

# Java - Introduction à l'AWT

Code: java-awt

## Originaux

**url: <http://tecfa.unige.ch/guides/tie/html/java-awt/java-awt.html>**

**url: <http://tecfa.unige.ch/guides/tie/pdf/files/java-awt.pdf>**

## Auteurs et version

- Daniel K. Schneider
- Version: 0.2 (modifié le 11/4/01)

## Prérequis

**Module technique précédent: java-intro**

## Modules

**Module technique suivant: java-swing**

## Objectifs

- AWT (Abstract Windowing Toolkit) de base
- Note: Ce matériel est lacunaire (et en aucun cas utile pour une auto-formation)

# 1. Table des matières

|  |    |
|--|----|
| 1. Table des matières  | 3  |
| 2. Programmation d'applications avec un GUI                            | 4  |
| 2.1 Introduction à l'AWT   | 4  |
| 2.2 Applications simples avec un GUI                                   | 5  |
| 3. Applets   | 9  |
| 3.1 Utilisation d'applets avec Java 1.1 "built-in" dans NS<6 et IE<5.5 | 9  |
| 3.2 Utilisation d'applets avec le java plugin                          | 10 |
| 3.3 Vos premiers applets   | 11 |
| 4. L'exemple Weatherchart  | 17 |
| 4.1 Layout   | 18 |

## 2. Programmation d'applications avec un GUI

### 2.1 Introduction à l'AWT

- AWT = Abstract Windowing Toolkit

#### A. Classes importantes dans le package AWT

| Catégorie                  | Fonctionnalités  | Packages/Classes   |
|----------------------------|--|--|
| graphisme                  | Dessin de formes, lignes, images, sélection de fontes, couleurs, etc,              | java.awt.Graphics  |
| composantes                | buttons, champs de texte, menus, scroll bars                                       | java.awt.Component   |
| layout managers            | affichent des composantes qui ont été mise dans un "containter"                    | java.awt.BorderLayout<br>java.awt.GridBagLayout<br>java.awt.FlowLayout, etc. |
| gestionnaires d'événements | récupération des gestes de l'utilisateur, avec des handlers, listeners et adapters | Souspackage:<br>java.awt.event   |
| manipulation d'images      | gifs et compagnie  | Souspackage:<br>java.awt.image   |

**url: (Java 1.1) <http://tecfa2.unige.ch/guides/java/jdk1.1/docs/api/packages.html>**

**url: (Java 2/jdk 1.2) <http://tecfa2.unige.ch/guides/java/jdk1.2/docs/api/index.html>**

## 2.2 Applications simples avec un GUI

### Exemple 2-1: Simple fenêtre avec bouton pour la fermer

<http://tecfa.unige.ch/guides/java/staf2x/ex/gui/ButtonTest2.java>

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

public class ButtonTest2 extends Frame implements ActionListener {

    /* The Button test Program Java 1.1 by T Abbott and J Bishop Oct 1997
    * a simplified version SMM / TECFA Nov 98
    * Prints a warning message, and
    * Illustrates Buttons, Listeners and the
    * handling of events. */
    Button OkButton;
    Label TheLabel;

    public ButtonTest2 () {
        /* The constructor is responsible for setting
        * up the initial buttons and colour background. */
        setLayout(new FlowLayout(FlowLayout.CENTER));
        setBackground(Color.black);
        setForeground(Color.cyan);

                // create and add a Label
        TheLabel = new Label("W A R N I N G");
        add(TheLabel);
    }
}
```

```
        // create & add a button, plus an ActionListener to the button
        OkButton = new Button("Ok");
        OkButton.addActionListener(this);
        add(OkButton);
    }

    // implement actionPerformed
    public void actionPerformed(ActionEvent e) {
        setVisible(false);
        dispose();
        System.exit (0);
    }

    public static void main(String[] args) {
        Frame f = new ButtonTest2();
        f.setTitle("Button Test 2");
        f.setSize(160,60);
        f.setVisible(true);
        // allows to close the application
        f.addWindowListener (new WindowAdapter () {
            public void windowClosing(WindowEvent e) {
                System.exit(0);
            }
        });
    }
}
```

Voir aussi une variante:

<http://tecfa.unige.ch/guides/java/staf2x/ex/gui/ButtonTest.java>

## A. Explications

Frame est la fenêtre de base:

```
public class ButtonTest2 extends Frame implements ActionListener {....}
```

- étend la classe `Frame`
- `Frame` hérite des classes `awt Window`, `Container`, `Component`

Gestion des événements: les `ActionListeners`

```
public class ButtonTest2 extends Frame implements ActionListener {....}
```

- implémente l'interface `java.awt.event.ActionListener`
- Cette "classe" exige que l'on implémente la méthode `actionPerformed` à qui les événements sont passés

Gestion du Layout et composantes interactives:

```
setLayout(new FlowLayout(FlowLayout.CENTER));
```

- définit un `Layout manager` pour le frame: ici un `FlowLayout` centré

```
OkButton = new Button("Ok");
```

```
OkButton.addActionListener(this);
```

```
add(OkButton);
```

- on crée un objet bouton
- on rajoute ce bouton dans le "`ActionListener`" pour le `Frame`
- finalement on rajoute le bouton dans le conteneur (le frame ici).

Implémentation de `actionPerformed`:

```
public void actionPerformed(ActionEvent e) {
```

```
setVisible(false);  
dispose();  
System.exit (0);  
}
```

- Pour tout Frame avec un Action Listener, on doit implémenter la méthode `actionPerformed`
- normalement il aurait fallu tester pour savoir quel geste l'utilisateur a fait. Voir Bishop (1998) 303ff.
- Ici il n'y a qu'un bouton et il suffit d'agir (on ferme l'application).



## 3. Applets

### 3.1 Utilisation d'applets avec Java 1.1 "built-in" dans NS<6 et IE<5.5

| Syntaxe: applet Tag dans HTML  |   | Exemples     |
|--|---|--------------|
| <pre>&lt;APPLET CODE="AppletSubclass.class" WIDTH="anInt" HEIGHT="anInt"&gt;   &lt;PARAM NAME=parameter1Name VALUE="aValue"&gt;   &lt;PARAM NAME=parameter2Name VALUE="anotherValue"&gt; &lt;/APPLET&gt;</pre>   |   |              |
| WIDTH  | largeur de l'applet dans la page HTML             | 200          |
| HEIGHT   | hauteur de l'applet dans la page HTML             | 174          |
| CODE   | nome de la classe Java qui implémente l'applet    | Simple.class |
| <p>Voir: <a href="http://tecfa.unige.ch/guides/java/tutorial/applet/appletonly/appletTag.html">http://tecfa.unige.ch/guides/java/tutorial/applet/appletonly/appletTag.html</a><br/> ou un bon manuel HTML pour une liste complète des paramètres "système"<br/> ou <a href="http://java.sun.com/products/jdk/1.1/docs/guide/misc/applet.html">http://java.sun.com/products/jdk/1.1/docs/guide/misc/applet.html</a></p> |   |              |
| PARAM NAME   | paramètres définis par le programmeur de l'applet | "Hello"      |

Illustration:

```
<APPLET CODE=Simple.class WIDTH=100 HEIGHT=200> </APPLET>
```

## 3.2 Utilisation d'applets avec le java plugin

- La balise applet active l'engin Java interne au navigateur ou encore le plugin Java par défaut (NS 6)
- Pour activer un java plugin 2 (JRE 1.2, 1.3, etc.) dans un ancien Netscape ou tout IE il faut utiliser d'autres balises

Documentation chez SUN

**url: <http://java.sun.com/products/plugin/1.2/docs/tags.html>**

**url: <http://java.sun.com/products/plugin/1.3/docs/tags.html>**

## 3.3 Vos premiers applets

### Exemple 3-1: Simple Virus Warning

Source: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/virus/Virus.java>

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

public class Virus extends Applet {
    /* A Warning box in an applet by J M Bishop  Oct 1996
     *                                     Java 1.1      */
    static private final int line = 15;
    static private final int letter = 5;
    public void paint(Graphics g) {
        g.drawRect(2*letter, 2*line, 33*letter, 6*line);
        g.drawString("W A R N I N G", 9*letter, 4*line);
        g.drawString("Possible virus detected", 4*letter, 5*line);
        g.drawString("Reboot and run virus", 5*letter, 6*line);
        g.drawString("remover software", 7*letter, 7*line);
    }
}
```

Page HTML: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/virus/virus.html>

```
<H1>Virus Applet</H1>
<APPLET CODE="Virus.class" WIDTH=200 HEIGHT=200>
</APPLET>
<hr> Voici notre petit applet <hr>
```

## A. La structure de base d'un applet

### Exemple 3-2: Applet vide qui illustre qq "méthodes de gestion"

HTML: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/simple/simpleempty.html>

Source: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/simple/SimpleEmpty.java>

```
import java.applet.Applet; import java.awt.Graphics;
public class SimpleEmpty extends Applet {
// execution lorsque l'applet est chargé (initialisation des trucs "statiques")
    public void init() {
        }
// execution lorsque l'applet démarre (aspets plus dynamiques)
// au début ou lorsque l'utilisateur revient sur la page
    public void start() {
        }
// execution lorsque l'applet s'arrete par exemple l'utilisateur fait un "back"
    public void stop() {
        }
// execution lorsque l'applet est détruit
    public void destroy() {
        }
    void uneMethodeVide() {
        }
    public void paint(Graphics g) {
//on fait quand-meme un minimum
        g.drawString("Hello World", 5, 15);
    }
}
```

### Exemple 3-3: Une version plus compliquée:

HTML: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/simple/Simple.java>

Source: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/simple/simple.html>

```
StringBuffer buffer;
public void init() {
    buffer = new StringBuffer();
    addItem("initializing... ");
}
public void start() {
    addItem("starting... ");
}
public void stop() {
    addItem("stopping... ");
}
public void destroy() {
    addItem("preparing for unloading...");
}
void addItem(String newWord) {
    System.out.println(newWord);
    buffer.append(newWord);
    repaint();
}
public void paint(Graphics g) {
    //Draw a Rectangle around the applet's display area.
    g.drawRect(0, 0, size().width - 1, size().height - 1);
    //Draw the current string inside the rectangle.
    g.drawString(buffer.toString(), 5, 15);
}
```

## B. La gestion des événements

Exemple 3-4: Virus Warning Interactif (exemple nul, faut écrire une méthode init)

HTML: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/virus/virus2.html>

Source: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/virus/Virus2.java>

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

public class Virus2 extends Applet implements ActionListener {
    /* A Warning box in an applet adapted from J M Bishop Oct 1996 - Java 1.1
    * Must be run via its corresponding html file in a browser or the appletviewer */

    static private final int line = 15; static private final int letter = 5;
    // flag pour voir si c'est la première fois
    static private boolean FirstTime = true;
    static private int hits = 0;

    public void paint(Graphics g) {
        Button KillButton;
        KillButton = new Button("Kill me");
        KillButton.setSize(100,30);
        KillButton.setLocation(50,170);
        KillButton.addActionListener(this);
        add(KillButton);
    }
}
```

```
if (FirstTime)
{
KillButton.setVisible(true);
g.drawRect(2*letter, 2*line, 33*letter, 6*line);
g.drawString("W A R N I N G", 9*letter, 4*line);
g.drawString("Possible virus detected", 4*letter, 5*line);
g.drawString("Reboot and run virus", 5*letter, 6*line);
g.drawString("remover software", 7*letter, 7*line);
}
else
{
g.drawRect(2*letter, 2*line, 33*letter, 6*line);
String bye = "Good BYE [ " + hits + " ]";
g.drawString(bye, 9*letter, 4*line);
g.drawString("OH CRUEL WORLD !", 5*letter, 6*line);
}

}

public void actionPerformed(ActionEvent e) {

FirstTime = false;
hits++;
repaint();

}
}
```

- Cet exemple fonctionne selon la même logique que l'exemple 2-1 "Simple fenêtre avec bouton pour la fermer" [4]
- Il n'est pas nécessaire de gérer la fermeture d'un applet (Le browser le fait pour vous par défaut)
- Il faut implémenter différemment l'affichage des composantes GUI. Ici on spécifie tout dans la méthode paint (ce qui est faux, car on dessine plusieurs fois un même bouton, l'un sur l'autre)
- La méthode main n'existe plus (le browser en crée une automatiquement)

### Exercice 1: Jouez avec les paramètres de l'applet "BarChart"

[url: http://tecfa2.unige.ch/guides/java/jdk1.1/demo/BarChart/example1.html](http://tecfa2.unige.ch/guides/java/jdk1.1/demo/BarChart/example1.html)

### Exercice 2: Visitez JARS ou d'autres archives

[url: http://www.jars.com/](http://www.jars.com/)



## 4. L'exemple Weatherchart

url: <http://tecfa.unige.ch/guides/java/staf2x/ex/gui/weather/>

Un petit programme qui permet d'ouvrir un fichier texte pour lire et afficher graphiquement les données pluviométriques d'une ville sur plusieurs années.

- L'interface contient plusieurs boutons qui rendent la navigation entre les années plus facile.
- Le programme comporte 4 fenêtres pour la version application (seulement 3 pour la version applet) :
  - la fenêtre principale
  - la fenêtre d'informations sur la programme
  - la fenêtre pour naviguer plus rapidement entre les années
  - la fenêtre pour ouvrir un fichier (seulement dans la version application)

## 4.1 Layout

```
TheAboutDialog = new AboutDialog();
TheAboutDialog.setVisible(false);
/* The windows can be drawn with the data from the file. */
this.setSize(400,450);
// we use a border Layout (default would be flow)
this.setLayout(new BorderLayout());

// Top Panel
// first row (open, Savanna Label, about)
TheTopPanel = new Panel();
TheOpenCloseButton = new Button("Open...");
TheOpenCloseButton.addActionListener(this);
TheTopPanel.add(TheOpenCloseButton);
TheTopPanel.add(new Label("Savanna Rainfall Chart:"));
TheAboutButton = new Button("About...");
TheAboutButton.addActionListener(this);
TheTopPanel.add(TheAboutButton);
this.add(TheTopPanel, "North");

// Canvas Line
TheChartCanvas = new ChartCanvas();
this.add(TheChartCanvas, "Center");

// Display commands
TheFirstButton = new Button("First");
TheFirstButton.addActionListener(this);
ThePreviousButton = new Button("Previous");
ThePreviousButton.addActionListener(this);
```

```
TheGoToButton = new Button("Go To...");
TheGoToButton.addActionListener(this);
TheNextButton = new Button("Next");
TheNextButton.addActionListener(this);
TheLastButton = new Button("Last");
TheLastButton.addActionListener(this);
TheBottomPanel = new Panel();
TheBottomPanel.add(TheFirstButton);
TheBottomPanel.add(ThePreviousButton);
TheBottomPanel.add(TheGoToButton);
TheBottomPanel.add(TheNextButton);
TheBottomPanel.add(TheLastButton);
this.add(TheBottomPanel, "South");
```

... à suivre ...

Voir le AWT Tutorial de Sun

**url: <http://tecfa.unige.ch/guides/java/awt-tutorial/OLDui/> (à TECFA)**

