Using Transmission in Unity

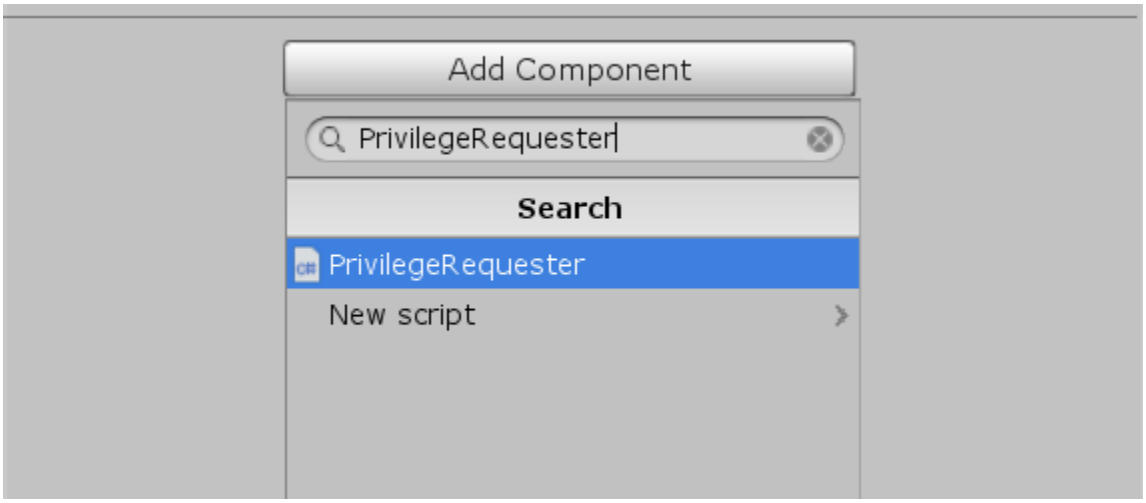
Set up Transmission and Spatial Alignment in your scene

Transmission is simply Magic Leap's way of sharing content between two or more players in a scene. For example, if I spawn a red cube in the scene, Transmission will allow all the other players in the scene to see that same red cube in the world.

However, Transmission alone will not guarantee that you will see it in the exact same location. This is where **Spatial Alignment** comes in. By adding the Spatial Alignment script to your scene, you will be able to accurately share data using Transmission. To set up Transmission and Spatial Alignment into your scene, follow the steps below.

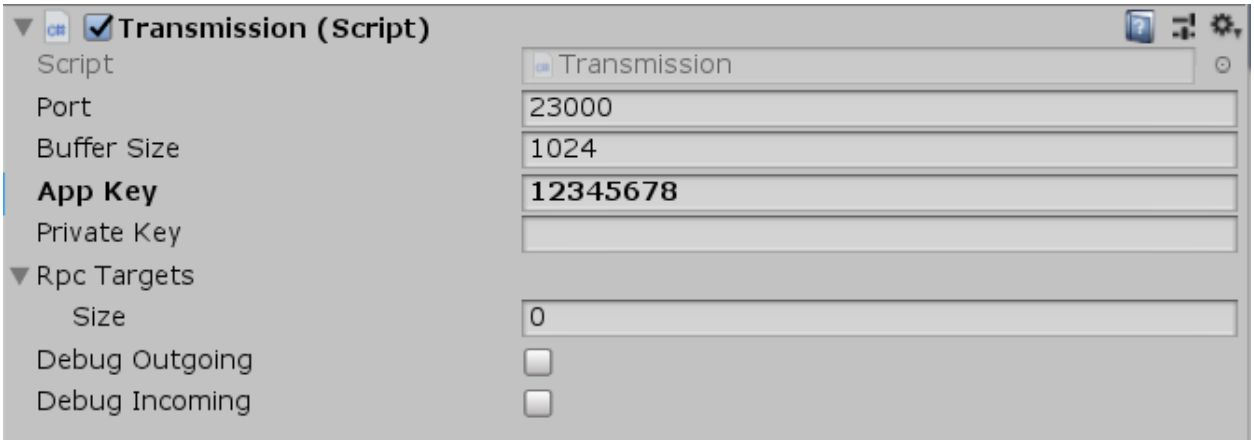
STEP 1: Add the Privilege Requestor Script to your scene

Click on Add Component to any active GameObject in your scene and add **Privilege Requestor**. Then, under the *Privileges* section of the Privilege Requestor component, set *Size* to 1 and *Element 0* to Local Area Network as shown below



STEP 2: Add the Transmission Prefab to your scene

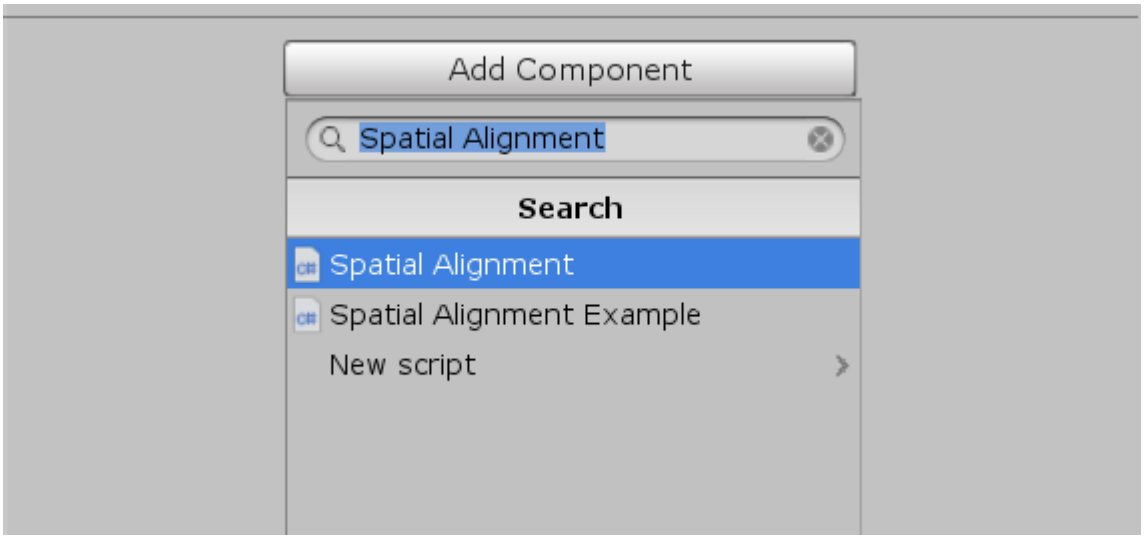
Look for the **Transmission** prefab asset under **MagicLeap-Tools > Prefabs > Networking** and add it anywhere in your scene. For Transmission to work, you will need to add an **App Key** so that the Transmission system will connect you to other peers running apps with the same App key. It is crucial that everyone share the same App Key in their .mpk file.



Adding a **Private Key** is optional and you may leave it blank, but if more than one Transmission app is running on the same network, you can use this to further restrict users on the network. All users with the same App Key **and** Private Key will be connected by Transmission.

STEP 3: Add Spatial Alignment

Add the **Spatial Alignment** script to any empty GameObject in the scene. Do not add this to more than one GameObject.



STEP 4: Put all your transmission prefabs in a Resources folder

Create a folder in your project and title it **Resources**. The folder can be anywhere in your project. Drag all the prefabs that you wish to use Transmission on, into this folder.

Note: Not every spawned object needs to use Transmission. Just the ones that you want everyone to see.

STEP 5: Add the TransmissionObject component

Add the **TransmissionObject** component to each of the Transmission prefabs in the Resources folder you created.

STEP 6: Instantiating the prefabs

To spawn a prefab that is using transmission, you will need to use the **Spawn()** function from the Transmission class. First, add **using MagicLeapTools;** to your script.

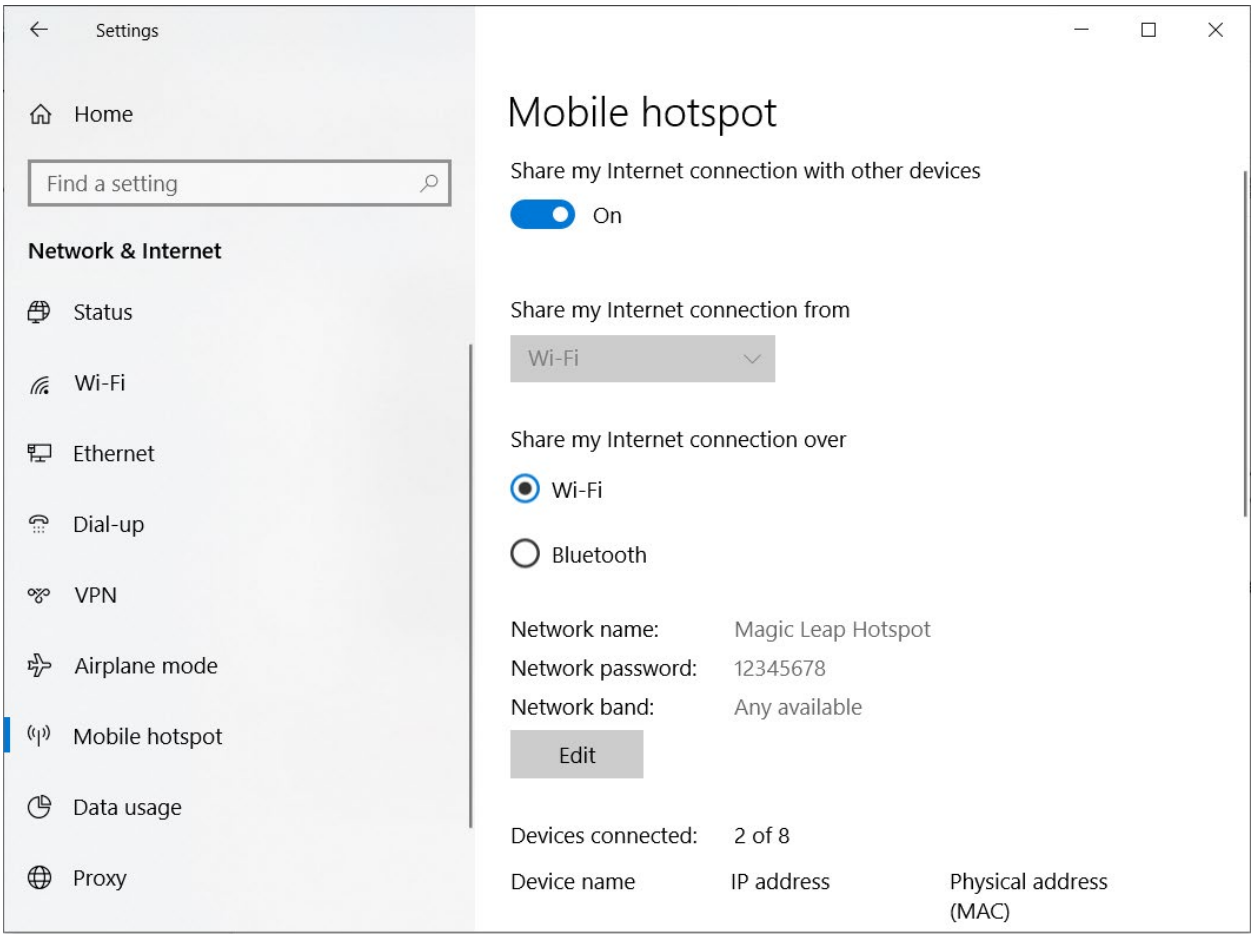
Next, create a **TransmissionObject** variable that will store your transmission object and use the Spawn() function to instantiate it. If you use the Instantiate() function provided by Unity instead, **Transmission will not work.** You must use the Spawn() function as shown below.

Again, make sure that all the prefabs that you are instantiating in the scene are in a Resources file somewhere in your project.

STEP 7: Set up a Local Area Network

Once you are ready to build your project and test it with multiple headsets, you will need to make sure all your headsets are connected to the same Local Area Network (LAN). You can do this by setting up a mobile hotspot on your phone or on your laptop and then having all the headsets connect to it.

To set up a mobile hotspot on a Windows PC or laptop, go to **Settings > Network & Internet > Mobile Hotspot**


Settings
Home
Find a setting
Network & Internet
Status
Wi-Fi
Ethernet
Dial-up
VPN
Airplane mode
Mobile hotspot
Data usage
Proxy
Mobile hotspot
Share my Internet connection with other devices
On
Share my Internet connection from
Wi-Fi
Share my Internet connection over
Wi-Fi
Bluetooth
Network name:
Magic Leap Hotspot
Network password:
12345678
Network band:
Any available
Edit
Devices connected:
2 of 8
Device name
IP address
Physical address (MAC)

Set up spectator headsets

A spectator headset is a headset that is able to see everything in the world and any changes done to it but not interact with it. The master headset creates the world and interacts with it. This is useful for quickly showcasing the abilities of an application to multiple people without having to pass a single headset around so everyone can see.

To set up a spectator headset, you must make sure all the headsets are connected to the same Local Area Network (LAN). You can do this by setting up a mobile hotspot on your phone or on your laptop and then having all the headsets connect to it. As well as this, all .mpk files downloaded to each headset must have the same app key.

STEP 1: Download the master scene

Make sure your master scene is already set up with Transmission and Spatial Alignment. If you haven't done so, simply follow the steps outlined in the **Set up Transmission and Spatial Alignment in your scene** section. The master scene is simply the scene you created that you wish to showcase to a spectator or group of spectators.

Download your master scene as an .mpk file to the headset you wish to be the master headset.

Transmission.Peers.Length doesn't update on start (will give you a value of 0 until after start)
No access to global variables on start nor on enable