

GUI in Java con l'AWT – 2

Stefano Mizzaro

Dipartimento di matematica e informatica
Università di Udine
<http://www.dimi.uniud.it/mizzaro/>
mizzaro@uniud.it
Programmazione, lezione 21
18 maggio 2015

Stefano Mizzaro - AWT2

3

Riassunto

- GUI in Java: AWT
- Concetti:
 - Componenti
 - Eventi
 - Ascoltatori
 - (Layout)
- Documentazione
 - Libri, Fotocopie
 - (Tutorial Sun)

Stefano Mizzaro - AWT2

2

Cosa manca

- Associazioni componenti – eventi
- Gestione layout
- Campionario componenti
 - (Component, Container, Window)
 - Frame, Panel, Button, Label, TextComponent, TextArea, TextField, (Canvas)
 - Per ogni componente:
 - Cosa fa/è, quali eventi genera, esempi di uso
 - Classi "ausiliarie": Color, (Font), Dimension
- (Disegno: Canvas e paint(). Applet)
- Eh...

Stefano Mizzaro - AWT2

3

Scaletta

- Associazione componenti ↔ eventi
- Layout
- Campionario di componenti

Stefano Mizzaro - AWT2

4

Nomi

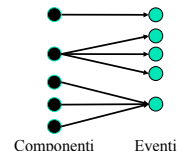
- Componenti
 - Button
 - Canvas
 - Checkbox
 - CheckboxMenuItem
 - Choice
 - Component
 - Container
 - Dialog
 - Frame
 - Label
 - List
 - MenuItem
 - Panel
 - Scrollbar
 - TextArea
 - TextField
 - Window
- Eventi
 - ActionEvent
 - AdjustmentEvent
 - ComponentEvent
 - ContainerEvent
 - FocusEvent
 - ItemEvent
 - KeyEvent
 - MouseEvent
 - MouseMotionEvent
 - TextEvent
 - WindowEvent

Stefano Mizzaro - AWT2

5

Componenti-eventi

- Quali componenti generano quali eventi?
- Ogni componente può generare più eventi
- Ogni evento può essere generato da più componenti



- E poi c'è anche l'ereditarietà...

Stefano Mizzaro - AWT2

6

Componenti, eventi ed eredità

- Eredità fra i componenti (Component genera eventi key ⇒ tutti i componenti lo faranno)
- Eredità fra gli eventi (se un componente genera eventi key ⇒ genera anche i "sopraeventi")

Componenti Eventi

Componenti Eventi

Stefano Mizzaro - AWT2 7

Componenti – eventi (1/2)

	action	adjustment	component	container	focus	item	key	mouse	mouse motion	text	window
Button	●		•		•		•	•	•		
Canvas			•		•		•	•	•		
Checkbox			•		•	•	•	•	•		
CheckboxMenuItem						•					
Choice			•		•	•	•	•	•		
Component			•		•		•	•	•		
Container			•	•	•		•	•	•		
Dialog			•	•	•		•	•	•		•

Componenti – eventi (2/2)

	action	adjustment	component	container	focus	item	key	mouse	mouse motion	text	window
Frame			•	•	•		•	•	•		●
Label			•		•		•	•	•		
List	•		•		•	•	•	•	•		
MenuItem	•										
Panel			•	•	•		•	•	•		
Scrollbar		•	•		•		•	•	•		
TextArea			•		•		•	•	•	•	
TextField	•		•		•		•	•	•	•	
Window			•	•	•		•	•	•		•

Stefano Mizzaro - AWT2 10

Componenti - Eventi - Ascoltatori

- L'evento XxxEvent è gestito da XxxListener
- XxxListener va registrato nel componente
 - addXxxListener sono definiti nelle (ed ereditati da) classi "opportune":
 - addWindowListener è definito in Window ed ereditato da Frame e Dialog
 - addActionListener è definito in Button, List, MenuItem, TextField
- ...vedi stampe... e javadoc

Stefano Mizzaro - AWT2 10

Scaletta

- Associazione componenti ↔ eventi
- Layout
- Campionario di componenti

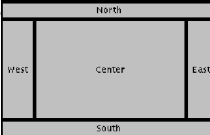
Stefano Mizzaro - AWT2 11

I gestori di layout

- Posizionamento dei componenti in un contenitore
- Ci sono vari gestori di layout (implementano **LayoutManager**)
 - Ne vediamo due: BorderLayout e FlowLayout
- C'è un gestore di layout di default per ogni componente
- Lo si può cambiare con il metodo **setLayout (LayoutManager)**
 - in Container
 - (ovviamente: è un contenitore)
 - E quindi in tutte le sottoclassi...

Stefano Mizzaro - AWT2 12

BorderLayout



- 5 posizioni (N, S, E, W, C)
 - `BorderLayout.NORTH` (`SOUTH EAST WEST CENTER`)
- Il **Center** cerca di essere più grande possibile
- I componenti vengono "stirati"
- Default di **Frame**
- Vediamo la documentazione
- Vediamo un esempio (`ProvaLayout.java`)

Stefano Mizzaro - AWT2 13

FlowLayout

- Come le parole di un testo
- Centrato, ma si può cambiare
 - parametro costruttore: `FlowLayout.LEFT`, ...
 - (sono `int`, cfr. codice sorgente API)
- Default di **Panel**
- Vediamo la documentazione
- I componenti **non** vengono stirati (utile mettere un **Panel** in un **Frame**...)
- Vediamo un esempio (`ProvaFlowLayout.java`)

Stefano Mizzaro - AWT2 14

Esempio

```

...
Panel p = new Panel();
p.setLayout(new BorderLayout());
p.add(new Label("Nord"), BorderLayout.NORTH);
p.add(new Label("Sud"), BorderLayout.SOUTH);
p.add(new Label("Est"), BorderLayout.EAST);
p.add(new Label("Ovest"), BorderLayout.WEST);
p.add(new Label("Centro"), BorderLayout.CENTER);
...

```

Stefano Mizzaro - AWT2 15

Pregi dei layout

- Meglio del posizionamento assoluto
 - Distanza in pixel dall'angolo in alto a sx.
- Il layout manager posiziona e dimensiona automaticamente (quando serve) i componenti in un contenitore
- NON** serve invocare i metodi di un gestore di layout: vengono invocati automaticamente quando necessario

Stefano Mizzaro - AWT2 16

Scaletta

- Associazione componenti ↔ eventi
- Layout
- Campionario di componenti

Stefano Mizzaro - AWT2 17

Tutti i componenti

<ul style="list-style-type: none"> Button (Canvas) Checkbox CheckboxGroup Choice Dialog Frame Label List Panel Scrollbar ScrollPane 	<ul style="list-style-type: none"> TextComponent TextArea TextField Componenti per menu: <ul style="list-style-type: none"> Menu MenuComponent MenuItem CheckboxMenuItem PopupMenu MenuBar MenuShortcut
---	---

Stefano Mizzaro - AWT2 18

abstract Component

- Il componente più generico (sopraclasse)
- Metodi principali (disponibili anche per sottoclassi):
 - `void setVisible(boolean)`
 - `void setBounds(int int int int)`
 - `Dimension getSize()`
 - `void setSize(Dimension d)`
 - `void setSize(int width, int height)`
 - `void setFont(Font)`
 - `void setBackground(Color) (Foreground)`
 - `Color getBackground() (Foreground)`
 - `void addXxxListener (XxxListener)`

Stefano Mizzaro - AWT2 19

3 classi "ausiliarie"

- **Color**
- **Dimension**
- **(Font)**

Stefano Mizzaro - AWT2 20

Color

- Ogni istanza rappresenta un colore
- Definisce alcuni colori "già pronti" (costanti `Color.BLACK`, `Color.WHITE`, ...)
- Vedi documentazione API
- Metodi: costruttori (sovraccarichi: RGB,...)

```
Component c;
c = new ...;
c.setBackground(new Color(0,0,255));
c.setBackground(Color.BLUE);
```

Stefano Mizzaro - AWT2 21

Dimension

- Larghezza (**width**) e altezza (**height**) di un **Component** in un unico oggetto
- Metodi:
 - Costruttore
 - `double getWidth()`
 - `double getHeight()`
- Vedi documentazione API
- Vedi anche il sorgente `Dimension.java`

Stefano Mizzaro - AWT2 22

abstract Container

- Contenitore di componenti
 - Vedi sorgente: lista di componenti...
- Metodi:
 - `void add(Component)`
 - `void setLayout(LayoutManager)`
 - `void addContainerListener(...)`

Stefano Mizzaro - AWT2 23

Window

- Finestra generica
- Di solito non si usa, si usano le sottoclassi
 - Che ereditano alcuni metodi utili
- Metodi
 - `void pack()`
 - `void dispose()`
 - `void addWindowListener()`

Stefano Mizzaro - AWT2 24

Frame

- Finestra principale di un'applicazione
- Metodi principali:
 - Costruttore
 - `void setTitle(String)`
 - `void setMenuBar(MenuBar)`
- Uso tipico: già visto...

Stefano Mizzaro - AWT2 25

Panel

- Contenitore generico: usato per contenere altri componenti
- Panel dentro Frame**
- Panel dentro Panel**
- Metodi:
 - Costruttore:
 - `public Panel()`
 - `public Panel(LayoutManager)`
- Layout di default: **FlowLayout**

Stefano Mizzaro - AWT2 26

Uso tipico di Panel

```
import java.awt.*;
class ProvaPanel extends Frame {
    public ProvaPanel () {
        super("Layout");
        add(new Button("Nord"), BorderLayout.NORTH);
        add(new Button("Sud"), BorderLayout.SOUTH);
        add(new Button("Est"), BorderLayout.EAST);
        add(new Button("Ovest"), BorderLayout.WEST);
        Panel p = new Panel();
        p.add(new Button("Centro"));
        add(p, BorderLayout.CENTER);
        setVisible(true);
        pack();
    }
    ...
}
```

`add(new Button("Centro"), BorderLayout.CENTER);`

Stefano Mizzaro - AWT2 27

Button

- Pulsante che l'utente può premere
- Con etichetta testuale
- Quando premuto genera un **ActionEvent**
- Che va gestito con un **ActionListener**
- L'**ActionEvent** ha il metodo **getActionCommand()** che restituisce una **String** che può essere usata per individuare il bottone
 - Anche se di solito meglio 1 pulsante 1 ascoltatore

Stefano Mizzaro - AWT2 28

Metodi di Button

- `setLabel` (anche con il costruttore)
- `getLabel`
- `setActionCommand` (di default la label)
- `getActionCommand`
- `addActionListener(ActionListener l)`
- ...

Stefano Mizzaro - AWT2 29

Uso tipico di Button (già visto)

```
class ... {
    ...
    Button b = new Button("Exit");
    this.add(b);
    Ascoltatore a = new Ascoltatore();
    b.addActionListener(a);
    ...
    class Ascoltatore implements ActionListener {
        public void actionPerformed(ActionEvent e) {
            //... Cosa fare quando si preme il pulsante
            //qui ci può essere e.getActionCommand()
        }
    }
}
```

Stefano Mizzaro - AWT2 30

Label

- Etichetta, testo non selezionabile
- Presenta informazioni
 - (poche informazioni. Se di più, c'è `TextArea` non editabile, vedi dopo)
- Genera solo gli eventi di `Component`
- Di solito non ascoltati
 - (serve solo a dare informazioni all'utente)

Stefano Mizzaro - AWT2 31

Metodi di Label

- `void setText(String)`
 - (anche costruttore)
- `void setAlignment(int)`
 - `Label.LEFT/CENTER/RIGHT` (anche costruttore)
- `String getText()`
- `int getAlignment()`

Stefano Mizzaro - AWT2 32

TextComponent

- Generico componente contenente testo
- Costruttore con accessibilità di package, quindi non accessibile dalla ns. applicazione
 - (ma accessibile dalle altre classi di `java.awt`)
 - Non è astratta, ma non la possiamo istanziare
- Genera eventi `TextEvent` gestiti con `TextListener` (`textValueChanged()`)
- Si usano le due 2 sottoclassi (che ereditano):
 - `TextArea`
 - `TextField`

Stefano Mizzaro - AWT2 33

Metodi di TextComponent

- `void addTextListener(TextListener)`
- `String getText()`
- `void setText(String)`
- `void setEditable(boolean)`
- `boolean isEditable()`
- `String getSelectedText()`
- `void selectAll()`
- ...

Stefano Mizzaro - AWT2 34

TextArea

- Area di testo su più righe
- Genera gli stessi eventi di `TextComponent`

Stefano Mizzaro - AWT2 35

Metodi di TextArea

- Quelli di `TextComponent`
- Altri (vedi javadoc API):
 - Costruttori
 - `void append(String)`
 - `int getColumns()`
 - `int getRows()`
 - `void setColumns(int)`
 - `void setRows(int)`

Stefano Mizzaro - AWT2 36

Uso tipico di `TextArea`

```
class ... {
    TextArea t = new TextArea("Testo iniziale",5,20);
    ...
    class AscoltatoreTesto implements TextListener {
        public void textValueChanged(TextEvent e) {
            if(t.getText().equals("Basta"))
                t.setEditable(false);
        }
    }
    ...
    ... {
        t.addTextListener(new AscoltatoreTesto());
    }
}
```

- (Cfr. `ProvaTextArea.java`)
- Oppure `setText(...)` per visualizzare messaggi su un `TextArea` non editabile

Stefano Mizzaro - AWT2 37

`TextField`

- Campo di testo (1 riga)
- Oltre agli eventi di `TextComponent`
- Genera anche `ActionEvent` quando si preme `return`
 - E il `getActionCommand()` restituisce il testo inserito

Stefano Mizzaro - AWT2 38

Uso tipico di `TextField` (1/2)

```
import java.awt.*;
import java.awt.event.*;
class MyFrame extends Frame {
    private TextField f;
    private TextArea t;
    public MyFrame () {
        super("Prova Text");
        t = new TextArea("Hai inserito:\n",5,20);
        t.setEditable(false);
        f = new TextField("Inserisci testo");
        f.selectAll();
        f.addActionListener(new Ascoltatore());
        this.add(f, BorderLayout.NORTH);
        this.add(t, BorderLayout.CENTER);
        this.pack();
        this.setVisible(true);
    }
    ...
}
```

Stefano Mizzaro - AWT2 39

Uso tipico di `TextField` (2/2)

```
...
class Ascoltatore implements ActionListener {
    public void actionPerformed(ActionEvent e) {
        t.append(e.getActionCommand() + "\n");
        f.selectAll();
    }
}

public class ProvaText {
    public static void main (String[] args) {
        MyFrame f = new MyFrame();
    }
}
```

- Nota: comodità della classe interna

Stefano Mizzaro - AWT2 40

Senza classe interna (IMP!)

```
class MyFrame {
    ...
    f.addActionListener(new Ascoltatore(f, t));
    ...
}

class Ascoltatore implements ActionListener {
    private TextField f;
    private TextArea t;
    public Ascoltatore (TextField f, TextArea t) {
        this.f = f; this.t = t;
    }
    public void actionPerformed(ActionEvent e) {
        t.append(e.getActionCommand() + "\n");
        f.selectAll();
    }
}
```

Stefano Mizzaro - AWT2 41

Altri componenti (che non abbiamo visto...)

- `Canvas`
- `Checkbox`
- `CheckboxGroup`
- `Choice`
- `Dialog`
- `List`
- `Scrollbar`
- `ScrollPane`
- Componenti per menu
 - `MenuComponent`
 - `MenuItem`
 - `Menu`
 - `CheckboxMenuItem`
 - `PopupMenu`
 - `MenuBar`
 - `MenuShortcut`

Stefano Mizzaro - AWT2 42



Riassunto

- Associazione componenti ↔ eventi
- Layout
- Campionario di componenti
 - (Component, Container, Window)
 - Frame, Panel, Button, Label, TextComponent, TextArea, TextField
 - Per ogni componente:
 - Cosa fa/è, quali eventi genera, esempi di uso
- Classi "ausiliarie"
 - Color, (Font), Dimension
- Fate prove ed esercizi!
 - Anche con i componenti non visti!

Stefano Mizzaro - AWT2 43