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

Azure DevOps is a set of services that gives you the tools you need to do just that.

With Azure DevOps, you can build, test, and deploy any application, either to the cloud or on premises.

**On Premises :**

On-premise is a more traditional IT (Information Technology) infrastructure method where the physical hardware, software, and data are stored on-site.

On-premise consists of an IT infrastructure comprising systems, hardware applications, and software applications.

With an on-premise system, you get complete ownership of your servers, and you can personally oversee the maintenance of your systems.

If a business chooses to adopt an on-premise system, it will be responsible for installing in-house servers and related hardware. The company will also have to employ IT staff to continually maintain the infrastructure to prevent malfunctions in the system that can lead to data breaches and data losses.

on-premise software requires a license per server, after which the servers are completely the company's responsibility, including [management](https://www.knowledgehut.com/blog/others/what-is-management) and security.

On-premise systems are ideal if your business requires customizable hardware or systems that can be built to match your purposes. However, there is less scope for scalability with on-premise systems. With on-premise, you do not have to rely on internet access or connectivity to access your required software, and as a result, there is less scope for wastage of time.

**On**[**Cloud**](https://www.geeksforgeeks.org/cloud-computing/)**:** Cloud refers to the delivery of on-demand computing services over the internet on “Pay As U Use “services, in simple words rather than managing files and Services on the local storage device you can do the same over the Internet in a cost-efficient manner.

With a Cloud-based enrolment model, there is no convincing motivation to purchase any additional establishment or licenses.

|  | **On Premises** | **On Cloud** |
| --- | --- | --- |
| **1.** | Control of user is more. | Control of user is less as third parties are involved. |
| **2.** | Infrastructure is not easy to scale. | Infrastructure is easy to scale. |
| **3.** | Internet connectivity is not need all the time. | Internet is must for the services of the cloud. |
| **4.** | These services run within the enterprise only. | The services of cloud depends on the third parties so these are not only accessed within the enterprise. |
| **5.** | These services are not quite flexible. | The services of cloud are highly flexible. |
| **6.** | Not available on a subscription basis. | Services are available for purchase. |
| **7.** | For hardware and software updates, enterprise is responsible. | For hardware and software updates, third party is responsible. |
| **8.** | Cost is fixed. | Cost is not fixed, as additional services comes with additional charges. |
| **9.** | Data is easily portable. | Data is not easily portable. |
| **10.** | The deployment happens in the local environment. | The deployment happens on the internet. |
| **11.** | Security is more. | Security is less as all the information is stored in the cloud. |
| **12.** | These services are used in large companies. | These services are used in small and mid sized companies. |
| **13.** | Implementation time is more. | Implementation time is less. |

DevOps practices that enable transparency, cooperation, continuous delivery and continuous deployment become embedded in your software development lifecycle.

**What DevOps is not**

When considering what DevOps is, it's also important to make sure we learn what it's *not*. DevOps is not:

* A methodology
* A specific piece of software
* A quick fix for an organization's challenges
* Just a team or a job title (although these titles are reasonably common in the industry)

Azure DevOps provides several tools you can use for better team collaboration. It also has tools for automated build processes, testing, version control, and package management.

These tools are great whether you're in the cloud or on-premises. It also doesn't matter if we're deploying to Linux or Windows or another platform.

Azure DevOps is a suite of services that provide a solution for anyone who wants an enterprise-grade tool chain.

How can u provide value to customer

* **Deploy more frequently**

In fact, some teams deploy up to dozens of times per day.

Practices such as monitoring, continuous testing, database change management, and integrating security earlier in the software development process help elite performers deploy more frequently, and with greater predictability and security.

* **Reduce lead time from commit to deploy**

Lead time is the time it takes for a feature to make it to the customer. By working in smaller batches, automating manual processes, and deploying more frequently, elite performers can achieve in hours or days what once took weeks or even months.

* **Reduce change failure rate**

A new feature that fails in production or that causes other features to break can create a lost opportunity between you and your users. As high-performing teams mature, they reduce their change failure rate over time.

* **Recover from incidents more quickly**

When incidents do occur, devops are able to recover more quickly. Acting on metrics helps devops recover more quickly while also deploying more frequently.

| **Service** | **Description** |
| --- | --- |
|  | [Azure Boards](https://azure.microsoft.com/services/devops/boards). These are agile tools that help us plan, track, and discuss our work, even with other  teams. |
|  | [Azure Pipelines](https://azure.microsoft.com/services/devops/pipelines)-  These will let us build, test, and deploy with CI/CD that works with  any language, platform, and cloud. |
|  | [Azure Test Plans](https://azure.microsoft.com/services/devops/test-plans).  These are manual and exploratory testing tools.  Screenshot of a hand-drawn illustration of a CI pipeline. The Build, Test, and Verify stages act on code. The build artifact is the output.  A *pipeline* defines the continuous integration process for the app. It's made up of steps  called *tasks*  . You can think of it as a script that defines how your build, test,  and deployment steps are run.  The pipeline runs when you submit code changes  .  You can configure the pipeline to run automatically, or you can run it manually.  You connect your pipeline to a source repository like GitHub, Bitbucket, or Subversion.  A build agent  builds or deploys the code.  When your build or deployment runs, the system begins one or more jobs.  An agent is an installable software that runs one build or deployment job at a time.  Because we're using Azure Pipelines, we can use a Microsoft-hosted agent.  With Microsoft-hosted agents, maintenance and upgrades are taken care of for us. Each time we run a pipeline, we'll get a fresh virtual machine.  .The final product of the pipeline is a *build artifact*  . Think of an artifact as  the smallest  compiled unit that we need to test or deploy the app. For example, an artifact can be:   * A Java or .NET app packaged into a *.jar* or *.zip* file. * A C++ or JavaScript library. * A virtual machine, cloud, or Docker image.  Check your knowledge Top of Form  **1.**  What is DevOps?    A job title for QA engineers    The union of people, process, and products to enable continuous delivery of value to our customers    A methodology that teams implement quickly to solve all their challenges  **2.**  DevOps is:    A gradual process    Only for startups    A piece of specialized software Check your knowledge Top of Form  **1.**  Which of these is an example of a build artifact?    The code compiler used to build the application.    A Windows Installer (.msi) file that contains a C++ desktop application.    An email that summarizes the build run.  Bottom of Form  Bottom of Form |