

# TP 5A - Génie Logiciel

## Patrons de conception: Stratégie

Nicolas Delanoue

(1 séance de 1h20)

Objectif de ce TP : Se familiariser avec le patron de conception (Design Pattern) Strategy. Amélioration d'un code existant (notion de *refactoring*).

### 1 Cahier des charges et prototype

On dispose d'une structure de données contenant des chaînes de caractères. On souhaite pouvoir afficher ces données sur la console ou dans une fenêtre graphique. A terme, on souhaiterait également envoyer ces données automatiquement par email ou les poster sur un serveur. Le choix du format d'affichage est configuré par le lanceur du programme (**Main**).

L'archive `TPStrategy_code_sujet.zip` donne un exemple de programme supportant deux formats d'affichage (voir la figure 1 pour la modélisation UML statique et la figure 2 pour la modélisation dynamique).

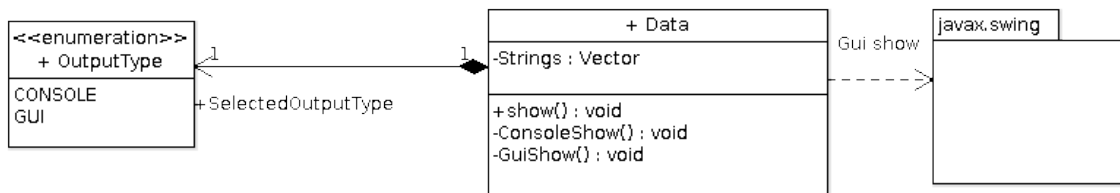


FIGURE 1 – Diagramme de classe UML du projet sujet.

### 2 Travail à réaliser

Ce programme présente deux inconvénients :

- L'ajout d'une autre méthode d'affichage impliquerait la modification et l'alourdissement de la classe `Data`.
- La classe `Data` présente une dépendance vers le package `javax.swing` (IHM) que l'on souhaite éliminer.

On souhaite améliorer la conception du programme par rapport à ces deux aspects. Pour cela, on propose d'implémenter le patron de conception *Strategy* pour gérer l'affichage (voir la figure 3 pour la modélisation générique de ce patron de conception).

Le travail à réaliser consiste à proposer une nouvelle modélisation du programme et à adapter le programme initial en conséquence.

On modélisera également le déroulement du `Main` au moyen du diagramme de séquence approprié.

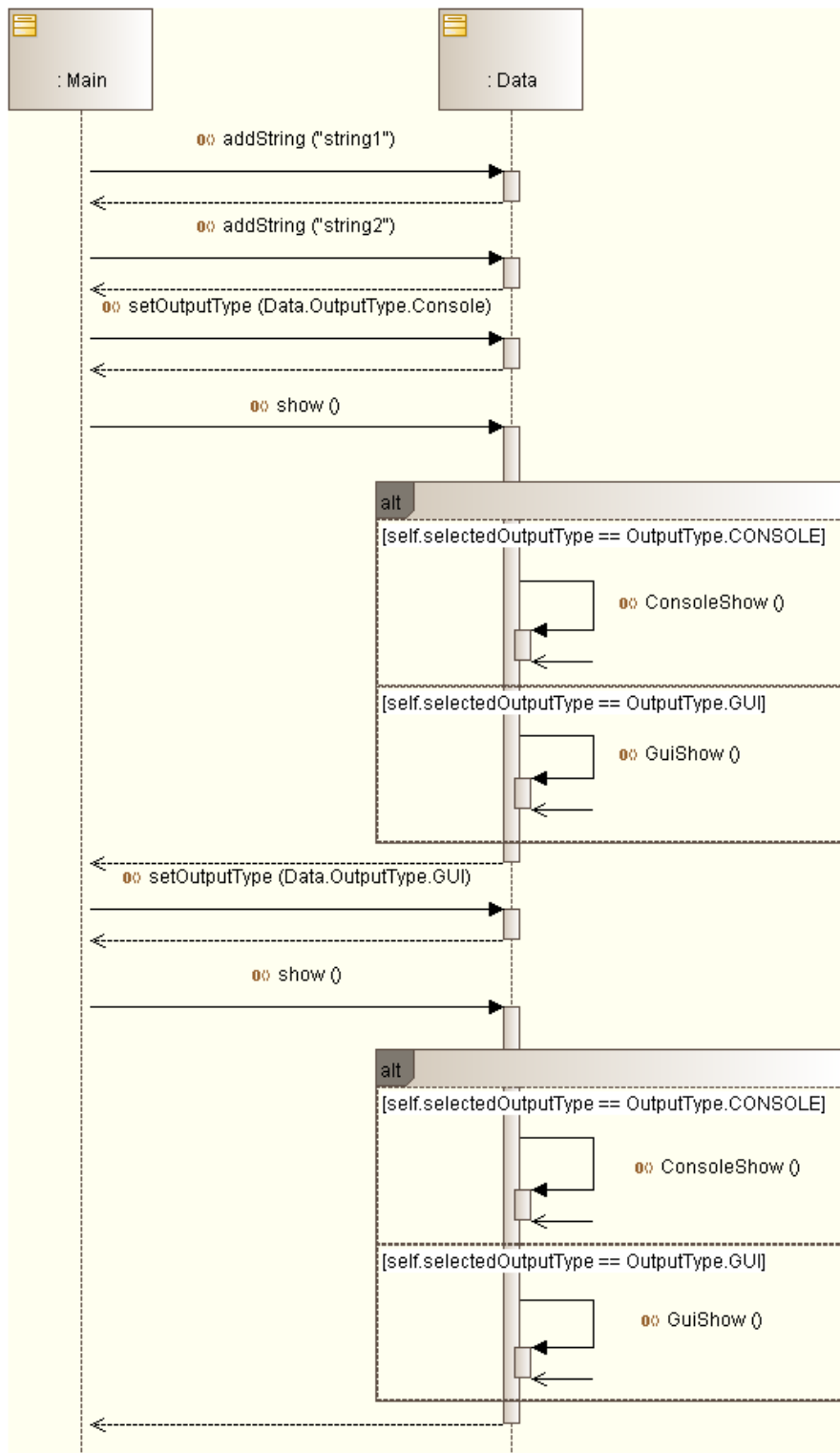


FIGURE 2 – Diagramme de séquence modélisant le déroulement du Main.

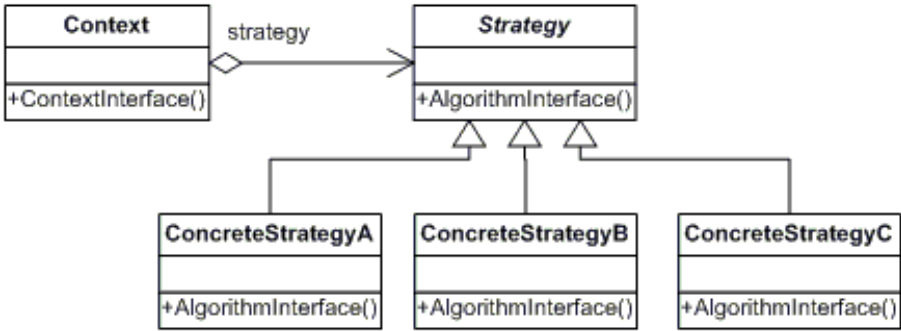


FIGURE 3 – Modélisation générique de Stratégie.