

- 【Android 自定义控件实战】 (29)
- 【Android 自定义控件之起步】 (7)
- 【Android 快速开发】 (12)
- 【Android 原生开发游戏】 (3)
- 【Java 并发专题】 (15)
- 【android 进阶之路】 (75)
- 【Java 设计模式】 (10)
- 【Android 百度地图】 (4)
- 【html5 css3精彩案例】 (14)
- 【Android github 控件】 (10)
- 【Android 基础】 (16)
- 【Javascript 】 (9)
- 【rabbitMQ 用法】 (5)
- 【Android微知识点】 (4)

友情链接

- 郭霖的博客
- 夏安明的博客
- 任玉刚的博客
- 亓斌的博客
- Github开源项目的配套视频
- 敬佩的孔老师
- foruok的订阅号程序视界
- OpenCV大神shiter
- 专为Android程序员的导航
- 泡在网上的日子

博客专栏



HTML5 & CSS3 实战

文章：11篇

阅读：204447



设计模式融入生活

文章：10篇

阅读：118341



Android 精彩案例

文章：67篇

阅读：5862959

阅读排行

- Android Https相关完全解析 ... (1588582)
- Android Fragment 真正的完... (671931)
- Android RecyclerView 使用... (668741)
- Android OkHttp完全解析 是... (497038)
- Android 自定义View (一) (313373)
- Android 属性动画 (Property... (281697)
- Android 屏幕适配方案 (250050)
- Android Fragment 真正的完... (240572)
- Android 手把手教您自定义Vi... (223355)

可以看到支持宽高，以及margin。

也就是说，大家只要在开发过程中使用 PercentRelativeLayout 、 PercentFrameLayout 替换 FrameLayout 、 RelativeLayout 即可。

是不是很简单，不过貌似没有LinearLayout，有人会说LinearLayout有weight属性 weight属性只能支持一个方向呀~~哈，没事，刚好给我们一个机会去自定义一个 PercentLinearLayout 。

好了，本文分为3个部分：

- PercentRelativeLayout 、 PercentFrameLayout 的使用
- 对上述控件源码分析
- 自定义 PercentLinearLayout

二、使用

关于使用，其实及其简单，并且github上也有例子， android-percent-support-lib-sample。我们就简单过一下：

首先记得在build.gradle添加：

```
1 | compile 'com.android.support.percent:22.2.0'
```

(一) PercentFrameLayout

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <android.support.percent.PercentFrameLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7
8     <TextView
9         android:layout_width="0dp"
10        android:layout_height="0dp"
11        android:layout_gravity="left|top"
12        android:background="#44ff0000"
13        android:text="width:30%,height:20%"
14        app:layout_heightPercent="20%"
15        android:gravity="center"
16        app:layout_widthPercent="30%"/>
17
18    <TextView
19        android:layout_width="0dp"
20        android:layout_height="0dp"
21        android:layout_gravity="right|top"
22        android:gravity="center"
23        android:background="#4400ff00"
24        android:text="width:70%,height:20%"
25        app:layout_heightPercent="20%"
26        app:layout_widthPercent="70%"/>
27
28    <TextView
29        android:layout_width="0dp"
30        android:layout_height="0dp"
```

微信关注我的公众号

Android 沉浸式状态栏攻略 ...
(212232)

文章搜索

最新评论

- Android 屏幕适配方案
qq_27788349 : hihi. 可以讲讲图片怎么也适配吗？
- Android 自定义View (三) 圆环交替 等待...
wsmdyys : @yuxiang88679:请问一下是怎么解决颜色冲突时有边线的问题的
- Android 仿火萤视频桌面 神奇的LiveWall...
独钓寒江雪_ : 一句“果然很简单”然后自己搞出来了！大神就是厉害
- Android 带你玩转实现游戏2048 其实20...
Super_wqp : button + 二维数组感觉容易理解一点
- 为RecyclerView打造通用Adapter 让Rec...
ofuufu : 我用DWRRefreshLayout,刷新和加载更多一次搞定: <https://github.com/>...
- Java爬虫，信息抓取的实现
qq_37782822 : @weixin_36934512:哥们你的微信加不上啊，我也遇到同类问题，加我微信：舞出我人生
- Java爬虫，信息抓取的实现
qq_37782822 : 百度的测试报异常org.jsoup.UnsupportedMimeTypeException: Un...
- Android 自定义View (一)
张二向日葵 : 超赞！
- 学会编写Android Studio插件 别停留在...
郁闷的中交兵 : 赞一个啊，大佬那个代码补全怎么弄啊，我想写一个gl的着色语言的代码补全
- Android RecyclerView 使用完全解析 体...
jacksonrickqml : @Mark_Liu:你的分析是对的，可以直接在大神的code上修改：getItemOffsets方...

统计

微信公众号

```
32         android:layout_gravity="bottom"
33         android:background="#770000ff"
34         android:text="width:100%,height:10%"
35         android:gravity="center"
36         app:layout_heightPercent="10%"
37         app:layout_widthPercent="100%"/>
38
39
40     </android.support.percent.PercentFrameLayout>
```

3个TextView，很简单，直接看效果图：



(二) PercentRelativeLayout

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <android.support.percent.PercentRelativeLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:clickable="true">
8
9      <TextView
10         android:id="@+id/row_one_item_one"
11         android:layout_width="0dp"
12         android:layout_height="0dp"
13         android:layout_alignParentTop="true"
14         android:background="#7700ff00"
15         android:text="w:70%,h:20%"
16         android:gravity="center"
17         app:layout_heightPercent="20%"
18         app:layout_widthPercent="70%"/>
19
20      <TextView
21         android:id="@+id/row_one_item_two"
22         android:layout_width="0dp"
23         android:layout_height="0dp"
24         android:layout_toRightOf="@+id/row_one_item_one"
25         android:background="#396190"
26         android:text="w:30%,h:20%"
27         app:layout_heightPercent="20%"
28         android:gravity="center"
29         app:layout_widthPercent="30%"/>
30
```

微信关注我的公众号

```
31
32     <ImageView
33         android:id="@+id/row_two_item_one"
34         android:layout_width="match_parent"
35         android:layout_height="0dp"
36         android:src="@drawable/tangyan"
37         android:scaleType="centerCrop"
38         android:layout_below="@+id/row_one_item_one"
39         android:background="#d89695"
40         app:layout_heightPercent="70%"/>
41
42     <TextView
43         android:layout_width="0dp"
44         android:layout_height="0dp"
45         android:layout_below="@id/row_two_item_one"
46         android:background="#770000ff"
47         android:gravity="center"
48         android:text="width:100%,height:10%"
49         app:layout_heightPercent="10%"
50         app:layout_widthPercent="100%"/>
51
52 </android.support.percent.PercentRelativeLayout>
53
54
```

ok，依然是直接看效果图：



使用没什么好说的，就是直观的看一下。

三、源码分析

其实细想一下，Google只是对我们原本熟悉的RelativeLayout和FrameLayout进行的功能的扩展，使其支持了percent相关的属性。

那么，我们考虑下，如果是我们添加这种扩展，我们会怎么做：

- 通过LayoutParams获取child设置的percent相关属性的值

- onMeasure的时候，将child的width,height的值，通过获取的自定义属性的值进行计算（eg: 容器的宽 * fraction ），计算后传入给child.measure(w,h);

ok，有了上面的猜想，我们直接看 PercentFrameLayout 的源码。

```
1 public class PercentFrameLayout extends FrameLayout {
2     private final PercentLayoutHelper mHelper = new PercentLayoutHelp
3
4     //省略了，两个构造方法
5
6     public PercentFrameLayout(Context context, AttributeSet attrs) {
7         super(context, attrs);
8     }
9
10
11
12     @Override
13     public LayoutParams generateLayoutParams(AttributeSet attrs) {
14         return new LayoutParams(getContext(), attrs);
15     }
16
17     @Override
18     protected void onMeasure(int widthMeasureSpec, int heightMeasureSpec) {
19         mHelper.adjustChildren(widthMeasureSpec, heightMeasureSpec);
20         super.onMeasure(widthMeasureSpec, heightMeasureSpec);
21         if (mHelper.handleMeasuredStateTooSmall()) {
22             super.onMeasure(widthMeasureSpec, heightMeasureSpec);
23         }
24     }
25
26     @Override
27     protected void onLayout(boolean changed, int left, int top, int right, int bottom) {
28         super.onLayout(changed, left, top, right, bottom);
29         mHelper.restoreOriginalParams();
30     }
31
32     public static class LayoutParams extends FrameLayout.LayoutParams
33         implements PercentLayoutHelper.PercentLayoutParams {
34         private PercentLayoutHelper.PercentLayoutInfo mPercentLayoutInfo;
35
36         public LayoutParams(Context c, AttributeSet attrs) {
37             super(c, attrs);
38             mPercentLayoutInfo = PercentLayoutHelper.getPercentLayoutInfo(c, attrs);
39         }
40         //省略了一些代码...
41
42         @Override
43         public PercentLayoutHelper.PercentLayoutInfo getPercentLayoutInfo() {
44             return mPercentLayoutInfo;
45         }
46
47         @Override
48         protected void setBaseAttributes(TypedArray a, int widthAttr, int heightAttr) {
49             PercentLayoutHelper.fetchWidthAndHeight(this, a, widthAttr, heightAttr);
50         }
51     }
52 }
```

代码是相当的短，可以看到 PercentFrameLayout 里面首先重写了generateLayoutParams方法，当然了，由于支持了一些新的layout_属性，那么肯定需要定义对应的LayoutParams。

（一）percent相关属性的获取

可以看到PercentFrameLayout.LayoutParams在原有的FrameLayout.LayoutParams基础上，实现了PercentLayoutHelper.PercentLayoutParams接口。

这个接口很简单，只有一个方法：

```
1 public interface PercentLayoutParams {
2     PercentLayoutInfo getPercentLayoutInfo();
3 }
```

而，这个方法的实现呢，也只有一行：`return mPercentLayoutInfo;`，那么这个mPercentLayoutInfo在哪完成赋值呢？

看PercentFrameLayout.LayoutParams的构造方法：

```
1 public LayoutParams(Context c, AttributeSet attrs) {
2     super(c, attrs);
3     mPercentLayoutInfo = PercentLayoutHelper.getPercentLayoutInfo(c, attrs);
4 }
```

可以看到，将attrs传入给getPercentLayoutInfo方法，那么不用说，这个方法的内部，肯定是获取自定义属性的值，然后将其封装到PercentLayoutInfo对象中，最后返回。

代码如下：

```
1 public static PercentLayoutInfo getPercentLayoutInfo(Context context,
2     AttributeSet attrs) {
3     PercentLayoutInfo info = null;
4     TypedArray array = context.obtainStyledAttributes(attrs, R.styleable.PercentLa
5     float value = array.getFraction(R.styleable.PercentLayout_Layout_layout_widthP
6         -1f);
7     if (value != -1f) {
8         if (Log.isLoggable(TAG, Log.VERBOSE)) {
9             Log.v(TAG, "percent width: " + value);
10        }
11        info = info != null ? info : new PercentLayoutInfo();
12        info.widthPercent = value;
13    }
14    value = array.getFraction(R.styleable.PercentLayout_Layout_layout_heightPercen
15    if (value != -1f) {
16        if (Log.isLoggable(TAG, Log.VERBOSE)) {
17            Log.v(TAG, "percent height: " + value);
18        }
19        info = info != null ? info : new PercentLayoutInfo();
20        info.heightPercent = value;
21    }
22    value = array.getFraction(R.styleable.PercentLayout_Layout_layout_marginPercen
23    if (value != -1f) {
24        if (Log.isLoggable(TAG, Log.VERBOSE)) {
25            Log.v(TAG, "percent margin: " + value);
26        }
27        info = info != null ? info : new PercentLayoutInfo();
28        info.leftMarginPercent = value;
29        info.topMarginPercent = value;
30        info.rightMarginPercent = value;
31        info.bottomMarginPercent = value;
32    }
33    value = array.getFraction(R.styleable.PercentLayout_Layout_layout_marginLeftPe
34        -1f);
35    if (value != -1f) {
36        if (Log.isLoggable(TAG, Log.VERBOSE)) {
37            Log.v(TAG, "percent left margin: " + value);
38        }
39        info = info != null ? info : new PercentLayoutInfo();
40        info.leftMarginPercent = value;
```

```

41     }
42     value = array.getFraction(R.styleable.PercentLayout_Layout_layout_marginTopPer
43         -1f);
44     if (value != -1f) {
45         if (Log.isLoggable(TAG, Log.VERBOSE)) {
46             Log.v(TAG, "percent top margin: " + value);
47         }
48         info = info != null ? info : new PercentLayoutInfo();
49         info.topMarginPercent = value;
50     }
51     value = array.getFraction(R.styleable.PercentLayout_Layout_la
52         -1f);
53     if (value != -1f) {
54         if (Log.isLoggable(TAG, Log.VERBOSE)) {
55             Log.v(TAG, "percent right margin: " + value);
56         }
57         info = info != null ? info : new PercentLayoutInfo();
58         info.rightMarginPercent = value;
59     }
60     value = array.getFraction(R.styleable.PercentLayout_Layout_layout_marginBottom
61         -1f);
62     if (value != -1f) {
63         if (Log.isLoggable(TAG, Log.VERBOSE)) {
64             Log.v(TAG, "percent bottom margin: " + value);
65         }
66         info = info != null ? info : new PercentLayoutInfo();
67         info.bottomMarginPercent = value;
68     }
69     value = array.getFraction(R.styleable.PercentLayout_Layout_layout_marginStartP
70         -1f);
71     if (value != -1f) {
72         if (Log.isLoggable(TAG, Log.VERBOSE)) {
73             Log.v(TAG, "percent start margin: " + value);
74         }
75         info = info != null ? info : new PercentLayoutInfo();
76         info.startMarginPercent = value;
77     }
78     value = array.getFraction(R.styleable.PercentLayout_Layout_layout_marginEndPer
79         -1f);
80     if (value != -1f) {
81         if (Log.isLoggable(TAG, Log.VERBOSE)) {
82             Log.v(TAG, "percent end margin: " + value);
83         }
84         info = info != null ? info : new PercentLayoutInfo();
85         info.endMarginPercent = value;
86     }
87     array.recycle();
88     if (Log.isLoggable(TAG, Log.DEBUG)) {
89         Log.d(TAG, "constructed: " + info);
90     }
91     return info;
92 }

```

是不是和我们平时的取值很类似，所有的值最终封装到PercentLayoutInfo对象中。

ok，到此我们的属性获取就介绍完成，有了这些属性，是不是onMeasure里面要进行使用呢？

(二) onMeasure中重新计算child的尺寸

```

1  @Override
2  protected void onMeasure(int widthMeasureSpec, int heightMeasureSpec) {
3      mHelper.adjustChildren(widthMeasureSpec, heightMeasureSpec);
4      super.onMeasure(widthMeasureSpec, heightMeasureSpec);
5      if (mHelper.handleMeasuredStateTooSmall()) {
6          super.onMeasure(widthMeasureSpec, heightMeasureSpec);
7      }
8  }

```

微信关注我的公众号

可以看到onMeasure中的代码页很少，看来核心的代码都被封装在mHelper的方法中，我们直接看mHelper.adjustChildren方法。

```

1  /**
2   * Iterates over children and changes their width and height to one calculated fro
3   * values.
4   * @param widthMeasureSpec Width MeasureSpec of the parent ViewGr
5   * @param heightMeasureSpec Height MeasureSpec of the parent View
6   */
7  public void adjustChildren(int widthMeasureSpec, int heightMeasur
8      //...
9      int widthHint = View.MeasureSpec.getSize(widthMeasureSpec);
10     int heightHint = View.MeasureSpec.getSize(heightMeasureSpec);
11     for (int i = 0, N = mHost.getChildCount(); i < N; i++) {
12         View view = mHost.getChildAt(i);
13         ViewGroup.LayoutParams params = view.getLayoutParams();
14
15         if (params instanceof PercentLayoutParams) {
16             PercentLayoutInfo info =
17                 ((PercentLayoutParams) params).getPercentLayoutInfo();
18             if (Log.isLoggable(TAG, Log.DEBUG)) {
19                 Log.d(TAG, "using " + info);
20             }
21             if (info != null) {
22                 if (params instanceof ViewGroup.MarginLayoutParams) {
23                     info.fillMarginLayoutParams((ViewGroup.MarginLayoutParams) par
24                         widthHint, heightHint);
25                 } else {
26                     info.fillLayoutParams(params, widthHint, heightHint);
27                 }
28             }
29         }
30     }
31 }

```

通过注释也能看出，此方法中遍历所有的孩子，通过百分比的属性重新设置其宽度和高度。

首先在widthHint、heightHint保存容器的宽、高，然后遍历所有的孩子，判断其LayoutParams是否是PercentLayoutParams类型，如果是，通过params.getPercentLayoutInfo拿出info对象。

是否还记得，上面的分析中，PercentLayoutInfo保存了percent相关属性的值。

如果info不为null，则判断是否需要处理margin；我们直接看fillLayoutParams方法（处理margin也是类似的）。

```

1  /**
2   * Fills {@code ViewGroup.LayoutParams} dimensions based on percentage values.
3   */
4  public void fillLayoutParams(ViewGroup.LayoutParams params, int widthHint,
5      int heightHint) {
6      // Preserve the original layout params, so we can restore them after the m
7      mPreservedParams.width = params.width;
8      mPreservedParams.height = params.height;
9
10     if (widthPercent >= 0) {
11         params.width = (int) (widthHint * widthPercent);
12     }
13     if (heightPercent >= 0) {
14         params.height = (int) (heightHint * heightPercent);
15     }
16     if (Log.isLoggable(TAG, Log.DEBUG)) {
17         Log.d(TAG, "after fillLayoutParams: (" + params.width + ", " + params.
18     )
19 }

```

微信关注我的公众号

首先保存原本的width和height，然后重置params的width和height

为 (int) (widthHint * widthPercent) 和 (int) (heightHint * heightPercent);。

到此，其实我们的百分比转换就结束了，理论上就已经实现了对于百分比的支持，不过Google还考虑了一些细节。

我们回到onMeasure方法：

```

1  @Override
2      protected void onMeasure(int widthMeasureSpec, int heightMeasureSpec) {
3      mHelper.adjustChildren(widthMeasureSpec, heightMeasureSpec);
4      super.onMeasure(widthMeasureSpec, heightMeasureSpec);
5      if (mHelper.handleMeasuredStateTooSmall()) {
6          super.onMeasure(widthMeasureSpec, heightMeasureSpec);
7      }
8  }
```

下面还有个mHelper.handleMeasuredStateTooSmall的判断，也就是说，如果你设置的百分比，最终计算出来的MeasuredSize过小的话，会进行一些操作。

代码如下：

```

1  public boolean handleMeasuredStateTooSmall() {
2      boolean needsSecondMeasure = false;
3      for (int i = 0, N = mHost.getChildCount(); i < N; i++) {
4          View view = mHost.getChildAt(i);
5          ViewGroup.LayoutParams params = view.getLayoutParams();
6          if (Log.isLoggable(TAG, Log.DEBUG)) {
7              Log.d(TAG, "should handle measured state too small " + view + " " + pa
8          }
9          if (params instanceof PercentLayoutParams) {
10             PercentLayoutInfo info =
11                 ((PercentLayoutParams) params).getPercentLayoutInfo();
12             if (info != null) {
13                 if (shouldHandleMeasuredWidthTooSmall(view, info)) {
14                     needsSecondMeasure = true;
15                     params.width = ViewGroup.LayoutParams.WRAP_CONTENT;
16                 }
17                 if (shouldHandleMeasuredHeightTooSmall(view, info)) {
18                     needsSecondMeasure = true;
19                     params.height = ViewGroup.LayoutParams.WRAP_CONTENT;
20                 }
21             }
22         }
23     }
24     if (Log.isLoggable(TAG, Log.DEBUG)) {
25         Log.d(TAG, "should trigger second measure pass: " + needsSecondMeasure);
26     }
27     return needsSecondMeasure;
28 }
```

首先遍历所有的孩子，拿出孩子的layoutparams，如果是PercentLayoutParams实例，则取出info。如果info不为null，调用 shouldHandleMeasuredWidthTooSmall 判断：

```

1  private static boolean shouldHandleMeasuredWidthTooSmall(View view, PercentLayoutInfo
2      int state = ViewCompat.getMeasuredWidthAndState(view) & ViewCompat.MEASURED_ST
3      return state == ViewCompat.MEASURED_STATE_TOO_SMALL && info.widthPercent >= 0
4          info.mPreservedParams.width == ViewGroup.LayoutParams.WRAP_CONTENT;
5  }
```

这里就是判断，如果你设置的measuredWidth或者measureHeight过小的话，并且你在布局文件中layout_w/h 设置的是WRAP_CONTENT的话，将params.width / height= ViewGroup.LayoutParams.WRAP_CONTENT，然后重新测量。

哈，onMeasure终于结束了~~~现在我觉得应该代码结束了吧，尺寸都设置好了，还重画干嘛？
but，你会发现onLayout也重写了，我们又不改变layout规则，在onLayout里面干

```

1  @Override
2      protected void onLayout(boolean changed, int left, int top, int right, int bottom) {
3          super.onLayout(changed, left, top, right, bottom);
4          mHelper.restoreOriginalParams();
5      }

```

继续看 mHelper.restoreOriginalParams

```

1  /**
2   * Iterates over children and restores their original dimensions that were changed
3   * percentage values. Calling this method only makes sense if you previously called
4   * {@link PercentLayoutHelper#adjustChildren(int, int)}.
5   */
6  public void restoreOriginalParams() {
7      for (int i = 0, N = mHost.getChildCount(); i < N; i++) {
8          View view = mHost.getChildAt(i);
9          ViewGroup.LayoutParams params = view.getLayoutParams();
10         if (Log.isLoggable(TAG, Log.DEBUG)) {
11             Log.d(TAG, "should restore " + view + " " + params);
12         }
13         if (params instanceof PercentLayoutParams) {
14             PercentLayoutInfo info =
15                 ((PercentLayoutParams) params).getPercentLayoutInfo();
16             if (Log.isLoggable(TAG, Log.DEBUG)) {
17                 Log.d(TAG, "using " + info);
18             }
19             if (info != null) {
20                 if (params instanceof ViewGroup.MarginLayoutParams) {
21                     info.restoreMarginLayoutParams((ViewGroup.MarginLayoutParams) params);
22                 } else {
23                     info.restoreLayoutParams(params);
24                 }
25             }
26         }
27     }
28 }

```

噢，原来是重新恢复原本的尺寸值，也就是说onMeasure里面的对值进行了改变，测量完成后。在这个地方，将值又恢复成如果布局文件中的值，上面写的都是0。恢复很简单：

```

1
2  public void restoreLayoutParams(ViewGroup.LayoutParams params) {
3      params.width = mPreservedParams.width;
4      params.height = mPreservedParams.height;
5  }

```

你应该没有忘在哪存的把~忘了的话，麻烦Ctrl+F ‘mPreservedParams.width’。

也就是说，你去打印上面写法，布局文件中views的 v.getLayoutParams().width，这个值应该是0。

这里感觉略微不爽~这个0没撒用处呀，还不如不重置~~

好了，到此就分析完了，其实主要就几个步骤：

微信关注我的公众号

- LayoutParams中属性的获取
- onMeasure中，改变params.width为百分比计算结果，测量
- 如果测量值过小且设置的w/h是wrap_content，重新测量
- onLayout中，重置params.w/h为布局文件中编写的值

可以看到，有了RelativeLayout、FrameLayout的扩展，竟然没有LinearLayout！
我们的核心代码都由 PercentLayoutHelper 封装了，自己扩展下LinearLayout也不复

三、实现PercentLinearlayout

可能有人会说，有了weight呀，但是weight能做到宽、高同时百分比赋值嘛？

好了，代码很简单，如下：

(一) PercentLinearLayout

```
1 package com.juliengenoud.percentsamples;
2
3 import android.content.Context;
4 import android.content.res.TypedArray;
5 import android.support.percent.PercentLayoutHelper;
6 import android.util.AttributeSet;
7 import android.view.ViewGroup;
8 import android.widget.LinearLayout;
9
10 /**
11  * Created by zhy on 15/6/30.
12  */
13 public class PercentLinearLayout extends LinearLayout
14 {
15
16     private PercentLayoutHelper mPercentLayoutHelper;
17
18     public PercentLinearLayout(Context context, AttributeSet attrs)
19     {
20         super(context, attrs);
21
22         mPercentLayoutHelper = new PercentLayoutHelper(this);
23     }
24
25
26     @Override
27     protected void onMeasure(int widthMeasureSpec, int heightMeasureSpec)
28     {
29         mPercentLayoutHelper.adjustChildren(widthMeasureSpec, heightMeasureSpec);
30         super.onMeasure(widthMeasureSpec, heightMeasureSpec);
31         if (mPercentLayoutHelper.handleMeasuredStateTooSmall())
32         {
33             super.onMeasure(widthMeasureSpec, heightMeasureSpec);
34         }
35     }
36
37     @Override
38     protected void onLayout(boolean changed, int l, int t, int r, int b)
39     {
40         super.onLayout(changed, l, t, r, b);
41         mPercentLayoutHelper.restoreOriginalParams();
42     }
43
44     @Override
45     public LayoutParams generateLayoutParams(AttributeSet attrs)
```

微信关注我的公众号

```

46     {
47         return new LayoutParams(getContext(), attrs);
48     }
49
50
51     public static class LayoutParams extends LinearLayout.LayoutParams
52         implements PercentLayoutHelper.PercentLayoutParams
53     {
54         private PercentLayoutHelper.PercentLayoutInfo mPercentLayoutI
55
56         public LayoutParams(Context c, AttributeSet attrs)
57         {
58             super(c, attrs);
59             mPercentLayoutInfo = PercentLayoutHelper.getPercentLayoutInfo(c, attrs);
60         }
61
62         @Override
63         public PercentLayoutHelper.PercentLayoutInfo getPercentLayoutInfo()
64         {
65             return mPercentLayoutInfo;
66         }
67
68         @Override
69         protected void setBaseAttributes(TypedArray a, int widthAttr, int heightAttr)
70         {
71             PercentLayoutHelper.fetchWidthAndHeight(this, a, widthAttr, heightAttr);
72         }
73
74         public LayoutParams(int width, int height) {
75             super(width, height);
76         }
77
78
79         public LayoutParams(ViewGroup.LayoutParams source) {
80             super(source);
81         }
82
83         public LayoutParams(MarginLayoutParams source) {
84             super(source);
85         }
86
87     }
88
89 }
90

```

如果你详细看了上面的源码分析，这个代码是不是没撒解释的了~

(二) 测试布局

```

1  <?xml version="1.0" encoding="utf-8"?>
2
3
4  <com.juliengenoud.percentsamples.PercentLinearLayout
5      xmlns:android="http://schemas.android.com/apk/res/android"
6      xmlns:app="http://schemas.android.com/apk/res-auto"
7      android:layout_width="match_parent"
8      android:layout_height="match_parent"
9      android:orientation="vertical">
10
11     <TextView
12         android:layout_width="0dp"
13         android:layout_height="0dp"
14         android:background="#ff44aacc"
15         android:text="width:60%,height:5%"
16         android:textColor="#ffffff"
17         app:layout_heightPercent="5%"
18         app:layout_marginBottomPercent="5%"

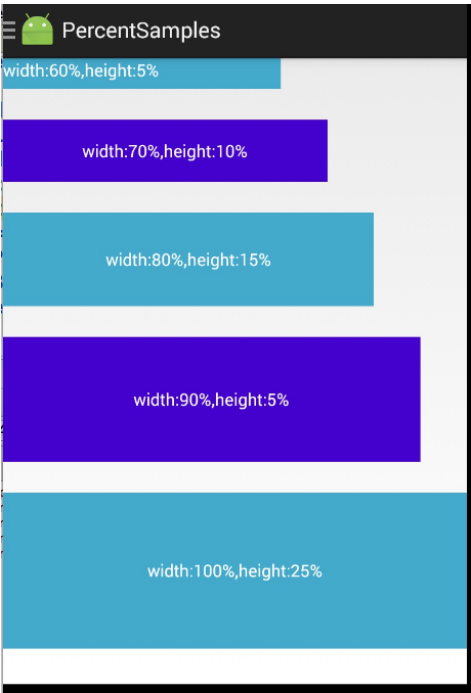
```

微信关注我的公众号

```
19         app:layout_widthPercent="60%"/>
20
21     <TextView
22         android:layout_width="0dp"
23         android:layout_height="0dp"
24         android:background="#ff4400cc"
25         android:gravity="center"
26         android:textColor="#ffffff"
27         android:text="width:70%, height:10%"
28         app:layout_heightPercent="10%"
29         app:layout_marginBottomPercent="5%"
30         app:layout_widthPercent="70%"/>
31     <TextView
32         android:layout_width="0dp"
33         android:layout_height="0dp"
34         android:background="#ff44aacc"
35         android:gravity="center"
36         android:text="width:80%, height:15%"
37         android:textColor="#ffffff"
38         app:layout_heightPercent="15%"
39         app:layout_marginBottomPercent="5%"
40         app:layout_widthPercent="80%"/>
41     <TextView
42         android:layout_width="0dp"
43         android:layout_height="0dp"
44         android:background="#ff4400cc"
45         android:gravity="center"
46         android:text="width:90%, height:5%"
47         android:textColor="#ffffff"
48         app:layout_heightPercent="20%"
49         app:layout_marginBottomPercent="10%"
50         app:layout_widthPercent="90%"/>
51
52     <TextView
53         android:layout_width="match_parent"
54         android:layout_height="0dp"
55         android:background="#ff44aacc"
56         android:gravity="center"
57         android:text="width:100%, height:25%"
58         android:textColor="#ffffff"
59         app:layout_heightPercent="25%"
60         app:layout_marginBottomPercent="5%"
61     />
62
63
64 </com.juliengenoud.percentsamples.PercentLinearLayout>
65
```

我们纵向排列的几个TextView，分别设置宽/高都为百分比，且之间的间隔为5%p。

(三)效果图



ok，到此，我们使用、源码分析、扩展PercentLinearLayout就结束了。

添加PercentLinearLayout后的地址：[点击查看](#)

扩展下载:[android-percent-support-extend](#) 包含android studio, eclipse项目，以及上述源码。

~~have a nice day ~~

新浪微博

群号：463081660

微信公众号：hongyangAndroid
(欢迎关注，第一时间推送博文信息)



顶 136 踩 11

- 上一篇 Android 自己实现 NavigationView [Design Support Library(1)]
- 下一篇 Android 增强版百分比布局库 为了适配而扩展

相关文章推荐

- Android 增强版百分比布局库 为了适配而扩展
- Android 百分比布局支持库介绍
- 安卓按比例布局，layout_weight用法
- Android百分比布局之layout_w

微信关注我的公众号

- Android中的百分比布局
 - android 百分比布局 for eclipse
 - Android中实现百分比布局的方式
- Android 百分比布局 (Percent Support Library)
 - Android 自定义百分比布局
 - Android 百分比布局库(percent-support-lib) 解...

猜你在找

- 机器学习之概率与统计推断

机器学习之凸优化

响应式布局全新探索

深度学习基础与TensorFlow实践

前端开发在线峰会
- 机器学习之数学基础

机器学习之矩阵

探究Linux的总线、设备、驱动模型


深度学习之神经网络原理与实战技巧

TensorFlow实战进阶：手把手教你做图

查看评论

xiangzilun05135楼 2017-06-28 11:32发表


我用RecyclerView是可以的，感觉ListView已经不好使了

keymonliuhai134楼 2017-03-17 14:10发表


有谁能告诉我一下，如果item里面我要实现圆角图片，应该怎么做，感觉用了之后圆角不能显示了。

行书开发133楼 2017-03-10 10:47发表


一篇优秀的文章，源码看的很到位!

u010203975132楼 2017-03-08 14:45发表

洋神，我在公司的时候做适配也是用这个比例布局去搞的，图片嘛我就用最大分辨率的那一套，因为最大的话被压缩还看得清，如果太小的话被拉伸就会变模糊了！但我发现如果两款适配分辨率差太大的话，比如一台是手机(比较小的尺寸)，一台是平板，这样的话在平板上就会显得太大了，因为是用比例去计算长度和宽度的，平板尺寸和分辨率比较大，同样是20%，在手机和平板计算出来差距还是有点大的！这样好像这种比例布局的适配只适合同一类别的设备，尺寸和屏幕分辨率不能差太大，太大了肯定有一端会显得太大或者太小了。不知道有没有可以同时适配手机和平板的方案？

努力奔跑的蜗牛哥131楼 2017-02-22 23:19发表

This support library should not use a different version (22) than the compileSdkVersion (25) less... (Ctrl+F1)
There are some combinations of libraries, or tools and libraries, that are incompatible, or can lead to bugs. One such incompatibility is compiling with a version of the Android support libraries that is not the latest version (or in particular, a version lower than your targetSdkVersion.)出现这个

youyuge34Re: 2017-03-02 19:32发表

回复努力奔跑的蜗牛哥：因为博文文章比较老了，gradle引用的support包版本还是21（对应api21），而你用的编译的api是新的25，将gradle中引用的support包版本改成25以上的就行了



恭喜您，文章已被推荐收录



boyceboy

129楼 2017-02-10 16:01发表

插个眼



John Vincent

128楼 2016-12-30

鸿神，用了你这个布局后，适配虽然好很多了，但是EventBus3.0用不了了。报错 EventBus: No subscribers registered for event class org.greenrobot.eventbus.NoSubscriberEvent



小小程序源

Re: 2017-01-09

回复John Vincent：你确定这两者有直接关系么，之所以会报这个错是因为你当前类没有观察者注册，EventBus找不到带注解@Subscribe的观察者方法，自然会报错。你将EventBus.register(this);这句话干掉就行了。



ITluochen

127楼 2016-12-18 12:53发表

在percentlinearLayout 中添加radiogroup 设置百分比不显示呢？博主



u010289384

126楼 2016-11-29 17:51发表

我自己也一直在网上找安卓的屏幕适配啊，这应该算是最完美的解决方案了



baidu_36851077

125楼 2016-11-28 17:33发表

请问大神，这个不能兼容下拉刷新的组件应该怎么办？？



敲代码的鸡

124楼 2016-11-16 00:40发表

我脾气向来很好。但是看见你的评论气打一处来，哪怕是时隔一年的评论让人看了也很恼火，人家分享自己的经验让我们受益。你是不是首先要说声谢谢。只知道一味的索取。别人有什么义务去提升你的经验。任何时候希望你有一颗懂得感恩和回报的心！



一杯粗茶走江湖

123楼 2016-11-01 11:48发表

对啊，我也发现listview的item用这个的话自定义的属性识别不了



sg_vgt

122楼 2016-10-29 16:47发表

感觉很有用，至少适配不再头疼了。感觉身边没人用呢，现在用的多吗？。只说优点，有什么缺点啊？



sg_vgt

121楼 2016-10-29 16:46发表

感觉很有用，至少适配不再头疼了。感觉身边没人用呢，现在用的多吗？。只说优点，有什么缺点啊？



sysuningwy

120楼 2016-10-10 21:21发表

鸿神，我在想谷歌没有写PercentLinearLayout是不是因为LinearLayout有了weight属性



胡锅

119楼 2016-09-08 16:17发表

请问博主现在自己在项目中采用那种适配方案？是你自己的那一套还是谷歌出的这个一套？

微信关注我的公众号

- 

十二月青

PercentLinearLayout、PercentFrameLayout、PercentRelativeLayout之间可以相互嵌套使用吗，就像LinearLayout里可以嵌套RelativeLayout，RelativeLayout里又可以嵌套LinearLayout。我有试过，没成功，不知道是方法不对还是就是不能？

118楼 2016-09-07 14:15发表
- 

十二月青

PercentLinearLayout、PercentFrameLayout、PercentRelativeLayout之间可以相互嵌套使用吗，就像LinearLayout里可以嵌套RelativeLayout，RelativeLayout里又可以嵌套LinearLayout。我有试过，没成功，不知道是方法不对还是就是不能？

117楼 2016-09-07
- 

qq_36045623

楼主威武，终于更加深入的了解适配问题了，希望楼主作品不断，鸿洋天下！！

116楼 2016-09-04
- 

qq_35528319

大神，请问gifview怎么做屏幕适配

115楼 2016-08-30 11:51发表
- 

qq_26856477

我在fragment 的适配器重写view时;view = LayoutInflater.from(getApplicationContext()).inflate(resourceId,null);

114楼 2016-08-25 19:16发表
- 

qq_26856477

113楼 2016-08-25 19:06发表
- 

qq_26856477

回复qq_26856477 : view = LayoutInflater.from(getApplicationContext()).inflate(resourceId,null);报错

Re: 2016-08-25 19:17发表
- 

qq_26856477

回复qq_26856477 : No static field PercentLayout_Layout_layout_aspectRatio of type I in class Landroid/support/percent/R\$styleable; or its superclasses (declaration of 'android.support.percent.R\$styleable' appears in

Re: 2016-08-25 19:17发表
- 

qyy530

低版本兼容时出问题,界面是白色的怎么办？

112楼 2016-07-12 13:41发表
- 

haowan123wode

就比如鸿神的 那张唐嫣图片， 现在 屏幕密度不变，高度不变的情况下， 宽度变成了 原来的两倍， 图片不就是直接拉伸了一倍么 完全失真了。 就算图片不是全屏的， 只要按上面 只有宽度发生变化， 图片宽度 按照百分比 分配的话 肯定是要拉伸的啊

111楼 2016-07-02 10:34发表
- 

haowan123wode

就比如鸿神的 那张唐嫣图片， 现在 屏幕密度不变，高度不变的情况下， 宽度变成了 原来的两倍， 图片不就是直接拉伸了一倍么 完全失真了。 就算图片不是全屏的， 只要按上面 只有宽度发生变化， 图片宽度 按照百分比 分配的话 肯定是要拉伸的啊

110楼 2016-07-02 10:35发表
- 

haowan123wode

图片是怎么适配啊， 就算是百分比 图片难道每个屏幕都要切一套么

109楼 2016-07-01 19:57发表

微信关注我的公众号



Martin-Rayman

108楼 2016-03-28 18:55发表

我想问下~乃们在Java中获取PercentLinearLayout.LayoutParams params = new PercentLinearLayout.LayoutParams(ViewGroup.LayoutParams.MATCH_PARENT, height);会不会报空指针呢？



xych45

107楼 2016-03-28 18:55发表

不得不佩服



陶的的

106楼 2016-03-28 18:55发表

洋神，亲测在使用中会出现1到2像素的误差，有没有解决办法，还是原本就是这样的？



江三沐洛

105楼 2016-03-17 17:40发表

布局写成style有什么方法吗？普通的写自定义属性的方法对这个不适合吧



江三沐洛

104楼 2016-03-17 17:40发表

对于listview这些，item中的宽高好像不适用这种方法啊



autonomousjack

103楼 2016-02-19 15:34发表

貌似对话框使用这种方式无效，这是为什么？



寫佩謫

102楼 2016-01-07 14:32发表

试试看，现在慢慢都转AS了



猴子下山写代码

Re: 2016-01-20 12:08发表

回复寫佩謫：为什么我导入库之后，有layout可以用，但是layout和里面的控件的属性不能设置啊，就是app:layout_heightPercent 之类的属性不能设置。



wuwh___

Re: 2016-02-01 10:11发表

回复猴子下山写代码：xmlns:app="http://schemas.android.com/apk/res-auto"



yubaoma

101楼 2016-01-04 20:08发表

mark。楼主牛逼。



maoxian007

100楼 2016-01-01 11:51发表

楼主 谁说 linerLayout的 weight 只是支持一个方向啊？？这句话 是不是有错误？



西装领带和我爱的姑娘

Re: 2016-01-07 22:14发表

回复maoxian007：LinearLayout的权重 要么高度 要么宽度 只能二选一 这样解释懂了么。。。



maoxian007

99楼 2016-01-01 11:49发表

回复东东车：吊了



老甩哥

98楼 2015-12-21 14:51发表

微信关注我的公众号

Error:(2) Attribute "layout_widthPercent" has already been defined???这是为何啊



lpslpsl

97楼 2015-12-18 11:38发表

用as导入库失败。。。就用的compile那句



鱼常晾

96楼 2015-12-14

你能不能说一下Eclipse如何集成这个库，为啥总说AS呢，你不知道现在还有很多人用Eclipse嘛，显得你很NB嘛？



老唐酸菜

Re: 2016-10-31 10:00发表

傻逼一样



酷菜

Re: 2016-08-22 22:19发表

回复鱼常晾：用别人写的东西 如果真的不懂 请以低姿态去请教别人 别人会很乐意告诉你的 你都用别人的东西 至少证明别人比你牛逼都什么时候了 既然还不学as 虽然我也是花了半年才适应as



凶残的程序员

Re: 2016-03-16 12:00发表

回复鱼常晾：为啥总坐动车呢，你不知道现在还有很多人坐火车吗？拿来主义不说谢谢就算了，Android 开发不用AS开发就算了，最后那句反问我是受不了



阿四大手法

Re: 2016-03-14 16:55发表

回复鱼常晾：你就不会查查怎么从as打jar包给Eclipse吗？也是搞笑



pkxutao

Re: 2015-12-16 17:46发表

回复鱼常晾：小蚂蚁对老鹰说，为什么你总是在天上飞呢，不在地上走呢，你不知道很多动物都是在地上走的吗，显得你很牛逼吗？



abner0914

Re: 2015-12-14 16:19发表

回复鱼常晾：那你就不会去学学as么？在这暴露你的无知！啧啧



squery

95楼 2015-12-03 09:53发表

这个由于是自定义控件，当我们在xml中搭建布局的时候preview无法显示预览效果，这个咋解决啊，不能写一点布局就在真机上测一下吧



下雨下雨下不停

Re: 2016-03-10 10:29发表

回复squery：哦知道了，要用全类名才可以



下雨下雨下不停

Re: 2016-03-10 09:47发表

回复squery：是啊，这是个问题，请问你的这个问题解决了么现在？



guider_

94楼 2015-11-28 22:16发表

大大您好 我是您微博的第4000位粉丝 么么哒



100*100的布局怎么写啊。。



zoomself

92楼 2015-11-22 21:51发表

今天看了你的RecyclerView的文章，然后又用了这个PercentRelativeLayout，发现原本可用的RecyclerView的item点击事点不可用了，换回原来的LinearLayout 又可以了，新手对于源码底层什么的确实看起来费劲，不知道大神能否给予答复，谢啦



qq_30658363

91楼 2015-11-12

如果用代码写布局，能用百分比布局写吗？怎么写呢，有人写过吗？写过的 加我qq,572961981,谢谢大家



不懂才觉得高深

90楼 2015-10-28

上次看了大神这篇文章,这个版本的app,直接让美工那边按百分比切了,今天又仔细看了一遍,感谢分享



恢兔子

89楼 2015-10-23 14:05发表

LinearLayout 有问题，里面的控件无法使用layout_gravity居中属性center不起作用



huangsco

88楼 2015-10-20 16:59发表

very good!



palatine

87楼 2015-10-14 17:34发表

多谢楼主,学习了



Android你好安卓

86楼 2015-10-13 14:28发表

java.lang.NoClassDefFoundError: Failed resolution of: Lcom/zyh/android/percent/support/R\$styleable;
eclipse里面怎么会报这个错？



小哈的秘密城堡

85楼 2015-09-22 11:16发表

我想问下，这个怎么用在项目里啊？下载下来例子不会用。



Lipt0n

84楼 2015-09-09 17:25发表

如果希望显示宽高为1:1 要怎么设置?还是要手动计算?



鸿洋_

Re: 2015-09-09 20:10发表

回复Lipt0n：详见：<http://blog.csdn.net/Imj623565791/article/details/46767825>



Lipt0n

83楼 2015-09-09 17:24发表

如果希望 宽和高是1:1显示有什么办法解决?需要计算后写入百分比?



不年少还青春

82楼 2015-09-07 14:49发表

java.lang.NoClassDefFoundError: android.support.v4.view.MarginLayoutParamsCompat

eclipse出现这个错，什么问题啊



Mr_Tony

Re: 2015-10-26 13:36发表

回复不年少还青春：v4包比较老，换个新的



阿四大手法

81楼 2015-09-04 17:46发表

导入这个包后，跟imageloader的包冲突，imageloader不能用了。这是怎么回事？？？



鸿洋_

Re: 2015-09-04

回复阿四大手法：自己检查下，这个情况是不可能的。



阿四大手法

80楼 2015-09-04 17:25发表

今天载了新的包，在里面看到了PercentLinearLayout文件。



老夫cxd

Re: 2017-05-27 15:14发表

回复阿四大手法：没有哇，我刚刚看了，没有，不要误导网友



kqw攻城狮

79楼 2015-09-01 15:07发表

这种方式如果使用的图片的话应该会拉伸把，怎么解决呢



cl12821

78楼 2015-08-27 17:20发表

请问下，eclipse要使用这个百分比库，需要怎么配置？



jjj930901

77楼 2015-08-20 11:26发表

Process 'command 'C:\Program Files\Java\jdk1.8.0_05\bin\java.exe' finished with non-zero exit value 2

我在 compile 'com.android.support.percent:22.2.0' 之后运行程序会出现这个错误其他地方我也没有compile percent这是为什么~~~



jjj930901

76楼 2015-08-20 11:26发表

Error:Execution failed for task ': Process 'command 'C:\Program Files\Java\jdk1.8.0_05\bin\java.exe' finished with non-zero exit value 2

我在 compile 'com.android.support.percent:22.2.0' 之后运行程序会出现这个错误其他地方我也没有compile percent这是为什么~~~



Jokeeeeeee

75楼 2015-08-05 17:44发表

mark



JabariWu

74楼 2015-08-05 11:43发表

通过大量资料分析得出以下结论：
翔哥喜欢唐嫣，Trinea爱陈乔恩！
膜拜哥吧，哈哈



u010523832

73楼 2015-08-03 12:33发表

没有什么卵用，layout 只能用，等到像html 那样的百分比再说，各种控件都能用



renshaodong

72楼 2015-07-29 16:27发表

没有jar包呢

微信关注我的公众号



i_a_jun

71楼 2015-07-27 15:55发表

大神你好！
我在<https://github.com/hongyangAndroid/android-percent-support-extend>这个路径下没有看到eclipse项目。都是android studio项目。



一树梨花Pw

70楼 2015-07-16 14:06发表

这回面试可以好好回答了，膜拜！



Line_WH

69楼 2015-07-16 14:06发表

'也就是说，你去打印上面写法，布局文件中view的v.getLayoutParams().width，这个值应该是0。

这里感觉略微不爽~这个0没撒用处呀，还不如不重置~~

如果不重置，在父视图重绘的时候会导致叠加计算percent，导致值不正确！



zy_zhangyuan88

68楼 2015-07-09 17:46发表

真牛逼，大神



XD---SJS

67楼 2015-07-07 23:45发表

给大神跪了，打算认真看完大神所有博客，谢谢分享!



p1aywind

66楼 2015-07-07 19:53发表

大神就是大神，游戏撸码两不误



beckett1216

65楼 2015-07-07 14:48发表

转载到一个论坛里注明出处可以不？楼主



Yang_Farley

64楼 2015-07-06 21:33发表

Mark

查看更多评论

发表评论

用户名： xfh_y_

评论内容：



提交

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场