

Conception Orientée Objet
UML 2

PLAN

- Partie 1:
 - Notions sur le Génie Logiciel (GL)
 - Concepts de l'approche objet
 - UML
 - Diagrammes statiques (structurels)
 - **Diagrammes comportementaux**

REFERENCES

- UML 2 Analyse et conception; Joseph Gabay, David Gabay.
- UML 2 de l'apprentissage à la pratique. Laurent AUDIBERT.
- Développement de logiciel à objets avec UML. Jean-Marc Jézéquel, Noël Plouzeau, Yves Le Traon.
- UML, Introduction au génie logiciel et à la modélisation. Delphine Longuet.
- Introduction au génie logiciel; L. Du Bousquet.
- Méthodes de Conception Orientés Objet (MCOO). Verlain FOUNDIKOU.

DIAGRAMMES COMPORTEMENTAUX

- Les diagrammes comportementaux sont focalisés sur la description de la partie dynamique du système à modéliser.
- 7 diagrammes sont proposés par UML 2 :
 - diagramme des cas d'utilisation (DCU),
 - diagramme d'état-transition (machine d'état, DET),
 - diagramme d'activité (DAC),
 - Diagrammes d'interaction:
 - diagramme de séquence (DSE),
 - diagramme de communication (DCO),
 - diagramme global d'interaction (DGI),
 - diagramme de temps (DTP).

DIAGRAMME DES CAS D'UTILISATION (DCU)

- Les cas d'utilisation constituent un moyen de recueillir et de décrire les besoins des acteurs du système.
- Peuvent être utilisés comme moyen **d'organisation** du développement du logiciel;
- Particulièrement pour la structuration et le déroulement des **tests du logiciel**.
- Un diagramme de cas d'utilisation capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit.

DIAGRAMME DES CAS D'UTILISATION (DCU)

- Un cas d'utilisation permet de décrire l'interaction entre les acteurs (utilisateurs du cas) et le système.
- La description de l'interaction est réalisée suivant le point de vue de l'utilisateur.
- La représentation d'un cas d'utilisation met en jeu 3 concepts :
 - l'acteur,
 - le cas d'utilisation
 - l'interaction entre l'acteur et le cas d'utilisation.

DIAGRAMME DES CAS D'UTILISATION (DCU)

ACTEUR

- Utilisateur type qui a toujours le même comportement vis-à-vis d'un cas d'utilisation.
- Les utilisateurs d'un système appartiennent à une ou plusieurs classes d'acteurs selon les rôles qu'ils tiennent par rapport au système.
- Une même personne physique peut se comporter en autant d'acteurs différents que le nombre de rôles qu'elle joue vis-à-vis du système:
 - Exemple: l'administrateur d'un système de messagerie peut être aussi utilisateur de cette même messagerie.
 - Il sera considéré, en tant qu'acteur du système, dans le rôle d'administrateur d'une part
 - et dans celui d'utilisateur d'autre part.
- Un acteur peut aussi être un **système externe** avec lequel le cas d'utilisation va interagir.

DIAGRAMME DES CAS D'UTILISATION (DCU)

ACTEUR

- **Formalisme:**
 - Un acteur peut se représenter symboliquement par un "bonhomme" et être identifié par son nom.
 - Il peut aussi être formalisé par une classe stéréotypée "acteur"



Figure. Représentations d'un acteur

DIAGRAMME DES CAS D'UTILISATION (DCU)

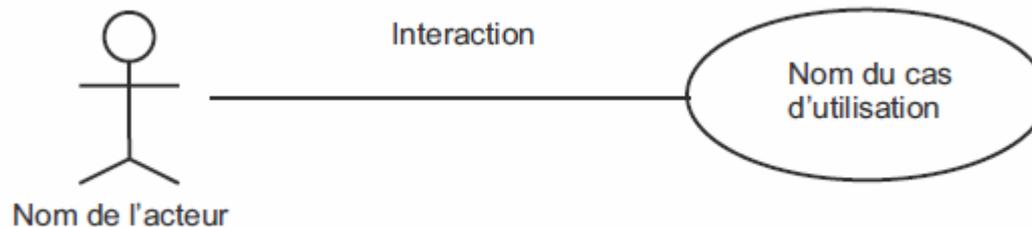
CAS D'UTILISATION ET INTERACTION

- Un **cas d'utilisation** correspond à un certain nombre d'**actions** que le système devra exécuter en réponse à un besoin d'un acteur.
- Un cas d'utilisation doit produire un **résultat observable** pour un ou plusieurs acteurs ou parties prenantes du système.
- Une **interaction** permet de décrire les échanges entre un acteur et un cas d'utilisation.

DIAGRAMME DES CAS D'UTILISATION (DCU)

CAS D'UTILISATION ET INTERACTION

- **Formalismes:**
 - Un cas d'utilisation se représente par un ovale (ellipse) dans lequel figure son intitulé (et optionnellement, au-dessus du nom, un stéréotype).
 - L'interaction entre un acteur et un cas d'utilisation se représente comme une association. Elle peut comporter des multiplicités comme toute association entre classes.



Formalisme de base de représentation d'un cas d'utilisation

- Pour satisfaire le besoin exprimé par l'acteur, chaque cas d'utilisation doit être décrit sous forme textuelle (bien identifier les traitements à réaliser par le système).

DIAGRAMME DES CAS D'UTILISATION (DCU)

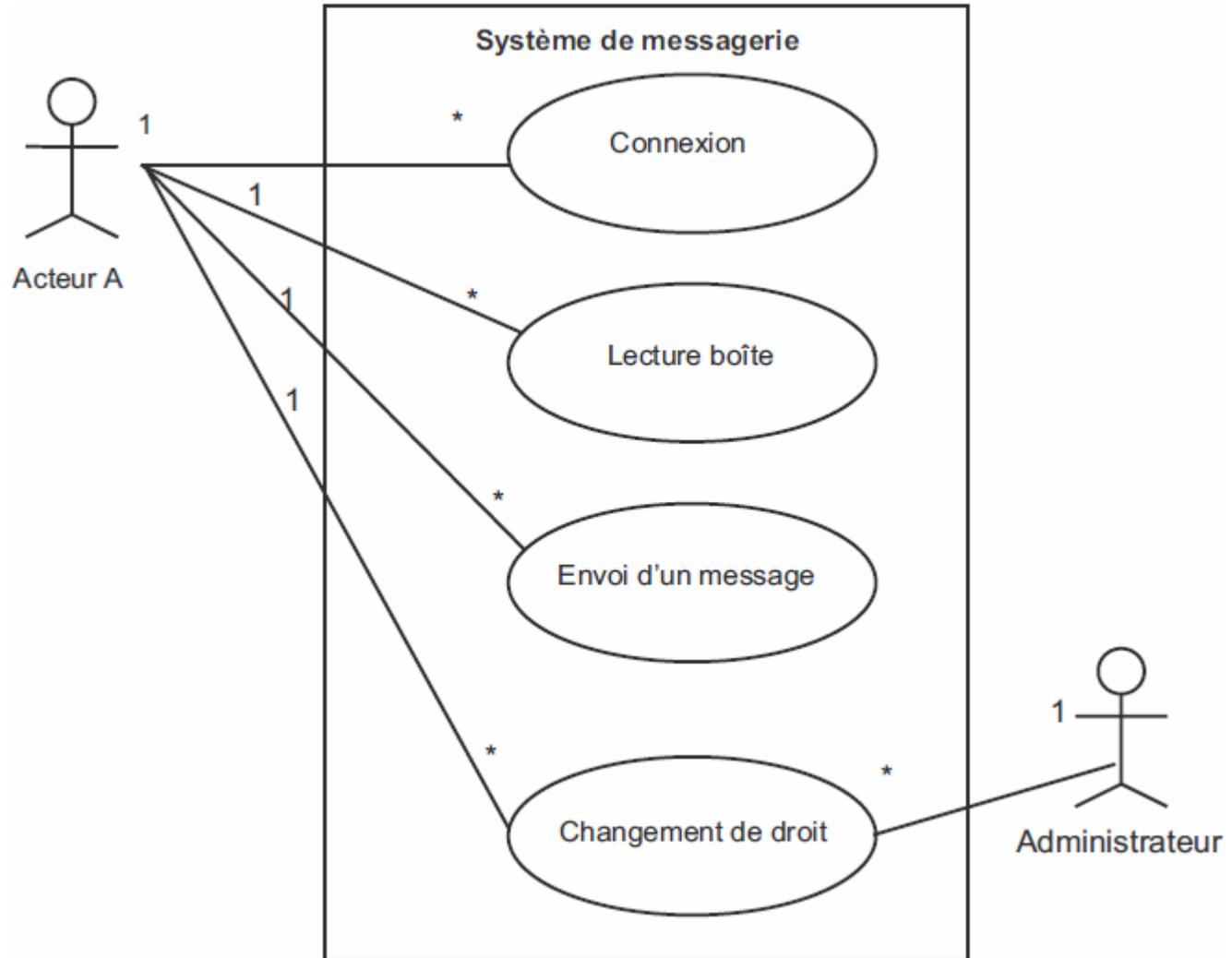
REPRÉSENTATION D'UN DIAGRAMME DE CAS D'UTILISATION

- Tout système peut être décrit par un certain nombre de cas d'utilisation correspondant aux besoins exprimés par l'ensemble des utilisateurs.
- À chaque utilisateur, vu comme acteur, correspondra un certain nombre de cas d'utilisation du système.
- L'ensemble de ces cas d'utilisation se représente sous forme d'un diagramme.

DIAGRAMME DES CAS D'UTILISATION (DCU)

REPRÉSENTATION D'UN DIAGRAMME DE CAS D'UTILISATION

- Exemple:



Exemple d'un système de messagerie comportant 4 cas d'utilisation

DIAGRAMME DES CAS D'UTILISATION (DCU)

REPRÉSENTATION D'UN DIAGRAMME DE CAS D'UTILISATION

- Exemple 2: Distributeur de billets

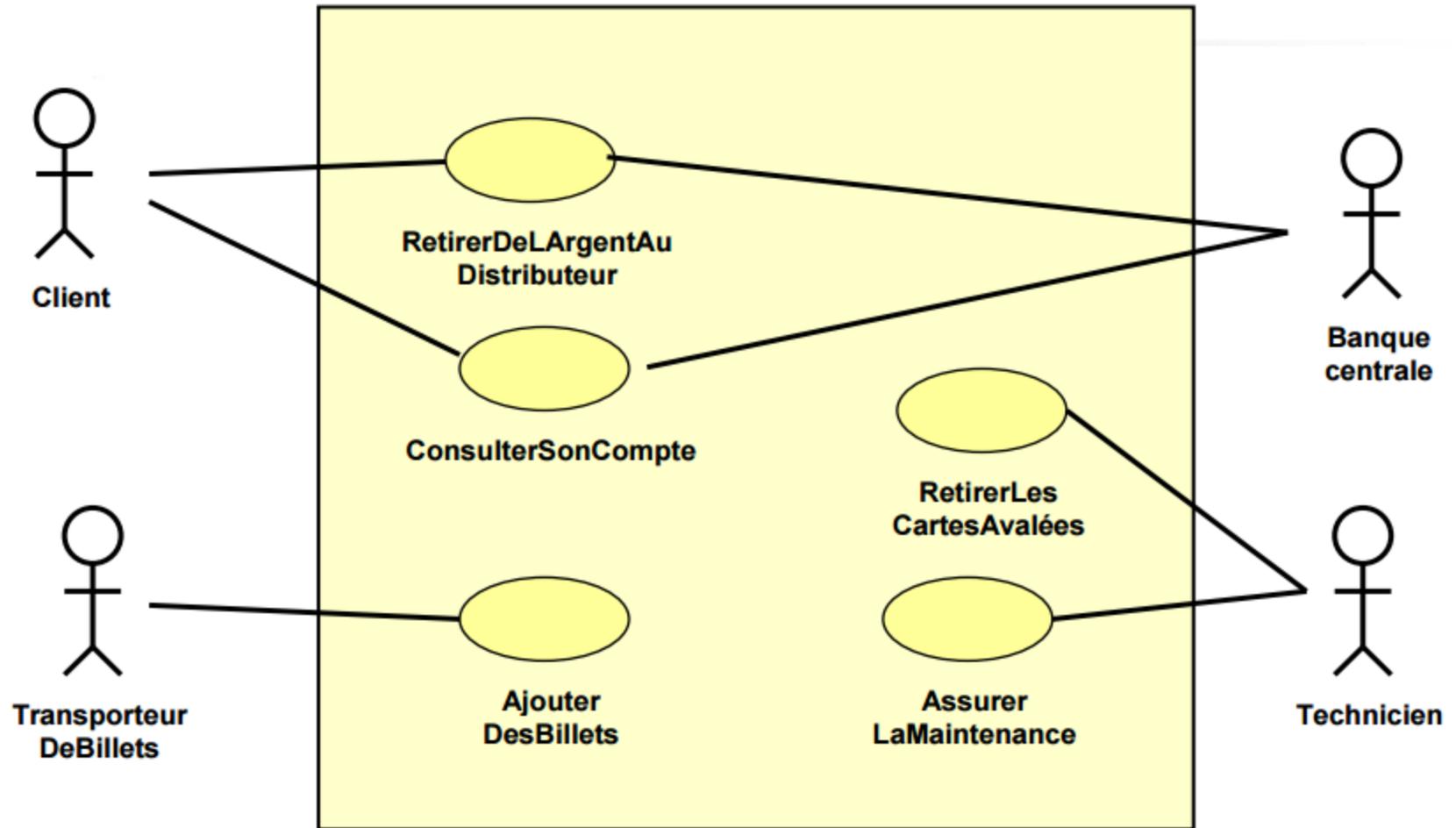


DIAGRAMME DES CAS D'UTILISATION (DCU)

RELATION ENTRE CAS D'UTILISATION

- Pour optimiser la formalisation des besoins en ayant recours à la réutilisation de cas d'utilisation, 3 relations peuvent être décrites entre cas d'utilisation :
 - une relation d'inclusion (« include ») , (obligatoire)
 - une relation d'extension (« extend »), (optionnel)
 - une relation de généralisation.

RELATION D'INCLUSION:

- Un cas 'A' inclut un cas 'B' si le comportement décrit par le cas 'A' inclut le comportement du cas 'B'.

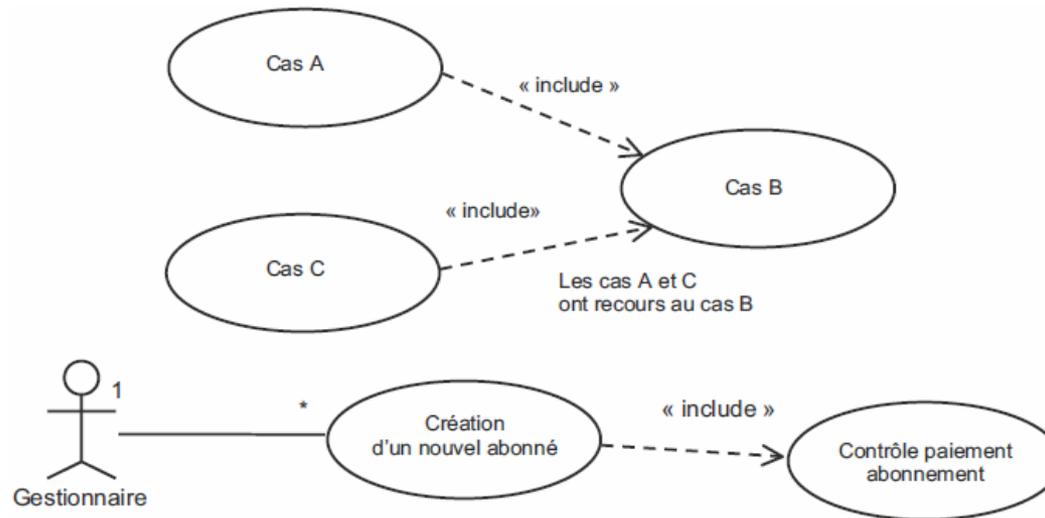


DIAGRAMME DES CAS D'UTILISATION (DCU)

RELATION D'EXTENSION:

- On dit qu'un cas d'utilisation B étend un cas d'utilisation A lorsque le cas d'utilisation B **peut être** appelé au cours de l'exécution du cas d'utilisation A.
- Exécuter A peut éventuellement entraîner l'exécution de B.
- contrairement à l'inclusion, l'extension est optionnelle.
- Cette dépendance est symbolisée par le stéréotype « **extend** »
- L'extension peut intervenir à un point précis du cas étendu.
- Ce point s'appelle le point d'extension.
- Il porte un nom, qui figure dans un compartiment du cas étendu sous la rubrique ***point d'extension***, et est éventuellement associé à une contrainte indiquant le moment où l'extension intervient.
- Une extension est souvent soumise à condition.
- Graphiquement, la condition est exprimée sous la forme d'une note.

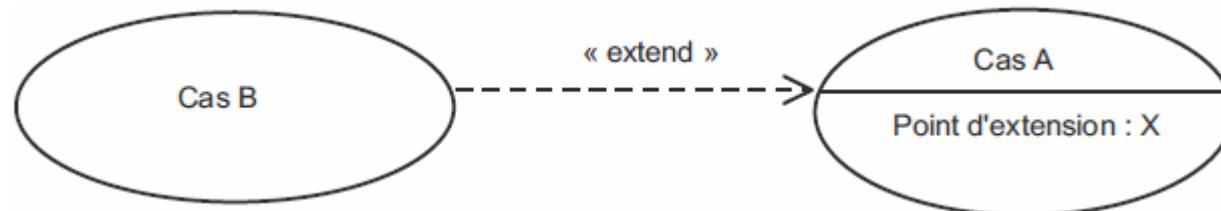


DIAGRAMME DES CAS D'UTILISATION (DCU)

RELATION D'EXTENSION:

- Exemple d'une relation d'extension entre cas d'utilisation :

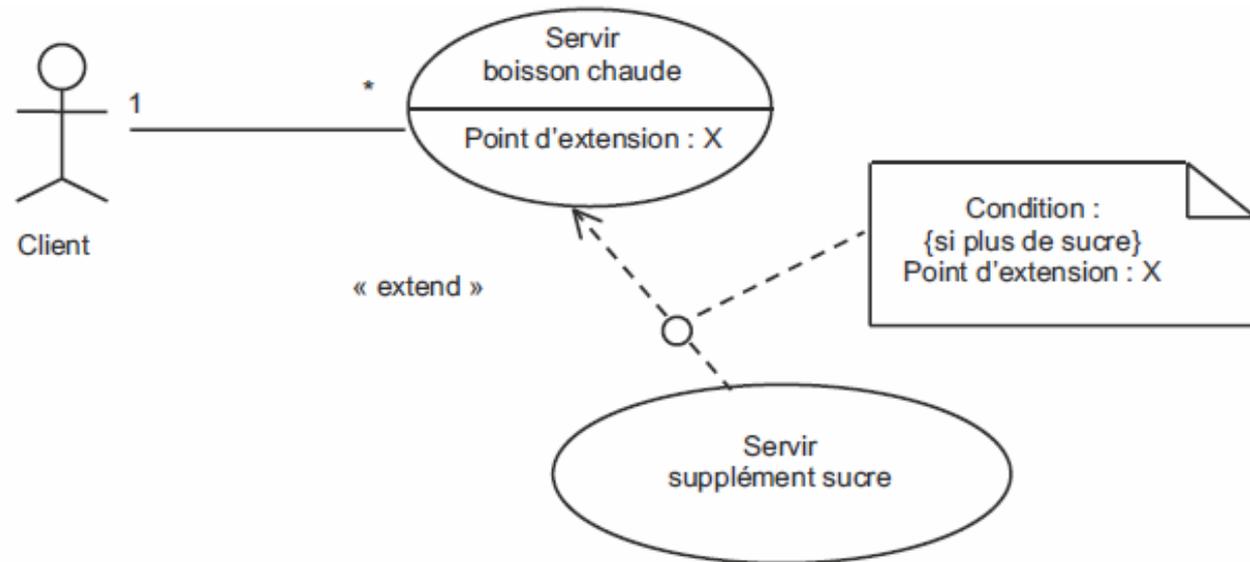


DIAGRAMME DES CAS D'UTILISATION (DCU)

RELATION DE GÉNÉRALISATION (ENTRE CU) :

- Une relation de généralisation de cas d'utilisation peut être définie conformément au principe de la spécialisation-généralisation déjà présentée pour les classes.
- Exemple consultation compte bancaire :

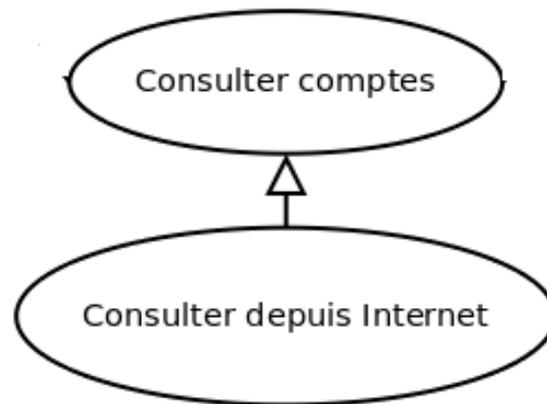


DIAGRAMME DES CAS D'UTILISATION (DCU)

DESCRIPTION TEXTUELLE D'UN CU:

- À chaque cas d'utilisation doit être associée une description textuelle des interactions entre l'acteur et le système et les actions que le système doit réaliser.
- UML ne propose pas de présentation type de cette description textuelle.
- Référence (Alistair COCKBURN, *Rédiger des cas d'utilisation efficaces*, Eyrolles, 2001.)

DIAGRAMME DES CAS D'UTILISATION (DCU)

DESCRIPTION TEXTUELLE D'UN CU:

- Articulée en six points :

1. **Objectif** - Décrire sommairement le contexte et les résultats attendus du cas d'utilisation.
2. **Acteurs concernés** - Le ou les acteurs concernés par le cas doivent être identifiés en précisant globalement leur rôle.
3. **Pré conditions** - Si certaines conditions particulières sont requises avant l'exécution du cas, elles sont à exprimer à ce niveau.
4. **Post conditions** - Par symétrie, si certaines conditions particulières doivent être réunies après l'exécution du cas, elles sont à exprimer à ce niveau.
5. **Scénario nominal** - Il s'agit là du scénario principal qui doit se dérouler sans incident et qui permet d'aboutir au résultat souhaité.
6. **Scénarios alternatifs** - Les autres scénarios, secondaires ou correspondant à la résolution d'anomalies, sont à décrire à ce niveau.
Le lien avec le scénario principal se fait à l'aide d'une numérotation hiérarchisée (1.1a, 1.1b...) rappelant le numéro de l'action concernée.

DIAGRAMME DES CAS D'UTILISATION (DCU)

EXERCICE DCU: ENNONCÉ

- Une bibliothèque universitaire souhaite automatiser sa gestion. Cette bibliothèque est gérée par un gestionnaire chargé des inscriptions et des relances des lecteurs quand ceux-ci n'ont pas rendu leurs ouvrages au-delà du délai autorisé. Les bibliothécaires sont chargés de gérer les emprunts et la restitution des ouvrages ainsi que l'acquisition de nouveaux ouvrages.

Il existe trois catégories d'abonné. Tout d'abord les étudiants qui doivent seulement s'acquitter d'une somme forfaitaire pour une année afin d'avoir droit à tous les services de la bibliothèque. L'accès à la bibliothèque est libre pour tous les enseignants. Enfin, il est possible d'autoriser des étudiants d'une autre université à s'inscrire exceptionnellement comme abonné moyennant le versement d'une cotisation. Le nombre d'abonné externe est limité chaque année à environ 10 % des inscrits.

Un nouveau service de consultation du catalogue général des ouvrages doit être mis en place.

Les ouvrages, souvent acquis en plusieurs exemplaires, sont rangés dans des rayons de la bibliothèque. Chaque exemplaire est repéré par une référence gérée dans le catalogue et le code du rayon où il est rangé.

Chaque abonné ne peut emprunter plus de trois ouvrages. Le délai d'emprunt d'un ouvrage est de trois semaines, il peut cependant être prolongé exceptionnellement à cinq semaines.

DIAGRAMME DES CAS D'UTILISATION (DCU)

EXERCICE DCU: CORRIGÉ[1/2]

- 5 types d'acteurs peuvent être identifiés :
 1. étudiant,
 2. externe,
 3. emprunteur,
 4. gestionnaire,
 5. bibliothécaire.

- 6 cas d'utilisation peuvent être identifiés :
 1. inscription à la bibliothèque,
 2. consultation du catalogue,
 3. emprunt d'ouvrages,
 4. restitution d'ouvrages,
 5. approvisionnement d'ouvrages,
 6. relance emprunteur.

DIAGRAMME DES CAS D'UTILISATION (DCU)

EXERCICE DCU: CORRIGÉ [2/2]

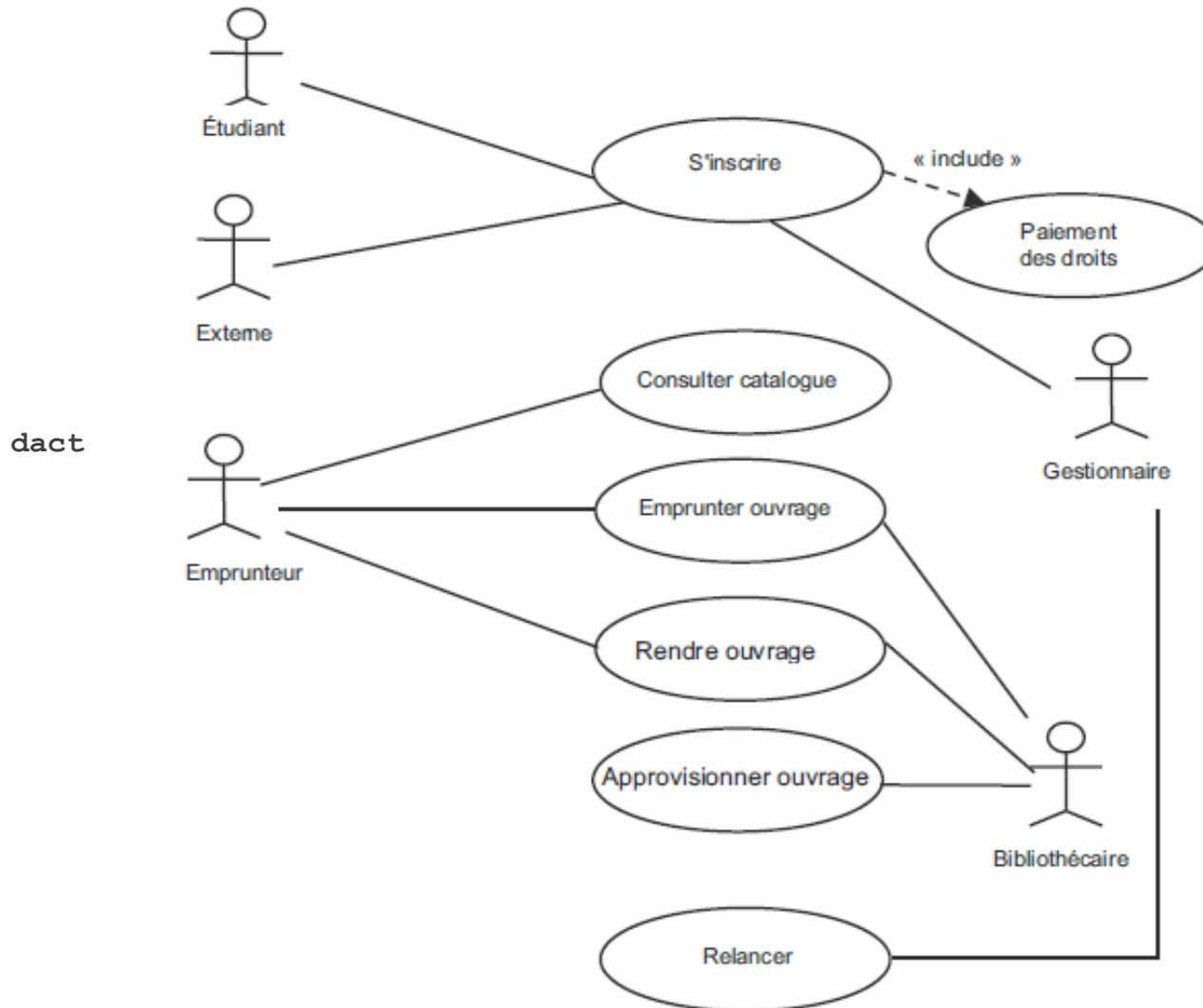


DIAGRAMME D'ÉTAT-TRANSITION (DET)

- Le diagramme d'états-transitions représente la façon dont évoluent (*i.e.* cycle de vie) les objets appartenant à une même classe.

PRÉSENTATION GÉNÉRALE ET CONCEPTS DE BASE

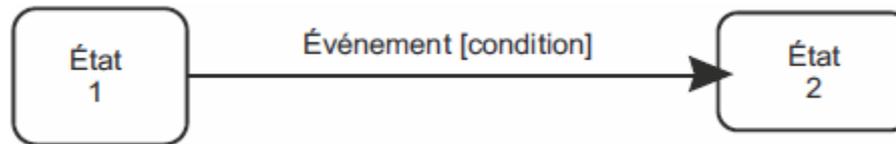
- État-transition et événement:
 - L'état d'un objet est défini, à un instant donné, par l'ensemble des valeurs de ses propriétés.
 - Le passage d'un état à un autre état s'appelle **transition**.
 - Un **événement** est un fait survenu qui déclenche une transition.
- Il existe 4 types d'événements :
 - Type appel de méthode (call): C'est le type le plus courant que nous traiterons dans la suite de la présentation.
 - Type signal - Exemple : clic de souris, interruption d'entrées-sorties... La modélisation de la réception ou l'émission d'un signal est traitée dans le diagramme d'activité.
 - Type changement de valeur (vrai/faux): C'est le cas de l'évaluation d'une expression booléenne.
 - Type écoulement du temps: C'est un événement lié à une condition de type after (durée) ou when (date).

DIAGRAMME D'ÉTAT-TRANSITION (DET)

PRÉSENTATION GÉNÉRALE ET CONCEPTS DE BASE

- **Formalisme et Exemple:**

- Un objet reste dans un état pendant une certaine durée.
- La durée d'un état correspond au temps qui s'écoule entre le début d'un état déclenché par une transition i et la fin de l'état déclenché par la transition $i+1$.
- Une condition, appelée « garde », peut être associée à une transition.



Formalisme d'état-transition

- Exemple: pour un employé donné d'une entreprise, nous pouvons considérer les deux états significatifs suivants : état recruté et état en activité.

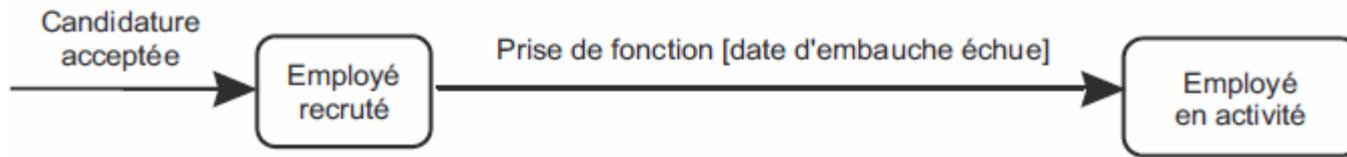
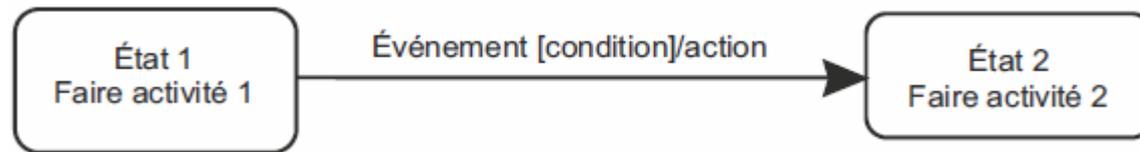


DIAGRAMME D'ÉTAT-TRANSITION (DET)

ACTION ET ACTIVITÉ

- Une action est une opération instantanée qui ne peut être interrompue; (Elle est associée à une transition)
- Une activité est une opération d'une certaine durée qui peut être interrompue, (elle est associée à un état d'un objet).

- **Formalisme:**



- **Exemple:**

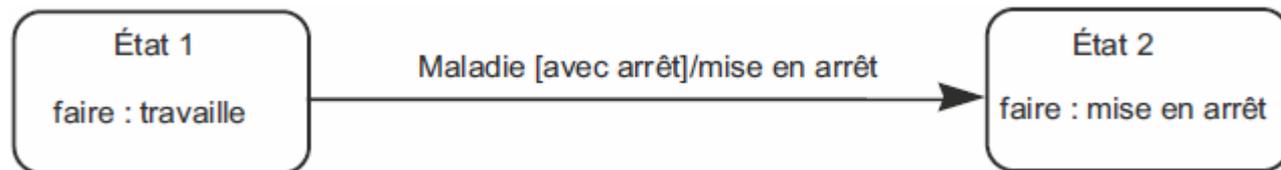


DIAGRAMME D'ÉTAT-TRANSITION (DET)

REPRÉSENTATION DU DIAGRAMME D'ÉTAT-TRANSITION D'UN OBJET

- L'enchaînement de tous les états caractéristiques d'un objet constitue le diagramme d'état.
 - Un diagramme d'états débute toujours par un état initial et se termine par un ou plusieurs états finaux, sauf dans le cas où le diagramme d'états représente une boucle.
 - À un événement peut être associé un message composé d'attributs.
-
- **Formalisme :**

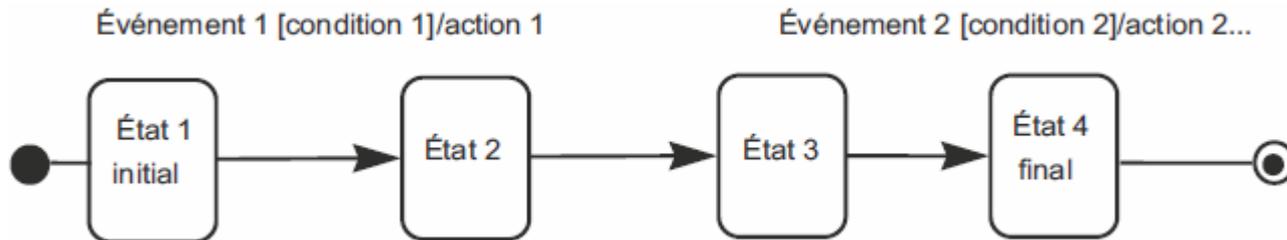


Figure. Formalisme de représentation des états initial et final

DIAGRAMME D'ÉTAT-TRANSITION (DET)

REPRÉSENTATION DU DIAGRAMME D'ÉTAT-TRANSITION D'UN OBJET

- Exemple 1:

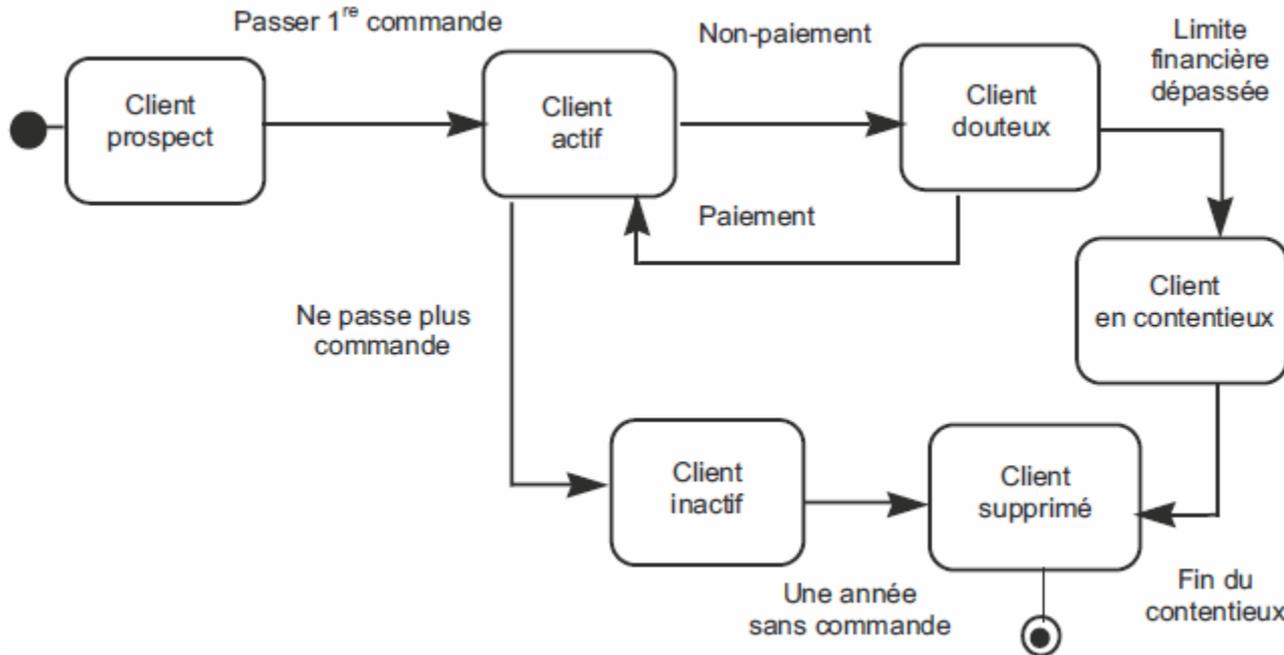


Figure. Diagramme d'état-transition de l'objet client d'une gestion commerciale

DIAGRAMME D'ÉTAT-TRANSITION (DET)

REPRÉSENTATION DU DIAGRAMME D'ÉTAT-TRANSITION D'UN OBJET

- Exemple 2:

- **En prévision d'arrivée:**
si la date prévisionnelle est postérieure à la date du jour.

- **En activité :**
état qui correspond à un personnel ayant une date d'arrivée renseignée.

- **Parti :** état qui correspond à un personnel ayant une date de départ renseignée.

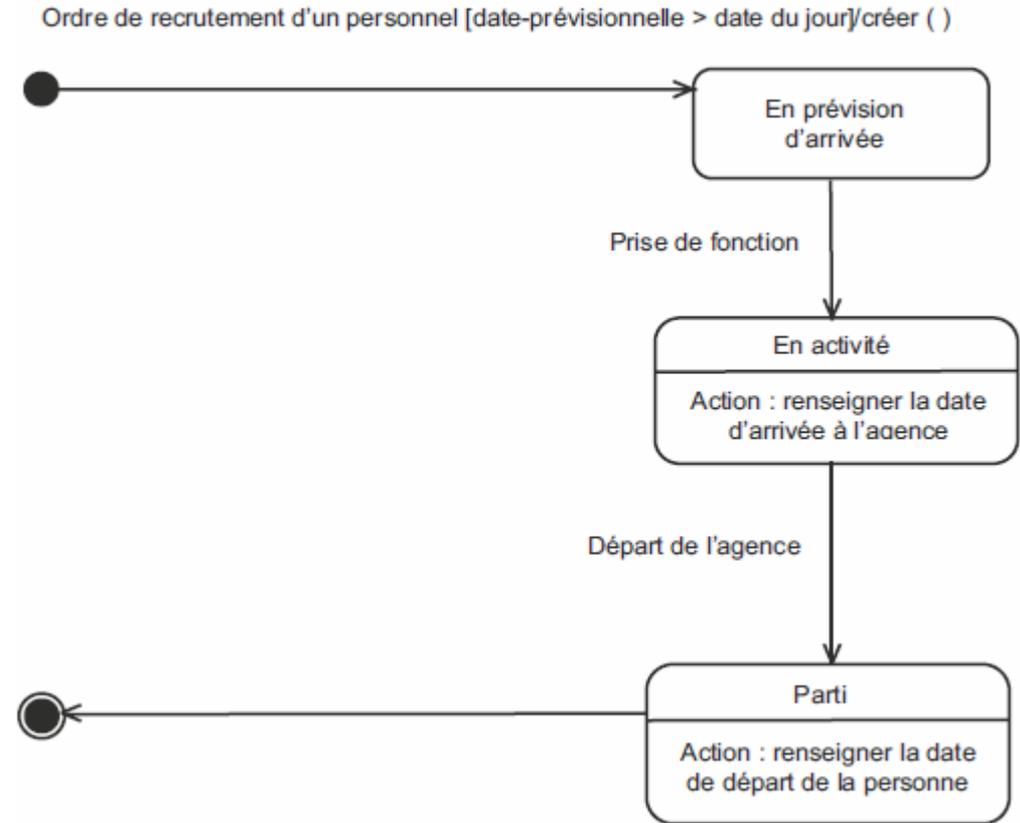


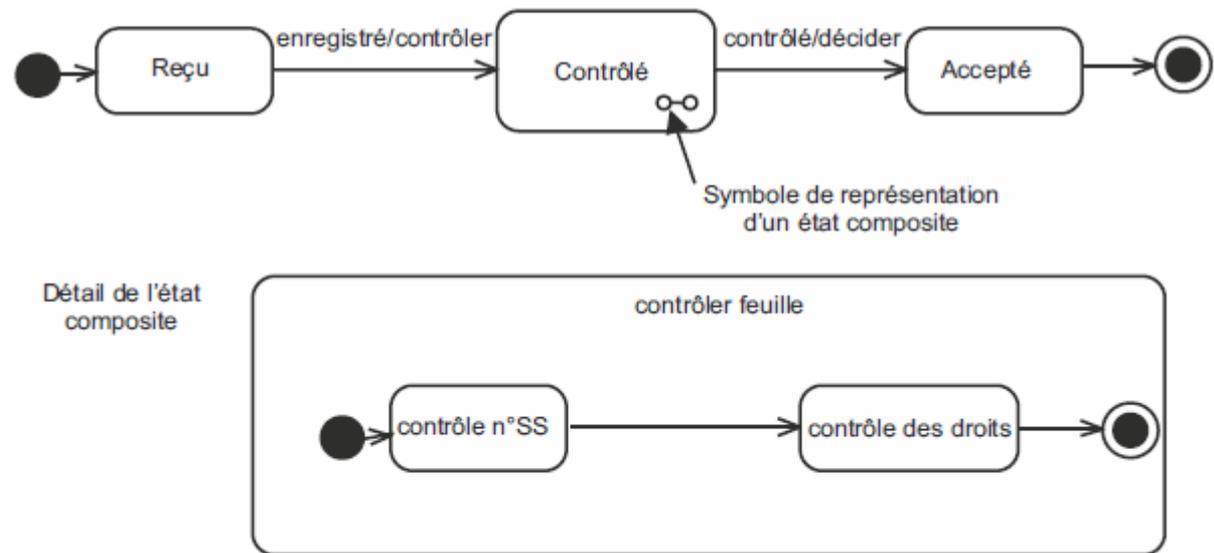
Figure. Diagramme d'état-transition de l'objet « personnel »

COMPLÉMENTS SUR LE DIAGRAMME D'ÉTAT-TRANSITION

COMPOSITION ET DÉCOMPOSITION D'ÉTAT

- Il est possible de décrire un diagramme d'état-transition à plusieurs niveaux.
- À un premier niveau, le diagramme comprendra des états **élémentaires** et des états **composites**.
- Les états composites seront ensuite décrits à un niveau élémentaire dans un autre diagramme.
- On peut aussi parler d'état composé et d'état composant.

— Dans cet exemple, l'état contrôlé est un état composite qui fait l'objet d'une description individualisée à un second niveau que l'on appelle aussi **sous-machine d'état**.



Exemple d'état composite

COMPLÉMENTS SUR LE DIAGRAMME D'ÉTAT-TRANSITION

POINT D'ENTRÉE ET DE SORTIE

- Sur une sous-machine d'état, il est possible de repérer un point d'entrée et un point de sortie particuliers.
- **Formalisme:**

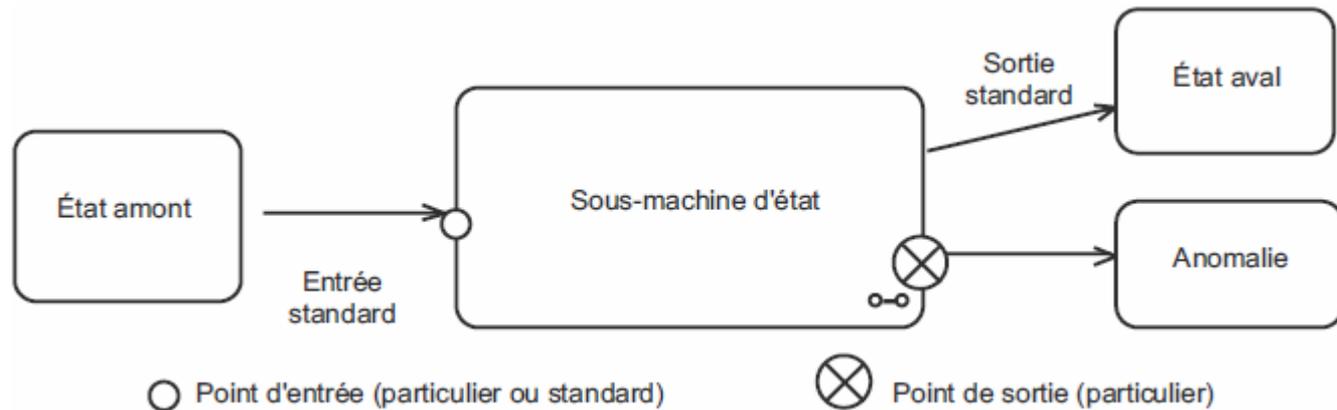


Figure. Exemple d'une sous-machine d'état avec point d'entrée et de sortie

COMPLÉMENTS SUR LE DIAGRAMME D'ÉTAT-TRANSITION

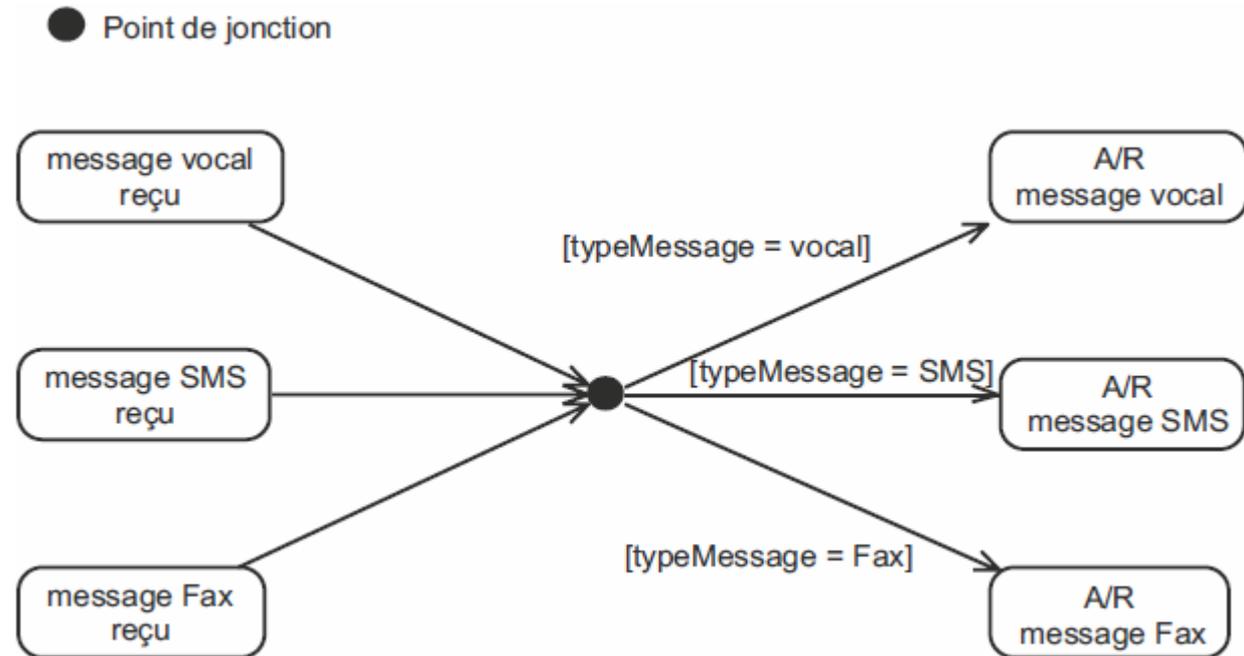
POINT DE JONCTION

- Artefact graphique (pseudo-état) permettant le partage des segments de transition,
- L'objectif est d'aboutir à une notation plus compacte (plus lisible) des chemins alternatifs.
- Un point de jonction peut avoir plusieurs segments de transition entrante et plusieurs segments de transition sortante.
- Par contre, il ne peut avoir d'activité interne ni des transitions sortantes dotées de déclencheurs d'événements.
- Il ne s'agit pas d'un état qui peut être actif au cours d'un laps de temps fini.
- Lorsqu'un chemin passant par un point de jonction est emprunté (transition associée est déclenchée) toutes les gardes le long de ce chemin doivent s'évaluer à vrai dès le franchissement du premier segment.

COMPLÉMENTS SUR LE DIAGRAMME D'ÉTAT-TRANSITION

POINT DE JONCTION

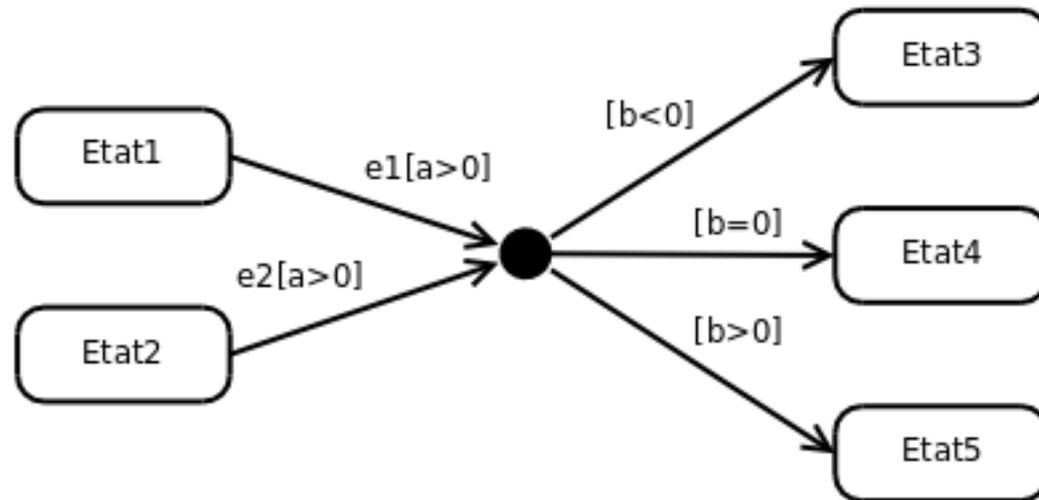
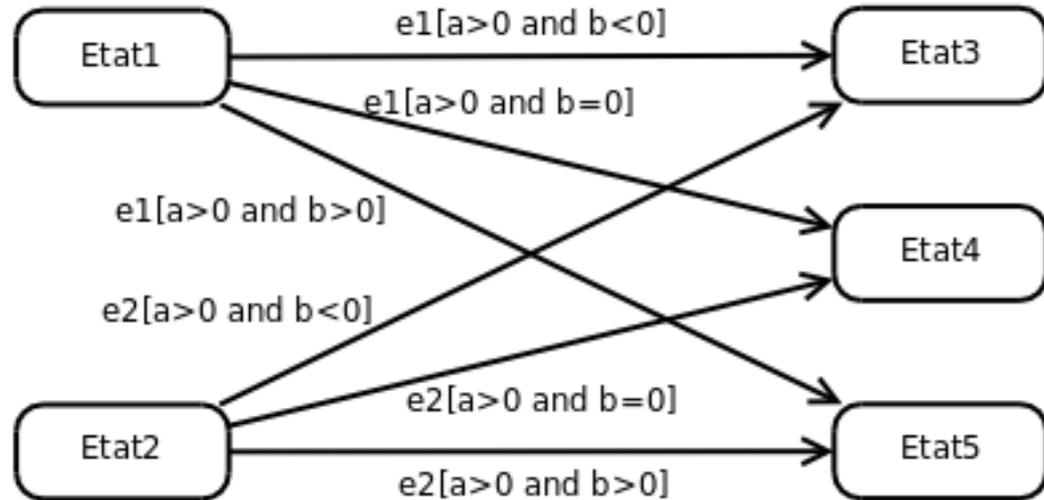
- Lorsque l'on veut relier plusieurs états vers d'autres états, un **point de jonction** permet de décomposer une transition en deux parties.
- À l'exécution, **un seul parcours sera emprunté**, c'est celui pour lequel toutes les conditions de garde seront satisfaites.
- **Exemple:**



COMPLÉMENTS SUR LE DIAGRAMME D'ÉTAT-TRANSITION

POINT DE JONCTION

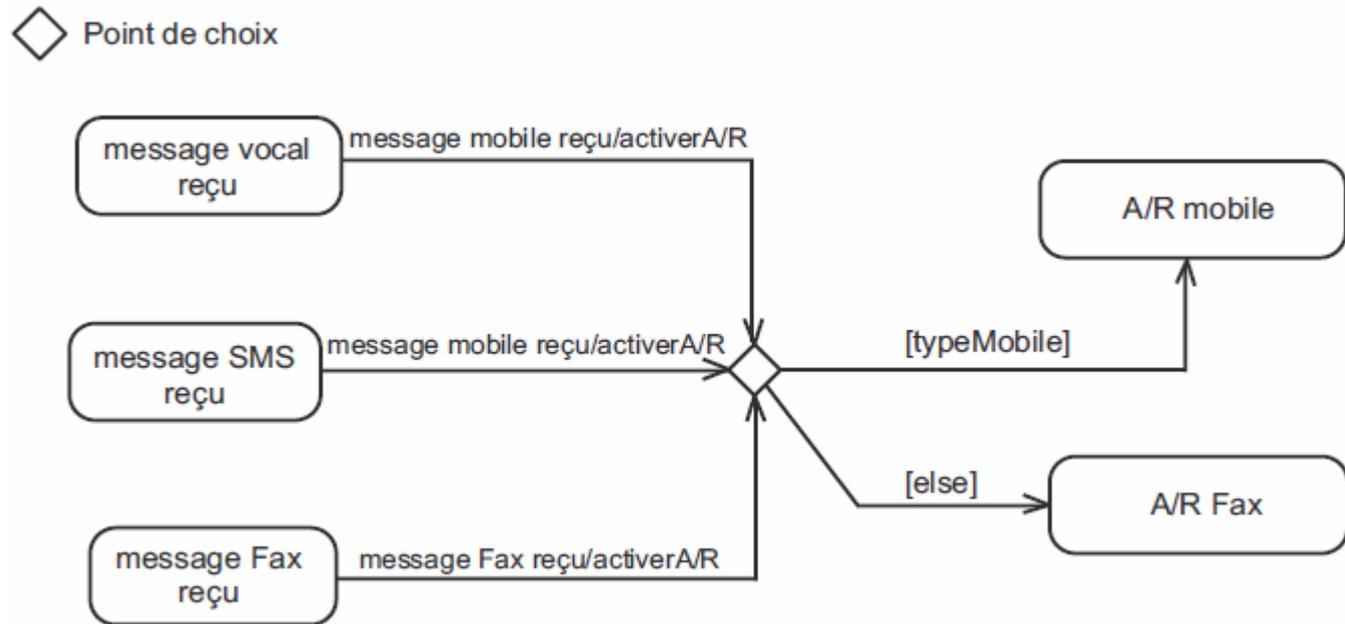
- Exemple 2:



COMPLÉMENTS SUR LE DIAGRAMME D'ÉTAT-TRANSITION

POINT DE CHOIX (DE DÉCISION)

- Le **point de choix** se comporte comme un test de type :
si condition faire action1 sinon faire action2.
- Formalisme et Exemple:



COMPLÉMENTS SUR LE DIAGRAMME D'ÉTAT-TRANSITION

ÉTAT HISTORIQUE

- Un état historique, également qualifié d'*état historique plat*, est un pseudo-état qui mémorise le dernier sous-état actif d'un état composite.
- Graphiquement, il est représenté par un cercle contenant un *H*.
- **Exemple:**

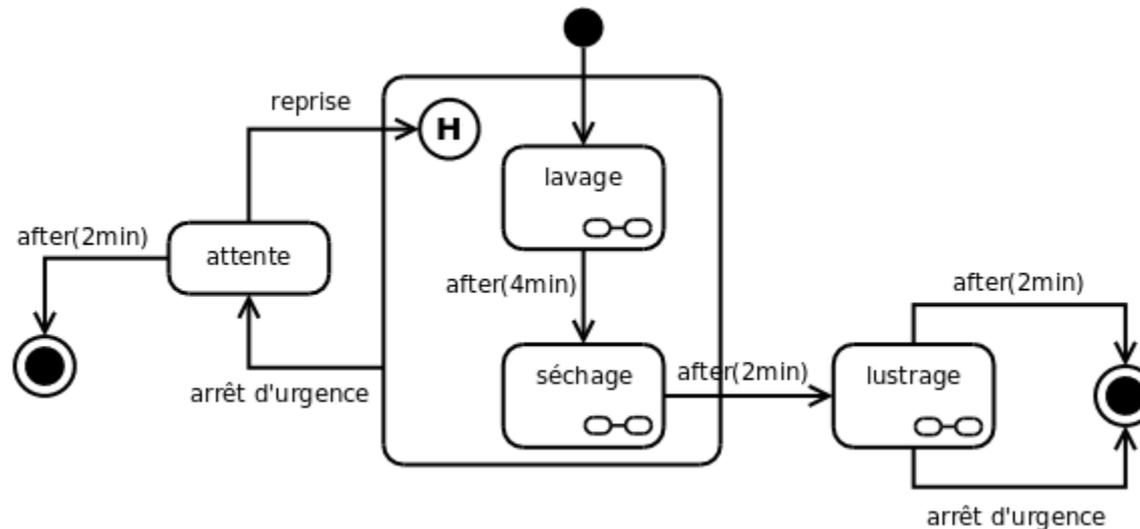


DIAGRAMME D'ÉTAT-TRANSITION (DET)

EXERCICE : ENNONCÉ

- Soit à représenter le diagramme d'état-transition d'un objet personnel en suivant les événements de gestion depuis le recrutement jusqu'à la mise en retraite.

Après le recrutement, une personne est considérée en activité dès sa prise de fonction dans l'entreprise. Au cours de sa carrière, nous retiendrons seulement les événements : congé de maladie et prise de congé annuel. En fin de carrière, nous retiendrons deux situations : la démission et la retraite.

DIAGRAMME D'ÉTAT-TRANSITION (DET)

EXERCICE: CORRIGÉ

- Pour cette proposition de solution 'type', nous avons retenu les états caractéristiques : recruté, activité, en congé, en arrêt, parti et retraite.

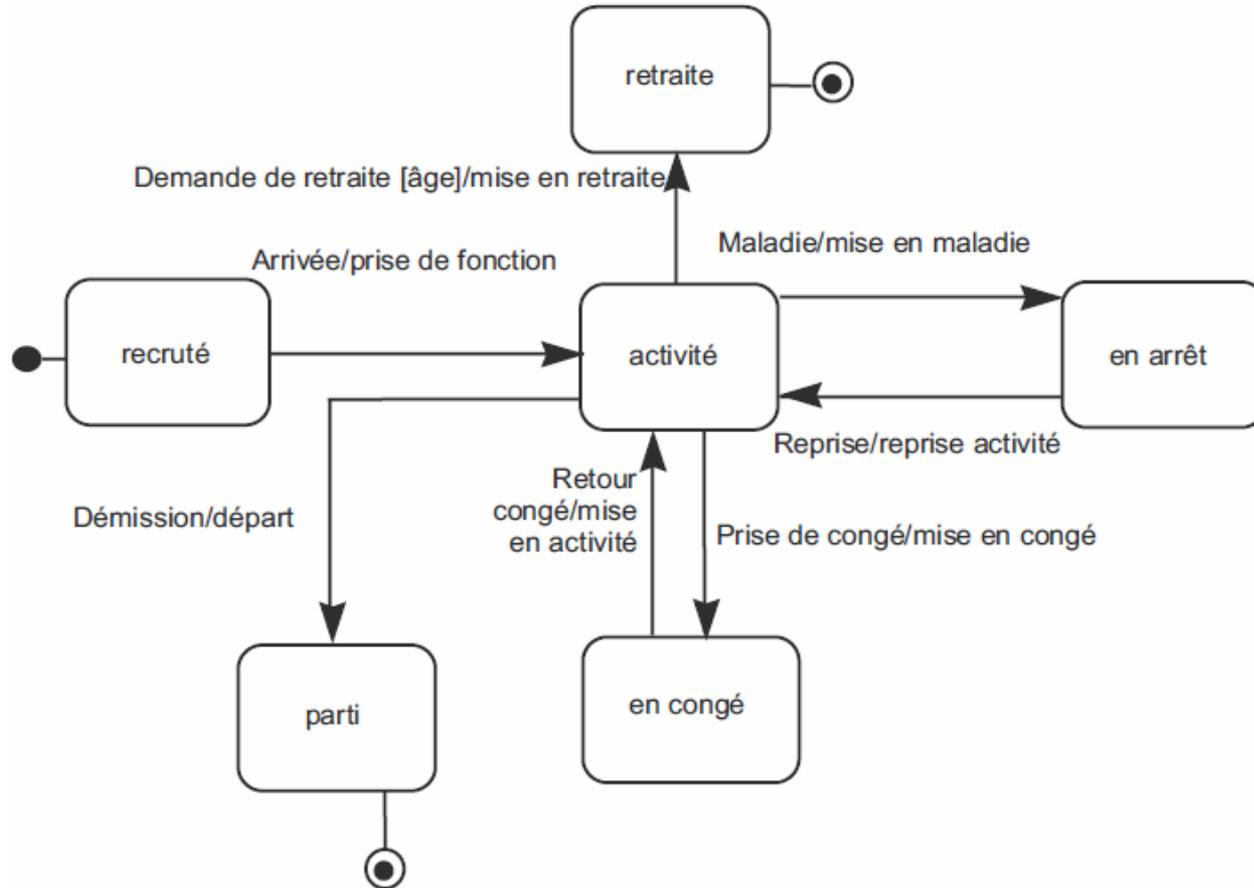


DIAGRAMME D'ACTIVITÉ (DAC)

- Les diagrammes d'activités permettent de mettre l'accent sur les traitements.
- Ils montrent l'enchaînement des activités qui concourent au processus.
- Présente des points communs avec le DET puisqu'il concerne le comportement interne des opérations ou des cas d'utilisation.
- Cependant le comportement visé ici s'applique aux flots de contrôle et aux flots de données propres à un ensemble d'activités et non plus relativement à une seule classe.

DIAGRAMME D'ACTIVITÉ (DAC)

CONCEPTS DE BASE :

- Les concepts communs (ou très proches) entre DAC et DET sont :
 - transition,
 - ● noeud initial (état initial),
 - ⊙ noeud final (état final),
 - ⊗ noeud de fin flot (état de sortie),
 - ◇ noeud de décision (choix).

Le formalisme reste identique pour ces noeuds de contrôle

- Les concepts spécifiques au diagramme d'activité sont :
 - noeud de **bi**furcation,
 - noeud de jonction,
 - noeud de fusion,
 - pin d'entrée et de sortie,
 - flot d'objet,
 - partition.

DIAGRAMME D'ACTIVITÉ (DAC)

ACTION:

- Une action est le plus petit traitement qui puisse être exprimé en UML.
- Une action a une incidence sur l'état du système ou en extrait une information.
- Les actions sont des étapes discrètes à partir desquelles se construisent les comportements.
- La notion d'action est à rapprocher de la notion d'instruction élémentaire d'un langage de programmation (comme C++ ou Java).
- Une action peut être, par exemple :
 - une affectation de valeur à des attributs ;
 - un accès à la valeur d'une propriété structurelle (attribut...);
 - la création d'un nouvel objet ou lien ;
 - un calcul arithmétique simple ;
 - l'émission d'un signal ;
 - la réception d'un signal ;
 - ...

DIAGRAMME D'ACTIVITÉ (DAC)

ACTION :

- **Formalisme et exemple :**

- Une action est représentée par un rectangle dont les coins sont arrondis comme pour les états du DET.



TRANSITION ET FLOT DE CONTROLE

- Dès qu'une action est achevée, une **transition** automatique est déclenchée vers l'action suivante.
- Il n'y a pas d'événement associé à la transition.
- L'enchaînement des actions constitue le **flot de contrôle**.
- **Formalisme et exemple :**

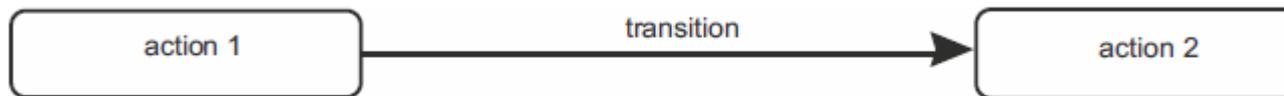


Figure – Formalisme de base du diagramme d'activité : actions et transition

DIAGRAMME D'ACTIVITÉ (DAC)

ACTIVITÉ :

- Définit un comportement décrit par un séquençement organisé d'unités dont les éléments simples sont les actions.
- Le flot d'exécution est modélisé par des nœuds reliés par les transitions.
- Le flot de contrôle reste dans l'activité jusqu'à ce que les traitements soient terminés.
- Une activité est composée de trois types de nœuds :
 - nœud d'exécution (action, transition),
 - nœud de contrôle:
 - nœud initial,
 - nœud final,
 - flux de sortie,
 - nœud de bifurcation,
 - nœud de jonction,
 - nœud de test-décision,
 - nœud de fusion-test,
 - pin d'entrée et de sortie,
 - nœud d'objet.
- Une activité est un comportement → peut recevoir des paramètres en entrée et en produire en sortie.

DIAGRAMME D'ACTIVITÉ (DAC)

ACTIVITÉ :

- Formalisme et exemple de représentation d'une activité:

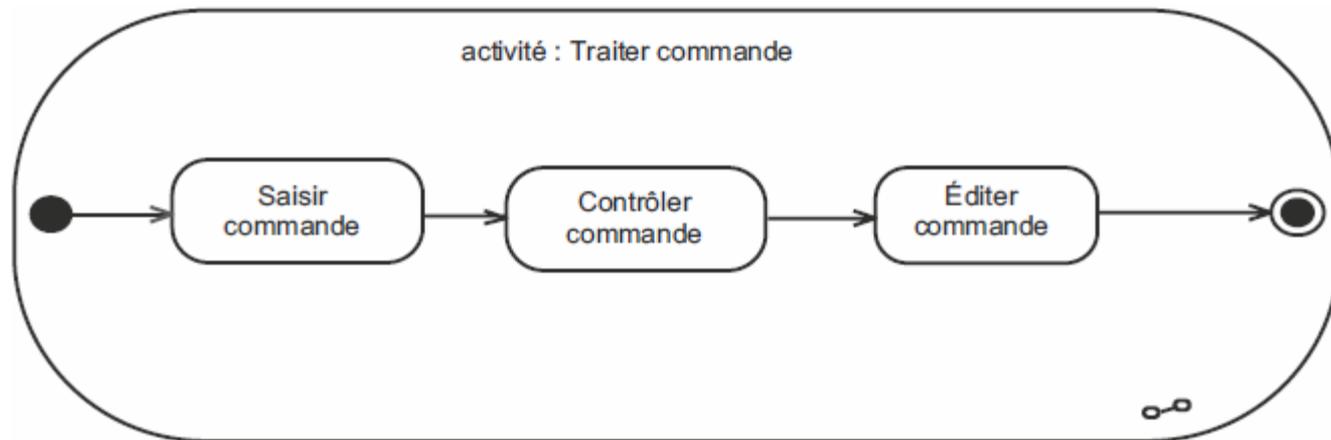


DIAGRAMME D'ACTIVITÉ (DAC)

NOEUD DE BIFURCATION (FOURCHE) :

- Un nœud de bifurcation, également appelé nœud de débranchement est un nœud de **contrôle** qui sépare un flot en plusieurs flots concurrents.
- Un tel nœud possède donc un arc entrant et plusieurs arcs sortants.
- Graphiquement, on représente un nœud de bifurcation par un trait plein.

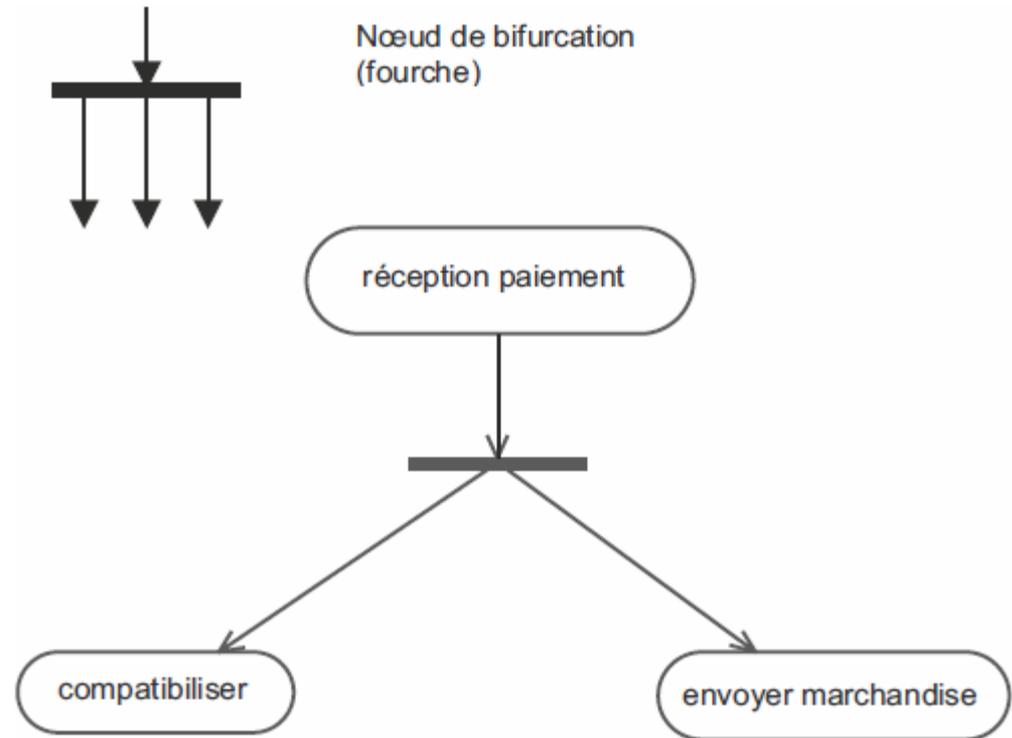


Figure — Exemple 1 d'activités avec nœud de bifurcation

DIAGRAMME D'ACTIVITÉ (DAC)

NOEUD DE BIFURCATION (FOURCHE) :

- Exemple 2 :

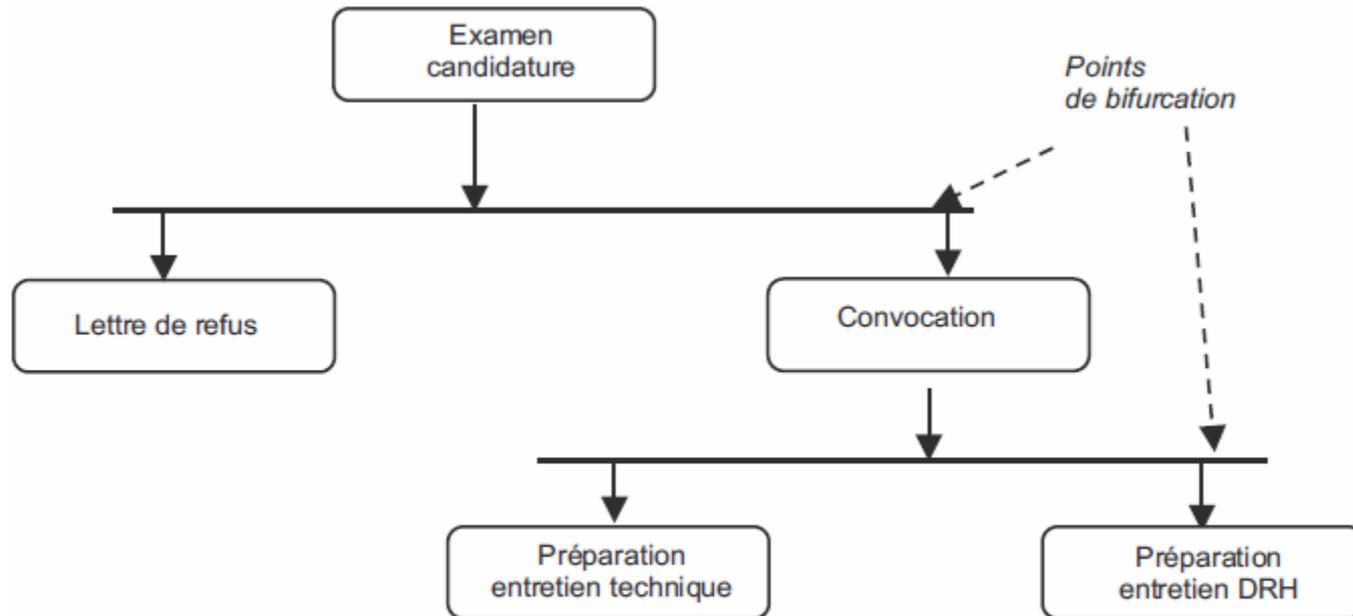


Figure — Exemple 2 de diagramme d'activité avec bifurcation de flots de contrôle

DIAGRAMME D'ACTIVITÉ (DAC)

NOEUD DE JONCTION (SYNCHRONISATION) :

- nœud de jonction est un nœud de contrôle qui synchronise des flots multiples.
- Un tel nœud possède plusieurs arcs entrants et un seul arc sortant.
- Graphiquement, on représente un nœud de jonction par un trait plein.

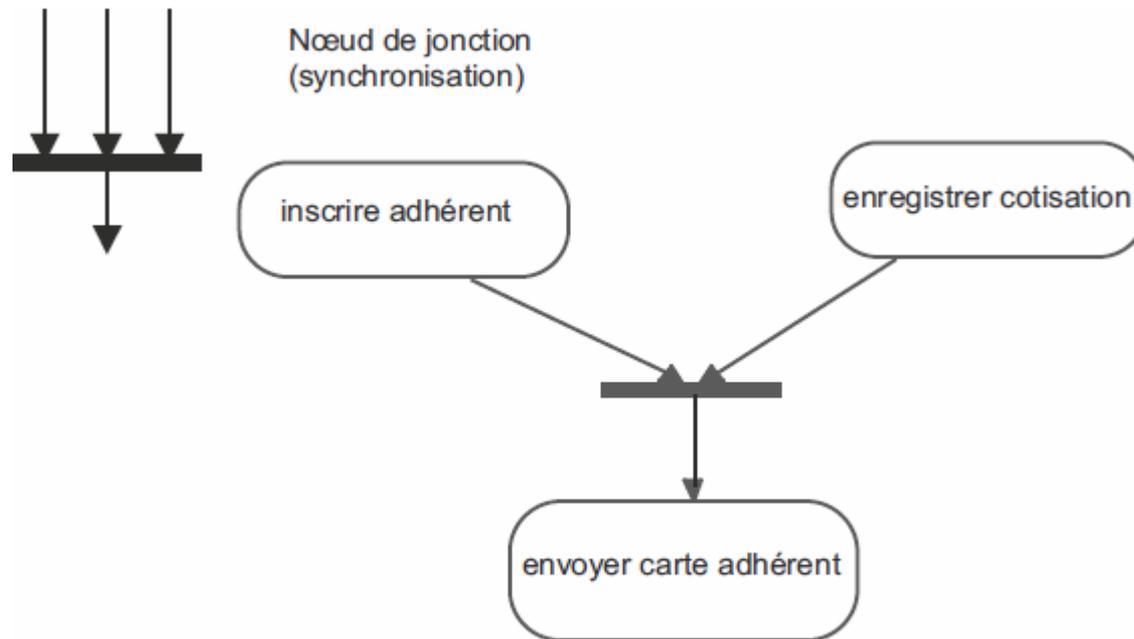


Figure — Exemple d'activités avec nœud de jonction

DIAGRAMME D'ACTIVITÉ (DAC)

NOEUD DE DÉCISION (TEST-DÉCISION) :

- Un noeud de test-décision est un nœud de contrôle qui permet de faire un choix entre plusieurs flots sortants en fonction des conditions de garde de chaque flot.
- Un noeud de test-décision n'a qu'un seul flot en entrée.
- On peut aussi utiliser seulement deux flots de sortie :
 - le premier correspondant à la condition vérifiée
 - l'autre traitant le cas sinon.
- Graphiquement, on représente un nœud de décision par un losange :



symbole du nœud de décision-test

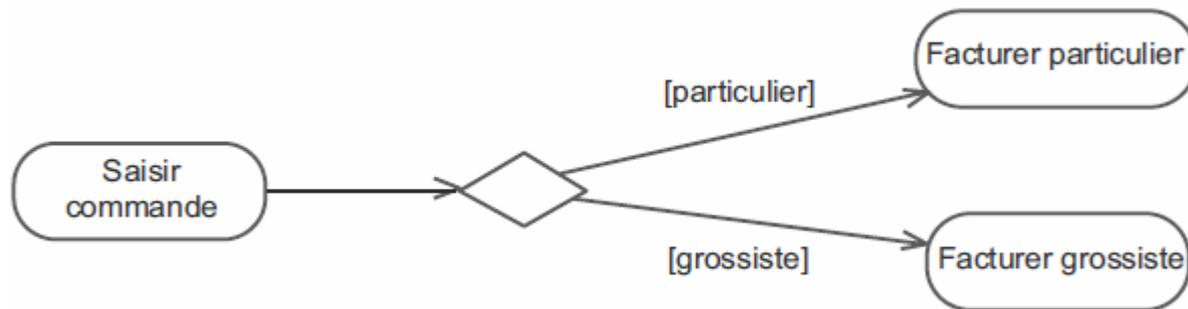


Figure – Formalisme et exemple 1 d'activités avec nœud de test-décision

DIAGRAMME D'ACTIVITÉ (DAC)

NOEUD DE DÉCISION (TEST-DÉCISION) :

- Exemple 2 :

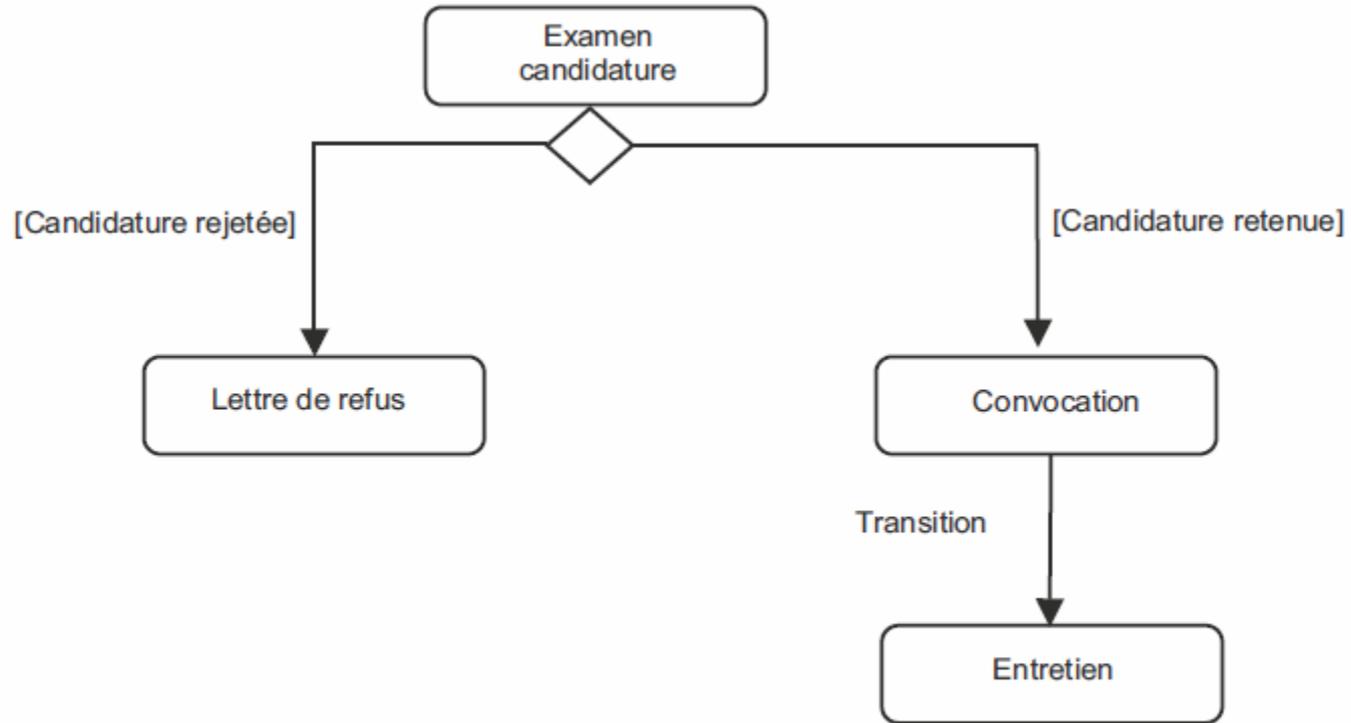


Figure — Exemple 2 de diagramme d'activités avec un nœud de test-décision

DIAGRAMME D'ACTIVITÉ (DAC)

NOEUD DE FUSION-TEST :

- Un nœud de fusion est un nœud de contrôle qui rassemble plusieurs flots alternatifs entrants en un seul flot sortant.
- Il n'est pas utilisé pour synchroniser des flots concurrents (c'est le rôle du nœud de jonction), mais pour accepter un flot parmi plusieurs.
- Le flot sortant est donc exécuté dès qu'**un des flots** entrants est activé.
- Graphiquement, on représente un nœud de fusion par un losange.

DIAGRAMME D'ACTIVITÉ (DAC)

NOEUD DE FUSION-TEST :

- Formalisme et Exemple :

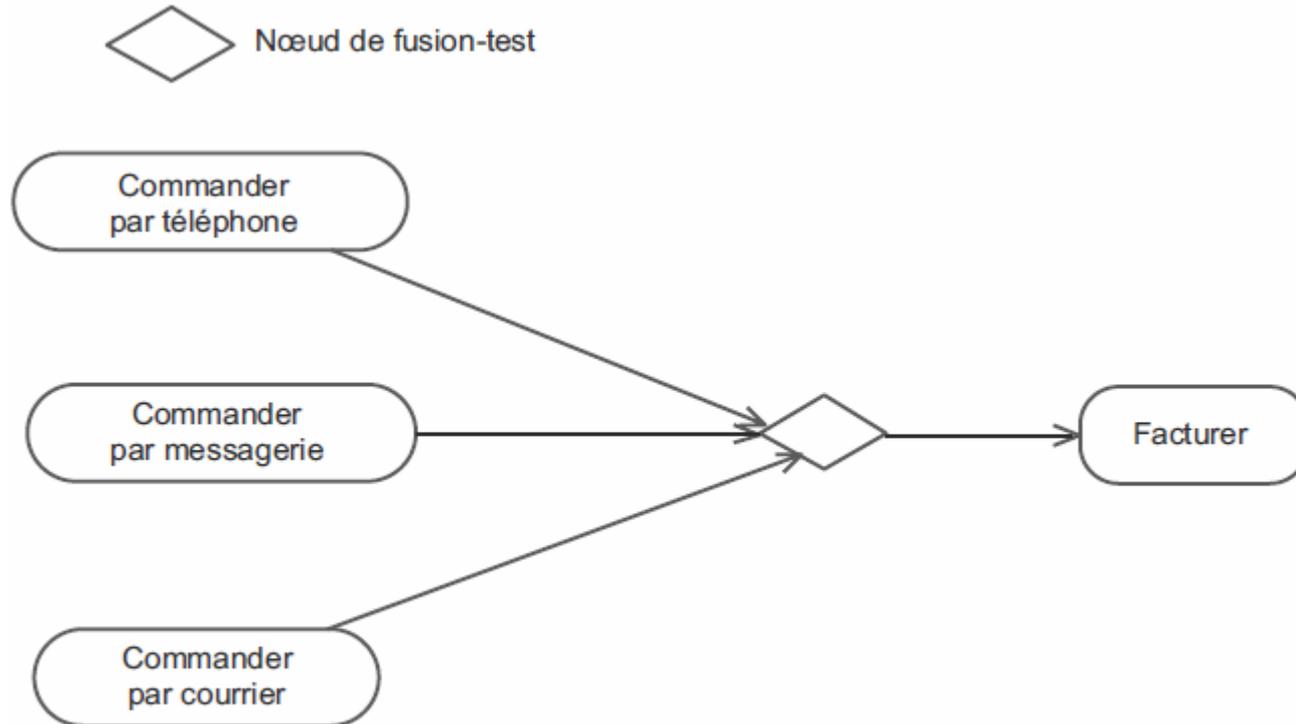


Figure — Formalisme et exemple de diagramme d'activités avec un nœud de fusion-test

DIAGRAMME D'ACTIVITÉ (DAC)

REMARQUES :

- Graphiquement il est possible de:
 - fusionner un nœud de fusion et un nœud de décision, et donc d'avoir un losange possédant plusieurs arcs entrants et sortants. Il est également possible de fusionner un nœud de décision ou de fusion avec un autre nœud, comme un nœud de fin de flot sur la figure, ou avec une activité. Cependant, pour mieux mettre en évidence un branchement conditionnel, il est préférable d'utiliser un nœud de décision (losange).
 - fusionner un nœud de bifurcation et un nœud d'union, et donc d'avoir un trait plein possédant plusieurs arcs entrants et sortants.

DIAGRAMME D'ACTIVITÉ (DAC)

PIN D'ENTRÉE ET DE SORTIE:

- Un pin d'entrée ou de sortie représente un paramètre que l'on peut spécifier en entrée ou en sortie d'une action.
- Un nom de donnée et un type de donnée peuvent être associés au pin.
- Un paramètre peut être de type objet.
- Formalisme et exemple:
 - Chaque paramètre se représente dans un petit rectangle.
 - Le nom du paramètre ainsi que son type sont aussi à indiquer.

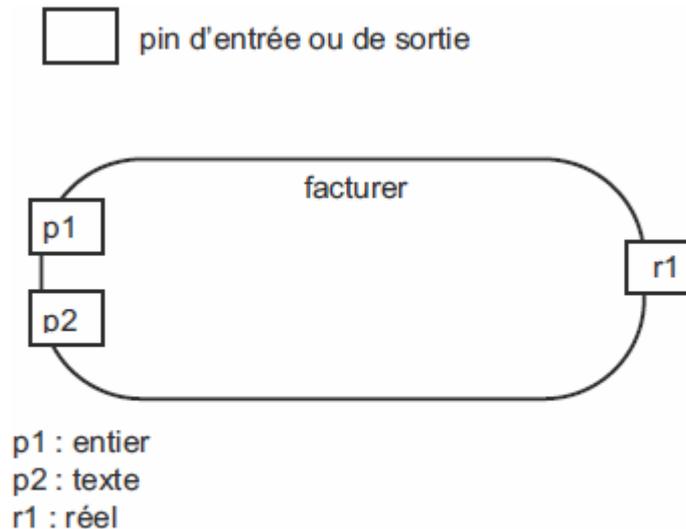


Figure – Formalisme et exemple d'activité avec pin d'entrée et de sortie

DIAGRAMME D'ACTIVITÉ (DAC)

FLOT DE DONNÉES ET NOEUD D'OBJET:

- Un noeud d'objet permet de représenter le flot de données véhiculé entre les actions.
- Les objets peuvent se représenter de deux manières différentes :
 - en utilisant le pin d'objet
 - en représentant explicitement un objet.
- Formalisme et exemple de représentation de flot de données et noeud d'objet:

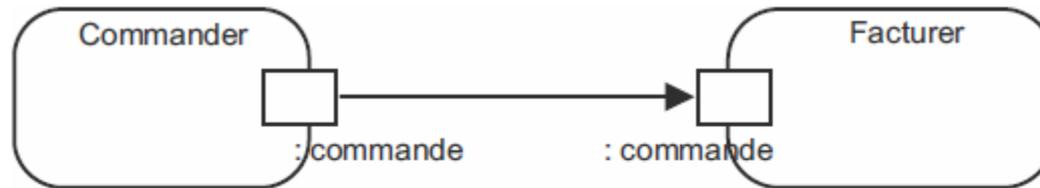


DIAGRAMME D'ACTIVITÉ (DAC)

PARTITION :

- UML permet d'organiser la présentation du diagramme d'activité en couloir d'activités.
- Chaque couloir correspond à un domaine de responsabilité d'un certain nombre d'actions.
- Les flots d'objets sont aussi représentés dans le diagramme.
- L'ordre relatif des couloirs de responsabilité n'est pas significatif.

REPRÉSENTATION DU DIAGRAMME D'ACTIVITÉ

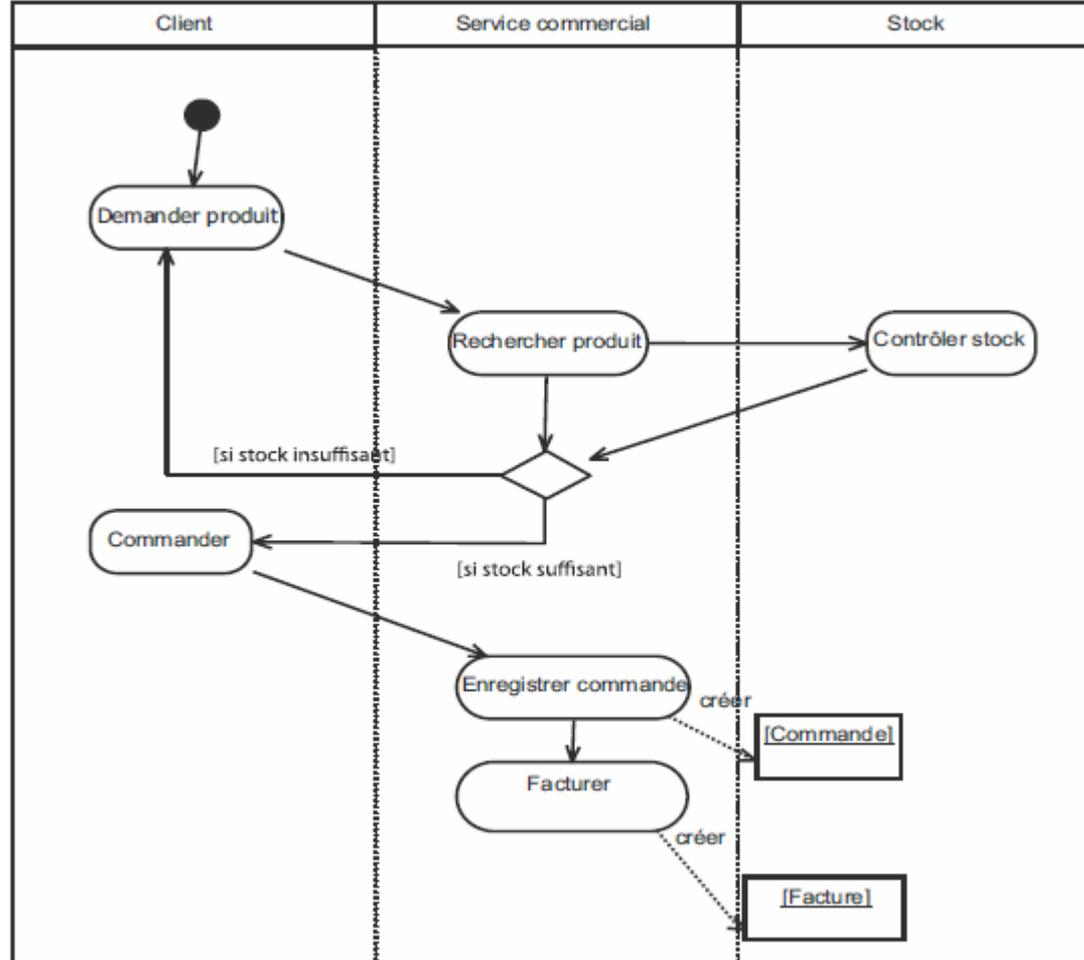


Figure — Exemple de diagramme d'activité avec couloir d'activité

DIAGRAMME D'ACTIVITÉ (DAC)

REPRÉSENTATION D' ACTIONS DE COMMUNICATION:

- Dans un diagramme d'activité, (comme dans un diagramme de temps), des interactions de communication liées à certains types d'événement peuvent se représenter.
- Les types d'événement concernés:
 - signal,
 - écoulement du temps.
- Formalisme et exemple:

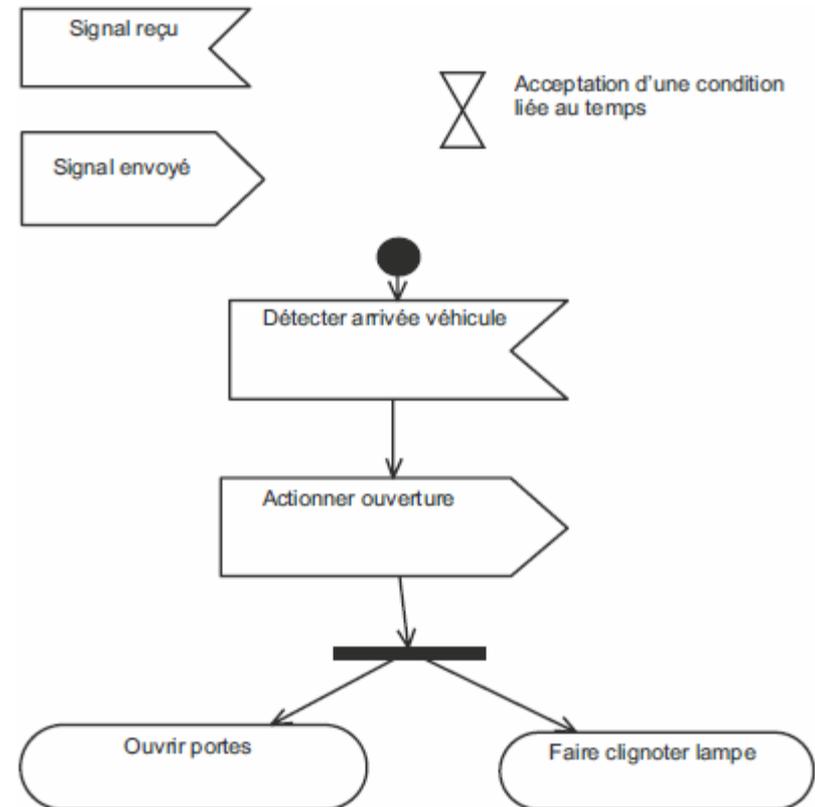


Figure – Formalisme et exemple de diagramme d'activité avec actions de communication

DIAGRAMME D'ACTIVITÉ (DAC)

EXERCICE : ENNONCÉ

En reprenant l'exercice relatif à la gestion de la bibliothèque traité dans les cas d'utilisation nous pouvons élaborer le diagramme d'activité correspondant.

- Deux acteurs ont été identifiés :
 - Bibliothécaire chargé de l'approvisionnement des ouvrages, de la gestion du catalogue et de l'enregistrement des emprunts et retours d'ouvrages ;
 - Gestionnaire, chargé de l'inscription des adhérents et de la relance des adhérents ayant dépassé le délai de restitution des ouvrages.

DIAGRAMME D'ACTIVITÉ (DAC)

EXERCICE : CORRIGÉ

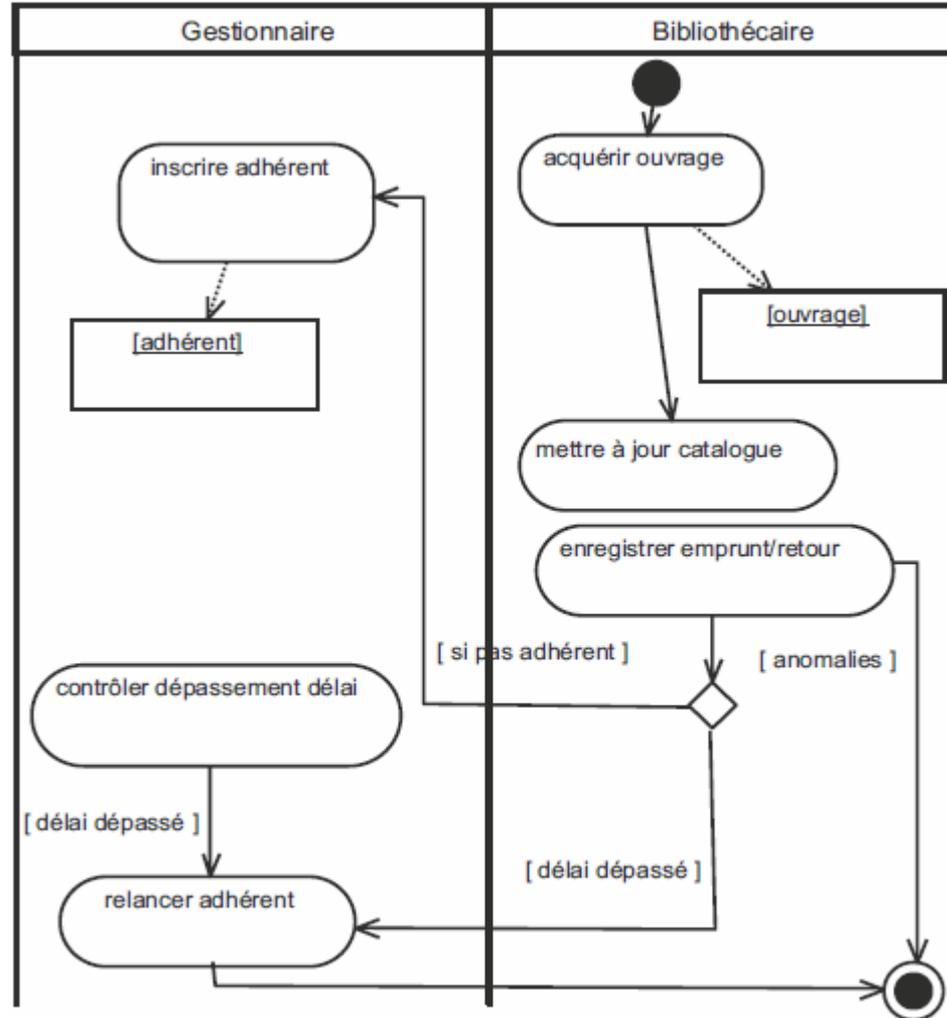


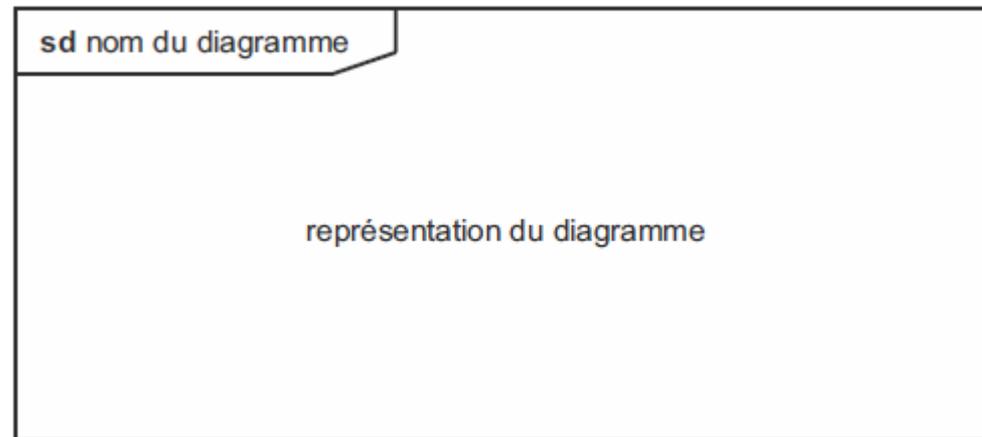
Figure — Diagramme d'activité de l'exercice 1

DIAGRAMMES COMPORTEMENTAUX

- Les diagrammes comportementaux sont focalisés sur la description de la partie dynamique du système à modéliser.
- 7 diagrammes sont proposés par UML 2 :
 - diagramme des cas d'utilisation (DCU),
 - diagramme d'état-transition (machine d'état, DET),
 - diagramme d'activité (DAC),
 - Diagrammes d'interaction:
 - **diagramme de séquence (DSE)** ,
 - diagramme de communication (DCO),
 - diagramme global d'interaction (DGI),
 - diagramme de temps (DTP).

DIAGRAMME DE SÉQUENCE (DSE)

- L'objectif du diagramme de séquence est de représenter les interactions entre objets en indiquant la chronologie des échanges.
- Cette représentation peut se réaliser par cas d'utilisation en considérant les différents scénarios associés.
- Un diagramme de séquence se représente globalement dans un grand rectangle avec indication du nom du diagramme en haut à gauche :



sd : abréviation de « sequence diagramm »

Figure — Formalisme général du cadre d'un diagramme de séquence

DIAGRAMME DE SÉQUENCE (DSE)

- **Remarques :**

- Un DSE indique les objets que l'acteur va manipuler et les opérations qui font passer d'un objet à l'autre.
- On peut représenter les mêmes opérations par un diagramme de communication (cf. plus loin).
- Le DSE et diagramme de communication sont deux vues différentes, mais logiquement équivalentes (on peut construire l'une à partir de l'autre) d'une même chronologie. Ce sont des diagrammes d'interaction.
- Contrairement au diagramme de communication, le temps est représenté explicitement dans un DSE par une dimension (la dimension verticale) et s'écoule de haut en bas.

DIAGRAMME DE SÉQUENCE (DSE)

- **Ligne de vie:**
 - Une ligne de vie représente l'ensemble des opérations exécutées par un objet.
 - Un message reçu par un objet déclenche l'exécution d'une opération.
 - Le retour d'information peut être implicite (cas général) ou explicite à l'aide d'un message de retour.
- **Message synchrone et asynchrone**
 - Dans un diagramme de séquence, deux types de messages peuvent être distingués :
 - **Message synchrone:** l'émetteur reste en attente de la réponse à son message avant de poursuivre ses actions.
La flèche avec extrémité **pleine** symbolise ce type de message. Le message retour peut ne pas être représenté car il est inclus dans la fin d'exécution de l'opération de l'objet destinataire du message.
 - **Message asynchrone:** l'émetteur n'attend pas la réponse à son message, il poursuit l'exécution de ses opérations.
C'est une flèche avec une extrémité **non pleine** qui symbolise ce type de message.

DIAGRAMME DE SÉQUENCE (DSE)

- Formalisme et exemple:

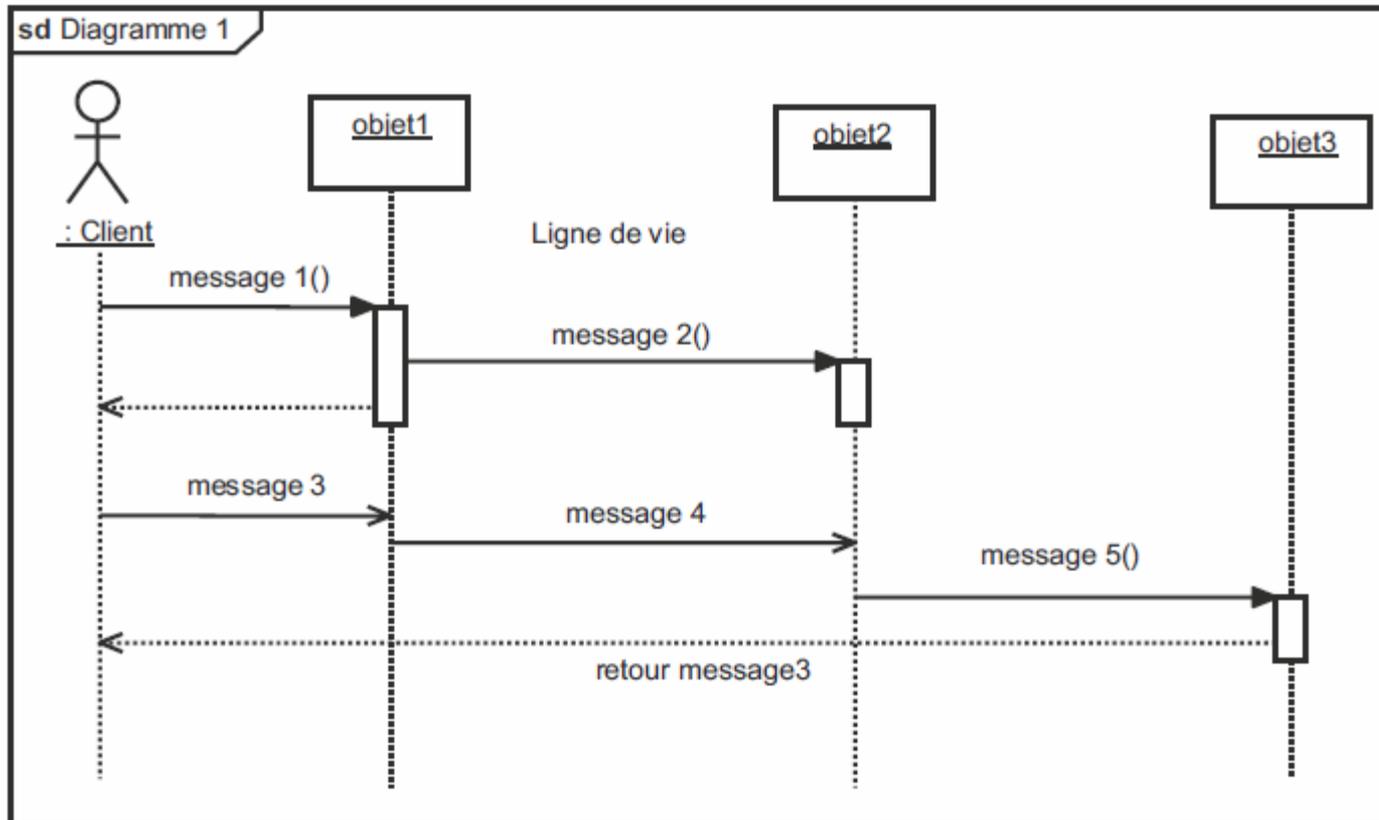


Figure — Formalisme du diagramme de séquence

DSE : OPÉRATIONS PARTICULIÈRES

- Création et destruction d'objet:
 - Si un objet est créé par une opération, celui-ci n'apparaît qu'au moment où il est créé.
 - Si l'objet est détruit par une opération, la destruction se représente par « X ».

Exemple :

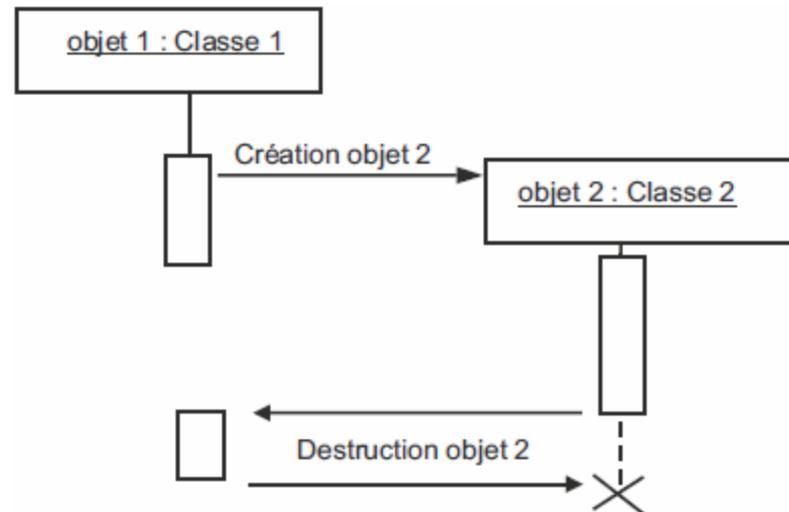


Figure – Exemple type de création et de destruction d'objet

Remarque : Il est aussi possible dans certains outils de modélisation d'indiquer plus simplement la création d'une nouvelle instance d'objet en utilisant le mot-clé « **create** »

DSE : OPÉRATIONS PARTICULIÈRES

- **Contrainte temporelle:**

- Des contraintes de chronologie entre les messages peuvent être spécifiées.
- Lorsque le diagramme de séquence est utilisé pour représenter un sous-ensemble du logiciel à réaliser, il est possible d'indiquer le pseudo-code exécuté par un objet pendant le déroulement d'une opération.
- Lorsque l'émission d'un message requiert une certaine durée, il se représente sous la forme d'un trait oblique.

Exemple:

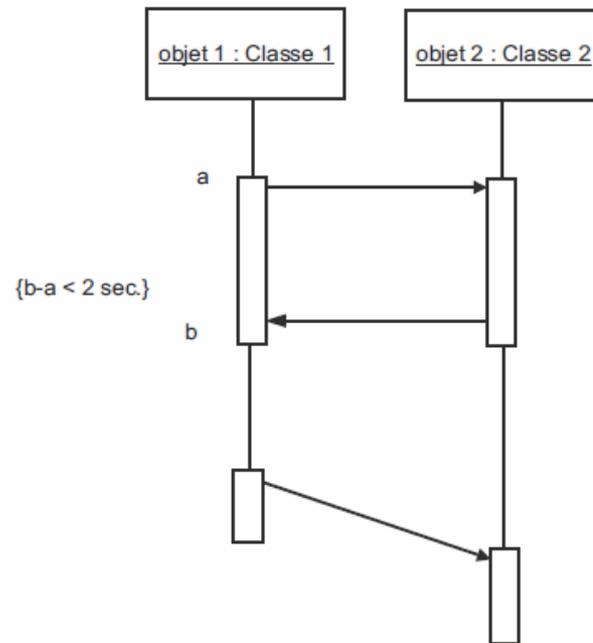


Figure — Exemple type de représentation de contrainte temporelle

DSE : FRAGMENT D'INTERACTION

- Types de fragments d'interaction:
 - Dans un DSE, il est possible de distinguer des sous-ensembles d'interactions qui constituent des fragments.
 - Un **fragment d'interaction se représente globalement comme un diagramme de** séquence dans un rectangle avec indication dans le coin à gauche du nom du fragment.
 - Un port d'entrée et un port de sortie peuvent être indiqués pour connaître la manière dont ce fragment peut être relié au reste du diagramme.
 - Dans le cas où aucun port n'est indiqué c'est l'ensemble du fragment qui est appelé pour exécution.

DSE : FRAGMENT D'INTERACTION

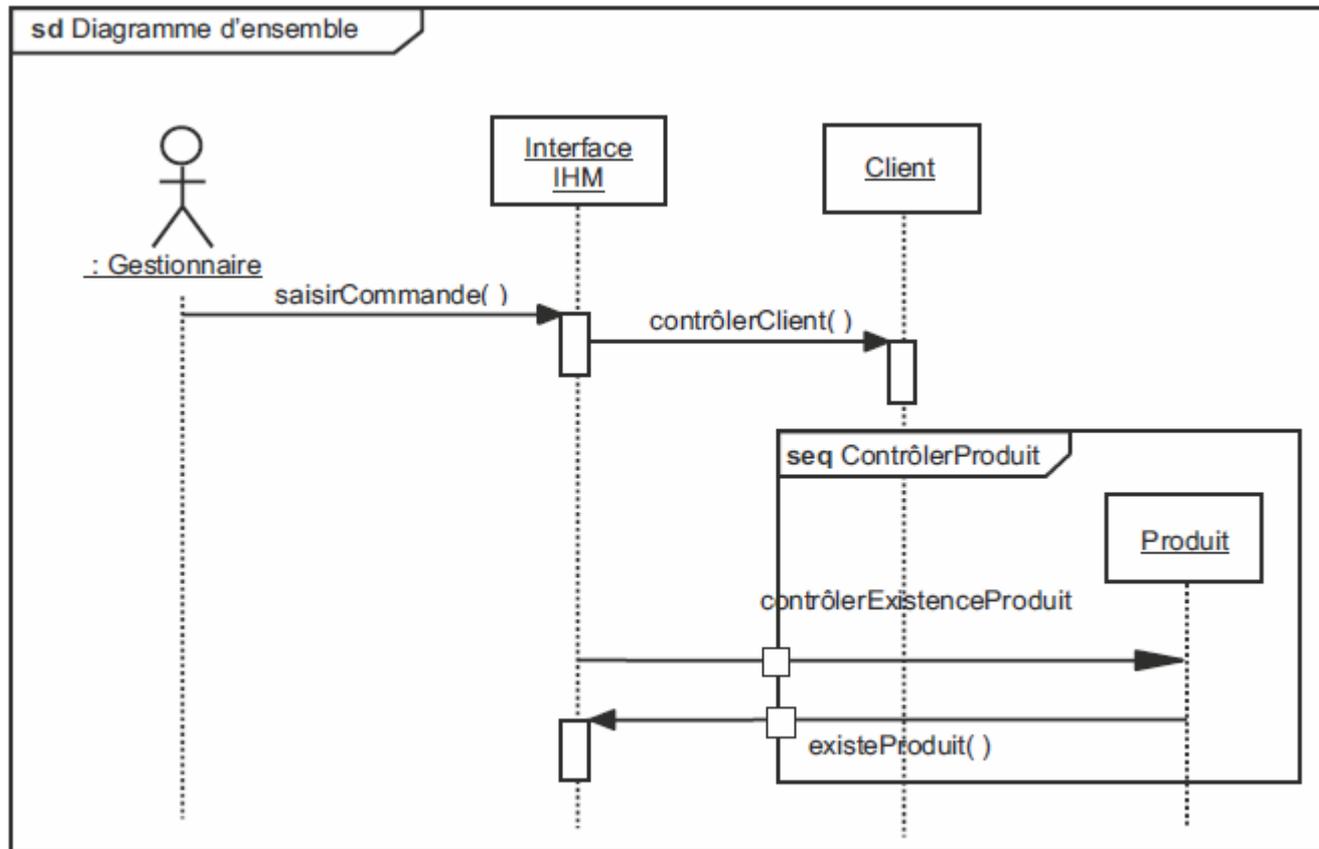


Figure – Exemple de fragment d'interaction avec port d'entrée et de sortie

- Dans cet exemple, le fragment «ContrôlerProduit » est représenté avec un port d'entrée et un port de sortie.

DSE : FRAGMENT D'INTERACTION COMBINÉ

- Un fragment d'interaction dit combiné correspond à un ensemble d'interaction auquel on applique un opérateur.
- Un fragment combiné se représente globalement comme un diagramme de séquence avec indication dans le coin à gauche du nom de l'opérateur.
- La liste suivante regroupe les opérateurs d'interaction par fonctions :
 - les opérateurs de choix et de boucle:
 - **alternative, option, break et loop ;**
 - les opérateurs contrôlant l'envoi en parallèle de messages :
 - **parallel et critical region ;**
 - les opérateurs contrôlant l'envoi de messages :
 - **ignore, consider, assertion et negative ;**
 - les opérateurs fixant l'ordre d'envoi des messages :
 - **weak sequencing , strict sequencing.**

DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateur alt:*

- L'opérateur **alt** correspond à une instruction de test avec une ou plusieurs alternatives possibles.
- Il est aussi permis d'utiliser les clauses de type sinon.
- **Formalisme et exemple:** L'opérateur alt se représente dans un fragment possédant au moins deux parties séparées par des pointillés.

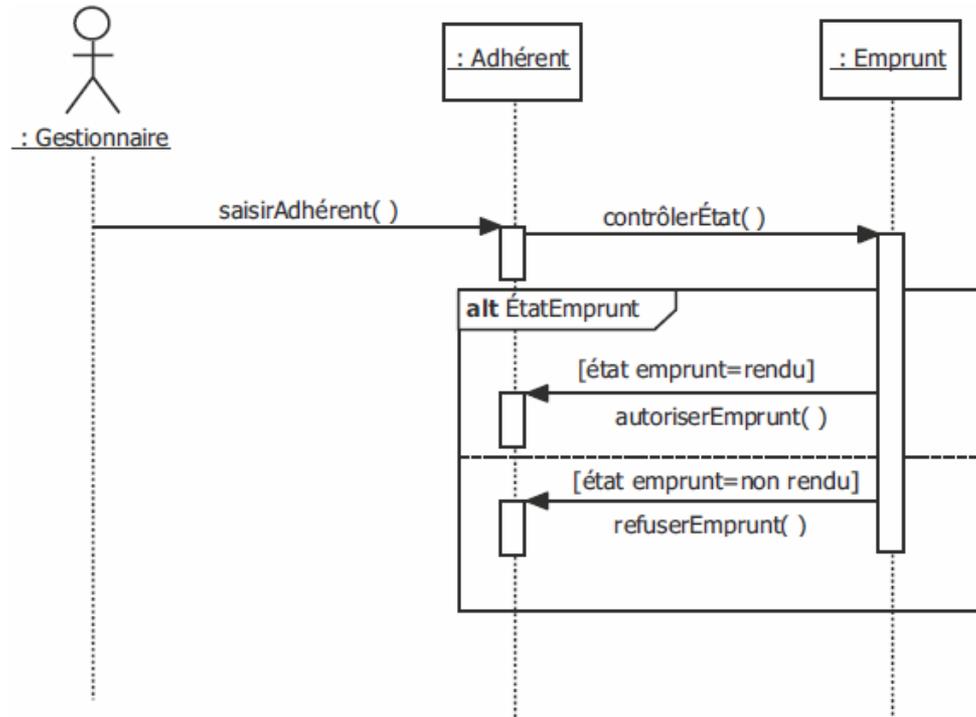


Figure – Exemple de fragment d'interaction avec l'opérateur alt

DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateur opt:*

- L'opérateur **opt** (optional) correspond à une instruction de **test sans alternative** (sinon).
- **Formalisme et exemple**

L'opérateur **opt** se représente dans un fragment possédant une seule partie

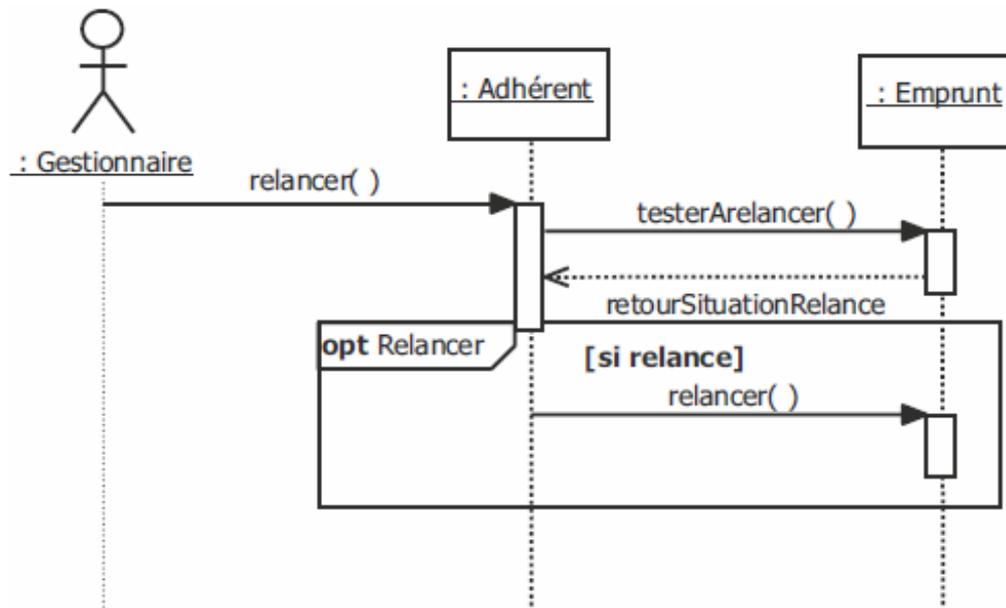


Figure – Exemple de fragment d'interaction avec l'opérateur **opt**

DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateur loop:*
 - L'opérateur **loop** correspond à une instruction de boucle qui **permet d'exécuter une** séquence d'interaction tant qu'une condition est satisfaite.
 - Il est aussi possible d'utiliser une condition portant sur un nombre minimum et maximum d'exécution de la boucle en écrivant : `loop min, max`.
 - Dans ce cas, la boucle s'exécutera au minimum `min` fois et au maximum `max` fois.
 - Il est aussi possible de combiner l'option `min/max` avec la condition associée à la boucle.

DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateur loop:*

- Formalisme et exemple:

- L'opérateur loop se représente dans un fragment possédant une seule partie et englobant toutes les interactions faisant partie de la boucle.

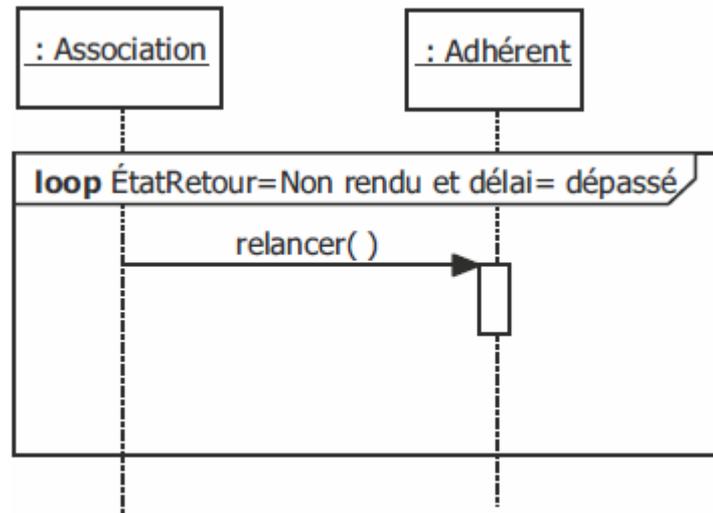


Figure – Exemple de fragment d'interaction avec l'opérateur loop

DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateur par:*
 - L'opérateur **par** (parallel) permet de représenter deux séries d'interactions qui se déroulent en parallèle.
 - **Formalisme et exemple**
 - L'opérateur 'par' se représente dans un fragment possédant deux parties séparées par une ligne en pointillé.

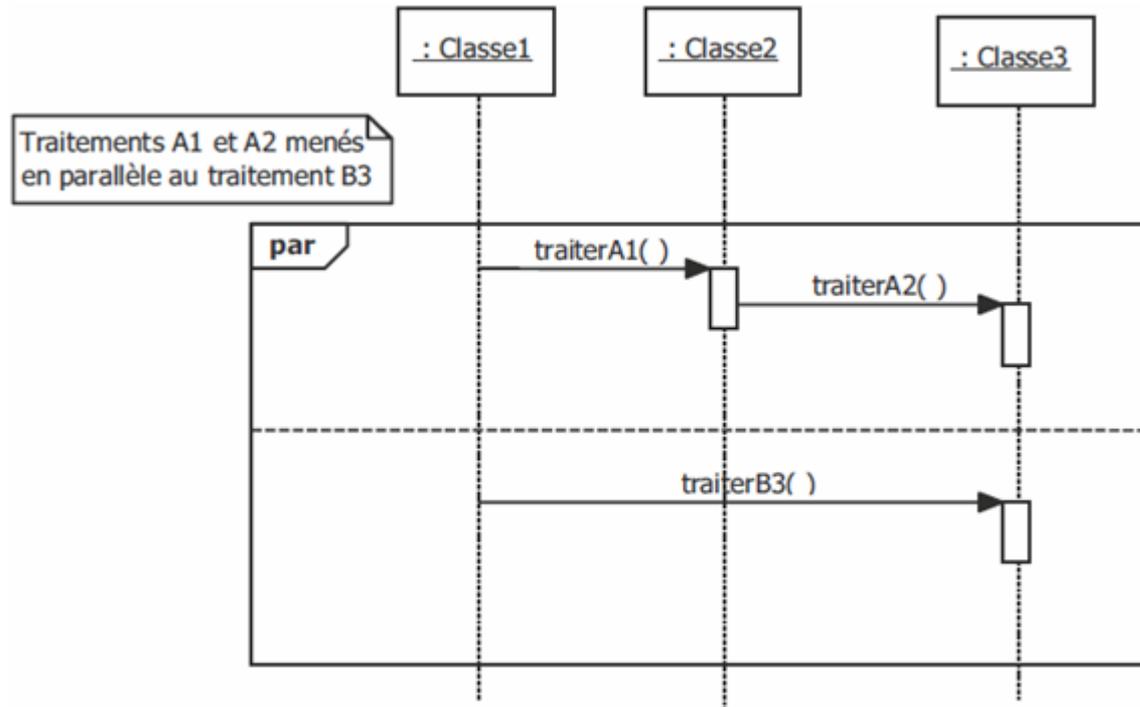


Figure — Exemple de fragment d'interaction avec l'opérateur par

DSE : FRAGMENT D'INTERACTION COMBINÉ

Opérateur Break:

- L'opérateur **break** permet de représenter une situation **exceptionnelle correspondant** à un scénario de rupture par rapport au scénario général.
- Le scénario de rupture s'exécute si la condition de garde est satisfaite.

Les opérations `annulerOp1()`, `annulerOp2()` et `afficherAide()` ne seront exécutées que si la touche F1 est activée sinon le fragment est ignoré et la séquence de traitement passe directement de l'opération `Op2()` à l'opération `Op3()`

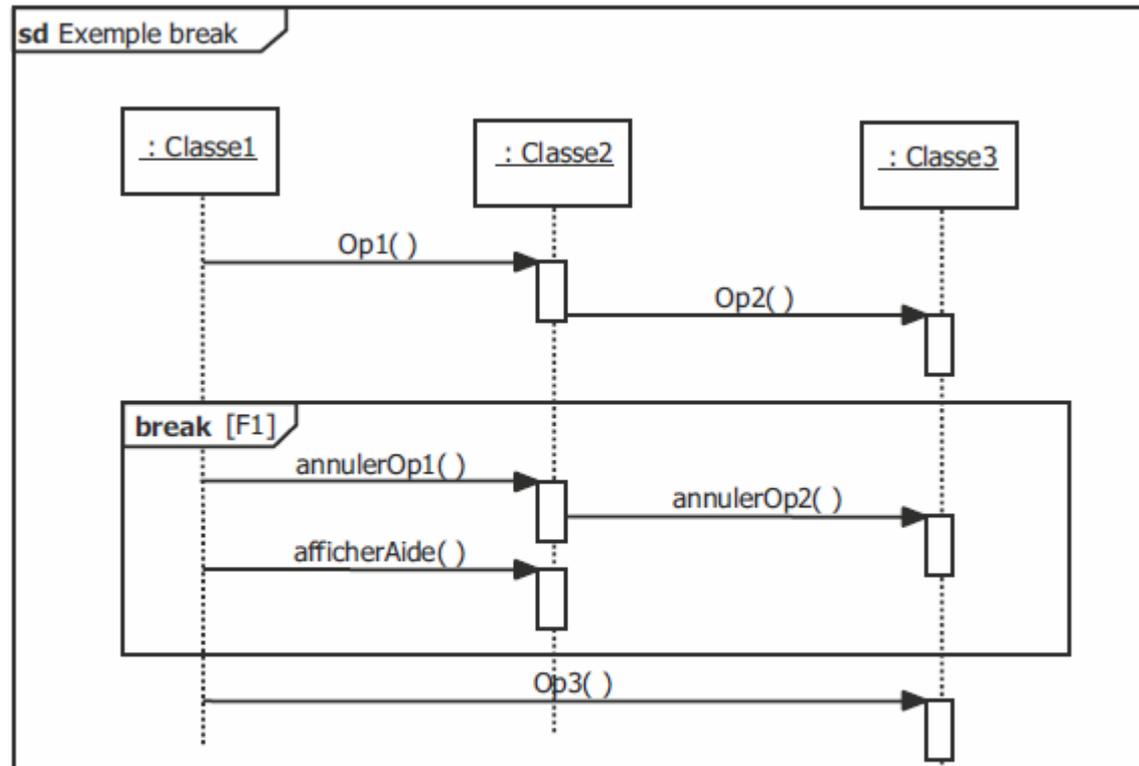


Figure 3.48 — Exemple de fragment d'interaction avec l'opérateur break

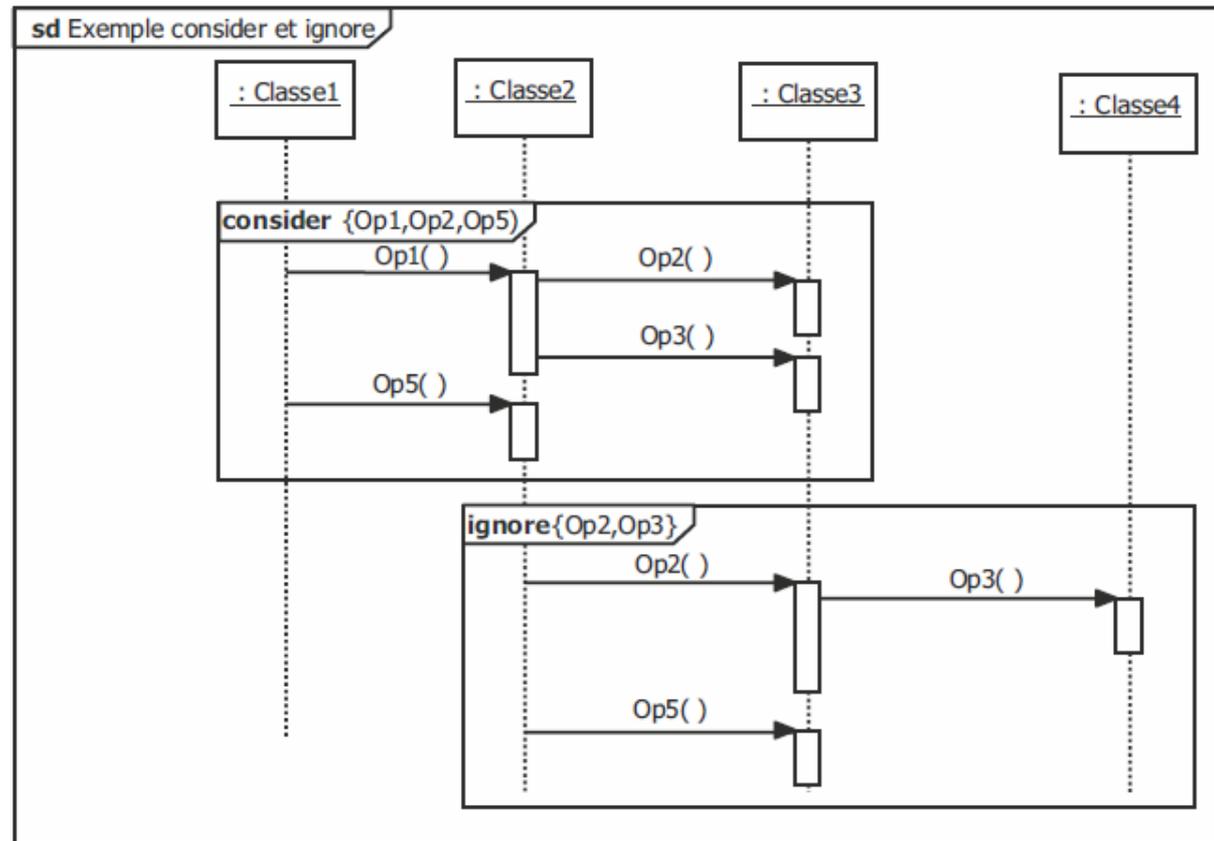
DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateurs ignore et consider :*

- Les opérateurs **ignore et consider** sont utilisés pour des fragments d'interactions dans lesquels on veut montrer que certains messages peuvent être soit absents sans avoir d'incidence sur le déroulement des interactions (ignore), soit obligatoirement présents (consider).

- Dans le fragment **consider**, les messages Op1, Op2 et Op5 doivent être obligatoirement présents lors de l'exécution du fragment sinon le fragment n'est pas exécuté

- Dans le fragment **ignore**, les messages Op2 et Op3 peuvent être absents lors de l'exécution du fragment.



DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateurs critical:*
 - L'opérateur **critical** permet d'indiquer qu'une séquence d'interactions ne peut être interrompue compte tenu du caractère critique des opérations traitées.
 - On considère que le traitement des interactions comprises dans la séquence critique est atomique.

- les opérations Op1(), Op2() et Op3() du fragment critical doivent s'exécuter sans interruption.

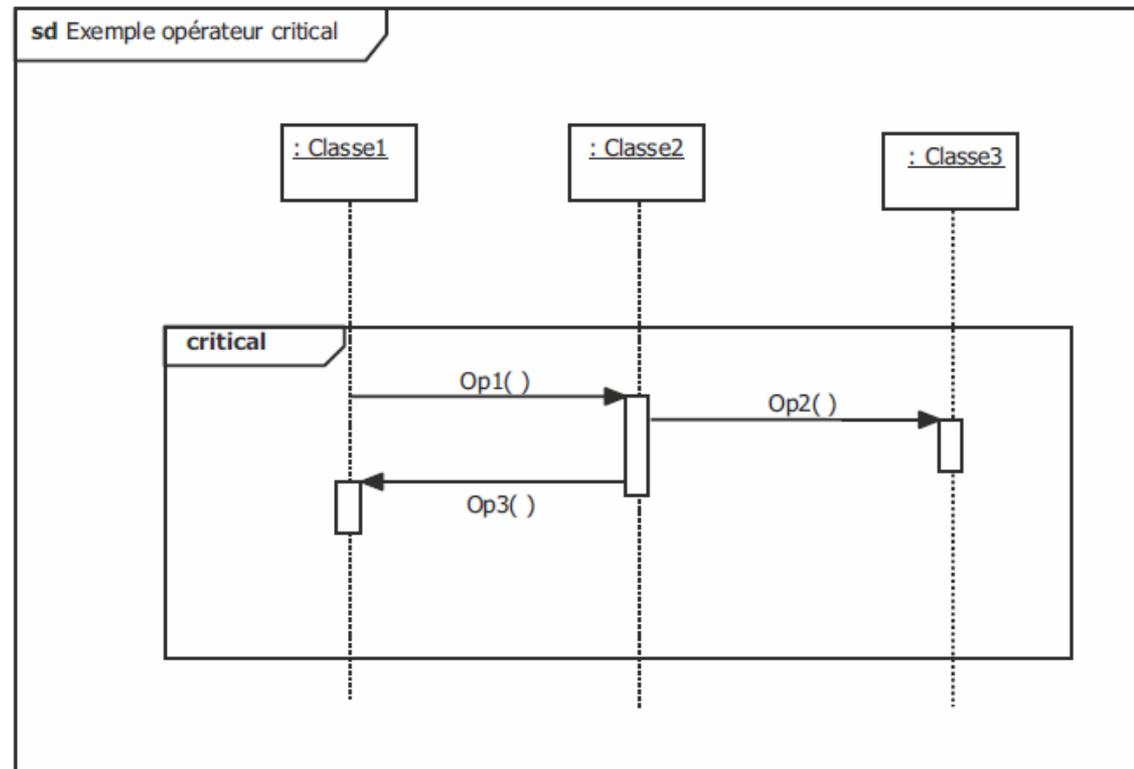


Figure – Exemple de fragment d'interaction avec l'opérateur critical

DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateurs negative:*

- L'opérateur **neg** (**negative**) permet d'indiquer qu'une séquence d'interactions est invalide.

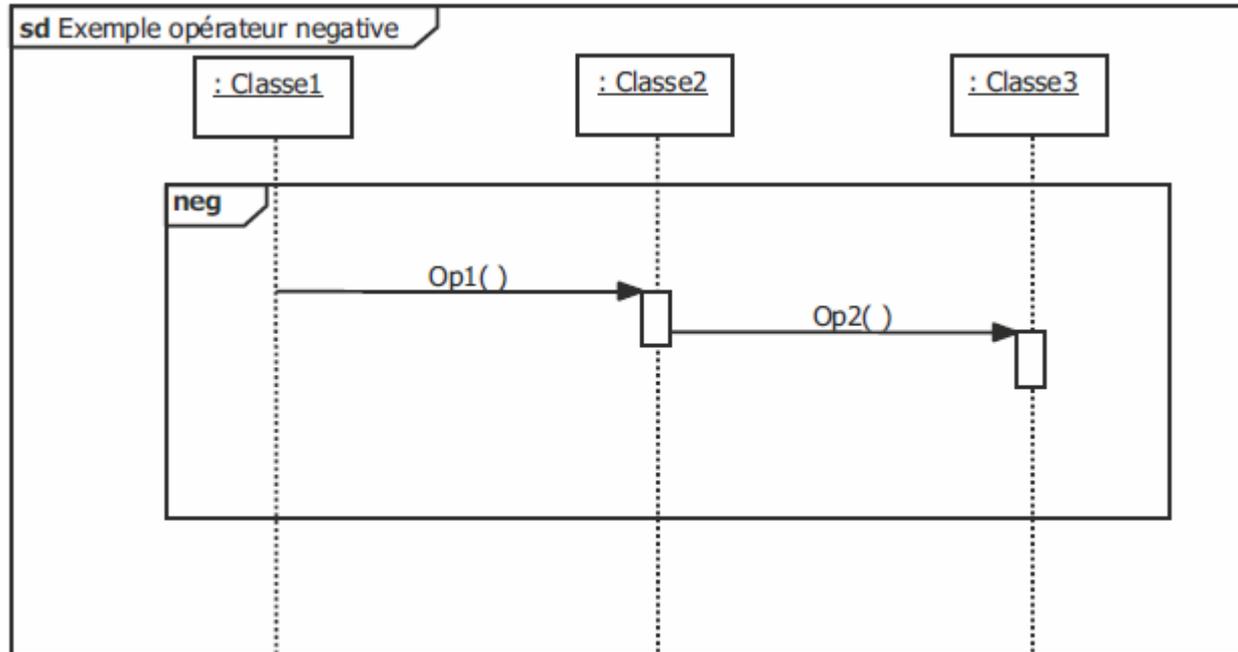


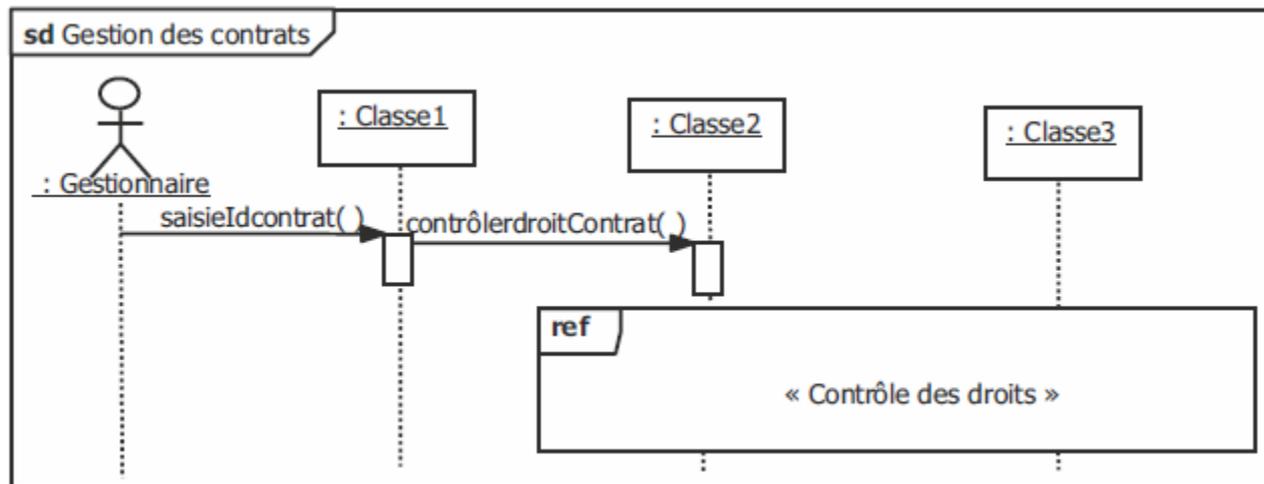
Figure – Exemple de fragment d'interaction avec l'opérateur neg

- les opérations `Op1()` et `Op2()` du fragment `neg` sont invalides.
- Une erreur sera déclenchée dans ce cas à l'exécution du fragment.

DSE : FRAGMENT D'INTERACTION COMBINÉ

- *Opérateur Ref:*

- L'opérateur ref permet d'appeler une séquence d'interactions décrite par ailleurs constituant ainsi une sorte de sous-diagramme de séquence.
- Dans cet exemple, on fait appel à un fragment « Contrôle des droits » qui est décrit par ailleurs.



Exemple de fragment d'interaction avec l'opérateur ref

AUTRE UTILISATION DU DIAGRAMME DE SÉQUENCE

- Le diagramme de séquence peut être aussi utilisé pour documenter un cas d'utilisation.
- Les interactions entre objets représentent, dans ce cas, des flux d'informations échangés et non pas de véritables messages entre les opérations des objets.

• Exemple :

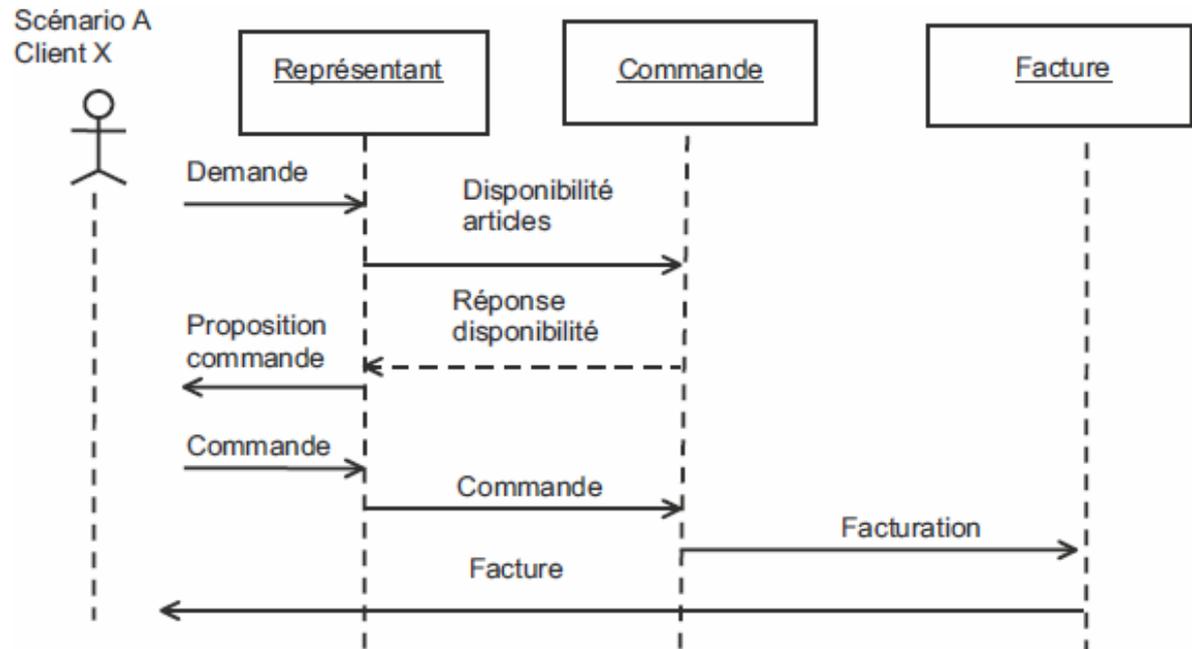
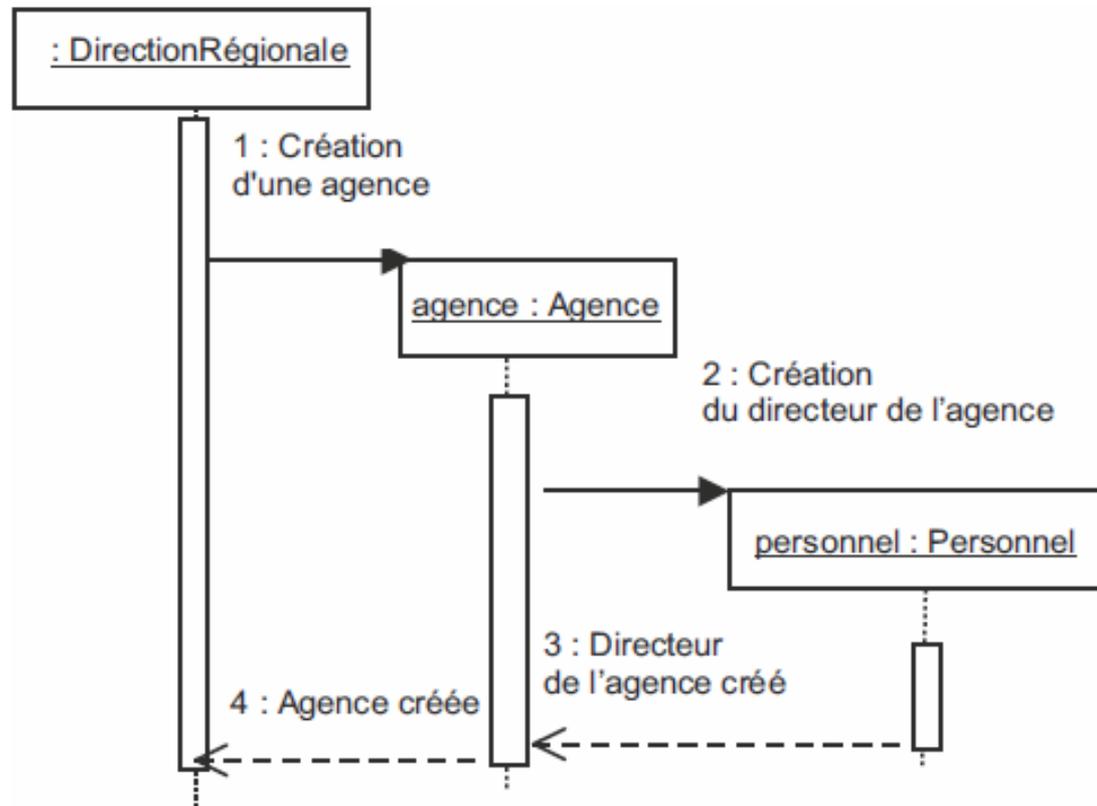


Figure : – Exemple de diagramme de séquence associé à un cas d'utilisation

DIAGRAMME DE SÉQUENCE (DSE)

- Exercice:

En se référant au sujet de 'Agence régionale' du diagramme de classe, nous donnons à titre d'exemple, le diagramme de séquence. Ce scénario correspond à la création d'une agence intégrant la création d'une personne.



DIAGRAMMES COMPORTEMENTAUX

- Les diagrammes comportementaux sont focalisés sur la description de la partie dynamique du système à modéliser.
- 7 diagrammes sont proposés par UML 2 :
 - diagramme des cas d'utilisation (DCU),
 - diagramme d'état-transition (machine d'état, DET),
 - diagramme d'activité (DAC),
 - Diagrammes d'interaction:
 - diagramme de séquence (DSE),
 - **diagramme de communication (DCO),**
 - diagramme global d'interaction (DGI),
 - diagramme de temps (DTP).

DIAGRAMME DE COMMUNICATION (DCO)

- Un des diagrammes d'**interactions**.
- Le diagramme de communication met plus l'accent sur l'aspect spatial des échanges que l'aspect temporel.
- Le DCO est souvent utilisé pour illustrer un cas d'utilisation ou pour décrire une opération.
- **Rôle:**
 - Chaque participant à un échange de message correspondant à une ligne de vie dans le diagramme de séquence se représente sous forme d'un **rôle dans le diagramme de** communication.
 - Un rôle est identifié par :
 - `<nom de rôle> : <nom du type>`

Une des deux parties de cette identification est obligatoire ainsi que le séparateur « : ».

Le **nom du rôle correspond au nom de l'objet dans le cas où l'acteur ou la classe** ont un rôle unique par rapport au système.

Le **nom du type correspond au nom de la classe** lorsque l'on manipule des objets.
- **Exemple**
 - administrateur : utilisateur

Pour un utilisateur qui est vu au travers de son rôle d'administrateur.

DIAGRAMME DE COMMUNICATION (DCO)

- *Message:*

- Un **message** correspond à un **appel d'opération** effectué par un rôle **émetteur** vers un rôle récepteur.
- Le sens du message est donné par une flèche portée au-dessus du lien reliant les participants au message (origine et destinataire).
- Chaque message est identifié par <numéro> : nom ()
- Plus précisément l'identification d'un message doit respecter la syntaxe suivante :

[n° du msg préc. reçu] «.» n° du msg [clause d'itération] [condition] «:» nom du msg.

- Numéro du message précédent reçu : permet d'indiquer la chronologie des messages.
- Numéro du message : numéro hiérarchique du message de type 1.1, 1.2... avec utilisation de lettre pour indiquer la simultanéité d'envoi de message.
- Clause d'itération : indique si l'envoi du message est répété. La syntaxe est * [spécification de l'itération].
- Condition : indique si l'envoi du message est soumis à une condition à satisfaire.

DIAGRAMME DE COMMUNICATION (DCO)

- *Message*:
 - **Exemples**:

1.2.1 * [3 fois] pour un message à adresser trois fois de suite.

1.2a et 1.2b pour deux messages envoyés en même temps.

Exemple récapitulatif de désignation de message :

1.2a.1.1[si t > 100] : lancer()

Ce message signifie :

- 1.2a : numéro du message reçu avant l'envoi du message courant.
- 1.1 : numéro de message courant à envoyer.
- [si t > 100] : message à envoyer si t > 100.
- lancer() : nom du message à envoyer.

DIAGRAMME DE COMMUNICATION (DCO)

- *Formalisme:*
 - Les rôles correspondent à des objets.
 - Le lien entre les rôles est représenté par un trait matérialisant le support des messages échangés.
 - La figure 3.58 donne le formalisme de base du diagramme de communication.



Figure – Formalisme de base du diagramme de communication

DIAGRAMME DE COMMUNICATION (DCO)

- *Exemple:*

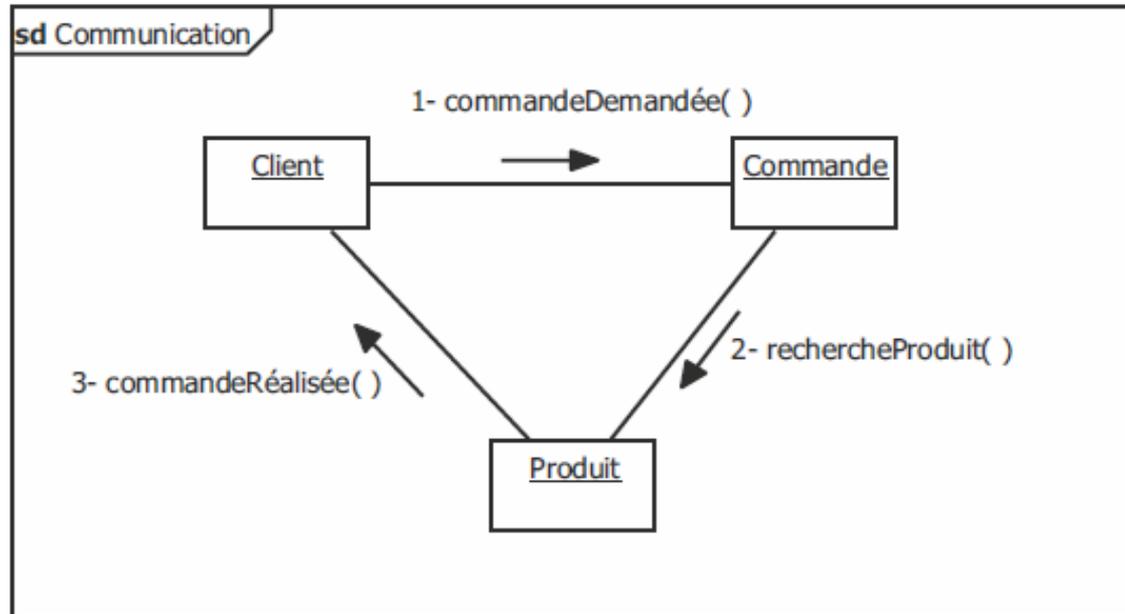


Figure — Exemple de diagramme de communication

DIAGRAMME DE COMMUNICATION (DCO)

- *Exemple exercice Agence:*

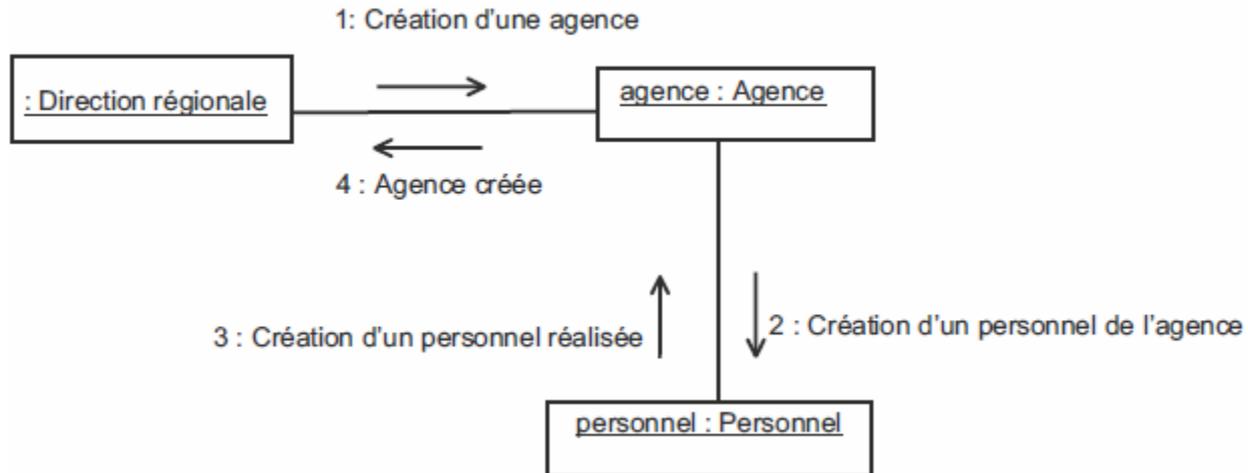


Figure – Exemple d'ensemble du diagramme de communication

DIAGRAMME GLOBAL D'INTERACTION (DGI)

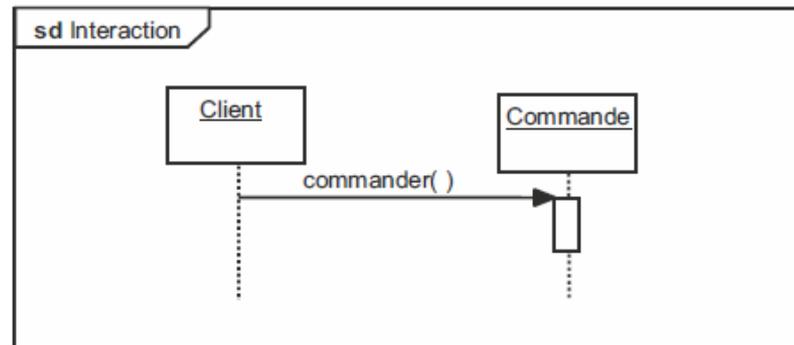
- Le diagramme global d'interaction permet de représenter une **vue générale** des interactions décrites dans le diagramme de séquence et des flots de contrôle décrits dans le diagramme d'activité.
- Le diagramme global d'interaction privilégie la vue générale des flux de contrôle dans lesquels les noeuds sont des interactions ou des utilisations d'interactions (opérateur ref).
- Autrement dit, le DGI est un diagramme d'activité dans lequel on représente des fragments d'interaction ou des utilisations d'interactions.
- Ainsi, il est possible de représenter :
 - des choix de fragments d'interactions (fusion) ;
 - des déroulements parallèles de fragments d'interactions (débranchement et jonction) ;
 - des boucles de fragments d'interaction.
- Les lignes de vie concernées par le DGI peuvent être citées dans l'en-tête du diagramme mais ne sont pas à représenter graphiquement.

DIAGRAMME GLOBAL D'INTERACTION (DGI)

- Concepts manipulés:

Le diagramme global d'interaction utilise les concepts du diagramme d'activité auquel on ajoute deux compléments :

- Les fragments d'interaction du diagramme de séquence: Il s'agit de la notion de fragment d'interaction vue dans le diagramme de séquence mais qui ne doit pas être détaillé à ce niveau.



- Les utilisations de fragments d'interaction: Il est aussi possible de faire appel à des fragments d'interaction à l'aide de l'opérateur ref.

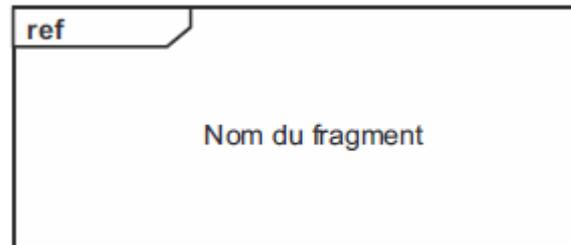


DIAGRAMME GLOBAL D'INTERACTION (DGI)

- Exemple :

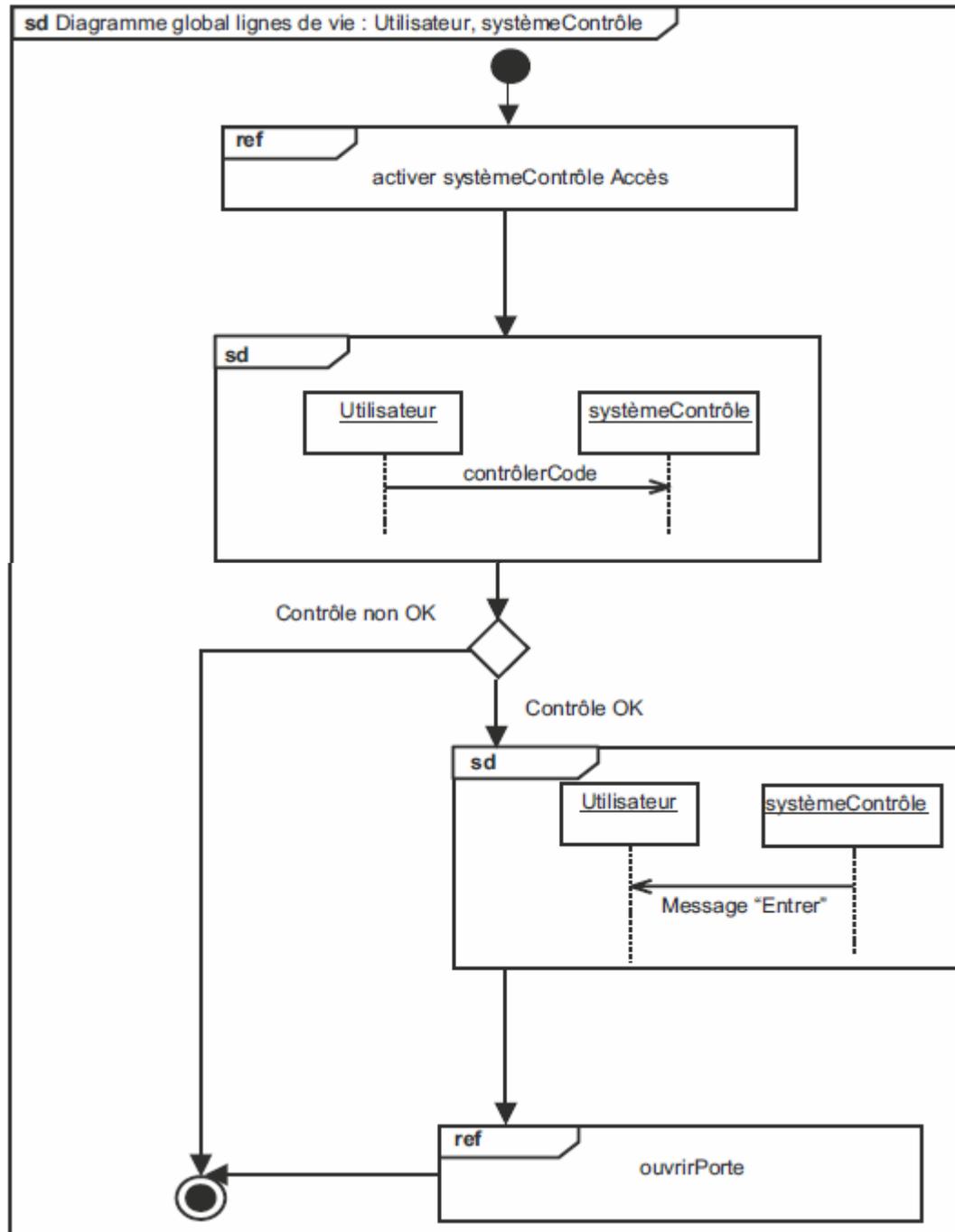


DIAGRAMME DE TEMPS (DTP)

- Le **diagramme de temps permet de représenter les états et les interactions d'objets dans** un contexte où le temps a une forte influence sur le comportement du système à gérer.
- Autrement dit, le diagramme de temps permet de mieux représenter des changements d'états et des interactions entre objets liés à des contraintes de temps.
- Pour cela, le diagramme de temps utilise en plus des lignes de vie, les concepts suivants :
 - Des états ou des lignes de temps conditionnées avec deux représentations graphiques possibles.
 - Des représentations propres aux aspects temporels :
 - échelle de temps,
 - contrainte de durée,
 - Événements ...

DIAGRAMME DE TEMPS (DTP)

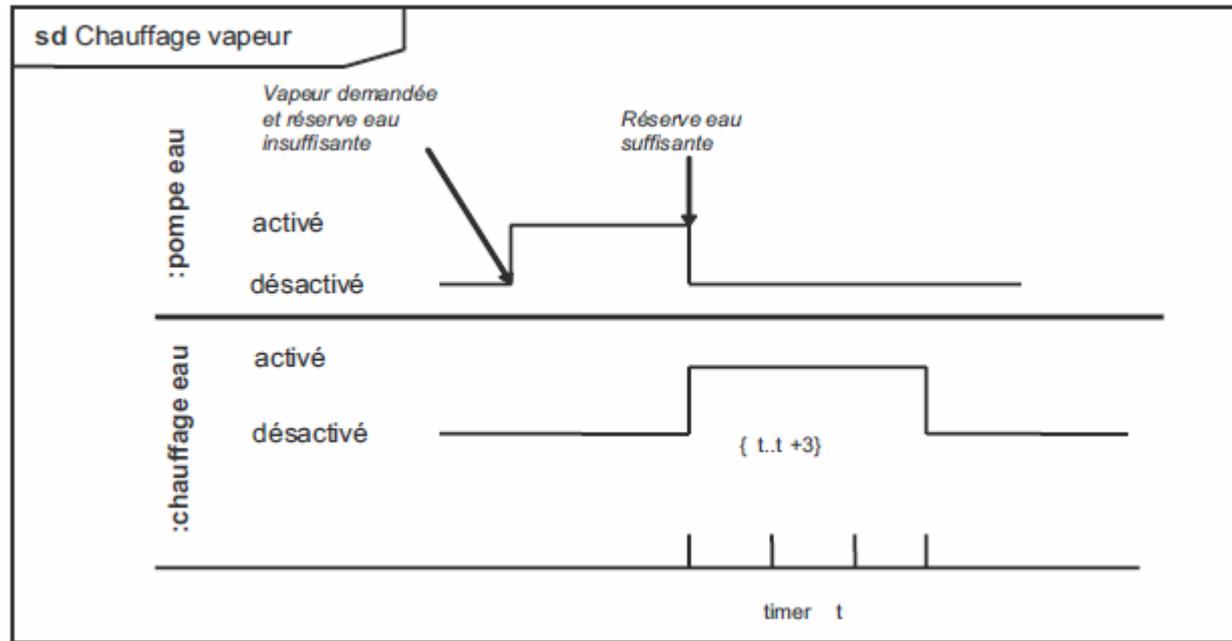
- 3 Concepts de bases sont manipulés:
 - **Ligne de vie** - Elle représente l'objet que l'on veut décrire. Elle se dessine de manière horizontale. Plusieurs lignes de vie peuvent figurer dans un diagramme de temps.
 - **État ou ligne de temps conditionnée** - Les différents états que peut prendre l'objet d'étude sont listés en colonne permettant ainsi de suivre le comportement de l'objet ligne par ligne (une ligne pour un état).
 - **États linéaires** - Il s'agit du même concept que le précédent, mais la représentation de la succession des états est faite de manière linéaire à l'aide d'un graphisme particulier.

DIAGRAMME DE TEMPS (DTP)

- **Représentation et exemples :**
- Soit à représenter le dispositif de chauffe d'un fer à repasser à vapeur au moment de sa mise en service selon les règles suivantes :
 - la pompe à eau qui remplit la chambre de chauffe s'active dès que le témoin d'eau interne le demande ;
 - la pompe à eau se désactive dès que le niveau d'eau nécessaire est atteint ;
 - le chauffage de l'eau, permettant de produire la vapeur, se met en action à la première mise en service du fer à repasser dès que le niveau d'eau de la chambre de chauffe est suffisant ;
 - le chauffage initial de l'eau dure 3 mm permettant ainsi de produire la vapeur.
- Dans cet exemple, nous avons deux objets à étudier : pompe à eau et chauffage de l'eau.
- Nous allons considérer pour chacun d'entre eux deux états significatifs : activé et désactivé.

DIAGRAMME DE TEMPS (DTP)

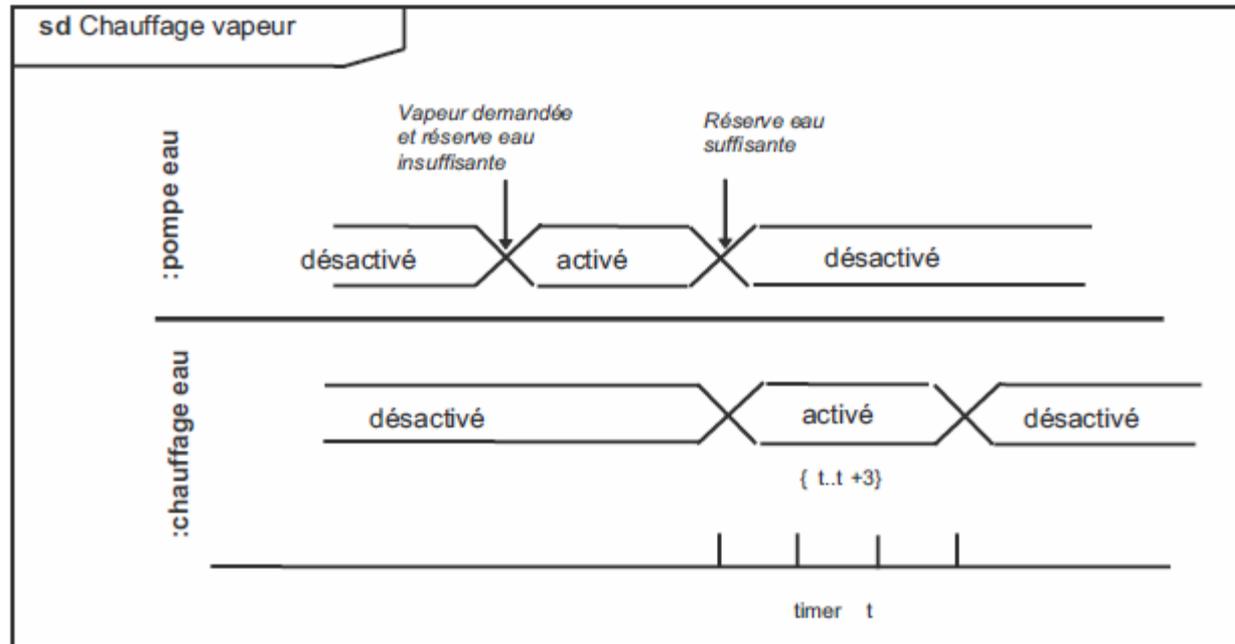
- Représentation et exemples "chauffage vapeur"



Exemple de diagramme de temps avec représentation « en escalier »

DIAGRAMME DE TEMPS (DTP)

- Représentation et exemples "chauffage vapeur"



Exemple de diagramme de temps avec représentation linéaire