

MICROOH 麦可网

Android-从程序员到架构师之路

出品人：Sundy

讲师：高焕堂（台湾）

<http://www.microoh.com>

A08_c

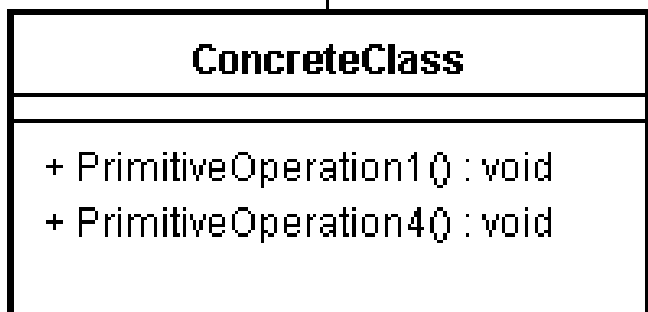
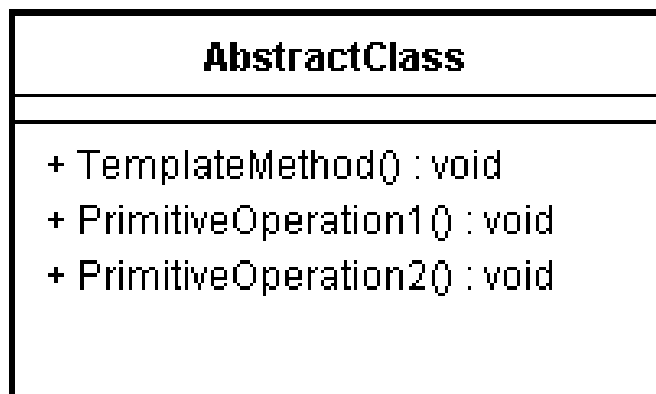
简介设计模式(c)

By 高煥堂

5、GoF的 Template Method模式

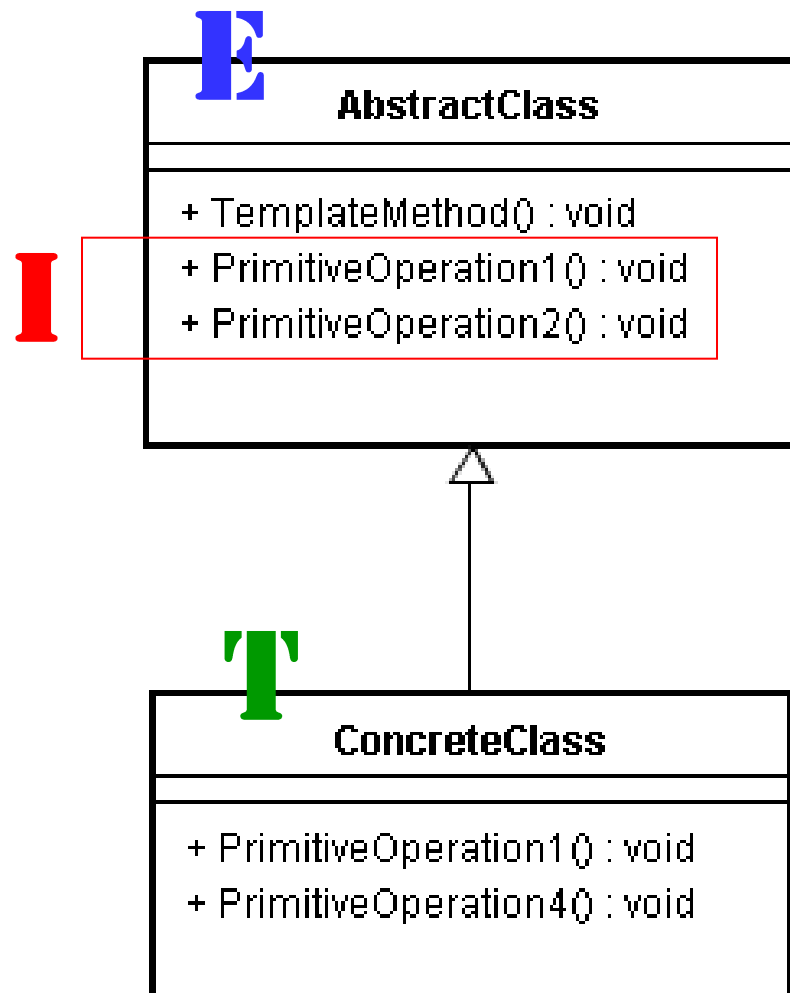
Template Method模式

- 在前面各节里，我们介绍过，控制反转(IoC: Inversion of Control)是<基类/子类>结构里的重要机制。Template Method模式是实现IoC的一种基本模式。

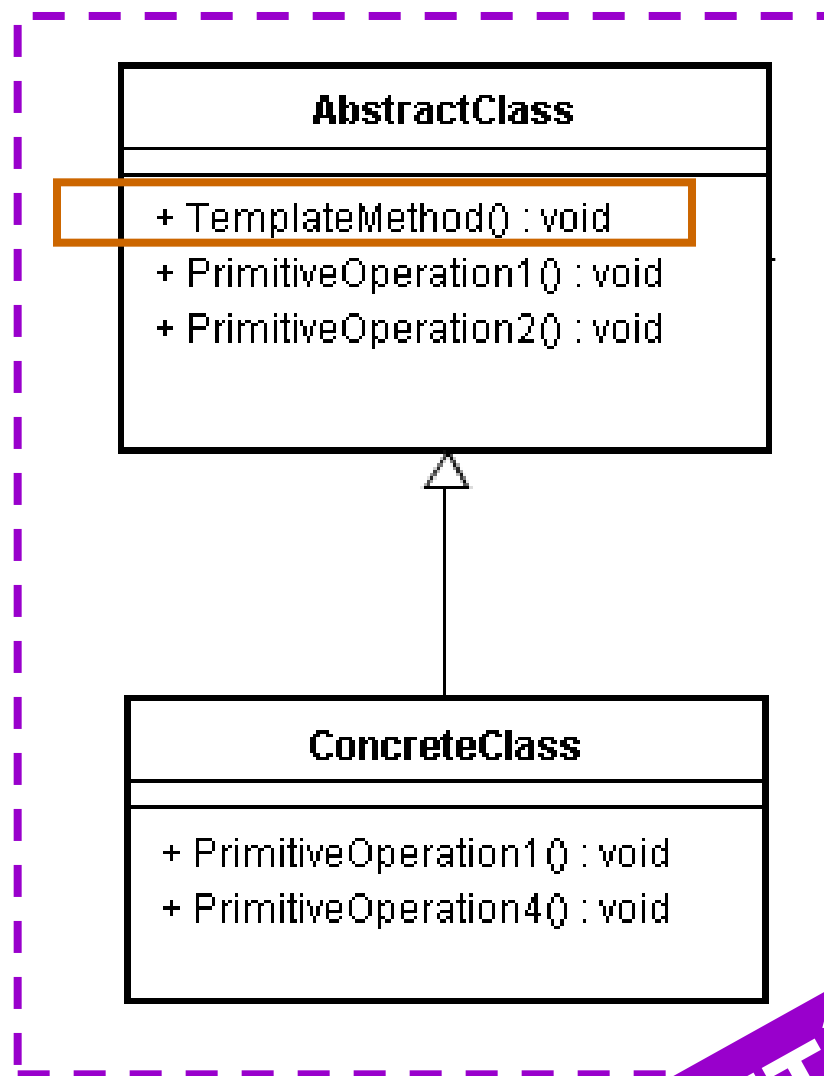


```
TemplateMethod(){  
.....  
PrimitiveOperation1();  
.....  
PrimitiveOperation2();  
..... }
```

GoF的TM模式

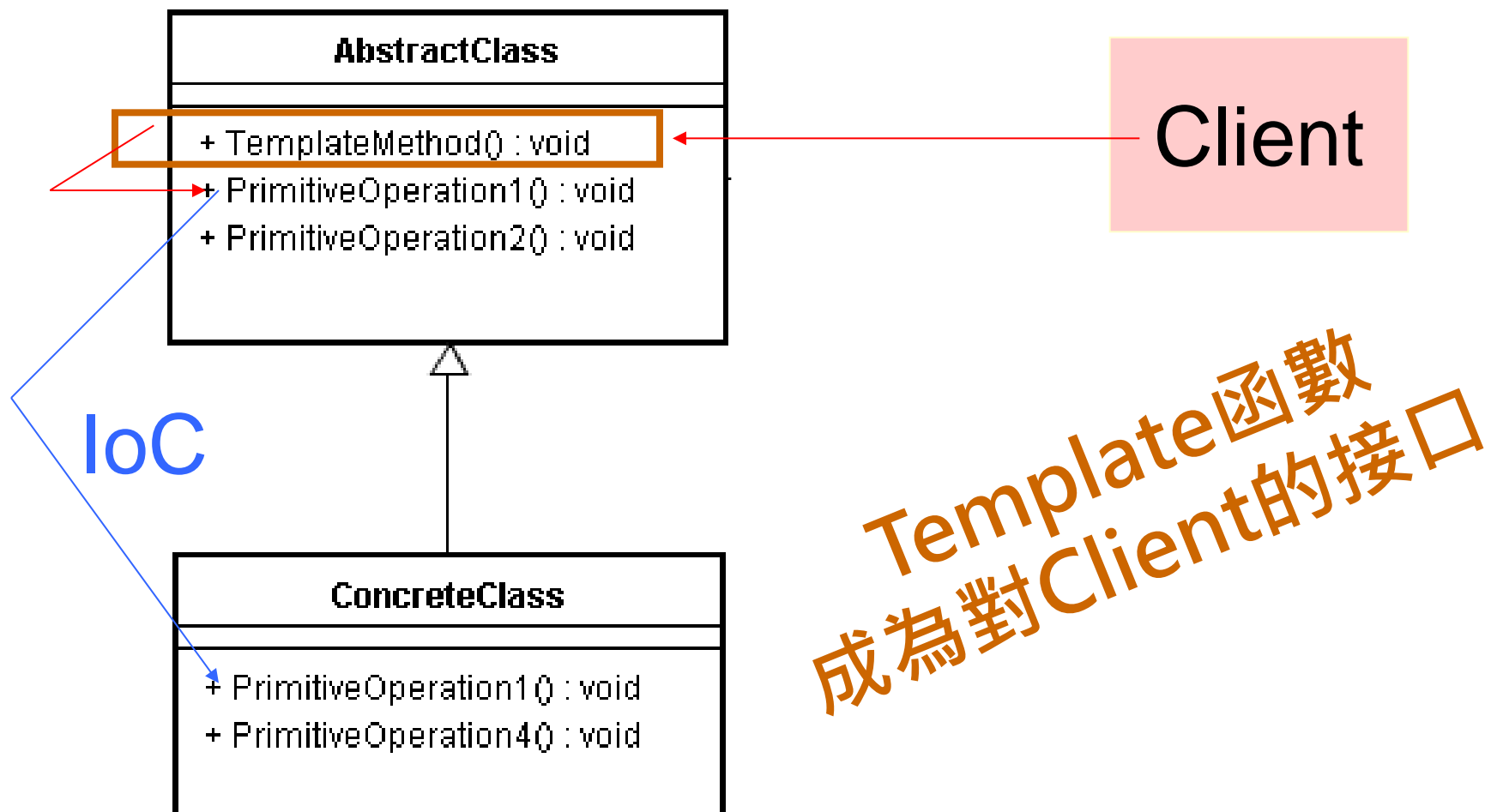


它是一個EIT造形



加上一個
Template函數

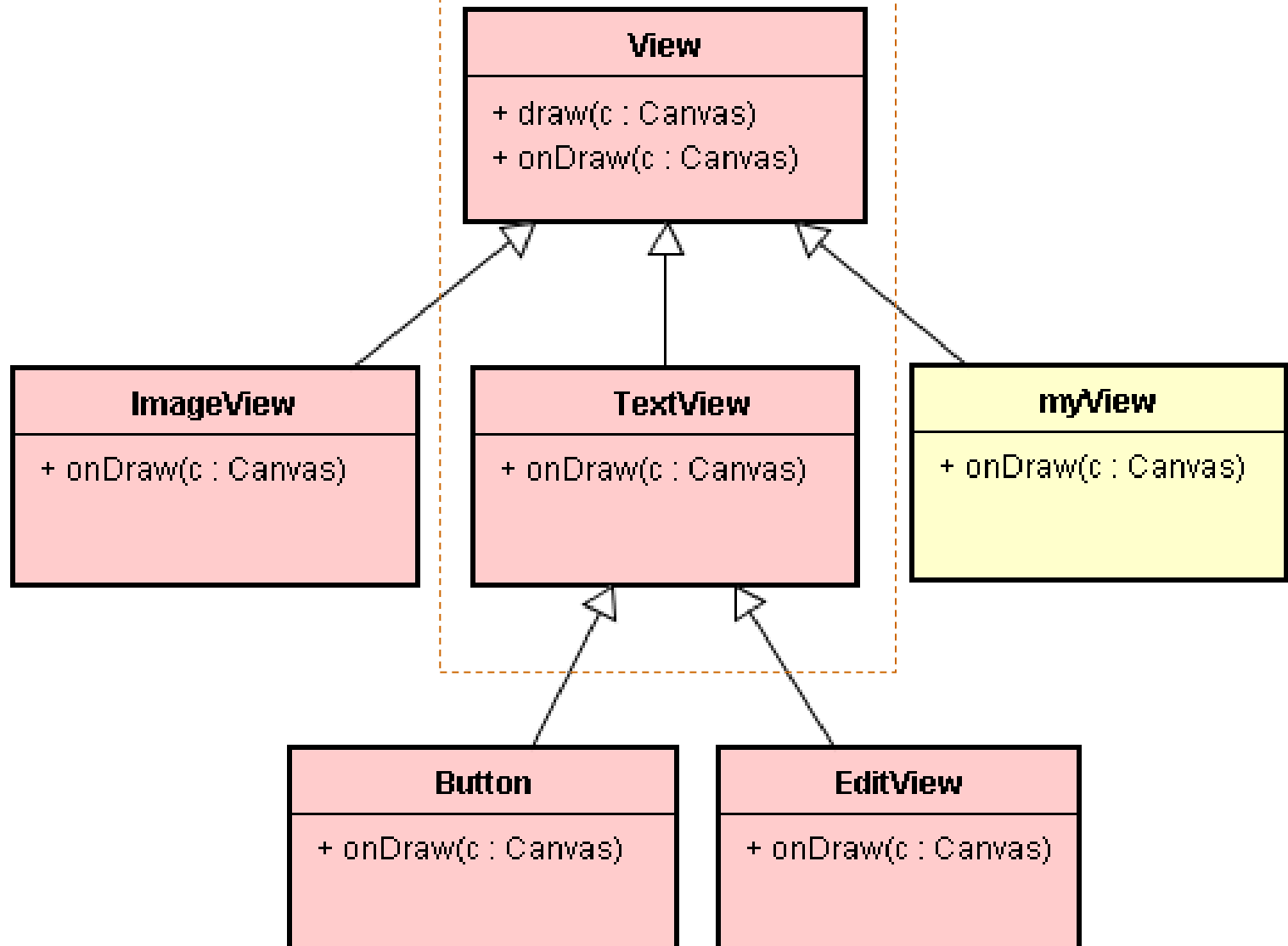
EIT造形



6、范例： Android + TM模式

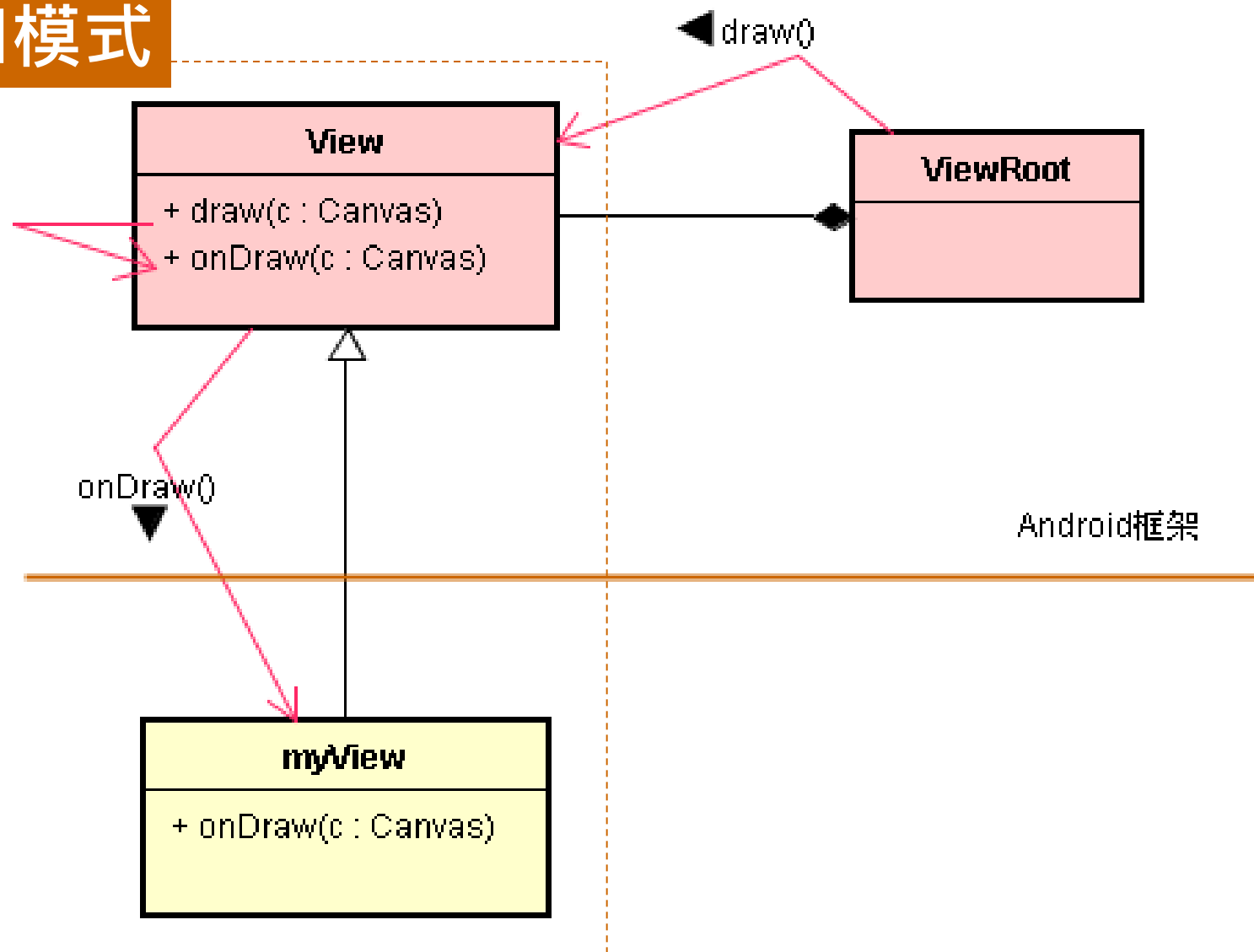
- Android的绘图是使用画布(Canvas)来把图显示于View的窗口里，并且从View类别而衍生子类别，提供更多功能来将图形或图片绘制于画布上。

TM模式



- 在View类别里有个onDraw()函数，View类别体系里的每一个类别都必须覆写(Override)这个onDraw()函数，来执行实际绘图的动作。

TM模式



```
// myView.java
```

```
//.....
```

```
public class myView extends View {
```

```
    private Paint paint= new Paint();
```

```
    private int line_x = 100, line_y = 100;
```

```
    private float count = 0;
```

```
myView(Context ctx) { super(ctx); }
```

```
@Override protected void onDraw(Canvas canvas) {
```

```
    super.onDraw(canvas);
```

```
    if( count > 12)
```

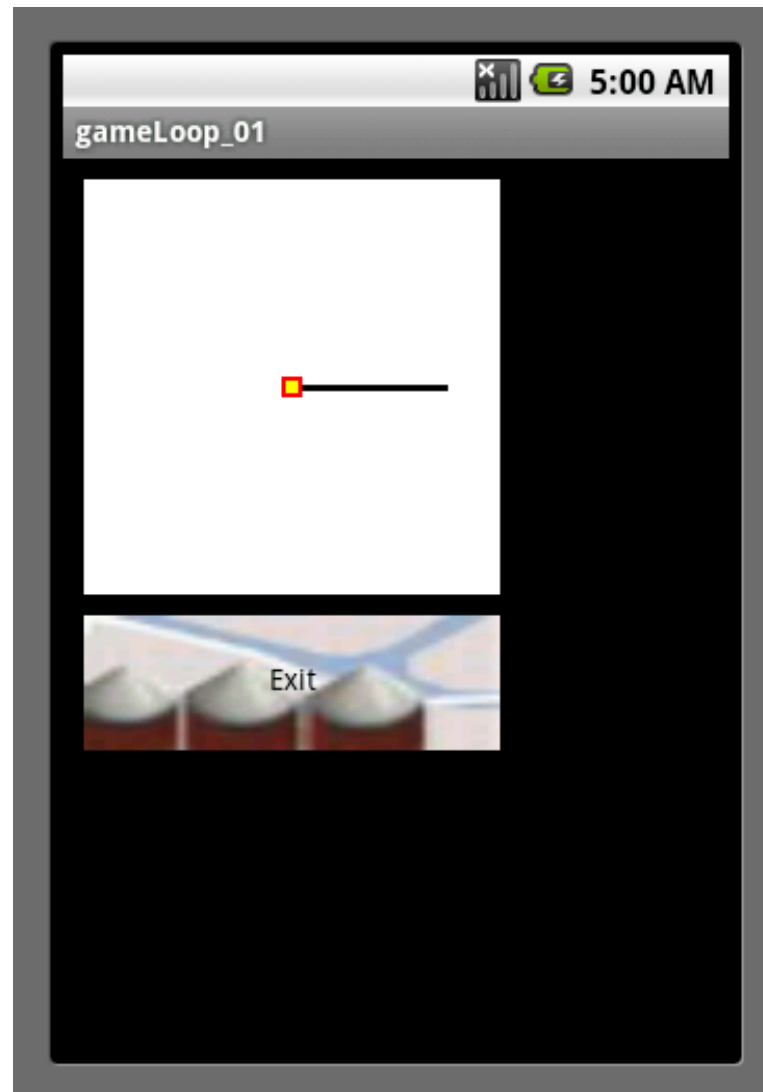
```
        count = 0;
```

```
    int x = (int) (75.0 * Math.cos(2*Math.PI * count/12.0));
```

```
    int y = (int) (75.0 * Math.sin(2*Math.PI * count/12.0));
```

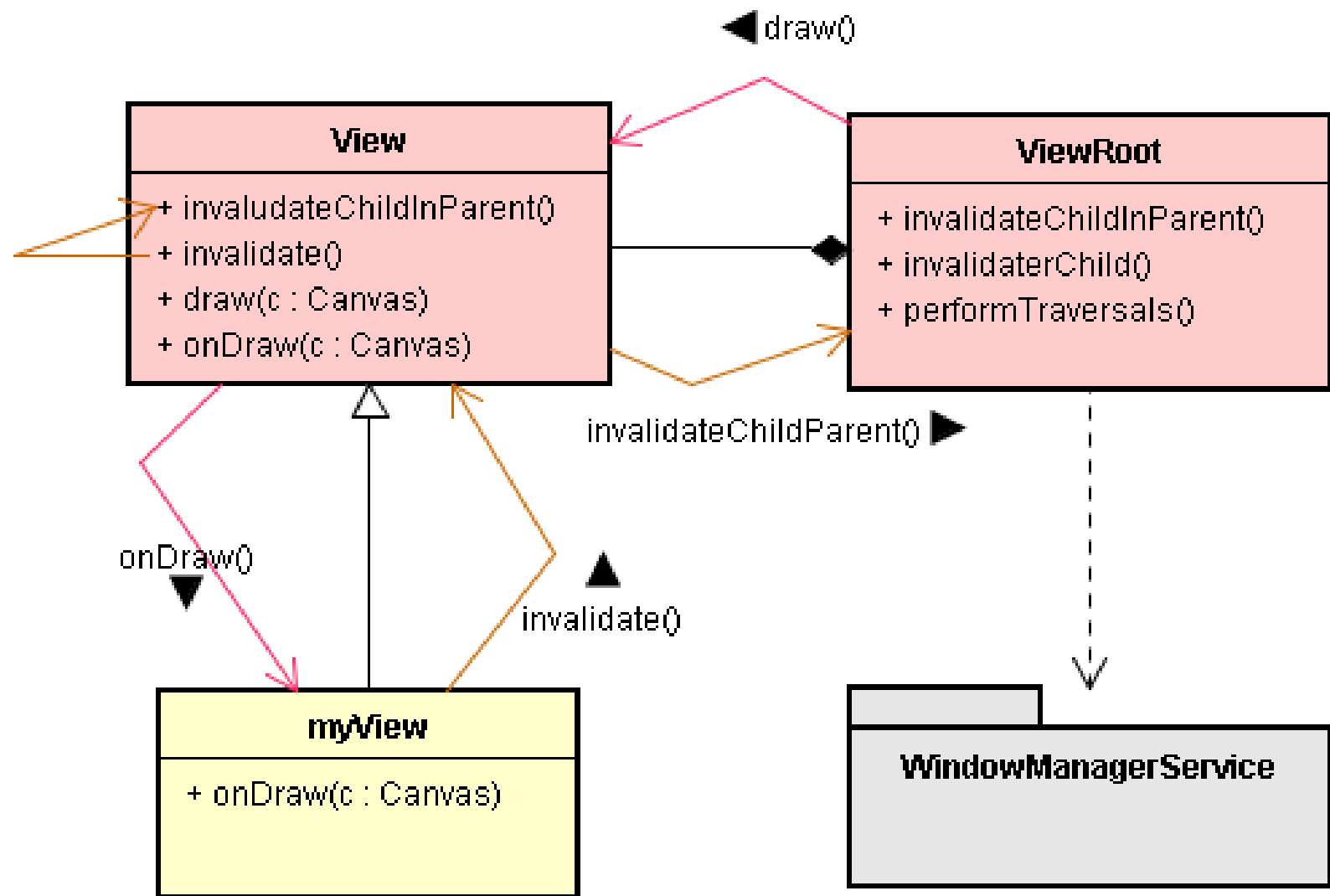
```
    count++;
```

```
canvas.drawColor(Color.WHITE);
paint.setColor(Color.BLACK);
paint.setStrokeWidth(3);
canvas.drawLine(line_x, line_y, line_x+x, line_y+y, paint);
paint.setStrokeWidth(2);
paint.setColor(Color.RED);
canvas.drawRect(line_x-5, line_y - 5,
                line_x+5, line_y + 5, paint);
paint.setColor(Color.YELLOW);
canvas.drawRect(line_x-3, line_y - 3, line_x+3,
                line_y + 3, paint);
    }
}
```



基于TM模式的扩充： 以游戏的绘图循环(Game Loop)为例

- 游戏的基本动作就是不断的进行：绘图和刷新(Refresh)画面。其中，onDraw()函数实践画图，将图形绘制于View的画布(Canvas)上，并显示出来；而invalidate()函数则启动画面的刷新，重新呼叫一次onDraw()函数。



```
// myView.java
```

```
// .....
```

```
public class myView extends View {  
    private Paint paint= new Paint();  
    private int line_x = 100, line_y = 100;
```

```
myView(Context ctx) { super(ctx); }
```

```
@Override protected void onDraw(Canvas canvas) {
```

```
    super.onDraw(canvas);
```

```
    //-----
```

```
    if( count > 12) count = 0;
```

```
    int x = (int) (75.0 * Math.cos(2*Math.PI * count/12.0));
```

```
    int y = (int) (75.0 * Math.sin(2*Math.PI * count/12.0));
```

```
    count++;
```

```
    //-----
```

```
canvas.drawColor(Color.WHITE);  
paint.setColor(Color.BLACK);  
paint.setStrokeWidth(3);  
canvas.drawLine(line_x, line_y, line_x+x, line_y+y, paint);
```

```
paint.setStrokeWidth(2);  
paint.setColor(Color.RED);  
canvas.drawRect(line_x-5, line_y - 5, line_x+5, line_y + 5, paint);  
paint.setColor(Color.YELLOW);  
canvas.drawRect(line_x-3, line_y - 3, line_x+3, line_y + 3, paint);
```

```
try {  
    Thread.sleep(1000);  
}catch (InterruptedException ie) {}
```

```
invalidate();
```

```
}
```

```
}
```

```
// myActivity.java
```

```
// .....
```

```
public class myActivity extends Activity  
                                implements OnClickListener {
```

```
    private myView mv = null;
```

```
    private Button ibtn;
```

```
@Override
```

```
protected void onCreate(Bundle icle) {
```

```
    super.onCreate(icle);
```

```
    public void show_layout_01(){
```

```
        LinearLayout layout = new LinearLayout(this);
```

```
        layout.setOrientation(LinearLayout.VERTICAL);
```

```
        mv = new myView(this);
```

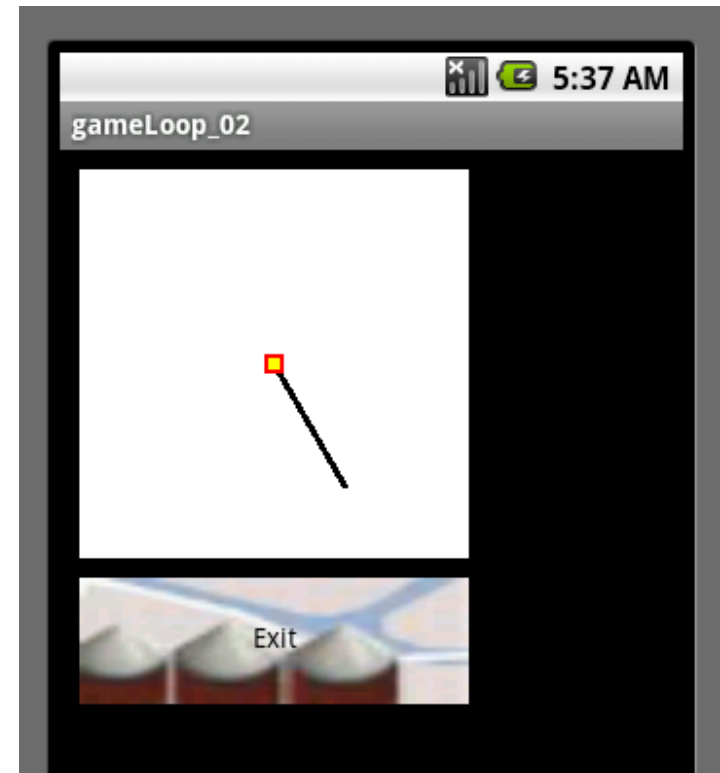
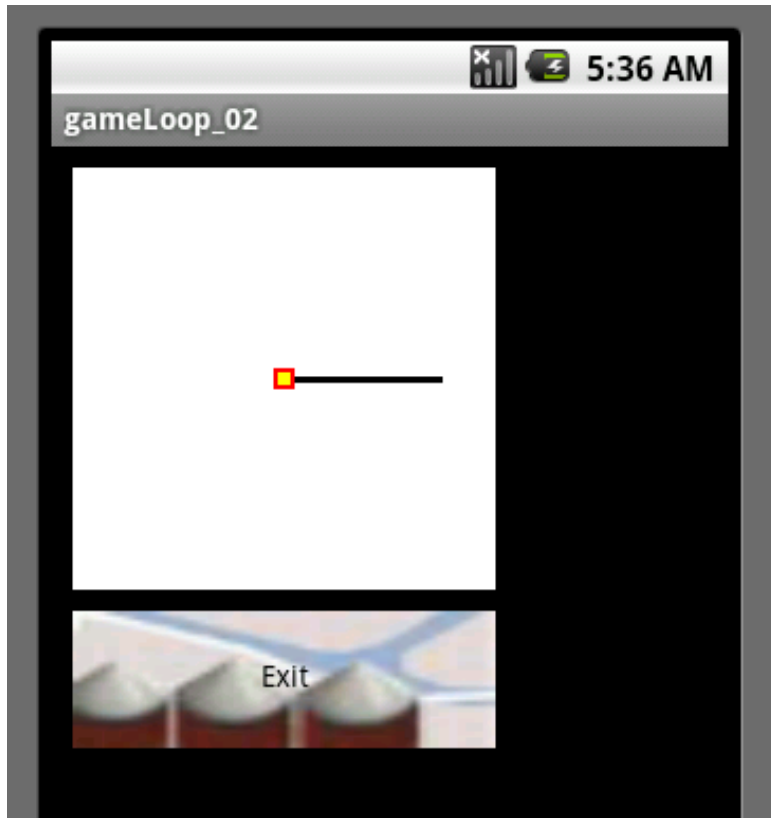
```
        LinearLayout.LayoutParams param =
```

```
            new LinearLayout.LayoutParams(200, 200);
```

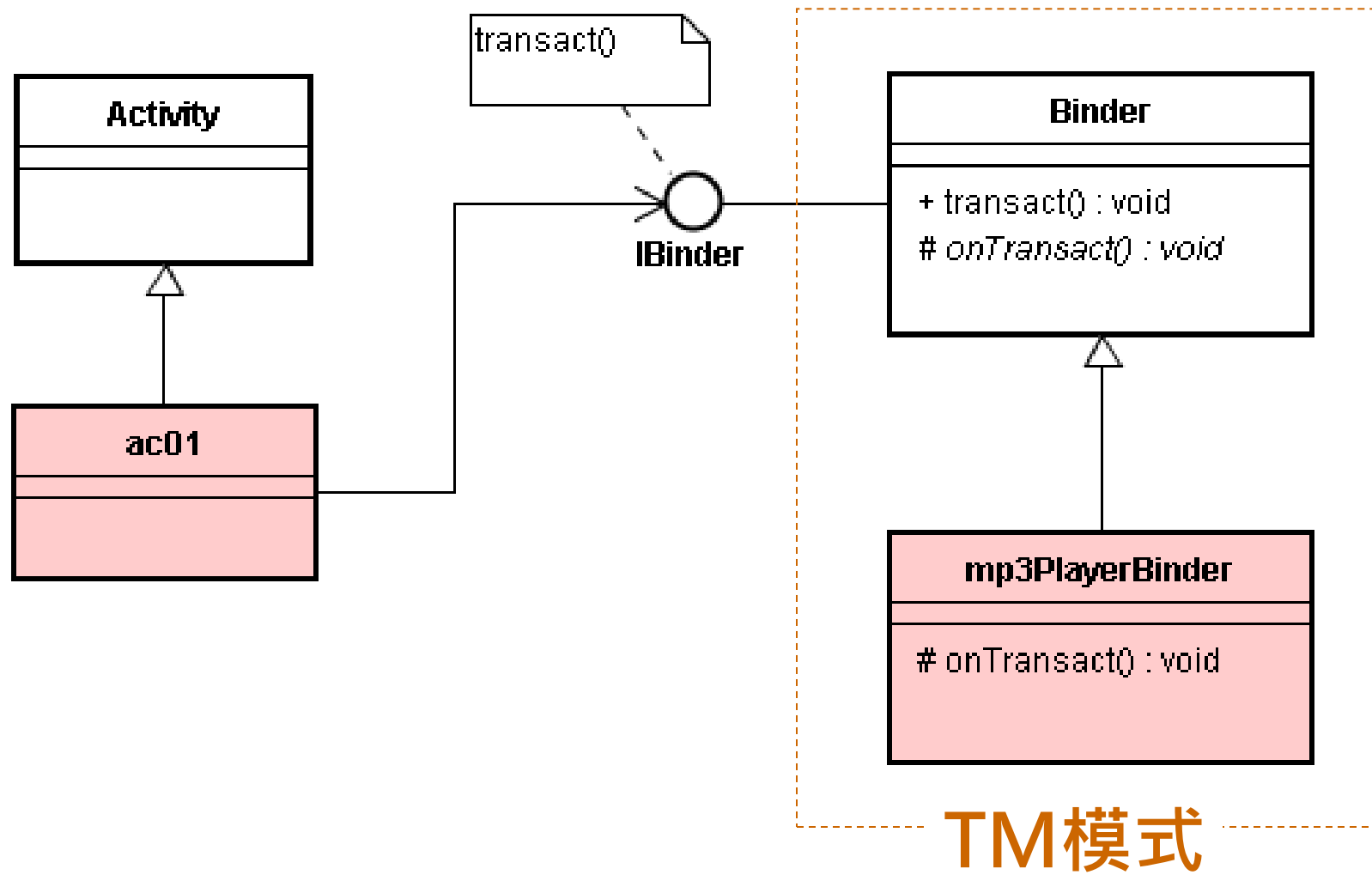
```
        param.topMargin = 10;    param.leftMargin = 10;
```

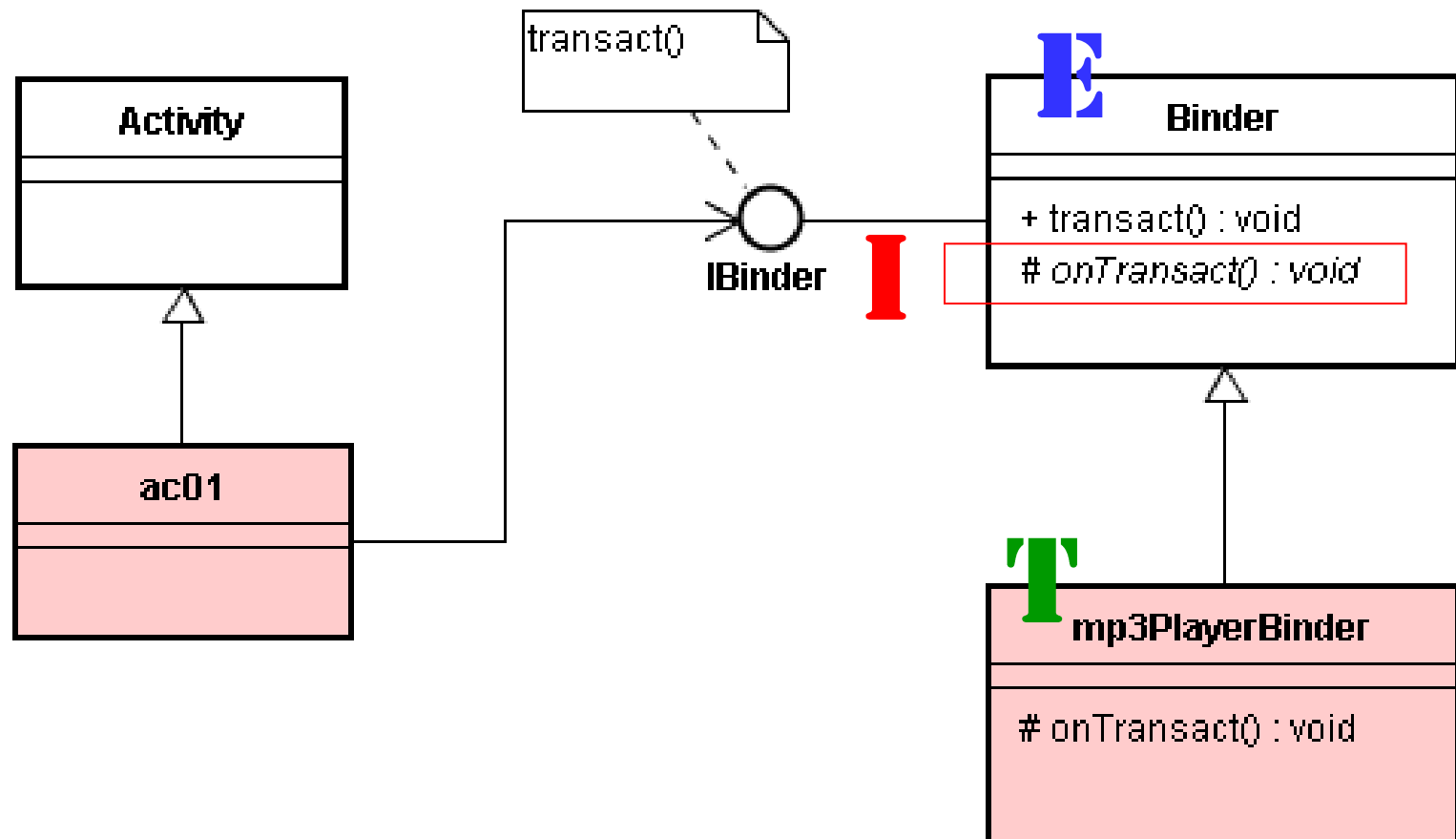
```
        layout.addView(mv, param);
```

```
ibtn = new Button(this);
ibtn.setOnClickListener(this);
ibtn.setText("Exit");
ibtn.setBackgroundResource(R.drawable.gray);
LinearLayout.LayoutParams param1 =
    new LinearLayout.LayoutParams(200, 65);
param1.topMargin = 10; param1.leftMargin = 10;
layout.addView(ibtn, param1);
//-----
setContentView(layout);
}
public void onClick(View v) {
    finish();
}
}
```



Android里处处可见
TM模式的应用







~ Continued ~