

4-7

執行緒的程式範例

執行緒的程式範例

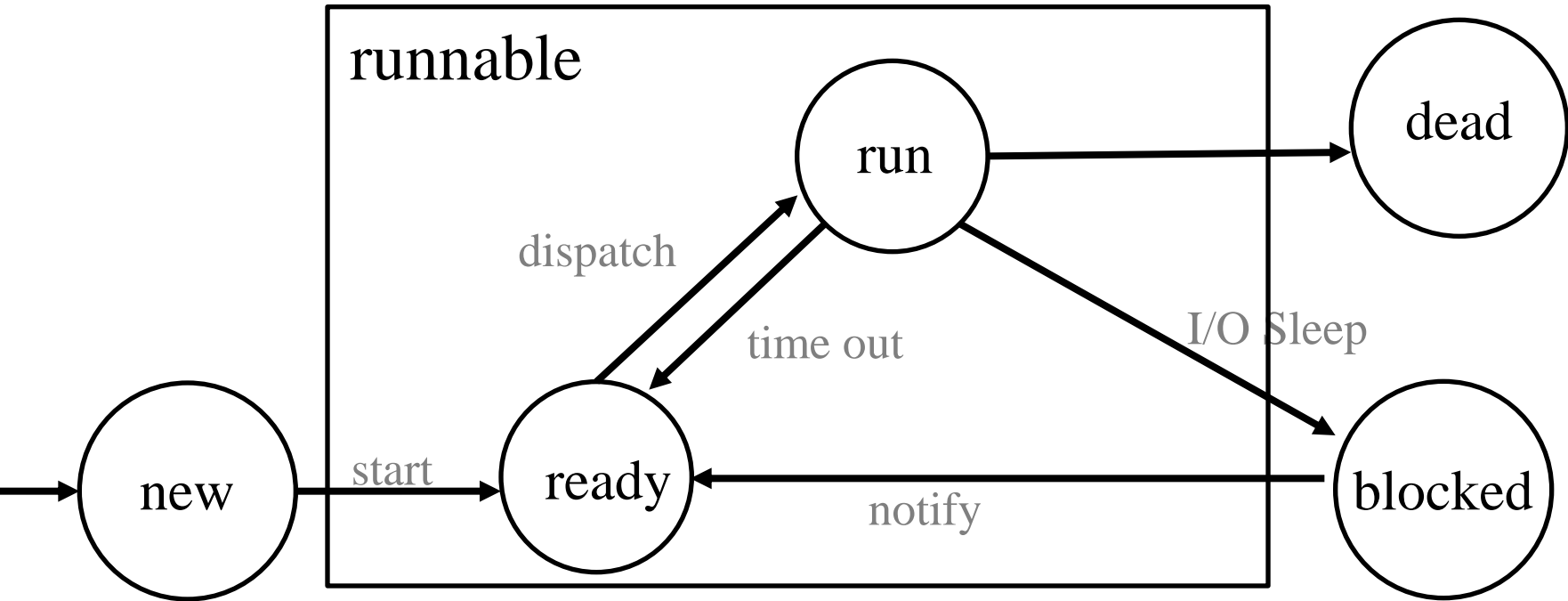
- 學習完本單元，您將可以：
 - 認識Java執行緒程式撰寫範例

Java 執行緒範例

```
Class children extends thread
{  public void run() {
    system.out.println( "children thread" );}
}
public class parent
{  public static void main(string args[]){
    children t1 = new children();
    children t2 = new children();
    t1.start();
    t2.start();
    system.out.println( "parent thread" );}
}
```



Java 執行緒狀態變化



執行緒排程(Thread Scheduling)

- 處理程序內部排程(Process local scheduling)
 - 執行緒之排程由所依附的處理程序內部的執行緒排程器排程。
- 系統全域排程(System global scheduling)
 - 由作業系統對處理程序排程。



Java之分時排程法

- 優先等級(Priority)

- 排程器執行緒之優先等級8。
- 一般執行緒之優先等級2。
- 在執行狀態之執行緒之優先等級6。

- 演算法

- 排程器執行緒從佇列內選取一個執行緒，並給予此執行緒優先等級6。
- 排程器執行緒進入懸置狀態(sleep)。
- 被選取的執行緒因為優先等級6最高，所以擁有中央處理器執行。
- 排程器執行緒被喚醒並占有中央處理器，然後將原來佔有中央處理器執行的執行緒之優先等級改為2。

Java之分時排程法程式範例

```
Public class threadscheduler extends thread
{
    :
    private void schedulerblockwait(){
        try {
            sleep(timeslice);
        } catch(interruptedexception e) {}
    }
    public void run() {
        thread active;
        this.setpriority(8);
        while (true) {
            active =(thread)queue.getnext();
            if ((active != null) && (active.alive() )) {
                active.setpriority(6);
                schedulerblockwait();
                active.setpriority(2);
            }
        }
    }
}
```

