

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_d

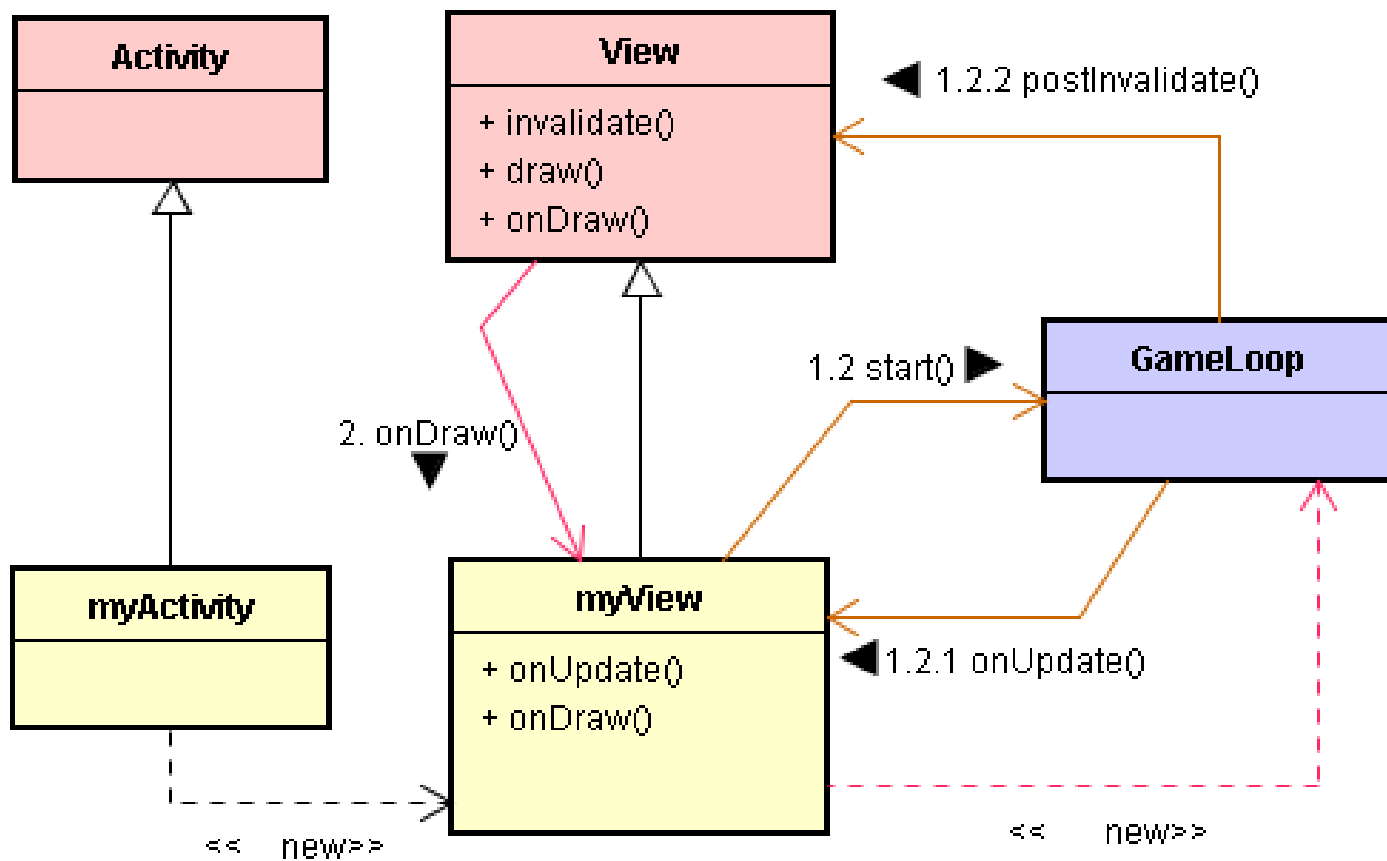
应用Android的UI框架(d)

-- 以设计游戏循环(*GameLoop*)为例

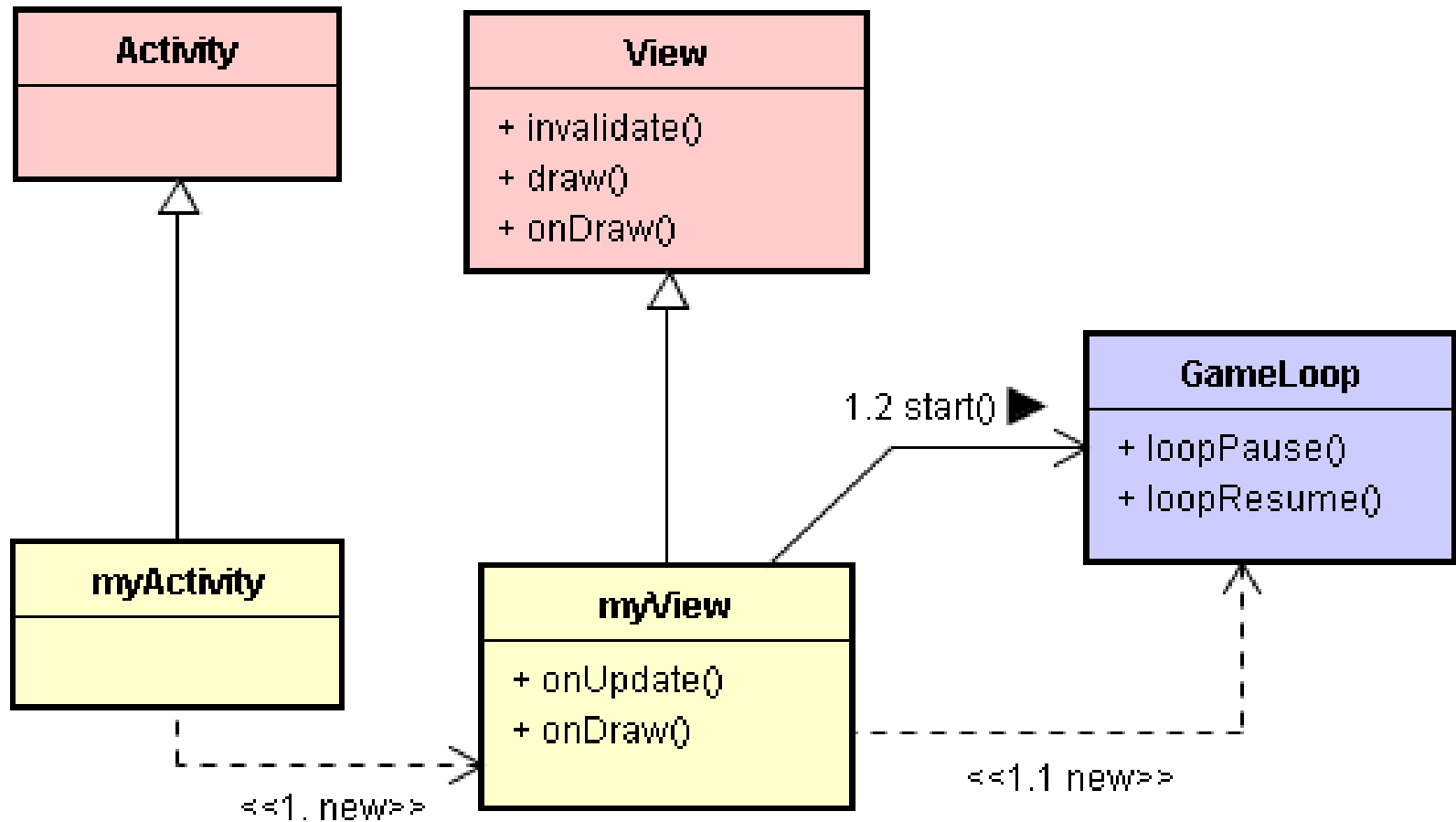
By 高焕堂

6、只诞生一次 GameLoop对象

复习：

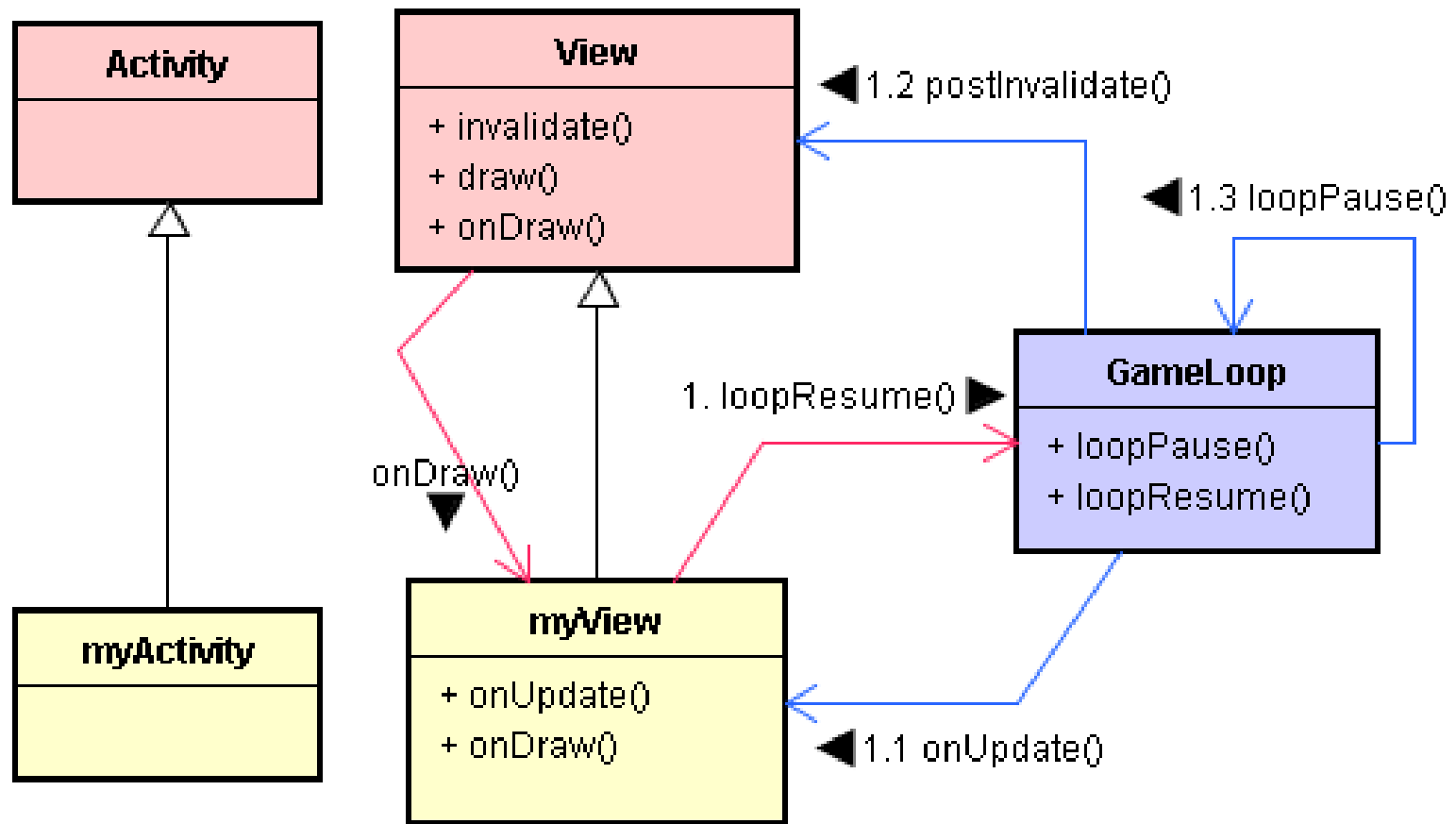


- 每次执行onDraw()时，都会重新诞生一次GameThread对象，也诞生一次游戏线程去调用postInvalidate()函数。似乎是UI线程控制着游戏线程，这样游戏线程就不能扮演主控者的角色了。
- 于是，可换一个方式：一开始先诞生一个游戏线程，并且使用while(true)来创建一个无限循环(Endless Loop)，让游戏线程持续绕回圈，而不会停止。



- 在诞生myView时，就诞生GameLoop对象，且调用其start()函数来启动游戏线程。此时游戏线程处于<暂停>状态，虽然继续绕回圈，但是并不会调用postInvalidate()函数。接着，由Android框架调用myView的onDraw()函数来绘图和显示。

- 绘图完毕，立即调用GameLoop的loopResume()函数，让GameLoop从<暂停>状态转移到<执行>状态。此时，这游戏线程就去调用postInvalidate()函数，触发UI线程重新调用myView的onDraw()函数。如下图：



```
// GameLoop.java
```

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

```
public class GameLoop extends Thread {  
    private myView mView;  
    private boolean bRunning;  
  
    GameLoop(myView v){  
        mView = v; bRunning = false; }  
    public void run() {  
        while(true){  
            if(bRunning){  
                mView.onUpdate();  
                mView.postInvalidateDelayed(1000);  
                loopPause();  
            }  
        }  
    }  
    public void loopPause(){          bRunning = false; }  
    public void loopResume(){        bRunning = true; }  
}
```

- 其中，loopPause()函数将bRunning设定为false，游戏线程就处于<暂停>状态。loopResume()函数将bRunning设定为true，游戏线程就处于<执行>状态，就会调用myView的onUpdate()函数，去更新绘图的设定。
- 然后调用postInvalidate()函数，触发UI线程去重新调用onDraw()函数。

```
// myView.java
```

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

```
public class myView extends View {  
    private Paint paint= new Paint();  
    private int x, y;  
    private int line_x = 100, line_y = 100;  
    private float count;  
    private GameLoop loop;
```

```
    myView(Context ctx) {  
        super(ctx);  
        init();  
        loop = new GameLoop(this);  
        loop.start();  
    }
```

```
public void init(){  
    count = 0;  
    x = (int) (75.0 * Math.cos(2*Math.PI * count/12.0));  
    y = (int) (75.0 * Math.sin(2*Math.PI * count/12.0));  
}  
public void onUpdate(){ // 游戏线程执行的  
    if( count > 12) count = 0;  
    x = (int) (75.0 * Math.cos(2*Math.PI * count/12.0));  
    y = (int) (75.0 * Math.sin(2*Math.PI * count/12.0));  
    count++;  
}
```

```
@Override protected void onDraw(Canvas canvas) {  
    super.onDraw(canvas);  
    canvas.drawColor(Color.WHITE);  
    paint.setColor(Color.BLUE);  
    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.CYAN);  
    canvas.drawRect(line_x-3, line_y - 3, line_x+3,  
                    line_y + 3, paint);  
    //-----  
    loop.loopResme();  
}  
}
```

- 请注意：onUpdate()函数是由游戏线程所执行的；而onDraw()则是由UI线程所执行的。

Thanks...



高煥堂