

UML

Diagrammes états-transitions

1. États et transitions

Diagramme états-transitions

Objectif : Décrire le **comportement dynamique** d'une entité (logiciel, composant, objet...)

Comportement décrit par états + transitions entre les états

- **État** : abstraction d'un **moment de la vie d'une entité** pendant lequel elle satisfait un ensemble de conditions
- **Transition** : **changement d'état**

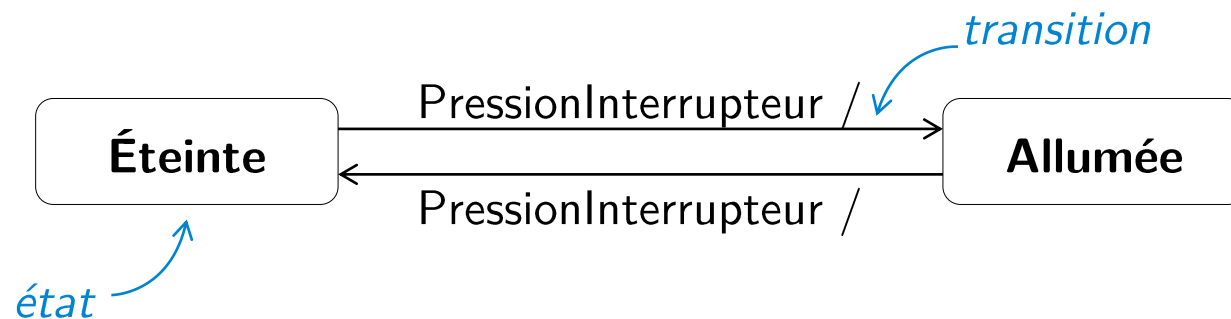


Diagramme états-transitions

Objectif : Décrire le **comportement dynamique** d'une entité (logiciel, composant, objet...)

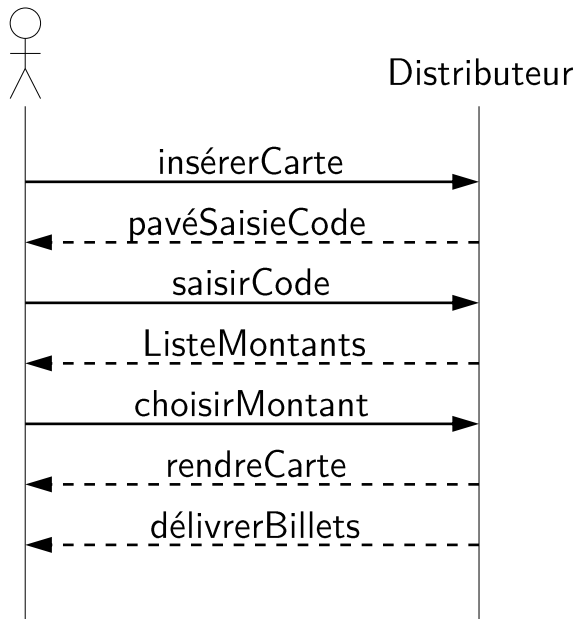
Comportement décrit par états + transitions entre les états

- **État** : abstraction d'un **moment de la vie d'une entité** pendant lequel elle satisfait un ensemble de conditions
- **Transition** : **changement** d'état

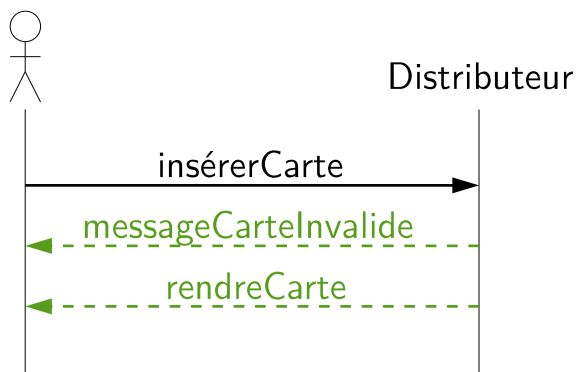
Intérêt :

- **Vue synthétique** de la dynamique de l'entité
- Regroupement d'un **ensemble de scénarios**

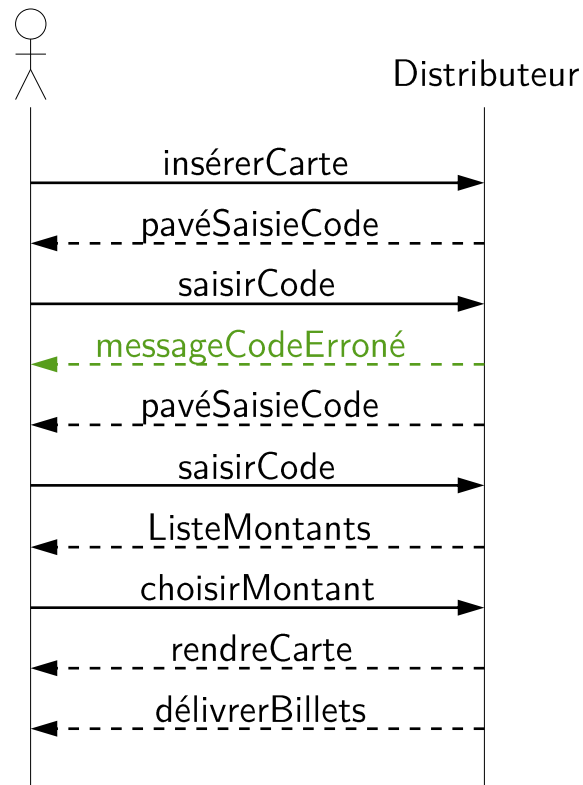
Exemple : distributeur automatique



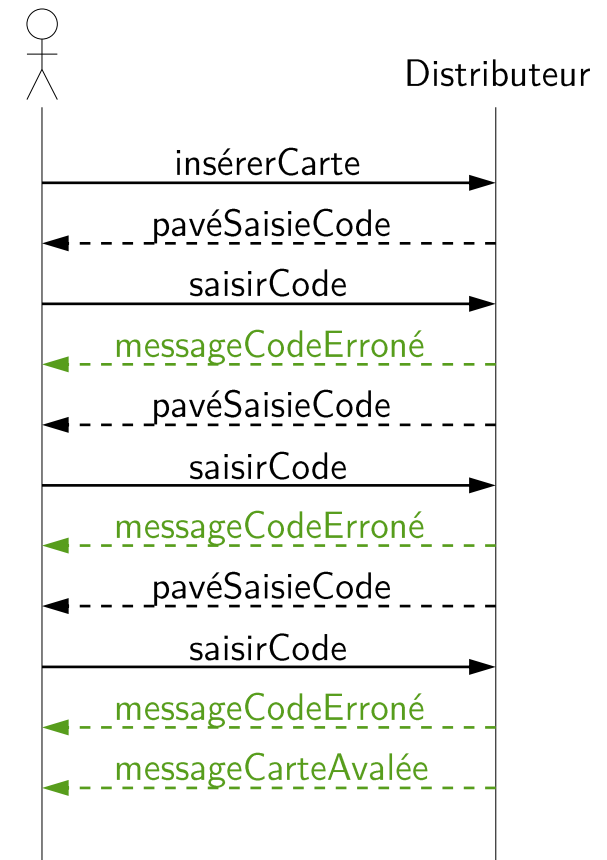
Scénario principal



Carte invalide





Une erreur de code



Trois erreurs de code

État

Types d'états

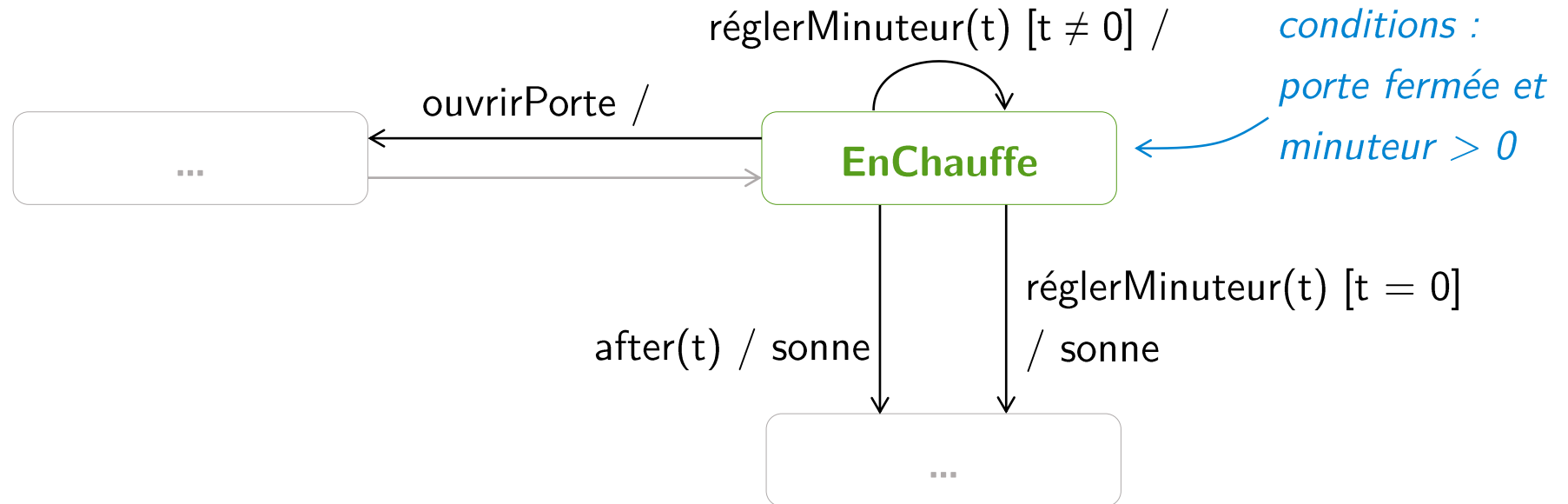
- **État initial** 
Initialisation du système, exécution du constructeur de l'objet
- **État final** 
Fin de vie du système, destruction de l'objet
- **États intermédiaires** : étapes de la vie du système, de l'objet

ÉtatSimple

État

Caractéristiques d'un état :

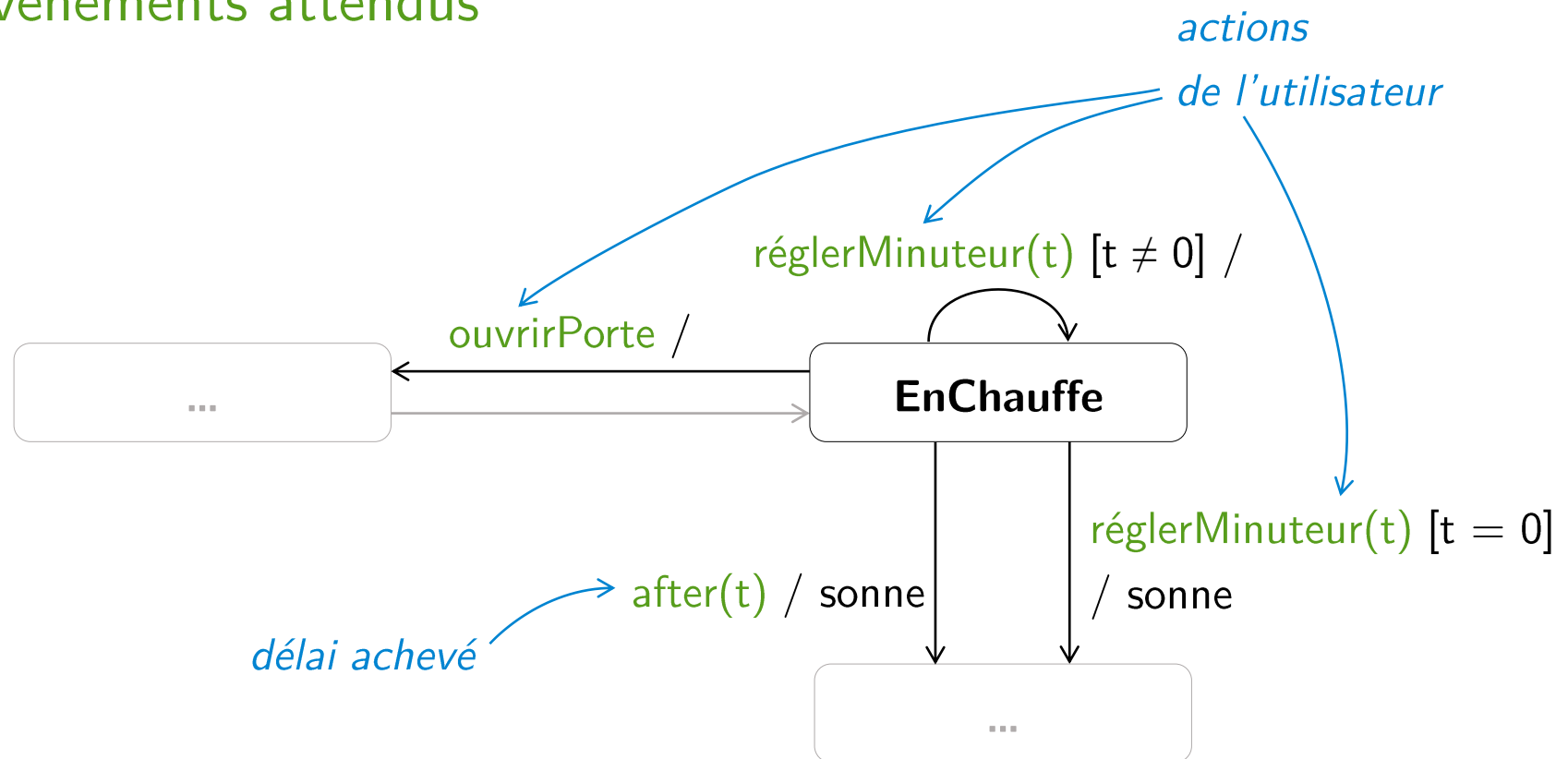
- Conditions vérifiées
- Événements attendus



État

Caractéristiques d'un état :

- Conditions vérifiées
- Événements attendus



Événement

Événement : Fait instantané venant de l'extérieur du système et survenant à un instant donné

Types d'événements :

- **Signal** : réception d'un message asynchrone
- **Appel d'une opération** (synchrone) : liée aux cas d'utilisation, opération du diagramme de classes...
- **Satisfaction d'une condition** booléenne : **when**(*cond*), évaluée continuellement jusqu'à ce qu'elle soit vraie
- **Temps**
 - Date relative : **when**(date = *date*)
 - Date absolue : **after**(*durée*)

Action

Action : Réaction du système à un événement

Caractéristiques : atomique, instantanée, non interruptible

Exemples d'actions (syntaxe laissée libre) :

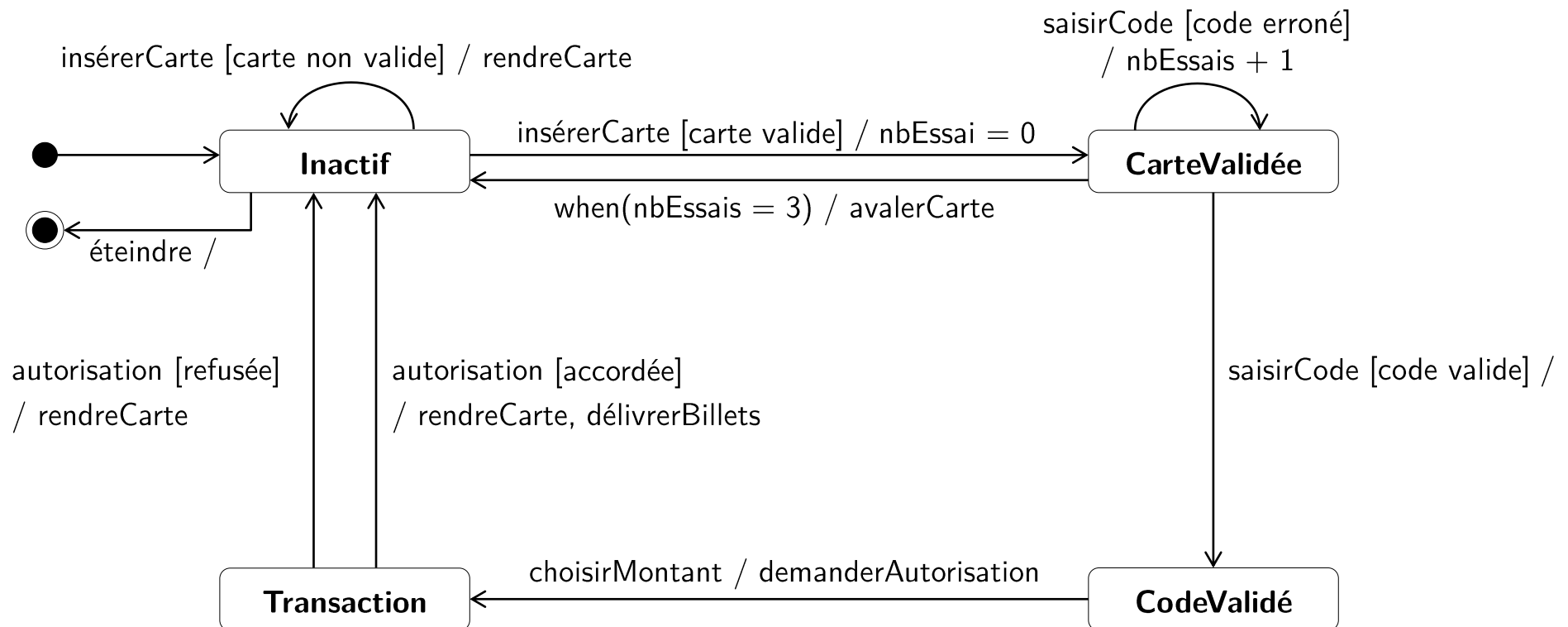
- affectation
- envoi d'un signal
- appel d'une opération
- création ou destruction d'un objet

Transition



Lorsque l'événement se produit, si la condition est vérifiée, alors l'action est effectuée

Diagramme états-transitions correspondant



Utilisation des diagrammes états-transitions

En phase d'analyse :

- Description de la **dynamique du système** vu de l'extérieur
- Synthèse des scénarios liés aux **cas d'utilisation**
- Événements = **action des acteurs**

En phase de conception :

- Description de la **dynamique d'un objet** particulier
- Événements = **appels d'opérations**

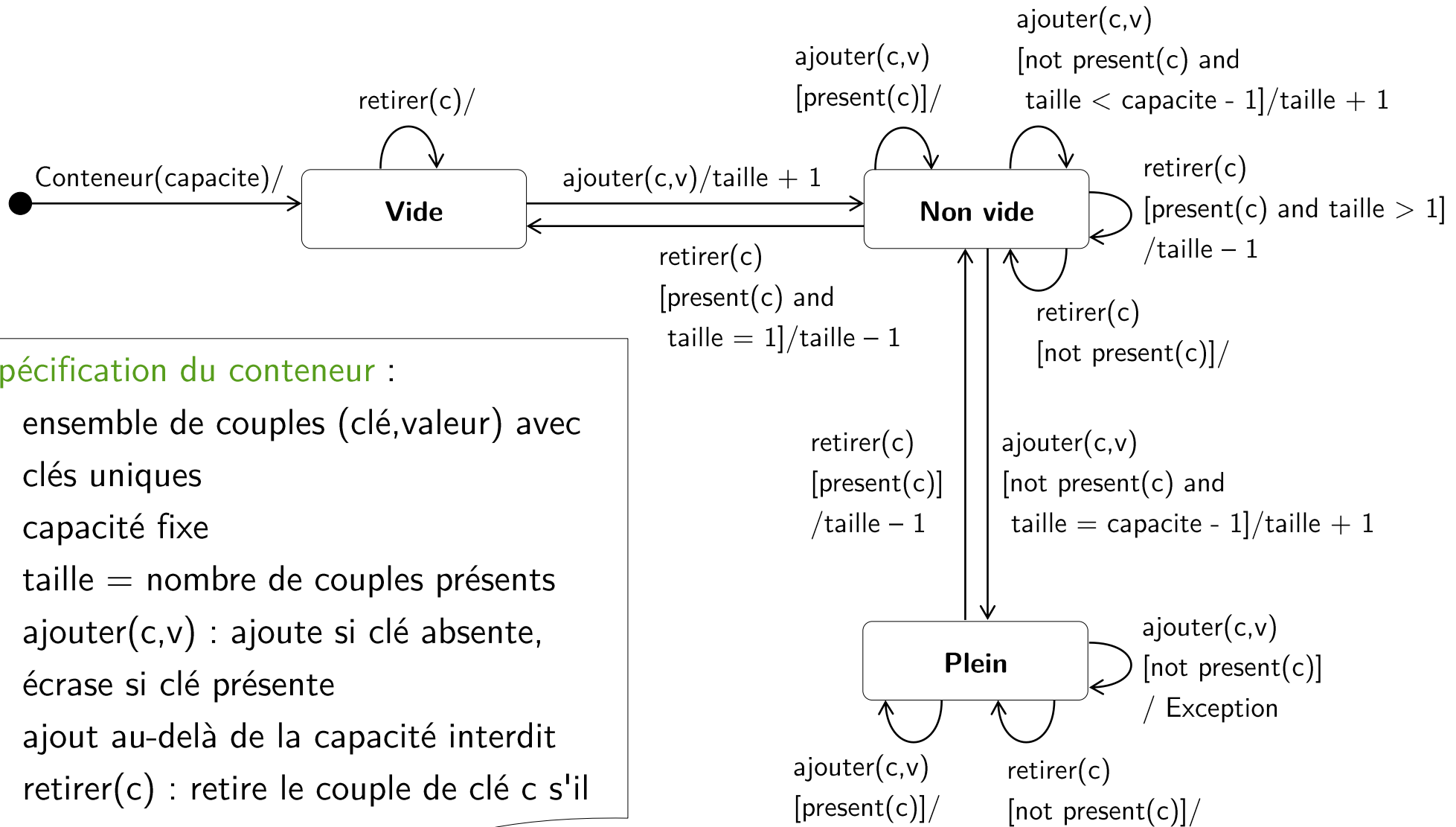
Diagramme états-transitions d'un objet

Spécification du conteneur :

- ensemble de couples (clé,valeur) avec clés uniques
- capacité fixe
- taille = nombre de couples présents
- ajouter(c,v) : ajoute si clé absente, écrase si clé présente
- ajout au-delà de la capacité interdit
- retirer(c) : retire le couple de clé c s'il existe

Conteneur
capacite : int taille : int
Conteneur(cap : int) ajouter(c:Object,v:Object) retirer(c:Object) present(c:Object) : boolean

Diagramme états-transitions d'un objet



Spécification du conteneur :

- ensemble de couples (clé,valeur) avec clés uniques
- capacité fixe
- taille = nombre de couples présents
- ajouter(c,v) : ajoute si clé absente, écrase si clé présente
- ajout au-delà de la capacité interdit
- retirer(c) : retire le couple de clé c s'il existe

Suite

- États composites
- États orthogonaux
- Entry, exit, activités et transitions internes