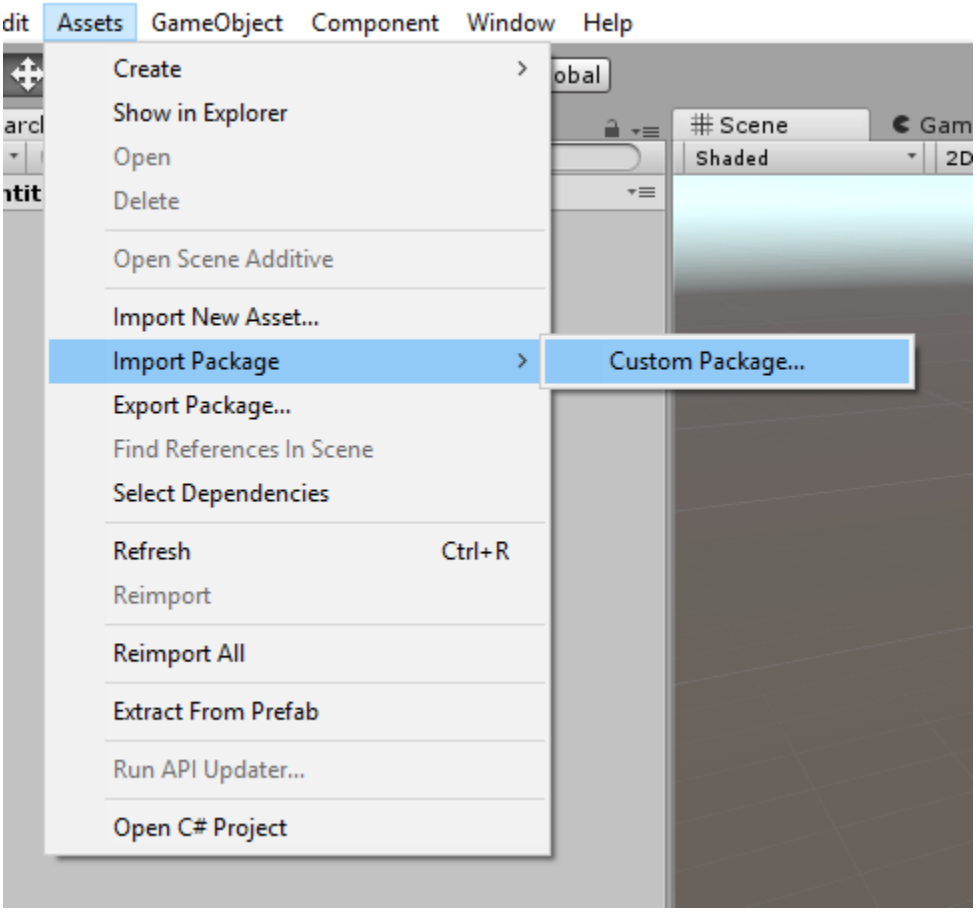
How to Use the Unity Streaming Plugin

This plugin for the Unity Game Engine provides the ability to create an audio and video stream that can be connected to and played from inside Digistar, and on the dome. It makes use of the NDI protocol to achieve a low-latency, high quality stream. This plugin works inside the Unity editor as well as in standalone builds, so you can test your project directly from the Unity editor. In order for Digistar to connect to the stream, the machine running Unity must be connected to the non-realtime network. Currently the plugin only works on Windows computers with DirectX 11 support. The plugin supports Unity version 2018.3 and newer.

Adding the Plugin to a Unity Project

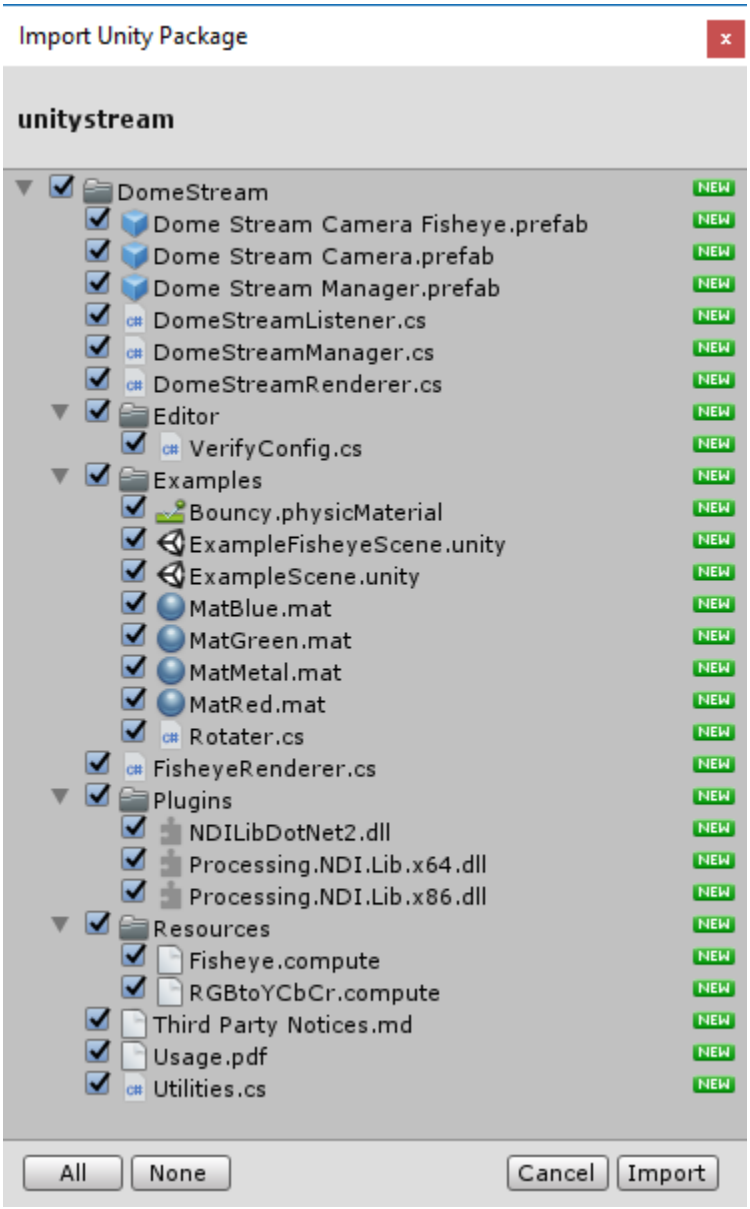
1. Import the UnityStream package

   a. On the editor's main menu, select Assets\Import Package\Custom Package...



   b. Navigate to D6Software\Utilities\UnityStream\ and select the UnityStream.unitypackage file.

c. Select the Assemblies and Scripts folders, and everything they contain (optionally select the Examples folder)



d. Select Import

e. If you see compiler errors regarding 'unsafe code', navigate to 'Edit\Project Settings', select 'Player', and inside 'Other Settings', check 'Allow 'unsafe' code'.

   i. The 'unsafe' code we use in the plugin is a C# feature that allows us to directly interact with system memory. We need to use this to improve the streaming performance.

Configuring the Unity Project

1. In the first scene of your project, add the 'Dome Stream Manager' prefab. When the 'Dome Stream Manager' script is initialized, it will start streaming, and when it is destroyed it will stop streaming. It is useful to check the 'Don't Destroy' checkbox on the script so that the object won't be destroyed when a new scene is loaded.
2. The Dome Stream Manager requires a video and audio source from the scene to capture output for the stream. It uses a scene Camera and Audio Listener for these sources. If the checkbox 'Auto Find Sources' on the Dome Stream Manager is checked, then whenever a new scene is loaded the scene will be searched for a valid Camera and Audio Listener. In this way, you can configure your scenes to use different cameras that the stream will automatically detect and use as necessary.
3. The plugin supports normal orthographic/perspective perspective rendering (i.e., a normal rectangular display), or fulldome fisheye rendering.
    a. The normal orthographic/perspective setup is very straightforward: the output of the configured camera will be streamed directly from the plugin.
    b. The fisheye setup is slightly more complicated. In order to generate a fisheye image of the scene, we render a cubemap of the scene from the 'original' scene camera, and then convert that cubemap into a fisheye image that is then streamed from the plugin. In effect, there are two cameras: a 'scene camera' that renders the scene as normal, and a 'stream camera' which takes the output of the scene camera and streams it. Please also take the following into consideration when using the fisheye rendering:
        i. Because of the extra rendering involved with creating a fisheye image, large and complex scenes will have a large impact on the performance of the application.
        ii. Some image effects do not work correctly with the fisheye rendering. Most notable are screenspace dependent effects, such as reflections.
4. The two camera prefab objects in the plugin: Dome Stream Camera, and Dome Stream Camera Fisheye can be used for reference, or as-is in your project.

Other Considerations

1. There are two example scenes included in the plugin: DomeStream\Examples\ExampleScene and DomeStream\Examples\ExampleFisheyeScene. Please consult these scenes for an example of how to configure the plugin components.
2. Because the stream only renders the output of a single camera, the Unity UI canvas must also be configured to be rendered with the source camera if you want it to be included in the stream
3. There can only be a single stream on the local network with the same name at a given time. If for some reason you need to have multiple simultaneous streams, make sure they have different names set on the Dome Stream Manager.

4. Depending on your network settings, the Windows firewall may block the stream. If you experience any issues finding the stream, try running the Unity application as an administrator, and adding configuring the Windows firewall to allow the application through all networks.
5. The plugin allows for streaming at both 30 and 60 fps, but currently Digistar only supports playback at 30 fps. While a 60 fps will still play in Digistar, it is recommended to only stream at 30 fps to avoid unnecessary performance overhead.
6. If you would like to test the stream outside of Digistar, download and install the NDI Tools package from NewTek. The NDI Studio Monitor will allow you to connect to streams on the local network.

Playing the Stream In Digistar

To view a stream from a Unity application built with the plugin, inside of Digistar, follow the directions for NDI streaming in 'Use Video Input'. Once the Unity application is streaming, it will show up in the list of available sources.