Documentation

ImaginationOverflow Deferred Deep Linking for Android is a direct implementation of Google Play Install Referrer API [https://developer.android.com/google/play/installreferrer/], it enables your unity applications or games to return the install referrer information in order to track installs, give users rewards, etc.

Info

The plugin only works for Play Store Installs

Warning

Due to Google annoucement [https://android-developers.googleblog.com/2019/11/still-using-installbroadcast-switch-to.html], the plugin doesn't configure the *com.android.vending.INSTALL_REFERRER* BroadcastReceiver, commonly used to track installs and referrers, since it will be deprecated on March 1, 2020.

Integration

Integrating with the plugin is as simple as registering an event in order to get the information, and call a method to start retrieving the data. This is an asynchronous operation but the plugin ensures that all event handlers are run in the Unity main thread.

```
using ImaginationOverflow.AndroidInstallReferrer;
using ImaginationOverflow.AndroidInstallReferrer.JavaInterop;
....

InstallReferrerManager.Instance.ReferrerInformationCollected +=
Instance_ReferrerInformationCollected;
InstallReferrerManager.Instance.FetchInformationCollected();
```

In the event handler you have access to all referral information stored by the play store, or in case of an error, the reason for the error and its details:

```
private void Instance_ReferrerInformationCollected(InstallReferrerInfo
data)
{

    if (string.IsNullOrEmpty(data.Error) == false)
    {
        Debug.LogErrorFormat("Deferred Deep Link Error!\n{0}\nIs
Exception: {1}", data.Error, data.IsException);
        return;
    }

    var referrerInfo = data.InstallReferrer;
    var referrerTime = data.ReferrerClickTime;
    var installTime = data.InstallTime;
    var playInstant = data.GooglePlayInstant;
}
```

Features

Caching

The plugin, by default, automatically caches the referrer info after the first data collection, this avoids unnecessary calls to the native API, internally we use PlayPrefs [https://docs.unity3d.com/ScriptReference/PlayerPrefs.html] to store this information. If this behavior is not intended you can deactivate it by setting **CacheReferrerInformation** to *false* before calling the *FetchInformationCollected* method.

```
//
//  Deactivate caching
//
InstallReferrerManager.Instance.CacheReferrerInformation = false;

(...)
```

The plugin reads from the PlayPrefs in the static constructor of the InstallReferrerManager class, that means that you can test if there are any cached data before you register your handler and call the *FetchInformationCollected* method:

```
if (InstallReferrerManager.Instance.ReferrerInfo != null)
{
    //do something
}
```

Note

When caching is **enabled** and cached data is available, the plugin won't query the *Google Play Install Referrer API*, it will automatically trigger the **ReferrerInformationCollected** event.

Note

On **Unity Editor** the caching feature is **always disabled**, this was made by design to allow continuous testing.

Universal Deep Link Integration

The plugin also integrates directly with our Universal Deep Link plugin [https://universaldeeplinking.imaginationoverflow.com/], so if you enable this option the **ReferrerInformationCollected** event won't ever be called, and instead, the **LinkActivated** will.

To enable the Universal Deep Link integration, you just need to set **FetchInformationCollected** only argument as true:

```
InstallReferrerManager.Instance.FetchInformationCollected(triggerUnivers
```

The generated link passed to the Universal Deep Link plugin will have the following format:

```
androidInstallReferrer://info?InstallReferrer={0}&GooglePlayInstant=
{1}&InstallTime={2}&ReferrerClickTime={3}&Error={4}&IsException={5}

//
//  {0} - install referrer [URL encoded]
//  {1} - GooglePlayInstant information
//  {2} - Install Time [Unix timestamp]
//  {3} - Referrer Click Time [Unix timestamp]
//  {4} - Error [URL Encoded]
//  {5} - Is Exception
//
```

To retrieve all arguments you can use the following snippet:

```
private void
Instance_LinkActivated(ImaginationOverflow.UniversalDeepLinking.LinkActi
 la)
{
    if (la.Uri.StartsWith("androidInstallReferrer"))
    {
        //
        //  Deferred activation
        //
        var installReferrer =
WWW.UnEscapeURL(la.QueryString["InstallReferrer"]);
        var playInstant =
bool.Parse(la.QueryString["GooglePlayInstant"]);
        var installTime =
long.Parse(la.QueryString["InstallTime"]).UnixTimestampToDateTime();
        var referrerClickTime =
long.Parse(la.QueryString["ReferrerClickTime"]).UnixTimestampToDateTime(

        var error = WWW.UnEscapeURL(la.QueryString["Error"]);
        var isException = bool.Parse(la.QueryString["IsException"]);
    }

    ...
}
```

Note

The *UnixTimestampToDateTime* function, is an extension method of long included with the plugin. To use it don't forget to include the **ImaginationOverflow.AndroidInstallReferrer.Helpers** namespace.

About Referrer Installs

Referrer installs work by adding custom information to the Play Store URL, the Play Store app stores this information and relays them to the target app/game after it's opened. The play store URLs have the following template:

```
https://play.google.com/store/apps/details?id=[APP/GAME_ID]
```

To add referral information you need to include on this store URL the **referrer** argument, this argument should be URL Encoded [https://en.wikipedia.org/wiki/Percent-encoding].

```
https://play.google.com/store/apps/details?id=[APP/GAME_ID]&referrer=
[MY_ARGUMENTS]
```

Although you can add anything that you wish to these arguments, they usually contain Urchin Tracking Module parameters [https://en.wikipedia.org/wiki/UTM_parameters], these are usually used for Google Analytics (and similar services) to track referral information automatically.

Considering one of our games, Sudoku Zenkai as an example for referral link usage:

```
Package ID: com.imaginationoverflow.sudokuzenkai
Arguments:
utm_source=imaginationoverflow.com&utm_medium=docs&puzzle=123
Arguments Encoded:
imaginationoverflow.com%26utm_medium%3Ddocs%26puzzle%3D123

https://play.google.com/store/apps/details?
id=com.imaginationoverflow.sudokuzenkai&referrer=imaginationoverflow.com
```

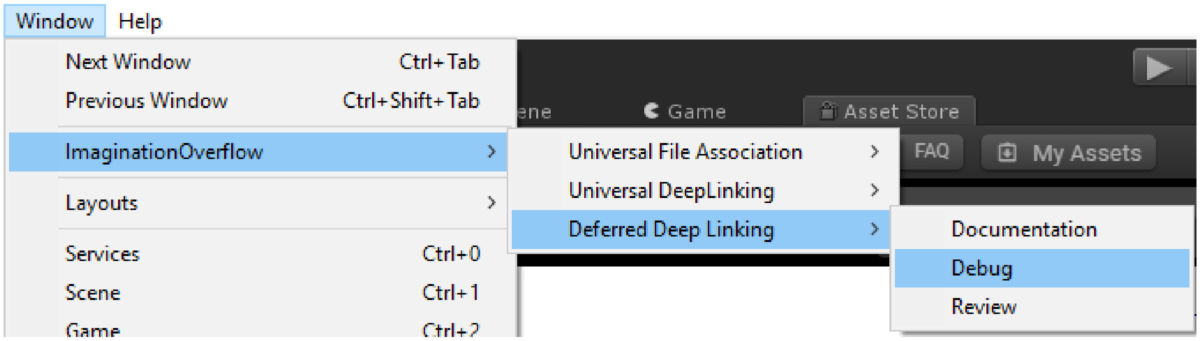
⚠ **Warning**

Link generation, **Link distribution**, and **Referral Tracking** are out of this plugin scope, the plugin only purpose is to retrieve the information and give it to you, the plugin doesn't handle anything else not stated on this documentation.

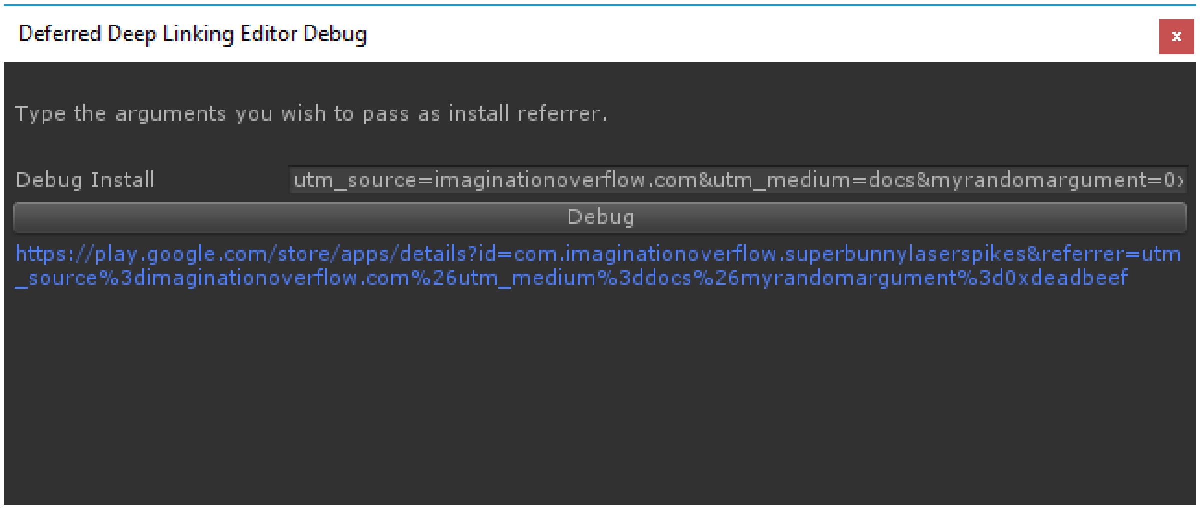
Testing

Editor

In the plugin, we included a quick and simple way for you to test the integration right from the Unity Editor. To use it go to Window -> ImaginationOverflow -> Deferred Deep Linking:



A Debug window will popup where you can configure the arguments you wish to pass as referrer information:



Type the arguments you wish to pass as install referrer.

Debug Install utm_source=imaginationoverflow.com&utm_medium=docs&myrandomargument=0x

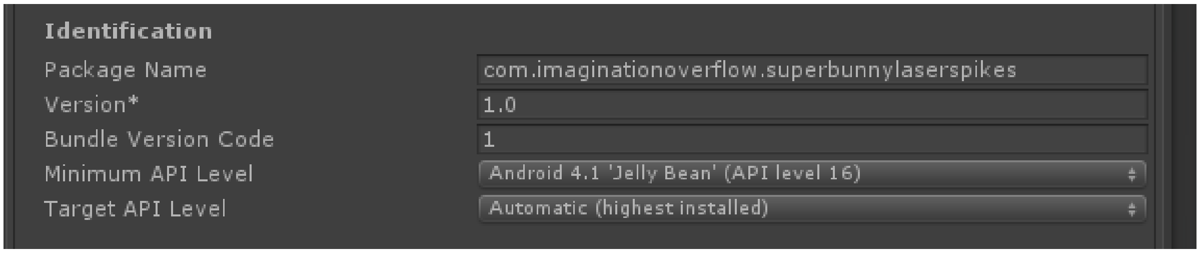
https://play.google.com/store/apps/details?id=com.imaginationoverflow.superbunnylaserspikes&referrer=utm_source%3dimaginationoverflow.com%26utm_medium%3ddocs%26myrandomargument%3d0xdeadbeef

After you complete the integration with the Deferred Deep Linking plugin, simply run your project, and press the **Debug** button, the **ReferrerInformationCollected** will be triggered with the configured referrer information.

The debug window contains a selectable label that allows you to preview how your links will look like once your app/game is on the store.

Device

To test the plugin integration while developing you can use the google play store alpha or beta tracks, but if you wish to test without deploying your app to the play store you can test by changing your application package name to any currently available app/game on the Play Store, as an example, lets use one of our games Super Bunny Laser Spikes, which package name **com.imaginationoverflow.superbunnylaserspikes**



Identification

Package Name	com.imaginationoverflow.superbunnylaserspikes
Version*	1.0
Bundle Version Code	1
Minimum API Level	Android 4.1 'Jelly Bean' (API level 16)
Target API Level	Automatic (highest installed)

Then get your referral link ready:

```
https://play.google.com/store/apps/details?id=
[TARGET_PACKAGE_NAME]&referrer=[YOUR_ARGUMENTS_URL_ENCODED]

URL Decoded arguments:
utm_source=imaginationoverflow.com&utm_medium=docs&myrandomargument=0xde


Final Link
https://play.google.com/store/apps/details?
id=com.imaginationoverflow.superbunnylaserspikes&referrer=utm_source%3Di
```

Tip

To encode/decode your arguments you can use on **Unity** the WWW.(EscapeURL/UnEscapeURL) methods, on standard .NET HttpUtility.(UrlEncode/UrlDecode) or this online tool [https://www.urlencoder.org/]

On your device, make sure that the app isn't installed, and open the referral link, the play store should appear and the install button should be visible.

Finally, install your app/game via *adb* or press Build and Run on the Unity Editor:

```
adb install myAwesomeGame.apk
```

You should see the store page Install button changing to Open, press it and you will be able to see the referral information that you passed along on **ReferrerInformationCollected** event callback.