

# Programmation Graphique en Swing

## PO2 (2016-2017)

### Chapitre 2 Composants Graphiques

# Plan

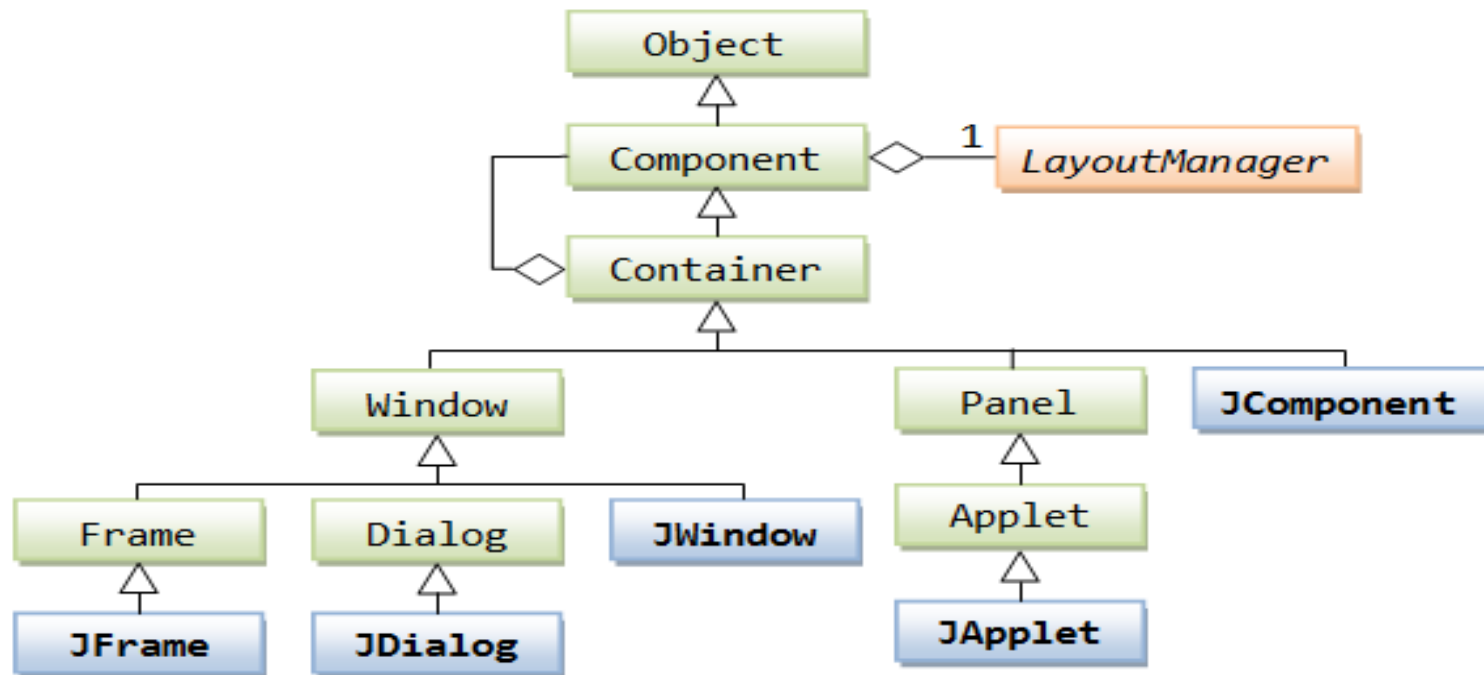
- Swing en Java
- GUI
- Conteneurs et composants
  - Exemple : JetDeComposantContentPane
  - Exemple : JetDeComposantJPanel
- Bordure
  - Exemple : JetDeComposantBorder

# Swing

- AWT (Abstract Window Toolkit)
  - La première API graphique de Java
- Swing
  - Une extension complémentaire à AWT avec une meilleure portabilité
  - Re-implémente les composants de AWT
  - Ajout de nouveaux composants dérivés des composants AWT
  - Applique le MVC (modèle, vue, contrôle) pour l'implémentation de ses composants
- Deux types d'objets graphiques de Swing :
  - Conteneurs : JFrame, JPanel, ...
  - Composants élémentaires: JButton , JLabel, JList, JTextField, ...

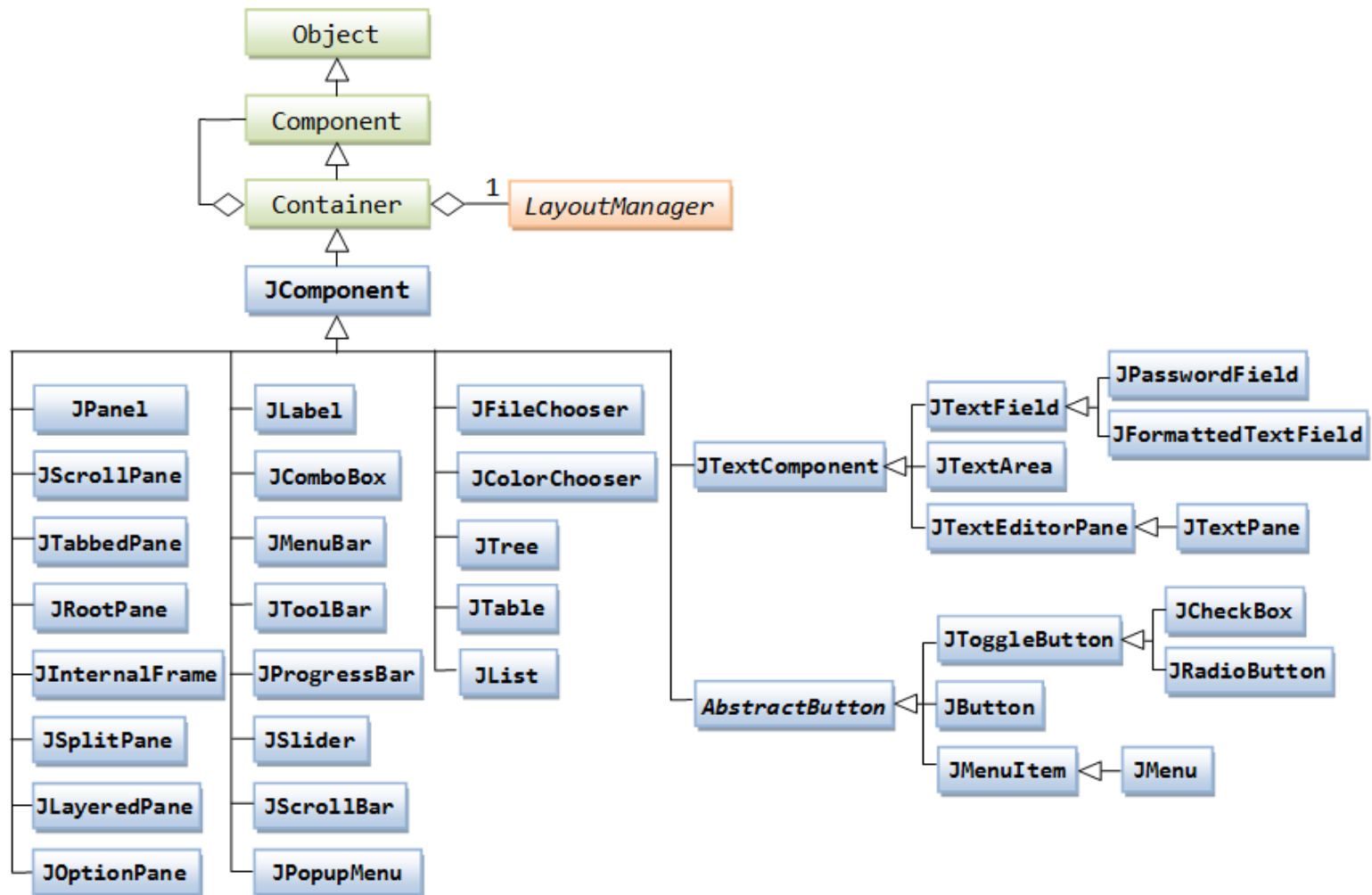
# Swing

- Conteneurs :
  - Top-level : JFrame, JDialog, JApplet
  - Seconde niveau : JPanel



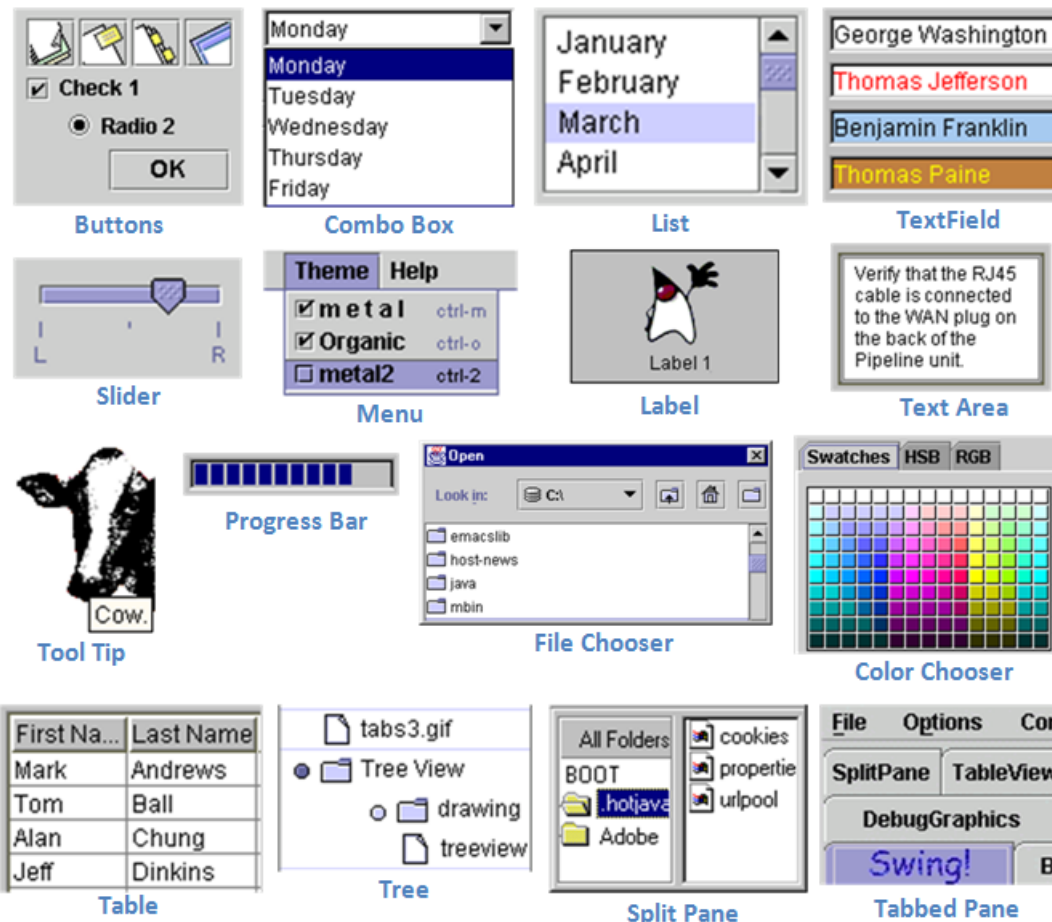
# Swing

- Composants élémentaires



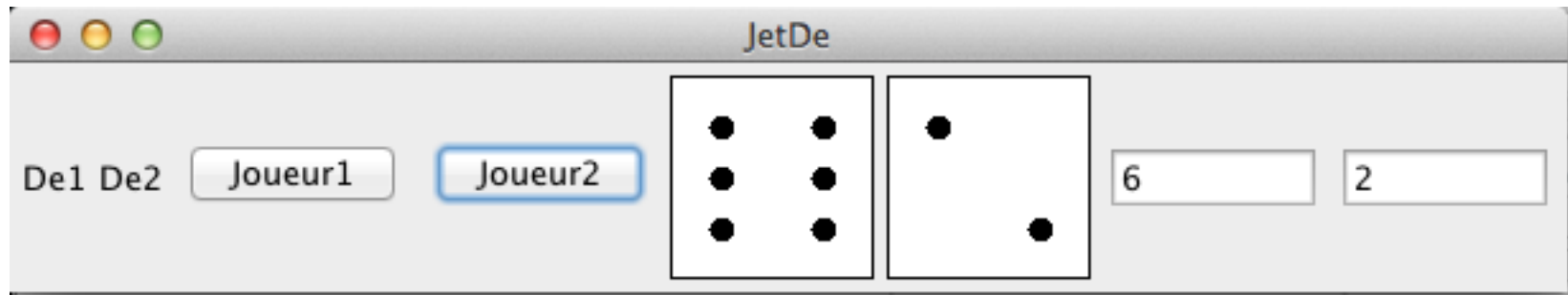
# Swing

- Composants élémentaires



# GUI

- GUI = la création des conteneurs/composants + la mise en place des composants + le traitement des événements



# Conteneur/Composant

- Une application Swing contient un conteneur de premier niveau (JFrame, JDialog, JApplet), et un conteneur de deuxième niveau comme content-pane de type `java.awt.Container`.
- Des composants sont ajoutés soit directement dans content-pane, soit dans un JPanel ou un Box qui sont convertis en content-pane ou ajoutés dans content-pane.



# Conteneur/Composant

- Un JPanel est un conteneur intermédiaire :
  - zone rectangulaire, par défaut, sans décoration graphique.
  - accueille graphiquement plusieurs composants.

# Conteneur/Composant

- Un JPanel peut être converti en content-pane :

```
this.setContentPane(panel);  
this.pack();  
this.setVisible(true);
```

- Un JPanel peut être aussi ajouté dans content-pane :

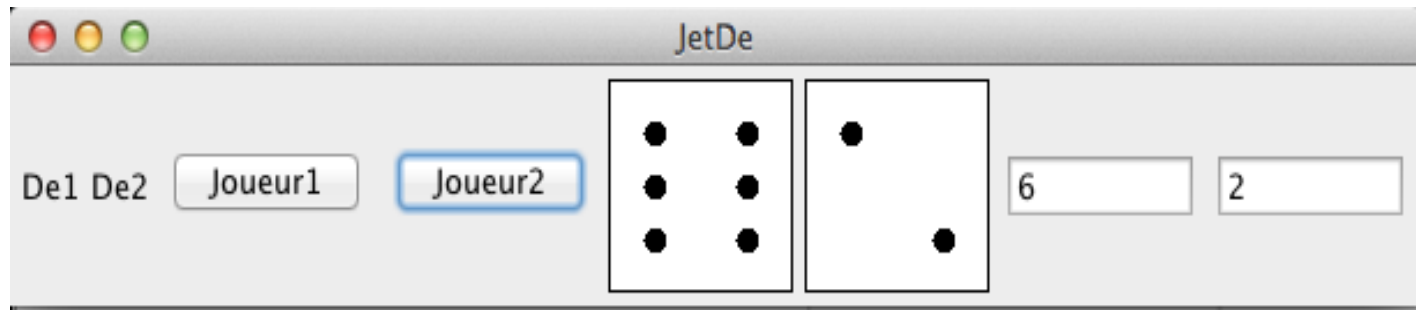
```
this.getContentPane().add(panel);  
this.pack();  
this.setVisible(true);
```

# Fonction main()

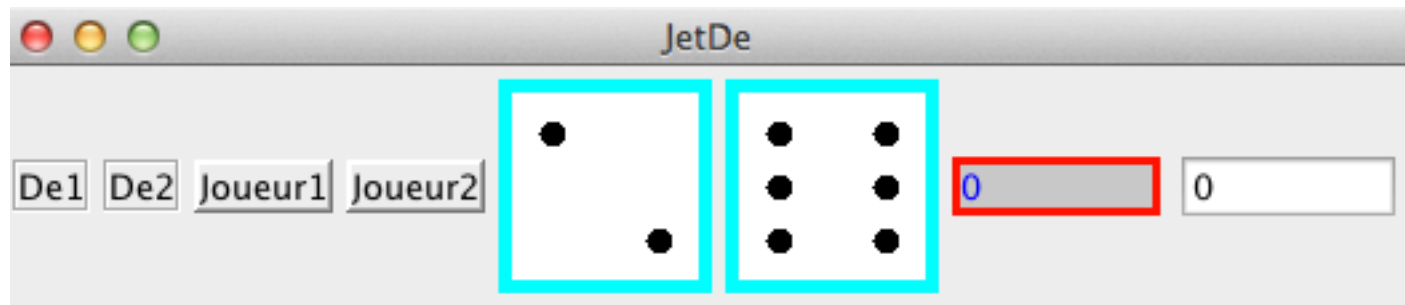
- Cette « recette » de création de GUI est nécessaire, car la méthode main et les réactions des composants graphiques s'effectuent dans des threads distincts.

# Bordure

- Sans décoration de bordure :



- Avec décoration de bordure :



# Bordure

- Utiliser une méthode static de la classe `BorderFactory` pour décoder la bordure d'un composant.
- Exemple : `JetDeComposantBorder`

```
label1.setBorder(BorderFactory.createEtchedBorder());  
label2.setBorder(BorderFactory.createEtchedBorder());
```

```
jetButton1.setBorder(BorderFactory.createRaisedBevelBorder());  
jetButton2.setBorder(BorderFactory.createRaisedBevelBorder());
```

```
deGauche.setBorder(BorderFactory.createLineBorder(Color.CYAN, 5));  
deDroit.setBorder(BorderFactory.createLineBorder(Color.CYAN, 5));
```

```
textDeGauche.setBackground(new Color(200, 200, 200));  
textDeGauche.setForeground(Color.BLUE);  
textDeGauche.setBorder(BorderFactory.createLineBorder(Color.RED, 3));
```

# Buttons

- Exemple : JetDeComposantGroupe



# Exemples

- Conteneurs et composants
  - Exemple : ComposantContentPane
  - Exemple : ComposantJPanel
- Bordure
  - Exemple : ComposantBorder
- Boutton
  - Exemple : ComposantGroupe