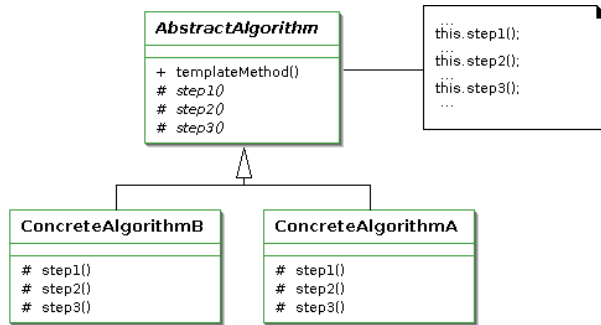


Design Pattern : template method

Intent

Define the skeleton of an algorithm in an operation, deferring some steps to subclasses. Template Method lets subclasses redefine certain steps of an algorithm without changing the algorithm's structure.

Structure



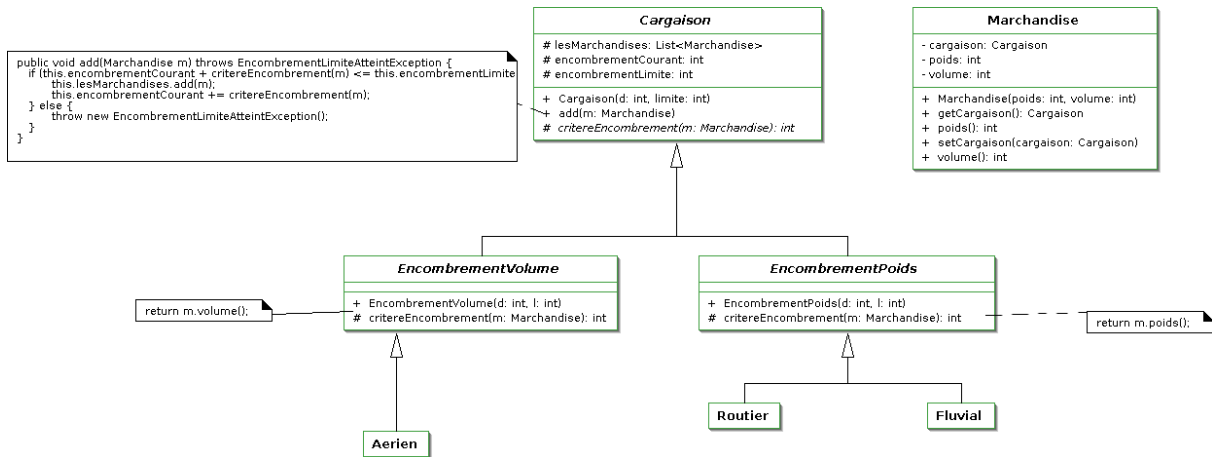
Eléments caractéristiques

- une méthode décrivant un algorithme qui fait appel à des méthodes abstraites de la classe, ces méthodes sont définies dans les sous-classes, permettant à ces sous-classes de faire varier le comportement de l’algorithme.

Exemples rencontrés

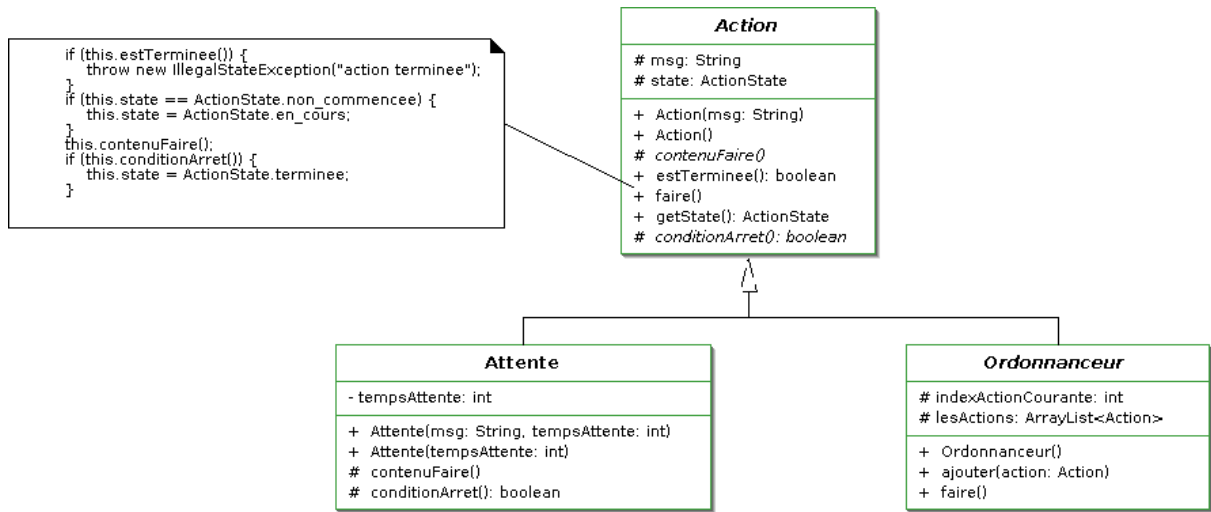
Cargaison : gestion de l’encombrement

Gestion de l’encombrement : la méthode add s’appuie sur la méthode abstraite critereEncombrement



Action : méthode faire()

Pour toutes les actions il faut gérer les passages de l’état *non_commencée* à *en_cours* puis à *terminée*. Cette partie de code est donc «factorisée». Le reste du traitement de faire() s’appuie ensuite sur les méthodes abstraites contenuFaire et conditionArret



Epreuves : ajouteResultat

On peut considérer que la méthode `ajouteResultat` satisfait ce design pattern car elle dépend de la méthode abstraite `creerResultat`.

