ba",  
    "DataImportCollectionId": "ca970401-9b06-4720-9407-ba32684e9499",  
    "DatabaseCollation": "SQL_Latin1_General_CI_AS",  
    "CommandExecutionCount": 0,  
    "CommandExecutionTime": 0.0,  
    "TfsVersion": "Dev15.M117",  
    "DatabaseTotalSize": 181,  
    "DatabaseBlobSize": 0,  
    "DatabaseTableSize": 181,  
    "DatabaseLargestTableSize": 8,  
    "ActiveUserCount": 8,  
    "TenantId": "72f988bf-86f1-41af-91ab-2d7cd011db47",  
    "Region": "CUS",  
    "ValidationChecksumVersion": 1,  
    "ValidationChecksum": "  
      "66516G8u850KY6XKj6MMSTy3krNjhUFFCh4zyZMXqm7ZDLVpFpiIi0zDnJcoZmjHgDzvoCNS/9PwGm28hBgPg=="  
    ",  
    "Identities": [  
      "S-1-5-21-1374400868-3601225936-2087002269-500",  
      "S-1-5-21-2127521184-1604012920-1887927527-11008431",  
      "S-1-5-21-2127521184-1604012920-1887927527-15795496"  
    ]  
}
```

## Determine the import type

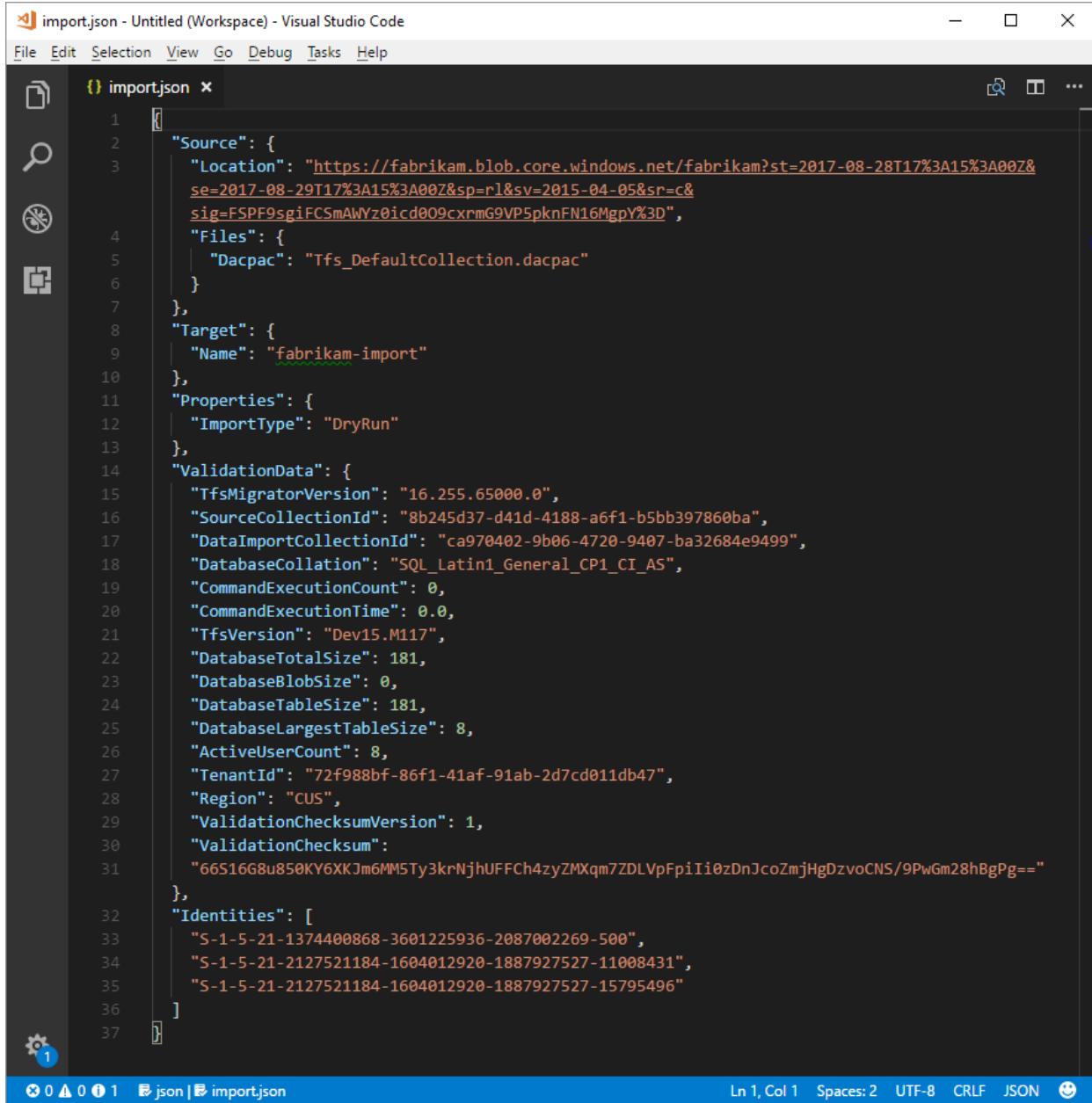
Imports can be queued as either a dry run or a production run. The `ImportType` parameter determines the type of import.

import type:

- **DryRun**: Use a dry run for test purposes. The system deletes dry runs after 21 days.
- **ProductionRun**: Use a production run when you want to keep the resulting import and use the organization full time in Azure DevOps Services after the import finishes.

#### TIP

We always recommend that you complete a dry-run import first.



The screenshot shows a Visual Studio Code window with a dark theme. The title bar reads "import.json - Untitled (Workspace) - Visual Studio Code". The editor pane displays the following JSON code:

```
1  {
2      "Source": {
3          "Location": "https://fabrikam.blob.core.windows.net/fabrikam?st=2017-08-28T17%3A15%3A00Z&se=2017-08-29T17%3A15%3A00Z&sp=r&sv=2015-04-05&sr=c&sig=FSPF9sg1FCSmAwYz0icd009cxrmG9VP5pknFN16MgpY%3D",
4          "Files": {
5              "Dacpac": "Tfs_DefaultCollection.dacpac"
6          }
7      },
8      "Target": {
9          "Name": "fabrikam-import"
10     },
11     "Properties": {
12         "ImportType": "DryRun"
13     },
14     "ValidationData": {
15         "TfsMigratorVersion": "16.255.65000.0",
16         "SourceCollectionId": "8b245d37-d41d-4188-a6f1-b5bb397860ba",
17         "DataImportCollectionId": "ca970402-9b06-4720-9407-ba32684e9499",
18         "DatabaseCollation": "SQL_Latin1_General_CI_AS",
19         "CommandExecutionCount": 0,
20         "CommandExecutionTime": 0.0,
21         "TfsVersion": "Dev15.M117",
22         "DatabaseTotalSize": 181,
23         "DatabaseBlobSize": 0,
24         "DatabaseTableSize": 181,
25         "DatabaseLargestTableSize": 8,
26         "ActiveUserCount": 8,
27         "TenantId": "72f988bf-86f1-41af-91ab-2d7cd011db47",
28         "Region": "CUS",
29         "ValidationChecksumVersion": 1,
30         "ValidationChecksum": "66516GBu850KY6XKJm6MM5Ty3krNjhUFFCh4zyZMXqm7ZDLVpFpiII0zDnJcoZmjHgDzvoCNS/9PwGm28hBgPg=="
31     },
32     "Identities": [
33         "S-1-5-21-1374400868-3601225936-2087002269-500",
34         "S-1-5-21-2127521184-1604012920-1887927527-11008431",
35         "S-1-5-21-2127521184-1604012920-1887927527-15795496"
36     ]
37 }
```

The status bar at the bottom shows: 0 0 0 1 json | import.json Ln 1, Col 1 Spaces: 2 UTF-8 CRLF JSON 😊

## Dry-run organizations

Dry-run imports help teams test the migration of their collections. Organizations are expected not to remain around forever but to exist for a short time. In fact, before a production migration can be run, any completed dry-run organizations will need to be deleted. All dry-run organizations have a *limited existence and are automatically deleted after a set period of time*. Information about when the organization will be deleted is included in the success email you should receive after the import finishes. Be sure to take note of this date and plan accordingly.

Most dry-run organizations have 15 days before they're deleted. Dry-run organizations can also have a 21-day expiration if more than 100 users have a basic or greater license at *import time*. After the specified time period,

the dry-run organization is deleted. You can repeat dry-run imports as many times as you need before you do a production migration. You need to delete any previous dry runs before you attempt a new one. When your team is ready to perform a production migration, you'll need to manually delete the dry-run organization.

For more information about post-import activities, see the [post import](#) article.

If you encounter any import problems, see [Troubleshoot import and migration errors](#).

## Run an import

Your team is now ready to begin the process of running an import. We recommend that you start with a successful dry-run import before you attempt a production-run import. With dry-run imports, you can see in advance how an import will look, identify potential issues, and gain experience before you head into your production run.

### NOTE

If you need to repeat a completed production-run import for a collection, as in the event of a rollback, contact Azure DevOps Services [Customer Support](#) before you queue up another import.

### NOTE

Azure administrators can prevent users from creating new Azure DevOps organizations. If the Azure AD tenant policy is turned on, your import will fail to finish. Before you begin, verify that the policy isn't set or that there is an exception for the user that is performing the migration. For more information, see [Restrict organization creation via Azure AD tenant policy](#).

## Considerations for rollback plans

A common concern for teams that are doing a final production run is what their rollback plan will be if anything goes wrong with import. This is why we highly recommend doing a dry run to make sure that you're able to test the import settings you provide to the data migration tool for Azure DevOps.

Rollback for the final production run is fairly simple. Before you queue the import, you detach the team project collection from Azure DevOps Server or Team Foundation Server, which will make it unavailable to your team members. If for any reason you need to roll back the production run and bring the on-premises server back online for your team members, you can do so. You simply attach the team project collection on-premises again and inform your team that they'll continue to work normally while your team regroups to understand any potential failures.

## Queue an import

### IMPORTANT

Before you proceed, ensure that your collection was [detached](#) prior to generating a DACPAC file or uploading the collection database to a SQL Azure VM. If you don't complete this step, the import will fail. In the event that your import fails, see [Troubleshoot import and migration errors](#).

You start an import by using the data migration tool's `import` command. The import command takes an import specification file as input. It parses the file to ensure that the provided values are valid and, if successful, it queues an import to Azure DevOps Services. The import command requires an internet connection, but does *not* require a connection to your Azure DevOps Server instance.

To get started, open a Command Prompt window, and change directories to the path to the data migration tool. We recommended that you take a moment to review the help text provided with the tool. Run the following

command to see the guidance and help for the import command:

```
Migrator import /help
```

The command to queue an import will have the following structure:

```
Migrator import /importFile:{location of import specification file}
```

Here is an example of a completed import command:

```
Migrator import /importFile:C:\DataMigrationToolFiles\import.json
```

After the validation passes, you'll be asked to sign in to Azure AD. It's important to sign in with an identity that's a member of the same Azure AD tenant as the identity map log file was built against. The user that signs in becomes the owner of the imported organization.

**NOTE**

Each Azure AD tenant is limited to five imports per 24-hour period. Only imports that are queued count against this cap.

When your team initiates an import, an email notification is sent to the user that queued the import. About 5 to 10 minutes after it queues the import, your team can go to the organization to check on the status. After the import finishes, your team is directed to sign in, and an email notification is sent to the organization owner.

## Related articles

- [Migrate options](#)
- [Post-import](#)

# Import large collections

4/2/2021 • 12 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server | TFS

For databases that the data migration tool warns are too big, a different data packaging approach is required to migrate to Azure DevOps Services. If you're unsure whether your collection exceeds the size threshold, you should run a data migration tool validation on the collection. The validation lets you know whether you need to use the SQL Azure VM method for import.

## Determine if you can reduce the collection size

Before you proceed, we recommend checking to see whether your [old data can be cleaned up](#). Over time, collections can build up very large volumes of data. This is a natural part of the DevOps process, but you might find that you don't need to retain all of the data. Some common examples of no longer relevant data are older workspaces and build results.

Cleaning older, no-longer-relevant artifacts could remove a lot more space than you might expect, and it could determine whether you use the DACPAC import method or a SQL Azure VM.

### IMPORTANT

After you've deleted older data, it *can't* be recovered unless you restore an older backup of the collection.

If you're under the DACPAC threshold, follow the instructions to [generate a DACPAC]](migration-import.md#dacpac-file) for import. If you still can't get the database under the DACPAC threshold, you need to set up a SQL Azure VM to import to Azure DevOps Services.

## Set up a SQL Azure VM to import to Azure DevOps Services

Let's walk through how to accomplish this. At a high level, you'll:

- Set up a SQL Azure VM.
- (Optional) Restrict access to Azure DevOps Services IPs only.
- Configure IP firewall exceptions.
- Restore your database on the VM.
- Configure your collection for import.
- Configure the import specification file to target the VM

## Set up a SQL Azure VM

You can set up a SQL Azure VM from the Azure portal with just a few clicks. To learn how, see [Use the Azure portal to provision a Windows virtual machine with SQL Server](#).

Azure DevOps Services is available in several [Azure regions](#) across the globe.

## IMPORTANT

To ensure that the import starts successfully, it's critical to place your data in the correct region. If you set up your SQL Azure VM in a location other than the regions listed in the following table, the import will fail to start.

If you're using this import method, determine where to create your SQL Azure VM by referring to the following table. Creating your VM in a region other than those in this list is not supported for running an import.

DESIRED IMPORT REGION	SQL AZURE VM REGION
Central United States	Central US, East US, East US 2, North Central US, South Central US, West Central US, West US, West US 2
Western Europe	North Europe, West Europe
Australia East	Australia Central, Australia East, Australia Southeast
Brazil South	Brazil South
South India	Central India, South India, West India
Central Canada	Canada Central, Canada East
Asia Pacific (Hong Kong)	East Asia, Southeast Asia
UK South	UK South, UK West

Although Azure DevOps Services is available in multiple regions in the US, only the Central United States region accepts new organizations. Companies can't import their data into other US Azure regions at this time.

## NOTE

DACPAC customers should consult the region table in the ["Step 3: Upload the DACPAC file" section](#). The preceding guidelines are for SQL Azure VMs only.

Here are a few more SQL Azure VM configurations that we recommend:

- Use D Series VMs, because they're optimized for database operations.
- Ensure that the D Series VMs have at least 28 gigabytes (GB) of RAM. For imports, we recommend Azure D12 V2 VM sizes.
- [Configure the SQL temporary database](#) to use a drive other than drive C. Ideally the drive should have ample free space; at least equivalent to your database's [largest table](#).
- If your source database is still over 1 terabyte (TB) after you've [reduced its size](#), you need to [attach additional 1-TB disks](#) and combine them into a single partition to restore your database on the VM.
- If your collection databases are over 1 TB in size, consider using an SSD for both the temporary database and collection database. Also, consider using larger VMs with 16 virtual CPUs (vCPUs) and 128 GB of RAM.
- You need to have a public facing IP address for the service to reach this machine.

(Optional) Restrict access to Azure DevOps Services IPs only

We highly recommend that you restrict access to your VM to only IPs from Azure DevOps Services. You do this by allowing connections only from the set of Azure DevOps Services IPs that are involved in the collection database import process. The IPs that need to be granted access to your collection database depend on the region you're importing into. The following tables can help you identify the correct IPs. The only port that's required to be opened to connections is the standard SQL connection port 1433.

First, no matter what Azure DevOps Services region you import into, you must grant the following IP addresses access to your collection database.

**NOTE**

In the following table, the two IP addresses listed with x.x.x.0/23 indicate a range. Allow the entire /23 range. For example, if you're importing into the Central United States region, allow the /23 range for 20.37.158.0. For IP addresses with x.x.x.0/24, allow the /24 range.

SERVICE	IP ADDRESS
Azure DevOps Services Identity Service	168.62.105.45, 40.81.42.115

Next, grant access to the Regional Identity Service. You need to grant an exception for the data migration tool instance only in the region that you're importing into.

SERVICE	IP ADDRESS
Regional Identity Service - Central United States	13.89.236.72, 52.165.41.252, 52.173.25.16, 13.86.38.60, 20.45.1.175, 13.86.36.181, 52.154.53.1, 52.158.209.56, 20.37.138.122, 20.37.158.0/23, 20.37.139.247, 20.37.158.5
Regional Identity Service - West Europe	20.67.123.240, 52.166.54.85, 13.95.233.212, 52.236.145.119, 52.142.235.223, 52.236.147.103, 23.97.221.25, 52.233.181.148, 52.149.110.153, 51.144.61.32, 52.236.147.236, 40.74.28.0/23
Regional Identity Service - Australia East	13.75.145.145, 40.82.217.103, 20.188.213.113, 104.210.88.194, 40.81.62.114, 20.37.194.0/24
Regional Identity Service - Brazil South	20.40.114.3, 191.235.90.183, 191.232.38.181, 191.233.25.175, 191.235.226.0/24
Regional Identity Service - India South	104.211.227.29, 40.81.75.130, 52.172.54.122, 52.172.49.252, 20.41.194.0/24
Regional Identity Service - Canada Central	52.237.19.6, 40.82.190.38, 52.228.82.0/243
Regional Identity Service - Asia Pacific (Hong Kong)	52.175.28.40, 40.81.25.218, 13.94.26.58, 20.189.107.0/24
Regional Identity Service - UK South	40.81.159.67, 51.104.26.0/24

Next, grant access to the data migration tool for Azure DevOps itself. You need to grant an exception for the data migration tool instance only in the region that you're importing into.

SERVICE	IP ADDRESS
Data migration tool - Central United States	52.173.74.9, 52.165.184.188, 20.45.1.234, 13.86.39.123
Data migration tool - West Europe	40.115.43.138, 13.95.15.128, 52.236.146.105, 40.67.219.89, 40.119.145.63, 52.142.236.228, 52.142.238.75
Data migration tool - Australia East	13.75.134.204, 40.82.219.41, 20.40.124.19
Data migration tool - Brazil South	104.41.24.164, 20.40.115.123
Data migration tool - India South	13.71.120.31, 40.81.76.137
Data migration tool - Canada Central	52.237.18.100, 52.237.24.61, 40.82.191.163
Data migration tool - Asia Pacific (Hong Kong)	13.75.106.194, 40.81.27.181
Data migration tool - UK South	40.81.153.223, 51.105.8.98, 51.104.26.2

Next, grant Azure DevOps Services access. Again, you need to grant an exception for the Azure DevOps Services instance only in the region that you're importing into.

SERVICE	IP ADDRESS
Azure DevOps Services - Central United States	13.89.236.72, 52.165.41.252, 52.173.25.16, 13.86.38.60, 20.45.1.175, 13.86.36.181, 52.158.209.56
Azure DevOps Services - West Europe	52.166.54.85, 13.95.233.212, 52.236.145.119, 52.142.235.223, 52.236.147.103, 23.97.221.25, 52.233.181.148, 52.149.110.153, 51.144.61.32, 52.236.147.236
Azure DevOps Services - Australia East	13.75.145.145, 40.82.217.103, 20.188.213.113, 104.210.88.194, 40.81.62.114
Azure DevOps Services - Brazil South	20.40.114.3, 191.235.90.183, 191.232.38.181, 191.233.25.175
Azure DevOps Services - India South	104.211.227.29, 40.81.75.130, 52.172.54.122, 52.172.49.252
Azure DevOps Services - Canada Central	52.237.19.6, 40.82.190.38
Azure DevOps Services - Asia Pacific (Hong Kong)	52.175.28.40, 40.81.25.218, 13.94.26.58
Azure DevOps Services - UK South	40.81.159.67, 51.105.8.98, 51.104.26.2, 51.104.26.5

Next, grant Azure Pipelines Releases service access. You need to grant an exception for the Azure DevOps Services instance only in the region that you're importing into.

## Release Management IPs

SERVICE	IP ADDRESS
Releases service - United States	23.102.153.83, 23.101.127.247, 23.100.85.250, 13.86.39.233, 40.80.217.53, 52.232.229.122
Releases service - West Europe	13.95.223.69, 104.45.64.13
Releases service - Australia East	13.73.204.151, 20.40.176.135
Releases service - Brazil South	191.235.94.154, 20.40.116.69
Releases service - India South	52.172.15.233, 40.81.79.60
Releases service - Canada Central	52.237.28.171, 40.82.189.127
Releases service - Asia Pacific (Hong Kong)	13.107.6.175, 40.81.29.43
Releases service - UK South	40.81.156.207

Next, grant Azure Artifacts access. Again, you need to grant an exception for the Azure DevOps Services instance only in the region that you're importing into.

### Azure Artifacts IPs

Add exceptions for all three services that make up Azure Artifacts.

SERVICE	IP ADDRESS
Azure Artifacts - United States	52.173.148.93, 104.43.253.181, 23.99.179.148, 40.80.222.154, 40.119.0.130, 40.119.0.139, 13.86.125.169, 20.41.44.47, 40.90.219.165
Azure Artifacts - West Europe	104.46.45.12, 52.236.148.212
Azure Artifacts - Australia East	13.73.100.166, 20.40.176.15, 40.81.59.69
Azure Artifacts - Brazil South	191.234.179.224, 20.40.115.214
Azure Artifacts - India South	52.172.11.191, 40.81.74.79
Azure Artifacts - Canada Central	52.237.24.224, 40.85.224.121, 13.71.189.199, 40.82.188.122
Azure Artifacts - Asia Pacific (Hong Kong)	52.229.175.18, 65.52.162.53, 40.83.74.71, 40.81.27.130
Azure Artifacts - UK South	51.145.120.132

SERVICE	IP ADDRESS
Azure Artifacts Feed - United States	52.173.251.89, 20.45.1.3, 40.67.190.224, 20.41.58.125, 40.119.1.14, 20.45.1.249

SERVICE	IP ADDRESS
Azure Artifacts Feed - West Europe	40.118.19.43, 52.236.146.118
Azure Artifacts Feed - Australia East	13.70.143.138, 20.40.176.80
Azure Artifacts Feed - Brazil South	191.235.93.87, 20.40.116.17
Azure Artifacts Feed - India South	52.172.8.41, 40.81.79.49
Azure Artifacts Feed - Canada Central	52.237.19.70, 40.82.188.254
Azure Artifacts Feed - Asia Pacific (Hong Kong)	52.229.163.155, 40.81.28.59, 40.81.59.77
Azure Artifacts Feed - UK South	51.145.120.49

SERVICE	IP ADDRESS
Azure Artifacts Blob - United States	70.37.94.103, 40.78.129.25, 40.67.155.236, 52.230.216.163, 20.45.3.51
Azure Artifacts Blob - West Europe	23.97.221.25
Azure Artifacts Blob - Australia East	40.127.86.30, 20.188.213.113, 40.82.221.14
Azure Artifacts Blob - Brazil South	191.235.90.183
Azure Artifacts Blob - India South	52.172.54.122
Azure Artifacts Blob - Canada Central	52.237.16.145, 52.237.16.145, 52.233.38.115, 40.82.187.186
Azure Artifacts Blob - Asia Pacific (Hong Kong)	13.94.26.58
Azure Artifacts Blob - UK South	51.143.174.59, 40.81.152.41

## Test Plans IPs

Add exceptions for Test Plans IP addresses only in the region you're migrating into.

SERVICE	IP ADDRESS
Test Plans - United States	52.253.227.131, 40.91.89.233, 20.41.47.199, 40.91.117.40, 40.91.126.113, 20.37.141.154
Test Plans - West Europe	40.119.145.57
Test Plans - Australia East	20.40.177.101
Test Plans - Brazil South	20.40.118.62

SERVICE	IP ADDRESS
Test Plans - India South	40.81.72.10
Test Plans - Canada Central	40.82.184.28
Test Plans - Asia Pacific (Hong Kong)	52.184.81.26
Test Plans - UK South	40.81.159.9

### Analytics IPs (Azure DevOps Server 2019 or later only)

If you included preview features with your import, add an exception for the analytics IPs only in your target import region.

SERVICE	IP ADDRESS
Analytics service - United States	20.41.43.22, 20.36.236.83, 20.41.40.50, 52.143.251.221, 52.242.212.199, 13.86.33.148, 13.86.39.80
Analytics service - West Europe	52.236.146.143, 52.236.146.9, 52.149.108.23
Analytics service - Australia East	20.40.179.159
Analytics service - Brazil South	20.40.113.248
Analytics service - India South	40.81.73.58
Analytics service - Canada Central	40.82.185.214
Analytics service - Asia Pacific (Hong Kong)	40.81.25.239
Analytics service - UK South	40.81.159.247

## Configure IP firewall exceptions

Granting exceptions for the necessary IPs is handled at the Azure networking layer for your SQL Azure VM. To get started, go to your SQL Azure VM in the [Azure portal](#). In **Settings**, select **Networking**. This will take you to the network interface page for your SQL Azure VM. The data migration tool requires the Azure DevOps Services IPs to be configured for inbound connections only on port 1431. You can grant exceptions for the IPs by selecting **Add inbound port rule** in the networking settings.

PROTOCOL	SOURCE	DESTINATION	ACTION	
TCP	Any	Any	Allow	...
TCP	Any	Any	Allow	...
Any	VirtualNetwork	VirtualNetwork	Allow	...
Any	AzureLoadBalancer	Any	Allow	...
Any	Any	Any	Deny	...

On the Add inbound security rule pane, select **Advanced** to configure an inbound port rule for a specific IP.

Add inbound security rule X

**Advanced**  

**Service** i

Custom ▼

**\* Port range** i

8080 ✓

**\* Priority** i

1510

**\* Name**

Port\_8080 ✓

**Description**

In the **Source** drop-down list, select **IP Addresses**, enter an IP address that needs to be granted an exception, set the **Destination port range** to **1433** and, in the **Name** box, enter a name that best describes the exception you're configuring.

Depending on other inbound port rules that have been configured, you might need to change the default priority for the Azure DevOps Services exceptions so they don't get ignored. For example, if you have a "deny on all inbound connections to 1433" rule with a higher priority than your Azure DevOps Services exceptions, the data migration tool might be unable to make a successful connection to your database.

\* Source ⓘ

IP Addresses ▾

\* Source IP address range ⓘ

168.62.105.45 ✓

\* Source port range ⓘ

\*

\* Destination ⓘ

Any ▾

\* Destination port range ⓘ

1433 ✓

\* Protocol

Any TCP UDP

\* Action

Allow Deny

\* Priority ⓘ

1010 ✓

\* Name

VSTS\_Identity\_Service ✓

Description

[Empty text area]

Repeat adding inbound port rules until all necessary Azure DevOps Services IPs have been granted an exception. Missing one IP could result in your import failing to start.

## Restore your database on the VM

After you set up and configure an Azure VM, you need to take your detached backup from your Azure DevOps Server instance to your Azure VM. Azure has [several documented methods](#) for how to accomplish this task. The collection database needs to be restored on your SQL instance and doesn't require Azure DevOps Server to be installed on the VM.

## Configure your collection for import

After your collection database has been restored on your Azure VM, configure a SQL login to allow Azure DevOps Services to connect to the database to import the data. This login allows only *read* access to a single

database.

To start, open SQL Server Management Studio on the VM, and then open a new query window against the database to be imported.

Set the database's recovery to simple:

```
ALTER DATABASE [<Database name>] SET RECOVERY SIMPLE;
```

Create a SQL login for the database, and assign that login the 'TFSEXECROLE':

```
USE [<database name>]
CREATE LOGIN <pick a username> WITH PASSWORD = '<pick a password>'
CREATE USER <username> FOR LOGIN <username> WITH DEFAULT_SCHEMA=[dbo]
EXEC sp_addrolemember @rolename='TFSEXECROLE', @membername='<username>'
```

Following our Fabrikam example, the two SQL commands would look like the following:

```
ALTER DATABASE [Foo] SET RECOVERY SIMPLE;

USE [Foo]
CREATE LOGIN fabrikam WITH PASSWORD = 'fabrikamimport1!'
CREATE USER fabrikam FOR LOGIN fabrikam WITH DEFAULT_SCHEMA=[dbo]
EXEC sp_addrolemember @rolename='TFSEXECROLE', @membername='fabrikam'
```

#### NOTE

Be sure to enable [SQL Server and Windows authentication mode](#) in SQL Server Management Studio on the VM. If you don't enable authentication mode, the import will fail.

## Configure the import specification file to target the VM

Update the import specification file to include information about how to connect to the SQL Server instance.

Open your import specification file and make the following updates:

1. Remove the DACPAC parameter from the source files object.

The import specification before the change is shown in the following code:

```
"Source": {
  "Location": "<Provide the SASKey to the Azure storage container with the collection and import files.>",
  "Files": {
    "Dacpac": "Tfs_DefaultCollection.dacpac"
  }
},
```

The import specification after the change is shown in the following code:

```
"Source": {
  "Properties": {
    "ConnectionString": "Data Source=8.8.8.8;Initial Catalog=Tfs_Foo;Integrated Security=False;
User ID=fabrikam;Password=fabrikam1!;Encrypt=True;TrustServerCertificate=True"
  }
},
```

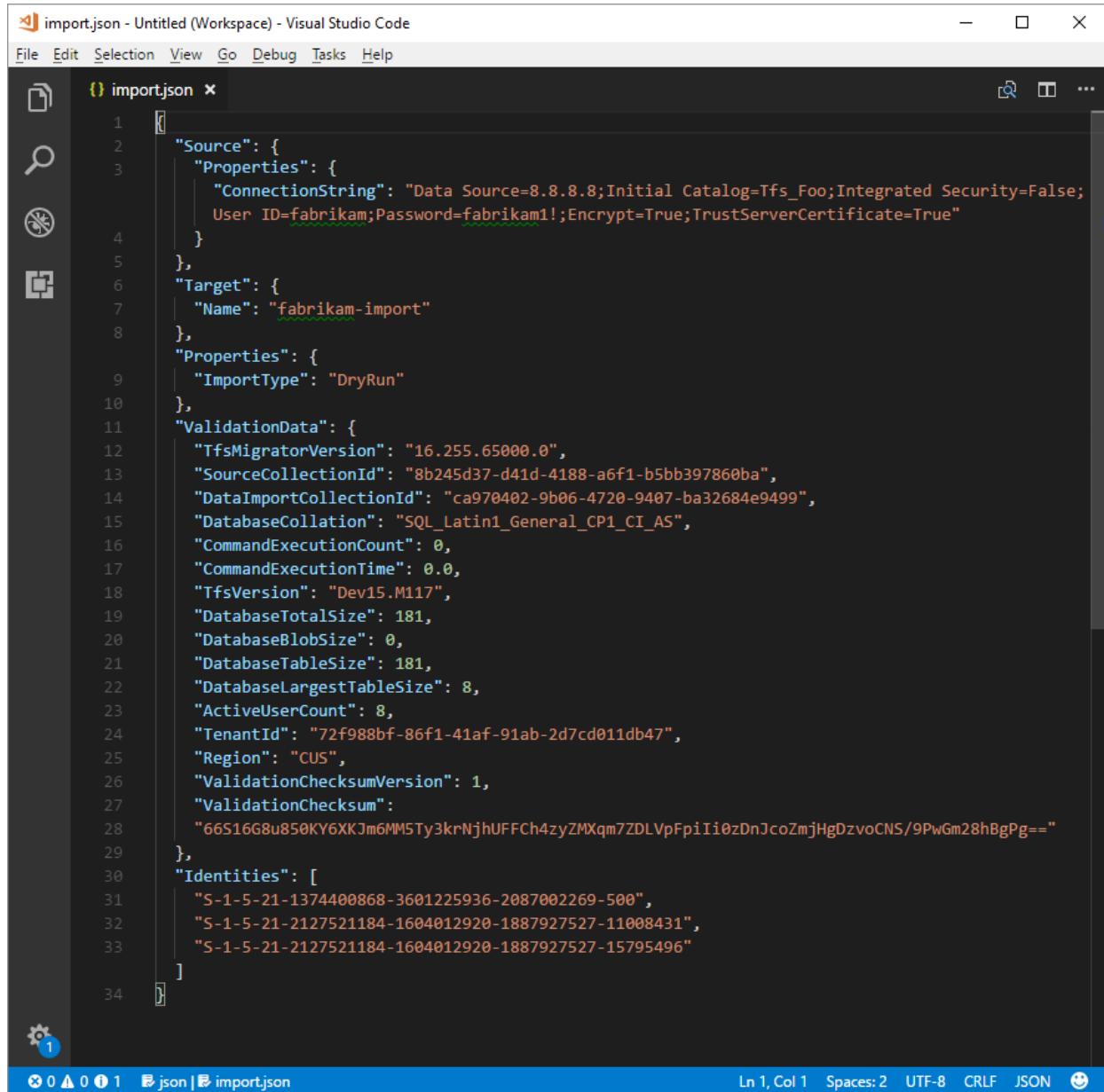
2. Fill out the required parameters and add the following properties object within your source object in the specification file.

```

"Properties":
{
    "ConnectionString": "Data Source={SQL Azure VM Public IP};Initial Catalog={Database
Name};Integrated Security=False;User ID={SQL Login Username};Password={SQL Login
Password};Encrypt=True;TrustServerCertificate=True"
}

```

Following the Fabrikam example, after you apply the changes, the import specification would look like the following:



```

import.json - Untitled (Workspace) - Visual Studio Code
File Edit Selection View Go Debug Tasks Help
import.json x
1 [
2     "Source": {
3         "Properties": {
4             "ConnectionString": "Data Source=8.8.8.8;Initial Catalog=Tfs_Foo;Integrated Security=False;
5             User ID=fabrikam;Password=fabrikam1!;Encrypt=True;TrustServerCertificate=True"
6         }
7     },
8     "Target": {
9         "Name": "fabrikam-import"
10    },
11    "Properties": {
12        "ImportType": "DryRun"
13    },
14    "ValidationData": {
15        "TfsMigratorVersion": "16.255.65000.0",
16        "SourceCollectionId": "8b245d37-d41d-4188-a6f1-b5bb397860ba",
17        "DataImportCollectionId": "ca970402-9b06-4720-9407-ba32684e9499",
18        "DatabaseCollation": "SQL_Latin1_General_CI_AS",
19        "CommandExecutionCount": 0,
20        "CommandExecutionTime": 0.0,
21        "TfsVersion": "Dev15.M117",
22        "DatabaseTotalSize": 181,
23        "DatabaseBlobSize": 0,
24        "DatabaseTableSize": 181,
25        "DatabaseLargestTableSize": 8,
26        "ActiveUserCount": 8,
27        "TenantId": "72f988bf-86f1-41af-91ab-2d7cd011db47",
28        "Region": "CUS",
29        "ValidationChecksumVersion": 1,
30        "ValidationChecksum":
31            "66S16G8u850KY6XKJm6MM5Ty3krNjhUFFCh4zyZMXqm7ZDLVpFpiIi0zDnJcoZmjHgDzvoCNS/9PwGm28hBgPg=="
32    },
33    "Identities": [
34        "S-1-5-21-1374400868-3601225936-2087002269-500",
35        "S-1-5-21-2127521184-1604012920-1887927527-11008431",
36        "S-1-5-21-2127521184-1604012920-1887927527-15795496"
37    ]
38]

```

0 1 json import.json Ln 1, Col 1 Spaces: 2 UTF-8 CRLF JSON 😊

Your import specification is now configured to use a SQL Azure VM for import. Proceed with the rest of preparation steps to import to Azure DevOps Services. After the import has finished, be sure to delete the SQL login or rotate the password. Microsoft does not retain the login information after the import has finished.

## Related articles

- [Validation and import processes](#)

# Validate and resolve errors related to process templates

4/2/2021 • 9 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server | TFS

As part of the migration import process, the data migration tool checks the process used by the projects in the collection. Fix any errors that get flagged.

After resolving the errors, rerun the data migration tool's `validate` command to verify that all errors have been fixed.

## NOTE

It's recommended that you use the [Migration Guide](#) to progress through your import. The guide links to the technical documentation as needed.

With the release of Azure DevOps Server 2019 the TFS Database Import Service was rebranded to Migrate to Azure DevOps. This includes Tfsmigrator becoming the data migration tool or migrator for short. This service still works exactly the same as the old Import Service. If you're on an older version of on-premises with TFS as the branding you can still use this feature to migrate to Azure DevOps as long as you upgrade to one of the supported versions.

## Process validation types

During validation, the data migration tool determines the target process model for each project. It automatically assigns one of the following two process models to each project in the collection:

- **Inherited process model:** If the project was created with the Agile, Scrum, or CMMI process template, and was never customized.
- **Hosted XML process model:** If the project process appears to have been customized. A customized process contains custom fields, work item types, or other types of customizations.

When the Hosted XML process is the targeted process model, the data migration tool validates if the customizations can be migrated. The data migration tool generates two files during the validation:

- **DataMigrationTool.log:** Contains the set of process validation errors found in the collection. Fix all process errors found to proceed with your migration.
- **TryMatchOobProcesses.log:** Lists for each project the target process model - Inheritance or Hosted XML. For projects that are set to target the Hosted XML process model, it explains why they are considered to be customized. You don't have to fix these errors, but they give you guidance what to do in case you want to migrate to the Inheritance process model. Note that once a collection is imported, you can migrate a project to an Inheritance process model.

Most customers have a mix of projects within a collection. Some projects use a default process template and others use custom process templates. The data migration tool checks and validates each project accordingly. It is very possible that you'll have a mix of projects, some mapped to an Inherited process and others to a Hosted XML process.

We recommend that for any project that has not been customized, that you review the `TryMatchOobProcesses.log` to determine if there are any errors. If so, make the adjustments accordingly so

that the project can be mapped to an Inherited process upon data import.

## Update to a system process

If you started with an older version of Azure DevOps Server, odds are your projects are still using an older process template. If those projects have not been updated using the [Configure Features Wizard](#) then the data migration tool will find process errors. In some rare cases, if your process is very old, even the Configure Features Wizard won't be able to resolve the errors.

Here are some examples of error messages you may receive:

```
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402571: Required element PortfolioBacklog is missing from Process Configuration.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402571: Required element BugWorkItems is missing from Process Configuration.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402571: Required element FeedbackRequestWorkItems is missing from Process Configuration.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402571: Required element FeedbackResponseWorkItems is missing from Process Configuration.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402574:  
ProcessConfiguration doesn't specify required TypeField Team.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402574:  
ProcessConfiguration doesn't specify required TypeField RemainingWork.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402574:  
ProcessConfiguration doesn't specify required TypeField Order.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402574:  
ProcessConfiguration doesn't specify required TypeField Effort.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402574:  
ProcessConfiguration doesn't specify required TypeField Activity.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402574:  
ProcessConfiguration doesn't specify required TypeField ApplicationStartInformation.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402574:  
ProcessConfiguration doesn't specify required TypeField ApplicationLaunchInstructions.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF402574:  
ProcessConfiguration doesn't specify required TypeField ApplicationType.  
Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml:: TF400572: The Project Process  
Settings must be configured for this feature to be used.
```

If you have never customized your project (added fields, work item types, etc.), then fixing these errors is actually pretty simple. If you have customized your process, then this approach won't work. You'll need to manually change the process templates so that your customizations don't get overwritten.

First, make sure you know what process your project started as. Is it Scrum, Agile or CMMI? In this example, let us assume Agile. Next, go to the [Process Customization Scripts](#) provided on GitHub and download the repo. In this instance, we are going to focus on contents in the `Import` folder.

Use the `ConformProject.ps1` script to conform a project of your choosing to the Agile system process. This will update the entire project to be Agile.

```
.\ConformProject.ps1 "<collection url>" "<project name>" "c:\process-customization-scripts\import\agile"
```

Make sure you do this for each and every project.

## Resolve process errors

Are your process templates customized? Are you using an older outdated process template? If so, you'll most likely have process validation errors. The data migration tool does an exhaustive check against your process templates. It checks to make sure that it is valid for Azure DevOps Services. Odds are that you'll need to make

some adjustments and apply them to your collection.

#### NOTE

If you are using an OOB Agile, Scrum, or CMMI process, you probably won't see any errors in the **DataMigrationTool.log**. Instead, check the **TryMatchOobProcesses.log** for errors. If you are error free, then your project will map to an OOB process.

There are several customizations that won't work in Azure DevOps Services. Make sure you review the [list of customizations](#) that are supported.

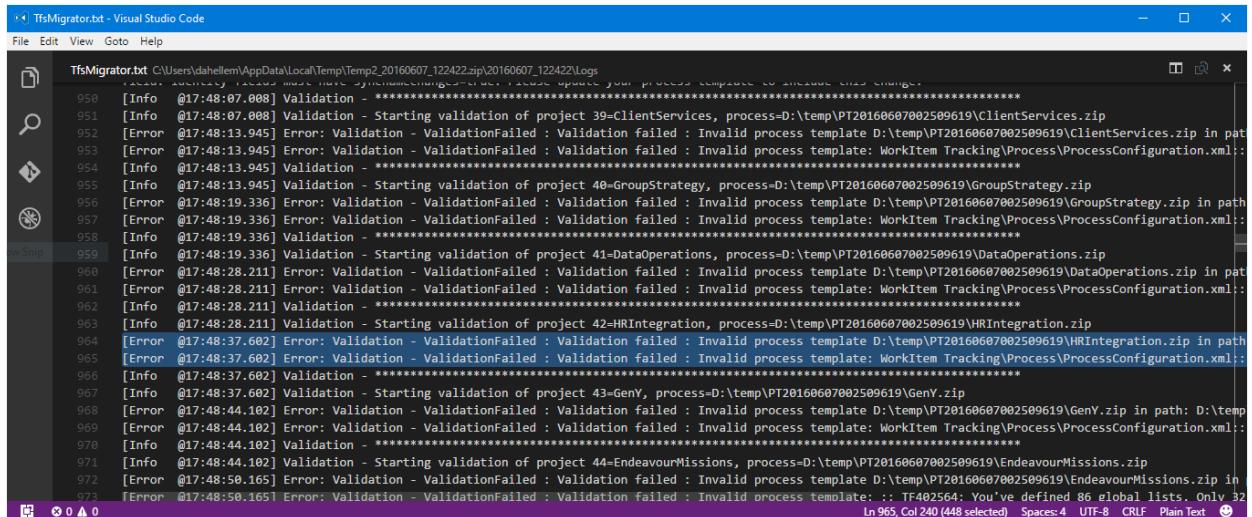
If you have projects that are using an older process template, the data migration tool will find several errors. This is because your process templates hasn't been updated to match the most recent process templates. To start, try running the [Configure Features Wizard](#) for each project. This will attempt to update your process templates with the most recent features. Doing so should drastically reduce the error count.

Finally, make sure you have `witadmin` on the machine that you intend to use to fix the process errors. This can be your local desktop. The `witadmin` command line tool is used in the automated scripts and is required whenever making changes to the process templates.

## Step 1 - Review errors

**DataMigrationTool.log** file will be generated and contains the list of errors that the validation process found. To view the logs, open **DataMigrationTool.log** file. Search for the string "Validation - Starting validation of project 1". Each project is validated. Scan through all the projects and search for any lines that contain a prefix of [Error

....



```
TfsMigrator.txt C:\Users\dahelleml\AppData\Local\Temp\Temp2_20160607_122422.zip\20160607_122422\Logs
File Edit View Goto Help

950 [Info @17:48:07.008] Validation - *****
951 [Info @17:48:07.008] Validation - Starting validation of project 39=ClientServices, process=D:\temp\PT20160607002509619\ClientServices.zip
952 [Error @17:48:13.945] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\ClientServices.zip in path
953 [Error @17:48:13.945] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::
954 [Info @17:48:13.945] Validation - *****
955 [Info @17:48:13.945] Validation - Starting validation of project 40=GroupStrategy, process=D:\temp\PT20160607002509619\GroupStrategy.zip
956 [Error @17:48:19.336] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\GroupStrategy.zip in path
957 [Error @17:48:19.336] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::
958 [Info @17:48:19.336] Validation - *****
959 [Info @17:48:19.336] Validation - Starting validation of project 41=DataOperations, process=D:\temp\PT20160607002509619\DataOperations.zip
960 [Error @17:48:28.211] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\DataOperations.zip in path
961 [Error @17:48:28.211] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::
962 [Info @17:48:28.211] Validation - *****
963 [Info @17:48:28.211] Validation - Starting validation of project 42=HRIntegration, process=D:\temp\PT20160607002509619\HRIntegration.zip
964 [Error @17:48:37.602] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\HRIntegration.zip in path
965 [Error @17:48:37.602] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::
966 [Info @17:48:37.602] Validation - *****
967 [Info @17:48:37.602] Validation - Starting validation of project 43=GenY, process=D:\temp\PT20160607002509619\GenY.zip
968 [Error @17:48:44.102] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\GenY.zip in path: D:\temp
969 [Error @17:48:44.102] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::
970 [Info @17:48:44.102] Validation - *****
971 [Info @17:48:44.102] Validation - Starting validation of project 44=EndeavourMissions, process=D:\temp\PT20160607002509619\EndeavourMissions.zip
972 [Error @17:48:58.165] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\EndeavourMissions.zip in path: :: TF402564: You've defined 86 global lists. Only 32
973 [Error @17:48:58.165] Error: Validation - ValidationFailed : Validation failed : Invalid process template :: TF402564: You've defined 86 global lists. Only 32
Ln 965, Col 240 (448 selected) Spaces:4 UTF-8 CRLF Plain Text
```

For a list of validation errors, see [Resolve validation errors for process import](#). For each validation error, we have provided the error number, description, and the method to resolve.

## Step 2 - Fix errors

Once you've determined which projects have errors and the error details, fix the errors. Fixing the errors requires that you change the XML syntax and apply the changes back to the project.

#### NOTE

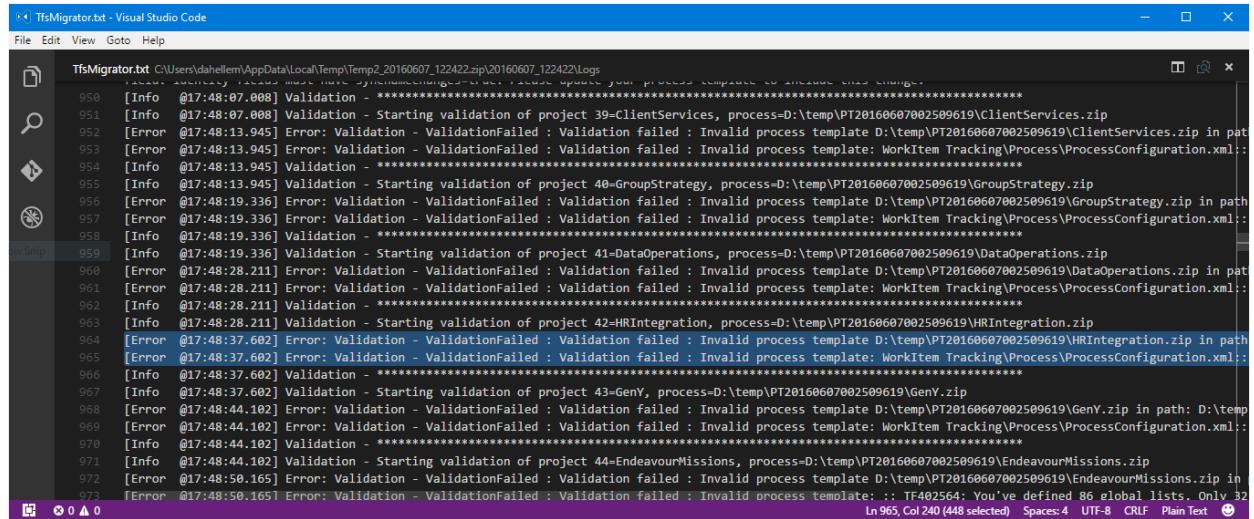
We recommend you don't use TFS Power Tools to do this work. Instead, we highly recommended that you modify the XML manually.

To get the process template from the project add the `/SaveProcesses` parameter when running the data migration tool command.

```
Migrator validate /collection:{collection URL} /SaveProcesses
```

This command will extract the XML from the project and place it into the same folder as the logs. Extract the zip files to your local machine so that you can edit the files.

Now, fix the XML. Use the logs from the `DataMigrationTool.log` file to determine the errors for each project.



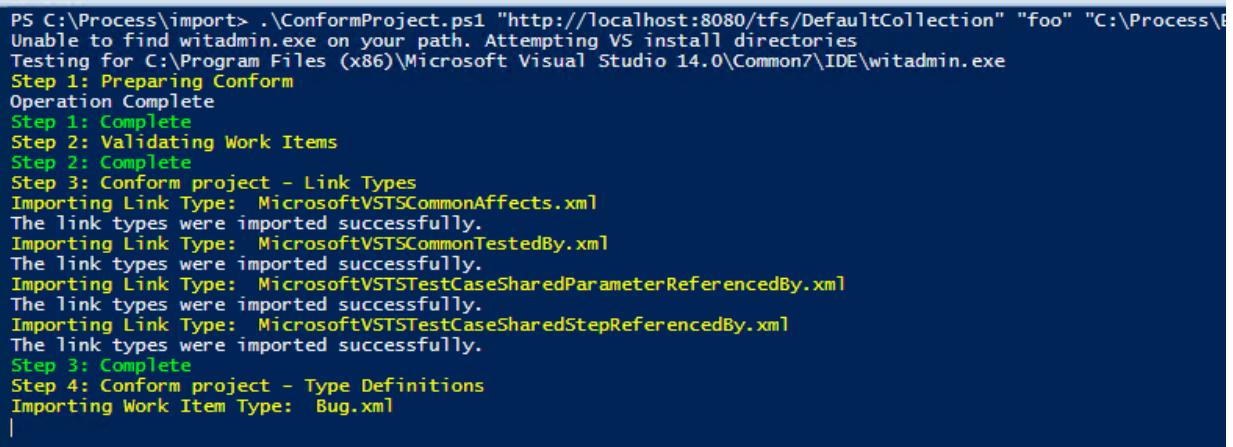
```
TfsMigrator.txt C:\Users\dahellem\AppData\Local\Temp\Temp2_20160607_122422.zip\b20160607_122422\Logs
950 [Info @17:48:07.008] Validation - *****
951 [Info @17:48:07.008] Validation - Starting validation of project 39-ClientServices, process=D:\temp\PT20160607002509619\ClientServices.zip
952 [Error @17:48:13.945] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\ClientServices.zip in path: D:\temp\PT20160607002509619\ClientServices.zip
953 [Error @17:48:13.945] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::10
954 [Info @17:48:13.945] Validation - *****
955 [Info @17:48:13.945] Validation - Starting validation of project 40-GroupStrategy, process=D:\temp\PT20160607002509619\GroupStrategy.zip
956 [Error @17:48:19.336] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\GroupStrategy.zip in path: D:\temp\PT20160607002509619\GroupStrategy.zip
957 [Error @17:48:19.336] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::10
958 [Info @17:48:19.336] Validation - *****
959 [Info @17:48:19.336] Validation - Starting validation of project 41-DataOperations, process=D:\temp\PT20160607002509619\DataOperations.zip
960 [Error @17:48:28.211] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\DataOperations.zip in path: D:\temp\PT20160607002509619\DataOperations.zip
961 [Error @17:48:28.211] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::10
962 [Info @17:48:28.211] Validation - *****
963 [Info @17:48:28.211] Validation - Starting validation of project 42-HRIntegration, process=D:\temp\PT20160607002509619\HRIntegration.zip
964 [Error @17:48:37.602] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\HRIntegration.zip in path: D:\temp\PT20160607002509619\HRIntegration.zip
965 [Error @17:48:37.602] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::10
966 [Info @17:48:37.602] Validation - *****
967 [Info @17:48:37.602] Validation - Starting validation of project 43-GenY, process=D:\temp\PT20160607002509619\GenY.zip
968 [Error @17:48:44.102] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\GenY.zip in path: D:\temp\PT20160607002509619\GenY.zip
969 [Error @17:48:44.102] Error: Validation - ValidationFailed : Validation failed : Invalid process template: WorkItem Tracking\Process\ProcessConfiguration.xml::10
970 [Info @17:48:44.102] Validation - *****
971 [Info @17:48:44.102] Validation - Starting validation of project 44-EndeavourMissions, process=D:\temp\PT20160607002509619\EndeavourMissions.zip
972 [Error @17:48:50.165] Error: Validation - ValidationFailed : Validation failed : Invalid process template D:\temp\PT20160607002509619\EndeavourMissions.zip in path: D:\temp\PT20160607002509619\EndeavourMissions.zip
973 [Error @17:48:50.165] Error: Validation - ValidationFailed : Validation failed : Invalid process template: :: TF402564: You've defined 86 global lists. Only 32
In 965, Col 240 (448 selected) Spaces:4 UTF-8 CRLF Plain Text ☺
```

Some errors will require you to do use a `witadmin changefield` command. Changing a field name is the most common example. To save yourself some time, we recommend you run the `witadmin changefield` command and then re-run the data migration tool. Doing this will re-export the XML with the corrected names. Otherwise, you'll need to manually fix the fields in the XML syntax as well.

Once you make a fix, apply the changes back to the Azure DevOps Server. To do this, depending on the changes you made, you'll need to run one or more `witadmin` commands. To make this easier for you, we created a PowerShell script to automate the process. The script contains all of the `witadmin` commands needed to conform the entire process.

You can get the scripts at [Process Customization Scripts](#). Use the `import/ConformProject.ps1` script.

```
.\conformproject.ps1 "<collection url>" "<project name>" "<process template folder>"
```



```
PS C:\Process\import> .\ConformProject.ps1 "http://localhost:8080/tfs/DefaultCollection" "foo" "C:\Process\B
Unable to find witadmin.exe on your path. Attempting VS install directories
Testing for C:\Program Files (x86)\Microsoft Visual Studio 14.0\Common7\IDE\witadmin.exe
Step 1: Preparing Conform
Operation Complete
Step 1: Complete
Step 2: Validating Work Items
Step 2: Complete
Step 3: Conform project - Link Types
Importing Link Type: MicrosoftVSTSCommonAffects.xml
The link types were imported successfully.
Importing Link Type: MicrosoftVSTSCommonTestedBy.xml
The link types were imported successfully.
Importing Link Type: MicrosoftVSTSTestCaseSharedParameterReferencedBy.xml
The link types were imported successfully.
Importing Link Type: MicrosoftVSTSTestCaseSharedStepReferencedBy.xml
The link types were imported successfully.
Step 3: Complete
Step 4: Conform project - Type Definitions
Importing Work Item Type: Bug.xml
|
```

When the script has completed, re-run the data migration tool to validate the collection. Follow steps 1 through 3 until the data migration tool generates no more validation errors.

**TIP**

If you are new to XML and `witadmin`, we suggest you make one fix at a time and then conform. Continue this loop until all errors are resolved.

## Common validation errors

**VS402841: Field X in work item type Bug has syncnamechanges=false but has rules making it an identity field. Identity fields must have syncnamechanges=true. Please update your process template to include this change.**

In Azure DevOps Services we added a rule so that every identity field must have the `syncnamechanges=true` attribute. In Azure DevOps Server that rule does not apply. Therefore, the data migration tool will identify this as an issue. Don't worry, making this change on Azure DevOps Server on-prem will not cause any harm.

Run the `witadmin changefield` command. Syntax for the command looks similar to the following:

```
witadmin changefield /collection:http://AdventureWorksServer:8080/tfs/DefaultCollection /n:fieldname  
/syncnamechanges:true
```

For more information on the `witadmin changefield` command see [Manage work item fields](#).

**TF402556: For field System.IterationId to be well defined, you must name it Iteration ID and set its type to Integer.**

This error is typical for old process templates. Try running the [Configure Features Wizard](#) on each project.

Alternatively you can run the follow `witadmin` command:

```
witadmin changefield /collection:http://AdventureWorksServer:8080/tfs/DefaultCollection /n:fieldname  
/name:newname
```

**TF402571: Required element BugWorkItems is missing from Process Configuration.**

This error typically occurs when a process hasn't been updated in a while. Try running the [configure features wizard](#) on each project to resolve.

**TF402564: You've defined XX global lists. Only 64 are allowed.**

By default, Azure DevOps Services will support 64 global lists. You'll typically run across this error if you have a large amount of build pipelines. The global list named Builds - `TeamProjectName` gets created for each new build pipeline. You'll need to remove the outdated global lists.

## Related articles

- [Migration and process model FAQs](#)
- [witadmin : Customize and manage objects for tracking work](#)
- [Differences between Azure DevOps Services and Azure DevOps Server process template customizations](#)
- [Configure features after Azure DevOps Server upgrade](#)
- [Resolve validation errors](#)
- [Define global lists in Azure DevOps Server](#)
- [Process customization PowerShell scripts](#)

# Post import

4/29/2021 • 5 minutes to read • [Edit Online](#)

## Azure DevOps Services | Azure DevOps Server | TFS

An organization is ready for use once an import has completed successfully. However, there are common tasks that you should perform before opening the organization up to all of your users. Below is a list of the most common after import tasks that should be completed. Tasks are listed in recommended order of completion.

### NOTE

It's recommended that you use the [Migration Guide](#) to progress through your import. The guide links to the technical documentation as needed.

With the release of Azure DevOps Server 2019 the TFS Database Import Service has been rebranded to become data migration tool for Azure DevOps. This includes TfsMigrator becoming the data migration tool or migrator for short. This service still works exactly the same as the old Import Service. If you're on an older version of on-premises with TFS as the branding you can still use this feature to migrate to Azure DevOps as long as you upgrade to one of the supported versions.

## Immediately after import

Immediately after the organization becomes available you will want to take a small team and perform spot checks on the organization. It's recommended that this team consists of the project collection administrators. This shouldn't be an in-depth check, but rather making sure that major pieces from your collection were brought over. Did your source code get imported? Are you seeing your build history? Are all of our area paths still present? It's best to confirm these artifacts are present before opening the organization to the entirety of your user base.

After spot checking the organization you will want to consider if you want to rename it. [Renaming an organization](#) is a simple operation, but it has [large impacts](#) on users currently using the organization. Some examples being Team Explore connections breaking or bookmarks no longer working. Getting a rename out of the way while it's just a small group of users using the organization allows the rest of the users to come in and configure their connections once.

## Set up billing

To pay for users or services in Azure DevOps Services, like hosted build and deployment agents, you need to [set up billing](#) for your organization. If you import more than one collection, you should ensure all your organizations are set up for billing with the same Azure subscription, and that your subscription is enabled for [multi-organization billing](#). You can then assign as many Basic users as you need free of charge during the calendar month in which you run the import.

## Manage users and access

Your organization includes 5 free users with [Basic](#) access. Basic includes features like Git and Team Foundation version control, tools for Agile planning and Java teams, and more. Also, you can add [Visual Studio subscribers](#) for free—they get basic features plus additional features—based on their subscription level. Also, you can add [Stakeholder](#) for free, which allows them to have partial access to Agile tools, create work items, and view backlogs and boards.

As Visual Studio subscribers log in to the organization, they are automatically detected. For all other users, you need to [assign paid access](#). Keep in mind, if you automate access using [group rules](#), the rules only apply to existing users if you [remove direct assignments](#), which were applied to users during import.

**Behavior change**—Starting between Monday, November 11th and Wednesday, November 13th, the default access behavior for imports will change. Previously, all imports tried to give users an equivalent access level post import. This means that users that had *Basic* received Basic access, and other users started with *Stakeholder* access. Once this change happens, all users will start out with free *Stakeholder* access. **You will continue to be able to assign Basic access to any users who need it at no cost, until the end of the calendar month during which your import is run.** If you have any questions or concerns about this change, feel free to [contact us](#).

## Builds

Next, you will want to configure your build agents. As part of the migration, all of your build pipelines have been brought over, but agents and pools need to be reconfigured against the new organization. Azure DevOps Services offers the ability to use a Microsoft-hosted pool of build agents that you can use, or you can connect your self-hosted build agent(s). It's important to note that only one self-hosted build agent is included for free. After that there is a [fee](#) for having additional self-hosted build agents. To pay for Microsoft-hosted and self-hosted build agents you will need to link a subscription to your organization. See the following resources for details on performing this task:

- [Build Agents](#)

If you plan on using your existing on-premises private build agents, there is one more recommended step that needs to be taken after registering them to your new organization. Clearing their cache will ensure that you don't encounter any build issues related to older TFVC or Git pointers to your on-premises collection. See [refreshing caches on client computers](#) for details on how to accomplish this task.

## Release management

If you used Release Management in Azure DevOps Server then your release pipelines and history data will be included with your import. However, like builds, [agents](#) and pools need to be reconfigured against the new organization.

## Azure Artifacts

If you used Azure Artifacts in your collection, then you will need to install the Azure Artifacts [extension](#) in your organization post import to view your Azure Artifacts data.

## Azure Boards

If you have an existing GitHub Enterprise Server connection associated with your Azure DevOps Server, it will not work as expected. Work item mentions within GitHub may be delayed or never show up in Azure DevOps Services. This problem occurs because the callback url associated with GitHub is no longer valid.

To resolve the problem, consider the following:

- **Remove and re-create the connection:** Remove and re-create the connection to the GitHub Enterprise Server repository. Follow the sequence of steps provided in [Connect from Azure Boards](#) documentation.
- **Fix the webhook url:** Go to GitHub's repository settings page and edit the webhook url to point out to the migrated Azure DevOps Services organization url:  
`https://dev.azure.com/{OrganizationName}/_apis/work/events?api-version=5.2-preview`

## Notify your teams

After getting your builds running and license subscription configured, it's recommended that the organization be opened up to all users for validation. This is when individual users can ensure that all of the content is in place, they have the right access level, and that they can pull code. Be sure to point users to our [documentation](#) on connecting to Azure DevOps Services from all of our supported IDEs and Team Explorer.

Users of TFVC with local workspaces will need to remap their workspaces against the new organization and Git users will have to reconfigure their remotes to be able to pull code.

If anything is reported as missing from the migrated organization, please reach out to [AzureDevOpsImport@microsoft.com](mailto:AzureDevOpsImport@microsoft.com). For other functional issues, please reach out to [customer support](#).

# Troubleshoot import and migration errors

5/18/2021 • 19 minutes to read • [Edit Online](#)

## Azure DevOps Services | Azure DevOps Server | TFS

The data migration tool flags errors that you need to correct prior to performing a migration to Azure DevOps Services. This article describes the most common warnings and errors that you may receive when preparing to migrate. After correcting each error, run the **migrator validate** command again to verify resolution of all errors.

### NOTE

We recommended that you use the [Migration guide](#) to progress through your import. The guide links to the technical documentation as needed.

With the release of Azure DevOps Server 2019, the TFS Database Import Service was re-branded to become the data migration tool for Azure DevOps. The data migration tool, **TfsMigrator** has been renamed **migrator** for short. The service still works exactly the same as the previous import service. If you're on an older version of on-premises with TFS as the branding, you can still use **migrator** to migrate to Azure DevOps as long as you upgrade to one of the supported versions. For details, see [Migrate data from Azure DevOps Server to Azure DevOps Services](#).

## Resolve size warnings

Extra-large collections may generate one of the following messages after running the data migration tool. If you receive any of these warnings or errors, we recommend that you try to [reduce your database's size](#).

### Database size above recommended size

The following warning means you need to use the SQL Azure VM method to complete your import. Once a database reaches a certain size, it becomes faster to setup a SQL Azure VM to complete the import to Azure DevOps Services. To setup the VM and complete your import, follow the instructions linked from the warning message.

The database is currently {Database Size}GBs. This is above the recommended size of {DACPAC Size Limit}GBs to use the DACPAC import method. Please see the following page to learn how to import using a SQL Azure VM: <https://aka.ms/AzureDevOpsImportLargeCollection>

This warning DOES NOT mean that your collection is too large for import.

### Table size above recommended size

Similar to the previous warning, the following warning means you must use the SQL Azure VM method to complete the import. Follow the instructions linked from the warning message to setup the VM and complete your import.

The largest table size is currently {Table size}GBs. This is above the recommended size of {Size limit}GBs to use the DACPAC import method. Please see the following page to learn how to import using a SQL Azure VM: <https://aka.ms/AzureDevOpsImportLargeCollection>

This warning DOES NOT mean that your collection is too large for import.

### Database metadata size above recommended size

The following warning means that your database is approaching the limit for total metadata size. Metadata size refers to the size of your database without including files, code, and other binary data. We recommend that you [reduce the size](#) of your database before import. Reducing the size provides the additional benefit of speeding up your import.

The database metadata size is currently {Metadata Size}GBs. This is above the recommended size of {Warning Size}GBs. It's recommended that you consider cleaning up older data as described in [Cleaning up old data] (/azure/devops/server/upgrade/clean-up-data).

The warning **DOES NOT** mean that your collection is too large for import, rather its metadata size is larger than the vast majority of other databases.

#### Database metadata size above maximum supported size

Unlike the previous warnings, the following error **WILL** block you from moving forward with your migration.

It indicates that the volume of metadata in your collection is too large. To proceed with the import, you need to [reduce](#) the size below the indicated limit.

The database metadata size is currently {Metadata Size}GBs. This is above the maximum supported size of {Metadata Limit}GBs.

## Resolve collation warnings

Collation warnings refer to your collection database's collation. Collations control the way string values are sorted and compared. Collections that aren't using either SQL\_Latin1\_General\_CI\_AS or Latin1\_General\_CI\_AS will generally receive one of the **warning** messages.

#### No native support

Receiving the following warning means that you need to consider collation implications before performing the import.

The collection database's collation '{collation}' is not natively supported in Azure DevOps Services. Importing your collection will result in your collation being converted to one of the supported Azure DevOps Services collations. See more details at <https://aka.ms/AzureDevOpsImportCollations>

This warning **DOES NOT** mean that you can't import your collection.

This warning requires you to acknowledge acceptance of the warning. Accepting the warning allows the data migration tool to continue import preparations.

When you import a non-supported collation into Azure DevOps Services, the collation is transformed to a supported collation. While this transform generally works without issue, unexpected results post import or import failures could occur.

For instance, customers may notice different ordering for strings containing non-English characters. Non-English characters like 'é' may become equivalent to the English 'e' after import. It's important that you complete and verify a dry run import when importing a collection with a non-supported collation.

#### No native support, no internet connection

If the data migration tool can't connect to the internet, it can't validate conversion of your collation. It's only a warning, so you can continue with your migration process. However, when you run the **prepare** command, an internet connection is required and collation conversion is validated at that time.

The collections database's collation '{collation}' is not natively supported in Azure DevOps Services. It could not be validated that the collation can be converted during import to a supported Azure DevOps Services collation, as there was no internet connection. Please run the command again from a machine with an internet connection. See more details at <https://aka.ms/AzureDevOpsImportCollations>

## Unsupported database collation

Generally you can convert a non-supported collation to a supported collation at import time. However, some collations can't be converted. If your collection uses one of these collations, you'll receive the following **error** message.

The collection database's collation '{collation}' is not supported for import to Azure DevOps Services. It will need to be changed to a supported collation before it can be imported. See more details at <https://aka.ms/AzureDevOpsImportCollations>

In order to continue, you need to [change your collection's collation](#) to one of the supported collations on Azure DevOps Services.

## Resolve identity errors

Identity errors aren't common when validating a collection, but when they do occur you need to fix them prior to migration to avoid undesired results. Generally, identity problems stem from valid operations on previous versions of TFS that are no longer valid on your current Azure DevOps Server version. For example, while it was once allowed for some users to be members of a built-in valid users group, it isn't in the more recent versions.

The following sections provide guidance for resolving the most common identity errors.

### ISVError: 100014

This error indicates that a permission is missing from a system security group. For example, every collection that you create has Project Collection Valid Users and Project Collection Administrators groups. The system creates them by default. These groups don't support editing of their permissions.

This error indicates that one or more groups is missing a permission that it's expected to have. To resolve this error, use the [TFSSecurity.exe](#) command to apply the expected permissions onto the flagged system groups. Your first step is to identify which [TFSSecurity](#) command(s) you need to run.

#### Project Collection Valid Users error message

Examine the error message(s) the data migration tool highlighted. If the flagged group ends with "0-0-0-0-3", such as in the example below, you need to fix a missing permission for the **Project Collection Valid Users** group.

Run the following command, replace the scope with the one from the error message and specify your collection URL.

```
TFSSecurity.exe /a+ Identity "{scope}\\" Read sid:{Group SID} ALLOW /collection:{collectionUrl}
```

You determine the scope and group SID from the error message.

```
ISVError:100014 Missing permission for group:Microsoft.TeamFoundation.Identity;S-1-9-XXXXXXXXXX-XXXXXXX-XXXXXXX-XXXXXXX-XXXXXXX-0-0-0-3 for scope:397c326b-b97c-4510-8271-75aac13de7a9. Expected:1 and Actual:0
```

The final command appears similar to the following entry:

```
TFSSecurity.exe /a+ Identity "397c326b-b97c-4510-8271-75aac13de7a9\\\" Read sid:S-1-9-XXXXXXXXXX-XXXXXXXXXX-  
XXXXXXXXXX-XXXXXXXXXX-0-0-0-0-3 ALLOW /collection:https://localhost:8080/defaultcollection
```

#### Project Collection Administrators error message

Carefully examine the error message(s) the data migration tool highlighted. If the flagged group that ends with "0-0-0-0-1", such as in the example below, then you will need to fix a missing permission for the **Project Collection Administrators** group. Run the following commands against **TFSSecurity.exe**, replace the scope with the one from the error message and specify your collection.

```
TFSSecurity.exe /a+ Identity "{scope}\" Read sid:{Group SID} ALLOW /collection:{collectionUrl}  
TFSSecurity.exe /a+ Identity "{scope}\" Write sid:{Group SID} ALLOW /collection:{collectionUrl}  
TFSSecurity.exe /a+ Identity "{scope}\" Delete sid:{Group SID} ALLOW /collection:{collectionUrl}  
TFSSecurity.exe /a+ Identity "{scope}\" ManageMembership sid:{Group SID} ALLOW /collection:{collectionUrl}
```

In the following example, take the scope and group SID from the error message and add them to the preceding command.

```
ISVError:100014 Missing permission for group:Microsoft.TeamFoundation.Identity;S-1-9-XXXXXXXXXX-XXXXXXXXXX-  
XXXXXXXXXX-XXXXXXXXXX-0-0-0-0-1 for scope:0c7c2216-fa4b-4107-a203-82b324a147ef. Expected:15 and Actual:0
```

The final command appears similar to the following entry:

```
TFSSecurity.exe /a+ Identity "0c7c2216-fa4b-4107-a203-82b324a147ef\\\" Read sid:S-1-9-XXXXXXXXXX-XXXXXXXXXX-  
XXXXXXXXXX-XXXXXXXXXX-0-0-0-0-1 ALLOW /collection:https://localhost:8080/defaultcollection  
TFSSecurity.exe /a+ Identity "0c7c2216-fa4b-4107-a203-82b324a147ef\\\" Write sid:S-1-9-XXXXXXXXXX-XXXXXXXXXX-  
XXXXXXXXXX-XXXXXXXXXX-0-0-0-0-1 ALLOW /collection:https://localhost:8080/defaultcollection  
TFSSecurity.exe /a+ Identity "0c7c2216-fa4b-4107-a203-82b324a147ef\\\" Delete sid:S-1-9-XXXXXXXXXX-  
XXXXXXXXXX-XXXXXXXXXX-XXXXXXXXXX-0-0-0-0-1 ALLOW /collection:https://localhost:8080/defaultcollection  
TFSSecurity.exe /a+ Identity "0c7c2216-fa4b-4107-a203-82b324a147ef\\\" ManageMembership sid:S-1-9-XXXXXXXXXX-  
XXXXXXXXXX-XXXXXXXXXX-XXXXXXXXXX-0-0-0-0-1 ALLOW /collection:https://localhost:8080/defaultcollection
```

When you need to correct multiple errors, we recommend that you create a batch file to automate execution of the commands. Once you've executed the commands, you need to rerun the data migration **validate** tool to verify resolution. If some errors still persist, contact [Azure DevOps Services customer support](#).

#### ISVError: 300005

ISVError: 300005 indicates that a non-group identity is a member of an everyone group, more commonly known as the Valid Users groups. Valid Users groups are default groups defined for all projects and collections. They're non-editable groups that only contain other Azure DevOps security groups as members. This error indicates that an AD group or user identity has a direct membership in a Valid Users group.

#### IMPORTANT

Ensure that you have a backup of your collection and configuration databases before running the following commands to resolve the error.

Since you can't directly edit Valid Users groups, you need to correct the invalid membership by running a SQL statement against the configuration database to remove the offending identity. Carefully examine the error

messages the data migration tool highlights. Copy down the GroupSid, MemberId, and Scopeld as you'll need to place these values into the following command.

```
DECLARE @p6 dbo.typ_GroupMembershipTable  
  
INSERT into @p6 values('{GroupSid}','Microsoft.TeamFoundation.Identity','{MemberId}',0)  
  
EXEC prc_UpdateGroupMembership  
@partitionId=1,@scopeId='{ScopeId}',@idempotent=1,@incremental=1,@insertInactiveUpdates=0,@updates=@p6,@even  
tAuthor='9EE20697-5343-43FC-8FC5-3D5D455D21C5',@updateGroupAudit=0
```

Below is an example ISVError: 300005 message from the data migration tool.

```
ISVError:300005 Unexpected non group identity was found to have direct membership to everyone group.  
GroupSid:S-1-9-1551374245-3746625149-2333054533-2458719197-2313548623-0-0-0-3, MemberId:76050ddf-4fd8-  
48c4-a1ff-859e44364519, ScopeId:7df650df-0f8b-4596-928d-13dd89e5f34f
```

Copy the GroupSid, MemberId, and Scopeld into the SQL command.

```
DECLARE @p6 dbo.typ_GroupMembershipTable  
  
INSERT into @p6 values('S-1-9-1551374245-3746625149-2333054533-2458719197-2313548623-0-0-0-  
3','Microsoft.TeamFoundation.Identity','76050ddf-4fd8-48c4-a1ff-859e44364519',0)  
  
EXEC prc_UpdateGroupMembership @partitionId=1,@scopeId='7df650df-0f8b-4596-928d-  
13dd89e5f34f',@idempotent=1,@incremental=1,@insertInactiveUpdates=0,@updates=@p6,@eventAuthor='9EE20697-  
5343-43FC-8FC5-3D5D455D21C5'
```

Run the completed command against the Azure DevOps Server configuration database. You'll need to repeat this command for each ISVError: 300005 instance reported. You can batch errors with the same scope ID into a single command. Once you've executed the commands, rerun the data migration tool validate again to ensure that the errors have been corrected. If the errors still persist, contact [Azure DevOps Services customer support](#).

#### IMPORTANT

To address these errors, the collection must be attached.

If you receive a -1 result when you run the command, ensure that your collection database that produced the error is attached to your Azure DevOps Server instance and that you're running the command on the configuration database.

#### Azure Active Directory timeout exception

On rare occasions, you may receive an Azure Active Directory (Azure AD) timeout error when running the data migration tool prepare command.

```
Exception Message: Request failed (type AadGraphTimeoutException)
```

This error means that the requests to Azure AD to find the matching Azure AD identities for users in your collection timed out. Generally, you can resolve this error by waiting to run the **prepare** command at a less busy time of the day, such as after regular business hours.

In the event that the error continues, you should undertake a few troubleshooting steps. First, you will want to test your connection to Azure AD from the machine running the **prepare** command. Execute the following steps to retrieve information on a user in your Azure AD.

Open PowerShell in elevated mode and replace 'someone@somecompany.com' in the following command with

your Azure AD user identity.

```
//Install the AzureAD PowerShell module - ensuring to select Yes to All  
Install-Module AzureAD  
  
// Install the MSOnline PowerShell module - ensuring to select Yes to All  
Install-Module MSOnline  
  
// Connect to AAD and use your AAD credentials (someone@somecompany.com) to login when the pop-up appears  
Connect-MsolService  
  
// Try to retrieve information on a user from your AAD  
Get-MsolUser -UserPrincipalName someone@somecompany.com
```

If any of the above steps fail or you're unable to look up a user's identity, a connection issue may exist between the machine running the **prepare** command and Azure AD. Run a network trace while executing the **prepare** command to ensure that nothing within your network is interfering with calls reaching Azure AD. If you've confirmed that the problem isn't with your network, contact Azure support for assistance with troubleshooting. If you're able to retrieve user information, open your log file from the **prepare** attempt and look for a line similar to the following entry.

```
Number of active users is {Number of Users}.
```

If this number is in the high five-digits or even six-digits ranges, it may indicate that the volume of identities being mapped requires more time than the timeout limit provides. Inspect your collection for inclusions of large AD groups such as an 'everyone' group. If possible, remove these groups and try again. If you still can't resolve this error, contact [Azure DevOps Services customer support](#).

## Resolve process errors

See the separate [Process Templates](#) page for details on resolving common process errors.

## Resolve field validation errors

### VS403442

Field name conflicts sometimes occur between your local collection and an Azure DevOps Services system field.

```
In order to migrate successfully, you must rename field *{TFSfieldReferenceName}*.
Given name *{TFSfieldName}* is reserved for field *{VSTSfieldReferenceName}*.
```

To resolve this error, change the name of your collection field. Use the **witadmin changefield** command from [witadmin](#).

```
witadmin changefield /collection:http://AdventureWorksServer:8080/DefaultCollection /n:TFSfieldReferenceName
/name:newFieldName
```

### VS403443

The following error indicates a field name conflict exists between your local collection and a specific Azure DevOps Services field.

```
In order to migrate successfully, you must rename field *{TFSfieldReferenceName}* to *{VSTSfieldName}*. Given name for *{TFSfieldReferenceName}* is *{TFSfieldName}*
```

To resolve this error, use the **witadmin changefield** command. For details, see [witadmin](#).

```
witadmin changefield /collection:http://AdventureWorksServer:8080/DefaultCollection /n:TFSfieldReferenceName  
/name:VSTSfieldName
```

#### VS403444

The following error indicates a field type conflict exists between your local collection and Azure DevOps Services.

Using [witadmin](#), you can change the data type only for HTML or PlainText fields.

```
In order to migrate successfully, you must set type of field *{TFSfieldReferenceName}* to *{Type}*. Given  
type for *{TFSfieldReferenceName}* is *{collectionType}*.
```

If your field type is HTML or PlainText, then you can change its type to the required type.

```
witadmin changefield /collection:http://AdventureWorksServer:8080/DefaultCollection /n:TFSfieldReferenceName  
/type:PlainText | HTML
```

#### NOTE

If your field type is something different than HTML or PlainText and field data isn't important or the field isn't used in any project, then we recommend you delete the field.

```
witadmin deletefield /collection:http://AdventureWorksServer:8080/DefaultCollection /n:TFSfieldReferenceName
```

#### IMPORTANT

Deleting a field results in a loss of field data across the collection.

## Resolve import errors

Hit a failure when running your import? Failures in the import space fall into one of two categories.

- Verification failures occur when the import fails to start. This failure indicates that the data migration tool attempted to queue an import, but returned an error instead.
- Import failures happen when the import was queued successfully in the data migration tool, but failed after that point. The individual that queued the import receives a failure email.

#### Verification failures

Verification failure issues indicate that your import request isn't valid. Follow the recommended guidance provided below based on the error messages you receive. Then try to queue the import again.

#### VS403254

The region that you entered for your Azure DevOps Services import isn't supported.

```
VS403254: Region {0} may not be used for the Import, it is not a supported region.
```

Open your import specification file and update the region that you've provided with the correct short name for

the [region](#).

## VS403249

The organization name your team has selected is already in use by an existing organization. All Azure DevOps Services imports go into a new organization that is created at import time.

```
VS403249: The organization {0} already exists. Please select a different name and try the import again.
```

Select a different organization name and update the import specification file before retrying the import.

## VS403250 & VS403286

The DACPAC isn't built off a detached collection.

```
VS403250: The dacpac is not a detached Azure DevOps Server Collection database.
```

```
VS403286: The dacpac is from a Azure DevOps Server Configuration database. You must use a detached Azure DevOps Server Collection database.
```

[Detach](#) your collection database and generate the DACPAC again.

## VS403243

Unable to make a connection to the database using the provided SQL Connection String.

```
VS403243: Unable to connect to the database using the provided SQL Connection String {0}.
```

Review the parameters that were provided to ensure they're correct and try again.

## VS403260 & VS403351

The collection database isn't detached.

```
VS403260: The database is not detached.
```

```
VS403351: The DACPAC or source database is missing an expected table. It's possible that the database was not correctly detached from Azure DevOps Server.
```

[Detach](#) your collection database and retry the import queue.

## VS403261

The connection string must be encrypted otherwise the password is sent in the clear.

```
VS403261: The SQL connection string must use encryption.
```

Add **Encrypt=true** to your SQL connection string.

## VS403262

The connection string must use SQL Authentication.

```
VS403262: The SQL connection string must use SQL Authentication, Integrated Authentication is not supported.
```

Add **Integrated Security=False** to your SQL connection string.

## VS403263

Your SQL sign in user account doesn't have the required database role.

```
VS403263: The User ID {0} must be member of the database role {1}.
```

Make sure the user account for sign in is assigned the '['TFSEEXECROLE'](#)' role.

#### NOTE

There is a known issue with using `sp_addrolemember` to add 'TFSEEXECROLE' to an existing SQL login. The role membership isn't applied until all open connections using that identity are closed. If you receive the VS403263 error and have confirmed your identity has the role, we recommend that you create a new identity for your import. Details on how to create a new SQL login that's ready to be used for import can be found at [Import large collections](#).

## VS403264

The connection string doesn't point to an Azure DevOps Server collection database.

```
VS403264: The database is not a Azure DevOps Server Collection database, it cannot be used for import.
```

Verify or correct the connection string points to your collection database.

## VS40325

The Azure DevOps Server Update has queued the file migration job. Imports can't be performed until this job has completed. The completion time for this job is dependent on the size of the collection.

```
VS403255: The collection cannot be imported due to an ongoing post upgrade job. Please wait and try again later
```

You can track job progress by running the following query on the collection database:

```
SELECT COUNT (*) as remaining_files_to_migrate
FROM   tbl_FileReference
WHERE  PartitionId > 0
       AND Migrate fileId IS NOT NULL
```

Once the number of files remaining to migrate is zero, you can run the data migration tool.

## VS403282

A new line character exists in the source location value. This character could have remained after copying the SAS key from your windows console.

```
VS403282: The source location parameter contains a new line character. Please ensure the SAS key is defined on a single line in the import specification file.
```

Remove the line break and try again.

## VS403271

Your import files and DACPAC aren't located in the **required** Azure region to complete the import to your target Azure DevOps Services region.

VS403271: It appears that your DACPAC was uploaded to East US. It's required that customers targeting Central US for import put their DACPACs in Central US. Please move your DACPAC to Central US and requeue the import.

Create a new Windows Azure storage account in the required region and copy your files. The following example shows how to copy your data using AzCopy.

```
AzCopy.exe /Source:https://accountCUS.blob.core.windows.net/mycontainer /SourceKey:"primary access key"  
/Dest:https://accountCUS.blob.core.windows.net/mycontainer /DestKey:"primary access key" /S
```

## VS403316

Inconsistencies were detected in some TFVC files within your collection.

VS403316: An inconsistency was detected in some TFVC files for this collection. The inconsistency needs to be corrected prior to running an import to Azure DevOps Services. Please reach out to <https://aka.ms/AzureDevOpsImportSupport> for assistance with addressing this issue.

Work with Azure DevOps Services [customer support](#). Open a support ticket and they'll work with you to resolve the error.

## VS403366

The data migration tool was unable to connect to the SQL Azure VM.

VS403366: A problem occurred while attempting to connect to your database. Please verify that your connection string is correct and that all required IP addresses for Azure DevOps Services have been provided exceptions for your machines firewall.

List of Azure DevOps Services IPs:

Verify that you've entered the information correctly in your connection string and that you can connect to the VM.

The IPs that the error message lists are for Azure DevOps Services. Azure DevOps Services IPs can change temporarily during deployments. Add them to your firewall exceptions and try queuing the import again. For a list of IP addresses, see [Import large collections, Restrict access to Azure DevOps Services IPs only](#)

## VS403373

The data migration tool doesn't support importing multiple copies of the **SAME** collection. However, it **DOES** support importing **split** copies of a collection. Change the GUID for the *DataImportCollectionID*.

From SQL Server Management Studio (SSMS), open the extended properties for the split copies that you haven't imported yet. Add a newly generated GUID to the "TFS\_DATAIMPORT\_COLLECTIONID" property. Then rerun the **prepare** command and use the new **import.json** file to queue the import.

## VS403379

Data import will fail as one or more projects found in this collection are in the soft-deleted stage. Please restore the soft-deleted project(s) or delete them permanently before running the data import.

VS403379: Data import will fail as one or more projects found in this collection are in the soft-deleted stage. Please restore the soft-deleted project(s) or delete them permanently before running the data import.

Verify the collection against which you are running the data migration tool has projects in the soft-deleted stage.

Once a project is deleted, it remains in a soft-delete state for 28 days during which the deleted project can be restored. You can read about how to restore a deleted project in [Restore a project](#). If you have projects in the soft-deleted stage, remove them completely or restore them back before running data import.

### Import failures

When an import fails, the individual that queued the import receives an email notification. Most of the time this email includes a reason for the failure. If it does, use the troubleshooting steps provided in the email and this page to resolve the errors and retry your import.

If the error is more complex, then the email you receive provides instructions on how to file a [customer support case](#). After submitting a customer support case, your team will need to roll back by bringing your Azure DevOps Server instance back online and reattach your collection. Your team members can then continue working. We recommended you not attempt the import again until the failure causing issue is resolved.

## Related articles

- [Validate and import](#)
- [Post-import](#)

# Default permissions and access for Azure DevOps

5/12/2021 • 23 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

To use Azure DevOps features, users must be added to a security group with the appropriate permissions and granted access to the web portal. Limitations to select features are based on the *access level* and *security group* to which a user is assigned. The **Basic** access level and higher supports full access to all Azure Boards features. **Stakeholder** access level provides partial support to select features, allowing users to view and modify work items, but not use all features. **Stakeholder** access is available to support free access to a limited set of features by an unlimited set of stakeholders.

The most common built-in security groups—**Readers**, **Contributors**, and **Project Administrators**—and team administrator role grant permissions to specific features.

In general, use the following guidance when assigning users to an access level and security group:

- Grant **Basic** access or higher and add to the **Contributors** security group full-time workers who contribute to the code base or manage projects.
- Grant **Stakeholder** access and add to the **Contributors** security group managers or users who don't actively contribute to the code base but want to check project status and provide direction, feedback, feature ideas, and business alignment to a team.
- Grant **Stakeholder** access and add to the **Project Administrators** security group users tasked with managing project resources. If they also need to contribute to the code base, then you must assign them **Basic** or higher-level access.
- Grant **Stakeholder** access and add to the **Project Collection Administrators** security group users tasked with managing organization or collection resources. If they also need to contribute to the code base, then you must assign them **Basic** or higher-level access.

To learn more about administrative tasks see [About user, team, project, and organization-level settings](#). For a complete reference of all built-in groups and permissions, see [Permissions and groups](#). For information about access levels, see [About access levels](#).

In the tables provided in this article, a ✓ indicates that the corresponding access level or security group has access to a feature by default.

For a comparison chart of Stakeholder versus Basic access, see the [Feature matrix](#). To assign or change an access level, see [Add users and assign licenses](#). If you need to [grant specific users select permissions](#), you can do so.

## Azure Boards

You can plan and track work from the web portal **Boards** hub, and using Eclipse, Visual Studio, Excel, Project, and other clients. For an overview of work tracking features, see [About Agile tools](#). To change permissions, see [Set permissions and access for work tracking](#).

Users granted Stakeholder access are granted different access to features depending on whether it is a private or a public project. For private projects, Stakeholders have limited access to select work tracking functions, whereas for public projects, Stakeholders enjoy full access to work tracking features. To learn more, see [About access levels, Stakeholder access](#).

# Work tracking

You can plan and track work from the web portal **Work** hub, and using Eclipse, Visual Studio, Excel, Project, and other clients. For an overview of work tracking features, see [About Agile tools](#).

## NOTE

Team administrators can configure settings for their team's tools. Organization owners and members of the Project Administrators group can configure settings for all teams. To be added as an administrator, see [Add team administrators](#) or [Add administrators, set permissions at the project-level or project collection-level](#).

Access to the following tasks are controlled by each user's access level or by permission assignments. Members of the Readers, Contributors, or Project Administrators group are assumed to have Basic access or greater.

## General work item feature access

You can use work items to track anything you need to track. To learn more, see [Understand how work items are used to track issues, tasks, and epics](#).

## NOTE

You can change the work item type or move work items to another project within a project collection. These features require that the data warehouse is disabled. With the data warehouse disabled, you can use the [Analytics Service](#) to support your reporting needs. To learn more about disabling the data warehouse, see [Disable the data warehouse and cube](#).

## Task or permission

Stakeholder

Readers

Contributors

Project admins

---

### View work items in this node (Area Path permission)

- ✓
- ✓
- ✓
- ✓

---

### Edit work items in this node (Area Path permission)

- ✓
- ✓
- ✓
- ✓
- ✓

---

### Create tag definition (*Stakeholders can assign existing tags to work items, but can't add new tags*)

- ✓

✓

✓

---

**Change work item type** (Project-level permission)

✓

✓

✓

---

**Move work items out of this project** (Project-level permission)

✓

✓

---

Email work items

✓

✓

✓

---

Apply a work item template

✓

✓

✓

---

Delete and restore work items

(Project-level permission) (able to restore from the Recycle bin)

✓

✓

---

**Permanently delete work items** (Project-level permission)

✓

---

[Provide feedback](#) (through the Microsoft Feedback client)

✓

✓

✓

---

[Request feedback](#)

✓

✓

**NOTE**

Work items are subject to rules applied to them. Conditional rules based on user or group membership are cached for your web browser. If you find yourself restricted to update a work item, you may have encountered one of these rules. If you believe you've encountered an issue that doesn't apply to you, see [Work item form IndexDB caching issues](#). To learn more about conditional rules, see [Add a rule to a work item type \(Inheritance process\)](#) or [Apply or ignore rules based on user or group \(On-premises XML process\)](#).

**Boards feature access**

You use **Boards** to implement Kanban methods. Boards present work items as cards and support quick status updates through drag-and-drop.

---

**Task****Stakeholder****Readers****Contributors****Team admins**

---

View boards and open work items

- ✓
- ✓
- ✓
- ✓

---

View boards and open work items

- ✓
- ✓
- ✓
- ✓

---

Add work items to a board; update status through drag-and-drop

- ✓
- ✓
- ✓

---

Reorder work items or reparent child items through drag-and-drop; update a field on a card

- ✓
- ✓

---

Add work items to a board; update status, reorder, or reparent child items through drag-and-drop; update a field on a card

- ✓



---

Add work items to a board; update status through drag-and-drop



---

Add child items to a checklist



---

Assign to a sprint (from card field)



---

Assign to a sprint



---

Configure board settings

*(Stakeholders assigned as a team administrator or Project Administrator can configure team settings)*



---

### **Backlogs features access**

**Backlogs** display work items as lists. A product backlog represents your project plan and a repository of all the information you need to track and share with your team. Portfolio backlogs allow you to group and organize your backlog into a hierarchy.

---

#### **Task**

#### **Stakeholders**

#### **Readers**

#### **Contributors**

#### **Team Admins**

---

View backlogs and open work items





---

Add work items to a backlog (*Stakeholders can only add items to the bottom of the backlog*)

Read only



---

Use bulk edit features



---

Add child items to a backlog item; prioritize or reorder a backlog; parent items using the Mapping pane; Assign items to a sprint using the Planning pane



---

Add child items to a backlog item; prioritize or reorder a backlog; parent items using the Mapping pane; Assign items to a sprint using drag-and-drop



---

Configure team settings, backlog levels, show bugs, work days off

*(Stakeholders assigned as a team administrator or Project Administrator can configure team settings)*



---

### Sprints feature access

You use sprint tools to implement Scrum methods. The **Sprints** set of tools provide filtered views of work items that a team has assigned to specific iteration paths or sprints.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS
View sprint backlogs, taskboards, and open work items	✓	✓	✓	✓
Add work items to a sprint backlog <i>(Stakeholders can add backlog items to the bottom of a sprint backlog)</i>	✓		✓	✓
Add work items to a taskboard <i>(Stakeholders can add backlog items but not tasks)</i>			✓	✓
Prioritize/reorder a sprint backlog or taskboard; add child items to a backlog item; reassign items to a sprint using the Planning pane			✓	✓

View team capacity (work details)	✓	✓	✓	✓
Set team capacity			✓	✓
Use bulk edit features	✓		✓	✓
Define sprints, set sprint dates				✓
Customize a sprint backlog or taskboard, configure team settings <i>(Stakeholders assigned as a team administrator or Project Administrator can configure team settings)</i>	✓			✓

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS
View sprint backlogs, taskboards, and open work items	✓	✓	✓	✓
Add work items to a sprint backlog <i>(Stakeholders can add backlog items to the bottom of a sprint backlog)</i>	✓		✓	✓
Add work items to a taskboard <i>(Stakeholders can add backlog items but not tasks)</i>			✓	✓
Prioritize/reorder a sprint backlog or taskboard; add child items to a backlog item; reassign items to another using drag-and-drop			✓	✓
View team capacity (work details)	✓	✓	✓	✓
Set team capacity			✓	✓
Use bulk edit features	✓		✓	✓
Define sprints, set sprint dates				✓
Customize a sprint backlog or taskboard, configure team settings <i>(Stakeholders assigned as a team administrator or Project Administrator can configure team settings)</i>	✓			✓

## Queries and semantic search

[Queries](#) are filtered lists of work items based on criteria that you define by using a query editor. [Adhoc searches](#) are powered by a semantic search engine.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	PROJECT ADMINS
View and run managed queries	✓	✓	✓	✓
Create and save managed My queries	✓		✓	✓

Contribute, delete, and manage permissions of Shared queries and folders <i>(Stakeholders can't save Shared queries even if granted permissions)</i>				✓
View query charts		✓	✓	✓
Create query charts			✓	✓
Powerful semantic work-tracking search	✓	✓	✓	✓

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS
View and run managed queries	✓	✓	✓	✓
Create and save managed My queries	✓		✓	✓
Contribute, delete, and manage permissions of Shared queries and folders <i>(Stakeholders can't save Shared queries even if granted permissions)</i>				✓
View query charts		✓	✓	✓
Create query charts			✓	✓

## Delivery plans feature access

[Delivery plans](#) display work items as cards against a calendar view. This format can be an effective communication tool with managers, partners, and stakeholders for a team.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	PROJECT ADMINS
View delivery plans	✓	✓	✓	✓
Create, edit, or delete a delivery plan <i>(Contributors can only edit or delete plans that they create)</i>			✓	✓
Manage permissions for a delivery plan <i>(Contributors can only manage permissions for plans that they create)</i>				✓

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	PROJECT ADMINS
View delivery plans		✓	✓	✓
Create, edit, or delete a delivery plan <i>(Contributors can only edit or delete plans that they create)</i>			✓	✓
Manage permissions for a delivery plan <i>(Contributors can only manage permissions for plans that they create)</i>				✓

## Additional permissions

In addition to the permissions set at the [project level via the built-in groups](#), you can set permissions for the following objects: [area and iteration paths](#) and individual [queries and query folders](#).

## Azure Repos

You can manage your source code from the web portal **Repos** hub, or using Xcode, Eclipse, IntelliJ, Android Studio, Visual Studio, or Visual Studio Code.

Stakeholders for private projects have no access to **Repos**. Stakeholders for public projects have the same access to **Repos** as **Contributors**.

## Code: Source control

You can connect to your code from the web portal **Code** hub, or using Xcode, Eclipse, IntelliJ, Android Studio, Visual Studio, or Visual Studio Code. Stakeholders for private projects have no access to **Code**.

### Git

You can use [Git repositories](#) to host and collaborate on your source code. For an overview of code features and functions.

---

#### Permission

Readers

Contributors

Build Admins

Project Admins

---

**Read** (clone, fetch, and explore the contents of a repository); also, can create, comment on, vote, and **Contribute to pull requests**

- ✓
  - ✓
  - ✓
  - ✓
- 

**Contribute to a repository, Create branches, Create tags, and Manage notes**

- ✓
  - ✓
  - ✓
- 

**Bypass policies when pushing** to a repository

- ✓
- 

**Create repository, Delete repository, and Rename repository**

- ✓
- 

**Edit policies, Force push (rewrite history, delete branches and tags), Manage permissions, Remove others' locks**



---

Bypass policies when completing pull requests (not set for any security group)

---

By default, the project-level Readers groups have read-only permissions.

---

#### Permission

Contributors

Build Admins

Project Admins

---

**Branch Creation:** At the repository level, can push their changes to branches in the repository. Does not override restrictions in place from [branch policies](#). At the branch level, can push their changes to the branch and lock the branch.



**Contribute:** At the repository level, can push their changes to branches in the repository. Does not override restrictions in place from [branch policies](#). At the branch level, can push their changes to the branch and lock the branch.



---

**Note Management:** Can push and edit Git notes to the repository. They can also remove notes from items if they have the **Force** permission.



---

**Tag Creation:** Can push tags to the repository, and can also edit or remove tags if they have the **Force** permission.



---

**Administer:** Delete and rename repositories: If assigned to the top-level **Git repositories** entry, can add additional repositories. At the branch level, users can set permissions for the branch and unlock the branch. The Administer permission set on an individual Git repository does not grant the ability to rename or delete the repository. These tasks require Administer permissions at the Git repositories top-level.



---

**Rewrite and destroy history (force push):** Can force an update to a branch and delete a branch. A force

update can overwrite commits added from any user. Users with this permission can modify the commit history of a branch.



By default, the Project Collection Build Service can read from all repositories. Any pipeline which runs within the project collection scope can potentially read any repository in the organization or collection. To remove this permission for a repository, change the **Read** permission to **Deny** for the Project Collection Build Service.

## TFVC

[Team Foundation Version Control \(TFVC\)](#) provides a centralized version control system to manage your source control.

### NOTE

Tasks such as create, delete, or rename a TFVC repository are not supported. Once a TFVC repository is created you can't delete it. Also, you can only have one TFVC repository per project. This is different from Git repositories which allow for adding, renaming, and deleting multiple repositories.

## Permission

### Readers

### Contributors

### Build Admins

### Project Admins

## Check in, Label, Lock, Merge, Pend a change in a server workspace, Read

Read only



## Administer labels, Manage branches, Manage permissions, Revise other users' changes, Undo other users' changes, Unlock other users' changes



## Azure Pipelines

You can define and manage your builds and releases from the web portal **Pipelines** hub. For an overview of pipelines features and functions, see [Continuous integration on any platform](#).

### NOTE

When the **Free access to Pipelines for Stakeholders** preview feature is enabled for the organization, Stakeholders get access to all **Build** and **Release** features. This is indicated by the preview icon shown in the following table. Without this feature enabled, stakeholders can only view and approve releases. To learn more, see [Provide Stakeholders access to edit build and release pipelines](#).

Task	Stakeholders	Readers	Contributors	Build Admins	Project Admins	Release Admins
View release pipelines	<input type="checkbox"/>	✓	✓	✓	✓	✓
Define builds with continuous integration	<input type="checkbox"/>		✓	✓	✓	
Define releases and manage deployments	<input type="checkbox"/>		✓		✓	✓
Approve releases	✓		✓		✓	✓
Azure Artifacts (5 users free)	<input type="checkbox"/>		✓		✓	✓
Queue builds, edit build quality	<input type="checkbox"/>		✓	✓	✓	
Manage build queues and build qualities	<input type="checkbox"/>			✓	✓	
Manage build retention policies, delete and destroy builds	<input type="checkbox"/>		✓	✓	✓	
Administer build permissions	<input type="checkbox"/>			✓	✓	
Manage release permissions	<input type="checkbox"/>				✓	✓
Create and edit task groups	<input type="checkbox"/>		✓	✓	✓	✓
Manage task group permissions	<input type="checkbox"/>			✓	✓	✓
Can view library items such as variable groups	<input type="checkbox"/>	✓	✓	✓	✓	✓

Task	Stakeholders	Readers	Contributors	Build Admins	Project Admins	Release Admins
Use and manage library items such as variable groups	□			✓	✓	✓

## Build

Task	Stakeholders	Readers	Contributors	Build Admins	Project Admins
View builds	✓	✓	✓	✓	✓
View build pipeline	✓	✓	✓	✓	✓
Administer build permissions				✓	✓
Delete or Edit build pipeline			✓	✓	✓
Delete or Destroy builds				✓	✓
Edit build quality			✓	✓	✓
Manage build qualities				✓	✓
Manage build queue				✓	✓
Override check-in validation by build					✓
Queue builds			✓	✓	✓
Retain indefinitely				✓	✓
Stop builds				✓	✓
Update build information					✓

## Release

Task	Stakeholders	Readers	Contributors	Project Admins	Release Admins
Approve releases	✓		✓	✓	✓

View releases	✓	✓	✓	✓	✓
View release pipeline		✓	✓	✓	✓
Administer release permissions				✓	✓
Delete release pipeline or release stage			✓	✓	✓
Delete releases			✓	✓	✓
Edit release pipeline				✓	✓
Edit release stage			✓	✓	✓
Manage deployments				✓	✓
Manage release approvers			✓	✓	✓
Manage releases				✓	✓

## Task groups

You use task groups to encapsulate a sequence of tasks already defined in a build or a release pipeline into a single reusable task. Task group permissions follow a hierarchical model. You can set defaults for all permissions at the project-level and over-write on an individual task group pipeline. You [define and manage task groups](#) in the **Task groups** tab in Azure Pipelines.

TASK	STAKEHOLDER S	READERS	CONTRIBUTOR S	BUILD ADMINS	PROJECT ADMINS	RELEASE ADMINS
Administer task group permissions				✓	✓	✓
Delete task group				✓	✓	✓
Edit task group				✓	✓	✓

## Build and Release

You can define and manage your builds and releases from the web portal, **Build and Release**. For an overview of pipelines features and functions, see [Continuous integration on any platform](#). From the web portal, you can set permissions for all or individual builds and releases. See [Set build and release permissions](#).

### Build

Task	Stakeholders	Readers	Contributors	Build Admins	Project Admins
View builds		✓	✓	✓	✓
View build definition		✓	✓	✓	✓
Administer build permissions				✓	✓
Delete or Edit build definitions			✓	✓	✓
Delete or Destroy builds				✓	✓
Edit build quality			✓	✓	✓
Manage build qualities				✓	✓
Manage build queue				✓	✓
Override check-in validation by build					✓
Queue builds			✓	✓	✓
Retain indefinitely				✓	✓
Stop builds				✓	✓
Update build information					✓

## Release

Task	Stakeholders	Readers	Contributors	Project Admins	Release Admins
Approve releases	✓		✓	✓	✓
View releases	✓	✓	✓	✓	✓
View release definition		✓	✓	✓	✓
Administer release permissions				✓	✓

Delete release definition or release stage		✓	✓	✓
Delete releases		✓	✓	✓
Edit release definition			✓	✓
Edit release stage		✓	✓	✓
Manage deployments			✓	✓
Manage release approvers		✓	✓	✓
Manage releases			✓	✓

## Azure Test Plans

### Test

You can define and manage manual tests from the web portal, **Test Plans** or **Test**. For an overview of manual test features and functions, see [Testing overview](#). You set test permissions at the project level from **Project Settings > Permissions**.

---

Task

Stakeholder

Readers

Contributors

Project Admins

---

Access Azure Test Plans (formerly Test Manager, purchased separately)

✓

✓

---

Create and delete test runs

✓

✓

---

Provide feedback using the Test & Feedback extension

✓

✓

✓



---

Request feedback using the Test & Feedback extension



---

Exploratory testing, view test runs



---

Manage test plans and test suites

Manage test configurations and test environments



---

Exploratory testing



---

Apply a work item template to a test case



---

Delete delete test plans, test cases, and other test related work items (able to restore from the Recycle bin)



---

Permanently delete test plans, test cases, and other test related work items (same as Permanently delete work items)



---

## Azure Artifacts

You can manage feeds from the web portal, [Artifacts](#). Users granted Stakeholder or Basic access, or higher can access Azure Artifacts features. To set permissions, see [Secure feeds using permissions](#).

You can manage feeds from the web portal, [Artifacts](#). Users granted Basic access or higher can access Azure Artifacts features. Users granted Stakeholder access have no access to Azure Artifacts. To set permissions, see [Secure feeds using permissions](#).

# Package management

You can manage feeds from the web portal, **Build and release > Packages**. Users granted Basic access or higher can access Package management features. Users granted Stakeholder access have no access. To set permissions, see [Secure feeds using permissions](#).

Feeds have four permission roles: Readers, Collaborators, Contributors, and Owners. Owners can add user accounts or security groups to any role.

PERMISSION	READER	COLLABORATOR	CONTRIBUTOR	OWNER
List, install, and restore packages	✓	✓	✓	✓
Push packages			✓	✓
Unlist/deprecate packages			✓	✓
Delete/unpublish package				✓
Promote a package to a view			✓	✓
Add/remove upstream sources				✓
Save packages from upstream sources		✓	✓	✓
Edit feed permissions				✓

By default, the Project Collection Build Service is a Contributor and your project team is a Reader.

## NOTE

To access a feed in a different organization, a user must be given access to the project hosting that feed.

Feeds have three permission roles: Readers, Contributors, and Owners. Owners can add user accounts or security groups to any role.

PERMISSION	READER	CONTRIBUTOR	OWNER
List and restore/install packages	✓	✓	✓
Push packages		✓	✓
Unlist/deprecate packages		✓	✓
Delete/unpublish package			✓
Edit feed permissions			✓

PERMISSION	READER	CONTRIBUTOR	OWNER
Rename and delete feed			✓

By default, the Project Collection Build Service is a Contributor and your project team is a Reader.

#### NOTE

To access a feed in a different organization, a user must be given access to the project hosting that feed.

## Notifications, alerts, and team collaboration tools

To manage notifications, see [Manage personal notifications](#) and [Manage team notifications](#).

#### NOTE

There are no UI permissions associated with managing notifications. Instead, you can manage them using the [TFS Security command line tool](#).

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	ORGANIZATION OWNER/PROJECT ADMINS
Set personal notifications or alerts	✓		✓	✓	✓
Set team notifications or alerts				✓	✓
Set project-level notifications or alerts					✓
READMEs	See Note 1	✓	✓	✓	✓
View Project Wikis	✓	✓	✓	✓	✓
View Code Wikis		✓	✓	✓	✓
Provision or create a Wiki					✓
Publish Code as Wiki			✓	See Note 2	See Note 2
View the project page	✓	✓	✓	✓	✓
Edit the project page					✓
Navigate using the Project pages	✓	✓	✓	✓	✓
Request feedback		✓	✓	✓	✓
Provide feedback	✓	✓	✓	✓	✓
Powerful semantic code search	✓	✓	✓	✓	✓

Powerful semantic work tracking search	✓	✓	✓	✓	✓
--	---	---	---	---	---

#### Notes

1. Can view project READMEs, but not READMEs defined for a repository.
2. Project Admins or Team Admins with contribute permission can publish code as wiki. Project Admins have this permission by default.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	ORGANIZATION OWNER/PROJECT ADMINS
Set personal notifications or alerts	✓		✓	✓	✓
Set team notifications or alerts				✓	✓
Set project-level notifications or alerts					✓
READMEs	See Note 1	✓	✓	✓	✓
View Project Wikis	✓	✓	✓	✓	✓
View Code Wikis		✓	✓	✓	✓
Provision or create a Wiki					✓
Publish Code as Wiki			✓	See Note 2	See Note 2
View the project page	✓	✓	✓	✓	✓
Edit the project page					✓
Navigate using the Project pages	✓	✓	✓	✓	✓
Request feedback		✓	✓	✓	✓
Provide feedback	✓	✓	✓	✓	✓
Powerful semantic code search	✓	✓	✓	✓	✓
Powerful semantic work tracking search	✓	✓	✓	✓	✓

#### Notes

1. Can view project READMEs, but not READMEs defined for a repository.
2. Project Admins or Team Admins with contribute permission can publish code as wiki. Project Admins have this permission by default.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	ORGANIZATION OWNER/PROJECT ADMINS
Set personal notifications or alerts	✓		✓	✓	✓

Set team notifications or alerts				✓	✓
Set project-level notifications or alerts					✓
Participate in Team (chat) rooms			✓	✓	✓
READMEs <i>Can view project READMEs, but not READMEs defined for a repository.</i>	Partial access	✓	✓	✓	✓
Request feedback		✓	✓	✓	✓
Provide feedback	✓	✓	✓	✓	✓

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	ORGANIZATION OWNER/PROJECT ADMINS
Set personal notifications or alerts	✓		✓	✓	✓
Set team notifications or alerts				✓	✓
Set project-level notifications or alerts					✓
Participate in Team (chat) rooms			✓	✓	✓
Request feedback		✓	✓	✓	✓
Provide feedback	✓	✓	✓	✓	✓

## Dashboards, charts, reports, and widgets

You can define and manage team and project dashboards from the web portal, [Dashboards](#). For an overview of dashboard and chart features, see [Dashboards](#). You can set [individual dashboard permissions](#) to grant or restrict the ability to edit or delete dashboards.

Users granted Stakeholder access to private projects can't view or create query charts. Stakeholder access to public projects can view and create query charts.

You can define and manage team dashboards from the web portal, [Dashboards](#). For an overview of dashboard and chart features, see [Dashboards](#). You set [dashboard permissions at the team level](#) from the team dashboard page.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	PROJECT ADMINS
View work item query charts (from the Queries page)		✓	✓	✓	✓
Create work item query and test tracking charts <sup>1</sup>			✓	✓	✓

View team and project dashboards (including work item query charts added to dashboards)	✓	✓	✓	✓	✓
Add and configure team dashboards <sup>1</sup>			✓	✓	✓
Add and configure project dashboards <sup>1</sup>			✓	✓	✓

**Notes:**

1. Public project Stakeholders have full access to all features.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	PROJECT ADMINS
View charts and dashboards	✓	✓	✓	✓	✓
Create work item and test tracking charts			✓	✓	✓
Add and configure dashboards			With permissions set	✓	✓

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	PROJECT ADMINS
View team dashboard home page	✓	✓	✓	✓	✓
Create work item and test tracking charts			✓	✓	✓

## Dashboards and charts

You can pin charts to a team dashboard Home page.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	PROJECT ADMINS
View work item query charts (from the Queries page)		✓	✓	✓	✓
Create work item query and test tracking charts <sup>1</sup>			✓	✓	✓
View team and project dashboards (including work item query charts added to dashboards)	✓	✓	✓	✓	✓
Add and configure team dashboards <sup>1</sup>			✓	✓	✓
Add and configure project dashboards <sup>1</sup>			✓	✓	✓

**Notes:**

1. Public project Stakeholders have full access to all features.

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	PROJECT ADMINS

View charts and dashboards	✓	✓	✓	✓	✓
Create work item and test tracking charts			✓	✓	✓
Add and configure dashboards			With permissions set	✓	✓

TASK	STAKEHOLDERS	READERS	CONTRIBUTORS	TEAM ADMINS	PROJECT ADMINS
View team dashboard home page	✓	✓	✓	✓	✓
Create work item and test tracking charts			✓	✓	✓

## Power BI Integration and Analytics views

From the web portal [Analytics views](#), you can create and manage Analytics views. An Analytics view provides a simplified way to specify the filter criteria for a Power BI report based on the Analytics Service data store. The Analytics Service is the reporting platform for Azure DevOps. To learn more, see [What is the Analytics Service?](#).

You set [permissions](#) for the service at the project level, and for shared Analytics views at the object level. Users with Stakeholder access have no access to view or edit Analytics views.

TASK	READERS	CONTRIBUTORS	PROJECT ADMINS
View Analytics	✓	✓	✓
View a shared Analytics view		✓	✓
Edit and delete Analytics views			✓

## Related articles

- [Add users to a project or team](#)
- [Security and permission management tools](#)
- [Permissions and groups reference](#)
- [About access levels](#)
- [Web portal navigation](#)
- [Troubleshoot permissions](#)

# About access levels

4/12/2021 • 16 minutes to read • [Edit Online](#)

Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

Access levels grant or restrict access to select web portal features. This is in addition to permissions granted through security groups, which provide or restrict specific tasks. Access levels enable administrators to provide their user base access to the features they need and only pay for those features.

## IMPORTANT

To view the content available for your platform, make sure that you select the correct version of this article from the version selector which is located above the table of contents. Feature support differs depending on whether you are working from Azure DevOps Services or an on-premises version of Azure DevOps Server, renamed from Team Foundation Server (TFS).

To learn which on-premises version you are using, see [What platform/version am I using?](#)

When you add a user or group to a team or project, they're automatically granted access to those features supported by the default access level and those supported by the security group to which they are added. Most users can access most features by being assigned to the **Basic** access level and **Contributors** security group. For a simplified overview of the permissions assigned to the most common groups **Readers**, **Contributors**, and **Project Administrators** as well as the **Stakeholder** access group, see [Permissions and access](#).

To add user accounts or groups to specific access levels, see [Manage users and access](#). Make sure to set each user's access level based on what you've purchased for that user.

To add user accounts or groups to specific access levels, see [Change access levels](#). Make sure to set each user's access level based on what you've purchased for that user.

## Supported access levels

Assign users or groups of users to one of the following access levels:

- **Stakeholder:** Provides partial access, can be assigned to unlimited users for free. Assign to users with no license or subscriptions who need access to a limited set of features.
- **Basic:** Provides access to most features. Assign to users with a Visual Studio Professional subscription, an Azure DevOps Server CAL, and to users for whom you're paying for Basic access in an organization.
- **Basic + Test Plans:** Provides access to all features included in Basic, as well as Azure Test Plans. Assign to users with a Visual Studio Test Professional or MSDN Platforms subscription, and to users for whom you're paying for Basic + Test Plans access in an organization.
- **Visual Studio subscription:** Assign to users who already have a Visual Studio subscription. The system automatically recognizes the user's subscription—Visual Studio Enterprise, Visual Studio Professional, Visual Studio Test Professional, or MSDN Platform—and enables any other features that are included in their subscription level. If you assign Basic or Stakeholder, they also receive their Visual Studio subscription benefits upon sign-in.

**TIP**

As a best practice when adding new users, we recommend assigning the Visual Studio subscriber level when appropriate (as opposed to Basic) to prevent being charged the Basic rate before the user signs in for the first time.

- **Stakeholder:** Provides partial access, can be assigned to unlimited users for free. Assign to users with no license or subscriptions who need access to a limited set of features.
- **Basic:** Provides access to most features. Assign to users with an Azure DevOps Server CAL, with a Visual Studio Professional subscription, and to users for whom you're paying for Basic access in an organization.
- **Basic + Test Plans:** Provides access for users who have a monthly Test Manager subscription, Visual Studio Test Professional, or MSDN Platforms subscription.
- **VS Enterprise:** Provides access to premium features. Assign to users with a subscription to Visual Studio Enterprise.
- **Stakeholder:** Provides partial access, can be assigned to unlimited users for free. Assign to users with no license or subscriptions who need access to a limited set of features.
- **Basic:** Provides access to most features. Assign to users with a CAL or with a Visual Studio Professional subscription.
- **Advanced** (legacy access level, deprecated in Azure DevOps Server 2019): Provides access to premium features. Only assign to users with a subscription to MSDN Platforms or Visual Studio Test Professional.
- **VS Enterprise:** Provides access to premium features. Assign to users with a subscription to Visual Studio Enterprise.
- **Stakeholder:** Provides partial access, can be assigned to unlimited users for free. Assign to users with no license or subscriptions who need access to a limited set of features.
- **Basic:** Provides access to most features. Assign to users with a CAL or with a Visual Studio subscription.
- **Advanced** (TFS 2017): Provides access to premium features. Only assign to users with a subscription to MSDN Platforms or Visual Studio Test Professional.
- **VS Enterprise** (TFS 2017.1 and later versions): Provides access to premium features. Assign to users with a subscription to Visual Studio Enterprise.

The following table indicates those features available for each supported access level. Visual Studio Test Professional and MSDN Platform subscriptions grant access to the same features as Visual Studio Enterprise.

---

**Feature****Stakeholder**

**Basic &  
Visual Studio Professional**

**Basic + Test Plans &  
Visual Studio Enterprise**

---

**Feature****Stakeholder**

**Basic &  
Visual Studio Professional**

**Basic + Test Plans &  
Visual Studio Enterprise**

---

## Feature

### Stakeholder

Basic &  
Visual Studio Professional

Advanced &  
Visual Studio Enterprise

---

## Feature

### Stakeholder (Limited)

Basic  
(Standard)

Advanced  
(Full)

---

### Administer organization

Can configure resources when also added to a security group or role: team administrator, Project Administrator, or Project Collection Administrator.

- ✓
  - ✓
  - ✓
- 

### Advanced backlog and sprint planning tools

Includes full access to all [backlog](#) and [sprint planning](#) tools.

- ✓
  - ✓
- 

### Advanced home page

Includes [access to projects, work items, and pull requests defined across projects you work in](#).

- ✓
  - ✓
- 

### Advanced portfolio management

Includes full access to define features and epics from a [portfolio backlog](#) or [Kanban board](#).

- ✓
  - ✓
- 

### Agile boards

Stakeholders have limited access to [Kanban boards](#) and [Taskboards](#). Stakeholders can add work items and update status through drag-and-drop, but can't update fields displayed on cards (except for the work item State) and can't [view or set capacity](#).

- ✓
- ✓



## Agile boards

Stakeholders have limited access to [Kanban boards](#) and [Taskboards](#). Stakeholders can't add work items, drag-and-drop cards to update status, update fields displayed on cards, nor [view or set capacity](#).



---

## Agile Portfolio Management

Includes limited access to [portfolio backlogs](#) and [Kanban boards](#). Stakeholders can't change the backlog priority order, can't assign items to an iteration, use the mapping pane, or exercise forecasting.



## Agile Portfolio Management

Includes limited access to [portfolio backlogs](#) and [Kanban boards](#). Stakeholders can't change the backlog priority order, can't assign items to an iteration, use the mapping pane, or exercise forecasting.



## Artifacts

Includes full access to all Azure Artifacts features, up to 2 GiB free storage.



---

## Artifacts

Includes full access to all Azure Artifacts features, up to 2 GiB free storage.



---

## Author Release Pipelines and Manage Releases

Includes defining [release pipelines](#), [multi-stage continuous deployment \(CD\) pipelines](#), and [using approvals and gates to control deployments](#); when the [Free access to Pipelines Preview feature is enabled](#), Stakeholders gain access to all Azure Pipelines features.



---

## Basic backlog and sprint planning tools

Includes limited access to add and modify items on [backlogs](#) and [sprint backlogs and taskboards](#). Stakeholders can't assign items to an iteration, use the mapping pane, or forecasting.



---

## Build

Includes full access to all features to [manage continuous integration and continuous delivery of software](#).





### Chart Authoring

Can create work tracking [query charts](#).



---

### Chart Viewing

Can only view work tracking query charts. Stakeholders can't view query charts from the Queries page, however can view them when added to a dashboard.



---

### Code

Includes full access to all features to manage code using [Git repositories](#) or using [Team Foundation Version Control \(TFVC\)](#) Team Foundation Version Control (TFVC).



---

### Delivery Plans

Includes full access to add and view Delivery plans.



**Request and Manage Feedback** Includes full access to request and manage feedback on working software.



---

### Standard Features

Includes [working across projects](#), [View dashboards](#), [View wikis](#), and [Manage personal notifications](#). Stakeholders can't view markdown README files defined for repositories and can only read wiki pages.



---

### Team rooms

Requires TFS 2017 or earlier versions. Deprecated for TFS 2018 and later versions.



---

### Test services in build and release

Includes [running unit tests with your builds](#), [reviewing](#), and [analyzing](#) test results.





## Test Case Management

Includes [adding test plans and test suites](#), [creating manual test cases](#), [deleting test artifacts](#), and [testing different configurations](#).



## Test Execution and Test Analysis

Includes running [manual](#), [tracking test status](#), and [automated tests](#).



## Test summary access to Stakeholder license

Includes [requesting Stakeholder feedback using the Test & Feedback extension](#).



## View My Work Items

Access to [add and modify work items](#), [follow work items](#), [view and create queries](#), and [submit, view, and change feedback responses](#). Stakeholders can only assign existing tags to work items (can't add new tags) and can only save queries under My Queries (can't save under Shared Queries).



## View Releases and Manage Approvals

Includes [viewing releases](#) and [approving releases](#); when the [Free access to Pipelines Preview feature is enabled](#) feature is enabled, Stakeholders gain access to all Azure Pipelines features.



## Stakeholder access

With Stakeholder access, users can create and modify work items and create and save queries. They have limited access to several Azure Boards features. They also can view and approve release pipelines and perform administrative tasks when granted administrative permissions or added to an administrative group.

To get started as a Stakeholder, see [Get started as a Stakeholder](#).

### Public versus private feature access

Stakeholder access grants access to features differently depending on whether you're working from a private or a public project. To learn more about public projects, see [What is a public project?](#).

Service, application, or setting	Private project	Public project
Dashboards	Partial access	Full access
Wiki (Project wiki)	Partial access	Full access
Wiki (Code wiki)	No access	Full access
Azure Boards	Partial access	Full access
Azure Repos	No access	Full access
Azure Pipelines	Full access	Full access
Azure Test Plans	No access	No access
Azure Artifacts	Full access	Full access
Notifications	Full access	Full access
Semantic search	Full access	Full access
Project settings	Partial access	Partial access
Organization settings	Partial access	Partial access

### Features not available to users with Stakeholder access

If a Stakeholder needs access to one or more of the following features—which support the daily work of product owners, team leads, developers, testers, and project administrators—you need to provide them **Basic** access.

#### NOTE

Even if Stakeholders are explicitly granted permissions to some features, they are disallowed access to the feature due to their access level. Stakeholders that choose a feature that's not available to them receive an error message indicating that they don't have permissions to complete the task.

#### For Private projects:

- Change the priority of an item within a backlog or board
- Delete work items or move work items to another project
- Change fields on cards on a Kanban board or Taskboard, except for State field
- Drag-and-drop work items from a Backlog to the Mapping pane (parent a work item) or Planning pane (to assign to a sprint)
- Add new work item tags
- Create shared queries, view query charts, and modify the home page
- View Delivery Plans
- Access the full set of features under **Pipelines**, **Repos**, or **Test Plans**.

#### For Public projects:

- View Delivery Plans (a Marketplace extension)
- Access the full set of features under **Repos** or **Test Plans**.
- Change the priority of an item within a backlog or board

- Delete work items or move work items to another project
- Change fields on cards on a Kanban board or Taskboard, except for State field
- Drag-and-drop work items from a Backlog to the Mapping pane (parent a work item) or Planning pane (to assign to a sprint)
- Add new work item tags
- Create shared queries, view query charts, and modify the home page
- View Delivery Plans (a Marketplace extension)
- Access the full set of features under **Pipelines, Repos, or Test Plans.**
  
- Drag-and-drop work items from one column to another on a Kanban board or Taskboard to change the work item state
- Change the priority of an item within a backlog or board
- Delete work items or move work items to another project
- Change fields on cards on a Kanban board or Taskboard
- Drag-and-drop work items from a Backlog to the Mapping pane (parent a work item) or Planning pane (to assign to a sprint)
- Add new work item tags
- Create shared queries, view query charts, and modify the home page
- View Delivery Plans (a Marketplace extension)
- Access the full set of features under **Pipelines, Repos, or Test Plans.**
  
- Drag-and-drop work items from one column to another on a Kanban board or Taskboard to change the work item state
- Change the priority of an item within a backlog or board
- Delete work items or move work items to another project
- Change fields on cards on a Kanban board or Taskboard
- Drag-and-drop work items from a Backlog to the Mapping pane (parent a work item) or Planning pane (to assign to a sprint)
- Add new work item tags
- Create shared queries, view query charts, and modify the home page
- View Delivery Plans (a Marketplace extension)
- Access the full set of features under **Code, Build and Release, or Test.**
  
- Change the priority of an item within a backlog
- Delete work items
- Add work items, drag-and-drop work items, or change fields on cards on a Kanban board
- Add new work item tags
- Create shared queries, view charts, and modify dashboards
- View Delivery Plans (a Marketplace extension)
- Access the full set of features under **Code, Build and Release, or Test**
- Participate in team rooms, which capture interactive, detailed conversations about the project.
  
- Change the priority of an item within a backlog
- Delete work items
- Add work items, drag-and-drop work items, or change fields on cards on a Kanban board
- Add new work item tags
- Create shared queries, view charts, and modify the home page
- Access the full set of features under **Code, Build and Release, or Test**
- Participate in team rooms, which capture interactive, detailed conversations about the project.

## Visual Studio subscription access

Visual Studio subscribers are entitled to **Visual Studio subscription** features as a subscriber benefit. When you add those users, be sure to assign them the **Visual Studio subscription** access level.

The system automatically recognizes their subscription and enables any other features that are included, based on their subscription level.

## VS Enterprise access

Visual Studio Enterprise subscribers are entitled to **VS Enterprise** access as a subscriber benefit. When you add those users, be sure to assign them the **VS Enterprise** access level.

With VS Enterprise access, users have access to any fee-based, Marketplace extension published by Microsoft Marketplace extension published by Microsoft that is included for active Visual Studio Enterprise subscribers.

For TFS 2017.2 and later versions, assign **VS Enterprise** to those users for whom you've purchased Visual Studio Enterprise. These include a TFS CAL plus the rights to access VS Enterprise features. (For users with MSDN Platforms subscriptions or Test Professional, assign the Basic access level and the Test Manager extension for Azure Test Plans.) To learn more, see [Assign paid extension access to users](#). For example, for users with Visual Studio Test Professional or Visual Studio Enterprise, assign them [access to the Test Manager extension for Azure Test Plans](#).

## Advanced access

Users assigned Advanced access can manage test cases when you have [purchased the Test Manager extension](#) for Azure Test Plans and assigned to the user accounts to gain full access to [Web-based test case management tools](#).

Users assigned Advanced access have all the Basic features, plus [web-based test case management tools](#). You can [buy monthly access](#) or add users who already have a Visual Studio Test Professional with MSDN or MSDN Platforms subscription.

For TFS 2017 and earlier versions, you should assign the **Advanced** level to those users for whom you've purchased the full Test feature set. Here are the purchasing options:

- Higher-level Visual Studio subscriptions: Visual Studio Test Professional, Visual Studio Enterprise, or MSDN Platforms subscriptions. These include a TFS CAL plus the rights to access the full set of Test features.
- A paid Azure DevOps user (which includes a TFS CAL) plus the [Test Manager extension](#).

For TFS 2017.2, Assign **Advanced** access to those users for whom you've purchased MSDN Platforms or Visual Studio Test Professional subscriptions. These include a TFS CAL plus the rights to access Test Manager. To learn more, see [Get extensions for TFS](#), [Assign paid extension access to users](#).

### NOTE

With TFS 2017.1, the Advanced access level was temporarily disabled. Updating to TFS 2017.2 re-enables it. If you are on TFS 2017.1 and have users with Visual Studio Test Professional or MSDN Platforms subscriptions, you should assign them Basic access. In addition, you need to open **Users** for the project collections in which they are a member and [assign them the Test Manager extension for Azure Test Plans](#). To learn more, see [Buy access to TFS or TFS Test](#).

## Programmatic mapping of access levels

You can manage access levels programmatically using the [az devops user add](#) (Azure DevOps Services only) or the [User Entitlement - Add REST API](#). The following table provides a mapping of the access level selected

through the user interface and the `AccountLicenseType` and `msdnLicenseType` parameters.

ACCESS LEVEL (USER INTERFACE)	ACCOUNTLICENSETYPE	MSDNLICENSETYPE
Stakeholder	stakeholder	none
Basic	express	none
Basic + Test Plans	advanced	none
Visual Studio subscriber	none	eligible
Visual Studio Enterprise	none	enterprise

**NOTE**

The `earlyAdopter` `AccountLicenseType` is an internal value used solely by Microsoft.

You can manage access levels programmatically using the [User Entitlement - Add REST API](#). The following table provides a mapping of the access level selected through the user interface and the `AccountLicenseType` and `msdnLicenseType` parameters.

ACCESS LEVEL (USER INTERFACE)	ACCOUNTLICENSETYPE	MSDNLICENSETYPE
Stakeholder	stakeholder	none
Basic	express	none
Basic + Test Plans	advanced	none
Visual Studio subscriber	none	eligible
Visual Studio Enterprise	none	enterprise

You can manage access levels programmatically using the [User Entitlement - Add REST API](#). The following table provides a mapping of the access level selected through the user interface and the `AccountLicenseType` and `msdnLicenseType` parameters.

ACCESS LEVEL (USER INTERFACE)	ACCOUNTLICENSETYPE	MSDNLICENSETYPE
Stakeholder	stakeholder	none
Basic	express	none
Advanced	advanced	none
MSDN Platforms	none	platforms
VS Enterprise	none	enterprise

What features are available to users who are added to two different

## access levels?

If a user belongs to a group that has **Basic** access and another group that has **VS Enterprise** access, the user has access to all features available through **VS Enterprise**, which is a superset of **Basic**.

## Service account access

Azure DevOps Server [service accounts](#) are added to the default access level. If you make Stakeholder the default access level, you must add the service accounts to Basic or Advanced/VS Enterprise access.

Service accounts don't require a CAL or other purchase.

## Related articles

- [Free access to Pipelines Preview](#)
- [Manage users and access](#)
- [Get started as a Stakeholder](#)
- [Export a list of users and their access levels](#)
- [Default permissions and access](#)
- [Change access levels](#)
- [Get started as a Stakeholder](#)
- [Export a list of users and their access levels](#)
- [Default permissions and access](#)
- [Compare features between plans](#)

# Azure DevOps Services status

5/11/2021 • 3 minutes to read • [Edit Online](#)

## Azure DevOps Services

We have a team of engineers around the world who look after the health of Azure DevOps 24 hours a day. Their primary goal is to ensure that our customers are always productive and successful with our service. From time to time, like any online service, our service experiences performance slowdowns and stability issues. In these cases, we aim to respond quickly to restore the service. It's our top priority to communicate the incident status and our next steps to mitigate the issue. We do this through the [Azure DevOps Services status portal](#).

If you're experiencing a problem with any of our Azure DevOps Services, you can check the service health to determine if we're already working on the issue. Many of the events we post are based on our Customer Impact Assessment (CIA). The CIA is modeled after our availability model that measures real customer experiences representing both reliability and performance.

## Services within the product suite

Azure DevOps is a product suite of service offerings. The [geographic region](#) indicates where an organization is hosted in the cloud. The data residency, sovereignty, compliance, and resilience requirements are honored within the geographical boundaries.

In addition to the specific Azure DevOps Services, the matrix also displays two other categories: **Core** and **Other**. The **Core** category encompasses the set of features that are fundamental to all five services, such as authentication or the web portal. The **Other** category corresponds to features that complement the suite, such as extensions.

For more information about pricing and acquisition, see the [pricing and acquisition page](#).

## Service health matrix

The service status portal provides a two-dimensional matrix view of active events mapped to a given service and geography. To help clarify which specific aspects of the service are affected, we communicate impact of each of these services by geographic region in the service matrix.

## Service health indicators

The Azure DevOps Services status portal indicates the status of Azure DevOps services according to the following indicators. These indicators reflect the severity of a service health event based on the number of customers affected by the issue. Typically, the highest severity events impact a large percentage of our customers and render some parts of the product unusable.

 **Healthy**    **Degraded**    **Unhealthy**    **Advisory**

- **Healthy:** Indicates the service is broadly available.
- **Degraded:** Indicates a lower-severity event that affects the performance of a service feature, but doesn't impact broad service availability.
- **Unhealthy:** Indicates a high-severity event that affects the performance of a service and it's broad availability.
- **Advisory:** Indicates that a service is under investigation to determine the performance and availability

impact.

## Service status and event logs

You can access more information on active events from the [Status history page](#). This page provides a view into current active events and past events. Each event under investigation or previously investigated is logged in the form of an event log. Each log has other associated information such as the impacted service, geography, and event duration. Choose the provided hyperlink to view the event log, which provides detailed information on the event under investigation.

You can also filter the logs to adjust the scope of your search into past events. In addition, you can use the REST API build automated alerting solutions to help you stay on top of events.

## When and how to report availability issues

If you're experiencing an issue with Azure DevOps and see a corresponding event that's communicated on the service health portal, we're already working to restore normal operations of the service. You don't need to take any further action to notify us.

However, if you don't see your issue reported on the Azure DevOps Services health page, you can report your issue using the Report a Service Availability Issue form. Or, you can ask a question through the [Azure DevOps Services virtual support agent](#).

For issues not related to availability, refer to our [Developer Community portal](#).

## RSS feed

You can use [the RSS feed](#) to subscribe and receive information in your feed reader.

## Use REST APIs to build automated solutions

The [Azure Resource health REST API](#) can retrieve the current health status of each of the Azure DevOps Services. You can use it to build an automated solution to [monitor the infrastructure incidents](#).

### NOTE

Looking for Azure DevOps REST APIs? See the latest [Azure DevOps REST API reference](#).

For information about .NET client libraries, see [.NET client libraries for Azure DevOps](#).

## Related articles

- [Azure Service Health overview](#)
- [Blog post: How do you measure quality of a service?](#)

# Data protection overview

3/6/2021 • 21 minutes to read • [Edit Online](#)

## Azure DevOps Services

Azure DevOps Services is a cloud-hosted application for your development projects, from planning through deployment. Based on the capabilities of Visual Studio Team Foundation Server, with additional cloud services, Azure DevOps manages your source code, work items, builds, and tests. It uses platform as a service (PaaS) infrastructure and many Azure services, including Azure SQL, to deliver a reliable, globally available service for your development projects.

This article discusses the steps that Microsoft takes to help keep your projects safe, available, secure, and private. Also, it describes the role you play in keeping your projects safe and secure.

This article is intended for organization administrators and IT professionals who manage their project assets daily. It will be most useful to individuals who are already familiar with Azure DevOps and want to know more about how Microsoft protects assets stored in Azure DevOps.

## Our commitment

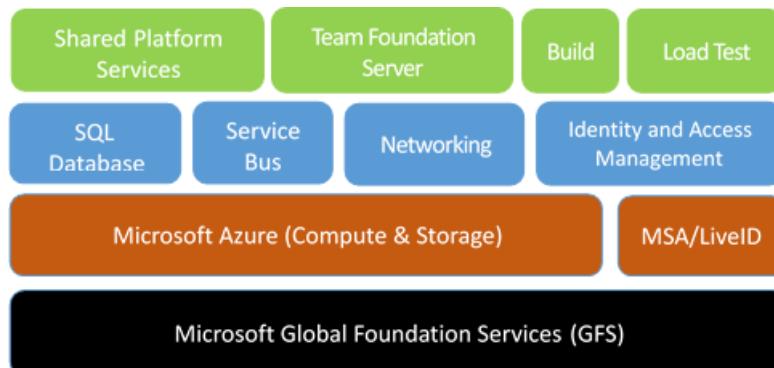
Microsoft helps to ensure that your projects remain safe and secure, without exception. When stored in Azure DevOps, your projects benefit from multiple layers of security and governance technologies, operational practices, and compliance policies. Microsoft enforces data privacy and integrity both at rest and in transit.

The threats you face boil down to four basic categories: data availability, service availability, service security, and data privacy. This article explores specific threats within each category, and explains what Azure DevOps does to address them. First, the article describes how data is stored and how Azure DevOps manages access to your data.

Because proper data protection also requires the active engagement of administrators and users, you need to know steps you should take to protect your project assets from unauthorized disclosure and tampering. You need to be explicit about granting permissions to user access points in order to have confidence that only the right people are accessing data within Azure DevOps.

Whatever your approach, you should consider all data potentially "at risk", no matter where it is or how it is being used. This is true for both data in the cloud as well as data stored in a private datacenter. Thus, it's important to classify your data, its sensitivity and risk, and the damage it might do if it's compromised. Also, categorize your data relative to an overall information security management policy.

## Built on Azure



Azure DevOps Services is hosted entirely in Azure datacenters and uses many of the core Azure services, including compute, storage, networking, Azure SQL, identity and access management, and Azure Service Bus.

Azure DevOps Services uses Azure Storage as the primary repository for service metadata and customer data. Depending on the type of data and the storage and retrieval needs, Azure DevOps Services uses Azure Blob Storage (for binary large objects) and Azure SQL data storage. To understand the Azure DevOps Services approach to data protection, some background on these elements is important.

- **Azure Blob Storage** stores large chunks of unstructured data. All projects use the Azure Blob Storage service. This data includes potentially sensitive or private information, such as the contents of source files and the attachments on work items. For most projects, the majority of storage in use is this type of unstructured blob storage. For more information, see [Azure Blob Storage](#).
- **Azure SQL Database storage** stores the structured and transactional aspects of your organization, including project metadata, the versioned source control history, and work item details. Database storage gives you fast access to the important elements of your project, and provides indexes into the blob storage to look up files and attachments. For more information, see [Azure SQL Database](#).

Administrators can manage access to resources by [granting or restricting permissions](#) on user identities or groups. Azure DevOps uses federated authentication of user identities via [Azure Active Directory](#) (Azure AD) and Microsoft accounts.

During authentication, the user is routed to the authentication provider, where they provide their credentials. After the authentication provider has verified the user's credentials, Azure DevOps issues an authentication cookie to the user, which allows the user to remain authenticated against Azure DevOps.

In this way, the user's credential information is never shared directly with Azure DevOps. For each Azure DevOps resource that the user attempts to access, permissions are validated based on the user's explicit permissions, as well as permissions inherited through group membership. Administrators can use access controls to protect [access to the organization](#), project collections, team projects, and team scoped data and functionality. Administrators can also protect more specific assets like version control folders and work item area paths.

## Data availability

Azure DevOps Services uses many of the Azure Storage features to ensure data availability in the case of hardware failure, service disruption, or region disaster. Additionally, the Azure DevOps team follows procedures to protect data from accidental or malicious deletion.

### Data redundancy

To protect data in the case of hardware or service failures, Azure Storage geo-replicates customer data between two regions in the same geography. For example, Azure can geo-replicate data between North and West Europe or between North and South United States.

For Azure Blob Storage, customer data is replicated three times within a single region, and is replicated asynchronously to a second region in the same geography. As such, Azure always maintains the equivalent of six copies of your data. This enables you to fail over to a separate region if there's a major outage or disaster, while also having local redundancy for hardware failures within a region. For Azure SQL Database storage, daily backups are maintained offsite if there's a regional disaster.

## **NOTE**

Regarding data redundancy and failover:

- There's an inherent delta, measured in minutes, when Microsoft replicates your data between the primary and secondary region.
- Failover to the secondary region is a decision that Microsoft must make centrally, as it affects all customers on the affected scale unit. Except in extreme circumstances, Microsoft opts to not fail over so that customer data isn't lost.
- Azure DevOps offers a 99.9 percent uptime SLA guarantee, and refunds a portion of the monthly charges if that SLA is missed in a specific month.
- Because there is only one region in Brazil, customer data in Brazil is replicated to the South Central US region for disaster recovery purposes.

## **Mistakes happen**

To protect against accidental deletion of data, Microsoft also takes point-in-time backups of both the blobs in Azure Blob Storage, and the databases in Azure SQL Database. There's a separate copy of all blobs, and changes are appended to each storage account. Because this data is immutable, there's no need to rewrite any existing storage as part of the backup procedures.

Backups are a standard part of Azure SQL Database, and Azure DevOps Services makes use of this. In both cases, these backups are also replicated in a paired region, helping to ensure that you recover from a regional outage.

A further protection is that Microsoft can recover entire organizations for up to 28 days after deletion. This is because Microsoft performs a "soft delete" for organization deletion operations.

## **Practice is critical**

Having multiple, redundant backups of your data is good but without practice, restoring can be unpredictable. It's been said that "backups never fail, it's the restores that do." While technically incorrect, the sentiment is right.

Microsoft regularly practices restoring various datasets from backup. Geo-redundant storage from Azure is tested regularly. Also, from time to time, Microsoft restores from backups to recover from human error, such as when a customer has inadvertently deleted a project in Azure DevOps. There are many permutations of disaster and data corruption scenarios, and Microsoft continues to plan and run new tests regularly.

## **Service availability**

Azure DevOps Services offers distributed denial-of-service (DDoS) protections and live site response to help ensure that you have access to your organization and associated assets.

### **DDoS protections**

In some cases, a malicious DDoS attack can affect service availability. Azure has a DDoS defense system that helps prevent attacks against our service. It uses standard detection and mitigation techniques such as SYN cookies, rate limiting, and connection limits. The system is designed to withstand attacks not only from the outside but also from within Azure.

For application-specific attacks that can penetrate the Azure defense systems, Azure DevOps establishes application and organization level quotas and throttling. This helps prevent any overuse of key service resources during an attack or accidental misuse of resources.

### **Live site response**

In rare circumstances, you might require a live site response to a problem with service availability. Microsoft has an operations team available 24x7, to rapidly identify the issue and to engage the necessary development team resources. Those resources then address the problem. They also aim to update the service status page within minutes of detecting an issue that affects the service. After the team has addressed an issue, they identify the

root cause of the issue and track the necessary changes to prevent similar issues in the future.

Azure DevOps live site management processes focus on your experience and the health of your service. These processes minimize the time to detect, respond to, and mitigate problems. All engineering disciplines are involved and responsible, so there are continual improvements evolving out of direct experience. This means that monitoring, diagnostics, resiliency, and quality assurance processes are improved over time.

Live site management in Azure DevOps has three distinct tracks: telemetry, incident management, and live site review. Here's what these tracks entail:

Telemetry	Incident management	Live-site review
<ul style="list-style-type: none"><li><b>Alerts</b> – define health alerts for failure modes</li><li><b>Diagnostics</b> – deliver instrumentation data and operational reports</li><li><b>Troubleshooting guides</b> – guidance for investigating an alert is defined by the feature, and then refined by the Service Engineer</li><li><b>Failure mode testing</b> – the Service Delivery (SD) team performs failure testing to ensure alerts fire as expected</li><li><b>Onboarding</b> – the feature team works with their Service Engineer (SE) to onboard new alerts to the 24 x 7 team</li></ul>	<ul style="list-style-type: none"><li><b>Detection</b> – product alerts detect health issues and start the Live Site Incident (LSI) process</li><li><b>Triage</b> – The 24 x 7 team receives all critical alerts and confirms impact using Azure DevOps guidance</li><li><b>Escalation</b> – both Dev and Ops have individuals in an on-call rotation. SE is the initial escalation path. The SE calls Dev, as needed</li><li><b>Incident management</b> – a bridge is managed by the SE who engages Dev, and Partners to troubleshoot</li><li><b>Resolution</b> – communication and service restoration are actively driven until customer impact is eliminated</li></ul>	<ul style="list-style-type: none"><li><b>Goal</b> – weekly review of LSI ensure that leadership has visibility into live site health and repeat issues</li><li><b>Cadence</b> – Incidents from prior week have root cause documented, then reviewed on weekly basis</li><li><b>Audience</b> – VS Leadership, Partner team when they drive impact. Developer attends to provide details on Service incident</li><li><b>Ownership</b> – Dev. Owns reviews for App and Deploy issues. SD owns for Platform issues.</li><li><b>Driving improvements</b> – Bugs and problem work items are logged for gaps (e.g. – missing alerts) and repeat root cause</li></ul>

The operations team also monitors the availability metrics for individual organizations. These metrics provide insights into specific conditions that might affect only some of our customers. Investigations into this data can often result in targeted improvements to address customer-specific issues. In some cases, Microsoft might even contact you directly to understand your experience and work with you to improve the service.

Microsoft publishes a service-level agreement (SLA) and provides a financial guarantee to ensure that we meet this agreement each month. For more information, see [SLA for Azure DevOps](#).

Sometimes partner teams or dependencies have incidents that affect Azure DevOps. All partner teams follow similar approaches to identifying, resolving, and learning from these service outages.

## Service security

Service security requires constant vigilance, from proper design and coding techniques to operational factors. Microsoft actively invests in the prevention of security holes and in breach detection. If there's a breach, Microsoft uses security response plans to minimize data leakage, loss, or corruption. For more information, see [About security, authentication, and authorization](#).

### Secure by design

Azure DevOps Services is designed to be secure. It makes use of the Microsoft Security Development Lifecycle at the core of its development process, and the Microsoft Operational Security Assurance program guides its cloud operation procedures. These methodologies specify the following requirements:

- Threat modeling during service design.

- Following design and code best practices.
- Verifying security with standard tooling and testing.
- Limiting access to operational and customer data.
- Gating rollout of new features through a rigid approval process.

The Azure DevOps Services team has annual training requirements for all engineers and operations personnel, and sponsors informal "brown bag" meetings hosted by Microsoft engineers. After they've solved an issue raised in a brown bag meeting, they share what they've learned with the rest of the team.

A cloud service is only as secure as the host platform. Azure DevOps uses PaaS for much of its infrastructure. PaaS automatically provides regular updates for known security vulnerabilities. VMs hosted in Azure use infrastructure as a service (IaaS), such as for a [hosted build service](#). Such images receive regular updates to include the latest security patches available from Windows Update. The same update rigor applies for on-premises machines, including those used for deployment, monitoring, and reporting.

The Azure DevOps Services team conducts regular, security-focused penetration testing of Azure DevOps. Using the same techniques and mechanisms as malicious attackers, penetration testing tries to exploit the live production services and infrastructure of Azure DevOps. The goal is to identify real-world vulnerabilities, configurations, errors, or other security gaps in a controlled process. The team reviews the results to identify other areas of improvement and to increase the quality of the preventative systems and training.

## Credential security

Your credentials in Azure DevOps are stored using industry best practices. Learn more about [credential storage](#).

## Reporting security issues

If during your penetration testing you believe you've discovered a potential security flaw related to the Azure DevOps service, report it to Microsoft within 24 hours. For more information, see [Report a computer security vulnerability](#).

### IMPORTANT

Although notifying Microsoft of penetration testing activities is no longer required, you must still comply with the [Microsoft Cloud Unified Penetration Testing Rules of Engagement](#).

## Bounty program

Azure DevOps participates in the [Microsoft Online Services Bounty Program](#). This program rewards security researchers who report issues to us, and encourages more people to help keep Azure DevOps secure. For more details, see the [Azure DevOps Bounty Program](#).

## Restricting access

Microsoft maintains strict control over who gets access to our production environment and customer data. Access is only granted at the level of least privilege required and only after proper justifications are provided and verified. If a team member needs access to resolve an urgent issue or deploy a configuration change, they must apply for "just-in-time" access to the production service. Access is revoked as soon as the situation is resolved.

Access requests and approvals are tracked and monitored in a separate system. All access to the system correlates against these approvals and if unapproved access is detected, an alert is raised for the operations team to investigate.

If the username and password for one of our developers or operation staff were stolen, data is still protected because we use two-factor authentication for all remote system access. This means that additional authentication checks via smart card or a phone call to a pre-approved number must take place before any remote access to the service is permitted.

In addition, Microsoft uses secrets to manage and maintain the service, such as RDP passwords, SSL certificates, and encryption keys. These are all managed, stored, and transmitted securely through the Azure portal. Any access to these secrets requires specific permission, which is logged and recorded in a secure manner. All secrets are rotated on a regular cadence, and can be rotated on-demand if there's a security event.

The Azure DevOps operations team uses hardened administrator workstations to manage the service. These machines run a minimal number of applications and operate in a logically segmented environment. Operations team members must provide specific credentials with two-factor authentication to access the workstations. All access is monitored and securely logged. To isolate the service from outside tampering, applications such as Outlook and Office, which are often targets of spear-phishing and other types of attacks, aren't permitted in this environment.

### **Intrusion protection and response**

To ensure data isn't intercepted or modified while in transit between you and Azure DevOps, we encrypt it via HTTPS and SSL.

Also, data we store on your behalf in Azure DevOps is encrypted as follows:

- For data stored in Azure SQL databases, Azure DevOps uses [Transparent Data Encryption \(TDE\)](#). This protects against the threat of malicious activity by doing real-time encryption of the database, associated backups, and transaction log files at rest.
- Azure Blob Storage connections are encrypted to protect your data in transit. To protect data at rest stored in Azure Blob Storage, Azure DevOps uses [Azure Storage Service Encryption \(SSE\)](#).

The Azure infrastructure helps the Azure DevOps Services team to log and monitor key aspects of the service. This helps ensure that activities within the service are legitimate, and detects breaches or attempted breaches. In addition, all deployment and administrator activities are securely logged, as is operator access to production storage. Real-time alerts are raised because the log information is automatically analyzed to uncover potentially malicious or unauthorized behavior.

Where a possible intrusion has been detected or high priority security vulnerability has been identified, the team has a clear security incident response plan. This plan outlines responsible parties, steps required to secure customer data, and how to engage with security experts at Microsoft. The team also notifies any organization owners if data is potentially disclosed or corrupted, so that they can take appropriate steps to remedy the situation.

Finally, to help combat emerging threats, Azure DevOps Services employs an "Assume Breach" strategy. A highly specialized group of security experts within Microsoft, known as the Red Team, assumes the role of sophisticated adversaries. This team tests breach detection and response, to accurately measure readiness and the impacts of real-world attacks. This strategy strengthens threat detection, response, and defense of the service. It also allows the team to validate and improve the effectiveness of the entire security program.

## **Data privacy**

You should have confidence that your data is being handled appropriately and for legitimate uses. Part of that assurance involves appropriately restricting usage so that your data is used only for legitimate reasons.

### **General Data Protection Regulation (GDPR)**

The General Data Protection Regulation (GDPR) is the biggest change in data protection laws in Europe since the 1995 introduction of the European Union (EU) Data Protection Directive 95/46/EC. To learn more about the GDPR regulation, see the [overview page in the Microsoft Trust Center](#).

### **Data residency and sovereignty**

Azure DevOps is available in the following eight geographies across the world: United States, Canada, Europe, United Kingdom, India, Australia, Asia Pacific, and Brazil. By default, your organization is assigned to your closest

geography, but you do have the option to choose a different geography. If you change your mind later, it's possible to migrate your organization to a different geography, with the assistance of Microsoft support.

Azure DevOps doesn't move or replicate customer data outside of the chosen geography. Instead, your data is geo-replicated to a second region within the same geography. The only exception is Brazil, which replicates data to the South Central US geography for disaster recovery purposes.

#### **NOTE**

For builds and releases running on Microsoft-provided macOS agents, your data will be transferred to a third-party data center in the US.

To learn more, see [Azure DevOps data location](#).

#### **Law enforcement access**

In some cases, third parties such as law enforcement entities might approach Microsoft to obtain access to customer data stored within Azure DevOps. Microsoft attempts to redirect the requests to the organization owner for resolution. When compelled by court order to disclose customer data to a third party, Microsoft makes a reasonable effort to notify the organization owner in advance, unless we are legally prohibited from doing so.

Some customers require their data storage in a particular geographic location to ensure a specific legal jurisdiction for any law enforcement activities. All customer data, such as source code, work items, test results, and geo-redundant mirrors and offsite backups, are maintained within the one of the geographies mentioned in the previous section.

#### **Microsoft access**

From time to time, Microsoft employees need to obtain access to customer data stored within Azure DevOps. As a precaution, all employees who have or might ever have access to customer data must pass a background check, which verifies previous employment and criminal convictions. In addition, we permit access to the production systems only when there's a live site incident or other approved maintenance activity, which is logged and monitored.

Because not all data within our system is treated the same, data is classified to distinguish between customer data (what you upload to Azure DevOps), organization data (information used when signing up for or administering your organization), and Microsoft data (information required for or collected through the operation of the service). Based on the classification, Microsoft controls usage scenarios, geo-location requirements, access restrictions, and retention requirements.

#### **Microsoft promotional use**

Microsoft occasionally wants to contact customers to let them know about additional features and services that might be useful. Because not all customers want to be contacted about these offers, you can opt in and opt out of marketing email communications.

Microsoft never uses customer data to target specific offers for specific users or organizations. Instead, we use organization data and aggregate usage statistics at the organization level to determine groups of organizations that should receive specific offers.

## **Building confidence**

In addition to these protections, you can be confident in other efforts Microsoft makes on behalf of Azure DevOps. These include internal adoption policies at Microsoft, the level of transparency provided into the state of our service, and progress towards receiving certification of our information security management systems.

#### **Internal adoption**

Teams across Microsoft are adopting Azure DevOps internally. The Azure DevOps team moved into an organization in 2014 and uses it extensively. More broadly, we have established guidelines to enable the adoption plans for other teams.

Obviously, large teams move more gradually than smaller ones, given their investments in existing DevOps systems. For teams able to move quickly, we have established a project classification approach. It assesses risk tolerance, based on project characteristics, to determine if the project is appropriate for Azure DevOps. For larger teams, the adoption typically occurs in phases, with more planning.

Additional requirements for internal projects include associating the organization with the Microsoft.com Azure Active Directory to ensure proper user identity life cycle and password complexity. For more sensitive projects, two-factor authentication is also required.

### Compliance certifications

Some of you want to understand third-party evaluation of our data security procedures. Azure DevOps has achieved the following certifications:

- ISO 27001:2013
- HIPAA (Health Insurance Portability and Accountability Act)
- BAA (Business Associate Agreement)
- EU Model Clauses
- SOC 1 Type 2
- SOC 2 Type 2

The SOC audit for Azure DevOps covers controls for data security, availability, processing integrity, and confidentiality. The SOC reports for Azure DevOps are available through the [Microsoft Service Trust Portal](#). You can also request a copy of these SOC reports.

## Steps you can take

Proper data protection requires your active engagement, as well as that of your administrators and users. Your project data stored within Azure DevOps is only as secure as the end-user access points. It's important to match the level of permission strictness and granularity for those organizations with the level of sensitivity of your project.

### Classify your data

The first step is to classify your data based on its sensitivity and risk horizon, and the damage that might occur if it's compromised. Many enterprises have existing classification methods that can be reused when projects move to Azure DevOps. For more information, you can download the "Data classification for cloud readiness" document from Microsoft Trustworthy Computing.

### Adopt Azure Active Directory

Another way to improve the security of your end users' credentials is to use Azure Active Directory (Azure AD) to manage your organization's access to Azure DevOps. Azure AD allows your IT department to manage its end-user access policy, including password complexity, password refreshes, and expiration if the user leaves your organization. Through Active Directory federation, you can directly link Azure AD to your organization's central directory, so you have only one location to manage these details for your enterprise.

The following table compares Microsoft account and Azure AD characteristics relative to Azure DevOps access:

PROPERTIES	MICROSOFT ACCOUNT	AZURE AD
Identity creator	User	Organization

Properties	Microsoft Account	Azure AD
Single username / password for all work assets	No	Yes
Password lifetime & complexity control	User	Organization
Azure DevOps membership limits	Any MSA	Organization's directory
Traceable identity	No	Yes
Organization & IP ownership	Unclear	Organization
2-factor authentication enrollment	User	Organization
Device-based conditional access	No	Organization

Learn more about [configuring this support for your organization](#).

### Require two-factor authentication

In some cases, you might want to restrict access to your organization by requiring more than one factor to sign in. You can require multiple factors with Azure AD. For example, you can require phone authentication, in addition to a username and password, for all authentication requests.

### Use BitLocker

For sensitive projects, you can use BitLocker on your Windows laptop or desktop computer. BitLocker encrypts the entire drive on which Windows and your data reside. When BitLocker is enabled, it automatically encrypts any file you save on that drive. If your laptop or desktop machine falls into the wrong hands, BitLocker prevents unauthorized access of local copies of data from your projects.

### Limit use of alternate authentication credentials

The default authentication mechanism for Git-related tooling is alternate authentication (sometimes referred to as basic authentication). This mechanism allows the end user to set up an alternate username and password for use during Git command-line operations. This username and password combination can also be used to access any other data for which that user has permissions. By its nature, alternate authentication credentials are less secure than the default federated authentication.

You can still make choices for increased security. For example, all communication is sent over HTTPS, and there are password complexity requirements. Nevertheless, your organization should evaluate if additional policies are required to meet your project security requirements. You can disable alternate authentication credentials altogether if you decide that it doesn't meet your organization's security requirements. For more information, see [Change application connection & security policies for your organization](#).

### Secure access to your organization

Azure AD provides the ability for administrators to control access to Azure resources and applications such as Azure DevOps. With conditional access control in place, Azure AD checks for the specific conditions you set for a user to access an application. After access requirements are met, the user is authenticated and can access the application.

Azure DevOps supports enforcing certain types of conditional access policies (for example, IP fencing) for custom Azure DevOps authentication mechanisms. These mechanisms include personal access tokens, alternate authentication, OAuth, and SSH keys. If your users are accessing Azure DevOps through a third-party client, only IP-based policies (IPv4 based only) are honored.

## Additional resources

- [Azure DevOps home page](#)
- [Azure DevOps data location](#)
- [Microsoft privacy statement](#)
- [Azure DevOps support](#)
- [What features and services do I get with Azure DevOps?](#)
- [Azure trust center](#)
- [Microsoft Security Development Lifecycle](#)
- [Revoke personal access tokens for organization users](#)

# Data locations for Azure DevOps

3/26/2021 • 2 minutes to read • [Edit Online](#)

## Azure DevOps Services

You can choose the location for your data during initial sign-up and creation of your organization. Azure DevOps operates in the following geographical locations ("geos").

## Data locations

Azure DevOps data is available in the following eight geographies across the world:

- Australia
- Brazil
- Canada
- Asia Pacific
- Europe
- India
- United Kingdom
- United States

We default your organization to your closest geography. However, you can choose a different geography. Later on, if you change your mind, you can [migrate your organization to a different geography](#).

## Customer data

Except as noted below, Azure DevOps maintains all customer data within your selected geography. Customer data includes the following data types:

- source code
- work items
- test results
- geo-redundant mirrors and offsite backups

Azure DevOps works with and uses many Microsoft Azure services. For details on customer data retention by location, see [Data residency in Azure](#).

## Profile data

Azure DevOps stores information that's global in nature, such as user identities and profile information as follows:

- EU-based users: profile data is in EU data center
- US-based users: profile data is in US data center
- Users from all other countries/regions: profile data is in US data center

## Transferring your data

Except as noted below, Microsoft doesn't transfer customer data outside of your selected geography.

If needed, you can transfer your data using preview, beta, or other pre-release services. These services typically store your data in the United States, but may store it globally.

**NOTE**

Microsoft will transfer your data if it needs to do any of the following actions:

- provide customer support
- troubleshoot the service
- comply with legal requirements

**NOTE**

Microsoft doesn't control or limit the geographies from which you or your users may access your data.

**NOTE**

Because there's only one region in Brazil, customer data is replicated to south-central United States for disaster recovery and load balancing purposes. For more information, see [Data residency in Azure](#).

**NOTE**

For builds and releases running on Microsoft-provided macOS agents, your data will be transferred to a third-party data center in the US.

These two data center locations are owned and managed by a third party with information security certification assurances, such as ISO 27001 and SOC 2 Type II report.

## Related articles

- [Get started with Azure DevOps](#)
- [Data protection overview](#)

# How we store your credentials for Azure DevOps Services

3/6/2021 • 2 minutes to read • [Edit Online](#)

## Azure DevOps Services

### IMPORTANT

Azure DevOps no longer supports Alternate Credentials authentication since the beginning of March 2, 2020. If you're still using Alternate Credentials, we strongly encourage you to switch to a more secure authentication method (for example, personal access tokens). [Learn more](#).

## Credential security

Microsoft is committed to ensuring that your projects remain safe and secure, without exception. In Azure DevOps, your projects benefit from multiple layers of security and governance technologies, operational practices, and compliance policies. We enforce data privacy and integrity both at rest and in transit. In addition, we adhere to the following practices with respect to the credentials or secrets that Azure DevOps stores. To learn more about how to choose the right authentication mechanism, see [Guidance for authentication](#).

## Personal access tokens (PATs)

- We store a hash of the PAT
- Raw PAT is generated in-memory on the server side as 32 bytes randomly generated through RNGCryptoServiceProvider then shared with the caller as a base-32-encoded string. This value is NOT stored
- PAT hash is generated in-memory on the server side as an *HMACSHA256Hash* of the raw PAT using a 64-byte symmetric signing key stored in our key vault
- Hash is stored in our database

## Secure shell (SSH) keys

- We store a hash of the enclosing organization ID and the SSH public key
- Raw public key is provided directly by the caller over SSL
- SSH hash is generated in-memory on the server side as an *HMACSHA256Hash* of the organization ID and raw public key using a 64-byte symmetric signing key stored in our key vault
- Hash is stored in our database

## OAuth credentials (JWTs)

- These are issued as fully self-describing JSON web tokens (JWTs) and are NOT stored in our service
- The claims in JWTs issued and presented to our service are validated using a certificate stored in our key vault

# Launch Visual Studio via Azure DevOps Services

4/21/2021 • 2 minutes to read • [Edit Online](#)

## Azure DevOps Services

When you first open [Visual Studio 2015](#), you can sign in and connect to [Azure DevOps Services](#).

If you're already signed in to Visual Studio or using Visual Studio 2017, [connect to Azure DevOps Services](#).

Once you're connected, you can store or share code in free, unlimited, private, cloud-based Git repositories or Team Foundation Version Control (TFVC). Organize and manage your work with Agile tools for DevOps, continuous integration, and continuous delivery. Your team can build often, test early, and ship faster.

To set up Visual Studio without Azure DevOps Services, learn how to [get started](#). To host your own server, learn how to [install and set up Azure DevOps Server](#).

Azure DevOps Services is free for [up to five users with access to Basic features](#) and for unlimited [Visual Studio subscribers](#) and [Stakeholders who can access limited features](#). Learn [what else you get with Azure DevOps Services](#). If you want, you can also use Azure DevOps Services with any IDE or code editor, like the following examples:

- [Eclipse, Android Studio, or IntelliJ](#)
- Xcode (see [Git](#) or [TFVC](#))
- [Visual Studio Code](#)

## How do I set up Visual Studio 2015 for Azure DevOps Services when I sign in?

1. [Download and install Visual Studio](#), if you don't have the version you want already. [Which versions can I use with Azure DevOps Services?](#)

If you have a Visual Studio subscription that includes the Visual Studio IDE, get the version that's available with your subscription.

2. Start Visual Studio, and then sign in to create your profile.

This profile saves your settings and roams with you when you sign in to Visual Studio on any computer.

[Why else should I sign in?](#) If you're a Visual Studio subscriber, use the sign in address for your subscription.



## Welcome. Sign in to Visual Studio.

Visual Studio will automatically keep you signed in, sync your settings between devices, and connect to online developer services.

[Learn more](#)

[Sign in](#)

[Not now, maybe later.](#)

[Can't sign in?](#)

3. Enter your sign in address, and then enter your password.
4. Add your Visual Studio profile details. You only need to add these details once.

The screenshot shows a profile setup form for Visual Studio. It includes fields for Full name (Jamal Hartnett), Contact e-mail (jamalhartnett@outlook.com), Phone number, and Country/Region (United States). The 'Full name' field has a question mark icon next to it.

Full name \* ?  
Jamal Hartnett

Contact e-mail \*  
jamalhartnett@outlook.com

Phone number

Country/Region \*  
United States

5. Give your organization a name, and confirm its location.

The screenshot shows a form for creating an organization. It asks for a URL (https://fabrikam) and a location (.visualstudio.com). It also states that the account will be hosted in the South Central US region. There is a 'Change options' link and a note about Microsoft's use of contact information. A 'Continue' button is at the bottom.

Create a [Visual Studio Team Services site](#) (optional)  
https://fabrikam .visualstudio.com

Your account will be hosted in the **South Central US** region.

[Change options](#)

Microsoft may use your contact information to provide updates and special offers about Visual Studio. You can unsubscribe at any time.

By clicking **Continue**, you agree to the [Terms of Service](#) and [Privacy Statement](#).

[Continue](#)

[How can I create an organization later or change its location?](#)

6. Create your first project to store your code, work items, backlog, builds, tests, and other assets. Name your project, select a process to organize your work, and choose the version control to manage your code.

 Visual Studio®

## Create your first team project

Welcome. Your account, <https://fabrikam.visualstudio.com/>, is created and ready to go. Now create your first team project where you'll host your code and backlog. [Learn more](#)

Project name: \*

Process template: \*

Version control: \*

 Git [?](#)     Team Foundation Version Control [?](#)

Create a README.md file to describe this project

[Not now, maybe later](#) [Create project](#)

Not sure which to choose? Learn which [process](#) and version control ([Git](#) or [TFVC](#)) work best for you.

7. If you're a new Visual Studio user, you can change your settings here, or change them later in Visual Studio options.

 Visual Studio®

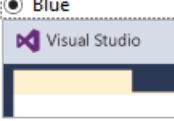
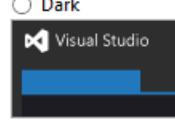
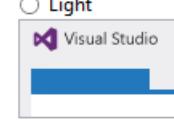
Hello, Jamal Hartnett

 [jamalhartnett@outlook.com](mailto:jamalhartnett@outlook.com)  
[View your Visual Studio profile](#)

Start with a familiar environment

Development Settings:

Choose your color theme

**Blue**     **Dark**     **Light** 

You can always change these settings later.

[Start Visual Studio](#)

These changes are saved with your profile, and your settings roam with you wherever you sign in.

8. To view your new organization, sign in to <https://dev.azure.com/{yourorganization}>.

## Next steps

[Add users to your organization](#)

## Related articles

- Add code to [Git](#) or [TFVC](#).
- [Create your backlog](#) to organize your work, manage your process, or [customize your process](#).

# About projects and scaling your organization

5/21/2021 • 11 minutes to read • [Edit Online](#)

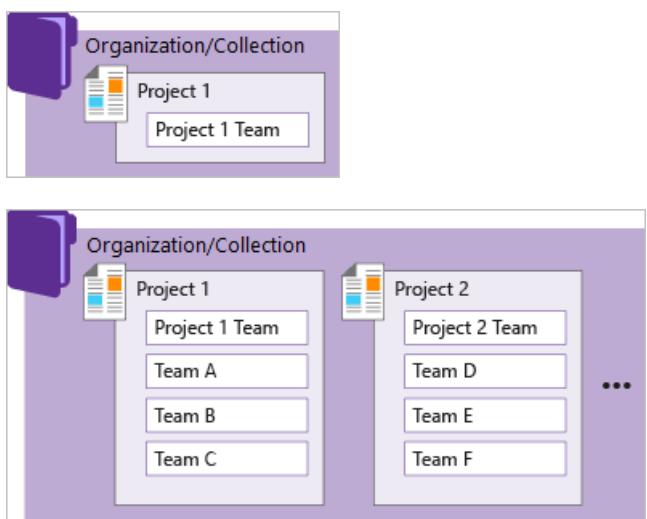
Azure DevOps Services | Azure DevOps Server 2020 | Azure DevOps Server 2019 | TFS 2018 - TFS 2013

A project provides a repository for source code and a place for users to plan, track progress, and collaborate on building software solutions. A project represents a fundamental container where data is stored when added to Azure DevOps.

When you create your project, a team of the same name is automatically created. This is sufficient for small teams. However, for enterprise-level organizations, it may be necessary to scale up, to create additional teams and projects. These additions can be created within the single account or collection.

**Single project and team defined within an organization or collection**

**Multiple projects and teams defined within an organization or collection**



The collection-project-team structure provides teams a high level of autonomy to configure their tools in ways that work for them. It also supports administrative tasks to occur at the appropriate level. As your organization grows, your tools can grow to support a [culture of team autonomy and organizational alignment](#).

## How do you manage work across the enterprise?

*How do you scale your DevOps and Agile tools to support your growing enterprise?*

When you connect to Azure DevOps, you connect to an organization or project collection. Within that container, one or more projects may be defined. At least one project must be created to use the system.

You can scale your organization in the following ways:

- To support different business units, you can add projects
- Within a project, you can add teams
- Add repositories and branches
- To support continuous integration and deployment, you can add agents, agent pools, and deployment pools

- To manage a large number of users, you can manage access through Azure Active Directory

You can scale your on-premises Azure DevOps deployment in the following ways:

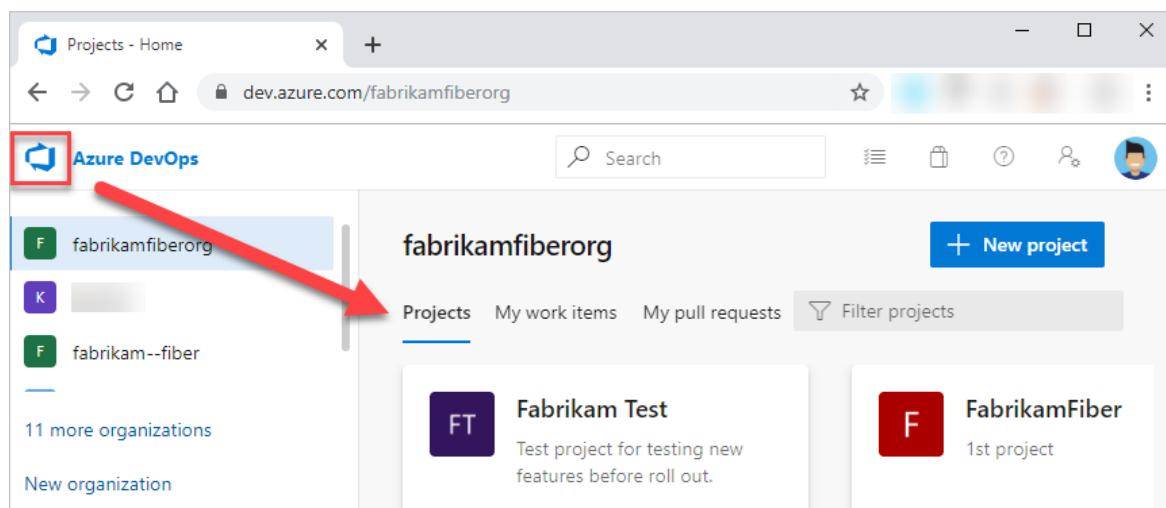
- To increase performance, you can add server instances
- To support different business units, you can add project collections and projects
- Within a project, you can add teams
- Add repositories and branches
- To support continuous integration and deployment, you can add agents, agent pools, and deployment pools
- To manage a large number of users, you can manage access through Active Directory

Azure DevOps Services and Azure DevOps Server are enterprise-ready platforms. These platforms support teams of any size, from tens to thousands. Azure DevOps Services, our cloud service, provides a scalable, reliable, and globally available hosted service. It's backed by a 99.9% SLA, monitored by our 24x7 operations team, and available in local data centers around the world.

## How to view projects

You can view the projects defined for your organization by opening the **Projects** page.

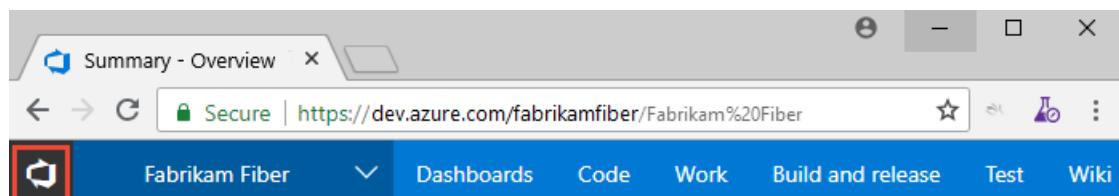
1. Select  **Azure DevOps** to open Projects.



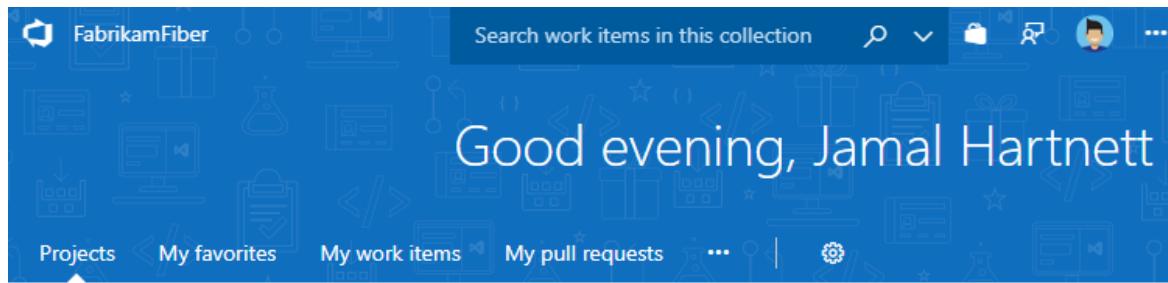
2. From there, you can choose a project from the set of projects listed.

To create or list projects, see [Create a project](#)

1. Select  **Azure DevOps** to open Projects.



2. From there, you can choose a project from the set of projects listed.



1. Choose the name of the server.

2. From there, you can choose a project from the set of projects listed.

## Limit user visibility for projects using the Project-Spaced Users group

By default, users added to an organization can view all organization and project information and settings.

The **Limit user visibility for projects** preview feature for the organization limits user access in two ways:

- Restricting views that display list of users, list of projects, billing details, usage data, and more that is accessed through **Organization Settings**.
- Limiting the set of people or groups that appear through people-picker search selections and the ability to @mention people.

### IMPORTANT

The limited visibility features described in this section apply only to interactions through the web portal. With the REST APIs or `azure devops` CLI commands, project members can access the restricted data.

### Limit access to Organization settings

To restrict select users, such as Stakeholders, Azure Active Directory guest users, or members of a particular security group, you can enable the **Limit user visibility for projects** preview feature for the organization. Once that is enabled, any user or group added to the **Project-Spaced Users** group, are restricted from accessing the **Organization Settings** pages, except for **Overview** and **Projects**; and are restricted to accessing only those projects to which they've been added to.

To enable this feature, see [Manage or enable features](#).

#### NOTE

All security groups are organization-level entities, even those groups that only have permissions to a specific project. From the web portal, users without access to a project can't see those groups which only have permissions to a specific project. However, you can discover the names of all groups in an organization using the **azure devops** CLI tool or our REST APIs. To learn more, see [Add and manage security groups](#).

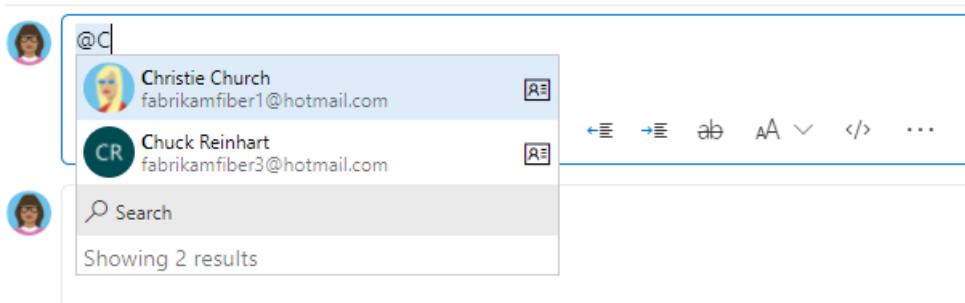
### Limit visibility within people pickers

For organizations that manage users and groups using Azure Active Directory (Azure AD), people pickers provide support for searching all users and groups added to Azure AD, not just those users and groups added to your project. people pickers support the following Azure DevOps functions:

- Selection of a user identity from a work tracking identity field such as **Assigned To**
- Selection of a user or group using **@mention** in a work item discussion or rich-text field, a pull request discussion, commit comments, or changeset or shelveset comments
- Selection of a user or group using **@mention** from a wiki page

As shown in the following image, you simply start typing into a people picker box until you find a match to a user name or security group.

#### Discussion



#### WARNING

When the **Limit user visibility for projects** preview feature is enabled for the organization, project-scoped users are unable to search for users who were added to the organization through Azure Active Directory group membership, rather than through an explicit user invitation. This is an unexpected behavior and a resolution is being worked on. To self-resolve this issue, disable the **Limit user visibility for projects** preview feature for the organization.

Users and groups who are added to the **Project-SScoped Users** group can only see and select users and groups in the project they are connected to from a people picker. To scope people pickers for all project members, see [Manage your project, Limit identity search and selection](#).

### Historical data remains visible

Identities that have been added to a comment, discussion, or assignment continue to be visible to all project members. For example, work items that were assigned to a user who has since left a project, the user's name on that work item remains visible to everyone in the project, even to users with the new restriction. The same is true for @mentions in PRs, comments, discussions, and more.

### When to add another project

In general, we recommend that you use a single project to support your organization or enterprise. A single project minimizes the maintenance of administrative tasks and supports the most optimized / full-flexibility **cross-link object** experience.

Even if you have many teams working on hundreds of different applications and software projects, you can most easily manage them within a single project. A project serves to isolate data stored within it. You can't easily move data from one project to another. When you move data from one project to another, you typically lose the history associated with that data.

For more information about when to add another project, see [How many projects do you need?](#).

### Reasons to add another project

You may want to add another project in following instances:

- To prohibit or manage access to the information contained within a project to select groups
- To support custom work tracking processes for specific business units within your organization
- To support entirely separate business units that have their own administrative policies and administrators
- To support testing customization activities or adding extensions before rolling out changes to the working project
- To support an Open Source Software (OSS) project

You may want to add another project in following instances:

- To prohibit or manage access to the information contained within a project
- To support custom work tracking processes for specific business units within your organization
- To support entirely separate business units that have their own administrative policies and administrators
- To support testing customization activities or adding extensions before rolling out changes to the working project

## Private and public projects

You can add public and private projects to your organization. You can also [change the visibility of a project from private to public](#).

Private projects require that you add and manage user access. Users must sign in to gain access to a project, even if it's read-only access. All users added to a project have access to the project and organization information. For details, see [Resources granted to project members](#).

A public project, doesn't require users to sign in to gain read-only access to many of the services. Public projects provide support to share code with others and to support continuous integration/continuous deployment (CI/CD) of open-source software. To learn more about public projects, see [What is a public project?](#).

## Structure your project

When you add a project, look at using the following elements to structure it to support your business needs:

- [Create a Git repository](#) for each subproject or application, or [create root folders within a TFVC repository](#) for each subproject. If you're using TFVC and heading toward a combined project model, create root folders for different teams and projects, just as you would create separate repos in Git. Folders can be secured as needed and workspace mappings can control what segments of the repo you're actively using.
- [Define area paths](#) to support different subprojects, products, features, or teams.
- [Define iteration paths \(also known as sprints\)](#) that can be shared across teams.
- [Add a team](#) for each product team that develops a set of features for a product. Each team you create automatically creates a security group for that team, which you can use to manage permissions for a team. See also, [Portfolio management](#).
- [Grant or restrict access to select features and functions](#) using custom security groups.
- [Create query folders](#) to organize queries for teams or product areas into folders.

- [Define or modify notifications](#) set at the project level.

## Customizing and configuring projects

You can configure and customize most services and applications to support your business needs or the way your teams work. Within each project, you can do the following tasks. For a comprehensive view of what resources can be configured, see [About team, project, and organizational-level settings](#).

- **Dashboards:** Each team can [configure their set of dashboards](#) to share information and monitor their progress.
- **Source control:** For each [Git repository](#), you can apply branch policies and define branch permissions. For TFVC repositories, you can [set check-in policies](#).
- **Work tracking:** You can add fields, change the workflow, add custom rules, and add custom pages to the work item form of most work item types. You can also add custom work item types. For details, see [Customize an inheritance process](#).
- **Azure Pipelines:** You can fully customize your build and release pipelines, define build steps, release environments, and deployment schedule. For details, see [Build and Release](#).
- **Azure Test Plans:** You can define and configure test plans, test suites, test cases, and test environments. You can also add test steps within your build pipelines. For details, see [Exploratory & Manual Testing](#) and [continuous testing for your builds](#).
- **Dashboards:** Each team can [configure their set of dashboards](#) to share information and monitor their progress.
- **Source control:** For each [Git repository](#), you can apply branch policies and define branch permissions. For TFVC repositories, you can [set check-in policies](#).
- **Work tracking:** You can add fields, change the workflow, add custom rules, and add custom pages to the work item form of most work item types. You can also add custom work item types. For details, see [Customize the On-premises XML process model](#).
- **Build and Release:** You can fully customize your build and release pipelines, define build steps, release environments, and deployment schedule. For details, see [Build and Release](#).
- **Test:** You can define and configure test plans, test suites, test cases, and test environments. You can also add test steps within your build pipelines. For details, see [Exploratory & Manual Testing](#) and [continuous testing for your builds](#).

## When to add a team, scaling Agile tools across the enterprise

As your organization grows, add teams to provide them the Agile tools that each team can configure to meet their workflow. To learn more, see the following articles.

- [Scale Agile to large teams](#)
- [About teams and Agile tools](#)
- Manage a [portfolio of backlogs](#) and gain insight into each team's progress and the progress of all programs.
- Use [Delivery plans](#) to review the schedule of stories or features your teams plan to deliver. Delivery plans show the scheduled work items by sprint (iteration path) of selected teams against a calendar view.
- Incrementally adopt [practices that scale](#) to create greater rhythm and flow within your organization, engage customers, improve project visibility, and develop a productive workforce.
- Structure projects to gain [visibility across teams](#) or to support [epics, release trains, and multiple backlogs](#) to [support the Scaled Agile Framework](#).

To review stories and short videos on how Microsoft transitioned from waterfall to Agile, see Scaling Agile Across the Enterprise.

# Clients that support connection to a project

In addition to connecting through a web browser, you can connect to a project from the following clients:

- [Visual Studio \(Professional, Enterprise, Test Professional\)](#)
  - [Visual Studio Code](#)
  - [Visual Studio Community](#)
  - [Eclipse: Team Explorer Everywhere](#)
  - [Office Excel](#)
  - [Azure Test Plans](#) (formerly Test Manager)
  - [Microsoft Feedback Client](#)
- 
- [Visual Studio \(Professional, Enterprise, Test Professional\)](#)
  - [Visual Studio Code](#)
  - [Visual Studio Community](#)
  - [Eclipse: Team Explorer Everywhere](#)
  - [Office Excel](#)
  - [Office Project](#)
  - [PowerPoint Storyboarding](#)
  - [Azure Test Plans](#) (formerly Test Manager)
  - [Microsoft Feedback Client](#)

See also, [Compatibility with Azure DevOps Server versions](#).

## Q & A

### **Q: Can I move or transfer a project to another organization or collection?**

A: Not without losing data. You can't move a project from one collection/organization to another collection/organization without losing data. You can manually copy resources and leave some behind, or use a third-party tool, such as [OpsHub Visual Studio Migration Utility](#), that copies data using the REST APIs.

### **Q: What programmatic tools support projects?**

A. See [Projects REST API](#).

Also, you can use the `az devops project` commands.

## Related articles

- [Get started as an administrator](#)
- [Web portal navigation](#)
- [What do I get with a project?](#)
- [Understand differences between Azure DevOps](#)