

# Interfaz gráfico de usuario

El Entorno Integrado de Desarrollo (IDE) dispone de un modo de diseño, similar a otras herramientas de programación de programas gráficos (GUI). En la superficie del applet se disponen los componentes, como paneles y controles. Para ello, se selecciona con el ratón un componente en la paleta correspondiente y se sitúa sobre el applet o sobre otro componente. Ahora bien, la herramienta de diseño no dispone de opciones para situar los componentes en lugares precisos, alinearlos, etc, como ocurre en Windows, ya que en Java existen los denominados gestores de diseño que no sitúan los componentes en posiciones absolutas, sino que la disposición se determina mediante un algoritmo. El más simple de los gestores de diseño es *FlowLayout* y el más complicado es *GridBagLayout*.

Para el programador acostumbrado a diseñar ventanas y diálogos en el entorno Windows le parecerá extraña esta forma de proceder y pensará que con este sistema será difícil elaborar un interfaz gráfico de usuario.

Como veremos, se puede crear un diseño complejo mediante el gestor *GridBagLayout* y también, mediante la aproximación de paneles anidados. En la parte del curso dedicada a estudiar ejemplos completos se describen ampliamente estas dos aproximaciones.

No obstante, el programador debe percibir la diferencia entre applets y aplicaciones. Los applets están insertados en una página web y se ejecutan en la ventana del navegador. El applet comparte espacio con texto, imágenes y otros elementos multimedia. El usuario tiene la libertad de moverse por la página, y por otras páginas a través de los enlaces. La percepción que tiene el usuario del applet es completamente distinta de la percepción que tiene de una aplicación que llama completamente su atención.

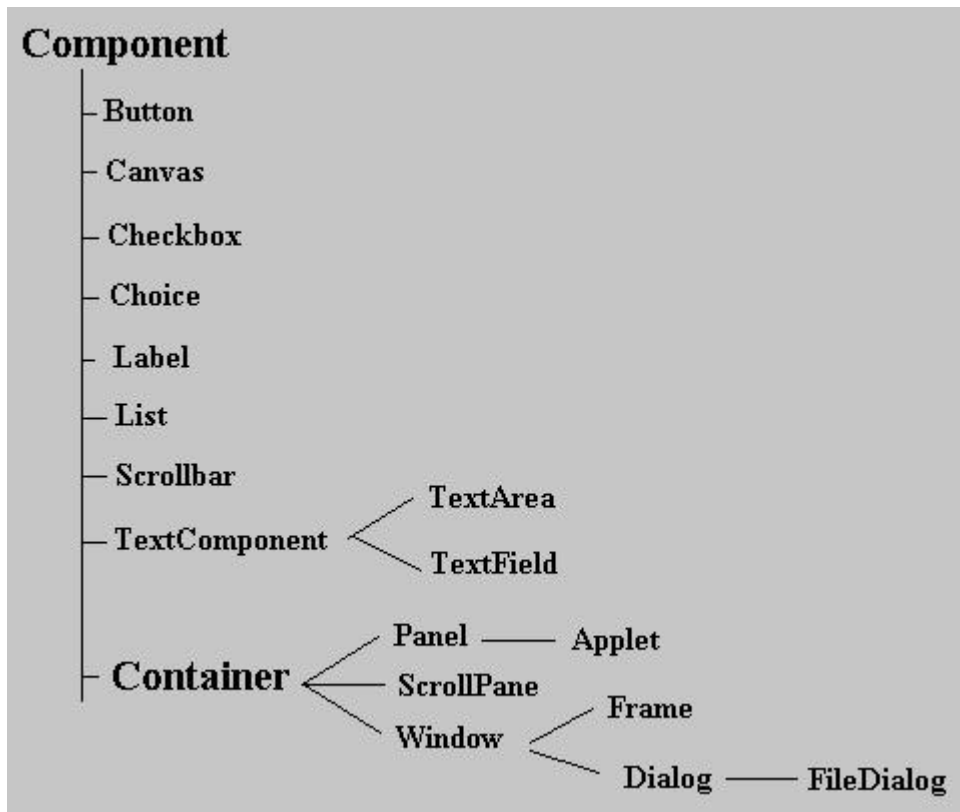
El programador debe tener en cuenta estas diferencias de percepción, y debe esforzarse en crear un diseño de modo que el usuario encuentre evidente el manejo del applet a primera vista.

## La paleta de componentes

Las paletas de componentes están situados en la ventana superior del IDE. Observamos en la paleta que hay dos tipos de componentes:

- los controles
- los paneles

Los controles derivan de la clase *Component*, y los paneles derivan de la clase *Container*. Los paneles (un applet es un panel especializado) pueden contener otros componentes (paneles y controles). La jerarquía de clases se muestra en la figura



La librería AWT no es muy rica en componentes, tal como se ve en la figura, por lo que JBuilder viene acompañado por otras paletas de componentes. Sin embargo, hemos de tener cuidado en usar solamente componentes AWT para los applets que vayamos a publicar en Internet. Los navegadores no ejecutan applets que utilicen componentes no estándar.

Por debajo de los controles AWT están los controles nativos, esto presenta algunas dificultades, por ejemplo, los controles son siempre rectangulares y opacos. La versión Java 2.0 sustituye la librería AWT por la denominada Swing en la que se elimina la dependencia de los controles nativos, el número y el tipo de componentes puede satisfacer con creces las aspiraciones de cualquier programador exigente. Además, podemos crear nuestros propios controles, comprarlos o encargarlos a medida, y situarlos en la paleta de componentes. En el capítulo dedicado a la tecnología de componentes o [javaBeans](#) nos ocuparemos de este asunto.

## Añadir componentes al applet

Una vez que hemos creado el applet con el asistente, se pulsa con el ratón la pestaña **Design** situada en la parte inferior del panel de contenido. Al seleccionar esta pestaña, no cambia el panel de navegación, pero cambia el panel de estructura (inferior izquierda) y el panel de contenido (a la derecha). El panel de estructura nos muestra los componentes **this** (el applet) y los botones que se han situado sobre el applet.

Nos aseguraremos que el applet (**this**) tenga establecido el gestor de diseño *XYLayout*. Este gestor permite el posicionamiento absoluto de los componentes, sirviendo de gran

ayuda en la etapa de diseño. Una vez terminado, hemos de cambiarlo por los gestores estándar que estudiamos en estas páginas.

Cuando se selecciona un componente, por ejemplo un botón, aparece su hoja de propiedades a la derecha. Dicha hoja contiene una lista de propiedades del componente y al lado un editor asociado. Por ejemplo, la propiedad **name** es el nombre con el que se conoce al botón en el código fuente. Podemos cambiar el valor por defecto y poner un nombre significativo. Para los controles se acostumbra poner un nombre que consta de dos partes:

- El prefijo indica el tipo de control. Por ejemplo, **btn** indica botón, **ch**, caja de selección, **list**, lista etc.
- El sufijo indica la funcionalidad o el título del control, *btnAceptar*, para el botón Aceptar, *btnCancelar*, etc.

La propiedad **font** tiene un editor asociado que es un diálogo en el que podemos elegir el tipo de fuente, el tamaño, y el estilo. En este caso elegimos Dialog, 16 puntos, estilo negrita (bold).

La propiedad **label** tiene un editor asociado en el que podemos cambiar el título del botón.

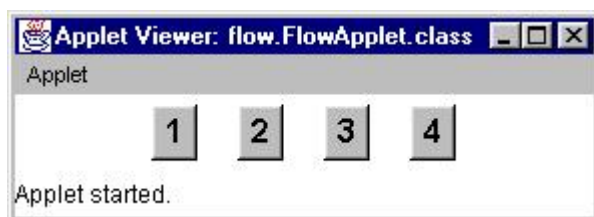
## El gestor *FlowLayout*

Vamos al panel de estructura y situamos el cursor en **this** (el applet) y observamos su hoja de propiedades. La propiedad **layout** tiene un editor asociado que es una caja de selección, elegimos el elemento *FlowLayout*. Veremos como los botones se alinean en el centro y en la parte superior del applet.

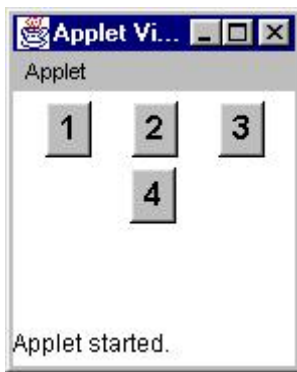
Volvemos al panel de estructura y situamos el cursor en *flowLayout1*, aparece su hoja de propiedades

Podemos cambiar su nombre por defecto, en el editor asociado a la propiedad **name**, el alineamiento (0 es a la izquierda, 1 al centro y 2 a la derecha) y el espaciado horizontal y vertical entre controles, en los editores asociados a las propiedades **alignment**, **hgap** y **vgap**. En este caso hemos elegido un alineamiento en el centro (por defecto) y un espaciado horizontal de 20 y un espaciado vertical de 5 (por defecto).

*FlowLayout* es un gestor que pone los controles en una línea, como puede verse en la figura



Si se cambia el tamaño del applet y los controles no caben en una línea, pasan a la línea siguiente, como puede verse en la figura.



## El código fuente

Si pulsamos con el ratón en la pestaña titulada **Source**, vemos el código fuente que ha generado el IDE.

Los controles son objetos de la clase *Button*, y el gestor de diseño es un objeto de la clase *FlowLayout*. Una vez inicializados los miembros dato, en la función miembro *init* se establecen sus propiedades y se añaden al applet mediante la función *add*, una vez establecido el gestor de diseño mediante *setLayout*. Los pasos son los siguientes

1. Crear los botones (objetos de la clase *Button*) y el gestor de diseño (objeto de la clase *FlowLayout*)

```
Button btn1 = new Button();  
FlowLayout flowLayout1 = new FlowLayout();
```

2. Establecer sus propiedades en *init*

```
btn1.setFont(new Font("Dialog", 1, 16));  
btn1.setLabel("1");  
flowLayout1.setHgap(20);
```

3. Establecer el gestor de diseño del applet (o de un panel) mediante *setLayout*

```
this.setLayout(flowLayout1);
```

4. Añadir los controles al applet (o a un panel) mediante *add*

```
this.add(btn1, null);
```

Lo que se ha dicho para un applet vale para cualquier panel, ya que un applet no es otra cosa que un panel especializado.

```
public class FlowApplet extends Applet {  
    Button btn1 = new Button();
```

```

Button btn2 = new Button();
Button btn3 = new Button();
Button btn4 = new Button();
FlowLayout flowLayout1 = new FlowLayout();

public void init(){
    setBackground(Color.white);
    btn1.setFont(new Font("Dialog", 1, 16));
    btn1.setLabel("1");
    btn2.setFont(new Font("Dialog", 1, 16));
    btn2.setLabel("2");
    btn3.setFont(new Font("Dialog", 1, 16));
    btn3.setLabel("3");
    btn4.setFont(new Font("Dialog", 1, 16));
    btn4.setLabel("4");
    flowLayout1.setHgap(20);
    this.setLayout(flowLayout1);
    this.add(btn1, null);
    this.add(btn2, null);
    this.add(btn3, null);
    this.add(btn4, null);
}
}

```

## El gestor *BorderLayout*

Los pasos para establecer el gestor *BorderLayout* son distintos a los empleados para el gestor *FlowLayout*.

1. Crear los botones (objetos de la clase *Button*) y el gestor de diseño (objeto de la clase *BorderLayout*)

```

Button btnOeste = new Button();
BorderLayout borderLayout1 = new BorderLayout();

```

2. Establecer sus propiedades en *init*

```

btnOeste.setFont(new Font("Dialog", 1, 16));
btn1.setLabel("Oeste");

```

3. Añadir los controles al applet (o a un panel) mediante *add*, indicando en el segundo argumento la posición que ocupará cada control en el panel mediante miembros estáticos de la clase *BorderLayout*.

```

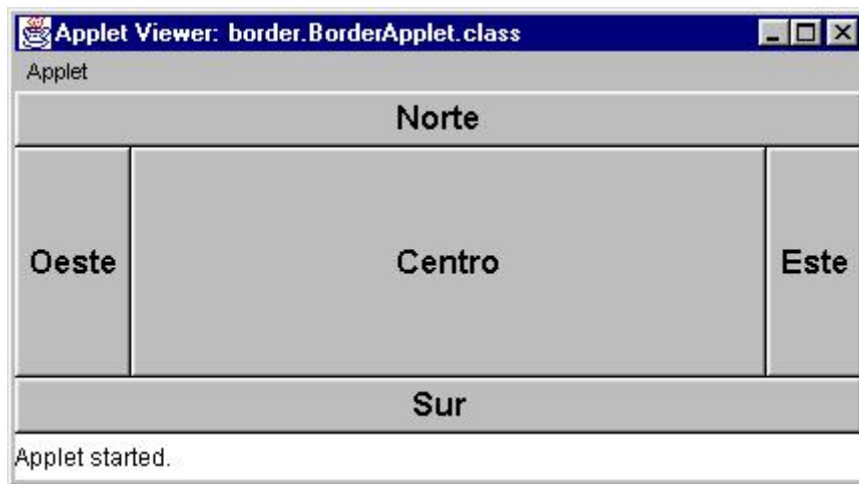
this.add(btnOeste, BorderLayout.WEST);

```

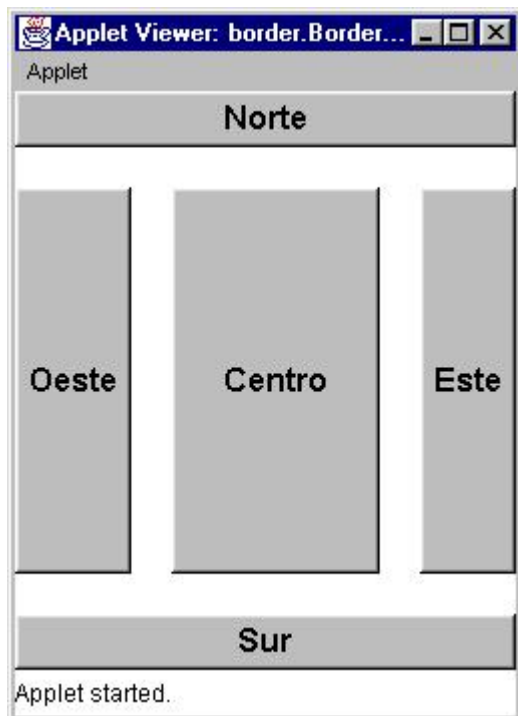
En el código fuente se ha marcado en letra negrita las diferencias y semejanzas entre los distintos gestores.

En este ejemplo, se han creado cinco botones cuyos títulos son *Oeste*, *Norte*, *Sur*, *Este* y *Centro*. Cuando se aplica el gestor *BorderLayout* al applet los cinco botones se

disponen como se muestra en la figura. Si alguno de los botones no está en la posición correcta se puede arrastrar con el ratón a la posición adecuada.



Vamos al panel de estructura y situamos el cursor sobre el gestor *borderLayout1*. En su hoja de propiedades podemos establecer un espaciado entre los botones tal como se muestra en la figura. Con este gestor de diseño los botones ocupan completamente el panel. Cuando se cambia las dimensiones del applet los botones cambian su tamaño para adaptarse a las nuevas dimensiones del panel, tal como puede verse en la figura inferior.



```
public class BorderApplet extends Applet {  
    Button btnOeste = new Button();  
    Button btnEste = new Button();  
    Button btnNorte = new Button();  
    Button btnSur = new Button();  
    Button btnCentro = new Button();  
}
```

```
BorderLayout BorderLayout1 = new BorderLayout();

public void init() {
    setBackground(Color.white);
    this.setSize(new Dimension(336, 253));
    this.setLayout(BorderLayout1);
    btnOeste.setFont(new Font("Dialog", 1, 16));
    btnOeste.setLabel("Oeste");
    btnEste.setFont(new Font("Dialog", 1, 16));
    btnEste.setLabel("Este");
    btnNorte.setFont(new Font("Dialog", 1, 16));
    btnNorte.setLabel("Norte");
    btnSur.setFont(new Font("Dialog", 1, 16));
    btnSur.setLabel("Sur");
    btnCentro.setFont(new Font("Dialog", 1, 16));
    btnCentro.setLabel("Centro");
    BorderLayout1.setVgap(20);
    BorderLayout1.setHgap(20);
    this.add(btnOeste, BorderLayout.WEST);
    this.add(btnEste, BorderLayout.EAST);
    this.add(btnNorte, BorderLayout.NORTH);
    this.add(btnSur, BorderLayout.SOUTH);
    this.add(btnCentro, BorderLayout.CENTER);
}
}
```