



Conception Orientée Objet

Diagramme d'état-transition

Tianxiao LIU
Master IISC 1^{ère} Année
CY Cergy-Paris Université
<http://depinfo.u-cergy.fr/~tliu/coo.php>

Plan

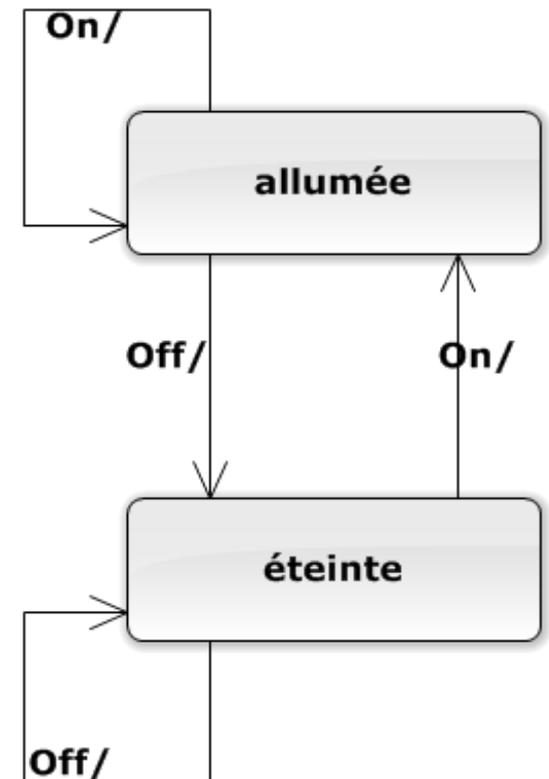
- Principes
- Notions de base
- Points de décision
- Hiérarchie d'état-transition
- Gestion de concurrence (parallélisme)

Principes

- Un graphe : un automate à états finis
 - *Statechart* ou *State Machine*
 - Une machine
 - Comportement de sorties ne dépend pas seulement de l'état de ses entrées
 - Cela dépend aussi d'une historique de sollicitations passées : **état**
 - Effet d'une action sur un objet dépend de son état interne (varier au cours du temps)

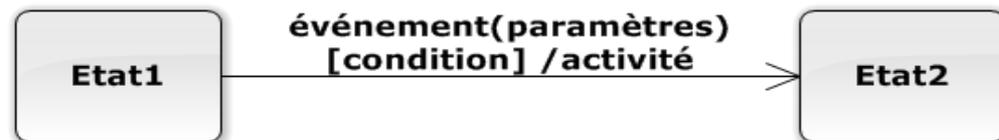
Principes

- Un exemple d'illustration : lampe à torche
 - Deux boutons-poussoirs : ON et OFF
 - Ex. une pression sur **ON** ne produit pas d'effet si la lampe est déjà allumée
- La réaction d'une instance Lampe à cet événement dépend de son état interne



Notions de base

- Etat, événement et transition
 - **Etat** : Représente une période dans la vie d'un objet dans laquelle ce dernier attend un événement ou accomplit une activité
 - **Événement** reçu → **Transition** déclenchée qui fera basculer l'objet dans un nouvel état
 - Etiquette de transition
 - **nom-événement (liste-paramètres) [condition] /activité**



- Etat initial et état final



Notions de base

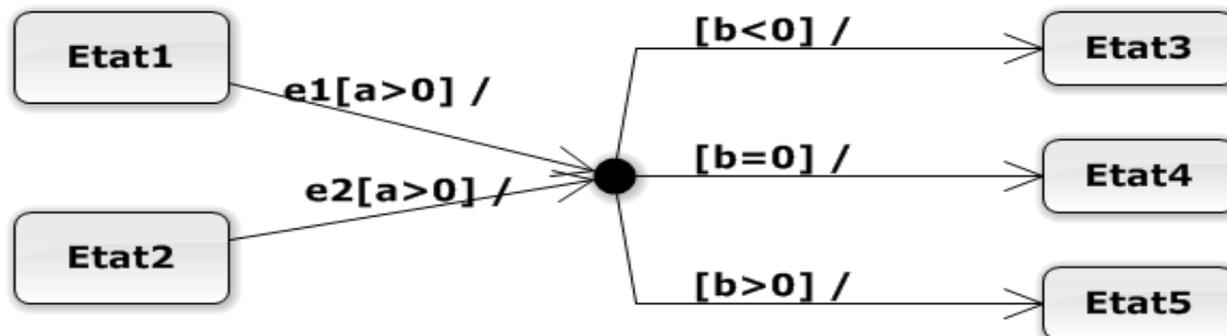
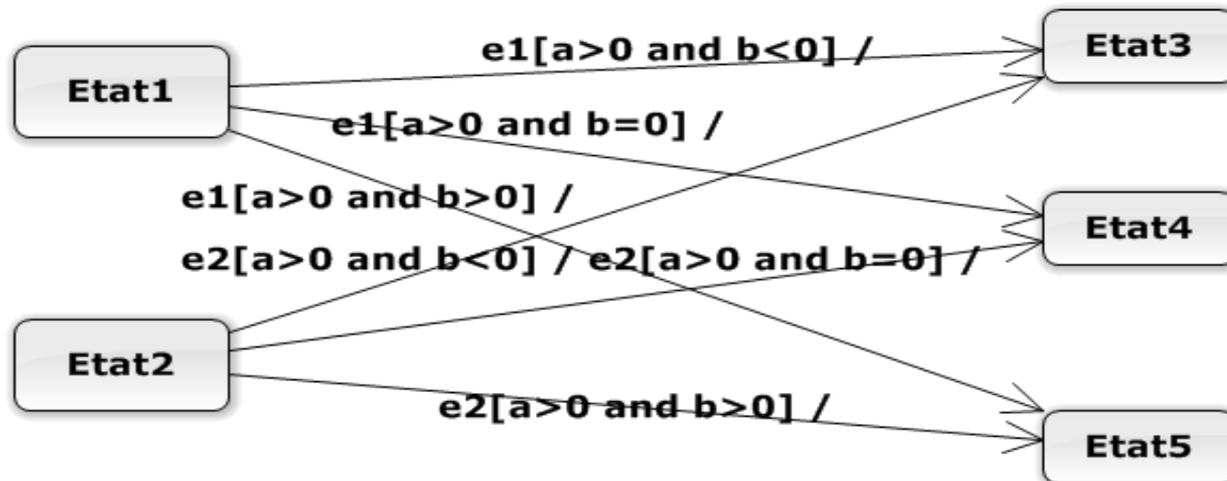
- Types d'événement
 - **call** : un appel de méthode sur l'objet
 - nom-événement (liste de paramètres)
 - nom-paramètre : type-paramètre
 - **change** : valeur d'une condition booléenne
 - when (condition-booléenne)
 - **signal** : réception d'un signal asynchrone
 - **after** : écoulement d'une durée déterminée
 - after (une durée)
 - **do/** : transition interne (voir explication plus tard)

Points de décision

- Point de jonction
 - **Artefact graphique** permettant de partager des segments de transition
 - **Plusieurs** transitions peuvent viser/quitter un point de jonction.
 - Tous les chemins à travers le point de jonction sont **valides**.
 - Motivation : une version équivalente mais plus **compacte**, sans sémantique particulière
- Point de choix
 - Vérification de la condition
 - Possibilité d'utiliser **else**

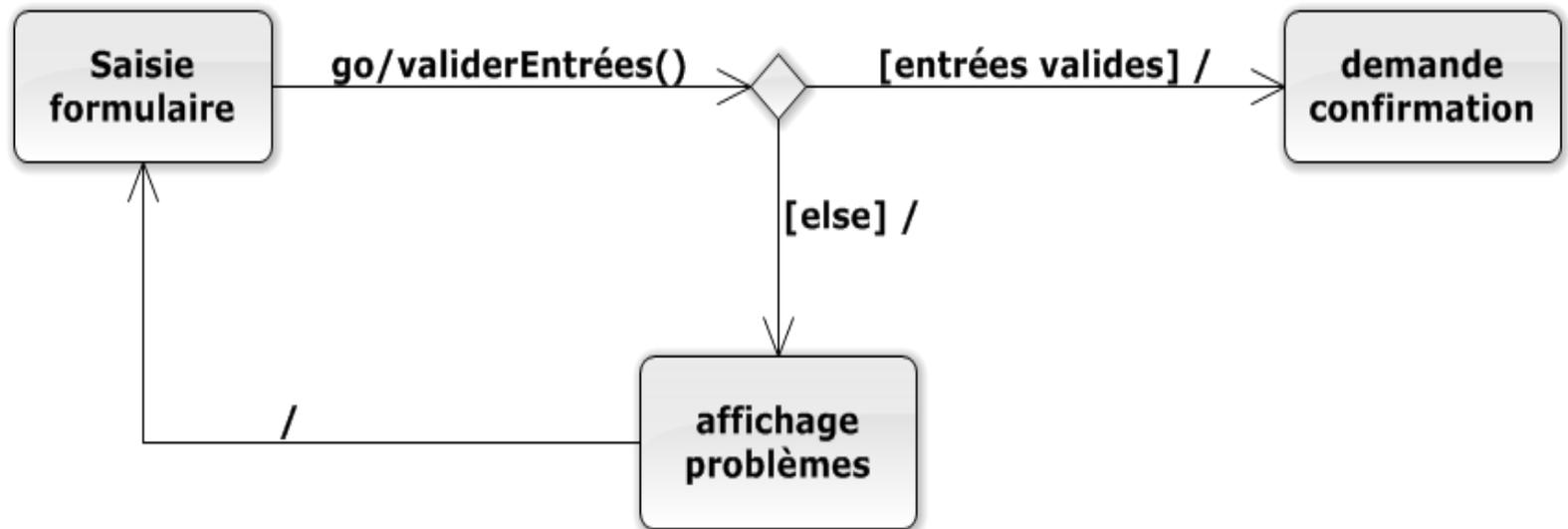
Points de décision

- Point de jonction : exemple



Points de décision

- Point de choix : exemple



Hiérarchie d'état-transition

- Etat et transition internes (syntaxe)
 - Nom de l'état
 - Transition interne
 - **entry/** : définit une activité à effectuer à chaque fois que l'on rentre dans cet état
 - **do/** : définit une activité continue qui est réalisée tant que l'on se trouve dans cet état, ou jusqu'à ce que le calcul associé soit terminée
 - **exit/** : définit une activité à effectuer quand on quitte cet état

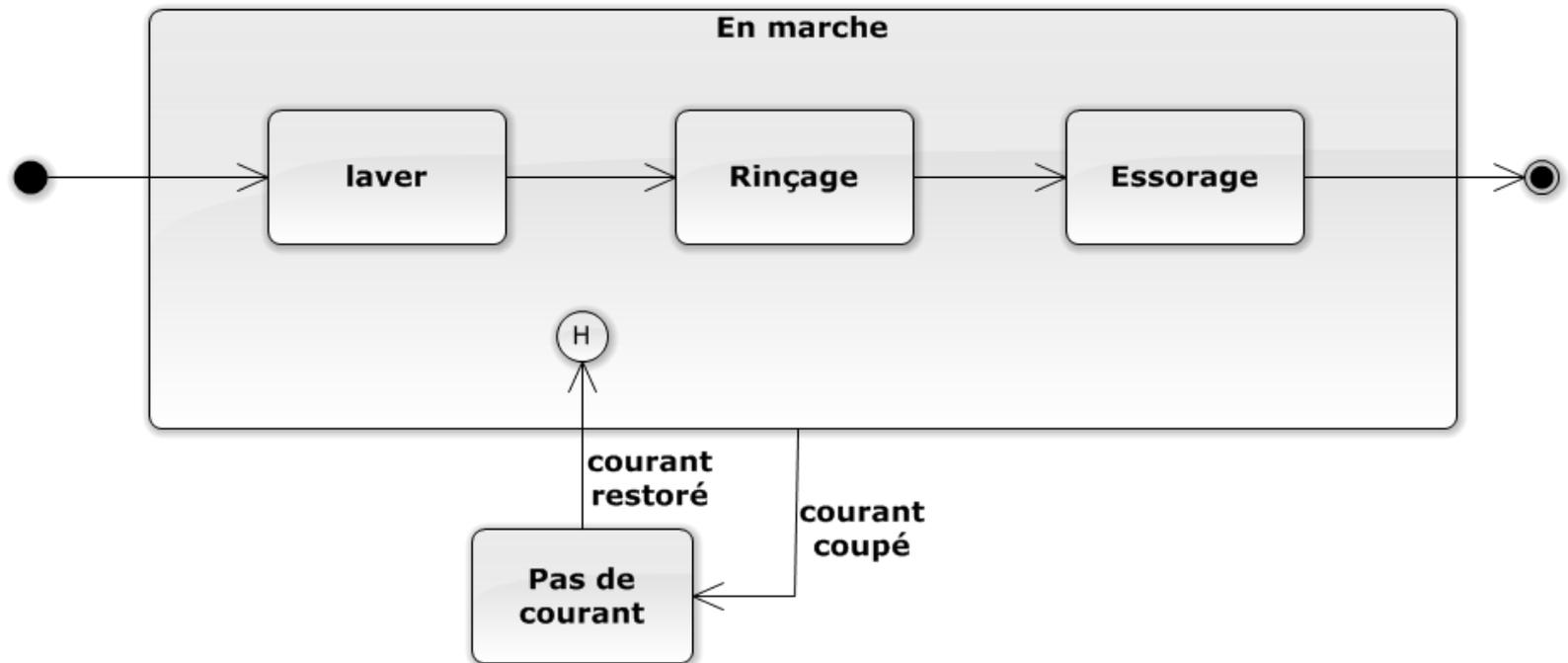
NomEtat
entry / activité1
do / activité2
exit / activité3

Hiérarchie d'états-transition

- **Etat composite**
 - Graphiquement décomposé en N sous-états
 - Récursivité *illimitée*
- **Transition et état composite**
 - Sortant d'un état composite (source) : tous les sous-états de la source sont franchissables
 - Entrant dans un état composite (cible) : équivalent à une transition ayant pour cible l'état initial de l'état composite

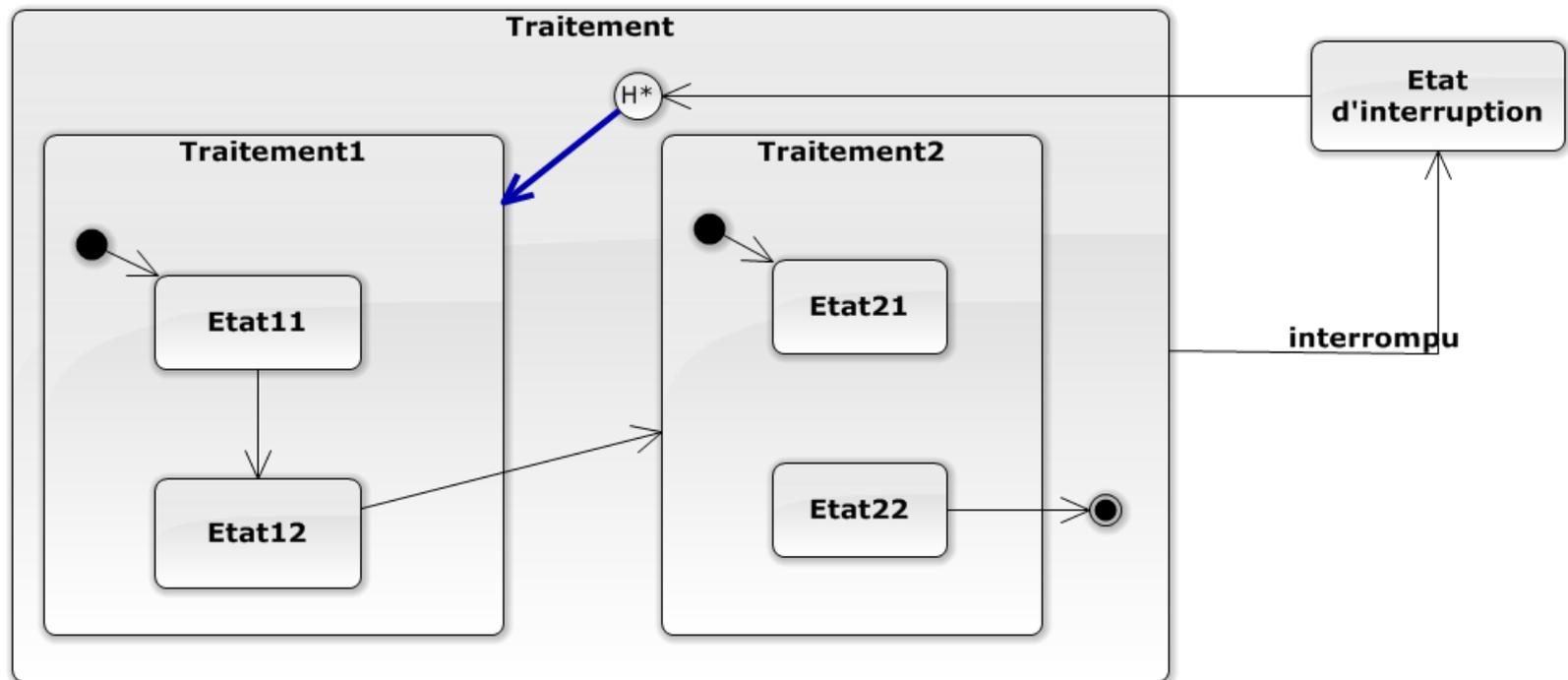
Hiérarchie d'états-transition

- **Historique** et état composite
 - Permet de trouver l'état avant l'interruption (l'état composite d'un **seul niveau**)



Hiérarchie d'états-transition

- **Historique profond**
 - Retrouver l'état avant interruption de n'importe quel niveau



Gestion de concurrence : parallélisme

- **Régions concurrentes**

- Permettent de représenter des zones où l'action est réalisée par des flots d'exécutions **parallèles**
- Chaque région peut avoir un état initial et final
- Transition complexe de type **fork** **=====**
- Transition complexe de type **join** **=====**
- **Similaires à ceux du diagramme d'activité**

