

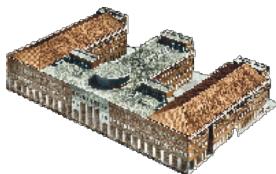
Interacción Persona Computador

Laboratorio - Sesión 1



Diseño de GUI con Java Swing y NetBeans (*)

C. Hernández / M.Gonzalo / A. Martínez



(*) Alguna parte de este material está basado en C.Vaca (2011).

INTRODUCCIÓN

Introducción

- Hace años ...

```
Terminal — ssh — 80x24

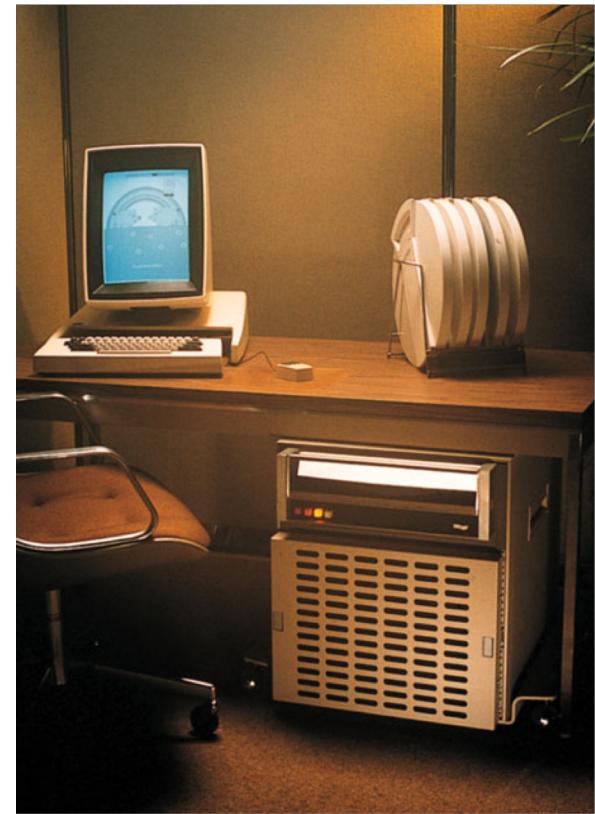
.es
ssh: connect to host saturno.dcs.eup.uva.es port 22: Connection refused
MacBook-de-Alejandra-Martinez-Mones:~ amartine$ ssh amartine@saturno.dcs.eup.uva
.es -p 22222
amartine@saturno.dcs.eup.uva.es's password:
Linux saturno 2.6.26-2-686 #1 SMP Thu Aug 19 03:44:10 UTC 2010 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jan 19 18:37:37 2011 from 239.red-83-61-242.dynamicip.rima-tde.n
et
[amartine@saturno ~]$ ls
11 C cursos Dia1 Ejercicios_Resueltos scripts Tema11
[amartine@saturno ~]$ cd 11
[amartine@saturno 11]$ ls
aleatorios_practicas.c a.out portafolio
[amartine@saturno 11]$ ./a.out
GRUPO GEA3 Miembro: 2
Quiere repetir (0-No / Otro-Si):0
[amartine@saturno 11]$
```

Introducción

- Algo de historia: Xerox Alto (1973)
 - Primer ordenador (no comercializado) que incorpora las ideas de las interfaces gráficas de usuario (GUIs)



Introducción

- ¿Qué implicaciones tiene la introducción de GUI en la programación?

Introducción

- La E/S en consola
 - El control lo tiene el sistema
 - La entrada es síncrona: se produce cuando el sistema se la pide al usuario.
- La E/S a través de interfaces gráficas
 - El usuario tiene el control sobre el sistema
 - Los controles no deben bloquear el sistema, ya que el usuario puede decidir actuar sobre cualquier otro componente de la IU
 - La entrada es asíncrona, compuesta por eventos.

Introducción

- Un programa de consola obedece a este estilo de interacción:

Realizar una tarea

Pedir datos al usuario

Esperar hasta que el usuario introduzca los datos

Procesar la entrada ...

Introducción

- Un programa “interactivo” debe hacer al menos:

```
repetir {  
    esperar la entrada del usuario  
    procesar la entrada  
    (opcionalmente) actualizar pantalla  
} hasta “salir”
```

Introducción

- Las interfaces gráficas de usuario (GUI) ofrecen al usuario ventanas, cuadros de diálogo, barras de herramientas, botones, listas desplegables y otros elementos con los que ya estamos acostumbrados a tratar.
- Su desarrollo tiene algo más de complejidad que la E/S por consola:
 - Los componentes de la interfaz deben ser programados
 - Necesitamos mecanismos para ponerlos en pantalla
 - Necesitamos mecanismos para el control de eventos
- Se desarrollan utilizando APIs (Application Programming Interface)

DISEÑO DE UNA GUI

Pasos en el diseño

1. Decidir la estructura de la interfaz
 - Qué componentes gráficos se van a utilizar
 - Cómo se van a relacionar estos componentes
2. Decidir la disposición (layout) de los componentes
3. Decidir el comportamiento de la interfaz: gestión de eventos
 - Conviene mantener la interfaz y la lógica lo más independientes posibles

Elementos básicos de una GUI

- Componentes GUI (widgets)
 - Un programa gráfico es un conjunto de componentes anidados
 - ventanas, contenedores, menús, barras, botones, campos de texto, etc.
- Disposición (layout): cómo se colocan los componentes para lograr una GUI cómoda de utilizar
 - Layout managers: Gestionan la organización de los componentes gráficos de la interfaz
- Eventos: interactividad, respuesta a la entrada del usuario
 - Desplazamiento del ratón, selección en un menú, botón pulsado, etc.

Componentes

- Existen dos tipos de componentes: contenedores y componentes atómicos
 - Los contenedores sirven para organizar los componentes contenidos en los mismos.

Controles y eventos

- Algunos componentes son controles: permiten reaccionar ante eventos del usuario.
- El comportamiento se especifica programando las respuestas a dichos eventos.
 - Alguna respuesta supondrá invocar funcionalidades de la lógica de la aplicación

Gestión de eventos en GUI

- Un evento es la acción que un usuario realiza sobre la GUI.
 - E.g., pulsar un botón “Salir, elegir una opción de menú, etc.
- La interacción es gestionada mediante “componentes” externos al programa que se encargan de:
 - Detectar eventos
 - Determinar los métodos que están interesados en ellos y/o pueden manejarlos
 - Enviar mensajes a esos métodos o invocar código del programa encargado de manejarlos

Manejadores de eventos (1)

- Se encargan de controlar los eventos, esperan a que el evento se produzca y realizan una serie de acciones. Según el evento, necesitaremos un Listener que lo controle.
- La aplicación registra pares (evento, manejador) para indicar al gestor que invoque al manejador cuando se produzca un evento del tipo determinado.
- En estas interfaces, el bucle de mensajes está oculto al programador.

Manejadores de eventos (2)

- En Java se llaman “Listener”
- El listener de Java tiene una serie de interfaces asociadas a los diferentes eventos
 - ActionListener → Una acción (e.g. click sobre botón)
 - KeyListener → Evento de teclado
 - FocusListener → Cambio en el foco de teclado
 - MouseListener → Evento de ratón
 - ...
- La clase interesada en procesar un evento implementa la interfaz correspondiente, y el objeto creado con esa clase se registra usando el método “addActionListener” .
- Cuando el evento ocurre, se invoca el método “actionPerformed” de este objeto

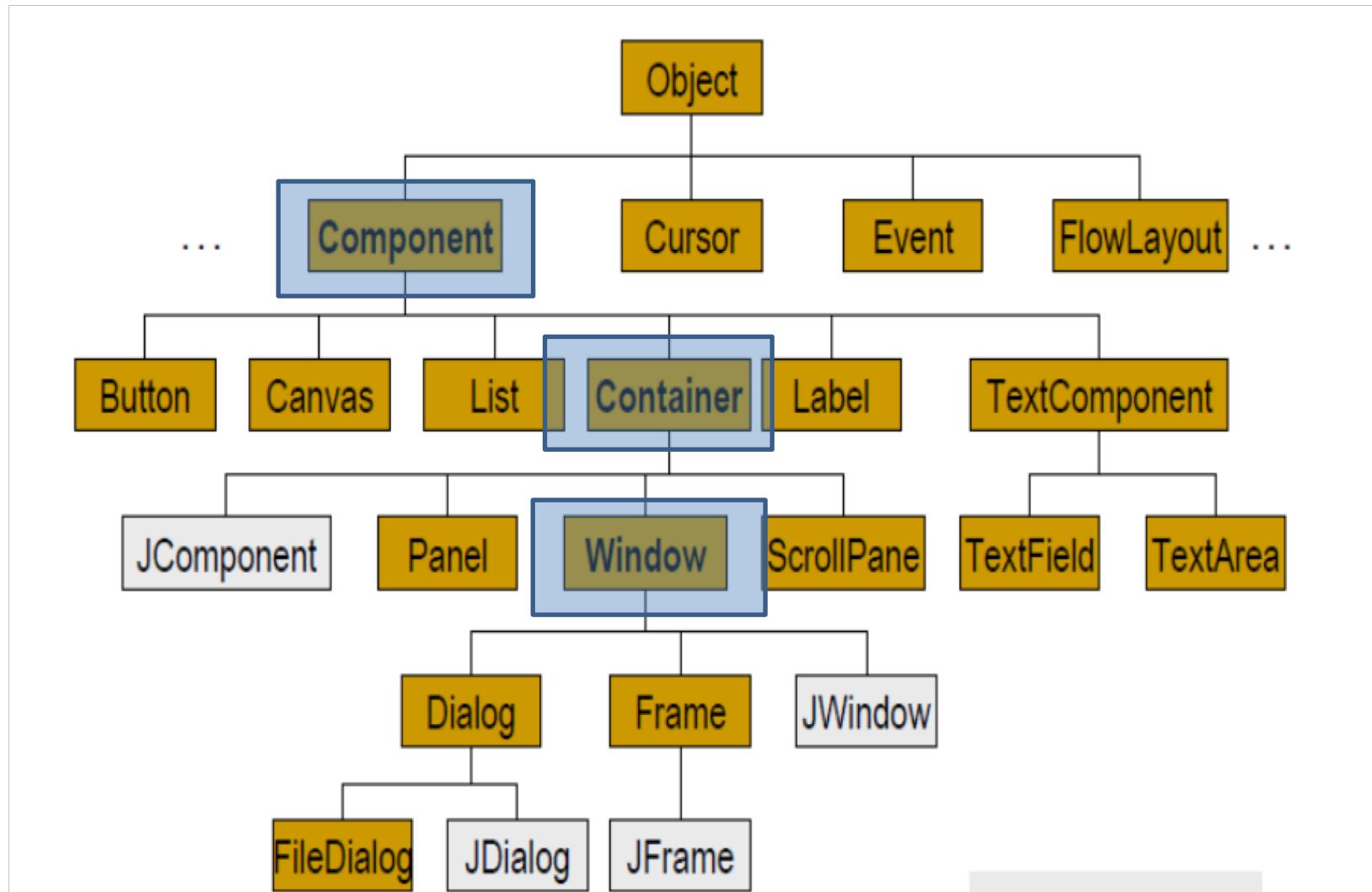
COMPONENTES DE UNA GUI

Con especial atención a Java AWT y Java Swing

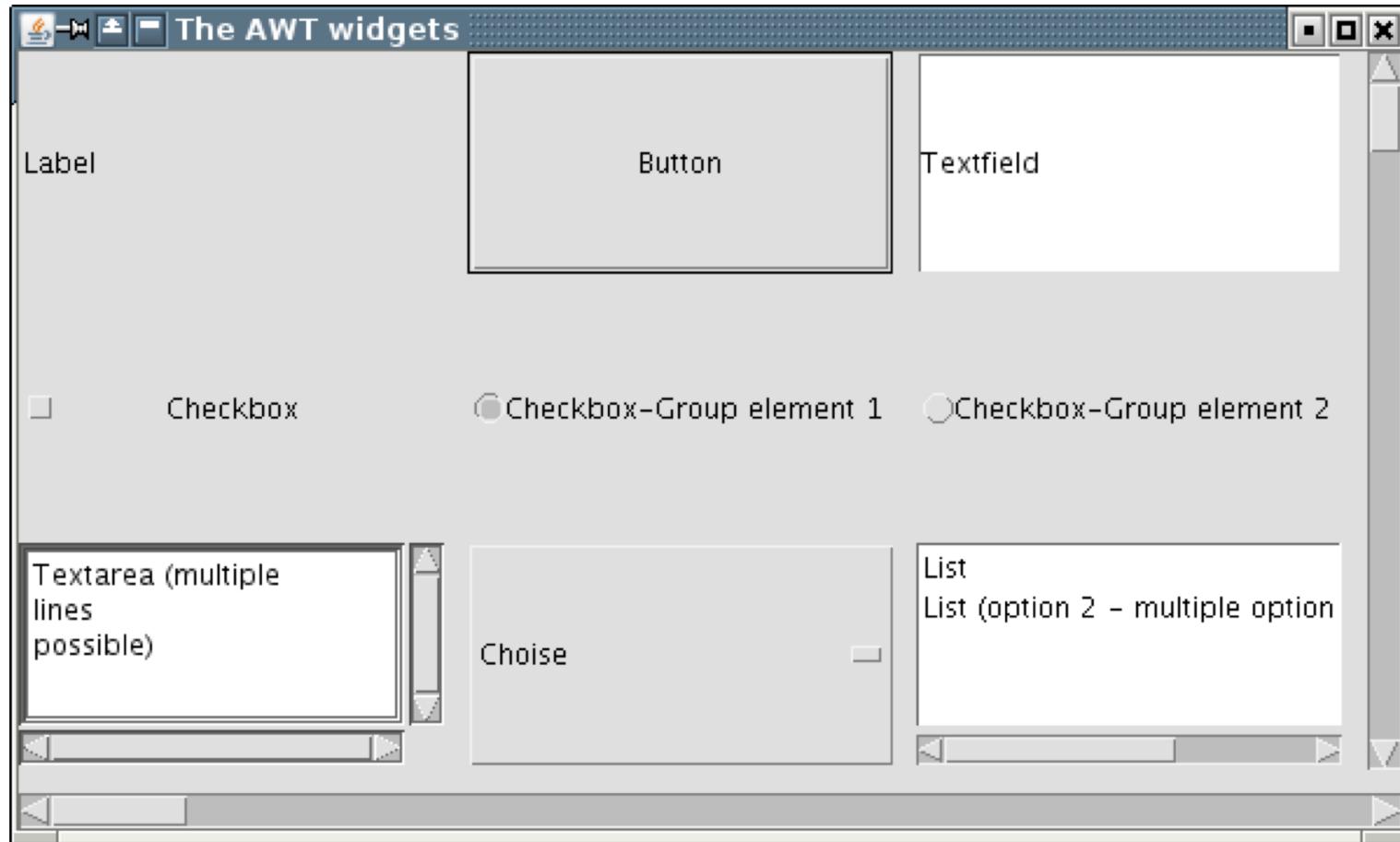
GUI en Java

- Desde la primera versión de Java, se añadió un toolkit para crear interfaces de usuario llamado AWT (Abstract Windowing Toolkit)
 - AWT era parte del JDK (Java Foundation Classes)
 - Su funcionalidad dependía del sistema operativo
 - Pero no soportaba interfaces complejas

Componentes AWT



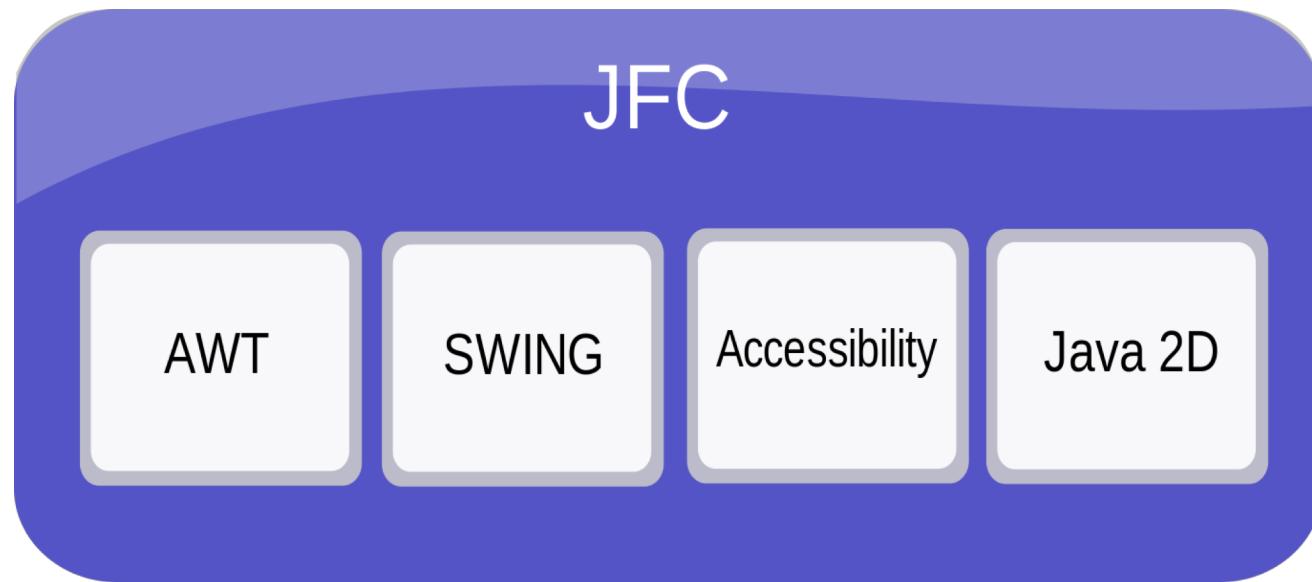
Componentes AWT (ejemplos)



Fuente: Wikipedia

Desarrollo de GUI en Java

- En 1997, Java anunció la JFC (Java Foundation Classes), un marco para el desarrollo de GUI
- La mayor parte de JFC era un nuevo conjunto de componentes de interfaz llamado Swing



Características de Swing (1 / 2)

- Independiente de plataforma
 - porque está escrito totalmente en Java
- Arquitectura altamente modular
 - ‘Framework’ basado en componentes
- Utiliza el patrón de diseño de MVC
 - Separación entre la vista y la lógica
- Personalizable:
 - Plaf: Pluggable “Look and Feel”
- No reemplaza a AWT. Se apoya en él.
 - Añade JComponents

Características de Swing (2 / 2)

- Lightweight UI: no depende de la IU del sistema operativo para “pintar” los componentes.
 - Aparecen en cada plataforma tal y como fueron diseñados.
 - Casi todos los componentes de Swing son lightweight (ligeros) excepto:
 - JApplet, JFrame, JDialog, JWindow

¿Applet o Aplicación?

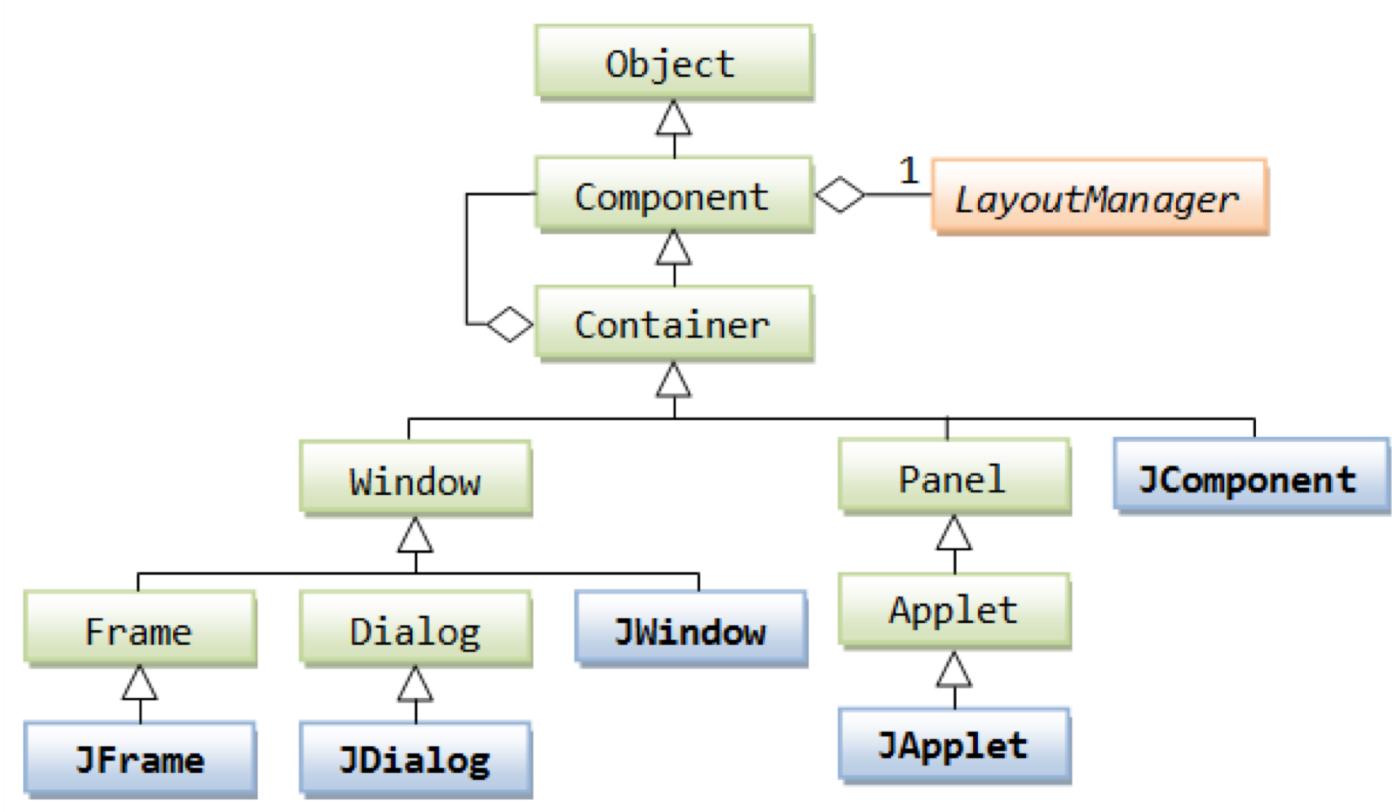
- Con Swing se puede crear dos tipos de GUIs.
- Aplicación independiente
 - Solo necesitan la máquina virtual de java (JVM)
 - Código reside en la máquina en la que se ejecuta
 - La aplicación se ejecuta con `java <ClaseMain>`
 - La aplicación finaliza cuando se cierra la ventana
- Applet
 - Programa se ejecuta dentro de un navegador
 - Código es descargado desde el servidor web
 - JVM está contenido dentro del navegador
 - Por problemas de seguridad no pueden realizar ciertas tareas.

Elementos de Swing

- Componentes
 - Component - Superclase de todas las clases de la interfaz gráfica.
 - Elementos visuales de la interfaz
- Layout Managers
 - Administradores de diseño - Gestionan la posición relativa y el tamaño de los componentes de una ventana
- Containers
 - Clase abstracta, que representa un componente que puede contener otros componentes gráficos

Jerarquía de clases Swing

Top-level containers

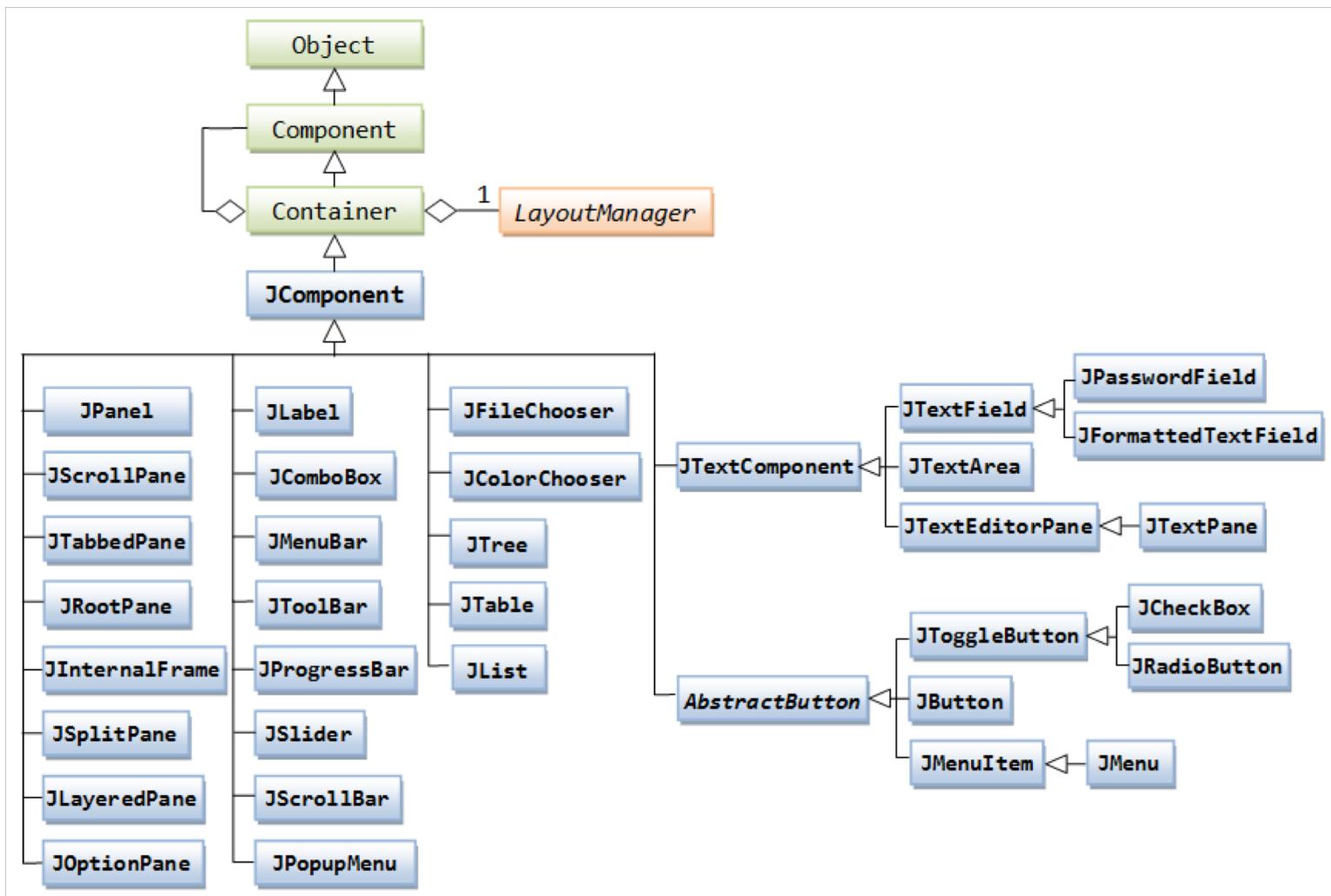


http://pervasive2.morselli.unimo.it/~nicola/courses/IngegneriaDelSoftware/java/J4a_GUI_2.html

Contenedores: tipos

- Top-level contenedores
 - No pueden ser añadidos a otros contenedores
 - JFrame, se emplea para crear la ventana principal de una aplicación en Swing.
 - JDialog, para ventanas de interacción con el usuario.
 - JApplet
- Contenedores intermedios:
 - Agrupan componentes para que puedan ser manejados como uno solo
 - JPanel, agrupa a otros componentes.
 - JScrollPane, incluye barras de desplazamiento.
 - Etc.

Jerarquía de clases Swing: JComponent

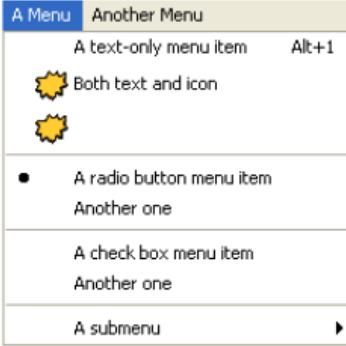
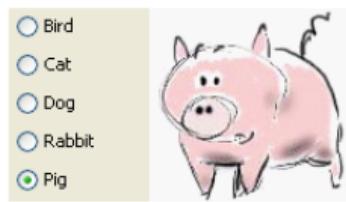


http://pervasive2.morselli.unimo.it/~nicola/courses/IngegneriaDelSoftware/java/J4a_GUI_2.html

Componentes

- Son componentes básicos que no contienen otros componentes
 - JLabel, JButton, JTextField, JTextArea, ...
- Clases de soporte:
 - Graphics, Color, Font, ...
- Widgets
 - label
 - button
 - checkbox
 - choice
 - texfield ...

Elementos de Swing

 JButton	 JCheckBox	 JComboBox	 JList
 JMenu	 JRadioButton	 JSlider	
 JSpinner	 JTextField	 JPasswordField	

Disposición de los componentes

- La “disposición” (“layout”) determina el tamaño y posición de los componentes en sus contenedores.
- Disposición declarativa
 - Java: Layout managers – LayoutManager interface
 - Web: CSS y estilos
- Disposición procedural
 - Escribir código para calcular posiciones y tamaños
- Asistida por los IDE
 - NetBeans – Free Design

Diseño de GUI con NetBeans

NetBeans

- IDE desarrollado en Java
- Vamos a trabajar con la versión instalada en Linux
- Descripción de las principales partes de la herramienta
- Realización del tutorial recomendado
 - <https://netbeans.org/kb/docs/java/gui-functionality.html>
 - <https://netbeans.org/kb/docs/java/quickstart-gui.html> (opcional)

Resumen

- Las GUI suponen un cambio en la forma de diseñar y programar, puesto que la Entrada/Salida del usuario sigue otros principios diferentes a la basada en consola
- Las GUI se estructuran alrededor de componentes
 - Los componentes encapsulan algunos patrones de diseño software como el “listener”, “MVC”, “comando” ...
 - Proporcionan una forma estandarizada de organizar la interfaz
- Existen diferentes tipos de componentes, y su elección depende de diferentes aspectos
 - Tarea para la que se usan, espacio en pantalla, eficiencia ...