河北师范大学软件学院
Software College of Hebei Normal University

百度地图实战

Android教研室

- 定位SDK介绍
- 环境配置
- 定位功能

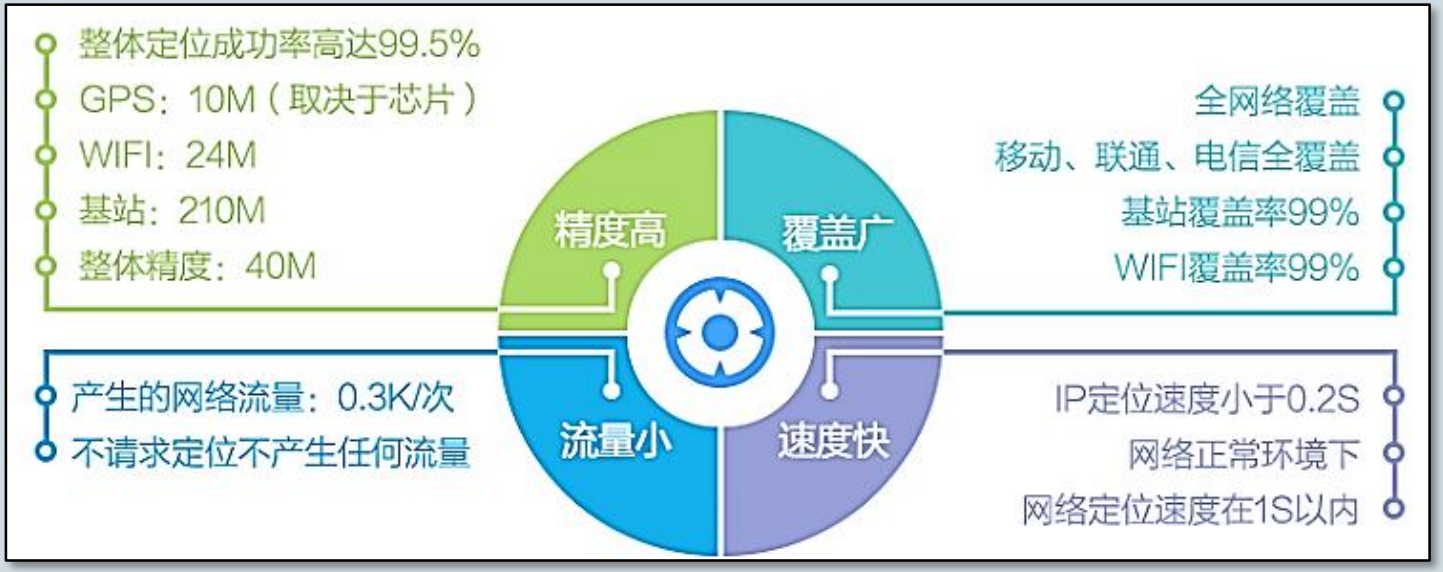
定位SDK介绍

百度地图Android定位SDK是为Android移动端应用提供的一套简单易用的定位服务接口，专注于为广大开发者提供最好的综合定位服务。通过使用百度定位SDK，开发者可以轻松为应用程序实现智能、精准、高效的定位功能。

该套定位SDK免费对外开放，接口使用无次数限制。在使用前，您需先申请密钥（AK）才可使用。

定位SDK介绍

百度地图Android定位SDK提供GPS、基站、WiFi等多种定位方式，适用于室内、室外多种定位场景，具有出色的定位性能：定位精度高、覆盖率广、网络定位请求流量小、定位速度快。



定位SDK介绍

❖ 定位原理

使用百度Android定位SDK必须注册GPS和网络使用权限。定位SDK采用GPS、基站、Wi-Fi信号进行定位。当应用程序向定位SDK发起定位请求时，定位SDK会根据应用的定位因素（GPS、基站、Wi-Fi信号）的实际情况（如是否开启GPS、是否连接网络、是否有信号等）来生成相应定位依据进行定位。

用户可以设置满足自身需求的定位依据：

若用户设置GPS优先，则优先使用GPS进行定位，如果GPS定位未打开或者没有可用位置信息，且网络连接正常，定位SDK则会返回网络定位（即Wi-Fi与基站）的最优结果。为了使获得的网络定位结果更加精确，请打开手机的Wi-Fi开关。

- 定位SDK介绍
- 环境配置
- 定位功能

环境配置

配置环境

1. 导入库文件（liblocsSDK7.so）。
2. 添加定位service（AndroidManifest.xml）。

```
<service
    android:name="com.baidu.location.f"
    android:enabled="true"
    android:process=":remote">
<intent-filter>
<action android:name="com.baidu.location.service_v5.0.0"/>
</intent-filter>
</service>
```

3. 添加权限（AndroidManifest.xml）。

环境配置

```
<uses-permission android:name="android.permission.FOREGROUND_SERVICE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="com.android.launcher.permission.READ_SETTINGS" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<!-- 这个权限用于进行网络定位-->
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<!-- 这个权限用于访问GPS定位-->
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<!-- 获取运营商信息，用于支持提供运营商信息相关的接口-->
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<!-- 用于读取手机当前的状态-->
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
```

- 定位SDK介绍
- 环境配置
- 定位功能

定位功能

- 定位SDK使用步骤

1. 创建定位服务的客户端（LocationClient）。
2. 对定位服务客户端进行设置（LocationClientOption）。
3. 实现监听接口（BDAbstractLocationListener）。
4. 开启定位服务（LocationClient.start()）

定位功能 – 定位服务的客户端类

- Class **LocationClient**

  – 定位服务的客户端类。

LocationClient类必须在主线程中声明，需要Context类型的参数。Context需要是全进程有效的Context,推荐用getApplicationConext获取全进程有效的Context。

```
public LocationClient mLocationClient = null;
public void onCreate() {
    mLocationClient
        = new LocationClient(getApplicationContext());
}
```

定位功能 – 定位服务的客户端类

- 主要方法介绍：

void **setLocOption**(LocationClientOption locOption)	设置 定位SDK参数
void **start**()	启动定位
void **stop**()	停止定位
void **registerLocationListener**(BDLocationListener listener)	注册定位监听函数
void **registerNotify**(BDNotifyListener mNotify)	注册位置提醒监听
void **removeNotifyEvent**(BDNotifyListener mNotify)	取消注册的位置提醒监听

更多方法请查阅百度地图API帮助文档

定位功能 – 配置定位SDK参数类

- Class **LocationClientOption**
  - 配置定位SDK参数类。

配置定位SDK各配置参数，比如定位模式、定位时间间隔、坐标系类型等。

定位功能 – 配置定位SDK参数类

- 主要的参数有：定位模式、返回坐标类型、是否打开GPS等。
- 定位模式分为三种：
  1. 高精度定位模式：同时使用网络定位和GPS定位，优先返回最高精度的定位结果。
  2. 低功耗定位模式：不使用GPS,只使用网络定位（WiFi和基站）。
  3. 仅用设备定位模式： 不使用网络定位，只使用GPS进行定位。但是此模式下不支持室内环境的定位。

定位功能 – 配置定位SDK参数类

- 主要方法介绍：

void **setCoorType**(java.lang.String coorType)	设置坐标类型
void **setIsNeedAddress**(boolean isNeed)	设置是否需要地址信息，默认为无地址
void **setLocationMode**(LocationClientOption .LocationMode mode)	设置定位模式
void **setOpenGps**(boolean openGps)	设置是否打开gps进行定位
void **setScanSpan**(int scanSpan)	设置扫描间隔，单位是毫秒 当<1000(1s)时，定时定位无效

更多方法请查阅百度地图API帮助文档

定位功能 - 配置定位SDK参数类

- 坐标类型：
  - 国测局经纬度坐标系： "gcj02"
  - 百度墨卡托坐标系："bd09"
  - 百度经纬度坐标系："bd09ll"
- 定位模式：
  - 高精度定位模式：Hight_Accuracy
  - 低功耗定位模式：Battery_Saving
  - 仅用设备定位模式： Device_Sensors

定位功能 - 配置定位SDK参数类

```
// 设置定位的相关配置
LocationClientOption option
        = new LocationClientOption();
option.setOpenGps(true);          // 打开GPS
option.setCoorType("bd0911");     // 坐标类型
option.setScanSpan(1000);         // 扫描间隔
// 定位模式
option.setLocationMode(LocationClientOption
        .LocationMode.Hight_Accuracy);
// 通过配置参数对定位客户端进行设置
mLocationClient.setLocOption(option);
```

定位功能 - 定位请求回调接口

❖实现定位请求回调接口（BDLocationListener）

```
// 定位服务客户端设置回调接口
locationClient.registerLocationListener(locationListener);

class MyLocationListener extends BDAbstractLocationListener{
    // 异步返回的定位结果
    @Override
    public void onReceiveLocation(BDLocation location) {
        // 处理定位信息
    }
}
```

定位功能 – 定位请求回调接口

- Class **BDLocation**

– 回调的百度坐标类。

内部封装了如经纬度、半径等属性信息。

java.lang.String **getAddrStr**()	获取详细地址信息，仅在网络定位下使用
double **getLatitude**()	获取纬度坐标
double **getLongitude**()	获取经度坐标
float **getDirection**()	gps定位结果时，行进的方向，单位度
float **getSpeed**()	获取速度，仅gps定位结果时有速度信息，单位公里/小时，默认值0.0f

更多方法请查阅百度地图API帮助文档

定位功能 – 定位请求回调接口

- 定位成功与否，定位的错误码都是通过BDLocation.getLocType()这个方法得到的。

- 61 ： GPS定位结果
- 62 ： 扫描整合定位依据失败。此时定位结果无效。
- 63 ： 网络异常，没有成功向服务器发起请求。此时定位结果无效。
- 65 ： 定位缓存的结果。
- 66 ： 离线定位结果。通过requestOfflineLocaiton调用时对应的返回结果
- 67 ： 离线定位失败。通过requestOfflineLocaiton调用时对应的返回结果
- 68 ： 网络连接失败时，查找本地离线定位时对应的返回结果
- 161： 表示网络定位结果
- 162~167： 服务端定位失败
- 502：key参数错误
- 505：key不存在或者非法
- 601：key服务被开发者自己禁用
- 602：key mcode不匹配
- 501～700：key验证失败

定位功能 – 定位请求回调接口

- 此时可以完成了一个基本的地图定位功能。如果要显示在地图上，就像百度那样出现一个点表示地图的定位点，就需要用到下面两个类：
  - **MyLocationConfiguration**(配置定位图层显示方式)
  - **MylocationData**(定位数据)。

定位功能 – 定位请求回调接口

- Class **MyLocationConfiguration**
  - 配置定位图层显示方式类。

通过BaiduMap.setMyLocationConfigeration()方法设置定位图层信息。

```
public MyLocationConfiguration
            (MyLocationConfiguration.LocationMode mode,
             boolean enableDirection,
             BitmapDescriptor customMarker)
```

参数:

mode - 定位图层显示方式，默认为 LocationMode.NORMAL 普通态

enableDirection - 是否允许显示方向信息

customMarker - 设置用户自定义定位图标，可以为 null

定位功能 – 定位请求回调接口

❖定位图层显示方式：

- MyLocationConfiguration.LocationMode.COMPASS
  罗盘态，显示定位方向圈，保持定位图标在地图中心
- MyLocationConfiguration.LocationMode.FOLLOWING
  跟随态，保持定位图标在地图中心
- MyLocationConfiguration.LocationMode.NORMAL
  普通态： 更新定位数据时不对地图做任何操作

定位功能 – 定位请求回调接口

- Class **MyLocationData**

– 定位数据类。

MyLocationData包含一个内部类Builder用于构建对象。

```
// 构造定位数据
MyLocationData locData = new MyLocationData.Builder()
                    .accuracy(radius)    // 精度
                    .direction(direction)// 方向
                    .latitude(latitude)  // 纬度
                    .longitude(longitude)// 经度
                    .build();  // 构建生成定位数据对象
// 通过BaiduMap.setMyLocationData()设置定位数据
bdMap.setMyLocationData(locData);
```

定位功能 – 定位请求回调接口

通过定位的onReceiveLocation()方法可以得到定位的经纬度，然后可以通过animateMapStatus()方法把定位到的点移动到地图中心。

```
LatLng ll = new LatLng(mCurrentLantitude,
                       mCurrentLongitude);
MapStatusUpdate u
        = MapStatusUpdateFactory.newLatLng(ll);
mBaiduMap.animateMapStatus(u);
```

定位功能 – 开启定位

- 通过定位服务客户端的start()函数来开启定位服务，通过stop()函数来关闭定位服务。
- 因为定位也是比较耗电的，所以通常我们在Activity 的onStart中开启定位，在onStop中关闭定位，这样应用最小化时就不会一直进行GPS请求定位了。

定位功能 - 开启定位

```
protected void onStart() {
    mBaiduMap.setMyLocationEnabled(true);  // 开启图层定位
    if (!mLocationClient.isStarted()) {
        mLocationClient.start();
    }
    super.onStart();
}
protected void onStop() {
    mBaiduMap.setMyLocationEnabled(false); // 关闭图层定位
    mLocationClient.stop();
    super.onStop();
}
```

定位功能

- 示例效果：


河北师范大学(新校区)
师大新校区·理科群3
师大新校区·物理科学与信息工程学院
师大·新校区理科群2
河北师范大学新校区·排球场
师大新校区·物流系
河北师范大学·计算机网络中心
河北师范大学·行政楼
河北师范大学·公教楼a座
河北师范大学·公教楼e座
河北师范大学学生公寓7栋
河北师范大学学生公寓11栋
50米


河北师范大学(新校区)
师大新校区·理科群3
师大新校区·物理科学与信息工程学院
师大·新校区理科群2
河北师范大学新校区·排球场
河北师范大学新校区·篮球场
师大·新校区理科群5
河北师范大学·数信学院
师大新校区·物流系
河北师范大学·计算机网络中心
师大新校区·资源环境系
河北师范大学软件学院微软职业技术学院
河北师范大学·行政楼
河北师范大学·公教楼a座
河北师范大学新校区·研究生院
河北师范大学·公教楼e座
河北师范大学学生公寓
50米

Thank You!