

1 El patrón de delegación

Todos los componentes tienen un “evento principal”. Por ejemplo, los botones son presionados y los campos de texto reciben un “enter”. Cuando este evento ocurre, los componentes **notifican** a todos los interesados. La notificación se lleva a cabo mediante la invocación del método `actionPerformed`, que tiene que ser implementado por el objeto interesado (que llamaremos “suscriptor”).

2 Suscripción

Como se acaba de mencionar, los suscriptores deben avisar en algún momento que quieren ser notificados de los eventos que ocurren en los componentes. Por ejemplo, si un objeto `x` quiere escribir “Clic” cada vez que el botón `b` es presionado, debería escribirse el siguiente código para que se le avise a `x` que `b` fue presionado:

```
b.addActionListener(x);
```

3 ActionPerformed

El objeto `x`, debe tener implementado el método `actionPerformed`, que es invocado cada vez que ocurre un evento

```
class X implements ActionListener {
    void actionPerformed(ActionEvent e) {
        con.println("Clic");
    }
}
```

El método recibe como único parámetro un objeto de la clase `ActionEvent`. Este objeto tiene toda la información del evento, particularmente quién fue la fuente del evento (`e.getSource()`).

4 Ejemplo completo

```
import java.awt.*;
import java.awt.event.*;
class Ejemplo {
    public static void main(String args[]) {
        Frame f=new Frame("Mi ventana");
        Contador c=new Contador();
```

```
        b.addActionListener(c);
        f.pack();
        f.show();
    }
}
class Contador implements ActionListener {
    int cont;
    Contador() {
        this.cont=0;
    }
    public void actionPerformed(ActionEvent e) {
        this.cont++;
        System.out.println("Clic " + this.cont);
    }
}
```

Todo se podría poner también dentro de la clase `Ejemplo`

```
import java.awt.*;
import java.awt.event.*;
class Ejemplo implements ActionListener {
    int cont;
    public static void main(String args[]) {
        new Ejemplo().start();
    }
    void start() {
        this.cont=0;
        Frame f=new Frame("Mi ventana");
        Button b=new Button("Clic");
        f.add("Center",b);
        b.addActionListener(this);
        f.pack();
        f.show();
    }
    public void actionPerformed(ActionEvent e) {
        this.cont++;
        System.out.println("Clic " + this.cont);
    }
}
```

Los TextFields generan los mismos eventos que los botones. Cada vez que uno presiona la tecla Enter dentro de un TextField se genera un evento que debe ser comunicado a los suscriptores al TextField.