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

B03_a

应用Android的UI框架(a)

-- 以设计游戏循环(GameLoop)为例

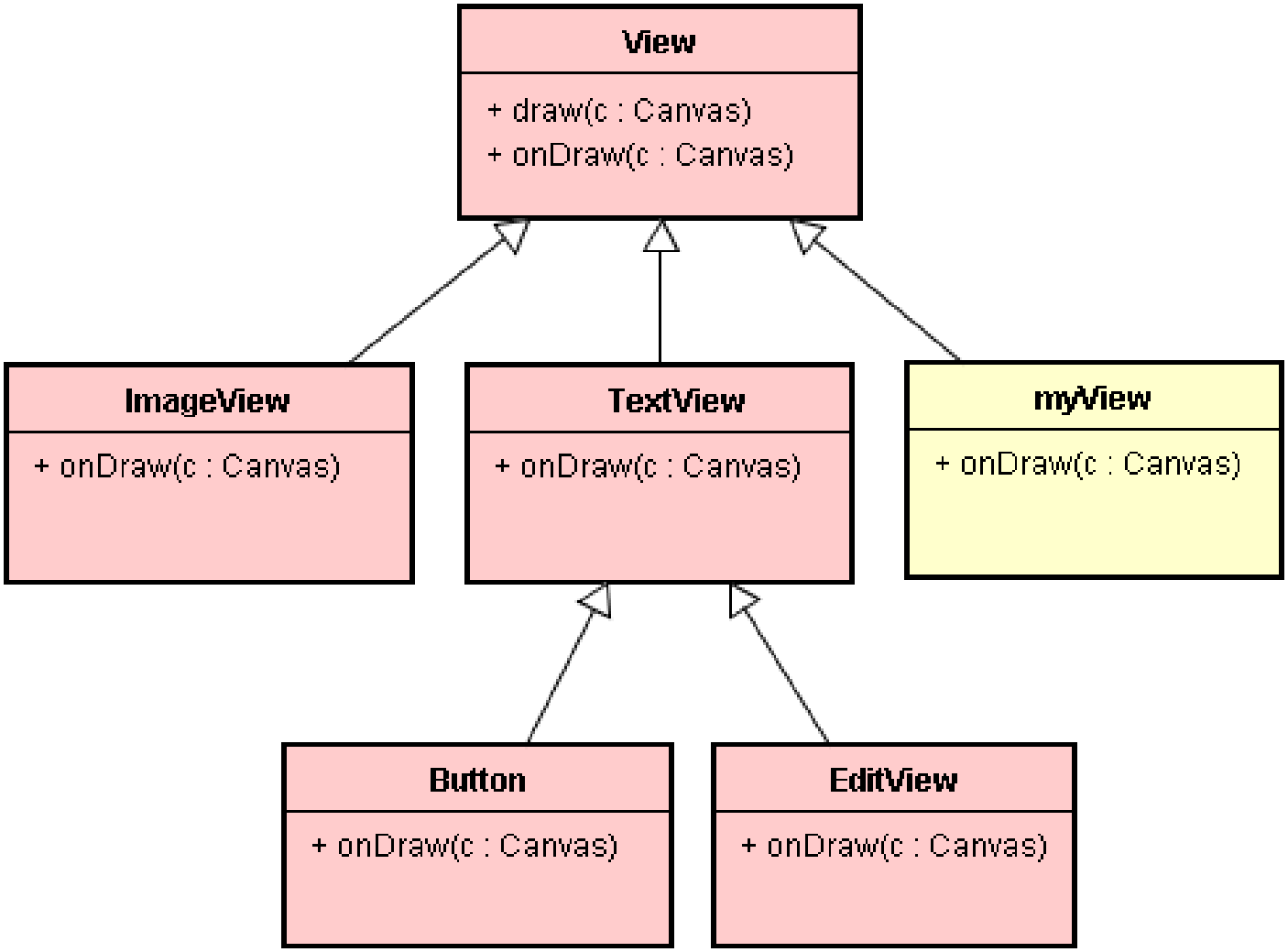
By 高煥堂

内容

1. UI线程、View与onDraw()函数
2. 基本游戏循环(GameLoop)
3. 使用UI线程的MQ(Message Queue)
4. 诞生一个小线程，担任游戏线程
5. 小线程調用postInvalidate()
6. 设计一个GameLoop类别
7. 只诞生一次GameLoop对象

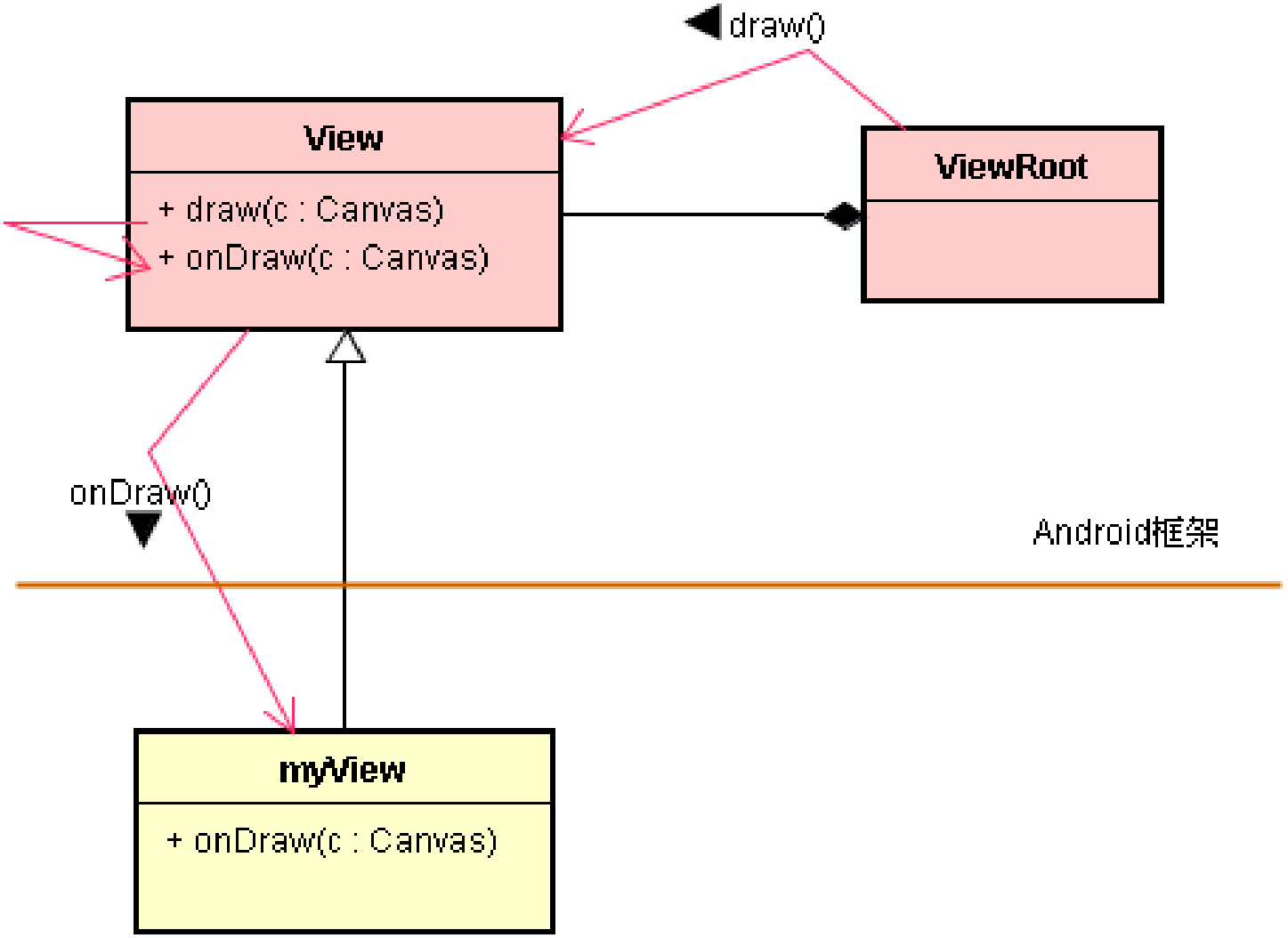
1、UI线程、View与onDraw()函数

- 游戏的UI画面通常是由大量美工贴图所构成的，并不会使用一般的Layout来布局，而是使用画布(Canvas)来把图片显示于View的窗口里。
- 在View类里有个onDraw()函数，View类体系里的每一个类都必须覆写(Override) 这个onDraw()函数，来执行实际绘图的动作。


View
+ draw(c : Canvas)
+ onDraw(c : Canvas)
ImageView
+ onDraw(c : Canvas)
TextView
+ onDraw(c : Canvas)
myView
+ onDraw(c : Canvas)
Button
+ onDraw(c : Canvas)
EditView
+ onDraw(c : Canvas)

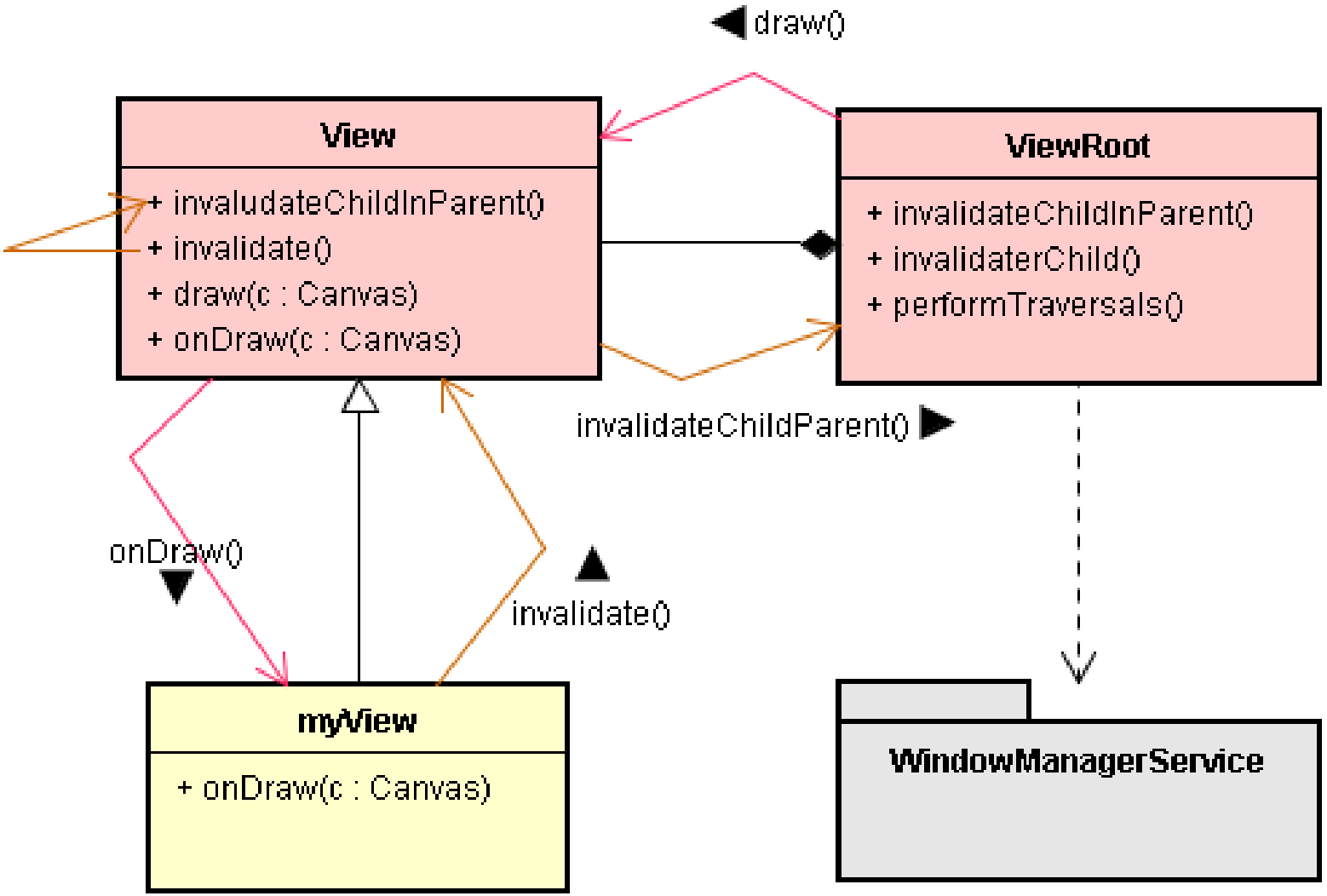
- 游戏的基本动作就是不断的进行：绘图和刷新(Refresh)画面。其中，onDraw()函数实践画图，将图形绘制于View的画布(Canvas)上，并显示出来；而invalidate()函数则启动画面的刷新，重新調用一次onDraw()函数。

- 当我们设计myView子类别时，也必须覆写onDraw()函数。在程序执行时，Android框架会进行反向調用到myView的onDraw()函数来进行画图动作。如下图：


draw()
View
+ draw(c : Canvas)
+ onDraw(c : Canvas)
ViewRoot
onDraw()
Android框架
myView
+ onDraw(c : Canvas)

2、基本游戏循环(GameLoop)

- 游戏的基本动作就是不断的绕回圈(Loop)，重复绘图和刷新画面的动作。最简单的循环实现方式是：在onDraw()函数里調用invalidate()函数，就能刷新画面(重新調用一次onDraw()函数)了。

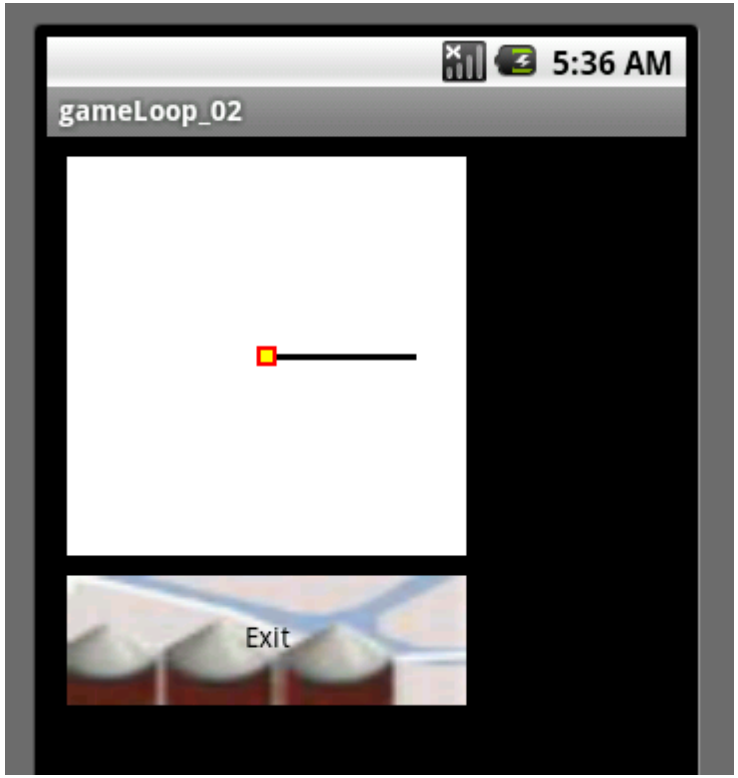

draw()
View
+ invaludateChildInParent()
+ invalidate()
+ draw(c : Canvas)
+ onDraw(c : Canvas)
ViewRoot
+ invalidateChildInParent()
+ invalidaterChild()
+ performTraversals()
invalidateChildParent()
onDraw()
invalidate()
myView
+ onDraw(c : Canvas)
WindowManagerService

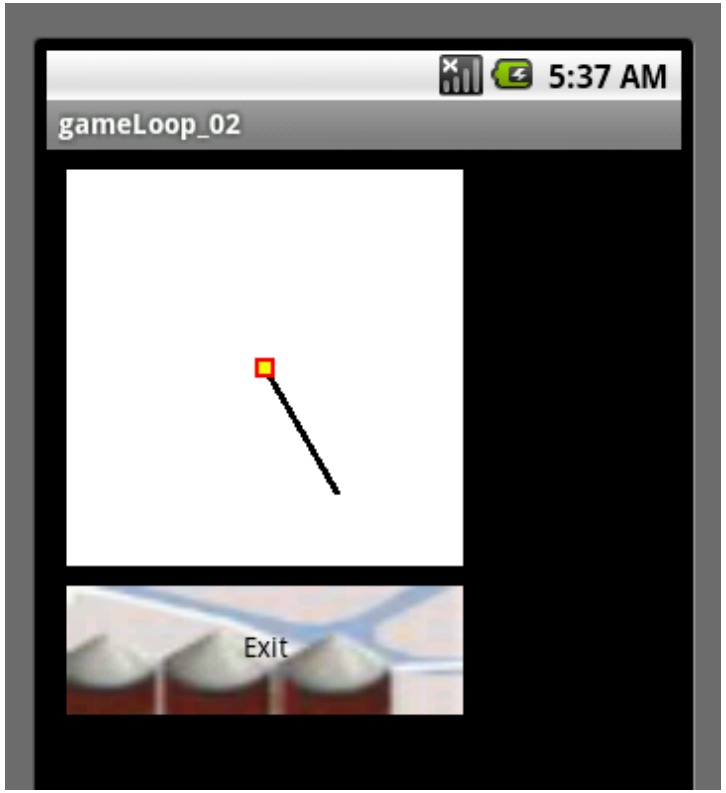
```
// myView.java
// .........
public class myView extends View {
    private Paint paint= new Paint();
    private int line_x = 100, line_y = 100;
    private float count = 0;

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

- Android中提供了invalidate()来实现画面的刷新：即触发框架重新执行onDraw()函数来绘图及显示。


5:36 AM
gameLoop_02
Exit


5:37 AM
gameLoop_02
Exit

~ Continued ~