



# Analyse Orientée Objet

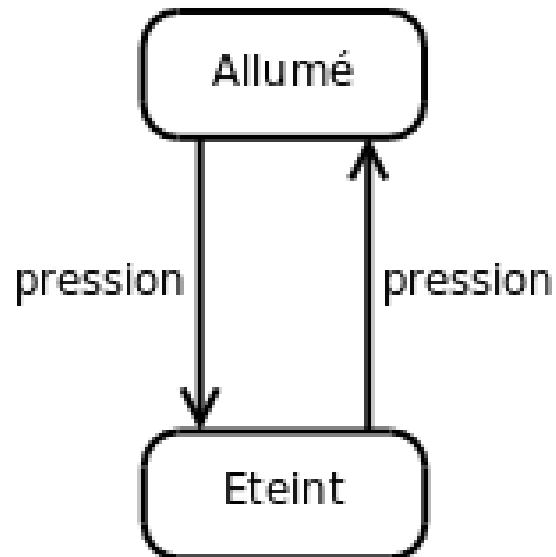
Cours 6 : Diagramme d'états – transitions

# Vie d'un objet

- Les diagrammes de classes et d'objets sont des descriptions statiques d'un objet ou d'un ensemble d'objets
- Pour décrire la vie d'un objet, on a besoin de diagrammes **dynamiques**.
- Un objet a une vie :
  - il naît dans certaines conditions;
  - il change d'état en fonction de traitement d'évènements qui le concernent;
  - enfin il meurt.
- Toutes les informations de la vie d'un objet sont contenues dans les diagrammes de séquence dans lesquels l'objet intervient.

# Diagramme d'états-transitions

- UML propose une modélisation de la vie d'un objet à l'aide d'un diagramme d'états - transitions.
- C'est un **automate d'états finis**.
- Il décrit les états et les changements d'états d'un objet suite aux interactions avec d'autres objets.



# Etats

- Un **état** est caractérisé par sa stabilité et sa durée.
- Il existe deux état particuliers :
  - état **initial** symbolisé par ●
  - état **final** symbolisé par ⊙
- Etat **simple** et état **composite (super-état)**

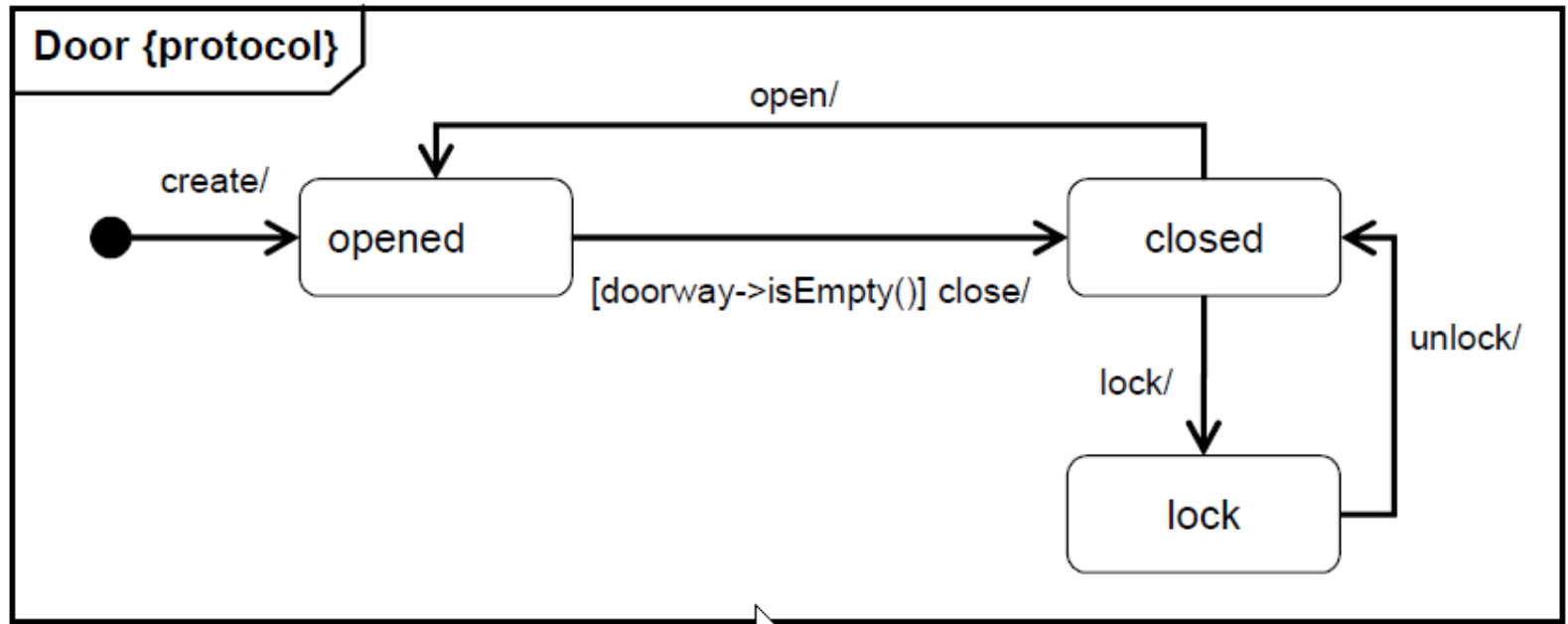
# Transitions

- Une **transition** représente pour l'objet, le passage **instantané** d'un état à un autre état.



- Une transition se produit suite à un **événement**
- Une transition peut être conditionnée à l'aide d'une **garde** : expression booléenne exprimée en langage naturel
- Si un événement déclencheur se produit et que la condition de garde est vérifiée, l'objet passe de l'état source en état cible et exécute une **activité** (effet de transition)

# Example



# Types d'événement

- Événement d'appel (call) :
  - réception de l'appel d'une opération par un objet
  - méthodes déclarées au diagramme de classes
- Événement de type signal :
  - la réception d'un signal envoyé par un objet
  - communication asynchrone à sens unique entre deux objets
- Événement de changement :
  - changement de valeur de faux à vrai d'une expression booléenne : **when (expression)**.
- Événement temporel :
  - épuisement d'un délai (timeout) : **after (durée)**.
  - survenue d'une date : **when(date = <date>)**

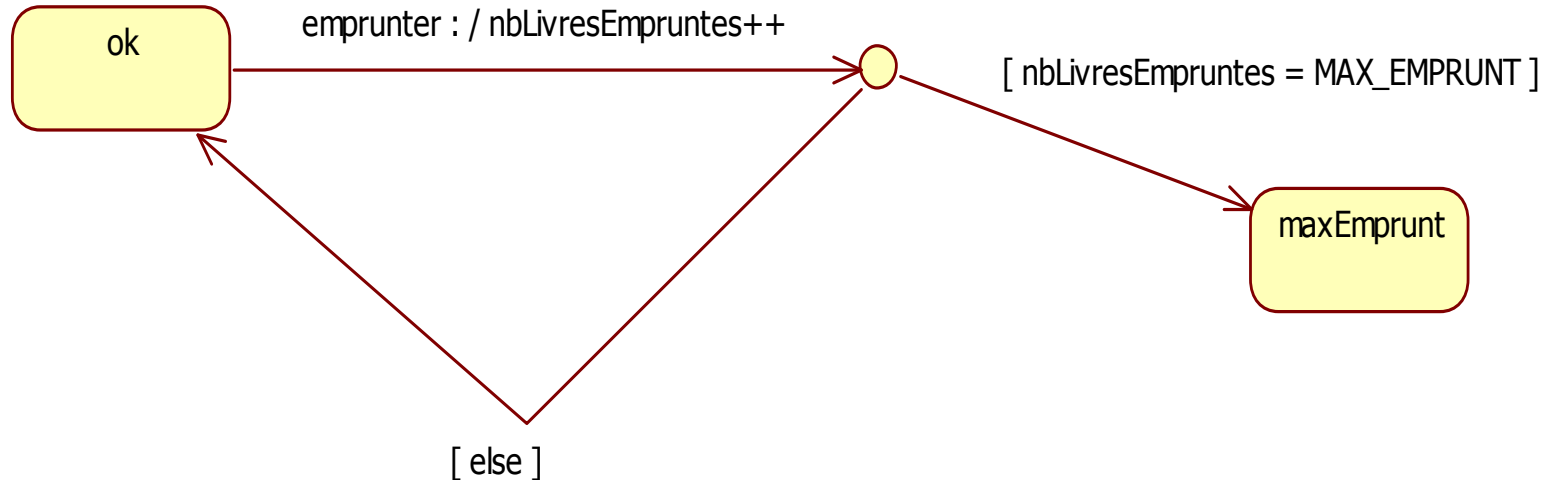
# Types d'activité

- une opération primitive comme une instruction d'assignation
- envoi d'un signal
- appel d'une opération
- une liste d'activités



# Point de décision

- Un point de décision possède une entrée et au moins deux sorties



# Transition interne

- Un seul état
- Quatre déclenchements d'action :
  - **entry / activité** : activité est déclenchée à l'entrée dans l'état.
  - **exit / activité** : activité est déclenchée à la sortie de l'état.
  - **on événement / activité** : activité est déclenchée chaque fois que l'événement se produit.
  - **do / activité** : activité liée à l'état quand il y en a une.

# Exemple

## Saisie mot de passe

entry / ne plus afficher les entrées du clavier

exit / réafficher les entrées du clavier

on aide / afficher l'aide

do / gérer la saisie de l'utilisateur

# Super-état

- C'est un état qui contient des états (notés **sous-états**) et des transitions.
- Il contient un **historique** qui permet de mémoriser le dernier sous état actif du super-état.
- Un super-état est un état. On peut donc passer d'un super-état à un autre état via une transition. Dans ce cas, on mémorise dans l'historique le dernier sous-état actif pour y revenir plus tard.
- Un sous-état est un état. On peut donc passer d'un sous-état à un autre état via une transition.
- Un super-état peut contenir un état de naissance. Il sert à préciser le premier sous-état actif.

# Modélisation de l'objet : lavage d'une voiture

