

# ***Plus Networking - Using Android Native Plugin***

*Version 1.2*

Plus Networking is design to handle network related functionality using Google Play real time multiplayer.

Plus Networking can instantiate players across network, sync transformers & rigid bodies, call functions over network.

## **How to setup example scene to test:**

- 1) Create new blank project then download & import Android native plugin from asset store & install it.
- 2) To avoid facebook related errors - If you don't want to use facebook API of it, simply remove it from its setting page(window>Stan's Asset>Android Native>Edit Settings) or configure it.
- 3) Download & import Plus Networking from asset store. Open Assets>Plus Networking>UsingAndroidNativePlugin>PlusNetworking>Example>RTM\_Example.scene.
- 4) In build setting add this scene & switch to android platform then fill your all necessary settings in player setting including Bundle identifier, Version, bundle version code etc & sign the apk using your keystore.
- 5) Also update manifest.xml with above values.
- 6) Setup app in google developer console & also make sure to setup in "Game Services" tab. Go to Game Services>LinkedApps & create one linked app if you have not & make sure you have enabled real time multiplayer in that. Add testers in "Testing" tab. Login using these tester IDs in device to test networking example.
- 7) Copy text from "Get Resources" from developer console's leaderboard or achievement page & replace text in ids.xml file in Plugins>Android>Res>Values.
- 8) Set number of min max players in Plus Network Manager - for two devices min 1 max 2. Build & run in both devices.
- 9) Press connect in both devices, you will see both players in each other devices. Select your color player & do quick match. Move your player, you can see multiplayer working..!

## **Implementation**

### **Importing :**

You need to have ANP plugin installed first to run Plus Networking. Than you can import Plus Networking without any errors.

Delete other plugin folder "UsingGooglePlugin" or do not import it.

### **Components:**

You have a ready to use component scripts to setup your Networking.

#### (A) Plus Networking manager

You have to create one empty game object. Drag this component script on it.

##### (1) Min Max Players :

Assign minimum & maximum players (max 8 players allowed in one room)

##### (2) Spawn Position :

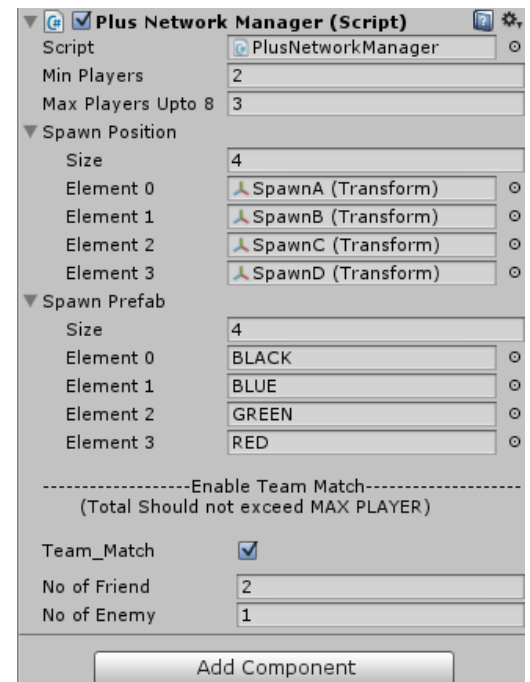
Drag empty game object (transforms) of positions (where you want your player prefabs to be spawn)

##### (3) Spawn Prefab :

Drag your player prefabs here

##### (4) Team Match :

You can choose if you want a team match here, then you can assign number of friend player & number of enemy player. Please note the total of local player  
+ friend player + enemy player should not exceed your maximum player value



#### (B) Plus Network Transform:

Attach this component script to your player prefabs.

##### (1) Data transfer rate :

This will set data transfer rate per second so setting it high will send more data per second & low will send less data per second. ( Note : According to Google documentation if you want to send data higher then 50 then it is recommended to use non reliable data type )

##### (2) Packet Type :

Sets packet type, for more info on this visit:

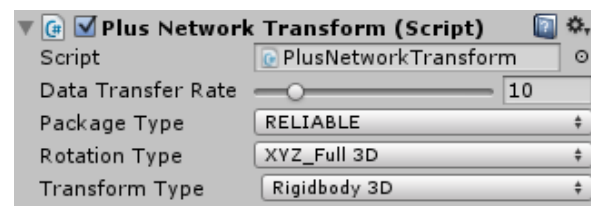
[https://developers.google.com/games/services/common/concepts/realtimeMultiplayer#sending\\_game\\_data](https://developers.google.com/games/services/common/concepts/realtimeMultiplayer#sending_game_data)

##### (3) Rotation Type :

Select the rotation axis you want to sync across network

##### (4) Transform Type :

Select the transform axis you want to sync across network



#### (C) Plus Network Identity:

Attach this component script to your player prefab.

#### (D) Scripting Implementation:

##### (1) Network function call - For calling network methods

PlusNetworkManager.PNInstance.SendMethodName("YourMethodNameHere");

##### (2) Tagging

You can use three default tagging system for specific functionality of Local, Friend & Enemy players.

For Local Player : tag is "plusMyPlayer"  
eg. if (gameobject.tag == "plusMyPlayer")  
{  
    Your Logic goes here..  
}

Same way you have tag "plusFriend" for friend players & tag "plusEnemy" for enemy players.

### 3) For advance setup

You can use index numbers of players for any particular functionality  
So for example : if you want to make player [0] a "GameOwner", you can customize tag him by replacing

```
player.tag = "plusMyPlayer";
```

line to

```
if(i == 0)  
{  
    Assign any tag & apply Your Logic for player index 0..  
}  
else  
{  
    player.tag = "plusMyPlayer";  
}
```

### (E) Related documentations:

**(1) Setting up Google Services RTM in developer console link :**

[https://developers.google.com/games/services/console/configuring#enabling\\_multiplayer\\_support](https://developers.google.com/games/services/console/configuring#enabling_multiplayer_support)

**(2) Adding Tester account link:**

<https://developers.google.com/games/services/console/testpub>

**(3) Android Native Plugin Documentation link:**

<https://unionassets.com/android-native-plugin/manual>

For support contact us at [gamozome@gmail.com](mailto:gamozome@gmail.com)

If you like this product please do rate and review on asset store.

Asset Store : <http://u3d.as/k9h>