

扫码服务使用文档

Version=3.9.2

Version History

Version	Author	Date	Description
3.9.0	李鹏	20181218	1:添加 zebra 扫码库的参数说明。 2:整理文档格式,添加文档版本历史。
3.9.1	李鹏	20181219	1:修正文档中页眉的错误描述. 2:添加 zebra 扫码库的参数说明。
3.9.2	李鹏	20181219	1:修复参数定义错误. 2:修正参数描述不准确的错误.

1. 简介

本文档介绍了慧银扫码服务的使用说明，主要包括接口描述，参数说明，以及调用服务的方法

2. 接口及参数介绍

2.1. 接口介绍

2.1.1. 同步扫码 `scanBarcode`

本接口是同步调用接口。推荐放在线程中操作

应用调用该接口后，扫码服务按照扫码参数的定义打开摄像头，并开始扫码。扫到结果后，关闭摄像头并立即返回结果。

参数：

`ScanParameter`

返回值：

`ScanResult`, 扫码结果可获得内容见 3.2.3

2.1.2. 开始连续扫码 `startScan`

本接口是异步调用接口。表示开始连续扫码。

应用调用该接口后，扫码服务按照扫码参数的定义打开摄像头，并开始扫码。每一次扫到结果后，在回调中返回结果，回调内的操作执行完后开始下一次扫码。

参数：

`ScanParameter`, `IScanCallBack`

返回值：

`void`

2.1.3. 扫码结果回调接口 foundBarcode

连续扫码内参数 `IScanCallBack` 需实现的方法。调用者可以通过这个接口拿到 `ScannerResult`。这个接口调用时候扫码服务处于 `pause` 状态，调用返回后，才继续进行下一次扫码动作。

可以通过 `stopScan` 关闭正在处于 `pause` 中的扫描服务。

参数：

`ScanResult`

返回值：

`Void`

2.1.4. 停止连续扫码 stopScan

停止连续扫码，并关闭扫描服务的 UI。停止后，其他调用者可以调用 `startScan`，或者 `scanBarcode` 接口。

调用该接口时，如果是同步扫码，`scanBarcode` 方法会返回 `resultCode = 0` 的 `ScanResult`。如果是连续扫码，则不会调用它的 `IScanCallBack` 回调

返回值：

`Boolean`

2.2. 参数介绍

2.2.1. 扫码参数 ScanParameter

`ScanParameter` 对外采用以下方法设置参数，且这些参数采用键值对的方式记录。

方法：`set(String key, String value)` (key 不区分大小写)

关键字 Key	Value 类型	取值范围和默认值	说明
<code>window_top</code>	类型:int	默认值: 0, 取值范围: >0	窗口左上角距离屏幕顶。（仅在悬浮窗模式下生效） 单位 dp
<code>window_left</code>	类型:int	默认值: 0, 取值范围: >0	距离屏幕左边的距离。（仅在悬浮窗模式下生效） 单位 dp

window_width	类型:int	默认值: 屏幕宽 取值范围: >0	窗口宽。(仅在悬浮窗模式下生效) 单位 dp
window_height	类型:int	默认值: 屏幕高 取值范围: >0	窗口高。(仅在悬浮窗模式下生效) 单位 dp
enable_scan_section	类型:boolean	默认值: true 取值范围: True/false	False, 整个显示窗口都是扫码区域。会去掉扫码框 True, 自定义扫码区域, 不需要扫全部图片。扫码区域会有边框, 其他部分半透明显示。 扫码框始终居中, 但可以调整宽高
scan_section_width	类型:int	默认值: 300dp 取值范围: >0	扫码框的宽。
scan_section_height	类型:int	默认值: 300dp 取值范围: >0	扫码框的高。
display_scan_line	类型:String	默认值: moving 取值范围: No/fixed/moving	扫码区域内显示指示的红线。 不显示/固定中间/上下移动
enable_flash_icon	类型:boolean	W1 上默认值:true Q1 上默认值:false 取值范围: True/false	是否显示打开/关闭散光灯的悬浮按钮。
enable_switch_icon	类型: boolean	默认值: true 取值范围: True/false	是否显示切换摄像头按钮。
enable_indicator_light	类型: boolean	默认值: false 取值范围: True/false	是否显示扫描指示灯, 仅在 Q1 设备上支持该功能。
decodeformat	类型:String	默认值: 取值范围: 见附录条码格式	解码范围。默认是 BARCODE_ALL。 多个解码范围用“,”分隔
decoder_mode	类型:int	默认值: 2	扫码模式

		取值范围: 0/1/2	0, mode1; 1, mode2; 2, mode3;
enable_return_image	类型:boolean	默认值: false 取值范围: True/false	是否返回识别出的Bitmap
camera_index	类型:int	默认值: 0 取值范围: 0/1/2	0, 主扫码头(定焦/定焦); 1, 第二摄像头(变焦); 2, 客显摄像头。
scan_time_out	类型:long 单位是毫秒	默认值: -1 取值范围: >0	小于等于 0 代表永远扫描, 大于 0 代表尽在这段时间进行扫描, 超过则返回扫描超时。 仅对同步接口有效。
scan_section_border_color	类型:int	默认值: Color.WHITE	扫码框边框颜色 传入的颜色为 int 值, 可使用 Color.argb 方法
scan_section_corner_color	类型:int	默认值: Color.argb(0xFF, 0x21, 0xDB, 0xD5)	扫码框边角颜色
scan_section_line_color	类型:int	默认值: Color.RED	扫码线颜色
scan_tip_text	类型: String	默认值: “将图片对准扫描框即可自动扫描”	扫码框下面的提示文字
scan_tip_textSize	类型:int	默认值: 15	扫码框下面的提示文字 字体大小 单位 sp
scan_tip_textColor	类型:int	默认值: Color.WHITE	扫码框下面的提示文字 颜色
scan_tip_textMargin	类型:int	默认值: 30	扫码框下面的提示文字 距离底部距离 单位 dp
flash_light_state	类型:boolean	默认值:false	初始化闪光灯开关状态 true:打开 false:关闭
indicator_light_state	类型:boolean	默认值:false	初始化指示灯开关状态 true:打开 false:关闭
scan_mode	类型:String	默认值:dialog	扫码框模式 dialog: dialog 主题的 activity

			overlay: 悬浮窗
scan_camera_exposure	类型: int	默认值:0	<p>摄像头曝光补偿值</p> <p>有时扫描手机上的二维码会出现屏幕太亮扫不到码的情况, 这时可以设置为负值。</p> <p>具体数值由于不用摄像头不一样, 需要自己通过 Camera 去获得支持的取值范围。</p> <p>另外, 只有在变焦摄像头下设置才会影响扫码效果, 定焦摄像头无效</p>
scan_time_limit	类型: int	默认值: 50	<p>最大解码时间</p> <p>提高数值能增加成功率但是扫码速率会下降。</p> <p>经测试 50 为最合适, 不建议修改该值</p>
enable_mirror_scan	类型: boolean	默认值: true	<p>是否支持镜像扫码</p> <p>默认为打开</p>
enable_hands_free	类型: boolean	默认值: true	<p>HandsFree 模式会启用移动侦测和移动补光. 该模式被用于连续扫码的业务场景.</p> <p>True:启用 HandsFree 模式.</p> <p>False:关闭 HandsFree 模式.</p> <p>注:</p> <p>该功能仅对 zebra 扫码头开放.</p>
enable_ui_by_zebra	类型: boolean	默认值: true	<p>True:启用默认扫码 UI.</p> <p>False:隐藏默认扫码 UI.</p> <p>隐藏默认扫码 UI 可加快扫码的启动速度.</p> <p>注:</p> <p>该功能仅对 zebra 扫码头开放.</p>
enable_mobile_phone_screen_mode	类型: boolean	默认值: false	<p>手机电子屏扫码模式, 该模式会提高扫码识别率.</p> <p>True:启用手机电子屏扫码模式.</p> <p>False:关闭手机电子屏扫码模式</p> <p>启用手机电子屏扫码模</p>

			式会导致每次解码的解码时间变长. 注: 该功能仅对 zebra 扫码头开放.
enable_upca_country	类型: boolean	默认值: true	True:在 UPC_A 解码后传递国家编号. False:在 UPC_A 解码后隐藏首位的国家编号. 注: 该功能仅对 zebra 扫码头开放.
enable_decoding_illumination	类型: boolean	默认值: true	True:打开补光灯. False:关闭补光灯. 注: 该功能仅对 zebra 扫码头开放.
enable_motion_illumination	类型: boolean	默认值: false	True:补光灯常亮, False:没有物体移动时补光灯自动关闭. 注: 该功能仅对 zebra 扫码头的 handsfree 模式.

2.2.2. 悬浮窗模式

悬浮窗模式下用户可以自己定义扫码框外的 UI,并可以同时操作自己的 UI 和扫码框,这样客户可以自己定义切换摄像头,开关闪光灯,开关指示灯的按钮,以及标题栏之类。切换摄像头,开关闪光灯,开关指示灯通过本身扫码框的按钮或者发送广播的方式控制。

切换摄像头: 广播 Action : com.wizarpos.scanner.setcamera

广播传值 Key: overlay_config

Value: 0, 主扫码头(定焦); 1, 第二摄像头(变焦); 2,客显摄像头。

切换闪光灯: 广播 Action : com.wizarpos.scanner.setflashlight

广播传值 Key: overlay_config

Value: true, 打开; false, 关闭

切换指示灯: 广播 Action : com.wizarpos.scanner.setindicator

广播传值 Key : overlay_config

Value: true, 打开; false, 关闭

以上值可以在直接复制, 需注意大小写或使用 ScanParameter 内的常量

示例: 打开闪光灯

```
Intent intent = new Intent();
intent.setAction(ScanParameter.BROADCAST_SET_FLASHLIGHT);
intent.putExtra(ScanParameter.BROADCAST_VALUE, true);
sendBroadcast(intent);
```


需要注意，该模式下扫码窗口无法捕捉到 `back` 和 `home` 键，需要开发人员重写 `onKeyDown` 方法，捕捉到 `back` 和 `home` 键操作后调用 `stopScan` 方法，否则会出现自己页面消失而扫码窗口还存在的情况。

2.2.3. Zebra 摄像头扫码

使用 `zebra` 摄像头需要满足以下条件。

- 1:终端上存在 `zebra` 摄像头。
- 2:需要在扫码参数上设置扫码头为主扫码头(0)。

注意:

- 1:当屏幕熄灭后将无法使用 `zebra`。

使用 `zebra` 摄像头扫码时请自己设置参数来隐藏 UI

通过设置参数 `enable_ui_by_zebra` 为 `false` 来实现该功能。

2.2.4. 扫码结果 ScanResult

域	类型	说明
<code>resultCode</code>	<code>Int</code>	<code>>=0</code> : 成功; <code><0</code> : 失败。见错误码定义
<code>text</code>	<code>String</code>	字符串结果(当发生错误时不返回)(使用了 <code>uft-8</code> 格式, 如要其他格式获取到 <code>rawBuffer</code> 后自己转换)
<code>rawBuffer</code>	<code>Byte[]</code>	字节流结果
<code>bitmap</code>	<code>Bitmap</code>	扫描成功的图像(需设置参数 <code>enable_return_image</code> 为 <code>true</code> 才有值)
<code>barcodeFormat</code>	<code>String</code>	条码格式, 见附录表格内条码格式部分

2.2.5. 错误码

值	说明
1	成功
0	用户取消
2	扫码窗口完全显示的通知
-1	扫码服务被占用
-2	扫码头/摄像头无法打开

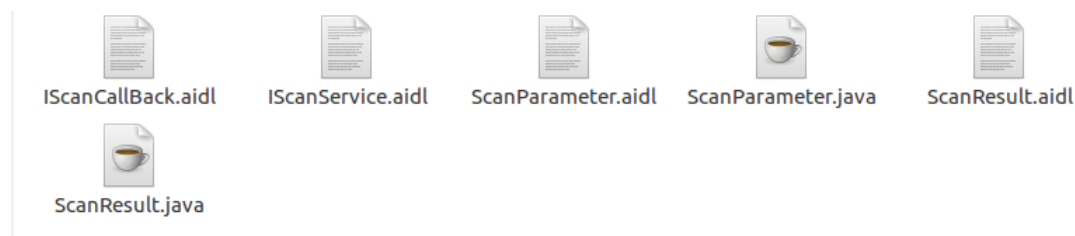
-3	扫描超时
-4	参数错误

3. 使用说明

3.1. 扫码服务集成

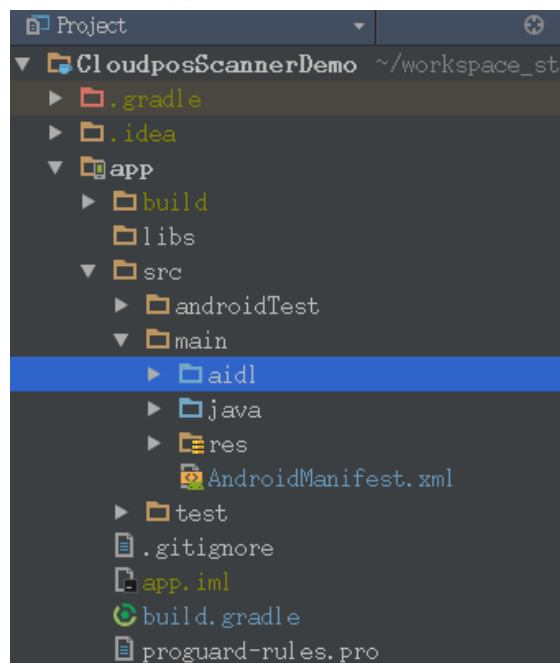
本扫码服务使用的是通过 AIDL 方式调起系统内的扫码服务，所以需要加项目中加入必要的 aidl 文件，以下是在 Eclipse 和 Android Studio 中集成方式。

需要集成的文件一共有以下几个：

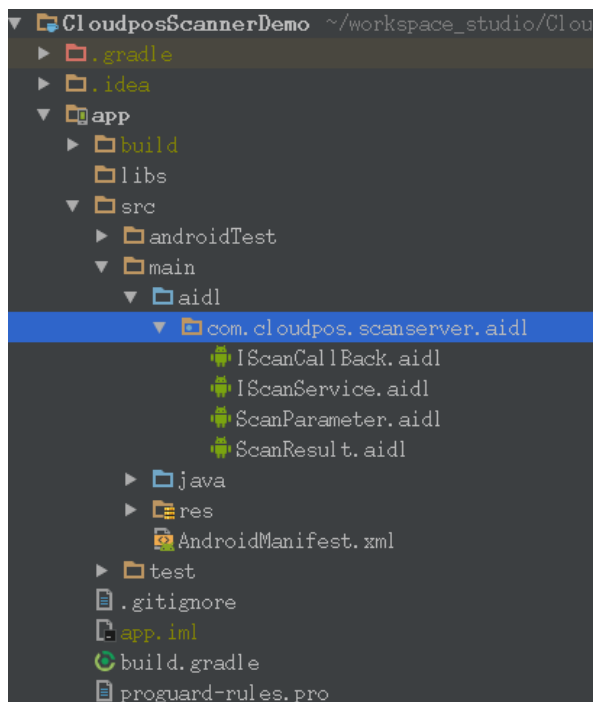


Eclipse 中只要把以上文件放在包名“com.cloudpos.scanserver.aidl”下即可

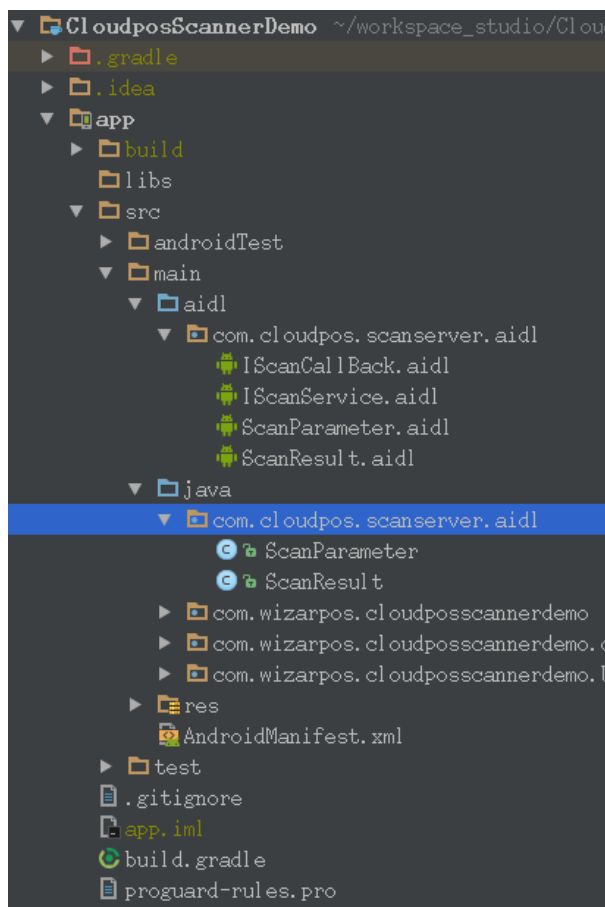
Android Studio 需要把 aidl 文件和 java 文件分开放，在项目下“src--main”目录下右键新建文件夹，名称为 aidl



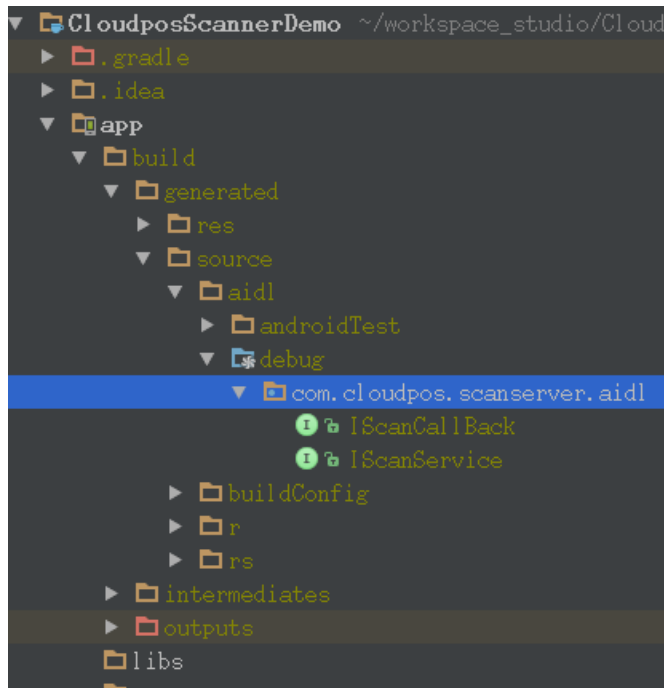
在 aidl 文件夹下新建包名“com.cloudpos.scanserver.aidl”，并把 4 个 aidl 文件放在包下



在“src—main—java”文件夹下新建包名“com.cloudpos.scanserver.aidl”，并把 2 个 java 文件放在包下



clean project，如果在“build—generated—source—aidl--debug”下成功编译出文件就可以调用服务了



剩下的是绑定 AIDL 服务，你可以用自己的方法也可以用我们提供的一个接口，接口使用方法如下

将我们提供的接口和实现的 java 文件放在项目的任意包名下



AidlController.java



IAIDLListener.java

在需要绑定服务时使用以下方法

```
AidlController.getInstance().startScanService(this, this);
```

然后让类实现 IAIDLListener 接口，并重写其中方法即可拿到扫码服务

```
private IScannService scanService; //扫码服务
private ServiceConnection scanConn;

@Override
public void serviceConnected(Object objService, ServiceConnection connection) {
    if(objService instanceof IScannService){
        scanService = (IScannService) objService;
        scanConn = connection;
    }
}
```

通过 scanService 就可以调用扫码服务的接口

在需要解绑服务时使用以下方法

```
@Override
```

```
protected void onDestroy() {
    if(scanService != null){
        this.unbindService(scanConn);
        scanService = null;
        scanConn = null;
    }
    super.onDestroy();
}
```

详细代码可以看提供的 demo 项目

4. 附录

4.1. 条码格式

使用方法：

```
ScanParameter parameter = new ScanParameter();
parameter.set(ScanParameter.KEY_DECODEFORMAT, " AZTEC, QR");
```

设置解码范围时除了以下所有条码格式外，还可使用以下三个特殊字符串，	
BARCODE_ALL	包括以下所有条码格式
BARCODE_1D	包括条码格式内所有 1D Barcode
BARCODE_2D	包括条码格式内所有 2D Barcode
条码格式	
AZTEC	2D barcode
DATAMATRIX	2D barcode
QR	2D barcode
MAXICODE	2D barcode
PDF417	2D barcode
CODABAR	1D barcode
CODE39	1D barcode
CODE93	1D barcode
CODE128	1D barcode
EAN8	1D barcode
EAN13	1D barcode
ITF	1Dbarcode(Interleaved Two of Five)
RSS_14	1D barcode
RSS_EXPANDED	1D barcode
UPCA	1D barcode
UPCE	1D barcode

CODE11	1D barcode
--------	------------