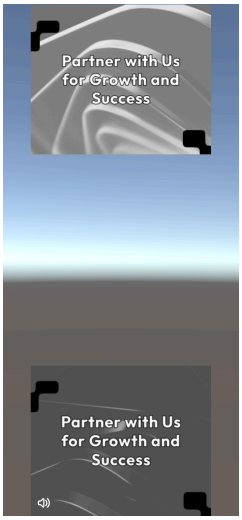
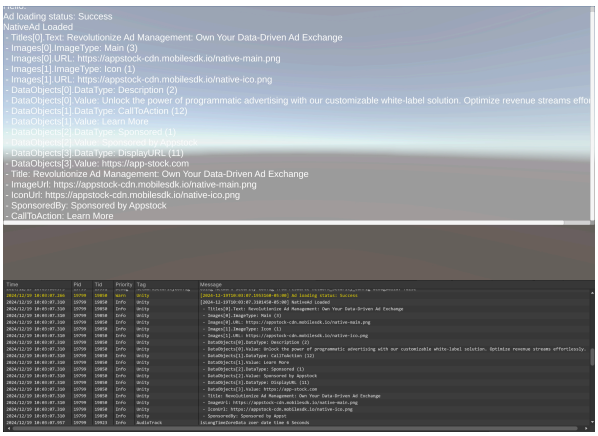
Overview

Appstock SDK SDK is a native library that monetizes iOS/Android applications.

"AppstockSDK for Unity" is a C# facade for using the native library within Unity.

Supported formats

Banner	Interstitial	Rewarded	Native
Overlay over the app anchored to screen edges.	Fullscreen ads to distract your users with.	Provide bonuses to users without making them pay actual money.	Let you (the developer) the freedom to seamlessly integrate them into your app.
			

Package Contents

This code is mostly a code library so you will rarely (if ever) need to browse through the assets.

Editor/

This folder contains an XML for External Dependency Manager for Unity (EDM4U).

See Installation: Native dependencies for more details.

Plugins/

This is where the code exists as a set of managed (DLL) and native (AAR/framework) plugins.

The meta files have already been configured for most cases.

See iOS Simulator on arm64 for more details.

Installation

Installing the package

Use the Package Manager window to Add UPM package.

Native dependencies

This package includes XML (at `Editor/NativeDependencies.xml`) for External Dependency Manager for Unity (EDM4U).

There are declared the versions of (AppstockSDK) native libraries to pull from CocoaPods / Maven.

If you already have EDM4U package installed, it should handle the rest.

Otherwise, you would need to manually (or via script) add these iOS/Android dependencies to the project upon build/export.

In case of any breaking changes within newer versions of native libraries, try updating this plugin package, or downgrade those dependencies back, down to the exact minimal version specified in XML.

Requirements

This package was developed with:

IDE	version
Unity	`2021.3.45f1` and `6000.0.29f1`
Xcode	`16.1`
Android Studio	`2024.2.1`

Which entails the following:

Build parameter	value	Notes
Minimal iOS target	`12.0`	-
Android Minimal SDK	`22`	-
Android Compile SDK	`34`	does not restrict from targeting higher
Swift	`6`	-
Java	`17`	-
C#	`9.0`	-
.NET Standard	`2.1`	see .NET profile support

Unity 2021/2022 Android issues

> **TIP**
>
> You are strongly advised to use Unity 6 for Android builds as everything works fine there.

We have encountered an error containing the following lines:

```
Execution failed for task ':launcher:mergeExtDexDebug'.

> Could not resolve all files for configuration ':launcher:debugRuntimeClasspath'.

> Failed to transform (...).aar
```

This is similar to the issues described in

- Gradle build issues for Android API SDK 35 in Unity 2022.3LTS
- Unable to Create Builds for Android SDK 35 with Unity 2022.3.40

despite not being directly related to Android 35.

iOS blocks insecure requests

Native SDK binaries might attempt to access ad exchange servers via "insecure" HTTP (rather than HTTPS), which can fail due to iOS feature -- see Preventing Insecure Network Connections.

To allow such communication in your app, `Info.plist` should be modified.

For more info see:

- App Transport Security has blocked a cleartext HTTP (http://) resource load since it is insecure.

Here is a modified script that can be put inside `Editor` folder to automate this task.

```
using UnityEditor;
using UnityEditor.Build;
using System.IO;
using UnityEditor.Build.Reporting;
using UnityEditor.iOS.Xcode;

// adds an ATS exception domain to the Info.plist
public class InfoPlistUpdater : IPostprocessBuildWithReport
{
    public int callbackOrder { get { return 0; } }

    public void OnPostprocessBuild(BuildReport report)
    {
        BuildTarget buildTarget = report.summary.platform;
        string pathToBuiltProject = report.summary.outputPath;

        if (buildTarget == BuildTarget.iOS)
        {
            // Get plist
            string plistPath = pathToBuiltProject + "/Info.plist";
            PlistDocument plist = new PlistDocument();
            plist.ReadFromString(File.ReadAllText(plistPath));

            PlistElementDict allowsDict =
plist.root.CreateDict("NSAppTransportSecurity");

            allowsDict.SetBoolean("NSAllowsArbitraryLoads", true);

            PlistElementDict exceptionsDict =
allowsDict.CreateDict("NSExceptionDomains");

            PlistElementDict domainDict = exceptionsDict.CreateDict("mobilesdk.io");
            domainDict.SetBoolean("NSExceptionAllowsInsecureHTTPLoads", true);
```

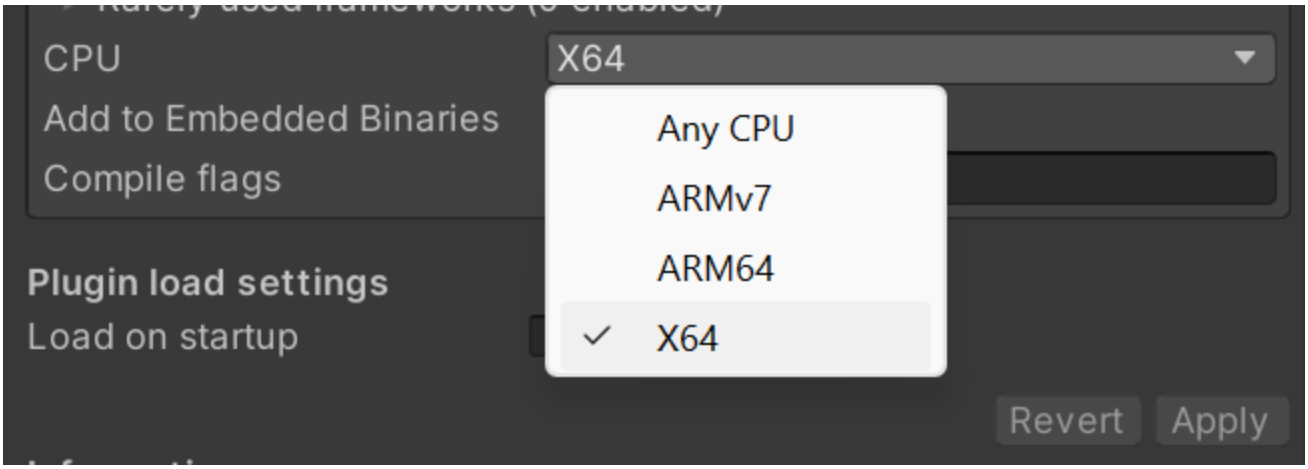
```
            domainDict.SetBoolean("NSIncludesSubdomains", true);

            // Write to file
            File.WriteAllText(plistPath, plist.WriteToString());
        }
    }
}
```

iOS Simulator on arm64

The package has two versions of native plugin (in `Plugins/iOS`):

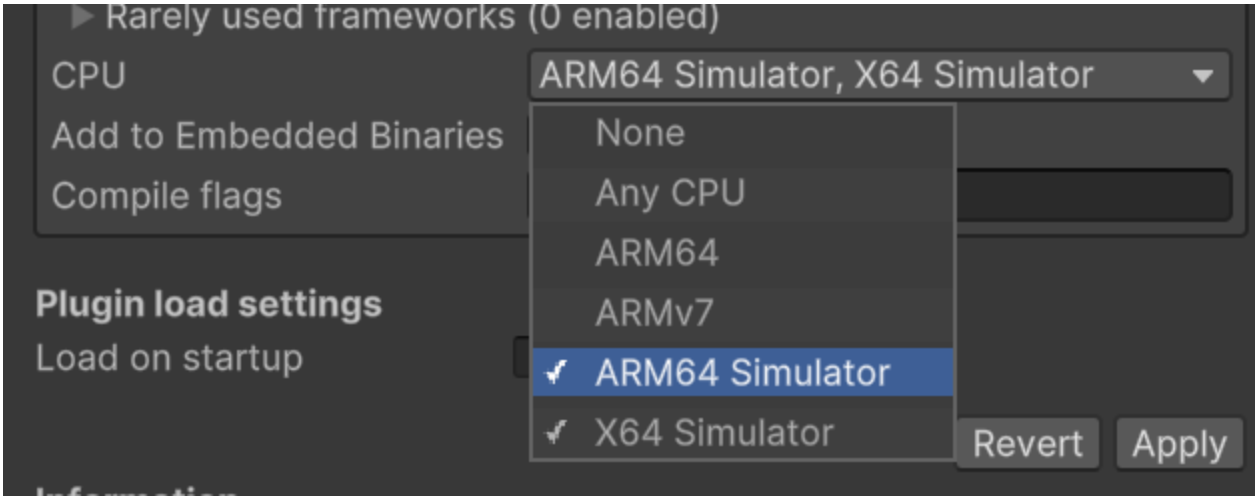
location	build destination	arhs	meta settings
AppstockSDKForUnity.framework	`generic/platform=iOS`	arm64	ARM64
AppstockSDKForUnity_Sim.framework	`generic/platform=iOS Simulator`	x64, arm64	X64

The simulator version is only marked as x64 compatible due to Unity 2021.3 LTS not having arm64 simulator flag in serialization version 2 of plugin meta files that it uses.



If you are using Unity 6, `X64` translates to `X64 Simulator`.

You may want to tick `ARM64 Simulator` as well and apply the changes once the package is imported.



SDK Initialization

> **TIP**
>
> Do this once, before attempting to load any ads.

Call InitializeSdk and pass your partner key.

You might find it useful to wrap this call in a dedicated MonoBehaviour to be invoked from Start -- *(see Order of execution for event functions)* -- once attached to some object on the scene.

"Banner" sample provides such a demo:

```
using System;
using AppstockSDK.Api;
using UnityEngine;

namespace AppstockSDK.Demo.Banner
{
    public class SdkInitializer : MonoBehaviour
    {
        // Start is called before the first frame update
        private void Start()
        {
            Debug.Log($"[{DateTime.Now:O}] Attempting to init SDK...");
            Appstock.InitializeSdk("appstock-demo");
        }
    }
}
```

Banner Ad

> **NOTE**
>
> Make sure you've done SDK Initialization before attempting to load any ads.

Use BannerAd to load and display a banner.

> **TIP**
>
> You can find a functional code in "Banner" sample.

Declare variable

BannerAd implements IDisposable. Calling Dispose() immediately destroys the respective View -- i.e. removes the banner from the screen.

> **WARNING**
>
> Losing the reference to BannerAd instance without calling Dispose() may result in a banner being stuck on screen for the remainder of the app's lifecycle.

Thus you must keep a reference in some relatively persistent variable.

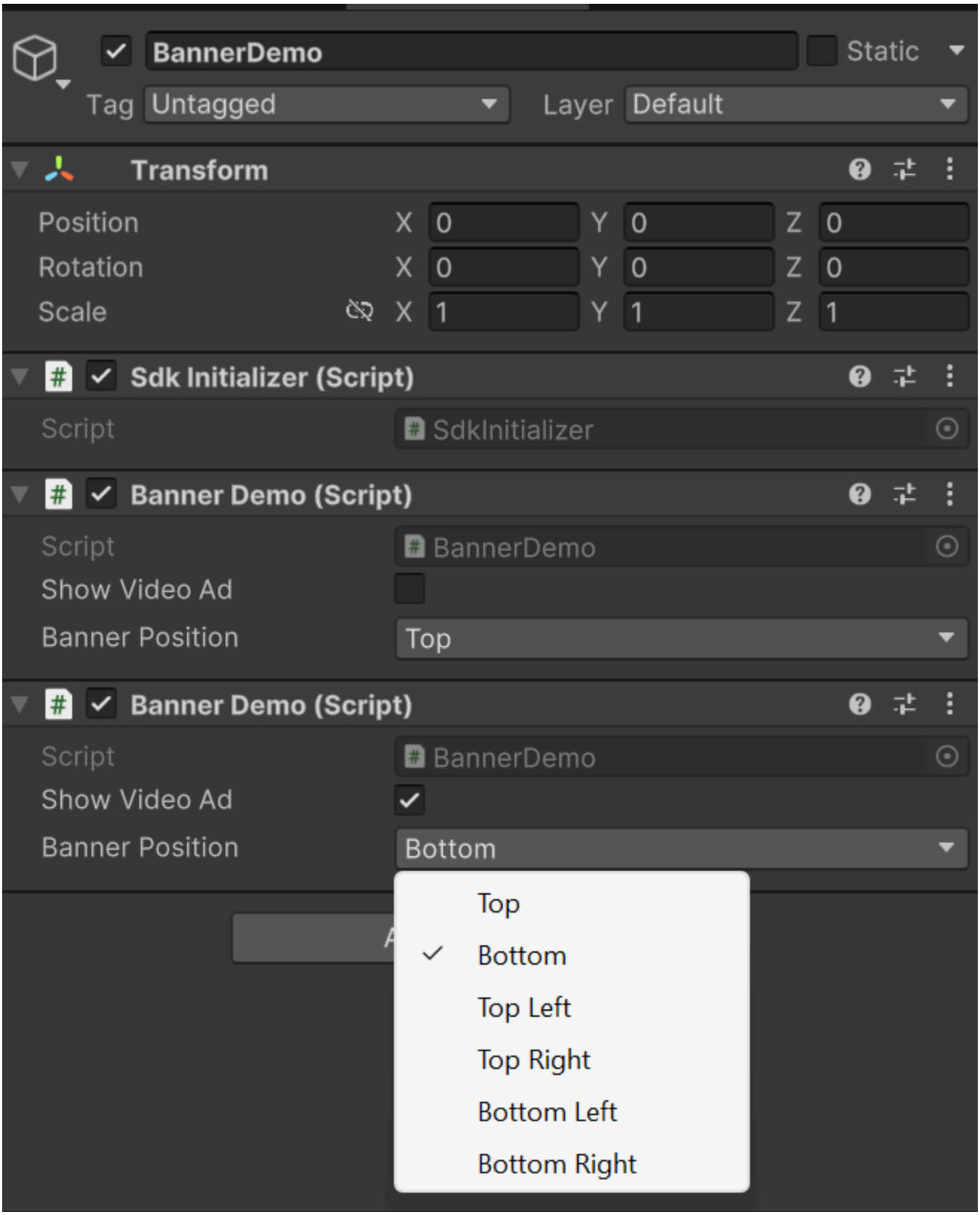
For example, an i-var on MonoBehaviour.

```
using System;
using System.Collections;
using AppstockSDK.Api;
using UnityEngine;

#nullable enable

namespace AppstockSDK.Demo.Banner
{
    [RequireComponent(typeof(SdkInitializer))]
    public class BannerDemo : MonoBehaviour
    {
        [SerializeField] private bool showVideoAd;
        public AnchoredAdPosition bannerPosition;
```

```
        private IBannerAd? _adUnit;
```

For info on SdkInitializer component see SDK Initialization.

The other two properties -- `showVideoAd` and `bannerPosition` let us customize the position and content of the banner in editor (on scene).



Instantiation

Pass desired AdSize into the constructor followed by other parameters.

```
_adUnit = showVideoAd
    ? new BannerAd(new(320, 250))
```

```
    {
        PlacementID = "8",
        AdUnitFormat = AdFormat.Video,
        AnchoredPosition = bannerPosition,
        AdPosition = bannerPosition.ToAdPosition(),
    }
    : new BannerAd(new(320, 250))
    {
        PlacementID = "4",
        AnchoredPosition = bannerPosition,
        AdPosition = bannerPosition.ToAdPosition(),
    };
```

Loading the Ad

Once all properties are set, call LoadAd().

```
_adUnit.LoadAd();
```

Listening to events

BannerAd exposes few events that may or may not be of interest to you.

Relatively important may be OnAdFailed.

In the sample the script subscribes to all events and logs whenever they are fired.

```
SubscribeToEvents(_adUnit);
_adUnit.LoadAd();


private void SubscribeToEvents(IBannerAd adUnit)
{
    adUnit.OnAdLoaded += OnAdUnitLoaded;
    adUnit.OnAdFailed += OnAdUnitFailed;
    adUnit.OnAdClicked += OnAdUnitClicked;
    adUnit.OnAdClosed += OnAdUnitClosed;
}


private void OnAdUnitLoaded(AdInfo? adInfo)
{
    Debug.Log($"[{DateTime.Now:O}] (video: {showVideoAd}) Ad Loaded. Info:
{adInfo?.ToString() ?? "(null)"}.");
```

```
}

private void OnAdUnitFailed(AdError? adError)
{
    Debug.LogError($"[{DateTime.Now:O}] (video: {showVideoAd}) Ad
Failed: {adError?.Message}.");
}

private void OnAdUnitClicked() => Debug.Log($"[{DateTime.Now:O}] (video:
{showVideoAd}) Ad Clicked.");

private void OnAdUnitClosed()
{
    Debug.Log($"[{DateTime.Now:O}] (video: {showVideoAd}) Ad Closed.");
}
```

Hiding the banner

Use Hide() / Show() to (temporarily?) change banner's visibility without destroying the view completely.

```
public void SetHidden(bool hidden)
{
    if (_adUnit is null)
    {
        return;
    }
    if (hidden)
    {
        _adUnit.Hide();
    }
    else
    {
        _adUnit.Show();
    }
}
```

Destroying the banner

Unsubscribe from events, call Dispose() and clear the reference i-var.

```
private void UnsubscribeFromEvents(IBannerAd adUnit)
{
    adUnit.OnAdLoaded -= OnAdUnitLoaded;
```

```
    adUnit.OnAdFailed -= OnAdUnitFailed;
    adUnit.OnAdClicked -= OnAdUnitClicked;
    adUnit.OnAdClosed -= OnAdUnitClosed;
}


public void DropAdUnit()
{
    if (_adUnit is null)
    {
        return;
    }
    UnsubscribeFromEvents(_adUnit);
    _adUnit.Dispose();
    _adUnit = null;
}
```

Interstitial Ad

> **NOTE**
>
> Make sure you've done SDK Initialization before attempting to load any ads.

Use InterstitialAd to load and display an interstitial.

> **TIP**
>
> You can find a functional code in "Interstitial" sample.

Declare variable

InterstitialAd implements IDisposable.

Dispose() severs the event chain (your callbacks will no longer be called) and lets the engine free the resources (e.g. held by underlying AndroidJavaObject).

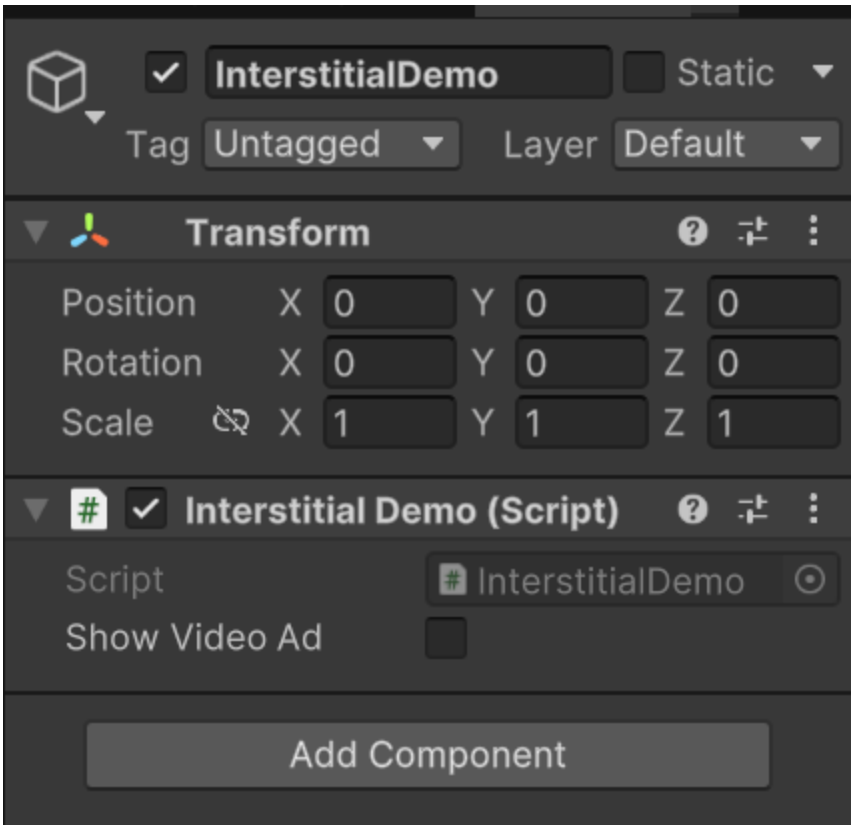
Calling Dispose() before losing the reference might have performance benefits.

Thus we shall keep a reference in some relatively persistent variable.

For example, an i-var on MonoBehaviour.

```
using System;
using System.Collections;
using AppstockSDK.Api;
using UnityEngine;

#nullable enable

namespace AppstockSDK.Demo.Interstitial
{
    public class InterstitialDemo : MonoBehaviour
    {
        [SerializeField] private bool showVideoAd;
        private IInterstitialAd? _adUnit;
```

The other property -- `showVideoAd` lets us customize the content of the interstitial in editor (on scene).



Instantiation

Call the constructor and set other parameters.

```
_adUnit = showVideoAd
    ? new InterstitialAd()
    {
        PlacementID = "7",
        AdUnitFormats = new []{ AdFormat.Video },
    }
    : new InterstitialAd()
    {
        PlacementID = "5",
    };
```

Loading the Ad

Once all properties are set, call LoadAd().

```
_adUnit.LoadAd();
```

Listening to events

InterstitialAd exposes few events.

OnAdLoaded is useful to know you can call Show(Action?).

OnAdFailed and OnAdClosed are useful to dispose of the no-longer-useful ad unit.

In the sample the script subscribes to all events and logs whenever they are fired.

```
SubscribeToEvents(_adUnit);
_adUnit.LoadAd();


private void SubscribeToEvents(IInterstitialAd adUnit)
{
    adUnit.OnAdLoaded += OnAdUnitLoaded;
    adUnit.OnAdDisplayed += OnAdUnitDisplayed;
    adUnit.OnAdFailed += OnAdUnitFailed;
    adUnit.OnAdClicked += OnAdUnitClicked;
    adUnit.OnAdClosed += OnAdUnitClosed;
}


private void OnAdUnitLoaded(AdInfo? adInfo)
{
    Debug.Log($"[{DateTime.Now:O}] Ad Loaded. Info: {adInfo?.ToString() ??
"(null)"}.");
    _adUnit?.Show();
}

private void OnAdUnitFailed(AdError? adError)
{
    Debug.LogError($"[{DateTime.Now:O}] Ad Failed: {adError?.Message}.");
    DropAdUnit();
}

private void OnAdUnitDisplayed() => Debug.Log($"[{DateTime.Now:O}] Ad Displayed.");
private void OnAdUnitClicked() => Debug.Log($"[{DateTime.Now:O}] Ad Clicked.");

private void OnAdUnitClosed()
{
    Debug.Log($"[{DateTime.Now:O}] Ad Closed.");
    DropAdUnit();
}
```

Destroying the ad unit

Unsubscribe from events, call Dispose() and clear the reference i-var.

```
private void UnsubscribeFromEvents(IInterstitialAd adUnit)
{
    adUnit.OnAdLoaded -= OnAdUnitLoaded;
    adUnit.OnAdDisplayed -= OnAdUnitDisplayed;
    adUnit.OnAdFailed -= OnAdUnitFailed;
    adUnit.OnAdClicked -= OnAdUnitClicked;
    adUnit.OnAdClosed -= OnAdUnitClosed;
}


private void DropAdUnit()
{
    if (_adUnit is null)
    {
        return;
    }
    UnsubscribeFromEvents(_adUnit);
    _adUnit.Dispose();
    _adUnit = null;
}
```

Rewarded Ad

> **NOTE**
>
> Make sure you've done SDK Initialization before attempting to load any ads.

Use RewardedAd to load and display an rewarded.

> **TIP**
>
> You can find a functional code in "Rewarded" sample.

Declare variable

RewardedAd implements IDisposable.

Dispose() severs the event chain (your callbacks will no longer be called) and lets the engine free the resources (e.g. held by underlying AndroidJavaObject).

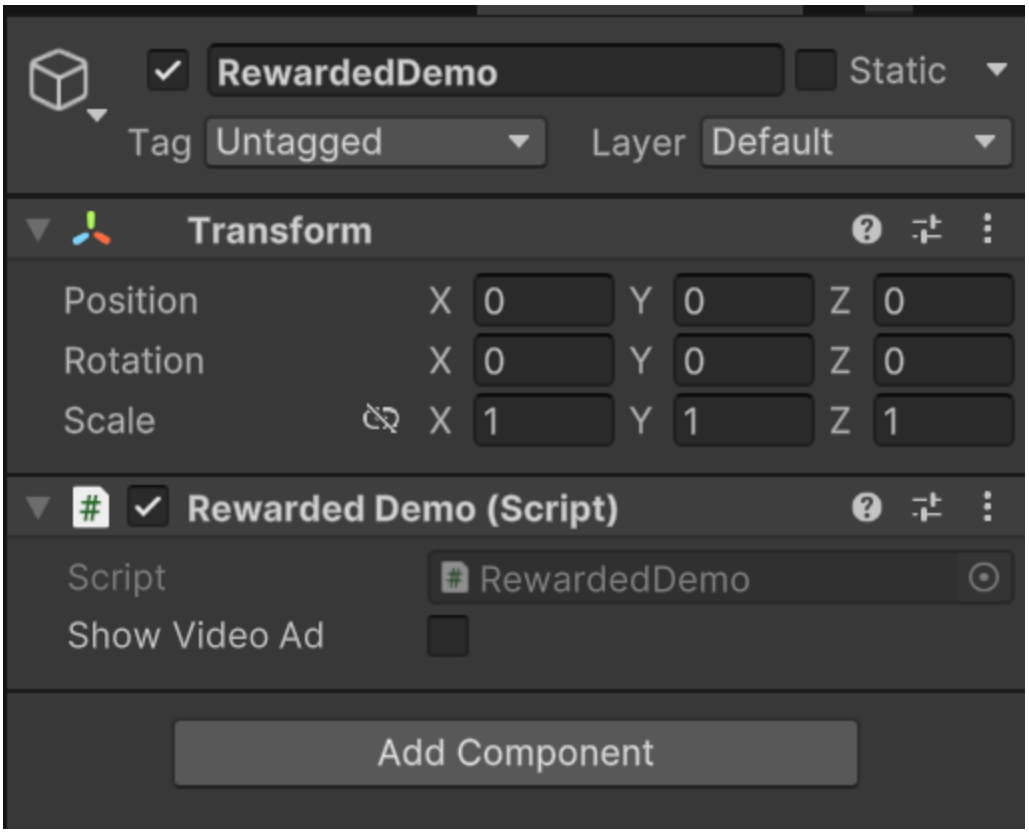
Calling Dispose() before losing the reference might have performance benefits.

Thus we shall keep a reference in some relatively persistent variable.

For example, an i-var on MonoBehaviour.

```
using System;
using System.Collections;
using AppstockSDK.Api;
using UnityEngine;

#nullable enable

namespace AppstockSDK.Demo.Rewarded
{
    public class RewardedDemo : MonoBehaviour
    {
        [SerializeField] private bool showVideoAd;
        private IRewardedAd? _adUnit;
```

The other property -- `showVideoAd` lets us customize the content of the rewarded in editor (on scene).



Instantiation

Call the constructor and set other parameters.

```
_adUnit = new RewardedAd()
{
    PlacementID = showVideoAd ? "16" : "12",
};
```

Loading the Ad

Once all properties are set, call LoadAd().

```
_adUnit.LoadAd();
```

Listening to events

RewardedAd exposes few events.

OnAdLoaded is useful to know you can call Show(Action?).

OnAdFailed and OnAdClosed are useful to dispose of the no-longer-useful ad unit.

OnReward is important to award users the bonuses.

In the sample the script subscribes to all events and logs whenever they are fired.

```
SubscribeToEvents(_adUnit);
_adUnit.LoadAd();


private void SubscribeToEvents(IRewardedAd adUnit)
{
    adUnit.OnAdLoaded += OnAdUnitLoaded;
    adUnit.OnAdDisplayed += OnAdUnitDisplayed;
    adUnit.OnAdFailed += OnAdUnitFailed;
    adUnit.OnAdClicked += OnAdUnitClicked;
    adUnit.OnAdClosed += OnAdUnitClosed;
    adUnit.OnReward += OnAdUnitRewarded;
}


private void OnAdUnitLoaded(AdInfo? adInfo)
{
    Debug.Log($"[{DateTime.Now:O}] Ad Loaded. Info: {adInfo?.ToString() ??
"(null)"}.");
    _adUnit?.Show();
}

private void OnAdUnitFailed(AdError? adError)
{
    Debug.LogError($"[{DateTime.Now:O}] Ad Failed: {adError?.Message}.");
    DropAdUnit();
}

private void OnAdUnitDisplayed() => Debug.Log($"[{DateTime.Now:O}] Ad Displayed.");
private void OnAdUnitClicked() => Debug.Log($"[{DateTime.Now:O}] Ad Clicked.");

private void OnAdUnitClosed()
{
    Debug.Log($"[{DateTime.Now:O}] Ad Closed.");
    DropAdUnit();
}
private void OnAdUnitRewarded(AdReward? adReward) => Debug.Log($"[{DateTime.Now:O}]
Ad Rewarded: {adReward}.");
```

Destroying the ad unit

Unsubscribe from events, call Dispose() and clear the reference i-var.

```
private void UnsubscribeFromEvents(IRewardedAd adUnit)
{
    adUnit.OnAdLoaded -= OnAdUnitLoaded;
    adUnit.OnAdDisplayed -= OnAdUnitDisplayed;
    adUnit.OnAdFailed -= OnAdUnitFailed;
    adUnit.OnAdClicked -= OnAdUnitClicked;
    adUnit.OnAdClosed -= OnAdUnitClosed;
    adUnit.OnReward -= OnAdUnitRewarded;
}


private void DropAdUnit()
{
    if (_adUnit is null)
    {
        return;
    }
    UnsubscribeFromEvents(_adUnit);
    _adUnit.Dispose();
    _adUnit = null;
}
```

Introduction to Native Ads

What?

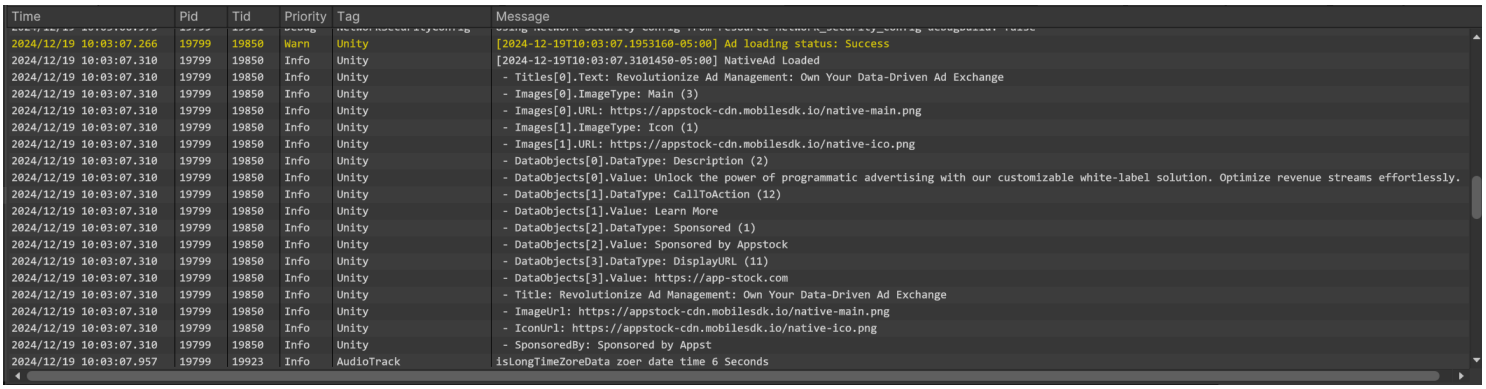
Here is an excerpt from OpenRTB Dynamic Native Ads API Specification Version 1.2.

> Native advertising is an online advertising method in which the advertiser attempts to gain attention by providing content in the context of the user's experience. Native ad formats match both the form and function of the user experience in which it is placed. This is in contrast to traditional banner or interstitials ads, which are displayed in a separate space of predefined and universal size, without regard to their surroundings.

What again?

The ads that have no 'predefined form', but rather a set of pieces you make look like a natural part of your application.

Here is an example of data you might receive:



Time	Pid	Tid	Priority	Tag	Message
2024/12/19 10:03:07.266	19799	19850	Warn	Unity	[2024-12-19T10:03:07.1953160-05:00] Ad loading status: Success
2024/12/19 10:03:07.310	19799	19850	Info	Unity	[2024-12-19T10:03:07.3101450-05:00] NativeAd Loaded
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- Titles[0].Text: Revolutionize Ad Management: Own Your Data-Driven Ad Exchange
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- Images[0].ImageType: Main (3)
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- Images[0].URL: https://appstock-cdn.mobilesdk.io/native-main.png
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- Images[1].ImageType: Icon (1)
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- Images[1].URL: https://appstock-cdn.mobilesdk.io/native-ico.png
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- DataObjects[0].DataType: Description (2)
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- DataObjects[0].Value: Unlock the power of programmatic advertising with our customizable white-label solution. Optimize revenue streams effortlessly.
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- DataObjects[1].DataType: CallToAction (12)
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- DataObjects[1].Value: Learn More
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- DataObjects[2].DataType: Sponsored (1)
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- DataObjects[2].Value: Sponsored by Appstock
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- DataObjects[3].DataType: DisplayURL (11)
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- DataObjects[3].Value: https://app-stock.com
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- Title: Revolutionize Ad Management: Own Your Data-Driven Ad Exchange
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- ImageUrl: https://appstock-cdn.mobilesdk.io/native-main.png
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- IconUrl: https://appstock-cdn.mobilesdk.io/native-ico.png
2024/12/19 10:03:07.310	19799	19850	Info	Unity	- SponsoredBy: Sponsored by Appst
2024/12/19 10:03:07.957	19799	19923	Info	AudioTrack	isLongTimeZoreData zoer date time 6 Seconds

As you can see, it can be given pretty much any reasonable shape.

Few examples:

- A message via in-app communication system.
  - chat/email/forum message.
  - post on a questing board.
  - entry within in-world 'news' feed.
- Content for in-world 2D/3D ad box.
  - If you have a city-like location within a game, placing a few ad boxes around might make it more realistic.
  - In a 2D platformer you might replaces some unreachable tiles with an ad container.

Why?

- The immersion is less disrupted:
  - The style of ad matches the rest of the app:
    - fonts
    - color scheme
    - layout (both locations and sizes of elements)
  - User input is not interrupted (compared to interstitials).
- You do not lose limited screen space (compared to banners).

How?

- Build a request describing what elements you can display.
  - see Building Native Ad Request
- Call LoadAd(Action<INativeAd?, AdError?>).
- Apply the content from INativeAd to the GameObject on scene.
  - see Applying Native Ad Response

Building Native Ad Request

> **NOTE**
>
> Make sure you've done SDK Initialization before attempting to load any ads.

Long story short

- Get a populated AdUnitData.
- Call an extension method BuildAdLoader(AdUnitData).

```
_nativeAdLoader = nativeConfig.BuildAdLoader();
_nativeAdLoader.LoadAd(OnAdLoadResult);
```

> **TIP**
>
> The resulting INativeAdLoader has LoadAd(Action<INativeAd?, AdError?>) you needed.
>
> Now you can get back to step 2 of How to do Native Ads.

NativeConfigPreset asset

AdUnitData has SerializableAttribute.

Which means you can have it edited via Unity Editor.

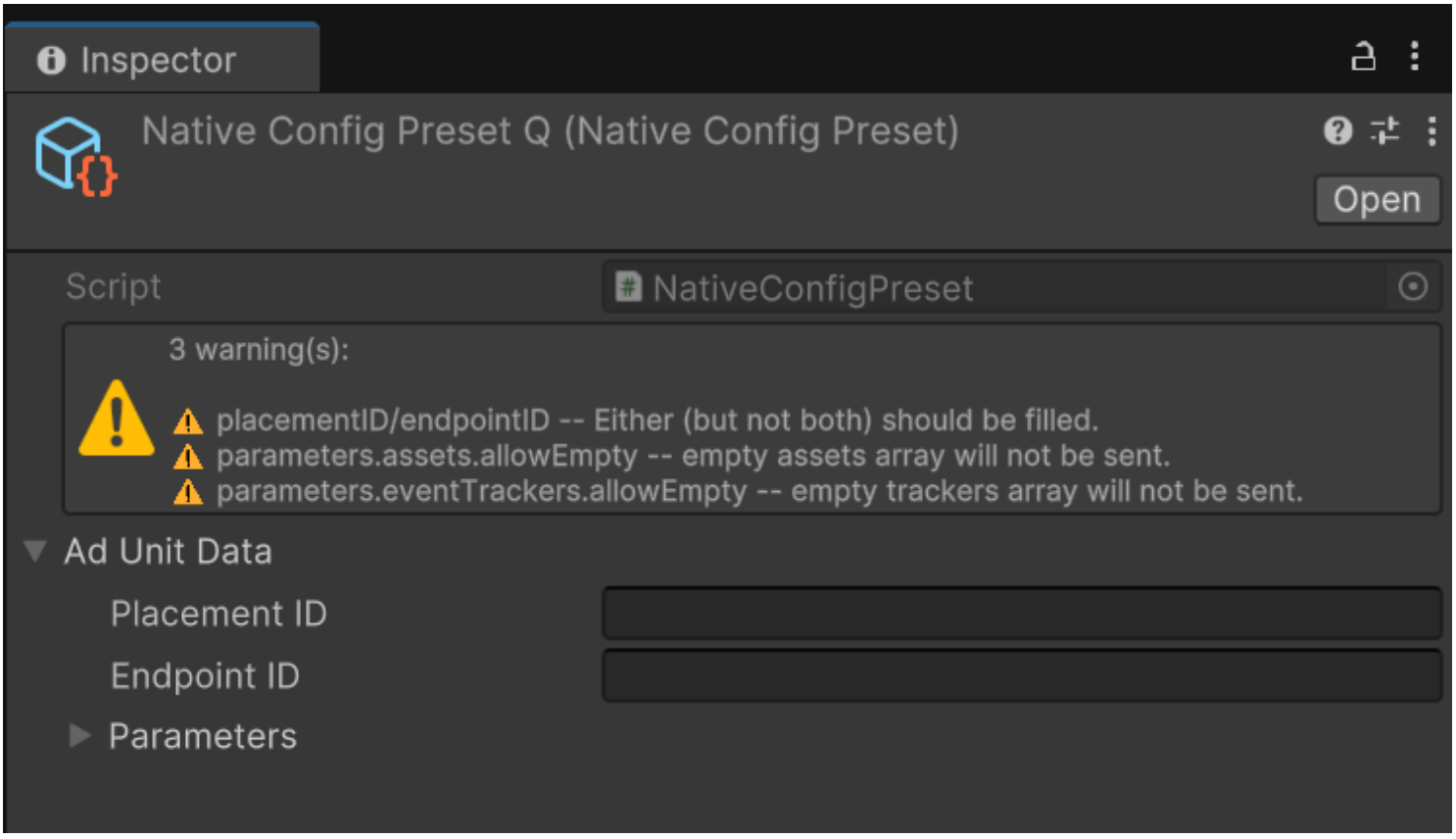
Most of the data is pretty static, so the logical thing to do is wrap AdUnitData into a ScriptableObject.

The "Native" sample has a `NativeConfigPreset` for that:

```
using System;
using AppstockSDK.Api;
using UnityEngine;
using AppstockSDK.Api.Native.Data.Request;


namespace AppstockSDK.Demo.Native
{
    [Serializable]
    [CreateAssetMenu(fileName = "NativeConfigPreset", menuName = "Appstock/Native
```

```
Config Preset")]
    public class NativeConfigPreset : ScriptableObject
    {
        public ConfigWarnings warnings;
        public AdUnitData adUnitData;

        public INativeAdLoader BuildAdLoader()
            => adUnitData.BuildAdLoader();


    }
}
```

Once imported, you can create assets of this class via respective menu entry (highlighted above).



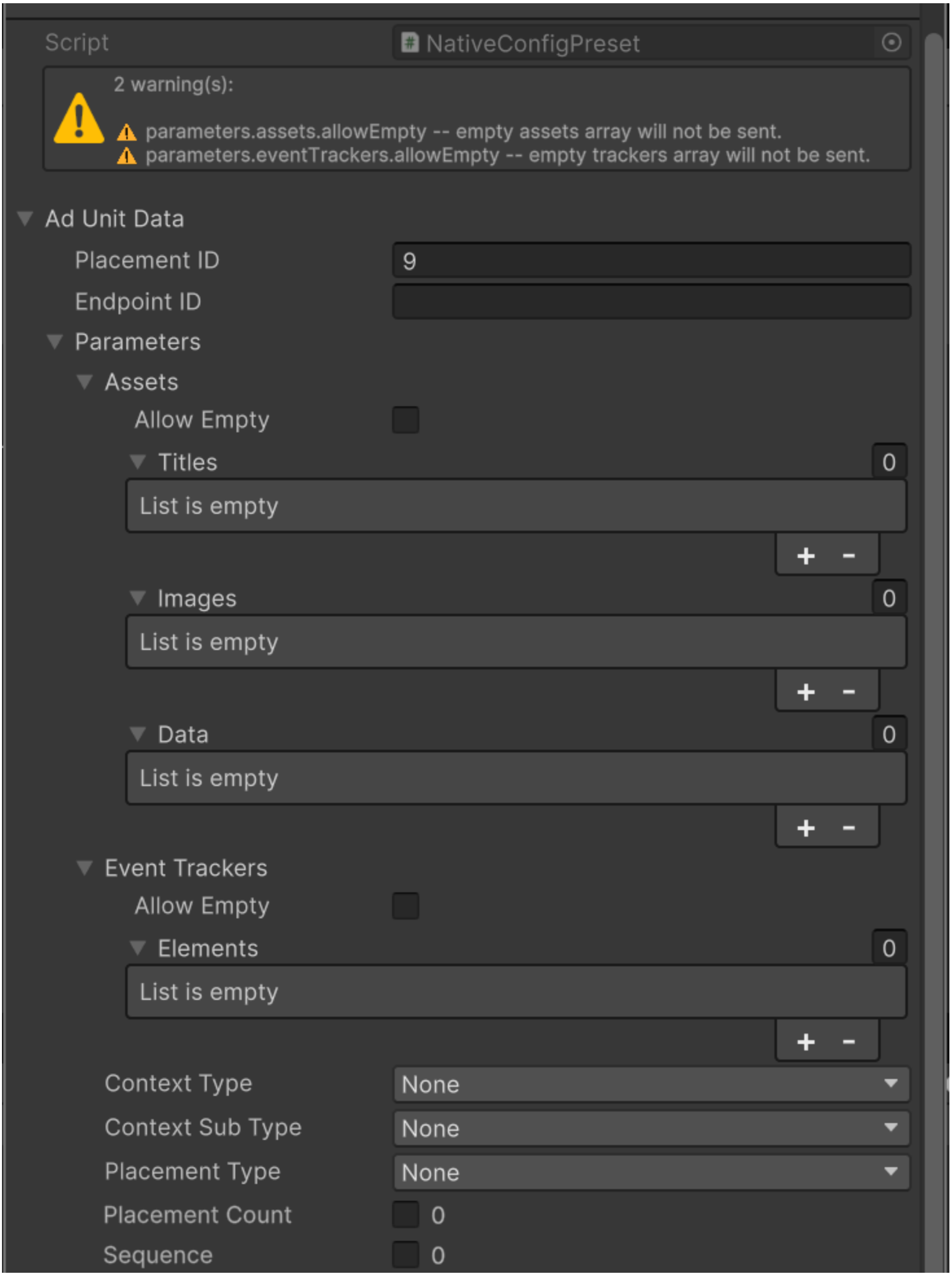
ConfigWarnings serves as an effectively-readonly container to display any errors directly in inspector during the editing process.

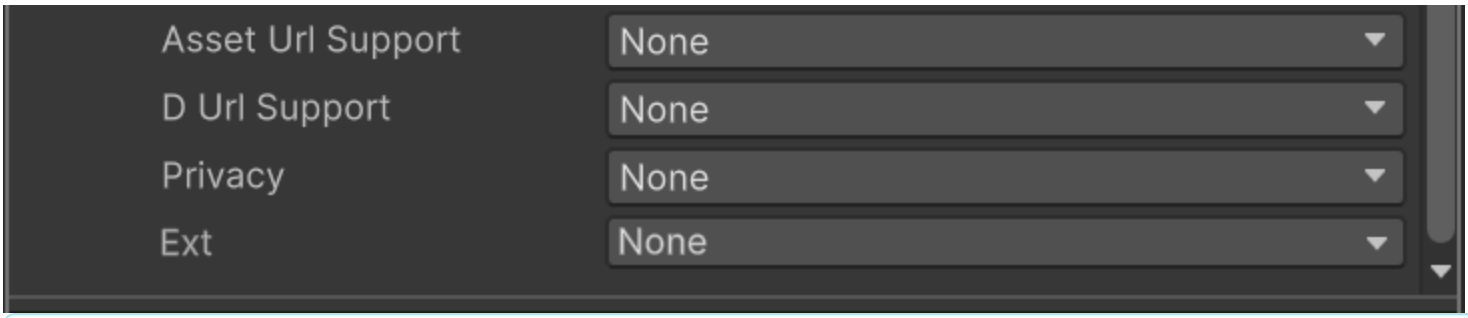
Placement/Endpoint ID

Fill in placementID or endpointID with a value generated on the Appstock SDK platform's UI.

Ad Parameters

Fill in the Parameters to the best of your abilities.

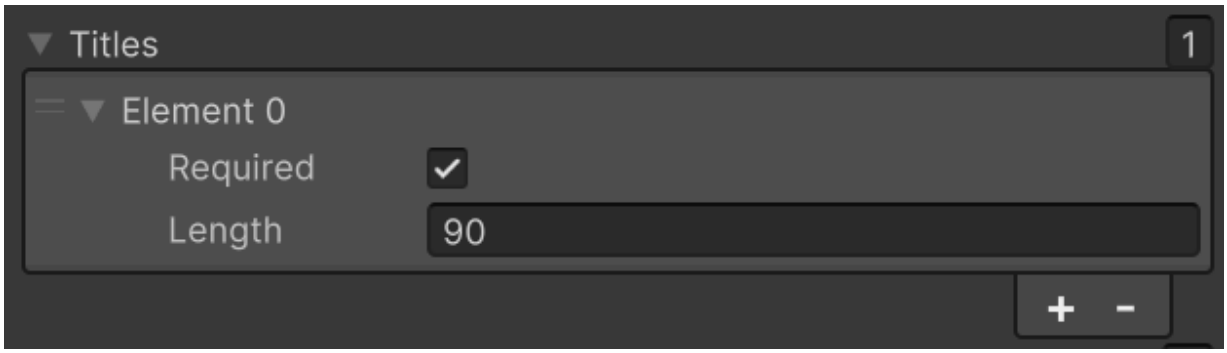

Script
NativeConfigPreset
2 warning(s):
parameters.assets.allowEmpty -- empty assets array will not be sent.
parameters.eventTrackers.allowEmpty -- empty trackers array will not be sent.
Ad Unit Data
Placement ID
9
Endpoint ID
Parameters
Assets
Allow Empty
Titles
0
List is empty
Images
0
List is empty
Data
0
List is empty
Event Trackers
Allow Empty
Elements
0
List is empty
Context Type
None
Context Sub Type
None
Placement Type
None
Placement Count
0
Sequence
0

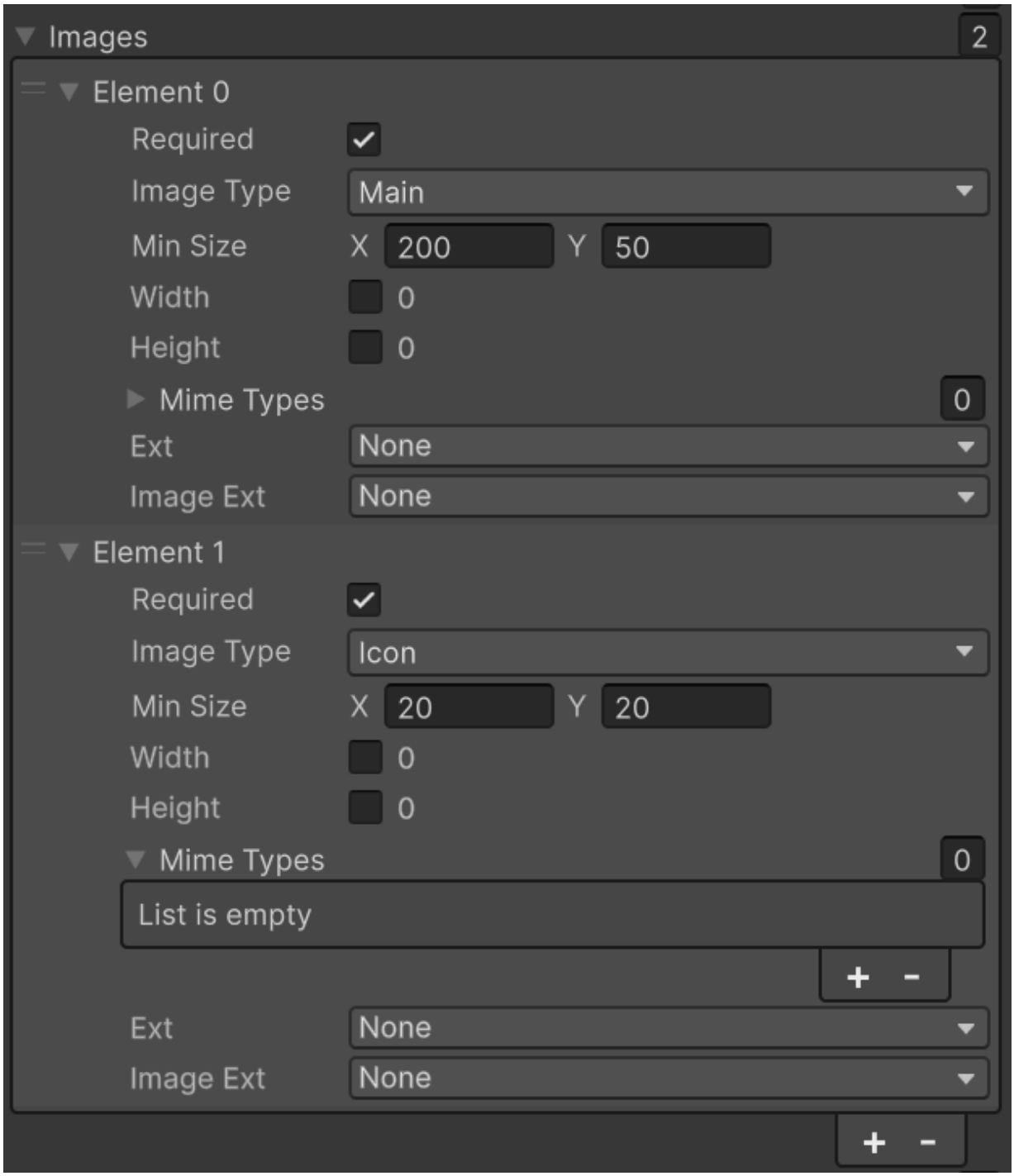


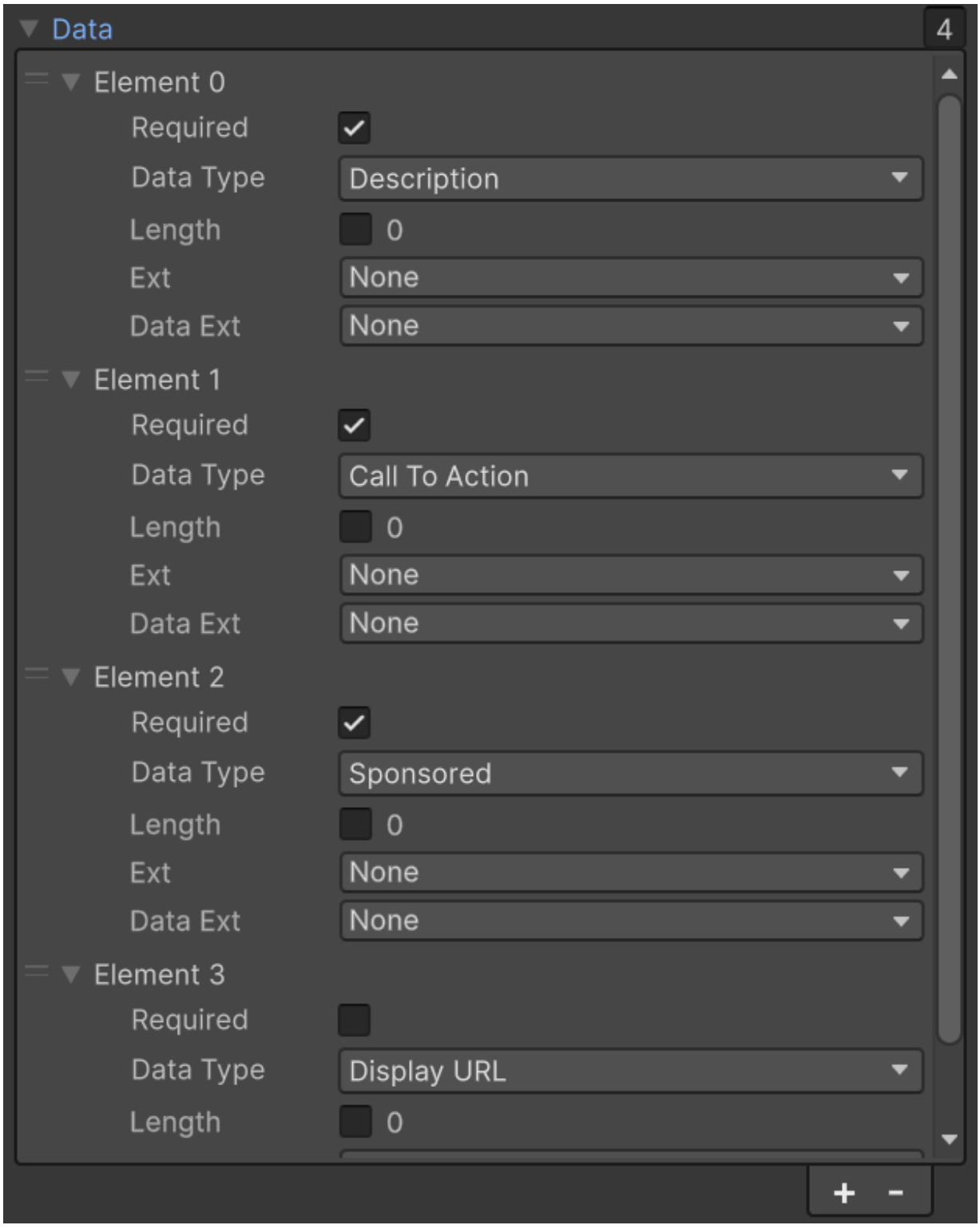
> **TIP**
>
> Both AdUnitData and Parameters are structs.
>
> This allows you to reuse a single `NativeConfigPreset` asset for different placements or contexts if needed.

Ad Assets

assets (of type Assets) has 3 homogeneous arrays for each type of assets:

Asset Type	Config Samples
AssetTitle[]	

Asset Type	Config Samples
Asset Image[]	

Asset Type	Config Samples
AssetData[]	

> **NOTE**
>
> If all of the array are null or empty, the combined `assets` field will not be written to in native library (by default) and the warning is displayed in inspector.
>
> allowEmpty lets you override this behaviour and intentionally pass the empty array down into native library (rather than keep default which most likely is `null`).

Ext Slots

Some data structures expose public fields of type ExtSlot.

See Ext Slot page for more info.

Using data from asset to load ad

> **TIP**
>
> You can find a functional code in "Native" sample.

Declaring variables

Expose `NativeConfigPreset` inspectable variable on the relevant MonoBehaviour.

Prepare INativeAdLoader private variable.

```
#nullable enable

namespace AppstockSDK.Demo.Native
{
    public class NativeDemo : MonoBehaviour
    {
        [SerializeField] private NativeConfigPreset? nativeConfig;
        [SerializeField] private TMP_Text? text;
        private INativeAdLoader? _nativeAdLoader;
        private INativeAd? _nativeAd;
```

`text` is the sample is used for displaying the content of INativeAd. It is for illustration purposes only.

`_nativeAd` will be used later (once received).

Loading the ad

Build the ad loader.

```
_nativeAdLoader = nativeConfig.BuildAdLoader();
```

In this case the convenience method by scriptable object was used.

```
public INativeAdLoader BuildAdLoader()
    => adUnitData.BuildAdLoader();
```

If you need to adjust the data before building the loader, use a local variable, e.g.:

```
AdUnitData dataCopy = nativeConfig.adUnitData;
dataCopy.placementID = "99";
dataCopy.parameters.sequence = 17;
_nativeAdLoader = dataCopy.BuildAdLoader();
```

Load the ad.

```
_nativeAdLoader.LoadAd(OnAdLoadResult);
```

The signature of `OnAdLoadResult` callback method is:

```
private void OnAdLoadResult(INativeAd? nativeAd, AdError? status)
```

If it failed to load, `nativeAd` will be null and `status` may contain an error message.

> **WARNING**
>
> Do not check if `status` is null to determine if ad has loaded.
>
> `status` may contain success message.

If the ad did successfully load, `nativeAd` will be not null.

> **TIP**
>
> For applying the received data please continue reading Applying Native Ad Response.

Applying Native Ad Response

This page explains how to make use of INativeAd.

On how to build the request and load the ad see Building Native Ad Request.

> **TIP**
>
> You can find a functional code in "Native" sample.

Where we left off

As was mentioned on the previous page, we have 3 (still relevant) i-vars.

```
[SerializeField] private TMP_Text? text;
private INativeAdLoader? _nativeAdLoader;
private INativeAd? _nativeAd;
```

We will use `text` to dump all the data from our INativeAd -- for illustration purposes.

In the real app you will populate your own GUI (or 2D/3D) components.

`_nativeAd` still has the default value (null).

`_nativeAdLoader` still has the loader that was used to start loading the ad.

```
_nativeAdLoader.LoadAd(OnAdLoadResult);
```

Logging helper

`LogToUI` method is a simple wrapper to add messages both to UI and debug console:

```
private void LogToUI(string message, LogType logType = LogType.Log)
{
    if (text != null)
    {
        text.text += "\n" + message;
    }
    Debug.LogFormat(logType, LogOption.None, gameObject, "[{0:O}] {1}",
DateTime.Now, message);
}
```

Ad Load Callback

First, check if we did get the actual ad.

```
_nativeAdLoader = null;
if (nativeAd is null)
{
    LogToUI($"Failed to load ad: {status?.Message}", LogType.Error);
    return;
}
```

Save a reference to the ad.

```
_nativeAd = nativeAd;
```

> **NOTE**
>
> INativeAd extends IDisposable. Calling Dispose() releases native objects and stops impression tracking.
>
> You should dispose of native ad once done, but preserve a reference until that point.

Event handling

Subscribing to events

```
SubscribeToEvents(nativeAd);


private void SubscribeToEvents(INativeAd nativeAd)
{
    nativeAd.OnAdClicked += OnAdClicked;
    nativeAd.OnAdImpression += OnAdImpression;
    nativeAd.OnAdExpired += OnAdExpired;
}
```

Expiration handling

```
private void OnAdExpired()
{
    LogToUI("Ad Expired.");
    using (_nativeAd)
    {
        if (_nativeAd is not null)
        {
            UnsubscribeFromEvents(_nativeAd);
            _nativeAd.Dispose();
            _nativeAd = null;
        }
    }
}
```

Unsubscribing

```
private void UnsubscribeFromEvents(INativeAd nativeAd)
{
    nativeAd.OnAdClicked -= OnAdClicked;
    nativeAd.OnAdImpression -= OnAdImpression;
    nativeAd.OnAdExpired -= OnAdExpired;
}
```

Extracting all the things

```
private void UnsubscribeFromEvents(INativeAd nativeAd)
{
    nativeAd.OnAdClicked -= OnAdClicked;
    nativeAd.OnAdImpression -= OnAdImpression;
```

```
    nativeAd.OnAdExpired -= OnAdExpired;
}
```

Iteration over typed asset collections

Use foreach to iterate over Titles:

```
foreach (var nextTitle in nativeAd.Titles)
{


}
```

> **NOTE**
>
> ITitleContent, IImageContent and IDataContent implement IDisposable.
>
> Calling Dispose() allows the engine to free native resources sooner on some platforms.
>
> See "Best practice" / "Garbage collection" section of Call Java and Kotlin plug-in code from C# scripts for details.

Since we no longer need titles, once we extract the strings, we will use using as recommended.

```
foreach (var nextTitle in nativeAd.Titles)
{
    using (nextTitle)
    {


    }
}
```

Similar loops can be made for Images / DataObjects.

Inside the loop(s) you can access all the properties exposed by (respective) asset type(s):

- ITitleContent
  - Text
- IImageContent
  - ImageType

    - URL
- IDataContent
    - DataType
    - Value

You might want to use a switch over ImageContentType / DataContentType to determine which of your components (on scene) should handle curent ad asset.

Convenience accessors

Few direct access string properties can be helpful to avoid filtering through all the assets (if you know what you are looking for).

- Title
- ImageUrl
- IconUrl
- SponsoredBy
- CallToAction

Enabling impression tracking

Call RegisterView(GameObject, IEnumerable<GameObject>?).

As a `container` pass the most relevant ancestor (Transform-wise) of ad elements.

As a `clickableObjects` -- GameObjects with attached colliders (Collider / Collider2D) or Selectable components.

```
nativeAd.RegisterView(gameObject, null);
```

Disposing of native ad

```
UnsubscribeFromEvents(_nativeAd);
_nativeAd.Dispose();
_nativeAd = null;
```

Used in Expiration handling but may also be needed when GameObject gets destroyed.

SDK Configuration

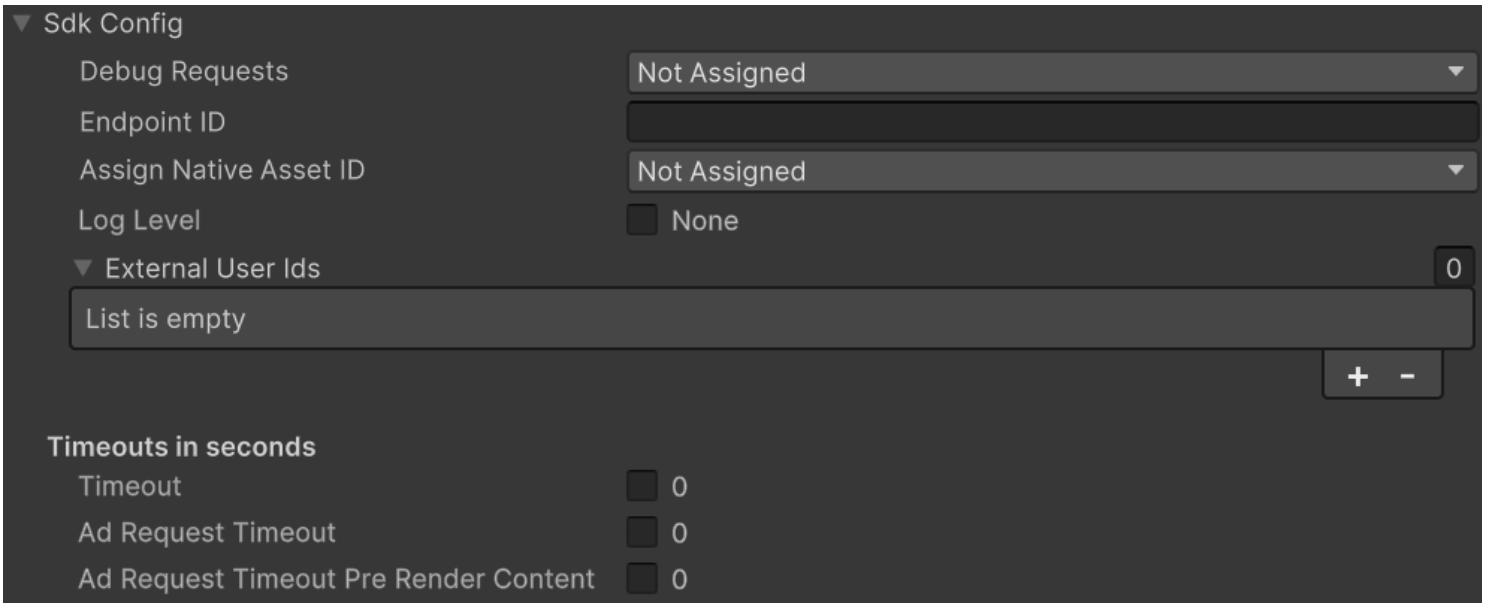
ISdkProxy provides few global configuration options.

Sdk property of Appstock can be used to get the instance of ISdkProxy.

> **WARNING**
>
> ISdkProxy::ExternalUserIds and SdkConfig::externalUserIds expose a potentially-deprecated property of native SDK library.
>
> ITargetingProxy::ExternalUserIds / TargetingData::externalUserIds should be used instead in most cases.
>
> See Improving targeting / External User IDs.

Data-driven approach

SdkConfig is a data container -- with a SerializableAttribute -- for specifying settings via Unity Editor.



Applying the config

- Expose SdkConfig variable on your MonoBehaviour
  - Or put it inside some ScriptableObject and assign through that.
- Call extension method Apply(ISdkProxy, SdkConfig) on Appstock.Sdk and pass your SdkConfig.

```
#nullable enable

namespace AppstockSDK.DevApp
{
    public class TargetingDemo : MonoBehaviour
    {
        public SdkConfig sdkConfig = new();
        public TargetingData targetingData = new();

        // Start is called before the first frame update
        private IEnumerator Start()
        {
            Debug.Log($"[{DateTime.Now:O}] Applying config...");
            Appstock.Sdk.Apply(sdkConfig);
```

> **TIP**
>
> You can find a complete code in "ConfigAndTargeting" sample.

Taking a config snapshot

SdkConfigSnapshot can retrieve all readable properties from ISdkProxy.

```
var sdkConfigSnapshot = new SdkConfigSnapshot(Appstock.Sdk);
```

It can be serialized and logged into console

```
var snapshotJson = JsonUtility.ToJson(sdkConfigSnapshot, prettyPrint: true);
Debug.Log($"(snapshot) {snapshotJson}");
```

or compared to the expected config -- to log differences for later manual inspection.

```
var unequalFields = sdkConfigSnapshot.UnequalFields(sdkConfig).ToList();
if (unequalFields.Any())
{
    var configJson = JsonUtility.ToJson(sdkConfig, prettyPrint: true);
    Debug.Log($"(config) {configJson}");
    Debug.LogWarning($"[DIFF-FIELDS] ({unequalFields.Count}): {string.Join(",
", unequalFields)}.");
}
else
```

```
{
    Debug.Log("SDK config applied successfully.");
}
```

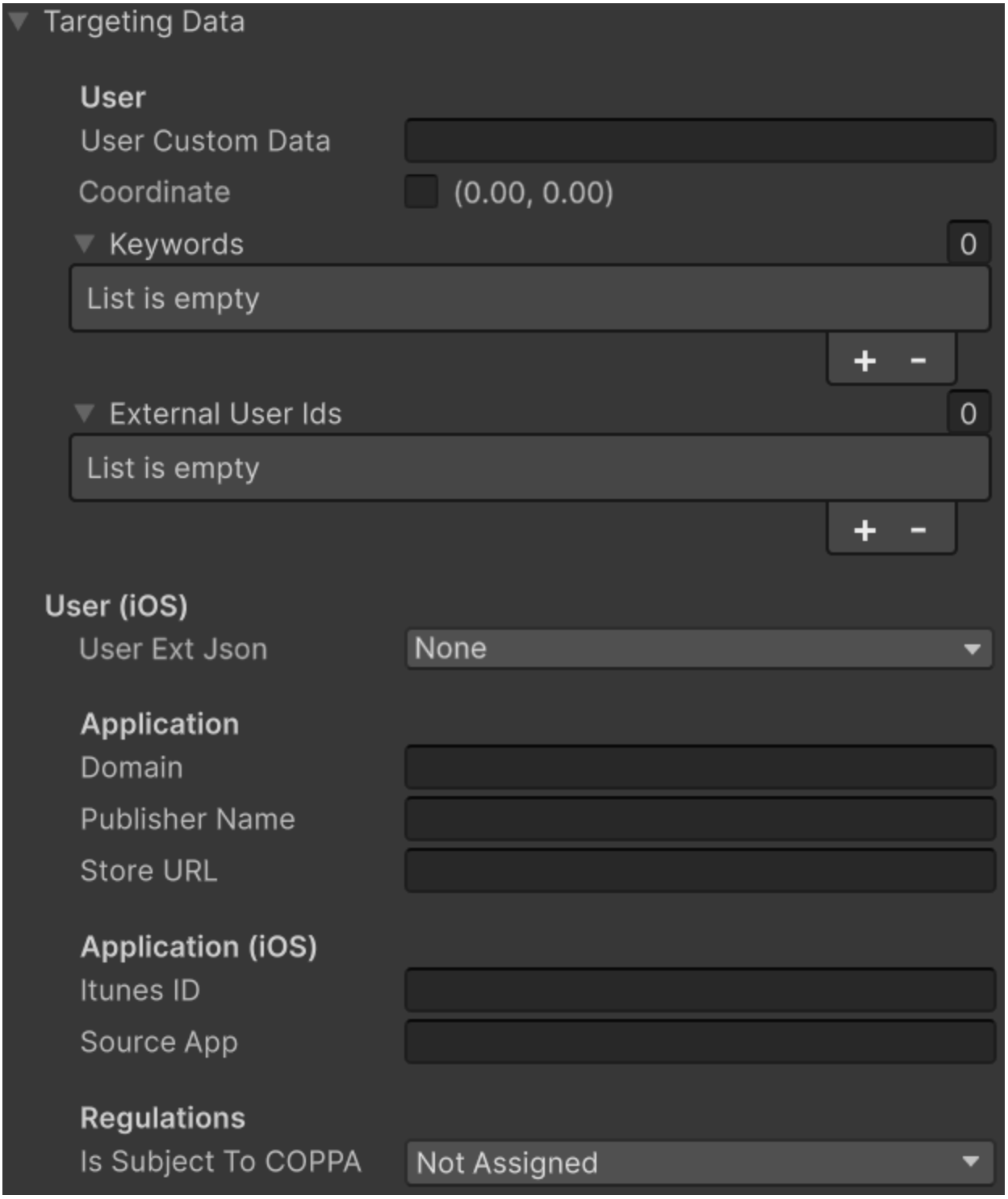
Improving targeting

ITargetingProxy provides few global configuration options.

Sdk property of Appstock can be used to get the instance of ITargetingProxy.

Data-driven approach

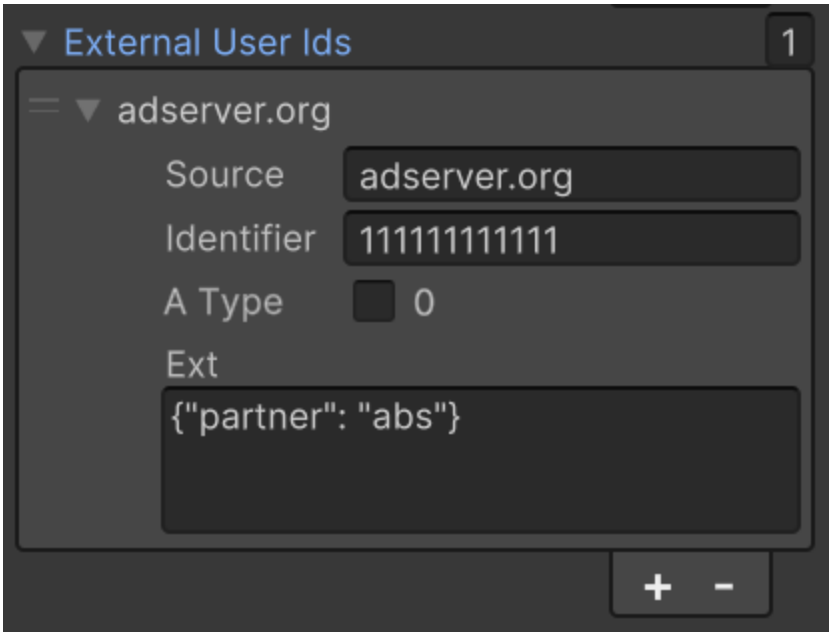
TargetingData is a data container -- with a SerializableAttribute -- for specifying settings via Unity Editor.



External User IDs

If you happen to know external ID of this user on other services, you can add them as ExternalUserID elements.

> **WARNING**
>
> If you populate ext, make sure it is a valid JSON object.
>
> Other JSON value types are NOT supported.



Applying the data

- Expose TargetingData variable on your MonoBehaviour
  - Or put it inside some ScriptableObject and assign through that.
- Call extension method Apply(ITargetingProxy, TargetingData) on Appstock.Sdk and pass your TargetingData.

```
#nullable enable

namespace AppstockSDK.DevApp
{
    public class TargetingDemo : MonoBehaviour
    {
        public SdkConfig sdkConfig = new();
        public TargetingData targetingData = new();

        // Start is called before the first frame update
```

```
    private IEnumerator Start()
    {
        Debug.Log($"[{DateTime.Now:O}] Applying config...");
        Appstock.Sdk.Apply(sdkConfig);

        Debug.Log($"[{DateTime.Now:O}] Attempting to init SDK...");
        Appstock.InitializeSdk("appstock-demo");

        Debug.Log($"[{DateTime.Now:O}] Letting SDK init to finish...");
        yield return new WaitForSeconds(1);

        Debug.Log($"[{DateTime.Now:O}] Applying targeting data...");
        Appstock.Targeting.Apply(targetingData);
```

> **TIP**
>
> You can find a complete code in "ConfigAndTargeting" sample.

Ext Slots

Some data structures expose public fields of type ExtSlot.

See Ext Slot page for more info.

Taking a snapshot of active data

Extension method TakeSnapshot(ITargetingProxy) can retrieve all readable properties from ITargeting Proxy.

```
var targetingSnapshot = Appstock.Targeting.TakeSnapshot();
```

It can be serialized and logged into console

```
var snapshotJson = JsonUtility.ToJson(targetingSnapshot, prettyPrint: true);
Debug.Log($"(snapshot) {snapshotJson}");
```

or compared to the expected data -- to log differences for later manual inspection.

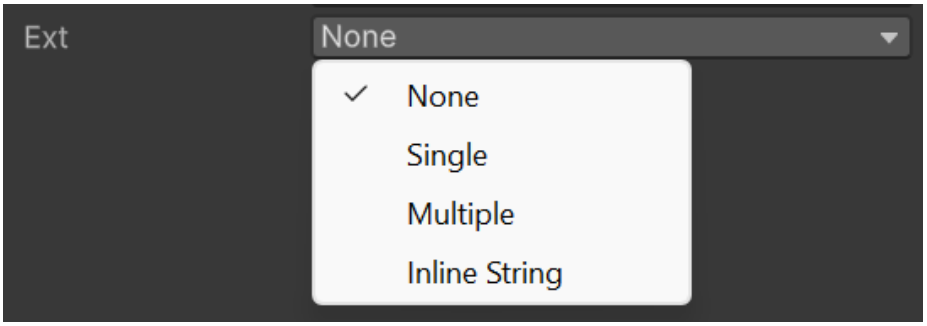
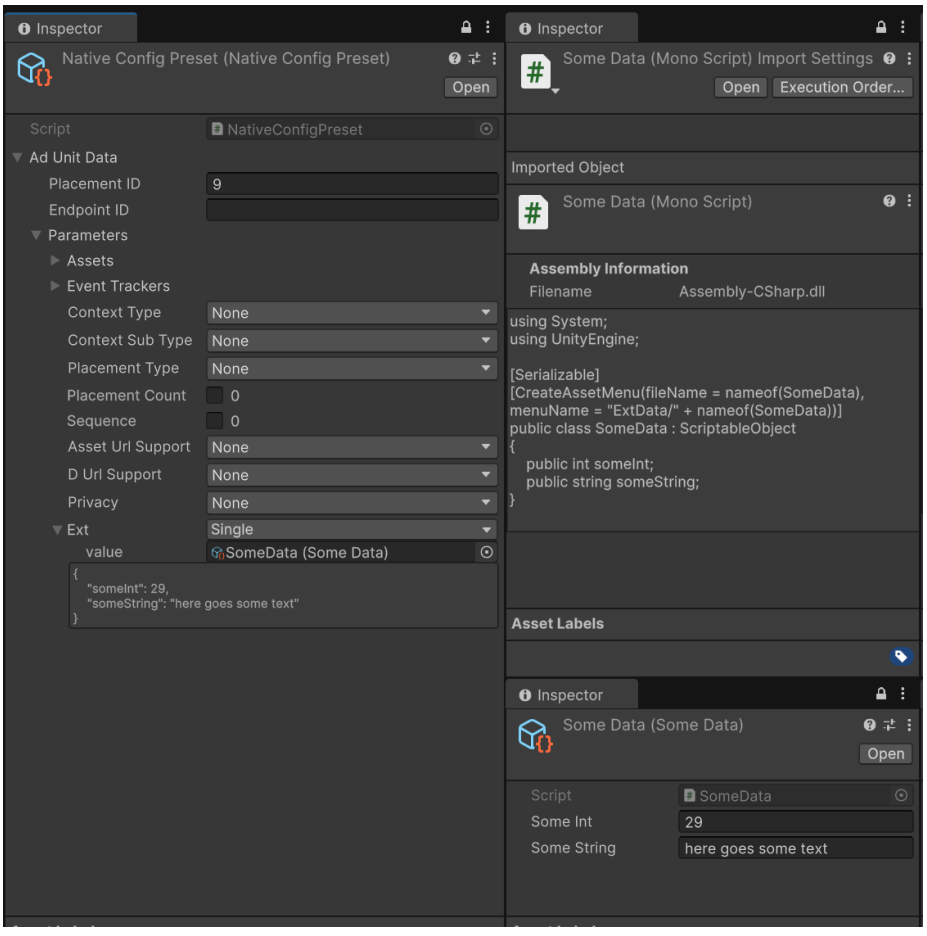
```
var unequalFields = targetingData.UnequalFields(targetingSnapshot,
fieldsToIgnore: new[]
{
    nameof(targetingData.userExtJson) // the ordering is not guaranteed
```

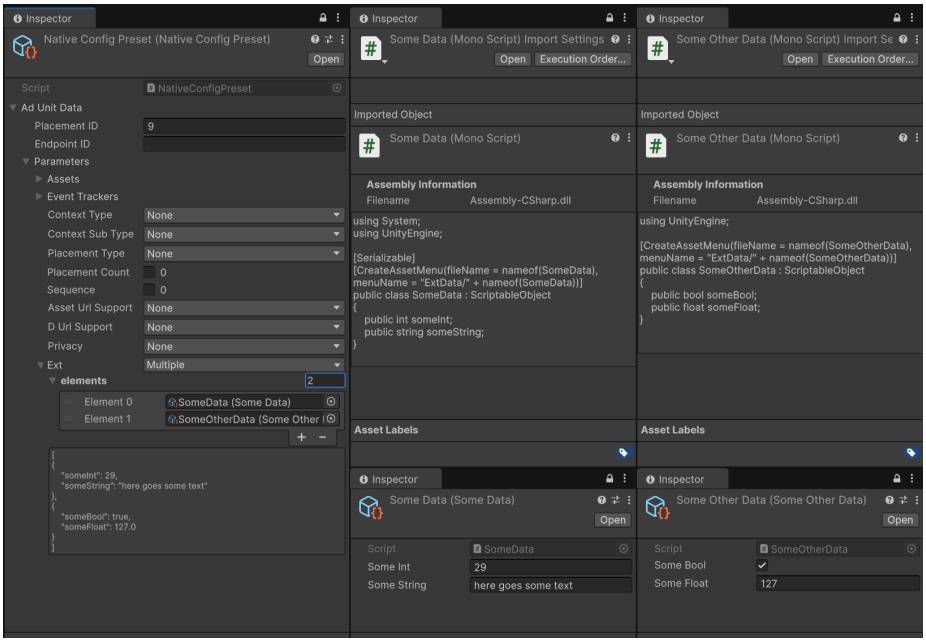
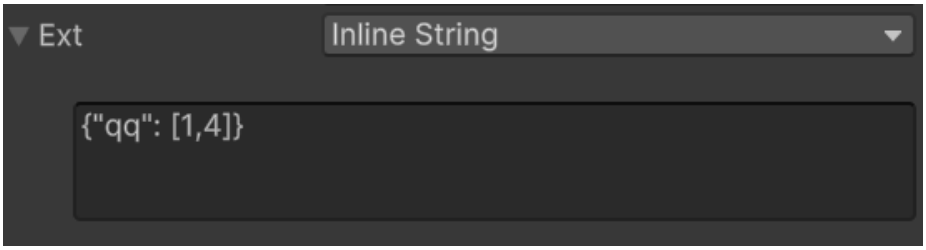
```
upon retrieval
}).ToList();
if (unequalFields.Any())
{
    var configJson = JsonUtility.ToJson(targetingData, prettyPrint: true);
    Debug.Log($"(config) {configJson}");
    Debug.LogWarning($"[DIFF-FIELDS] ({unequalFields.Count}): {string.Join(",
", unequalFields)}.");
}
else
{
    Debug.Log("Targeting data applied successfully.");
}
```

Ext Slot

Some data structures expose public fields of type ExtSlot.

Any JSON data is allowed inside.

Mode	Description	Scrrenshot
None	Default value. No content.	
Single	Object form. Accepts a single ScriptableObject asset. You can use any necessary structure for your data class. Has preview.	

Mode	Description	Scrrenshot
Multiple	Array form. Accepts multiple ScriptableObject assets. You can use any necessary structure for your data class(-es). Has preview.	
Inline String	String form. Fill in with a valid JSON string.	

Also provides implicit casting operators for all 3 cases:

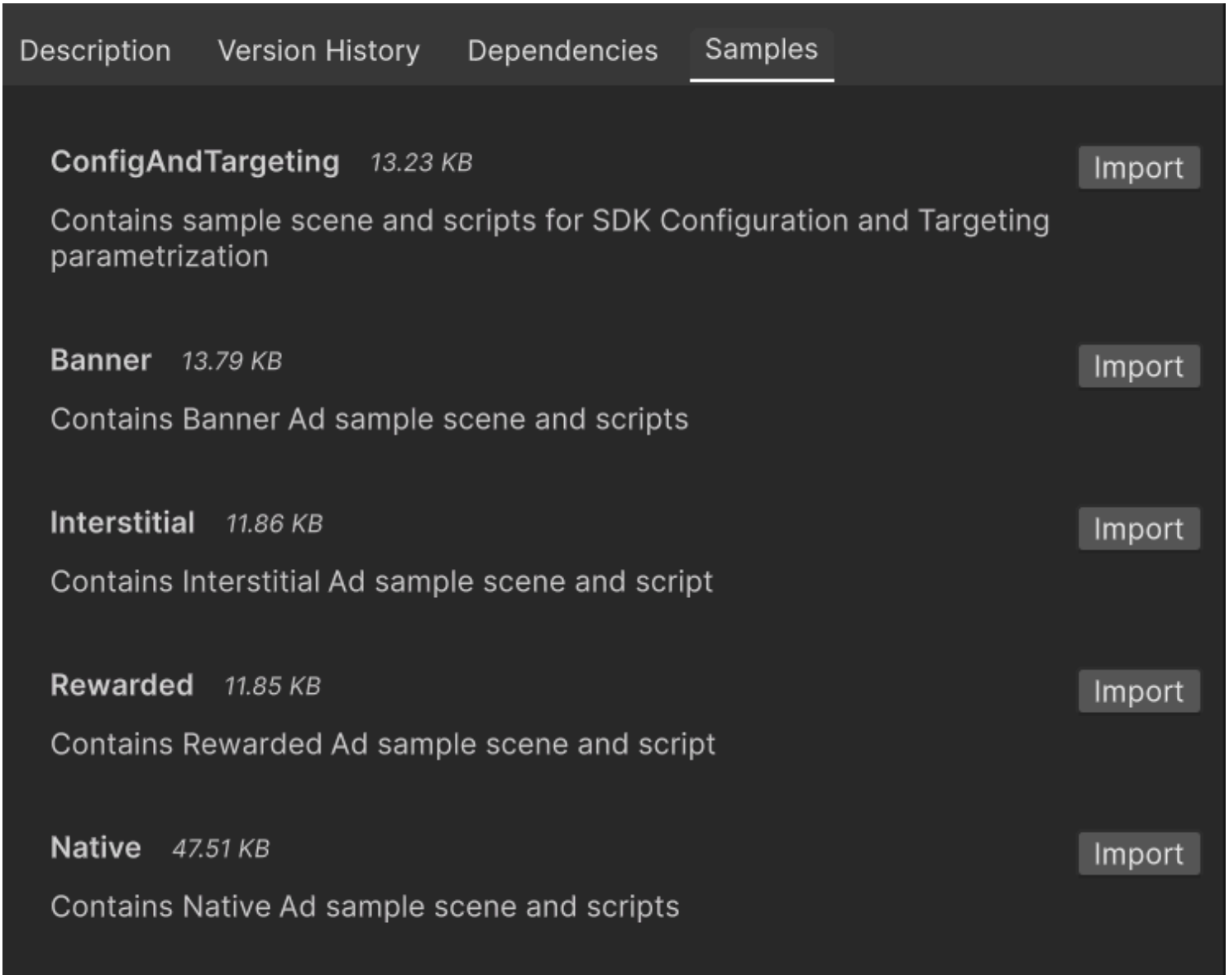
- implicit operator ExtSlot(string?)
- implicit operator ExtSlot(ScriptableObject?)
- implicit operator ExtSlot(ScriptableObject[]?)

Samples

There are 5 samples provided with the package.

Sample name	Related topics
Banner	SDK Initialization and Banner Ad
Interstitial	Interstitial Ad
Rewarded	Rewarded Ad
Native	Native Ads
ConfigAndTargeting	SDK Configuration and Improved Targeting

You can find them in Details panel of Package Manager window.



Namespace AppstockSDK.Android

Classes

SdkEnums

Exposes a collection of existing converters between managed and native enum values.

SdkImp

ISdkImp implementation for Android.

Interfaces

IJavaEnum

Converter between managed and native enum types.

Interface IJavaEnum

Namespace: AppstockSDK.Android

Assembly: Appstock.Android.dll

Converter between managed and native enum types.

```
public interface IJavaEnum
```

Properties

Converter

Gets managed-to-native converter delegate.

```
Func<object, AndroidJavaObject?> Converter { get; }
```

Property Value

Func<object, AndroidJavaObject>

A delegate that can convert managed enum value to native one.

ConvertibleEnumValues

Gets convertible values of managed enum.

```
IEnumerable<object> ConvertibleEnumValues { get; }
```

Property Value

IEnumerable<object>

Managed enum values for which the name of native enum is known.

EnumType

Gets the type of managed enum.

```
Type EnumType { get; }
```

Property Value

Type

Managed enum type.

Reverter

Gets native-to-managed converter delegate.

```
Func<AndroidJavaObject, object?> Reverter { get; }
```

Property Value

Func<AndroidJavaObject, object>

A delegate that can convert native enum value to managed one.

Class SdkEnums

Namespace: AppstockSDK.Android

Assembly: Appstock.Android.dll

Exposes a collection of existing converters between managed and native enum values.

```
public static class SdkEnums
```

Inheritance

object ← SdkEnums

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Properties

JavaEnums

Get the converters.

```
public static IEnumerable<IJavaEnum> JavaEnums { get; }
```

Property Value

IEnumerable<IJavaEnum>

Collection of existing converters.

Class SdkImp

Namespace: AppstockSDK.Android

Assembly: Appstock.Android.dll

ISdkImp implementation for Android.

```
public sealed class SdkImp : ISdkImp
```

Inheritance

object ← SdkImp

Implements

ISdkImp

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.ReferenceEquals(object, object) , object.ToString()

Properties

SdkInitializer

Gets SDK Initializer object.

```
public ISdkInitializer SdkInitializer { get; }
```

Property Value

ISdkInitializer

Proxy for a static method in the native SDK library for a specific platform.

SdkProxy

Gets SDK proxy object.

```
public ISdkProxy SdkProxy { get; }
```

Property Value

ISdkProxy

Managed object that exposes global settings of native SDK.

TargetingProxy

Gets SDK targeting proxy object.

```
public ITargetingProxy TargetingProxy { get; }
```

Property Value

ITargetingProxy

Managed object that exposes targeting settings of native SDK.

Methods

CreateBanner(AdSize)

Creates a banner of specified size.

```
public IBannerAd CreateBanner(AdSize adSize)
```

Parameters

`adSize` AdSize

Size of the banner view.

Returns

IBannerAd

Managed object that exposes controls over native ad view.

CreateInterstitial()

Creates an interstitial ad unit.

```
public IInterstitialAd CreateInterstitial()
```

Returns

IInterstitialAd

Managed object that exposes controls over native interstitial ad unit.

CreateNativeAdLoader(AdUnitData)

Creates a loader for native ad.

```
public INativeAdLoader CreateNativeAdLoader(AdUnitData adUnitData)
```

Parameters

`adUnitData` AdUnitData

Serializable template for native ad request.

Returns

INativeAdLoader

Managed object that exposes controls over native ad loader.

CreateRewarded()

Creates a rewarded ad unit.

```
public IRewardedAd CreateRewarded()
```

Returns

IRewardedAd

Managed object that exposes controls over native rewarded ad unit.

Namespace AppstockSDK.Api

Main namespace of the package.

The most useful classes

- Initialization
  - Appstock
- Configuration and Targeting
  - SdkConfig
  - TargetingData
- Basic Ads
  - BannerAd
  - InterstitialAd
  - RewardedAd

Classes

Appstock
: Facade for static methods and cross-platform getters.

BannerAd
: Banner ad. Anchored to some part of the screen. Call Dispose() to remove when done. Or use Show() / Hide() to toggle visibility.

BannerAdPositionExtensions
: Extensions for AnchoredAdPosition.

InterstitialAd
: A full page ad experience at natural transition points, such as a page change or an app launch. Interstitials use a close button that removes the ad from the user's experience.

OptionalBoolExtensions
: Extensions for OptionalBool.

RewardedAd
: Rewarded ads reward users with in-app items for interacting with video ads, playable ads, or surveys.

SdkConfig
: Data container for writeable properties of ISdkProxy.

SdkConfigSnapshot
: Data container for readable properties of ISdkProxy.

SdkProxyExtensions
: Extension methods for ISdkProxy.

TargetingData

TargetingProxyExtensions
Extension methods for ITargetingProxy.

Structs

AdError
Representation of an error.

AdInfo
The info about the ad.

AdReward
The reward for the ad.

AdSize
The size of a banner ad.

ExtSlot
Data container for exchange-specific extensions to OpenRTB.

ExternalUserID
External ID of the user that could be used for ad targeting.
https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids.md

SerializableOptional<T>
Represents Nullable<T> for simple value types.

Interfaces

IAdEventSource
Events shared by all non-native ads.

IAdUnit
Properties required for any ad unit to function.

IAnchoredAd
Ad that is anchored to the screen.

IAutoRefreshableAd
Ad that can auto-refresh.

IBannerAd
Common properties for banner facade and platform-specific implementations.

ICloseableAd
Ad that can be closed.

IConcealableAd
Ad that can be loaded and shown.

IInterstitialAd
: Common properties for interstitial ad facade and platform-specific implementations.

ILoadableAd
: Ad that can be loaded.

IModalAd
: Ad that can be loaded and shown.

IModalAdEventSource
: Events shared by interstitial and rewarded ads.

IMonoformattedAd
: Single format supported.

IMultiformattedAd
: Multiple formats supported.

IMuteableAd
: Ad that can be muted.

INativeAdLoader
: A native object that can be used to load a native ad (once built).

IResizableAd
: Shared by non-native ad units.

IRewardedAd
: Common properties for rewarded ad facade and platform-specific implementations.

IRewardingAdEventSource
: Events for rewarding ads.

ISdkProxy
: Managed object that exposes global settings of native SDK.

ISkippableAd
: Ad that can be skipped.

ITargetingProxy
: Managed object that exposes targeting settings of native SDK.

Enums

AdElementPosition
: Where the ad element (image/button) should be placed.

AdFormat
: Which ad content format to display.

AdPosition

Where the ad itself is logically.

AnchoredAdPosition

Where the banner itself should be placed.

ExtSlot.Mode

What fields of slots should be used to build the value.

LogLevel

The desired verbosity of logs by native library.

OptionalBool

Represents `bool?` -- i.e. Nullable<T> of bool -- that can be `true`, `false` or `null`.

Enum AdElementPosition

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Where the ad element (image/button) should be placed.

```
public enum AdElementPosition
```

Fields

`BottomCenter = 5`

Bottom center of safe area.

`BottomLeft = 4`

Bottom left of safe area.

`BottomRight = 6`

Bottom right of safe area.

`None = 0`

Not assigned (keep default).

`TopCenter = 2`

Top center of safe area.

`TopLeft = 1`

Top left corner of safe area.

`TopRight = 3`

Top right of safe area.

Struct AdError

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Representation of an error.

```
public struct AdError
```

Inherited Members
ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

Message

Gets or sets the message.

```
public string Message
```

Field Value

string

The message from underlying error in native library.

Enum AdFormat

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Which ad content format to display.

```
public enum AdFormat
```

Fields

`Banner = 1`

Banner, i.e. static content.

`Native = 4`

Native content format.

`None = 0`

Not assigned (keep default).

`Video = 2`

Video content.

Struct AdInfo

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

The info about the ad.

```
public readonly struct AdInfo : IEquatable<AdInfo>
```

Implements
IEquatable<AdInfo>

Inherited Members
object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Constructors

AdInfo(double?)

Initializes a new instance of the AdInfo class.

```
public AdInfo(double? price)
```

Parameters

`price` double?

Price of the ad.

Fields

Price

Price of the ad.

```
public readonly double? Price
```

Field Value

double?

Price of the ad.

Methods

Equals(AdInfo)

Indicates whether the current object is equal to another object of the same type.

```
public bool Equals(AdInfo other)
```

Parameters

`other` AdInfo

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(object?)

Indicates whether this instance and a specified object are equal.

```
public override bool Equals(object? obj)
```

Parameters

`obj` object

The object to compare with the current instance.

Returns

bool

true if `obj` and this instance are the same type and represent the same value; otherwise, false.

GetHashCode()

Returns the hash code for this instance.

```
public override int GetHashCode()
```

Returns

int

A 32-bit signed integer that is the hash code for this instance.

ToString()

Returns the fully qualified type name of this instance.

```
public override string ToString()
```

Returns

string

The fully qualified type name.

Enum AdPosition

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Where the ad itself is logically.

```
public enum AdPosition
```

Fields

`Footer = 5`

Ad is placed inside page footer.

`Header = 4`

Ad is placed inside page header.

`Sidebar = 6`

Ad is placed inside page sidebar.

`Unknown = 0`

Not assigned (unknown).

Struct AdReward

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

The reward for the ad.

```
public readonly struct AdReward : IEquatable<AdReward>
```

Implements
IEquatable<AdReward>

Inherited Members
object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Constructors

AdReward(string?, int, string?)

Initializes a new instance of the AdReward class.

```
public AdReward(string? rewardType, int count, string? extJsonString)
```

Parameters

`rewardType` string

What kind of reward user earned.

`count` int

How many units of reward user earned.

`extJsonString` string

Exchange-specific extensions to OpenRTB.

Fields

Count

Gets or sets reward count.

```
public readonly int Count
```

Field Value

int

How many reward units user earned.

ExtJsonString

Gets or sets `ext` value.

```
public readonly string? ExtJsonString
```

Field Value

string

Exchange-specific extensions to OpenRTB.

Remarks

Most likely to be a JSON object.

RewardType

Gets or sets reward type.

```
public readonly string? RewardType
```

Field Value

string

Type of reward user earned.

Methods

Equals(AdReward)

Indicates whether the current object is equal to another object of the same type.

```
public bool Equals(AdReward other)
```

Parameters

`other` AdReward

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(object?)

Indicates whether this instance and a specified object are equal.

```
public override bool Equals(object? obj)
```

Parameters

`obj` object

The object to compare with the current instance.

Returns

bool

true if `obj` and this instance are the same type and represent the same value; otherwise, false.

GetHashCode()

Returns the hash code for this instance.

```
public override int GetHashCode()
```

Returns

int

A 32-bit signed integer that is the hash code for this instance.

ToString()

Returns the fully qualified type name of this instance.

```
public override string ToString()
```

Returns

string

The fully qualified type name.

Struct AdSize

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

The size of a banner ad.

```
public readonly struct AdSize : IEquatable<AdSize>, IFormattable
```

Implements
IEquatable<AdSize>, IFormattable

Inherited Members
object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Constructors

AdSize(int, int)

Creates a new AdSize.

```
public AdSize(int width, int height)
```

Parameters

`width` int

The width of the ad in density-independent pixels.

`height` int

The height of the ad in density-independent pixels.

Fields

Banner

Gets a standard banner ad size.

```
public static readonly AdSize Banner
```

Field Value

AdSize

Interactive Advertising Bureau (IAB) banner ad size (320x50 density-independent pixels).

Height

Gets the height.

```
public readonly int Height
```

Field Value

int

The height of the ad in density-independent pixels.

IABBanner

Gets a standard full banner ad size.

```
public static readonly AdSize IABBanner
```

Field Value

AdSize

Interactive Advertising Bureau (IAB) full banner ad size (468x60 density-independent pixels).

Leaderboard

Gets a standard leaderboard ad size.

```
public static readonly AdSize Leaderboard
```

Field Value

AdSize

Interactive Advertising Bureau (IAB) leaderboard ad size (728x90 density-independent pixels).

MediumRectangle

Gets a standard medium rectangle ad size.

```
public static readonly AdSize MediumRectangle
```

Field Value

AdSize

Interactive Advertising Bureau (IAB) medium rectangle ad size (300x250 density-independent pixels).

Width

Gets the width.

```
public readonly int Width
```

Field Value

int

The width of the ad in density-independent pixels.

Methods

Equals(AdSize)

Indicates whether the current object is equal to another object of the same type.

```
public bool Equals(AdSize other)
```

Parameters

`other` AdSize

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(object?)

Indicates whether this instance and a specified object are equal.

```
public override bool Equals(object? obj)
```

Parameters

`obj` object

The object to compare with the current instance.

Returns

bool

true if `obj` and this instance are the same type and represent the same value; otherwise, false.

GetHashCode()

Returns the hash code for this instance.

```
public override int GetHashCode()
```

Returns

int

A 32-bit signed integer that is the hash code for this instance.

ToString()

Returns the fully qualified type name of this instance.

```
public override string ToString()
```

Returns

string

The fully qualified type name.

ToString(string?, IFormatProvider?)

Formats the value of the current instance using the specified format.

```
public string ToString(string? format, IFormatProvider? formatProvider)
```

Parameters

`format` string

The format to use.
-or-
A null reference (`Nothing` in Visual Basic) to use the default format defined for the type of the IFormattable implementation.

`formatProvider` IFormatProvider

The provider to use to format the value.
-or-
A null reference (`Nothing` in Visual Basic) to obtain the numeric format information from the current locale setting of the operating system.

Returns

string

The value of the current instance in the specified format.

Operators

implicit operator Vector2Int(AdSize)

Defines an implicit conversion of AdSize to a Vector2Int.

```
public static implicit operator Vector2Int(AdSize size)
```

Parameters

`size` AdSize

The AdSize to convert.

Returns

Vector2Int

The converted Vector2Int.

implicit operator AdSize(Vector2Int)

Defines an implicit conversion of Vector2Int to a AdSize.

```
public static implicit operator AdSize(Vector2Int size)
```

Parameters

`size` Vector2Int

The Vector2Int to convert.

Returns

AdSize

The converted AdSize.

Enum AnchoredAdPosition

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Where the banner itself should be placed.

```
public enum AnchoredAdPosition
```

Extension Methods
BannerAdPositionExtensions.ToAdPosition(AnchoredAdPosition)

Fields

`Bottom = 1`

Bottom center of safe area.

`BottomLeft = 4`

Bottom left of safe area.

`BottomRight = 5`

Bottom right of safe area.

`Top = 0`

Top center of safe area.

`TopLeft = 2`

Top left corner of safe area.

`TopRight = 3`

Top right of safe area.

Class Appstock

Namespace: AppstockSDK.Api

Assembly: Appstock.Api.dll

Facade for static methods and cross-platform getters.

```
public static class Appstock
```

Inheritance

object ← Appstock

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Properties

Sdk

Gets SDK proxy object.

```
public static ISdkProxy Sdk { get; }
```

Property Value

ISdkProxy

Managed object that exposes global settings of native SDK.

See Also

ISdkProxy

Targeting

Gets SDK targeting proxy object.

```
public static ITargetingProxy Targeting { get; }
```

Property Value

ITargetingProxy

Managed object that exposes targeting settings of native SDK.

See Also
ITargetingProxy

Methods

InitializeSdk(string)

Initialize the SDK.

```
public static void InitializeSdk(string partnerKey)
```

Parameters

`partnerKey` string

Determines the Appstock server URL. The Appstock account manager should provide you with this key.

Class BannerAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Api.dll

Banner ad. Anchored to some part of the screen. Call Dispose() to remove when done. Or use Show() / Hide() to toggle visibility.

```
public class BannerAd : IBannerAd, IDisposable, IAdUnit, ILoadableAd,
IMonoformattedAd, IResizableAd, IConcealableAd, IAnchoredAd,
IAutoRefreshableAd, IAdEventSource
```

Inheritance

object ← BannerAd

Implements

IBannerAd, IDisposable, IAdUnit, ILoadableAd, IMonoformattedAd, IResizableAd, IConcealableAd, IAnchoredAd, IAutoRefreshableAd, IAdEventSource

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

BannerAd(AdSize)

Initializes a new instance of the BannerAd class.

```
public BannerAd(AdSize adSize)
```

Parameters

`adSize` AdSize

Banner view size.

Properties

AdPosition

Gets or sets ad position (for exchange purposes).

```
public AdPosition AdPosition { get; set; }
```

Property Value

AdPosition

Where the ad itself is logically (header/footer etc.).

See Also
AdPosition

AdSizes

Sets ad sizes for the ad.

```
public IEnumerable<AdSize> AdSizes { set; }
```

Property Value

IEnumerable<AdSize>

Sizes that ad's content may take.

AdUnitFormat

Sets format for the ad unit.

```
public AdFormat AdUnitFormat { set; }
```

Property Value

AdFormat

Ad format for the Ad.

AnchoredPosition

Gets or sets the position ad is anchored to.

```
public AnchoredAdPosition AnchoredPosition { get; set; }
```

Property Value

AnchoredAdPosition

Where the ad itself should be placed on screen (edge/corner).

See Also
AnchoredPosition

AutoRefreshDelay

Gets or sets auto refresh delay (interval).

```
public TimeSpan AutoRefreshDelay { get; set; }
```

Property Value

TimeSpan

How often to refresh the ad content.

EndpointID

Sets the endpoint ID for the ad.

```
public string? EndpointID { set; }
```

Property Value

string

Unique endpoint identifier generated on the Appstock platform's UI.

PlacementID

Sets the placement ID for the ad.

```
public string? PlacementID { set; }
```

Property Value

string

Unique placement identifier generated on the Appstock platform's UI.

Methods

Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

Dispose(bool)

Releases the unmanaged resources used by the class and optionally releases the managed resources.

```
protected virtual void Dispose(bool disposing)
```

Parameters

`disposing` bool

`true` to release both managed and unmanaged resources; `false` to release only unmanaged resources.

Hide()

Set the ad invisible.

```
public void Hide()
```

LoadAd()

Load the content of the ad.

```
public void LoadAd()
```

Show()

Set the ad visible.

```
public void Show()
```

StopAutoRefresh()

Disable auto-refreshing.

```
public void StopAutoRefresh()
```

Events

OnAdClicked

Fired when ad was clicked on.

```
public event Action? OnAdClicked
```

Event Type

Action

OnAdClosed

Fired when ad was closed (from fullscreen mode).

```
public event Action? OnAdClosed
```

Event Type

Action

OnAdFailed

Fired when ad failed to load.

```
public event Action<AdError?>? OnAdFailed
```

Event Type

Action<AdError?>

OnAdLoaded

Fired when ad finished loading.

```
public event Action<AdInfo?>? OnAdLoaded
```

Event Type

Action<AdInfo?>

Class BannerAdPositionExtensions

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Extensions for AnchoredAdPosition.

```
public static class BannerAdPositionExtensions
```

Inheritance

object ← BannerAdPositionExtensions

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

ToAdPosition(AnchoredAdPosition)

Converts AnchoredAdPosition to an AdPosition.

```
public static AdPosition ToAdPosition(this AnchoredAdPosition adPosition)
```

Parameters

`adPosition` AnchoredAdPosition

The AnchoredAdPosition to convert.

Returns

AdPosition

The converted AdPosition.

Struct ExtSlot

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Data container for exchange-specific extensions to OpenRTB.

```
[Serializable]
public struct ExtSlot : IEquatable<ExtSlot>
```

Implements

IEquatable<ExtSlot>

Inherited Members

ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Remarks

Custom property drawer included.

Fields

customJsonString

JSON to be directly used as slot value.

```
[TextArea]
public string? customJsonString
```

Field Value

string

JSON array/object to be used as slot value in InlineString mode.

elements

Gets or sets multiple elements of the slot value.

```
public ScriptableObject?[]? elements
```

Field Value

ScriptableObject[]

Data assets to be serialized into JSON and used as array element for slot value in Multiple mode.

mode

Gets or sets slot mode.

```
public ExtSlot.Mode mode
```

Field Value

ExtSlot.Mode

Mode of the slot.

See Also
ExtSlot.Mode

value

Gets or sets single value of the slot.

```
public ScriptableObject? value
```

Field Value

ScriptableObject

Data asset to be serialized into JSON and used as value of the slot in Single mode.

Properties

JsonString

Gets the effective JSON string.

```
public string? JsonString { get; }
```

Property Value

string

Effective JSON based on mode and data fields.

Methods

Equals(ExtSlot)

Indicates whether the current object is equal to another object of the same type.

```
public bool Equals(ExtSlot other)
```

Parameters

`other` ExtSlot

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(object?)

Indicates whether this instance and a specified object are equal.

```
public override bool Equals(object? obj)
```

Parameters

`obj` object

The object to compare with the current instance.

Returns

bool

true if `obj` and this instance are the same type and represent the same value; otherwise, false.

GetDiffMessage(ExtSlot)

Compares values of 2 ExtSlots.

```
public string? GetDiffMessage(ExtSlot other)
```

Parameters

`other` ExtSlot

The object to compare against.

Returns

string

`null` if effective values are equal, descriptive message otherwise.

GetHashCode()

Returns the hash code for this instance.

```
public override int GetHashCode()
```

Returns

int

A 32-bit signed integer that is the hash code for this instance.

ToJsonArrayString(IEnumerable<ScriptableObject?>, bool)

Converts ScriptableObjects to a string.

```
public static string ToJsonArrayString(IEnumerable<ScriptableObject?>
scriptableObjects, bool prettyPrint)
```

Parameters

`scriptableObjects` IEnumerable<ScriptableObject>

The ScriptableObjects to convert.

`prettyPrint` bool

Whether to make the output pretty-ish.

Returns

string

Serialized JSON array of `scriptableObjects`.

Operators

implicit operator ExtSlot(string?)

Defines an implicit conversion of string to a ExtSlot.

```
public static implicit operator ExtSlot(string? customJsonString)
```

Parameters

`customJsonString` string

The string to convert.

Returns

ExtSlot

The converted ExtSlot.

implicit operator ExtSlot(ScriptableObject?)

Defines an implicit conversion of ScriptableObject to a ExtSlot.

```
public static implicit operator ExtSlot(ScriptableObject? scriptableObject)
```

Parameters

`scriptableObject` ScriptableObject

The ScriptableObject to convert.

Returns

ExtSlot

The converted ExtSlot.

implicit operator ExtSlot(ScriptableObject[]?)

Defines an implicit conversion of ScriptableObjects to a ExtSlot.

```
public static implicit operator ExtSlot(ScriptableObject[]? scriptableObjects)
```

Parameters

`scriptableObjects` ScriptableObject[]

The ScriptableObjects to convert.

Returns

ExtSlot

The converted ExtSlot.

Enum ExtSlot.Mode

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

What fields of slots should be used to build the value.

```
[Serializable]
public enum ExtSlot.Mode
```

Fields

`InlineString = 3`

Send JSON as explicitly provided -- use customJsonString

`Multiple = 2`

Send JSON array -- use elements.

`None = 0`

No value (i.e. `ext` is `null`)

`Single = 1`

Send JSON object -- use value.

Struct ExternalUserID

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

External ID of the user that could be used for ad targeting.
https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids.md

```
[Serializable]
public struct ExternalUserID : IEquatable<ExternalUserID>
```

Implements

IEquatable<ExternalUserID>

Inherited Members

object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

aType

Gets or sets the type of user agent.

```
[Tooltip("Type of user agent the match is from.\nIt is highly recommended to set this, as many DSPs
separate app-native IDs from browser-based IDs and require a type value for ID resolution.\nSee
https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids
.md#list_agenttypes")]
public SerializableOptional<int> aType
```

Field Value

SerializableOptional<int>

> Type of user agent the match is from. It is highly recommended to set this, as many DSPs separate app-native IDs from browser-based IDs and require a type value for ID resolution.

Remarks

Refer to List:
https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids.md#list_agenttypes .

ext

Gets or sets `ext` content.

```
[TextArea]
[Tooltip("Optional vendor-specific extensions.")]
```

```
public string? ext
```

Field Value

string

Optional vendor-specific extensions.

identifier

Gets or sets the ID.

```
[Tooltip("Cookie or platform-native identifier.")]
public string? identifier
```

Field Value

string

Cookie or platform-native identifier.

source

Gets or sets the source of ID.

```
[Tooltip("Source or technology provider responsible for the set of included IDs.\nExpressed as a top-level domain.")]
public string? source
```

Field Value

string

Source or technology provider responsible for the set of included IDs.

Remarks

Expressed as a top-level domain.

Properties

WithoutDefaults

Strips away whitespace strings that Unity creates on serialization.

```
public readonly ExternalUserID WithoutDefaults { get; }
```

Property Value

ExternalUserID

"Clone" without whitespace strings.

Methods

Equals(ExternalUserID)

Indicates whether the current object is equal to another object of the same type.

```
public readonly bool Equals(ExternalUserID other)
```

Parameters

`other` ExternalUserID

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(object?)

Indicates whether this instance and a specified object are equal.

```
public override readonly bool Equals(object? obj)
```

Parameters

`obj` object

The object to compare with the current instance.

Returns

bool

true if `obj` and this instance are the same type and represent the same value; otherwise, false.

GetHashCode()

Returns the hash code for this instance.

```
public override readonly int GetHashCode()
```

Returns

int

A 32-bit signed integer that is the hash code for this instance.

ToString()

Returns the fully qualified type name of this instance.

```
public override readonly string ToString()
```

Returns

string

The fully qualified type name.

Interface IAdEventSource

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Events shared by all non-native ads.

```
public interface IAdEventSource
```

Events

OnAdClicked

Fired when ad was clicked on.

```
event Action OnAdClicked
```

Event Type

Action

OnAdClosed

Fired when ad was closed (from fullscreen mode).

```
event Action OnAdClosed
```

Event Type

Action

OnAdFailed

Fired when ad failed to load.

```
event Action<AdError?> OnAdFailed
```

Event Type

Action<AdError?>

OnAdLoaded

Fired when ad finished loading.

```
event Action<AdInfo?> OnAdLoaded
```

Event Type

Action<AdInfo?>

Interface IAdUnit

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Properties required for any ad unit to function.

```
public interface IAdUnit
```

Properties

EndpointID

Sets the endpoint ID for the ad.

```
string? EndpointID { set; }
```

Property Value

string

Unique endpoint identifier generated on the Appstock platform's UI.

PlacementID

Sets the placement ID for the ad.

```
string? PlacementID { set; }
```

Property Value

string

Unique placement identifier generated on the Appstock platform's UI.

Interface IAnchoredAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Ad that is anchored to the screen.

```
public interface IAnchoredAd
```

Properties

AdPosition

Gets or sets ad position (for exchange purposes).

```
AdPosition AdPosition { get; set; }
```

Property Value

AdPosition

Where the ad itself is logically (header/footer etc.).

See Also
AdPosition

AnchoredPosition

Gets or sets the position ad is anchored to.

```
AnchoredAdPosition AnchoredPosition { get; set; }
```

Property Value

AnchoredAdPosition

Where the ad itself should be placed on screen (edge/corner).

See Also

AnchoredPosition

Interface IAutoRefreshableAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Ad that can auto-refresh.

```
public interface IAutoRefreshableAd
```

Properties

AutoRefreshDelay

Gets or sets auto refresh delay (interval).

```
TimeSpan AutoRefreshDelay { get; set; }
```

Property Value

TimeSpan

How often to refresh the ad content.

Methods

StopAutoRefresh()

Disable auto-refreshing.

```
void StopAutoRefresh()
```

Interface IBannerAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Common properties for banner facade and platform-specific implementations.

```
public interface IBannerAd : IDisposable, IAdUnit, ILoadableAd, IMonoformattedAd,
IResizableAd, IConcealableAd, IAnchoredAd, IAutoRefreshableAd, IAdEventSource
```

Inherited Members

IDisposable.Dispose() , IAdUnit.PlacementID , IAdUnit.EndpointID , ILoadableAd.LoadAd() , IMonoformattedAd.AdUnitFormat , IResizableAd.AdSizes , IConcealableAd.Show() , IConcealableAd.Hide() , IAnchoredAd.AnchoredPosition , IAnchoredAd.AdPosition , IAutoRefreshableAd.AutoRefreshDelay , IAutoRefreshableAd.StopAutoRefresh() , IAdEventSource.OnAdLoaded , IAdEventSource.OnAdFailed , IAdEventSource.OnAdClicked , IAdEventSource.OnAdClosed

Interface ICloseableAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Ad that can be closed.

```
public interface ICloseableAd
```

Properties

CloseButtonArea

Sets close button area factor.

```
double CloseButtonArea { set; }
```

Property Value

double

the percentage of the device screen that the close button should cover.

Remarks

Allowed range - `0...1`.

Default value is ~`0.1`.

CloseButtonPosition

Sets position of the close button.

```
AdElementPosition CloseButtonPosition { set; }
```

Property Value

AdElementPosition

Where the close button appears on the screen.

Remarks

Allowed values: TopLeft, TopRight.

Default is TopRight.

Interface IConcealableAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Ad that can be loaded and shown.

```
public interface IConcealableAd
```

Methods

Hide()

Set the ad invisible.

```
void Hide()
```

Show()

Set the ad visible.

```
void Show()
```

Interface IInterstitialAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Common properties for interstitial ad facade and platform-specific implementations.

```
public interface IInterstitialAd : IDisposable, IAdUnit, IMultiformattedAd,
IResizableAd, ISkippableAd, IMuteableAd, ICloseableAd, IModalAd, ILoadableAd,
IAdEventSource, IModalAdEventSource
```

Inherited Members

IDisposable.Dispose() , IAdUnit.PlacementID , IAdUnit.EndpointID , IMultiformattedAd.AdUnitFormats , IResizableAd.AdSizes , ISkippableAd.SkipButtonArea , ISkippableAd.SkipButtonPosition , ISkippableAd.SkipDelay , IMuteableAd.Muted , IMuteableAd.SoundButtonVisible , ICloseableAd.CloseButtonArea , ICloseableAd.CloseButtonPosition , IModalAd.Loaded , IModalAd.Show(Action) , ILoadableAd.LoadAd() , IAdEventSource.OnAdLoaded , IAdEventSource.OnAdFailed , IAdEventSource.OnAdClicked , IAdEventSource.OnAdClosed , IModalAdEventSource.OnAdDisplayed

Interface ILoadableAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Ad that can be loaded.

```
public interface ILoadableAd
```

Methods

LoadAd()

Load the content of the ad.

```
void LoadAd()
```

Interface IModalAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Ad that can be loaded and shown.

```
public interface IModalAd : ILoadableAd
```

Inherited Members
ILoadableAd.LoadAd()

Properties

Loaded

Gets if ad has loaded.

```
bool Loaded { get; }
```

Property Value

bool

`true` if ad has successfully loaded, `false` otherwise.

Methods

Show(Action?)

Show the ad on screen.

```
void Show(Action? actionOnUIThread = null)
```

Parameters

`actionOnUIThread` Action

Something you want to execute immediately when the ad is being added to the screen

`actionOnUIThread` is invoked on **NATIVE UI THREAD** which **MAY DIFFER from UNITY'S MAIN THREAD**.

Interface IModalAdEventSource

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Events shared by interstitial and rewarded ads.

```
public interface IModalAdEventSource
```

Events

OnAdDisplayed

Fired when ad did appear on screen.

```
event Action OnAdDisplayed
```

Event Type

Action

Interface IMonoformattedAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Single format supported.

```
public interface IMonoformattedAd
```

Properties

AdUnitFormat

Sets format for the ad unit.

```
AdFormat AdUnitFormat { set; }
```

Property Value

AdFormat

Ad format for the Ad.

Interface IMultiformattedAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Multiple formats supported.

```
public interface IMultiformattedAd
```

Properties

AdUnitFormats

Sets formats for the ad unit.

```
IEnumerable<AdFormat> AdUnitFormats { set; }
```

Property Value

IEnumerable<AdFormat>

Ad formats that are allowed into the ad.

Interface IMuteableAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Ad that can be muted.

```
public interface IMuteableAd
```

Properties

Muted

Sets if ad is to be muted.

```
bool Muted { set; }
```

Property Value

bool

`true` if muted, `false` otherwise.

SoundButtonVisible

Sets if `Mute` button is visible when the ad is on display.

```
bool SoundButtonVisible { set; }
```

Property Value

bool

`true` if visible, `false` otherwise.

Interface INativeAdLoader

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

A native object that can be used to load a native ad (once built).

```
public interface INativeAdLoader : IDisposable
```

Inherited Members
IDisposable.Dispose()

Methods

LoadAd(Action<INativeAd?, AdError?>)

Sends a request and loads the INativeAd.

```
void LoadAd(Action<INativeAd?, AdError?> callback)
```

Parameters

`callback` Action<INativeAd, AdError?>

Invoked when loading succeeds or fails.

Remarks

Success indicated by non-null INativeAd.

Interface IResizableAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Shared by non-native ad units.

```
public interface IResizableAd
```

Properties

AdSizes

Sets ad sizes for the ad.

```
IEnumerable<AdSize> AdSizes { set; }
```

Property Value

IEnumerable<AdSize>

Sizes that ad's content may take.

Interface IRewardedAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Common properties for rewarded ad facade and platform-specific implementations.

```
public interface IRewardedAd : IDisposable, IAdUnit, IMultiformattedAd,
IResizableAd, IMuteableAd, ICloseableAd, IModalAd, ILoadableAd, IAdEventSource,
IModalAdEventSource, IRewardingAdEventSource
```

Inherited Members

IDisposable.Dispose() , IAdUnit.PlacementID , IAdUnit.EndpointID , IMultiformattedAd.AdUnitFormats , IResizableAd.AdSizes , IMuteableAd.Muted , IMuteableAd.SoundButtonVisible , ICloseableAd.CloseButtonArea , ICloseableAd.CloseButtonPosition , IModalAd.Loaded , IModalAd.Show(Action) , ILoadableAd.LoadAd() , IAdEventSource.OnAdLoaded , IAdEventSource.OnAdFailed , IAdEventSource.OnAdClicked , IAdEventSource.OnAdClosed , IModalAdEventSource.OnAdDisplayed , IRewardingAdEventSource.OnReward

Interface IRewardingAdEventSource

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Events for rewarding ads.

```
public interface IRewardingAdEventSource
```

Events

OnReward

Fired when user deserved the reward.

```
event Action<AdReward?> OnReward
```

Event Type

Action<AdReward?>

Interface ISdkProxy

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Managed object that exposes global settings of native SDK.

```
public interface ISdkProxy
```

Extension Methods
SdkProxyExtensions.Apply(ISdkProxy, SdkConfig)

Properties

AdRequestTimeout

Gets or sets timeout for creative requests.

```
TimeSpan AdRequestTimeout { get; set; }
```

Property Value

TimeSpan

The time interval allowed for a creative to load before it is considered a failure.

AdRequestTimeoutPreRenderContent

Gets or sets timeout for video and interstitial creatves.

```
TimeSpan AdRequestTimeoutPreRenderContent { get; set; }
```

Property Value

TimeSpan

The time interval allowed for video and interstitial creatives to load.

AssignNativeAssetID

Gets or sets whether the asset ID for native ads should be manually assigned.

```
bool AssignNativeAssetID { get; set; }
```

Property Value

bool

`true` if IDs should be assigned automatically, `false` if manually.

DebugRequests

Gets or sets debug mode for requests.

```
bool DebugRequests { get; set; }
```

Property Value

bool

`true` if debug mode is enabled, `false` otherwise

EndpointID

Gets or sets endpoint ID for ads.

```
string? EndpointID { set; }
```

Property Value

string

A unique identifier generated on the platform's UI.

ExternalUserIds

Gets or sets external IDs of the user.

```
IEnumerable<ExternalUserID> ExternalUserIds { get; set; }
```

Property Value

IEnumerable<ExternalUserID>

External IDs of the user that could be used for ad targeting.

Remarks

See https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids.md

See Also
ExternalUserID, ExternalUserIds

LogLevel

Gets or sets the log level for native library.

```
LogLevel LogLevel { get; set; }
```

Property Value

LogLevel

The desired verbosity of the logs.

OmSdkVersion

Gets the OM SDK version.

```
string? OmSdkVersion { get; }
```

Property Value

string

The version of native OM SDK library.

SdkVersion

Gets the SDK version.

```
string? SdkVersion { get; }
```

Property Value

string

The version of native SDK library.

Timeout

Gets or sets request timeout.

```
TimeSpan Timeout { get; set; }
```

Property Value

TimeSpan

The timeout for ad requests.

Interface ISkippableAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Ad that can be skipped.

```
public interface ISkippableAd
```

Properties

SkipButtonArea

Sets skip button area factor.

```
double SkipButtonArea { set; }
```

Property Value

double

the percentage of the device screen that the skip button should cover.

Remarks

Allowed range - `0...1`.

Default value is ~`0.1`.

SkipButtonPosition

Sets position of the skip button.

```
AdElementPosition SkipButtonPosition { set; }
```

Property Value

AdElementPosition

Where the skip button appears on the screen.

Remarks

Allowed values: TopLeft, TopRight.

Default is TopRight.

SkipDelay

Sets skip button appearing delay for the ad.

```
int SkipDelay { set; }
```

Property Value

int

Number of seconds after the start of playback before the skip or close button should appear.

Remarks

Default value is ~`10.0`.

Interface ITargetingProxy

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Managed object that exposes targeting settings of native SDK.

```
public interface ITargetingProxy
```

Extension Methods
TargetingProxyExtensions.Apply(ITargetingProxy, TargetingData) ,
TargetingProxyExtensions.TakeSnapshot(ITargetingProxy)

Properties

Coordinate

Gets or sets user's base location.

```
Vector2? Coordinate { get; set; }
```

Property Value

Vector2?

> Location of the user’s home base.

Remarks

This is not necessarily their current location.

Domain

Gets or sets app domain.

```
string? Domain { get; set; }
```

Property Value

string

Domain of the app (e.g., `mygame.foo.com`).

ExternalUserIds

Gets or sets external IDs of the user.

```
IEnumerable<ExternalUserID> ExternalUserIds { get; set; }
```

Property Value

IEnumerable<ExternalUserID>

External IDs of the user that could be used for ad targeting.

Remarks

See https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids.md

See Also
ExternalUserID

IsSubjectToCOPPA

Gets or sets if request is subject to the COPPA.

```
bool? IsSubjectToCOPPA { get; set; }
```

Property Value

bool?

`true` if this request is subject to the COPPA regulations established by the USA FTC, `false` otherwise.

ItunesID

Gets or sets iTunes ID.

```
string? ItunesID { get; set; }
```

Property Value

string

The app identifier in iTunes.

Keywords

Gets current keywords.

```
IEnumerable<string?> Keywords { get; }
```

Property Value

IEnumerable<string>

Keywords, interests, or intent.

See Also
AddKeyword(string)

PublisherName

Gets or sets publisher name.

```
string? PublisherName { get; set; }
```

Property Value

string

App's publisher name.

SourceApp

Gets or sets source app.

```
string? SourceApp { get; set; }
```

Property Value

string

ID of publisher app in Apple's App Store.

StoreURL

Gets or sets store URL.

```
string? StoreURL { get; set; }
```

Property Value

string

App store URL for an installed app.

UserCustomData

Gets or sets custom data for user.

```
string? UserCustomData { get; set; }
```

Property Value

string

Optional feature to pass bidder data that was set in the exchange's cookie.

Remarks

The string must be in base85 cookie safe characters and be in any format.

UserExtJson

Gets or sets `ext` content for ORTB `user`.

```
string? UserExtJson { get; set; }
```

Property Value

string

Exchange-specific extensions to OpenRTB.

Remarks

Should be a valid JSON object or array.

Methods

AddKeyword(string)

Adds a keyword.

```
void AddKeyword(string keyword)
```

Parameters

`keyword` string

Keyword, interest, or intent.

See Also

Keywords

Class InterstitialAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Api.dll

A full page ad experience at natural transition points, such as a page change or an app launch. Interstitials use a close button that removes the ad from the user's experience.

```
public class InterstitialAd : IInterstitialAd, IDisposable, IAdUnit,
IMultiformattedAd, IResizableAd, ISkippableAd, IMuteableAd, ICloseableAd, IModalAd,
ILoadableAd, IAdEventSource, IModalAdEventSource
```

Inheritance

object ← InterstitialAd

Implements

IInterstitialAd, IDisposable, IAdUnit, IMultiformattedAd, IResizableAd, ISkippableAd, IMuteableAd, ICloseableAd, IModalAd, ILoadableAd, IAdEventSource, IModalAdEventSource

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

InterstitialAd()

Initializes a new instance of the InterstitialAd class.

```
public InterstitialAd()
```

Properties

AdSizes

Sets ad sizes for the ad.

```
public IEnumerable<AdSize> AdSizes { set; }
```

Property Value

IEnumerable<AdSize>

Sizes that ad's content may take.

AdUnitFormats

Sets formats for the ad unit.

```
public IEnumerable<AdFormat> AdUnitFormats { set; }
```

Property Value

IEnumerable<AdFormat>

Ad formats that are allowed into the ad.

CloseButtonArea

Sets close button area factor.

```
public double CloseButtonArea { set; }
```

Property Value

double

the percentage of the device screen that the close button should cover.

Remarks

Allowed range - `0...1`.

Default value is ~`0.1`.

CloseButtonPosition

Sets position of the close button.

```
public AdElementPosition CloseButtonPosition { set; }
```

Property Value

AdElementPosition

Where the close button appears on the screen.

Remarks

Allowed values: TopLeft, TopRight.

Default is TopRight.

EndpointID

Sets the endpoint ID for the ad.

```
public string? EndpointID { set; }
```

Property Value

string

Unique endpoint identifier generated on the Appstock platform's UI.

Loaded

Gets if ad has loaded.

```
public bool Loaded { get; }
```

Property Value

bool

`true` if ad has successfully loaded, `false` otherwise.

Muted

Sets if ad is to be muted.

```
public bool Muted { set; }
```

Property Value

bool

`true` if muted, `false` otherwise.

PlacementID

Sets the placement ID for the ad.

```
public string? PlacementID { set; }
```

Property Value

string

Unique placement identifier generated on the Appstock platform's UI.

SkipButtonArea

Sets skip button area factor.

```
public double SkipButtonArea { set; }
```

Property Value

double

the percentage of the device screen that the skip button should cover.

Remarks

Allowed range - `0...1`.

Default value is ~`0.1`.

SkipButtonPosition

Sets position of the skip button.

```
public AdElementPosition SkipButtonPosition { set; }
```

Property Value

AdElementPosition

Where the skip button appears on the screen.

Remarks

Allowed values: TopLeft, TopRight.

Default is TopRight.

SkipDelay

Sets skip button appearing delay for the ad.

```
public int SkipDelay { set; }
```

Property Value

int

Number of seconds after the start of playback before the skip or close button should appear.

Remarks

Default value is ~`10.0`.

SoundButtonVisible

Sets if `Mute` button is visible when the ad is on display.

```
public bool SoundButtonVisible { set; }
```

Property Value

bool

`true` if visible, `false` otherwise.

Methods

Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

Dispose(bool)

Releases the unmanaged resources used by the class and optionally releases the managed resources.

```
protected virtual void Dispose(bool disposing)
```

Parameters

`disposing` bool

`true` to release both managed and unmanaged resources; `false` to release only unmanaged resources.

~InterstitialAd()

```
protected ~InterstitialAd()
```

LoadAd()

Load the content of the ad.

```
public void LoadAd()
```

Show(Action?)

Show the ad on screen.

```
public void Show(Action? actionOnUIThread = null)
```

Parameters

`actionOnUIThread` Action

Something you want to execute immediately when the ad is being added to the screen

`actionOnUIThread` is invoked on **NATIVE UI THREAD** which **MAY DIFFER from UNITY'S MAIN THREAD**.

Events

OnAdClicked

Fired when ad was clicked on.

```
public event Action? OnAdClicked
```

Event Type

Action

OnAdClosed

Fired when ad was closed (from fullscreen mode).

```
public event Action? OnAdClosed
```

Event Type

Action

OnAdDisplayed

Fired when ad did appear on screen.

```
public event Action? OnAdDisplayed
```

Event Type

Action

OnAdFailed

Fired when ad failed to load.

```
public event Action<AdError?>? OnAdFailed
```

Event Type

Action<AdError?>

OnAdLoaded

Fired when ad finished loading.

```
public event Action<AdInfo?>? OnAdLoaded
```

Event Type

Action<AdInfo?>

Enum LogLevel

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

The desired verbosity of logs by native library.

```
public enum LogLevel
```

Fields

`Debug = 3`

Errors, warnings and debug.

`Error = 1`

Errors only.

`None = 0`

No logs.

`Warning = 2`

Errors and warnings.

See Also

LogLevel
logLevel
logLevel

Enum OptionalBool

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Represents `bool?` -- i.e. Nullable<T> of bool -- that can be `true`, `false` or `null`.

```
[Serializable]
public enum OptionalBool
```

Extension Methods

OptionalBoolExtensions.IsTrue(OptionalBool) , OptionalBoolExtensions.ToNullableBool(OptionalBool)

Fields

`False = 1`

`false`

`NotAssigned = 0`

`null`

`True = 2`

`true`

Class OptionalBoolExtensions

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Extensions for OptionalBool.

```
public static class OptionalBoolExtensions
```

Inheritance

object ← OptionalBoolExtensions

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

IsTrue(OptionalBool)

Checks if OptionalBool is True.

```
public static bool IsTrue(this OptionalBool value)
```

Parameters

`value` OptionalBool

OptionalBool to check.

Returns

bool

`true` if `value` is True, `false` otherwise.

ToNullableBool(OptionalBool)

Converts OptionalBool to a Nullable<T> of bool.

```
public static bool? ToNullableBool(this OptionalBool value)
```

Parameters

`value` OptionalBool

The OptionalBool to convert.

Returns

bool?

The converted Nullable<T> of bool.

ToOptionalBool(bool?)

Converts Nullable<T> of bool to an OptionalBool.

```
public static OptionalBool ToOptionalBool(this bool? value)
```

Parameters

`value` bool?

The Nullable<T> of bool to convert.

Returns

OptionalBool

The converted OptionalBool.

Class RewardedAd

Namespace: AppstockSDK.Api

Assembly: Appstock.Api.dll

Rewarded ads reward users with in-app items for interacting with video ads, playable ads, or surveys.

```
public class RewardedAd : IRewardedAd, IDisposable, IAdUnit, IMultiformattedAd,
IResizableAd, IMuteableAd, ICloseableAd, IModalAd, ILoadableAd, IAdEventSource,
IModalAdEventSource, IRewardingAdEventSource
```

Inheritance

object ← RewardedAd

Implements

IRewardedAd, IDisposable, IAdUnit, IMultiformattedAd, IResizableAd, IMuteableAd, ICloseableAd, IModalAd, ILoadableAd, IAdEventSource, IModalAdEventSource, IRewardingAdEventSource

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

RewardedAd()

Initializes a new instance of the RewardedAd class.

```
public RewardedAd()
```

Properties

AdSizes

Sets ad sizes for the ad.

```
public IEnumerable<AdSize> AdSizes { set; }
```

Property Value

IEnumerable<AdSize>

Sizes that ad's content may take.

AdUnitFormats

Sets formats for the ad unit.

```
public IEnumerable<AdFormat> AdUnitFormats { set; }
```

Property Value

IEnumerable<AdFormat>

Ad formats that are allowed into the ad.

CloseButtonArea

Sets close button area factor.

```
public double CloseButtonArea { set; }
```

Property Value

double

the percentage of the device screen that the close button should cover.

Remarks

Allowed range - `0...1`.

Default value is ~`0.1`.

CloseButtonPosition

Sets position of the close button.

```
public AdElementPosition CloseButtonPosition { set; }
```

Property Value

AdElementPosition

Where the close button appears on the screen.

Remarks

Allowed values: TopLeft, TopRight.

Default is TopRight.

EndpointID

Sets the endpoint ID for the ad.

```
public string? EndpointID { set; }
```

Property Value

string

Unique endpoint identifier generated on the Appstock platform's UI.

Loaded

Gets if ad has loaded.

```
public bool Loaded { get; }
```

Property Value

bool

`true` if ad has successfully loaded, `false` otherwise.

Muted

Sets if ad is to be muted.

```
public bool Muted { set; }
```

Property Value

bool

`true` if muted, `false` otherwise.

PlacementID

Sets the placement ID for the ad.

```
public string? PlacementID { set; }
```

Property Value

string

Unique placement identifier generated on the Appstock platform's UI.

SoundButtonVisible

Sets if `Mute` button is visible when the ad is on display.

```
public bool SoundButtonVisible { set; }
```

Property Value

bool

`true` if visible, `false` otherwise.

Methods

Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

Dispose(bool)

Releases the unmanaged resources used by the class and optionally releases the managed resources.

```
protected virtual void Dispose(bool disposing)
```

Parameters

`disposing` bool

`true` to release both managed and unmanaged resources; `false` to release only unmanaged resources.

~RewardedAd()

```
protected ~RewardedAd()
```

LoadAd()

Load the content of the ad.

```
public void LoadAd()
```

Show(Action?)

Show the ad on screen.

```
public void Show(Action? actionOnUIThread = null)
```

Parameters

`actionOnUIThread` Action

Something you want to execute immediately when the ad is being added to the screen

`actionOnUIThread` is invoked on **NATIVE UI THREAD** which **MAY DIFFER from UNITY'S MAIN THREAD**.

Events

OnAdClicked

Fired when ad was clicked on.

```
public event Action? OnAdClicked
```

Event Type

Action

OnAdClosed

Fired when ad was closed (from fullscreen mode).

```
public event Action? OnAdClosed
```

Event Type

Action

OnAdDisplayed

Fired when ad did appear on screen.

```
public event Action? OnAdDisplayed
```

Event Type

Action

OnAdFailed

Fired when ad failed to load.

```
public event Action<AdError?>? OnAdFailed
```

Event Type

Action<AdError?>

OnAdLoaded

Fired when ad finished loading.

```
public event Action<AdInfo?>? OnAdLoaded
```

Event Type

Action<AdInfo?>

OnReward

Fired when user deserved the reward.

```
public event Action<AdReward?>? OnReward
```

Event Type

Action<AdReward?>

Class SdkConfig

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Data container for writeable properties of ISdkProxy.

```
[Serializable]
public class SdkConfig
```

Inheritance

object ← SdkConfig

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

For exposing controls via Unity Editor.

Fields

adRequestTimeout

Gets or sets timeout for creative requests.

```
[Tooltip("The time interval allowed for a creative to load before it is considered
a failure.")]
public SerializableOptional<float> adRequestTimeout
```

Field Value

SerializableOptional<float>

The time interval allowed for a creative to load before it is considered a failure.

Remarks

In seconds

See Also

AdRequestTimeout

adRequestTimeoutPreRenderContent

Gets or sets timeout for video and interstitial creatves.

```
[Tooltip("The time interval allowed for video and interstitial creatives to load.")]
public SerializableOptional<float> adRequestTimeoutPreRenderContent
```

Field Value

SerializableOptional<float>

The time interval allowed for video and interstitial creatives to load.

Remarks

In seconds

See Also

AdRequestTimeoutPreRenderContent

assignNativeAssetID

Gets or sets whether the asset ID for native ads should be manually assigned.

```
[Tooltip("Determines whether the asset ID for native ads should be manually assigned.")]
public OptionalBool assignNativeAssetID
```

Field Value

OptionalBool

`true` if IDs should be assigned automatically, `false` if manually.

See Also

AssignNativeAssetID

debugRequests

Gets or sets debug mode for requests.

```
[Tooltip("Enables or disables debug mode for requests.")]
public OptionalBool debugRequests
```

Field Value

OptionalBool

`true` if debug mode is enabled, `false` otherwise

See Also
DebugRequests

endpointID

Gets or sets endpoint ID for ads.

```
[Tooltip("A unique identifier generated on the platform's UI.")]
public string? endpointID
```

Field Value

string

A unique identifier generated on the platform's UI.

See Also
EndpointID

externalUserIds

Gets or sets external IDs of the user.

```
[Tooltip("External IDs of the user that could be used for ad targeting.")]
public List<ExternalUserID>? externalUserIds
```

Field Value

List<ExternalUserID>

External IDs of the user that could be used for ad targeting.

Remarks

See
https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids.md

See Also

ExternalUserID, ExternalUserIds, ExternalUserIds

logLevel

Gets or sets the log level for native library.

```
[Tooltip("Sets the desired verbosity of the logs.")]
public SerializableOptional<LogLevel> logLevel
```

Field Value

SerializableOptional<LogLevel>

The desired verbosity of the logs.

See Also

LogLevel

timeout

Gets or sets request timeout.

```
[Header("Timeouts in seconds")]
[Tooltip("The timeout for ad requests.")]
public SerializableOptional<float> timeout
```

Field Value

SerializableOptional<float>

The timeout for ad requests.

Remarks

In seconds

See Also

Timeout

Class SdkConfigSnapshot

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Data container for readable properties of ISdkProxy.

```
[Serializable]
public class SdkConfigSnapshot : IEquatable<SdkConfigSnapshot>, IEquatable<SdkConfig>
```

Inheritance

object ← SdkConfigSnapshot

Implements

IEquatable<SdkConfigSnapshot>, IEquatable<SdkConfig>

Inherited Members

object.Equals(object, object) , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

To simplify logging the state.

Constructors

SdkConfigSnapshot(ISdkProxy)

Collects all readable properties from `sdkProxy`.

```
public SdkConfigSnapshot(ISdkProxy sdkProxy)
```

Parameters

`sdkProxy` ISdkProxy

Proxy to get properties from.

Fields

adRequestTimeoutSeconds

Gets or sets timeout for creative requests.

```
public double adRequestTimeoutSeconds
```

Field Value

double

The time interval allowed for a creative to load before it is considered a failure.

Remarks

In seconds

See Also

AdRequestTimeout

adRequestTimeoutSecondsPreRenderContent

Gets or sets timeout for video and interstitial creatves.

```
public double adRequestTimeoutSecondsPreRenderContent
```

Field Value

double

The time interval allowed for video and interstitial creatives to load.

Remarks

In seconds

See Also

AdRequestTimeoutPreRenderContent

assignNativeAssetID

Gets or sets whether the asset ID for native ads should be manually assigned.

```
public bool assignNativeAssetID
```

Field Value

bool

`true` if IDs should be assigned automatically, `false` if manually.

See Also
AssignNativeAssetID

debugRequests

Gets or sets debug mode for requests.

```
public bool debugRequests
```

Field Value

bool

`true` if debug mode is enabled, `false` otherwise

See Also
DebugRequests

externalUserIds

Gets or sets external IDs of the user.

```
public List<ExternalUserID> externalUserIds
```

Field Value

List<ExternalUserID>

External IDs of the user that could be used for ad targeting.

Remarks

See
https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids.md

See Also

ExternalUserID, ExternalUserIds, ExternalUserIds

logLevel

Gets or sets the log level for native library.

```
public LogLevel logLevel
```

Field Value

LogLevel

The desired verbosity of the logs.

See Also
LogLevel

omSdkVersion

Gets the OM SDK version.

```
public string? omSdkVersion
```

Field Value

string

The version of native OM SDK library.

See Also
OmSdkVersion

sdkVersion

Gets the SDK version.

```
public string? sdkVersion
```

Field Value

string

The version of native SDK library.

See Also
SdkVersion

timeoutSeconds

Gets or sets request timeout.

```
public double timeoutSeconds
```

Field Value

double

The timeout for ad requests.

Remarks

In seconds

See Also
Timeout

Methods

Equals(SdkConfig)

Indicates whether the current object is equal to another object of the same type.

```
public bool Equals(SdkConfig other)
```

Parameters

`other` SdkConfig

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(SdkConfigSnapshot)

Indicates whether the current object is equal to another object of the same type.

```
public bool Equals(SdkConfigSnapshot other)
```

Parameters

`other` SdkConfigSnapshot

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(object?)

Determines whether the specified object is equal to the current object.

```
public override bool Equals(object? obj)
```

Parameters

`obj` object

The object to compare with the current object.

Returns

bool

true if the specified object is equal to the current object; otherwise, false.

GetHashCode()

Serves as the default hash function.

```
public override int GetHashCode()
```

Returns

int

A hash code for the current object.

UnequalFields(SdkConfig, string[]?)

Searches for value differences between read and written configs.

```
public IEnumerable<string> UnequalFields(SdkConfig other, string[]? fieldsToIgnore = null)
```

Parameters

`other` SdkConfig

Reference config for comparing the snapshot with.

`fieldsToIgnore` string[]

What fields should not be returned even when values differ.

Returns

IEnumerable<string>

Field (property) names. Occasionally with additional info.

Class SdkProxyExtensions

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Extension methods for ISdkProxy.

```
public static class SdkProxyExtensions
```

Inheritance

object ← SdkProxyExtensions

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

Apply(ISdkProxy, SdkConfig)

Sets properties in `sdkProxy` to values that are set in `sdkConfig`.

```
public static void Apply(this ISdkProxy sdkProxy, SdkConfig sdkConfig)
```

Parameters

`sdkProxy` ISdkProxy

Proxy to set properties in.

`sdkConfig` SdkConfig

Serializable container of values.

Struct SerializableOptional<T>

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Represents Nullable<T> for simple value types.

```
[Serializable]
public struct SerializableOptional<T> : IEquatable<SerializableOptional<T>> where T
: struct
```

Type Parameters

T

Type of potentially contained value.

Implements

IEquatable<SerializableOptional<T>>

Inherited Members

ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Remarks

Custom property drawer included.

Fields

enabled

Gets or sets if value is present.

```
public bool enabled
```

Field Value

bool

`true` if value is the value of whole property, `false` if the whole property is `null`.

value

Gets or sets value of internal container.

```
public T value
```

Field Value

T

Value container for when enabled is `true`.

Remarks

Ignored if enabled is `false`.

Methods

Equals(SerializableOptional<T>)

Indicates whether the current object is equal to another object of the same type.

```
public readonly bool Equals(SerializableOptional<T> other)
```

Parameters

`other` SerializableOptional<T>

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(object?)

Indicates whether this instance and a specified object are equal.

```
public override readonly bool Equals(object? obj)
```

Parameters

`obj` object

The object to compare with the current instance.

Returns

bool

true if `obj` and this instance are the same type and represent the same value; otherwise, false.

GetHashCode()

Returns the hash code for this instance.

```
public override readonly int GetHashCode()
```

Returns

int

A 32-bit signed integer that is the hash code for this instance.

Operators

explicit operator T?(SerializableOptional<T>)

Converts SerializableOptional<T> to a Nullable<T>.

```
public static explicit operator T?(SerializableOptional<T> optional)
```

Parameters

`optional` SerializableOptional<T>

The SerializableOptional<T> to convert.

Returns

T?

The converted Nullable<T>.

implicit operator SerializableOptional<T>(T?)

Converts Nullable<T> to a SerializableOptional<T>.

```
public static implicit operator SerializableOptional<T>(T? nullableValue)
```

Parameters

`nullableValue` T?

The Nullable<T> to convert.

Returns

SerializableOptional<T>

The converted SerializableOptional<T>.

Class TargetingData

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

```
[Serializable]
public class TargetingData : IEquatable<TargetingData>
```

Inheritance

object ← TargetingData

Implements

IEquatable<TargetingData>

Inherited Members

object.Equals(object, object) , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Fields

coordinate

Gets or sets user's base location.

```
[Tooltip("ORTB: user.geo.lat, user.geo.lon")]
public SerializableOptional<Vector2> coordinate
```

Field Value

SerializableOptional<Vector2>

Location of the user’s home base.

Remarks

This is not necessarily their current location.

See Also

Coordinate

domain

Gets or sets app domain.

```
[Header("Application")]
[Tooltip("ORTB: app.domain")]
public string? domain
```

Field Value

string

Domain of the app (e.g., `mygame.foo.com`).

See Also
Domain

externalUserIds

Gets or sets external IDs of the user.

```
[Tooltip("ORTB: usr.ext.eids")]
public List<ExternalUserID>? externalUserIds
```

Field Value

List<ExternalUserID>

External IDs of the user that could be used for ad targeting.

Remarks

See
https://github.com/InteractiveAdvertisingBureau/openrtb/blob/main/extensions/2.x_official_extensions/eids.md

See Also
ExternalUserID, ExternalUserIds

isSubjectToCOPPA

Gets or sets if request is subject to the COPPA.

```
[Header("Regulations")]
[Tooltip("ORTB: regs.coppa")]
```

```
public OptionalBool isSubjectToCOPPA
```

Field Value

OptionalBool

`true` if this request is subject to the COPPA regulations established by the USA FTC, `false` otherwise.

See Also
IsSubjectToCOPPA

itunesID

Gets or sets iTunes ID.

```
[Header("Application (iOS)")]
[Tooltip("ORTB: app.bundle")]
public string? itunesID
```

Field Value

string

The app identifier in iTunes.

See Also
ItunesID

keywords

Gets or sets keywords.

```
[Tooltip("user.keywords")]
public List<string?>? keywords
```

Field Value

List<string>

Keywords, interests, or intent.

See Also
Keywords, AddKeyword(string)

publisherName

Gets or sets publisher name.

```
[Tooltip("ORTB: app.publisher.name")]
public string? publisherName
```

Field Value

string

App's publisher name.

See Also

PublisherName

sourceApp

Gets or sets source app.

```
[Tooltip("ORTB: imp[].ext.skadn.sourceapp")]
public string? sourceApp
```

Field Value

string

ID of publisher app in Apple's App Store.

See Also

SourceApp

storeURL

Gets or sets store URL.

```
[Tooltip("ORTB: app.storeurl")]
public string? storeURL
```

Field Value

string

App store URL for an installed app.

See Also
StoreURL

userCustomData

Gets or sets custom data for user.

```
[Header("User")]
[Tooltip("ORTB: user.customdata")]
public string? userCustomData
```

Field Value

string

Optional feature to pass bidder data that was set in the exchange's cookie.

Remarks

The string must be in base85 cookie safe characters and be in any format.

See Also
UserCustomData

userExtJson

Gets or sets `ext` content for ORTB `user`.

```
[Header("User (iOS)")]
[Tooltip("ORTB: user.ext")]
public ExtSlot userExtJson
```

Field Value

ExtSlot

Exchange-specific extensions to OpenRTB.

Remarks

Should be a valid JSON object or array.

See Also

UserExtJson

Methods

Equals(TargetingData?)

Indicates whether the current object is equal to another object of the same type.

```
public bool Equals(TargetingData? other)
```

Parameters

`other` TargetingData

An object to compare with this object.

Returns

bool

true if the current object is equal to the `other` parameter; otherwise, false.

Equals(object?)

Determines whether the specified object is equal to the current object.

```
public override bool Equals(object? obj)
```

Parameters

`obj` object

The object to compare with the current object.

Returns

bool

true if the specified object is equal to the current object; otherwise, false.

GetHashCode()

Serves as the default hash function.

```
public override int GetHashCode()
```

Returns

int

A hash code for the current object.

UnequalFields(TargetingData, string[]?)

Searches for value differences between read and written data.

```
public IEnumerable<string> UnequalFields(TargetingData other, string[]? fieldsToIgnore
= null)
```

Parameters

`other` TargetingData

Reference data for comparing with.

`fieldsToIgnore` string[]

What fields should not be returned even when values differ.

Returns

IEnumerable<string>

Field (property) names. Occasionally with additional info.

Class TargetingProxyExtensions

Namespace: AppstockSDK.Api

Assembly: Appstock.Core.dll

Extension methods for ITargetingProxy.

```
public static class TargetingProxyExtensions
```

Inheritance

object ← TargetingProxyExtensions

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

Apply(ITargetingProxy, TargetingData)

Sets properties in `targetingProxy` to values that are set in `targetingData`.

```
public static void Apply(this ITargetingProxy targetingProxy,
TargetingData targetingData)
```

Parameters

`targetingProxy` ITargetingProxy

Proxy to set properties in.

`targetingData` TargetingData

Serializable container of values.

TakeSnapshot(ITargetingProxy)

Gets properties from `targetingProxy`.

```
public static TargetingData TakeSnapshot(this ITargetingProxy targetingProxy)
```

Parameters

`targetingProxy` ITargetingProxy

Proxy to get properties from.

Returns

TargetingData

Serializable container of values.

Namespace AppstockSDK.Api.Native.Data.Request

Classes

AdUnitDataExtensions

Extensions for AdUnitData.

Structs

AdUnitData

Top-level object to build native ad request (aka INativeAdLoader) from. Describes both WHAT and WHERE to request.

AssetData

Describes a requested data asset.

AssetImage

Describes a requested image asset.

AssetTitle

Describes a requested title asset.

Assets

A collection of assets to be requested.

ConfigWarnings

Holds validation warnings for displaying in Unity Editor.

EventTracker

The event trackers object specifies the types of events the bidder can request to be tracked in the bid response.

EventTrackers

Collection of event trackers.

Parameters

Content part of Native Ad request.

Enums

AssetData.DataType

Type of data asset.

AssetImage.ImageType

Type of image asset.

EventTracker.EventType

Type of event available for tracking.

EventTracker.TrackingMethod

Types of tracking available for the given event.

Parameters.ContextSubType

Next-level context in which the ad appears.

Parameters.ContextType

The context in which the ad appears - what type of content is surrounding the ad on the page at a high level.

Parameters.PlacementType

The FORMAT of the ad, separate from the surrounding context.

Struct AdUnitData

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Top-level object to build native ad request (aka INativeAdLoader) from. Describes both WHAT and WHERE to request.

```
[Serializable]
public struct AdUnitData
```

Inherited Members

ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

endpointID

Sets the endpoint ID for the ad.

```
[Tooltip("Unique endpoint identifier generated on the Appstock platform's UI.")]
public string? endpointID
```

Field Value

string

Unique endpoint identifier generated on the Appstock platform's UI.

parameters

Gets or sets ad parameters.

```
public Parameters parameters
```

Field Value

Parameters

Native ad content parameters.

placementID

Sets the placement ID for the ad.

```
[Tooltip("Unique placement identifier generated on the Appstock platform's UI.")]
public string? placementID
```

Field Value

string

Unique placement identifier generated on the Appstock platform's UI.

Properties

Warnings

Gets the warnings for current state.

```
public readonly IEnumerable<string> Warnings { get; }
```

Property Value

IEnumerable<string>

Warnings for unpopulated or conflicting values inside.

Class AdUnitDataExtensions

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Api.dll

Extensions for AdUnitData.

```
public static class AdUnitDataExtensions
```

Inheritance

object ← AdUnitDataExtensions

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

BuildAdLoader(AdUnitData)

Uses a data from `adUnitData` to build a loader for native ad.

```
public static INativeAdLoader BuildAdLoader(this AdUnitData adUnitData)
```

Parameters

`adUnitData` AdUnitData

Serializable data container with native ad request settings to build loader for.

Returns

INativeAdLoader

Built INativeAdLoader.

Struct AssetData

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Describes a requested data asset.

```
[Serializable]
public struct AssetData
```

Inherited Members

ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

dataExt

An optional extension for additional data.

```
[Tooltip("An optional extension for additional data.\nORTB: assets[i].data.ext")]
public ExtSlot dataExt
```

Field Value

ExtSlot

> This object is a placeholder that may contain custom JSON agreed to by the parties to support flexibility beyond the standard defined in ORTB specification.

Remarks

ORTB: `assets[i].data.ext`

dataType

Gets or sets data type.

```
[Tooltip("Type of the data element supported by the publisher.\nThe publisher can
display this information in an appropriate format.\nORTB: assets[i].data.type")]
public AssetData.DataType dataType
```

Field Value

AssetData.DataType

Type ID of the element supported by the publisher.

The publisher can display this information in an appropriate format.

Remarks

ORTB: `assets[i].data.type`

See Also
AssetData.DataType

ext

An optional extension for additional data.

```
[Tooltip("An optional extension for additional data.\nORTB: assets[i].ext")]
public ExtSlot ext
```

Field Value

ExtSlot

This object is a placeholder that may contain custom JSON agreed to by the parties to support flexibility beyond the standard defined in ORTB specification.

Remarks

ORTB: `assets[i].ext`

length

Gets or sets the maximum length.

```
[Tooltip("Maximum length of the text in the element's response.\nORTB:
assets[i].data.len")]
public SerializableOptional<int> length
```

Field Value

SerializableOptional<int>

Maximum length of the text in the element's response.

Remarks

ORTB: `assets[i].data.len`

required

Gets or sets if asset is required.

```
[Tooltip("If checked, exchange will not accept a bid without this
asset.\nORTB: assets[i].required")]
public bool required
```

Field Value

bool

`true` if required (exchange will not accept a bid without it), `false` otherwise.

Remarks

ORTB: `assets[i].required`

Enum AssetData.DataType

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Type of data asset.

```
[Serializable]
public enum AssetData.DataType
```

Fields

`Address = 9`

Address.

`CallToAction = 12`

CTA description - descriptive text describing a ‘call to action’ button for the destination URL.

Optional. Max 15 or longer.

`Custom = 500`

Reserved for Exchange specific usage numbered above 500

`Description = 2`

Descriptive text associated with the product or service being advertised. Longer length of text in response may be truncated or ellipsed by the exchange.

Recommended. Max 140 or longer.

`Description2 = 10`

Additional descriptive text associated with the product or service being advertised.

`DisplayURL = 11`

Display URL for the text ad. To be used when sponsoring entity doesn't own the content. IE sponsored by BRAND on SITE (where SITE is transmitted in this field).

`Downloads = 5`

Number downloads/installs of this product

`Likes = 4`

Number of social ratings or “likes” of the product being offered to the user.

`None = 0`

Not assigned.

Should not be used.

`Phone = 8`

Phone number.

`Price = 6`

Price for product / app / in-app purchase. Value should include currency symbol in localised format.

`Rating = 3`

Rating of the product being offered to the user. For example an app’s rating in an app store from 0-5.

Optional. 0-5 integer formatted as string.

`SalePrice = 7`

Sale price that can be used together with price to indicate a discounted price compared to a regular price. Value should include currency symbol in localised format.

`Sponsored = 1`

Sponsored By message where response should contain the brand name of the sponsor.

Required. Max 25 or longer.

See Also

dataType

Struct AssetImage

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Describes a requested image asset.

```
[Serializable]
public struct AssetImage
```

Inherited Members

ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

ext

An optional extension for additional data.

```
[Tooltip("An optional extension for additional data.\nORTB: assets[i].ext")]
public ExtSlot ext
```

Field Value

ExtSlot

> This object is a placeholder that may contain custom JSON agreed to by the parties to support flexibility beyond the standard defined in ORTB specification.

Remarks

ORTB: `assets[i].ext`

height

Gets or sets image height.

```
[Tooltip("Height of the image in pixels.\nORTB: assets[i].img.h")]
public SerializableOptional<int> height
```

Field Value

SerializableOptional<int>

Height of the image in pixels.

Remarks

ORTB: `assets[i].img.h`

imageExt

An optional extension for additional data.

```
[Tooltip("An optional extension for additional data.\nORTB: assets[i].img.ext")]
public ExtSlot imageExt
```

Field Value

ExtSlot

This object is a placeholder that may contain custom JSON agreed to by the parties to support flexibility beyond the standard defined in ORTB specification.

Remarks

ORTB: `assets[i].img.ext`

imageType

Gets or sets image type.

```
[Tooltip("Type of the image element supported by the publisher.\nThe publisher can
display this information in an appropriate format.\nORTB: assets[i].img.type")]
public AssetImage.ImageType imageType
```

Field Value

AssetImage.ImageType

Type ID of the image element supported by the publisher.

The publisher can display this information in an appropriate format.

Remarks

ORTB: `assets[i].img.type`

See Also

AssetImage.ImageType

mimeTypes

Gets or sets mime types.

```
[Tooltip("Whitelist of content MIME types supported.\nORTB: assets[i].img.mimes")]
public string[]? mimeTypes
```

Field Value

string[]

Whitelist of content MIME types supported.

Remarks

ORTB: `assets[i].img.mimes`

minSize

Gets or sets minimal image size.

```
[Tooltip("The minimum requested width and height of the image in pixels.\nThis
option should be used for any rescaling of images by the client.\nORTB:
assets[i].img.wmin, assets[i].img.hmin")]
public Vector2Int minSize
```

Field Value

Vector2Int

The minimum requested width and height of the image in pixels.

This option should be used for any rescaling of images by the client.

Remarks

ORTB: `assets[i].img.wmin`, `assets[i].img.hmin`

required

Gets or sets if asset is required.

```
[Tooltip("If checked, exchange will not accept a bid without this
asset.\nORTB: assets[i].required")]
public bool required
```

Field Value

bool

`true` if required (exchange will not accept a bid without it), `false` otherwise.

Remarks

ORTB: `assets[i].required`

width

Gets or sets image width.

```
[Tooltip("Width of the image in pixels.\nORTB: assets[i].img.w")]
public SerializableOptional<int> width
```

Field Value

SerializableOptional<int>

Width of the image in pixels.

Remarks

ORTB: `assets[i].img.w`

Enum AssetImage.ImageType

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Type of image asset.

```
[Serializable]
public enum AssetImage.ImageType
```

Fields

`Custom = 500`

Reserved for Exchange specific usage numbered above 500.

`Icon = 1`

Icon image.

Optional.

max height: at least 50

aspect ratio: 1:1

`Main = 3`

Large image preview for the ad.

At least one of 2 size variants required:

Small variant	**max height**	at least 200
	max width	at least 200, 267, or 382
	aspect ratio	1:1, 4:3, or 1.91:1
Large variant	**max height**	at least 627
	max width	at least 627, 836, or 1198

aspect ratio	1:1, 4:3, or 1.91:1

`None = 0`

Not assigned.

Should not be used.

See Also

imageType

Struct AssetTitle

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Describes a requested title asset.

```
[Serializable]
public struct AssetTitle
```

Inherited Members

ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

length

Gets or sets the maximum length.

```
[Tooltip("Maximum length of the text in the title element.\nORTB:
assets[i].title.len")]
public int length
```

Field Value

int

Maximum length of the text in the title element.

Remarks

Recommended to be 25, 90, or 140.

ORTB: `assets[i].title.len`

required

Gets or sets if asset is required.

```
[Tooltip("If checked, exchange will not accept a bid without this
asset.\nORTB: assets[i].required")]
public bool required
```

Field Value

bool

`true` if required (exchange will not accept a bid without it), `false` otherwise.

Remarks

ORTB: `assets[i].required`

Struct Assets

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

A collection of assets to be requested.

```
[Serializable]
public struct Assets
```

Inherited Members

ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

allowEmpty

Gets or sets if empty collection is a valid value.

```
[Tooltip("Whether collection should be forwarded to native library even
when empty.")]
public bool allowEmpty
```

Field Value

bool

`false` if empty collection prevents calling the respective setter in native library.

`true` if native setter should be called even when collection is empty.

data

Gets or sets the data assets.

```
[Tooltip("Data assets to request.")]
```

```
public AssetData[]? data
```

Field Value

AssetData[]

Data assets to request.

See Also
AssetData

images

Gets or sets the images.

```
[Tooltip("Image assets to request.")]
public AssetImage[]? images
```

Field Value

AssetImage[]

Image assets to request.

See Also
AssetImage

titles

Gets or sets the titles.

```
[Tooltip("Title assets to request.")]
public AssetTitle[]? titles
```

Field Value

AssetTitle[]

Title assets to request.

See Also

AssetTitle

Properties

Count

Gets the count of assets in collection.

```
public readonly int Count { get; }
```

Property Value

int

Assets count.

Struct ConfigWarnings

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Holds validation warnings for displaying in Unity Editor.

```
[Serializable]
public struct ConfigWarnings
```

Inherited Members
ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Remarks

Custom property drawer included.

Fields

warnings

Gets or sets warnings.

```
public List<string>? warnings
```

Field Value

List<string>

Validation warnings by contained class.

Remarks

Set in `OnValidate`.

Displayed in inspector.

See Also
https://docs.unity3d.com/Manual/execution-order.html#Editor

Struct EventTracker

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

The event trackers object specifies the types of events the bidder can request to be tracked in the bid response.

```
[Serializable]
public struct EventTracker
```

Inherited Members
ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

eventType

Gets or sets event type.

```
[Tooltip("Type of event available for tracking.\nORTB: eventtrackers[i].event")]
public EventTracker.EventType eventType
```

Field Value

EventTracker.EventType

Type of event available for tracking.

Remarks

ORTB: `eventtrackers[i].event`

methods

Gets or sets event tracking methods.

```
[Tooltip("Types of tracking available for the given event.\nORTB:
eventtrackers[i].methods")]
public EventTracker.TrackingMethod[]? methods
```

Field Value

TrackingMethod[]

Types of tracking available for the given event.

Remarks

ORTB: `eventtrackers[i].methods`

See Also

EventTracker.TrackingMethod

Enum EventTracker.EventType

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Type of event available for tracking.

```
[Serializable]
public enum EventTracker.EventType
```

Fields

`Custom = 500`

Exchange-specific.

`Impression = 1`

Impression.

`None = 0`

Not assigned.

Should not be used.

`ViewableImpression100 = 3`

100% in view for 1 second (ie GroupM standard).

`ViewableImpression50 = 2`

Visible impression using MRC definition at 50% in view for 1 second.

`ViewableVideoImpression50 = 4`

Visible impression for video using MRC definition at 50% in view for 2 seconds.

Enum EventTracker.TrackingMethod

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Types of tracking available for the given event.

```
[Serializable]
public enum EventTracker.TrackingMethod
```

Fields

`Custom = 500`

Exchange-specific.

Could include custom measurement companies such as moat, doubleverify, IAS, etc. - in this case additional elements will often be passed

`Image = 1`

Image-pixel tracking - URL provided will be inserted as a 1x1 pixel at the time of the event.

`Js = 2`

Javascript-based tracking - URL provided will be inserted as a js tag at the time of the event.

`None = 0`

Not assigned.

Should not be used.

Struct EventTrackers

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Collection of event trackers.

```
[Serializable]
public struct EventTrackers
```

Inherited Members

ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

allowEmpty

Gets or sets if empty collection is a valid value.

```
[Tooltip("Whether collection should be forwarded to native library even
when empty.")]
public bool allowEmpty
```

Field Value

bool

`false` if empty collection prevents calling the respective setter in native library.

`true` if native setter should be called even when collection is empty.

elements

```
public EventTracker[]? elements
```

Field Value

EventTracker[]

Native event trackers

Operators

implicit operator EventTrackers(EventTracker[]?)

Defines an implicit conversion of EventTracker array to a EventTrackers.

```
public static implicit operator EventTrackers(EventTracker[]? elements)
```

Parameters

`elements` EventTracker[]

The EventTracker array to convert.

Returns

EventTrackers

The converted EventTrackers.

See Also

EventTracker

Struct Parameters

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Content part of Native Ad request.

```
[Serializable]
public struct Parameters
```

Inherited Members

ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Fields

assetUrlSupport

Gets or sets aurlsupport flag.

```
[Tooltip("Whether the supply source / impression supports returning an assetsurl
instead of an asset object.\nORTB: aurlsupport")]
public OptionalBool assetUrlSupport
```

Field Value

OptionalBool

Whether the supply source / impression supports returning an assetsurl instead of an asset object.

Remarks

ORTB: `aurlsupport`

assets

Gets or sets assets.

```
[Tooltip("An array of Asset Objects.\nAny bid response must comply with the array of
elements expressed in the bid request.\nORTB: assets")]
public Assets assets
```

Field Value

Assets

An array of Asset Objects.

Any bid response must comply with the array of elements expressed in the bid request.

Remarks

ORTB: `assets`

contextSubType

Gets or sets context subtype.

```
[Tooltip("A more detailed context in which the ad appears\nORTB: contextsubtype")]
public Parameters.ContextSubType contextSubType
```

Field Value

Parameters.ContextSubType

A more detailed context in which the ad appears

Remarks

ORTB: `contextsubtype`

See Also

Parameters.ContextSubType

contextType

Gets or sets context type.

```
[Tooltip("The context in which the ad appears.\nORTB: context")]
public Parameters.ContextType contextType
```

Field Value

Parameters.ContextType

The context in which the ad appears.

Remarks

ORTB: `context`

See Also

Parameters.ContextType

dUrlSupport

Gets or sets durlsupport flag.

```
[Tooltip("Whether the supply source / impression supports returning a dco url
instead of an asset object.\nORTB: durlsupport")]
public OptionalBool dUrlSupport
```

Field Value

OptionalBool

Whether the supply source / impression supports returning a dco url instead of an asset object.

Remarks

ORTB: `durlsupport`

eventTrackers

Gets or sets event trackers.

```
[Tooltip("Specifies what type of event tracking is supported.\nORTB:
eventtrackers")]
```

```
public EventTrackers eventTrackers
```

Field Value

EventTrackers

Specifies what type of event tracking is supported.

Remarks

ORTB: `eventtrackers`

See Also
EventTracker

ext

An optional extension for additional data.

```
[Tooltip("An optional extension for additional data.\nORTB: ext")]
public ExtSlot ext
```

Field Value

ExtSlot

This object is a placeholder that may contain custom JSON agreed to by the parties to support flexibility beyond the standard defined in ORTB specification.

Remarks

ORTB: `ext`

placementCount

Gets or sets placements count.

```
[Tooltip("The number of identical placements in this Layout.\nORTB: plcmtcnt")]
public SerializableOptional<int> placementCount
```

Field Value

SerializableOptional<int>

The number of identical placements in this Layout.

Remarks

ORTB: `plcmtcnt`

placementType

Gets or sets placement type.

```
[Tooltip("The design/format/layout of the ad unit.\nORTB: plcmttype")]
public Parameters.PlacementType placementType
```

Field Value

Parameters.PlacementType

The design/format/layout of the ad unit.

Remarks

ORTB: `plcmttype`

See Also

Parameters.PlacementType

privacy

Gets or sets privacy support flag.

```
[Tooltip("Set to 1 when the native ad supports buyer-specific privacy
notice.\nORTB: privacy")]
public OptionalBool privacy
```

Field Value

OptionalBool

Set to 1 when the native ad supports buyer-specific privacy notice.

Remarks

ORTB: `privacy`

sequence

Gets or sets sequence number.

```
[Tooltip("0 for the first ad, 1 for the second ad, and so on.\nORTB: seq")]
public SerializableOptional<int> sequence
```

Field Value

SerializableOptional<int>

0 for the first ad, 1 for the second ad, and so on.

Remarks

ORTB: `seq`

See Also

parameters

Enum Parameters.ContextSubType

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

Next-level context in which the ad appears.

```
public enum Parameters.ContextSubType
```

Fields

`AppStore = 31`

Application store/marketplace.

`Article = 11`

Primarily article content (which of course could include images, etc as part of the article).

`Audio = 13`

Primarily audio content.

`ChatInstantMessage = 22`

Primarily chat/IM content.

`Custom = 500`

To be defined by the exchange.

`Email = 21`

Primarily email content.

`General = 10`

General or mixed content.

`Image = 14`

Primarily image content.

`None = 0`

Not assigned.

`ReviewSite = 32`

Product reviews site primarily (which may sell product secondarily).

`SellingProduct = 30`

Content focused on selling products, whether digital or physical.

`Social = 20`

General social content such as a general social network.

`UserGenerated = 15`

User-generated content - forums, comments, etc.

`Video = 12`

Primarily video content.

Remarks

This reflects the primary context, and does not imply no presence of other elements.

See Also

contextSubType

Enum Parameters.ContextType

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

The context in which the ad appears - what type of content is surrounding the ad on the page at a high level.

```
[Serializable]
public enum Parameters.ContextType
```

Fields

`Content = 1`

Content-centric context such as newsfeed, article, image gallery, video gallery, or similar.

`Custom = 500`

To be defined by the exchange.

`None = 0`

Not assigned.

`Product = 3`

Product context such as product listings, details, recommendations, reviews, or similar.

`Social = 2`

Social-centric context such as social network feed, email, chat, or similar.

See Also

contextType

Enum Parameters.PlacementType

Namespace: AppstockSDK.Api.Native.Data.Request

Assembly: Appstock.Core.dll

The FORMAT of the ad, separate from the surrounding context.

```
[Serializable]
public enum Parameters.PlacementType
```

Fields

`AtomicContent = 2`

In the atomic unit of the content - IE in the article page or single image page.

`Custom = 500`

To be defined by the exchange.

`FeedContent = 1`

In the feed of content - for example as an item inside the organic feed/grid/listing/carousel.

`None = 0`

Not assigned.

`OutsideContent = 3`

Outside the core content - for example in the ads section on the right rail, as a banner-style placement near the content, etc.

`RecommendationWidget = 4`

Recommendation widget, most commonly presented below the article content.

See Also

placementType

Namespace AppstockSDK.Api.Native.Data.Response

Interfaces

IDataContent
: Corresponds to the Data Object in the request, with the value filled in.

IImageContent
: Corresponds to the Image Object in the request, with the value filled in.

INativeAd
: Native object that exposes properties of the native ad response.

ITitleContent
: Corresponds to the Title Object in the request, with the value filled in.

Enums

DataContentType
: Type of data asset.

ImageContentType
: Type of image asset.

Enum DataContentType

Namespace: AppstockSDK.Api.Native.Data.Response

Assembly: Appstock.Core.dll

Type of data asset.

```
[Serializable]
public enum DataContentType
```

Fields

`Address = 9`

Address.

`CallToAction = 12`

CTA description - descriptive text describing a ‘call to action’ button for the destination URL.

Optional. Max 15 or longer.

`Custom = 500`

Reserved for Exchange specific usage numbered above 500

`Description = 2`

Descriptive text associated with the product or service being advertised. Longer length of text in response may be truncated or ellipsed by the exchange.

Recommended. Max 140 or longer.

`Description2 = 10`

Additional descriptive text associated with the product or service being advertised.

`DisplayURL = 11`

Display URL for the text ad. To be used when sponsoring entity doesn't own the content. IE sponsored by BRAND on SITE (where SITE is transmitted in this field).

`Downloads = 5`

Number downloads/installs of this product

`Likes = 4`

Number of social ratings or “likes” of the product being offered to the user.

`Phone = 8`

Phone number.

`Price = 6`

Price for product / app / in-app purchase. Value should include currency symbol in localised format.

`Rating = 3`

Rating of the product being offered to the user. For example an app’s rating in an app store from 0-5.

Optional. 0-5 integer formatted as string.

`SalePrice = 7`

Sale price that can be used together with price to indicate a discounted price compared to a regular price. Value should include currency symbol in localised format.

`Sponsored = 1`

Sponsored By message where response should contain the brand name of the sponsor.

Required. Max 25 or longer.

`Undefined = 0`

Not assigned.

Should not be used.

See Also

dataType
AssetData.DataType

Interface IDataContent

Namespace: AppstockSDK.Api.Native.Data.Response

Assembly: Appstock.Core.dll

Corresponds to the Data Object in the request, with the value filled in.

```
public interface IDataContent : IDisposable
```

Inherited Members
IDisposable.Dispose()

Properties

DataType

Gets or sets data type.

```
DataContentType? DataType { get; }
```

Property Value

DataContentType?

The type of data element.

See Also
dataType

Value

Gets or sets data value.

```
string? Value { get; }
```

Property Value

string

The formatted string of data to be displayed.

Remarks

Can contain a formatted value such as “5 stars” or “$10” or “3.4 stars out of 5”.

See Also

AssetData

Interface IImageContent

Namespace: AppstockSDK.Api.Native.Data.Response

Assembly: Appstock.Core.dll

Corresponds to the Image Object in the request, with the value filled in.

```
public interface IImageContent : IDisposable
```

Inherited Members
IDisposable.Dispose()

Properties

ImageType

Gets or sets image type.

```
ImageContentType? ImageType { get; }
```

Property Value

ImageContentType?

The type of image element.

See Also
imageType

URL

Gets or sets image URL.

```
string? URL { get; }
```

Property Value

string

URL of the image asset.

See Also

AssetImage

Interface INativeAd

Namespace: AppstockSDK.Api.Native.Data.Response

Assembly: Appstock.Core.dll

Native object that exposes properties of the native ad response.

```
public interface INativeAd : IDisposable
```

Inherited Members
IDisposable.Dispose()

Properties

CallToAction

Gets the CallToAction text.

```
string? CallToAction { get; }
```

Property Value

string

Value of first IDataContent asset with DataType equal to CallToAction.

DataObjects

Gets data assets.

```
IEnumerable<IDataContent?> DataObjects { get; }
```

Property Value

IEnumerable<IDataContent>

Data elements of ad assets collection.

IconUrl

Gets the Icon image URL.

```
string? IconUrl { get; }
```

Property Value

string

URL of first IImageContent asset with ImageType equal to Icon.

ImageUrl

Gets the Main image URL.

```
string? ImageUrl { get; }
```

Property Value

string

URL of first IImageContent asset with ImageType equal to Main.

Images

Gets image assets.

```
IEnumerable<IImageContent?> Images { get; }
```

Property Value

IEnumerable<IImageContent>

Image elements of ad assets collection.

SponsoredBy

Gets the Sponsored text.

```
string? SponsoredBy { get; }
```

Property Value

string

Value of first IDataContent asset with DataType equal to Sponsored.

Title

Gets the title string.

```
string? Title { get; }
```

Property Value

string

Text of first ITitleContent asset.

Titles

Gets title assets.

```
IEnumerable<ITitleContent?> Titles { get; }
```

Property Value

IEnumerable<ITitleContent>

Title elements of ad assets collection.

Methods

RegisterView(GameObject, IEnumerable<GameObject>?)

Adds trackers to the view.

```
bool RegisterView(GameObject container, IEnumerable<GameObject>? clickableObjects)
```

Parameters

`container` GameObject

The most relevant ancestor (Transform-wise) of ad elements.

`clickableObjects` IEnumerable<GameObject>

GameObjects with attached colliders or selectable components.

Returns

bool

`true` is tracking started successfully, `false` otherwise.

See Also
https://docs.unity3d.com/ScriptReference/Collider.html,
https://docs.unity3d.com/ScriptReference/Collider2D.html,
https://docs.unity3d.com/Packages/com.unity.ugui@3.0/api/UnityEngine.UI.Selectable.html

Events

OnAdClicked

Fired when ad is clicked.

```
event Action OnAdClicked
```

Event Type

Action

OnAdExpired

Fired when ad expired.

```
event Action OnAdExpired
```

Event Type

Action

OnAdImpression

Fired when impression is tracked.

```
event Action OnAdImpression
```

Event Type

Action

Interface ITitleContent

Namespace: AppstockSDK.Api.Native.Data.Response

Assembly: Appstock.Core.dll

Corresponds to the Title Object in the request, with the value filled in.

```
public interface ITitleContent : IDisposable
```

Inherited Members
IDisposable.Dispose()

Properties

Text

Gets or sets text value.

```
string? Text { get; }
```

Property Value

string

The text associated with the text element.

See Also

AssetTitle

Enum ImageContentType

Namespace: AppstockSDK.Api.Native.Data.Response

Assembly: Appstock.Core.dll

Type of image asset.

```
[Serializable]
public enum ImageContentType
```

Fields

`Custom = 500`

Reserved for Exchange specific usage numbered above 500.

`Icon = 1`

Icon image.

Optional.

max height: at least 50

aspect ratio: 1:1

`Main = 3`

Large image preview for the ad.

At least one of 2 size variants required:

Small variant	**max height**	at least 200
	max width	at least 200, 267, or 382
	aspect ratio	1:1, 4:3, or 1.91:1
Large variant	**max height**	at least 627
	max width	at least 627, 836, or 1198

aspect ratio	1:1, 4:3, or 1.91:1

See Also

imageType
AssetImage.ImageType

Namespace AppstockSDK.Common

Interfaces

ISdkImp

Builds platform-specific implementations.

ISdkInitializer

Proxy for a static method in the native SDK library for a specific platform.

Interface ISdkImp

Namespace: AppstockSDK.Common

Assembly: Appstock.Common.dll

Builds platform-specific implementations.

```
[RequireImplementors]
public interface ISdkImp
```

Properties

SdkInitializer

Gets SDK Initializer object.

```
ISdkInitializer SdkInitializer { get; }
```

Property Value

ISdkInitializer

Proxy for a static method in the native SDK library for a specific platform.

SdkProxy

Gets SDK proxy object.

```
ISdkProxy SdkProxy { get; }
```

Property Value

ISdkProxy

Managed object that exposes global settings of native SDK.

TargetingProxy

Gets SDK targeting proxy object.

```
ITargetingProxy TargetingProxy { get; }
```

Property Value

ITargetingProxy

Managed object that exposes targeting settings of native SDK.

Methods

CreateBanner(AdSize)

Creates a banner of specified size.

```
IBannerAd CreateBanner(AdSize adSize)
```

Parameters

`adSize` AdSize

Size of the banner view.

Returns

IBannerAd

Managed object that exposes controls over native ad view.

CreateInterstitial()

Creates an interstitial ad unit.

```
IInterstitialAd CreateInterstitial()
```

Returns

IInterstitialAd

Managed object that exposes controls over native interstitial ad unit.

CreateNativeAdLoader(AdUnitData)

Creates a loader for native ad.

```
INativeAdLoader CreateNativeAdLoader(AdUnitData adUnitData)
```

Parameters

`adUnitData` AdUnitData

Serializable template for native ad request.

Returns

INativeAdLoader

Managed object that exposes controls over native ad loader.

CreateRewarded()

Creates a rewarded ad unit.

```
IRewardedAd CreateRewarded()
```

Returns

IRewardedAd

Managed object that exposes controls over native rewarded ad unit.

Interface ISdkInitializer

Namespace: AppstockSDK.Common

Assembly: Appstock.Common.dll

Proxy for a static method in the native SDK library for a specific platform.

```
public interface ISdkInitializer
```

Methods

InitializeSdk(string)

Initialize the SDK.

```
void InitializeSdk(string partnerKey)
```

Parameters

`partnerKey` string

Determines the Appstock server URL. The Appstock account manager should provide you with this key.

Namespace AppstockSDK.Editor

Classes

ExtSlotPropertyDrawer

Custom PropertyDrawer for ExtSlot.

NativeConfigWarningsPropertyDrawer

Custom PropertyDrawer for ConfigWarnings.

SerializableOptionalPropertyDrawer

Custom PropertyDrawer for SerializableOptional<T>.

Class ExtSlotPropertyDrawer

Namespace: AppstockSDK.Editor

Assembly: Appstock.Core.Native.Data.Editor.dll

Custom PropertyDrawer for ExtSlot.

```
[CustomPropertyDrawer(typeof(ExtSlot))]
public class ExtSlotPropertyDrawer : PropertyDrawer
```

Inheritance

object ← GUIDrawer ← PropertyDrawer ← ExtSlotPropertyDrawer

Inherited Members

PropertyDrawer.CreatePropertyGUI(SerializedProperty) , PropertyDrawer.CanCacheInspectorGUI(SerializedProperty) , PropertyDrawer.attribute , PropertyDrawer.fieldInfo , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

GetPropertyHeight(SerializedProperty, GUIContent)

Specifies how tall the GUI for this field is in pixels.

```
public override float GetPropertyHeight(SerializedProperty property,
GUIContent label)
```

Parameters

`property` SerializedProperty

The SerializedProperty to make the custom GUI for.

`label` GUIContent

The label of this property.

Returns

float

The height in pixels.

OnGUI(Rect, SerializedProperty, GUIContent)

Uses IMGUI to draw GUI for the property.

```
public override void OnGUI(Rect position, SerializedProperty property,
GUIContent label)
```

Parameters

`position` Rect

Rectangle on the screen to use for the property GUI.

`property` SerializedProperty

The SerializedProperty to make the custom GUI for.

`label` GUIContent

The label of this property.

Class NativeConfigWarningsPropertyDrawer

Namespace: AppstockSDK.Editor

Assembly: Appstock.Core.Native.Data.Editor.dll

Custom PropertyDrawer for ConfigWarnings.

```
[CustomPropertyDrawer(typeof(ConfigWarnings))]
public class NativeConfigWarningsPropertyDrawer : PropertyDrawer
```

Inheritance

object ← GUIDrawer ← PropertyDrawer ← NativeConfigWarningsPropertyDrawer

Inherited Members

PropertyDrawer.CreatePropertyGUI(SerializedProperty) , PropertyDrawer.CanCacheInspectorGUI(SerializedProperty) , PropertyDrawer.attribute , PropertyDrawer.fieldInfo , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

GetPropertyHeight(SerializedProperty, GUIContent)

Specifies how tall the GUI for this field is in pixels.

```
public override float GetPropertyHeight(SerializedProperty property,
GUIContent label)
```

Parameters

`property` SerializedProperty

The SerializedProperty to make the custom GUI for.

`label` GUIContent

The label of this property.

Returns

float

The height in pixels.

OnGUI(Rect, SerializedProperty, GUIContent)

Uses IMGUI to draw GUI for the property.

```
public override void OnGUI(Rect position, SerializedProperty property,
GUIContent label)
```

Parameters

`position` Rect

Rectangle on the screen to use for the property GUI.

`property` SerializedProperty

The SerializedProperty to make the custom GUI for.

`label` GUIContent

The label of this property.

Class SerializableOptionalPropertyDrawer

Namespace: AppstockSDK.Editor

Assembly: Appstock.Core.Native.Data.Editor.dll

Custom PropertyDrawer for SerializableOptional<T>.

```
[CustomPropertyDrawer(typeof(SerializableOptional<>), true)]
public class SerializableOptionalPropertyDrawer : PropertyDrawer
```

Inheritance

object ← GUIDrawer ← PropertyDrawer ← SerializableOptionalPropertyDrawer

Inherited Members

PropertyDrawer.CreatePropertyGUI(SerializedProperty) ,
PropertyDrawer.GetPropertyHeight(SerializedProperty, GUIContent) ,
PropertyDrawer.CanCacheInspectorGUI(SerializedProperty) , PropertyDrawer.attribute ,
PropertyDrawer.fieldInfo , object.Equals(object) , object.Equals(object, object) ,
object.GetHashCode() , object.GetType() , object.MemberwiseClone() ,
object.ReferenceEquals(object, object) , object.ToString()

Methods

OnGUI(Rect, SerializedProperty, GUIContent)

Uses IMGUI to draw GUI for the property.

```
public override void OnGUI(Rect position, SerializedProperty property,
GUIContent label)
```

Parameters

`position` Rect

Rectangle on the screen to use for the property GUI.

`property` SerializedProperty

The SerializedProperty to make the custom GUI for.

`label` GUIContent

The label of this property.

Namespace AppstockSDK.Unity

Classes

SdkImp

ISdkImp stub for platforms other than IPhonePlayer, Android.

Class SdkImp

Namespace: AppstockSDK.Unity

Assembly: Appstock.Unity.dll

ISdkImp stub for platforms other than IPhonePlayer, Android.

```
public sealed class SdkImp : ISdkImp
```

Inheritance

object ← SdkImp

Implements

ISdkImp

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.ReferenceEquals(object, object) , object.ToString()

Properties

SdkInitializer

Gets SDK Initializer object.

```
public ISdkInitializer SdkInitializer { get; }
```

Property Value

ISdkInitializer

Proxy for a static method in the native SDK library for a specific platform.

SdkProxy

Gets SDK proxy object.

```
public ISdkProxy SdkProxy { get; }
```

Property Value

ISdkProxy

Managed object that exposes global settings of native SDK.

TargetingProxy

Gets SDK targeting proxy object.

```
public ITargetingProxy TargetingProxy { get; }
```

Property Value

ITargetingProxy

Managed object that exposes targeting settings of native SDK.

Methods

CreateBanner(AdSize)

Creates a banner of specified size.

```
public IBannerAd CreateBanner(AdSize adSize)
```

Parameters

`adSize` AdSize

Size of the banner view.

Returns

IBannerAd

Managed object that exposes controls over native ad view.

CreateInterstitial()

Creates an interstitial ad unit.

```
public IInterstitialAd CreateInterstitial()
```

Returns

IInterstitialAd

Managed object that exposes controls over native interstitial ad unit.

CreateNativeAdLoader(AdUnitData)

Creates a loader for native ad.

```
public INativeAdLoader CreateNativeAdLoader(AdUnitData adUnitData)
```

Parameters

`adUnitData` AdUnitData

Serializable template for native ad request.

Returns

INativeAdLoader

Managed object that exposes controls over native ad loader.

CreateRewarded()

Creates a rewarded ad unit.

```
public IRewardedAd CreateRewarded()
```

Returns

IRewardedAd

Managed object that exposes controls over native rewarded ad unit.

Namespace AppstockSDK.iOS

Classes

SdkImp

ISdkImp implementation for IPhonePlayer.

Class SdkImp

Namespace: AppstockSDK.iOS

Assembly: Appstock.iOS.dll

ISdkImp implementation for IPhonePlayer.

```
public sealed class SdkImp : ISdkImp
```

Inheritance

object ← SdkImp

Implements

ISdkImp

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.ReferenceEquals(object, object) , object.ToString()

Properties

SdkInitializer

Gets SDK Initializer object.

```
public ISdkInitializer SdkInitializer { get; }
```

Property Value

ISdkInitializer

Proxy for a static method in the native SDK library for a specific platform.

SdkProxy

Gets SDK proxy object.

```
public ISdkProxy SdkProxy { get; }
```

Property Value

ISdkProxy

Managed object that exposes global settings of native SDK.

TargetingProxy

Gets SDK targeting proxy object.

```
public ITargetingProxy TargetingProxy { get; }
```

Property Value

ITargetingProxy

Managed object that exposes targeting settings of native SDK.

Methods

CreateBanner(AdSize)

Creates a banner of specified size.

```
public IBannerAd CreateBanner(AdSize adSize)
```

Parameters

`adSize` AdSize

Size of the banner view.

Returns

IBannerAd

Managed object that exposes controls over native ad view.

CreateInterstitial()

Creates an interstitial ad unit.

```
public IInterstitialAd CreateInterstitial()
```

Returns

IInterstitialAd

Managed object that exposes controls over native interstitial ad unit.

CreateNativeAdLoader(AdUnitData)

Creates a loader for native ad.

```
public INativeAdLoader CreateNativeAdLoader(AdUnitData adUnitData)
```

Parameters

`adUnitData` AdUnitData

Serializable template for native ad request.

Returns

INativeAdLoader

Managed object that exposes controls over native ad loader.

CreateRewarded()

Creates a rewarded ad unit.

```
public IRewardedAd CreateRewarded()
```

Returns

IRewardedAd

Managed object that exposes controls over native rewarded ad unit.