

AppSearch

制作者 Doxygen 1.8.16

1 AllApp	1
2 继承关系索引	3
2.1 类继承关系	3
3 类索引	5
3.1 类列表	5
4 类说明	7
4.1 com.tcl.allapp.AllApp类 参考	7
4.1.1 详细描述	9
4.1.2 成员函数说明	9
4.1.2.1 getApps()	9
4.1.2.2 getCurentPkgInfo()	9
4.1.2.3 initAppClass()	9
4.1.2.4 onDestroy()	10
4.1.2.5 onStart()	10
4.1.2.6 stopTimer()	10
4.1.2.7 updateAppInfo()	10
4.1.2.8 updateAppInfoRunning()	10
4.1.2.9 updateAppList()	11
4.1.3 类成员变量说明	11
4.1.3.1 appCpu	11
4.1.3.2 appInfoList	11
4.1.3.3 appListening	11
4.1.3.4 appPermission	11
4.1.3.5 appStorage	11
4.1.3.6 lstApp	12
4.1.3.7 lstPermission	12
4.1.3.8 storageHandler	12
4.1.3.9 Tag	12
4.1.3.10 task	12
4.1.3.11 timer	12
4.1.3.12 timerHandler	12
4.1.3.13 TimerMs	12
4.1.3.14 timerRunning	13
4.1.3.15 txtCpu	13
4.1.3.16 txtMemory	13
4.1.3.17 txtStorage	13
4.2 com.tcl.allapp.AppCpu类 参考	13
4.2.1 详细描述	14
4.2.2 成员函数说明	14
4.2.2.1 convertToInt()	14

4.2.2.2 formatByte()	15
4.2.2.3 getCpuByCmd()	15
4.2.2.4 getPid()	15
4.2.2.5 updateCpu()	16
4.2.2.6 updateView()	17
4.3 com.tcl.allapp.AllApp.AppInfo类 参考	17
4.3.1 详细描述	18
4.3.2 类成员变量说明	18
4.3.2.1 icon	18
4.3.2.2 name	18
4.3.2.3 pkgName	18
4.3.2.4 sysFlg	18
4.3.2.5 version	18
4.4 com.tcl.allapp.AllApp.AppInfoAdapter类 参考	19
4.4.1 详细描述	19
4.4.2 构造及析构造函数说明	19
4.4.2.1 AppInfoAdapter()	19
4.4.3 成员函数说明	20
4.4.3.1 getItem()	20
4.4.3.2 getView()	20
4.4.4 类成员变量说明	20
4.4.4.1 appInfoList	21
4.5 com.tcl.allapp.AppListening类 参考	21
4.5.1 详细描述	21
4.6 com.tcl.allapp.AppPermission类 参考	21
4.6.1 详细描述	22
4.6.2 成员函数说明	22
4.6.2.1 getPermission()	22
4.6.2.2 updatePermission() [1/2]	22
4.6.2.3 updatePermission() [2/2]	23
4.7 com.tcl.allapp.AppStorage类 参考	23
4.7.1 详细描述	24
4.7.2 成员函数说明	24
4.7.2.1 updateStorage()	24
4.8 com.tcl.allapp.AppCpu.CpuInfo类 参考	24
4.8.1 详细描述	25
4.9 com.tcl.allapp.AllApp.StorageHandler类 参考	25
4.9.1 详细描述	25
4.9.2 成员函数说明	25
4.9.2.1 handleMessage()	25
4.10 com.tcl.allapp.AppStorage.StorageInfo类 参考	26
4.10.1 详细描述	26

4.10.2 构造及析构函数说明	26
4.10.2.1 StorageInfo() [1/2]	27
4.10.2.2 StorageInfo() [2/2]	27
4.10.3 成员函数说明	27
4.10.3.1 toString()	27
4.10.4 类成员变量说明	27
4.10.4.1 CREATOR	27
4.11 com.tcl.allapp.AppStorage.ThreadStorage类 参考	28
4.11.1 详细描述	28
4.12 com.tcl.allapp.AllApp.TimerHandler类 参考	28
4.12.1 详细描述	29
Index	31

Chapter 1

AllApp

AllApp

AllApp是一个应用助手，用于显示应用相关信息，主要有以下信息：

- 1.需要显示所有TV安装的应用名称，图标。
- 2.选中应用后，显示通过列表方式显示应用使用权限情况。
- 3.选中应用后，显示应用内存，CPU，存储，进程号，应用包名等信息进行显示。

作为一个信息展示软件，我们将所有信息在一个页面显示。

Chapter 2

继承关系索引

2.1 类继承关系

此继承关系列表按字典顺序粗略的排序:

com.tcl.allapp.AppCpu	13
com.tcl.allapp.AllApp.AppInfo	17
com.tcl.allapp.AppPermission	21
com.tcl.allapp.AppStorage	23
com.tcl.allapp.AppCpu.CpuInfo	24
Thread	
com.tcl.allapp.AppStorage.ThreadStorage	28
AppCompatActivity	
com.tcl.allapp.AllApp	7
BaseAdapter	
com.tcl.allapp.AllApp.AppInfoAdapter	19
BroadcastReceiver	
com.tcl.allapp.AppListening	21
Handler	
com.tcl.allapp.AllApp.StorageHandler	25
com.tcl.allapp.AllApp.TimerHandler	28
Parcelable	
com.tcl.allapp.AppStorage.StorageInfo	26

Chapter 3

类索引

3.1 类列表

这里列出了所有类、结构、联合以及接口定义等，并附带简要说明：

com.tcl.allapp.AllApp	
AllApp主类	7
com.tcl.allapp.AppCpu	
应用Cpu和Pid信息类	13
com.tcl.allapp.AllApp.AppInfo	
应用信息结构，用于保存所有应用的基本信息。	17
com.tcl.allapp.AllApp.AppInfoAdapter	
APP应用列表适配器	19
com.tcl.allapp.AppListening	
监听系统应用的安装与卸载	21
com.tcl.allapp.AppPermission	
应用权限信息类	21
com.tcl.allapp.AppStorage	
获取应用存储信息	23
com.tcl.allapp.AppCpu.CpuInfo	
Cpu信息结构	24
com.tcl.allapp.AllApp.StorageHandler	
静态Handler类，与Storage子线程传递消息	25
com.tcl.allapp.AppStorage.StorageInfo	
应用存储信息结构	26
com.tcl.allapp.AppStorage.ThreadStorage	
获取应用存储的线程	28
com.tcl.allapp.AllApp.TimerHandler	
静态Handler类	28

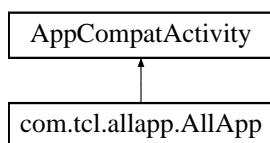
Chapter 4

类说明

4.1 com.tcl.allapp.AllApp类 参考

AllApp主类

类 com.tcl.allapp.AllApp 继承关系图:



类

- class `AppInfo`
应用信息结构，用于保存所有应用的基本信息。
- class `AppInfoAdapter`
*APP*应用列表适配器
- class `StorageHandler`
静态 *Handler*类，与 *Storage*子线程传递消息
- class `TimerHandler`
静态 *Handler*类

Public 成员函数

- void `initActivity ()`
组件初始化
- void `startTimer ()`
开启定时任务 每次`timer.cancel()`或者`task.cancel()`后，*Timer*和*Task*均消耗掉，需要再次*New*。
- void `stopTimer ()`
关闭定时器
- void `getApps (List< AppInfo > list)`
获取所有应用列表信息

- void `updateAppList` ()
更新所有应用列表
- void `updateAppInfo` (PackageInfo pkgInfo)
更新指定应用的所有信息
- void `updateAppInfoRunning` (PackageInfo pkgInfo)
更新指定运行应用的所有信息
- PackageInfo `getCurentPkgInfo` ()
获取当前光标所在 *Item* 的 *PackageInfo* 对象

静态 **Public** 成员函数

- static boolean `hasUsageStatsPermission` (final Context context)
查看是否有 *Usage Stats* 权限

Public 属性

- final String `Tag` = "AllApp"
- final int `TimerMs` = 2000

Protected 成员函数

- void `onCreate` (Bundle savedInstanceState)
Activity onCreate 函数
- void `onStart` ()
Activity onStart 函数
- void `onDestroy` ()
Activity onDestroy 函数

Private 成员函数

- void `initAppClass` ()
应用数据类初始

Private 属性

- List< `AppInfo` > `appInfoList`
- Timer `timer`
- boolean `timerRunning`
- TimerTask `task`
- `TimerHandler` `timerHandler`
- `StorageHandler` `storageHandler`
- `AppListening` `appListening`
- `AppPermission` `appPermission`
- `AppCpu` `appCpu`
- `AppStorage` `appStorage`
- ListView `lstApp`
- ListView `lstPermission`
- TextView `txtCpu`
- TextView `txtMemory`
- TextView `txtStorage`

4.1.1 详细描述

AllApp主类

AllApp是软件主类，控制软件整体逻辑，显示TV所有应用列表信息，并控制指定应用信息的更新。

4.1.2 成员函数说明

4.1.2.1 getApp()

```
void com.tcl.allapp.AllApp.getApp (
    List< AppInfo > list )
```

获取所有应用列表信息

注意：获取应用列表前，会将原有的所有数据清空

参数

<i>list</i>	将获取的应用列表信息保存在list中
-------------	--------------------

4.1.2.2 getCurentPkgInfo()

```
PackageInfo com.tcl.allapp.AllApp.getCurentPkgInfo ( )
```

获取当前光标所在Item的PackageInfo对象

返回

返回当前光标所在Item的PackageInfo对象

返回值

<i>null</i>	当前没有选择的Item
-------------	-------------

4.1.2.3 initAppClass()

```
void com.tcl.allapp.AllApp.initAppClass ( ) [private]
```

应用数据类初始

4.1.2.4 onDestroy()

```
void com.tcl.allapp.AllApp.onDestroy ( ) [protected]
```

Activity onDestroy函数

4.1.2.5 onStart()

```
void com.tcl.allapp.AllApp.onStart ( ) [protected]
```

Activity onStart函数

4.1.2.6 stopTimer()

```
void com.tcl.allapp.AllApp.stopTimer ( )
```

关闭定时器

4.1.2.7 updateAppInfo()

```
void com.tcl.allapp.AllApp.updateAppInfo (
    PackageInfo pkgInfo )
```

更新指定应用的所有信息

参数

<i>pkgInfo</i>	指定应用
----------------	------

4.1.2.8 updateAppInfoRunning()

```
void com.tcl.allapp.AllApp.updateAppInfoRunning (
    PackageInfo pkgInfo )
```

更新指定运行应用的所有信息

参数

<i>pkgInfo</i>	指定应用
----------------	------

4.1.2.9 updateAppList()

```
void com.tcl.allapp.AllApp.updateAppList ( )
```

更新所有应用列表

4.1.3 类成员变量说明

4.1.3.1 appCpu

```
AppCpu com.tcl.allapp.AllApp.appCpu [private]
```

应用Cpu、Pid和内存信息实例

4.1.3.2 appInfoList

```
List<AppInfo> com.tcl.allapp.AllApp.appInfoList [private]
```

所有应用信息列表

4.1.3.3 appListening

```
AppListening com.tcl.allapp.AllApp.appListening [private]
```

应用安装卸载监听对象

4.1.3.4 appPermission

```
AppPermission com.tcl.allapp.AllApp.appPermission [private]
```

应用信息实例

4.1.3.5 appStorage

```
AppStorage com.tcl.allapp.AllApp.appStorage [private]
```

应用存储信息实例

4.1.3.6 IstApp

```
ListView com.tcl.allapp.AllApp.lstApp [private]
```

应用列表显示

4.1.3.7 IstPermission

```
ListView com.tcl.allapp.AllApp.lstPermission [private]
```

应用权限显示

4.1.3.8 storageHandler

```
StorageHandler com.tcl.allapp.AllApp.storageHandler [private]
```

Storage消息传递Handler

4.1.3.9 Tag

```
final String com.tcl.allapp.AllApp.Tag = "AllApp"
```

软件Tag，用于logcat使用

4.1.3.10 task

```
TimerTask com.tcl.allapp.AllApp.task [private]
```

定时器任务

4.1.3.11 timer

```
Timer com.tcl.allapp.AllApp.timer [private]
```

定时器，用于定时查询信息

4.1.3.12 timerHandler

```
TimerHandler com.tcl.allapp.AllApp.timerHandler [private]
```

任务句柄

4.1.3.13 TimerMs

```
final int com.tcl.allapp.AllApp.TimerMs = 2000
```

定时器间隔，单位：milliseconds

4.1.3.14 timerRunning

`boolean com.tcl.allapp.AllApp.timerRunning [private]`

Timer是否正在运行

4.1.3.15 txtCpu

`TextView com.tcl.allapp.AllApp.txtCpu [private]`

应用Cpu、Pid信息显示

4.1.3.16 txtMemory

`TextView com.tcl.allapp.AllApp.txtMemory [private]`

应用内存信息显示

4.1.3.17 txtStorage

`TextView com.tcl.allapp.AllApp.txtStorage [private]`

应用存储信息显示

该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AllApp.java

4.2 com.tcl.allapp.AppCpu类 参考

应用Cpu和Pid信息类

类

- class [CpuInfo](#)
*Cpu*信息结构

Public 成员函数

- boolean [updateCpu](#) (Context context, PackageInfo pkgInfo, TextView cpuView, TextView memView)
更新 *CpuInfo* 实例对象信息

Private 成员函数

- boolean `getPid` (Context context, PackageInfo pkgInfo, `CpuInfo` cpuInfo)
 获取应用是否运行的状态，如果正在运行将将*pid*赋给*CpuInfo*实例
- void `updateView` (Context context, TextView cpuView, TextView memView, `CpuInfo` cpuInfo)
 将CPU 内存对象中的信息展示到界面上

静态 Private 成员函数

- static String `getCpuByCmd` (String pid)
 获取CPU占用率
- static int `convertToInt` (Object value, int defaultValue)
 格式转化工具，将*Object*对象转化为整数
- static String `formatByte` (int data)
 格式转化工具，将整数自动转化为对应内存单位大小的字符串

4.2.1 详细描述

应用Cpu和Pid信息类

4.2.2 成员函数说明

4.2.2.1 `convertToInt()`

```
static int com.tcl.allapp.AppCpu.convertToInt (  
    Object value,  
    int defaultValue ) [static], [private]
```

格式转化工具，将Object对象转化为整数

参数

<i>value</i>	任意对象
<i>defaultValue</i>	转化失败时返回默认值

返回

int

返回值

NONE	
------	--

4.2.2.2 formatByte()

```
static String com.tcl.allapp.AppCpu.formatByte (  
    int data ) [static], [private]
```

格式化工具，将整数自动转化为对应内存单位大小的字符串

参数

<i>data</i>	整数
-------------	----

返回

int

返回值

<i>NONE</i>	
-------------	--

4.2.2.3 getCpuByCmd()

```
static String com.tcl.allapp.AppCpu.getCpuByCmd (  
    String pid ) [static], [private]
```

获取CPU占用率

参数

<i>pid</i>	需要获取应用的进程号
------------	------------

返回

返回值

返回当前 <i>PID</i> 进程的 <i>CPU</i> 占用率	
------------------------------------	--

4.2.2.4 getPid()

```
boolean com.tcl.allapp.AppCpu.getPid (  

```

```
Context context,  
PackageInfo pkgInfo,  
CpuInfo cpuInfo ) [private]
```

获取应用是否运行的状态，如果正在运行将将pid赋给CpuInfo实例

参数

<i>context</i>	上下文环境
<i>pkgInfo</i>	应用信息实例对象
<i>cpuInfo</i>	应用Cpu和Pid信息类实例

返回

返回值

<i>None</i>	
-------------	--

4.2.2.5 updateCpu()

```
boolean com.tcl.allapp.AppCpu.updateCpu (  
    Context context,  
    PackageInfo pkgInfo,  
    TextView cpuView,  
    TextView memView )
```

更新CpuInfo实例对象信息

参数

<i>context</i>	上下文环境
<i>pkgInfo</i>	应用信息实例对象
<i>cpuView</i>	CPU和PID信息显示界面
<i>memView</i>	内存信息显示页面

返回

返回值

<i>None</i>	
-------------	--

4.2.2.6 updateView()

```
void com.tcl.allapp.AppCpu.updateView (
    Context context,
    TextView cpuView,
    TextView memView,
    CpuInfo cpuInfo ) [private]
```

将CPU 内存对象中的信息展示到界面上

参数

context	上下文环境
cpuView	CPU和PID信息显示界面
memView	内存信息显示界面
cpuInfo	CPU内存类对象

返回

返回值

None	
------	--

该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AppCpu.java

4.3 com.tcl.allapp.AllApp.AppInfo类 参考

应用信息结构，用于保存所有应用的基本信息。

Public 成员函数

- String toString ()
生成String函数

Public 属性

- String name
- String pkgName
- String version
- Drawable icon
- boolean sysFlg

4.3.1 详细描述

应用信息结构，用于保存所有应用的基本信息。

4.3.2 类成员变量说明

4.3.2.1 icon

```
Drawable com.tcl.allapp.AllApp.AppInfo.icon
```

应用图标

4.3.2.2 name

```
String com.tcl.allapp.AllApp.AppInfo.name
```

应用名称

4.3.2.3 pkgName

```
String com.tcl.allapp.AllApp.AppInfo.pkgName
```

应用包名称

4.3.2.4 sysFlg

```
boolean com.tcl.allapp.AllApp.AppInfo.sysFlg
```

是否是系统应用

4.3.2.5 version

```
String com.tcl.allapp.AllApp.AppInfo.version
```

应用版本

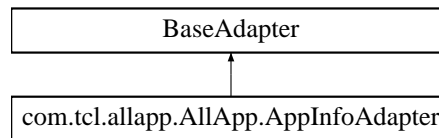
该类的文档由以下文件生成：

- AllApp/app/src/main/java/com/tcl/allapp/AllApp.java

4.4 com.tcl.allapp.AllApp.AppInfoAdapter类 参考

APP应用列表适配器

类 com.tcl.allapp.AllApp.AppInfoAdapter 继承关系图:



Public 成员函数

- [AppInfoAdapter](#) (List< [AppInfo](#) > list)
构造函数 适配器实例化时，需要一个应用信息列表实例，适配器本身不会产生应用信息列表数据。
- int [getCount](#) ()
获取 *Item* 的数量
- Object [getItem](#) (int pos)
获取 *Item* 对象
- long [getItemId](#) (int pos)
获取 *Item* 下标
- View [getView](#) (int pos, @Nullable View convertView, ViewGroup viewParent)
设置并返回一个 *Item* 视图

Private 属性

- List< [AppInfo](#) > [appInfoList](#)

4.4.1 详细描述

APP应用列表适配器

自行实现ListView的Item显示内容。

4.4.2 构造及析构函数说明

4.4.2.1 AppInfoAdapter()

```
com.tcl.allapp.AllApp.AppInfoAdapter.AppInfoAdapter (  
    List< AppInfo > list )
```

构造函数 适配器实例化时，需要一个应用信息列表实例，适配器本身不会产生应用信息列表数据。

参数

<i>list</i>	应用信息列表
-------------	--------

4.4.3 成员函数说明

4.4.3.1 getItem()

```
Object com.tcl.allapp.AllApp.AppInfoAdapter.getItem (
    int pos )
```

获取Item对象

参数

<i>pos</i>	Item的下标
------------	---------

4.4.3.2 getView()

```
View com.tcl.allapp.AllApp.AppInfoAdapter.getView (
    int pos,
    @Nullable View convertView,
    ViewGroup viewParent )
```

设置并返回一个Item视图

参数

<i>pos</i>	Item所在位置
<i>convertView</i>	Item的视图，若为Null，需要先初始化
<i>viewParent</i>	Item的分组

返回

返回Item视图

4.4.4 类成员变量说明

4.4.4.1 appInfoList

```
List<AppInfo> com.tcl.allapp.AllApp.AppInfoAdapter.appInfoList [private]
```

需要显示的应用信息列表

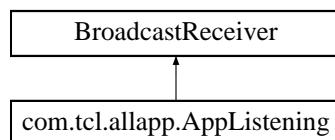
该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AllApp.java

4.5 com.tcl.allapp.AppListening类 参考

监听系统应用的安装与卸载

类 com.tcl.allapp.AppListening 继承关系图:



Public 成员函数

- **AppListening** ([AllApp](#) aa)
- void **onReceive** (Context context, Intent intent)
应用安装和卸载的监听函数

Private 属性

- final [AllApp](#) aa

4.5.1 详细描述

监听系统应用的安装与卸载

该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AppListening.java

4.6 com.tcl.allapp.AppPermission类 参考

应用权限信息类

Public 成员函数

- boolean [updatePermission](#) (Context context, PackageInfo pkgInfo, ListView listView)
更新显示应用权限信息
- List< String > [updatePermission](#) (Context context, PackageInfo pkgInfo)
返回一个 *list*

Private 成员函数

- List [getPermission](#) (Context context, String packageName)

4.6.1 详细描述

应用权限信息类

4.6.2 成员函数说明

4.6.2.1 getPermission()

```
List com.tcl.allapp.AppPermission.getPermission (  
    Context context,  
    String packageName ) [private]
```

List集合返回某APP拥有的权限信息

参数

<i>context</i>	Context 实例
<i>packageName</i>	APP包名

返回

返回该APP的权限list

4.6.2.2 updatePermission() [1/2]

```
List<String> com.tcl.allapp.AppPermission.updatePermission (  
    Context context,  
    PackageInfo pkgInfo )
```

返回一个list

参数

<i>context</i>	context实例
<i>pkgInfo</i>	package信息

返回

list

4.6.2.3 updatePermission() [2/2]

```
boolean com.tcl.allapp.AppPermission.updatePermission (
    Context context,
    PackageInfo pkgInfo,
    ListView listView )
```

更新显示应用权限信息

参数

<i>context</i>	Context实例
<i>pkgInfo</i>	package信息
<i>listView</i>	listview组件

返回

成功更新则返回是， 否则返回否

该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AppPermission.java

4.7 com.tcl.allapp.AppStorage类 参考

获取应用存储信息

类

- class [StorageInfo](#)
应用存储信息结构
- class [ThreadStorage](#)
获取应用存储的线程

Public 成员函数

- boolean [updateStorage](#) (Context context, PackageInfo pkgInfo, AllApp.StorageHandler handler)
获取应用存储信息

4.7.1 详细描述

获取应用存储信息

4.7.2 成员函数说明

4.7.2.1 updateStorage()

```
boolean com.tcl.allapp.AppStorage.updateStorage (  
    Context context,  
    PackageInfo pkgInfo,  
    AllApp.StorageHandler handler )
```

获取应用存储信息

参数

<i>context</i>	上下文环境
<i>pkgInfo</i>	应用包信息
<i>handler</i>	用于子线程传递消息的Handler

返回

返回值

<i>None</i>	
-------------	--

该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AppStorage.java

4.8 com.tcl.allapp.AppCpu.CpuInfo类 参考

Cpu信息结构

4.8.1 详细描述

Cpu信息结构

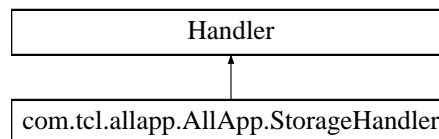
该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AppCpu.java

4.9 com.tcl.allapp.AllApp.StorageHandler类 参考

静态Handler类，与Storage子线程传递消息

类 com.tcl.allapp.AllApp.StorageHandler 继承关系图:



Public 成员函数

- **StorageHandler** ([AllApp](#) aa)
- void **handleMessage** (Message msg)
处理Storage消息

Private 属性

- final WeakReference< [AllApp](#) > **waa**

4.9.1 详细描述

静态Handler类，与Storage子线程传递消息

4.9.2 成员函数说明

4.9.2.1 handleMessage()

```
void com.tcl.allapp.AllApp.StorageHandler.handleMessage (  
    Message msg )
```

处理Storage消息

参数

<i>msg</i>	来自AppStorage的消息
------------	-----------------

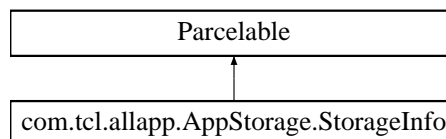
该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AllApp.java

4.10 com.tcl.allapp.AppStorage.StorageInfo类 参考

应用存储信息结构

类 com.tcl.allapp.AppStorage.StorageInfo 继承关系图:



Public 成员函数

- [StorageInfo](#) ()
构造函数
- [StorageInfo](#) (Parcel in)
*Parcel*构造函数
- int **describeContents** ()
- void [writeToParcel](#) (Parcel dest, int flags)
写入*Parcel*, 以便*Bundle*传递数据
- String [toString](#) (Context context)
生成应用存储信息

Public 属性

- final Parcelable.Creator< [StorageInfo](#) > **CREATOR**

4.10.1 详细描述

应用存储信息结构

4.10.2 构造及析构函数说明

4.10.2.1 StorageInfo() [1/2]

```
com.tcl.allapp.AppStorage.StorageInfo.StorageInfo ( )
```

构造函数

4.10.2.2 StorageInfo() [2/2]

```
com.tcl.allapp.AppStorage.StorageInfo.StorageInfo (
    Parcel in )
```

Parcel构造函数

参数

<i>in</i>	Parcel对象
-----------	----------

4.10.3 成员函数说明

4.10.3.1 toString()

```
String com.tcl.allapp.AppStorage.StorageInfo.toString (
    Context context )
```

生成应用存储信息

返回

返回存储信息

4.10.4 类成员变量说明

4.10.4.1 CREATOR

```
final Parcelable.Creator<StorageInfo> com.tcl.allapp.AppStorage.StorageInfo.CREATOR
```

初始值:

```
= new Parcelable.Creator<StorageInfo>() {
    public StorageInfo createFromParcel(Parcel in) {
        return new StorageInfo(in);
    }
    public StorageInfo[] newArray(int size) {
        return new StorageInfo[size];
    }
}
```

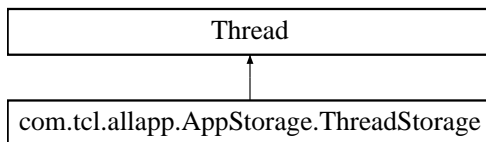
该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AppStorage.java

4.11 com.tcl.allapp.AppStorage.ThreadStorage类 参考

获取应用存储的线程

类 com.tcl.allapp.AppStorage.ThreadStorage 继承关系图:



Public 成员函数

- void **run** ()

Public 属性

- Context **context**
- PackageInfo **pkgInfo**
- AllApp.StorageHandler **handler**

4.11.1 详细描述

获取应用存储的线程

获取存储信息是耗时操作，需要在线程中实现

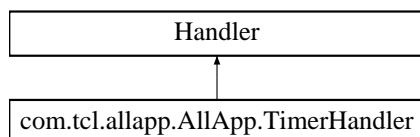
该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AppStorage.java

4.12 com.tcl.allapp.AllApp.TimerHandler类 参考

静态Handler类

类 com.tcl.allapp.AllApp.TimerHandler 继承关系图:



Public 成员函数

- **TimerHandler** ([AllApp](#) aa)
- void **handleMessage** (Message msg)

Private 属性

- final WeakReference< [AllApp](#) > **waa**

4.12.1 详细描述

静态Handler类

该类的文档由以下文件生成:

- AllApp/app/src/main/java/com/tcl/allapp/AllApp.java

Index

- appCpu
 - com.tcl.allapp.AllApp, 11
- AppInfoAdapter
 - com.tcl.allapp.AllApp.AppInfoAdapter, 19
- appInfoList
 - com.tcl.allapp.AllApp, 11
 - com.tcl.allapp.AllApp.AppInfoAdapter, 20
- appListening
 - com.tcl.allapp.AllApp, 11
- appPermission
 - com.tcl.allapp.AllApp, 11
- appStorage
 - com.tcl.allapp.AllApp, 11
- com.tcl.allapp.AllApp, 7
 - appCpu, 11
 - appInfoList, 11
 - appListening, 11
 - appPermission, 11
 - appStorage, 11
 - getApps, 9
 - getCurentPkgInfo, 9
 - initAppClass, 9
 - lstApp, 11
 - lstPermission, 12
 - onDestroy, 9
 - onStart, 10
 - stopTimer, 10
 - storageHandler, 12
 - Tag, 12
 - task, 12
 - timer, 12
 - timerHandler, 12
 - TimerMs, 12
 - timerRunning, 12
 - txtCpu, 13
 - txtMemory, 13
 - txtStorage, 13
 - updateAppInfo, 10
 - updateAppInfoRunning, 10
 - updateAppList, 11
- com.tcl.allapp.AllApp.AppInfo, 17
 - icon, 18
 - name, 18
 - pkgName, 18
 - sysFig, 18
 - version, 18
- com.tcl.allapp.AllApp.AppInfoAdapter, 19
 - AppInfoAdapter, 19
 - appInfoList, 20
 - getItem, 20
 - getView, 20
- com.tcl.allapp.AllApp.StorageHandler, 25
 - handleMessage, 25
- com.tcl.allapp.AllApp.TimerHandler, 28
- com.tcl.allapp.AppCpu, 13
 - convertToInt, 14
 - formatByte, 15
 - getCpuByCmd, 15
 - getPid, 15
 - updateCpu, 16
 - updateView, 16
- com.tcl.allapp.AppCpu.CpuInfo, 24
- com.tcl.allapp.AppListening, 21
- com.tcl.allapp.AppPermission, 21
 - getPermission, 22
 - updatePermission, 22, 23
- com.tcl.allapp.AppStorage, 23
 - updateStorage, 24
- com.tcl.allapp.AppStorage.StorageInfo, 26
 - CREATOR, 27
 - StorageInfo, 26, 27
 - toString, 27
- com.tcl.allapp.AppStorage.ThreadStorage, 28
- convertToInt
 - com.tcl.allapp.AppCpu, 14
- CREATOR
 - com.tcl.allapp.AppStorage.StorageInfo, 27
- formatByte
 - com.tcl.allapp.AppCpu, 15
- getApps
 - com.tcl.allapp.AllApp, 9
- getCpuByCmd
 - com.tcl.allapp.AppCpu, 15
- getCurentPkgInfo
 - com.tcl.allapp.AllApp, 9
- getItem
 - com.tcl.allapp.AllApp.AppInfoAdapter, 20
- getPermission
 - com.tcl.allapp.AppPermission, 22
- getPid
 - com.tcl.allapp.AppCpu, 15
- getView
 - com.tcl.allapp.AllApp.AppInfoAdapter, 20
- handleMessage
 - com.tcl.allapp.AllApp.StorageHandler, 25
- icon

- com.tcl.allapp.AllApp.AppInfo, [18](#)
- initAppClass
 - com.tcl.allapp.AllApp, [9](#)
- lstApp
 - com.tcl.allapp.AllApp, [11](#)
- lstPermission
 - com.tcl.allapp.AllApp, [12](#)
- name
 - com.tcl.allapp.AllApp.AppInfo, [18](#)
- onDestroy
 - com.tcl.allapp.AllApp, [9](#)
- onStart
 - com.tcl.allapp.AllApp, [10](#)
- pkgName
 - com.tcl.allapp.AllApp.AppInfo, [18](#)
- stopTimer
 - com.tcl.allapp.AllApp, [10](#)
- storageHandler
 - com.tcl.allapp.AllApp, [12](#)
- StorageInfo
 - com.tcl.allapp.AppStorage.StorageInfo, [26](#), [27](#)
- sysFlg
 - com.tcl.allapp.AllApp.AppInfo, [18](#)
- Tag
 - com.tcl.allapp.AllApp, [12](#)
- task
 - com.tcl.allapp.AllApp, [12](#)
- timer
 - com.tcl.allapp.AllApp, [12](#)
- timerHandler
 - com.tcl.allapp.AllApp, [12](#)
- TimerMs
 - com.tcl.allapp.AllApp, [12](#)
- timerRunning
 - com.tcl.allapp.AllApp, [12](#)
- toString
 - com.tcl.allapp.AppStorage.StorageInfo, [27](#)
- txtCpu
 - com.tcl.allapp.AllApp, [13](#)
- txtMemory
 - com.tcl.allapp.AllApp, [13](#)
- txtStorage
 - com.tcl.allapp.AllApp, [13](#)
- updateAppInfo
 - com.tcl.allapp.AllApp, [10](#)
- updateAppInfoRunning
 - com.tcl.allapp.AllApp, [10](#)
- updateAppList
 - com.tcl.allapp.AllApp, [11](#)
- updateCpu
 - com.tcl.allapp.AppCpu, [16](#)
- updatePermission
 - com.tcl.allapp.AppPermission, [22](#), [23](#)
- updateStorage
 - com.tcl.allapp.AppStorage, [24](#)
- updateView
 - com.tcl.allapp.AppCpu, [16](#)
- version
 - com.tcl.allapp.AllApp.AppInfo, [18](#)