Intent的中文意思是“意图，目的”的意思，可以理解为不同组件之间通信的“媒介”或者“信使”。

目标组件一般要通过Intent来声明自己的条件，一般通过组件中的<intent-filter>元素来过滤。

Intent在由以下几个部分组成：动作（action），数据（data），分类（Category），类型（Type），组件（Component），和扩展信息（Extra）。

Intent在寻找目标组件的时候有两种方法：第一，通过组件名称直接指定；第二，通过Intent Filter过滤指定

Intent启动不同组件的方法

组件名称  方法名称

Activity  startActivity()  startActivityForResult()

Service  startService()  bindService()

Broadcasts  sendBroadcast()  sendOrderedBroadcast()  sendStickyBroadcast()

常见的Activity Action Intent常量

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

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

常见的BroadcastIntent Action常量  BroadcastIntent

Action字符串常量  描述

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

Intent的Action和Data属性匹配

Action属性  Data属性  说明

ACTION\_VIEW  content://contacts/people/1  显示id为1的联系人信息

ACTION\_DIAL  content://contacts/people/1  将id为1的联系人电话号码显示在拨号界面中

ACITON\_VIEW  tel:123  显示电话为123的联系人信息

ACTION\_VIEW  http://www.google.com  在浏览器中浏览该网站

ACTION\_VIEW  file://sdcard/mymusic.mp3  播放MP3

ACTION\_VIEW  geo:39.2456,116.3523  显示地图

常见的Category常量

Category字符串常量  描述

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

常见的Extra常量

Extra键值字符串常量  描述

EXTRA\_BCC  装有邮件密送地址的字符串数组

EXTRA\_CC  装有邮件抄送地址的字符串数组

EXTRA\_EMAIL  装有邮件发送地址的字符串数组

EXTRA\_INTENT  使用ACTION\_PICK\_ACTIVITY动作时装有Intent选项的键

EXTRA\_KEY\_EVENT  触发该Intent的案件的KeyEvent对象

EXTRA\_PHONE\_NUMBER  使用拨打电话相关的Action时，电话号码字符串的键，类型为String

EXTRA\_SHORTCUT\_ICON  使用ACTION\_CREATE\_SHORTCUT在HomeActivity创建快捷方式时，对快捷方式的描述信息。

其中ICON和ICON\_RESOURCE描述的是快捷方式的图标，类型分别为Bitmap和ShortcutIconResource。INTENT描述的是快捷方式相对应的Intent对象。NAME描述的是快捷方式的名字。

EXTRA\_SHORTCUT\_ICON\_RESOURCE  EXTRA\_SHORTCUT\_INTENT  EXTRA\_SHORTCUT\_NAME  EXTRA\_SUBJECT  描述信息主题的键

EXTRA\_TEXT  使用ACTION\_SEND动作时，用来描述要发送的文本信息，类型为CharSequence

EXTRA\_TITLE  使用ACTION\_CHOOSER动作时，描述对话框标题的键，类型为CharSequence

EXTRA\_UID  使用ACTION\_UID\_REMOVED动作时，描述删除的用户id的键，类型为int

Android.telephony包中的类

类名  描述

CellLocation  表示设备位置的抽象类

PhoneNumberFormattingTextWather  监视一个TextView控件，如果有电话号码输入，则用formatNumber()方法处理电话号码

PhoneNumberUtils  包含各种处理电话号码字符串的使用工具

PhoneStateListener  监视手机中电话状态变化的监听类

ServiceState  包含电话状态和相关的服务信息

TelephonyManager  提供对手机中电话服务信息的访问

与短信服务相关的类主要在包android.telephony.gsm中

类名  描述

GsmCellLocation  表示GSM手机的基站位置

SmsManager  管理各种短信操作

SmsMessage  表示具体的短信

1.Intent的用法：

(1)用Action跳转

1、使用Action跳转，如果有一个程序的AndroidManifest.xml中的某一个 Activity的IntentFilter段中 定义了包含了相同的Action那么这个Intent就与这个目标Action匹配。如果这个IntentFilter段中没有定义 Type,Category，那么这个 Activity就匹配了。但是如果手机中有两个以上的程序匹配，那么就会弹出一个对话可框来提示说明。

Action 的值在Android中有很多预定义，如果你想直接转到你自己定义的Intent接收者，你可以在接收者的IntentFilter 中加入一个自定义的Action值（同时要设定 Category值为"android.intent.category.DEFAULT"），在你的Intent中设定该值为Intent的 Action,就直接能跳转到你自己的Intent接收者中。因为这个Action在系统中是唯一的。

2,data/type，你可以用Uri 来做为data,比如Uri uri = Uri.parse(http://www.google.com);

Intent i = new Intent(Intent.ACTION\_VIEW,uri);手机的Intent分发过程中，会根据http://www.google.com 的scheme判断出数据类型type 。手机的Brower则能匹配它，在Brower的Manifest.xml中的IntenFilter中 首先有ACTION\_VIEW Action,也能处理http:的type，

3， 至于分类Category，一般不要去在Intent中设置它，如果你写Intent的接收者，就在Manifest.xml的Activity的 IntentFilter中包含android.category.DEFAULT,这样所有不设置 Category（Intent.addCategory(String c);）的Intent都会与这个Category匹配。

4,extras（附 加信息），是其它所有附加信息的集合。使用extras可以为组件提供扩展信息，比如，如果要执行“发送电子邮件”这个动 作，可以将电子邮件的标题、正文等保存在extras里，传给电子邮件发送组件。

（2）用类名跳转

    Intent负责对应用中一次操作的动作、动作涉及数据、附加数据进行描 述，Android则根据此Intent的描述， 负责找到对应的组件，将 Intent传递给调用的组件，并完成组件的调用。Intent在这里起着实现调用者与被调用者之间的解耦作用。Intent传递过程中，要找 到目标消费者（另一个Activity,IntentReceiver或Service），也就是Intent的响 应者。

Intent intent = new Intent();

intent.setClass(context, targetActivy.class);

//或者直接用 Intent intent = new Intent(context, targetActivity.class);

startActivity(intent);

不过注意用类名跳转，需要在AndroidManifest.xml中申明activity

<activity android:name="targetActivity"></activity>

2.几种Intent的用法

android 中intent是经常要用到的。不管是页面牵转，还是传递数据，或是调用外部程序，系统功能都要用到intent。在做了一些intent的例子之后，整理了一下intent，希望对大家有用。由于intent内容太多，不可能真的写全，难免会有遗落，以后我会随时更新。如果你们有疑问或新的intent内容，希望交流。

★intent大全：

1.从google搜索内容

Intent intent = new Intent();

intent.setAction(Intent.ACTION\_WEB\_SEARCH);

intent.putExtra(SearchManager.QUERY,"searchString")

startActivity(intent);

2.浏览网页

Uri uri = Uri.parse("http://www.google.com");

Intent it  = new Intent(Intent.ACTION\_VIEW,uri);

startActivity(it);

3.显示地图

Uri uri = Uri.parse("geo:38.899533,-77.036476");

Intent it = new Intent(Intent.Action\_VIEW,uri);

startActivity(it);

4.路径规划

Uri uri = Uri.parse("http://maps.google.com/maps?f=dsaddr=startLat%20startLng&daddr=endLat%20endLng&hl=en");

Intent it = new Intent(Intent.ACTION\_VIEW,URI);

startActivity(it);

5.拨打电话

Uri uri = Uri.parse("tel:xxxxxx");

Intent it = new Intent(Intent.ACTION\_DIAL, uri);

startActivity(it);

6.调用发短信的程序

Intent it = new Intent(Intent.ACTION\_VIEW);

it.putExtra("sms\_body", "The SMS text");

it.setType("vnd.android-dir/mms-sms");

startActivity(it);

7.发送短信

Uri uri = Uri.parse("smsto:0800000123");

Intent it = new Intent(Intent.ACTION\_SENDTO, uri);

it.putExtra("sms\_body", "The SMS text");

startActivity(it);

String body="this is sms demo";

Intent mmsintent = new Intent(Intent.ACTION\_SENDTO, Uri.fromParts("smsto", number, null));

mmsintent.putExtra(Messaging.KEY\_ACTION\_SENDTO\_MESSAGE\_BODY, body);

mmsintent.putExtra(Messaging.KEY\_ACTION\_SENDTO\_COMPOSE\_MODE, true);

mmsintent.putExtra(Messaging.KEY\_ACTION\_SENDTO\_EXIT\_ON\_SENT, true);

startActivity(mmsintent);

8.发送彩信

Uri uri = Uri.parse("content://media/external/images/media/23");

Intent it = new Intent(Intent.ACTION\_SEND);

it.putExtra("sms\_body", "some text");

it.putExtra(Intent.EXTRA\_STREAM, uri);

it.setType("image/png");

startActivity(it);

StringBuilder sb = new StringBuilder();

sb.append("file://");

sb.append(fd.getAbsoluteFile());

Intent intent = new Intent(Intent.ACTION\_SENDTO, Uri.fromParts("mmsto", number, null));

// Below extra datas are all optional.

intent.putExtra(Messaging.KEY\_ACTION\_SENDTO\_MESSAGE\_SUBJECT, subject);

intent.putExtra(Messaging.KEY\_ACTION\_SENDTO\_MESSAGE\_BODY, body);

intent.putExtra(Messaging.KEY\_ACTION\_SENDTO\_CONTENT\_URI, sb.toString());

intent.putExtra(Messaging.KEY\_ACTION\_SENDTO\_COMPOSE\_MODE, composeMode);

intent.putExtra(Messaging.KEY\_ACTION\_SENDTO\_EXIT\_ON\_SENT, exitOnSent);

startActivity(intent);

9.发送Email

Uri uri = Uri.parse("mailto:xxx@abc.com");

Intent it = new Intent(Intent.ACTION\_SENDTO, uri);

startActivity(it);

Intent it = new Intent(Intent.ACTION\_SEND);

it.putExtra(Intent.EXTRA\_EMAIL, "me@abc.com");

it.putExtra(Intent.EXTRA\_TEXT, "The email body text");

it.setType("text/plain");

startActivity(Intent.createChooser(it, "Choose Email Client"));

Intent it=new Intent(Intent.ACTION\_SEND);

String[] tos={"me@abc.com"};

String[] ccs={"you@abc.com"};

it.putExtra(Intent.EXTRA\_EMAIL, tos);

it.putExtra(Intent.EXTRA\_CC, ccs);

it.putExtra(Intent.EXTRA\_TEXT, "The email body text");

it.putExtra(Intent.EXTRA\_SUBJECT, "The email subject text");

it.setType("message/rfc822");

startActivity(Intent.createChooser(it, "Choose Email Client"));

Intent it = new Intent(Intent.ACTION\_SEND);

it.putExtra(Intent.EXTRA\_SUBJECT, "The email subject text");

it.putExtra(Intent.EXTRA\_STREAM, "file:///sdcard/mysong.mp3");

sendIntent.setType("audio/mp3");

startActivity(Intent.createChooser(it, "Choose Email Client"));

10.播放多媒体

Intent it = new Intent(Intent.ACTION\_VIEW);

Uri uri = Uri.parse("file:///sdcard/song.mp3");

it.setDataAndType(uri, "audio/mp3");

startActivity(it);

Uri uri = Uri.withAppendedPath(MediaStore.Audio.Media.INTERNAL\_CONTENT\_URI, "1");

Intent it = new Intent(Intent.ACTION\_VIEW, uri);

startActivity(it);

11.uninstall apk

Uri uri = Uri.fromParts("package", strPackageName, null);

Intent it = new Intent(Intent.ACTION\_DELETE, uri);

startActivity(it);

12.install apk

Uri installUri = Uri.fromParts("package", "xxx", null);

returnIt = new Intent(Intent.ACTION\_PACKAGE\_ADDED, installUri);

13. 打开照相机

<1>Intent i = new Intent(Intent.ACTION\_CAMERA\_BUTTON, null);

    this.sendBroadcast(i);

<2>long dateTaken = System.currentTimeMillis();

    String name = createName(dateTaken) + ".jpg";

    fileName = folder + name;

    ContentValues values = new ContentValues();

    values.put(Images.Media.TITLE, fileName);

    values.put("\_data", fileName);

    values.put(Images.Media.PICASA\_ID, fileName);

    values.put(Images.Media.DISPLAY\_NAME, fileName);

    values.put(Images.Media.DESCRIPTION, fileName);

    values.put(Images.ImageColumns.BUCKET\_DISPLAY\_NAME, fileName);

    Uri photoUri = getContentResolver().insert(

            MediaStore.Images.Media.EXTERNAL\_CONTENT\_URI, values);

    Intent inttPhoto = new Intent(MediaStore.ACTION\_IMAGE\_CAPTURE);

    inttPhoto.putExtra(MediaStore.EXTRA\_OUTPUT, photoUri);

    startActivityForResult(inttPhoto, 10);

14.从gallery选取图片

Intent i = new Intent();

i.setType("image/\*");

i.setAction(Intent.ACTION\_GET\_CONTENT);

startActivityForResult(i, 11);

15. 打开录音机

Intent mi = new Intent(Media.RECORD\_SOUND\_ACTION);

startActivity(mi);

16.显示应用详细列表

Uri uri = Uri.parse("market://details?id=app\_id");

Intent it = new Intent(Intent.ACTION\_VIEW, uri);

startActivity(it);

//where app\_id is the application ID, find the ID

//by clicking on your application on Market home

//page, and notice the ID from the address bar

刚才找app id未果，结果发现用package name也可以

Uri uri = Uri.parse("market://details?id=<packagename>");

这个简单多了

17寻找应用

Uri uri = Uri.parse("market://search?q=pname:pkg\_name");

Intent it = new Intent(Intent.ACTION\_VIEW, uri);

startActivity(it);

//where pkg\_name is the full package path for an application

18打开联系人列表

    <1>

    Intent i = new Intent();

    i.setAction(Intent.ACTION\_GET\_CONTENT);

    i.setType("vnd.android.cursor.item/phone");

    startActivityForResult(i, REQUEST\_TEXT);

    <2>

    Uri uri = Uri.parse("content://contacts/people");

    Intent it = new Intent(Intent.ACTION\_PICK, uri);

    startActivityForResult(it, REQUEST\_TEXT);

19 打开另一程序

Intent i = new Intent();

ComponentName cn = new ComponentName("com.yellowbook.android2",

        "com.yellowbook.android2.AndroidSearch");

i.setComponent(cn);

i.setAction("android.intent.action.MAIN");

startActivityForResult(i, RESULT\_OK);

20.调用系统编辑添加联系人（高版本SDK有效）：

Intent it = newIntent(Intent.ACTION\_INSERT\_OR\_EDIT);

it.setType("vnd.android.cursor.item/contact");

//it.setType(Contacts.CONTENT\_ITEM\_TYPE);

it.putExtra("name","myName");

it.putExtra(android.provider.Contacts.Intents.Insert.COMPANY,  "organization");

it.putExtra(android.provider.Contacts.Intents.Insert.EMAIL,"email");

it.putExtra(android.provider.Contacts.Intents.Insert.PHONE,"homePhone");

it.putExtra(android.provider.Contacts.Intents.Insert.SECONDARY\_PHONE,

               "mobilePhone");

it.putExtra(  android.provider.Contacts.Intents.Insert.TERTIARY\_PHONE,

               "workPhone");

it.putExtra(android.provider.Contacts.Intents.Insert.JOB\_TITLE,"title");

startActivity(it);

21.调用系统编辑添加联系人（全有效）：

Intent intent = newIntent(Intent.ACTION\_INSERT\_OR\_EDIT);

intent.setType(People.CONTENT\_ITEM\_TYPE);

intent.putExtra(Contacts.Intents.Insert.NAME, "My Name");

intent.putExtra(Contacts.Intents.Insert.PHONE, "+1234567890");

intent.putExtra(Contacts.Intents.Insert.PHONE\_TYPE,Contacts.PhonesColumns.TYPE\_MOBILE);

intent.putExtra(Contacts.Intents.Insert.EMAIL, "com@com.com");

intent.putExtra(Contacts.Intents.Insert.EMAIL\_TYPE,                    Contacts.ContactMethodsColumns.TYPE\_WORK);

startActivity(intent);