

视频频道页面重构总结

By Long Luo

@2014.07.01

1. 重构前的一些缺点

1. 频道显示在VideoListActivity实现的，代码结构不够清晰；
2. 频道页面需要根据不同频道目前存在矩阵式显示和列表式显示2种方式，矩阵式显示嵌入在ListView中实现的，造成加载时绘制矩阵式页面需要更长的时间；
3. 后续频道页面会参考第三方视频应用，不同频道会有不同的展示方式，而目前功能可扩展性不够好；

2. 重构方案

重构方案:

1. 将频道页面独立出来，新建一个ChannelListActivity，作为所有频道页面的Activity，负责所有频道界面的绘制，便于后续扩展；
2. 针对不同频道，具有不同的布局方式，比如对于需要矩阵式显示的频道，使用矩阵式布局；
3. 矩阵式布局，替换原有的ListView加GridView显示，提高加载速度；

3. 具体实现

技术实现方案如下所示：

1. 新建ChannelListActivity，在VideoListActivity中用进入频道页的接口替换原有的接口，将相关处理在ChannelListActivity中实现；
2. 根据传入的频道ID不同，使用setContentViews()加载不同的布局方式，实现不同频道不同布局显示；
3. 新建ChannelGridAdapter，替换原有的ChannelListAdapter加ChannelGridListAdapter方式，提高加载速度；

4. 数据对比

我们通过TraceView工具比较前后2种方案的区别：

1. 启动时间：

比较两者启动时间，可以看出重构之后的方案是有大概20ms的提升。

	电影		电视剧		动漫	
1	248	168	188	196	170	173
2	190	121	178	127	156	167
3	176	129	153	145	186	132
4	153	123	198	132	167	143
5	178	135	169	146	152	135
平均(ms)	189	135.2	177.2	149.2	166.2	150

2. TraceView对比：

通过DDMS TraceView工具，我们再比较下2者性能上的区别：

重构之前：

重构之前的在绘制页面时需要绘制ListView，然后在每一个List中在绘制GridView，在TraceView获取到的数据如下：

252 android.graphics.BitmapFactory.nativeDecodeStream (Ljava/io/InputStream;[BLandroid/graphics/Rect;Landroid/gra	1.0%	23.152	0.4%	8.445	0.3%	24.125	0.1%	8.522	6+0	3.859	4.021
253 com.oppo/video/channel/ChannelListAdapter.getView (Landroid/view/View;Landroid/view/ViewGroup;Landroid/vie	1.0%	23.086	0.0%	0.339	0.3%	25.435	0.0%	0.346	2+0	11.543	12.718
Parents											
243 android.widget.HeaderViewListAdapter.getView (Landroid/view/View;Landroid/view/ViewGroup;Landroid/vie	100.0%	23.086			100.0%	25.435			2/2		
Children											
self		1.5%	0.339		1.4%	0.346					
47 android.view.LayoutInflater.inflate (Landroid/view/ViewGroup;Landroid/view/View;		79.2%	18.275		78.2%	19.879			2/13		
1262 android.widget.GridView.setAdapter (Landroid/widget/ListAdapter;V		7.3%	1.682		8.5%	2.158			2/2		
120 dalvik/system/VMDebug.startClassPrep (V		3.6%	0.841		3.3%	0.840			2/693		
2025 com.oppo/video/channel/ChannelGridListAdapter.setGroupPosition (I)V		2.3%	0.534		2.5%	0.635			2/2		
1212 android.widget.BaseAdapter.notifyDataSetChanged (V		2.2%	0.513		2.2%	0.561			2/3		
2704 com.oppo/video/channel/ChannelGridListAdapter.<init> (Landroid/content/Context;V		1.0%	0.226		0.9%	0.225			2/2		
2092 com.oppo/video/utlis/MyLog.v (Ljava/lang/String;Ljava/lang/String;V		0.8%	0.174		1.1%	0.292			2/8		
275 java/lang/StringBuilder.append (Ljava/lang/String;Ljava/lang/String;Builder;		0.5%	0.119		0.5%	0.120			2/482		
713 java/lang/StringBuilder.append (I)Ljava/lang/String;Builder;		0.5%	0.111		0.4%	0.110			2/89		
812 android.view.View.findViewById (I)Landroid/view/View;		0.3%	0.063		0.2%	0.061			2/324		
695 java/lang/StringBuilder.toString (I)Ljava/lang/String;		0.2%	0.042		0.2%	0.041			2/264		
4406 com.oppo/video/channel/ChannelListAdapter.ViewHolder.access\$300 (Lcom/oppo/video/channel/ChannelLi		0.1%	0.031		0.1%	0.031			8/8		
939 java/lang/StringBuilder.<init> (V		0.1%	0.031		0.1%	0.031			2/226		
828 android.widget.AbsListView.setOverScrollMode (I)V		0.1%	0.029		0.1%	0.027			2/10		
4598 com.oppo/video/channel/ChannelListAdapter.ViewHolder.<init> (Lcom/oppo/video/channel/ChannelListAda		0.1%	0.024		0.1%	0.023			2/2		
4501 android.widget.AdapterView.OnItemClickListener (Landroid/widget/AdapterView\$OnItemClickListener;V		0.1%	0.024		0.1%	0.025			2/3		

76 com.oppo/video/channel/ChannelGridListAdapter.getView (Landroid/view/View;Landroid/view/ViewGroup;Landroid/	6.9%	165.319	0.1%	1.859	2.1%	180.383	0.0%	1.877	9+0	18.369	20.043
Parents											
69 android.widget.AbsListView.obtainView (I)Z)Landroid/view/View;	100.0%	165.319			100.0%	180.383			9/9		
Children											
self		1.1%	1.859		1.0%	1.877					
47 android.view.LayoutInflater.inflate (Landroid/view/ViewGroup;Landroid/view/View;		67.4%	111.364		64.7%	116.618			5/13		
395 com.squareup.picasso/Picasso.with (Landroid/content/Context;)Lcom/squareup/picasso/Picasso;		8.3%	13.791		9.7%	17.542			9/9		
428 com.squareup.picasso/RequestCreator.into (Landroid/widget/ImageView;)V		7.6%	12.499		8.0%	14.444			9/9		
511 android.content.res.Resources.getString (I)Ljava/lang/Object;)Ljava/lang/String;		3.9%	6.506		3.8%	6.826			9/12		
458 android.widget.TextView.setText (Ljava/lang/Object;)Ljava/lang/CharSequence;)V		3.5%	5.814		3.3%	5.899			18/39		
688 java/lang/StringBuilder.append (Ljava/lang/Object;)Ljava/lang/String;Builder;		2.2%	3.572		2.0%	3.573			9/24		
275 java/lang/StringBuilder.append (Ljava/lang/String;)Ljava/lang/String;Builder;		1.6%	2.610		3.0%	5.401			54/482		
1275 com/squareup/picasso/Picasso.load (Ljava/lang/String;)Lcom/squareup/picasso/RequestCreator;		1.0%	1.641		1.0%	1.851			9/9		
850 com.oppo/video/utlis/MyLog.d (Ljava/lang/String;Ljava/lang/String;)V		0.7%	1.187		1.0%	1.875			18/66		
812 android.view.View.findViewById (I)Landroid/view/View;		0.5%	0.907		0.5%	0.909			15/324		
480 android.view.View.setVisibility (I)V		0.4%	0.712		0.4%	0.710			5/29		
120 dalvik/system/VMDebug.startClassPrep (V		0.4%	0.604		0.3%	0.603			2/693		
713 java/lang/StringBuilder.append (I)Ljava/lang/String;Builder;		0.3%	0.455		0.3%	0.455			9/89		
2180 com/squareup/picasso/Picasso.<clinit> (V		0.3%	0.442		0.2%	0.441			1/1		
695 java/lang/StringBuilder.toString (I)Ljava/lang/String;		0.2%	0.369		0.2%	0.374			18/264		
939 java/lang/StringBuilder.<init> (V		0.2%	0.260		0.1%	0.260			18/226		

从以上可以分析，在ChannelListAdapter的getView()就占据了1.0%的Incl Cpu Time，GridView的getView()占据了6.9%的Incl Cpu Time，合计占用了7.9%的CPU Time。

重构之后：

重构之后，我们仅需要在ChannelGridAdapter中绘制View，测量数据如下：

84 com/oppo/video/channel/ChannelGridAdapter.getView (Landroid/view/View;Landroid/view/View;	3.2%	86.607	0.0%	0.778	0.8%	100.009	0.0%	0.785	5+0	17.321	20.002
Parents											
77 android/widget/AbsListView.obtainView (IILandroid/view/View;	100.0%	86.607			100.0%	100.009			5/5		
Children											
self		0.9%	0.778		0.8%	0.785					
35 android/view/LayoutInflater.inflate (Landroid/view/ViewGroup;Landroid/view/View;		62.9%	54.472		62.3%	62.341			3/11		
355 com/squareup/picasso/Picasso.with (Landroid/content/Context;Lcom/squareup/pic		16.2%	14.030		17.9%	17.867			4/4		
670 com/squareup/picasso/RequestCreator.into (Landroid/widget/ImageView;I		6.4%	5.550		5.5%	5.548			4/4		
650 android/content/res/Resources.getString (IILjava/lang/Object;Ljava/lang/String;		3.2%	2.772		3.1%	3.063			4/7		
668 android/widget/TextView.setText (Ljava/lang/CharSequence;)V		2.5%	2.199		2.2%	2.194			8/22		
686 java/lang/StringBuilder.append (Ljava/lang/Object;Ljava/lang/StringBuilder;		1.7%	1.511		1.7%	1.714			5/22		
238 java/lang/StringBuilder.append (Ljava/lang/String;Ljava/lang/StringBuilder;		1.3%	1.139		1.3%	1.318			26/472		
1458 com/squareup/picasso/Picasso.load (Ljava/lang/String;Lcom/squareup/picasso/R		1.1%	0.987		1.0%	0.986			4/4		
129 dalvik/system/VMDebug.startClassPrep (JV		0.9%	0.744		0.9%	0.942			2/662		
1004 com/oppo/video/Utils.MyLog.d (Ljava/lang/String;Ljava/lang/String;)V		0.6%	0.546		1.3%	1.299			9/49		
1903 com/squareup/picasso/Picasso.<init> (V		0.6%	0.494		0.5%	0.494			1/1		
878 android/view/View.findViewById (ILandroid/view/View;		0.4%	0.307		0.3%	0.308			6/280		
1027 java/lang/StringBuilder.append (ILjava/lang/StringBuilder;		0.3%	0.270		0.3%	0.341			5/50		

从上图可以看出，getView()占据的CPU时间仅为**3.2%**，效率大大提高。

Created by Long Luo at 2014/7/1 15:45:01

Completed by Long Luo at 2014/7/2 14:42:29