

星與高精度地图 SDK 接入指南

(Android)

修订历史记录

日期	版本	说明	作者
2020.06.18	V1.0.0	创建	星與科技

广东星與科技有限公司

2020 年 6 月



法律声明

Copyright©2020 广东星奥科技有限公司

版权所有，保留所有权利

未经本公司书面许可，任何单位和个人不得以任何方式（包括仿制、复制、誊写或转译）本文档部分或全部内容。不得以任何形式或任何方式（电子、机械、影印、录制或其他可能的方式）进行商品的传播或用于任何商业、盈利目的。对本文档或其包含的任何产品、服务、信息、材料的任何部分进行使用、复制、修改、抄录、传播或其它产品捆绑使用、销售，均视为侵权，本公司必依法追究其法律责任。广东星奥科技有限公司有权对本文档内容进行更新，恕不另行通知。除非另有特殊约定，本手册仅作为使用指导，本手册所有陈述、信息均不承担任何形式的担保。本文档中提到的其它公司及其产品的商标所有权属于该商标的所有者。

广东星奥科技有限公司

联系邮箱：service@starcart.cn

官方网站：www.starcart.cn

目录

第一章	概述	1
第二章	开发指南	1
2.1	开发准备	1
2.1.1	开发环境	1
2.1.2	资源清单	1
2.2	SDK 使用流程	2
2.3	高精度地图接入详细步骤	3
2.3.1	创建 Android 工程	3
2.3.2	导入开发包	3
2.3.3	SDK 初始化	5
2.3.4	高精度地图加载	5
2.3.5	地图界面交互设置	6
2.3.6	设置地图样式	7
2.3.7	标签点操作	8
2.3.8	轨迹线操作	10
2.4	高精度导航服务接入详细步骤	11
2.4.1	创建 Android 工程	11
2.4.2	导入开发包	11
2.4.3	SDK 初始化	12
2.4.4	路径规划页面	12
2.4.5	定位引擎	14
2.4.6	车道级导航引导界面	15

第一章 概述

星與科技高精度地图依靠精准的位置测量和道路元素提取，达到 20 厘米级精度，结合北斗高精度定位网络，实现厘米级精准定位，可以帮助车辆实现 L2-L3 级别辅助驾驶，并为无人驾驶提供精准时空的基础环境。星與高精度地图 SDK 封装了高精度地图和导航服务，降低开发者集成难度，使开发者方便快捷的在 Android 设备上使用高精度地图和导航。

第二章 开发指南

2.1 开发准备

2.1.1 开发环境

Android Studio 3.0+, Android minSdkVersion 19+

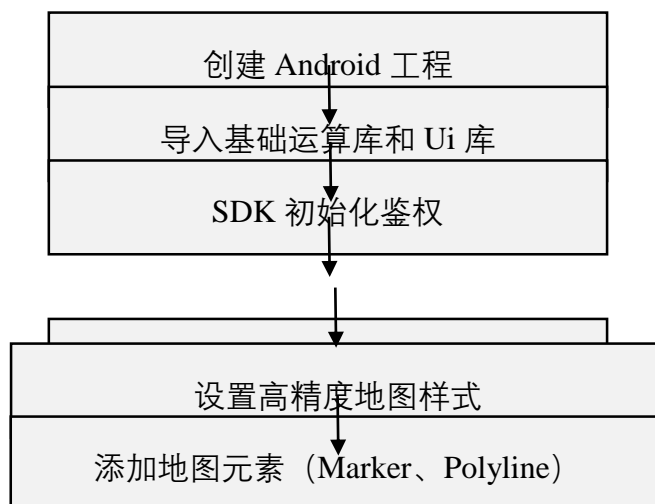
2.1.2 资源清单

- 1、starcart_mapsdk_core_vx.x.x.aar:基础计算库
- 2、starcart_mapsdk_ui_vx.x.x.aar:地图界面操作基础库
- 3、starcart_mapsdk_navi_vx.x.x.aar:车道级导航基础库

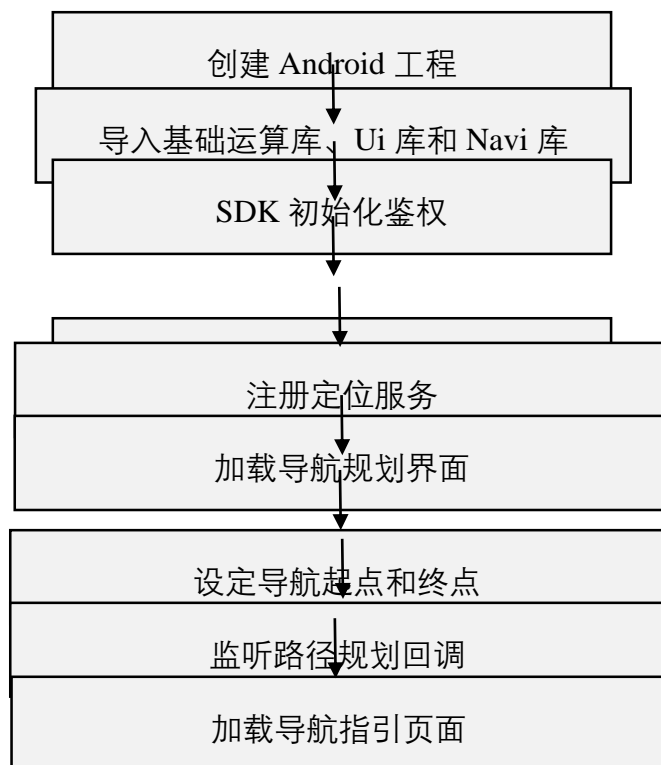
其中，x.x.x 为版本号，根据是否需要导航服务，可选择引入或不引用车道级导航基础库。

2.2 SDK 使用流程

➤ 2.2.1 高精度地图接入流程



➤ 2.2.2 高精度导航服务接入流程



2.3 高精度地图接入详细步骤

2.3.1 创建 Android 工程

在 Android Studio 创建 Android 工程，要求主模块下的 build.gradle 中设置最小开发版本要求为 19。如图：

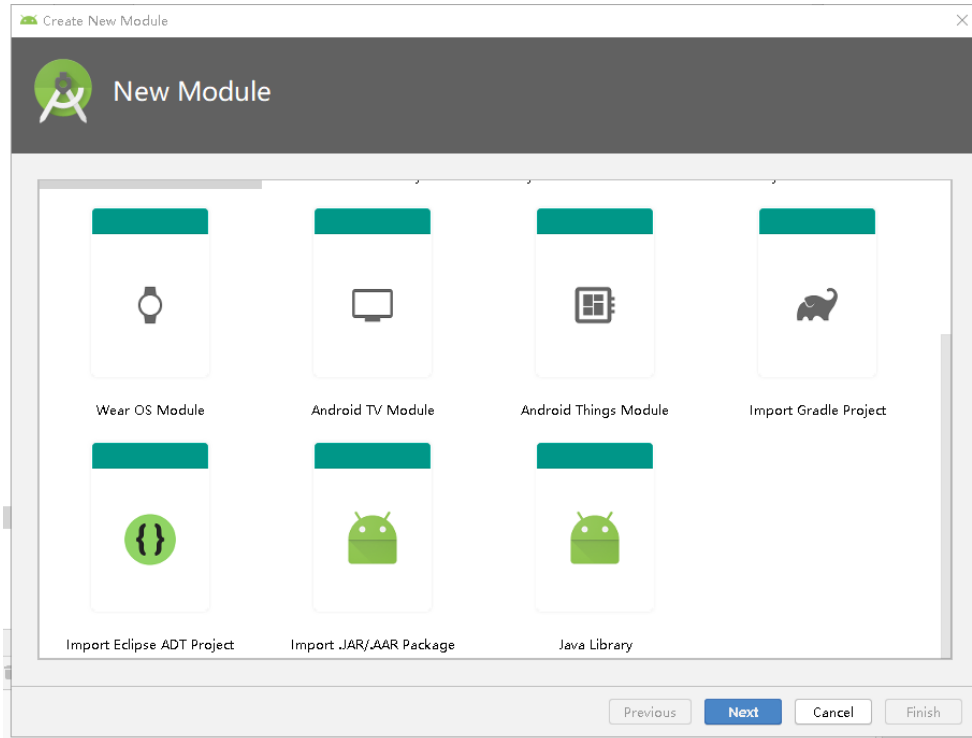
```
defaultConfig {  
    minSdkVersion 19
```

2.3.2 导入开发包

使用 Android Studio 功能菜单 “File->new->new module->Import .JAR/.AAR Package” 导入 2.1.2 所列 aar 文件：

- 1、starcart_mapsdk_core_vx.x.x.aar:基础计算库
- 2、starcart_mapsdk_ui_vx.x.x.aar:地图界面操作基础库

如图：




在模块 build.gardle 中添加如下配置：

```
compileOptions {
    sourceCompatibility JavaVersion.VERSION_1_8
    targetCompatibility JavaVersion.VERSION_1_8
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation "com.mapbox.mapboxsdk:mapbox-android-sdk:9.1.0"
    implementation "com.mapbox.mapboxsdk:mapbox-sdk-turf:5.2.1"
    implementation project(":core")
    implementation project(":ui")
    implementation project(":navi")
}
```

在主模块 AndroidManifest.xml 配置以下权限：

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.INTERNET" />
```

点击 Android Studio 的 Sync project () 按钮，等待项目构建完毕。如图：

```
✓ Build: completed successfully at 2019/9/
▼ ✓ Run build D:\AndroidStudioProjects\t
  ▶ ✓ Load build
  ▶ ✓ Configure build
  ▶ ✓ Calculate task graph
  ▶ ✓ Run tasks
```

2.3.3 SDK 初始化

通过星與位置申请 AppKey，在 Android Application 初始化时调用 SDKInitializer 进行初始化。

```
SDKInitializer.init(this, "你的 AppKey");
```

2.3.4 高精度地图加载

在需要使用地图的页面的布局文件中，添加 StarMapView 布局

```
<cn.starcart.maplib.StarMapView
  android:id="@+id/smv"
  android:layout_width="match_parent"
  android:layout_height="match_parent"/>
```

Activity 中添加地图加载回调监听


```

starMapView.setOnStarMapLoadCallback(new OnStarMapLoadCallback() {
    @Override
    public void onMapReady(StarMap starMap) {
        //地图加载成功
    }

    @Override
    public void onLoadFail(int errorCode) {
        //地图加载失败
    }
});

```

2.3.5 地图界面交互设置

通过 `StarMapView.getUISettings()` 获取地图界面交互设置助手类

UISettings, 具体操作方法有:

方法	作用
<code>showRecenterButton</code>	显示/隐藏内嵌回中按钮
<code>showCompassView</code>	显示/隐藏内嵌指南针
<code>showZoomController</code>	显示/隐藏内嵌地图缩放控制器
<code>showStarScaleBar</code>	显示/隐藏内嵌比例尺
<code>setMaxZoomLevel</code>	设置地图最大缩放级别
<code>setMinZoomLevel</code>	设置地图最小缩放级别
<code>setInitialZoomLevel</code>	设置地图初始缩放级别
<code>setMapStyle</code>	设置地图样式(样式说明见 2.3.6)
<code>cameraTo</code>	地图镜头中心点切换至指定位置

cameraUpdateTo	根据传入的传入动画效果参数，将地图镜头中心点切换至指定位置
----------------	-------------------------------

2.3.6 设置地图样式

地图样式是指地图底图的风格，如街道图、卫星图。可通过 `UISettings.setMapStyle` 设置地图样式，通过 `StarMap.changeMapStyle` 方法改变地图样式。内置的地图样式有：

样式名称	说明
STYLE_COMMON_STREET	英文街道底图，无高精度数据
STYLE_COMMON_SATELLITE	卫星底图，无高精度数据
STYLE_CHINESE_STREET	中文街道底图，无高精度数据
STYLE_OSM_STERRT	OSM 街道底图，无高精度数据
STYLE_OSM_SATELLITE	OSM 卫星底图，无高精度数据
STYLE_STARCART_HDMAP_VECTOR	高精度矢量地图，无底图
STYLE_STARCART_HDMAP_RASTER	高精度瓦片地图，无底图
STYLE_STARCART_HDMAP_STREET_VECTOR	高精度矢量地图，中文街道底图

STYLE_STARCART_HDMAP_STREET_RASTER	高精度瓦片地图, 中文街道底图
STYLE_STARCART_HDMAP_SATELLITE_VECTOR	高精度矢量地图, 卫星底图
STYLE_STARCART_HDMAP_SATELLITE_RASTER	高精度瓦片地图, 卫星底图
STYLE_STARCART_OSM_HDMAP_SATELLITE_VECTOR	高精度矢量地图, OSM 卫星底图
STYLE_STARCART_OSM_HDMAP_SATELLITE_RASTER	高精度瓦片地图, OSM 卫星底图
STYLE_STARCART_OSM_HDMAP_STREET_VECTOR	高精度矢量地图, OSM 街道底图
STYLE_STARCART_OSM_HDMAP_STREET_RASTER	高精度瓦片地图, OSM 街道底图

2.3.7 标签点操作

通过 StarMarkerBuilder 类, 可以在地图上指定位置添加标签点, 如

```
StarMarker maker=starMarkerBuilder.setName(“MY_LOCATION”)
    .showMarkInfo(true)
    .setMarkInfoText(“我的位置”)
    .addToMap(mStarMap, location);
```

StarMarkerBuilder 实现的方法有：

方法	作用
----	----

setName	设置标签点名称，必选项，且全局唯一，否则会报 StarMarkerAddException
setColor	设置标签点颜色，若未设置标签点图片，则会生成该颜色的小圆点图标，默认为蓝色
setIconSize	设置标签点尺寸，若未设置标签点图片，则会生成该尺寸的小圆点图标，默认为 50
showMarkInfo	是否在标签点旁显示文字备注信息
setMarkerIconBitmap	设置标签点图标，若设置该选项，则在地图上绘制该图片
setAutoBearingIcon	在更新定位时，是否自动切换图标朝向。例如设置图标为小车时，设置车头根据 Location 定位调整朝向
setMarkInfoText	设置标签点备注文字内容
setMarkInfoSize	设置标签点备注文字字体大小
addToMap	根据如上配置，将 StarMaker 添加到地图指定位置，并返回 StarMarker

StarMarker 添加到地图上之后，addToMap 方法返回 StarMarker，可通过以下方法对其进行操作：

方法	作用
----	----

showMarkInfo	显示/隐藏标签点文字备注信息
changeMarkerIconBitmap	更改标签点图标
setAutoBearingIcon	启用/停用标签点图标根据定位自动调整朝向
clearStarMarkers	清除指定地图上所有的标签点
jumpToLocation	标签点跳转到指定位置
animToLocation	标签点平滑移动到指定位置
setOnMarkerAnimPointChangeListener	监听标签点平滑移动事件，回调平滑移动过程中每一时刻标签点经纬度插值坐标
remove	将标签点从地图上移除

2.3.8 轨迹线操作

通过 StarPolyLineBuilder 类，可以根据指定位位置集合，在地图上生成轨迹线，如

```
StarPolyLine trackPolyline=new StarPolyLineBuilder ()
    .setColor(color)
    .setName(name)
    .setWidth(Float.parseFloat(polylineSize))
    .addToMap(mStarMap);
trackPolyline.onNewLocation(location);
```

StarPolyLineBuilder 可设置如下属性：

方法	作用
----	----

setName	设置轨迹线名称，必选且全局唯一， 否则将抛出 StarPolylineAddException
setColor	设置轨迹线颜色
setWidth	设置轨迹线粗细
addToMap	添加轨迹线到地图上，返回 StarPolyLine 对象

StarPolyLine 具有以下方法：

方法	作用
setWidth	改变轨迹线粗细
onNewLocation	添加轨迹点到轨迹线上，立即绘制
drawLocationList	更新轨迹点坐标，立即绘制
clearStarPolylines	清除地图上所有的轨迹线
animToLocation	轨迹线平滑连接到最新定位点
remove	移除轨迹线

2.4 高精度导航服务接入详细步骤

2.4.1 创建 Android 工程

配置同 2.3.1.

2.4.2 导入开发包

配置同 2.3.2，另需导入导航服务 SDK：

starcart_mapsdk_navi_vx.x.x.aar:车道级导航基础库

2.4.3 SDK 初始化

同 2.4.3, 需要先在 Application 中调用 SDKInitializer 进行初始化。

```
SDKInitializer.init(this, "你的 AppKey");
```

2.4.4 路径规划页面

路径规划 View 内嵌了高精度地图, 在布局文件中增加以下内容, 可使用内嵌的路径规划页面。

```
<cn.starcart.mapsdk.navi.StarNavigationRoutingView  
    app:initLatitude="23.12199887507846"  
    app:initLongitude="113.34587317490309"  
    app:initScale="16.5"  
    app:initTilt="0"  
    android:id="@+id/snrView"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"/>  
</androidx.constraintlayout.widget.ConstraintLayout>
```

其中 app:initLatitude 和 app:initLongitude 为地图初始化经纬度, app:initScale 为地图初始化缩放缩放级别, app:initTilt 为地图初始化倾斜角。

StarNavigationRoutingView 类实现了如下方法:

方法	作用
getStarMapView	获取路径规划 View 内嵌的高精度地图 View

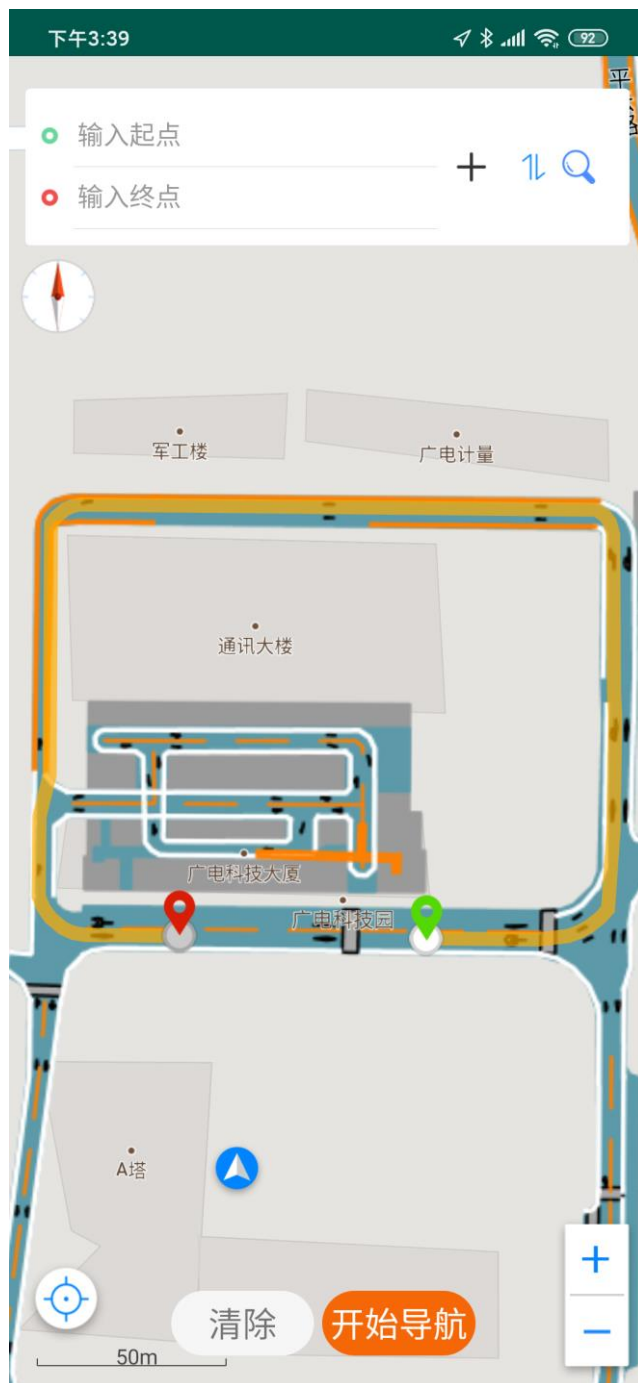
getStarMap	获取路径规划 View 内嵌的高精度地图
setOnStarMapLoadCallback	设置高精度地图加载回调
useEmbedNaviGuideUi	设置在路径规划完成后，是否使用内嵌的导航引导界面
setNaviActivityClass	若使用内嵌的导航引导界面，则应指定导航引导界面所在的 Activity，点击“开始导航”后，会跳转到该 Activity
setStarLocationEngine	设置定位引擎，路径规划页面会使用定位引擎更新位置信息
setLaneNaviRoutingCallback	设置车道级路径规划结果回调，若不使用内嵌的导航引导界面，则可以根据该结果自定义导航引导界面

可通过以下方式开始路径规划：

- 1, 先后点击地图两点，分别作为路径规划的起点和终点。
- 2, 使用POI搜索地图上的位置，添加路径规划的起点、途径点喝终点。（待开发）

点击完毕后请求路径规划引擎，规划结果绘制在地图上，并通过

LaneNaviRoutingCallback 回调给客户，如图：



2.4.5 定位引擎

StarLocationEngine 为导航服务提供定位信息，定位来源由 APP 端控制，可以是 Andorid 设备自身定位。也可以搭配相应的定位 SDK 或模组，使用星奥手机高精度定位算法输出结果、星奥手机助手定位结果等定位信息。

其开放函数有：

函数	说明
registLocationListener	注册定位监听回调
unregistLocationListener	反注册定位监听回调
clear	清除所有已注册的定位监听回调
active	启用/重新启用定位引擎
disactive	暂停定位引擎
onLocationChange	更新定位

2.4.6 车道级导航引导界面

若使用内嵌的车道级导航引导界面，需要在 Activity 的布局文件中增加以下内容：

```
<cn.starcart.mapsdk.navi.StarNavigationView
    app:initLatitude="23.12199887507846"
    app:initLongitude="113.34587317490309"
    app:initScale="18"
    app:initTilt="45"
    android:id="@+id/sngView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"/>
```

其中 app:initLatitude 和 app:initLongitude 为地图初始化经纬度，app:initScale 为地图初始化缩放缩放级别，app:initTilt 为地图初始化倾斜角。

并在 Activity 初始化后增加以下内容：

```
Intent intent=getIntent();
String naviRouteKey=intent.getStringExtra(NaviConst.KEY_ROUTING);
starNavigationView.setNaviRoutingKey(naviRouteKey);
```

naviRouteKey 代表导航路径规划数据的索引，StarNavigationGuideView 通过该索引获取导航引导数据。

StarNavigationGuideView 开放的方法有：

函数	说明
setNaviRoutingKey	初始化后设置导航路径规划数据索引，根据该索引获取使用的导航数据
setOnStarMapLoadCallback	高精度地图加载回调
setStarLocationEngine	设置定位引擎，导航引导页面会使用定位引擎更新位置信息

导航引导界面会根据定位信息，在界面上给出当前所在道路，下一路口指引、剩余里程、车速等信息。如图：

