



Modélisation UML

6- DIAGRAMME D'ETATS - TRANSITIONS

PRESENTATION

- **Pourquoi ?**
Représenter la façon dont évoluent les instances d'une classe (cycle de vie)
- **Quoi ?**
Séquences possibles d'états et d'action en réaction à des événements
- **Outil ?**
*Automates à états finis (graphe états-transitions)
« machine dont le comportement des sorties ne dépend pas uniquement des entrées, mais aussi de l'historique des sollicitations passées »*
- **Modélisation de la vue dynamique du système**
avec diagramme de séquence

ETATS

- **Définition**

*conjonction instantanée des valeurs des attributs d'un objet
caractérise par sa durée et sa stabilité*

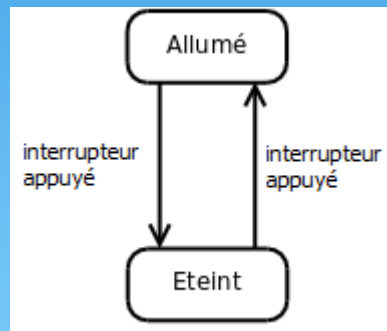
A UML state box diagram, which is a rounded rectangle with a thick black border. The word "Etat" is written inside the box.

Etat

- **Règle de nommage**

*participe passé (vérifié, refusé, allumé, ...)
nom commun (en attente, en vente, ...)*

ETATS - *exemple*



ETAT INITIAL – ETATS FINALS

- **Etat initial**

*indique l'état de départ du diagramme d'états-transitions
état de l'objet à sa création*



- **Etat(s) final(s)**

indique que le diagramme d'état (ou l'état enveloppant) est terminé



EVENEMENTS

- **Définition**

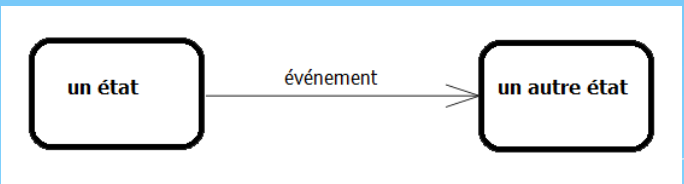
quelque chose qui se produit durant l'exécution du système et qui mérite d'être modélisé

- **Caractéristiques**

- se produit à un instant précis
- dépourvu de durée

- **Conséquences de la réception d'un événement**

- déclenchement d'une transition
- passage de l'objet dans un nouvel état



TYPE D'ÉVÉNEMENTS

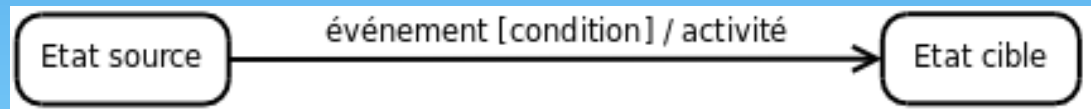
- **Signal (*signal*)**
événement asynchrone (ex : interruption E/S, clavier,...)
ex: **clavier** (*caractère = « F1 »*)
- **Appel (*call*)**
réception de l'appel d'une méthode par un objet
même syntaxe qu'un signal
ex: **acheter**
- **Changement (*change*)**
passage à VRAI d'une condition
ex: **when** (*nombreEssais = 3 ET codeErroné*)
- **Temporel**
fonction du temps absolu (*date*) ou relatif (*durée*)
ex: **after** (*2min*)
when (*date = « 24/12/2014 »*)

TRANSITIONS

- **Réponse d'un objet à la survenue d'un événement**
*lie généralement deux états (transition externe)
non déterministe si deux événements simultanés*
`<événement> [<garde>] / <activité>`
- **Condition de garde**
expression logique sur les attributs de l'objet et/ou paramètres de l'évt
FAUX *la transition ne se déclenche pas*
VRAI *la transition se déclenche et ses effets se produisent*
- **Activité**
action en résultat de l'événement
 - une opération primitive (assignation)
 - l'envoi d'un signal
 - l'appel d'une opération
 - une liste d'activités, ...*formalisme libre*

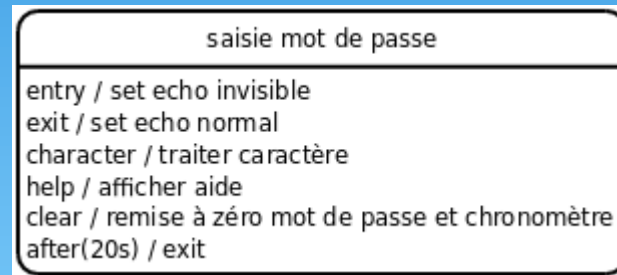
TRANSITION EXTERNE

- **Entre deux états**
état source – état cible
- **La plus fréquente**



TRANSITION INTERNE

- Pas d'état source, pas d'état cible (*même état*)
- *Représentation*



- **Mots clés**
 - *entry* : action lors de l'entrée dans l'état
 - *exit* : action quand on sort de l'état
 - *do* : action s'exécution à la fin de l'action entry.
 - *include* : invocation d'un sous-diagramme d'état

TRANSITION D'ACHEVEMENT

- Se produit à la fin des actions contenues dans l'état source
- Pas d'événement déclencheur explicite

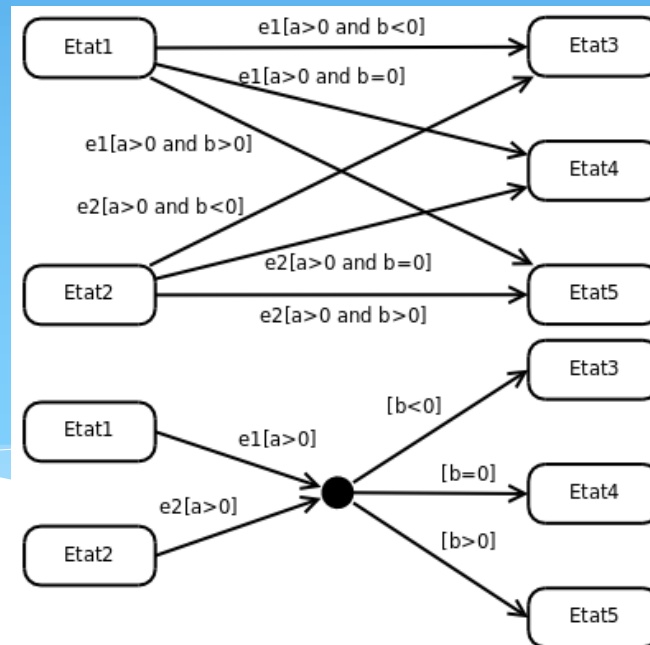
POINT DE CHOIX

- Alternatives pour le franchissement d'une transition
- Points de jonction
- Point de décision



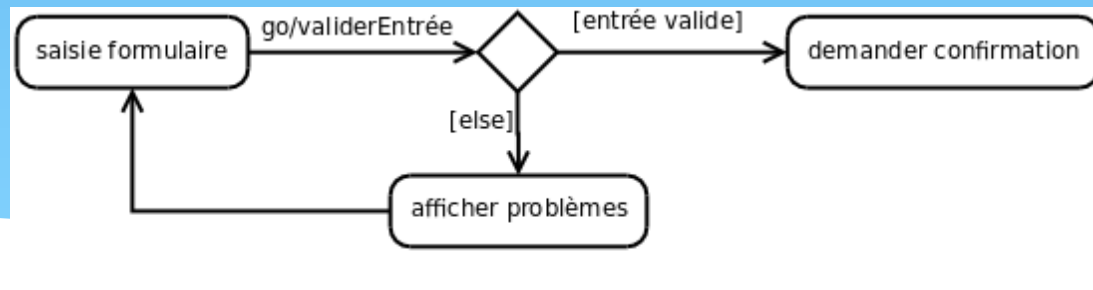
POINT DE JONCTION

- **Partage de segments de transition**
diagramme plus lisible
- **Evaluation des gardes du chemin (entrées et sorties)**
toutes vraies dès le départ



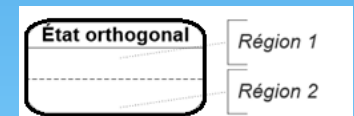
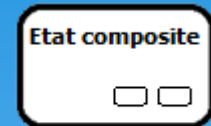
POINT DE DECISION

- Une entrée et au moins deux sorties
- Evaluation des gardes du chemin (entrées et sorties)
gardes de sorties évaluées une fois le point de décision atteint
- Garde [else]

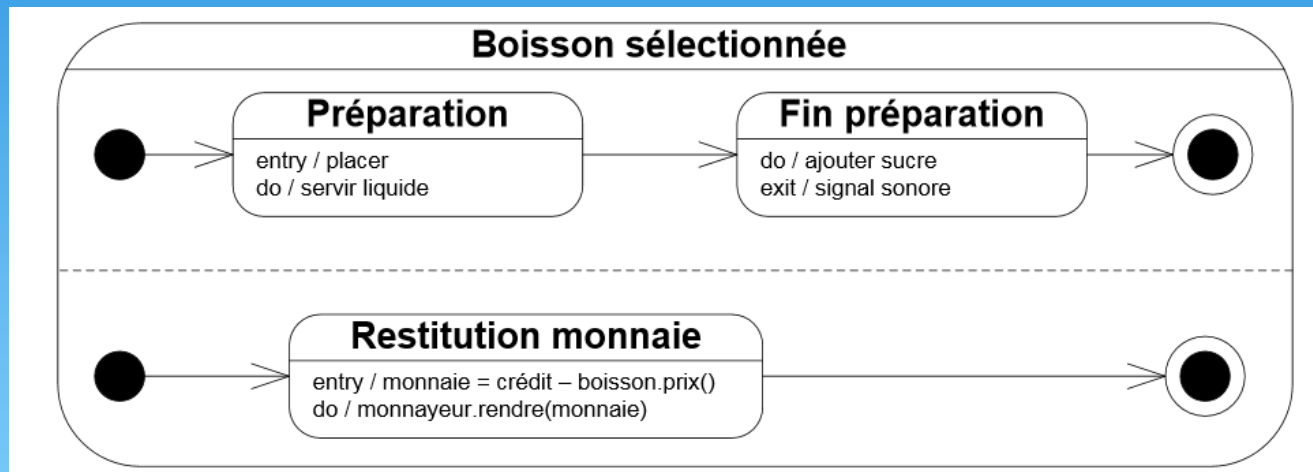


ETATS COMPOSITE

- **Etat composé d'une ou plusieurs régions**
- **Région** composé de plusieurs sous-états
dans chaque région : sous état initial, final
- **Etat orthogonal**
état composite contenant plus d'une région
- **Etats concurrents**
sous-états de deux régions différentes (un objet peut etre simultanément dans plusieurs états)
- **Intérêt**
raffinements successifs

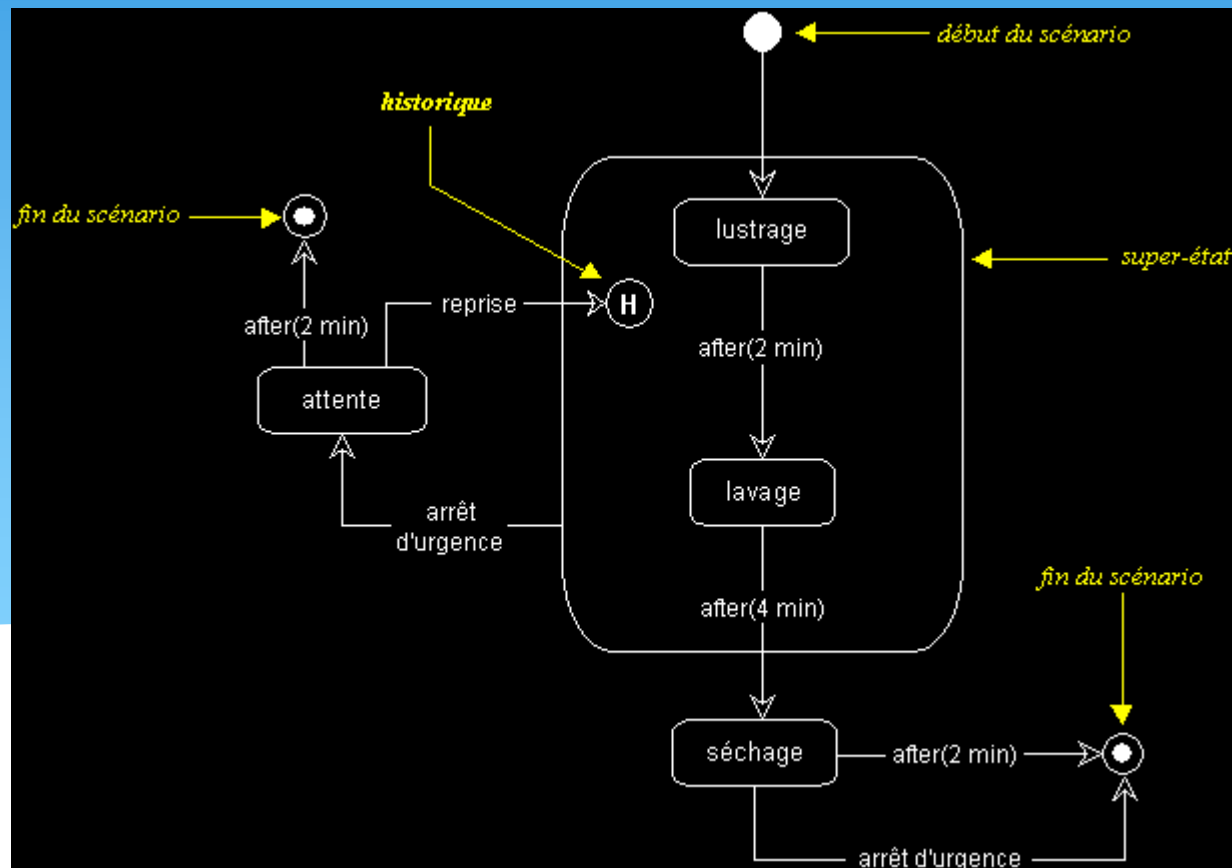


ETATS COMPOSITE - *exemple*



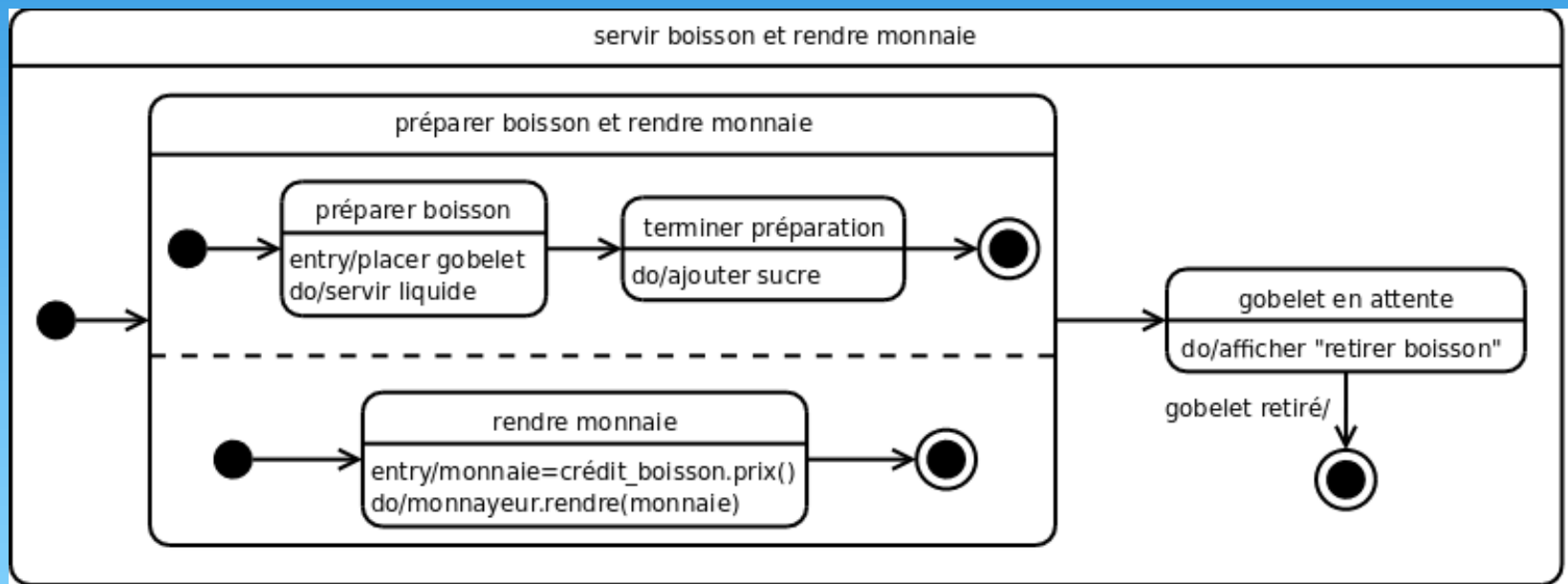
HISTORIQUE

- **Mémoire le dernier sous-état actif d'un état composite permet d'y revenir par la suite**



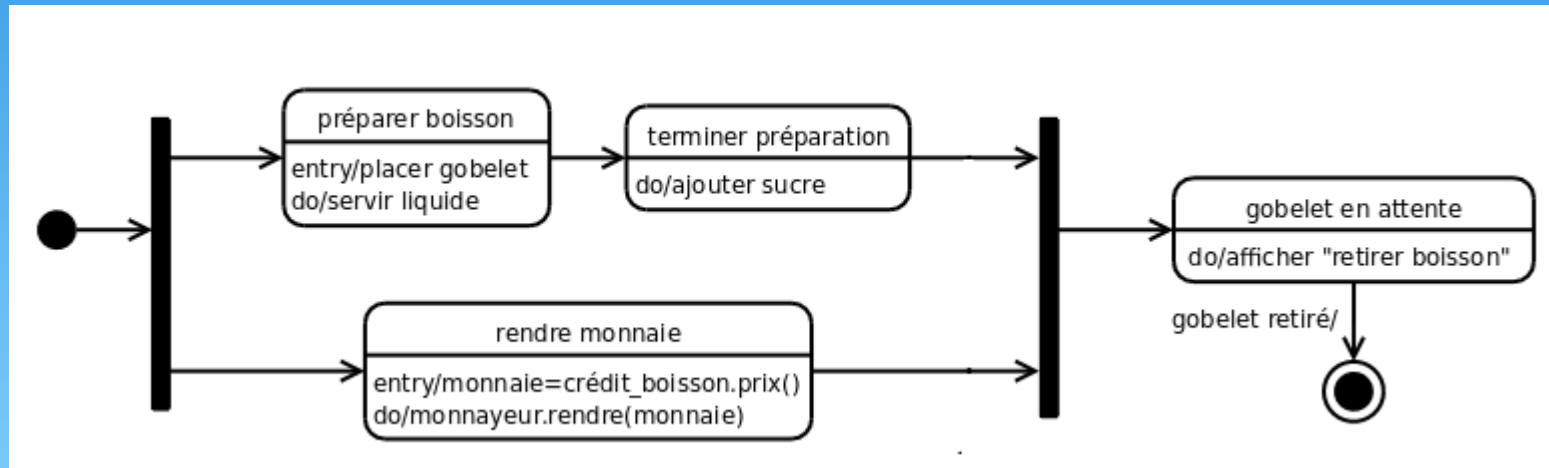
SYNCHRONISATION – V1

- Etat composite avec deux régions



SYNCHRONISATION – V2

- Transitions concurrentes (*souches*)





Modélisation UML

DIAGRAMMES D'ETATS - TRANSITIONS

EXERCICES

ECHECS

Considérons une classe Partie dont la responsabilité est de gérer le déroulement d'une partie de jeu d'échecs. Cette classe peut être dans deux états :

- le tour des blancs*
- le tour des noirs.*

Les événements à prendre en considération sont

- un déplacement de pièces de la part du joueur noir*
- un déplacement de pièces de la part du joueur blanc*
- la demande de prise en compte d'un échec et mat par un joueur*
- un échec et mat assure la victoire du dernier joueur. Dans ce cas, une activité « noirsGagnants » ou « blancsGagnants » est déclenchée*
- la demande de prise en compte d'un pat qui mène à une fin de partie, avec une égalité. Dans ce cas, une activité « égalité » est déclenchée.*

Représenter le diagramme d'états/transitions associé à la classe Partie.

MONOPOLY - Terrain

Décrire par un diagramme états-transitions les états par lesquels peut passer un terrain au jeu du Monopoly

Indication : on se concentrera sur les points suivants :

- à vendre / acheté
- nu / bati (de 0 à 4 maisons, un hotel)
- hypothéqué