Flurry SDK

iOS Upgrade Instructions

SDK version 7.x.x
Updated: 8/25/2015

Welcome to Flurry!

This file contains:

1. Introduction

The Flurry iOS SDK version 4.x.x, 5.x.x, 6.x.x and 7.x.x are modularized to allow our partners to select exactly the components they wish to integrate into their app. This will allow for more targeted updates and smaller distributions for partners that only integrate a subset of available Flurry services.

The Flurry iOS Analytics library is required for any integration. You may optionally add on the Flurry Ads library. More information on these libraries are contained in their respective README documents, however a brief description follows:

- **Flurry Analytics Agent** allows you to track the usage and behavior of your iPhone application on users' phones for viewing in the Flurry Developer Portal.
- **Flurry Ads Agent** allows you to earn revenue by offering App, Video, and Reengagement recommendations in your app. Supported integrations include banners and takeovers.

2. Upgrading to 7.x.x

The 7.x.x Flurry SDK has an object based FlurryAds API. If you are on SDK 4.x.x or 5.x.x, It is possible to use the static APIs as before, however we highly recommend migrating over to the object API. To upgrade from 6.x.x to 7.x.x simply replace the libraries. The Xcode project changes required to upgrade to 7.x.x is straightforward and should take less than 10 minutes. The process of replacing the static libraries is the same as in previous upgrades. In addition to this, your application's code will have to include headers and make code modifications for the new object APIs

Flurry Analytics

1. The Analytics API is largely unchanged and available via Flurry.h and the Flurry interface as before.

2. In the finder, drag Flurry/ into project's file folder. *(NOTE: Be sure to remove any existing Flurry library folders from your project's file folder before proceeding. Older versions were under FlurryLib or FlurryAnalytics)*

3. Now add it to your project: **Project > Add to project > Flurry**
   - Choose 'Recursively create groups for any added folders'

4. In your Application Delegate:
   - Import Flurry and inside "application:didFinishLaunchingWithOptions:"
     add: [Flurry startSession:@"YOUR_API_KEY"];

```
#import "Flurry.h"

- (BOOL)application:(UIApplication *)application
 didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {

    [Flurry startSession:@"YOUR_API_KEY"];
    //your code
}
```

Flurry Ads

1. Make sure to first follow steps for Flurry Analytics integration above. The Ads library is dependent on the Analytics library. Replace FlurryAds in existing calls that relate to serving ads with FlurryBanner or FlurryInterstitial. Please see the sample code snippet below and the Advertising README included in this release for more details on migrating from the FlurryAds API to the FlurryBanner and FlurryInterstitial APIs.

2. In the finder, drag FlurryAds/ into project's file folder.

3. Now add it to your project: **Project > Add to project > FlurryAds**
   - Choose 'Recursively create groups for any added folders'

4. In your Application Delegate:
   - In your source code, call Analytics' startSession selector. Note that initializing FlurryAds is no longer required for the object based APIs.

Below is a sample 5.x implementation of Analytics and Ads followed by an equivalent 6.x implementation

```
//Flurry 5.x SDK implementation


#import "Flurry.h"

- (BOOL)application:(UIApplication *)application
 didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {
    [Flurry startSession:@"YOUR_API_KEY"];
    [FlurryAds initialize:rootViewController];

   //your code
}
```

```
-(void)showAdInterstitial {

  [FlurryAds setAdDelegate:self];

  if ([FlurryAds adReadyForSpace: @"INTERSTITIAL_MAIN_VIEW"]) {
     // display a pre-fetched ad
     [FlurryAds displayAdForSpace: @"INTERSTITIAL_MAIN_VIEW"
                           onView: self.view
    viewControllerForPresentation: self];

  } else {
     // Fetch an ad (note: optimize ad
    // serving by fetching early)
    [FlurryAds fetchAdForSpace:@"INTERSTITIAL_MAIN_VIEW"
                         frame:self.view.frame
                          size:FULLSCREEN];
 }
}

-(void) showBanner {

   [FlurryAds fetchAndDisplayAdForSpace:@"BANNER_BOTTOM"
                                   view:self.view
                         viewController:self
                                   size:BANNER_TOP];
}
```

```
//Flurry 6.x, 7.x SDK implementation
#import "Flurry.h"
- (BOOL)application:(UIApplication *)application
 didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {

   [Flurry startSession:@"YOUR_API_KEY"];
   //your code
}

@property (strong)  FlurryAdInterstitial* adInterstitial;

-(void)showAdInterstitial {

  if ([self.adInterstitial ready]) {
     // display a pre-fetched ad
    [self.adInterstitial presentWithViewControler:self];
  } else {
     // Fetch an ad (note: optimize ad serving by fetching early)
    self.adInterstitial = [[FlurryAdInterstitial alloc]
                                  initWithSpace:@"INTERSTITIAL_MAIN_VIEW"];
    self.adInterstitial.adDelegate = self;

    [self.adInterstitial fetchAd];
  }
```

```
}


@property (strong)  FlurryAdBanner* adBanner;

-(void)showBanner {
   self.adBanner = [[FlurryAdBanner alloc] initWithSpace:@"BANNER_BOTTOM"];
   self.adBanner.adDelegate = self;
   [self.adBanner fetchAndDisplayAdInView:self.view
            viewControllerForPresentation:self];

}
```

Please let us know if you have any questions. If you need any help, just email support@flurry.com!