

如何将消息发送给 Whatsapp 联系人

Whatsapp 官网上没有找到在 Android 上进行消息发送相关的信息，但是有一个 iOS 相关的帖子 <https://www.whatsapp.com/faq/iphone/23559013>，原以为用它的 URL 在 Android 上也是可以使用的，结果试了下不行，看来错误地把 URL 当作 URI 了。

基础功能实现

后来根据 Android Intent 和 Intent Filters 的官方文档说明（如果英文不太好，参考中文的翻译文档，链接为下面的第二个地址）：

<http://developer.android.com/guide/components/intents-filters.html>

http://wiki.eoe.cn/page/Intents_and_Intent_Filters.html#comment

就想着查看了下 Whatsapp 的 Manifest 配置文件，看看能不能找出点蛛丝马迹，结果发现有如下的 Activity 信息

```
<activity
    android:name=".ContactPicker"
    android:configChanges="0x00000FB0"
    >
    <intent-filter
        >
        <action
            android:name="android.intent.action.PICK"
            >
        </action>
        <category
            android:name="android.intent.category.DEFAULT"
            >
        </category>
        <category
            android:name="com.whatsapp"
            >
        </category>
    </intent-filter>
    <intent-filter
        >
        <action
            android:name="android.intent.action.CREATE_SHORTCUT"
            >
        </action>
    </intent-filter>
    <intent-filter
        >
```

```
<action
    android:name="android.intent.action.SEND"
  >
</action>
<category
    android:name="android.intent.category.DEFAULT"
  >
</category>
<data
    android:mimeType="audio/*"
  >
</data>
<data
    android:mimeType="video/*"
  >
</data>
<data
    android:mimeType="image/*"
  >
</data>
<data
    android:mimeType="text/plain"
  >
</data>
<data
    android:mimeType="text/x-vcard"
  >
</data>
</intent-filter>
<intent-filter
  >
  <action
    android:name="android.intent.action.SEND_MULTIPLE"
  >
  </action>
  <category
    android:name="android.intent.category.DEFAULT"
  >
  </category>
  <data
    android:mimeType="audio/*"
  >
  </data>
  <data
```

```

        android:mimeType="video/*"
    >
</data>
<data
    android:mimeType="image/*"
    >
</data>
</intent-filter>
</activity>

```

至于是不是这个 Activity，简单写个测试就验证了，既然找到了这个对象，怎么将信息发送进去，继续 google 下 “com.whatsapp” 和 “com.whatsapp.ContactPicker” 字符串，结果就发现如下一些热帖：

<http://stackoverflow.com/questions/6394173/androidpick-action-in-intent>

<http://stackoverflow.com/questions/19081654/send-to-specific-contact-whatsapp>

<http://stackoverflow.com/questions/15462874/sending-message-through-whatsapp>

从中发现如果想要发送文字信息给 Whatsapp 进行 Whatsapp 内消息的发送，可以利用的 intent 中两个缺省的字段 Intent.EXTRA_TEXT 和 Intent.EXTRA_SUBJECT，于是就按照 “android.intent.action.SEND” 这个 Intent Filter 做下如下的代码尝试。

```

public static boolean sendWhatsApp(Context ctx, String text){
    boolean sendOk = false;
    if(checkApkExist(ctx, "com.whatsapp")){
        Intent vIt = new Intent("android.intent.action.SEND");
        vIt.setPackage("com.whatsapp");
        vIt.setType("text/plain");
        if(!Util.IsNullOrEmpty(text)){
            vIt.putExtra(Intent.EXTRA_TEXT, "This is a simple test");
            vIt.putExtra(Intent.EXTRA_SUBJECT, "Subject");
        }
        ctx.startActivity(vIt);
        sendOk = true;
    }
    return sendOk;
}

private static boolean checkApkExist(Context ctx, String packageName) {
    if (packageName == null || "".equals(packageName))
        return false;
    try {
        ApplicationInfo info = ctx.getPackageManager().getApplicationInfo(packageName,
PackageManager.GET_UNINSTALLED_PACKAGES);
        if(info != null){
            return true;
        }else{
            return false;
        }
    }
}

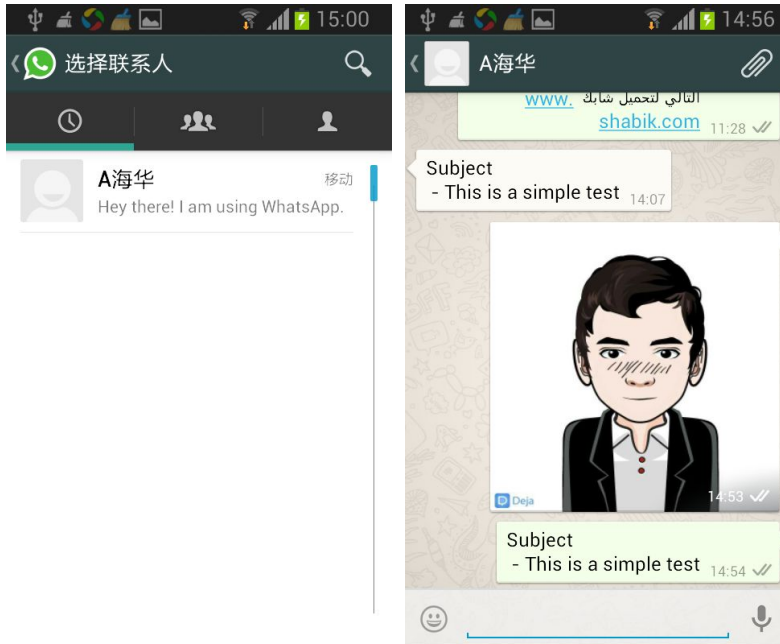
```

```

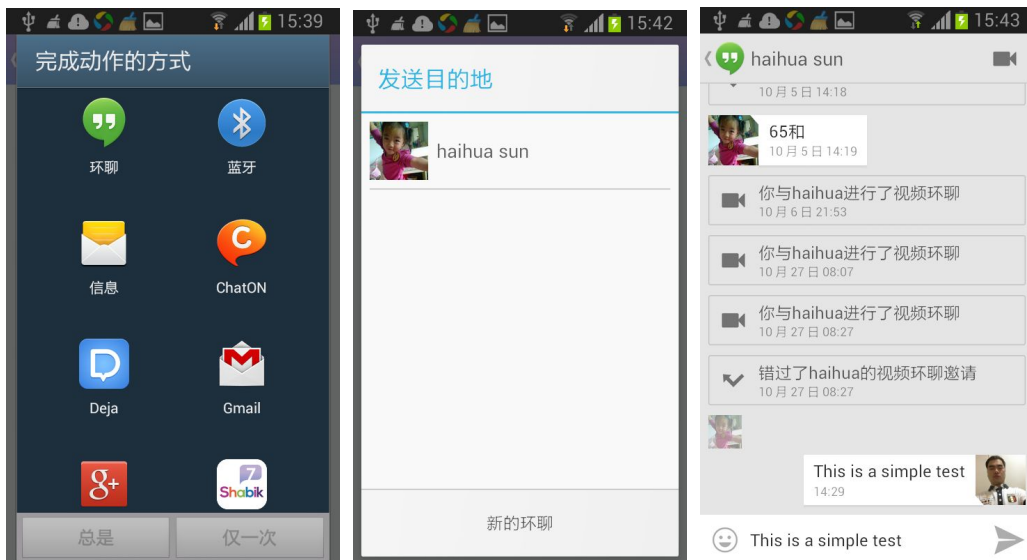
    }
} catch (NameNotFoundException e){
    return false;
}
}
}

```

上述的 checkApkExist 方法是用于判断当前是否有安装相应包程序的，发送效果如下：



假如上述代码中，去除 `vIt.setPackage("com.whatsapp");`这段代码之后，就会出现一个选择框的效果，这个就和很多程序中做分享的效果一致了，下面选择环聊做了简单测试。



假如要发图片的话也可以通过分享 Deja 头像的方式，来实现，下面将代码简单整理了一下，就先省略了具体的判断代码：

```

public static void shareImageToWhatsapp(Activity ctx, Bitmap shareBitmap, String subject,
String text) {
    Intent intent = new Intent(Intent.ACTION_SEND);
    intent.setPackage("com.whatsapp");

```

```

        intent.setType("image/*");
        Uri
            imageUri
            =
Uri.parse(MediaStore.Images.Media.insertImage(ctx.getContentResolver(), shareBitmap, null,
null));
        intent.putExtra(Intent.EXTRA_SUBJECT, subject);
        intent.putExtra(Intent.EXTRA_TEXT, text);
        intent.putExtra(Intent.EXTRA_STREAM, imageUri);
        ctx.startActivity(Intent.createChooser(intent, ctx.getTitle()));
    }

```

扩展深入

以前对 Intent 和 Intent Filters 可以说是一知半解的，对于官方的提供的上面的那个 API 文档，也没有吃透，不过经过这次的机会，对于 Intent 和 Intent Filter 可以说有了一个比较进一步的认识。

Intent 分为显式和隐式两种，简而言之，显式就是已经明显告知了所要调用的组件名称，而隐式是没有明确告知组件名称的调用方式，所以说**显式调用时虽然也是通过 Intent，但是显式调用不进行 Intent Filter 条件过滤**；隐式调用组件时，就要通过组建在 Manifest 中设置的 Intent Filter 来进行匹配了。

显式调用有两种，一种是程序内部明确知道是具体的 Activity，然后类似下述的直接调用：

```
startActivity(new Intent(this, ShareTargetActivity.class));
```

这种调用显然是能够通过 lib 或源码 import 到现有文件的方式。另外一种显式调用就是通过 Intent 的 setComponent 方法来显示调用，比如我们已经知道 Whatsapp 的组件名了，我们就可以通过以下方式调用：

```

public static boolean sendWhatsApp(Context ctx, String text){
    boolean sendOk = false;
    if(checkApkExist(ctx, "com.whatsapp")){
        Intent vIt = new Intent();
        ComponentName comp = new ComponentName( "com.whatsapp",
"com.whatsapp.ContactPicker");
        vIt.setComponent(comp);
        if(!Util.IsNullOrEmpty(text)){
            vIt.putExtra(Intent.EXTRA_TEXT, text);
            vIt.putExtra(Intent.EXTRA_SUBJECT, "Subject\n");
        }
        ctx.startActivity(vIt);
        sendOk = true;
    }
    return sendOk;
}

```

显式 Intent 时，压根就不用设置任何跟 Intent Filter 相关的参数，如果写了反而证明是不明真相的画蛇添足。

隐式 Intent 的调用就必须查看 Manifest 里面的配置了，而且根据 API 文档的说明，隐式 Intent 在进行过滤时，**action、category 和 data 是必须进行匹配的，至于 extra 和 flag 选项只是负责传参数，不作为过滤条件。**而一个 Intent Filter 只有一个 action，category 则是可以叠加的，至于 data 还是看 API 文档吧，不做过多展看了，通常我们都是忽略掉 category，在这里正好给我们提供了一个 category 的例子，因为 Whatsapp 里面的第一个 Intent Filter，即如下内容

```
<intent-filter
  >
  <action
    android:name="android.intent.action.PICK"
  >
</action>
<category
  android:name="android.intent.category.DEFAULT"
  >
</category>
<category
  android:name="com.whatsapp"
  >
</category>
</intent-filter>
```

我们可以通过下述方法进行隐式调用，就可以通过这个 Filter 的过滤，将参数正确传给 ContactPicker 这个 Activity，以下给出简单代码：

```
public static boolean sendWhatsApp(Context ctx, String text) {
    boolean sendOk = false;
    if (checkApkExist(ctx, "com.whatsapp")) {
        Intent vIt = new Intent("android.intent.action.PICK");
        vIt.addCategory("com.whatsapp");
        // vIt.setType("text/plain");
        if (!Util.IsNullOrEmpty(text)) {
            vIt.putExtra(Intent.EXTRA_TEXT, text);
            vIt.putExtra(Intent.EXTRA_SUBJECT, "Subject\n");
        }
        ctx.startActivity(vIt);
        sendOk = true;
    }
    return sendOk;
}
```

注意在上述这个例子中，注释掉的 setType 代码是必须注释的，否则程序将因为找不到 Activity 组件而报 android.content.ActivityNotFoundException 的异常错误。

同样我们也不能通过以下方式进行调用

```
public static boolean sendWhatsApp(Context ctx, String text){
    boolean sendOk = false;
    if(checkApkExist(ctx, "com.whatsapp")){
```

```
Intent vIt = new Intent("android.intent.action.SEND_MULTIPLE");
vIt.setPackage("com.whatsapp");
vIt.setType("text/plain");
if(!Util.IsNullOrEmpty(text)){
    vIt.putExtra(Intent.EXTRA_TEXT, text);
    vIt.putExtra(Intent.EXTRA_SUBJECT, "Subject");
}
ctx.startActivity(vIt);
sendOk = true;
}
return sendOk;
}
```

对照下 Manifest，应该就一目了然了，总体上，通过这个例子，应该能对 Intent 和 Intent Filter 有了更进一步的认识了。