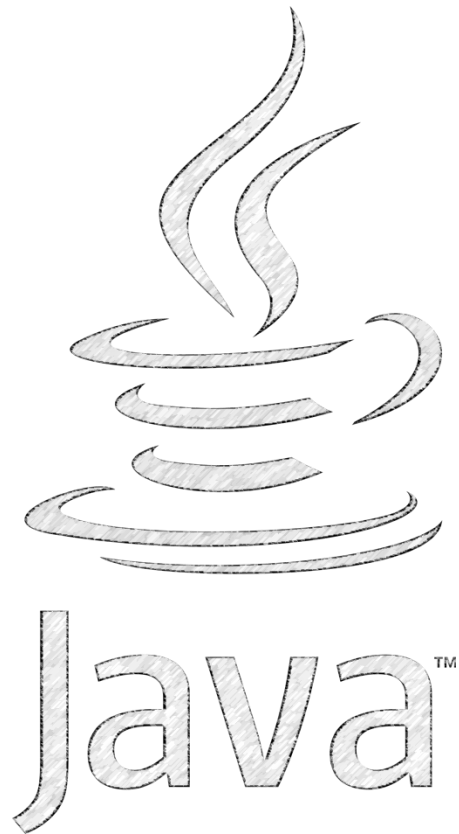


Programmation Orientée Objets avec Java

Stéphane Malandain, Yassin Rekik



Chapitre 9

Introduction à JavaFX

- Environnement
 - Installing Java
 - Installing IDE : Eclipse / IntelliJ
 - Installing JavaFX SDK
 - Running First JavaFX Apps
- Utilisation du Framework JavaFX
 - Stage – Scene – Layouts - Controls
- Utilisation de Scene Builder et CSS

- Eclipse : <https://www.eclipse.org/downloads/>
- IntelliJ IDEA : <https://www.jetbrains.com/idea/download/#section=mac>
- Netbeans : <https://netbeans.apache.org/>

Install JavaFX

- Java FX ne fait plus partie du SDK Java , il doit donc être ajouté à votre projet
- JavaFX est actuellement maintenu comme projet Open Source par : Gluon
- Install JavaFX : <https://gluonhq.com/products/javafx/>
- Aide pour configuration : <https://openjfx.io/openjfx-docs/#install-javafx>

« Hello World » en JavaFX

5

- With IntelliJ IDEA
 - Open New Java FX Project – Selecting the right Java SDK
 - File -> Project Structure -> Libraries
 - Add JavaFX : JavaFX folder/lib
 - Apply – Finish
 - Run -> Edit Configuration
 - VM options : `--module-path Path/lib --add-modules javafx.controls,javafx.fxml`

« Hello World » en JavaFX

6

- With Eclipse
 - Window (Help) -> Eclipse Marketplace
 - Install e(fx)clipse plugin and restart
 - File -> New -> Project : Java FX Project
 - Set the appropriate Java JRE
 - Window (Eclipse) -> Preferences -> User Libraries
 - New JavaFX -> Add external Jars
 - Add all jar in lib
 - Run -> Run configuration :
 - Arguments : `--module-path Path/lib --add-modules javafx.controls,javafx.fxml`

JavaFX : Structure interne

7

- Stage est la fenêtre dans laquelle les interfaces vont être visualisées
 - C'est une instance de la classe Stage (package javafx.stage).
- Stage contient une scène active qui définit l'interface graphique comme un graphe (Scene-Graph) - une structure de données arborescente des éléments visuels d'une application, tels que les contrôles, les formes, les images, la vidéo, le texte et les images.
 - La scène est une instance de la classe Scene (package javafx.scene).
- Les contrôles sont des composants de l'interface graphique, tels que
 - les Label qui affichent du texte
 - les TextField qui permettent à un programme de recevoir des données de l'utilisateur
 - les Button sur lesquels les utilisateurs cliquent pour lancer des actions, etc.

JavaFX Application Layout

8

- Une fenêtre d'application en JavaFX est connue sous le nom de Stage.
 - `package javafx.stage`
- Une scène contient une scène active qui est définie par un gestionnaire de Layout.
 - `package javafx.scene`
- La scène peut avoir d'autres Layout Manager pour organiser les contrôles dans une arborescence.
 - Les nœuds avec enfants sont des Containers – Layout Managers.
 - Les nœuds sans enfants sont des Widgets.

JavaFX App Window Structure

9

- Chaque élément visuel du graphe de scène est un nœud - une instance d'une sous-classe de Node (package javafx.scene), qui définit les attributs et les comportements communs à tous les nœuds
- À l'exception du premier nœud du graphe de scène, le nœud racine, chaque nœud du graphe de scène a un parent.
- Les nœuds peuvent subir des transformations (déplacement, rotation et mise à l'échelle, par exemple), avoir une opacité (transparence, transparence partielle ou opacité), des effets (ombres portées, flou, réflexion et éclairage, par exemple) et bien d'autres choses encore.

- Les nœuds avec enfants sont généralement des conteneurs qui organisent leurs nœuds enfants dans la scène.
- Les conteneurs contiennent des contrôles qui acceptent des widgets ou d'autres conteneurs.
- Lorsque l'utilisateur interagit avec un contrôle, celui-ci génère un événement.
- Les programmes peuvent répondre à ces événements - connus sous le nom de gestion d'événements - afin de spécifier ce qui doit se passer lors de chaque interaction avec l'utilisateur.
- Un gestionnaire d'événements est une méthode qui répond à une interaction utilisateur. Les gestionnaires d'événements d'une interface graphique FXML sont définis dans une classe dite "contrôleur".

Scene Graph

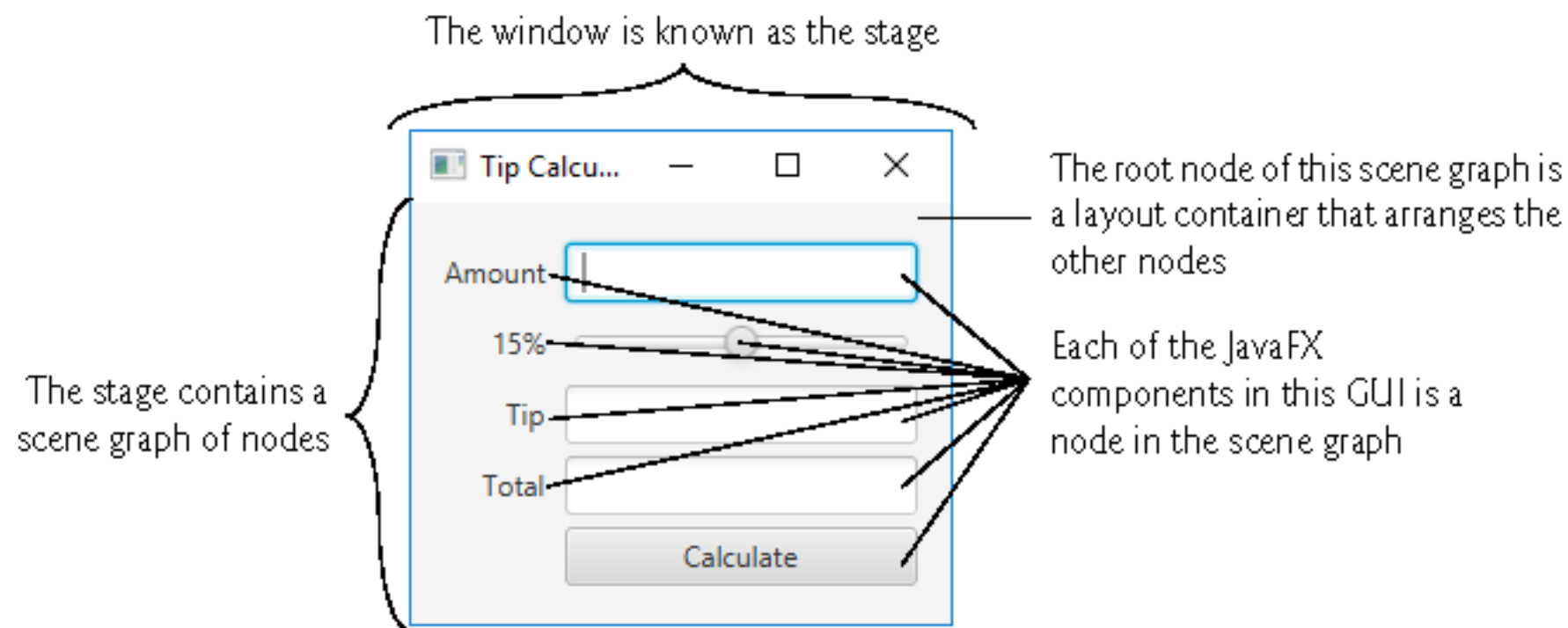
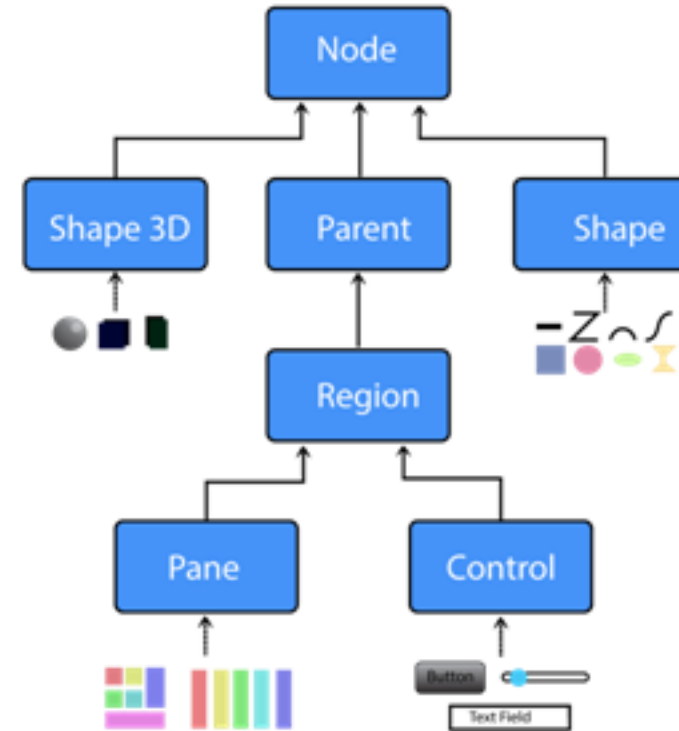
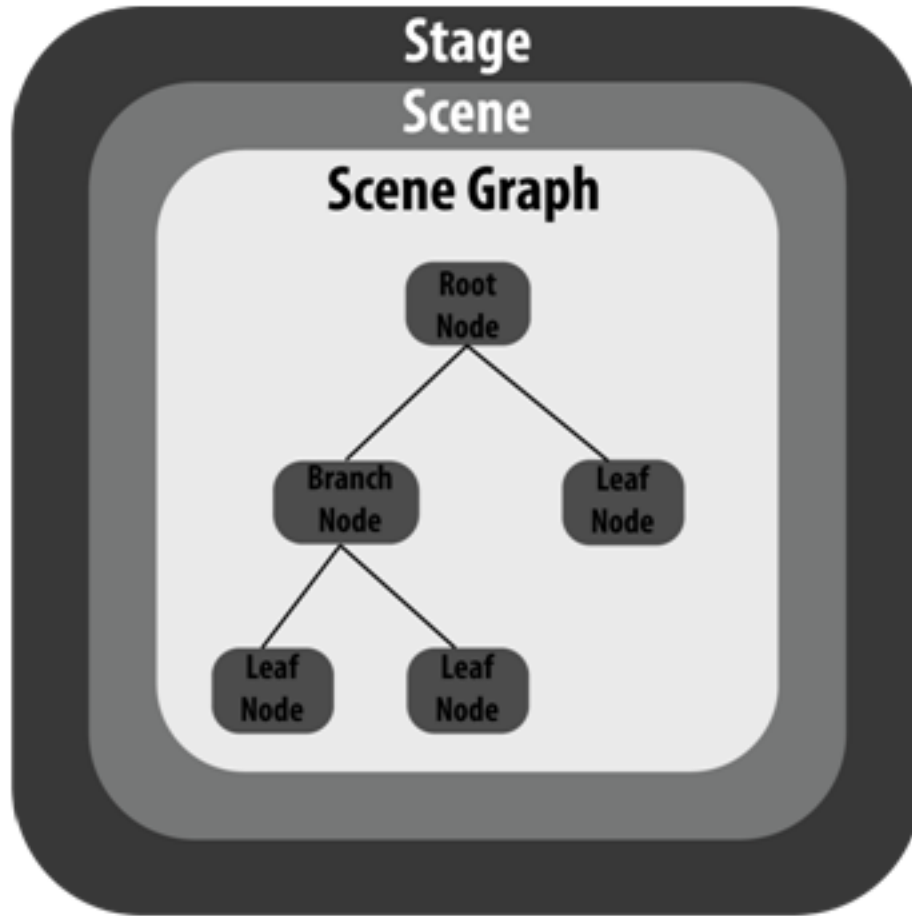


Fig. 12.1 | JavaFX app window parts.

Scene Graph

12



- Classe Application
 - Une sous-classe de la classe Application est nécessaire pour lancer une application JavaFX
- (package `javafx.application`).
- Lorsque la méthode principale de la sous-classe est appelée, la méthode statique de lancement de l'application est appelée :
 - La méthode statique `launch()` de l'application est appelée, ce qui transfère le contrôle à JavaFX pour démarrer l'application.
 - La méthode `launch()` amène le moteur d'exécution JavaFX à créer un objet de la sous-classe Application et à appeler sa méthode de démarrage.
 - La méthode `start` de la sous-classe Application crée l'interface graphique, l'attache à une scène et la place sur la scène que `start` reçoit en argument.

Exemple d'utilisation JavaFX

14

- A faire :
 - Créer une App simple
 - Simple « Click Me ! » App
 - Alternner entre différents Scenes
 - Utiliser plusieurs fenêtres : Multiple Windows
 - Communication entre différentes fenêtres
 - Fermeture d'application et gestion des événements
 - Utilisation des gestionnaires de positionnement : Layouts

JavaFX Classes

15

- <http://tutorials.jenkov.com/javafx/index.html>
- <https://www.tutorialspoint.com/javafx/index.htm>

Utilisation de Scene Builder

16

- Installer : Scene Builder
- Création d'un fichier fxml dans votre project
- Importation de fxml dans une scene
- Utilisation des Containers - Controls – Layouts
- Gestion des Callback → Controler

Exemple plus complet

17

