

# 百度移动统计 SDK

## 用户手册 (iOS3.0 版)

百度在线网络技术（北京）有限公司

（版权所有,翻版必究）

## 目录

一、	简介.....	3
二、	阅读对象.....	4
三、	版本支持.....	4
四、	集成使用.....	4
1.	添加SDK到项目.....	4
2.	参数申请.....	5
3.	代码集成.....	5
a)	启动功能.....	5
b)	页面统计.....	7
c)	渠道统计.....	7
d)	应用版本统计.....	8
e)	日志发送策略.....	8
五、	联系我们.....	9

版本	修改内容
2.0	增加日志发送策略
	增加应用版本统计
	废弃了启动 GPS 的接口 ( 防止开发者在非 GPS 应用中启用该选项而导致被苹果拒绝上架 )
2.1	更新 TouchJSON 库到最新版本
	支持 IOS6 SDK , XCode4.5 开发环境
	解决与友盟库同时添加的符号冲突
3.0	增加统计事件发生时长
	设置应用进入后台再回到前台为同一次 session 的间隔时间的方法

## 一、 简介

百度移动统计 SDK(IOS)是百度为 IOS 平台提供的统计工具 ( 以下简称 SDK )。该文档提供了对如何使用 SDK 的一个详细说明。建议阅读时下载我们的 API 用例 , 按照用例设置自己的工程。

如有其他问题可以参考网站的FAQ , 或者即使与我们联系 ( [apptongji@baidu.com](mailto:apptongji@baidu.com) )

## 二、 阅读对象

本文档面向所有使用该 SDK 的开发人员、测试人员以及对此感兴趣的其他用户。

## 三、 版本支持

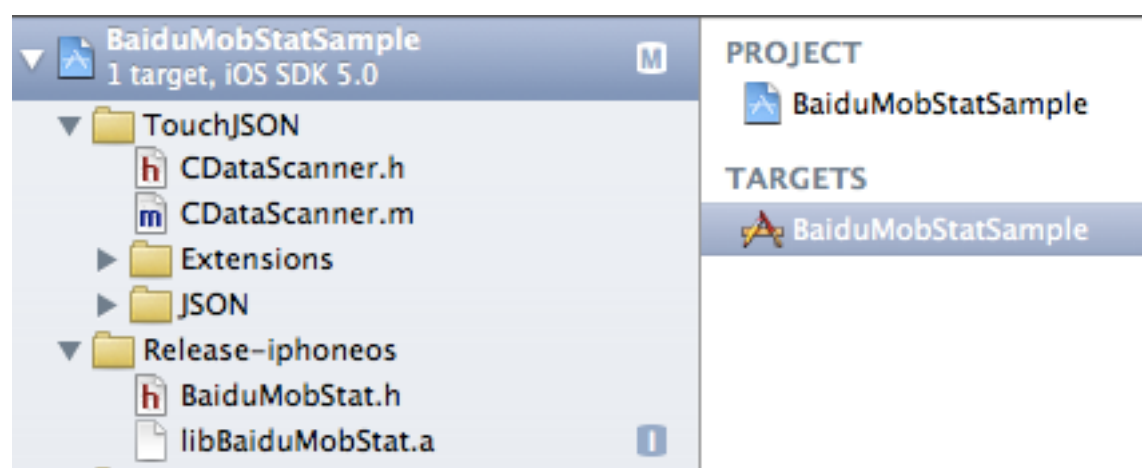
IOS3.0+

## 四、 集成使用

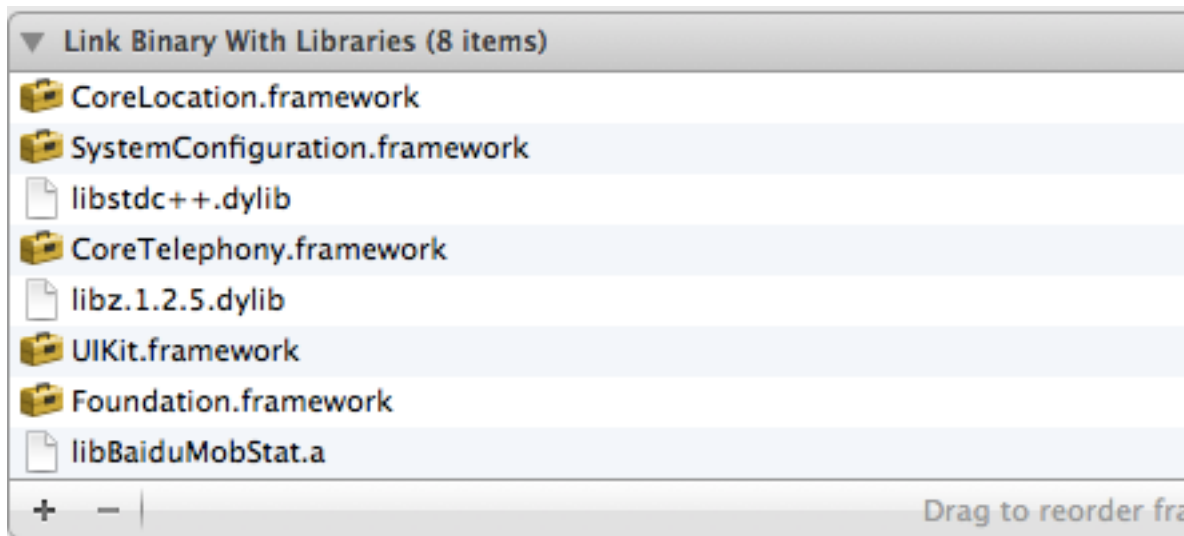
注：以下截图均在示例程序中可以找到

### 1. 添加 SDK 到项目

- 添加 sdk 静态库和第三方 JSON 库



➤ 添加系统依赖库



## 2. 参数申请

在百度移动统计平台(<http://mtj.baidu.com>) 申请应用ID ( APP KEY ) 用于标识您的应用程序。在百度移动统计平台([http:// mtj.baidu.com](http://mtj.baidu.com))的应用配置功能中创建EventId

## 3. 代码集成

### a) 启动功能

**注意：** 2.0 中移除了 `enableLocation` 接口。

在应用启动函数(`didFinishLaunchingWithOptions`)中调用如下代码即可完成启动功能；启动的过程中同时您可以配置是否打开崩溃日志收集 (`statTracker.enableExceptionLog`) 和是否启用 `gps`(`statTracker.enableLocation`)信息。

3.0 中新增设置应用进入后台再回到前台为同一次 session 的间隔时间的方法：

`sessionResumeInterva` 设置范围为[0~600s],超过 600s 则设为 600s，默认为 30s

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:
{
    BaiduMobStat* statTracker = [BaiduMobStat defaultStat];
    statTracker.enableExceptionLog = NO;
    statTracker.channelId = @"Cydia";
    statTracker.logStrategy = BaiduMobStatLogStrategyCustom;
    statTracker.logSendInterval = 1;
    statTracker.logSendWifiOnly = YES;
    statTracker.sessionResumeInterval = 60;
    //statTracker.enableExceptionLog = NO;
    [statTracker startWithAppId:@"ReplaceMeWithAppId"];
}
```

事件统计 ( 注意: eventId 需要预先在服务器端配,eventLabel 不可以为 )

```
- (void)tabBarController:(UITabBarController *)tabBarController didSelectViewController:(UIViewController *)viewController
{
    int index = tabBarController.selectedIndex;
    BaiduMobStat* statTracker = [BaiduMobStat defaultStat];
    [statTracker logEvent:@"TabClick" eventLabel:[NSString stringWithFormat: @"Tab-%d", index]];
}
```

### 3.0 中新增统计事件时长的方法

```
/**
 * v3.0 新增
 * 记录一次事件的时长，eventId请在网站上创建。未创建的eventId记录将无效。
 */
-(void) logEventWithDurationTime:(NSString*) eventId eventLabel:(NSString*)eventLabel;

/**
 * v3.0 新增
 * 记录一次事件的开始，eventId请在网站上创建。未创建的eventId记录将无效。
 */
-(void) eventStart:(NSString*) eventId eventLabel:(NSString*)eventLabel;

/**
 * v3.0 新增
 * 记录一次事件的结束，eventId请在网站上创建。未创建的eventId记录将无效。
 */
-(void) eventEnd:(NSString*) eventId eventLabel:(NSString*)eventLabel;
```

事件时长统计新增两种方法：

1. 在事件开始的时候调用 eventStart 方法，在事件结束时调用 eventEnd 方法。Sdk 自动计算事件发生时长。
2. 调用 logEventWithDurationTime 方法，并将事件时长作为第三个参数 duration 传

入。

#### a) 页面统计

```
#pragma mark - View lifecycle

-(void) viewWillAppear:(BOOL)animated
{
    NSString* cName = [NSString stringWithFormat:@"%s", self.tabBarItem.title, nil];
    [[BaiduMobStat defaultStat] pageviewStartWithName:cName];
}

-(void) viewWillDisappear:(BOOL)animated
{
    NSString* cName = [NSString stringWithFormat:@"%s", self.tabBarItem.title, nil];
    [[BaiduMobStat defaultStat] pageviewEndWithName:cName];
}
```

我们建议在 `UIViewController` 的 `viewWillAppear` 函数中调用 `pageviewStartWithName:xxx`。在 `viewWillDisappear` 函数中调用 `pageviewEndWithName:xxx`。Pageview 的名字为开发者自定义。

您也可以在程序的其他地方构造 pageview 的 start 和 end,只要在逻辑上构成了一段时间的页面访问即可。

#### b) 渠道统计

如下图,您可以在 `startWithAppId` 之前调用 `statTracker.channelId = @"你的渠道名"`,设置不同的渠道名,然后编译成不同的 ipa 文件发布。

如果 `channelId` 属性未设置,系统默认会采用 AppStore 为您的应用渠道。

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    BaiduMobStat* statTracker = [BaiduMobStat defaultStat];
    statTracker.enableExceptionLog = NO;
    statTracker.channelId = @"Cydia";
    statTracker.logStrategy = BaiduMobStatLogStrategyCustom;
    statTracker.logSendInterval = 1;
    statTracker.logSendWifiOnly = YES;
    statTracker.sessionResumeInterval = 60;
    //statTracker.enableExceptionLog = NO;
    [statTracker startWithAppId:@"c00b7d86"];
}
```

### c) 应用版本统计

如下图，您可以在应用配置文件 xxx.plist 中设置 Bundle version，SDK 会读取该值作为您应用的版本号。

Bundle creator OS Type code	String	????
Bundle version	String	1.1
Application requires iPhone environm	  Boolean	YES

### d) 日志发送策略

- 启动时发送：
 

每次启动并联网时会将之前保存在本地的日志发送。本次启动产生的日志将在下一次启动并联网时发送。

**BaiduMobStatLogStrategy = BaiduMobStatLogStrategyAppLaunch**
- 每日发送：
 

设置每日单次发送后，启动时发送数据时会做一次检查，距离上次成功发送数据的时间间隔是否超过 24 小时，如果超过 24 小时，则将之前保存在本地的日志全部发送。如果还不到 24 小时，则不发送。

**BaiduMobStatLogStrategy = BaiduMobStatLogStrategyAppDay**
- 自定义发送间隔：
 

开发者可以自定义发送间隔（单位为小时，1-24 都可以设置），如果开发者设为 N 小时，启动时发送数据时会做一次检查，距离上次成功发送数据的时间间隔是否超过 N 小时，如果超过 N 小时，则将之前保存在本地的日志全部发送。如果还不到 N 小时，则不发送。

**BaiduMobStatLogStrategy = BaiduMobStatLogStrategyCustom**  
**logSentInterval = N**
- 仅在 WIFI 发送：对上述所有发送间隔设置均有效
 

是：必须在 wifi 联网方式下才能发送数据

否：不论联网方式都会发送数据。



---

logSendWifiOnly = YES/NO

## 五、 联系我们

感谢您的阅读,如果有问题请 email 我们。

邮箱: [apptongji@baidu.com](mailto:apptongji@baidu.com)