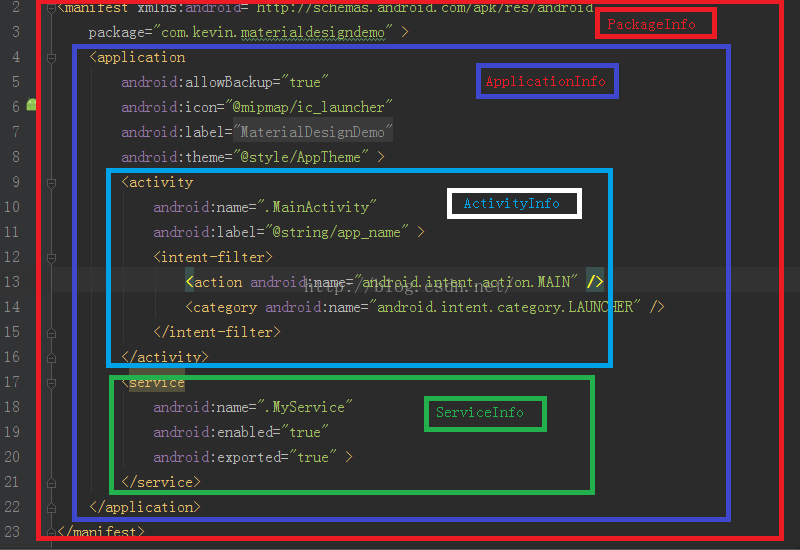
此文章已于 9:31:39 2018/3/30 重新发布到 Song\_UP

Android获取应用程序信息——PackageManager的基本使用

类别 Anddroid

在我们实际开发中，我们常需要获取应用的一些信息来使用，比如获取应用程序基本信息、获取permission是否注册等。这些信息获取都借助于Android给我们提供的PackageManager进行获取，这里我们就总结下PackageManager的基本用法。

首先看下我们常见的Manifest配置图：  
  
  
通过上图可以看出，Android给我们的每个节点提供了对应的信息存储实体类，所以这些类中应该存储了对应类别节点的一些信息，我们只需要通过PackageManager获取对应的组件信息即可。

# 一、常见类介绍

## 1、PackageManager包管理类

获取设备上的安装的应用程序包，一般通过getPackageManager();获取对象。      
常用方法：  
（1）、public abstract PackageInfo getPackageInfo(String packageName, int flags)根据包名获取对应的PackageInfo，注意，此处的flags标签：

* GET\_ACTIVITIES
* GET\_GIDS
* GET\_CONFIGURATIONS
* GET\_INSTRUMENTATION
* GET\_PERMISSIONS
* GET\_PROVIDERS
* GET\_RECEIVERS
* GET\_SERVICES
* GET\_SIGNATURES
* GET\_UNINSTALLED\_PACKAGES

（2）、public abstract int[] getPackageGids(String packageName)，根据包名获取group-ids  
  
（3）、public abstract PermissionInfo getPermissionInfo(String name, int flags)，根据包名和指定的flags获取指定的授权信息  
  
（4）、public abstract List<PermissionGroupInfo> getAllPermissionGroups(int flags);获取所以PermissGroup集合  
  
（5）、public abstract PermissionGroupInfo getPermissionGroupInfo(String name,  
    int flags)根据指定的Group名称获取PermissionGroupInfo对象。  
  
（6）、public abstract ApplicationInfo getApplicationInfo(String packageName,  
            int flags)，根据指定的包名获取ApplicationInfo信息。  
  
（7）、public abstract ActivityInfo getActivityInfo(ComponentName component,  
            int flags)，根据指定的组件，获取ActivityInfo信息  
  
（8）、public abstract ServiceInfo getServiceInfo(ComponentName component,  
            int flags)，根据指定组件获取ServiceInfo  
  
（9）、public abstract ProviderInfo getProviderInfo(ComponentName component,  
            int flags)，根据指定组件名称获取ProviderInfo信息  
  
（10）、public abstract List<PackageInfo> getInstalledPackages(int flags);获取所有安装的PackagInfo信息  
  
（11）、public abstract List<PackageInfo> getPackagesHoldingPermissions(  
            String[] permissions, int flags);获取具有特定权限的PackagInfo  
  
（12）、public abstract List<ApplicationInfo> getInstalledApplications(int flags);获取安装的ApplicationInfo信息  
  
（13）、public abstract boolean addPermission(PermissionInfo info);添加权限  
  
（14）、public abstract void removePermission(String name);移除权限

## 2、PackageInfo用于描述mainfest中所有描述信息。

## 常见字段：    （1）、public String packageName;包名     （2）、public String[] splitNames;    （3）、public int versionCode;版本号　　　　　 （4）、public String versionName;版本名称     （5）、public ApplicationInfo applicationInfo;     （6）、public long firstInstallTime;第一次安装时间    （7）、public long lastUpdateTime;上次更新时间      （8）、public ActivityInfo[] activities;所有的Activity信息      （9）、public ActivityInfo[] receivers; 所有的广播接收者     （10）、public ServiceInfo[] services;所有的服务信息      （11）、public ProviderInfo[] providers;获取ContentProvide      （12）、public PermissionInfo[] permissions;所有的权限信息 2、ApplicationInfo描述应用程序的基本信息

可以通过getApplicationInfo()方法获取该对象。     
常见字段：          
（1）、public String taskAffinity;        
（2）、public String permission;       
（3）、public String processName;         
（4）、public String className;        
（5）、public int descriptionRes;         
（6）、public static final int FLAG\_SYSTEM = 1<<0;系统应用      
（7）、public static final int FLAG\_EXTERNAL\_STORAGE = 1<<18;安装在外部存储卡       
（8）、 public static final int FLAG\_LARGE\_HEAP = 1<<20;支持大内存        
  
其它诸如ActivityInfo、ServiceInfo都类似，他们都是继承自ComponentInfo类，唯一可以需要注意的就是各个类中的FLAG标签的含义，Flag比较多，需要花较多时间记忆。

# 二、代码段测试

1、ApplicationInfo类测试：获取Application节点信息

**[java]** [view plain](https://blog.csdn.net/mr_dsw/article/details/51340254) [copy](https://blog.csdn.net/mr_dsw/article/details/51340254)

1. ApplicationInfo applicationInfo = getApplicationInfo();
2. Log.d("dsw",applicationInfo.className + "\n" +
3. applicationInfo.dataDir+"\n" +
4. applicationInfo.permission + "\n"
5. + applicationInfo.packageName + "\n"
6. + applicationInfo.processName + "\n"
7. + applicationInfo.taskAffinity + "\n"
8. );

输出结果：

**[java]** [view plain](https://blog.csdn.net/mr_dsw/article/details/51340254) [copy](https://blog.csdn.net/mr_dsw/article/details/51340254)

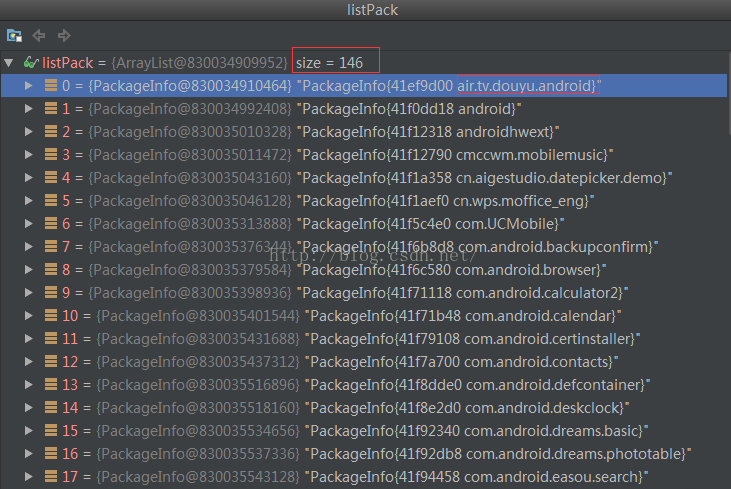
1. 05-07 18:38:19.368 8255-8255/com.kevin.materialdesigndemo D/dsw: **null**
2. 05-07 18:38:19.368 8255-8255/com.kevin.materialdesigndemo D/dsw: /data/data/com.kevin.materialdesigndemo
3. 05-07 18:38:19.368 8255-8255/com.kevin.materialdesigndemo D/dsw: **null**
4. 05-07 18:38:19.368 8255-8255/com.kevin.materialdesigndemo D/dsw: com.kevin.materialdesigndemo
5. 05-07 18:38:19.368 8255-8255/com.kevin.materialdesigndemo D/dsw: com.kevin.materialdesigndemo
6. 05-07 18:38:19.368 8255-8255/com.kevin.materialdesigndemo D/dsw: com.kevin.materialdesigndemo

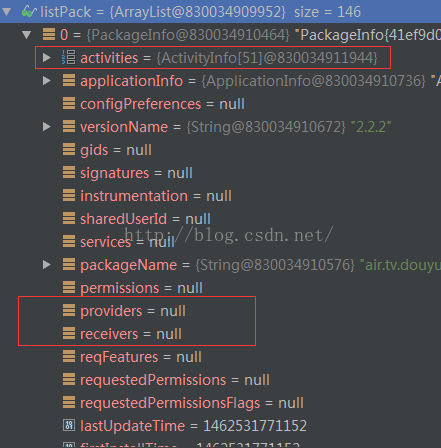
**[java]** [view plain](https://blog.csdn.net/mr_dsw/article/details/51340254) [copy](https://blog.csdn.net/mr_dsw/article/details/51340254)



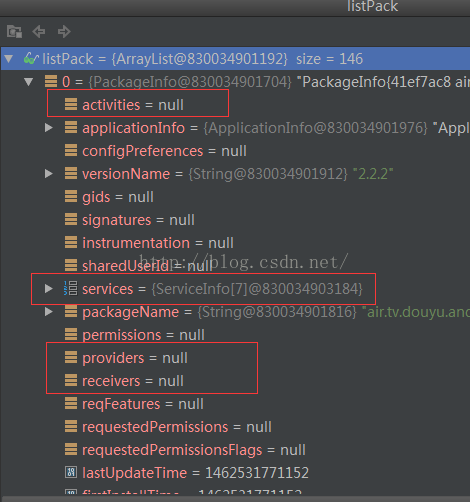
## 2、获取所有安装的Packages。

（1）、Flag = GET\_ACTIVITIES  
List<PackageInfo> listPack = packageManager.getInstalledPackages(PackageManager.GET\_ACTIVITIES);  
下面是我们获取的结果：





从上面的图中我们可以看到，已经获取了安装在手机上的所有Package的信息，获得PackageInfo集合，然后编辑PackageInfo可以获取每个Package里的详情信息。由于我们设置的是Activity，所以在对应的PackageInfo中activities中有值。（2）、Flag = GET\_SERVICES



此时，我们的services中就有值，而activities中为null。

## 3、获取指定应用的PackageInfo。

这个需要我们通过getPackageInfo(packName,flag)方法进行获取，其中packName是对应manifest中的package属性值。

**[java]** [view plain](https://blog.csdn.net/mr_dsw/article/details/51340254) [copy](https://blog.csdn.net/mr_dsw/article/details/51340254)

1. PackageManager packageManager = getPackageManager();
2. PackageInfo packageInfo = **null**;
3. **try** {
4. packageInfo = packageManager.getPackageInfo("com.kevin.materialdesigndemo",
5. PackageManager.GET\_ACTIVITIES);
6. Log.d("dsw", packageInfo.packageName + "\n"
7. + packageInfo.versionName + "\n"
8. );
9. } **catch** (PackageManager.NameNotFoundException e) {
10. e.printStackTrace();
11. }

这样就可以获取一个应用的PackageInfo信息。

## 4、获取应用程序中的permission。

**[java]** [view plain](https://blog.csdn.net/mr_dsw/article/details/51340254) [copy](https://blog.csdn.net/mr_dsw/article/details/51340254)

1. PackageManager packageManager = getPackageManager();
2. **try** {
3. PermissionInfo permissionInfo = packageManager.getPermissionInfo("android.permission.INTERNET",
4. PermissionInfo.PROTECTION\_NORMAL);
5. List<PermissionGroupInfo> list = packageManager.getAllPermissionGroups(PackageManager.PERMISSION\_GRANTED);
6. Log.d("dsw",
7. permissionInfo.group + "\n"
8. + permissionInfo.packageName + "\n"
9. + permissionInfo.name + "\n"
10. + permissionInfo.flags + "\n"
11. );
12. } **catch** (PackageManager.NameNotFoundException e) {
13. e.printStackTrace();
14. }

输出结果：

**[java]** [view plain](https://blog.csdn.net/mr_dsw/article/details/51340254) [copy](https://blog.csdn.net/mr_dsw/article/details/51340254)

1. 05-07 21:25:03.984 10392-10392/com.kevin.materialdesigndemo D/dsw: android.permission-group.NETWORK
2. 05-07 21:25:03.984 10392-10392/com.kevin.materialdesigndemo D/dsw: android
3. 05-07 21:25:03.984 10392-10392/com.kevin.materialdesigndemo D/dsw: android.permission.INTERNET
4. 05-07 21:25:03.984 10392-10392/com.kevin.materialdesigndemo D/dsw: 0

## 5、获取应用程序中的执行AcitivtyInfo。

**[java]** [view plain](https://blog.csdn.net/mr_dsw/article/details/51340254) [copy](https://blog.csdn.net/mr_dsw/article/details/51340254)

1. PackageManager packageManager = getPackageManager();
2. ComponentName componentName = **new** ComponentName("com.kevin.materialdesigndemo","com.dsw.demo.MainActivity");
3. **try** {
4. ActivityInfo activityInfo = packageManager.getActivityInfo(componentName, PackageManager.GET\_ACTIVITIES);
5. Log.d("dsw:activityInfo",activityInfo.name + "\n"
6. + activityInfo.packageName +"\n"
7. + activityInfo.targetActivity
8. );
9. } **catch** (PackageManager.NameNotFoundException e) {
10. e.printStackTrace();
11. }

注意这里组装的ComponentName对象，第一个值对应manifest下的package值，第二个是完整的包路径值的Activity的名称。  
结果：

**[java]** [view plain](https://blog.csdn.net/mr_dsw/article/details/51340254) [copy](https://blog.csdn.net/mr_dsw/article/details/51340254)

1. 05-07 21:33:16.524 16730-16730/com.kevin.materialdesigndemo D/dsw:activityInfo: com.dsw.demo.MainActivity
2. 05-07 21:33:16.524 16730-16730/com.kevin.materialdesigndemo D/dsw:activityInfo: com.kevin.materialdesigndemo
3. 05-07 21:33:16.524 16730-16730/com.kevin.materialdesigndemo D/dsw:activityInfo: **null**

补充：获取ServiceInfo、ProviderInfo的方法和这个相同，都是通过指定ComponentName组件进行获取。