

Sucesos, componentes, interfaces

En el modelo AWT 1.1 los sucesos (events) son generados por fuentes (sources). Uno o más objetos interesados (listeners) son notificados que tal o cual suceso se ha producido en una fuente particular.

Los objetos interesados en los sucesos, también llamados manejadores de sucesos (event handlers) pueden ser objetos de cualquier clase, siempre que dicha clase implemente un determinado interface. Son necesarios tres pasos para gestionar los sucesos.

1. Una clase que implemente un interface, por ejemplo *ActionListener*

```
class MiClase implements ActionListener{  
    //...  
}
```

2. La implementación del método del interface

```
public void actionPerformed(ActionEvent ev){  
    //código respuesta a la acción del usuario sobre el componente  
}
```

3. La relación entre la fuente de los sucesos (el componente o componentes) y el *objetoDeMiClase* que está interesado o que maneja el suceso.

```
componente.addActionListener(objetoDeMiClase);
```

4. La información acerca del suceso viene encapsulada en un objeto de una clase específica derivada de *Event*.

Las siguientes tablas nos servirán de referencia para programar las respuestas a las acciones del usuario o el sistema sobre los diversos componentes. Para cada tipo de suceso hay una clase separada que lo describe. Las clases son de dos tipos:

- Las que corresponden a sucesos de bajo nivel que puede recibir cualquier componente

ComponentEvent, FocusEvent, KeyEvent, MouseEvent.

- Las que corresponden a sucesos de más alto nivel (semantic events) que solamente tienen significado para ciertos componentes.

ActionEvent, AdjustmentEvent, ItemEvent

Un suceso de alto nivel (semantic) está compuesto por varios sucesos de bajo nivel. Por ejemplo, pulsar el botón izquierdo del ratón y liberarlo, son dos sucesos de bajo nivel. Ambos generan un clic del ratón que es a su vez un suceso de bajo nivel. Cuando dicho suceso ocurre sobre un botón se genera un suceso de alto nivel descrito por la clase *ActionEvent*. Cuando se pulsa con el ratón sobre un elemento de un control lista se genera un suceso descrito por la clase *ItemEvent*.

Los sucesos de alto nivel que tienen significado para los componentes que se especifican se muestran en la tabla 1

Tabla 1.

Suceso (Event)	Componente	Acción
ActionEvent	Button	Pulsar sobre el botón
	List	Hacer doble-clic sobre un elemento de la lista
	TextField	Pulsar la tecla retorno (Enter)
AdjustementEvent	Scrollbar	Cualquier acción sobre la barra de desplazamiento
ItemEvent	Choice	Seleccionar un elemento
	List	Seleccionar o deseleccionar un elemento de la lista
	Checkbox	Activar o desactivar la casilla de verificación

Para cada tipo de suceso la aplicación puede añadir objetos interesados (listeners) en dichos sucesos. Las clases que describen dichos objetos implementan interfaces. Cuando el suceso ocurre se llama a los métodos del interface que implementa la clase. En la Tabla 2 se recogen los interfaces para cada tipo de suceso.

Por ejemplo, un objeto (listener) interesado en los sucesos provenientes de un botón se describe mediante una clase que implementa el interface *ActionListener* y define la función *actionPerformed*. Cuando se pulsa el botón, dicho objeto llama a la función *actionPerformed* y le pasa un objeto de la clase *ActionEvent* que contiene la información relativa al suceso generado.

Como vemos hay una correspondencia entre el nombre del suceso y el nombre del interface, excepto para *MouseEvent*, que hay dos interfaces para el mismo suceso por razones de eficiencia.

Tabla 2.

Suceso (Event)	Interface (Listener)	Método
ActionEvent	ActionListener	actionPerformed
AdjustementEvent	AdjustementListener	adjustementValueChanged
FocusEvent	FocusListener	focusGained focusLost
ItemEvent	ItemListener	itemStateChanged
KeyEvent	KeyListener	keyTyped keyPressed

		keyReleased
MouseEvent	MouseListener	mouseClicked mouseEntered mouseExited mousePressed mouseReleased
	MouseMotionListener	mouseDragged mouseMoved

La fuente de los sucesos mantienen una lista de objetos interesados (listeners) en los mismos. Se añaden a la lista mediante una función denominada *addXXXListener*, donde XXX es el tipo de suceso.

```
boton.addActionListener (accion) ;
```

La tabla 3, muestra los objetos fuente de sucesos y los tipos de objetos interesados en dichos sucesos (listeners) que se pueden añadir (add)

Tabla 3.

Control	Interface (Listener)
Button	ActionListener
Choice	ItemListener
Checkbox	ItemListener
Component	FocusListener KeyListener MouseListener MouseMotionListener
List	ActionListener ItemListener