

TD Mise en œuvre de Design Patterns

Partie I - Création de cartes de paiement

Création en fonction du client

Les clients d'une banque sont classés en 2 catégories :

- Ceux qui ont droit au crédit,
- Ceux qui n'y ont pas droit

Lors de la demande d'une carte de paiement, les premiers reçoivent une carte de crédit à débit différé sur le compte alors que les seconds peuvent seulement avoir une carte de débit cad à débit immédiat sur le compte.

1) Quel pattern de conception permet-il de modéliser la création de la carte de paiement en fonction du client ?

2) Modélisez son utilisation par un diagramme de classes

Création à l'aide d'une fabrique

De plus, il existe 2 modèles de cartes de débit et de crédit, à savoir les cartes Visa et les cartes MasterCard

3) Modélisez à l'aide d'un diagramme de classes, la création d'une carte de paiement en fonction de sa famille (de crédit ou de débit) en utilisant le pattern Abstract Factory.

Autorisation des cartes de paiement

Lors d'un achat avec une carte de paiement, une autorisation doit être accordée ; Si la carte est une carte de débit (débit immédiat), l'autorisation est accordée si le solde du compte sur lequel la carte est débitée est suffisant. Si la carte est une carte de crédit (débit différé), l'autorisation est accordée si le montant mensuel des dépenses n'a pas dépassé le plafond.

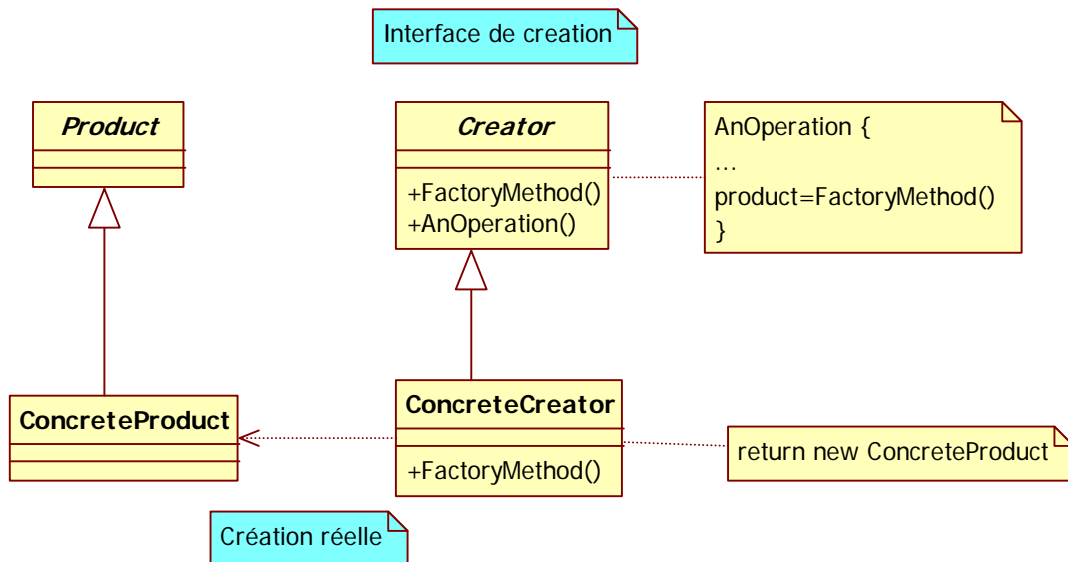
4) Quel pattern de conception permet-il de modéliser l'autorisation lors d'un achat avec une carte de paiement en fonction du modèle de carte.

5) Modélisez son utilisation par un diagramme de classes.

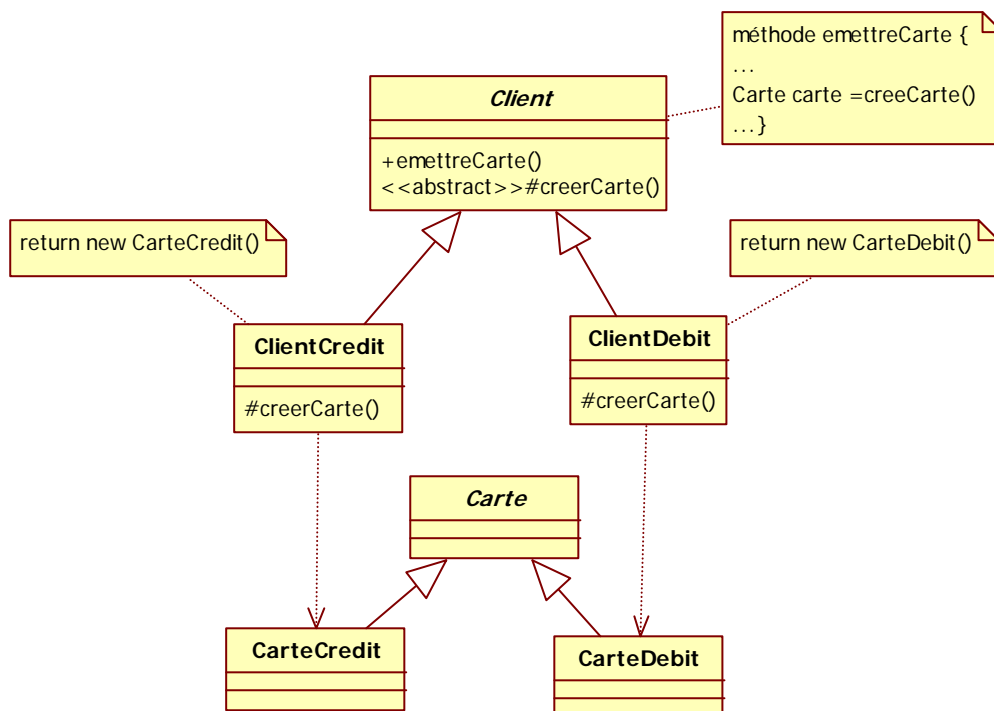
Correction - Création de cartes de paiement

Création en fonction du client

Le pattern *Factory Method* introduit une méthode abstraite de création d'un objet et reporte sur les sous-classes la création effective. L'intérêt est la possibilité d'ajouter facilement d'autres classes concrètes (ici il y en a 2).

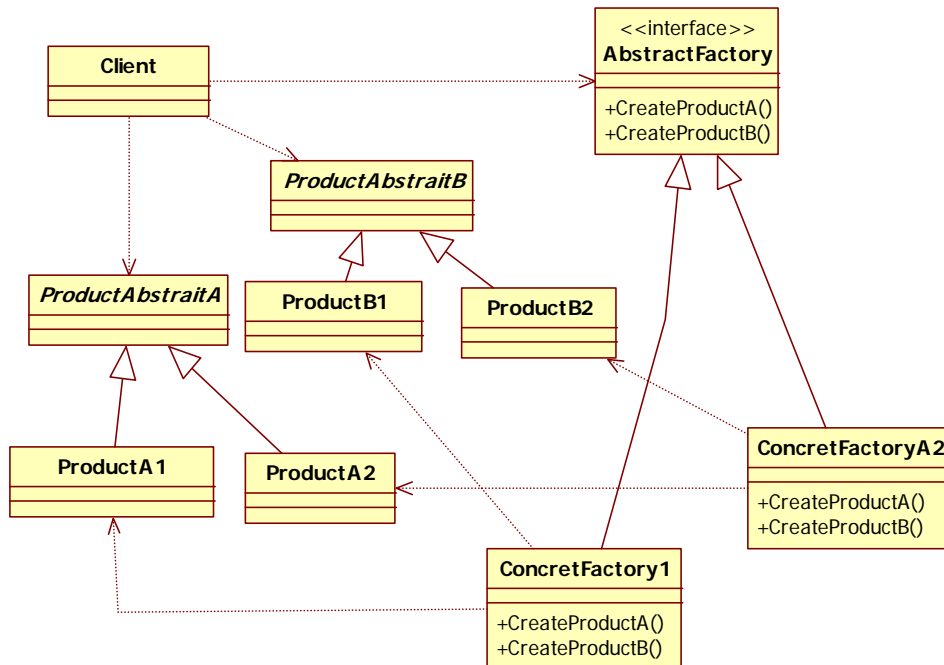


Application

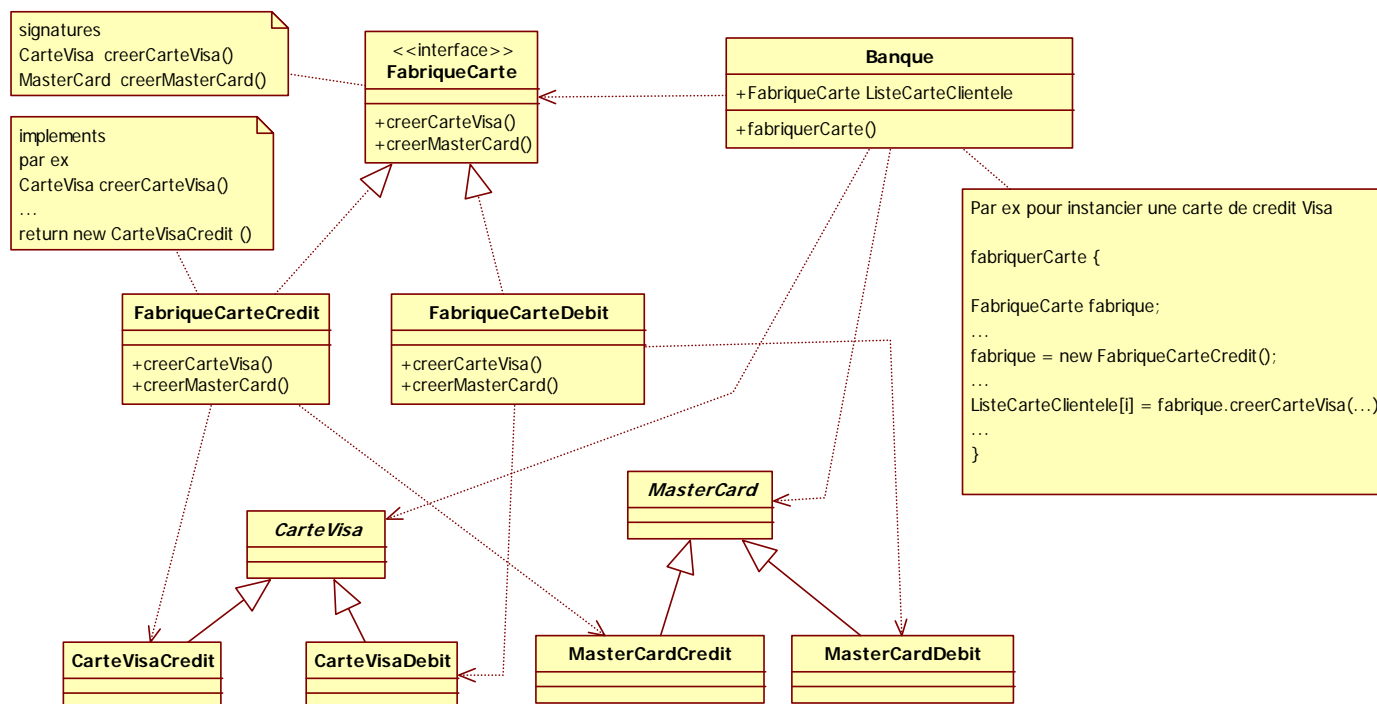


Création à l'aide d'une fabrique

Le pattern *Abstract Factory* permet de créer une collection d'objets de classes interdépendantes. Ici par ex la liste de toutes les cartes de la banque comporte indifféremment des références sur l'un des objets concrets de CarteVisaCredit, CarteVisaCredit, MasterCardDebit, MasterCardCredit, en laissant



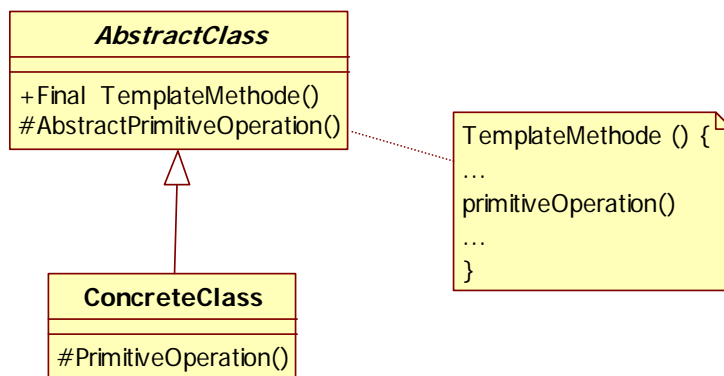
Application



Remarque : on peut de plus appliquer le pattern Singleton respectivement à FabriqueCarteCredit et FabriqueCarteDebit.

Autorisation des cartes de paiement

Le pattern *Template Method* permet de reporter dans des sous-classes une partie variable d'un algorithme. Ici, la demande d'autorisation est commune aux cartes de crédit et aux cartes de débit mais s'implante différemment.



Application

