

What is NuGetForUnity?

NuGetForUnity is a NuGet client built from scratch to run inside the Unity Editor. NuGet is a package management system which makes it easy to create packages that are distributed online and consumed by other .NET applications. NuGet is a [semantic versioning](#) framework that allows dependencies in the package.

You can learn more about NuGet here: nuget.org

NuGetForUnity is a tool that allows you to install and use available packages in the editor, install packages, and use available packages in the editor. A simple interface is provided to create and edit `.nuspec` files in the editor to define and build the NuGet package from within Unity.

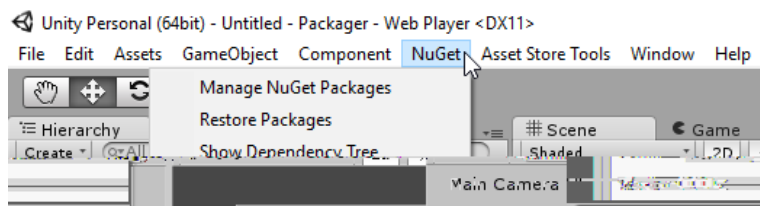


How do I install NuGetForUnity?

In all the listed Unity packages in the Unity Project, I can find [here](#).

How do I use NuGetForUnity?

To launch, select NuGet → Manage NuGet Packages



After installation (it can take some time depending on the size of the package), hold the mouse button like this:



The Online tab shows the package available in the NuGet repository.

Enable **Show All Versions** to list all versions of a package (don't forget to click on the 'g' icon).

Disable **Show All Versions** to only show the latest version of a package.

Enable **Show Pre-release** to list pre-release versions of a package (alpha, beta, pre-release candidate, etc).

Disable **Show Pre-release** to only show stable versions.

There are checkboxes in the Search bar to filter the results.

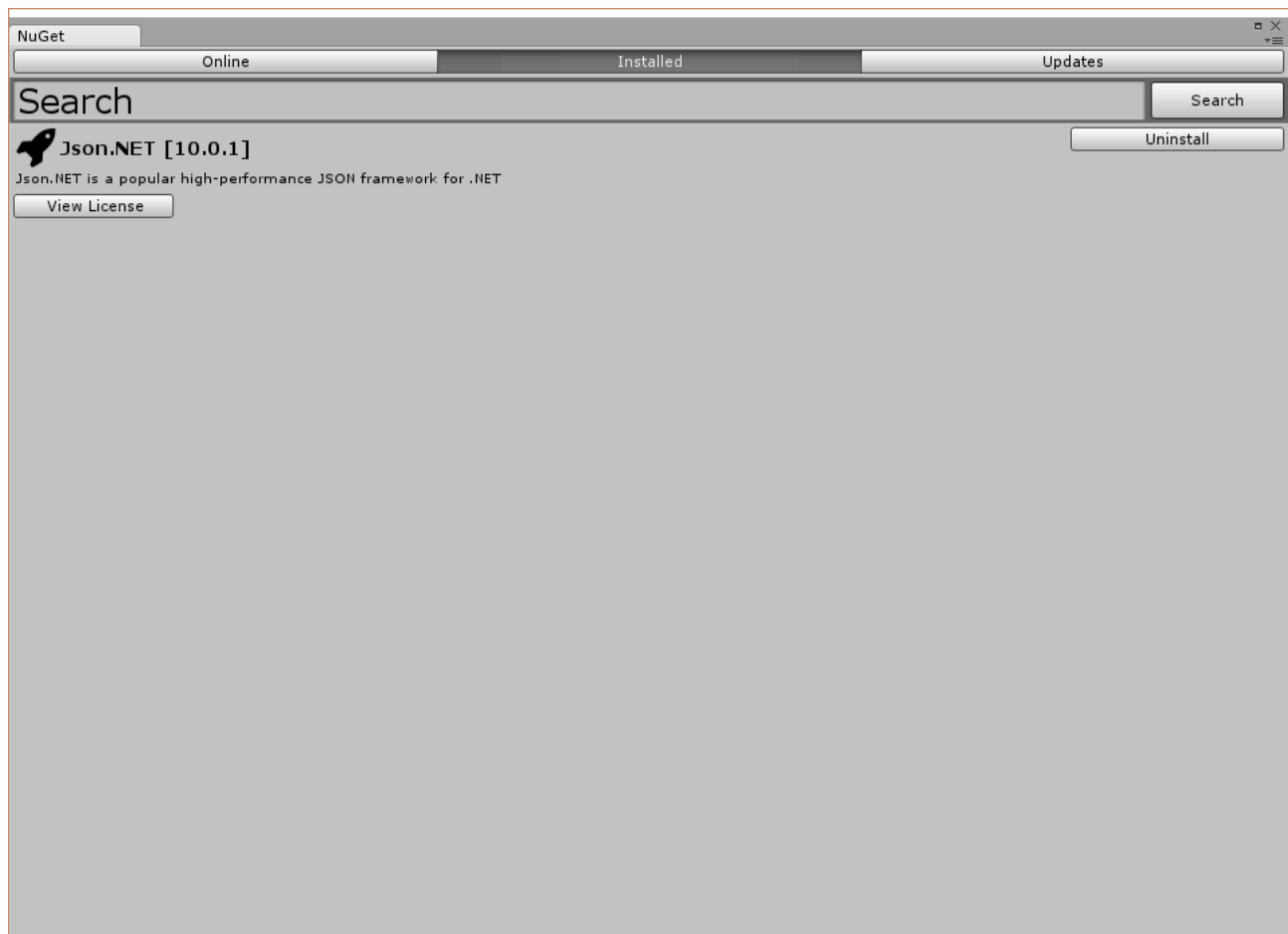
Press the **Refresh** button to refresh the list of packages. (Useful after changing a package in the Hierarchy and clicking on the package in the Hierarchy).

The name of the package, the version of the package (in parentheses), and a description are displayed.

Click the **View License** button to see the license in a web browser.

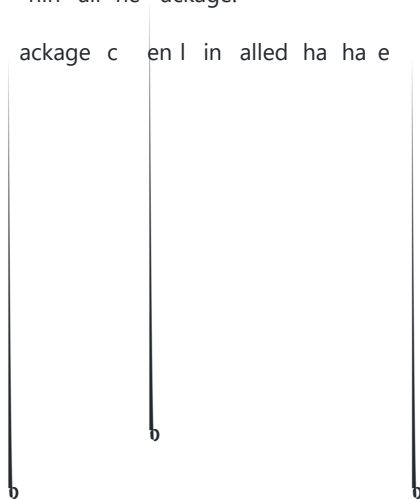
Click the **Install** button to install the package. Note: If the package is already installed, an **Uninstall** button will be displayed which lets you uninstall the package.

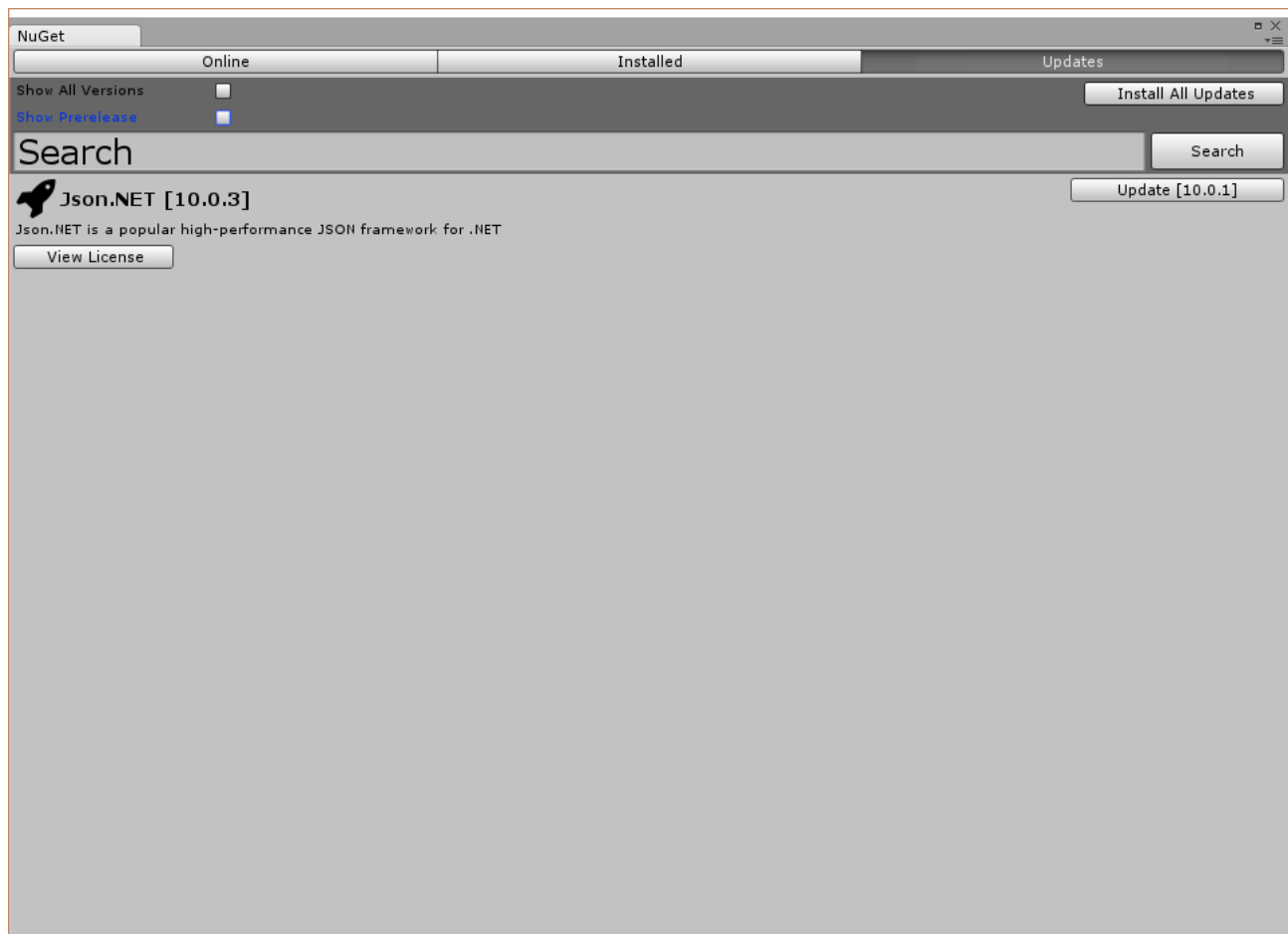
The **Installed** tab shows the packages already installed in the current Unity project.



Click the **Uninstall** button to uninstall the package.

The **Updates** tab shows the package when it is installed but has a newer version available in the feed.





The `e i n` in `backe` n the left i he ne `e i n n` mbe .The `e i n` in `backe` in the **Update** b n i he c en l in alled `e i n`.

Click the **Update** b n n in all the c en ackage and in all the ne ackage.

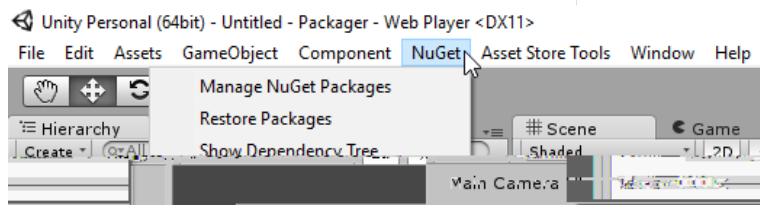
How does NuGetForUnity work?

N Ge F Uni l ad he *NuGet.config* file in he Uni jec (a ma icall c ea ed if he e i n' al ead ne) in de de e mine he e e i h l d ll ackage d n f m and h ackage . B defa l, hi e e i e he n ge . g ackage ce.

The default NuGet.config file:

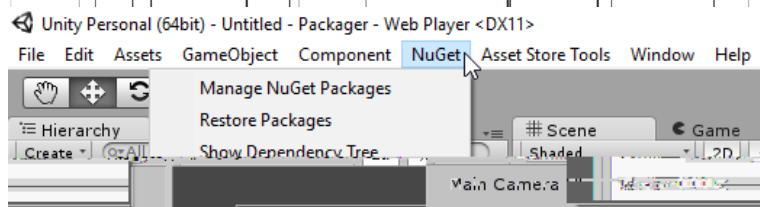
Y can change hi an he N Ge e e (ch a N Ge .Se e P Ge - ee bel). The **NuGet** → **Reload NuGet.config** men i em i ef l if a e edi ng he *NuGet.config* file.

See more information about `NuGet.config` file here: <https://docs.unity3d.com/Manual/NuGetConfiguration.html>



NuGet for Unity installs all packages in the local file system as defined in the `NuGet.config` file (located in the `Assets` folder). In the `NuGet.config` file, you can define the package source to use. By default, the package source is set to the Unity Package Manager. You can also define a custom package source. The package source is defined in the `packages.config` file. The package source is defined in the `packages.config` file.

When a package is installed, the `packages.config` file in the project is automatically updated with the specific package information, including all of the dependencies that are installed. This allows for the package to be used from each application. The package information is automatically updated when the package is installed. The package information is automatically updated when the package is installed.



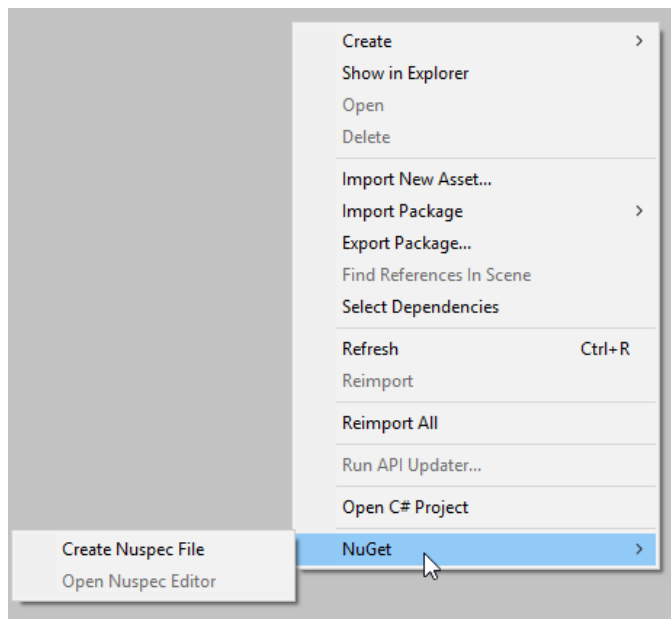
Note: Depending on the number of packages needed to be installed, the process could take a long time, especially if you are installing a large number of packages. If you are installing a large number of packages, the process could take a long time, especially if you are installing a large number of packages.

If a package is installed in the Unity Package Manager, the package is automatically updated when the package is installed. The package is automatically updated when the package is installed. The package is automatically updated when the package is installed.

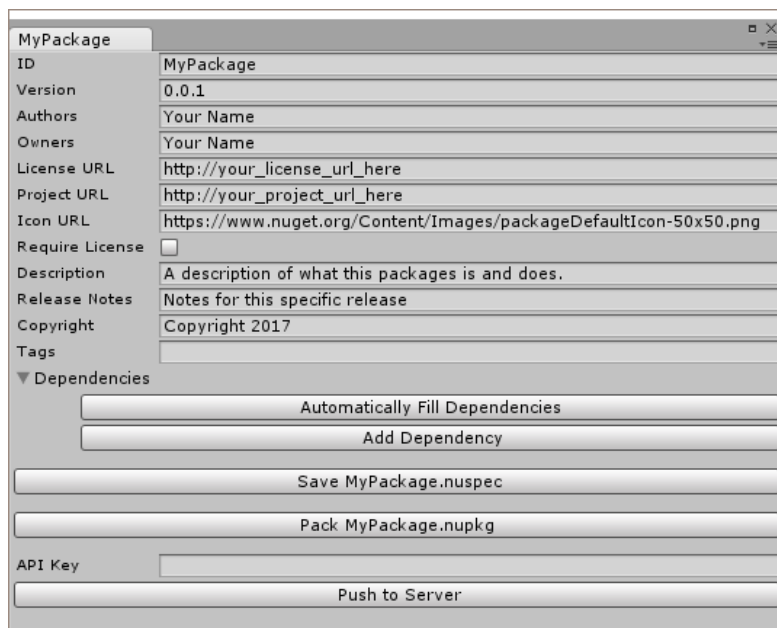
The `.nupkg` file is downloaded from the NuGet package source and cached locally in the `Assets` folder. The package is cached locally in the `Assets` folder. The package is cached locally in the `Assets` folder.

How do I create my own NuGet packages from within Unity?

First, you'll need to create a `.nuspec` file that defines the package. In the Project window, right-click the `Assets` folder and select `NuGet > Create Nuspec File`.



Select the new `.nuspec` file and hold the mouse button like this:



In the above information package (ID, Version, Authors, Description, etc). Be sure to include the dependencies as well as the package.

Press the Pack button on the package in the `.nupkg` file has been added in the folder.

Press the Push button on the package to the server. Be sure to enter the correct API Key to get the package to the server (if you have configured the server).

How do I create my own NuGet server to host NuGet packages?

You can use [NuGet.Server](#), [NuGet Gallery](#), or [NuGet](#), etc to create a NuGet server.

Alternatively, you can use a "local feed" which is a folder on the hard drive and use it.

Be sure to use the `URL` in the `NuGet.config` file and hold the button!

Read more information here: <https://d.c.n.g/e.g/c/ea/e/h/ing/-n-n/ge/-feed>

0