Sunshine Android Native Share:

Just import the package in your unity project and you are ready to use the plugin. You don't need to drag any prefab. The only thing you need to do is to call some static functions.

Video tutorial for setup is here https://youtu.be/GuCw5plwxtl

Setup:

Open androidManifest.xml file from "Plugins / Android / androidManifest.xml". replace the android:authorities name from provider block with some unique name. This will be your file provider path.

```
♦ SunShineNativeShare.cs
                                  × TestShare.cs
                                                                 × AndroidManifest.xml
■ Assembly-CSharp ▶ manifest ▶ application ▶ provider ▶ @android:authorities
       1 <?xml version="1.0" encoding="utf-8"?>
      2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3 xmlns:tools="http://schemas.android.com/tools"</pre>
                   package="com.SmileSoft.unityplugin">
           <application android:label="@string/app_name" android:icon="@drawable/app_icon" >
            <activity android:name="com.unity3d.player.UnityPlayerActivity" android:label="@string/app_name" android:screenOrientation="fullSensor" android:launu
     10
11
                 <action android:name="android.intent.action.MAIN" />
              12
     14
15
16
17
               <meta-data android:name="unityplayer.UnityActivity" android:value="true" />
           </activity>
              Kprovider
     18
19
                   android:name="and
                   android:authorities="com.SmileSoft.unit
                                                                                               replace this with Some unique name
                   android:exported="1
     22
23
                   android:grantUriPermissions="true">
                <meta-data
                    android:name="android.support.FILE_PROVIDER_PATHS" android:resource="@xml/file_provider_paths" />
              </provider>
           </application>
             <uses-sdk android:minSdkVersion="16" android:targetSdkVersion="27" />
     30 c/manifests
```

Then Again open SunShineNativeShare.cs script from "Plugins / SunShine Android Native Share / SunShineNativeShare.cs". Copy the provider path from androidmanifest file and paste it in "FileProviderName "variable.

** Strongly recommended that you should use your package name as your provider name **

```
SunShineNativeShare.cs + X TestShare.cs
Unity Native Share Free.Plugins

    SunShineNativeShare

    ShareText(string message, string shareDialogTitle)

          □public class SunShineNativeShare : MonoBehaviour
                public static string TYPE_IMAGE = "image/*";
    9
10
11
12
                private const string SHARE_PACKAGE_NAME = "com.SmileSoft.unityplugin";
    13
                private const string SHARE_CLASS_NAME = ".ShareFragment";
                private const string TEXT_SHARE_METHOD = "ShareText";
    15
16
17
                private const string SINGLE_FILE_SHARE_METHOD = "ShareSingleFile";
    18
                private const string FileProviderName =
                                                            "com.SmileSoft.unityplugin.ShareProvider_test";
                                                                                                                      Replace the provider path from manifest File
    19
20
21
    22
23
24
                public static void ShareText(string message, string shareDialogTitle)
    25
           #if UNITY_ANDROID
    26
27
                     using (AndroidJavaObject share_android_obj = new AndroidJavaObject(SHARE_PACKAGE_NAME + SHARE_CLASS_NAME))
                         share_android_obj.Call(TEXT_SHARE_METHOD, message, shareDialogTitle);
    28
29
30
31
                     Debug.Log("Native Share just work in android Platform");
    32
33
34
35
36
    37
38
                public static void ShareSingleFile(string path, string fileType, string message, string shareDialogTitle)
    39
            #if UNITY ANDROID
                     using (AndroidJavaObject share_android_obj = new AndroidJavaObject(SHARE_PACKAGE_NAME + SHARE_CLASS_NAME))
                         share\_android\_obj.Call(SINGLE\_FILE\_SHARE\_METHOD,\ FileProviderName,\ path,\ fileType,\ message,\ shareDialogTitle);
    45
    46
47
48
                    Debug.Log("Native Share just work in android Platform");
```

1. Share Text:

SunShineNativeShare.ShareText (string message, string share_dialog)

2. **Share Single File:** (free version only support image type file . Pro version supports all most all type of files)

SunShineNativeShare.ShareSingleFile(string path, string fileType, string message, string shareDialogTitle)

Ex: SunShineNativeShare.ShareSingleFile("path", SunShineNativeShare.TYPE_IMAGE,"Share Message","Share By sunshine")

3. Share Multiple File of Same Type: (Pro Version Only)

SunShineNativeShare. ShareMultipleFileOfSameType (string[] path, string fileType, string message, string shareDialogTitle)

4. Share Multiple File of Multiple Type: (Pro version Only)

SunShineNativeShare. ShareMultipleFileOfMultipleType (string[] path, string message, string shareDialogTitle)

** If you previously import the free version of this plugin then please delete the "SunShine Android Native Share" folder first for unexpected conflict of scripts before using the pro version. **