
TD 7 IJA

Les interfaces graphiques

JAVA

AWT & SWING

- Première bibliothèque graphique JAVA: AWT
 - Package java.awt
 - Utilisation de code natif
 - Composants limités
- Nouvelle bibliothèque: SWING
 - Package javax.swing
 - Plus riche et plus personnalisable
 - Ne remplace pas AWT mais fournit des composants plus performants

Hiérarchie des composants

Les composants graphiques sont placés dans des conteneurs (Containers):

- Composants élémentaires

- JLabel
- JButton
- JCheckBox
- JRadioButton
- JTextField
- JTextArea

- Composants complexes

- ButtonGroup
- JComboBox
- JList
- JScrollBar
- JMenuBar
- JPopupMenu

- Les containers

- JWindow
 - JFrame
 - JDialog
 - JFileDialog
- JPanel
 - Applet
- JTabbedPane
- JScrollPane

<http://java.sun.com/docs/books/tutorial/uiswing/components/components.html>

Disposition des composants (1/2)

Chaque conteneur utilise un gestionnaire de placement (Layout) pour la disposition des composants qu'il contient.



<http://java.sun.com/docs/books/tutorial/uiswing/layout/visual.html>

Disposition des composants (2/2)

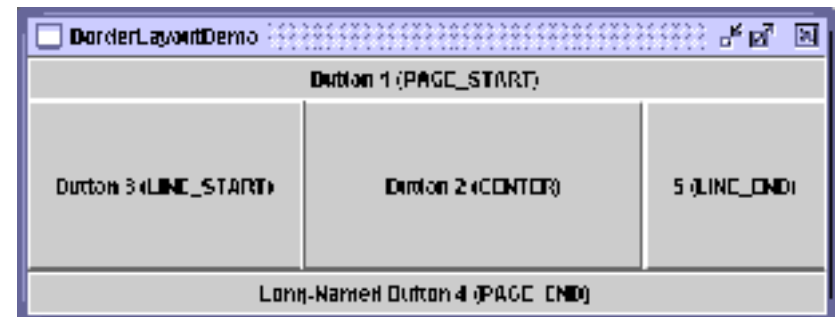
Exemples de dispositions

GridLayout



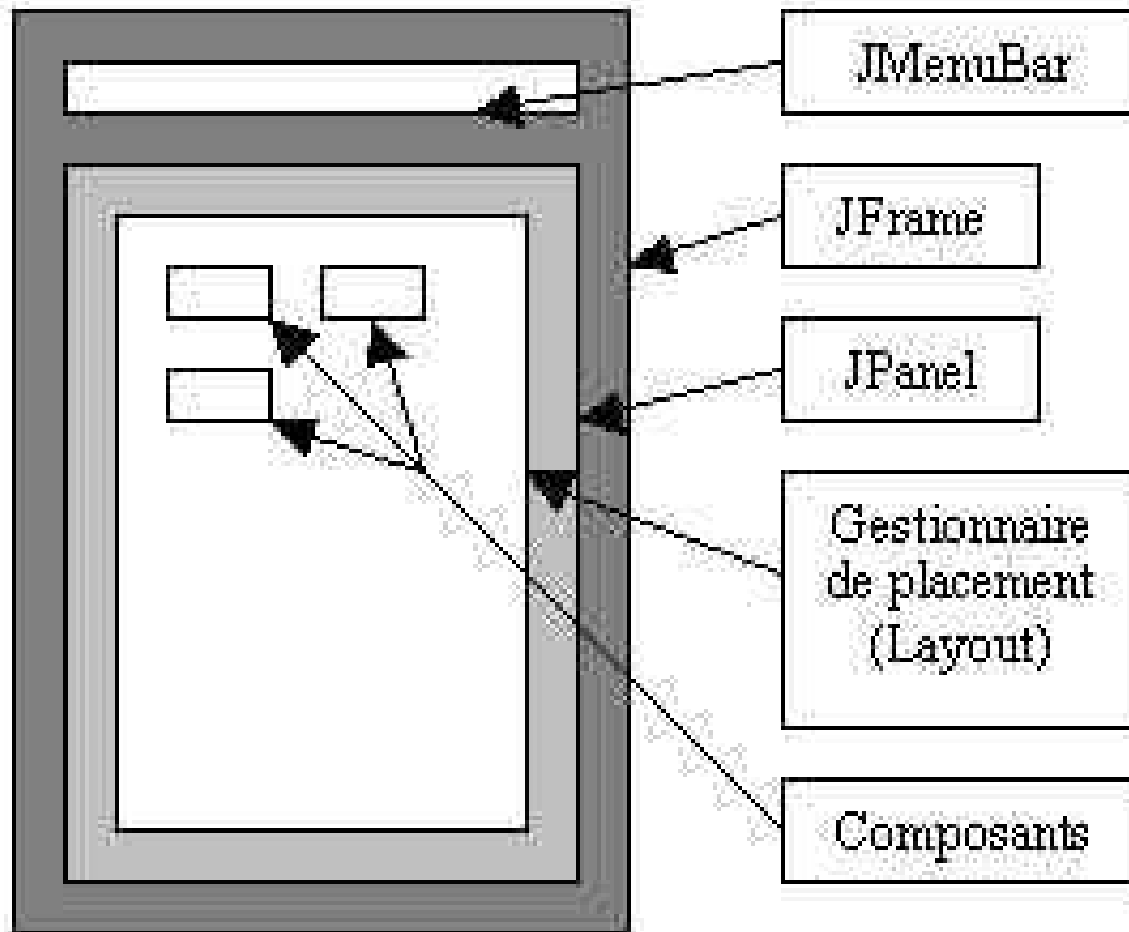
```
JFrame fenetre=new JFrame("GridLayoutDemo");
JPanel tmp = fenetre.getContentPane();
tmp.setLayout(new GridLayout(3,2));
tmp.add(new Button("Button 1"), 0,0);
tmp.add(new Button("Button 2"), 0,1);
tmp.add(new Button("Button 3"), 1,0);
tmp.add(new Button("Long-Named Button 4 "), 1,1);
tmp.add(new Button("5"), 2,0);
```

BorderLayout



```
JFrame fenetre=new JFrame("BorderLayoutDemo");
JPanel tmp = fenetre.getContentPane();
tmp.setLayout(new BorderLayout());
tmp.add(new Button("Button 1 (PAGE_START)",
BorderLayout.NORTH));
tmp.add(new Button("Button 3 (LINE_START)",
BorderLayout.WEST));
tmp.add(new Button("Button 2 (CENTER)")
BorderLayout.CENTER);
tmp.add(new Button("5 (LINE_END)")
BorderLayout.EAST);
tmp.add(new Button("Long-Named Button 4
(PAGE_END)") BorderLayout.SOUTH);
```

Composition d'une fenêtre JAVA



Exemple d'une fenêtre JAVA

// Création de la fenêtre

```
JFrame frame = new JFrame("ExempleSimple");
```

// Création du container

```
JPanel panel = new JPanel();
```

// Définition du gestionnaire de placement

```
panel.setBorder(new GridLayout(1,2))
```

// Création des composants

```
JLabel label = new JLabel("Entrer votre nom");
```

```
JTextField textField = new JTextField("toto");
```

// Ajout des composants au container

```
panel.add(label);
```

```
Panel.add(textField);
```

// Ajout du container à la fenêtre

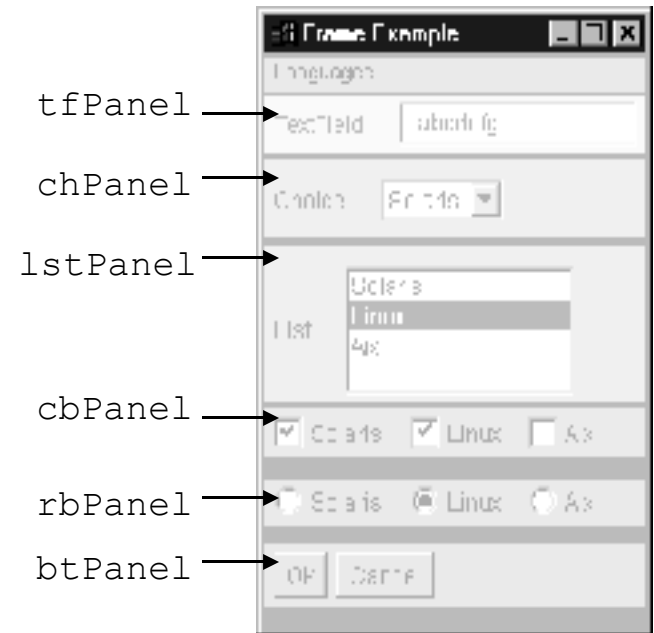
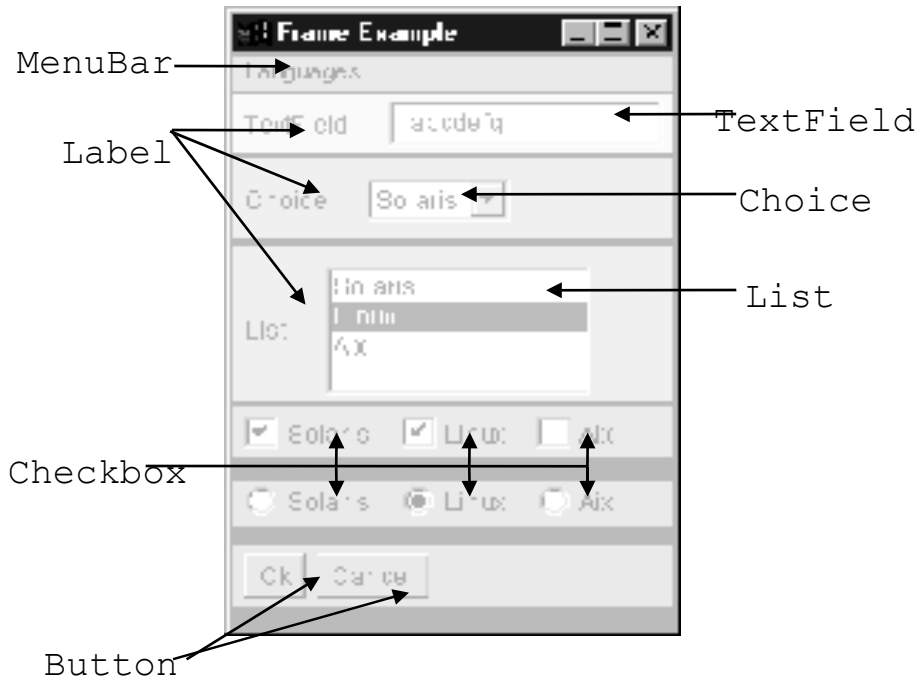
```
frame.getContentPane().add(panel); (*)
```

// Afficher la fenêtre

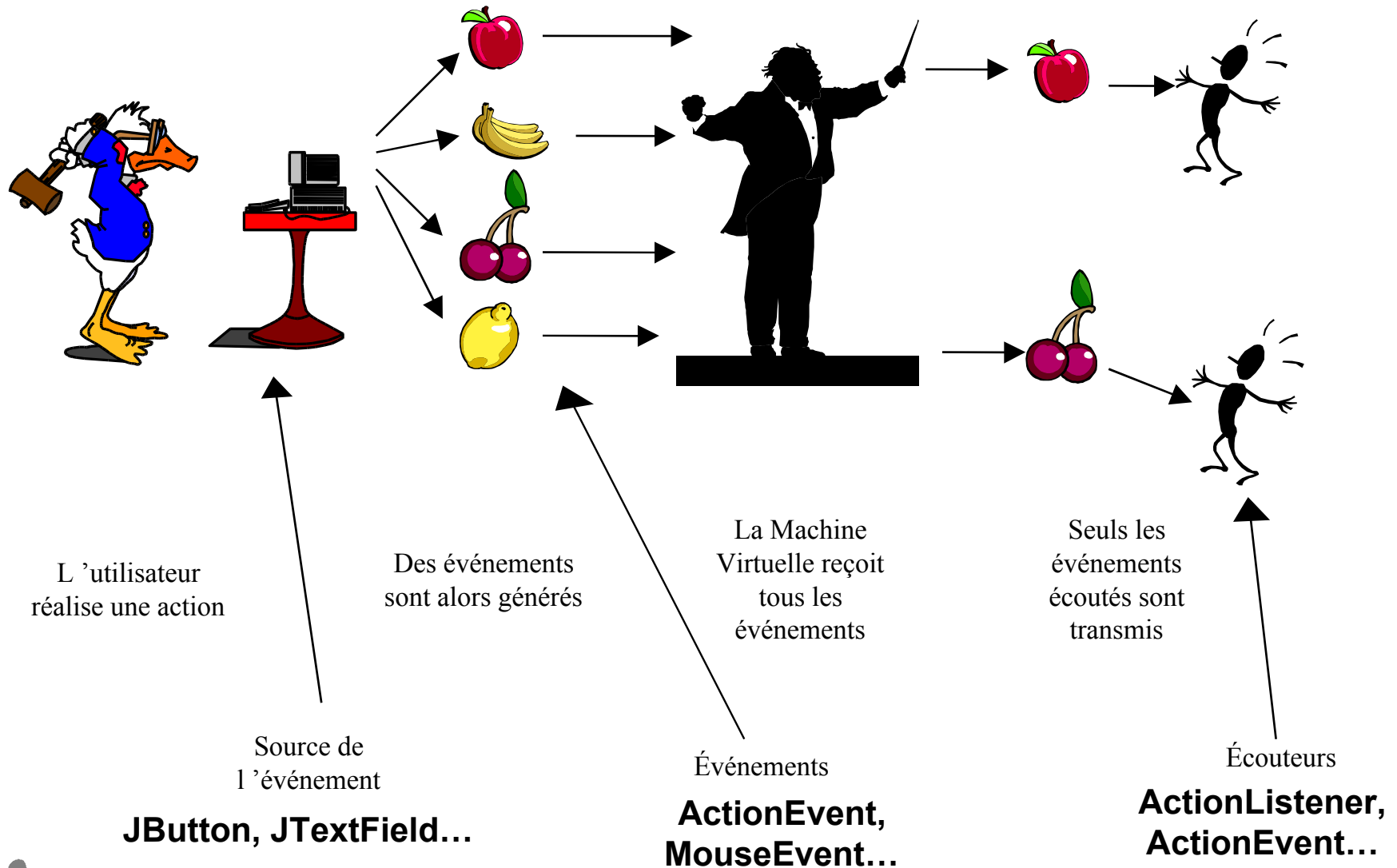
```
frame.pack();
```

```
frame.setVisible(true);
```

Exercice (à programmer)



Propagation des évènements



Un exemple

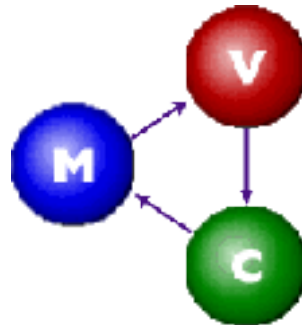
```
class BoutonAvecListener implements ActionListener{
    static JButton boutonJaune = new JButton("Jaune");

    public void actionPerformed(ActionEvent evt){
        if (evt.getSource()==boutonJaune)
            System.out.println("bouton Jaune Appuyé");
    }

    public static void main(String[] args)
    {
        JFrame fenetre = new JFrame();
        boutonJaune.addActionListener(new BoutonAvecListener());
        fenetre.getContentPane().add(boutonJaune);
        fenetre.pack();
        fenetre.show();
    }
}
```

Le modèle MVC

- Une interface graphique =
 - Un **m**odèle qui gère les données de l'application
 - Une **v**ue qui gère les composants graphiques
 - Un **c**ontrôleur qui intercepte les événements de la vue et effectue les actions nécessaires au niveau du modèle



Conception d'une IHM en Java

- Modèle : classe de traitement
 - Elle définit toutes les méthodes applicatives
- Vue : classe graphique de présentation
 - Elle définit la facette externe de l'application pour l'utilisateur
- Contrôleur : classe listener
 - Elle redirige les actions de l'utilisateur sur les classes de traitement (modèle)
- Remarque : le listener et la vue peuvent être contenus dans la même classe

Exercice

I) L'exemple inévitable (HelloWorld)

- 1) développer une fenêtre HelloWorld qui affiche « Hello !! » dans un label
- 2) Ajouter un bouton à la fenêtre. Le label affichera « Hello (n) » où n est le nombre de clicks sur le bouton

II) Interface graphique pour la gestion des étudiants

En utilisant le modèle MVC, développer une interface graphique pour ajouter, supprimer et afficher un étudiant à la classe GesEtudHash du TD précédent.