



Developpement Graphique

Les Applications graphiques sont composées d'éléments graphiques (objets) de **type bouton, checkBox, champs de saisie, fenêtres diverses** .

Ces objets sont associés à des objets appelés (composants chez Java)

Ils sont disponibles dans des bibliothèques de classes.



Developpement Graphique

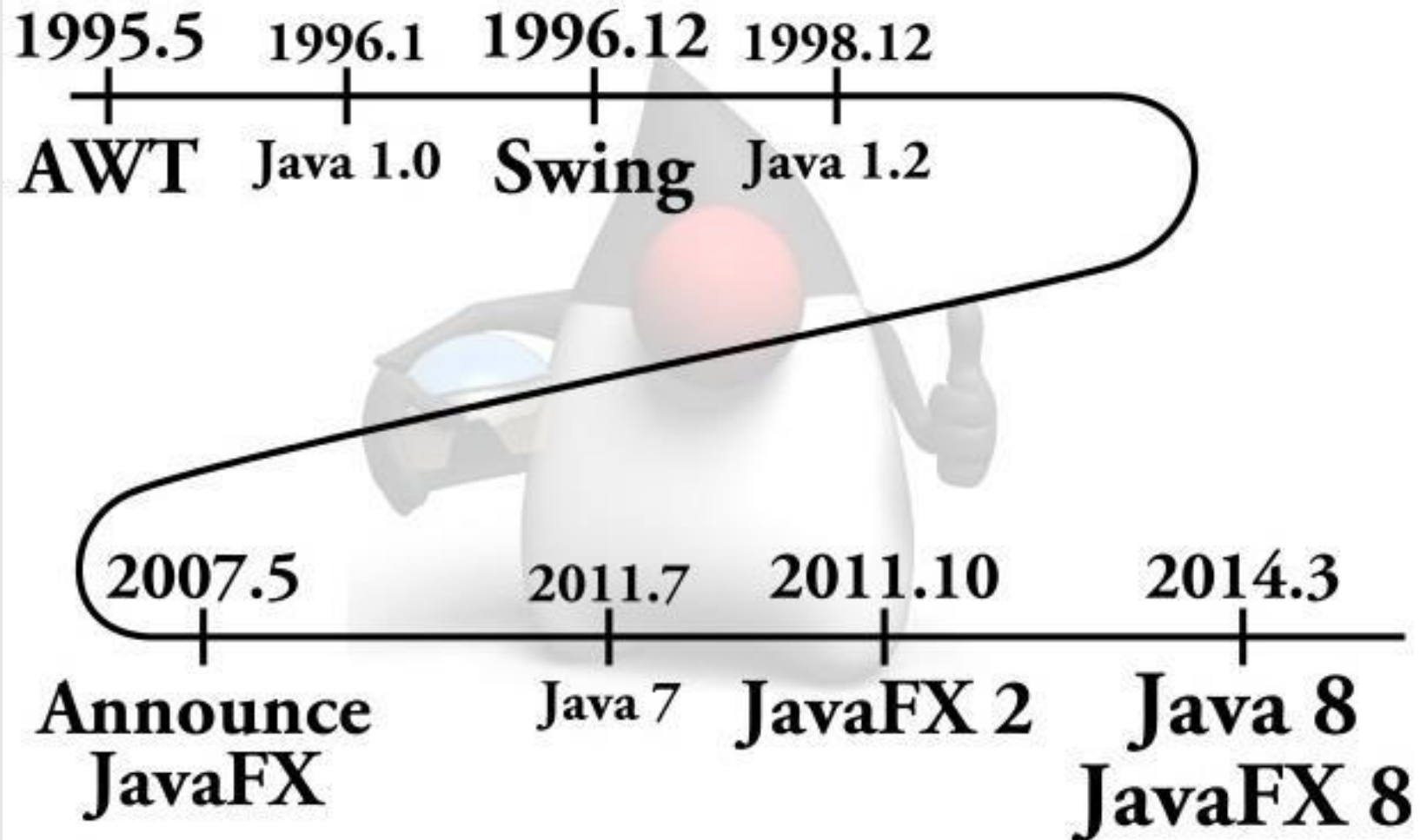
Exemples

A screenshot of a Java Swing window titled "Test Fxml". The window has a standard Mac OS-style title bar with minimize, maximize, and close buttons. Below the title bar is a menu bar with "Fichier", "Edition", and "?". The main content area has a light blue background and is titled "Exemple de composants graphiques". It contains several GUI components: a "CheckBox" with a small blue square icon, a "Date Picker" with a text field and a calendar icon, a "Slider" with a horizontal track and a circular knob, a "Combo Box" with a dropdown arrow, three radio buttons labeled "Oui", "Non", and "Ne sais pas", a text field with the placeholder text "votre question", and a "Validation" button at the bottom.



Développement Graphique

Historique





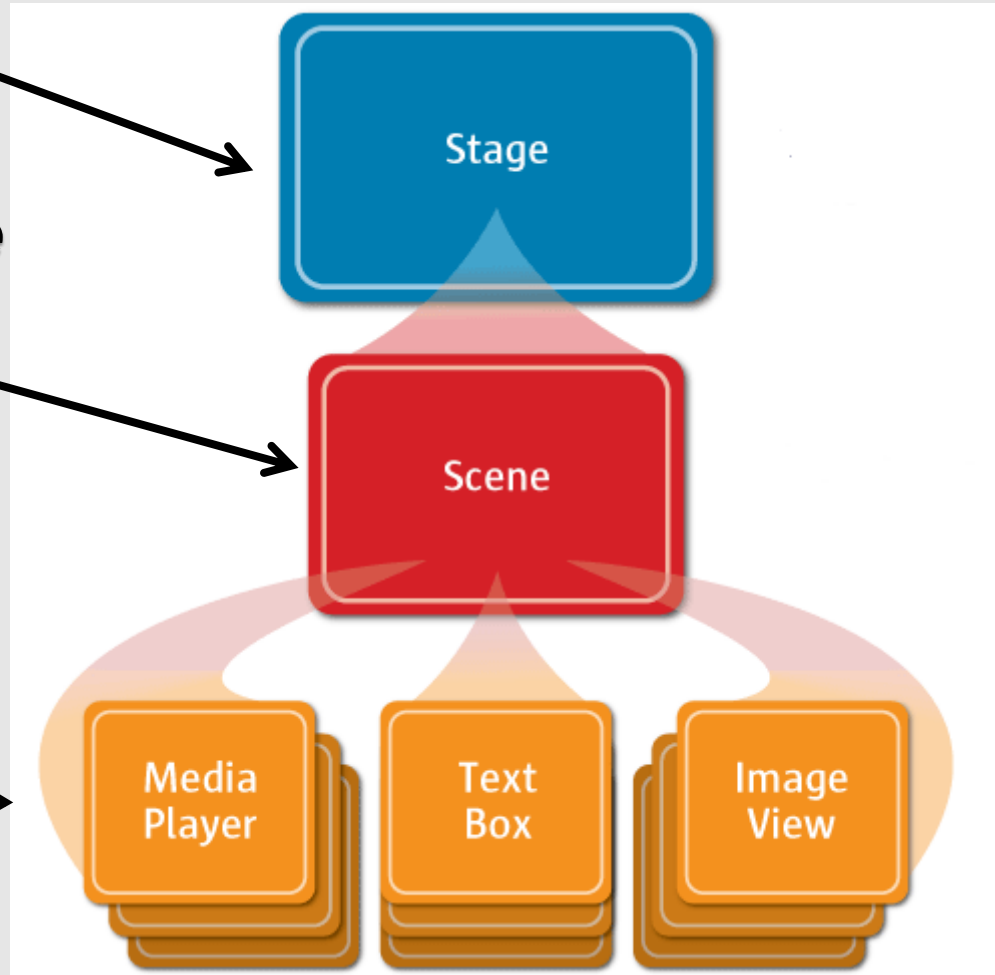
Développement Graphique

Structure d'une Application en JavaFX

Fenêtre principale

Scène (Container de composants)

Composants graphiques





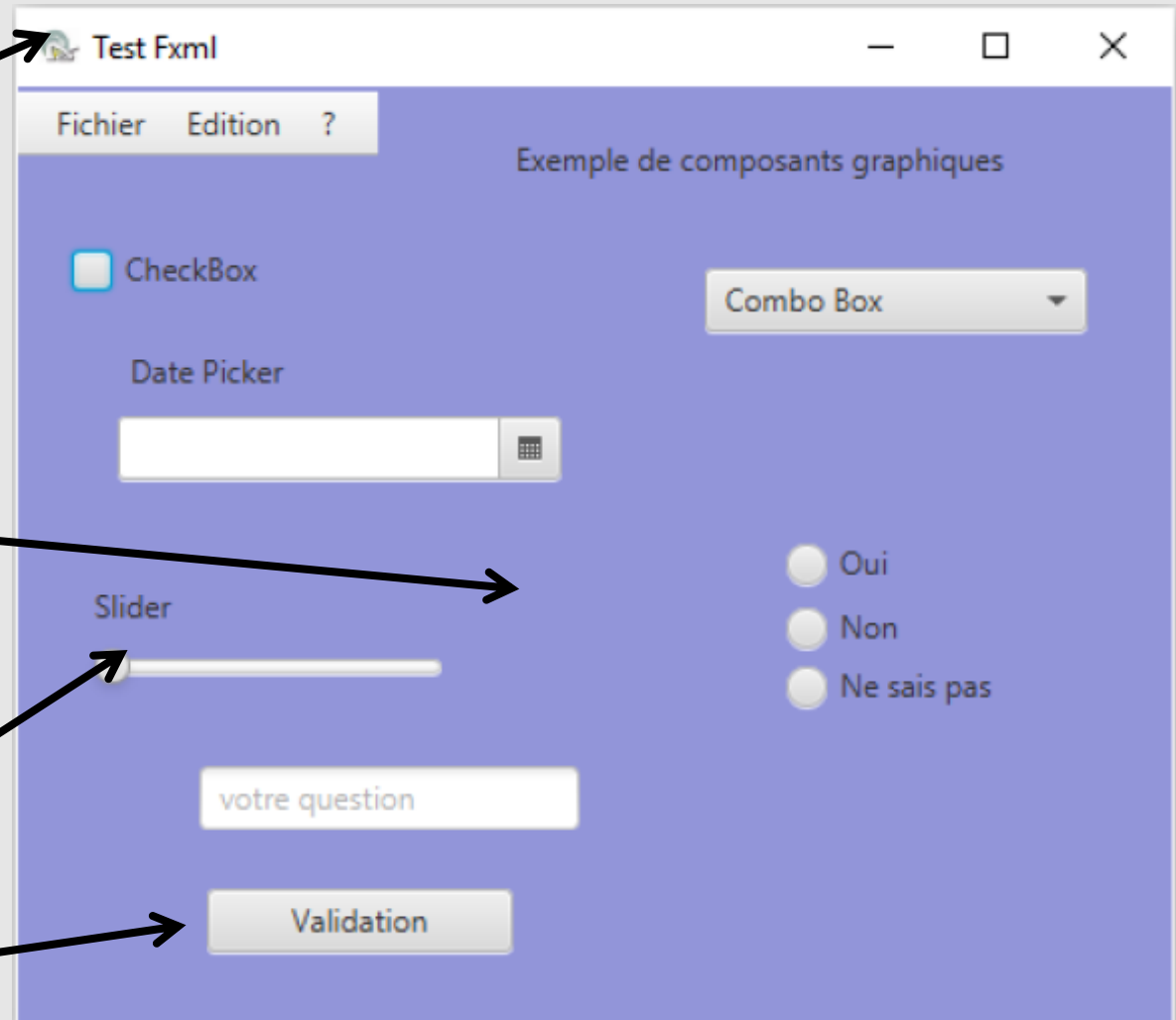
Developpement Graphique

Structure d'une Application en JavaFX

Fenêtre principale

Scène (Container de composants)

Composants graphiques





Développement Graphique

Création Projet sous IntelliJ

Choisir un projet de type JavaFX Application

The image shows the 'New Project' dialog in IntelliJ IDEA. On the left, a list of project types is shown, with 'JavaFX' selected and highlighted in blue. Other options include Java, Maven, Gradle, IntelliJ Platform Plugin, Java Enterprise, Spring Initializr, Quarkus, Micronaut, MicroProfile, Ktor, Raspberry PI, BeagleBone Black, and Korge. On the right, various configuration fields are visible: 'Name' is 'demo4', 'Location' is 'C:\Temp\demo4', 'Language' is 'Java' (with Kotlin and Groovy as options), 'Build system' is 'Maven' (with Gradle as an option), 'Test framework' is 'JUnit' (with TestNG as an option), 'Group' is 'com.example', 'Artifact' is 'demo4', and 'Project SDK' is 'liberica-17 BellSoft Liberica JDK vers'.

Java	Name:	demo4
Maven	Location:	C:\Temp\demo4
Gradle	Language:	Java Kotlin Groovy
IntelliJ Platform Plugin	Build system:	Maven Gradle
JavaFX	Test framework:	JUnit TestNG
Java Enterprise	Group:	com.example
Spring Initializr	Artifact:	demo4
Quarkus	Project SDK:	liberica-17 BellSoft Liberica JDK vers
Micronaut		
MicroProfile		
Ktor		
Raspberry PI		
BeagleBone Black		
Korge		



Développement Graphique

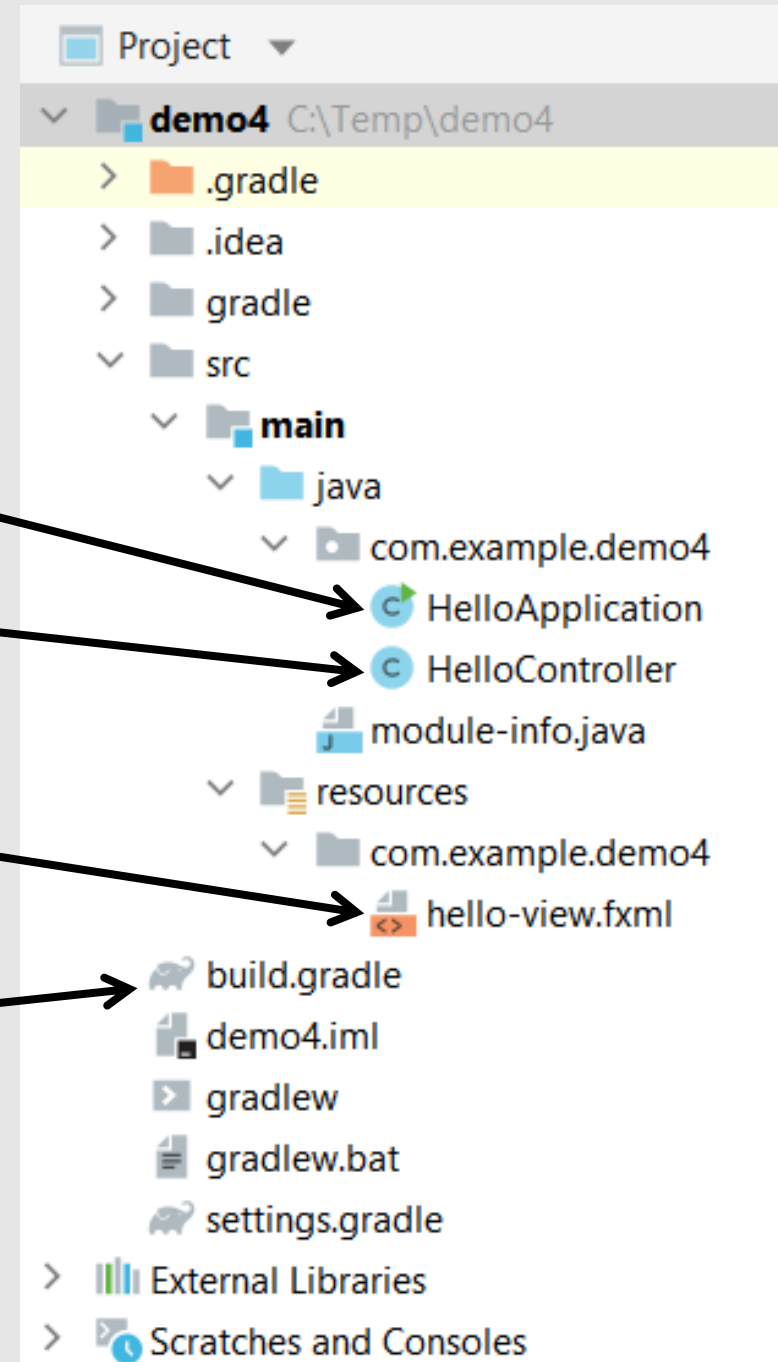
Création Projet sous IntelliJ

Classe principale

Classe contrôleur

Fichier fxml décrivant l'interface graphique

Gradle est un système de build proposant une approche flexible pour la construction de projets Java (nous le recroiseront !!)





Développement Graphique

La Classe Principe **A MODIFIER** pour avoir:

```
package sample;

import javafx.application.Application;
import javafx.event.EventHandler;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.stage.Stage;
import javafx.stage.WindowEvent;

public class Main extends Application {

    @Override
    public void start(Stage primaryStage) throws Exception{
        Parent root = FXMLLoader.load(getClass().getResource("sample.fxml"));
        primaryStage.setTitle("Binding");
        primaryStage.setScene(new Scene(root));
        primaryStage.show();
        primaryStage.setResizable(false);
        primaryStage.setOnCloseRequest(new EventHandler<WindowEvent>() {
            @Override
            public void handle(WindowEvent event) { System.exit(1); }
        });
    }

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

La méthode **main** appelle la méthode **launch(..** de la classe **Application** qui crée un objet **primaryStage** et lance la méthode **start(..**



Développement Graphique

La Classe Principe :

Création d'un objet Parent à partir du fichier fxml décrivant la vue

Interdiction du redimensionnement de la fenêtre

Création de l'objet Scene Association de la scene au stage

Affichage de la fenêtre

```
public class Main extends Application {  
  
    @Override  
    public void start(Stage primaryStage) throws Exception {  
        Parent root = FXMLLoader.load(getClass().getResource("sample.fxml"));  
        primaryStage.setTitle("Binding");  
        primaryStage.setScene(new Scene(root));  
        primaryStage.show();  
        primaryStage.setResizable(false);  
  
        primaryStage.setOnCloseRequest(new EventHandler<WindowEvent>() {  
            @Override  
            public void handle(WindowEvent event) { System.exit(1); }  
        });  
    }  
  
    public static void main(String[] args) { launch(args); }  
}
```

Récupère le click sur « X » à la fermeture de la fenêtre: pour fermeture propre du logiciel (si theads)



Développement Graphique

La Classe Controller **A MODIFIER** pour avoir:

```
package sample;

import javafx.fxml.Initializable;

import java.net.URL;
import java.util.ResourceBundle;

public class Controller implements Initializable {
    @Override
    public void initialize(URL url, ResourceBundle resourceBundle) {

    }
}
```



Développement Graphique

Le fichier fxml décrivant l'interface graphique

Le fichier fxml (la vue) est construit graphiquement dans le concepteur graphique Scene Builder et/ou dans l'éditeur IntelliJ.

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.*?>
<?import javafx.scene.layout.*?>

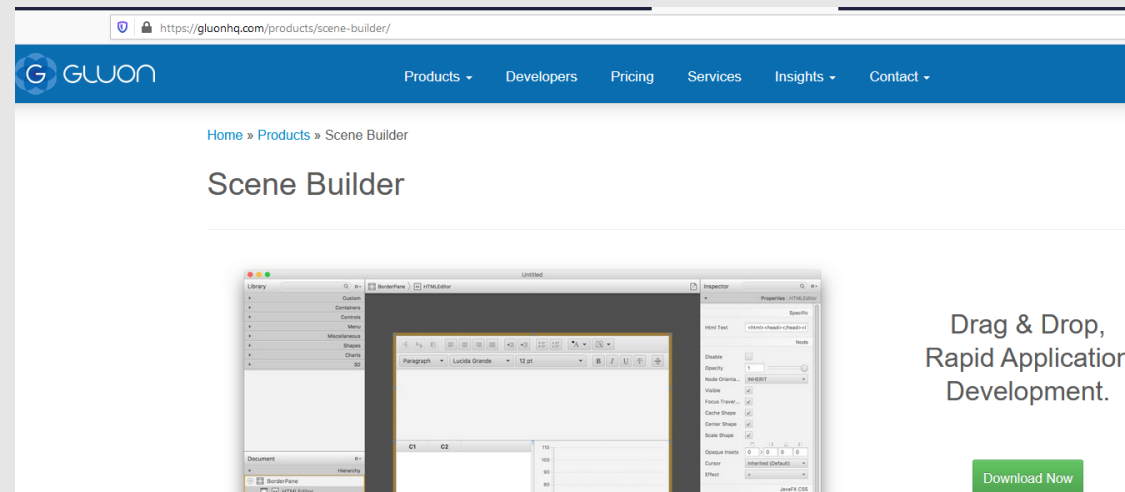
<AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0" prefWidth="450.0"
  stylesheets="@../css/mycss.css" xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1" fx:controller="sample.Controller">
  <children>
    <TextField fx:id="hauteur" alignment="CENTER" layoutX="68.0" layoutY="117.0" prefHeight="32.0" prefWidth="100.0" promptText="hauteur" />
    <TextField fx:id="largeur" alignment="CENTER" layoutX="68.0" layoutY="197.0" prefHeight="32.0" prefWidth="100.0" promptText="largeur" />
    <TextField fx:id="perimetre" alignment="CENTER" layoutX="268.0" layoutY="133.0" prefHeight="26.0" prefWidth="100.0" />
    <TextField fx:id="surface" alignment="CENTER" layoutX="268.0" layoutY="187.0" prefHeight="26.0" prefWidth="100.0" />
  </children>
</AnchorPane>
```

Important (ajouter si absent)



Complément pour Scene Builder externe

1. Download et installer la dernier version de Scene Builder (Gluon)



2. Dans IntelliJ

1. Settings
2. Languages and Frameworks
3. JavaFX.

Languages & Frameworks > JavaFX

Path to SceneBuilder: `C:\Program Files\SceneBuilder\SceneBuilder.exe`