



链滴

# Android-Uri To Path

作者: [wind](#)

原文链接: <https://ld246.com/article/1441014769169>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<pre>通过Uri得到实际路径，网上方法有很多，但我前几天试了一下，发现返回的值都是null，后来发现，Uri的格式在Android4.4之后有了很大的改动。</pre>

<p style="font-family:'microsoft yahei';font-size:14px;background-color:#FFFFFF;">  
content://media/external/images/media/17766(4.4前) &nbsp;<br />  
content://com.android.providers.media.documents/document/image:2706(4.4后)  
</p>

<ol style="font-family:'microsoft yahei';font-size:14px;background-color:#FFFFFF;">  
<li>  
4.4之前的方法&nbsp;<br />

4.4之前的方法网上都有很多，大多数都是基于cursor来实现的。这里主要讲4.4之后的。

</li>

<li>

<p>

4.4之后的方法&nbsp;<br />

代码在这里有了，大家可以对照来看。&nbsp;<br />

<a href="http://www.2cto.com/kf/201502/376975.html">http://www.2cto.com/kf/201502/376975.html</a>&nbsp;<br />

Uri分为三个部分&nbsp;<br />

content://com.android.providers.media.documents/document/image:2706&nbsp;<br />

scheme:content://&nbsp;<br />

authority: com.android.providers.media.documents&nbsp;<br />

path:document/image:2706

</p>

<p>

所以要得到uri对应的文件路径&nbsp;<br />

1、先找到文件是属于什么类型（通过authority来判断，源码中就根据authority的值来分为external\_document, media等），可以通过getAuthority()来得到。&nbsp;<br />

2、解析文件path的结构，这里拿MediaDocument举例&nbsp;<br />

String docId=DocumentContract.getDocumentId(uri);&nbsp;<br />

进入这个函数，发现包含getPathSegments(), 这就是将uri中的path解析成多个部分，如document和image:2706, image是类型, 2706就是ID啦。&nbsp;<br />

之后当然是判断类型咯，看看是image还是video还是audio，然后生成一个contentUri，这个时候后面的就代表的是路径而不是id了。&nbsp;<br />

例如，contentUri=content://+authority+/external/image/media&nbsp;<br />

现在已经知道了，uri对应的id是2706，之后怎么得到它的信息呢？对，跟4.4版本之前的一样也是用cursor，所以有：

</p>

</li>

</ol>

```
<pre class="prettyprint">cursor = context<span style="color:#444444;">.getContentResolver</span>().<span style="color:#444444;">.query</span>(uri, projection, selection, selectionArgs, null)<span style="color:#880000;">;</span></pre>
```

<pre>这是个比较关键的函数，cursor相当于一个数据库里的光标，ContentResolver是内容解析对，我们可以把要解析的内容想象成一个数据库，下面讲讲这几个参数的含义：

uri，就是contentUri了，包含路径的

projection，官方解释是a list of which columns return，意思就是返回哪些列，源码中是\_data，思就是返回\_data列

selection，源码中是\_id=?

selectionArgs，源码中是2706，即图片id，联合上面的参数就是在数据库寻找\_id=2706的数据</p></pre>

<p style="font-family:'microsoft yahei';font-size:14px;background-color:#FFFFFF;">

好了，我们现在可以看到，返回的cursor，就是指向\_id=2706的数据中的\_data列的数据，其实这

数据就是我们需要的path了。 &nbsp;  <br />

下面输出path

</p>

```
<pre class="prettyprint"><span>int</span> column_index=cursor.getColumnIndex(<span style="color:#008800;">"_data"</span>) <span style="color:#660066;">String</span> <span style="color:#000088;">data</span>=cursor.getString(<span>column_index</span>)</span></pre>
```

<p style="font-family:'microsoft yahei';font-size:14px;background-color:#FFFFFF;">

好了，其实也蛮简单的。。。但是我做了好久哎。。

</p>

<p style="font-family:'microsoft yahei';font-size:14px;background-color:#FFFFFF;">

<br />

</p>

<p style="font-family:'microsoft yahei';font-size:14px;background-color:#FFFFFF;">

<br />

</p>

<p style="font-family:'microsoft yahei';font-size:14px;background-color:#FFFFFF;">

<br />

</p>