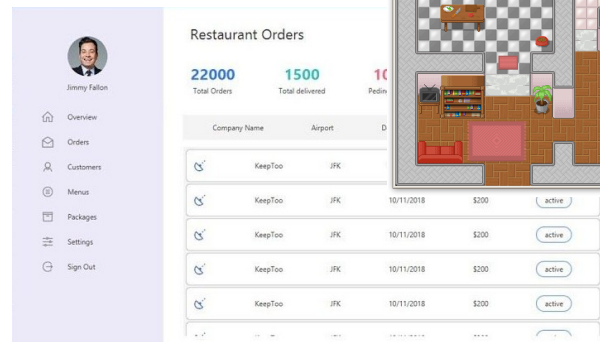
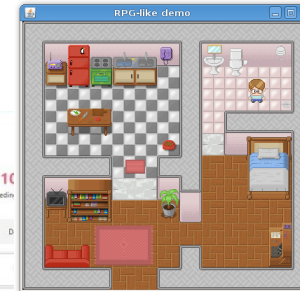
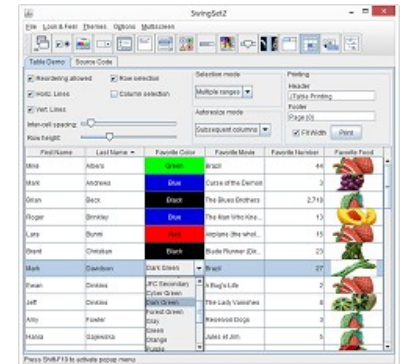


Interfaces Homme Machine

JAVA FX



Fabrice PELLEAU



Gestion des événements

- Un événement est
 - La réponse à une action utilisateur
 - clique sur un bouton, saisie d'un champ, sélection d'une option
- Le principe de délégation consiste
 - A faire correspondre
 - à un objet émetteur d'un événement
 - un objet porteur de la fonction à appliquer lorsque l'événement d'un type particulier est « potentiellement » déclenché.

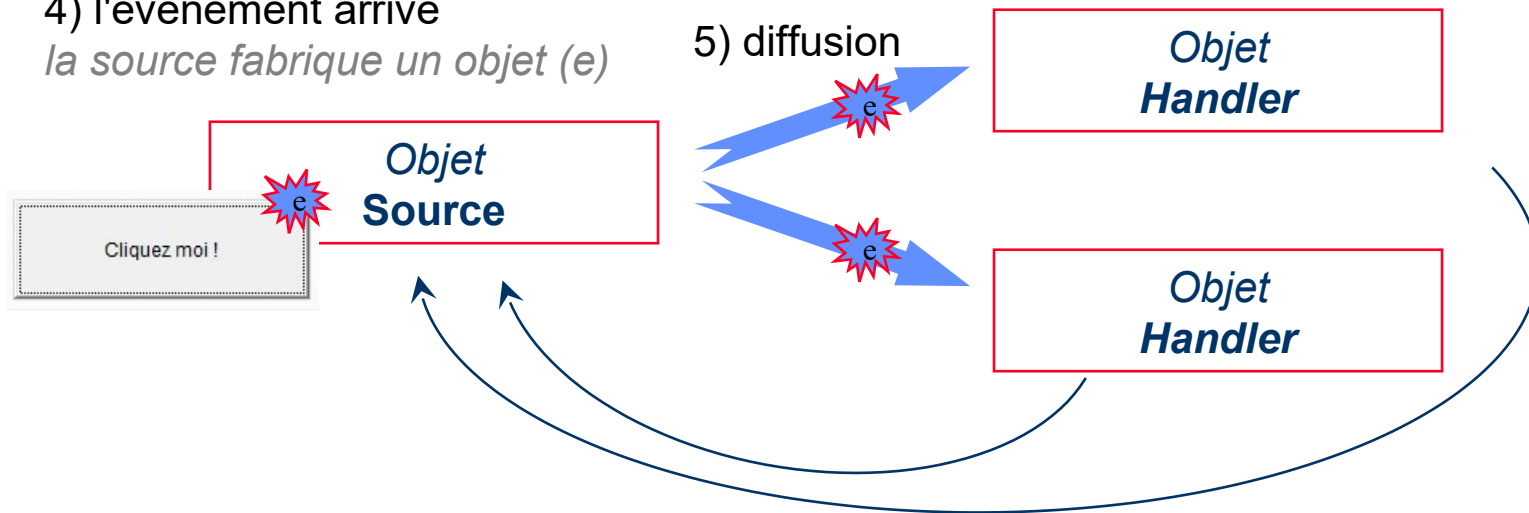
Gestion des événements

- Modèle basé sur la délégation
 - Source, Event, Listener/Handler

- 1) une classe « capable »
- 2) un objet de cette classe
- 3) abonnement (à une source)

4) l'événement arrive
la source fabrique un objet (e)

5) diffusion



Abonnement => `lasource.addEventHandler(leHandler)`

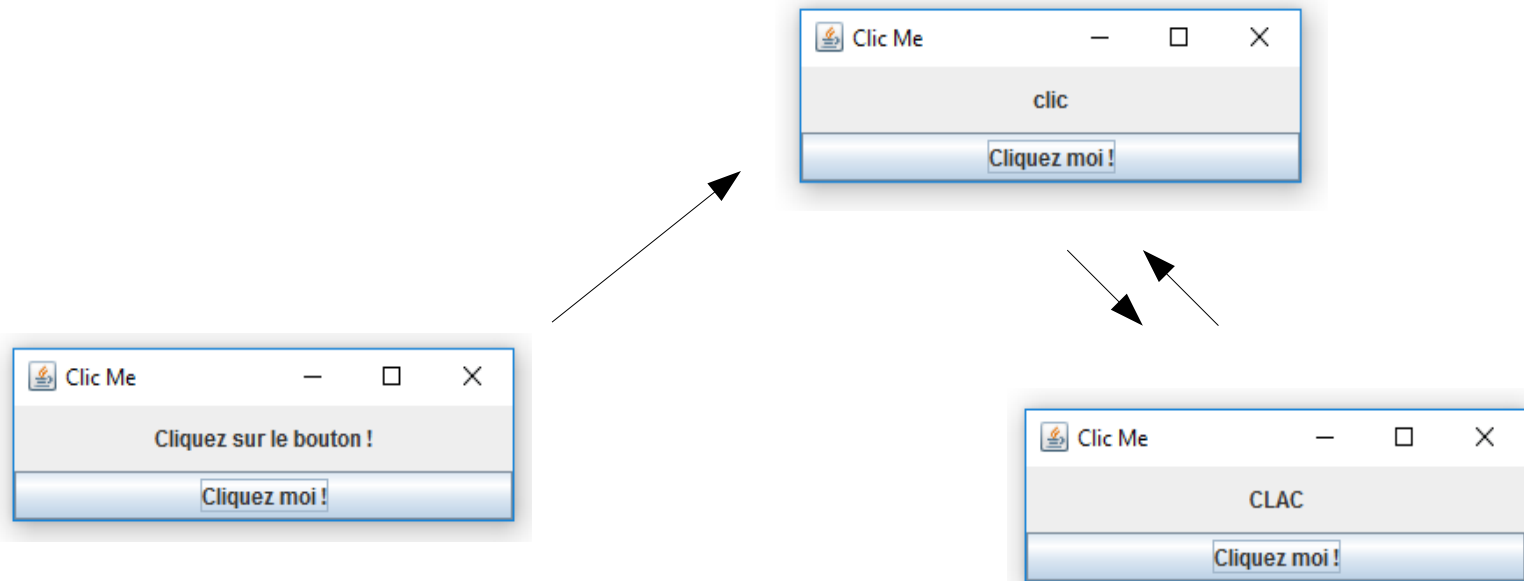
Implémentation d'un Handler

- 3 méthodes possibles :
 - Des classes/objets séparés
 - Pattern commande
 - Bcp de classes mais des classes simples
 - Une classe/objet centrale
 - Centralisation (moins de code)
 - Il faut distinguer les sources (garder un code clair)
 - Classes internes anonymes ou **Lambda**
 - Le Handler devient masqué
 - On le remplace par des méthodes claires
- JavaFX propose un grand nombre de méthodes pour définir les Handlers et la majorité des interfaces sont fonctionnelles
 - L'implémentation sous forme de Lambda est souvent privilégiée

`javafx.event.Event`

- Le déclenchement de chaque évènement provoque la création d'un objet dérivant de `Event`
 - Il est systématiquement reçu en paramètre de la méthode du Handler
 - Il contient toujours la source de l'évènement
 - Il contient un ensemble de caractéristiques propre à la nature de l'évènement (Key, Mouse, etc.)

Un Exemple



Cf. démo...

FXML

- Le langage FXML (dialecte XML) permet la définition déclarative des interfaces graphiques
- Il existe des éditeurs Graphiques (notamment Scene Builder)
- **Une vue FXML est associée à un objet Java (son « controller »).** Il porte :
 - Les initialisations algorithmiques
 - Les méthodes associées aux Handlers
 - Les relations entre les vues

Contrôleur de vue (*Initializable*)

- La vue fxml accède aux méthodes publiques du contrôleur ainsi qu'à ses méthodes et attributs privés si ils sont annotés par @FXML
 - *en général tout ce qui est à destination du fxml est privé et annoté*
- @FXML
 - Attributs private
 - Méthode `private void initialize()`
 - Autres méthodes privées