

Patrons de Conception

-

Factory Method

Simon Urli
urli@i3s.unice.fr

Master I MIAGE
2014-2015

Objectifs

“ Définir un point d’entrée unique pour la création d’un objet concret respectant une définition homogène de création de ce type d’objet.”

Classification : patron de création

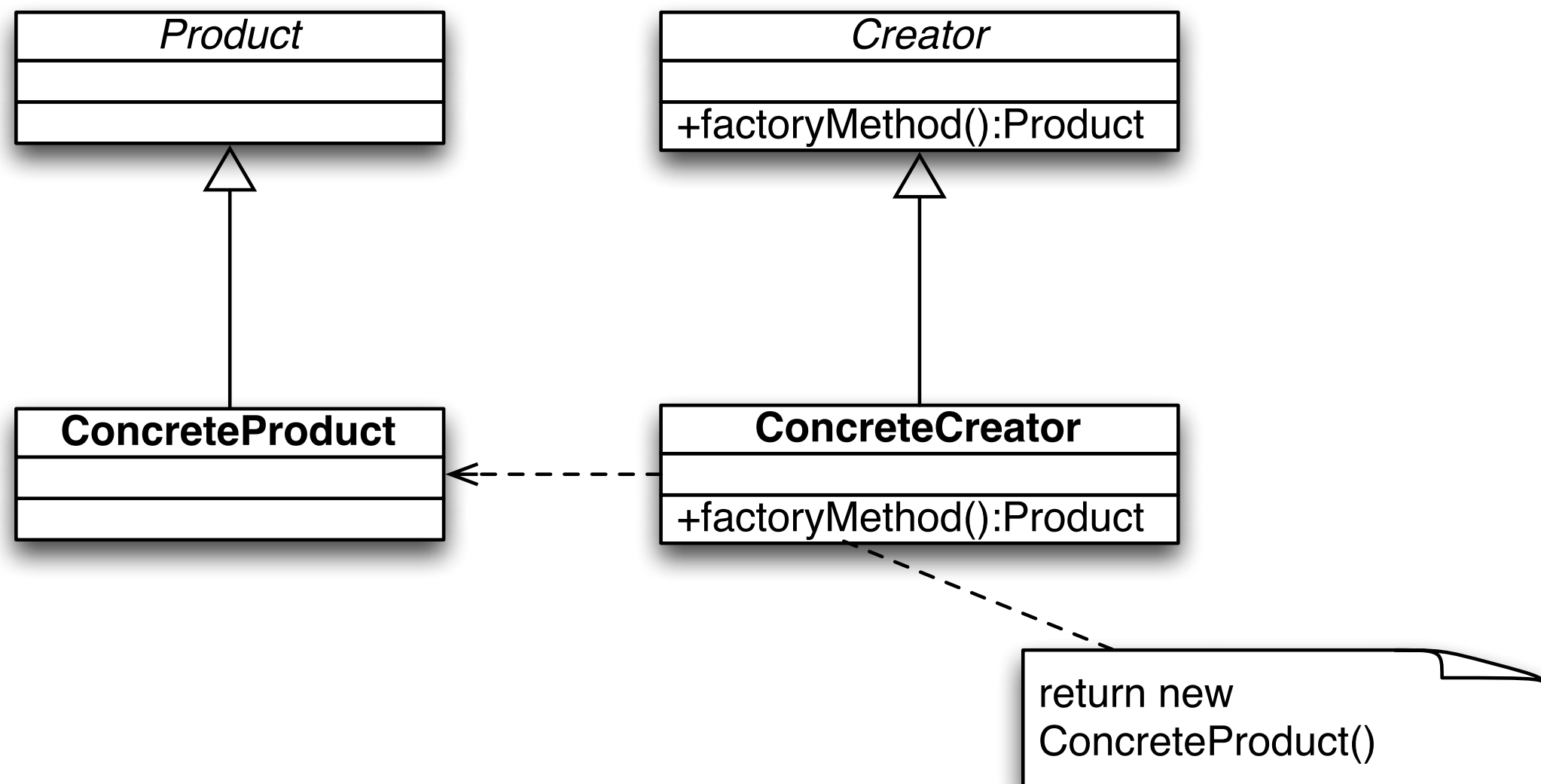
Exemple



Applications

- Besoin de gérer un unique point de création d'un objet (ajout de logging, etc)
- Besoin de gérer de manière homogène la création d'une famille d'objet
- Pas de maîtrise sur l'implémentation des objets, mais besoin de les utiliser

Structure



Implémentation

- **Product** : définit l'interface des objets créés par la factory
- **ConcreteProduct** : produit concret implémentant l'interface Product
- **Creator** : déclare la méthode de fabrication
- **ConcreteCreator** : implémente la méthode de fabrication pour un produit concret particulier

Exemple 1/2

```
public interface IWidget {  
    public Point2D getPosition();  
}  
  
public interface IWidgetFactory {  
    public IWidget createWidget(Point2D  
position);  
}
```

Exemple 2/2

```
public class Clock implements IWidget {  
    private Point2D position;  
    private int decalage_utc;  
    public Clock(Point2D position, int d) {  
        this.position = position;  
        this.decalage_utc = d;  
    }  
    public Point2D getPosition() {  
        return this.position;  
    }  
}  
  
public class ClockFactory implements IWidgetFactory {  
    public IWidget createWidget(Point2D position) {  
        return new Clock(position, 0);  
    }  
}
```


Exercice

- Sachant qu'une cohorte est une unité tactique de l'armée romaine constituée d'un seul type de soldat, et que l'armée romaine contenait plusieurs cohortes, modélisez (ou implémentez) les classes permettant de «créer» l'armée romaine.