



Azure Pipelines



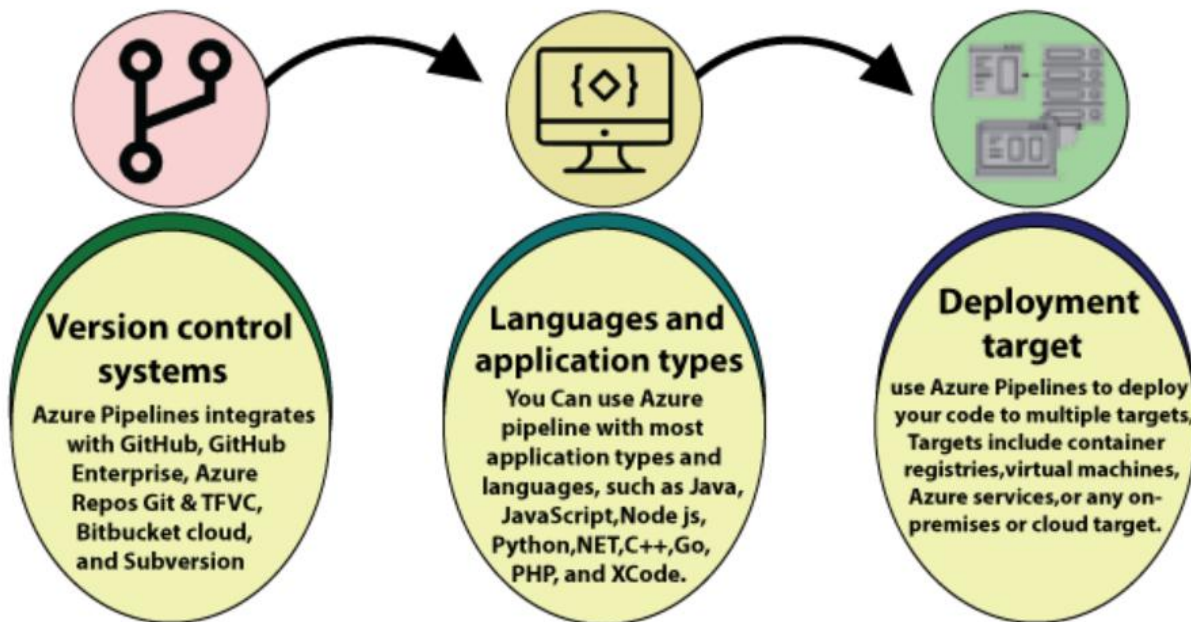
TechData-Infinity-Devops with MultiCloud



8. Azure Pipelines

Pipelines are the [CI/CD](#) tool that facilitates automated building, [testing](#), and deployment. Azure Pipelines supports any programming language or platform which enables users to create pipelines that support Windows, Linux, and macOS using cloud-hosted agents.

Azure Pipelines automatically builds and tests code projects to make them available to others. It works with just about any language or project type. Azure Pipelines combines continuous integration (CI) and continuous delivery (CD) to test and build your code and ship it to any target. The Azure pipeline has a lot of capabilities such as continuous integration and continuous delivery to regularly and consistently test and builds our code and ship to any target. There are three key distinct advantages of using Azure DevOps pipelines.



Version control system: Azure Pipelines integrates with GitHub, GitHub Enterprise, Azure Repos Git & TFVC, Bitbucket Cloud, and Subversion.

Language and application types: We can use Azure Pipeline with most application types and languages, such as Java, JavaScript, Node.js, Python, .Net, C++, Go, PHP, and Xcode.

Deployment target: We can use Azure Pipelines to deploy our code to multiple targets. Targets include – container registries, virtual machines, Azure services, or any on-premises or cloud target.

Azure DevOps Pipeline concepts

1. **Pipeline:** It is a workflow that defines how our test, build, and deployment steps are run.
2. **Stage:** It is a logical boundary in the pipeline. It can be used to mark the separation of concerns. Each stage contains one or more jobs.

TechData-Infinity-Devops with MultiCloud



3. **Job:** A stage can contain one or more jobs. Each job runs on an agent. It represents an execution boundary of a set of steps.
4. **Step:** It is the smallest building block of a pipeline. It can either be a script or a task. A task is simply an already created script offered as a convenience to you.
5. **Agent and Agent pools:** An agent is an installable software that runs one job at a time. Instead of managing each agent individually, you organize agents into agent pools.
6. **Artifact:** It is a collection of files or packages published by a run. The Artifact is made available to subsequent tasks, such as distribution or deployment.
7. **Trigger:** It is something that is set up to tell the pipeline when to run. We can configure a pipeline to run upon a push to the repository, at scheduled times, etc.
8. **Environment:** It is a collection of resources, where you deploy your application. It contains one or more virtual machines, containers, web apps, etc.
9. **Checks:** Checks define a set of validations required before a deployment can be performed.
10. **Runs:** It represents a single execution of a pipeline and collects the logs associated with running the steps and the results of running tests.

These pipelines are easily extensible through the extensions available in the marketplace. Besides, they support advanced workflows that can be used to facilitate:

- Multi-phase builds
- Test integrations
- Custom reporting functions

On top of that, Azure Pipelines provide native container support, enabling them to push containers to container registries from the pipeline directly. The pipelines offer flexibility to deploy to multiple environments from Kubernetes clusters to serverless functions and even deploy to other cloud providers such as AWS or GCP.