

Nfc开发指南

版本	更改人	日期	更改内容
V1.0	ct	2018-11-21	

目录

Nfc开发指南	1
文档概述	2
适用人群	2
文档目的	2
开发环境和工具	2
1.1开发环境和工具	2
1.2认识Android manifest 文件与NFC相关权限和功能	3
1.3允许设备访问NFC硬件	3
1.4 可以在android market中显示有NFC硬件	3
1.5 最低版本要求	3
1.6 Android NFC消息发送机制	3
1.7 Intent发送机制	4
1.8 Android 开发和配置	4
1.8.2 Activity代码	6

文档概述

NFC模块接口描述

适用人群

软件开发人员

软件测试人员

文档目的

为软件开发人员提供参考

开发环境和工具

开发平台：win7旗舰版 64

开发工具：Android studio

编译环境：ndkr16B java 1.8

1.1开发环境和工具

开发平台：win7旗舰版 64

开发工具：Android studio 3.1

编译环境：ndkr16B java 1.8

1.2认识Android manifest 文件与NFC相关权限和功能

NFC应用开发涉及到对NFC设备访问的权限，对系统最低版本的要求（Android 2.3.3及以上的版本才支持此功能），以及处理NFC消息的IntentFilter等一系列的设置，所以我们先对Android manifest文件中申明和NFC相关的权限和功能等进行简单的认识。

1.3允许设备访问NFC硬件

```
<uses-permission android:name="android.permission.NFC" />
```

1.4 可以在android market中显示有NFC硬件

```
<uses-feature android:name="android.hardware.nfc" android:required="true" />
```

1.5 最低版本要求

NFC是Android2.3（Level 10）才开始支持的，因此最低版本要求必须指定为10。

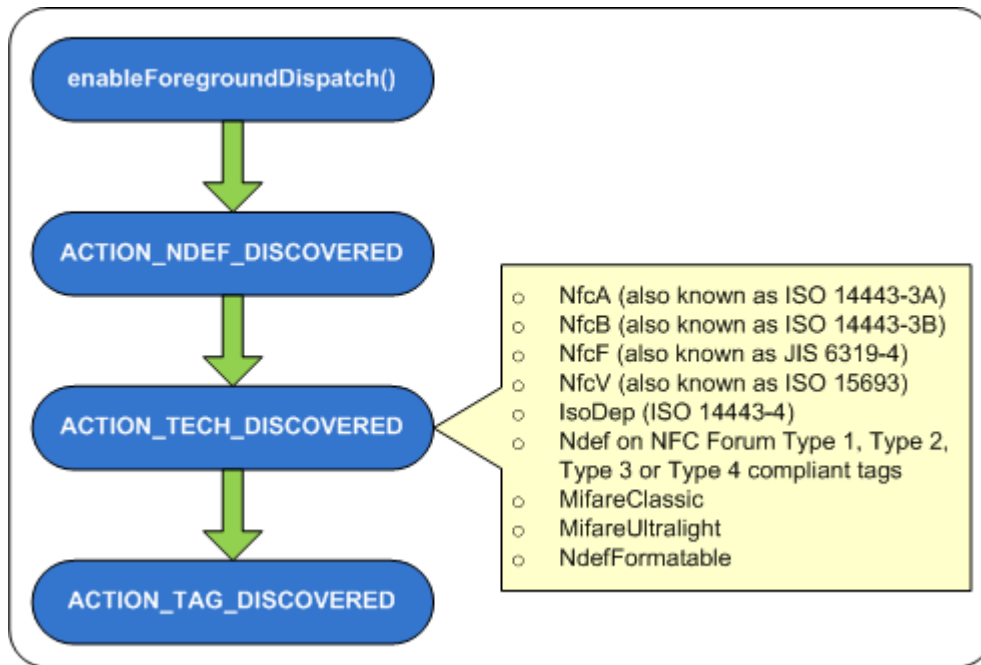
```
<uses-sdk  
  
    android:minSdkVersion="10"  
  
    android:targetSdkVersion="15" />
```

1.6 Android NFC消息发送机制

当Android设备检测到有NFC Tag时，理想的行为是触发最合适的Activity来处理检测到的Tag，这是因为NFC通常是在非常近的距离才起作用(<4cm)，如果此时需要用户来选择合适的应用来处理Tag，很容易断开与Tag之间的通信。因此你需要选择合适的Intent filter 只处理你想读写的Tag类型。

Android系统支持两种NFC消息发送机制：Intent 发送机制和前台Activity 消息发送机制。这两种方法基本上都是使用Intent-filter 来指明Activity可以处理的Tag类型，一个是使用Android的Manifest 来说明，一个是通过代码来声明。

下图显示当Android检测到Tag，消息发送的优先级：



1.7 Intent发送机制

当系统检测到Tag时，Android系统提供manifest 中定义的Intent filter 来选择合适的Activity来处理对应的Tag，当有多个Activity可以处理对应的Tag类型时，则会显示Activity选择窗口由用户选择：



1.8 Android 开发和配置

1.8.1 AndroidManifest.xml中

```

<uses-sdk android:minSdkVersion="10" />
<uses-permission android:name="android.permission.NFC" />
<!-- 要求当前设备必须要有NFC芯片 -->
<uses-feature
    android:name="android.hardware.nfc"
    android:required="true" />

```



```

<activity android:name=".NFCCard">

    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>

    <!-- 当检测到一个NFC标签时，系统自动创建一个相关的Intent对象，含有响应intent filter的Activity将处理这个Intent TECH格式过滤标签 -->
    <intent-filter>
        <action android:name="android.nfc.action.TECH_DISCOVERED" />
    </intent-filter>
    <meta-data
        android:name="android.nfc.action.TECH_DISCOVERED"
        android:resource="@xml/nfc_tech_filter" />

    <!-- NDEF格式 -->
    <intent-filter>
        <action android:name="android.nfc.action.NDEF_DISCOVERED" />
        <data android:mimeType="text/plain" />
    </intent-filter>

    <!-- TAG格式 -->
    <intent-filter>
        <action android:name="android.nfc.action.TAG_DISCOVERED" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>

```

nfc_tech_filter.xml为标签过滤用，请自行在res资源目录下新建xml文件夹，新建一个nfc_tech_filter.xml的文件

```
roidManifest.xml × nfc_tech_filter.xml ×
<?xml version="1.0" encoding="utf-8"?>
<resources xmlns:xliff="urn:oasis:names:tc:xliff:document:1.2">
  <tech-list>
    <tech>android.nfc.tech.NfcA</tech>
  </tech-list>
  <tech-list>
    <tech>android.nfc.tech.NfcB</tech>
  </tech-list>
  <tech-list>
    <tech>android.nfc.tech.NfcF</tech>
  </tech-list>
  <tech-list>
    <tech>android.nfc.tech.NfcV</tech>
  </tech-list>
  <tech-list>
    <tech>android.nfc.tech.Ndef</tech>
  </tech-list>
  <tech-list>
    <tech>android.nfc.tech.NdefFormatable</tech>
  </tech-list>
  <tech-list>
    <tech>android.nfc.tech.IsoDep</tech>
  </tech-list>
  <tech-list>
    <tech>android.nfc.tech.MifareClassic</tech>
  </tech-list>
  <tech-list>
    <tech>android.nfc.tech.MifareUltralight</tech>
  </tech-list>
</resources>
```

1.8.2 Activity代码

具体请参阅提供demo