

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>

A05

# 认识EIT造形与内涵

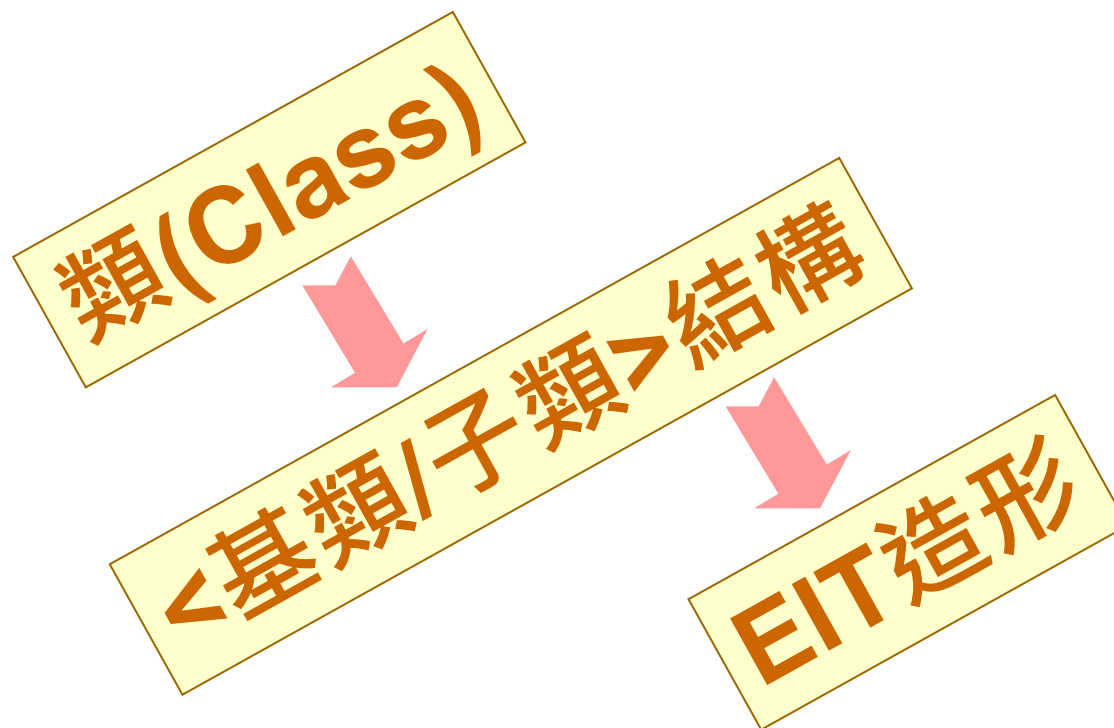
By 高煥堂

# 内容

1. 认识类(Class)与内涵
2. 认识EIT造形与内涵
3. 结语

# 1、认识类(Class)与内涵

# 回顾

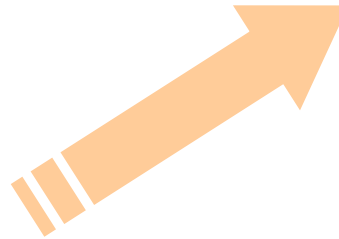
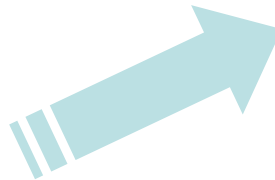
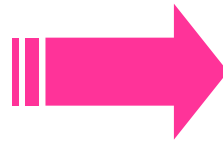
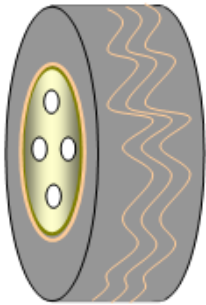


# 类(Class)是比较小的造形

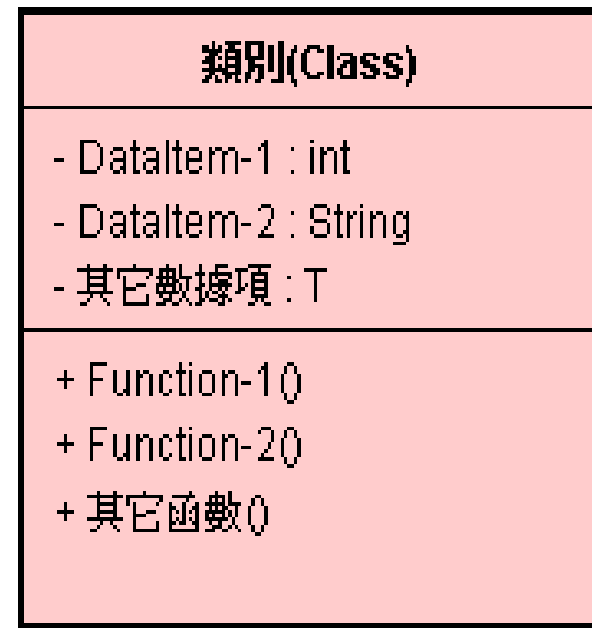
類別(Class)
<ul style="list-style-type: none"><li>- DataItem-1 : int</li><li>- DataItem-2 : String</li><li>- 其它數據項 : T</li></ul>
<ul style="list-style-type: none"><li>+ Function-1()</li><li>+ Function-2()</li><li>+ 其它函數()</li></ul>

- 各种不同内涵，可以透过类(Class)的形式(Form)来呈现出来。
- 类造形就像集装箱(Container)，可以容纳各种内涵。
- 集装箱的形式大多与内涵无关。

# 不同的內涵



# 类(Class)造形







內涵

+

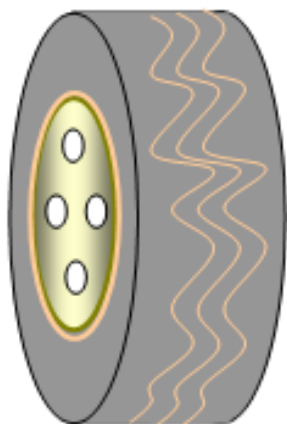
類別(Class)
- DataItem-1 : int - DataItem-2 : String - 其它數據項 : T
+ Function-1() + Function-2() + 其它函數()

類造形

=

Jeep車類

Jeep車
- 主人名稱 : char - 廠牌 : char
+ 啟動() : void + 響喇叭() : void + Others() : void



+

類別(Class)
- DataItem-1 : int - DataItem-2 : String - 其它數據項 : T
+ Function-1() + Function-2() + 其它函數()

類造形

內涵

||

Jeep輪胎類

Jeep輪胎
- 價錢 : float - 廠牌型號 : int - 其它 : void
+ 充氣() : void + 其它() : void



內涵

+

類別(Class)
- DataItem-1 : int - DataItem-2 : String - 其它數據項 : T
+ Function-1() + Function-2() + 其它函數()

類造形

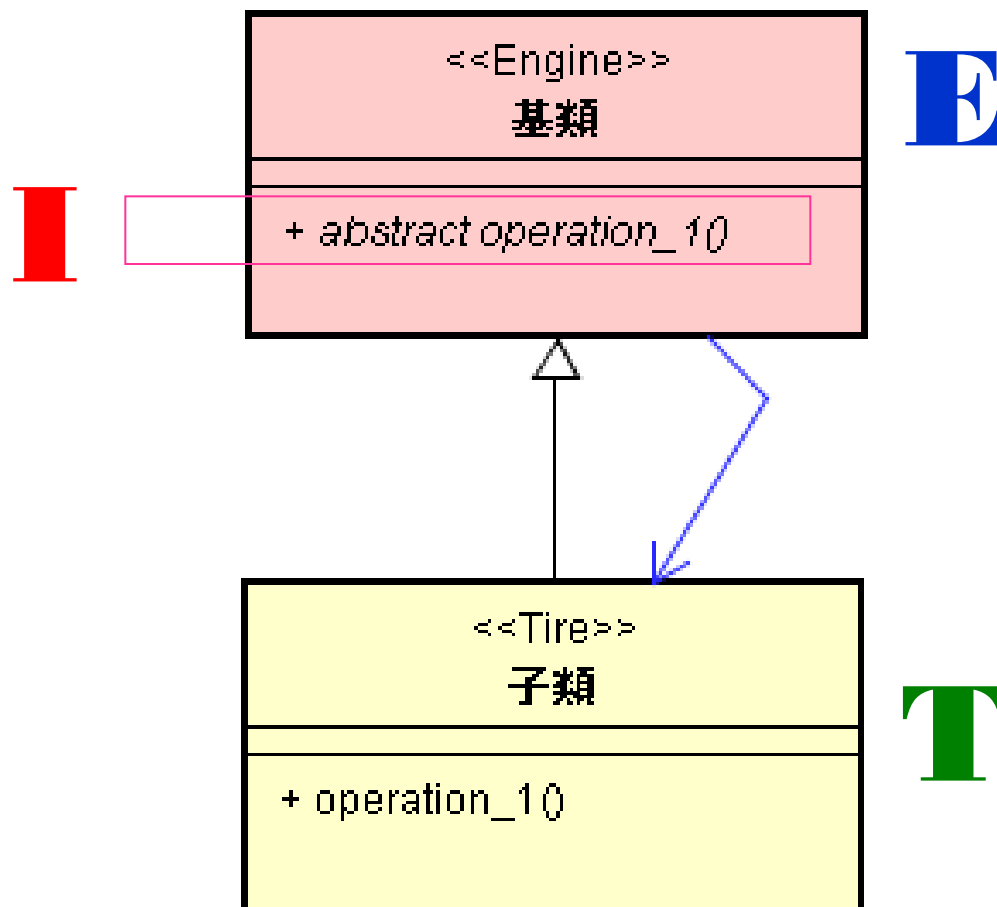
==

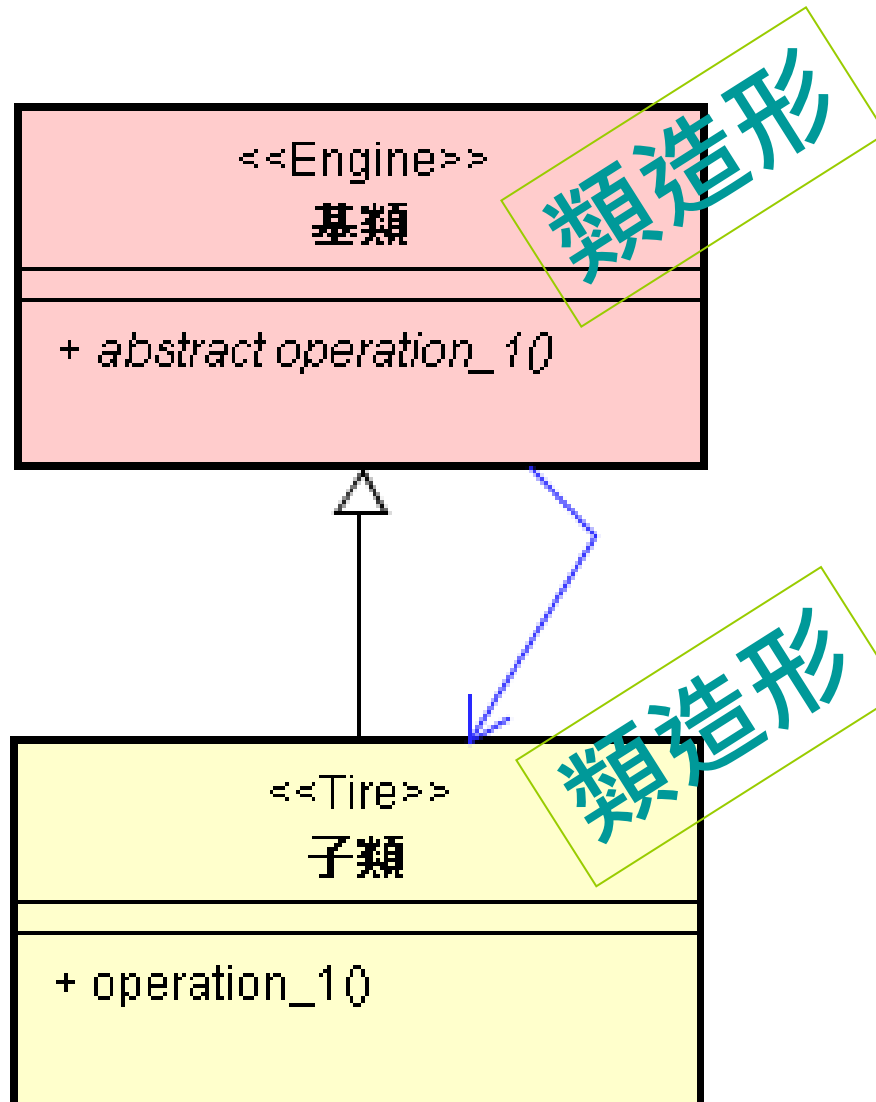
楓葉類

楓葉
- 顏色 : int - 大小 : int - 種類 : char
+ 飛() : void + 其它() : void

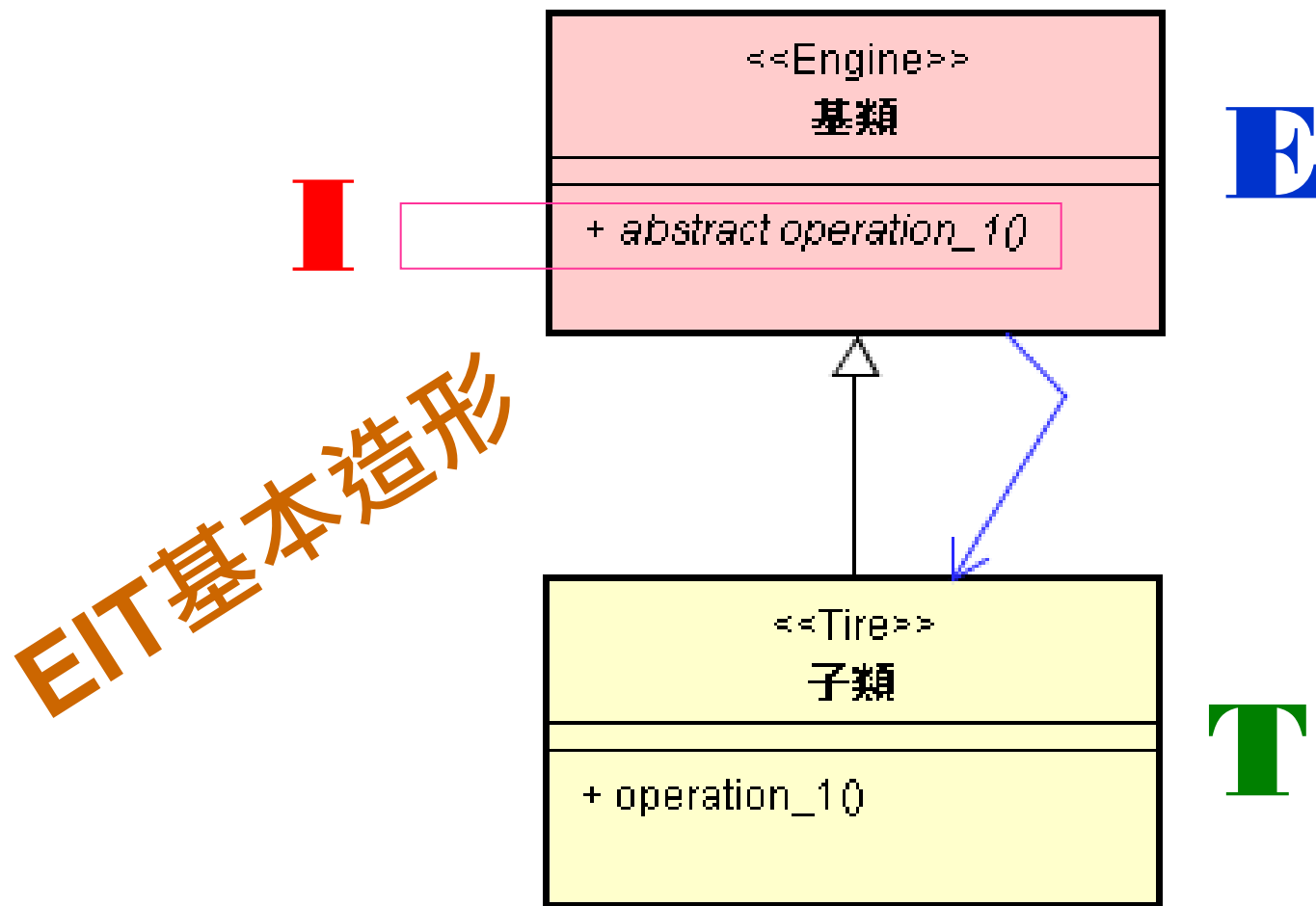
## 2、认识EIT造形与内涵

# EIT造形：类造形的组合





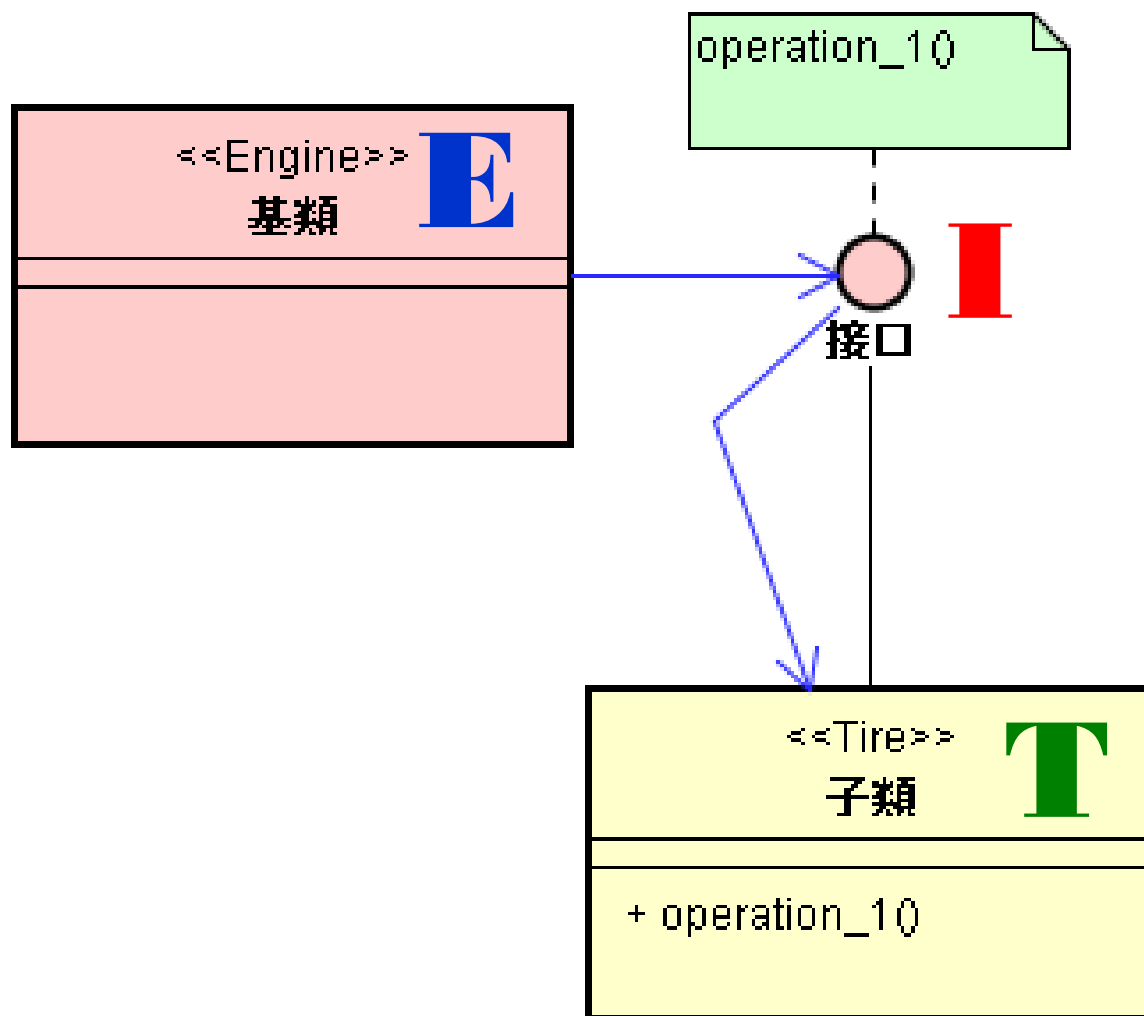
# 复习EIT的基本形与变形

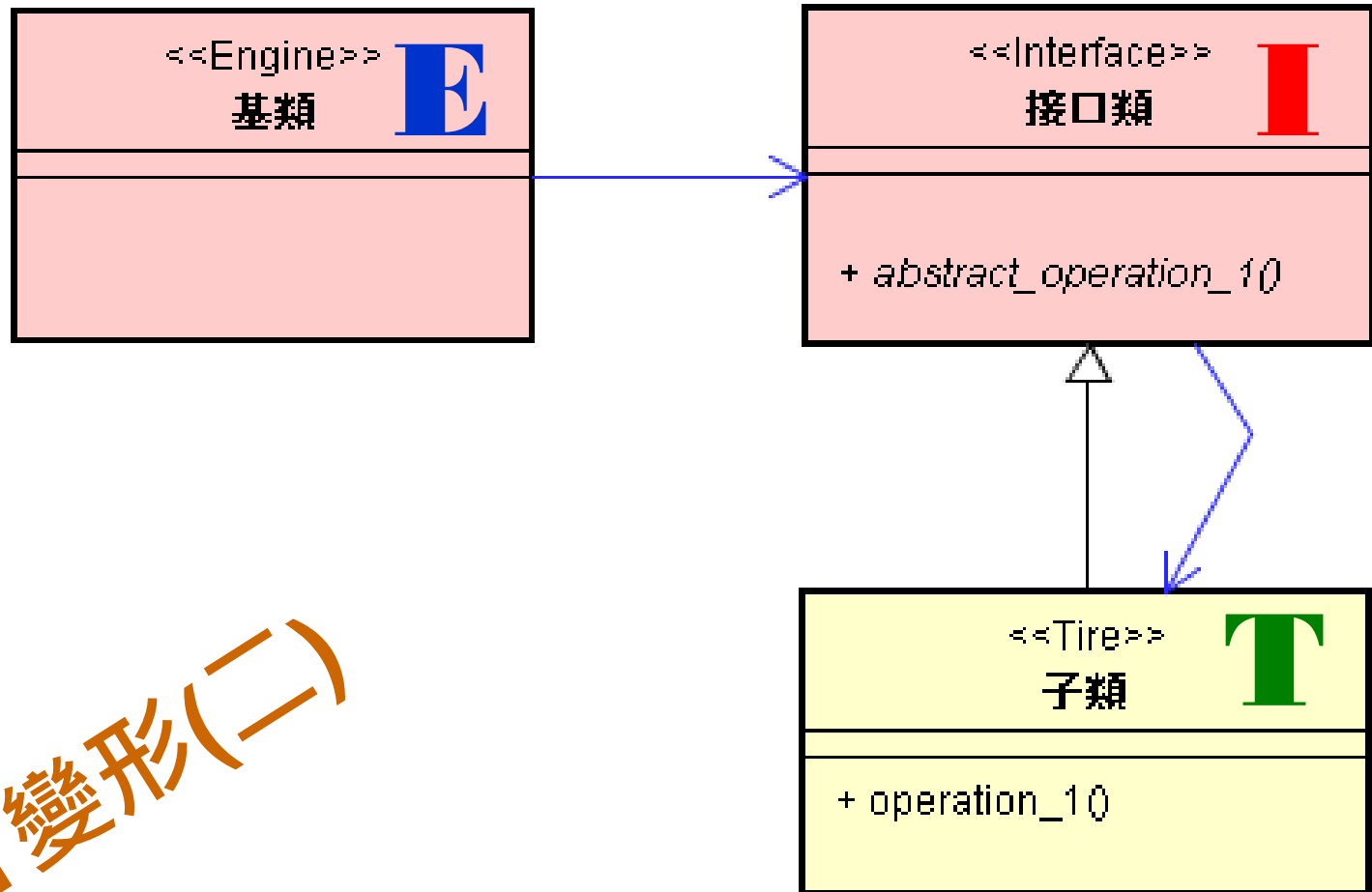


- EIT 造形就像比類(Class)大的集裝箱。
- 就像集裝箱也有不同形式(即變形)。
- EIT造形有2种变形，如下：

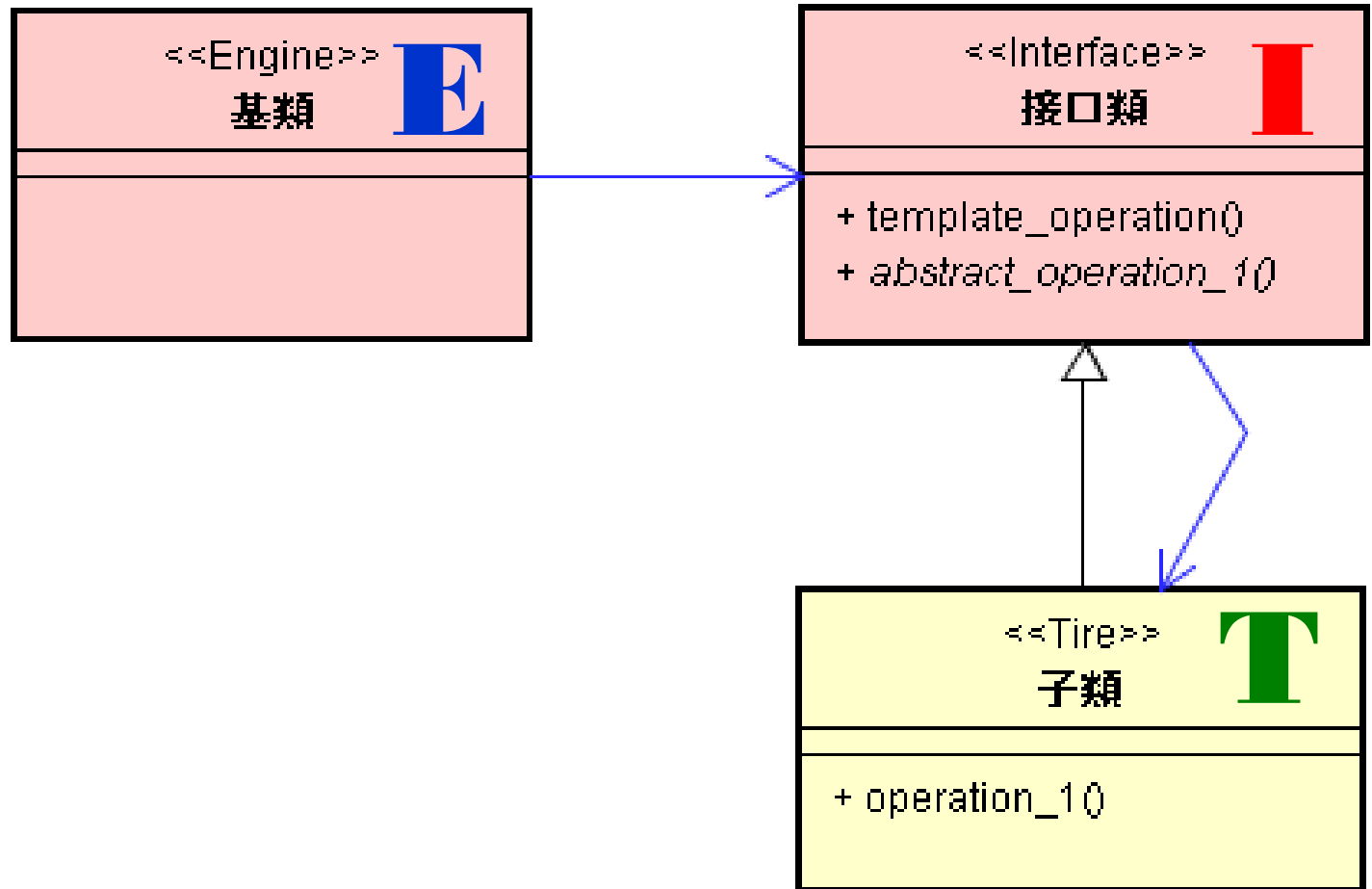


# EIT 變形(一)





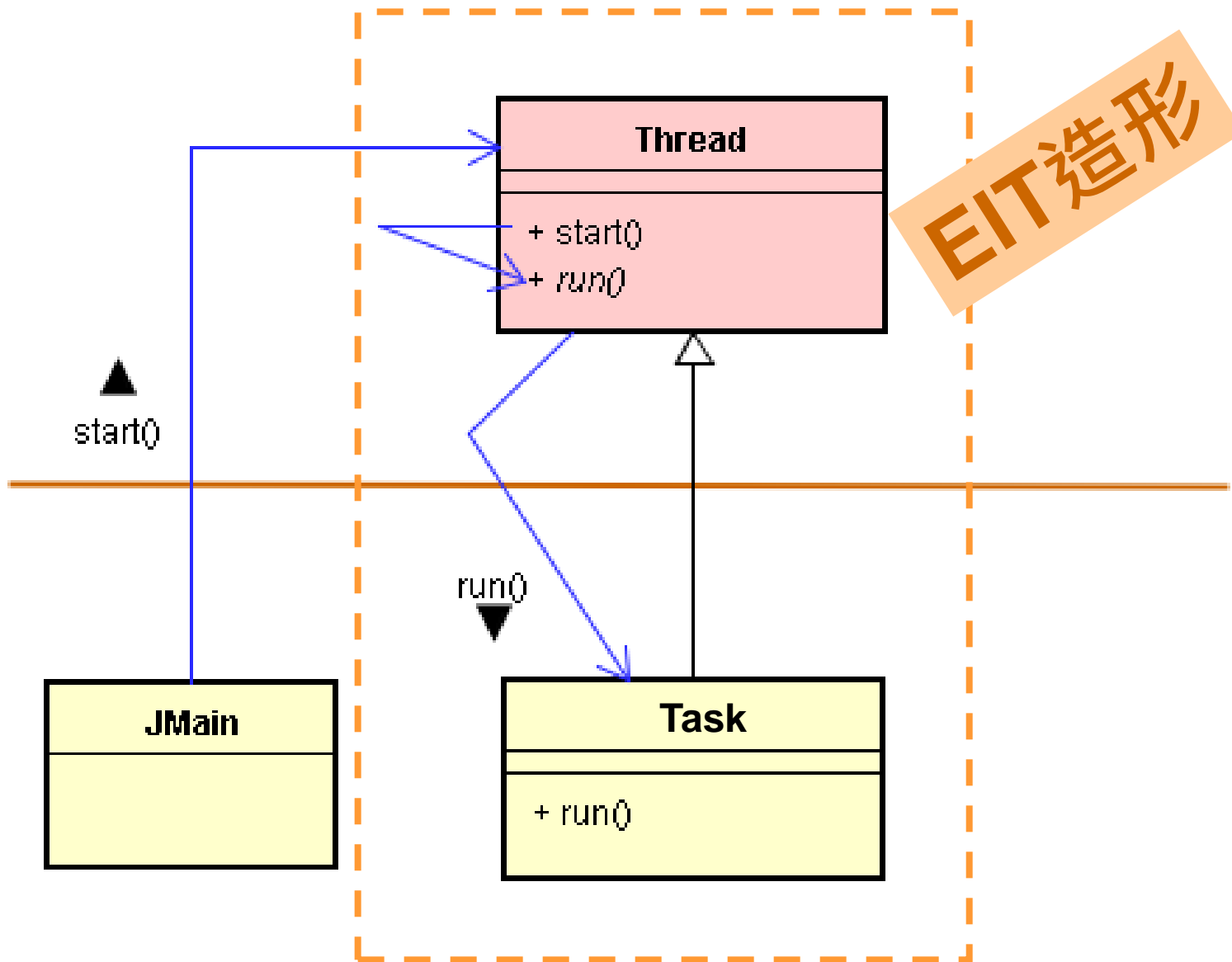
EIT 變形(二)



# 不同内涵 + EIT造形

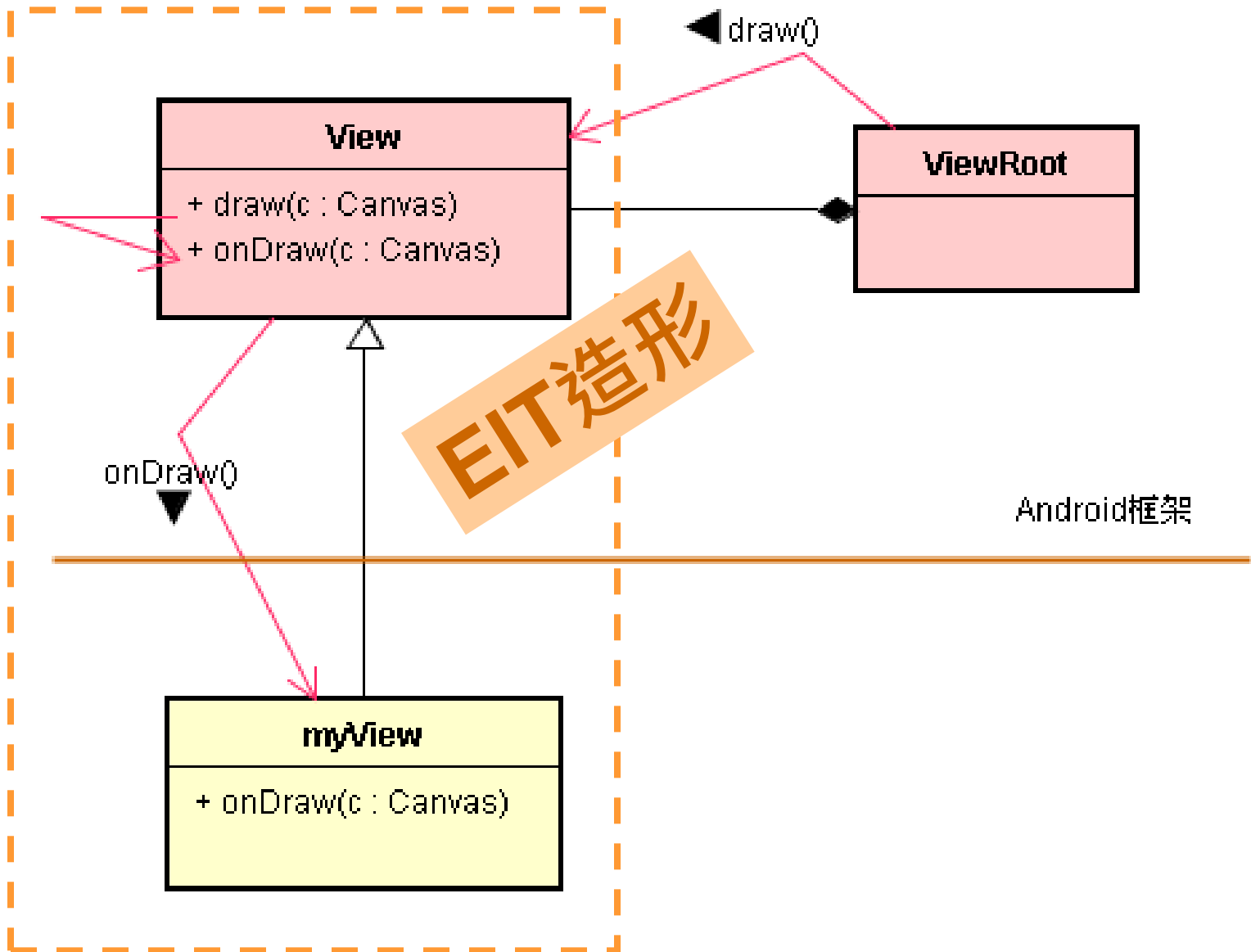
## 范例(一)

- 线程(Thread)内涵 + EIT造形



## 范例(二)

- UI窗口(View)内涵 + EIT造形



```
// myView.java
```

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

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

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

```
    private int line_x = 100;
```

```
    private int 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++;
```

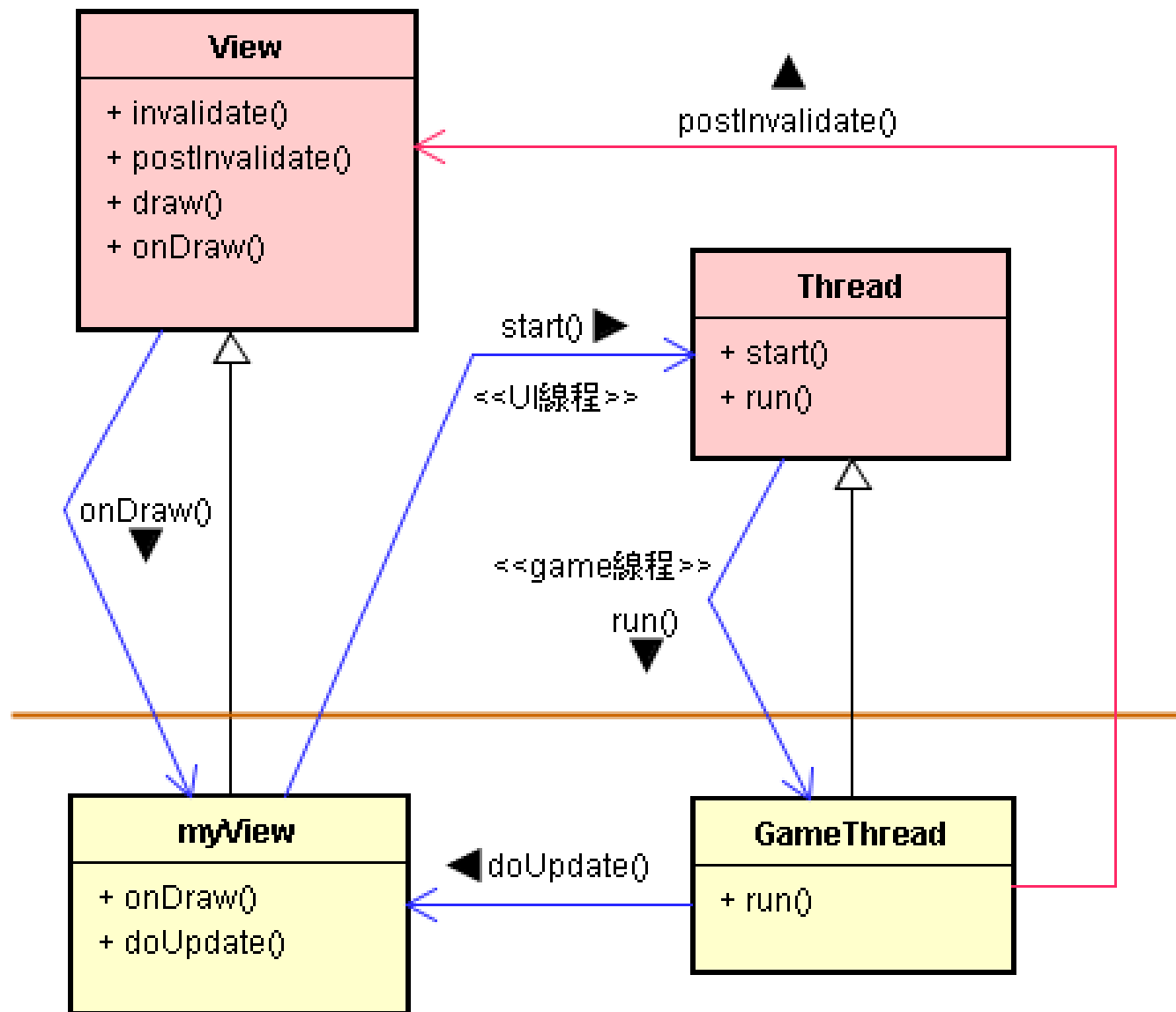
```
        //-----
```



```
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);  
}  
}
```

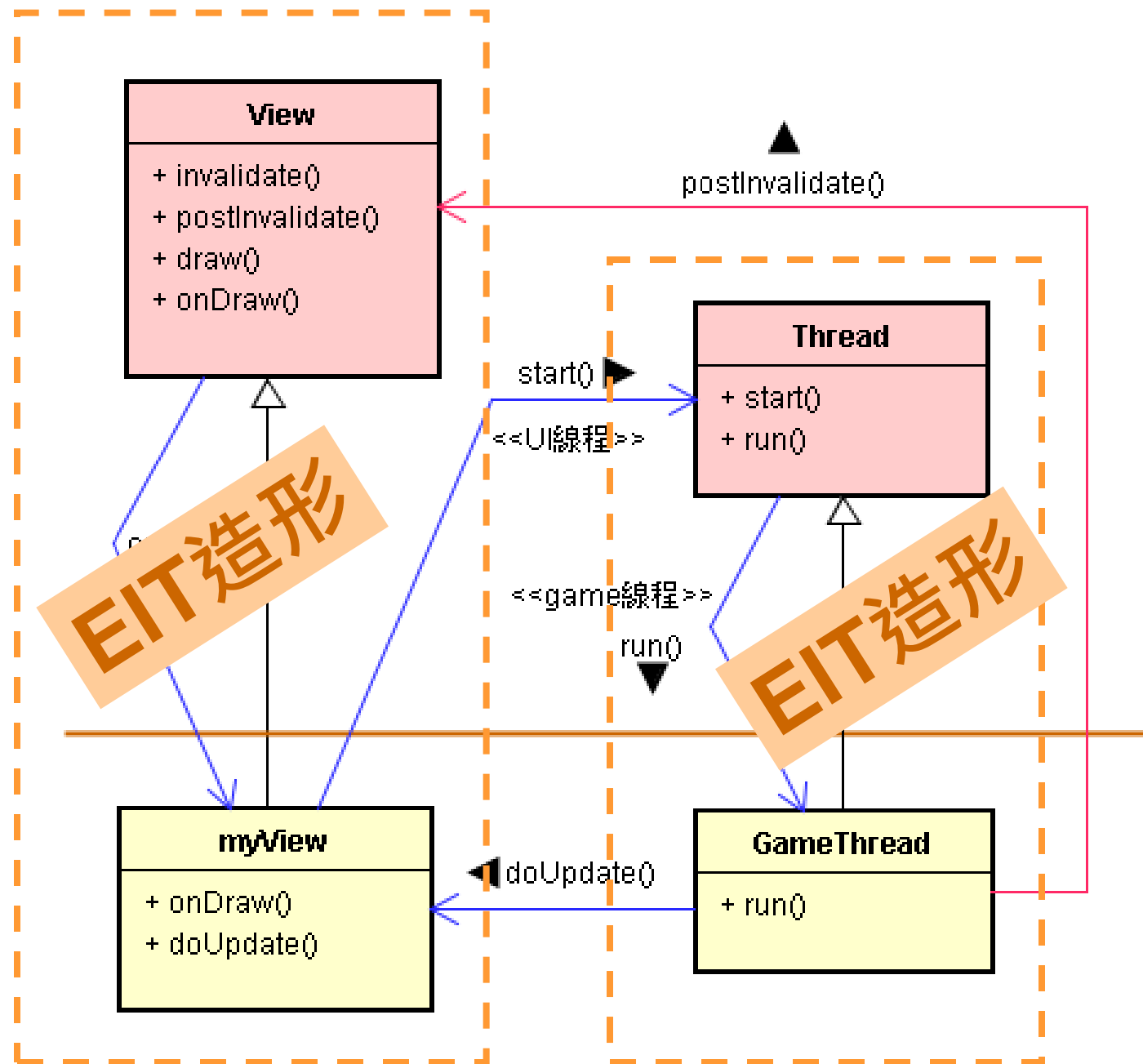
## 范例(三)

- 将上述两个有内涵的EIT造形，加以组合起来



- 将Thread造形与View造形做些组合，可以创造出双线程的绘图循环：View造形的UI线程与Thread造形的小线程，联合控制了绘图循环。

- 在游戏软件应用上，这个Thread造形里的小线程(由UI线程所诞生的)扮演一个特殊的角色：成为游戏的主控循环(Game Loop)，而UI线程则专注于响应UI 的事件，创造出两个线程完美分工。由于这个线程专注于游戏主控循环，所以又称为游戏线程(Game Thread)。



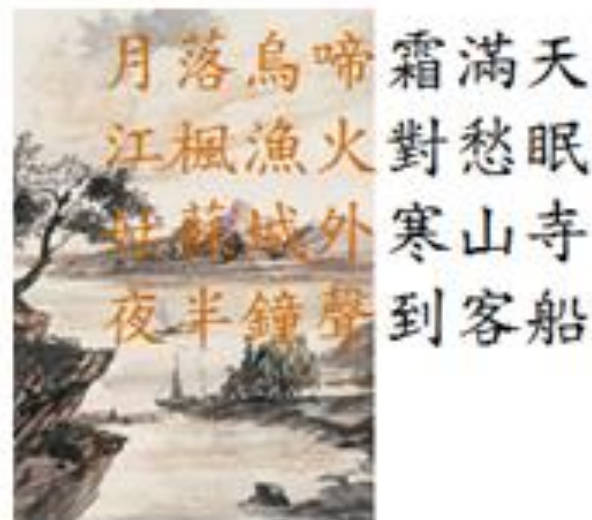
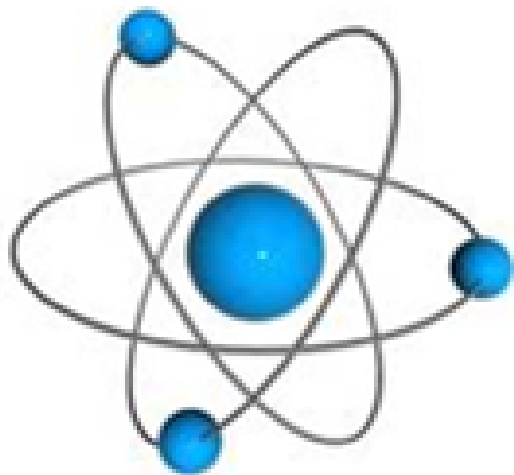
### 3、结语

- 造形特性：

形式简单、内涵复杂、重复组合

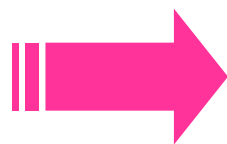
- 其它造形范例：

物理原子(Atom)、唐诗七言绝句之形



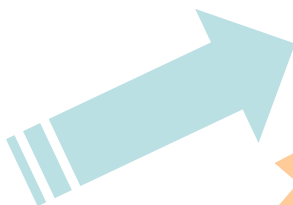


李白的创意

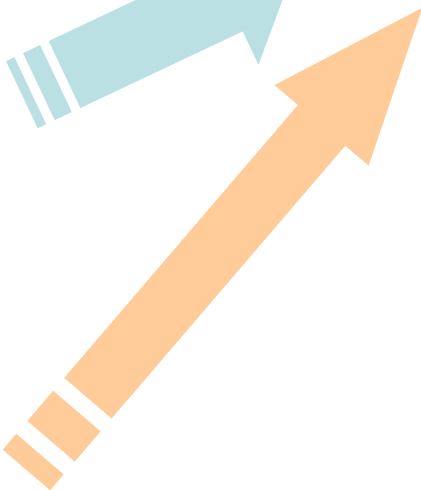


七言绝句  
唐诗造形

杜甫的意境



白居易感触



~ Continued ~