

## Gli applet

Stefano Mizzaro

Dipartimento di matematica e Informatica  
Università di Udine  
<http://www.dimi.uniud.it/~mizzaro>  
mizzaro@dimi.uniud.it  
21 maggio 2003

## Scaletta

- Gli applet
  - Esempi
  - Restrizioni
- Eventi del mouse
- Choice
- Menu

Stefano Mizzaro - Applet

2/25

## Gli applet

- Applet = “piccola applicazione”
- Non un programma indipendente, ma un programma integrato in una pagina web, in esecuzione sulla JVM del browser
- Restrizioni di sicurezza

Stefano Mizzaro - Applet

3/25

## La classe Applet

- `java.applet.Applet` è una sottoclasse di `java.awt.Panel`
- Da sovrascrivere per creare i nostri applet
- Non c'è un `main`:
  - i metodi di `Applet` vengono chiamati (implicitamente, dal browser)
  - e vanno quindi sovrascritti dal programmatore

Stefano Mizzaro - Applet

4/25

## Il nostro primo applet

```
import java.applet.*;
import java.awt.*;
public class CiaoATuttiApplet extends Applet {
    public void paint (Graphics g) {
        g.drawString("Ciao a tutti!", 5, 25);
    }
}
```

Stefano Mizzaro - Applet

5/25

## Il file html

```
<html>
<head>
    <title>Salve a tutti voi</title>
</head>
<body>
    <p>Il mio primo applet dice:
        <applet code="CiaoATuttiApplet.class"
            width="150" height="25">
    </applet></p>
</body>
</html>
```

Stefano Mizzaro - Applet

6/25

## Come funziona

- Il browser ha una JVM, in grado di eseguire codice java
- Abbiamo semplicemente sovrascritto `paint...`
- ...che viene invocato implicitamente

Stefano Mizzaro - Applet

7/25

## L'appletviewer

- Altro "attrezzo" del JDK
- Visualizza applet, ma non html
- Ha bisogno di un file con il tag "applet"
- ...a volte i browser fanno scherzi strani...

Stefano Mizzaro - Applet

8/25

## Uso dell'appletviewer

```

/*<applet code="CiaoATuttiApplet.class"
   width="150" height="25"></applet>*/
import java.applet.*;
import java.awt.*;
public class CiaoATuttiApplet extends Applet {
    public void paint (Graphics g) {
        g.drawString("Ciao a tutti!",5,25);
        g.fillOval(0,0,100,100);
    }
}

```

```

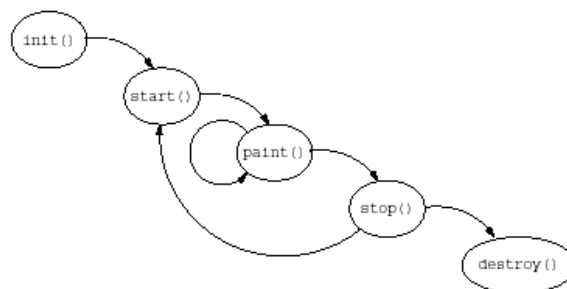
>javac CiaoATuttiApplet.java
>appletviewer CiaoATuttiApplet.java

```

Stefano Mizzaro - Applet

9/25

## "Ciclo di vita" di un applet



Stefano Mizzaro - Applet

10/25

## Metodi di applet

- Metodi (vuoti) da Applet :
  - `init`: inizializzazione (~costruttore)
  - `start`: inizio esecuzione
  - `stop`: fine esecuzione, arresto
  - `destroy`: rilascio risorse
  - (leggere variazioni fra i vari browser...)
- Da Component:
  - `paint`: quello che l'applet mostra

Stefano Mizzaro - Applet

11/25

## Invocazione metodi di Applet

```

/*<applet code="MetodiApplet.class" width="10"
   height="10"></applet>*/
import java.applet.Applet;
import java.awt.*;
public class MetodiApplet extends Applet {
    public void init() { System.err.println("init..."); }
    public void start() { System.err.println("start..."); }
    public void stop() { System.err.println("stop..."); }
    public void destroy() {System.err.println("destroy..."); }
    public void paint(Graphics g) {
        System.err.println("paint...");
    }
}

```

```

>javac MetodiApplet.java
>appletviewer MetodiApplet.java
(e con un browser...)

```

Stefano Mizzaro - Applet

12/25

## Applet per disegnare col mouse

- Vediamo un applet che ci consente di disegnare trascinando il mouse
- Avremo bisogno di:
  - Un applet contenente un Canvas
  - Ascoltatori di **eventi sul mouse** (click e drag) associati al Canvas
- Schema di disegno differente:
  - paint non disegna
  - Al drag, nell'ascoltatore, disegno un segmento

Stefano Mizzaro - Applet

13/25

## ScribbleApplet1.java (1/2)

```

/*<applet code="ScribbleApplet1.class"
   width="500" height="400"></applet>*/
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;

public class ScribbleApplet1 extends Applet {
    private Canvas c = new MyCanvas();
    public void init(){
        this.setLayout(new BorderLayout());
        this.add(c, BorderLayout.CENTER);
    }
}

```

Stefano Mizzaro - Applet

14/25

## ScribbleApplet1.java (2/2)

```

class MyCanvas extends Canvas {
    private int lastX, lastY;
    public MyCanvas() {
        addMouseListener(new AscoltaClick());
        addMouseMotionListener(new AscoltaDrag());
    }
    class AscoltaClick extends MouseAdapter {
        public void mousePressed(MouseEvent e) {
            lastX = e.getX(); lastY = e.getY();
        }
    }
    class AscoltaDrag extends MouseMotionAdapter {
        public void mouseDragged(MouseEvent e) {
            int x = e.getX(); int y = e.getY();
            getGraphics().drawLine(lastX, lastY, x, y);
            lastX = x; lastY = y;
        }
    }
}

```

```

>javac ScribbleApplet1.java
>appletviewer ScribbleApplet1.java

```

Stefano Mizzaro - Applet

16/25

## Eventi mouse(1/2)

- Non rispettano la regola dell'AWT "1 evento ↔ 1 ascoltatore"
- Una classe per gli eventi:
  - **MouseEvent**
- Due interfacce per gli ascoltatori:
  - **MouseListener** e **MouseMotionListener**
- Ci sono anche gli adapter:
  - **MouseAdapter** e **MouseMotionAdapter**

## Eventi mouse(2/2)

- **MouseListener**
  - Pulsante mouse premuto (**mousePressed**)
  - Pulsante mouse rilasciato (**mouseReleased**)
  - Click (premutato + rilasciato) (**mouseClicked**)
  - Cursore entra nel componente (**mouseEntered**)
  - Cursore lascia il componente (**mouseExited**)
- **MouseMotionListener**
  - Mouse mosso (**mouseMoved**)
  - Mouse trascinato (**mouseDragged**)
- **Component:**
  - **addMouseListener** e **addMouseMotionListener**

Stefano Mizzaro - Applet

17/25

## Aggiungiamo i colori

- Scelta del "colore della penna"
- Potremmo farlo con dei pulsanti, 1 per colore
- Ma non è il componente giusto
- Usiamo invece `java.awt.Choice`

Stefano Mizzaro - Applet

18/25

## ScribbleAppletColor (1/2)

```

/*<applet code="ScribbleAppletColor.class"
   width="500" height="400"></applet>*/
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
public class ScribbleAppletColor extends Applet {
    private Canvas c = new MyCanvas();
    private Choice ch = new Choice();
    private static final Color[] colors =
        {Color.BLACK, Color.RED, Color.GREEN, Color.BLUE};
    private static final String[] colorNames =
        {"Nero", "Rosso", "Verde", "Blu"};
    public void init(){
        this.setLayout(new BorderLayout());
        this.add(c, BorderLayout.CENTER);
        for (int i = 0; i < colorNames.length; i++)
            ch.add(colorNames[i]);
        ch.addItemListener(new AscoltaColore());
        this.add(ch, BorderLayout.EAST);
    }
}

```

## ScribbleAppletColor(2/2)

```

class AscoltaColore implements ItemListener {
    public void itemStateChanged(ItemEvent e) {
        c.setForeground(colors[ch.getSelectedIndex()]);
    }
}

class MyCanvas extends Canvas {
    // idem...
}

```

Stefano Mizzaro - Applet

20/25

## Choice

- “Scelta” di un’alternativa da un elenco
- Metodi (vedi documentazione API):
  - Costruttore
  - add(String)
  - addItemListener(ItemListener)
  - int getSelectedIndex()
  - String getSelectedItem()
- ItemEvent: alla scelta
- ItemListener
  - itemStateChanged(ItemEvent)

Stefano Mizzaro - Applet

21/25

## Restrizioni sugli applet

- Un applet non può:
  - accedere al file system del client
  - aprire connessioni di rete con un calcolatore che non sia il web server
  - stampare
  - creare processi figli
- Tutte le finestre aperte da un applet riportano l’indicazione che sono state create da un applet (esempio →)

Stefano Mizzaro - Applet

22/25

```

/*<applet code="OpenFrameApplet.class" width="10"
   height="10"></applet>*/
import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
public class OpenFrameApplet extends Applet {
    public void init() {
        final Frame f = new Frame();
        MenuBar mb = new MenuBar();
        Menu m = new Menu("File");
        MenuItem mi = new MenuItem("Esci");
        m.add(mi);
        mb.add(m);
        f.setMenuBar(mb);
        f.setVisible(true); f.pack();
        mi.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                f.dispose();
            }
        });
    }
}

```

```

>javac OpenFrameApplet.java
>appletviewer OpenFrameApplet.java

```

## Commenti

- MenuItem, Menu, MenuBar, add, setMenuBar (vedi documentazione API)
- Menu solo nei Frame
- Selezione voce da menu:
  - ActionEvent → ActionListener → actionPerformed
- final Frame f perché ascoltatore con classe anonima

Stefano Mizzaro - Applet

24/25

## Riassunto

- Gli applet
  - Esempi
  - Restrizioni
- Eventi del mouse (cenni)
- Choice
- Menu (cenni)