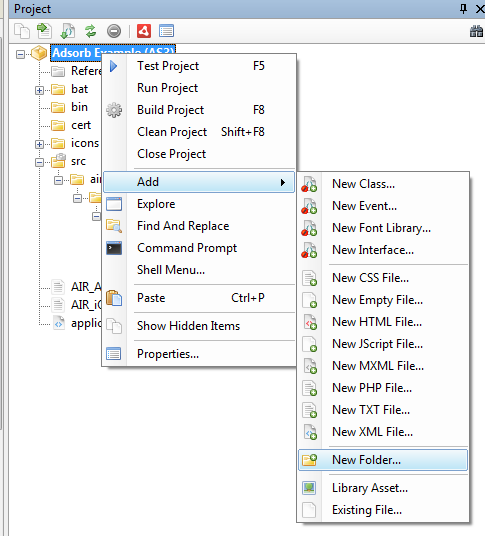
Adsorb Integration Guide

The examples given in this document are mostly using AS3, but the principles remain the same for all programming languages.  To find the functions / event listeners specific to your language, check the example file in the connector.

Basic integration with FlashDevelop

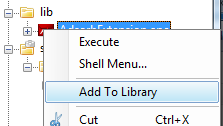
Starting with the very beginning, how to set up a FlashDevelop project to use Adsorb.

1. Start a new Project, select “AIR Mobile AS3 App”  (or open your existing game)
2. Create a “lib” folder in the root of your project.



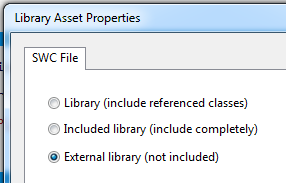
3.  Copy the \AdsorbConnector\air\_extension\AdsorbExtension.ane into the “lib” folder

4.  Right click the AdsorbExtension.ane and click “Add to Library”.



5.  Right click the AdsorbExtension.ane file, and click “Options”.

6.  Set it to “External”:



7.  Edit the “bat/Packager.bat” file.  Add “-extdir lib/” at the end of the line that packages the app, something like:

call adt -package -target %TYPE%%TARGET% %OPTIONS% %SIGNING\_OPTIONS% "%OUTPUT%" "%APP\_XML%" %FILE\_OR\_DIR% -extdir lib/

8.  Open the “application.xml” file.  Before the closing </application> tag, add a line including this extension:

<extensions>

       <extensionID>com.fgl.airconnector</extensionID>

</extensions>

9.  Initialize a static reference to the Adsorb class that you can use throughout your game.  Add a test call to show an ad.

    /\*\*

    \* ...

    \* @author FGL

    \*/

    public class Main extends Sprite

    {

        static private var adsorb:Adsorb;

        static private var eventDispatcher:EventDispatcher;

        public function Main()

        {

            // Adsorb Entry point

            // Initialize the static var

            adsorb = Adsorb.getInstance();

            // Show an Ad

            adsorb.showInterstitialAd();

        }

    }

10.  Create a test certificate by running the bat\CreateCertificate.bat file.

11.  Connect a Mobile device to your PC, make sure it has the “developer mode” checked.

12.  Run the game on a mobile device.  You should see a test ad displayed where you called “showInterstitialAd”.  A real ad will appear once FGL runs your final APK through the adsorb process.

Advanced Integration

**Showing Rewarded Videos**

This will show you how to set up the most common type of rewarded ad, the HELPER ad.  This reward should be placed in the game at a point where you can “help” the player.  Some common placements:

* give the player another life
* give the player more hints when they run out
* give the player more time to complete the level

It must immediately show when you want to offer the player the reward, you are not allowed to have a button that initiates the ad.

1. The first step is to listen for the events that pertain to HELPER rewards:

e.addEventListener(AdsorbEvent.ADSORB\_HELPER\_REWARDED\_AD\_READY, onHelperRewardedAdReady);

e.addEventListener(AdsorbEvent.ADSORB\_HELPER\_REWARDED\_AD\_UNAVAILABLE, onHelperRewardedAdUnavailable);

e.addEventListener(AdsorbEvent.ADSORB\_HELPER\_REWARDED\_AD\_DISMISSED, onHelperRewardedAdDismissed);

e.addEventListener(AdsorbEvent.ADSORB\_HELPER\_REWARDED\_AD\_FAILED, onHelperRewardedAdFailed);

e.addEventListener(AdsorbEvent.ADSORB\_HELPER\_REWARD\_GRANTED, onHelperRewardGranted);

1. Next, you need to tell ADsorb to load a helper Ad.  This can be done on startup or a short time before you know that the player will need one.  If you go the startup route, make sure you ping the server every once in a while if you get a “Unavailable” event.

adsorb.loadHelperAd ("Need more moves?", "Watch a short message and get an extra", "7 moves", "Tap to get moving", false);

    Replace the texts as you see fit for your game.

1. If a “onHelperRewardedAdReady” function fires, that means that your game is ready to show it when needed.