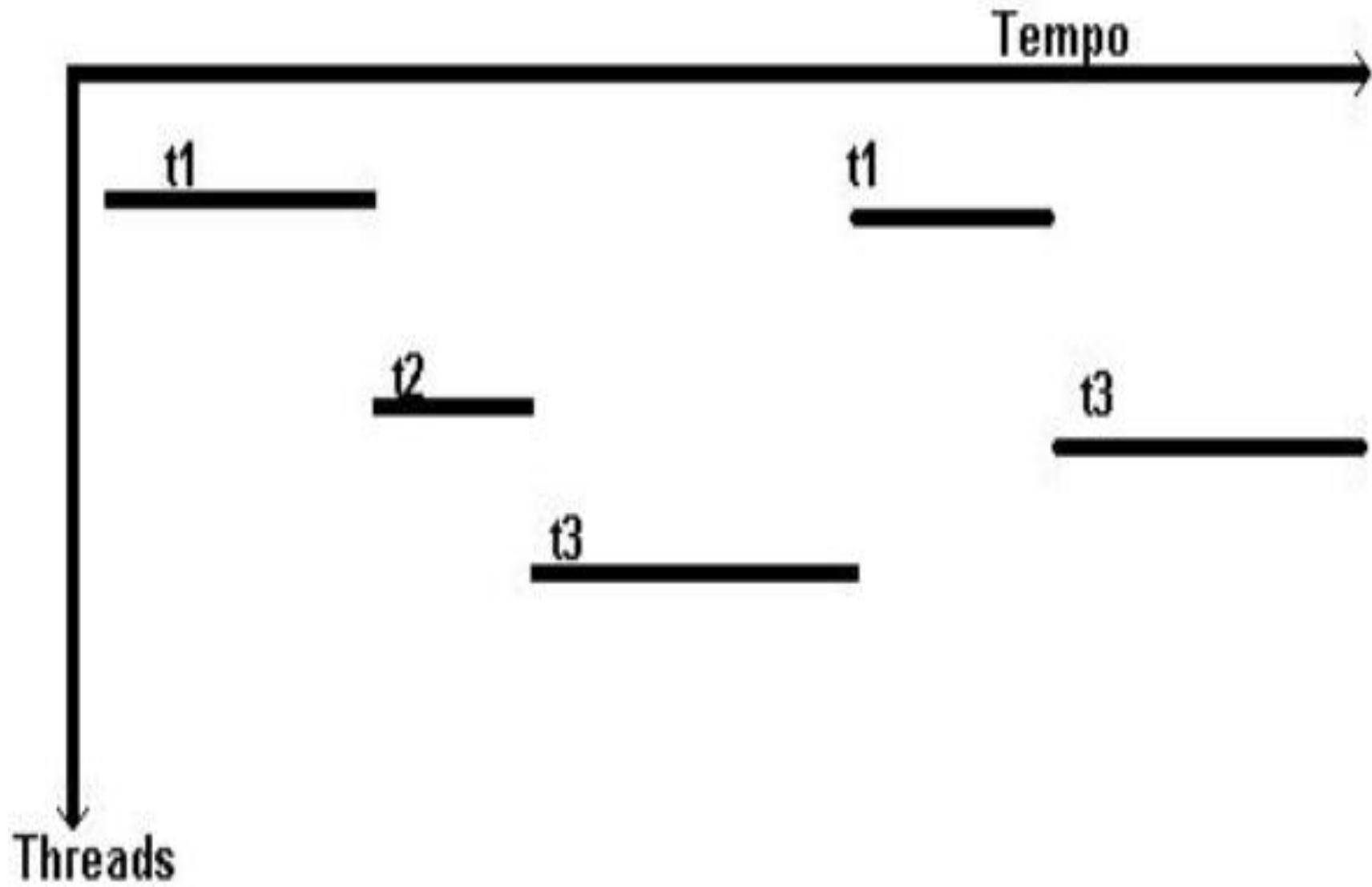


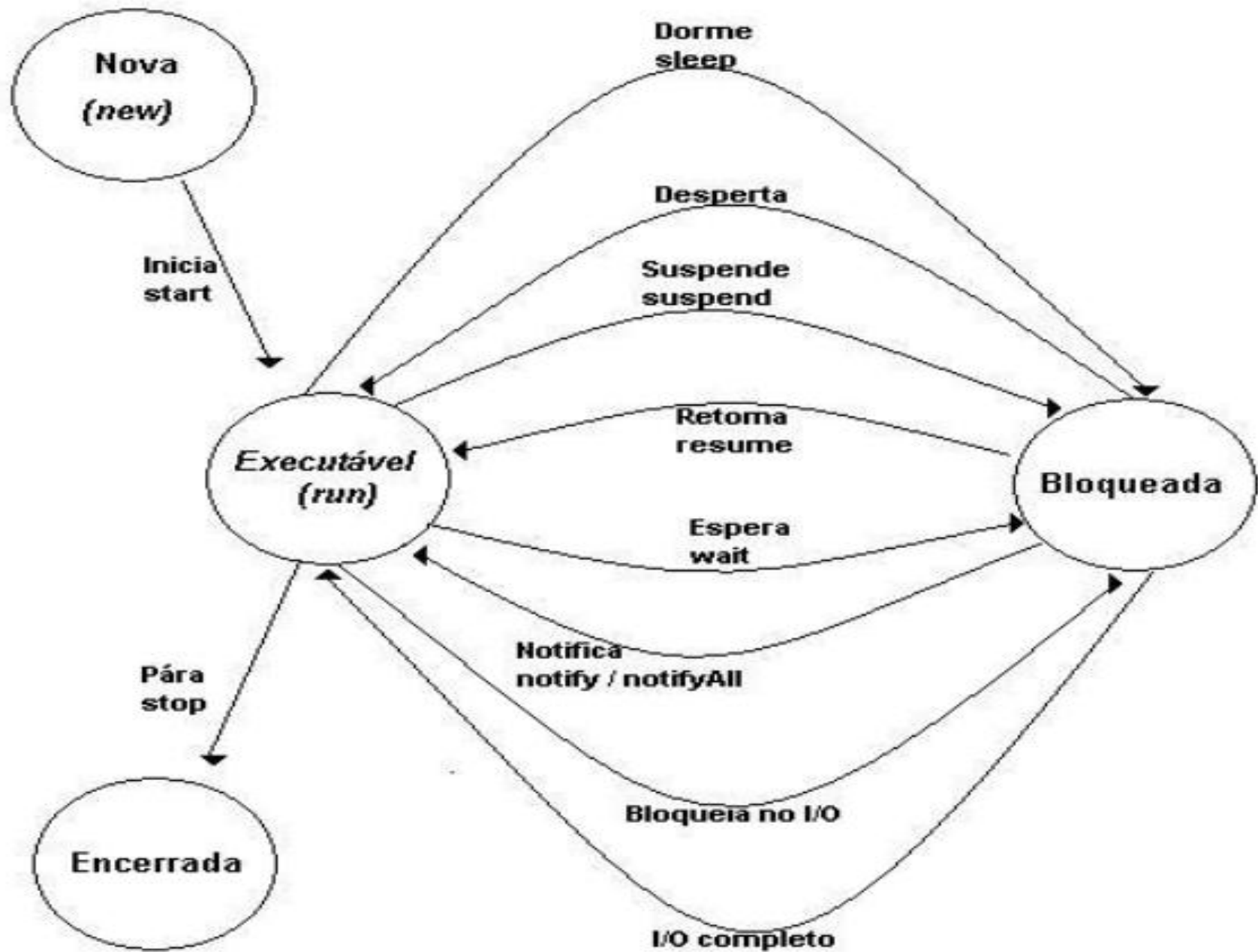
Threads



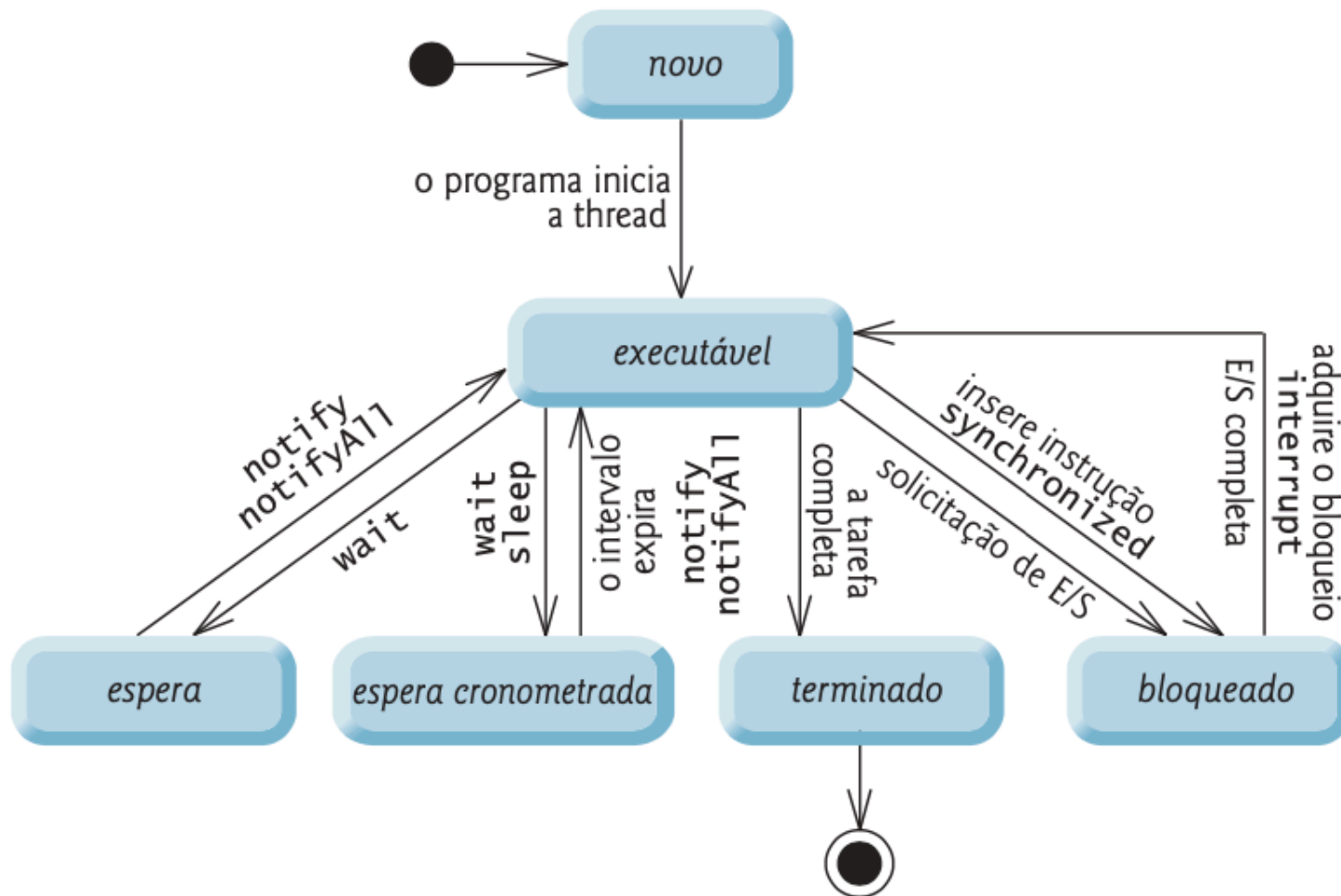
Threads

- Java é uma Linguagem Multithread
- Threads são como processo em um Sistema Operacional
- Caracterizam-se por dividir um programa em diversas linhas de execução invés de uma única.

Threads



Threads



Em Java, é possível implementar Threads de duas formas.

```
1: public class ProcessoThread extends Thread{  
  
}  
  
2: public class ProcessoThread implements Runnable{  
    @Override  
    public void run() {  
    }  
}
```

1. Runnable é uma interface, portanto sera possível estender outras classes.
2. Se voce não precisa modificar ou aprimorar a classe Thread, não há motivo para estendê-la

Observers

Observer (interface implements Observer) -> qualquer objeto que deseja ser notificado quando o estado de outro mudar

Observable (classe extends Observable) -> qualquer objeto que seu estado interessar a outros objetos.

Estes conceitos são portáteis para qualquer tipo de sistema onde objetos precisam se notificar mudanças entre si.

Observers

Requisitos Funcionais:

Objeto Observavel

Extends Observable

setChanged();

NotifyObservers();

Objeto Observador

Implements Observer

@Override

update(update(Observable obs, Object obj))

Metodo Main

ObservableObjeto observable = new ObservableObjeto();

ObserverObjeto observer = new ObserverObjeto();

observer.setObservable(observable);

observable.addObserver(observer);

Observers

```
public class ProcessoObservado extends Observable{  
    private int numero = 0;  
    public int getValue() {  
        return numero;  
    }  
    public void setValue(int numero) {  
        this.numero = numero;  
        setChanged();  
        notifyObservers();  
    }  
}
```


Observers

```
public class Observador implements Observer{  
    private ObservadoProcesso observado;  
    public void setObservado(ObservadoProcesso o){  
        this.observado=o;  
    }  
    @Override  
    public void update(Observable o, Object arg) {  
        System.out.println("algo ocorreu no objeto observado");  
        System.out.println("Numero: "+observado.getValue());  
    }  
}
```

Observers

```
public class Main {  
    public static void main(String a[]){  
        ObservadoProcesso processo = new ObservadoProcesso();  
        Observador observador = new Observador();  
        observador.setObservado(processo);  
        processo.addObserver(observador);  
        processo.setValue(34);  
    }  
}
```