

JFace Data binding

JFace permet de mettre en place un système permettant de mettre à jour les interfaces sur modification du modèle, et de mettre à jour le modèle, depuis la modification d'une interface :

-- Crédit à la modélisation

Sur le principe, chaque modèle va émettre un événement sur modification de ses membres, événement communiqué aux listeners associés au modèle.

-- Classe de base AbstractModel

La classe **AbstractModel** va permettre d'ajouter des listeners associés à des membres, et de déclencher des événements. Toute classe intégrée au modèle et utilisant le Data binding devra hériter de AbstractModel.

```
public abstract class AbstractModelObject {  
    private final PropertyChangeSupport propertyChangeSupport = new  
    PropertyChangeSupport(  
        this);  
  
    public void addPropertyChangeListener(PropertyChangeListener listener) {  
        propertyChangeSupport.addPropertyChangeListener(listener);  
    }  
  
    public void addPropertyChangeListener(String propertyName,  
        PropertyChangeListener listener) {  
        propertyChangeSupport.addPropertyChangeListener(propertyName, listener);  
    }  
  
    public void removePropertyChangeListener(PropertyChangeListener listener) {  
        propertyChangeSupport.removePropertyChangeListener(listener);  
    }  
  
    public void removePropertyChangeListener(String propertyName,  
        PropertyChangeListener listener) {  
        propertyChangeSupport.removePropertyChangeListener(propertyName,  
            listener);  
    }  
  
    protected void firePropertyChange(String propertyName, Object oldValue,  
        Object newValue) {  
        propertyChangeSupport.firePropertyChange(propertyName, oldValue,  
            newValue);  
    }  
}
```

-- Crédit à la modélisation

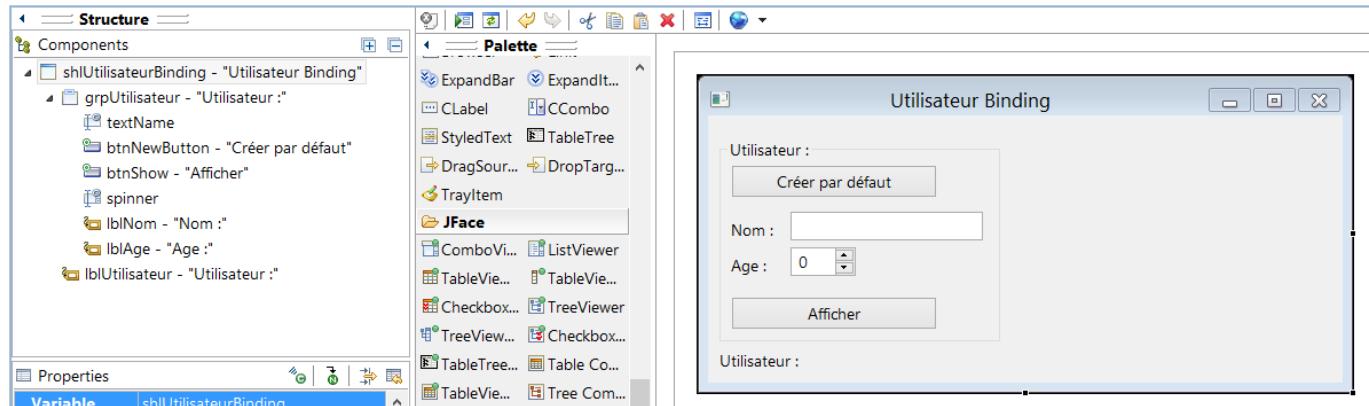
Création d'une classe Utilisateur :

- Utilisateur doit être un Bean (constructeur+getters/setters)
- Hériter de **AbstractModel**
- Appeler **firePropertyChange** sur la modification de ses membres

```
public class Utilisateur extends AbstractModelObject {  
    private String nom;  
    private int age;  
  
    public Utilisateur() {}  
  
    public Utilisateur(String nom, int age) {  
        this.nom = nom;  
        this.age = age;  
    }  
  
    /**  
     * @return the nom  
     */  
    public String getNom() {  
        return nom;  
    }  
  
    /**  
     * @param nom  
     *          the nom to set  
     */  
    public void setNom(String nom) {  
        String oldValue = this.nom;  
        this.nom = nom;  
        firePropertyChange("nom", oldValue, nom);  
    }  
  
    /**  
     * @return the age  
     */  
    public int getAge() {  
        return age;  
    }  
  
    /**  
     * @param age  
     *          the age to set  
     */  
    public void setAge(int age) {  
        int oldValue = this.age;  
        this.age = age;  
        firePropertyChange("age", oldValue, age);  
    }  
}
```

-- Création d'une vue

Créer la classe FormUtilisateur, à partir de l'assistant **SWT/window Application**



-- Implémentation du code

-- Définition du binding

Target	Model	Target Strategy	Model Strategy
textName.text(SWT.Modify)	user.nom	POLICY_UPDATE	POLICY_UPDATE
spinnerAge.selection	user.age	POLICY_UPDATE	POLICY_UPDATE

Code généré :

```
protected DataBindingContext initDataBindings() {
    DataBindingContext bindingContext = new DataBindingContext();
```

```
//  
    IObserveValue observeTextTextNameObserveWidget =  
WidgetProperties.text(SWT.Modify).observe(textName);  
    IObserveValue namePersonObserveValue =  
BeanProperties.value("nom").observe(user);  
        bindingContext.bindValue(observeTextTextNameObserveWidget,  
namePersonObserveValue, null, null);  
    //  
    IObserveValue observeSelectionSpinnerObserveWidget =  
WidgetProperties.selection().observe(spinnerAge);  
    IObserveValue ageUserObserveValue =  
BeanProperties.value("age").observe(user);  
        bindingContext.bindValue(observeSelectionSpinnerObserveWidget,  
ageUserObserveValue, null, null);  
    //  
    return bindingContext;  
}
```

From:
<http://slamwiki2.kobject.net/> - **SlamWiki 2.1**

Permanent link:
<http://slamwiki2.kobject.net/slam4/gui/jfacebinding?rev=1364429767>

Last update: **2019/08/31 14:39**

