

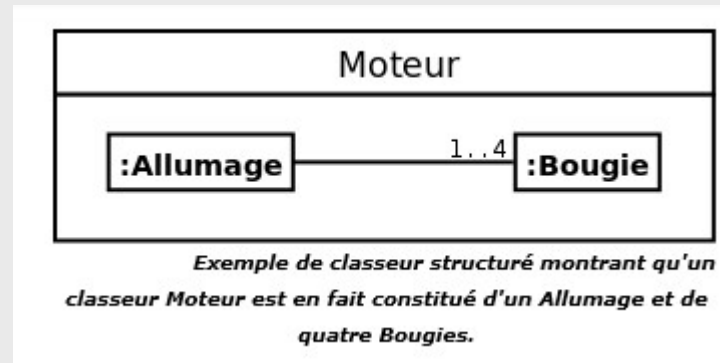
# Diagrammes d'interaction

# Diagrammes d'interaction

- 
- 
- Les diagrammes d'interaction permettent d'offrir une vue plus holistique **du comportement** d'un jeu d'objets.
- Les diagrammes d'interaction permettent d'établir un lien entre les diagrammes de cas d'utilisation et les diagrammes de classes : ils montrent comment des objets communiquent pour réaliser une certaine fonctionnalité. Ils apportent ainsi un aspect dynamique à la modélisation du système.
- Pour produire un diagramme d'interaction, il faut focaliser son attention sur un sous-ensemble d'éléments du système et étudier leur façon d'interagir pour décrire un comportement particulier.
- ❖ UML permet de décrire un comportement limité à un contexte précis de deux façons : dans le cadre d'un classeur structuré ou dans celui d'une collaboration.

# Diagrammes d'interaction

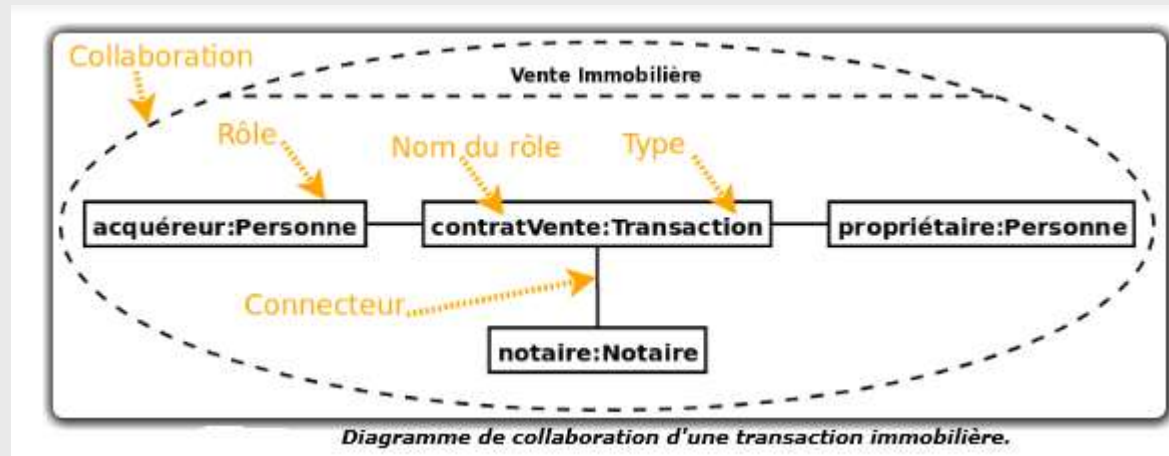
- Classeur structuré :



- Les classes découvertes au moment de l'analyse (celles qui figurent dans le diagramme de classes) ne sont parfois pas assez détaillées pour pouvoir être implémentées par des développeurs. UML propose de partir des classeurs découverts au moment de l'analyse (tels que les classes, mais aussi les sous-systèmes, les cas d'utilisation...) et de les décomposer en éléments suffisamment fins pour permettre leur implémentation. **Les classeurs ainsi décomposés s'appellent des classeurs structurés.**
- Un classeur structuré est donc la description de la structure d'implémentation interne d'une classe.

# Diagrammes d'interaction

## ■ Collaboration :



- Une collaboration permet de décrire la mise en œuvre d'une fonctionnalité par un jeu de participants. Un rôle est la description d'un participant. Contrairement aux paquetages et aux classeurs structurés, une collaboration ne détient pas les instances liées à ses rôles. Les instances existent avant l'établissement d'une instance de la collaboration, mais la collaboration les rassemble et précise des connecteurs entre elles. Une collaboration peut donc traverser plusieurs niveaux d'un système et un même élément peut apparaître dans plusieurs collaborations.
- Par exemple, pour implémenter un cas d'utilisation, il faut utiliser un ensemble de classes, et d'autres éléments, fonctionnant ensemble pour réaliser le comportement de ce cas d'utilisation. Cet ensemble d'éléments, comportant à la fois une structure statique et dynamique, est modélisé en UML par une collaboration.

# Diagrammes d'interaction

❖ Il existe deux types de diagrammes d'interaction :

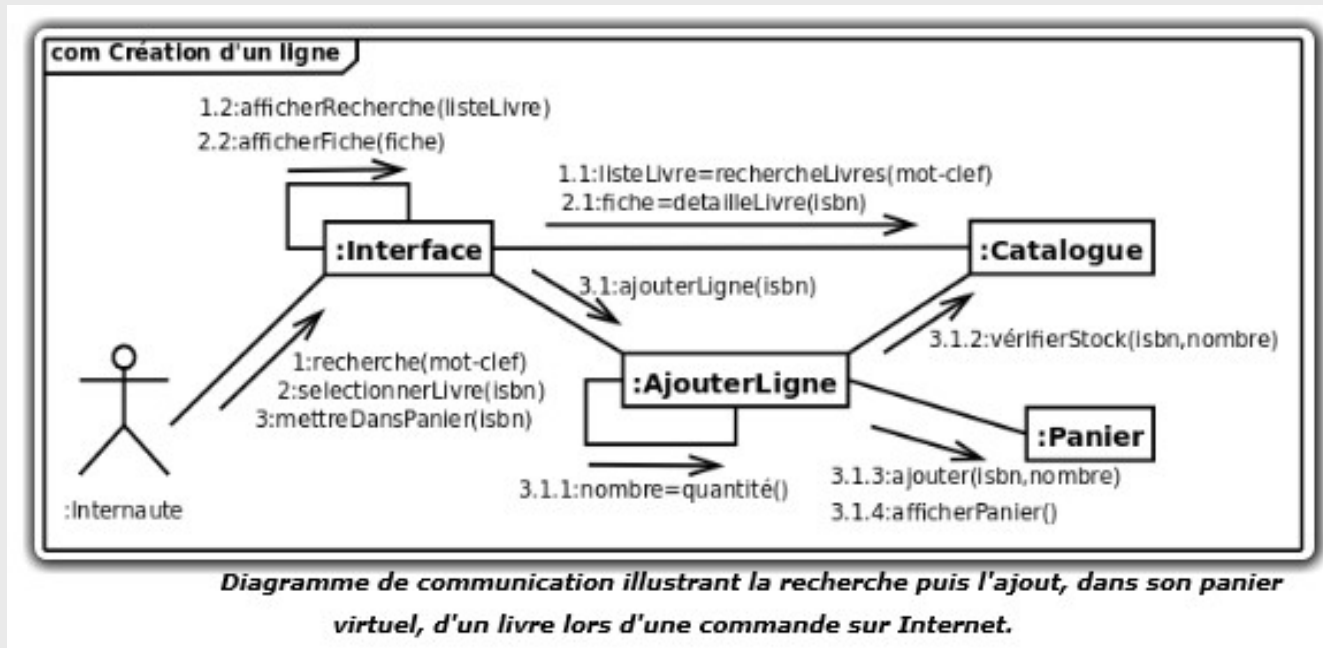
- Diagramme de communication ;
- Diagramme de séquence.

❖ **Diagramme de communication :**

- rend compte de l'organisation spatiale des participants à l'interaction, il est souvent utilisé pour illustrer un cas d'utilisation ou pour décrire une opération.
- aide à valider les associations du diagramme de classe en les utilisant comme support de transmission des messages.

❖ **Diagramme de séquence :**

# Objectif



# Objectif

- 
- 
- 

Elément dans le diagramme de communication	Description
<b>Représentation des lignes de vie</b>	[<nom_du_rôle>] : [<Nom_du_type>] Au moins un des deux noms doit être spécifié dans l'étiquette, les deux points (:) sont, quant à eux, obligatoires.
<b>Représentation des connecteurs</b>	Les relations entre les lignes de vie sont appelées connecteurs et se représentent par un trait plein reliant deux lignes de vie et dont les extrémités peuvent être ornées de multiplicités.
<b>Représentation des messages</b>	'['<cond>']' [<séq>] [ * [  ] ['<iter>'] ] : ] [<var> :=] <msg>([<par>])

<cond> : est une condition sous forme d'expression booléenne entre crochets.

<séq> : est le numéro de séquence du message. On numérote les messages par envoi et sous-envoi désignés par des chiffres séparés par des points : ainsi l'envoi du message 1.4.4 est postérieur à celui du message 1.4.3, tous deux étant des conséquences (i.e. des sous-envois) de la réception d'un message 1.4. La simultanéité d'un envoi est désignée par une lettre : les messages 1.6a et 1.6b sont envoyés en même temps.

<iter> : spécifie (en langage naturel, entre crochets) l'envoi séquentiel (ou en parallèle, avec ||) de plusieurs messages. On peut omettre cette spécification et ne garder que le caractère \* (ou \*||) pour désigner un message récurrent, envoyé un certain nombre de fois.

<var> : est la valeur de retour du message, qui sera par exemple transmise en paramètre à un autre message.

<msg> : est le nom du message.

<par> : désigne les paramètres (optionnels) du message.

# • Diagramme de séquence

- 
- 

Les principales informations contenues dans un diagramme de séquence sont les messages échangés entre les lignes de vie, présentées dans un ordre chronologique. Ainsi, contrairement au diagramme de communication, le temps y est représenté explicitement par une dimension (la dimension verticale) et s'écoule de haut en bas :

