

Cours R2.02

Introduction à l'Interaction Humain-Machine

Cours 1 : introduction

Contact

2

<https://gery.casiez.net>

Home Publications Teachings Service Students

Géry Casiez



Professeur des Universités en informatique à l'Université de Lille

Professor of Computer Science at Univ. Lille

Junior member (2018 - 2023) of Institut Universitaire de France ([IUF](#))

Adjunct Professor (2018 - 2028) at the [University of Waterloo](#), Canada

Member of the Inria [Loki](#) research team

Member of [CRISTAL](#)

Inria, office B103

Parc scientifique de la Haute-Borne

40, Avenue Halley - Bât B - Park Plaza

59650 Villeneuve d'Ascq

+33 (0)3 59 35 87 87

gery (dot) casiez (at) univ-lille (dot) fr

CRISTAL

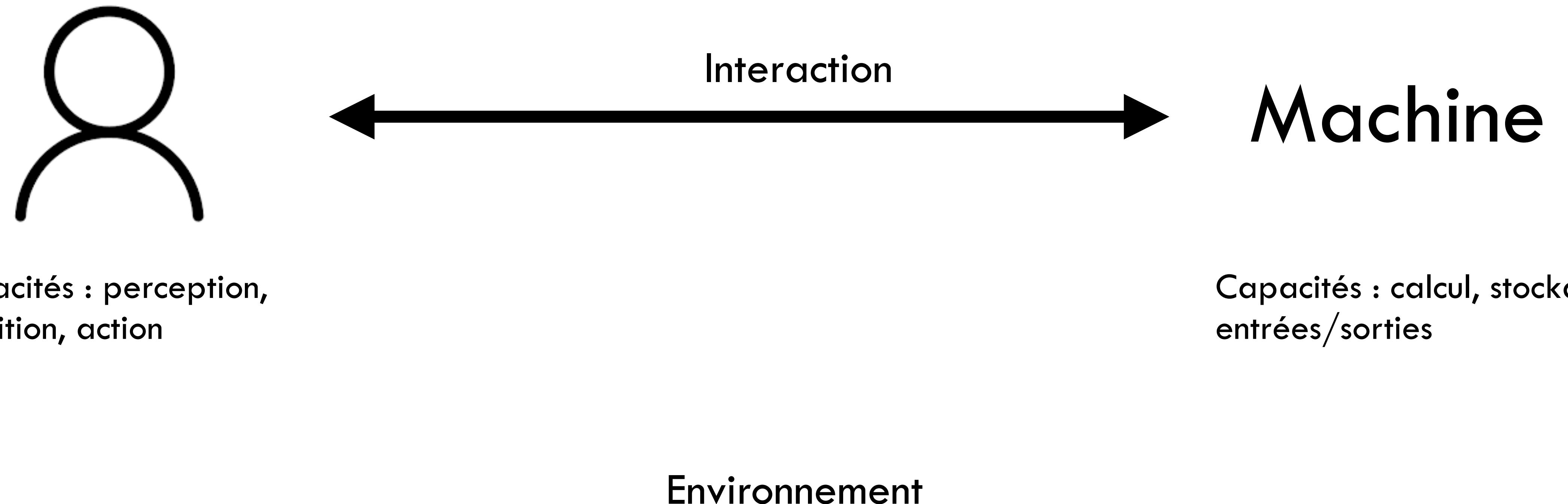
Bureau S1.20, ESPRIT, Cité Scientifique

+33 (0)3 28 77 85 44

I'm a Full Professor of Computer Science, Classe Exceptionnelle, at [Université de Lille](#). My research is in the area of Human-Computer Interaction (HCI), carried in the [Loki](#) research team, created in partnership between the [Inria centre at the University of Lille](#) and the Université de Lille within the Joint Research Unit [UMR 9189 CRISTAL](#). I teach in the CS department at [IUT de Lille](#). My [works](#) have been published at [ACM CHI](#), [ACM UIST](#), [IEEE VR](#) or [IHM](#), for which I've participated as a PC member or in the organization. In 2018, I have been appointed junior member of the [Institut Universitaire de France](#), became Adjunct-Professor at the [University of Waterloo](#) and deputy director of CRISTAL. My research interests include input interaction (design of input devices, interaction techniques, study of transfer functions, latency measure and compensation, and noise reduction in interactive systems), VR and 3D interaction (study of embodiment and design of interaction techniques and input devices), interactive documents, haptics, multitouch interaction and BCI.

Interaction Humain-Machine

3



Système interactif

4

Un système interactif n'est pas un système algorithmique.

Un **système interactif** est un système qui prend en compte, pendant son exécution, des informations communiquées par un ou plusieurs utilisateurs, et qui fournit, pendant son exécution **une représentation perceptible de son état interne**.

Pourquoi étudier l'IHM ?

5

De nombreux systèmes donnent l'impression de “tomber en marche”.

Pourquoi sont-ils si mal conçus ou mal implémentés ?

Paradoxe de l'Interaction Humain-Machine

Importance de l'IHM

6

La conception de l'interface peut prendre jusqu'à 80 % du temps de développement.

La qualité de l'interaction / l'interface constitue la vitrine du produit et conditionne son succès ou son échec.

Enjeux importants.

L'interface idéale

7

Ne demanderait aucun apprentissage.

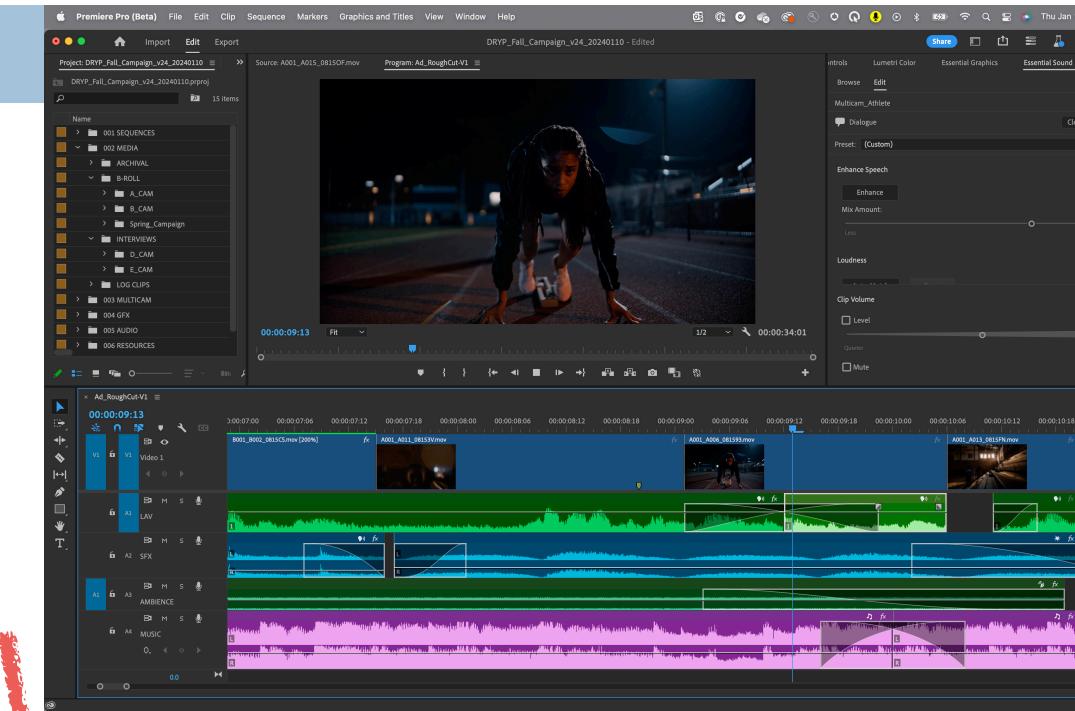
Permettrait aux utilisateurs de réaliser rapidement et sans erreur toutes leurs tâches avec peu d'effort.

Favoriserait la satisfaction subjective des utilisateurs.

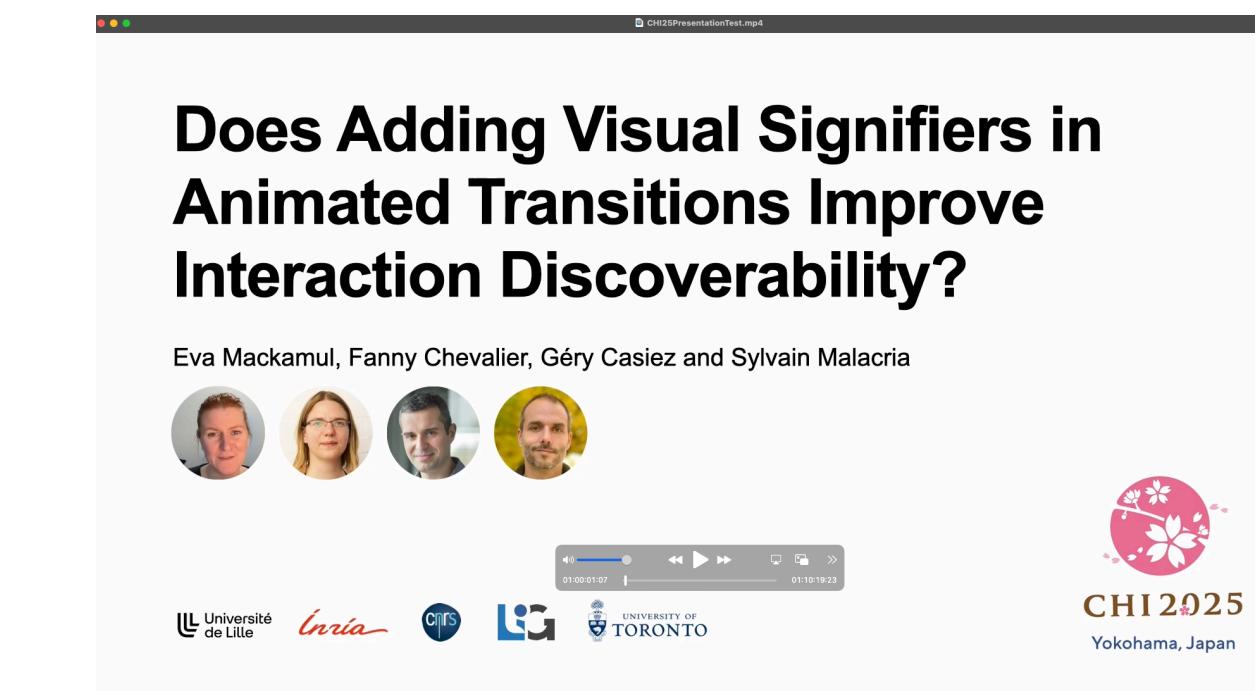
Compromis pouvoir d'expression / simplicité

8

Pouvoir d'expression



Adobe Premiere



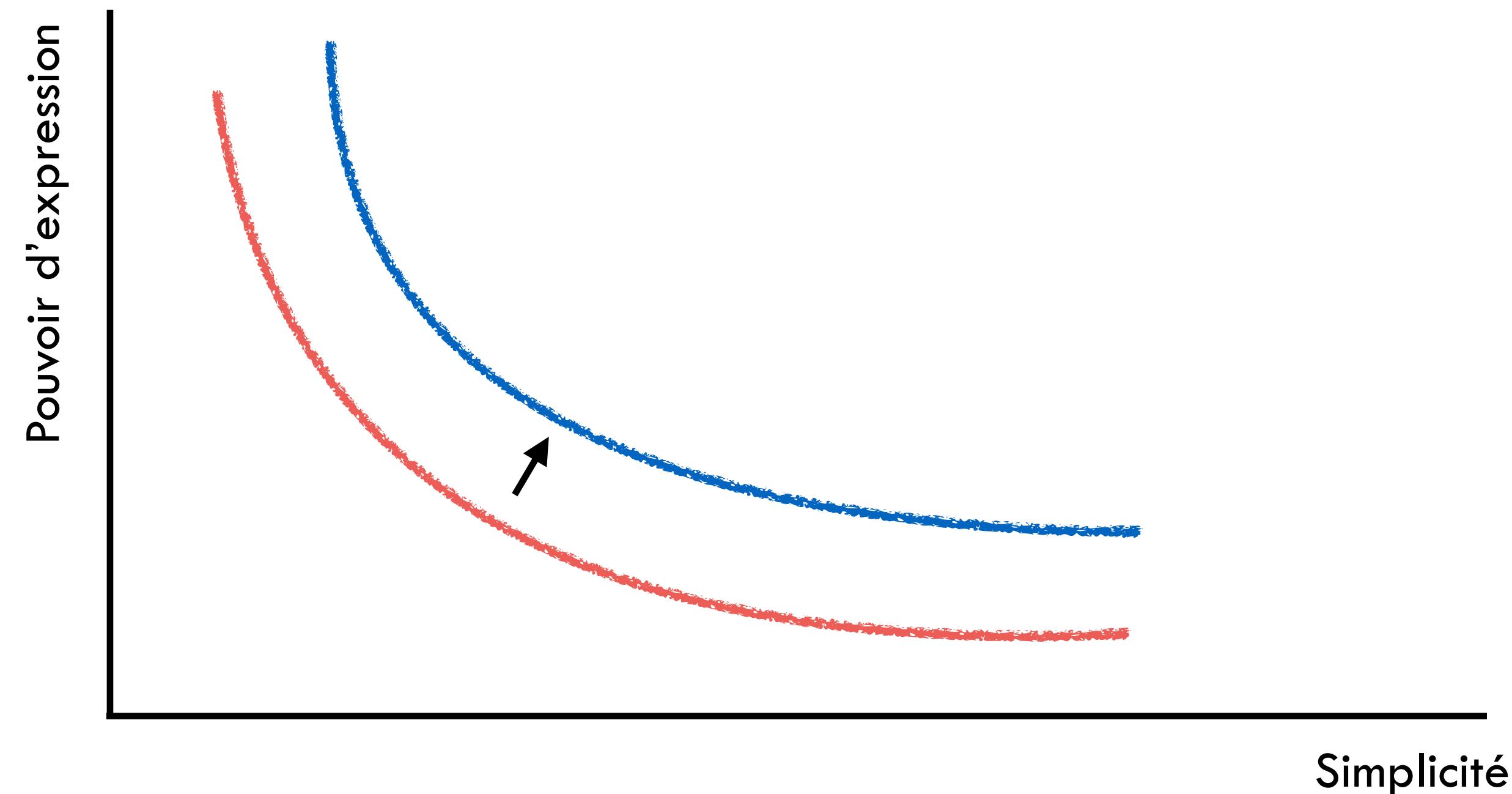
Quicktime

Simplicité

“simple things should be simple, complex things should be possible” [A. Kay]

Compromis pouvoir d'expression / simplicité

9



Présentation du cours

10

Objectifs

Savoir spécifier, concevoir, et développer une IHM

Maîtriser la programmation événementielle

Conception et prototypage d'IHM

Notions de théorie, heuristiques, évaluation

En pratique

Cours sur 9 semaines (1hCM + 1,5hTD + 3hTP/semaine)

Evaluation : DS + une partie de la SAÉ 2.01

Langage support : Java (JavaFX)

Lien avec les autres modules

11

Pré-requis

R2.01 Dev Objets

Pré-requis pour

Programmation Web – client riche

Conception et développement d'applications mobiles

+ SAÉ 2.01 & 2.02

Plan du cours en 9 semaines

12

1. Introduction à l'interaction, widgets

2. Programmation événementielle

3. Widgets et événements (1/2)

4. Widgets et événements (2/2)

5. Conception et prototypage (1/2)

6. Conception et prototypage (2/2)

7. Heuristiques et recommandations

8. Modèles et théories

9. Méthodes d'évaluation des IHM

Moodle

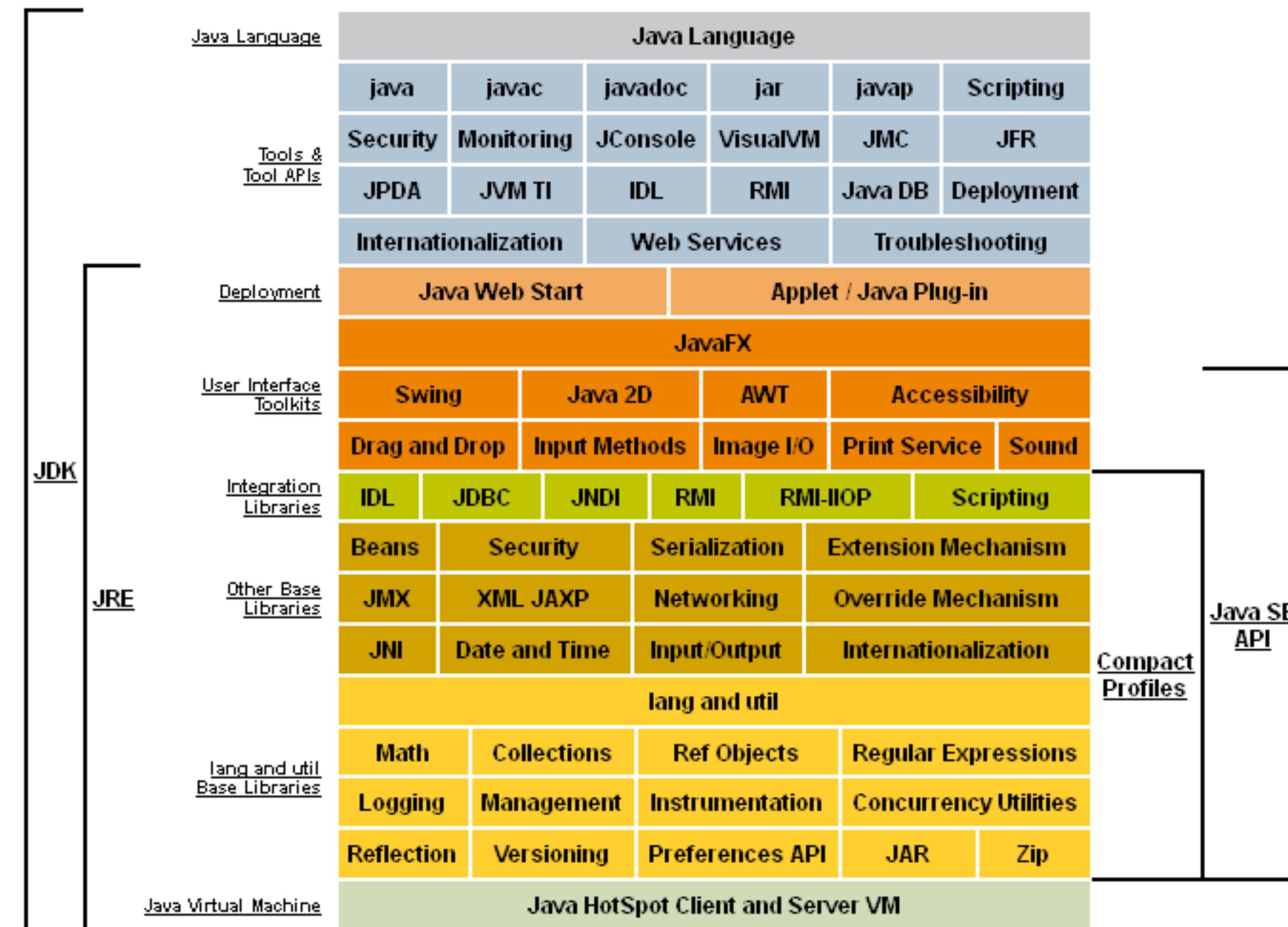
13

<https://moodle.univ-lille.fr/course/view.php?id=682>

The screenshot shows the Moodle course page for "R2.02 : Introduction à l'Interaction Humain-Machine". The top navigation bar includes links for "Moodle ULille", "Mes cours", "Ce cours", and language selection ("Français (fr)"). The main title "R2.02 : Introduction à l'Interaction Humain-Machine" is prominently displayed. Below the title, a breadcrumb navigation shows the path: "Accueil" > "Mes cours" > "Institut universitaire de technologie (IUT)" > "IUT - Site de Villeneuve d'Ascq" > "IUT - Informatique" > "BUT - Informatique" > "BUT S2" > "R2.02". The page has a decorative background featuring colorful abstract shapes.

Composants de la plateforme Java

14



La librairie AWT

15

La librairie AWT (Abstract Window Toolkit) a été introduite dès les premières versions de Java.

Utilisation de ressources propres au système encapsulées dans des abstractions.

L'affichage des composants est géré par le système

=> l'apparence des composants est différente selon les plateformes

Fonctionnalités assez rudimentaires.

La librairie Swing

16

Swing utilise des éléments d'AWT mais offre un grand nombre de nouveaux composants.

Les éléments graphiques ont la même apparence quelque soit le système d'exploitation.

Utilisation du Look-and-feel pour configurer la représentation visuelle.

Les noms des classes de la librairie Swing commencent par la lettre **J**.

La librairie JavaFX

17

Successeur de Swing

2008 : JavaFX 1.0

2011 : JavaFX 2.0

2014 : JavaFX 8 inclus dans JRE/JDK Java 8 + support multiplateformes

2018 : JavaFX redevient un module séparé à partir de Java 11

2021 : version 17 LTS

2025 : version 24

Stage : la fenêtre de l'application

```
import javafx.application.Application;  
import javafx.stage.Stage;
```

```
public class HelloJavaFXStage extends Application {
```

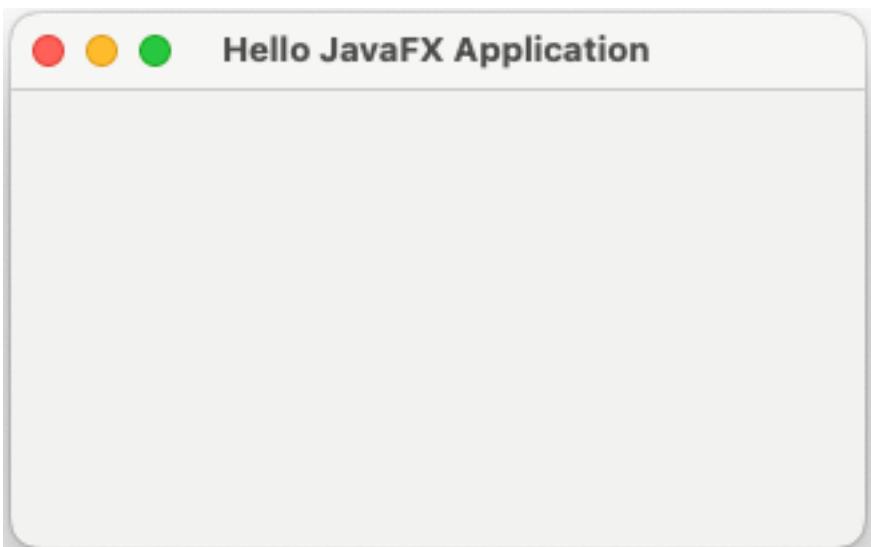
```
    @Override public void start(Stage stage) {  
        stage.setTitle("Hello JavaFX Application");  
        stage.show();  
    }
```

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

le point d'entrée d'une application JavaFX est une classe qui hérite de la classe abstraite Application

start prend en paramètre la fenêtre qui a été créée par l'application

on trouvera toujours cette ligne à la fin de start pour afficher la fenêtre



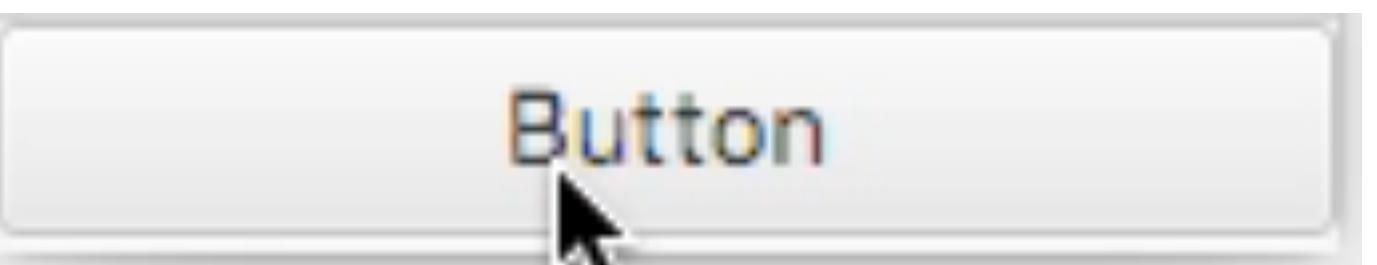
launch réalise un certain nombre de travaux préparatoires avant d'appeler la méthode start

Widgets / Control

19

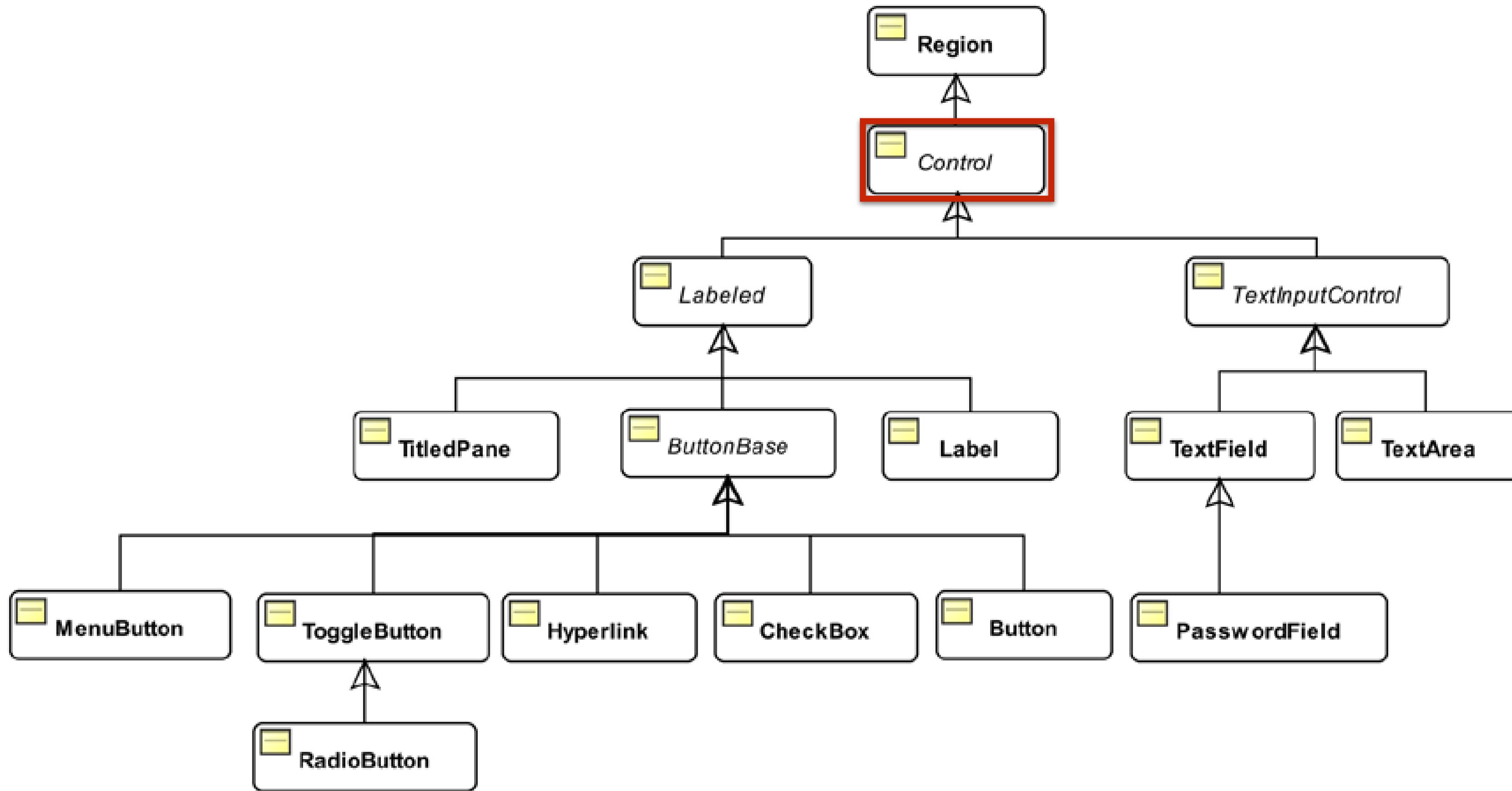
Le widget est un objet graphique interactif.

On parle de *control* avec JavaFX.



Aperçu des Controls

20



Scene



cours1/SimpleScene.java

21

```
public class SimpleScene extends Application {
```

```
    public void start(Stage stage) {  
        Button bouton = new Button("Bouton");
```

```
        Scene scene = new Scene(bouton, 300, 50);  
        stage.setScene(scene);
```

```
        stage.setTitle("Hello JavaFX Application with a Scene");  
        stage.show();  
    }
```

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

Stage contient un objet Scene qui contient les widgets de la fenêtre



Évolution des widgets

22

Widgets de base du macintosh (1984)

Bouton (button)

Potentiomètre (slider)

Case à cocher (check box)

Bouton radio (radio button)

Champ texte (text field)

Boites de dialogue pour les fichiers (file open/save dialog)

Ajouts ultérieurs : menus hiérarchiques, listes, combo box, onglets

Tâches élémentaires d'interaction

23

Quelques exemples

Saisie

Sélection

Déclenchement

Défilement

Spécification d'arguments et de propriétés

...

Saisie

24

Saisie de texte

Boite de saisie et clavier

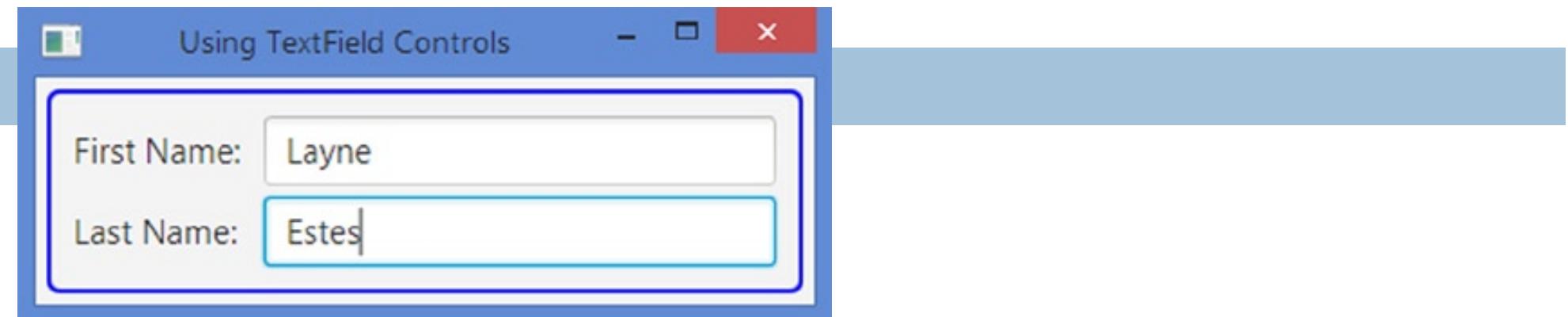
Saisie de quantités

Ex : potentiomètre

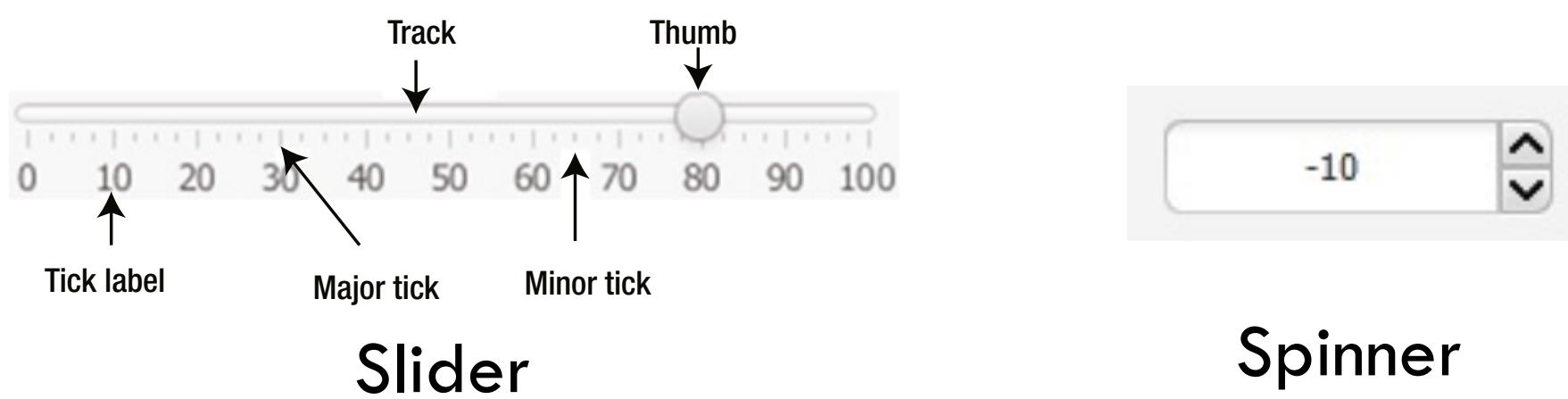
Saisie de positions

Pointage

Saisie de tracés



TextField



Slider

Spinner



Sélection



25

Choix d'un ou plusieurs éléments dans un ensemble

Cardinal fixe ou variable

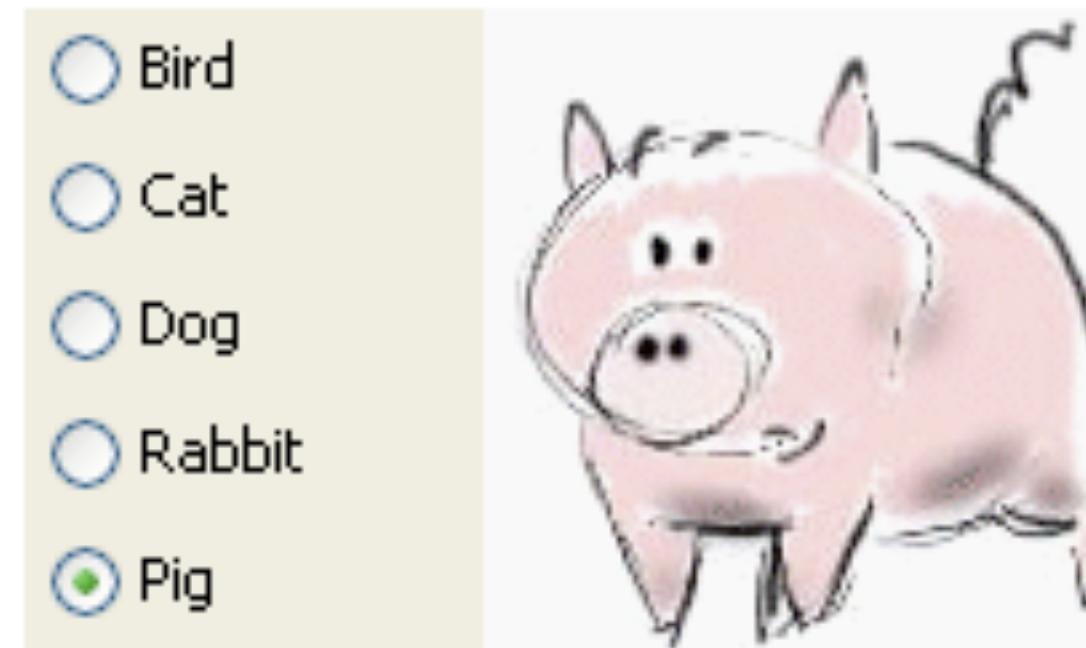
Cardinal petit ou grand

Exemples

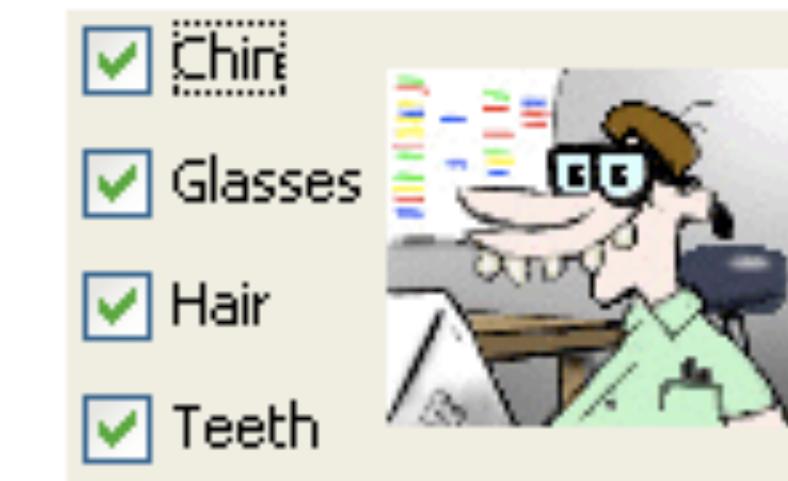
Cardinal fixe : menu, boutons radio, cases à cocher

Cardinal variable : pointage, liste, saisie de nom ou combinaison de ces deux dernières techniques

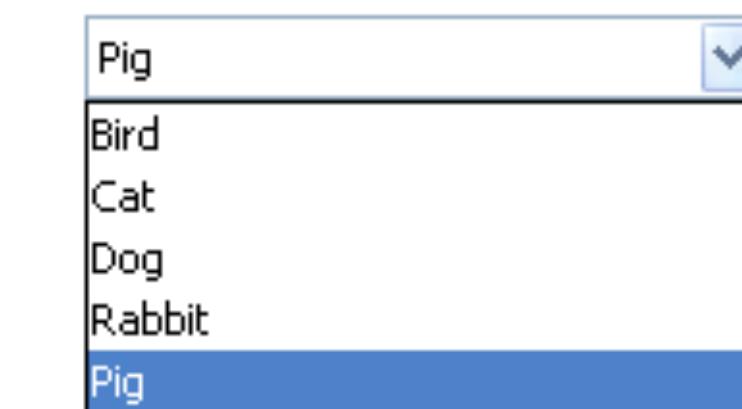
Sélection multiple : groupe ou intervalle, ajout et retrait



RadioButton



CheckBox



ComboBox



ListView

Sélection

26

Menus

Déroulants

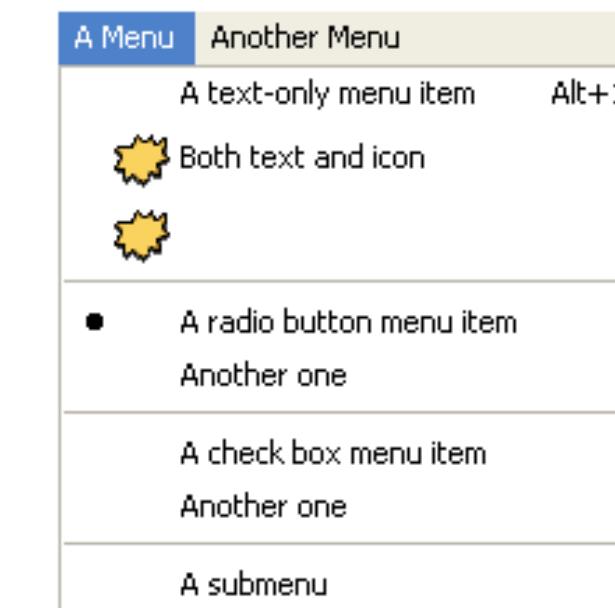
Pop-up

Palettes

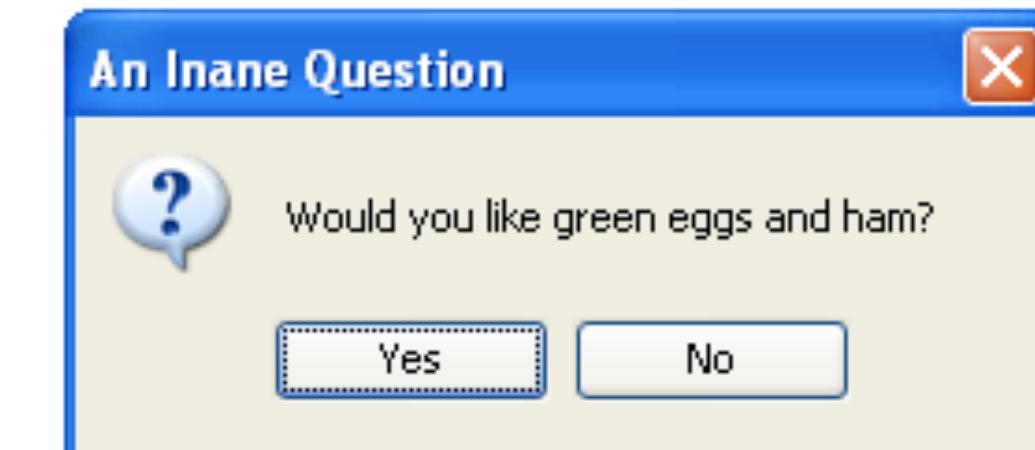
Organisation du menu

Hiérarchique

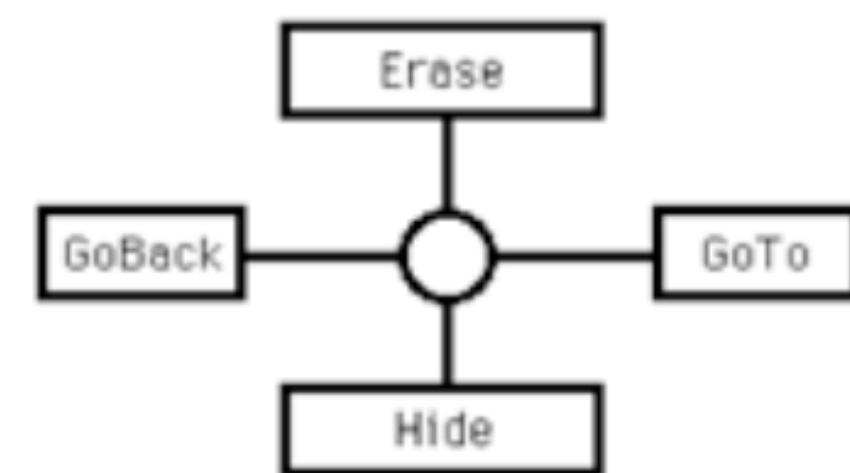
Circulaire



MenuBar



Stage



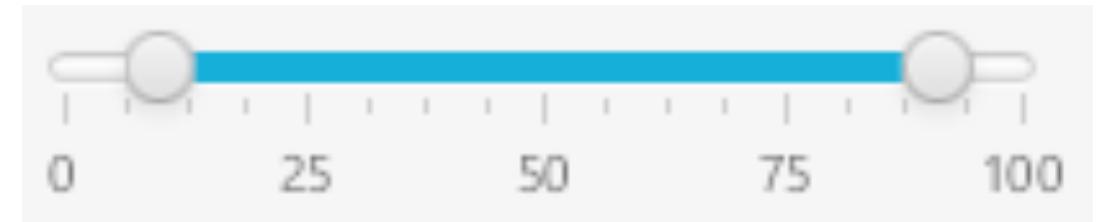
Sélection



27

Le range slider

Permet une sélection d'une plage de valeurs

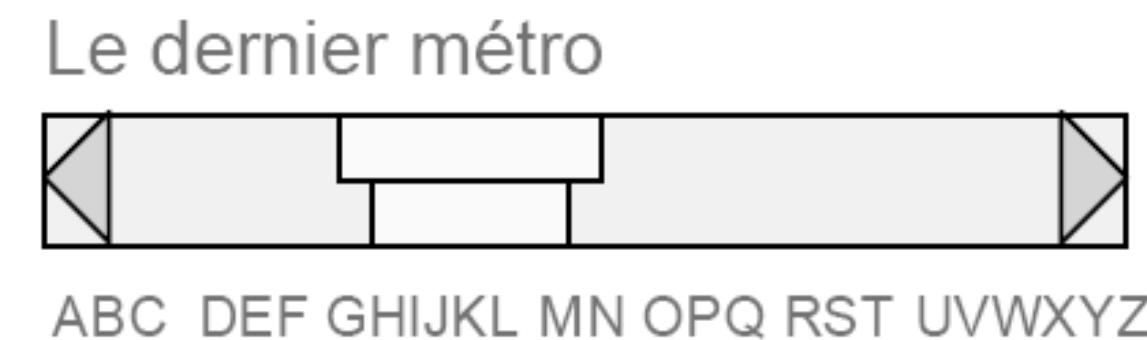


RangeSlider

L'alpha slider

Permet de sélectionner rapidement un élément dans une longue liste

Exemple : de 13 à 24 secondes pour trouver un film dans une liste de 10 000



Déclenchement

28

Boutons et menus

Cliquer-tirer (drag-and-drop)

L'action dépend de la source et de la destination

Entrée gestuelle

Spécification simultanée de la commande et de l'objet



Button



supprimer



déplacer



copier

Défilement



29

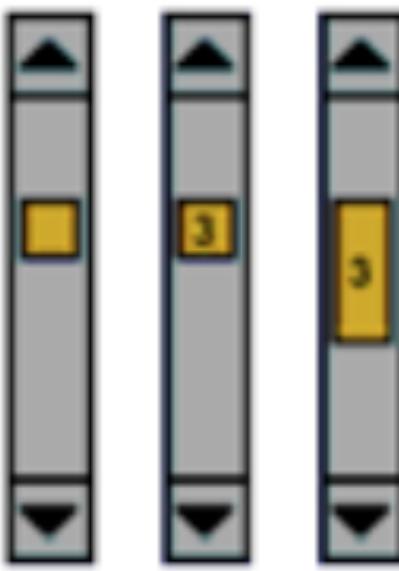
Barres de défilement

Sens du défilement ?

Découplage spatial

Défilement direct

Défilement automatique

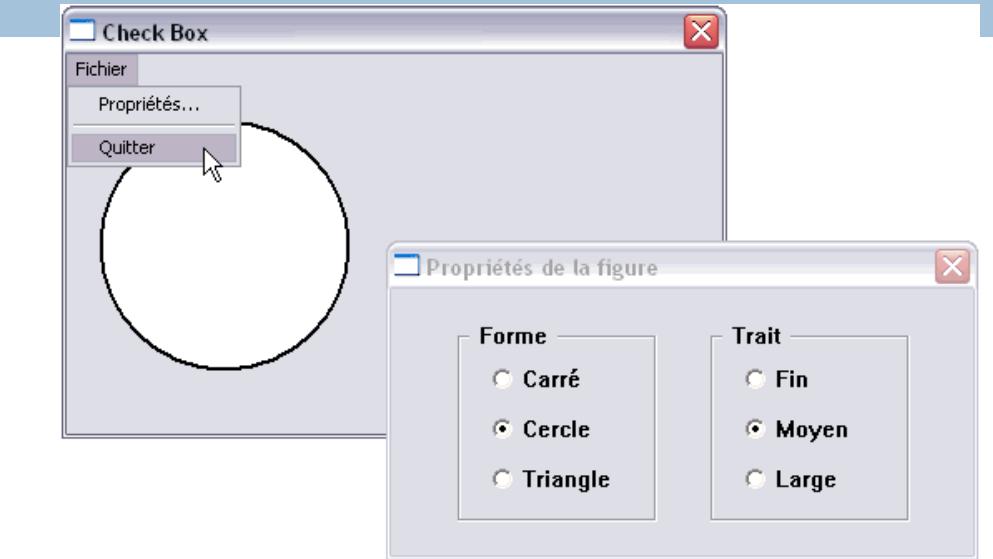


Spécification d'arguments et de propriétés

30

Boites de dialogue

Découplage temporel et spatial entre la spécification de la commande, de ses paramètres et son exécution

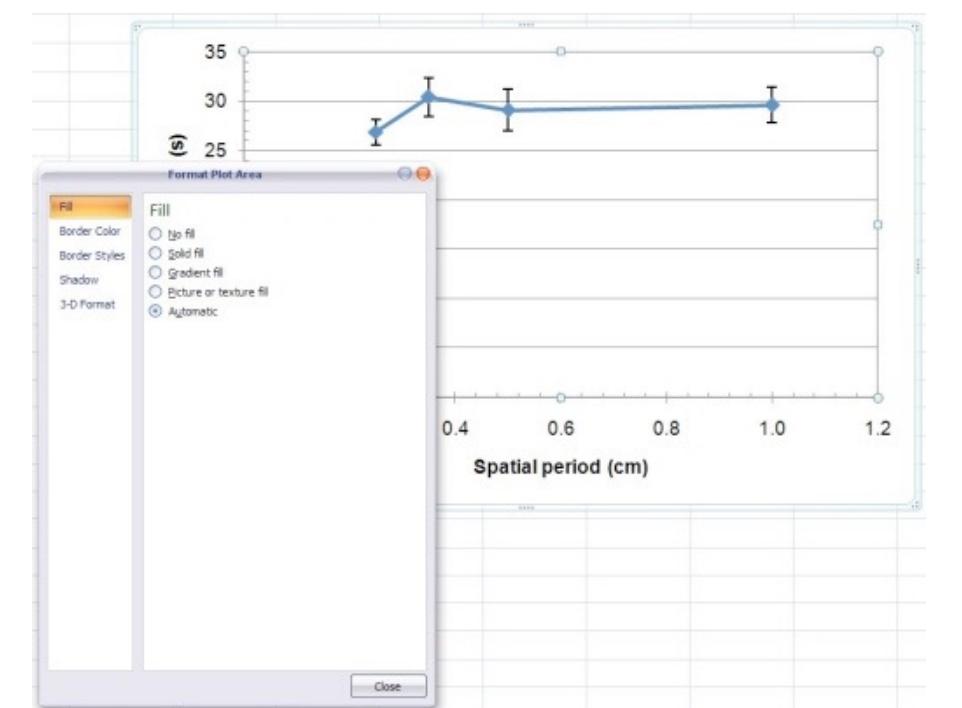


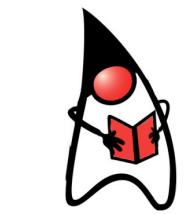
Boites modales ou non modales

Parties optionnelles, boites à onglet

Boites de propriétés

Effet immédiat des modifications sur les objets de la sélection





Stage

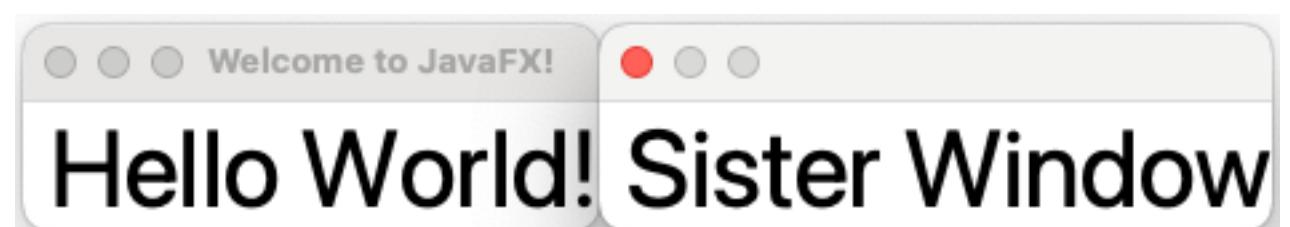
cours1/FenetreModale.java

31

Fenêtre modale

Une fenêtre B est modale par rapport à une fenêtre A si l'affichage de B empêche l'accès – et non la vue – à la fenêtre A.

```
Stage stage2 = new Stage();
stage2.initOwner(stage);
stage2.initModality(Modality.WINDOW_MODAL);
```



Enum Constants
Enum Constant and Description
APPLICATION_MODAL Defines a modal window that blocks events from being delivered to any other application window.
NONE Defines a top-level window that is not modal and does not block any other window.
WINDOW_MODAL Defines a modal window that blocks events from being delivered to its entire owner window hierarchy.