

CENTRALE  
L Y O N

# Interaction Personne-Machine

## JAVA - Swing

BTD/IPM

1

CENTRALE  
L Y O N

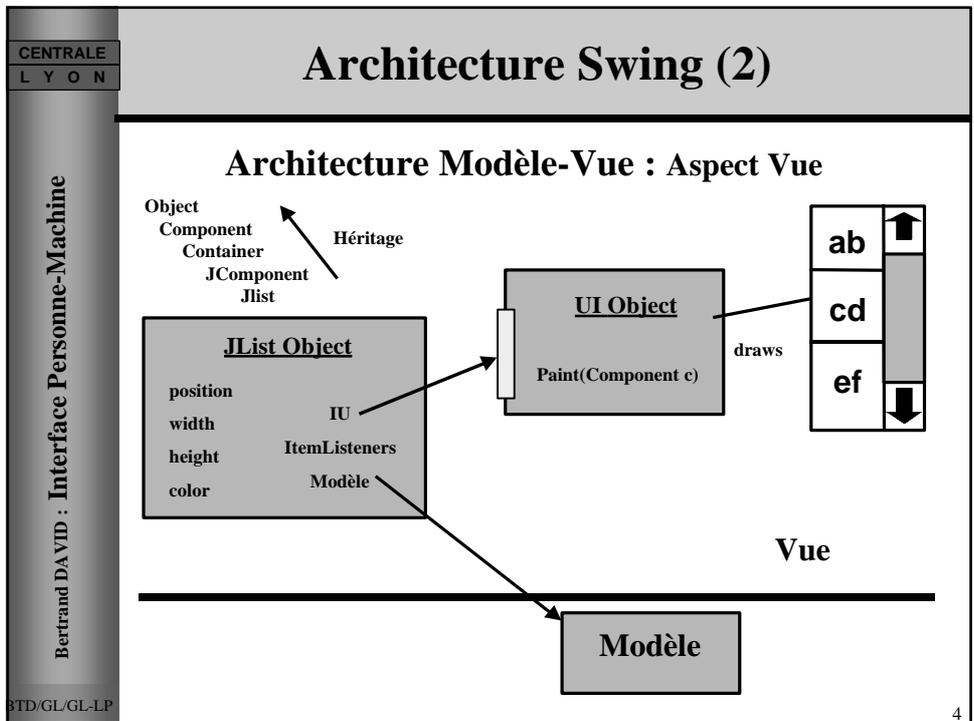
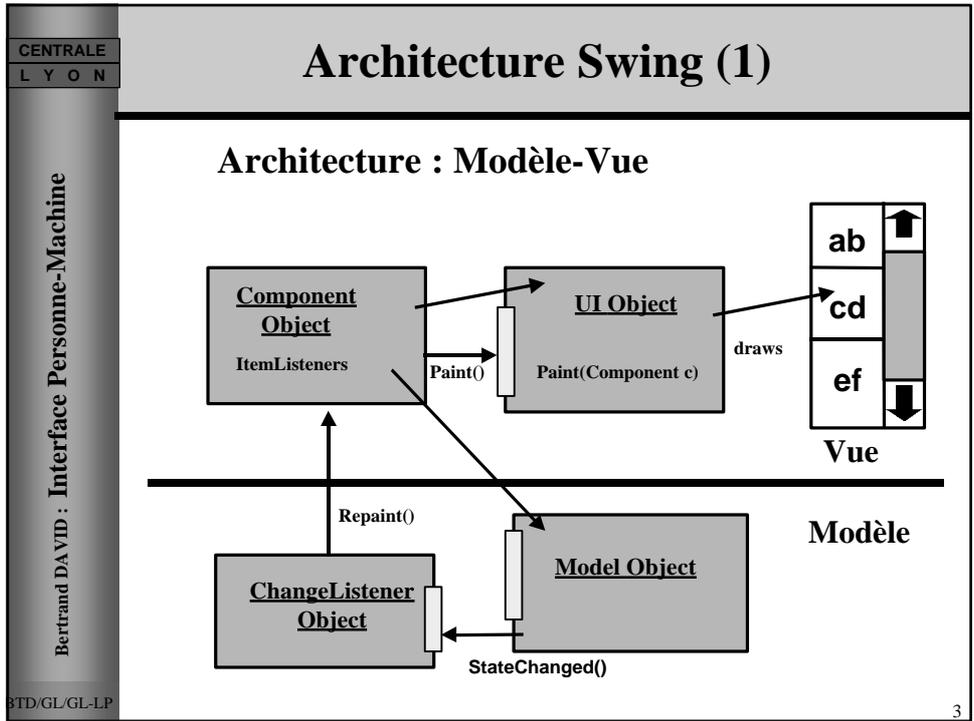
# SWING - hiérarchie de classe

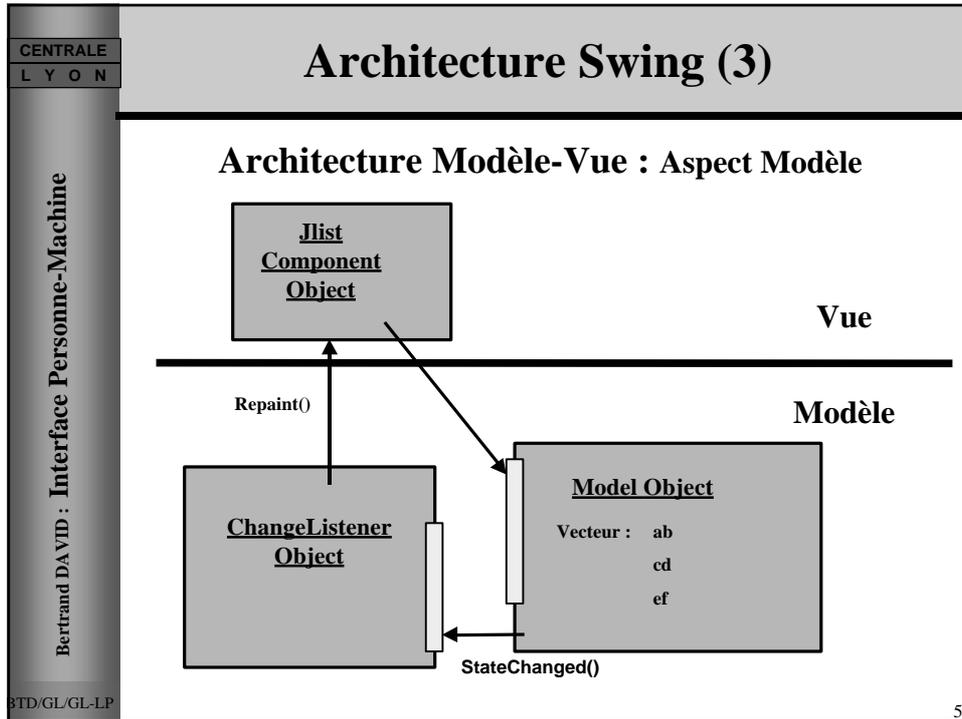
Bertrand DAVID : Interface Personne-Machine

|                |                 |
|----------------|-----------------|
| Object         | Object          |
| Component      | Component       |
| Container      | Container       |
| JComponent     | JComponent      |
| JComboBox      | JAbstractButton |
| JInternalFrame | JButton         |
| JLabel         | JToggleButton   |
| JLayeredPane   | JCheckBox       |
| JList          | JRadioButton    |
| JProgressBar   | JTextComponent  |
| JRootPane      | JEditorPane     |
| JScrollPane    | JTextPane       |
| JSlider        | JTextArea       |
| JSplitPane     | JTextField      |
| JTable         | JPasswordField  |
| JToolBar       |                 |
| JToolTip       | Manquent :      |
| JTree          | Menus, Frames   |
| JViewport      | et Applets      |

BTD/GL/GL-LP

2





CENTRALE  
L Y O N

### Java Beans

- Dans VisualAge pour Java, les beans sont les composants à utiliser lors de la programmation visuelle.
- Ces beans correspondent à des classes Java conformes aux spécifications JavaBeans.
- L'éditeur de composition visuelle permet de sélectionner les beans à partir d'une palette, spécifier leurs caractéristiques et connecter les beans entre eux.
- Les beans peuvent contenir d'autres beans ainsi que des connexions à des beans.

3TD/GL/GL-LP

6

CENTRALE  
L Y O N

## Beans (1)

Bertrand DAVID : Interface Personne-Machine

- Deux types de beans sont utilisés dans l'éditeur de composition visuelle :
  - ➔ Le **bean visuel** qui s'affiche dans le programme en cours d'exécution. Les beans visuels, tels que les fenêtres, les boutons et les zones de texte, constituent l'interface graphique utilisateur d'un programme.
  - ➔ Le **bean non visuel** qui n'apparaît pas dans le programme au moment de son exécution. Il représente un objet qui encapsule des données et implémente un comportement dans un programme.

8TD/GL/GL-LP 7

CENTRALE  
L Y O N

## Beans (2)

Bertrand DAVID : Interface Personne-Machine

- L'interface publique d'un bean détermine les interactions avec les autres composants du même type. Elle se compose des éléments suivants :
  - ➔ Les **Propriétés** qui sont des données accessibles par d'autres beans. Ces données peuvent correspondre aux propriétés logiques d'un bean, comme le solde d'un compte, la taille d'un envoi, ou le libellé d'un bouton.
  - ➔ Les **Evénements** qui correspondent à des signaux qui indiquent que quelque chose s'est passé. L'ouverture d'une fenêtre ou la modification de la valeur d'une propriété, par exemple, déclenchent un événement.
  - ➔ Les **Méthodes** qui correspondent aux opérations qu'un bean peut exécuter. Elles peuvent être déclenchées par des connexions à d'autres beans.

8TD/GL/GL-LP 8

|   |  |
|---|--|
| CENTRALE<br>L Y O N                         | <h2>Connexions</h2>  |
| Bertrand DAVID : Interface Personne-Machine | <ul style="list-style-type: none"><li>■ Editeur de composition visuelle<br/><p>L'éditeur de composition visuelle est l'outil de programmation visuelle intégré de VisualAge for Java.</p></li><li>■ Dans l'éditeur de composition visuelle, les connexions définissent les interactions entre les beans. On peut connecter des beans entre eux et des connexions entre elles.</li><li>■ Une connexion est constituée d'une source et d'une cible. L'extrémité de départ de la connexion constitue la source de la connexion et l'extrémité d'arrivée la cible.</li></ul> |
| BTD/GL/GL-LP                                | 9  |

|   |   |
|---|---|
| CENTRALE<br>L Y O N                         | <h2>Qu'est-ce qu'un JavaBean ?</h2>   |
| Bertrand DAVID : Interface Personne-Machine | <ul style="list-style-type: none"><li>■ Les JavaBeans sont des objets Java qui se comportent conformément à la spécification JavaBeans.</li><li>■ Les JavaBeans (ou beans) sont des composants logiciels réutilisables qui peuvent être manipulés dans un environnement de développement comme VisualAge pour Java.</li><li>■ Le modèle des signatures de méthode et de définition de classe d'un bean permet à des environnements tels que VisualAge pour Java de déterminer leurs propriétés et comportement.</li><li>■ La faculté d'un environnement de beans à déterminer les caractéristiques d'un bean est appelée introspection.</li></ul> |
| BTD/GL/GL-LP                                | 10  |

CENTRALE  
L Y O N

Bertrand DAVID : Interface Personne-Machine

## Caractéristiques des beans (1)

- Les beans possèdent trois sortes de caractéristiques :
  - ➔ Evénements
  - ➔ Méthodes
  - ➔ Propriétés
- Un bean expose une caractéristique lorsqu'il rend cette caractéristique disponible pour les autres beans.

BTD/GL/GL-LP

11

CENTRALE  
L Y O N

Bertrand DAVID : Interface Personne-Machine

## Caractéristiques des beans (2)

- Brève description de ces trois types de caractéristiques :
  1. Evénements sont les événements que le bean génère. D'autres beans peuvent signaler que ces événements les intéressent et être prévenus lorsqu'ils se produisent.
  2. Méthodes sont les actions qu'un bean expose pour que les autres beans les appellent. Les méthodes bean sont un sous-ensemble des méthodes publiques de la classe Java qui constitue le bean.
  3. Les propriétés sont les attributs exposés par un bean. Les propriétés peuvent être lues, écrites ou les deux. Elles peuvent présenter les caractéristiques décrites ci-après.

BTD/GL/GL-LP

12

|   |  |
|---|--|
| CENTRALE<br>L Y O N                         | <h2>Caractéristiques des beans (3) :</h2>  |
| Bertrand DAVID : Interface Personne-Machine | <ul style="list-style-type: none"><li>➔ Une propriété liée déclenche l'événement <code>propertyChange</code> lorsque sa valeur est modifiée.</li><li>➔ Une propriété contrainte permet à d'autres beans de déterminer si la valeur de la propriété peut être modifiée (elle déclenche l'événement <code>vetoableChange</code>).</li><li>-</li><li>➔ Une propriété indexée est un tableau. Elle expose donc des méthodes supplémentaires à des éléments d'adresse individuels.</li><li>➔ Une propriété cachée n'est pas visible. Elle est utilisable par des outils adaptés aux beans uniquement.</li><li>➔ Une propriété expert doit être manipulée uniquement par des utilisateurs experts.</li><li>-</li><li>➔ Une propriété normale est une propriété ni cachée, ni expert.</li></ul> |
| BTDA/DAVID : Interface Personne-Machine     | 13   |

|   |   |
|---|---|
| CENTRALE<br>L Y O N                         | <h2>Classes BeanInfo</h2>   |
| Bertrand DAVID : Interface Personne-Machine | <ul style="list-style-type: none"><li>➔ Les beans peuvent être associés à des classes <code>BeanInfo</code>. Ces classes décrivent explicitement les événements, méthodes et propriétés exposés par un bean.</li><li>➔ <code>VisualAge</code> pour Java peut générer des classes <code>BeanInfo</code> pour les beans. Le nom de la classe <code>BeanInfo</code> est le même que le nom du bean avec le suffixe "<code>BeanInfo</code>".</li><li>➔ La classe <code>BeanInfo</code> contient des méthodes publiques qui renvoient des informations relatives au bean, notamment la classe du bean, le nom de cette classe, ainsi que des informations relatives aux événements, les méthodes et les propriétés du bean !!!</li></ul> |
| BTDA/DAVID : Interface Personne-Machine     | 14  |