Méthodes d'analyse et de conception

Partie I : UML (*Unified Modeling Language*)

Chapitre 7 : Les diagrammes de composants

2ème Année Génie Informatique / Semestre 3

ENSA Khouribga

Pr. DARGHAM ABDELMAJID

Année académique : 2018/2019

Le diagramme de composants

- **Rôle** :
  - Le **diagramme de composant** permet de représenter les **composants logiciels** d'un système ainsi que les **liens qui existent entre ces composants**.
  - Il y a deux catégories de **composants logiciels** :
    - **Les composants métiers** qui sont propres à une entreprise.
    - **Les composants disponibles sur le marché** (par exemple **EJB**, **CORBA**, **.NET**, **ActiveX**).

Le diagramme de composants

- **Composant** :
  - Un **composant** est une **unité logicielle autonome**.
  - Chaque **composant** est assimilé à un **élément exécutable du système**. Il est caractérisé par :
    - Un **nom**;
    - Une **spécification externe** sous forme soit d'une ou plusieurs **interfaces requises**, soit d'une ou plusieurs **interfaces fournies**;
    - Un **port de connexion**.

Le diagramme de composants

- **Quelques exemples de composants** :
  - **Fichiers sources (.jar)**.
  - **Contrôles ActiveX**.
  - **JavaBeans**.
  - **EJB**.
  - **Java Servelts**.
  - **JSP (Java Server Pages)**.

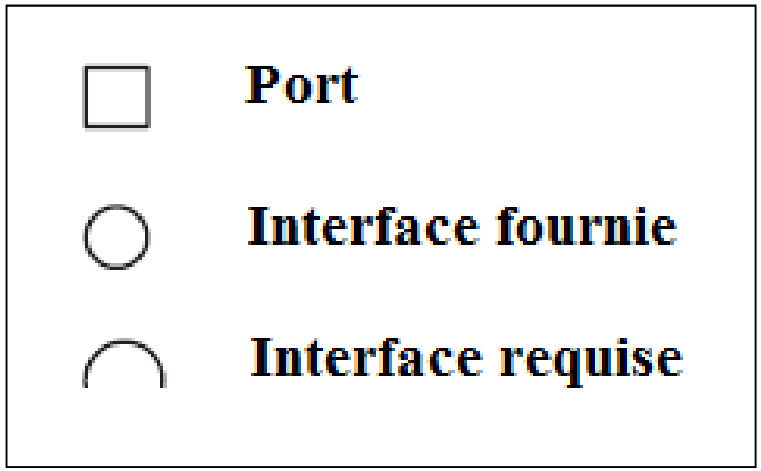
Le diagramme de composants

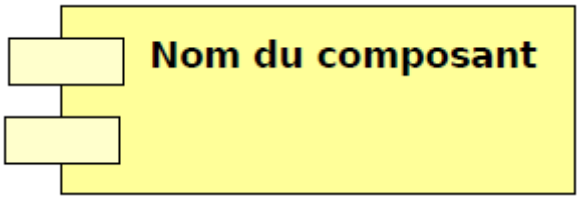
- **Composant** :
  - Le **port d'un composant** représente le **point de connexion** entre le **composant** et une **interface**.
  - L'identification d'un port permet d'**assurer une certaine indépendance** entre le **composant** et son **environnement extérieur**.

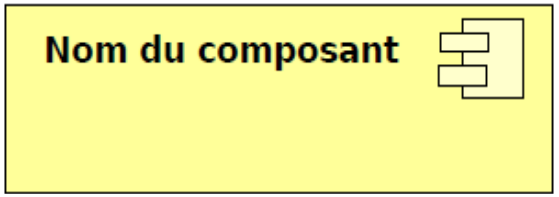
Le diagramme de composants

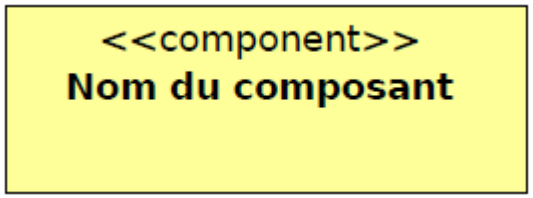
- **Types d'interfaces de composants** :
  - Une **interface fournie** spécifie les **opérations qu'un composant doit exécuter**.
  - Une **interface requise** décrit les **opérations** que **d'autres composants fournissent** pour que ce **composant fonctionne correctement** dans un environnement particulier.

Le diagramme de composants









Formalisme UML pour décrire les composants

Le diagramme de composants

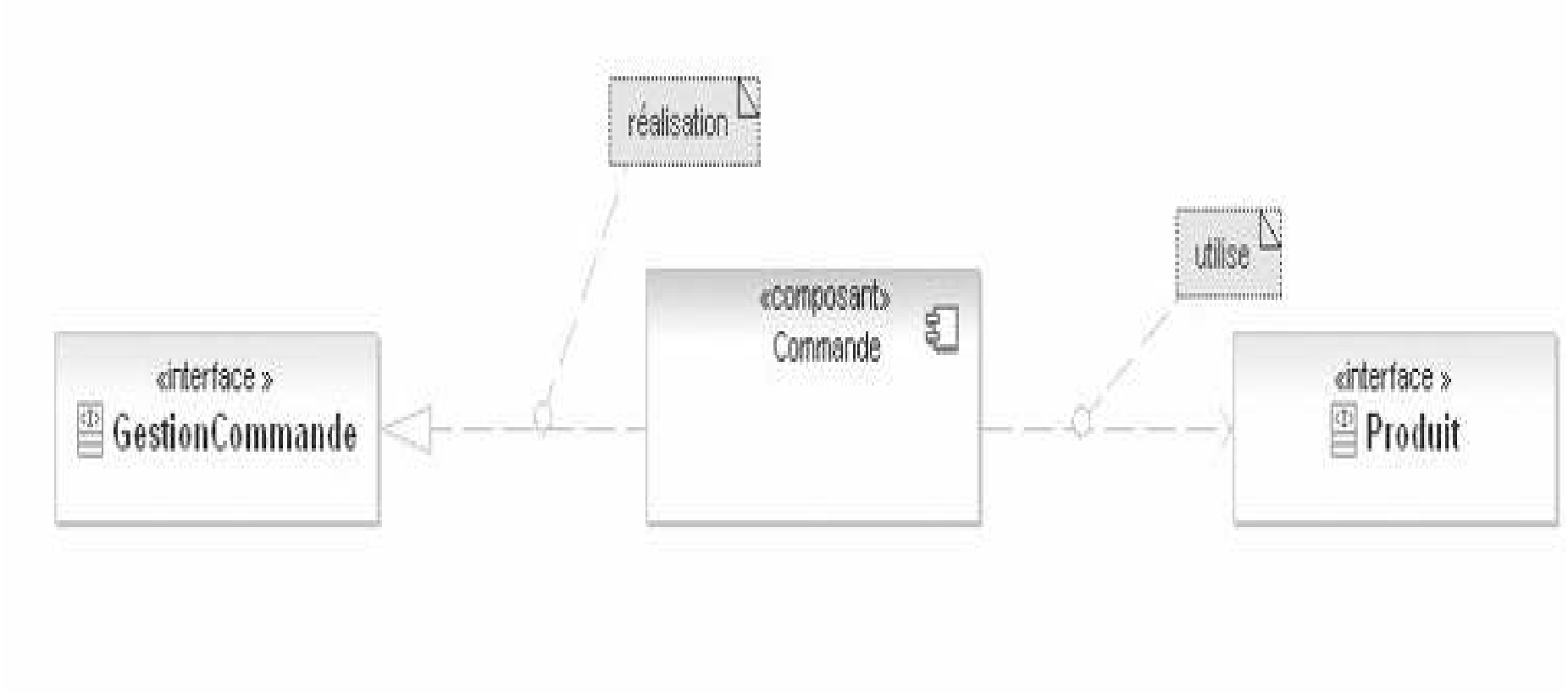
- **Représentation « boîte noire » :**
  - C'est une **vue externe du composant** qui présente ses **interfaces fournies et requises** sans entrer dans le détail de l'implémentation du composant.
  - Une **boîte noire** peut se représenter de différentes manières :
    - **Connecteur d'assemblage**
    - **Connecteur d'interfaces**
    - **Compartiment**

Le diagramme de composants

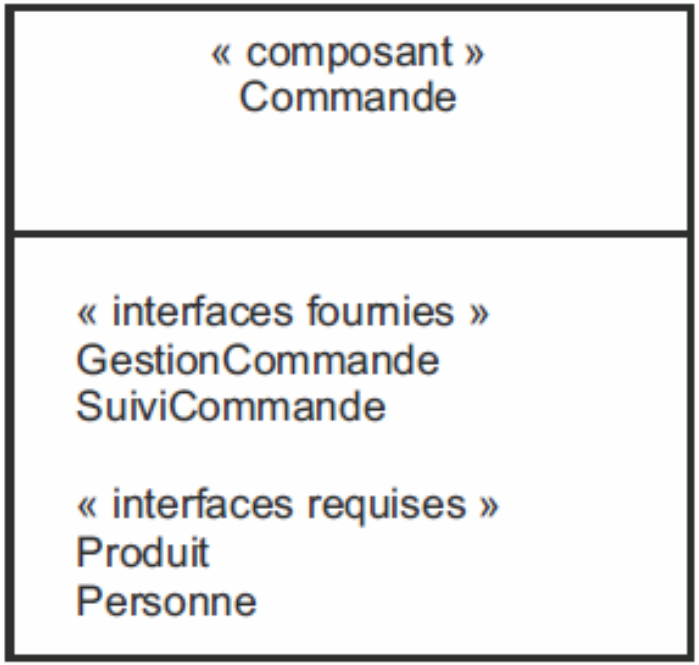
- **Connecteur d'assemblage :**
  - Une **interface fournie** se représente à l'aide d'un **trait** et d'un **petit cercle** et une **interface requise** à l'aide d'un **trait** et d'un **demi-cercle**.

- **Connecteur d'interfaces :**
  - Dans cette méthode de représentation, on utilise les **dépendances d'interfaces** « **utilise** » et « **réalise** » :
  - Pour une **interface fournie**, c'est une **relation de réalisation** partant du **composant** et allant vers **l'interface**.
  - Pour une **interface requise**, c'est une dépendance avec le mot-clé « **utilise** » partant du **composant** et allant vers **l'interface**.

Le diagramme de composants



- **Compartiment :**
  - Consiste à décrire sous **forme textuelle** les **interfaces fournies et requises** à l'intérieur d'un second **compartiment**.



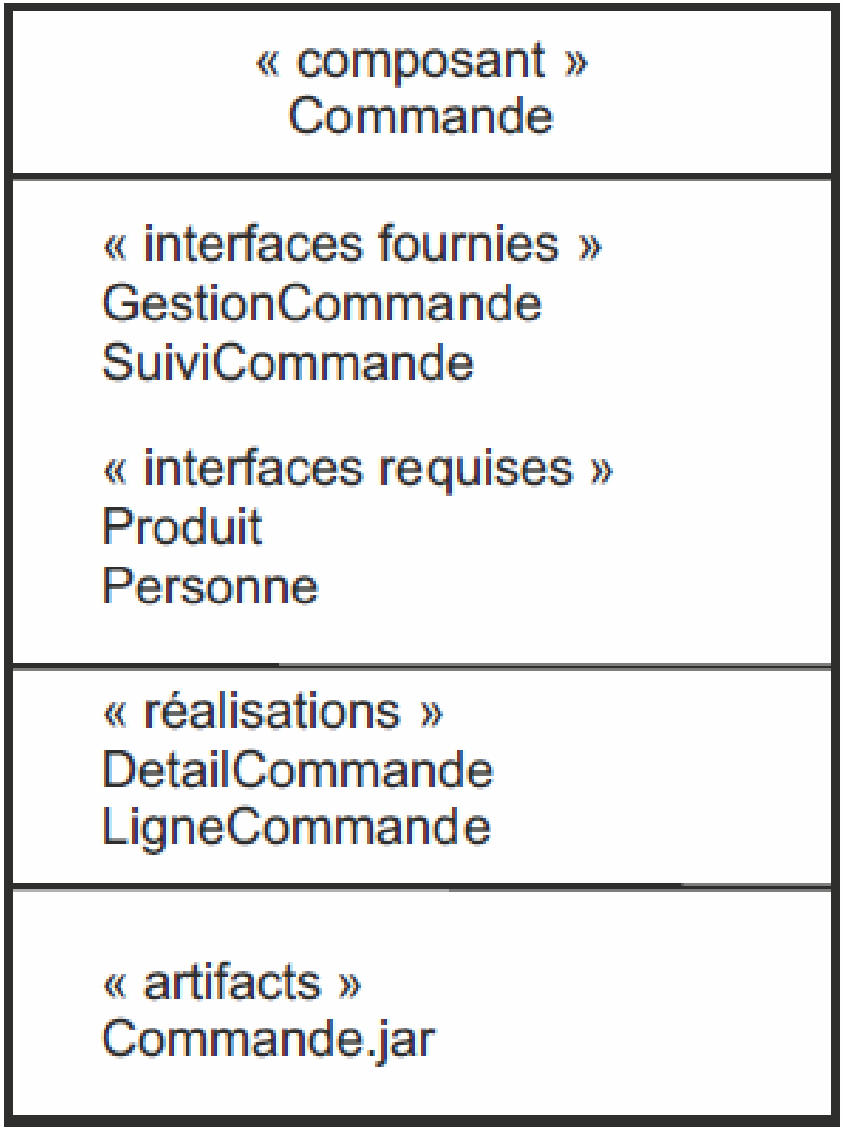
Le diagramme de composants

- **Représentation « boîte blanche » :**
  - C'est une **vue interne du composant** qui décrit son **implémentation** à l'aide de **classificateurs** (**classes**, **autres composants**) qui le composent.
  - Une **boîte blanche** peut se représenter de différentes manières :
    - **Compartiment**
    - **Dépendance**
    - **Ports et connecteurs**

Le diagramme de composants

- **Compartiment :**
  - Consiste à décrire sous **forme textuelle** :
    - Les **interfaces fournies et requises** à l'intérieur d'un **compartiment**.
    - Les **classificateurs** (**classes**, **autres composants**) dans un autre **compartiment**.
    - Les **artefacts** (élément logiciel, comme : **jar** et **dll**) qui représentent physiquement le composant dans un dernier **compartiment**.

Le diagramme de composants

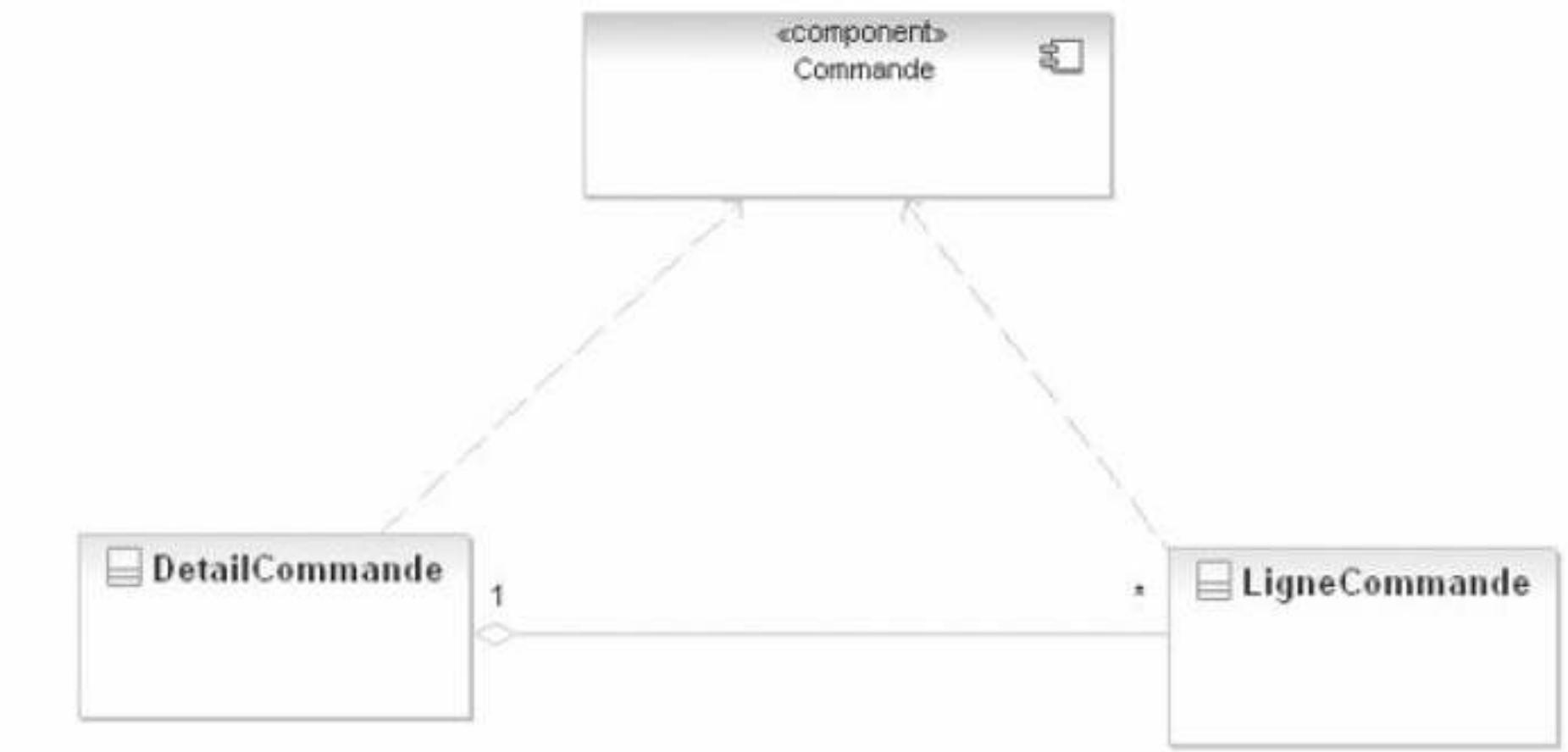


Le diagramme de composants

- **Dépendance :**
  - Une autre représentation interne du composant peut être aussi utilisée en ayant recours aux **dépendances**.
  - Ainsi, les **classificateurs** qui composent le **composant** sont reliés à celui-ci par une **relation de dépendance**.
  - Les **relations** entre les classificateurs (**association**, **composition**, **agrégation**) sont aussi modélisées.

- **Dépendance :**
  - Néanmoins, si elles sont trop complexes, elles peuvent être représentées sur un **diagramme de classe** relié au composant par une **note**.

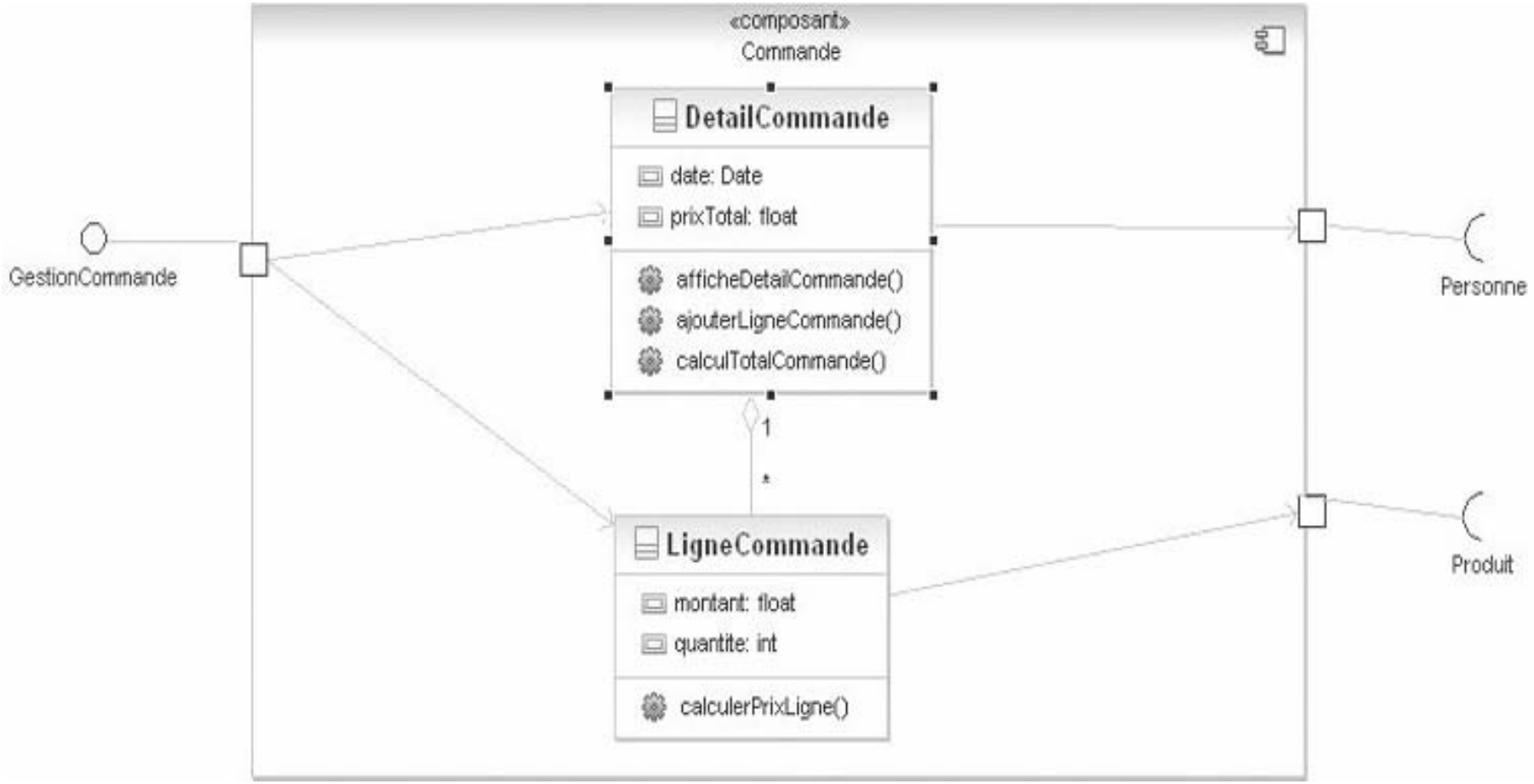
Le diagramme de composants



Le diagramme de composants

- **Ports & connecteurs :**
  - Le **port** est représenté par un **petit carré** sur le **composant**.
  - Les **connecteurs** permettent de relier les **ports** aux **classificateurs**.
  - Ils sont représentés par une **association navigable** et indiquent que toute information arrivée au **port** est transmise au **classificateur**.

Le diagramme de composants



Le diagramme de composants

- **Explication de l'exemple :**
  - Le **composant** « **Commande** » est constitué de deux **classes** (**classificateur**) reliées par une agrégation : « **DetailCommande** » et « **LigneCommande** ».
  - L'**interface fournie** « **GestionCommande** » est accessible de l'extérieur via un **port** et permet d'accéder via les connecteurs aux opérations des deux classes « **DetailCommande** » et « **LigneCommande** » (***ajouterLigneCommande***, ***calculTotal-Commande***, ***calculPrixLigne***).

Le diagramme de composants

- **Explication de l'exemple :**
  - L'**interface requise** « **Personne** » est nécessaire pour l'affichage du détail de la commande et est accessible via un port du composant Commande.
  - L'**interface requise** « **Produit** » est nécessaire pour le calcul du prix de la ligne de commande et est accessible via un **port** du composant « **Commande** ».
  - Les **connecteurs** permettent de relier les **ports** aux **classificateurs**.