

A quick introduction to Azure DevOps

Kevin Chant



A quick introduction to Azure DevOps

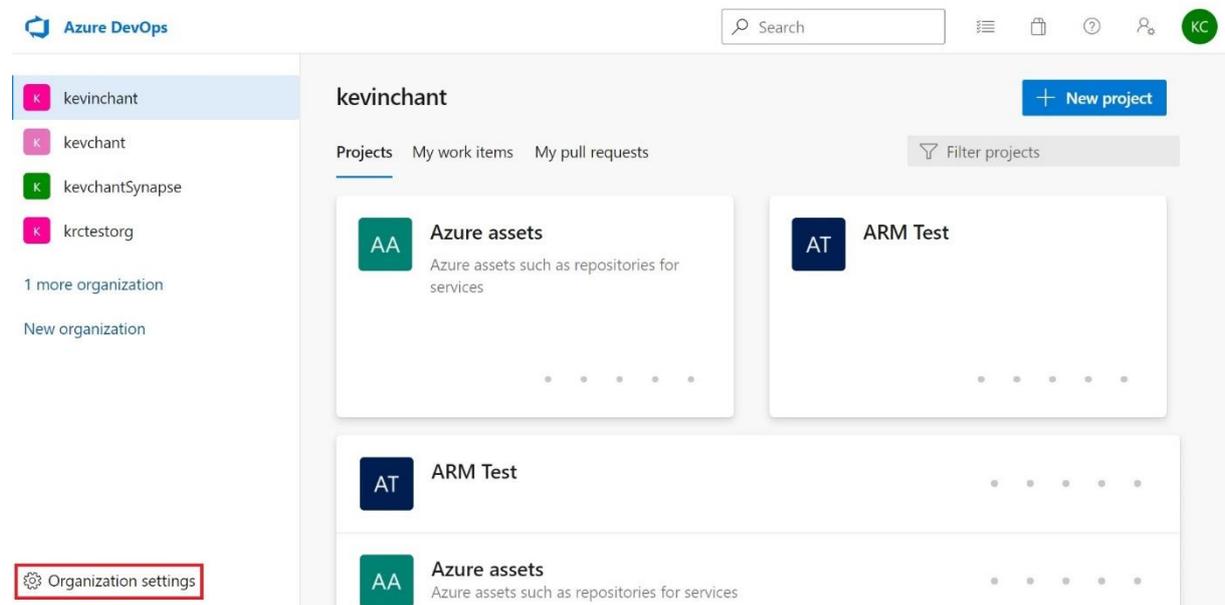
In this document, I want to do an introduction to [Azure DevOps](#). Because I have had to do this a lot over the last couple of years.

Azure DevOps is a service that you can use to do your application lifecycle management. In reality, there are two versions of Azure DevOps. Azure DevOps server which you install on your own server(s) and Azure DevOps cloud services.

We will focus on Azure DevOps cloud services in this post. However, most of it applies to Azure DevOps server as well.

Structure

Azure DevOps has an interesting structure when you first access it.



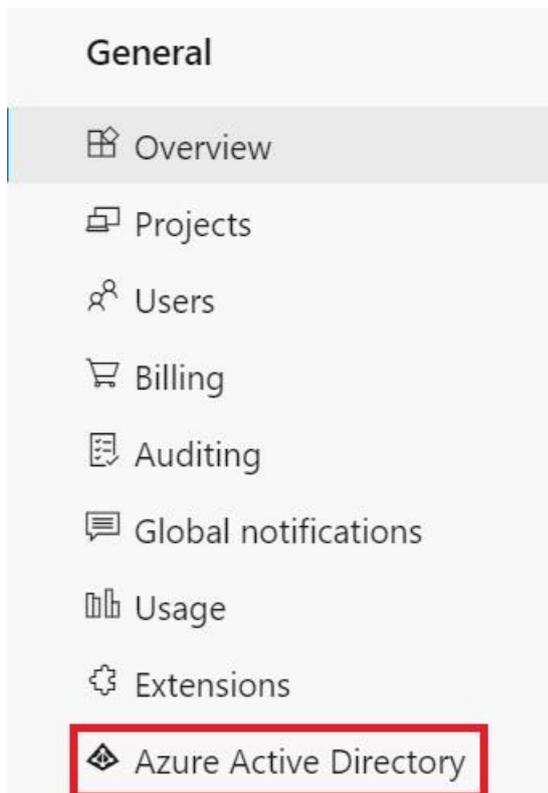
At the top you have the Azure DevOps icon, which represents your account. Below that on the left is a list of your organizations your account is a part of.

Now, these organizations can be a company you work for, if you work for a large enterprise it can be a division within it. Of course, it can also be one you have created for testing purposes as well which I covered in another post about [creating your own Azure DevOps organization](#). So that you can test things without worrying about breaking anything in your work environment.

If you have the right permissions, you can change various settings at the organization level. By clicking the icon highlighted above. I will quickly cover some interesting settings here.

Organization settings

In the General section you can change various things at the organization level. Including whether or not your organization is connected to an Azure Active Directory. Which is a requirement if you want to use CI/CD with Azure Synapse Analytics.



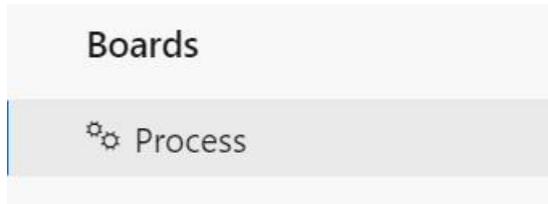
In the security section you can set various policies and permissions at the organization level.



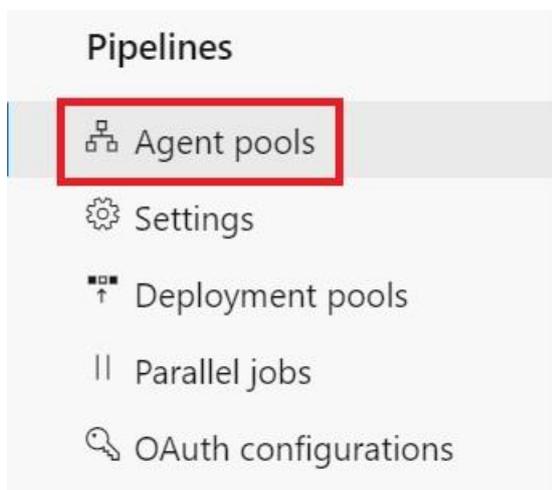
Within the Boards section you can change the process settings for each of the projects. You can think of these processes as a template that each Azure DevOps project uses. Azure DevOps provides processes based on Basic, Agile, Scrum or CMMI which can all be customized to suit your needs. Each project can only use one process.

One piece of advice I will give relating to processes it to try and choose the right one at the start. For example, if looking to use Agile and user stories select the Agile process or

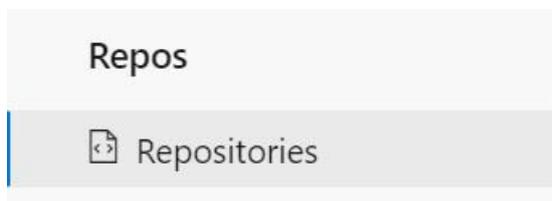
copy it so that you can customize one to suit your needs. Otherwise, it can cause issues further down the line.



Azure DevOps uses what it calls agents to do all the work specified in pipelines. Like building artifacts and deploying them to servers. You can manage them at an organization level so that they can be used in any project in Azure DevOps.



Currently you can change a couple of settings for your repositories at an organization level. With the most interesting one being the ability to change the default branch name for branches.



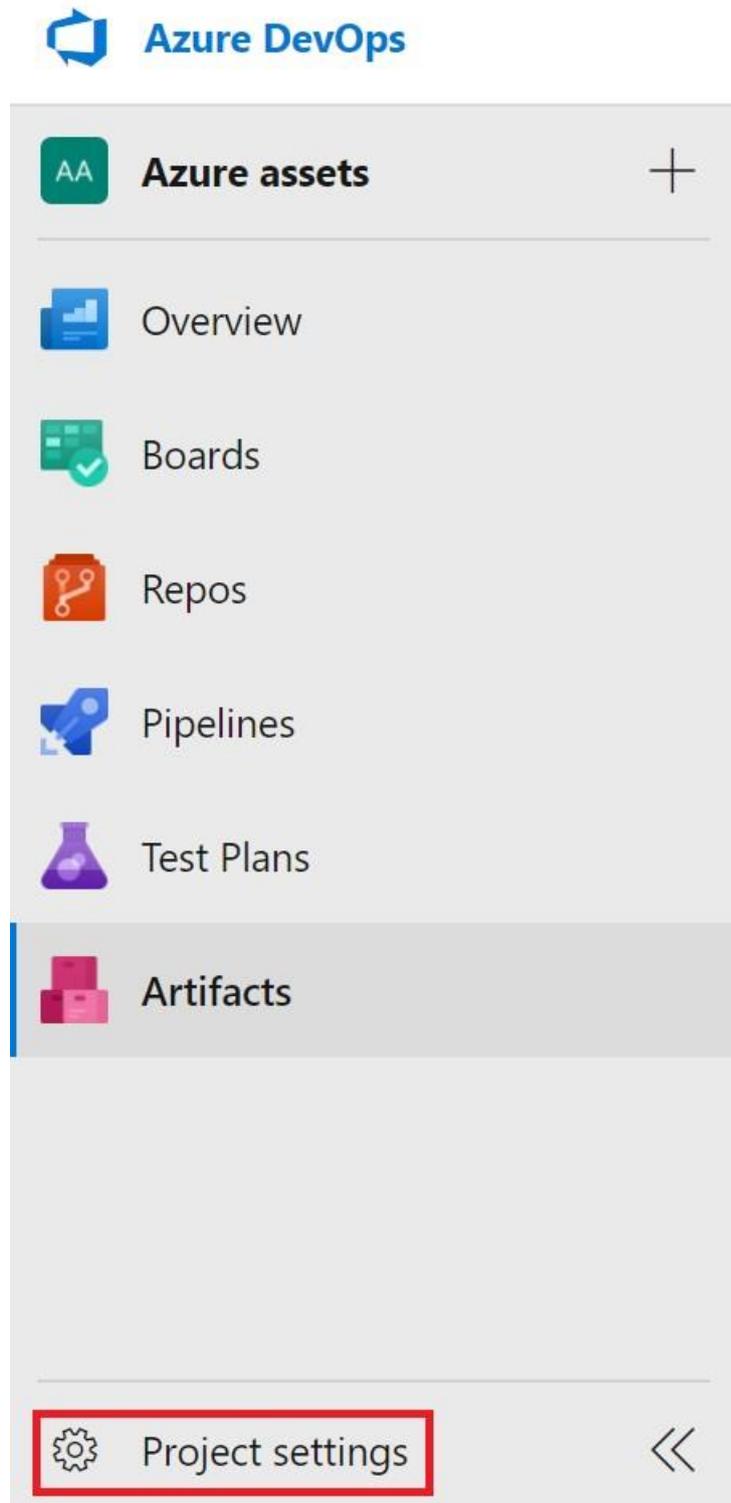
Finally, the artifacts section you can view the total storage used by all the artifacts in your organization.



Azure DevOps Projects

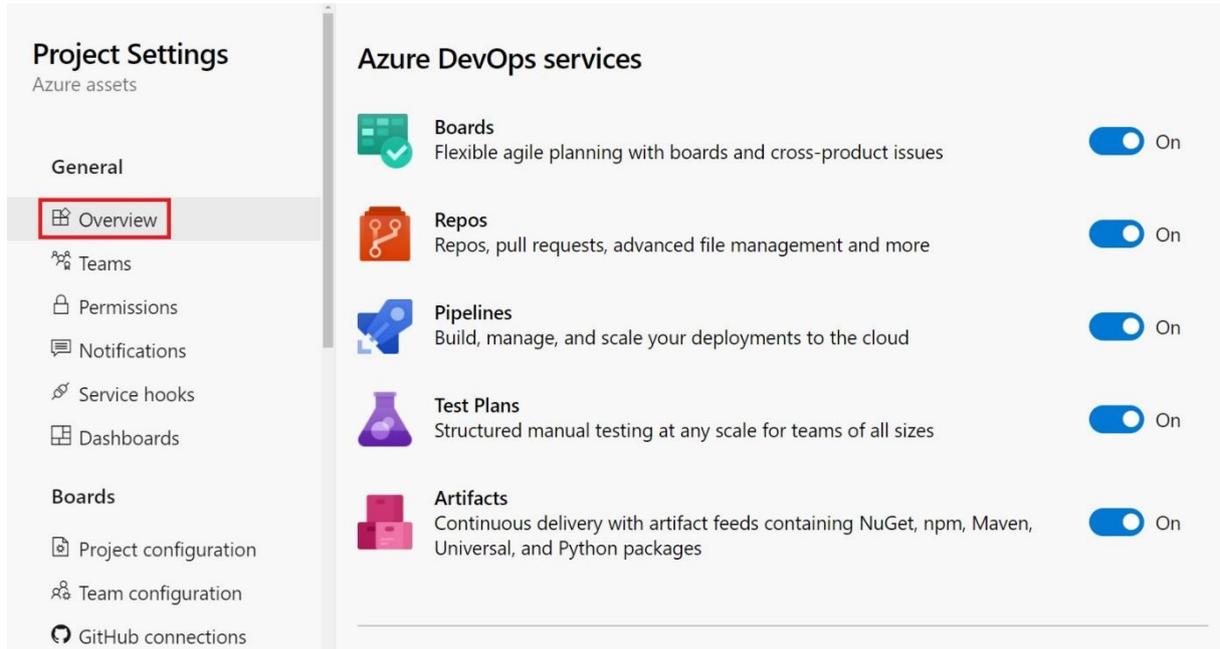
Each organization in Azure DevOps can contain one or more projects. How you define a project is entirely up to you. For example, it can be work that a team does together, or multiple teams or of course an actual project that is taking place.

When you first go into a project you usually see something that looks like the below on the left-hand side. Which are the default services you can use in your project.



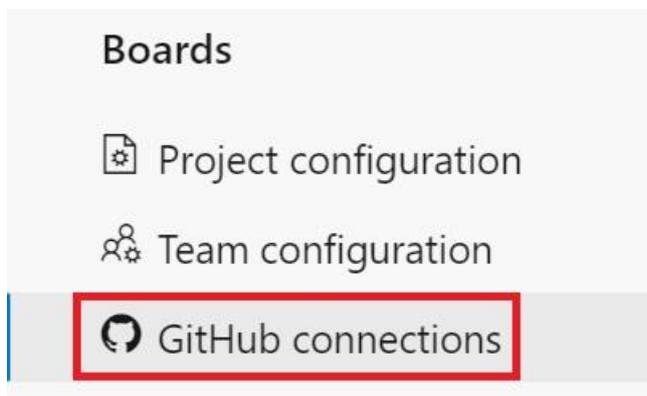
Plus, you can change various project settings if you have the right permissions. Just like with organization settings I will only cover some interesting ones here. My advice is to experiment with them.

In the General settings you can change various settings at the project level. One thing a lot of people do not realize is that you can turn the default services used for a project on or off.



Top tip though, if you turn off Azure Boards the Azure Test Plans will be turned off as well. Because they are connected.

You manage your boards in your project in the Boards section. You can also link GitHub repositories here, which allows you to add links to from GitHub to your work items.



Just like in organization settings you can change pipeline settings at the project level. For example, you can add connections to service connections to external services like your Azure and Docker accounts. So that you can use them during your CI/CD deployments.

Pipelines

 Agent pools

 Parallel jobs

 Settings

 Test management

 Release retention

 Service connections

 XAML build services

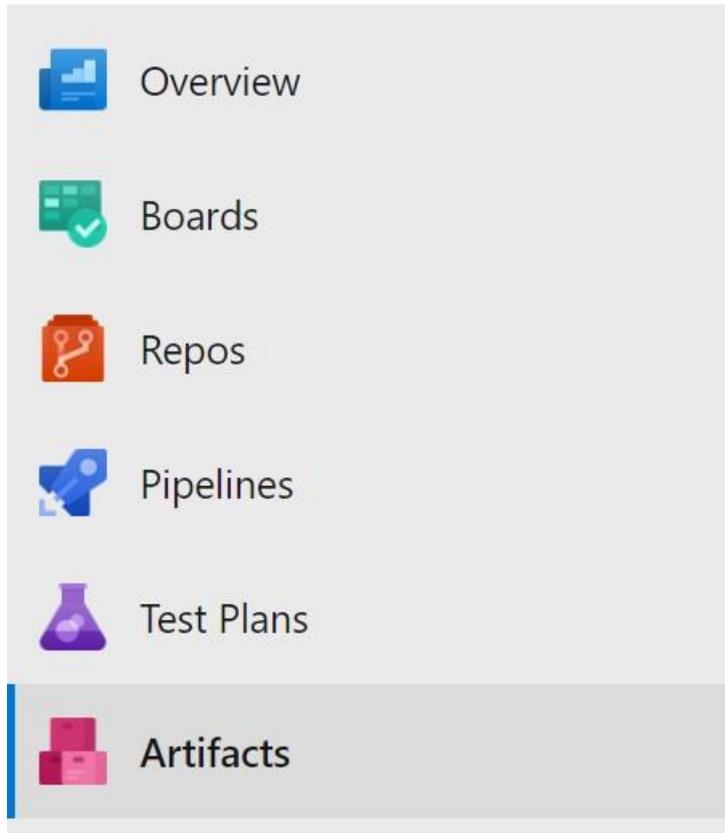
In the Repos section you can manage your existing repositories. In addition, you get the option to change the default branch name for new repositories again and set some interesting policies for your repositories.

Repos

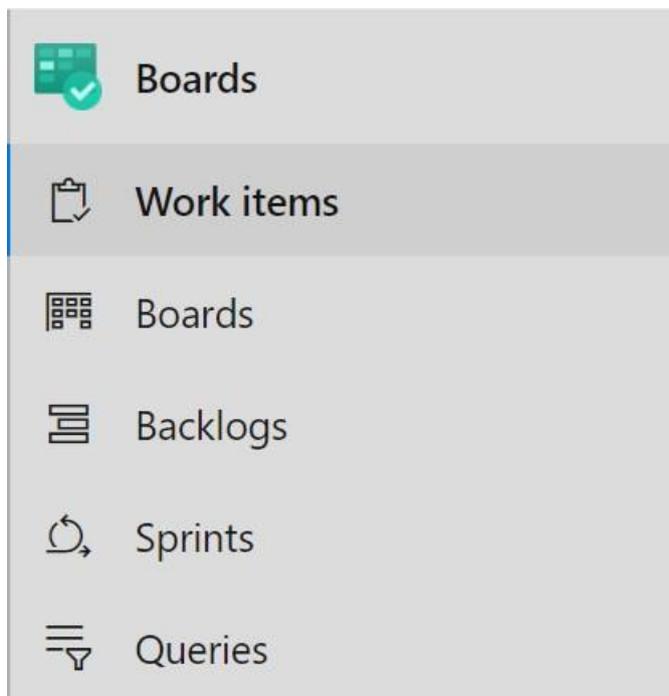
 Repositories

[Quick intro to the default services](#)

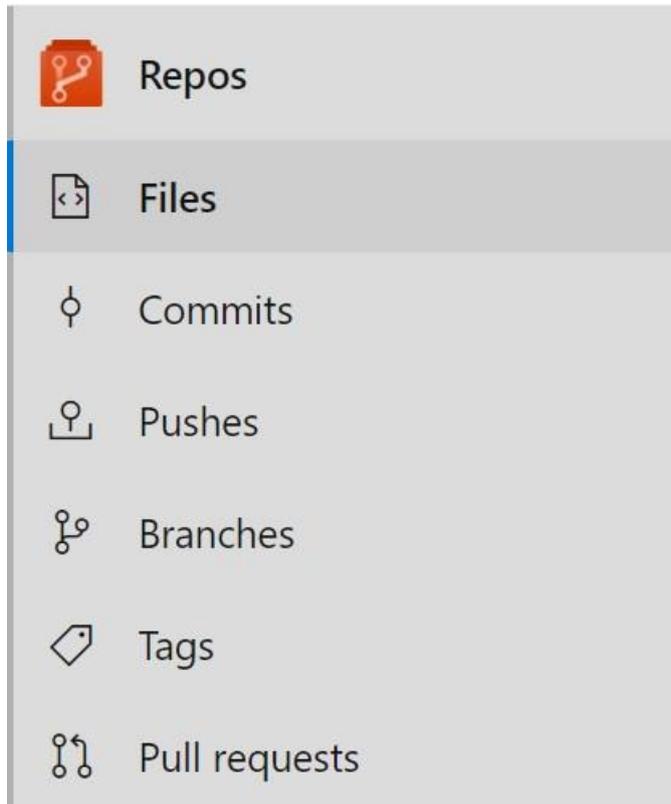
You usually see the below default services in Azure DevOps:



[Azure Boards](#) is where you manage your work items.

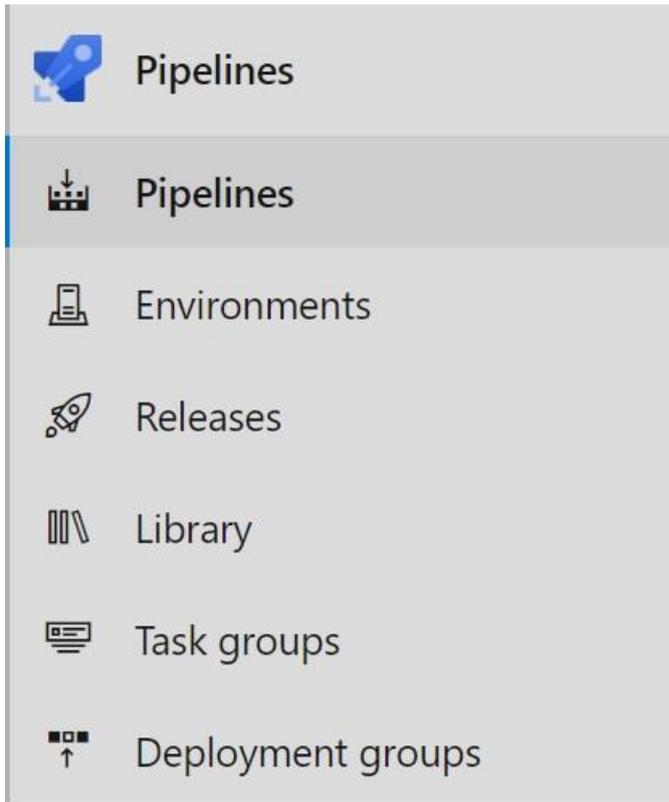


[Azure Repos](#) is where you can manage your Git and TFVC repositories.

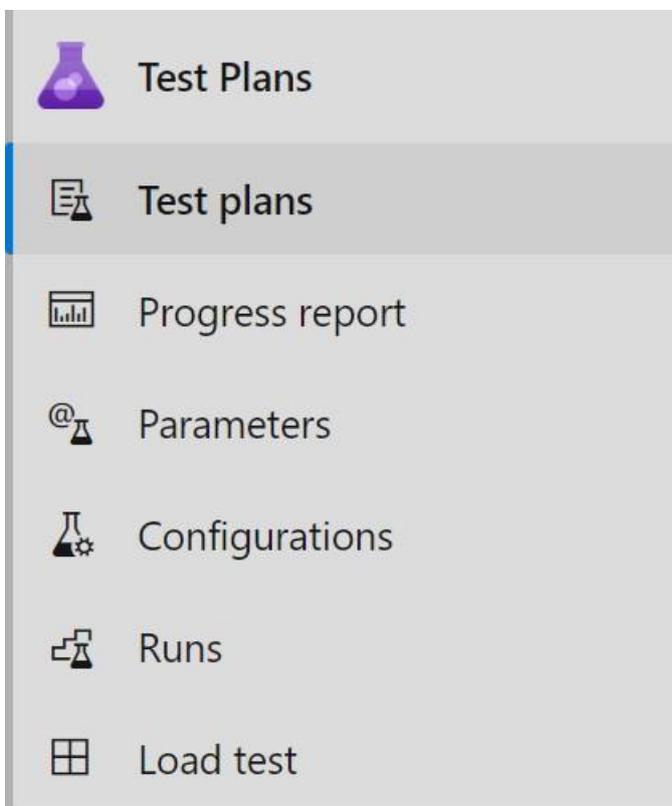


[Azure Pipelines](#) is where you manage your CI/CD pipelines. I recommend using YAML in the Pipelines feature for your CI/CD deployments. I know some demos show the Classic Editor and Releases working together but YAML is the way forwards.

Just to help raise awareness, the Environments feature is constantly evolving. Look to use it with your YAML pipelines for deployments to various environments to avoid pipeline sprawl.



You manage your manual test plans in [Azure Test Plans](#). I highly recommend using this, especially if you are using Azure Boards since they are tied together. In addition, using this instead of spreadsheets removes the problems with spreadsheet sprawl.



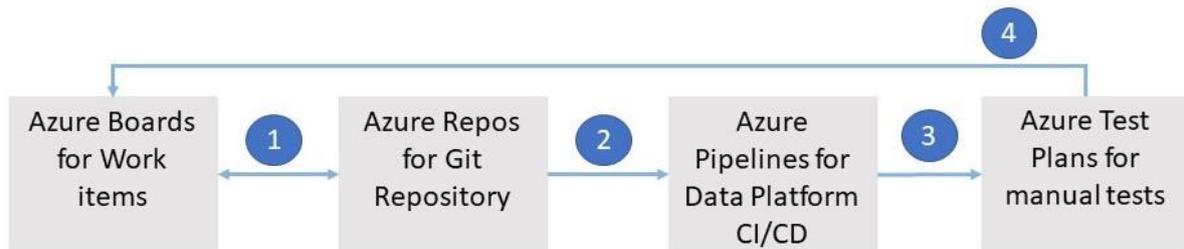
[Azure Artifacts](#) is the service which allows you to share packages in various formats. Including an interesting one called Universal Packages which is designed for large binaries.

However, I have also used this format for PowerShell modules.



Work together

To give you a better idea of how these services can work together for CI/CD deployments here's a diagram I created for a post about [using Azure Test Plans for Data Platform deployments](#).



Azure Artifacts is missing from the above diagram, but you can use it in conjunction with Azure Pipelines.

Additional services and features

You can install additional features and services in Azure DevOps by installing extensions through the marketplace. In fact, you can probably see some of these yourself if you log into an organization for an enterprise because they probably have some installed already. For example, SonarQube and Retrospectives.

Top tip, you might want to avoid installing the Delivery Plans extension because Delivery Plans 2.0 is now integrated within Azure DevOps. Here's the announcement about the fact that [Delivery Plans 2.0 is now GA](#).

Additional resources

I thought I would share some additional resources for those who want to learn about Azure DevOps.

For those who like videos, in one video Donavan Brown from Microsoft does a [five-minute introduction to Azure DevOps](#). In addition, you can watch him present a [one hour introduction to Azure DevOps](#).

If you want to watch more video, Pluralsight has a whole [path for the AZ-400 exam](#).

For those who prefer hands-on experience, Microsoft learn has various [Azure DevOps paths](#).

In addition, you can read Microsoft's official [Azure DevOps documentation](#).

About Sogeti

Sogeti is a leading provider of technology and engineering services. Sogeti delivers solutions that enable digital transformation and offers cutting-edge expertise in Cloud, Cybersecurity, Digital Manufacturing, Digital Assurance & Testing, and emerging technologies. Sogeti combines agility and speed of implementation with strong technology supplier partnerships, world class methodologies and its global delivery model, Rightshore®. Sogeti brings together more than 25,000 professionals in 15 countries, based in over 100 locations in Europe, USA and India. Sogeti is a wholly-owned subsidiary of Capgemini SE, listed on the Paris Stock Exchange.

Learn more about us at
www.sogeti.com