

# Chapitre V: Diagramme d'activité

Nour H. BEN SLIMEN ATTAOUI

# UML:DIAGRAMME D'ACTIVITE

## Définition

- ▶ Le diagramme d'activités est un automate à états finis, dans lequel tous les états sont des états d'activités et presque toutes les transitions sont déclenchées par l'achèvement des activités dans l'état source. Comme les autres diagrammes, le diagramme d'activité peut également contenir des contraintes et des notes. C'est un organigramme d'activités.
- ▶ On utilise le diagramme d'activités pour modéliser les aspects dynamiques du système: le comportement d'un objet lorsqu'il passe d'un état à un autre, le comportement d'une opération, le comportement d'un cas d'utilisation, etc.

# UML:DIAGRAMME D'ACTIVITE

## But de diagramme d'activité

- ▶ Diagramme d'activité est utilisé pour:
  - ▶ Modéliser un workflow dans un use case ou entre plusieurs use cases.
  - ▶ Spécifier une opération (décrire sa logique).
- ▶ Le diagramme d'activité est approprié pour modéliser la dynamique d'une tâche ou d'un processus métier.

# UML:DIAGRAMME D'ACTIVITE

## Formalisme

En règle générale, le diagramme d'activités comprend les éléments suivants:

- **Nœud de départ** : élément symbolisant le début de l'activité, que l'on représente par un cercle noir
- **Les états d'action et les états d'activité** : étape dans l'activité où les utilisateurs ou le logiciel exécutent une tâche donnée. Ils sont symbolisés par des rectangles aux bords arrondis.
- **Transition** : connecteur qui illustre le flux entre les étapes du diagramme.
- **Branche (ou Nœud de décision)** : embranchement conditionnel dans le flux, qui est représenté par un losange. Il comporte une seule entrée et au moins deux sorties.
- **Nœud de fin** : élément symbolisant l'étape finale de l'activité, que l'on représente par un cercle noir avec un contour.

# UML:DIAGRAMME D'ACTIVITE

## Formalisme

### Les états d'action et les états d'activité

C'est l'exécution d'un traitement tel que évaluer une expression, appeler une opération, créer ou détruire un objet, etc.

- ▶ Les actions ne peuvent pas être décomposées, ne peuvent pas être interrompues même si un événement survient. Le temps d'exécution d'un état d'action est considéré comme insignifiant.
- ▶ En revanche, les états d'activités peuvent être décomposés: il est possible de représenter un état d'activité par un autre diagramme d'activité. En plus les états d'activité peuvent être interrompus, leurs temps d'exécution est relativement lents.

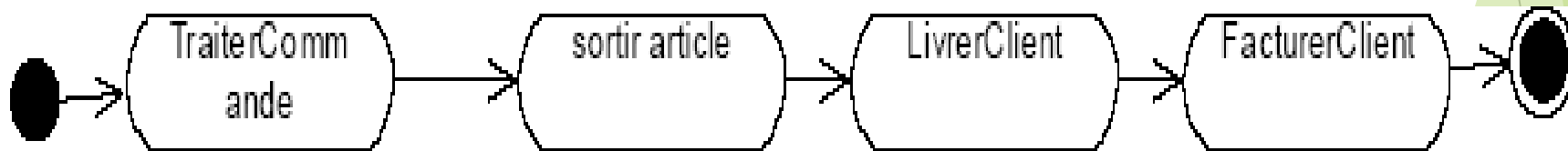
# UML:DIAGRAMME D'ACTIVITE

## Formalisme

### Les états d'action et les états d'activité

La notation est la même pour un état d'action et un état d'activité (les parenthèses indiquent qu'il s'agit d'une activité), cependant ce dernier peut comporter des éléments en plus tels que l'action au moment d'entrée dans l'état, l'action au moment de sortie de l'état, ainsi que la spécification des sous automates.

Exemple 1:



# UML:DIAGRAMME D'ACTIVITE

## Formalisme

### Les états d'action et les états d'activité

#### Exemple:



**Servir boisson ()** est une activité car elle peut comporter un automate tel que :



# UML:DIAGRAMME D'ACTIVITE

## Formalisme

### Les transitions

- ▶ C'est une relation entre deux états (flèche). C'est une transition sans déclencheur, car le contrôle passe immédiatement à l'état suivant une fois l'activité de l'état source est terminé.
- ▶ Les transitions simples et séquentielles sont les plus courantes.
- ▶ Cependant, on peut inclure une **branche** ou une **barre de synchronisation**.



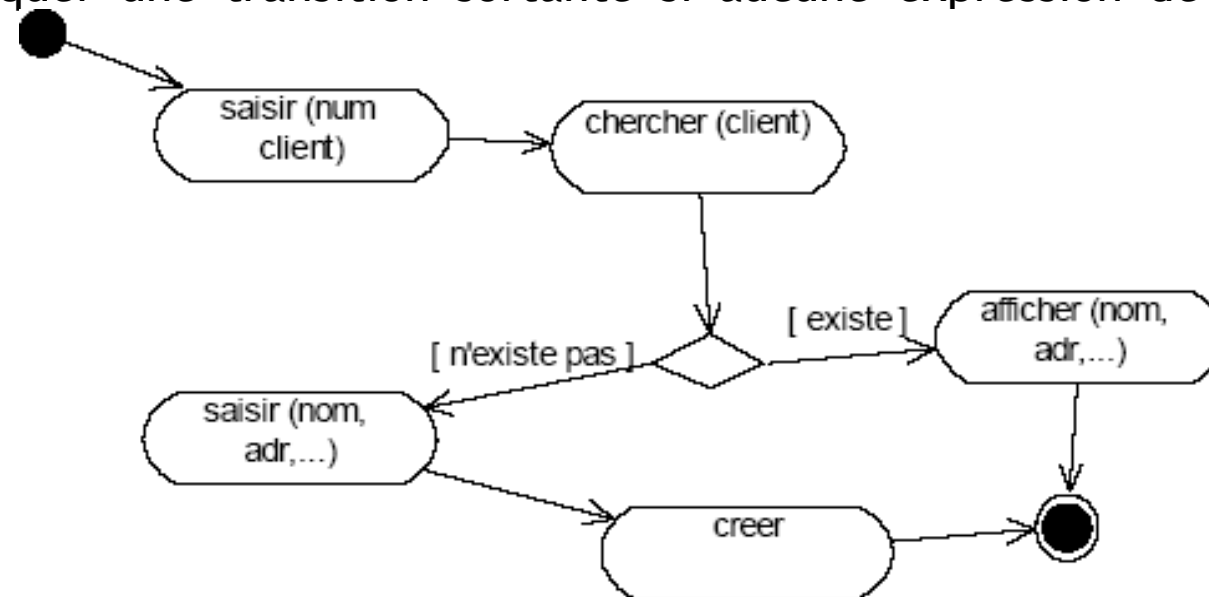
# UML:DIAGRAMME D'ACTIVITE

## Formalisme

### Les transitions

- **Une branche:** Elle spécifie que l'on peut prendre plusieurs chemins. Une branche peut avoir une transition entrante et deux ou plusieurs transitions sortantes. Sur chaque transition, on place une expression de garde (expression booléenne). Pour plus de facilité, on peut utiliser le mot clé **else** pour marquer une transition sortante si aucune expression de garde n'est vraie.

#### Exemple:

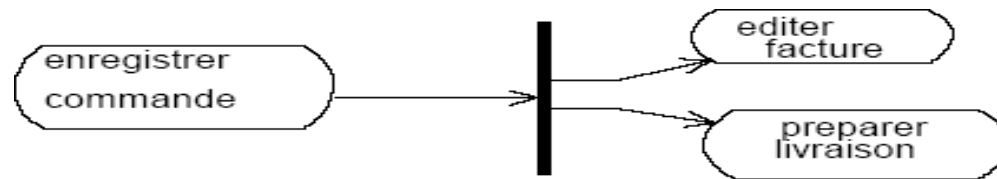


# UML:DIAGRAMME D'ACTIVITE

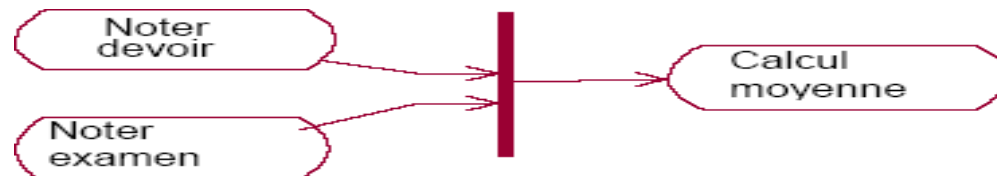
## Formalisme

### Les transitions

- **Une barre de synchronisation:** Il est possible de modéliser des états parallèles en utilisant une barre de synchronisation qui permet de spécifier la division (**fourche**) ou le regroupement (**jonction**) des états.
- **Une fourche:** peut avoir une transition entrante et deux ou plusieurs transitions sortantes. Après la fourche, les activités continuent en parallèle.



- **Une jonction:** peut avoir deux ou plusieurs transitions entrantes et une transition sortante. Avant la jonction, les activités ou les actions sont déclenchées en parallèle : la barre de synchronisation ne peut être franchie que lorsque les transitions en entrée de la barre de synchronisation ont été déclenchées.



# UML:DIAGRAMME D'ACTIVITE

## Formalisme

### Les travées (couloirs d'activités)

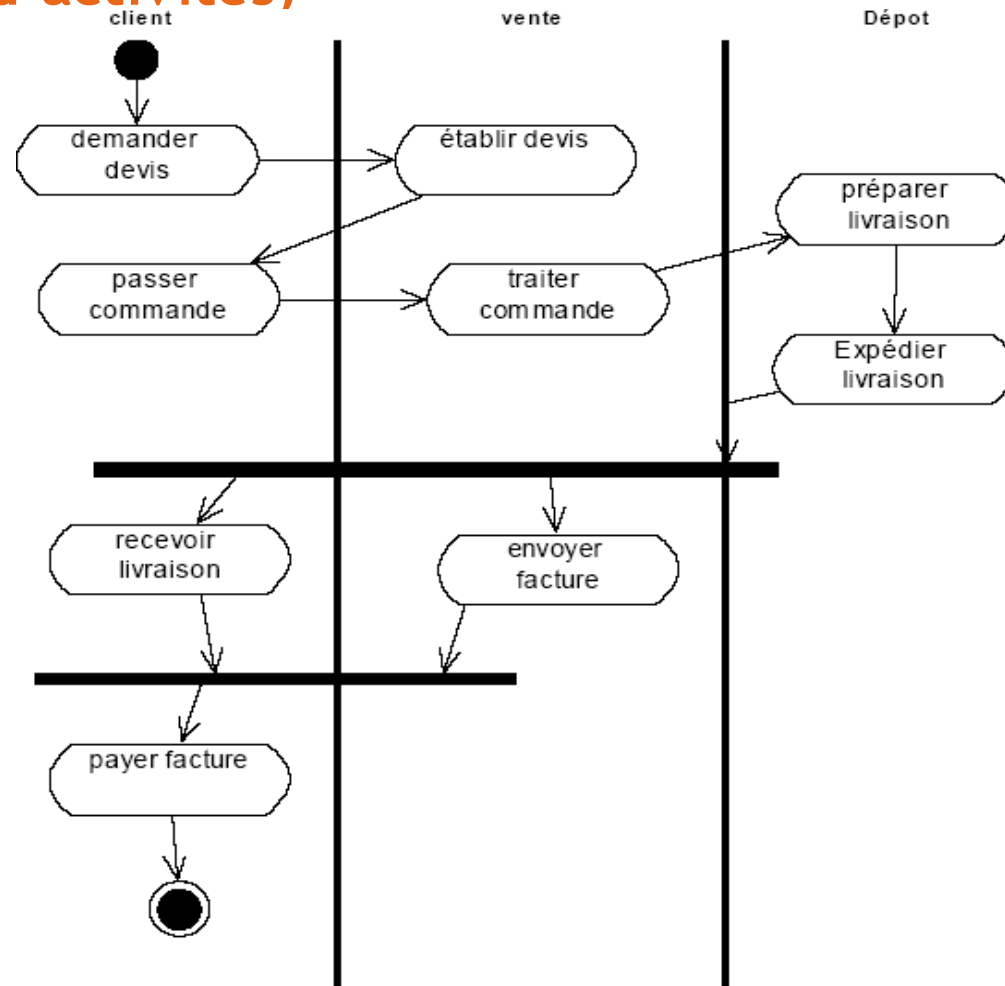
- Le diagramme d'activités peut être décomposé en des couloirs d'activités pour montrer la responsabilité de l'action au sein d'une organisation. C'est une division des états d'activités en groupes sur un diagramme d'activités, Chaque groupe (travée) représente un département ou quelqu'un responsable de ces activités. Chaque travée possède un nom unique dans le diagramme. Chaque activité appartient à une seule travée même si les transitions peuvent passer d'une travée à une autre.

# UML:DIAGRAMME D'ACTIVITE

## Formalisme

### Les travées (couloirs d'activités)

#### ► Exemple:



# Exercice 1

## Diagramme d'activité

- Le logiciel de gestion des réparations est destiné en priorité au chef d'atelier, il devra lui permettre de saisir les fiches de réparations et le travail effectué par les divers employés de l'atelier. Pour effectuer leur travail, les mécaniciens et autres employés de l'atelier vont chercher des pièces de rechange au magasin. Lorsque le logiciel sera installé, les magasiniers ne fourniront des pièces que pour les véhicules pour lesquels une fiche de réparation est ouverte; ils saisiront directement les pièces fournies depuis un terminal installé au magasin. Lorsqu'une réparation est terminée, le chef d'atelier va essayer la voiture. Si tout est en ordre, il met la voiture sur le parc clientèle et bouclera la fiche de réparation informatisée. Les fiches de réparations bouclées par le chef d'atelier devront pouvoir être importées par le comptable dans le logiciel comptable.
- 1. Concevez le modèle de cas d'utilisation du logiciel de gestion des réparations
- 2. Créer un diagramme d'activité pour tout le traitement d'une réparation.

# Exercice 2

## Diagramme d'activité

1. Créer un diagramme d'activité pour le use case « Créer une fiche de réparation » Pour créer une fiche de réparation, le chef d'atelier saisit les critères de recherche de voitures dans le système. Le logiciel de gestion des réparations lui donne la liste des voitures correspondant aux critères entrés. Si la voiture existe, le chef d'atelier va sélectionner la voiture. Le logiciel va, ensuite, fournir les informations sur le véhicule. Si la voiture est sous garantie, le chef devra saisir la date de demande de réparation. Si la voiture n'existe pas, le chef va saisir les informations concernant ce nouveau véhicule. Dans tous les cas, le chef d'atelier devra saisir la date de réception et de restitution. Si le dommage de la voiture est payé par l'assurance, le logiciel va fournir une liste d'assurances au chef d'atelier. Ce dernier sélectionnera l'assurance adéquate. Enfin, le logiciel enregistre la fiche de réparation.
2. Créer un diagramme d'activité pour le use case « Créer une fiche de réparation »