

Interfaces Graphiques Java

Swing
JavaFX
SWT
GWT

Swing

- Classes avancées / AWT
- **Difficultés** à personnaliser :
 - Model
 - Controller
 - Renderer
- Layout **complexes** à déboguer
- Plugin NetBeans pour construire

JavaFX

- RIA : Appli web + **riches** (multi touch, mobile, animations, ... 3D)
- Version 1 en **2008** : nouveau langage de script
- Version 2 en **2011** : API, **inclus dans java 7**
 - Interactions : HTML – Javascript – Swing
 - Décomposition en **XML / Java / CSS**
 - Autre vocabulaire similaire à AWT
- Version 3 en cours dans java 8

SWT

- « Standard Widget Toolkit » de **Eclipse** (inclus dans JSF : « Java Server Faces » pour le développement de plugin)
- download swt.zip : 5.6M
 - import org.eclipse.swt.graphics.*;
 - import org.eclipse.swt.widgets.* ;
 - import org.eclipse.swt.layout.*;
- **Vocabulaire :**
 - Frame → Display
 - Panel → Shell
- Lourd à l'installation (plugins ...)

GWT

- « Google **Web** Toolkit »
- Possibilité standalone en cours
- Proches servlets
- Lourd installation

JavaFX

code source

- **Code java** (controller) :
 - Vocabulaire :
 - Application
 - Frame → Stage
 - Panel → Scene
 - Layout → Pane
- Fichier **FXML** : description des composants (vue)
- Fichier **CSS** : rendu

Swing / JavaFX

DemoSwing.java

```
public static void main(String[] args) {
    JFrame frame = new JFrame("Raccord");
    frame.setSize(800, 800);

    // titre
    frame.setLayout(new BorderLayout());
    JLabel toppanetext = new JLabel("Raccord");
    toppanetext.setBackground(Color.blue.brighter());
    toppanetext.setOpaque(true);
    frame.add(toppanetext, BorderLayout.NORTH);

    // layout central
    GridBagLayout layout = new GridBagLayout();
    JPanel centerpane = new JPanel(layout);
    frame.add(centerpane, BorderLayout.CENTER);
    GridBagConstraints c = new GridBagConstraints();
    c.insets = new Insets(10,10,10,10);
    c.fill=GridBagConstraints.NONE;

    // menu deroulant
    JComboBox<File> files = new JComboBox<File>(
        (new File(System.getProperty("user.dir")).listFiles()));
    files.addItemListener(new ItemListener() {
```

DemoFX.java

test.css

ParfileExplorer.fxml

DemoFXController.java

```
@Override
public void start(Stage primaryStage) throws IOException {

    // version FX vs Swing
    BorderPane pane = new BorderPane();
    Label toppanetext = new Label("Raccord");
    pane.setTop(toppanetext);
    Label centerpanetext = new Label("Some data here");
    pane.setCenter(centerpanetext);

    // version avec FXML
    Pane pane = (Pane) FXMLLoader.load(getClass().getResource("ParfileExplorer.fxml"));
    Scene sc = new Scene(pane, 800,500);
    sc.getStylesheets().add("test.css");
    primaryStage.setTitle("Raccord");
    primaryStage.setScene(sc);
    primaryStage.show();

}

public static void main(String[] args) {
    launch(args);
}
```

JavaFX

```
<?import javafx.scene.paint.*?>
<?import javafx.scene.text.*?>
<?import javafx.geometry.*?>
```

```
<BorderPane layoutX="0.0" layoutY="0.0" prefHeight="400.0" prefWidth="608.0"
```

```
xmlns:fx="http://javafx.com/fxml"
fx:controller="DemoFXController"
```

```
<center>
```

```
<GridPane alignment="CENTER" fx:id="grid">
```

```
<padding> <Insets top="10" right="10" bottom="10" left="10" />
```

```
<children>
```

```
<Label text="Fichier a ouvrir" />
```

```
<ChoiceBox GridPane.columnIndex="0" fx:id="myCombo">
```

```
<items>
```

```
<FXCollections fx:factory="ObservableList" />
```

```
</items>
```

```
</ChoiceBox>
```

```
<TextArea editable="false" prefHeight="550" prefWidth="550" GridPane.columnIndex="0" GridPane.vgrow="ALWAYS" GridPane.grow="ALWAYS" fx:id="parfileTxt" />
```

```
public class DemoFXController implements Initializable {
```

```
@FXML
```

```
private ChoiceBox<String> myCombo ;
```

```
@FXML
```

```
private TextArea parfileTxt;
```

```
@FXML
```

```
private ObservableList<String> files;
```

```
@FXML
```

```
private Label title;
```

```
@Override // This method is called by the FXMLLoader when initialization complete
```

```
public void initialize(URL fxmLocation, ResourceBundle resources) {
```

```
title.getStyleClass().add("my-title"); // Label title : style du CSS
```

```
files = FXCollections.observableArrayList(
```

```
(new File(System.getProperty("user.dir"))).listFiles();
```

```
myCombo.setItems(files);
```

```
// Ecoute la sélection
```

```
myCombo.getSelectionModel().selectedIndexProperty()
```

```
.addListener(new ChangeListener<Number>() {
```

```
//IDEM DANS FXML : onAction="#handleChoiceBoxAction">
```

```
@Override
```

```
public void changed(
```

```
ObservableValue<? extends Number> observable,
```

```
Number oldValue, Number newValue) {
```

```
String file = files.get(newValue.intValue());
```

```
List<String> lines;
```

```
try {
```

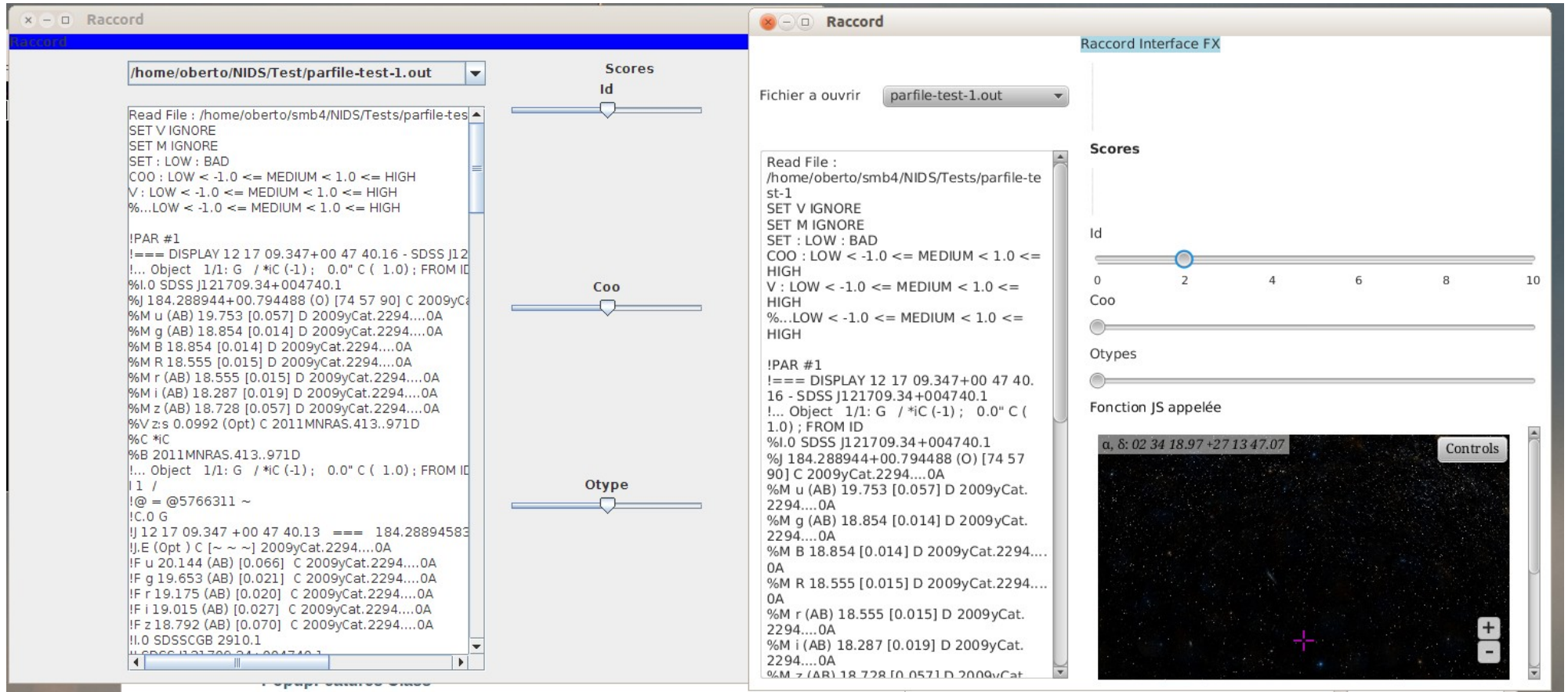
```
lines = Files.readAllLines((new File(file)).toPath(),
```

```
Charset.defaultCharset());
```

```
String text = "";
```

```
for (String string : lines) {
```


Swing / JavaFX



```
@FXML
WebView webView;
...
WebEngine webEngine = webView.getEngine();
String filepath = new File("/home/oberto/dev/demo/t2/src/AladinLite.html")
    .toURI().toURL().toString();
```

Java FX

Erreurs parfois cryptiques

Invalid property.

/home/oberto/dev/demo/t2/bin/ParfileExplorer.fxml:18

at javafx.fxml.FXMLLoader\$PropertyElement.<init>(FXMLLoader.java:1163)

at javafx.fxml.FXMLLoader.createElement(FXMLLoader.java:2372)

at javafx.fxml.FXMLLoader.processStartElement(FXMLLoader.java:2345)

- `<paddig>` → `<padding>`

Insets is not a valid type.

/home/oberto/dev/demo/t2/bin/ParfileExplorer.fxml:19

at javafx.fxml.FXMLLoader.createElement(FXMLLoader.java:2415)

at javafx.fxml.FXMLLoader.processStartElement(FXMLLoader.java:2345)

at javafx.fxml.FXMLLoader.load(FXMLLoader.java:2164)

- `<?import javafx.geometry.*?>`