

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>

A04

简介EIT造形

By 高煥堂

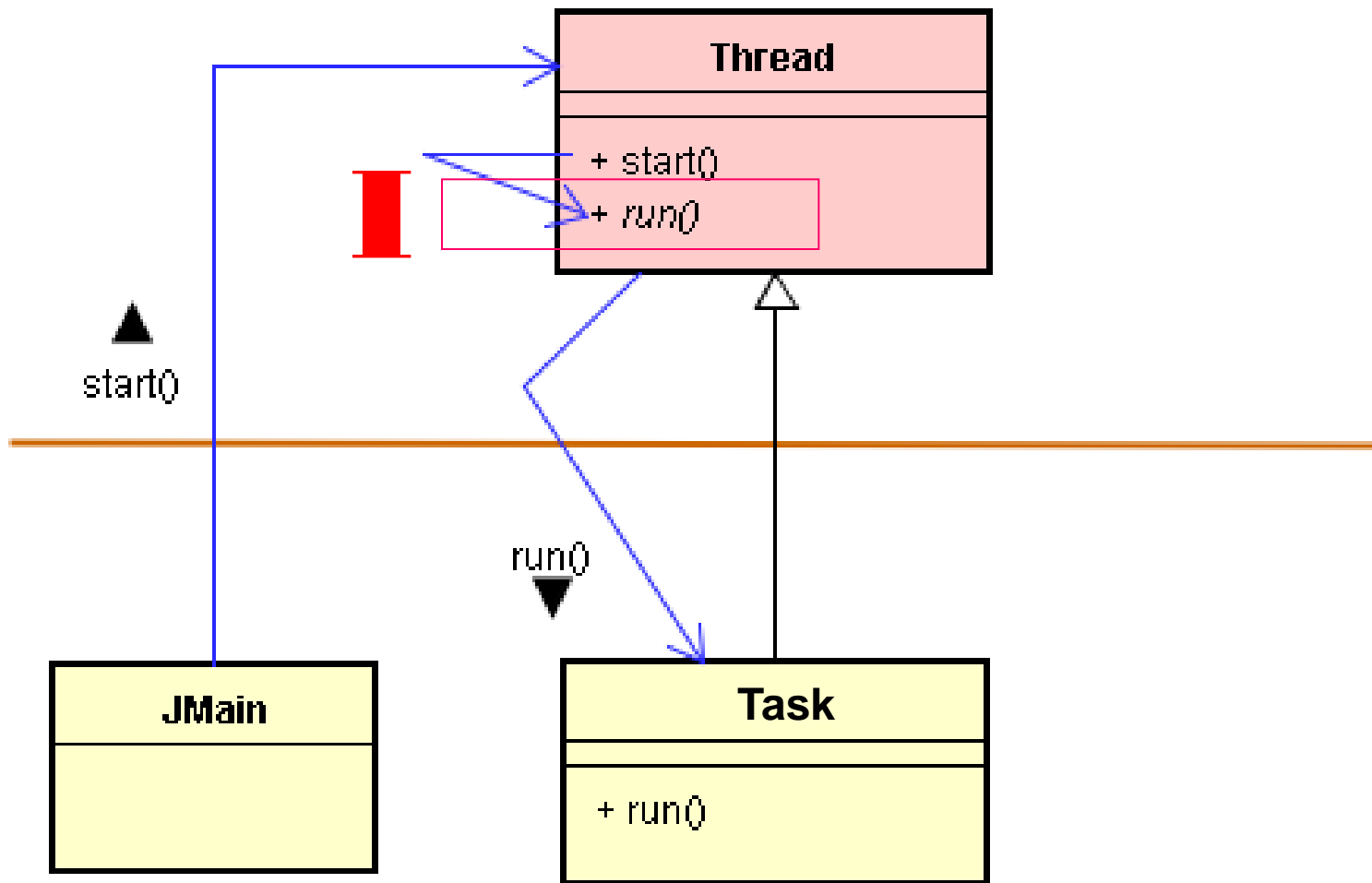
内容

1. 复习<基类/子类>的 扩充(extends)结构
2. 从<基类/子类>结构到EIT造形
3. EIT造形的基本形与变形

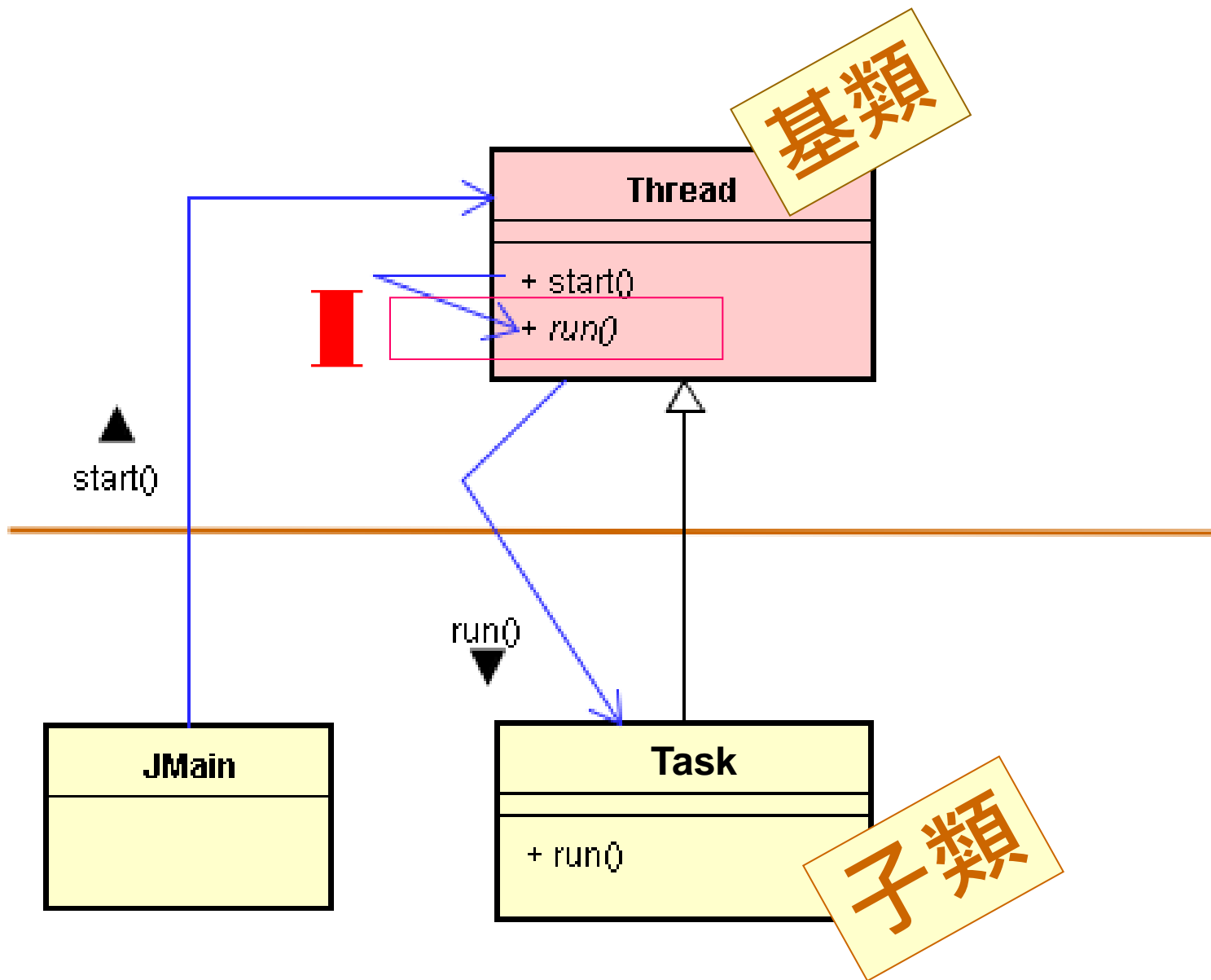
1、复习<基类/子类>的 扩充(extends)结构

典型的<基类/子类>代码结构

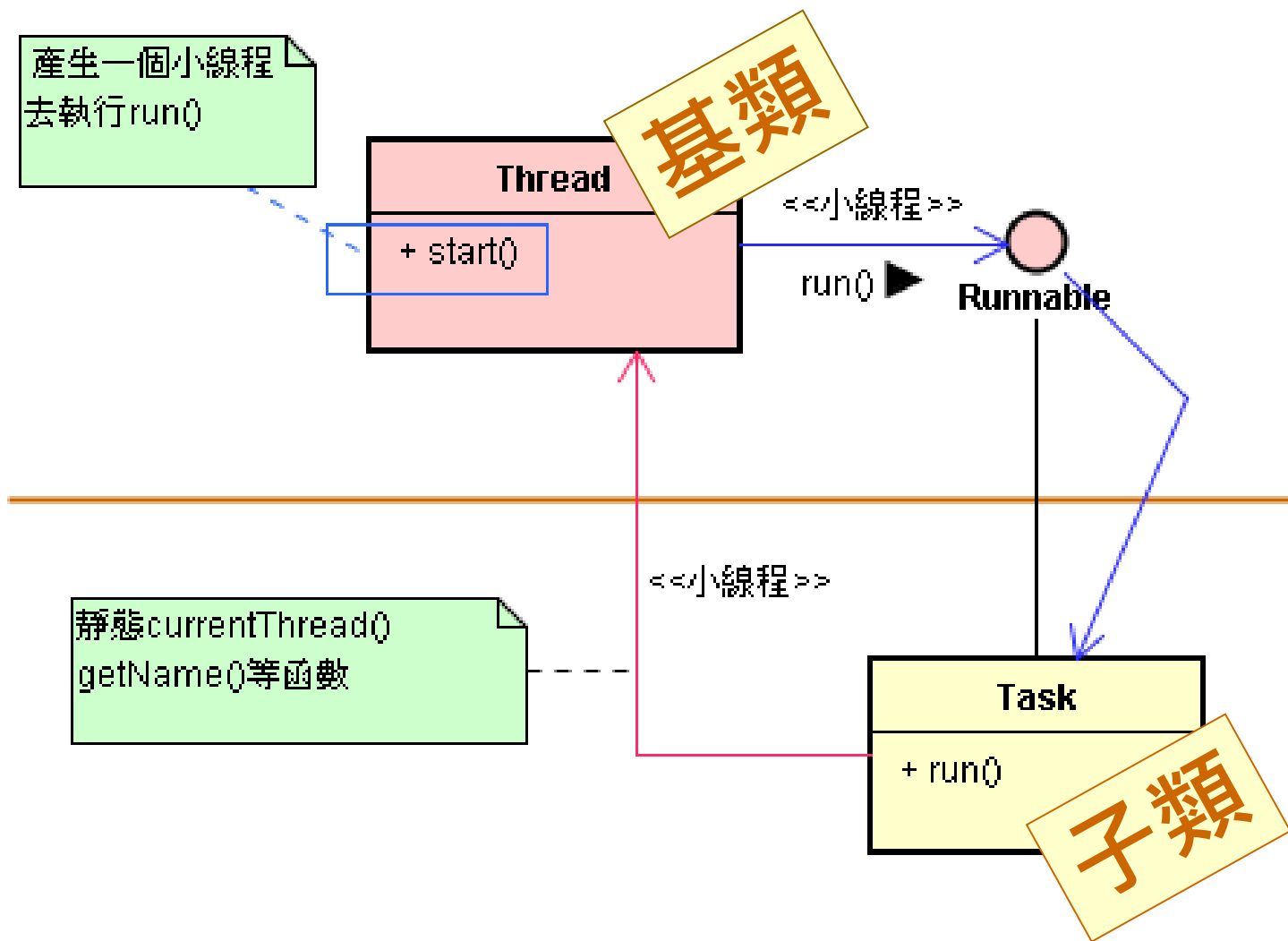
```
class Task extends Thread {  
    public void run() {  
        int sum = 0;  
        for (int i = 0; i <= 100; i++)  
            sum += i;  
        System.out.println("Result: " + sum);  
    }  
}
```



```
public class JMain {  
    public static void main(String[] args) {  
        Thread t = new Task();  
        t.start();  
        System.out.println("Waiting...");  
    }  
}
```



<基类/子类>代码结构的变形



- 于此图里，Thread基类会先诞生一个小线程，然后该小线程透过Runnable接口，呼叫(或执行)了Task类别的run()函数。

```
class Task implements Runnable {  
    public void run() {  
        int sum = 0;  
        for (int i = 0; i <= 100; i++)  
            sum += i;  
        System.out.println("Result: " + sum);  
    }  
}
```

```
public class JMain {  
    public static void main(String[] args) {  
        Thread t = new Thread(new Task());  
        t.start();  
        System.out.println("Waiting...");  
    }  
}
```

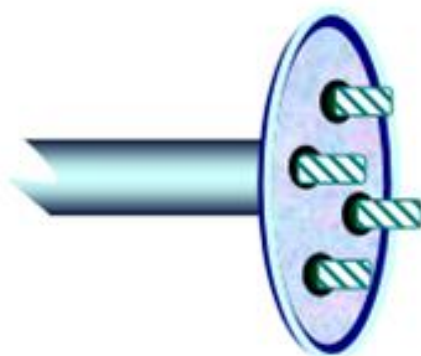
2、从<基类/子类>结构 到EIT造形

◎以汽车来做比喻

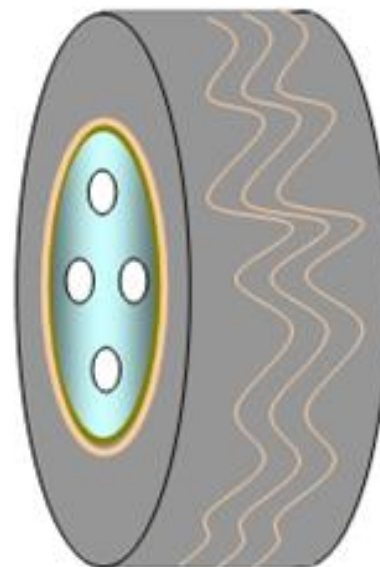




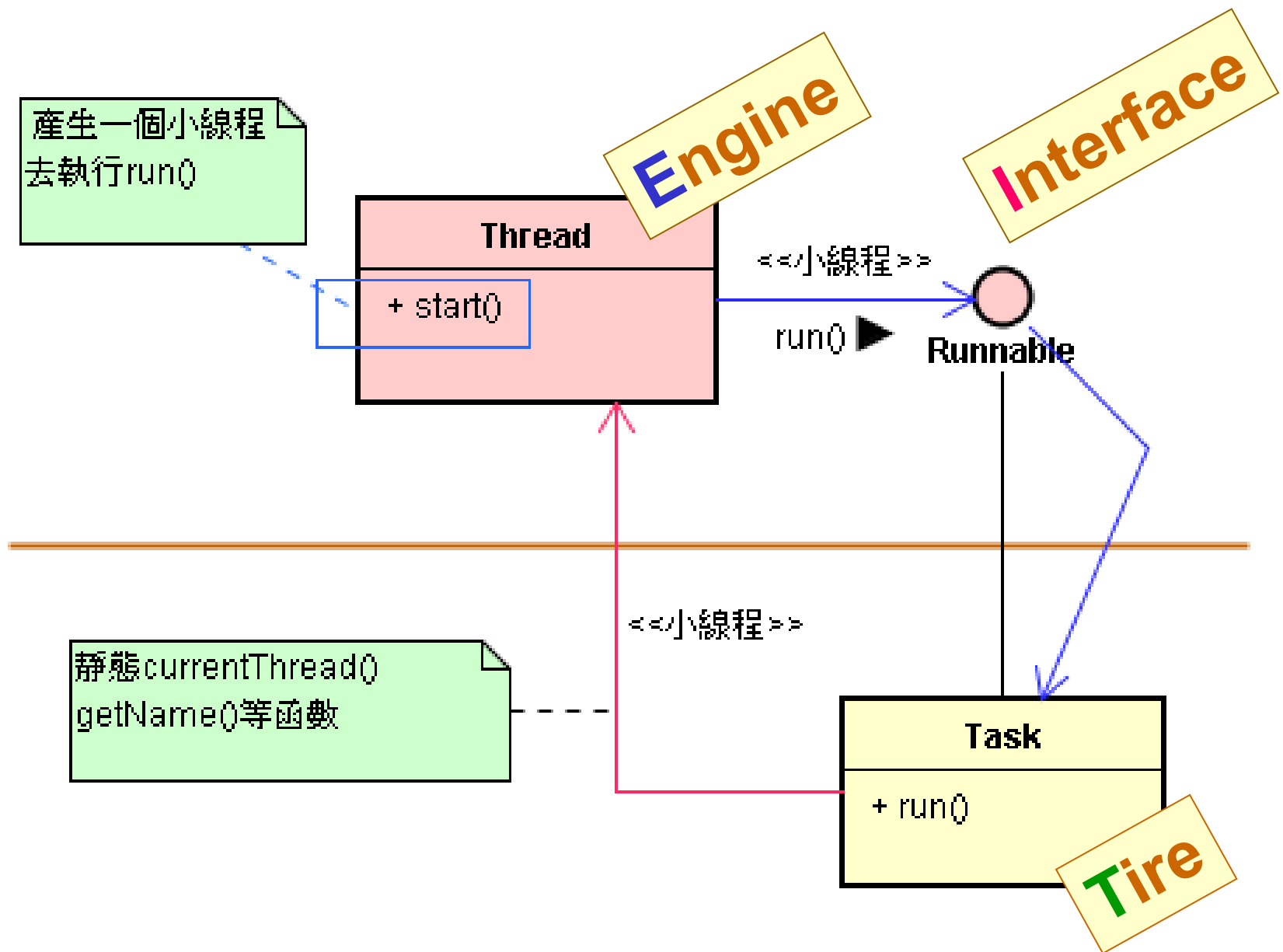
Engine

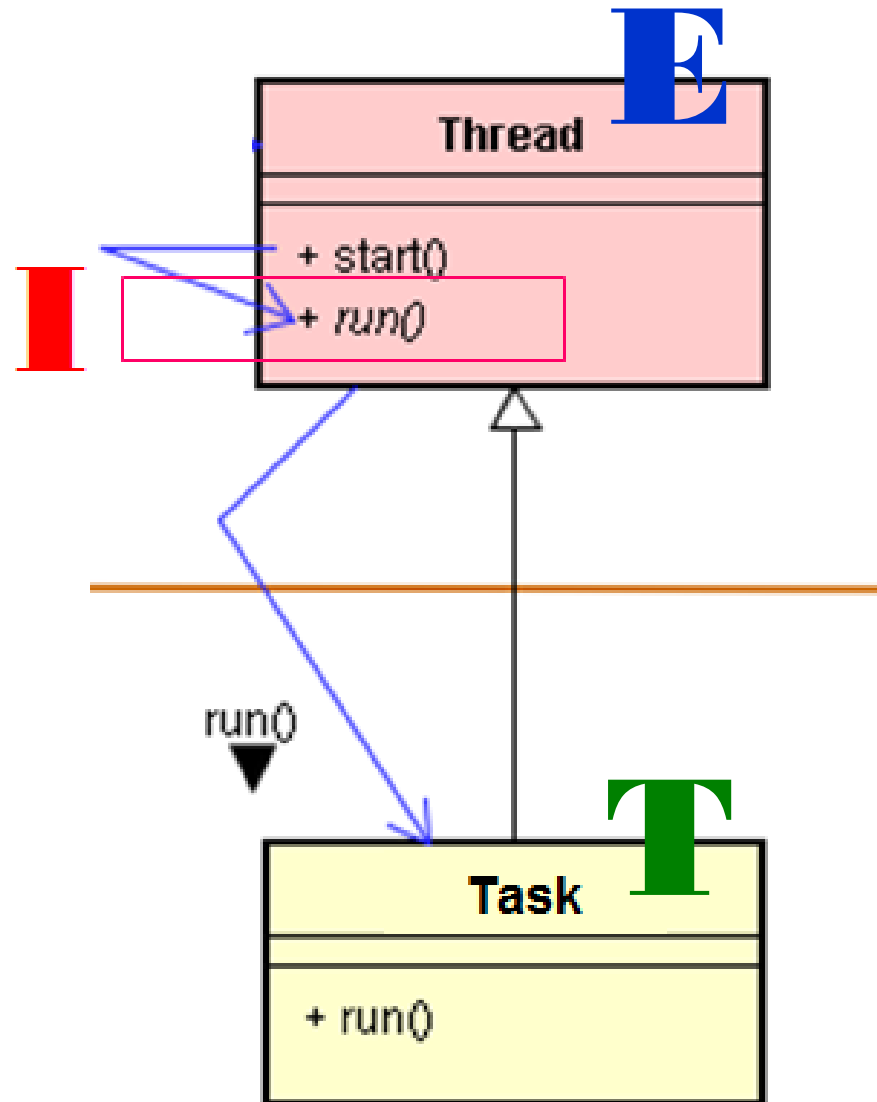


Interface



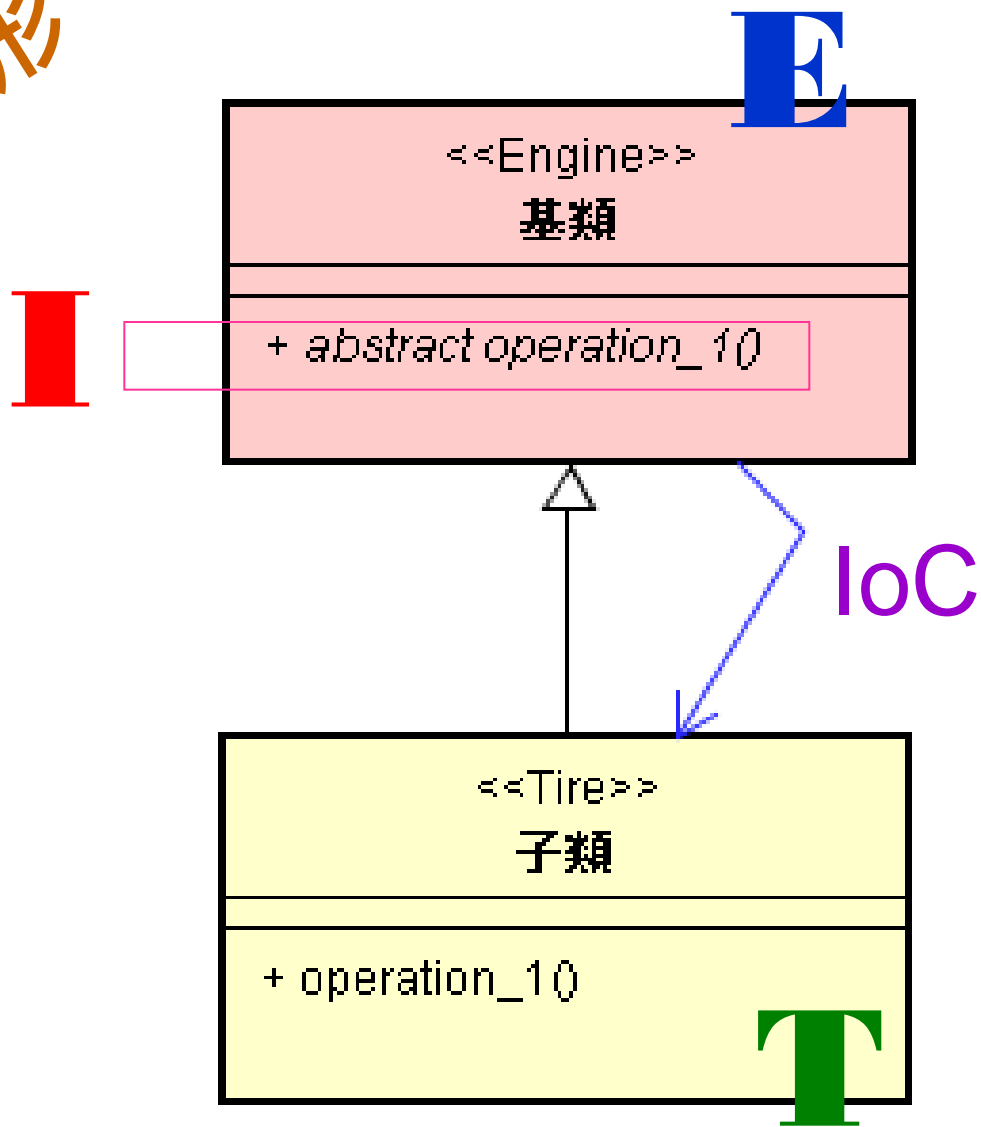
Tire



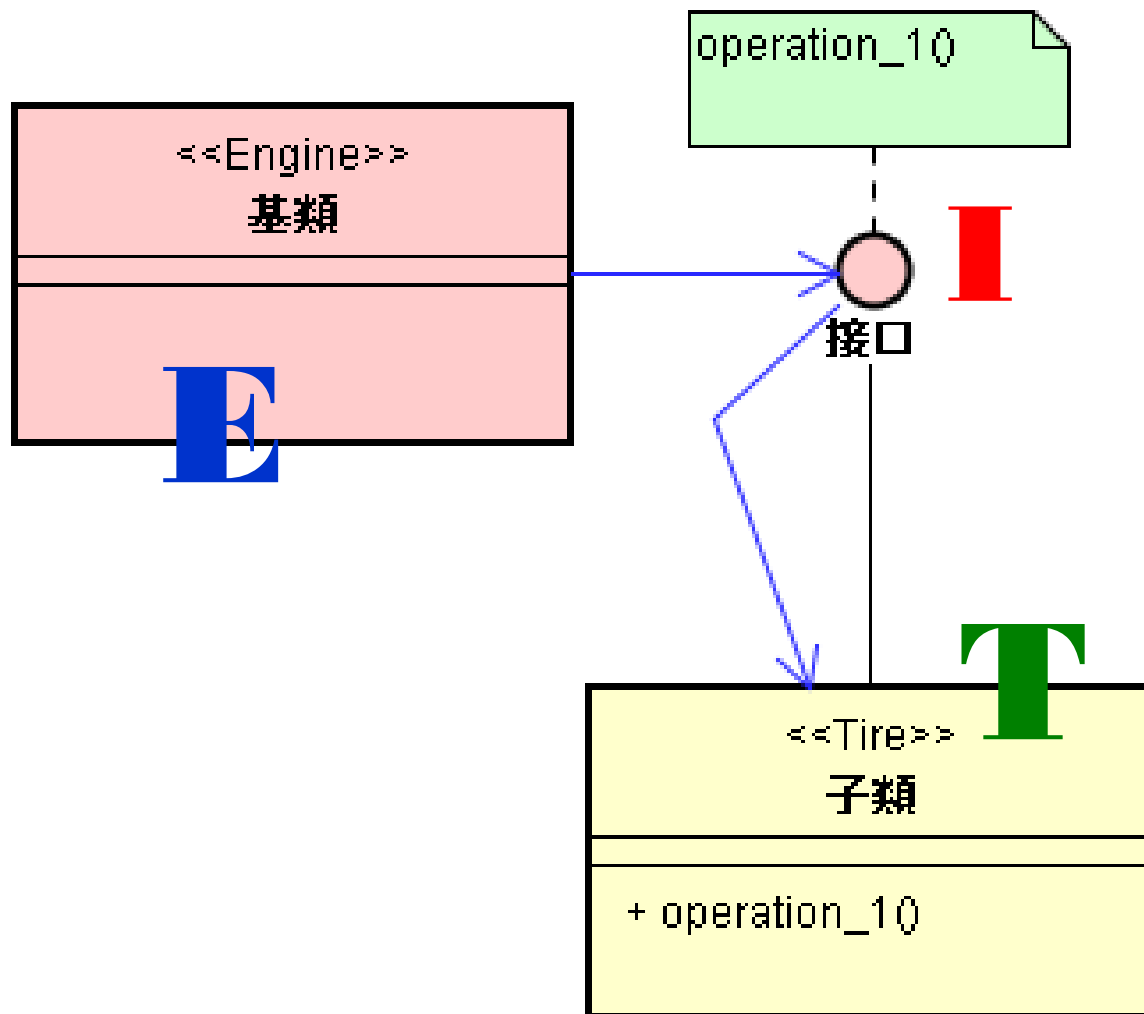


3、EIT造形的基本形 与变形

EIT基本形



EIT 變形



Q&A



高煥堂