**What are Packages?**

The software is delivered into packages - a collection of files and directories needed for a software product. And these packages are designed by the application developer after the completion of the development of the application code. As the software goes to different platforms, to the production stage, or the administrators for installation, software needs to be built into one or more packages.

**What Is Package Management and Why is it Needed?**

A package manager or package management system contains software tools to automate certain processes, such as installation, up-gradation, configuration, and removal of software packages for a computer’s operating system in a consistent manner. In simple words, the package manager maintains the database of software dependencies and version information to avoid software mismatches and missing prerequisites.

A package manager is a programming language’s tool built to create project environments and import external dependencies. Since the package management system maintains all the things related to packages and ensures the availability of such tools within your reach, you don’t have to search or make these tools on your own. Besides this, when you work on a project or library, you can package your project and publish it for others.

It helps you specify dependencies, a package name, author tags, keywords, and version number. By doing this, you can store your package in the online repositories and allow others to find your project.

**What is Azure Artifacts?**

Azure Artifacts is part of the [Azure DevOps](https://www.techtarget.com/searchwindowsserver/definition/Azure-DevOps-formerly-Visual-Studio-Team-Services) suite of tools. It serves as a centralized environment for developers to share and manage code, packages and [artifacts](https://www.techtarget.com/searchsoftwarequality/definition/artifact-software-development).

The service helps developers locate, install and publish NPM, NuGet, Python and Maven package types.

Azure Artifacts is also a package management solution. It is integrated into Azure DevOps and allows you to share various packages using public or private feeds. Any team can take advantage of these artifacts. It allows you to store other artifacts on your feed in the form of universal packages that can be customized according to your needs.

**Why Azure Artifacts Is Used as Package Manager?**

You can also build a solution internally on a server, but surely you will face lots of challenges like excessive management burden and security issues. For instance, if you host your own NuGet server, you will have to take care of the server maintenance, security patching, access management, and connectivity issues for developers and build systems.

Contrary, you can leverage the SaaS platform for your package management needs. Azure Artifacts is integrated into Azure DevOps, a SaaS platform. If you move to this platform, you can relieve your system administrators from management burden, and your organization from chaotic homegrown solutions. Azure Artifacts allows a single sign-in feature to your package feeds, higher uptime, and scalability. Plus, you will experience enhanced security for your package feeds with Azure DevOps’ built-in security feature.

**When should I use Azure Artifacts?**

The Azure Artifacts service automates the process of building and deploying software artifacts such as libraries, executables and documents. It supports a variety of popular languages and frameworks.

Because it is hosted in the cloud, Azure Artifacts comes with built-in security features and enables developers to build projects on top of what others created before them. It is encrypted at rest, high performant and reliable.

**What's included in Azure Artifacts?**

Azure Artifacts includes several features to manage software development artifacts. These include:

* a package management system that lets you host and share NuGet, NPM, Python and Apache Maven packages within your team or organization;
* an integrated build system that lets you automatically publish packages to Azure Artifacts as part of the build process.
* a dependency manager that simplifies how you keep track of dependencies between versions of your packages; and
* a user interface that makes it easy to browse and search for packages, view package details and manage account settings.

**Features and Benefits of Azure Artifacts**

Universal Packages for Any Kind of Usage: With the availability of the Universal Packages, Azure Artifacts has become a universal store for artifacts that you need for development and deployment. Adding to NuGet, npm, and Maven packages feed now Universal Packages that can be used to store any file or set of files. Visual Studio Team Services (VSTS) CLI assists you to create and consume Universal Packages. These Universal Packages can be used to store deployment inputs like installers, large datasets, or binary files that you need during development, or as a versioned container for your pipeline outputs.

* **Sharing is Easy:** With the views feature you can share subsets of the NuGet and npm package-versions in your feed with consumers. The goal of views is to share package-versions that have been tested, validated, or deployed but hold back packages that are still under development or not ready for public consumption. Views and upstream sources are designed to collaborate, produce, and consume packages at an enterprise scale.
* **Control your dependencies:** Upstream sources ensure single feed usage for the packages you produce and the packages you consume from remote feeds including public and private feeds. After enabling the upstream source, any user connected to the feed can install a package from the remote feed, and the feed will save a copy of the same.
* **Easy to use Symbols and the Symbol Server:** To debug compiled executables compiled from native code languages like C++, you need symbol files that contain debugging information. Artifacts make symbol support and publishing quick and simple. After selecting the “Index Sources and Publish Symbols” task, you can publish symbols to the Azure DevOps. The best part is that no advanced configuration or file sharing setup is needed to use this feature.
* **Credential Provider Authentication:** Azure Artifacts secure all the artifacts you publish, but earlier it was challenging to securely use NuGet packages, especially on Mac and Linux. However, with the new Azure Artifacts Credential Provider, you can automate the acquisition of credentials needed to restore NuGet packages as part of your .NET development workflow. Whether you’re using MS Build,.NET, or NuGet(.exe) on Windows, Mac, or Linux, you can use packages from an Azure Artifacts feed and the Credential Provider will automatically acquire and store a token on behalf of the NuGet client.

**Conclusion**

Azure Artifacts is an open-source toolchain for development teams to help with the management of their builds. You can set up a build in minutes using Azure Artifacts without worrying about the installation or configuration of build servers. It is an advanced package management system that ensures better security, accessibility, and scalability compared to previous and existing solutions.

**Rate card**

* 0 - 2 GiB = Free
* 2 - 10 GiB = ₹165.50 per GiB
* 10 - 100 GiB = ₹82.75 per GiB
* 100 - 1,000 GiB = ₹41.38 per GiB
* 1,000+ GiB = ₹20.69 per GiB

