

INF4067: Patrons de Construction

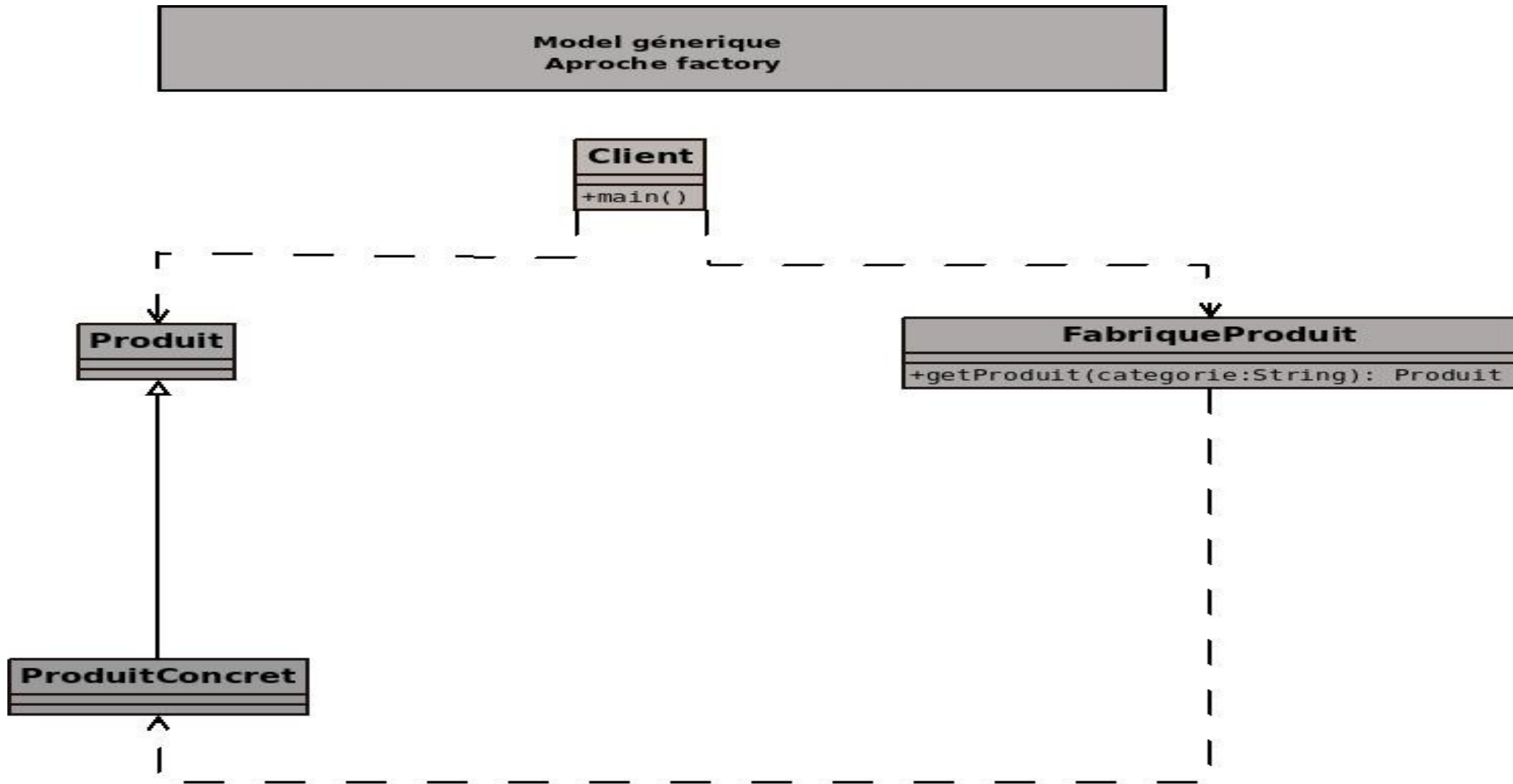
Exercice: on gère trois types de produits dans un système. Mais le programme qui se charge de manipuler ces produits ne connaît le type de produit à exécuter lors de l'exécution. On souhaite utiliser le pattern Fac tory **méthode** pour gérer cette situation

solution

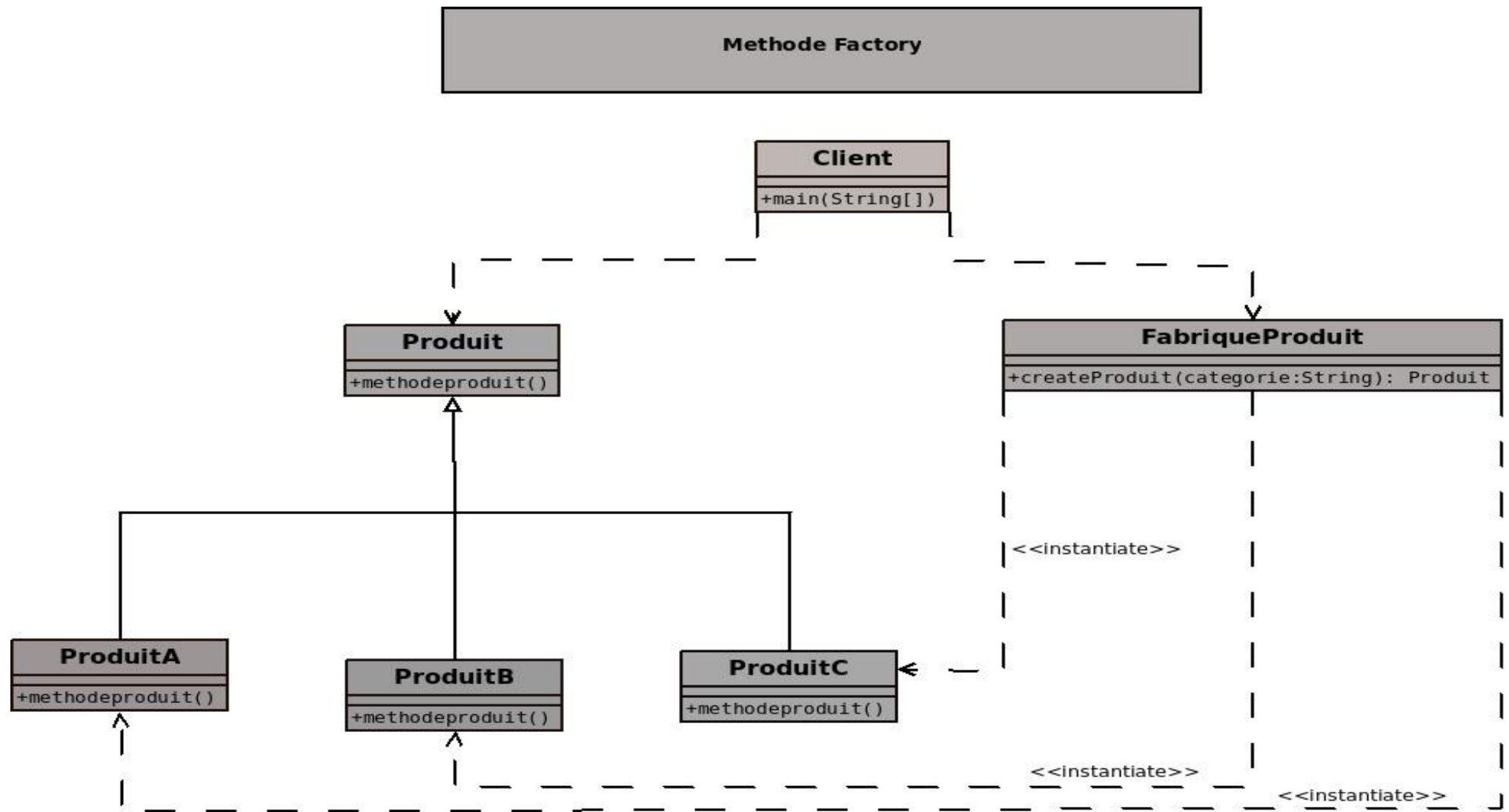
Pour résoudre ce problème , nous allons utiliser deux approches dont les models génériques sont :

Fac tory Méthode

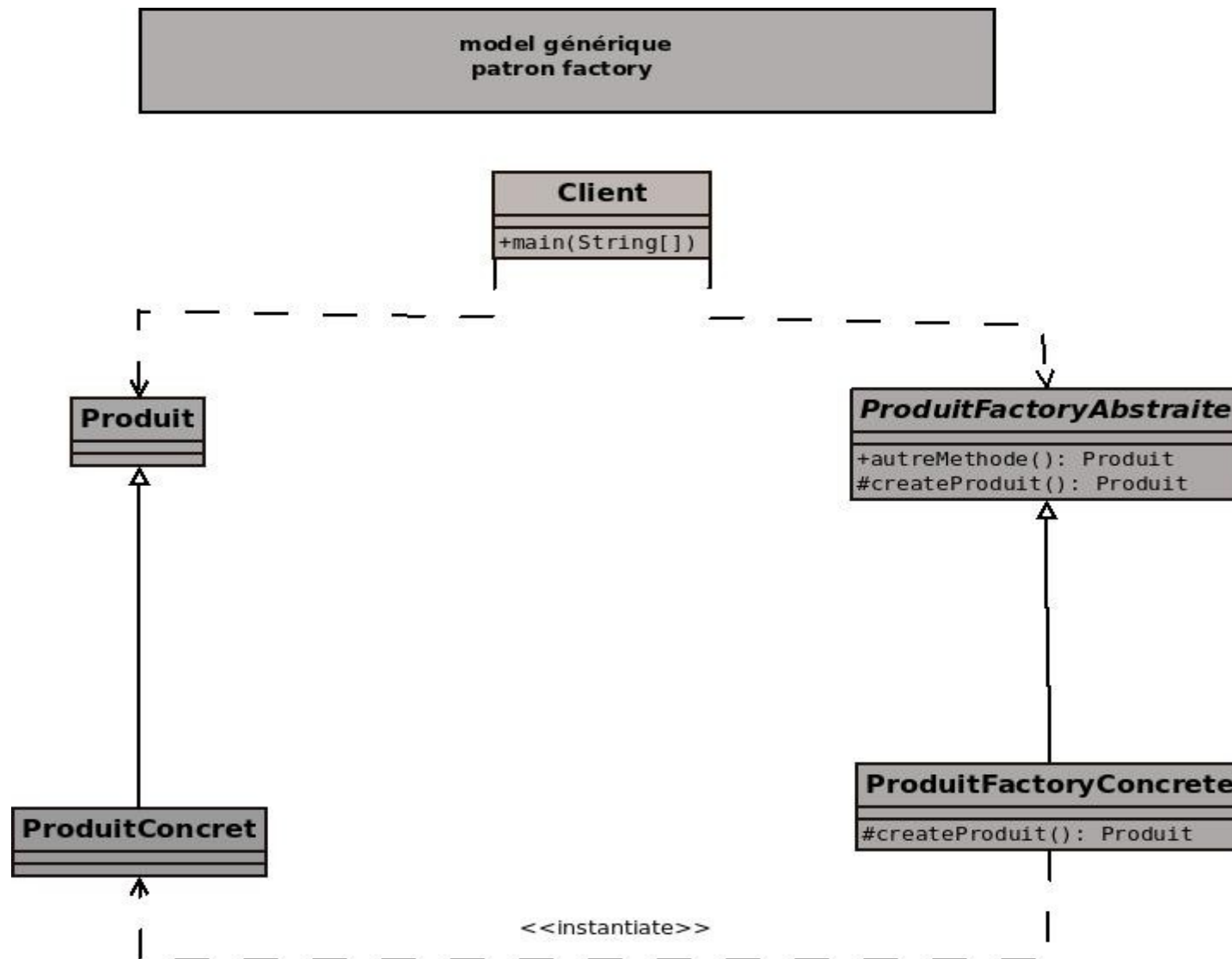
Factory Méthode



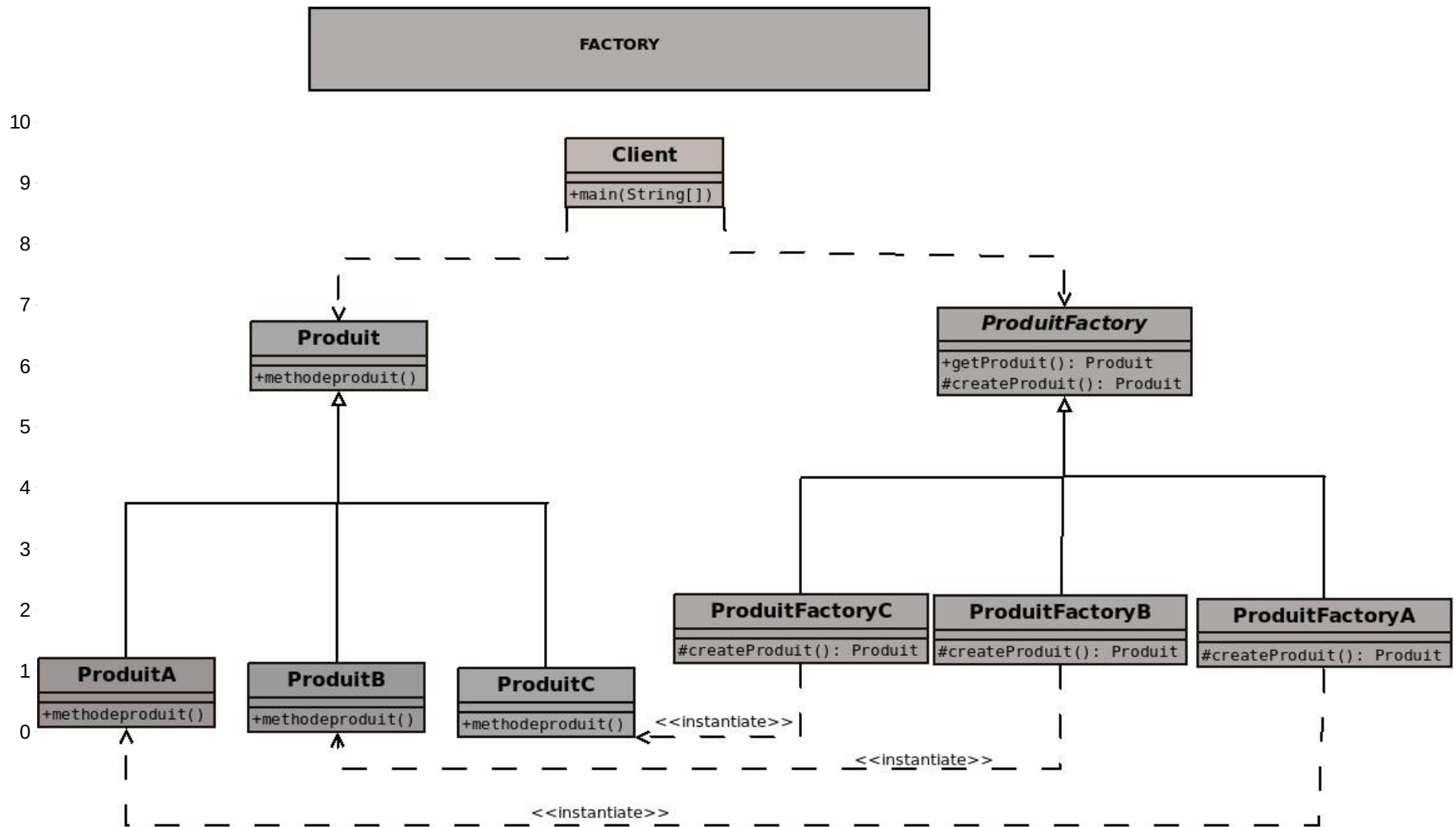
le model générique adapté à ceci pour la résolution du problème est :



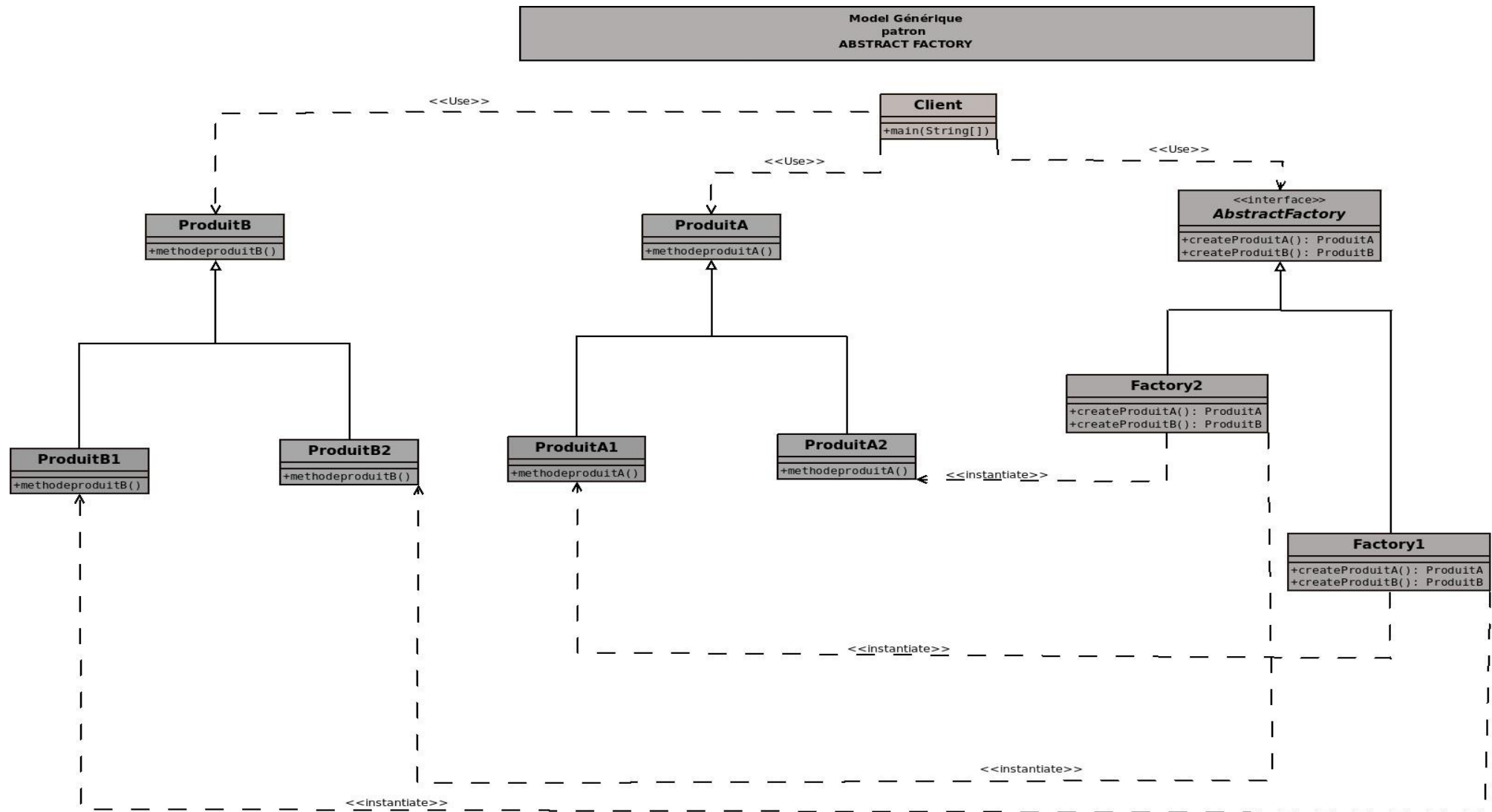
Deuxième Méthode Factory



le model générique adapté à ceci pour la résolution du problème est :

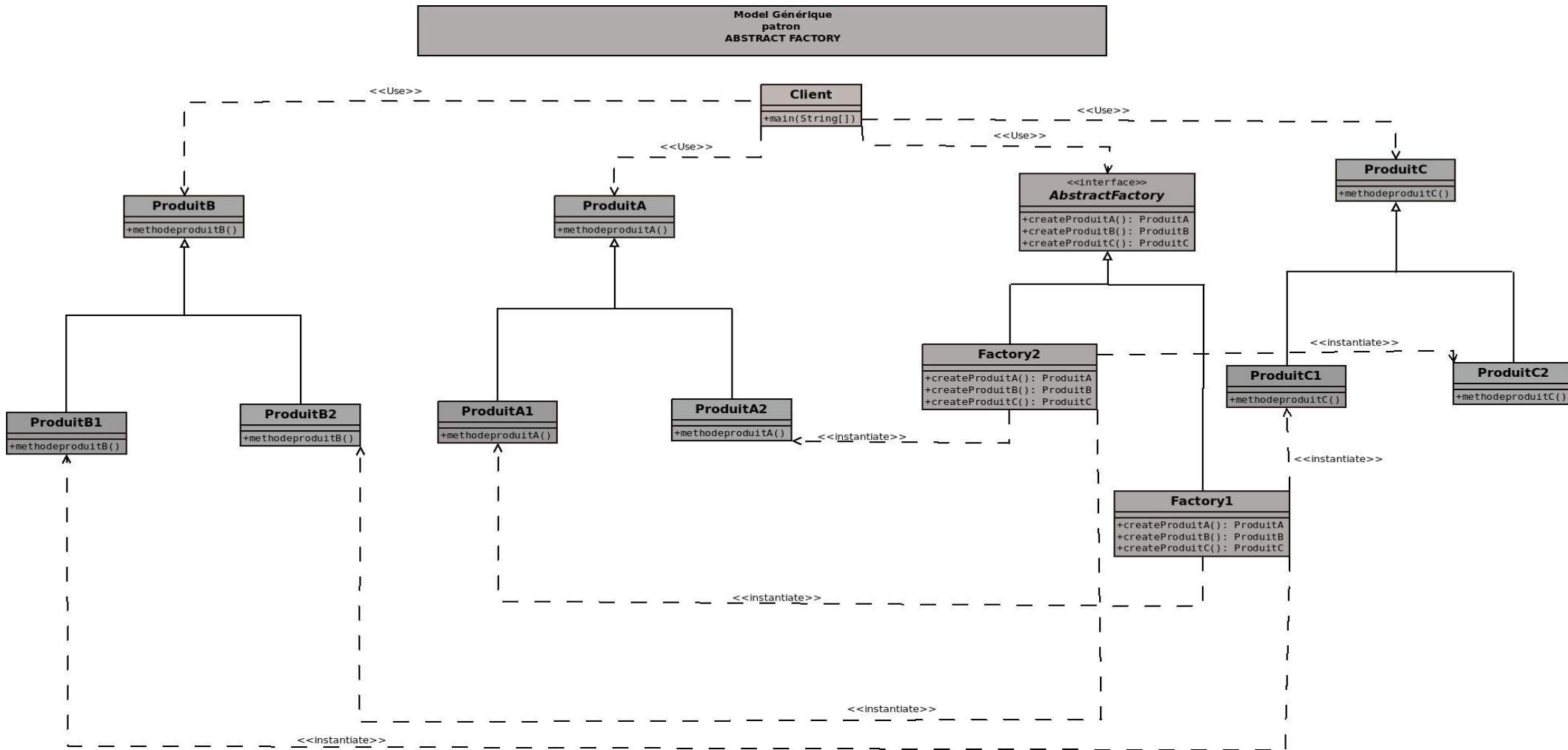


Model Générique Abstract Factory



Abstract factory

- Ajout d'un troisième type de produit



Le Patron Singleton

- Model générique de l'exemple vu en cours :

**Model Générique
Patron Singleton**

Singleton

```
-instance: Singleton  
+affiche(): void  
+moyenne(x:int,y:int): float  
+somme(x:int,y:int): int  
-Singleton()  
+getInstance(): Singleton
```


Le Patron Singleton: Modification

- Ajout des nouveaux constructeurs et nouvelles méthodes

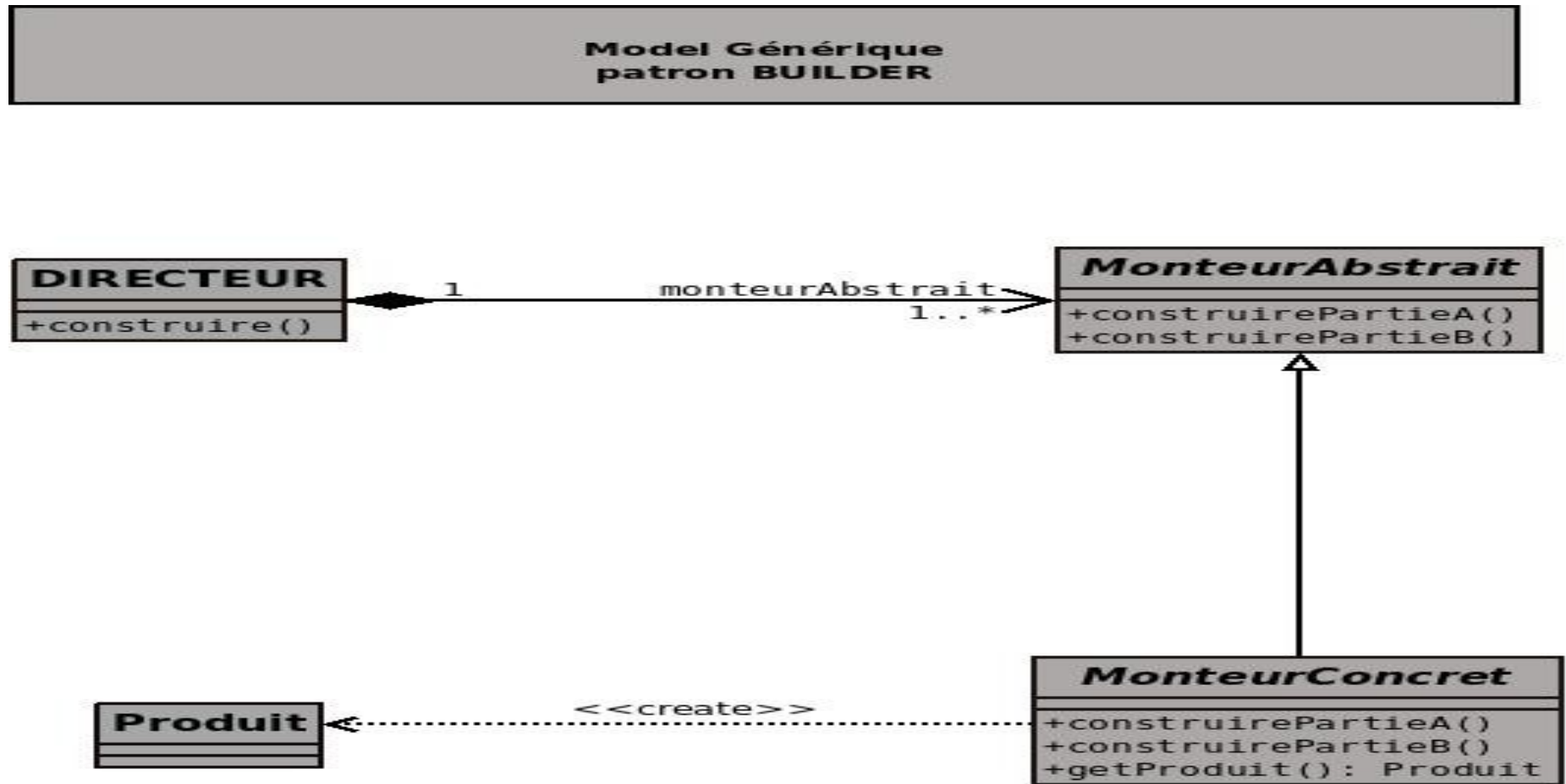
Patron Singleton

Arithmetique

```
-instance: Singleton
+x: int
+y: int
+nom: String
-Singleton()
-Singleton(x:int,y:int)
-Singleton(x:int,y:int,nom:String)
+getInstance(): Singleton
+moyenne(x:int,y:int): float
+somme(x:int,y:int): int
+Produit(x:int,y:int): int
+affiche(): void
```

Le Patron Builder

- Ayant pour but de construire un objet complexes en construisant ses parties et en les rassemblant, le model générique est le suivant :



Patron Buidel:Exemple

On souhaite fabriquer des pizza. On a 2 types de pizza : la pizza reine et le pizza piquante,

Pour fabriquer une pizza : on prépare la pate, on prépare et ajoute une sauce et on prépare et ajoute une garniture.

– pour la pizza reine, on utilise : pate = « croisée », sauce = « douce » et garniture = « jambon et champignon »

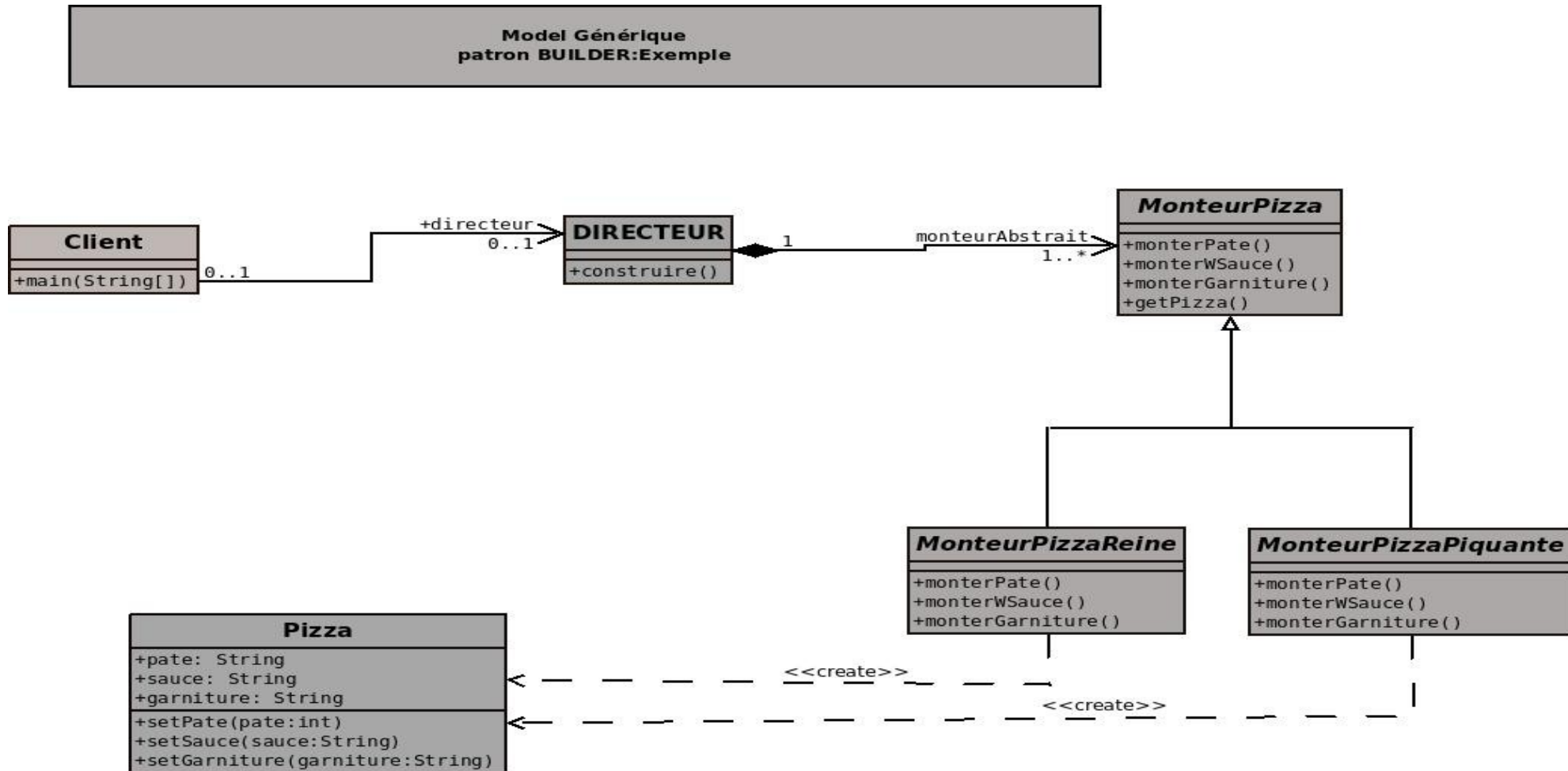
– pour la pizza piquante, on utilise : pate = « feuilletée », sauce = « piquante » et garniture = « pepperoni + salami »

On veut utiliser le design pattern Builder pour faciliter la fabrication des pizza.

1. Proposer une modélisation sous forme de diagramme de classe
2. Proposer le code source correspondant.

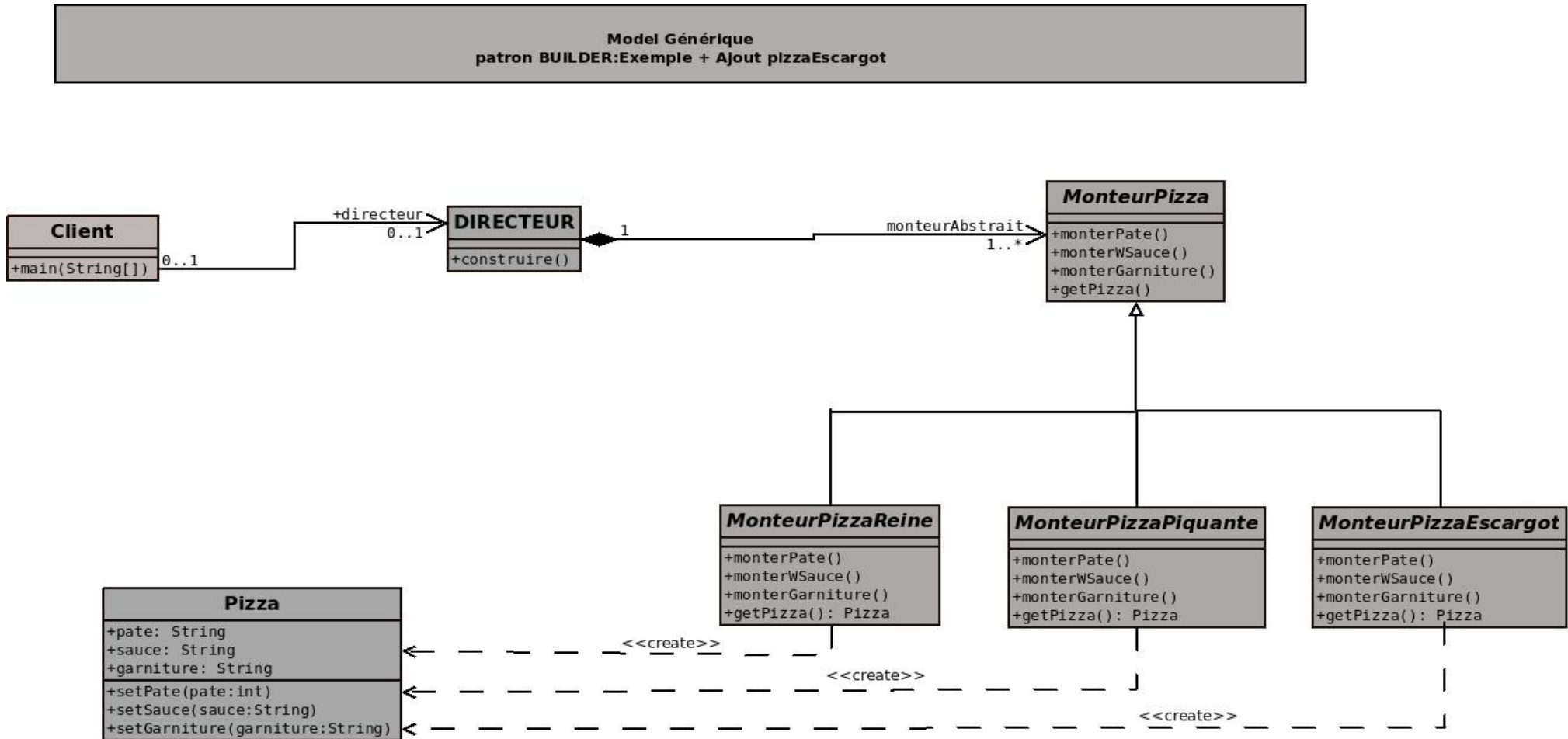
Model générique

- Le model générique pour résoudre ce problème est le suivant :



Ajout d'une Autre pizza

En ajoutant une autre pizza à l'exemple précédent, on a le model générique suivant



Liens d'accès aux dépôts Git

liens d'accès aux dépôts Git Hub :

Patron Factory:

<https://github.com/INF4067/FactoryAbstraite>

<https://github.com/INF4067/Factory>

Patron Abstract Factory :

<https://github.com/INF4067/abstractFactory>

<https://github.com/INF4067/patronAbstractfactory>

Patron Singleton :

<https://github.com/INF4067/Singleton>

<https://github.com/INF4067/PatronSingleton>