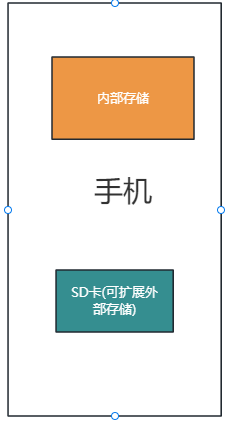
Android11文件存储适配分析

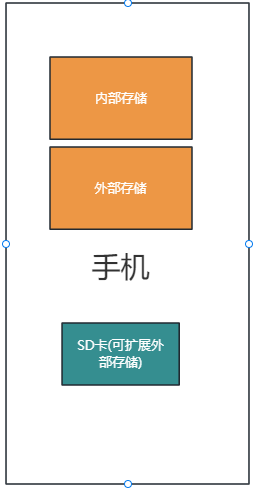
最近各大应用市场都宣布将采用targetSdkVersion>=30的等级要求作为应用上架的收录标准，并且将逐步清理低于targetSdkVersion 30的应用。这就意味着咱们的APP必须跟进升级，不然都不能正常上架。接到通知后，GMU终端组就开始进行了框架的升级适配工作，今天主要来聊聊关于文件存储的适配。

**一．手机存储**

首先，先介绍下Android系统中的存储区域划分。Android4.4以前，由于硬件发展受限，手机自身的存储空间有限，需要通过外置SD卡来扩展存储空间，如下图：

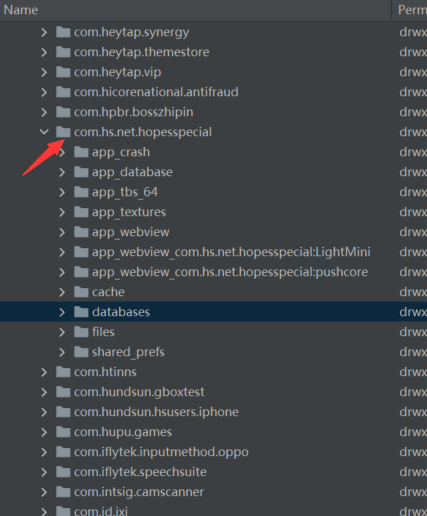


在Android 4.4 之后(含)，手机机身存储扩大了，如下图:手机的机身存储划分内部存储和外部存储。现在现在依然可以插入SD卡来扩充存储空间，这部分的存储空间称为扩展的外部存储空间。只不过现在机身存储都比较大，很少插入SD卡了，因此，插入SD卡的情况就不在此次的讨论范围内了。**我们所做的文件适配工作主要是针对手机内部存储和手机外部存储两部分来展开的**。

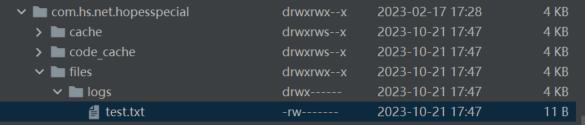


**二．内部存储和外部存储的适配**

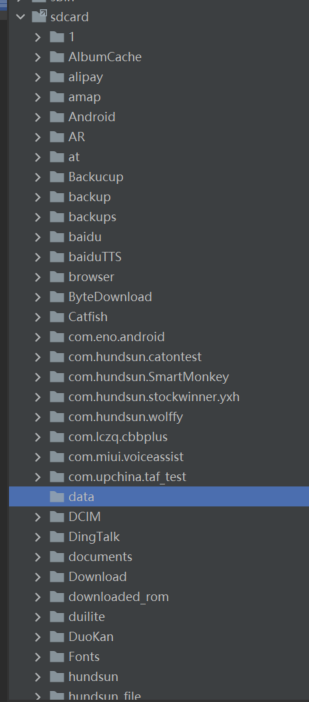
内部存储：Android的内部存储里给每个应用按照其包名各自划分了目录，假设App的包名为com.hs.net.hopesspecial,那么该文件在内部存储的目录为/data/data/com.hs.net.hopesspecial，这块内容可以用AndroidStudio侧边栏Device File Explore查看。如下图，这个块存储空间只允许App自己访问(除非有更高的权限，比如root),程序卸载后，该目录也会随之删除。



往内部存储空间读写数据是不需要读写权限的，直接使用File API进行读写即可。大致代码如下图，执行后我们可以看到data/data/ com.hs.net.hopesspecial/files/出现了logs文件夹，里面有test.txt文件，并且内容是hello world

手机外部存储:手机外部存储目录主要是指/sdcard里面的内容，对应的真正地址是/ storage/emulated/0/。



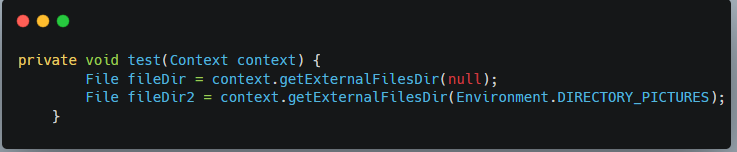
如上图所示，/sdcard/目录下的子目录看起来都比较眼熟,这些子目录分可以分为三部分：

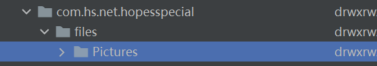
1. 共享存储空间

上图中的DCIM、Download、Music、Movies、Pictures、Ringtones 等这种系统为我们创建的文件夹就是共享存储空间。他们又分为两类: 1.媒体文件 2.文档和其他文件。

1. App外部存储-私有目录

Android/data/xxx(包名)/就是各个App的外部私有目录。读写这部分区域的数据和操作内部存储一样，不需要任何的权限，直接使用File API操作就行，如下图操作后，外部存储的的私有目录下就是生成对应的文件夹，然后可以使用文件流的形式进行读写文件。





1. 其他目录

除了上面提到的那些文件夹，剩下的都归类于其他目录，例如/sdcard/alipay、/sdcard/baidu等。

可以看到，针对内部存储和外部存储-私有目录，我们可以不用申请任何的权限并且直接使用File API进行读写文件的操作。这两部分的文件会跟随App卸载的时候一起清除掉，接下来重点讲讲共享存储空间和其他目录。

1.针对共享存储空间中的媒体文件，我们有两种方式访问，第一种是使用File API,这种方式在Android10上是不被允许的，但是google考虑到这样做很多三方库都无法正常使用，所以又在Android 11上重新放开了，虽然可以通过路径直接访问媒体文件，这些操作最终是被重定向到MediaStore API的，重定向过程可能会损耗一些性能，并且直接通过路径访问不一定比MediaStore API 访问快。 所以google还是建议非必要的话不要直接使用路径访问。这个我们可以根据实际情况来修改，如果工程中大量使用的File API,那我们从28升级到30的时候，这部分代码是可以不用修改的，可以减少适配工作量。第二种方式就是使用MediaStore。MediaStore中有Audio,Image,Video等内部类。这些内部类中记录着文件表的各个字段名，通过构造这些参数就可以插入相应的字段值以及获取对应的字段值。比如我们想要查询共享存储空间中的图片文件，可以这样来操作，如下图：



MediaStore.Images.Media.EXTERNAL\_CONTENT\_URI 意思是指定查询文件的类型是图片，并构造成Uri对象，Uri实现了Parcelable，能够在进程间传递。接收方(另一个进程收到后)，匹配Uri，解析出对应的字段，获取图片流，从而得到图片。另外App如果想访问其他应用共享的媒体文件，只需要申请READ\_EXTERNAL\_STORAGE权限即可。

2.针对共享存储空间中的文档和其他文件，我们同样也有两种方式来操作。第一种仍然是File API，直接构造路径进行访问。第二种则是通过SAF(Storage Access Framework),存储访问框架。相当于系统内置了文件选择器，通过它可以拿到想要访问的文件信息，代码大致使用如下：



3访问其他目录。通过上面我们可以看到三方应用App可以随便在/sdcard根目录下创建文件，并且App卸载后这些文件也不会删除，导致手机的容量越来越小，google为了整治这种乱象，在Android 11中不允许我们直接使用File API来操作其他目录，只能使用SAF.来操作。另外为了适配一些三方文件管理类的APP，google提供一个MANAGE\_EXTERNAL\_STORAGE的权限，允许你访问所有文件，使用ACTION\_MANAGE\_ALL\_FILES\_ACCESS\_PERMISSION intent操作将用户引导至一个系统设置页面，在该页面上，你可以启用”授予所有文件的管理权限”.当赋予了这个权限时，App就可以使用MediaStore API或者File API访问共享存储空间和其他目录中所有文件。但是针对其他应用的私有目录，仍然是无法访问的。另外注意一点，使用这个权限如果需要上架google play，大概率是不会通过的，因为google不认为一个非管理类APP需要使用此权限。

综上所述，适配Android11存储的核心就是对各个分区文件处理

* App自身产生的文件应该存放在自己的私有目录下即/sdcard/Android/data/packagename/ 和/data/data/packagename/
* 媒体类和文档类文件存放共享存储空间下，使用MediaStore和SAF得到Uri构造输入输出流或者使用File API
* 其他目录不能通过File API直接访问，只能通过SAF。
* 赋予所有文件的管理权限时，可以使用File API操作以上所有文件，其他应用的私有目录除外

**三．总结**

介绍完分区存储，这里再提一点，虽然Android11上强制开启了分区存储，但是我们在适配的时候仍然需要将android:requestLegacyExternalStorage设置成true,这是为了在Android10的机器上正常运行。以上就是关于Android11文件存储适配的一些分析介绍，希望对大家升级android11有所帮助