

Android抽象布局——include、merge、ViewStub

2013-11-11 17:18

44425人阅读

评论(16)

收藏

举报

分类: [android UI \(常用\) \(66\)](#) [android性能优化 \(10\)](#)

版权声明: 本

文为博主原创文章, 未经博主允许不得转载。

目录(?)

[+]

在布局优化中, Androi的官方提到了这三种布局<include />、<merge />、<ViewStub />, 并介绍了这三种布局各自的优点, 下面也是简单说一下他们的优点, 以及怎么使用, 记下来权当做笔记。

1、布局重用<include />

<include />标签能够重用布局文件, 简单的使用如下:

```
[html] view plain copy print ?
01. <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
02.     android:orientation="vertical"
03.     android:layout_width="match_parent"
04.     android:layout_height="match_parent"
05.     android:background="@color/app_bg"
06.     android:gravity="center_horizontal">
07.
08.     <include layout="@layout/titlebar"/>
09.
10.     <TextView android:layout_width="match_parent"
11.         android:layout_height="wrap_content"
12.         android:text="@string/hello"
13.         android:padding="10dp" />
14.
15.     ...
16.
17. </LinearLayout>
```

⌵ 载 Ⓜ

1)<include />标签可以使用单独的layout属性, 这个也是必须使用的。

2)可以使用其他属性。<include />标签若指定了ID属性, 而你的layout也定义了ID, 则你的layout的ID会被覆盖, [解决方案](#)。

3)在include标签中所有的android:layout_*都是有效的, 前提是必须要写layout_width和layout_height两个属性。

4)布局中可以包含两个相同的include标签, 引用时可以使用如下方法解决 ([参考](#)):

```
[html] view plain copy print ?
01. View bookmarks_container_2 = findViewById(R.id.bookmarks_favourite);
02.
03. bookmarks_container_2.findViewById(R.id.bookmarks_list);
```

2、减少视图层级<merge />

<merge/>标签在UI的结构优化中起着非常重要的作用，它可以删减多余的层级，优化UI。<merge/>多用于替换FrameLayout或者当一个布局包含另一个时，<merge/>标签消除视图层次结构中多余的视图组。例如你的主布局文件是垂直布局，引入了一个垂直布局的include，这是如果include布局使用的LinearLayout就没意义了，使用的话反而减慢你的UI表现。这时可以使用<merge/>标签优化。

[📄 载指](#)

```
[html] view plain copy print ? C ?
01. <merge xmlns:android="http://schemas.android.com/apk/res/android">
02.
03.     <Button
04.         android:layout_width="fill_parent"
05.         android:layout_height="wrap_content"
06.         android:text="@string/add"/>
07.
08.     <Button
09.         android:layout_width="fill_parent"
10.         android:layout_height="wrap_content"
11.         android:text="@string/delete"/>
12.
13. </merge>
```

[📄 载指](#)

现在，当你添加该布局文件时(使用<include />标签)，系统忽略<merge />节点并且直接添加两个Button。更多<merge />介绍可以参考《[Android Layout Tricks #3: Optimize by merging](#)》

3、需要时使用<ViewStub />

<ViewStub />标签最大的优点是当你需要时才会加载，使用他并不会影响UI初始化时的性能。各种不常用的布局想进度条、显示错误消息等可以使用<ViewStub />标签，以减少内存使用量，加快渲染速度。<ViewStub />是一个不可见的，大小为0的View。<ViewStub />标签使用如下：

```
[html] view plain copy print ? C ?
01. <ViewStub
02.     android:id="@+id/stub_import"
03.     android:inflatedId="@+id/panel_import"
04.     android:layout="@layout/progress_overlay"
05.     android:layout_width="fill_parent"
06.     android:layout_height="wrap_content"
07.     android:layout_gravity="bottom" />
```

当你想加载布局时，可以使用下面其中一种方法：

[📄 载指](#)

```
[java] view plain copy print ? C ?
```

```
01. ((ViewStub) findViewById(R.id.stub_import)).setVisibility(View.VISIBLE);
02. // or
03. View importPanel = ((ViewStub) findViewById(R.id.stub_import)).inflate();
```

当调用`inflate()`函数的时候，`ViewStub`被引用的资源替代，并且返回引用的view。这样程序可以直接得到引用的view而不用再次调用函数`findViewById()`来查找了。

注：`ViewStub`目前有个缺陷就是还不支持 `<merge />` 标签。

[加载](#)

更多`<ViewStub />`标签介绍可以参考《[Android Layout Tricks #3: Optimize with stubs](#)》

```
/**
 * @author 张兴业
 * http://blog.csdn.net/xyz\_lmn
 * 我的新浪微博: @张兴业TBOW
 */
```

参考:

<http://developer.android.com/training/improving-layouts/reusing-layouts.html>

<http://android-developers.blogspot.com/2009/03/android-layout-tricks-3-optimize-by.html>

<http://developer.android.com/training/improving-layouts/optimizing-layout.html#Lint>

<http://android-developers.blogspot.com/2009/02/android-layout-tricks-1.html>

<http://developer.android.com/training/improving-layouts/loading-ondemand.html>