

GoViral Social Networking Extension for iOS and Android

Copyright © 2012-2013 Milkman Games, LLC. All rights reserved.

<http://www.milkmangames.com>

For support, contact support@milkmangames.com

Before you begin:

To View full AS3 documentation, see 'docs/as3docs/index.html'.

This extension requires the AIR 3.5 SDK or higher, which you can get at

<http://www.adobe.com/devnet/air/air-sdk-download.html> .

Review 'example/GoViralExample.as' for a sample application.

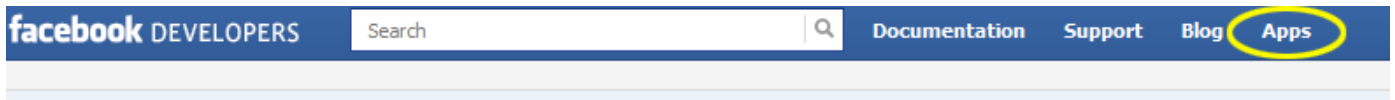
Note that GoViralExample.as is a Document Class. If you're a Flash Professional user and don't know how to use a document class, see the appendix "Using the Example Class in Flash CS6" at the end of this guide.

If you are migrating from an earlier version of the extension, please review this guide and see [releasenotes.txt](#) for details of changes.

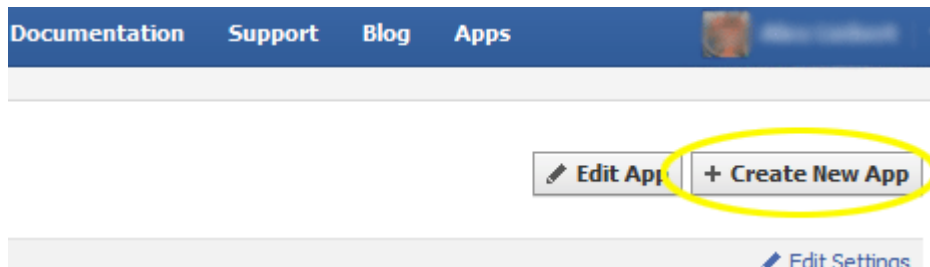
1. Facebook Developer Site Setup

This section covers setting up a Facebook mobile app in the Developer section of the Facebook website. Before you can use the Facebook features of the extension, you'll need to create the App on the Facebook Developer site and register for a Facebook App ID.

1. Go to <http://developers.facebook.com>. If you haven't already done so, register a developer with Facebook to create your first app.
2. After logging in, select 'Apps' from the top menu.





3. Select the '+ Create New App' button on the next screen. The 'Create New App' pop up dialog will appear.



4. Enter a display name for your application, and press 'Continue'.

5. A Facebook App ID and App Secret have now been created for your application. Take note of the **App ID**- you'll need it later to configure your application in Adobe AIR.
6. Complete the 'Basic Information' section, entering the appropriate email address, and app category.

7. **(iOS App Setup:)** Select '*Native iOS App*' to expand the iOS Section. For '*iOS Bundle ID*', enter your iOS Application Bundle ID. This ID will need to match both the ID you used in iTunes Connect for the App, and the ID you use in your AIR Application XML descriptor.
8. **(iOS App Setup:)** For '*iPhone App Store ID*', you'll need to enter the numeric iTunes App Store ID for your app. If your app is not yet live on the App Store, you can enter a different app's numeric ID here for now. Later, when your App has been approved by Apple, you can replace it with your own numeric ID. If you need a numeric ID to use temporarily during development, you may use 331975235, which is the ID for Adobe Photoshop Express iOS.
9. **(iOS App Setup:)** For '*Configured for iOS SSO*' and '*iOS Native Deep Linking*', select '*Enabled*'.
10. **(Android App Setup:)** Select '*Native Android App*' to expand the Android Section. Facebook for Android uses a different security method for authenticating the app as your own than Facebook does- your **Android Key Hash**. You will need to generate a Key Hash for the testing/debug version of your app, AND the release version. For now, enter any string in this box (such as "invalidkeyhash"). Later, you can use the native extension to retrieve your actual Hash. *After you complete Facebook.com setup, refer to the section, "Getting Your Android Key Hash" below.*

 **Native Android App** 

Configured for Android SSO: ☒ Enabled ☐ Disabled

Android Native Deep Linking: ☐ Enabled ☒ Disabled

Android Key Hash: [?]

Android Package Name:

Android Class Name:

11. **(Android App Setup:)** For '*Android Package Name*', enter your full Android app ID (such as 'air.com.yourcompany.yourappname'; for '*Android Class Name*', enter the package name plus '.AppEntry'- such as, 'air.com.yourcompany.yourappname.AppEntry'. Enable 'Configured for Android SSO'. **You must include the 'air.' prefix!**
12. Select '*Mobile Web*' and fill in '*Mobile Site URL*' with a site url for your app, or if it's mobile only, an Google Play or iTunes web URL. If your app is not live yet, you can use an existing web url, such as Angry Birds, which is <http://itunes.apple.com/us/app/angry-birds/id343200656?> . Leave the '?' at the end of the url- it is required.
13. Press '*Save Changes*' to complete the Basic Settings setup.

2. Install the AIR 3.5 (or higher) SDK in your IDE

The GoViral extension requires the AIR 3.5 SDK or higher. You can download the latest AIR SDK from <http://www.adobe.com/devnet/air/air-sdk-download.html>. If you haven't already installed the AIR 3.5 (or higher) SDK for your Flash CS6 or Flash Builder IDE, follow the instructions below:

Enabling the AIR 3.5 SDK in Flash Professional CS6:

1. Unzip the AIR 3.5 SDK package to a location on your hard drive.
2. Launch Flash Professional CS6.
3. Select Help > Manage AIR SDK...
4. Press the Plus (+) Button and navigate to the location of the unzipped AIR SDK
5. Press OK
6. Select File > Publish Settings
7. Select the AIR 3.5 SDK for iOS from the 'Target' Dropdown menu

Enabling the AIR 3.5 SDK in Flash Builder 4.6- Windows:

1. Unzip the AIR 3.5 SDK package to a location on your hard drive.
2. Close Flash Builder.
3. Locate the Flash Builder SDK directory. On the PC, usually c:\Program Files\Adobe\Adobe Flash Builder 4.6\sdk .
4. Make a copy of the current Flex SDK directory, and give it a descriptive name. For instance, copy the "4.6.0" SDK folder inside /sdks and name the copy "4.6.0_AIR35".
5. Copy and paste the contents of the AIR 3.5 SDK on top of the 4.6.0_AIR35 directory. Accept all changes.
6. Edit the flex-sdk-description.xml file inside the new directory, and change the value of the <name> tag to 'Flex 4.6.0 (AIR 3.5)'.
7. Open Flash Builder and choose Project > Properties > Flex Compiler > Configure Flex SDKs.
8. Press 'Add' and navigate to the new folder location.

Enabling the AIR 3.5 SDK in Flash Builder 4.6- Mac:

1. Copy the contents AIR 3.5 SDK package to a location on your hard drive.
2. Close Flash Builder.
3. Locate the Flash Builder SDK directory. On the Mac, it is usually /Applications/Adobe Flash Builder 4.6/sdks/. On the PC, c:\Program Files\Adobe\Adobe Flash Builder 4.6\sdk .
4. Create a new folder inside the SDK folder, called AIR35SDK and copy the contents of the SDK package into it.
5. Open the Terminal, and merge the AIR 3.5 SDK files into your current SDK directory:

```
sudo cp -Rp /Applications/Adobe\ Flash\ Builder\ 4.6/sdks/AIR35SDK/ /Applications/Adobe\ Flash\ Builder\ 4.6/sdks/4.6.0/
```
6. Edit the flex-sdk-description.xml file inside the new directory, and change the value of the <name> tag to 'Flex 4.6.0 (AIR 3.5)'.
7. Open Flash Builder and choose Project > Properties > Flex Compiler > Configure Flex SDKs.

8. Press 'Add' and navigate to the new folder location. Press 'Add' and navigate to the new folder location.

3. Include the GoViral API Library

Add the com.milkmangames.nativeextensions.GoViral.ane library to your project. These are in the /extension folder of the extension package zip file.

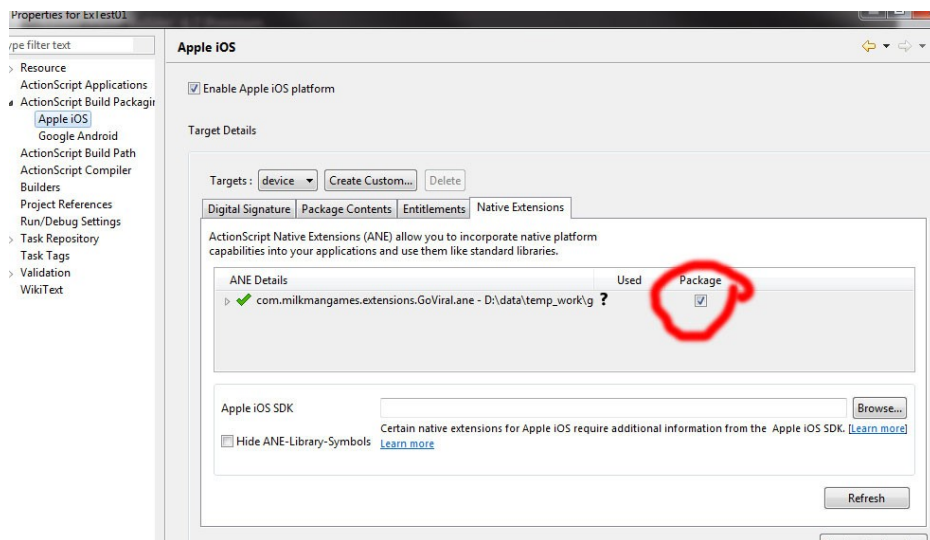
IMPORTANT: *During testing/development of an Android Facebook application, use the version of the extension in the '/extension/androiddebugane' folder that came with the extension zip package. Before public release, you should switch to the regular version.*

In Flash Professional CS6:

1. Create a new project of the type 'AIR for iOS' (or AIR for Android)
2. Select File > Publish Settings...
3. Select the wrench icon next to 'Script' for 'Actionscript Settings'
4. Select the Library Path tab
5. Press the 'Browse for Native Extension (ANE) File' button and select the com.milkmangames.extensions.GoViral.ane file. Press OK.
6. Select the wrench icon next to 'Target' for Player Settings
7. Select the 'Permissions' tab, and enable INTERNET and WRITE_EXTERNAL_STORAGE
8. Check the 'Manually manage permissions and manifest additions for this app' box.
9. Make sure you're using the AIR 3.5 SDK per the instructions in Section 2.
10. Select File>Publish Settings...
11. For 'Target' Select 'AIR 3.5 for iOS' (Or Android)

In Flash Builder 4.6+:

1. Go to *Project Properties*
2. Select *Native Extensions*
3. Choose *Add ANE...* and navigate to the com.milkmangames.extensions.GoViral.ane file
4. Make sure the 'package' box is checked for your target, under Actionscript Build Packaging:



5. Make sure you're using the AIR 3.5 SDK per the instructions in Section 2.

4. Update Your Application Descriptor

For extension to work, changes are required to the application XML file for your app. Modify the xml file created by your IDE with the following changes (if you're a Flash Professional user, make sure you've followed the steps above for 'Include the Library in Flash Professional', otherwise Flash might undo your changes as you make them):

You'll need to be using the AIR 3.5 SDK or higher, and include the extension in your Application Descriptor XML. For an example, see '**example/app.xml**'. (Remember, the <id> section must exactly match your ids from developer.facebook.com, so you can't copy that part of the example xml verbatim.)

1. Set your AIR SDK to 3.5 in the app descriptor file:

```
<application xmlns="http://ns.adobe.com/air/application/3.5">
```

2. Include a link to the extension in the descriptor:

```
<extensions>
<extensionID>com.milkmangames.extensions.GoViral</extensionID>
</extensions>
```

3. **For iOS:** Add your Facebook App ID to the XML Application Descriptor. Refer to Section 1 for information on locating your Facebook App ID in the Facebook Developer website. Add the following section, <iphone>, to the descriptor file, and **replace the 000000 in fb000000, and in the <string> below <key>FacebookAppID</key> with your Facebook App ID. Change only the parts show in red- the fb prefix is important.**

```
<iPhone>
<InfoAdditions>
    <![CDATA[
    <key>FacebookAppID</key>
    <string>000000</string>
    <key>CFBundleURLTypes</key>
    <array>
        <dict>
            <key>CFBundleURLSchemes</key>
            <array>
                <string>fb000000</string>
```

```

        </array>
    </dict>
</array>
]]>
</InfoAdditions>
</iPhone>

```

4. **For Android:** Set the Android ManifestAdditions required by the extension. You'll need to add the WRITE_EXTERNAL_STORAGE and INTERNET permissions, and add the LoginActivity. For compatibility with the newest versions of Android, you'll need to set targetSdkVersion and minSdkVersion. Add the following section, <android>, to the descriptor file:

```

<android>
<manifestAdditions><![CDATA[
<manifest android:installLocation="auto">
    <uses-sdk android:targetSdkVersion="12"/>
    <uses-sdk android:minSdkVersion="8"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.INTERNET"/>

    <application>

        <activity android:name="com.facebook.LoginActivity"
android:theme="@android:style/Theme.Translucent.NoTitleBar" android:label="Login"/>

    </application>
</manifest>
]]></manifestAdditions>

</android>

```

5. API Quick Start: Initialization

The GoViral extension can be up and running in a few simple calls. See '[example/GoViralExample.as](#)' for a full example, including Facebook methods, Twitter, and Email.

1. Import the API Classes:

```

import com.milkmangames.nativeextensions.*;
import com.milkmangames.nativeextensions.events.*;

```

2. First, initialize the API and create an instance of the GoViral object, by calling GoViral.create().

IMPORTANT: This initialization must occur in the entry point of your application. If you are using Flex/MXML, this means you must call create() in the initialize() event of the main class (not a View constructor or createComplete callback.) If you are using pure ActionScript, you must call create() inside the Constructor function of the Document Class. If you are using timeline code in Flash, call this once at the beginning of Frame 1.

Make sure the current platform is supported (as in, it's not a PC or Mac, etc.) by checking the GoViral.isSupported() method:

```

if(GoViral.isSupported())
{
    GoViral.create();
}
else {
    trace("GoViral only works on mobile!");
    return;
}

```

```
}
```

6. API Quick Start: Facebook

GoViral offers complete support for Facebook mobile APIs: Single Sign-On/Authentication, Wall Post Dialogs, Friend Invitations, and the entire Facebook Graph API.

1. Start by initializing the Facebook SDK, using your Facebook App ID. (The Facebook App ID can be found in your Facebook Developer site panel, as described in Section 1, Step 5.)

```
// replace 000000 with your facebook app id!
GoViral.goViral.initFacebook("000000", "");
```

2. Add any event listeners you may want for Facebook related events. The following example code shows all the available listeners.

```
// Listen for events.
GoViral.goViral.addListener(GVFacebookEvent.FB_DIALOG_CANCELED, onFacebookEvent);
GoViral.goViral.addListener(GVFacebookEvent.FB_DIALOG_FAILED, onFacebookEvent);
GoViral.goViral.addListener(GVFacebookEvent.FB_DIALOG_FINISHED, onFacebookEvent);
GoViral.goViral.addListener(GVFacebookEvent.FB_LOGGED_IN, onFacebookEvent);
GoViral.goViral.addListener(GVFacebookEvent.FB_LOGGED_OUT, onFacebookEvent);
GoViral.goViral.addListener(GVFacebookEvent.FB_LOGIN_CANCELED, onFacebookEvent);
GoViral.goViral.addListener(GVFacebookEvent.FB_LOGIN_FAILED, onFacebookEvent);
GoViral.goViral.addListener(GVFacebookEvent.FB_REQUEST_FAILED, onFacebookEvent);
GoViral.goViral.addListener(GVFacebookEvent.FB_REQUEST_RESPONSE, onFacebookEvent);
```

3. Log the user in to Facebook. You'd typically want to do this in response to pressing a 'Connect with Facebook' button or similar UI.

Note that as a feature of SSO (single sign-on), if your app is restarting after a previous session where the user has logged in, you may be automatically logged in after calling `initFacebook()` - in this case you will received the `FB_LOGGED_IN` event shortly after the app begins.

You can use the `GoViral.goViral.isFacebookAuthenticated()` method to determine if the user is logged in already. If they aren't, call `GoViral.goViral.authenticateWithFacebook()` to start a login prompt:

```
// if the user is not already logged in...
if(!GoViral.goViral.isFacebookAuthenticated())
{
    // show a connect with Facebook prompt.
    // this method takes a comma separated list of facebook READ permissions as a first
    // parameter. You should have at least one read permission.

    // you can refer to the facebook documentation at
    // http://developers.facebook.com/docs/authentication/permissions/ to determine which
    // permissions your app requires.

    GoViral.goViral.authenticateWithFacebook("basic_info");
}
```

Next, the user will be presented with the Facebook Connect prompt, and asked to grant your app permission to use Facebook on their behalf.

If the device is running iOS 6 or higher, iOS will present a dialog to accept any permissions without leaving the app.

If the official Facebook app is installed, the device is running iOS 5 or earlier, and the device supports

multitasking, iOS will task switch to the Facebook app to authenticate the user, then return to your app on completion.

If the official Facebook app is not installed, the device is running iOS 5 or earlier, and the device supports multi-tasking, iOS will task switch to Safari and present an authentication dialog there; on completion, your app will be returned to the foreground.

If the device does not support multitasking, an authentication dialog will be presented inside your own app.

On Android, authentication will use an activity from the official Facebook app if it is installed, or a web view if not. For authentication to complete on Android, your Android Key Hash must be properly entered on developer.facebook.com. *After building your APK file, refer to Section 10, "Getting Your Android Key Hash".*

A `GVFacebookEvent` indicating the result will be dispatched when the login succeeds, fails, or is cancelled by the user- with the type of `GVFacebookEvent.FB_LOGGED_IN`, `GVFacebookEvent.FB_LOGIN_FAILED`, `GVFacebookEvent.FB_LOGIN_CANCELED`.

Posting to the Facebook Wall

A common task your app may wish to perform is posting an update to the user's Facebook wall. The following example will display a Facebook dialog prompting the user to make a wall post, with a title, caption, message, description, link, and image URL:

```
// show a dialog to post the the user's wall
GoViral.goViral.showFacebookFeedDialog(
    "Posting from AIR",
    "This is a caption",
    "This is a message!",
    "This is a description",
    "http://www.milkmangames.com",
    "http://www.milkmangames.com/blog/wp-content/uploads/2012/01/v202.jpg"
);
```

When the dialog is dismissed, a `GVFacebookEvent` will be dispatched with the result, of the type `GVFacebookEvent.FB_DIALOG_FINISHED` if the Dialog finished, `GVFacebookEvent.FB_DIALOG_CANCELED` if the dialog was canceled, or `GVFacebookEvent.FB_DIALOG_FAILED` if the dialog failed.

Inviting Facebook Friends

The GoViral extension can prompt the user to invite their friends to your app, using `showFacebookRequestDialog()`. The following example displays the invite friends dialog, with a message and title. If a friend accepts the request on the website, they'll be directed to the App Store to get your app. If they accept in on their phone, the app will be loaded if its installed, or they'll be sent to the App Store to get it.

```
// show a dialog so the user can invite friends to app.
GoViral.goViral.showFacebookRequestDialog("Try out this app!", "My App Title");
```

When the dialog is dismissed, a `GVFacebookEvent` will be dispatched with the result, of the type `GVFacebookEvent.FB_DIALOG_FINISHED` if the Dialog finished, `GVFacebookEvent.FB_DIALOG_CANCELED` if the dialog was canceled, or `GVFacebookEvent.FB_DIALOG_FAILED` if the dialog failed.

Getting the User's Profile

You can retrieve the user's profile with the `requestMyFacebookProfile()` method. This will cause a

GVFacebookEvent.FB_REQUEST_RESPONSE event to be dispatched; its property friends is a Vector.<GVFacebookFriend>, that will contain the user's profile information.

The following code shows how to request the profile, and an implementation of an event listener function for handling the response. Refer to *example/GoViralExample.as* for a complete sample application that handles all Facebook events.

```
// listen for a response
GoViral.goViral.addEventListener(GVFacebookEvent.FB_REQUEST_RESPONSE,onFacebookResponse);
GoViral.goViral.addEventListener(GVFacebookEvent.FB_REQUEST_FAILED,onRequestFailed);

// request the user's profile.
GoViral.goViral.requestMyFacebookProfile();

function onFacebookResponse(e:GVFacebookEvent):void
{
    // the graphPath property is 'me' for a profile request.
    if(e.graphPath=="me")
    {
        var myProfile:GVFacebookFriend=e.friends[0];
        trace("my name is:"+myProfile.name);
        trace("my gender is:"+myProfile.gender);
        trace("My bio is:"+myProfile.bio);
    }
}

function onRequestFailed(e:GVFacebookEvent):void
{
    trace("something went wrong with the request:"+e.errorMessage);
}
```

Getting the User's Friends

You can retrieve the user's list of friends with the requestFacebookFriends() method. This will cause a GVFacebookEvent.FB_REQUEST_RESPONSE event to be dispatched; its property friends is a Vector.<GVFacebookFriend>, that will contain the user's friends' information.

The following code shows how to request the user's friends, and an implementation of an event listener function for handling the response. Refer to *example/GoViralExample.as* for a complete sample application that handles all Facebook events.

```
// listen for a response
GoViral.goViral.addEventListener(GVFacebookEvent.FB_REQUEST_RESPONSE,onFacebookResponse);
GoViral.goViral.addEventListener(GVFacebookEvent.FB_REQUEST_FAILED,onRequestFailed);

// request the user's friends list.
GoViral.goViral.requestFacebookFriends();

function onFacebookResponse(e:GVFacebookEvent):void
{
    // the graphPath property is 'me/friends' for a friends request.
    if(e.graphPath=="me/friends")
    {
        for each(var friend:GVFacebookFriend in e.friends)
        {
            trace("i have a friend named:"+friend.name);
        }
    }
}

function onRequestFailed(e:GVFacebookEvent):void
```

```
{
    trace("something went wrong with the request:"+e.errorMessage);
}
```

Posting a Photo to Facebook

You can post a `BitmapData` image to the user's Facebook photos with the `facebookPostPhoto()` method.

```
// posts a bitmapData called 'myBitmapData' to Facebook
GoViral.goViral.facebookPostPhoto("posted from ios sdk",myBitmapData);
```

When the request is finished, a `GVFacebookEvent.FB_REQUEST_RESPONSE` will be dispatched, with a `graphPath` property of `me/photos`. If an error occurs, `GVFacebookEvent.FB_REQUEST_RESPONSE` will be dispatched instead.

Handling Mobile App Ad Installs

If you are using the mobile ads feature of the Facebook Platform, you can notify Facebook of an install after your app has started with the following call:

```
// notify facebook of the install, for ad tracking purposes.
GoViral.goViral.publishInstall();
```

Calling the Facebook Graph API

GoViral gives you access to the entirety of the Facebook Graph API- so you can use any method Facebook supports- scores, achievements, videos, checkins, events, groups and more – directly from Actionscript 3.

The following code shows the method signature for `GoViral.goViral.facebookGraphRequest()`. For more information on the Graph API, refer to <https://developers.facebook.com/docs/reference/api/>.

facebookGraphRequest() METHOD

```
public function facebookGraphRequest(graphPath:String, httpMethod:String = GET,
params:Object = null, publishPermissions:String = null):void
```

Post a raw request to the Facebook Graph API. Results are returned as JSON. This gives you native access to the entire Facebook platform. Refer to the Facebook developer documentation for details on the many accessible API methods. Facebook now requires that any 'PUBLISH' permissions are requested separately from initial sign on. If the graph request requires any PUBLISH permissions, include them in the `publishPermissions` parameter. If the user has not yet accepted the permissions, they will be prompted to do so before the request is processed. After the request, a `GVFacebookEvent` will be dispatched with the result- `GVFacebookEvent.FB_REQUEST_RESPONSE` will be dispatched if the request succeeds, and the string value of its property `'jsonData'` will contain the raw JSON response. The property `'data'` will contain an Object of the parsed JSON for convenience. If the request fails, a `GVFacebookEvent.FB_REQUEST_FAILED` event will be dispatched.

Parameters:

graphPath:String-The graph path command you wish to execute, such as "me/friends"

to get user's friends. See the Facebook developer documentation for more details.

httpMethod:String (default =GET)- Optional. Default value is GVHttpMethod.GET. The httpMethod to use for the request, either GVHttpMethod.GET, GVHttpMethod.POST, or GVHttpMethod.DELETE. Refer to the Facebook developer documentation for details on the appropriate method.

params:Object(default =null)-Optional. An Object with string keys and values, for each additional parameter you want to send with the Graph request. See the Facebook developer documentation for more details.

publishPermissions:String- optional. A comma-separated list of any PUBLISH permissions required for the request.

7. API Quick Start: Twitter

Post a Status Update to Twitter

Posting a status update to Twitter is simple – just call `showTweetSheet()` with the message you'd like to post. (Twitter may not be available on iOS devices with a version below 5, or Android devices without Twitter. You can determine this before hand with the function `isTweetSheetAvailable()` :

```
if(GoViral.goViral.isTweetSheetAvailable())
{
    // show a twitter status post dialog
    GoViral.goViral.showTweetSheet("This is a native twitter post!");
}
```

Upon completion, a `GVTwitterEvent.TW_DIALOG_FINISHED`, `GVTwitterEvent.TW_DIALOG_CANCELED`, or `GVTwitterEvent.TW_DIALOG_FAILED` event will be dispatched.

Android User Note: Due to the limitations of interprocess communication on Android, the native Twitter app on the Android device may dispatch a `DIALOG_FINISHED` event, even if the dialog was canceled by the user.

Post an Image to Twitter

You may post a `bitmapData` image to Twitter using the `showTweetSheetWithImage()` method.

```
if(GoViral.goViral.isTweetSheetAvailable())
{
    // show a twitter status post dialog with bitmapData image
    GoViral.goViral.showTweetSheetWithImage("This is a native twitter post!",myBitmapData);
}
```

Upon completion, a `GVTwitterEvent.TW_DIALOG_FINISHED`, `GVTwitterEvent.TW_DIALOG_CANCELED`, or `GVTwitterEvent.TW_DIALOG_FAILED` event will be dispatched.

Android User Note: Due to the limitations of interprocess communication on Android, the native Twitter app on the Android device may dispatch a `DIALOG_FINISHED` event, even if the dialog was canceled by the user.

On Android, the extension will locate an app on that phone that responds to requests to send Tweets. Some Twitter Android apps from some Android manufacturers do not accept messages that have both custom status messages and images; in these cases, the image will be attached but a default status message may be displayed for the user to edit. If you are not seeing an image and message

attachment while using the official Twitter app from Google Play, update to the latest version, as Twitter has recently fixed a bug affecting this behavior.

8. API Quick Start: Email

Email is a great simple way to invite your users to share information from your app with their friends. GoViral supports plain emails, html emails, and emails with image attachments, which you may pass in as `BitmapData`.

Send an Email

The `showEmailComposer()` method will display a native iOS or Android Email Composer window to the user, prepopulated with the information of your choice. It takes a Subject, comma-separated list of 'to' addresses, and email body as its parameters. The last parameter is a Boolean value- set it to false if your email body is formatted as plain text; set it as true if your email body is formatted as html. (If you want to avoid making the request on devices where email is not set up, you may check `isEmailAvailable()`):

```
if(GoViral.goViral.isEmailAvailable())
{
    // show an email to who@where.com and john@doe.com, with subject 'this is a subject!',
    // and a plain text body of 'This is the body of the message.'
    GoViral.goViral.showEmailComposer(
        "This is a subject!",
        "who@where.com,john@doe.com",
        "This is the body of the message.",
        false
    );
}
```

On dismissal, a `GVMailEvent` will be dispatched indicating the result, with a type of `GVMailEvent.MAIL_SAVED` if the user saved but did not send the message, `GVMailEvent.MAIL_SENT` if the user sent the message, `GVMailEvent.MAIL_CANCELED` if the user canceled the dialog, or `GVMailEvent.MAIL_FAILED` if an error occurred sending.

Android User Note: Due to the limitations of interprocess communication on Android, the native email app on the Android device may dispatch a `MAIL_SENT` event, even if the dialog was canceled by the user.

Send an Email with Image Attachment

The `showEmailComposerWithBitmap()` method will display a native iOS or Android Email Composer window to the user, prepopulated with the information of your choice. It uses the same parameters as `showEmailComposer()`, with one additional parameter- the `BitmapData` image you'd like to attach to the mail. (If you want to avoid making the request on devices where email is not set up, you may check `isEmailAvailable()`):

```
if(GoViral.goViral.isEmailAvailable())
{
    // show an email to who@where.com and john@doe.com, with subject 'this is a subject!', a
    // and a plain text body of 'This has a pic attached.' Attaches a bitmapData image called
    // myBitmapData.
    GoViral.goViral.showEmailComposerWithBitmap(
        "This is a subject!",
        "who@where.com,john@doe.com",
        "This has a pic attached.",
        false,
        myBitmapData
    );
}
```

```
);
}
```

On dismissal, a `GVMailEvent` will be dispatched indicating the result, with a type of `GVMailEvent.MAIL_SAVED` if the user saved but did not send the message, `GVMailEvent.MAIL_SENT` if the user sent the message, `GVMailEvent.MAIL_CANCELED` if the user canceled the dialog, or `GVMailEvent.MAIL_FAILED` if an error occurred sending.

Android User Note: Due to the limitations of interprocess communication on Android, the native email app on the Android device may dispatch a `MAIL_SENT` event, even if the dialog was canceled by the user.

9. API Quick Start: Generic Share a Message or Image

All supported versions of Android can be used to post a message and/or image with no particular target, and allow the user to select an app from a list of installed apps to handle the post. Similar functionality is available in iOS 6 and higher.

Share a message

The `shareGenericMessage()` method allows you to send a message, and have the user select an app from their installed apps to handle the post. This feature is supported on all Android devices, and iOS devices running iOS 6 or higher. To ensure the feature is available on the current device, you should check `isGenericShareAvailable()` before calling `shareGenericMessage()` :

```
// show a dialog for the user to choose how to share a social message
if(GoViral.goViral.isGenericShareAvailable())
{
    GoViral.goViral.shareGenericMessage(
        "This is a subject!", // not all targets will use the 'subject'
        "I'm sharing this!", // this is the message to share
        false,               // set to 'true' if the message is html-formatted
        800,                 // on iOS 6 ipads, show the popup at 800 pts x
        800                  // on iOS 6 ipads, show the popup at 800 pts y
    );
}
```

After the action is completed, a `GVShareEvent.GENERIC_MESSAGE_SHARED` event will be dispatched.

Share a message and image

The `shareGenericMessageWithImage()` method behaves exactly like , but with an extra `BitmapData` paramter for sharing images :

```
// allows the user to choose
// plain text body of 'This is the body of the message.'

// show a dialog for the user to choose how to share a social message
if(GoViral.goViral.isGenericShareAvailable())
{
    GoViral.goViral.shareGenericMessageWithImage(
        "This is a subject!", // not all targets will use the 'subject'
        "I'm sharing this!", // this is the message to share
        false,               // set to 'true' if the message is html-formatted
        myBitmapData,        // the bitmapData image to share
        800,                 // on iOS 6 ipads, show the popup at 800 pts x
        800                  // on iOS 6 ipads, show the popup at 800 pts y
    );
}
```

After the action is completed, a `GVShareEvent.GENERIC_MESSAGE_SHARED` event will be dispatched.

Fast Social Sharing with Sina Weibo, Twitter, or Facebook (iOS 6 only)

iOS 6 introduces a new feature, the Social Composer View- which can be used for quickly sharing messages and images to built-in social services as Sina Weibo, Twitter, and Facebook. This feature is accessible to the extension via the `displaySocialComposerView()` method. You may specify a social service type from one of the constants `GVSocialServiceType.FACEBOOK`, `GVSocialServiceType.TWITTER`, or `GVSocialServiceType.SINAWEIBO`, as well as a message- and optionally, and image and URL. Before calling it, you should check `isSocialServiceTypeAvailable()` to make sure the service is enabled and the OS supports it.

```
// share with Sina Weibo in iOS 6:
if(GoViral.goViral.isSocialServiceAvailable(GVSocialServiceType.SINAWEIBO))
{
    GoViral.goViral.displaySocialComposerView(
        GVSocialServiceType.SINAWEIBO,    // GVSocialServiceType.FACEBOOK or .TWITTER also available
        "I'm sharing this!",               // this is the message to share
        myBitmapData,                     // the (optional) bitmapData image to share
        "http://www.milkmangames.com"      // the (optional) url to share
    );
}
```

After the action is completed, either `GVShareEvent.SOCIAL_COMPOSER_FINISHED` or `GVShareEvent.SOCIAL_COMPOSER_CANCELED` events will be dispatched, depending whether the user completed or canceled the dialog.

10. Getting Your Android Key Hash

Facebook uses a different method of authenticating your Application as yours on Android than it does on iOS; the *Android Key Hash*. Your Android Key Hash is a special key string that is unique to the keystore (.p12 file) you use to sign your application.

To get your Android Key Hash:

1. Build your application with the version of the .ane found in the '/extension/androiddebugane' folder.
2. Run your app on your Android phone. When `initFacebook()` is called, a pop-up window will display your Android Key Hash.
3. You may select the 'Mail' button the pop-up to email the Key Hash to yourself.
4. Take the Key Hash and enter it under 'Android Key Hash' on the developer.facebook.com page for your app.
5. **Facebook authentication will not work on Android until your Key Hash has been entered on developer.facebook.com.**
6. **The Key Hash may be different between your testing/debug app and your release/Google Play app. Be sure to use the correct Key Hash before releasing your app publicly.**
7. **Be sure to switch back to the release version of the ANE before releasing your app to Google Play- otherwise your Key Hash will be compromised and visible to all users.**

11. Troubleshooting Common Problems

"Why does login not work / the app crash / Twitter not work / email not work?"

- The settings in `application.xml` are not correct; refer to *Update Your Application Descriptor*. A common mistake is to forget the 'fb' prefix before your Facebook App ID inside the URL Scheme. Read the example

XML settings carefully- 90% of tech support requests can be solved by fixing missing or incorrect xml settings. If you're using Flash, make sure that you've ticked the box to Mangle Permissions and Manifest Additions Manually.

- The settings on developer.facebook.com are not correct; refer to Section 1, *Update Facebook Developer Site Setup*.
- *For Facebook, if you have sandbox mode enabled on developer.facebook.com, only developers registered with your app will be able to login.*

"How do I use the GoViralExample.as file in Flash Professional CS6?"

- First, create the application, add the extension, and update developer.facebook.com, by following this guide, Sections 1-4.
- Copy and paste GoViralExample.as into the same folder as your .fla. Do not copy and paste its contents on to the timeline. That will not work.
- In Flash properties, under 'Document Class', type 'GoViralExample' (no quotes) and press OK.
- Edit the GoViralExample.as file, and change the FACEBOOK_APP_ID value to your own facebook id.
- Build and install the application.

"Why doesn't the extension work when I run my swf on my Mac / iPhone / Windows Computer?"

- *The extension uses features that are built into the iOS and Android operating systems. The extension will only work when you run it on a phone or tablet.*

"I still need help! How can I get technical support?"

- Your purchase comes with free technical support by email – just send us a message at support@milkmandgames.com . We're here to help! Please remember that...:
- We're open during United States business hours, excluding U.S. Holidays, Monday-Friday, Pacific Standard Time. We strive to answer all support email within 24 hours (not including weekends and holidays) but usually do so much faster. Remember that we may not be in the same time zone.
- Please remember to mention: which extension you're having a problem with, what IDE you're using (such as 'Flash CS6' or 'Flash Builder 4.6'), and what device you're targeting. If you're experiencing an error message, please specify what that message is.
- *We don't provide tech support through blog comments, Facebook, or Twitter. Please email us and we'll be happy to help you out!*