

Introduction à UML

Diagramme d'état

Plan

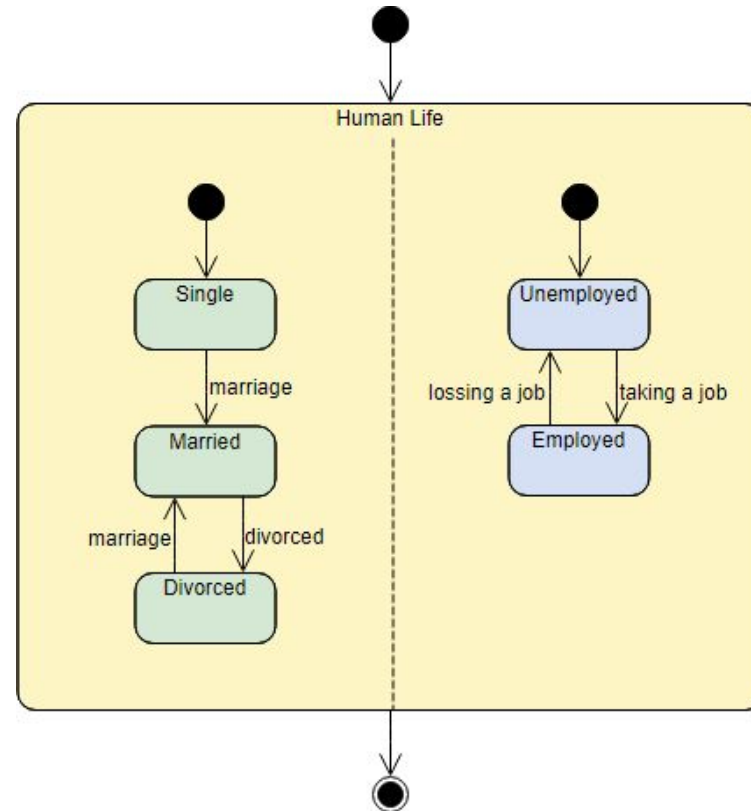
1. Introduction à la Spécification
2. Aperçu d'UML
3. Diagramme de Use Case et Description Textuelle
4. Diagramme d'Activité
5. Diagramme de Séquence
6. Diagramme de Classes
- 7. Diagramme d'États**

Big Picture

Chaque objet est à un moment donné dans un état particulier. Cet **état** modélise donc une situation durant laquelle un ensemble de conditions ne varient pas.

Le **diagramme d'état** modélise les changements d'état.

Example



Etude de cas : bibliothèque

Quel est le cycle de vie d'un livre, quels sont les différents états par lequel il peut passer ?

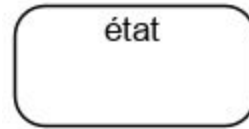
- Réservé
- Emprunté
- ...

Quelles sont les événements qui déclenchent le passage d'un état à l'autre ?

- Restitution par le client
- ...

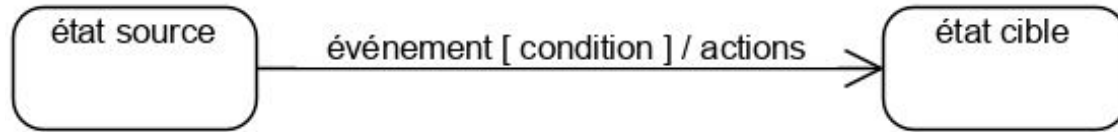
Notation

Rectangle aux coins arrondis qui indique la nature actuelle d'un objet.



Transition

Une **transition** est une relation entre deux états d'un même objet déclenchée par un événement extérieure.

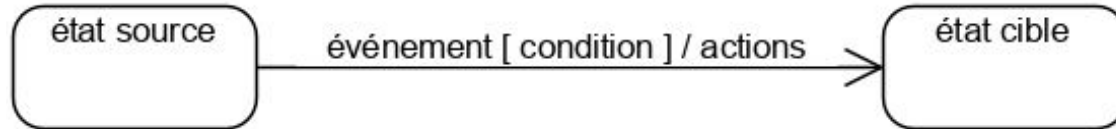


Cette relation indique que l'objet entre dans un second état et effectue certaines actions lorsqu'un événement se produit et que les conditions sont satisfaites.

Transition

Une transition comporte cinq parties :

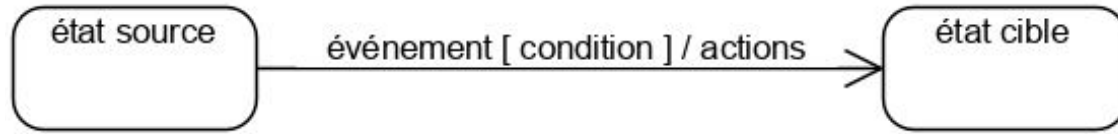
1. **Etat source** : état affecté par la transition.
2. **Etat cible** : état actif après l'achèvement de la transition.



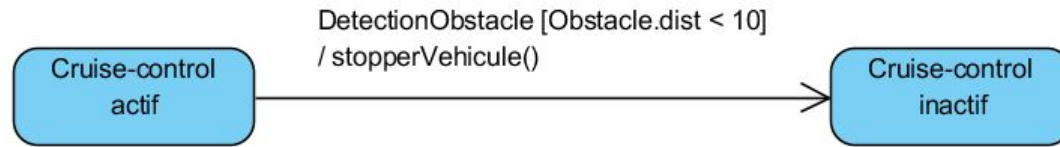
Transition

Une transition comporte cinq parties:

3. **Trigger** [0..*] : spécifie le ou les événements qui déclenchent la transition d'un état source lorsque la condition de garde est satisfaisante.
4. **Guard** [0..1] : condition de garde (expression booléenne) spécifie la condition qui rend possible la transition.
5. **Effect** [0..1] : spécifie une ou plusieurs actions à effectuer lorsque la transition est déclenchée.



Exemple

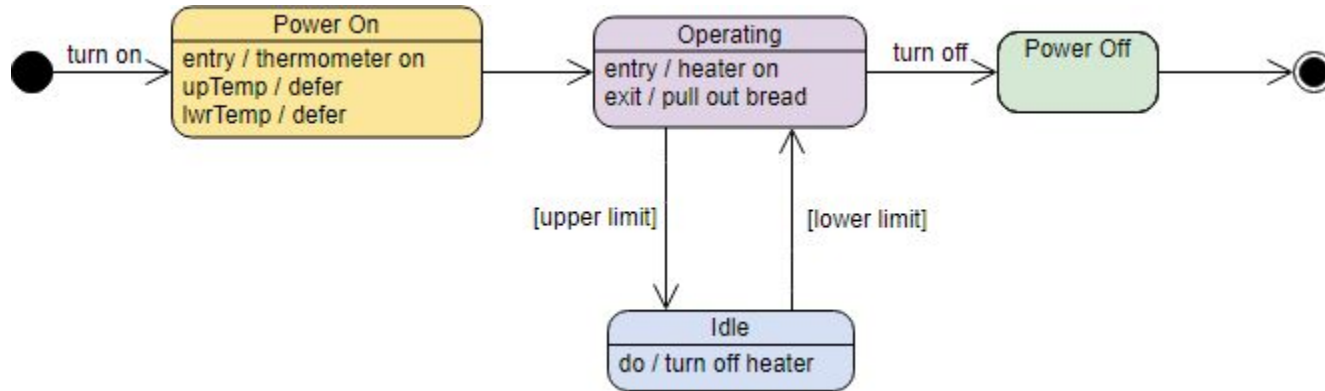


Types d'événement

1. **Call Event** : représente un **appel** à une opération particulière d'un objet
2. **Change Event** : modélise le fait qu'une **condition** devient vraie
3. **Signal Event** : représente la réception d'un **message** asynchrone
4. **Time Event** : spécifie un **moment** particulier dans le temps

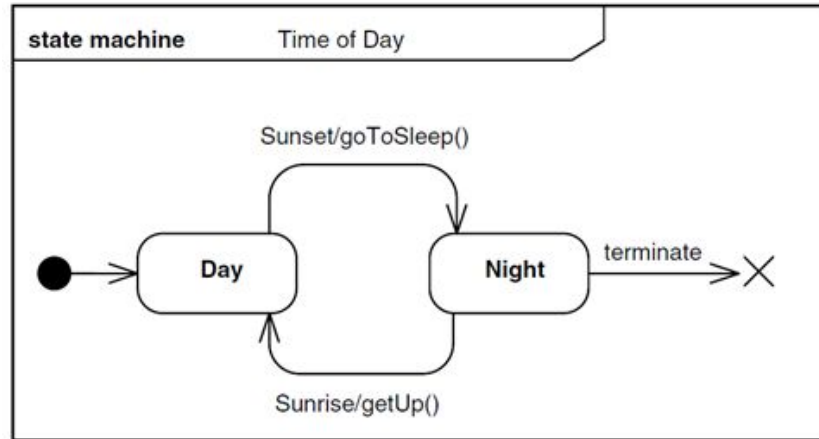
Pseudo-Etat Initial et Final

Chaque diagramme commence généralement par un **cercle noir** indiquant l'**état initial** et se termine par un **cercle encadré** indiquant l'**état final**.



Pseudo-Etat Terminate

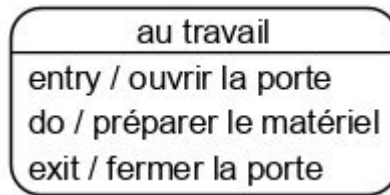
Pseudo-Etat impliquant la terminaison de la machine à état, c-à-d la destruction de l'objet. Il est représenté par une croix.



Activité interne

A la différence avec la transition, l'activité interne

1. Ne possède pas d'état cible.
2. L'état actif reste le même après le déclenchement.
3. Est représenté à l'intérieur de leur état.



Activité interne

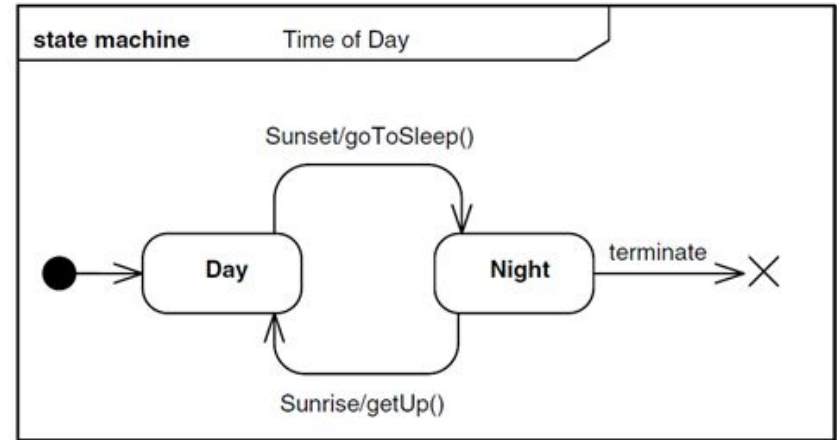
Les activités internes possèdent des noms d'événement prédéfinis correspondant à des déclencheurs particuliers :

- **entry** : comportement exécuté à l'entrée de l'état, souvent pour effectuer la configuration nécessaire à l'état.
- **do** : comportement exécuté dès que l'activité entry est terminé.
- **exit** : comportement exécuté quand l'objet quitte l'état, après le traitement de toute activité interne et avant toute activité de transition vers l'état suivant.

3 types d'états

1. Simple State

- Etat sans sous-état.



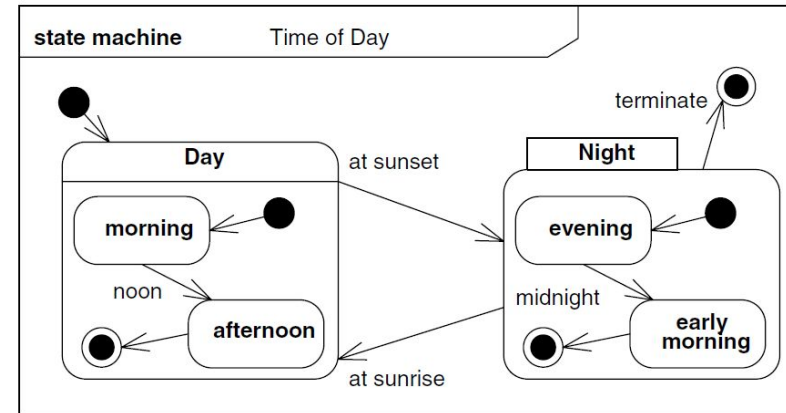
3 types d'états

1. Simple State

- Etat sans sous-état.

2. Composite State

- **Simple composite State**
 - Définition de sous-états et leurs transitions.



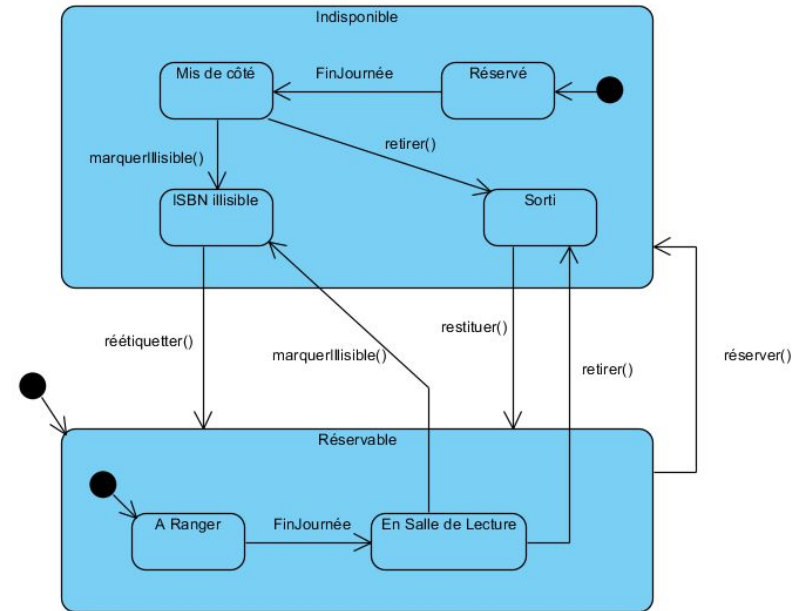
3 types d'états

1. Simple State

- Etat sans sous-état.

2. Composite State

- **Simple composite State**
 - Définition de sous-états et leurs transitions.
- **Orthogonal State**
 - Etat comprenant différentes régions indépendantes.



3 types d'états

1. Simple State

- Etat sans sous-état.

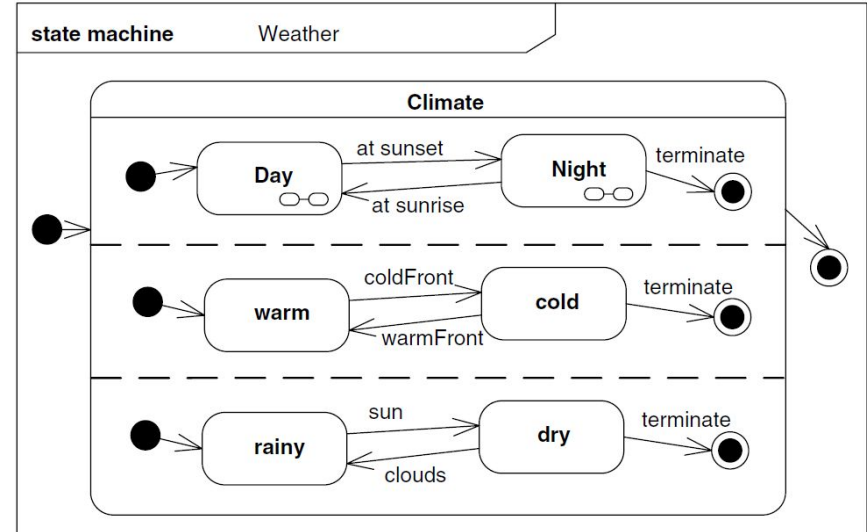
2. Composite State

- **Simple composite State**
 - Définition de sous-états et leurs transitions.
- **Orthogonal State**
 - Etat comprenant différentes régions indépendantes.

3. Submachine State

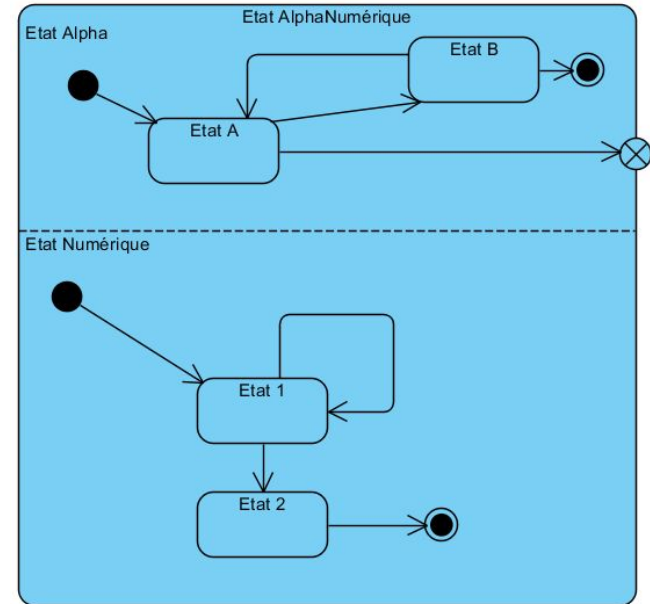
- Sous-état représenter par deux états reliés.

L' état orthogonal Climate est terminé lorsque tous les sous-états orthogonaux sont terminés



Exit Point

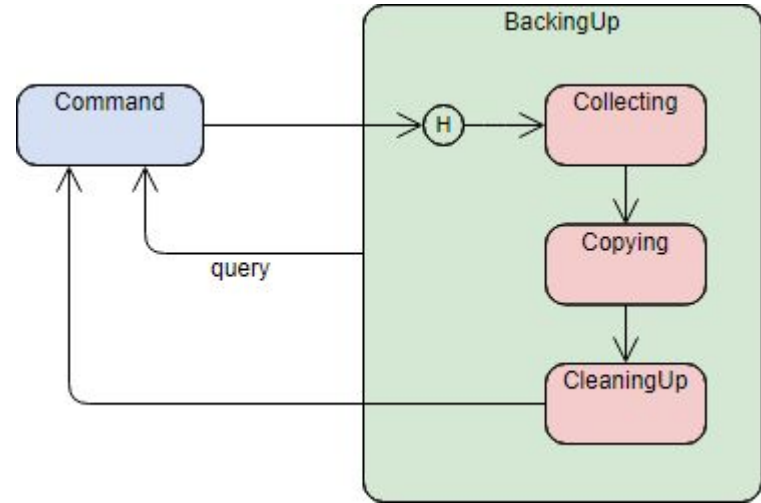
Lorsque l'**Exit Point** d'une région est atteint, l'état composite est considéré dans son entièreté comme terminé (même si d'autres états orthogonaux ne sont pas terminés).



History State

Un “History State” est utilisé pour rappeler l'état précédent d'une machine à états lorsqu'elle a été interrompue

1. **Shallow History.**
2. **Deep History** : identique sauf que le mécanisme de mémorisation du dernier état est également valable pour les sous-états composites de l'état.

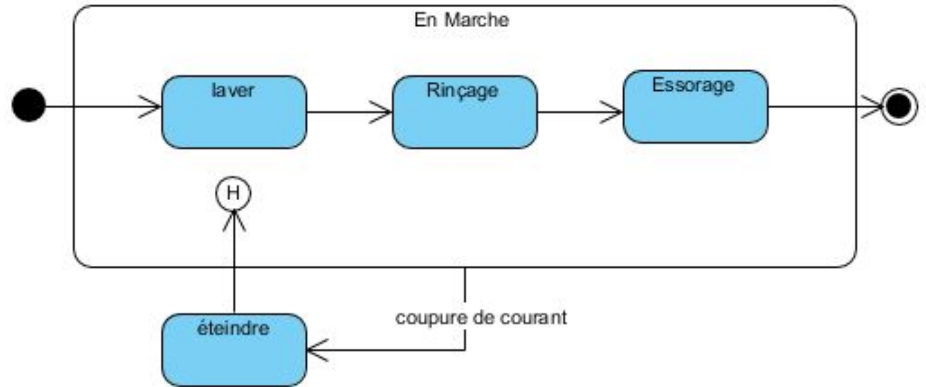


Exemple

Quand une machine à laver est en marche, elle passera de "Lavage" à "Rinçage" en passant à "Essorage".

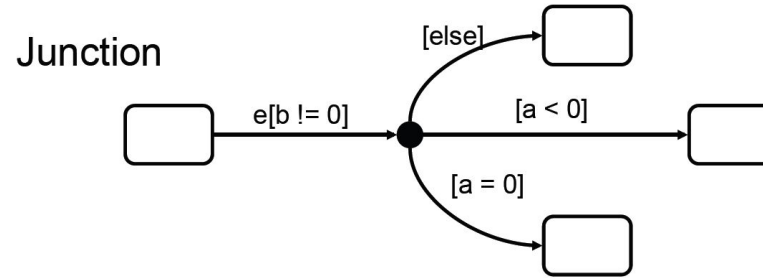
En cas de coupure de courant, le lave-linge cessera de fonctionner et passera à l'état "Éteindre".

Lorsque le courant est rétabli, l'état de fonctionnement est entré avec le symbole "Etat historique", ce qui signifie qu'il doit être rétabli là où il avait été arrêté.



Junction Point

Utiliser pour séparer ou rejoindre plusieurs transitions.



Exercices

Réaliser les exercices des diagrammes d'état.