

IntentFilter

版本号: 2018/08/30-1(16:16)

- [IntentFilter](#)
 - [scheme\(3\)](#)
 - [协议格式](#)
 - [使用](#)
 - [IntentFilter的匹配机制\(18\)](#)
 - [action](#)
 - [category](#)
 - [data](#)
 - [实例](#)
 - [参考资料](#)

scheme(3)

1、scheme是什么？

1. android中的scheme是一种页面内跳转协议，
2. 能够方便地跳转到其他页面
3. 如：根据服务器返回的信息，跳转到目标页面。
4. 如：可以通过通知消息栏，去点击跳转到需要的页面
5. 可以通过H5跳转到需要的页面。

协议格式

2、scheme的协议格式

```
url = "http://www.orangecpp.com:80/tucao?id=hello&name=lily";
```

```
url =          protocol + authority(host + port) + path + query
协议protocol=  http
域名authority= www.orangecpp.com:80
页面path=      /tucao
参数query=     id=hello&name=lily

authority =    host + port
主机host=     www.orangecpp.com
端口port=     80
```

使用

3、scheme的使用必须要配置intent-filter, 才能从其他app跳转到我们的app

1- data中设置scheme

```
<intent-filter>
    <!-- 协议部分，随便设置
        x1://goods:8888/goodsDetail?goodsId=10011002
    -->
    <data android:scheme="x1"
        android:host="goods"
        android:path="/goodsDetail"
        android:port="8888"/>
</intent-filter>
```

2-设置几项必须的action和category

```
<intent-filter>
    <data xxx/>

    <!-- 下面这几行也必须得设置 -->
    <category android:name="android.intent.category.DEFAULT"/>
    <action android:name="android.intent.action.VIEW"/>
    <category android:name="android.intent.category.BROWSABLE"/>
</intent-filter>
```

IntentFilter的匹配机制(18)

1、Activity的显式Intent

```
Intent intent = new Intent();
intent.setAction(MainActivity.this, SecondActivity.class);
startActivity(intent);
```

2、Activity的隐式Intent

```
Intent intent = new Intent();
intent.setAction("com.example.action");
intent.addCategory("com.example.category");
intent.setData(Uri.parse("scheme://www.baidu.com"))
startActivity(intent);
```

3、Intent的隐式调用注意点

1. 隐式调用 需要 Intent 能够匹配目标组件的 IntentFilter 中所设置的过滤信息
2. IntentFilter 过滤信息有 action、category、data -三个类别完全匹配时才算匹配成功
3. Activity 可以有多个 intent-filter，intent 只要匹配任何一组就算匹配成功。

```
<activity
    android:name="com.zte.ipstvclient.android.ipstvclient.activity.InitActivity"
    android:label="@string/app_name"
    android:launchMode="singleTop"
    android:configChanges="orientation|locale|fontScale|keyboardHidden"
    android:screenOrientation="unspecified"
    android:windowSoftInputMode="adjustPan">
<intent-filter>
    <action android:name="android.intent.action.MAIN" />
    <category android:name="android.intent.category.LAUNCHER" />
</intent-filter>
<intent-filter>
    <action android:name="android.intent.action.VIEW" />
    <category android:name="android.intent.category.DEFAULT" />
    <category android:name="android.intent.category.BROWSABLE" />
    <data
        android:host="127.0.0.1"
        android:scheme="zteottptcl2cc82f66de86" />
    </intent-filter>
</activity>
```

4、intent-filter的过滤信息

- 1. action
- 2. category
- 3. data

action

5、action的匹配规则

- 1. action是一个字符串，有系统预定义的action，也可以自定义action
- 2. action的匹配 需要字符串完全一样
- 3. 具有多个action时，只要一个匹配就算做匹配成功，如果 Intent中没有指定action则匹配失败
- 4. action 区分大小写
- 5. Intent只有一个action： setAction()
- 6. 举例： Intent设置了action，该action只要能在intent-filter中找到任何一个符合的，就表示匹配成功。

6、activity的预定义Action

activity预定义了很多Action，常用的如下：

常量名称	常量值	意义
ACTION_MAIN	android.intent.action.MAIN	应用程序入口
ACTION_VIEW	android.intent.action.VIEW	显示数据给用户
ACTION_ATTACH_DATA	android.intent.action.ATTACH_DATA	指明附加信息给其他地方的一些数据
ACTION_EDIT	android.intent.action.EDIT	显示可编辑的数据
ACTION_PICK	android.intent.action.PICK	选择数据
ACTION_CHOOSER	android.intent.action.CHOOSER	显示一个Activity选择器

常量名称	常量值	意义
ACTION_GET_CONTENT	android.intent.action.GET_CONTENT	获得内容
ACTION_DIAL	android.intent.action.GET_CONTENT	显示打电话面板
ACITON_CALL	android.intent.action.DIAL	直接打电话
ACTION_SEND	android.intent.action.SEND	直接发短信
ACTION_SENDTO	android.intent.action.SENDTO	选择发短信
ACTION_ANSWER	android.intent.action.ANSWER	应答电话
ACTION_INSERT	android.intent.action.INSERT	插入数据
ACTION_DELETE	android.intent.action.DELETE	删除数据
ACTION_RUN	android.intent.action.RUN	运行数据
ACTION_SYNC	android.intent.action.SYNC	同步数据
ACTION_PICK_ACTIVITY	android.intent.action.PICK_ACTIVITY	选择Activity
ACTION_SEARCH	android.intent.action.SEARCH	搜索
ACTION_WEB_SEARCH	android.intent.action.WEB_SEARCH	Web搜索
ACTION_FACTORY_TEST	android.intent.action.FACTORY_TEST	工厂测试入口点

7、broadcast的预定义Action

AndroidManifest中的 receiver 标签中使用

常量名称	意义
ACTION_TIME_TICK	系统时间每过一分钟发出的广播
ACTION_TIME_CHANGED	系统时间通过设置发生了变化
ACTION_TIMEZONE_CHANGED	时区改变
ACTION_BOOT_COMPLETED	系统启动完毕
ACTION_PACKAGE_ADDED	新的应用程序apk包安装完毕
ACTION_PACKAGE_CHANGED	现有应用程序apk包改变
ACTION_PACKAGE_REMOVED	现有应用程序apk包被删除
ACTION_UID_REMOVED	用户id被删除

category

8、category的匹配规则

- 1. category 是一个字符串，也有系统预定义和自定义的
- 2. intent有多个category，这些category都必须在intent-filter中找到一个相同的category

3. intent 中没有 category 时, Intent仍然可以匹配成功(系统会自动加上“android.intent.category.DEFAULT”, 也需要我们在intent-filter中指定, 这样才能接收 隐式调用)

9、category的预定义常量

常量名称	意义
CATEGORY_BROWSABLE	目标Activity能通过在网页浏览器中点击链接而激活（比如，点击浏览器中的图片链接）
CATEGORY_GADGET	表示目标Activity可以被内嵌到其他Activity当中
CATEGORY_HOME	目标Activity是HOME Activity，即手机开机启动后显示的Activity，或按下HOME键后显示的Activity
CATEGORY_LAUNCHER	表示目标Activity是应用程序中最优先被执行的Activity
CATEGORY_PREFERENCE	表示目标Activity是一个偏爱设置的Activity

data

10、data的组成

由两部分组成：MineType和URI

11、data的匹配规则

1. 规则与 action 类似：如果定义了 data ， 则 Intent 中必须定义可匹配的 data

```
<data
  android:mimeType="string"
  android:scheme="string"
  android:host="string"
  android:port="string"
  android:path="string"
  android:pathPattern="string"
  android:pathPrefix="string"/>
```

12、data的组成部分

字段	意义
MineType	MineType指的是媒体类型：例如imgage/jpeg, auto/mpeg4和viedo/*等，可以表示图片、文本、视频等不同的媒体格式
URI 可配置更多信息，类似于url。	
Scheme	URI的模式，如http、file、content。如果没有指定Scheme.那么整个URI无效。默认值为content 和 file。
Host	URI的主机名，如 www.baidu.com
Port	URI中的端口号，如80

字段	意义
path	用来匹配完整的路径，如： http://example.com/blog/abc.html ，这里将 path 设置为 /blog/abc.html 才能够进行匹配
pathPrefix	用来匹配路径的开头部分，拿上面的 Uri 来说，这里将 pathPrefix 设置为 /blog 就能进行匹配了
pathPattern	用表达式来匹配整个路径

13、匹配符号的讲解

1. * 用来匹配0次或更多，例如： a 可以匹配 a、aa、aaa
 2. . 可以匹配任意字符。例如： . 可以匹配 a、b、c等等
 3. 例如： .*html 可以匹配 “abhtml”、“chtml”，“html”，“sdf.html”

14、data的过滤实例:

```
<data android:mimeType="image/*" />

intent.setDataAndType(Uri.parse("file://abc"), "image/png");

1. Intent 中 mimeType 必须为 image/* 才能匹配
2. data 中没有指定 URI，因此默认为 content和file ——如果使用 Uri.parse("http://abc") 就会报错
3.必须调用 setDataAndType() 才能同时设置两个值，如果分开使用 setData()和setType() 会导致前者的值被清除

<data android:mimeType="video/mpeg" android:scheme="http" .../>
<data android:mimeType="audio/mpeg" android:scheme="http" .../>

intent.setDataAndType(Uri.parse("http://abc"), "video/mpeg");
or
intent.setDataAndType(Uri.parse("http://abc"), "audio/mpeg");

<data android:scheme="file" android:host="www.baidu.com"/>
等效于
<data android:scheme="file"/>
<data android:host="www.baidu.com"/>
```

实例

15、匹配scheme为http并且以 .pdf 结尾的路径，让别的程序打开网络pdf时，用户能选择我们的程序

```
<intent-filter>
  <action android:name="android.intent.action.VIEW"/>
  <category android:name="android.intent.category.DEFAULT"/>
  <data
    android:scheme="http"
    android:pathPattern=".*\\.pdf">
  </data>
</intent-filter>
```

16、别人分享信息，如何让我们的应用出现在选择框中？

```
<intent-filter>
    <action android:name="android.intent.action.SEND" />
    <category android:name="android.intent.category.DEFAULT" />
    <data mimeType="*/*" />
</intent-filter>
```

17、Service 和 BroadcastReceiver 中的 Intent-filter 同理，系统建议对于Service使用显式方式调用

18、隐式调用 Activity 不存在时的处理办法

1. Intent 的 resolveActivity()：找不到匹配 Activity 会返回 null
2. PackageManager 的 resolveActivity() ——返回最佳匹配的 Activity 信息
3. PackageManager 的 queryIntentActivity() 会返回所有成功匹配的 Activity 信息
4. queryIntentActivity() 的第二个参数需要使用 MATCH_DEFAULT_ONLY 标志位去仅仅匹配在 category 中声明了 android.intent.category.DEFAULT 的 Activity (这样只要有匹配的Activity,就会保证 startActivity 一定成功。因为没有指明 DEFAULT 的 Activity 是不能接受 隐式Intent 的)
5. 对于 Service 和 BroadcastReceiver，PackageManager 提供了类似方法去获取成功匹配的组件信息

```
<action android:name="android.intent.action.MAIN" />
<category android:name="android.intent.category.LAUNCHER">
```

共同作用表示为一个 入口Activity 并且会出现在系统的应用列表中

参考资料

1. [android-Scheme与网页跳转原生的三种方式](#)
2. [\[Intent,Action,大全](#)
3. [android intent和intent action大全](#)