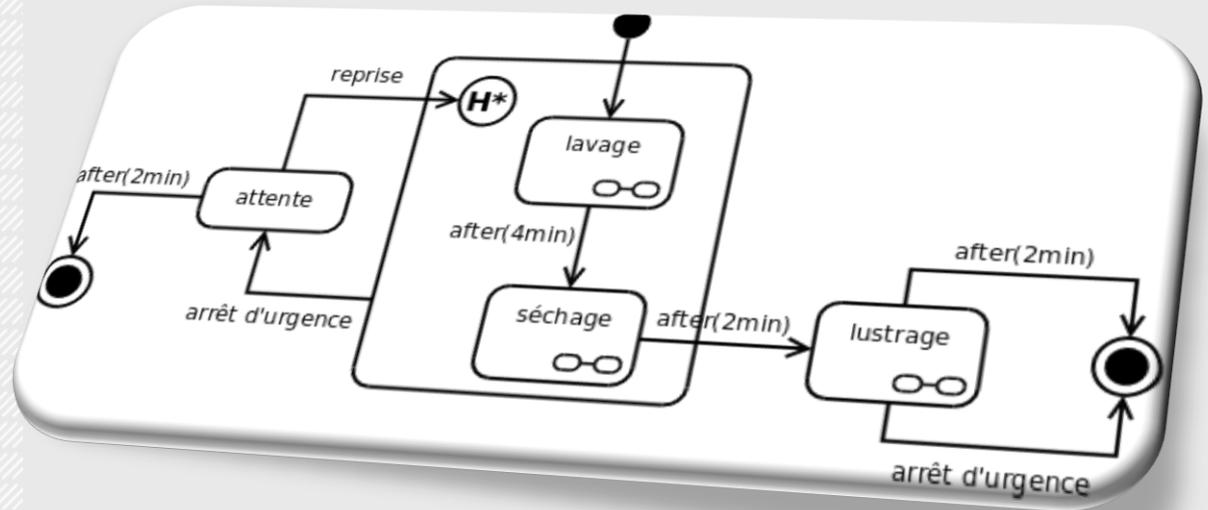


# UML : DIAGRAMME D'ETATS-TRANSITIONS



# UML : DIAGRAMME D'ETATS-TRANSITIONS

## DEFINITION:

- Décrit le **comportement** des objets d'une classe au moyen d'un automate d'états associés à la classe
- Le comportement est modélisé par un graphe :
  - Nœuds = **états possibles** des objets
  - Arcs = transitions d'état à état
- Une **transition** :
  - exécution d'une action
  - réaction de l'objet sous l'effet d'une occurrence **d'événement**

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## DEFINITION:

- Description du **cycle de vie** d'un objet d'une classe :
  - Les **états** qui peuvent être pris par les **objets** d'une classe
  - Les **événements** qui provoquent la **transition d'un état à un autre**
  - Les **actions** subies/provoquées qui accompagnent un **changement d'état**
  - Les **activités** qui surviennent tant que l'objet est dans un **état donné**.

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## OBJECTIFS:

- **Globalement** : étudier les états d'un Système d'Information
- **Plus particulièrement** : Comprendre le système d'information en s'intéressant aux **classes** qui présentent des traitements complexes
  - On se limite aux classes qui sont cruciales pour le champ de l'étude
  - On se limite aux états qui sont d'un intérêt pour le champ de l'étude
- Fournir une **représentation** dynamique du **comportement** des objets d'une classe
- Aider à déterminer les **événements** qui occasionnent les transitions
- Aider à déterminer les **opérations** qui vont permettre ces transitions

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## UTILITE :

- Décrire un processus (type workflow)
- Aider à concevoir les activités, les IHM
- Aider à construire les scénarios de test
- Aider à déterminer les états pour gérer les risques de dysfonctionnement

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## NOTION D'ETAT:

- Un état = étape dans le cycle de vie d'un objet
- Chaque objet possède à un instant donné un état particulier
- Chaque état est identifié par un nom.
- Un état est stable et durable

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## NOTION D'ETAT:

- Un état = étape dans le **cycle de vie** d'un objet
- Chaque objet possède à un instant donné un état particulier
- Chaque état est identifié par un nom.
- Un état est **stable** et **durable**
- Chaque diagramme d'états-transitions comprend un état
- Il est possible de n'avoir aucun état final : *un système qui ne s'arrête jamais.*



Etat initial

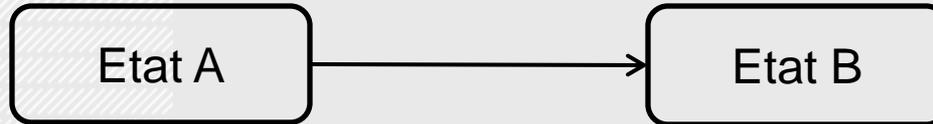


Etat final

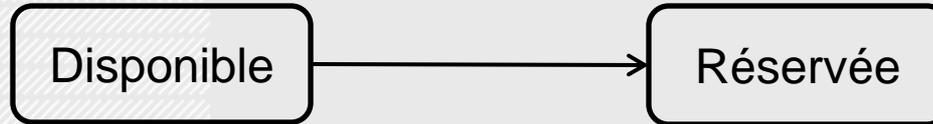
# UML : DIAGRAMME D'ETATS-TRANSITIONS

## NOTION DE TRANSITION:

- Les états sont reliés par des connexions unidirectionnelles appelées **transitions**



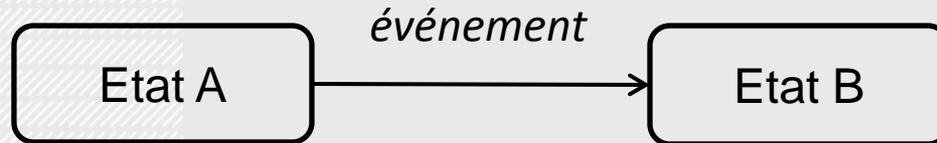
- Ex : place de parking



# UML : DIAGRAMME D'ETATS-TRANSITIONS

## NOTION D'ÉVÉNEMENT:

- Un événement correspond à l'occurrence d'une situation donnée dans le domaine étudié
- Un événement est une information instantanée qui doit être traitée à l'instant où il se produit



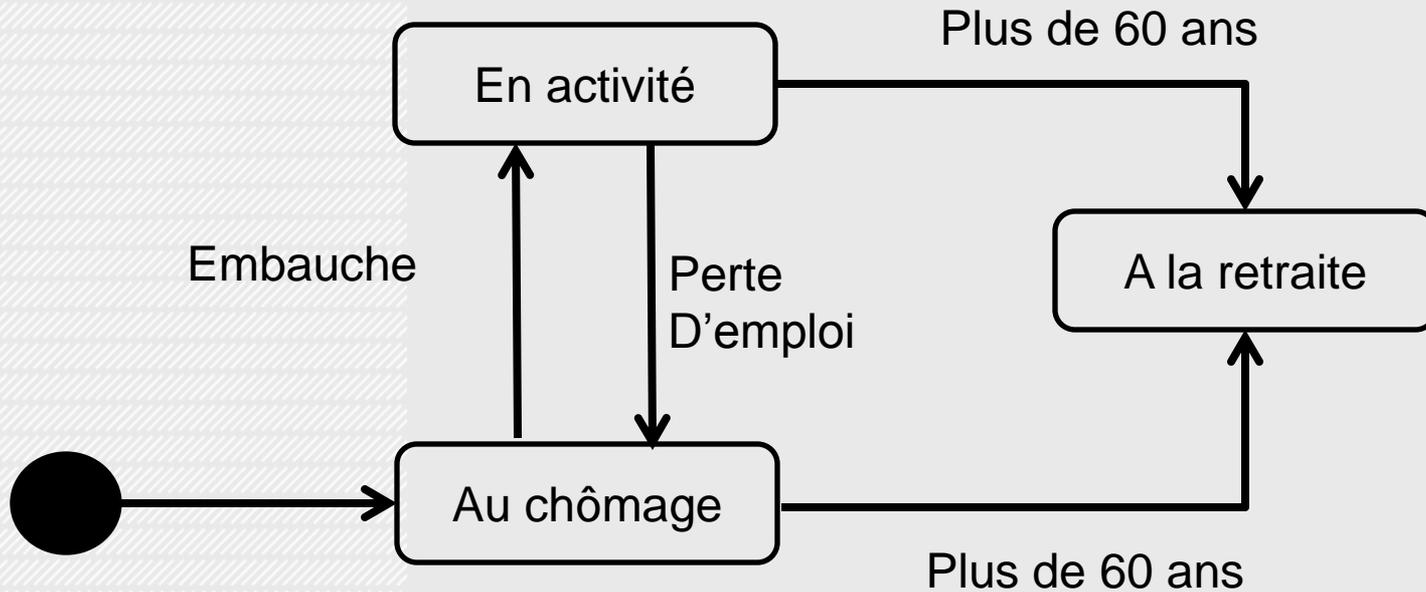
# UML : DIAGRAMME D'ETATS-TRANSITIONS

## NOTION D'ÉVÉNEMENT:

- Syntaxe d'un événement :
  - Nom de l'événement (Nom de paramètre : Type,.....)
- La description complète d'un événement est donnée par :
  - Nom de l'événement
  - Liste des paramètres
  - Objet expéditeur
  - Objet destinataire
  - Description textuelle

# UML : DIAGRAMME D'ETATS-TRANSITIONS

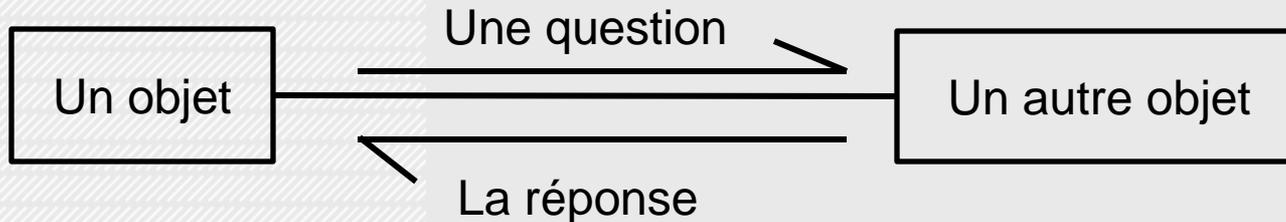
## NOTION D'EVENEMENT:



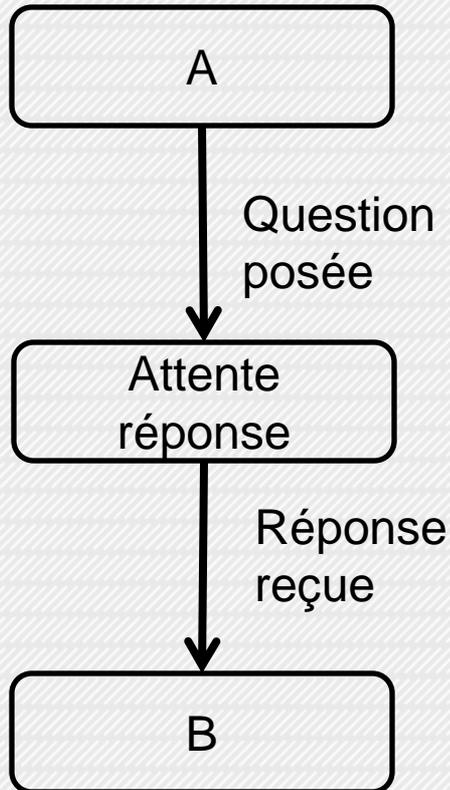
# UML : DIAGRAMME D'ETATS-TRANSITIONS

## COMMUNICATION ENTRE OBJETS PAR EVENEMENTS

- La communication est de type asynchrone, atomique et unidirectionnelle



# UML : DIAGRAMME D'ETATS-TRANSITIONS



## COMMUNICATION ENTRE OBJETS PAR EVENEMENTS

- L'objet émetteur de la requête se met en attente de la réponse de l'objet récepteur de la requête

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## EN RESUME :

- Etat d'un objet :
  - Situation d'un objet que l'on désire connaître et gérer.
- Transition :
  - Passage d'un objet d'un état à un autre. Elle est déclenchée par un événement
- Evénement :
  - Stimulus qui provoque une (ou plusieurs) transition(s). A chaque stimulus peut correspondre une **action** responsable des modifications de l'objet (les valeurs des attributs)

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## ETATS SPECIAUX:

- 2 états prédéfinis :
  - état de démarrage : obligatoire, unique
  - état de fin : optionnel, peut-être multiple

Création de l'objet



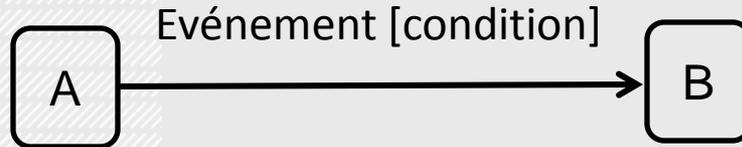
Fin de vie de l'objet



# UML : DIAGRAMME D'ETATS-TRANSITIONS

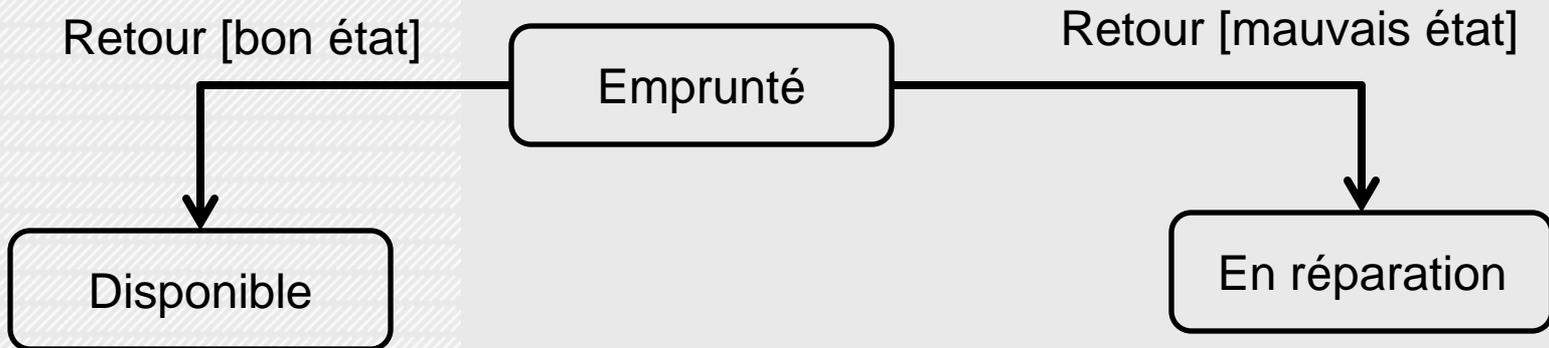
## NOTION DE GARDE:

- Une garde est une condition booléenne qui permet ou non le déclenchement d'une transition lors de l'occurrence d'un événement.



# UML : DIAGRAMME D'ETATS-TRANSITIONS

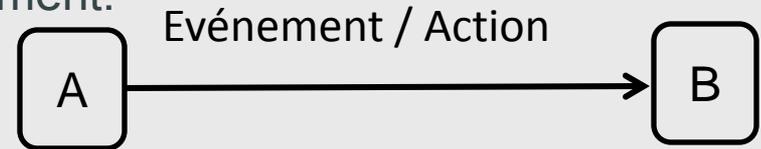
COMMUNICATION ENTRE OBJETS PAR EVENEMENTS :



# UML : DIAGRAMME D'ETATS-TRANSITIONS

## NOTIONS D'OPERATION ET D'ACTION :

- Action et activités : Le lien entre les **opérations** définies dans la spécification d'une **classe** et les **événements** apparaissant dans la diagramme d'états-transitions
- Chaque transition peut avoir une action à exécuter lorsqu'elle est déclenchée
- L'action est considérée comme instantanée et atomique
- Une action correspond à l'exécution d'une des opérations déclarées dans la classe de l'objet destinataire de l'événement.



- L'action a accès aux paramètres de l'événement ainsi qu'aux attributs de l'objet sur lequel elle s'applique

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## ACTIONS DANS UN ETAT :

- Les états peuvent également contenir des actions :

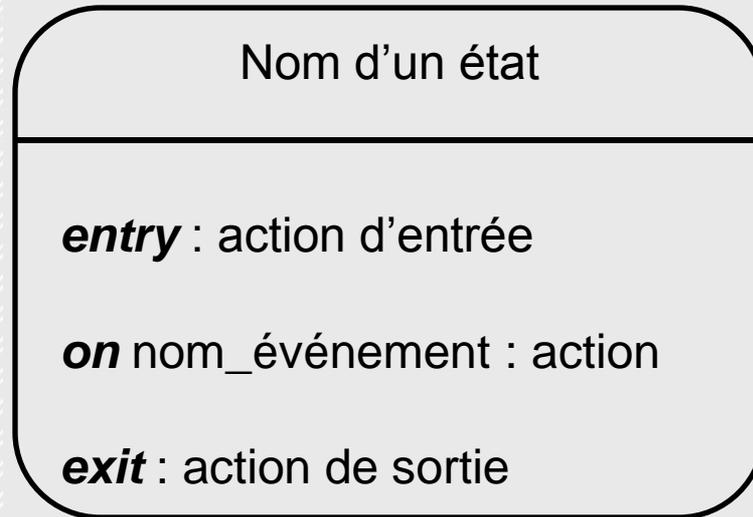
Elles sont exécutées :

- à l'entrée (*entry*) ou à la sortie (*exit*) de l'état
- lorsqu'une occurrence d'événement interne (*on*) survient  
(exécutée lors de l'occurrence d'un événement qui ne conduit pas à un autre état)

# UML : DIAGRAMME D'ETATS-TRANSITIONS

## ACTIONS DANS UN ETAT :

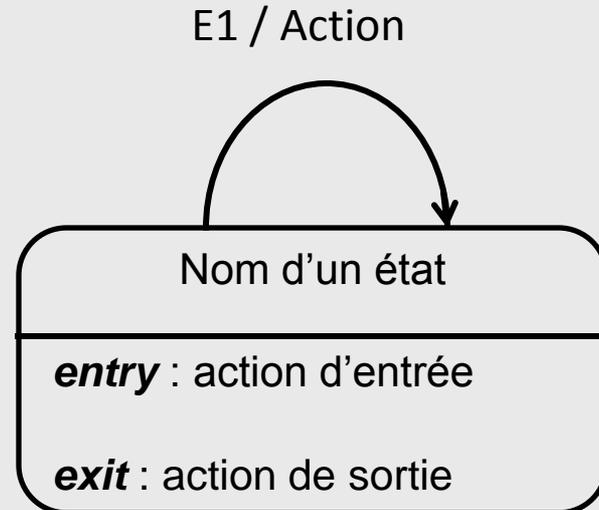
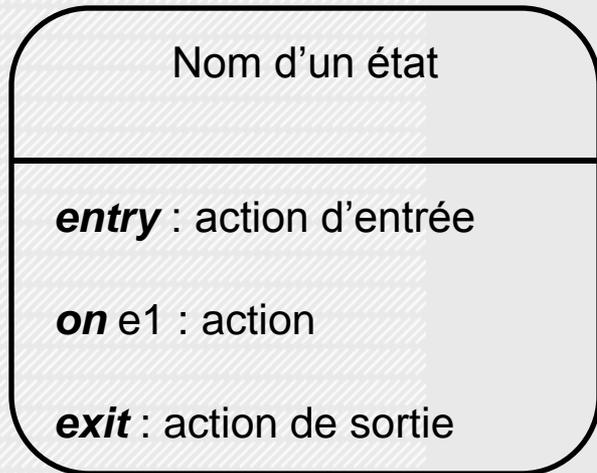
- Représentation :



# UML : DIAGRAMME D'ETATS-TRANSITIONS

## OPERATIONS, ACTIONS ET ACTIVITES :

- Un événement interne n'entraîne pas l'exécution des actions de sortie et d'entrée, contrairement au déclenchement d'une transition réflexive.



# UML : DIAGRAMME D'ETATS-TRANSITIONS

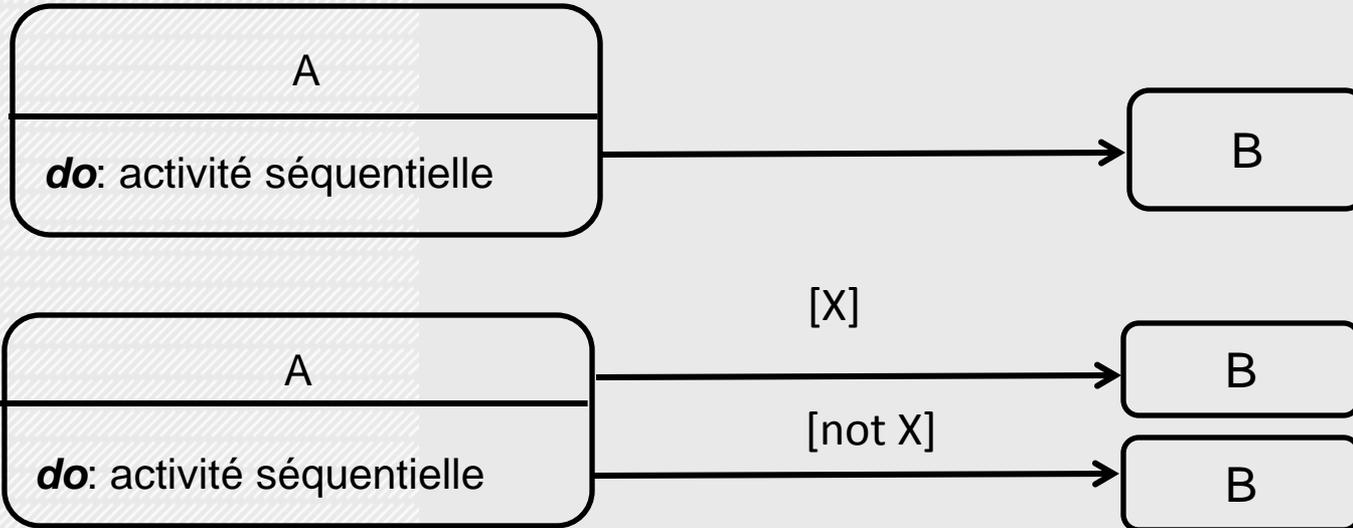
## OPERATIONS, ACTIONS ET ACTIVITES :

- Contrairement à une action, une activité est une opération qui dure un certain temps
- Les activités sont associées aux états
  - commencent quand on est entré dans l'état
  - s'exécutent jusqu'à la fin si elles ne sont pas interrompues par une transition sortante (donc tant que l'état ne change pas)
  - peuvent être interrompues car elles ne modifient pas l'état de l'objet
- Les activités sont notées dans la partie inférieure de l'état

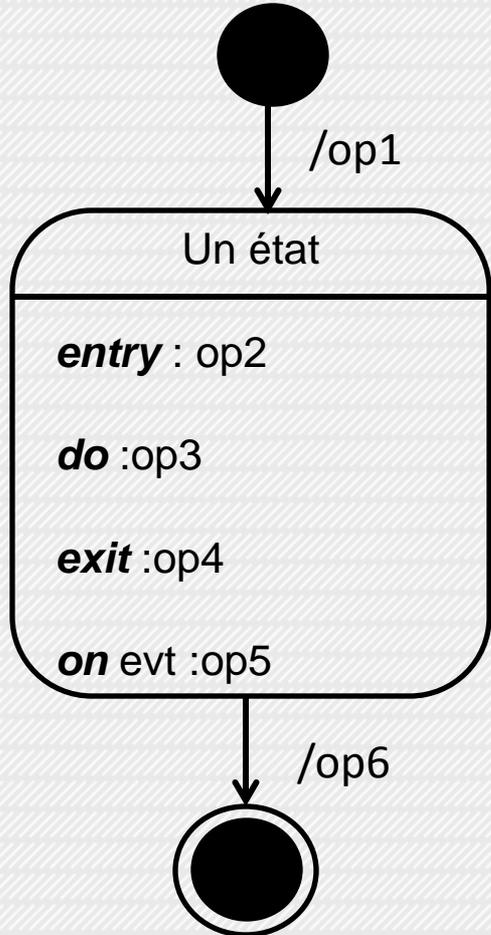
# UML : DIAGRAMME D'ETATS-TRANSITIONS

## OPERATIONS, ACTIONS ET ACTIVITES :

- Lorsqu'une activité se termine, les transitions automatiques (sans événement), mais éventuellement protégées par des gardes, sont déclenchées



# UML : DIAGRAMME D'ETATS-TRANSITIONS

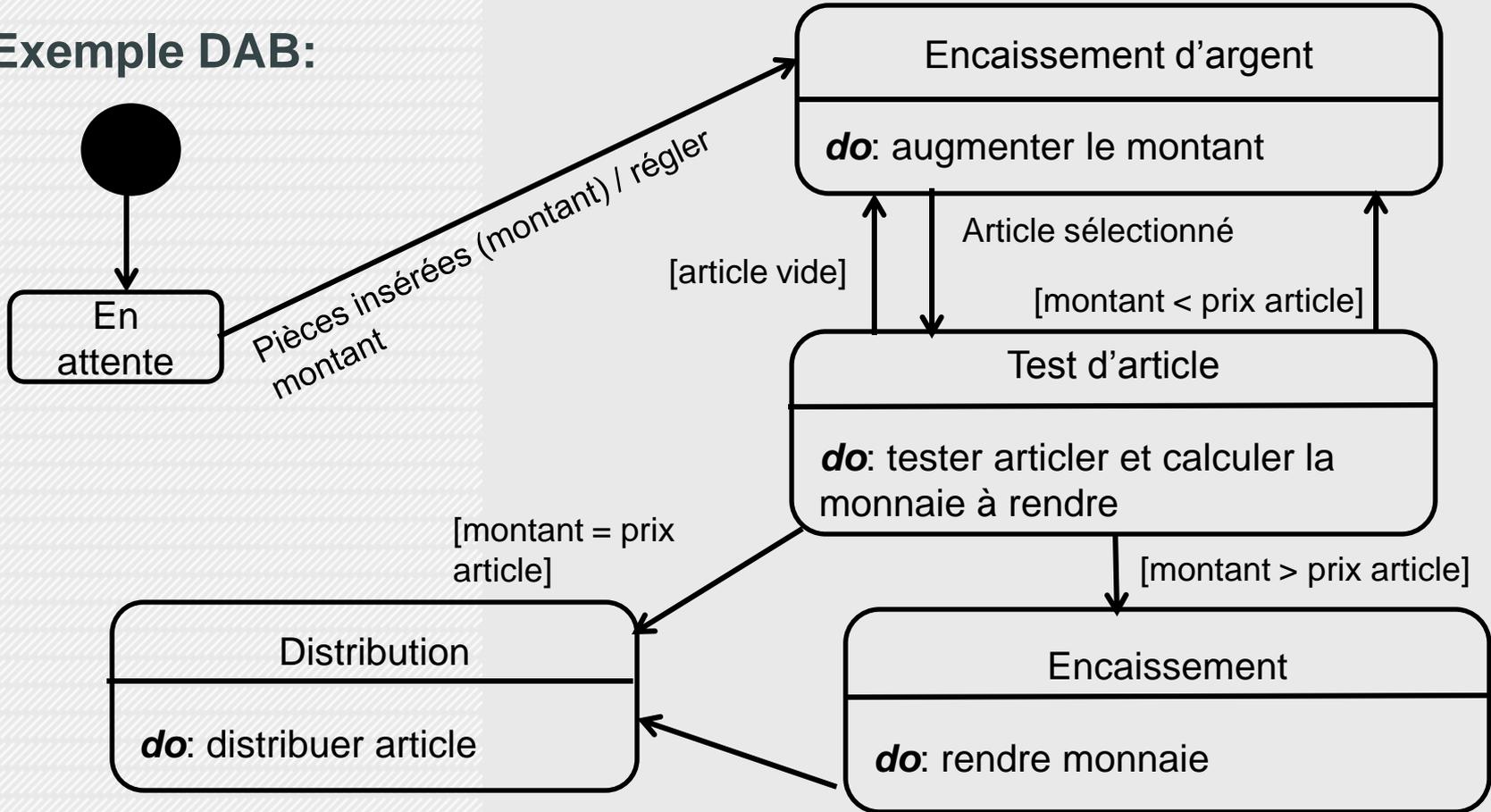


## OPERATIONS, ACTIONS ET ACTIVITES :

- 6 manières d'associer une opération à une transition :
  - l'action **associée** à la transition d'entrée (op1)
  - l'action **d'entrée** de l'état (op2)
  - **l'activité** dans l'état (op3)
  - l'action de **sortie** de l'état (op4)
  - l'action **associée** aux **événements internes** (op5)
  - l'action **associée** à la **transition** de la sortie de l'état (op6)

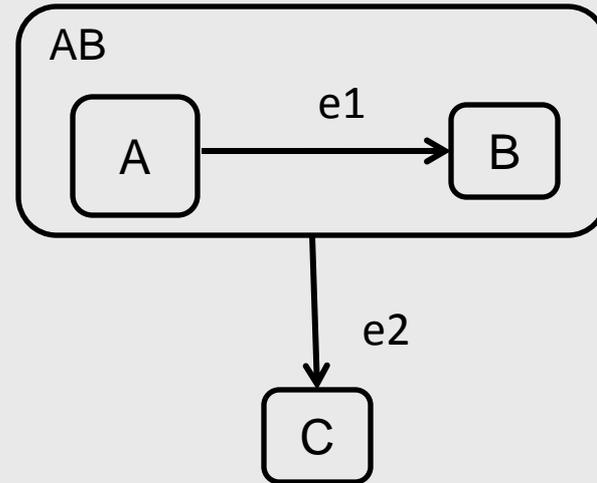
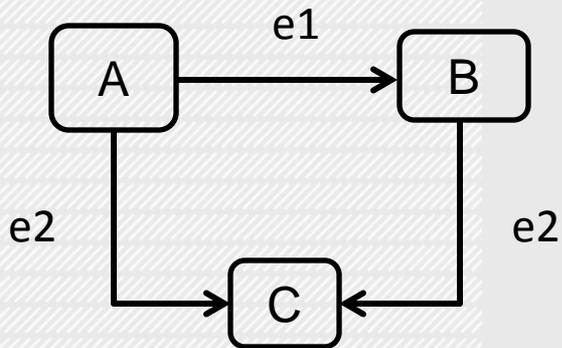
# UML : DIAGRAMME D'ETATS-TRANSITIONS

Exemple DAB:



# UML : DIAGRAMME D'ETATS-TRANSITIONS

Généralisation d'états:



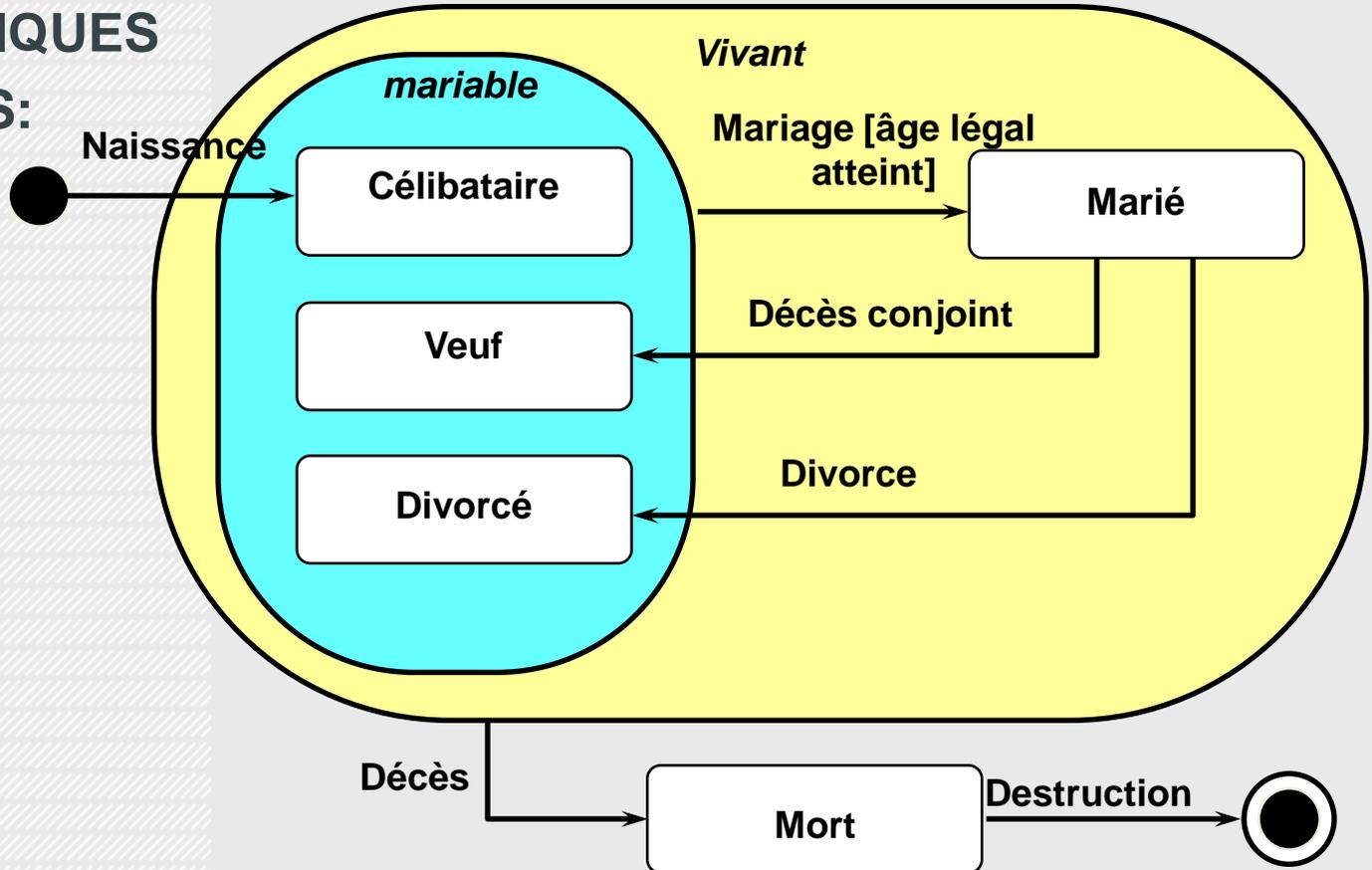
# UML : DIAGRAMME D'ETATS-TRANSITIONS

## ETATS IMBRIQUES - COMPOSITES:

- Si le diagramme d'état transition devient trop complexe, on peut utiliser des états imbriqués pour le simplifier
- Un super-état ou état composite est un état qui englobe d'autres états appelés sous-états
- Le nombre d'imbrication n'est pas limité (ne pas abusé sinon problème de lisibilité)

# UML : DIAGRAMME D'ETATS-TRANSITIONS

ETATS IMBRIQUES  
COMPOSITES:



# UML : DIAGRAMME D'ETATS-TRANSITIONS

EXERCICES