

# *Diagrammes d'états*



# Rôle

- Jusqu'à présent, on a vu :
  - les besoins (uses cases, scénarios)
  - l'aspect statique (diagramme de classes)
  - l'aspect dynamique entre composants (scénarios)

⇒ manque :

- aspect dynamique d'un objet

⇒ rôle des diagrammes d'état



# Notion d'Etat

- Notion primaire :
    - Définition :

L'état d'un objet caractérise l'ensemble des valeurs de ses champs à un instant donné
    - Exemple :

si un enfant est caractérisé par la couleur de ses cheveux et son âge, alors <blond, 15> est un état, <brun, 14> est un autre état
  - Notion abstraite (utilisée par la suite) :
    - Définition :

un état représente un ensemble d'états au sens primaire donné ci-dessus
    - Exemple :

si un enfant est caractérisé par la couleur de ses cheveux et son âge, alors « adolescent » (  $13 \leq \text{âge} \leq 19$ ) peut être un état
- 
-

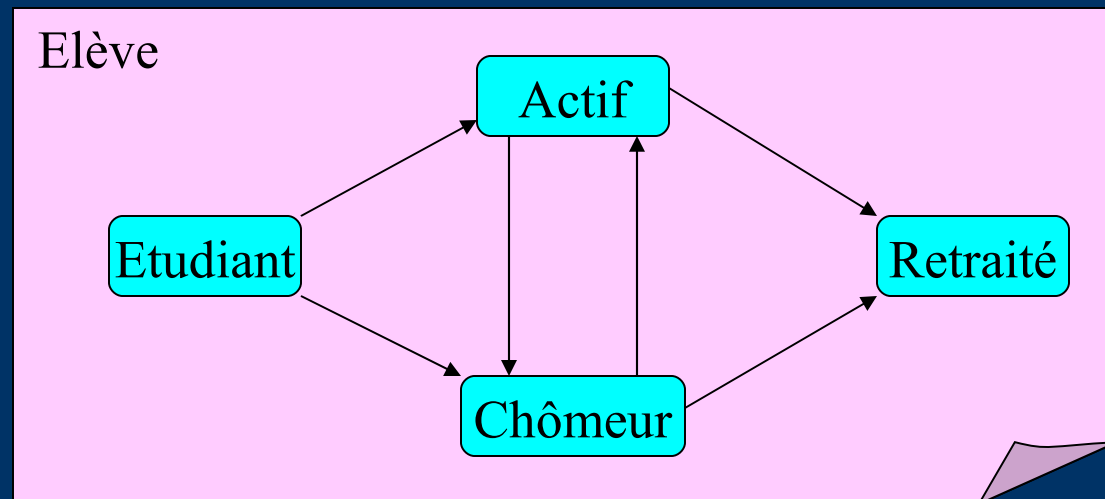
# *Particularité des états*

- Intérêt de la notion abstraite d'état :
  - éviter de multiplier le nombre d'états
  - se concentrer sur les caractéristiques essentielles intervenant dans le comportement d'un objet
- En règle général, des états sont :
  - soit disjoints
  - soit imbriqués

⇒ pas de recouvrement partiel

# Les diagrammes d'états

- Au plus un diagramme par classe
- Pour représenter
  - les états possibles des objets de la classe;
  - les transitions possibles entre ces états.
- Exemple (simplifié) :



# Les types d'états

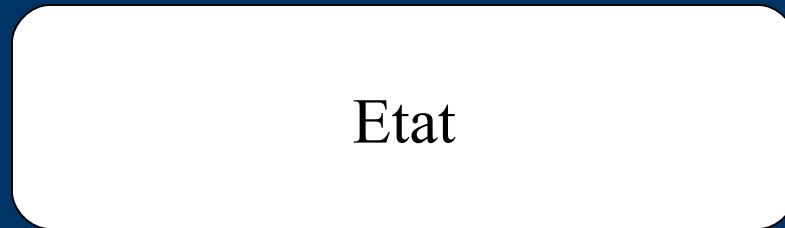
- Etat initial :
  - indispensable
  - tout objet commence dans cet état
  - notation : ●
- Etat final :
  - pas indispensable
  - passage obligé pour tout objet à détruire
  - notation : ⊙
- Etat intermédiaire :

Nom

N.B. : nom pas indispensable,  
mais **FORTEMENT** conseillé

# *Représentation des états intermédiaires*

- Deux représentations possibles :

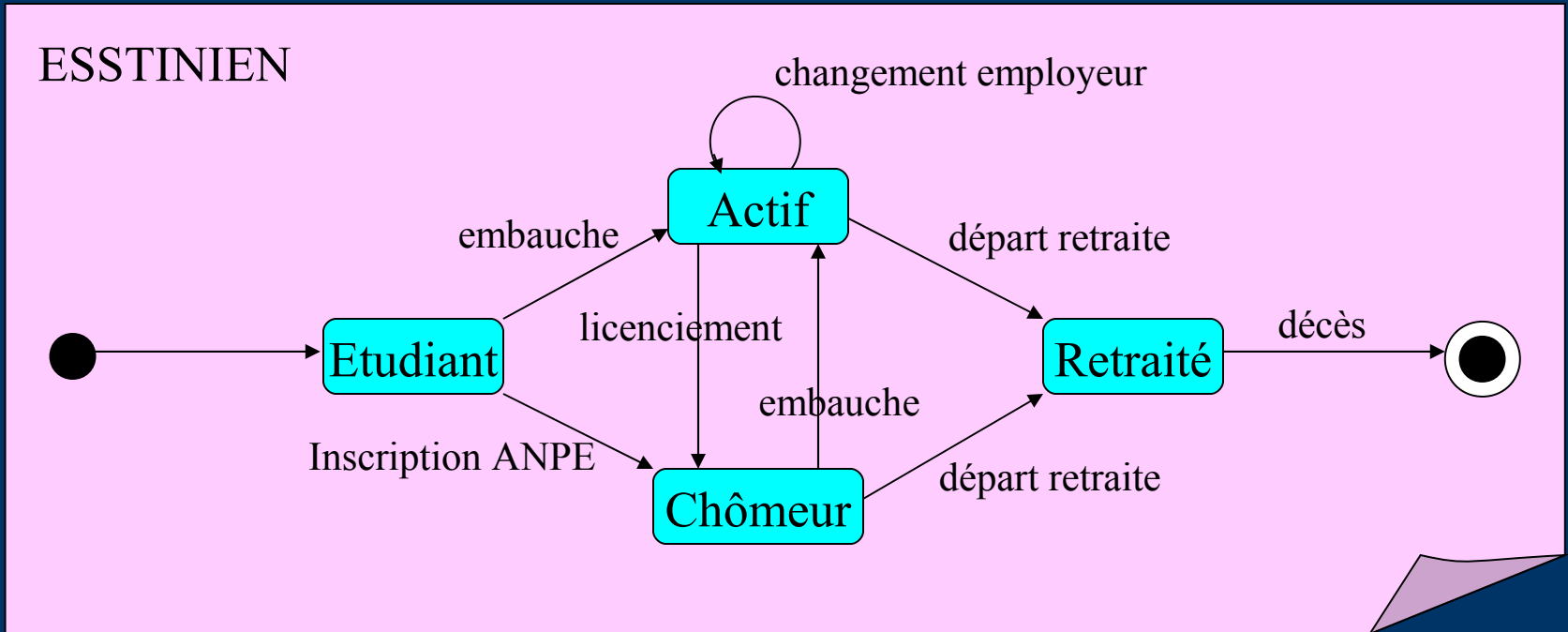


# Transitions et événements

- Transition :
    - Définition  
passage potentiel d'un état à un autre (éventuellement le même)
    - Représentation :  $\longrightarrow$
  - Événement :
    - (souvent) : message venant de l'*extérieur* de l'objet
    - (souvent) : figure sur les diagrammes de séquence
  - En général, un événement est associé à une transition
- 
-



# Exemple complété



# Événements : approfondissement

- Notation :

nom(paramètres) [condition]/action

Optionnel      Optionnel      Optionnel

- Explications

- Condition :

doit être vérifiée pour que l'événement soit pris en compte

- Action :

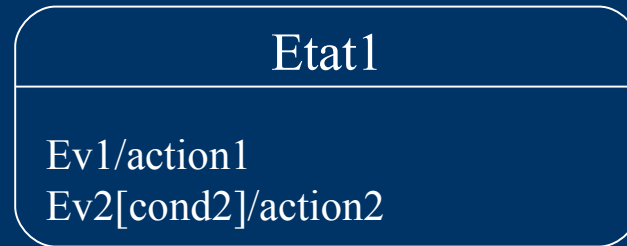
décrit une action à exécuter (souvent, une méthode de la classe) si l'événement survient et doit être pris en compte

---

---

# Transitions internes

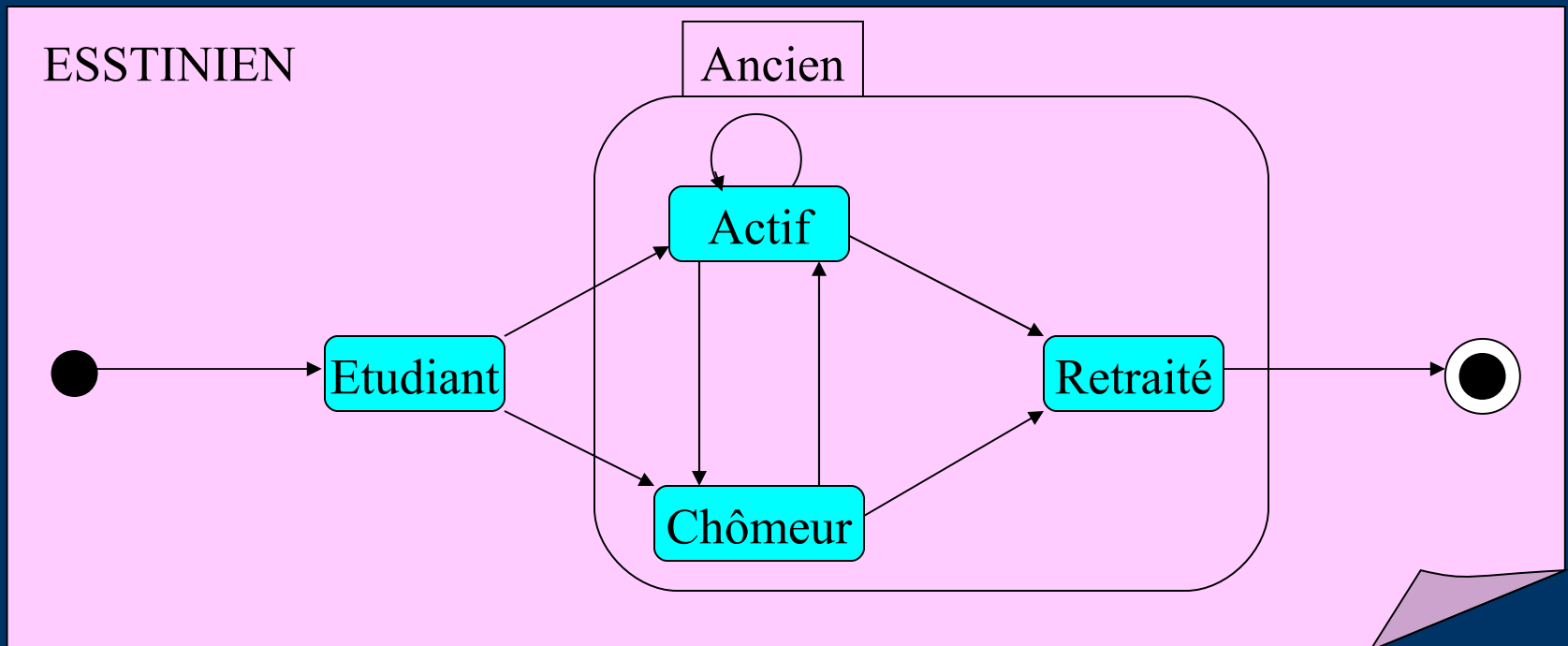
- Notation :  
comme des événements à l'intérieur d'un état.
- Exemple :



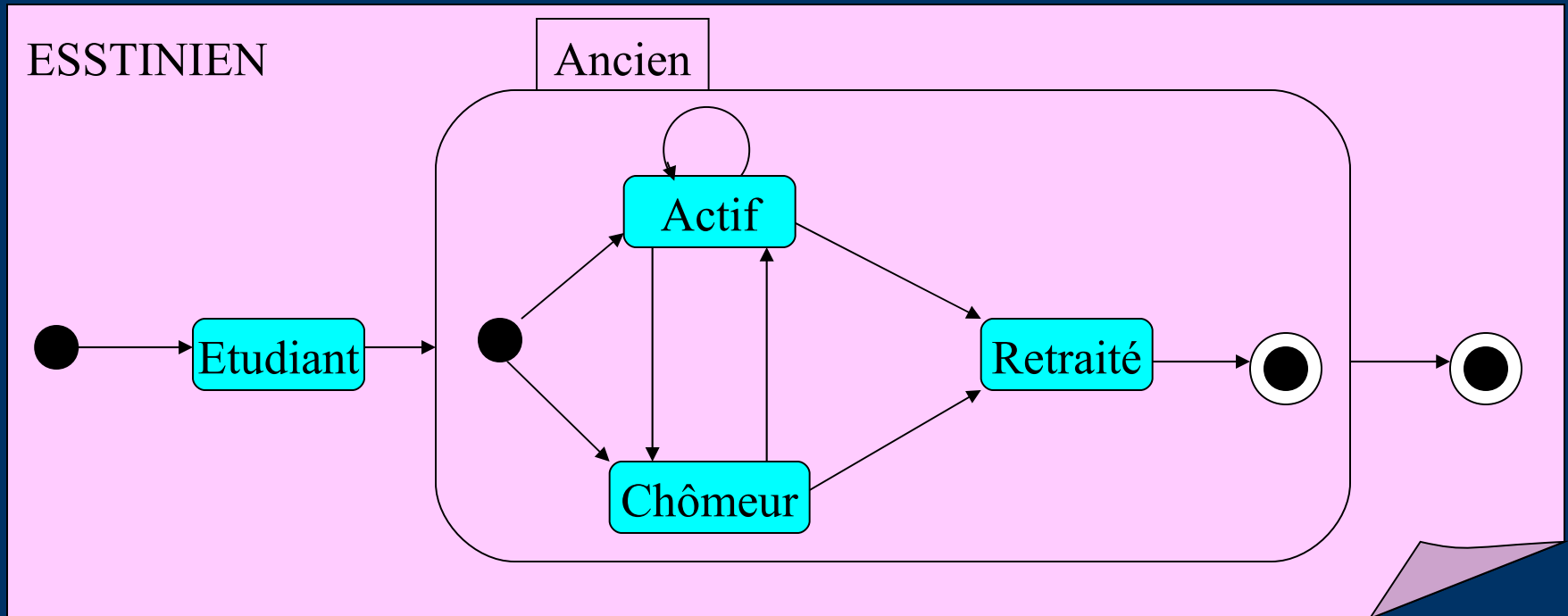
- Evénements particuliers :
  - entry : dès que l'on rentre dans l'état (par transition externe)
  - exit : dès que l'on quitte l'état (par transition externe)
  - do : l'action associée est exécutée tant que l'on est dans l'état

# Etats composites : Introduction

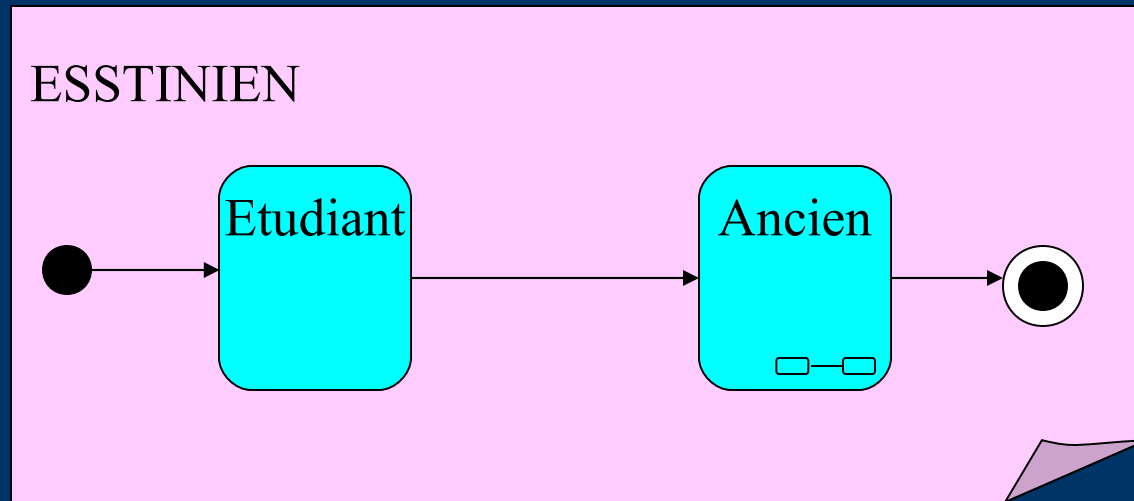
- Principe :  
Un état peut être composé de plusieurs sous-états.
- Exemple :



# Etats composites : Représentation conseillée

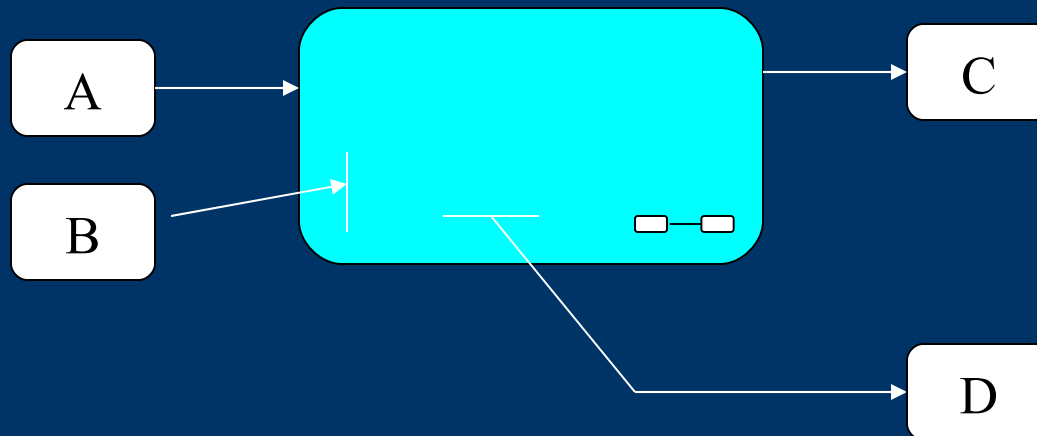


# *Etats composites :* *Représentation abstraite simple*



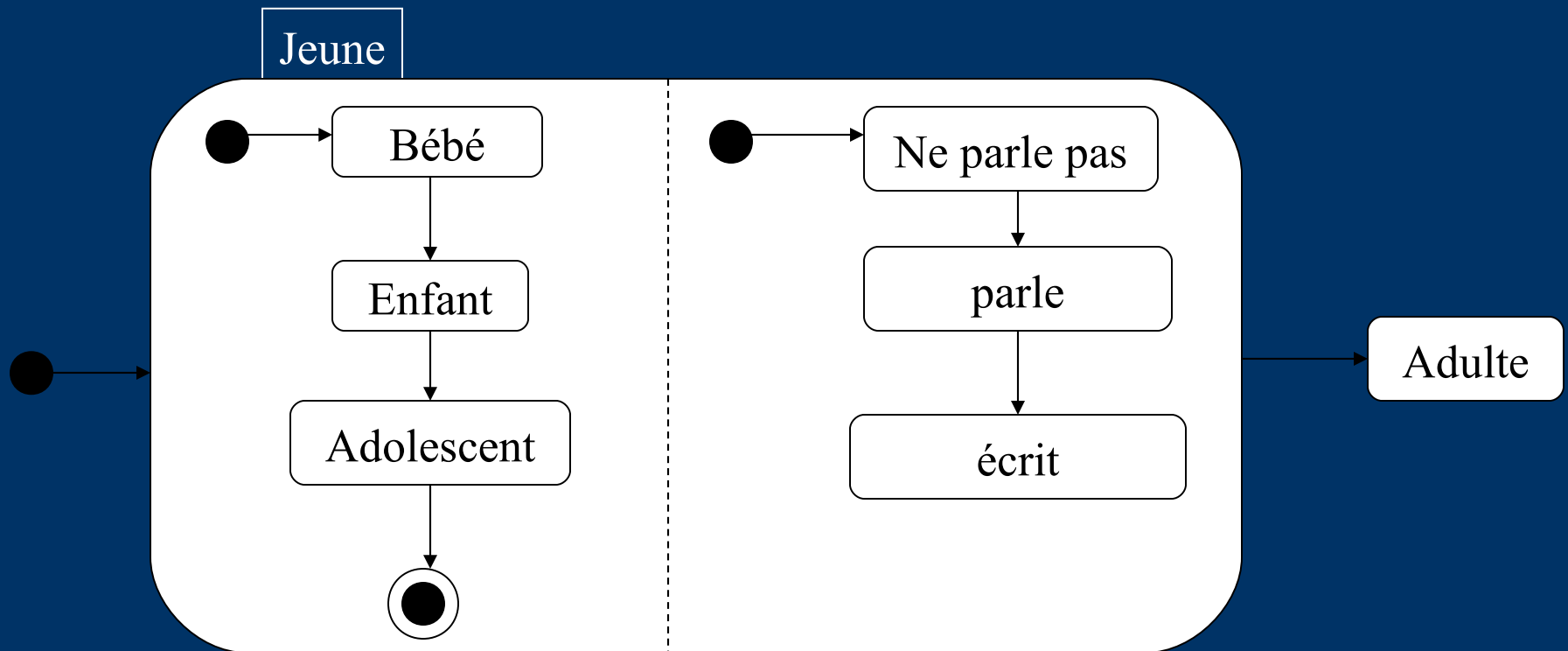
# *Etats composites :* *Représentation abstraite* *générale*

- Principe :
  - s'il y a entrée ou sortie directe avec des sous-états, ceux-ci sont représentés par de simples traits
- Exemple :



# États composites multi-régions (1)

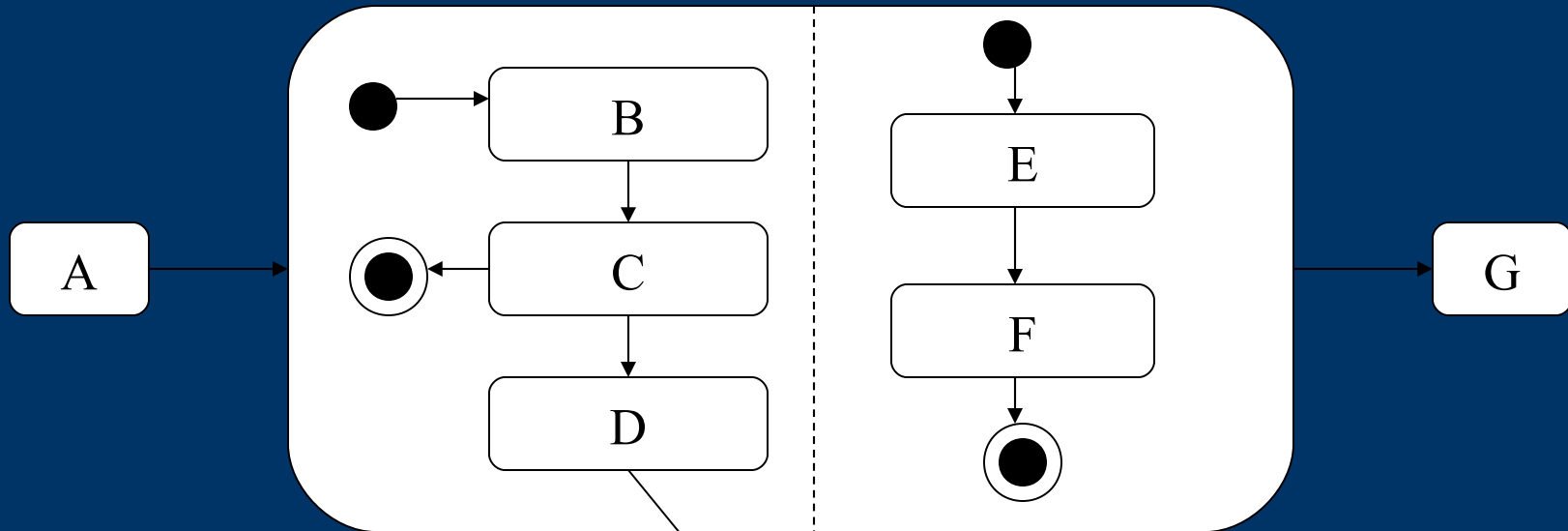
- Principe :  
état = composition (simultanée) de plusieurs sous-états
- Exemple :





# États composites multi-régions (2)

- Nouvel exemple :



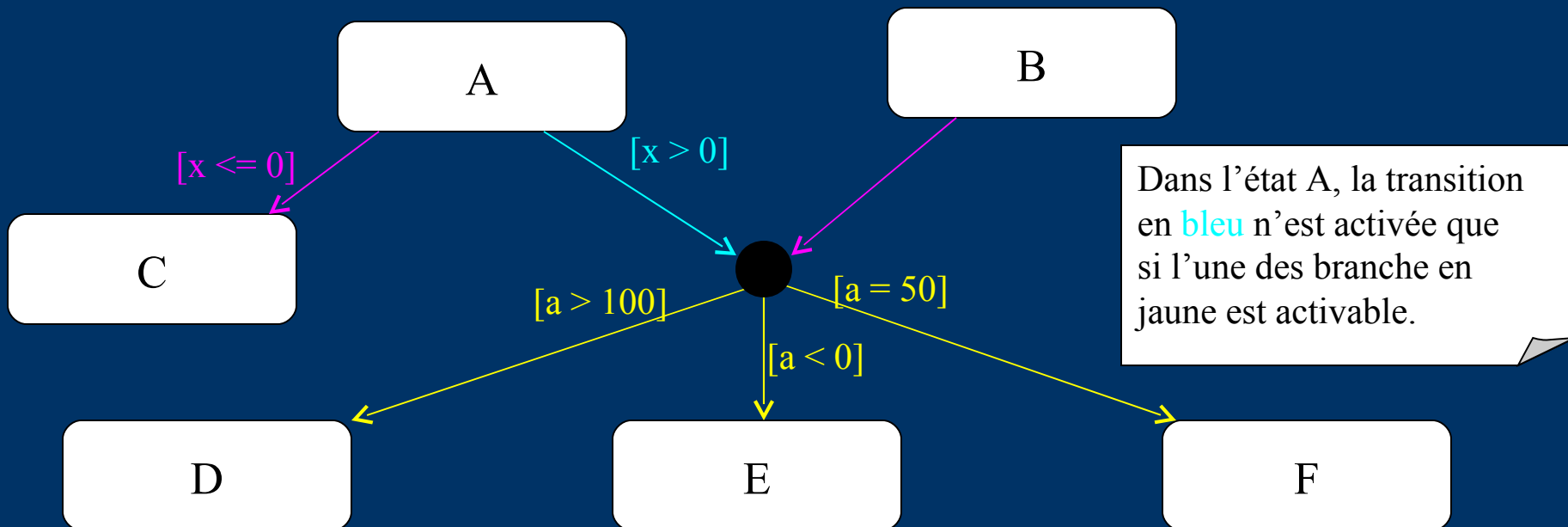
- Chemins possibles :

A, (B,E), (B,F), G  
A, (B,E), (C,E), G  
A, (B,E), (C,E), (D,E), (D,F), G  
A, (B,E), (C,E), (D,E), H



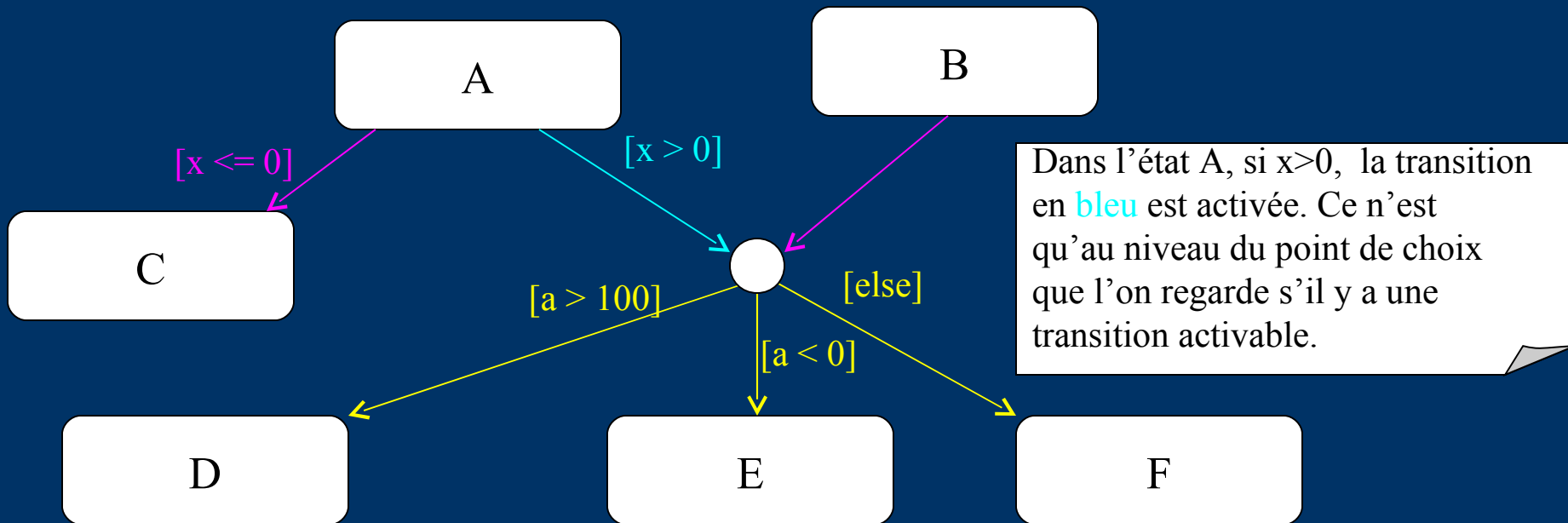
# Point de jonction

- But :  
regrouper ou diviser des transitions
- Exemple :

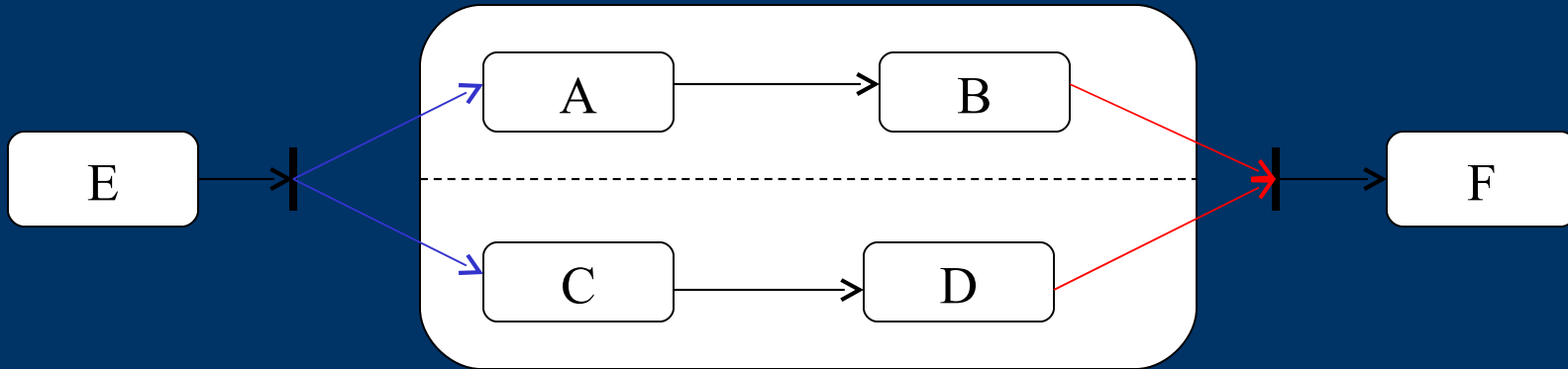


# Point de choix dynamique (UML1 ; UML2?)

- But :  
regrouper ou diviser des transitions !
- Exemple :

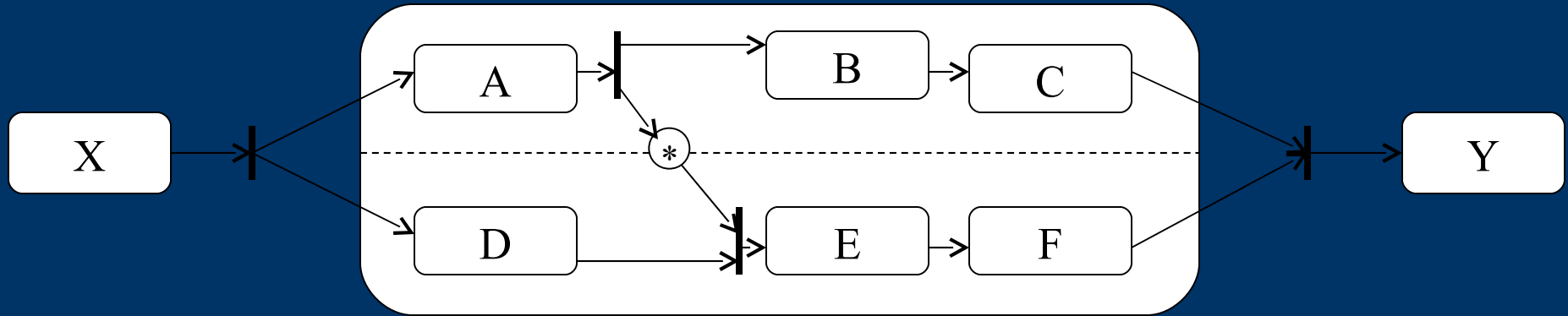


# Transitions simultanées



- Les états A et C sont atteints simultanément ;
- Les états B et D sont quittés simultanément.

# Synchronisation (UML1 ; UML2?)



- **Interprétation :**

On ne peut quitter D pour E que si dans le même temps, on quitte A



# Choix

