

Creación de Applets Simples

Junio 2004

Michael González: mgh@unican.es

Creación de Applets Simples

- 1. Introducción**
- 2. Estructura de un applet**
- 3. Inserción en una página web**
- 4. Dibujando sobre un applet**
- 5. Etiquetas y botones**
- 6. Entradas de Texto**
- 7. Ejemplo: regresión lineal**

1. Introducción

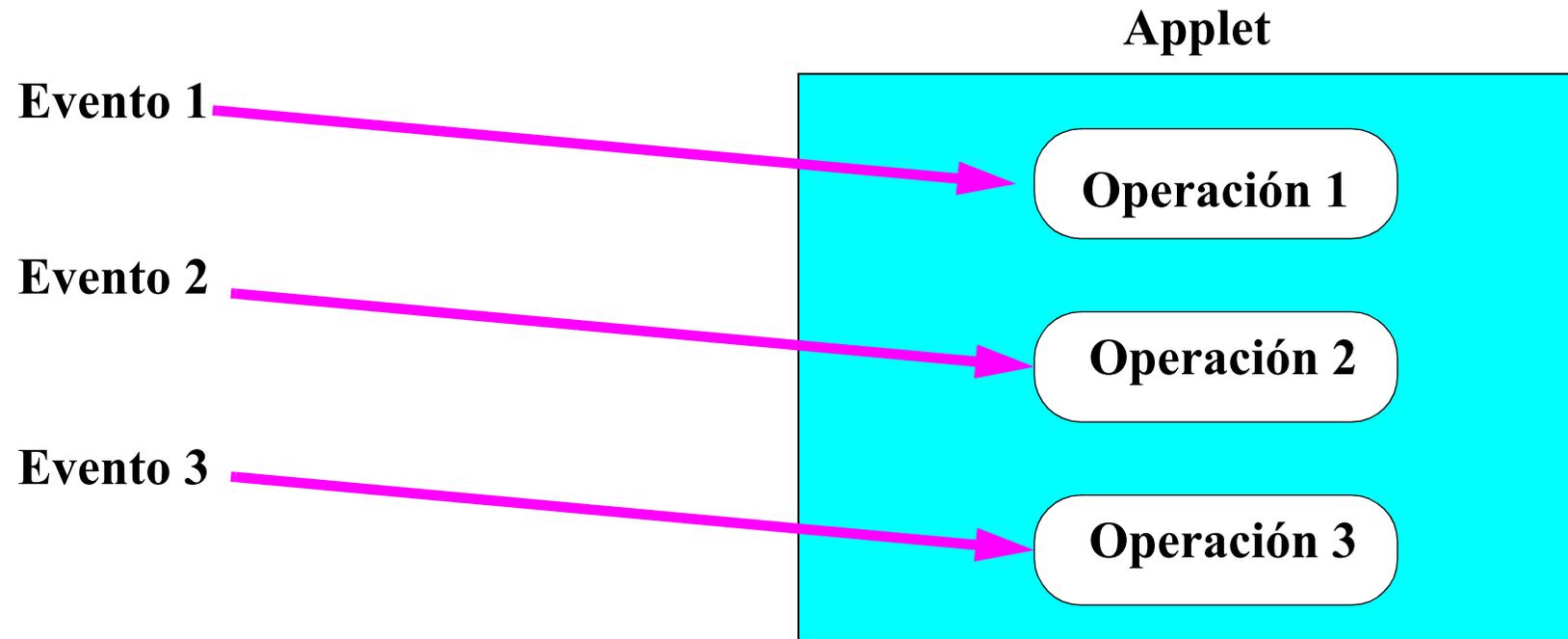
Los *applets* son objetos Java que:

- presentan una ventana gráfica para su ejecución
- tienen un conjunto bien definido de operaciones que:
 - les permiten formar parte de otros programas (p.e., *appletviewer*)
 - y estar integrado en páginas *web*
- permiten por tanto crear páginas web *dinámicas*
- tienen algunas *restricciones* especiales:
 - no tienen operación `main()`
 - están *gobernados por eventos*
 - la entrada/salida es diferente a la de las aplicaciones

Los **eventos** son sucesos que el sistema detecta relacionados con el programa, la entrada/salida, y el sistema de ventanas:

- **eventos de ratón**: mover, hacer click, ...
- **eventos de teclado**: pulsar o liberar una tecla
- **eventos de acción**: pulsación de un botón o menú
- **eventos de texto**: cambiar el valor de una entrada de texto
- **eventos de ventana**: cerrar una ventana, abrirla, minimizarla...
- **etc.**

Arquitectura gobernada por eventos



Arquitectura gobernada por eventos (cont.)

En un programa *gobernado por eventos*:

- hay operaciones que gestionan eventos
- el applet está esperando a que llegue un evento
- cuando el sistema detecta un evento llama a la operación asociada a ese evento
- la operación debe terminar pronto
- la operación no debe esperar a que algo ocurra (p.e., leer)
- el usuario inicia la acción, no al revés como en las aplicaciones

2. Estructura de un applet

```
import java.awt.*;
import java.applet.*;

public class SimpleApplet extends Applet {

    String mens;

    public void init() {
        mens="Inicializado";
    }

    public void start() {
        mens=mens+" Comienza";
    }

    public void paint(Graphics g) {
        g.drawString(mens,80,10);
    }
}
```

3. Inserción en una página web

Para insertar un applet en una página web:

- con Bluej: “*Run Applet in Web browser*”
- si la herramienta de creación de páginas web lo permite, añadir el applet en el lugar deseado
- si no, añadir con el editor de textos en el lugar deseado:

```
<applet codebase="classes" code="SimpleApplet"  
width=300 height=200>  
</applet>
```

donde:

- ***codebase***: directorio donde se encuentran las clases
- ***code***: nombre del applet
- ***width, height***: ancho y alto de la ventana

Visualización

Con un navegador de internet

- Ojo: suelen usar versiones viejas de Java
- Actualizar la máquina virtual java si es necesario

Con la herramienta appletviewer:

`appletviewer pagina.html`

4. Dibujando sobre un applet

La salida se hace dibujando sobre un entorno gráfico desde el método `paint()`:

- `drawString(String str, int x, int y)`
- `drawLine(int startX, int startY, int endX, int endY)`
- `drawRect(int x, int y, int ancho, int alto)`
- `drawOval(int x, int y, int ancho, int alto)`
- `drawArc(int x, int y, int ancho, int alto, int anguloInicial, int anguloBarrido)`
- `drawPolygon (int x[], int y[], int numPuntos)`

Todas las *coordenadas* en pixels, siendo (0,0) la esquina superior izquierda

Ángulos en grados, 0=horizontal

Ejemplo

```
import java.awt.*;
import java.applet.*;

public class Dibujo extends Applet {

    public void init() {
    }

    public void start() {
    }

    public void paint(Graphics g) {
        g.drawOval(10,10,100,100);
        g.drawOval(30,30,20,20);
        g.drawOval(70,30,20,20);
        g.drawArc(30,30,60,60,210,120);
    }
}
```

5. Etiquetas y botones

Las *etiquetas* son ventanas de la clase `Label` con un texto, que se pueden añadir a un applet (con `add()`)

Los *botones* de la clase `Button`, además de lo anterior, producen un evento de “acción” al ser pulsados

- hay que programar el applet para que atienda a eventos de acción; para ello:
 - debe implementar `ActionListener`
 - debe tener una operación `actionPerformed()`, que atiende a todos los eventos de acción
- hay que indicar que los eventos del botón son atendidos por el applet:
 - con la operación `addActionListener()`

Ejemplo

```
import java.awt.*;
import java.applet.*;
import java.awt.event.*;
public class Contador extends Applet implements ActionListener{

    Label mens;
    Button incr,decr;
    int cuenta=0;

    public void init() {
        mens=new Label ("Contador: "+cuenta);
        incr=new Button("Incrementar");
        decr=new Button("Decrementar");
        add(incr);
        add(decr);
        add(mens);
        incr.addActionListener(this);
        decr.addActionListener(this);
    }
}
```

Ejemplo (cont.)

```
public void actionPerformed(ActionEvent ev) {  
  
    String str = ev.getActionCommand();  
    if (str.equals("Incrementar")) {  
        cuenta++;  
    } else if (str.equals("Decrementar")) {  
        cuenta--;  
    }  
    mens.setText("Contador: "+cuenta);  
}  
  
public void start() {  
}  
  
public void paint(Graphics g) {  
}  
}
```

6. Entradas de Texto

Son objetos de la clase `TextField` que:

- presentan una ventana en pantalla en la que se puede teclear un texto
- se pueden añadir a un applet
- se puede leer su valor con `getText()`
- generan eventos de tipo texto

Ejemplo

```
import java.awt.*;  
import java.applet.*;  
import java.awt.event.*;  
  
public class Temperatura extends Applet  
    implements ActionListener  
{  
  
    Label mens;  
    Button af,ag;  
    TextField temp;
```

Ejemplo (cont.)

```
public void init() {
    mens=new Label ("                ");
    af=new Button("A Fahrenheit");
    ag=new Button("A Grados");
    temp= new TextField("0.0");
    add(af);
    add(ag);
    add(temp);
    add(mens);
    af.addActionListener(this);
    ag.addActionListener(this);
}
```

Ejemplo (cont.)

```
public void actionPerformed(ActionEvent ev) {  
  
    try {  
        double tempActual=Double.parseDouble(temp.getText());  
        String str = ev.getActionCommand();  
        if (str.equals("A Fahrenheit")) {  
            mens.setText("Temperatura "+tempActual+" C = "+  
                (tempActual*1.80+32.0)+" F");  
        } else if (str.equals("A Grados")) {  
            mens.setText("Temperatura "+tempActual+" F = "+  
                ((tempActual-32.0)/1.80)+" C");  
        }  
    } catch (NumberFormatException e) {  
        mens.setText("Error");  
    }  
}
```

Ejemplo (cont.)

```
public void start() {  
}  
  
public void paint(Graphics g) {  
}  
  
}
```

7. Ejemplo: regresión lineal

Se dispone de la clase `RegresionLineal2`, con la siguiente interfaz:

```
public class RegresionLineal2 {  
  
    public class NoCabe extends Exception {}  
  
    public RegresionLineal2(int maxPuntos)  
  
    public void inserta(double x, double y) throws NoCabe  
  
    public double coefA()  
    public double coefB()  
    public double correlacion()  
    public double valorY(double x)  
  
    public void dibuja(Graphics g)  
}
```

Ejemplo (cont.)

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

public class AppletRL extends Applet implements ActionListener {

    Button inserta, calcula;
    TextField num1,num2;
    RegresionLineal rl;
    String msg="";
    boolean calculated=false;
```

Ejemplo (cont.)

```
//Initialize the applet
public void init() {
    try {
        this.setSize(new Dimension(640,480));
        inserta=new Button("Inserta");
        calcula=new Button("Calcula");
        rl=new RegresionLineal(100);
        num1=new TextField(8);
        num2=new TextField(8);
        Label lab1=new Label("X");
        Label lab2=new Label("Y");

        add(lab1);
        add(num1);
        add(lab2);
        add(num2);
        add(inserta);
        add(calcula);
    }
}
```

Ejemplo (cont.)

```
        inserta.addActionListener(this);
        calcula.addActionListener(this);
    }
    catch(Exception e) {
        e.printStackTrace();
    }
}
```

Ejemplo (cont.)

```
public void actionPerformed(ActionEvent ev) {

    double x,y;
    String str = ev.getActionCommand();
    if (str.equals("Inserta")) {
        try {
            x=Double.parseDouble(num1.getText());
            y=Double.parseDouble(num2.getText());
            rl.inserta(x,y);
            msg="Insertados "+num1.getText()+" , "+num2.getText();
        } catch (Exception e) {
            msg="Error en datos          ";
        }

    } else if (str.equals("Calcula")) {
        calculated=true;
    }
    repaint(); // fuerza a que se llame a paint()
}
```

Ejemplo (cont.)

```
public void paint(Graphics g) {  
  
    if (calculated) {  
        g.drawString("Valores obtenidos. A="+r1.coefA()+  
            "    B="+r1.coefB()+  
            "    r="+r1.correlacion(),40,460);  
        r1.dibuja(g);  
    } else {  
        g.drawString(msg,40,60);  
    }  
}  
  
}
```