



# Analyse orienté objet

*Diagramme d'états*



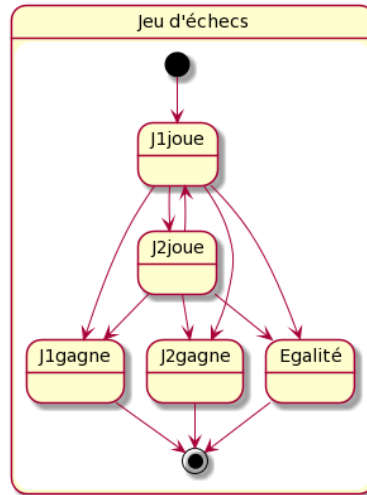
## Définition

- Un diagramme d'états permet de représenter les états d'une entité, les transitions entre ces états ainsi que les éventuelles conditions.



# Exemple

- Jeu d'échecs:



Source:

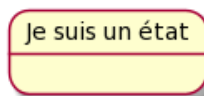
<https://www.planttext.com/?text=SoWkIImgAStDuU82iafI5PIyKaiLKjGFhqpEI4qkLb98B5O0iefBYgkvkAArOXLqTUqKI0opyajJkl0KHCG88cA4486eGL5ac9wOdeUc3qF8DJqn9xFay4ekLDEWwemmjHfXqmfH3nO5kfeBgXZEXwc42zHoEQJcfO320G00>

# Caractéristiques

- Il y a 5 représentations dans un diagramme d'états:
  - Etat
  - Transition
  - Evènement
  - Action/Activité
  - Garde

# Etat

- Notion d'état: modélise une situation durant laquelle un ensemble de conditions ne varie pas
- En UML, on représente un état avec un rectangle aux coins arrondis



Source:

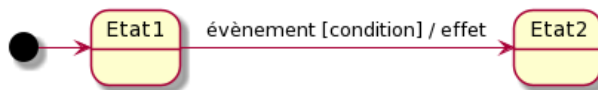
<https://www.planttext.com/?text=SoWkIImgAStDuU82iafI5PIyKXMAIpEB5Khp50wl10efAIGMAoHdf1QN57CvfEQbWDG20000>

# Transition

- Notion de transition: représentation d'un changement d'état (source) vers un autre état (cible)
- En UML, on représente une transition avec une flèche qui part d'un état source vers un état cible.

# Transition

- Une transition est composée de:
  - Un évènement (le déclencheur – obligatoire)
  - Garde (une condition – optionnelle)
  - Effet (comportement à effectuer après la transition – optionnel)
- Notation: Evènement [Garde] / Effet



Source:

<https://www.planttext.com/?text=SoWkIImgAStDuUAArefLq2tAJCyeqRLJS2r9B37a0fEeWaOALWg7LvOTNf6NcfkQLw8GdPoVbv9PafcV5wkWhv2QbfPQmiNba9gN0Z8J000>

0

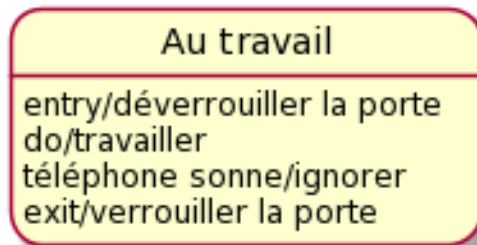
# Transition

- Les transitions internes existent également. Il s'agit d'une transitions où l'état ne change pas.
- Ils ont la même notation qu'une transition
- Ils n'ont pas de flèche, mais ils sont représentés dans l'état



# Transition

- Exemple:



Source: [https://www.planttext.com/?text=ROx13SCm30F\\_daB80aQ-VIMZ2B3G6b2bG9A3T2JFuSMQ1ddsH\\_18Wei7MhGtftlg89ZM1c6uO-K9qC7dvFC3IC8-kOo-avcsoam6ZB2f1Ls9eldkdUXoOdGUVNkf4BYAKAvFKRif7JNotxM5fFmUVG40](https://www.planttext.com/?text=ROx13SCm30F_daB80aQ-VIMZ2B3G6b2bG9A3T2JFuSMQ1ddsH_18Wei7MhGtftlg89ZM1c6uO-K9qC7dvFC3IC8-kOo-avcsoam6ZB2f1Ls9eldkdUXoOdGUVNkf4BYAKAvFKRif7JNotxM5fFmUVG40)

# Transition

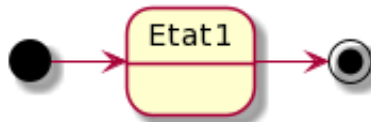
- Cependant, il existe 3 transitions réservées:
  - Entry: activité exécutée quand on rentre dans l'état
  - Do: une activité qui commence dès que « entry » est fini et continue d'être exécuté tant que l'état reste le même
  - Exit: activité exécutée quand on sort de l'état



# Pseudo-états

- Il existe deux pseudo-états:
  - Initial
  - Final
- Le premier représente l'initialisation. Sans lui, impossible de connaître le premier état
- Le second représente la terminaison. Sans lui, impossible de savoir si le système a un état final (non obligatoire)

# Pseudo-états



12



Source:

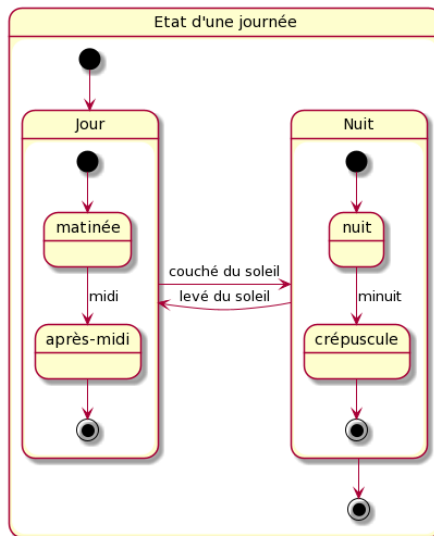
<https://www.planttext.com/?text=SoWkIImgAStDuUAArefLq2tAJCyeqRLJS2r9B37a0fC8GQ0ABYukXzly59060000>

# Type d'état

- Il existe des sous types d'état:
  - Etat simple (celui vu précédemment)
  - Etat composite:
    - Simple: définition de sous-états et de leurs transitions
    - Orthogonal: division en régions avec exécution en parallèle
  - Sous-machine (illustration plus loin dans le cours)

# Etat composite

- Simple:



14

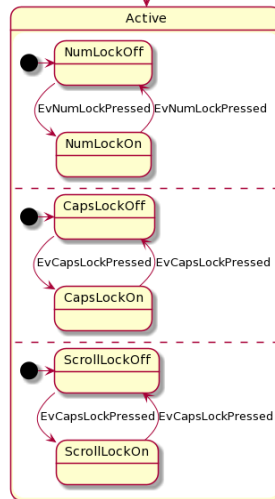


Source:

[https://www.planttext.com/?text=TP5D2i8m48NtESNGZI3q0bs86pT2fNjn4PBHHj9KykEcz3xs7BsOoMXhHSmYFzzxBpCXM-UvzQ7HZCMTHyXsSG6v2WRXsWPhnW4pu0v8BhcebS6ECOZZbTX7LqVd3ylupOuFbpTAAafNvUmuha-Gvnjek5Vfzbc00AbLMK1ApsfL4ezfGZtDYpOEGVdkfuY9T8RfG5JOSRW59u96i6M-N4lnJyNfzIHq0ca3kLMNsaUSMYb0j47Kum0oW6irAatEO\\_MWYUpV-G8qtl\\_OtjLPpnZRef7fdvu0](https://www.planttext.com/?text=TP5D2i8m48NtESNGZI3q0bs86pT2fNjn4PBHHj9KykEcz3xs7BsOoMXhHSmYFzzxBpCXM-UvzQ7HZCMTHyXsSG6v2WRXsWPhnW4pu0v8BhcebS6ECOZZbTX7LqVd3ylupOuFbpTAAafNvUmuha-Gvnjek5Vfzbc00AbLMK1ApsfL4ezfGZtDYpOEGVdkfuY9T8RfG5JOSRW59u96i6M-N4lnJyNfzIHq0ca3kLMNsaUSMYb0j47Kum0oW6irAatEO_MWYUpV-G8qtl_OtjLPpnZRef7fdvu0)

# Etat composite

- Orthogonal



Source: <https://plantuml.com/fr/state-diagram>

# Etat composite

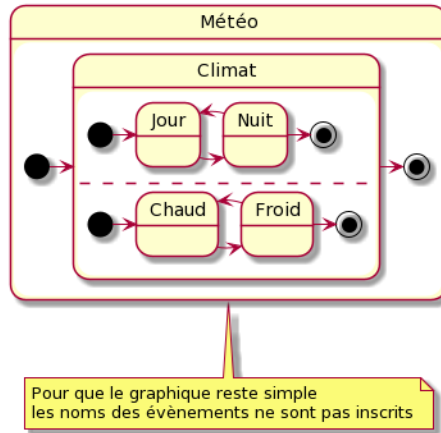
- Orthogonal:
  - Chaque « région » est indépendante
  - L'état « mère » est terminal seulement quand les « enfants » sont tous en état terminal





# Etat composite

- Orthogonal:



17



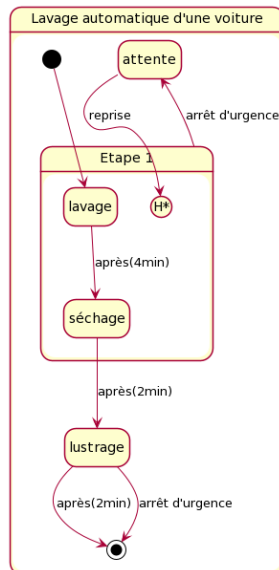
Source: [https://www.planttext.com/?text=PP51QWCn34NtEeNM1L-X109T15gwBrqu6iLZiAsfBMSJUf\\_CEUPYbJsPJ4EyyTVJbuJiJHQRf0GFe4e8twTHff4IlxprBbYv0EhvUIb6azof5\\_EAEowfqIfM\\_56SD5m56a\\_71\\_DC5xCsRDIOv\\_xRtfQkuQRMn5jYDoUQkizO2sxy\\_vHVWCVkRIsOhoThPDrypo8Sa8\\_Bcm1-rbL\\_2g4dF2KxzAu6YRBQimk39z1KniWXOWTY6i\\_JDLAWA4hLn55mi1bTp8Va90F53kiusAYg\\_\\_07](https://www.planttext.com/?text=PP51QWCn34NtEeNM1L-X109T15gwBrqu6iLZiAsfBMSJUf_CEUPYbJsPJ4EyyTVJbuJiJHQRf0GFe4e8twTHff4IlxprBbYv0EhvUIb6azof5_EAEowfqIfM_56SD5m56a_71_DC5xCsRDIOv_xRtfQkuQRMn5jYDoUQkizO2sxy_vHVWCVkRIsOhoThPDrypo8Sa8_Bcm1-rbL_2g4dF2KxzAu6YRBQimk39z1KniWXOWTY6i_JDLAWA4hLn55mi1bTp8Va90F53kiusAYg__07)

# Historique

- L'historique est un moyen de « sauvegarder » l'état.
- Il y a 2 types d'historique:
  - Plat: qui enregistre uniquement l'état du niveau actuel  $\rightarrow H$
  - Profond: qui enregistre l'état peu importe le niveau  $\rightarrow H^*$



# Historique



19



Source:[https://www.planttext.com/?text=ZP4zRW8n44Pxds944u94mR8LHKI3H94R88hHxWYihBrcF4P24VT9Sesz6FxpJLAal3E5tppnlRM2x84qodbtxHro6Z0owsZM5at6Prrlq36oGrQyWrh9xgtCI4e1BDNIEA908Fq1aLV0a7x52p1jTSIc6Q07lpNG70644HiZAI-XVO2Jg2LG5LZln0FtmsvwWgT3t2hkYNovTujW5qF7puUMsqVPxYTQQ7\\_UAu0IR7sbD2zrGEYsEcPKnUfih3wZYCUVYIT1q-aMqANfV-41ye9hwDrMISl94cvpyw\\_X2fkzgIRTCFF000](https://www.planttext.com/?text=ZP4zRW8n44Pxds944u94mR8LHKI3H94R88hHxWYihBrcF4P24VT9Sesz6FxpJLAal3E5tppnlRM2x84qodbtxHro6Z0owsZM5at6Prrlq36oGrQyWrh9xgtCI4e1BDNIEA908Fq1aLV0a7x52p1jTSIc6Q07lpNG70644HiZAI-XVO2Jg2LG5LZln0FtmsvwWgT3t2hkYNovTujW5qF7puUMsqVPxYTQQ7_UAu0IR7sbD2zrGEYsEcPKnUfih3wZYCUVYIT1q-aMqANfV-41ye9hwDrMISl94cvpyw_X2fkzgIRTCFF000)

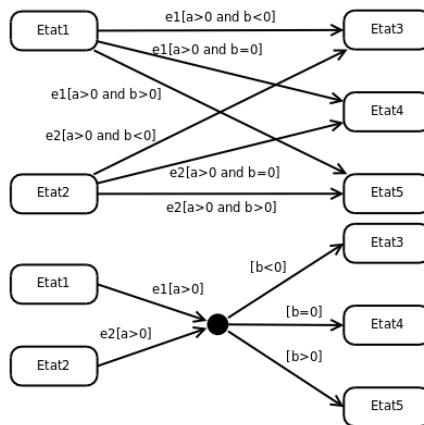
# Historique

- Si on avait remplacé  $H^*$  par  $H$  on devrait reprendre le lavage ou le séchage depuis le début
- Avec un  $H^*$  on reprend le lavage ou le séchage là où il a été interrompu
- PS: lavage et séchage ont leur propre état  
→ Il s'agit de sous-machines à état (schéma simplifié)

## Point de jonction

- Les points de jonction permettent de partager des segments de transition. Le but est de simplifier le chemin
- Attention ! Toutes les gardes sont évaluées dès le début !

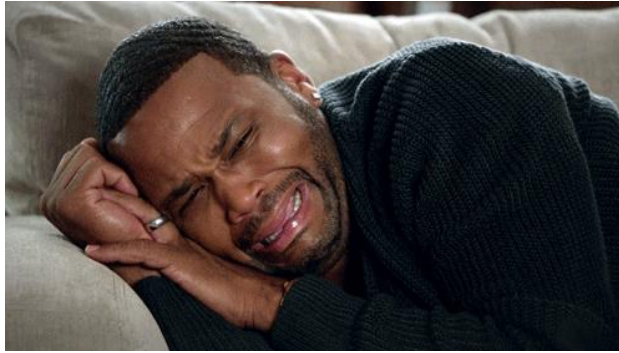
# Point de jonction



Source: [https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-etats-transitions#fig\\_ex\\_point\\_jonction](https://laurent-audibert.developpez.com/Cours-UML/?page=diagramme-etats-transitions#fig_ex_point_jonction)

## Point de jonction

- PS: PlantUML ne permet pas de faire un point de jonction :'(

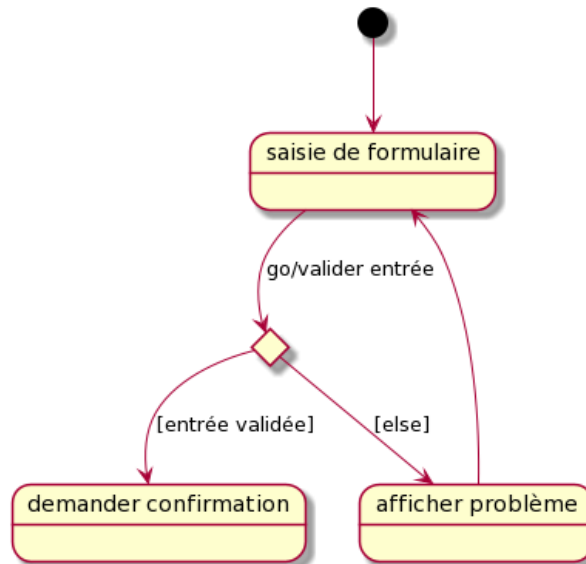


## Point de décision

- Un point de décision ressemble à un point de jonction, sauf que les gardes sont évaluées au moment où on a atteint la garde.
- Notation UML: un carré sur sa pointe (parfois un losange)
- Possède une garde particulière: [else]



# Point de décision



25



Source:

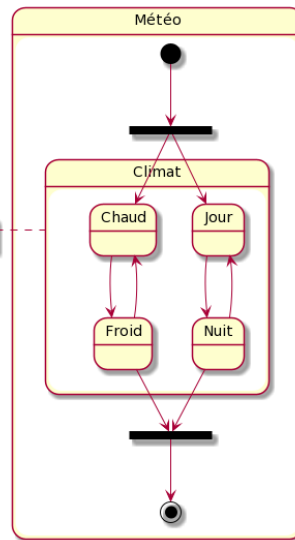
[https://www.planttext.com/?text=LOyz3eCm34RtdC9Yh8GgO4C8SG\\_4u2PEiPIVAWazKtiEBbOWC7Hx-lm-MnvYmf0MQyG6YQ28o949587smlu6EL01625Mbu9Qivmfm2luXra\\_zXJglr1aqQbDaDvf3XOJUvUTvdJax5aIT5s6IXTYl4rGblrng3eeprfu-liR3UyRoQMmVacS\\_SEIMnZ9H9h-qcPFimr7UuD9nFgwCP1J-zy\\_](https://www.planttext.com/?text=LOyz3eCm34RtdC9Yh8GgO4C8SG_4u2PEiPIVAWazKtiEBbOWC7Hx-lm-MnvYmf0MQyG6YQ28o949587smlu6EL01625Mbu9Qivmfm2luXra_zXJglr1aqQbDaDvf3XOJUvUTvdJax5aIT5s6IXTYl4rGblrng3eeprfu-liR3UyRoQMmVacS_SEIMnZ9H9h-qcPFimr7UuD9nFgwCP1J-zy_)

# Fork/Join

- Les « Forks » permettent de représenter une exécution concurrente
- Les « Joins » permettent de représenter l'attente d'une exécution concurrente
- Il s'agit d'une autre représentation d'un état orthogonal
- Ils sont représentés par des barres

# Fork/Join

Pour que le graphique reste simple  
les noms des évènements ne sont pas inscrits



27



Source:

[https://www.planttext.com/?text=NP0nRiCm34Ltde9pWTp0C084w50WGVQWWr3BjHfBT2Iglz3xrEVmnKeoRej4Y\\_v\\_fBv4RGkxp3ME0ABOutwPUPdf-kjxocTi6jtQTcKV598mtOljS3U6wFWAACiERJOjlb3DHBGOETJ0HbJSz\\_mH3ORxeDtWQcV8bB7dJE76J3tq\\_JEB-m8uFRsQrr2m6TG9rM8YcM7qFIFrwoo0HttrfMe1txERXg0c-oAz9SHfz22bWebYmKx4Cdyktyb7drYeD55Yd5p1aCfR3bp0fmxrBjYAqd\\_\\_0G00](https://www.planttext.com/?text=NP0nRiCm34Ltde9pWTp0C084w50WGVQWWr3BjHfBT2Iglz3xrEVmnKeoRej4Y_v_fBv4RGkxp3ME0ABOutwPUPdf-kjxocTi6jtQTcKV598mtOljS3U6wFWAACiERJOjlb3DHBGOETJ0HbJSz_mH3ORxeDtWQcV8bB7dJE76J3tq_JEB-m8uFRsQrr2m6TG9rM8YcM7qFIFrwoo0HttrfMe1txERXg0c-oAz9SHfz22bWebYmKx4Cdyktyb7drYeD55Yd5p1aCfR3bp0fmxrBjYAqd__0G00)

## Exercice

- Faites un diagramme d'état de l'énoncé (slide suivant)
- Utilisez un état composite

## Exercice

Quand un livre arrive dans une bibliothèque, il est marqué comme « à ranger ». A la fin de la journée, il est mis en salle de lecture. Il peut être retiré par un membre pour une lecture à domicile. Il est considéré comme « sorti ». Une fois la lecture fini, il est restitué à la bibliothèque. Un livre dans la salle de lecture peut avoir un ISBN illisible à cause de l'usure. Il est alors réétiqueté et revient dans la bibliothèque. Un livre dans la bibliothèque est reservable pour le lendemain. A la fin de la journée, il est mis de côté. Il peut-être retiré et devient considéré comme « sorti ». Il est également possible que personne ne vienne chercher le livre. On vérifie si l'ISBN est toujours lisible et dans le cas échéant, il est remis à la bibliothèque.



# PlantUML

- Documentation:  
<https://plantuml.com/fr/state-diagram>

