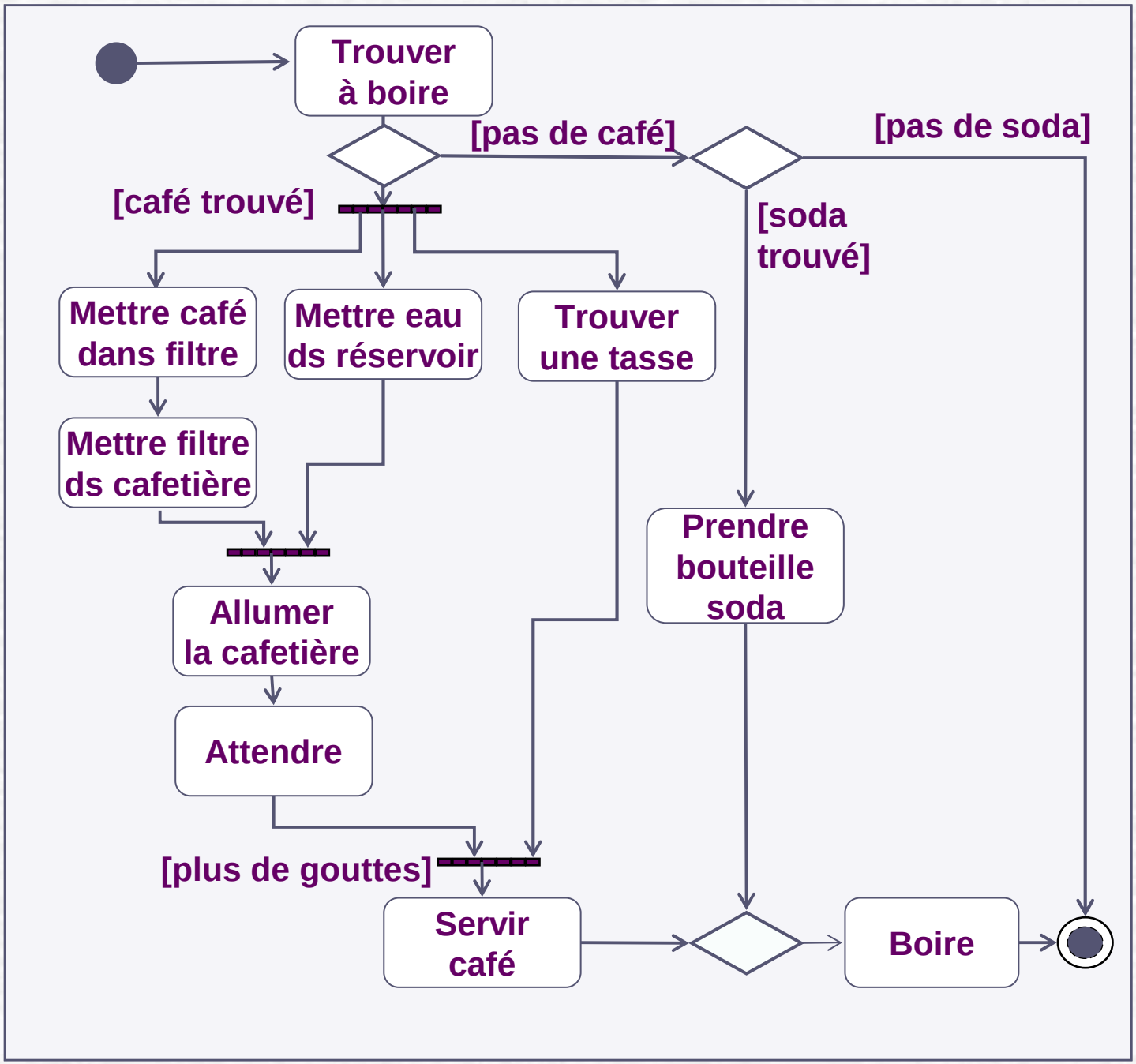
Isima

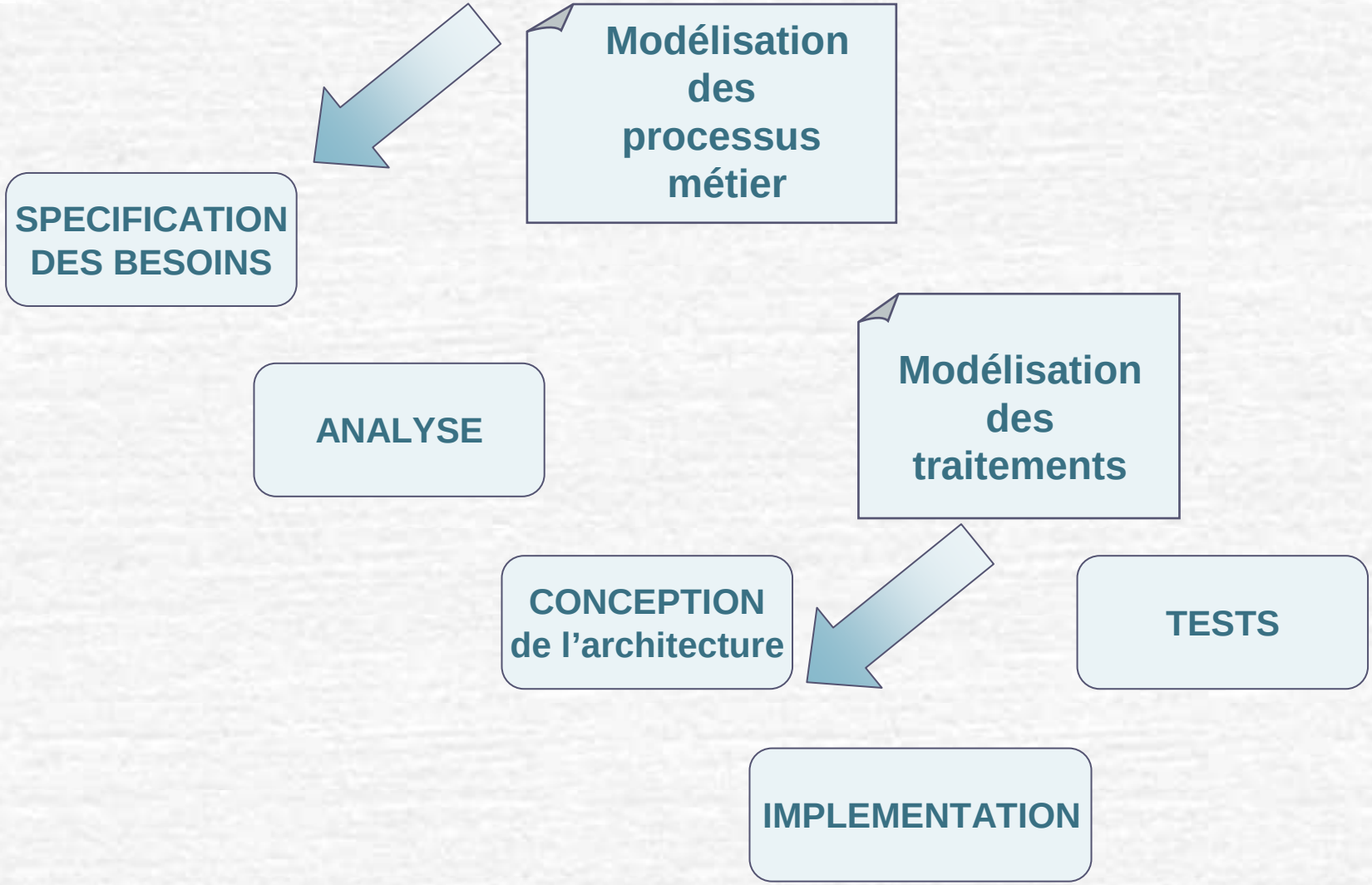
Chapitre 8

LES DIAGRAMMES D'ACTIVITÉS

EXEMPLE


Trouver à boire
[pas de café]
[pas de soda]
[café trouvé]
[soda trouvé]
Mettre café dans filtre
Mettre eau ds réservoir
Trouver une tasse
Mettre filtre ds cafetière
Prendre bouteille soda
Allumer la cafetière
Attendre
[plus de gouttes]
Servir café
Boire

DIAGRAMMES D'ACTIVITES

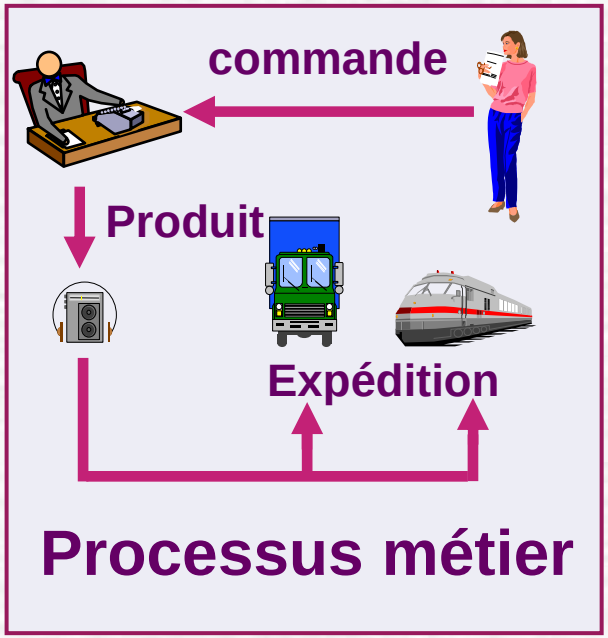


DIAGRAMMES D'ACTIVITES

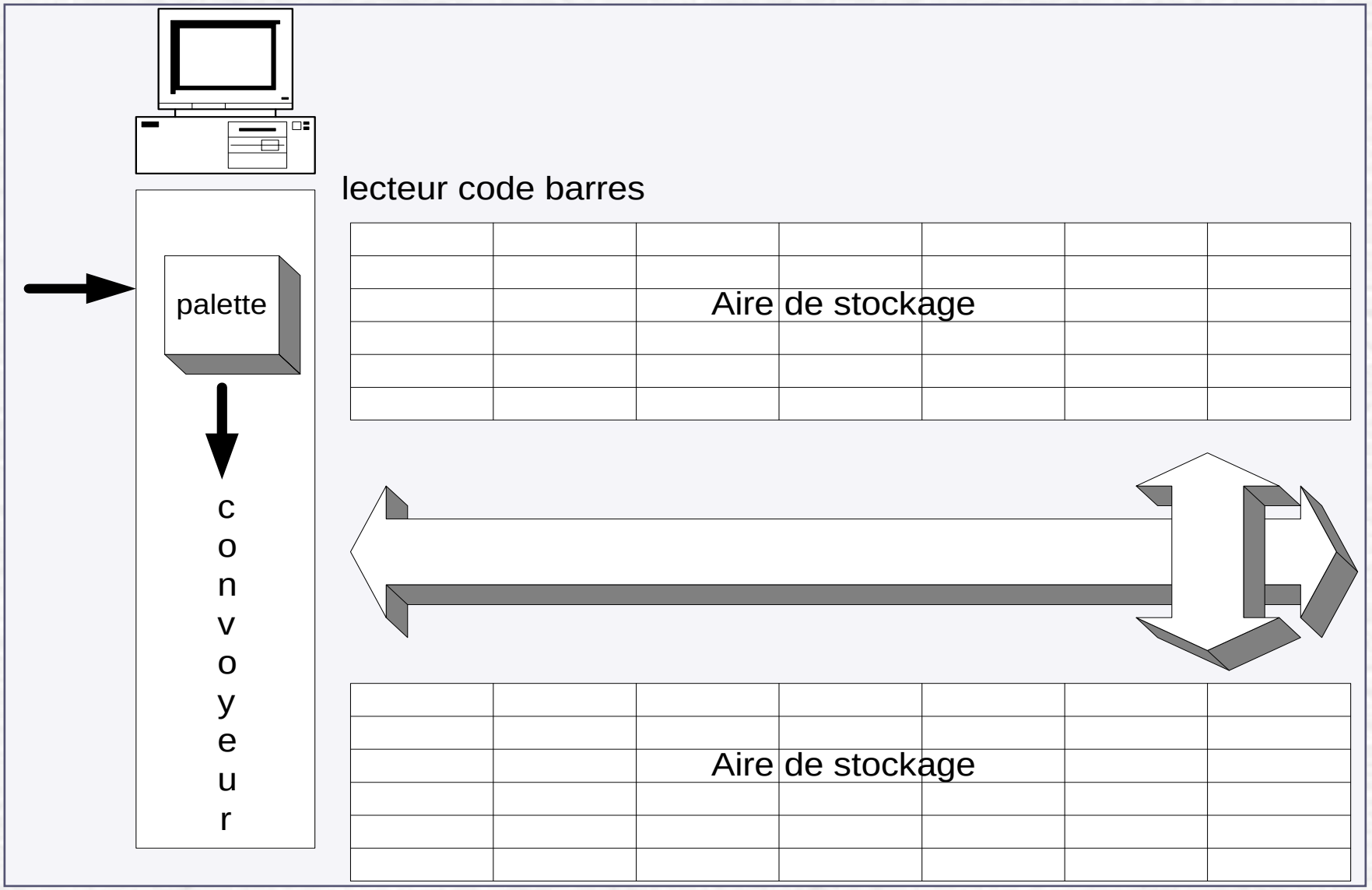
- S'intéressent aux étapes d'une tâche complexe à accomplir.
- Modélisent les processus : enchaînements d'activités.
- Décrivent le flot de contrôle entre activités.
- S'utilisent à plusieurs niveaux :
  - Processus métiers de haut niveau (par exemple étude de l'existant).
  - Alternative aux diagrammes de séquences pour décrire un cas d'utilisation.
  - Description des délégations entre objets dans un système logiciel.
  - Description d'algorithmes (difficultés pour décrire des algorithmes complexes)

DOMAINE D'APPLICATION

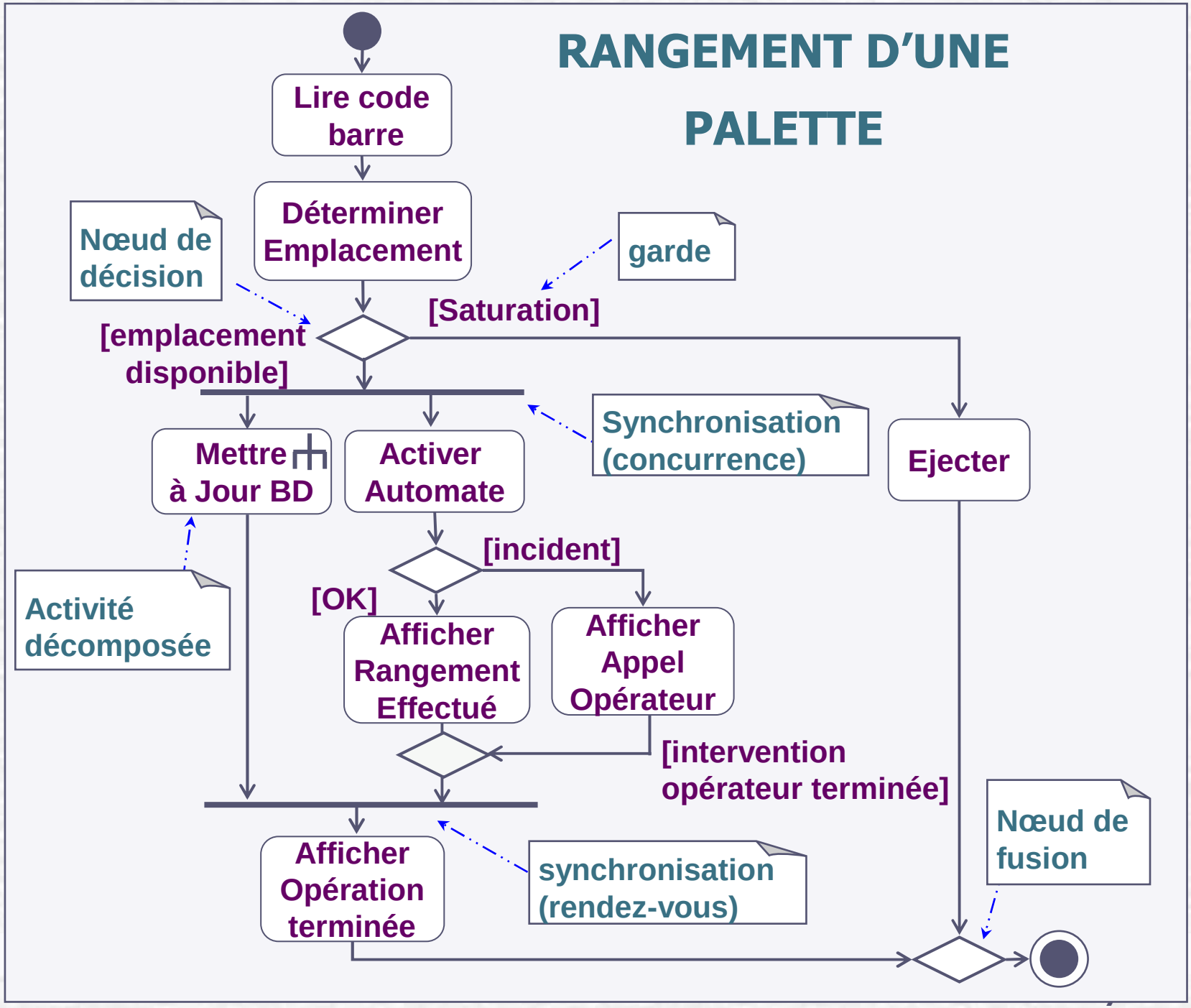
- Processus métier :
  - Le système (d'information, industriel),
  - Un sous-système,
- Cas d'utilisation
  - Description d'une famille de scénarios,
  - Enchaînements d'écrans dans une IHO,
- Logique complexe
  - d'une opération, ou d'une classe,
  - d'une collaboration (la dynamique d'un ensemble d'objets).



EXEMPLE : PROCESSUS INDUSTRIEL



Isima


RANGEMENT D'UNE PALETTE
Lire code barre
Déterminer Emplacement
Nœud de décision
garde
[Saturation]
[emplacement disponible]
Synchronisation (concurrence)
Mettre à Jour BD
Activer Automate
Ejecter
[incident]
[OK]
Activité décomposée
Afficher Rangement Effectué
Afficher Appel Opérateur
[intervention opérateur terminée]
Afficher Opération terminée
synchronisation (rendez-vous)
Nœud de fusion

Isima

DIAGRAMMES D'ACTIVITÉS

- Syntaxe proche des diagrammes d'états
- Contiennent :
  - Des activités (peuvent se redécomposer en un autre diagramme d'activités)
  - Des actions (non décomposables) et des attentes,
  - Des transitions automatiques : le flot de contrôle reste dans l'activité jusqu'à ce que le traitement soit terminé.
  - Des nœuds de décision (associés à des gardes) et de fusion,
  - Des barres de synchronisation
  - Des objets (pour monter qui est modifié par une action).

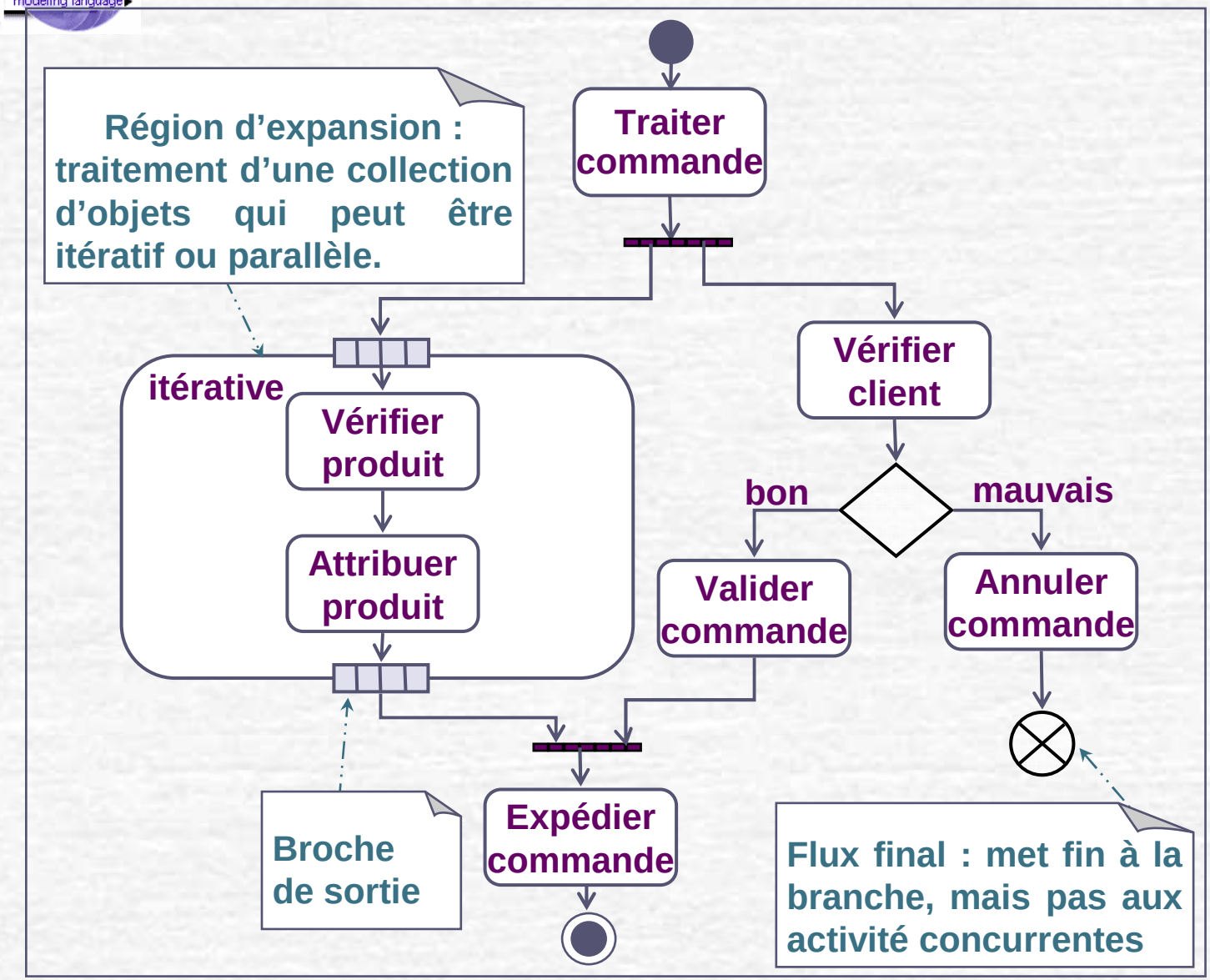
DIAGRAMMES D'ACTIVITÉS

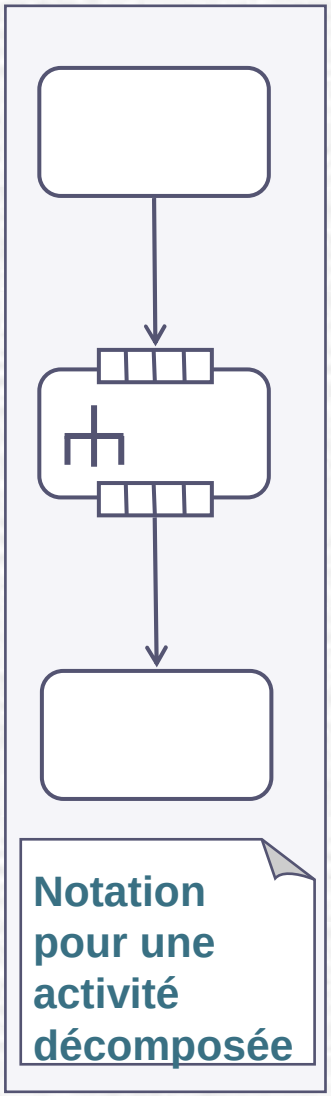
- Un nœud de décision est représenté par un losange d'où partent toutes les alternatives obligatoirement exclusives.
- Un autre losange (nœud de « fusion ») matérialise la fusion des branches d'une décision.
- Une barre de synchronisation spécifie le parallélisme : en sortie d'une barre on indique plusieurs flots de contrôles concurrents.
  - à l'exécution du système, ces activités peuvent être physiquement concurrentes, séquentielles ou entrelacées.
- Une autre barre de synchronisation représente le rendez-vous entre flots de contrôles parallèles : la barre n'est franchie que lorsque chaque flot amont est terminé.
- Etat initial et final peuvent être représentés sur le diagramme.

DIAGRAMMES D'ACTIVITÉS

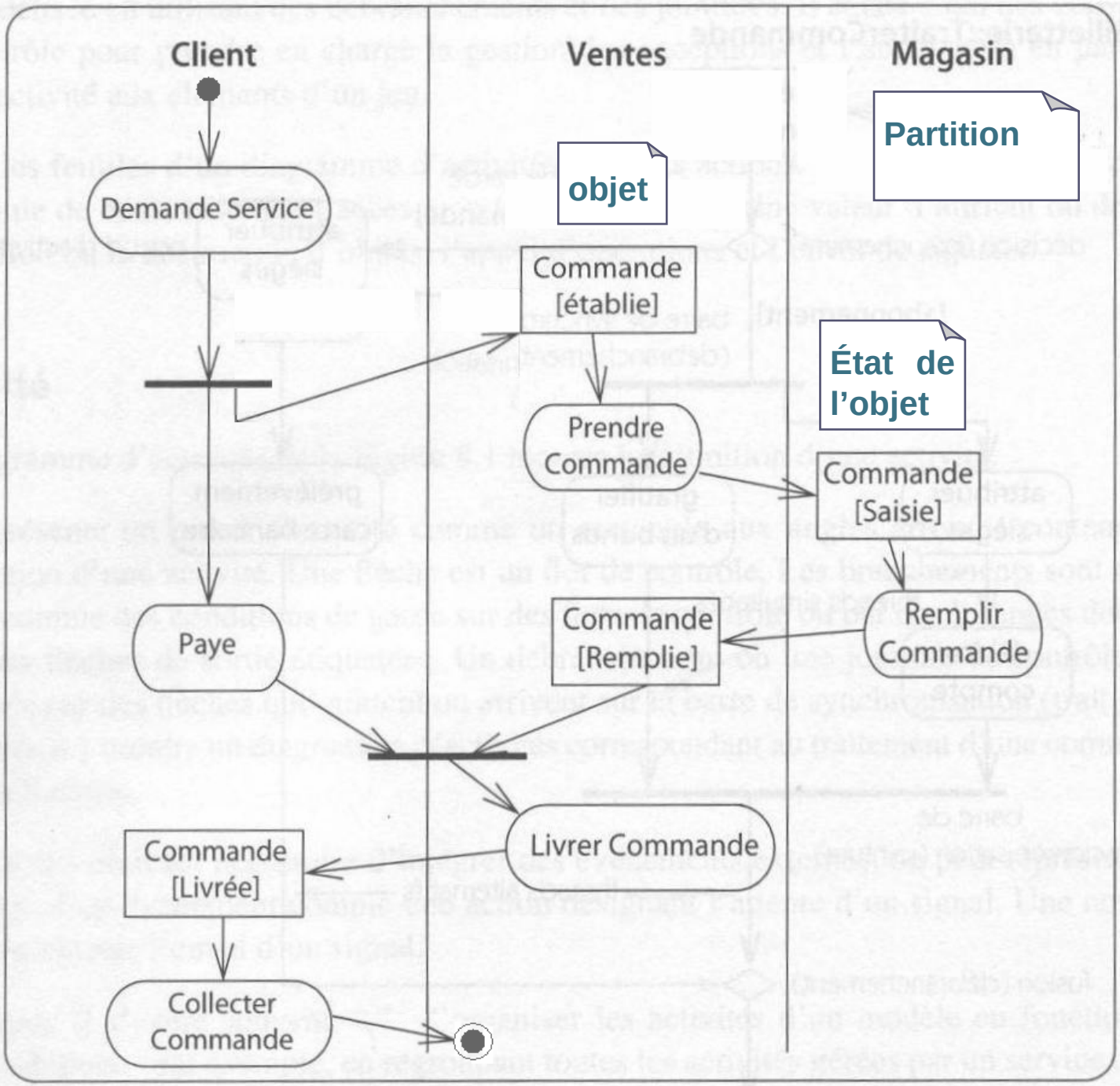
- Les itérations sont représentées par une « région d'expansion » : partie d'un diagramme où les actions sont exécutées pour tous les éléments d'une collection.
  - Les actions peuvent être concurrentes ou itératives.
- Le symbole ⊗ représente un flux qui s'arrête même si l'activité n'est pas finie (exemple : dans une collection on rejette un élément).
- Les diagrammes peuvent être découpés en partitions (swim lanes), pour montrer les différentes responsabilités dans un mécanisme ou une organisation.
  - Chaque responsabilité assurée par un objet est allouée à une partition donnée.
  - La position relative des partitions n'est pas significative, les transitions peuvent traverser librement les frontières.
  - Les partitions peuvent être représentées en couloirs ou en damiers.

REGION D'EXPANSION

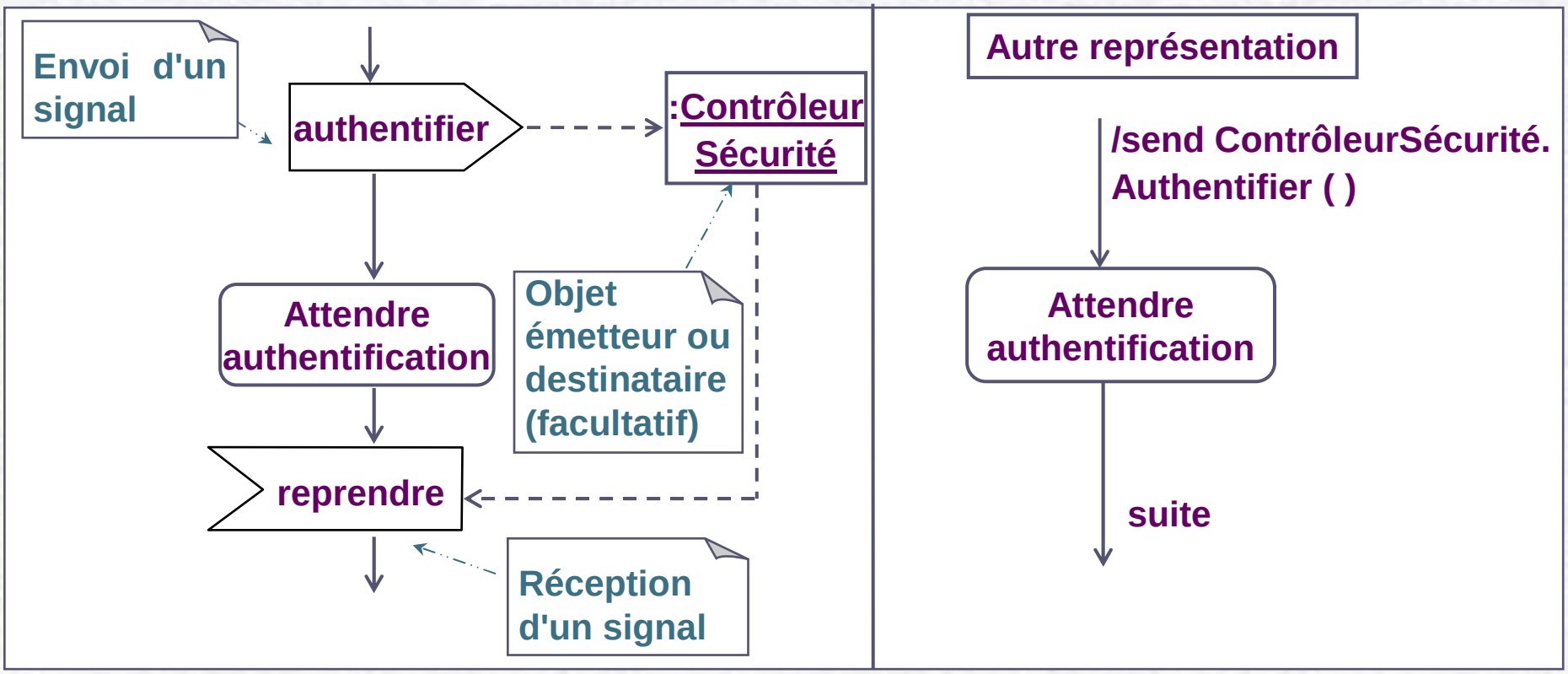


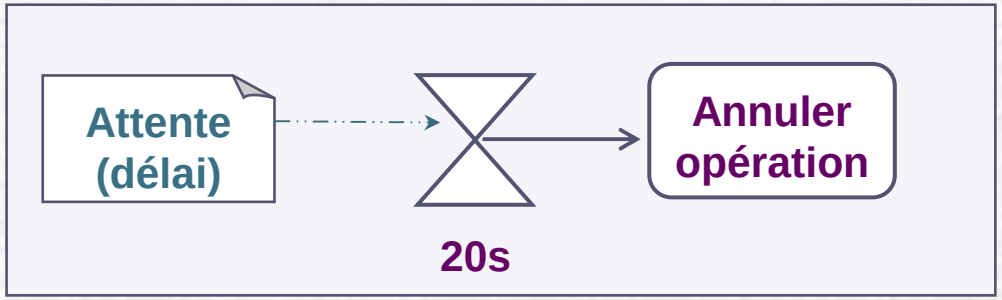


PARTITIONS


Client
Ventes
Magasin
Partition
objet
Demande Service
Commande [établie]
État de l'objet
Prendre Commande
Commande [Saisie]
Paye
Commande [Remplie]
Remplir Commande
Livrer Commande
Commande [Livrée]
Collecter Commande

ENVOI ET RÉCEPTION DE SIGNAUX





MODÉLISATION DES FLUX

- Établir l'objectif.
- Sélectionner les objets qui ont les responsabilités de haut niveau.
- Identifier les pré-conditions de l'état initial et les post-conditions de l'état final.
- Spécifier les actions et les placer sur le diagramme.
- Regrouper en Action Composite les actions complexes ou les ensembles d'actions qui apparaissent à plusieurs reprises et développer un diagramme d'activités pour chacun.
- Tracer les transitions. Commencer avec les flux séquentiels, puis les branchements et enfin les flux parallèles.
- Insérer les objets importants dans les diagrammes.
- Montrer les changements de valeur et d'état utiles à la compréhension de l'objectif.

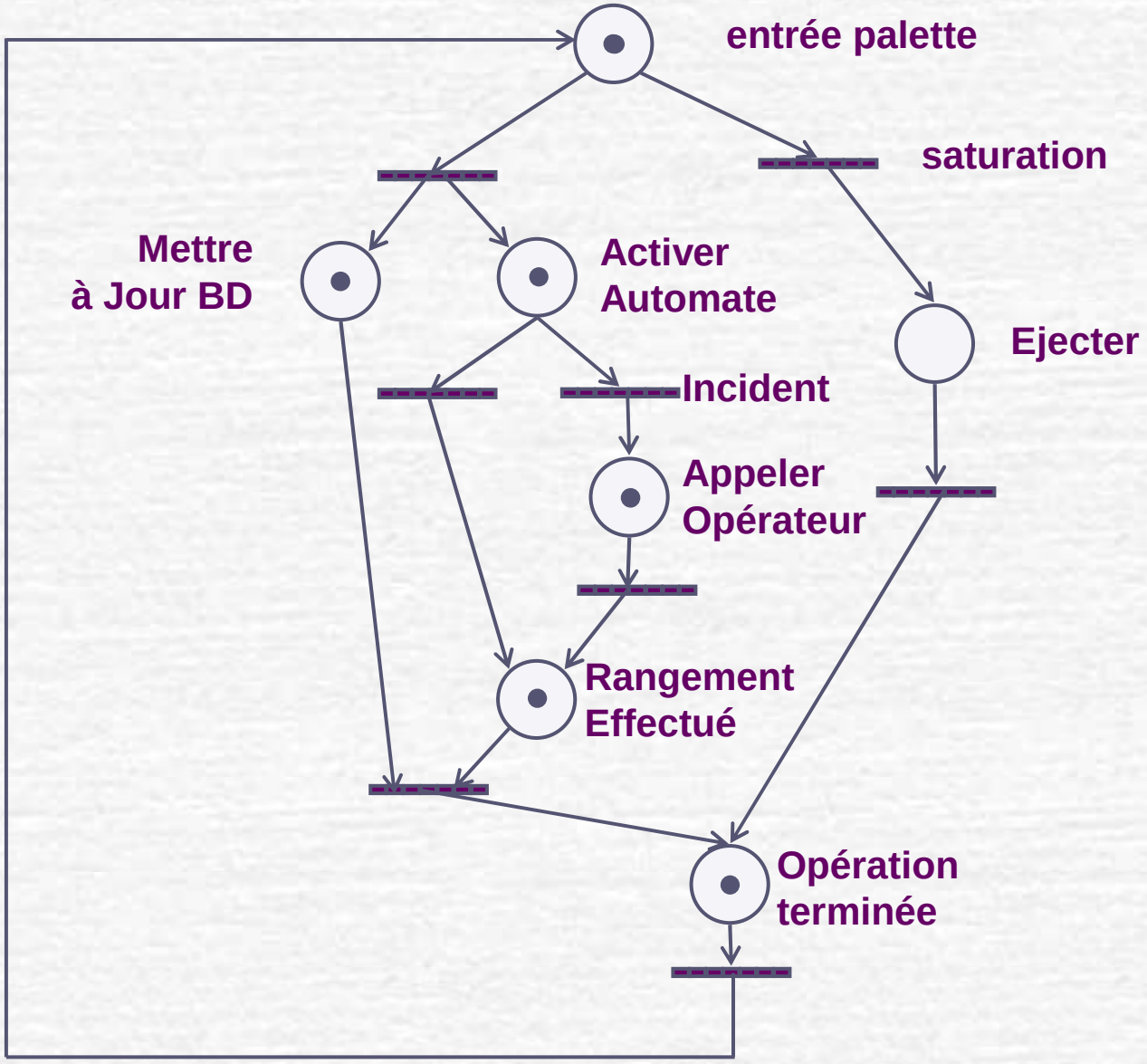
EXERCICE 1 : organisation d'un examen

- Le service scolarité :
  - planifie l'examen,
  - prépare les copies,
  - puis, une fois l'examen corrigé, saisit et affiche les notes,
  - et archive les copies.
- L'enseignant prépare un sujet et corrige les copies.
- Les étudiants rédigent une solution et prennent connaissance de leur note après affichage.
- Rédiger un diagramme d'activités (en remettant les choses dans l'ordre).

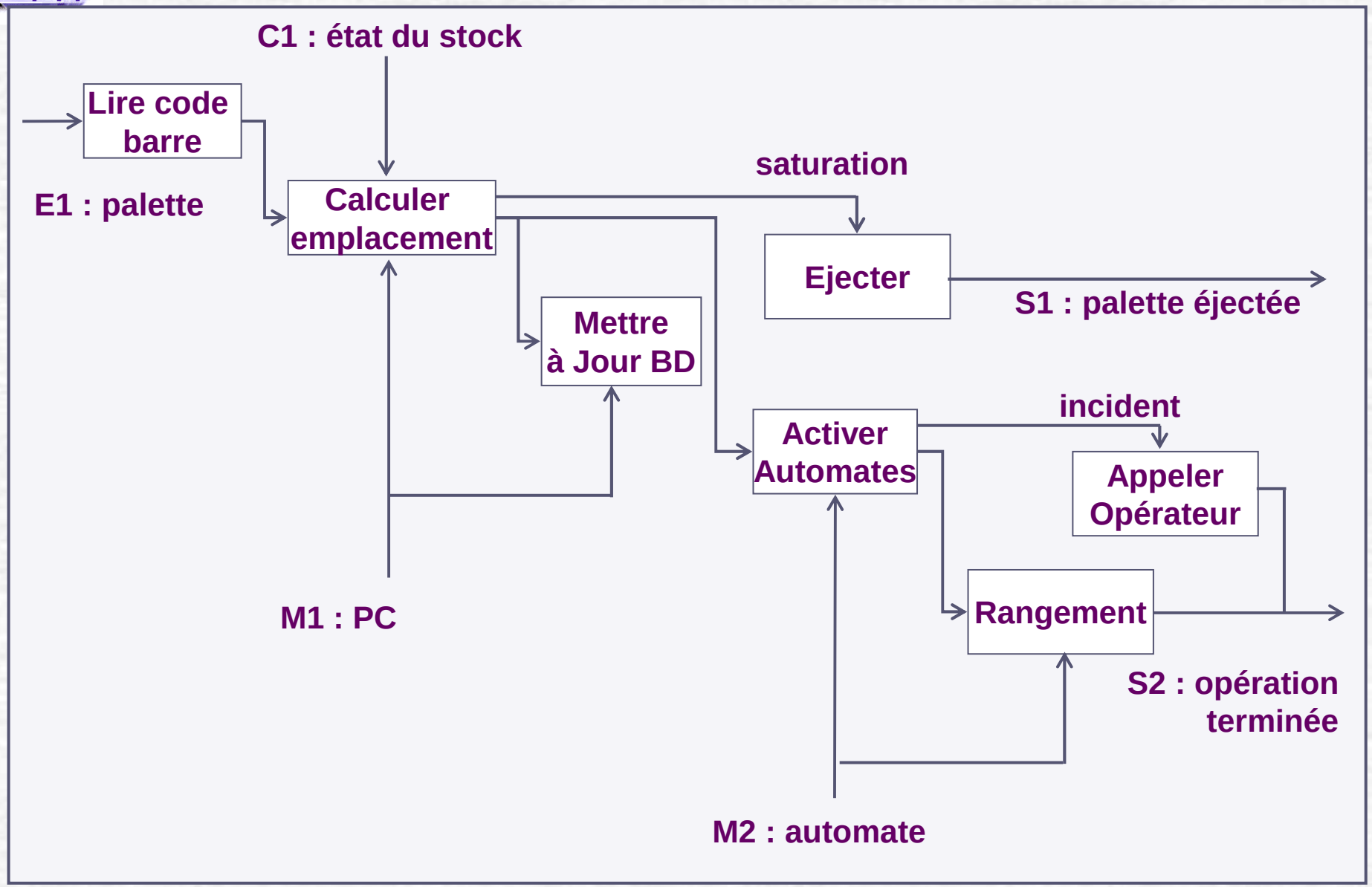
Exercice 2 : Atelier

- Pour créer une fiche de réparation, le chef d'atelier saisit les critères de recherche de voitures dans le système.
- Le logiciel de gestion des réparations lui donne la liste des voitures correspondant aux critères entrés.
- Si la voiture existe dans la liste, le chef d'atelier va sélectionner la voiture.
- Le logiciel va, ensuite, fournir les informations sur le véhicule.
- Si la voiture est sous garantie, le chef saisit la date de demande de réparation.
- Si la voiture n'existe pas, le chef saisit les informations concernant ce nouveau véhicule.
- Dans tous les cas, le chef d'atelier entre la date de réception et de restitution.
- Si le dommage de la voiture est payé par l'assurance, le logiciel fournit une liste d'assurances au chef d'atelier.
- Ce dernier sélectionne l'assurance adéquate.
- Enfin, le logiciel enregistre la fiche de réparation.

AUTRE FORMALISME : RESEAU DE PETRI



AUTRE FORMALISME : SADT / IDEF0



Chapitre 8

LA CONCEPTION

L'ACTIVITÉ DE CONCEPTION

- La spécification et l'analyse des besoins ont permis de définir le système à construire (Quoi).
- L'activité de conception, s'intéresse à la façon de construire le système (Comment).
- Elle vise à construire une **solution** qui est conforme aux besoins du système

CONCEPTION

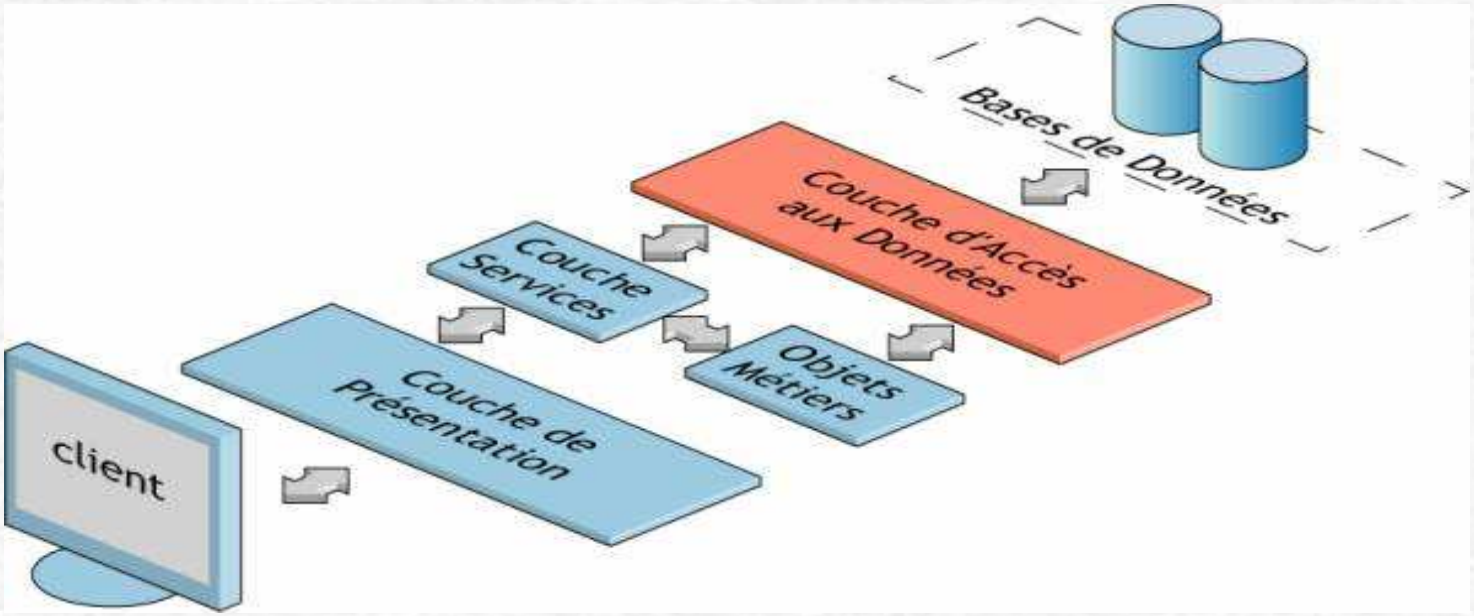
- Organiser le développement,
- Concevoir et documenter précisément la solution informatique (le code),
- Passer de l'analyse objets à l'architecture,
- Combiner les différents points de vue de la modélisation pour obtenir les détails de fabrication en langage objets,
- Répondre à toutes les questions qui concernent la manière de réaliser le système.

CONCEPTION

- Définition des classes à implémenter, en utilisant :
  - Les diagrammes de classes d'analyse pour préciser la structure des classes techniques :
    - Concevoir les associations,
    - Concevoir les attributs (sdd, conteneurs),
    - Définir les méthodes.
    - Intégrer patrons (design patterns) et composants sur étagère (COTS).
  - Les diagrammes d'interactions (communication entre objets),
  - Les diagrammes d'activités (délégations entre objets, méthodes).

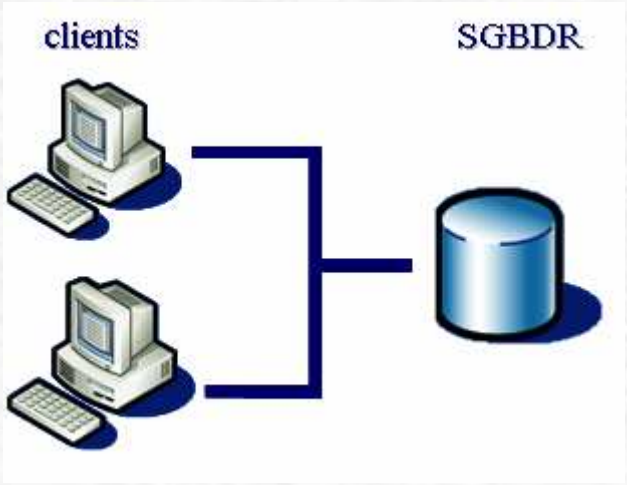
ARCHITECTURE

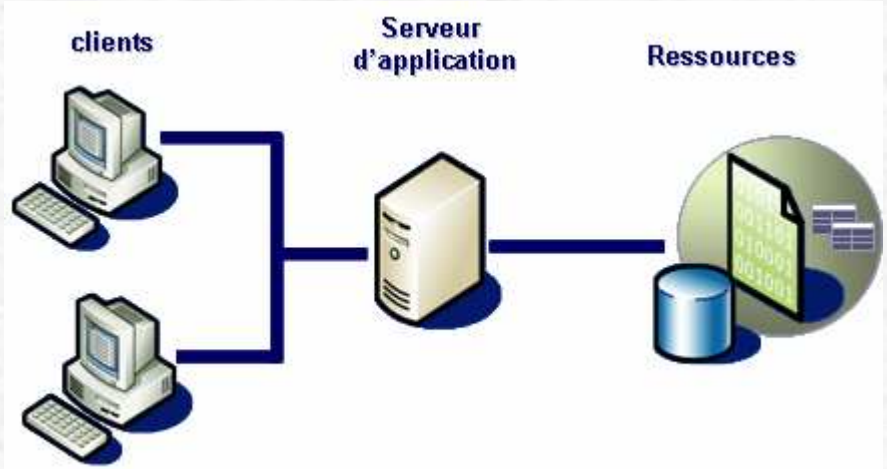
- Organisation des classes de conception en configuration logicielle : paquets sous-systèmes (cohésion, couplage).
- Séparation des responsabilités :
  - La présentation,
  - La logique applicative (services),
  - Le domaine métier,
  - L'accès aux données.



ARCHITECTURE

- Passage du modèle objets au modèle physique (solutions qui traduisent des choix techniques) :
  - Type d'architecture (clients/serveur, n tiers…),





  - Localisation des composants (nœuds du réseau),
  - Migration des objets.

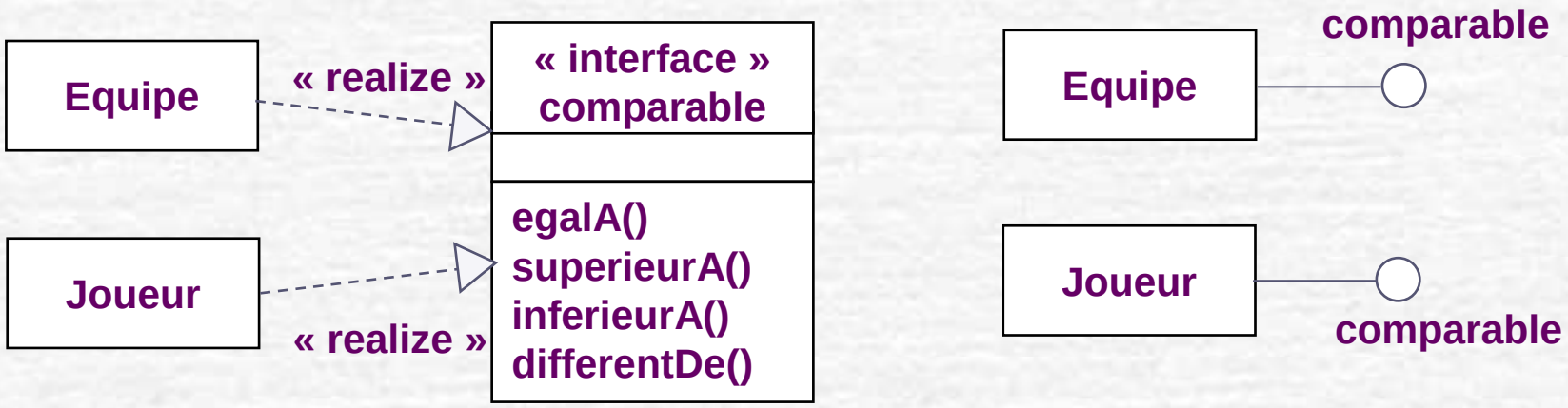
LA PERSISTANCE

- Type de gestion de données.
- Retouche du diagramme de classes :
  - Objets volumineux,
  - Chargement trop long,
  - Navigation,
  - Persistance trop étendue,
- Transformation de l'héritage (cohérence du modèle et navigabilité).
- Transformation des associations (requêtes, contraintes d'intégrité, économies de stockage).
- Passage au modèle relationnel (Cf. annexe 3 poly 1).

Chapitre 9

CLASSES NOTATIONS AVANCEES

INTERFACES


Equipe
« realize »
« interface »
comparable
Joueur
« realize »
egalA()
superieurA()
inferieurA()
differentDe()
Equipe
comparable
Joueur
comparable

NAVIGABILITE DES ASSOCIATIONS

notation

dirige ▶

Exemple

CLASSES GENERIQUES

Généricité : classe paramétrée

STEREOTYPES

«métaclasse»
TypeXX

«utilitaire»
Maths

«interface»
Runnable

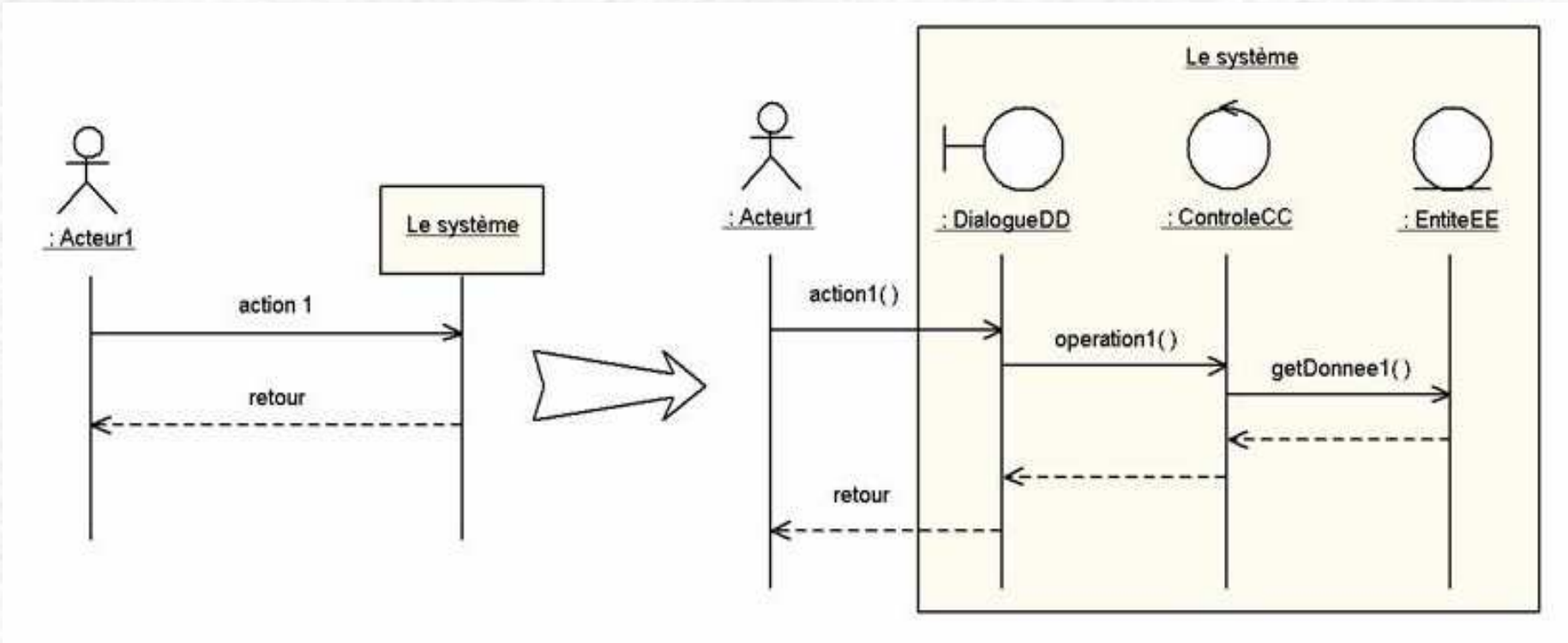
boundary
(classe d'IHO)

control

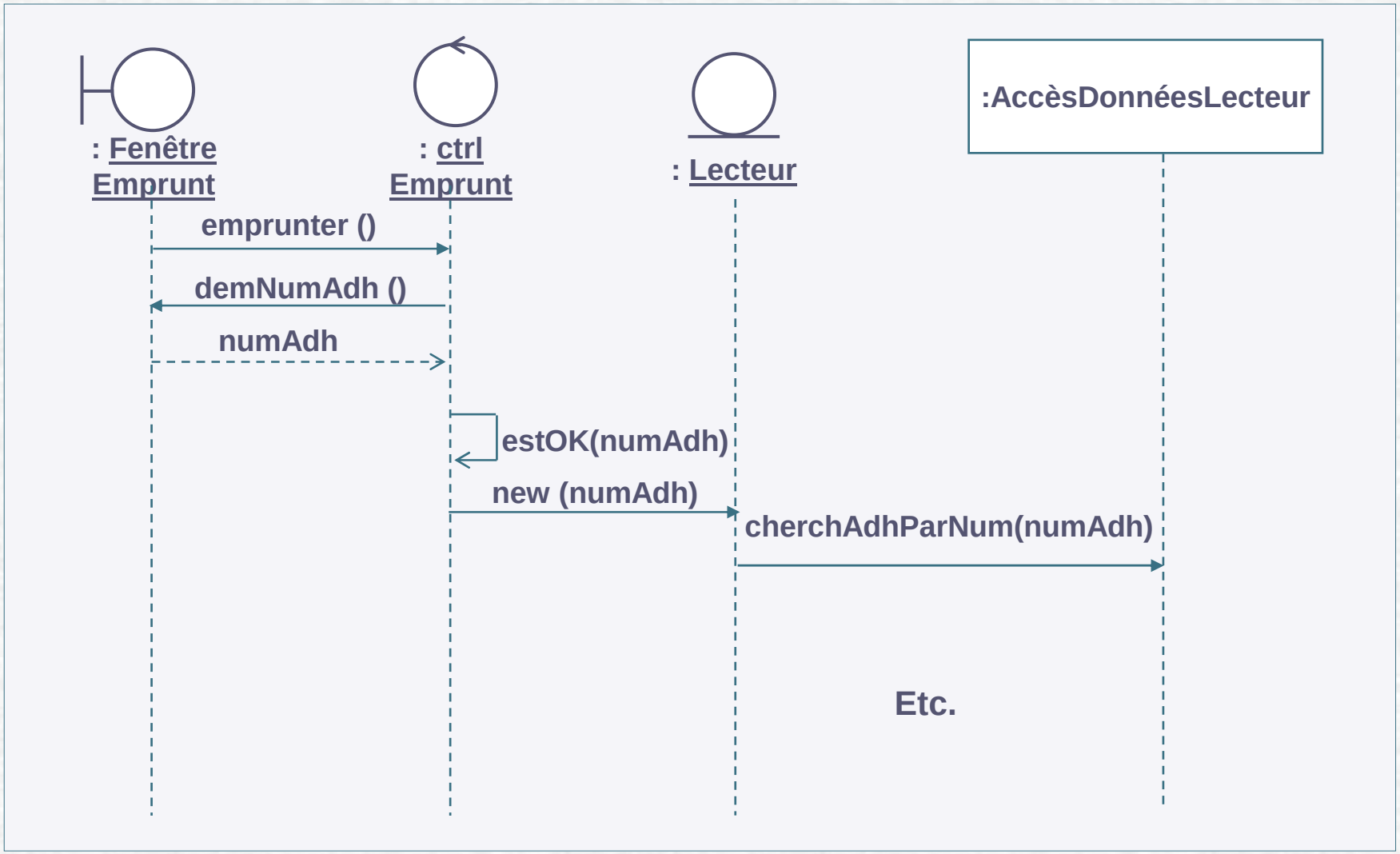
entity
(classe métier)

exemple : les classes d'analyse de Jacobson

TECHNIQUES DE CONCEPTION



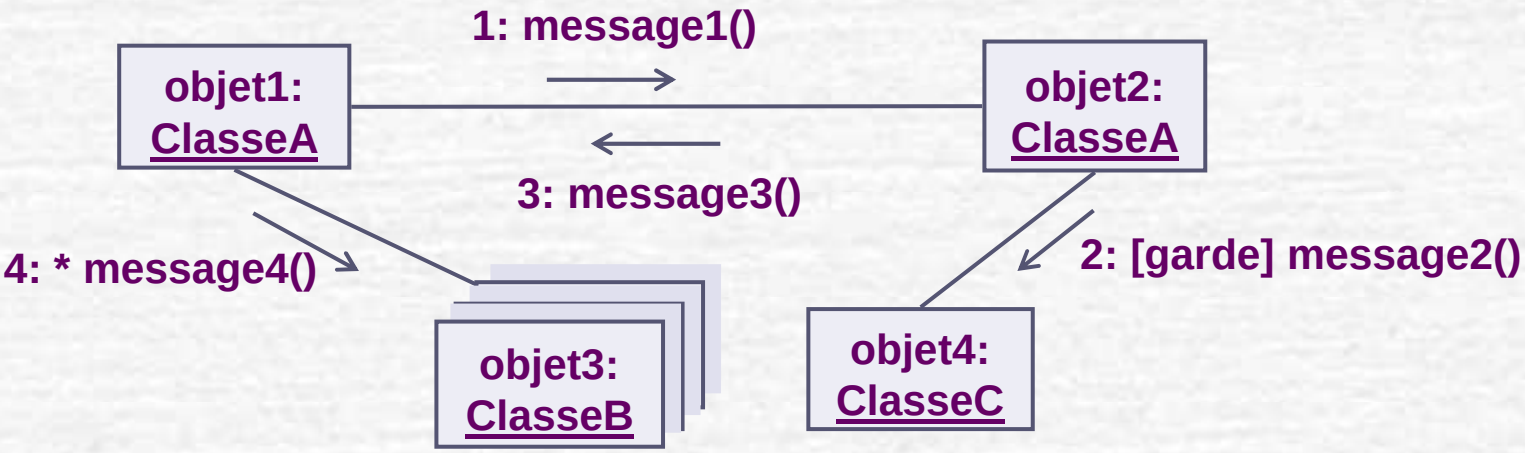
EXEMPLE : MEDUSE



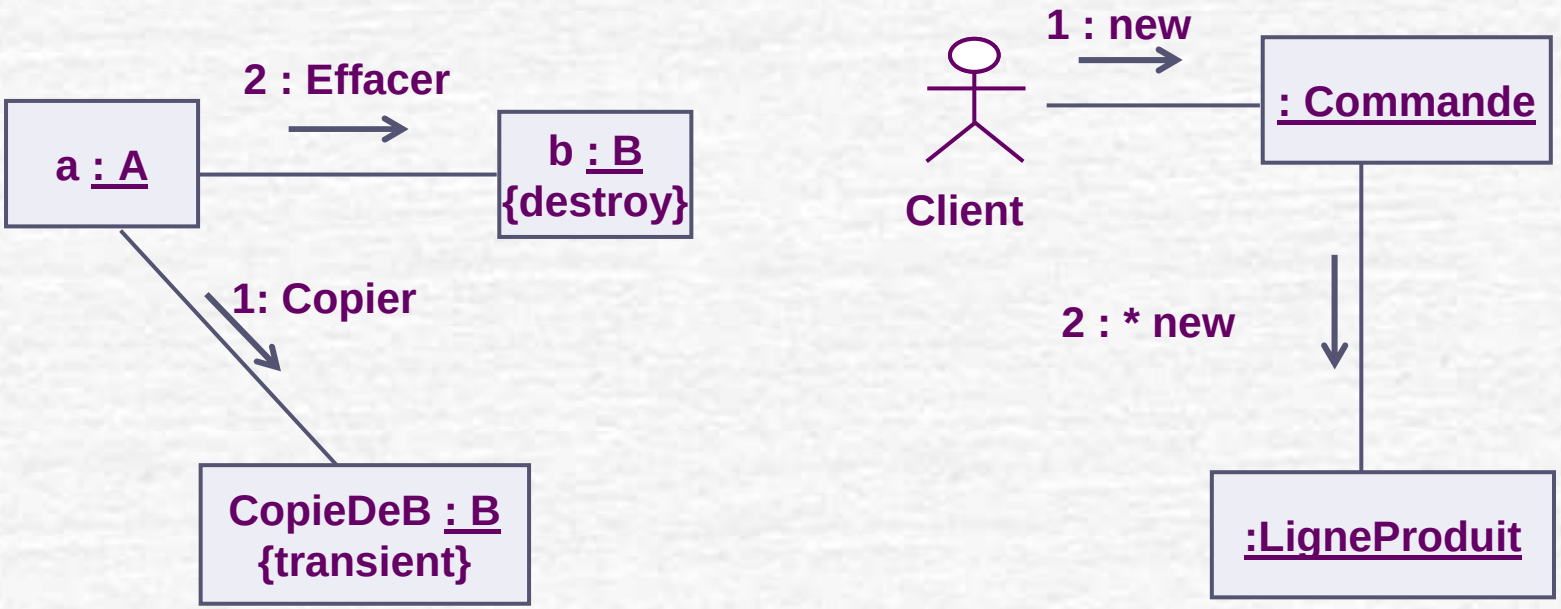
Chapitre 10

DIAGRAMMES DE COMMUNICATION (COD)

DIAGRAMMES DE COMMUNICATION

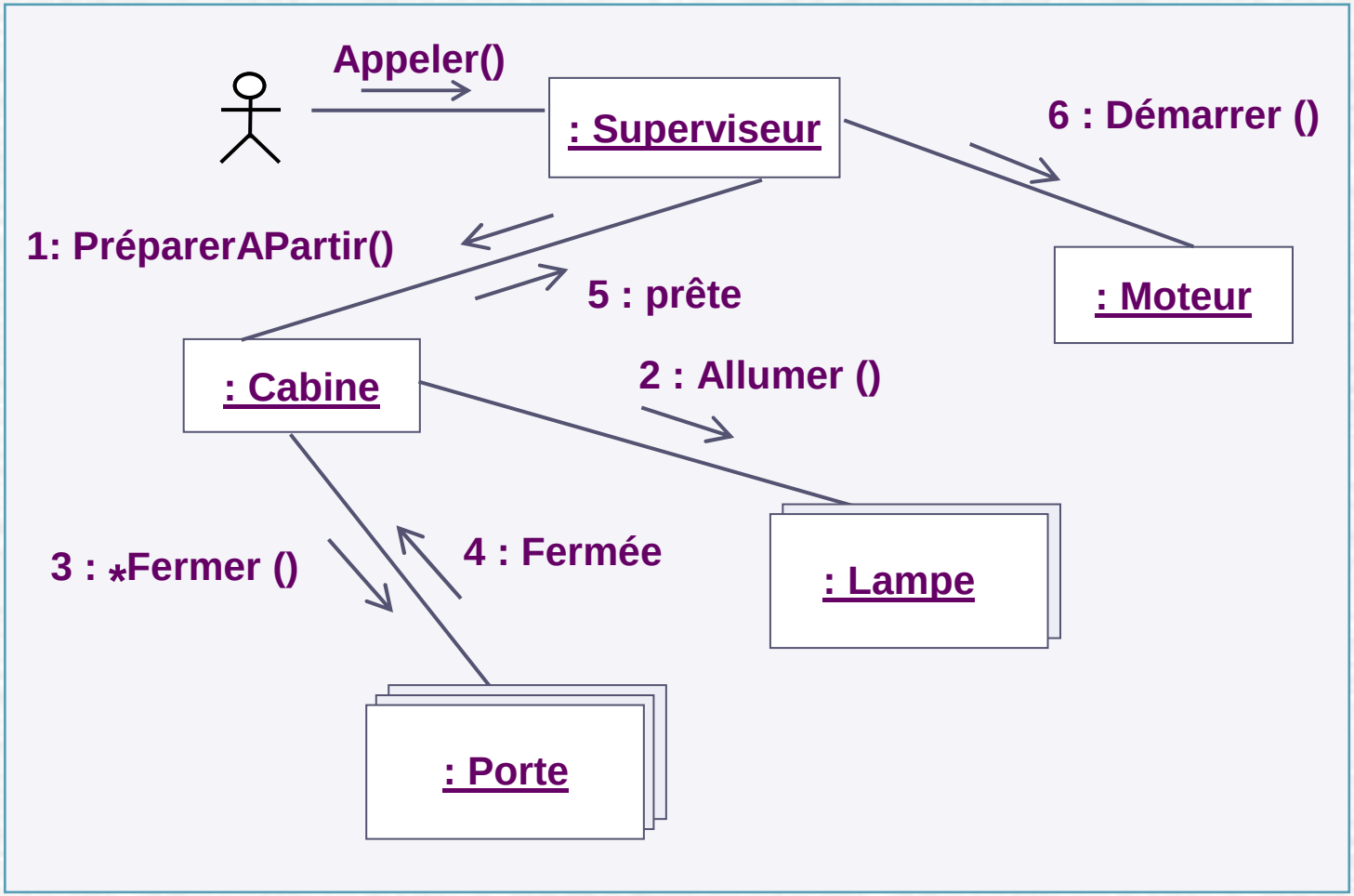


EXEMPLES

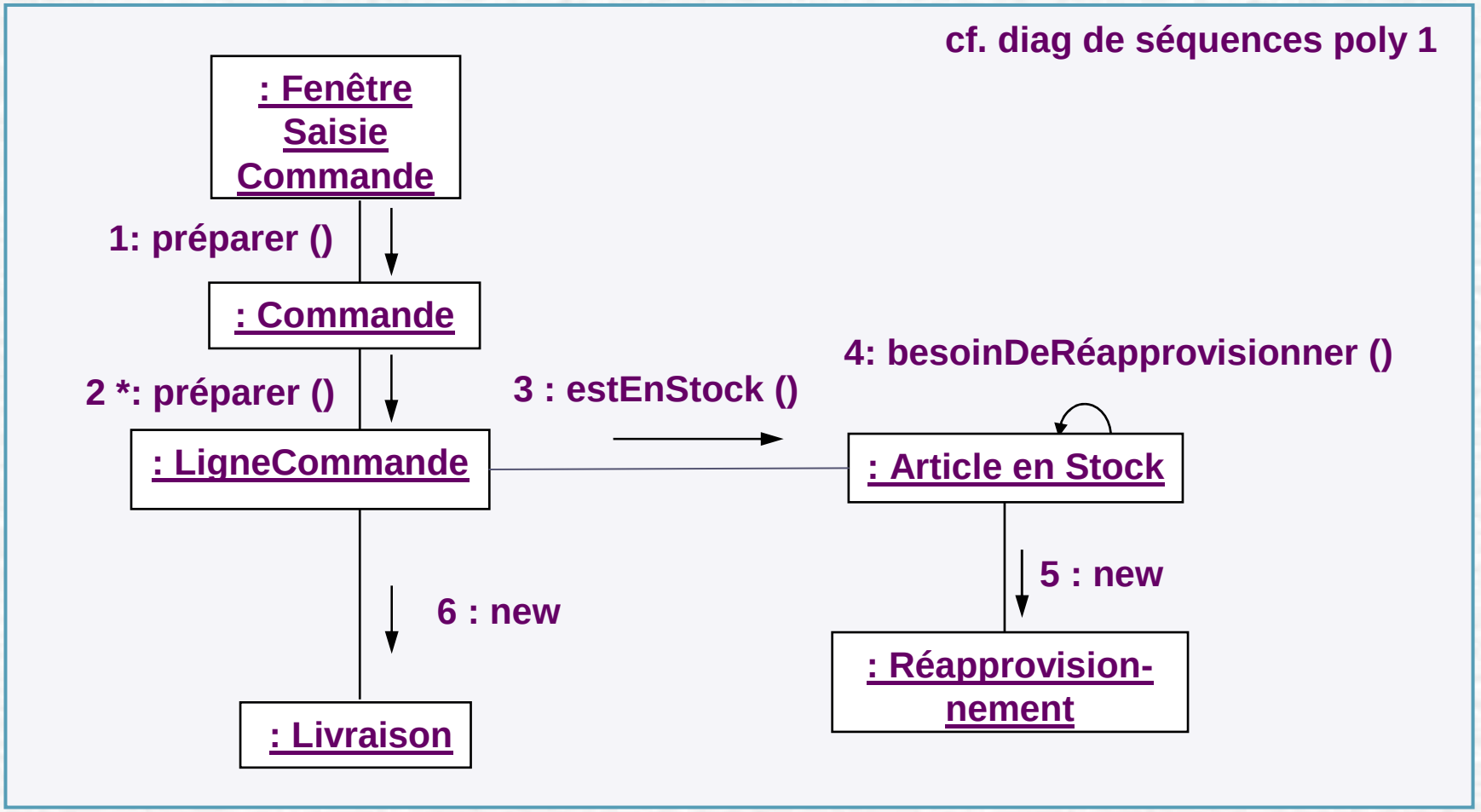

2 : Effacer
a : A
b : B
{destroy}
1: Copier
CopieDeB : B
{transient}
1 : new
Client
: Commande
2 : * new
:LigneProduit

EXEMPLE

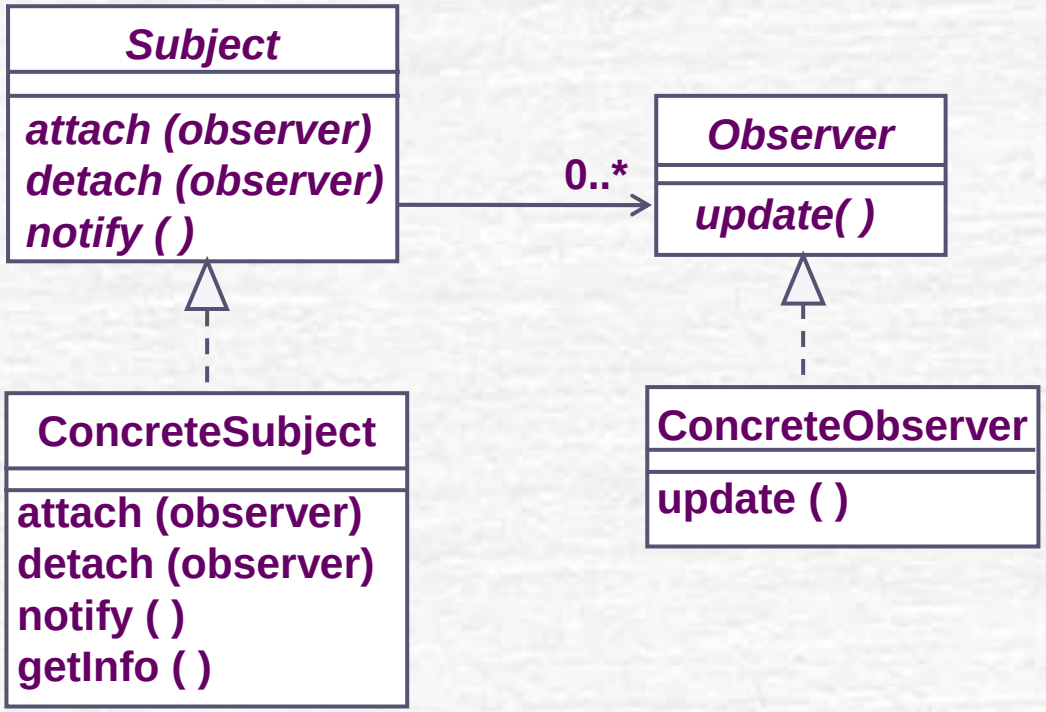
Ascenseur :: appel



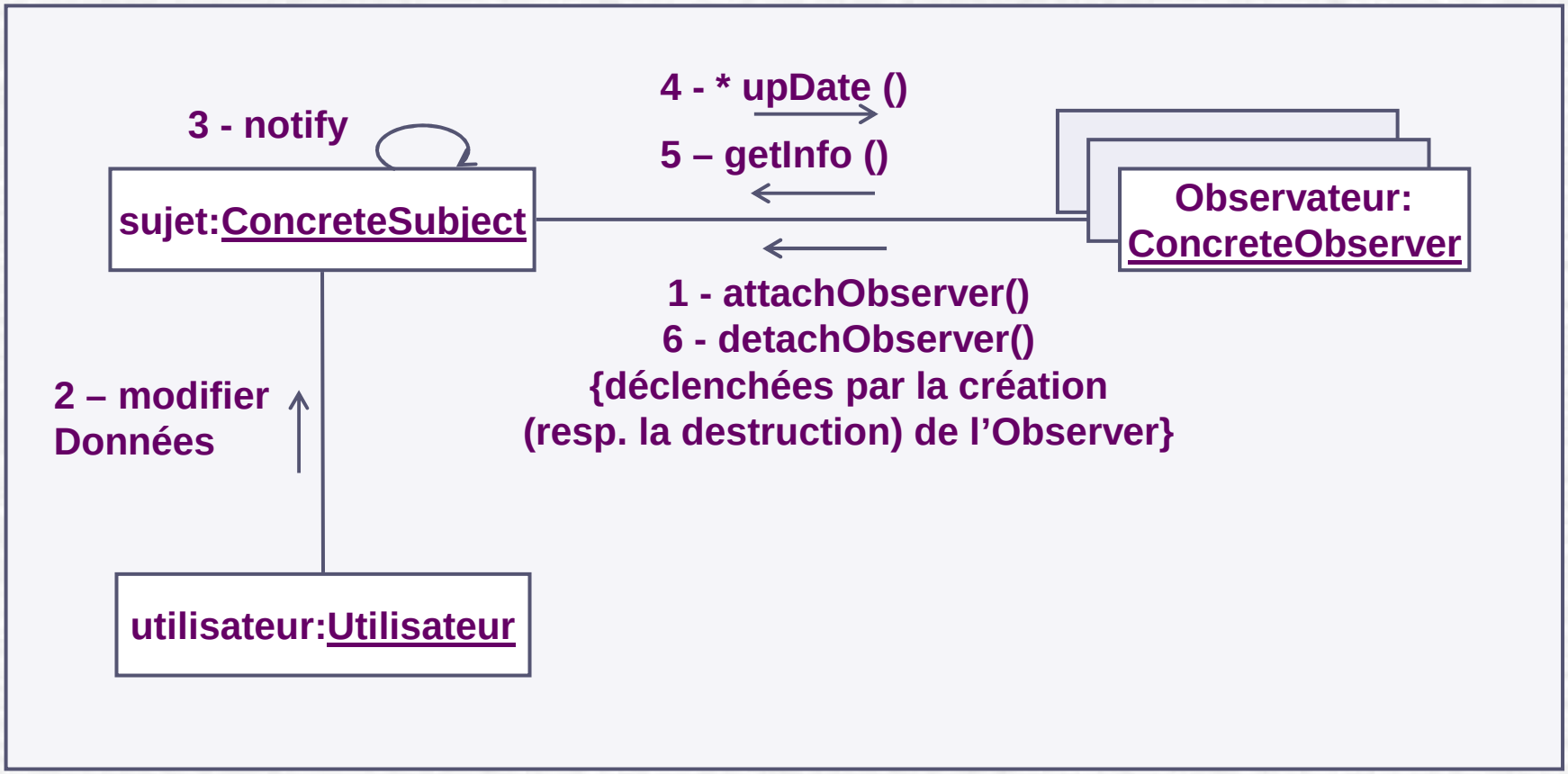
COMPARAISON COMMUNICATION/SEQUENCES



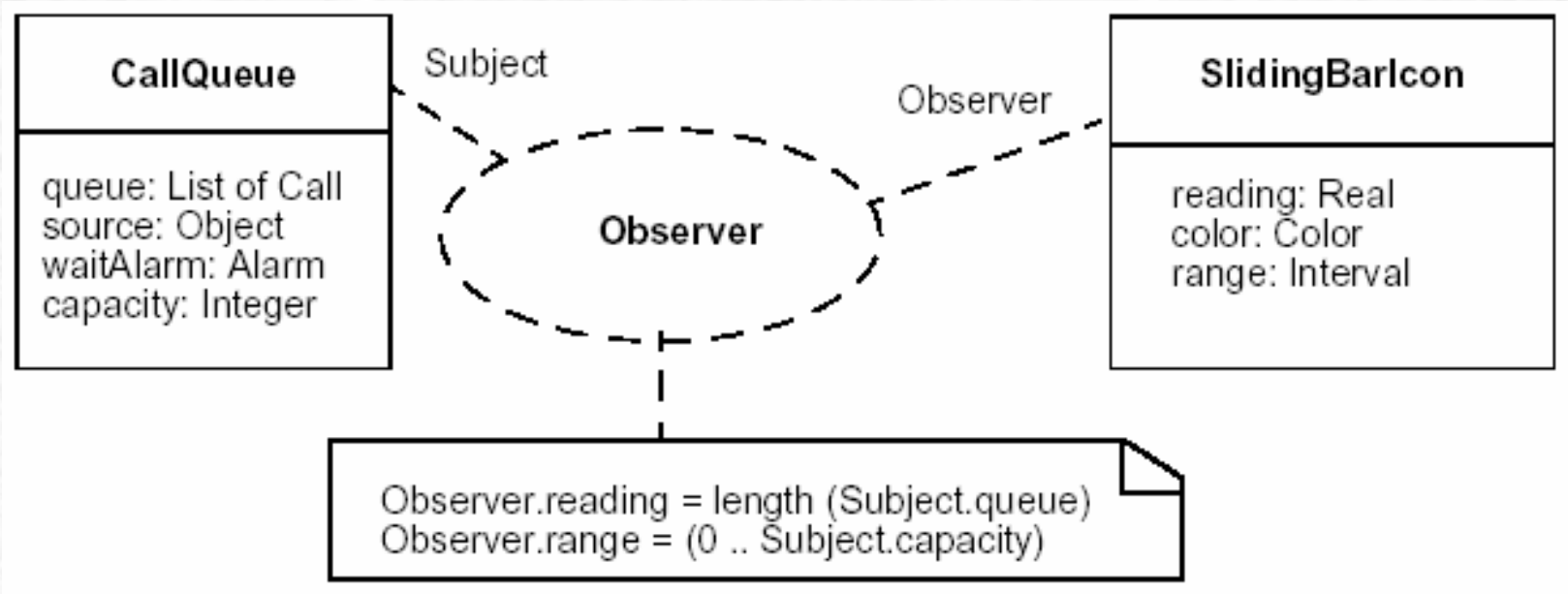
PATRON OBSERVER



LE PATRON OBSERVER



REPRÉSENTATION DE LA COLLABORATION

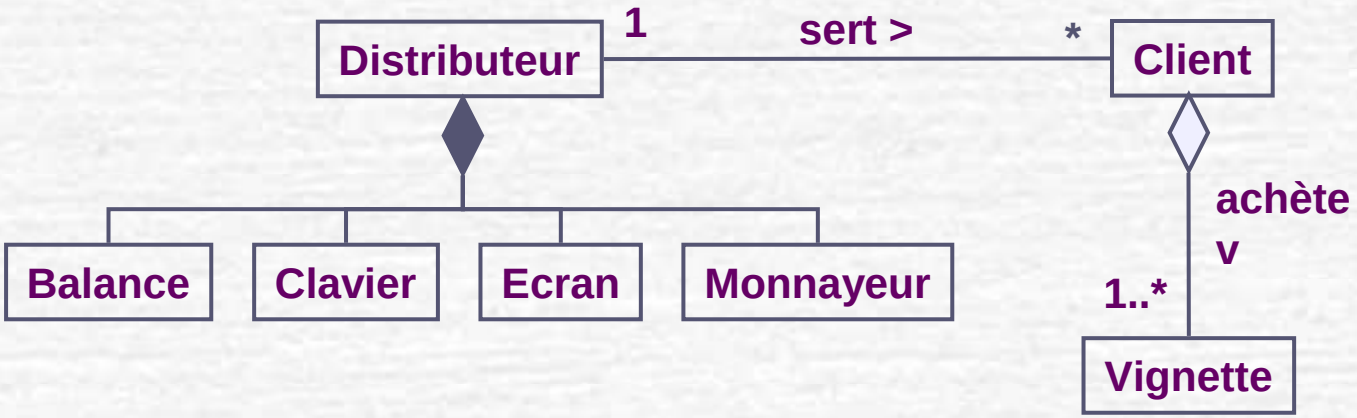


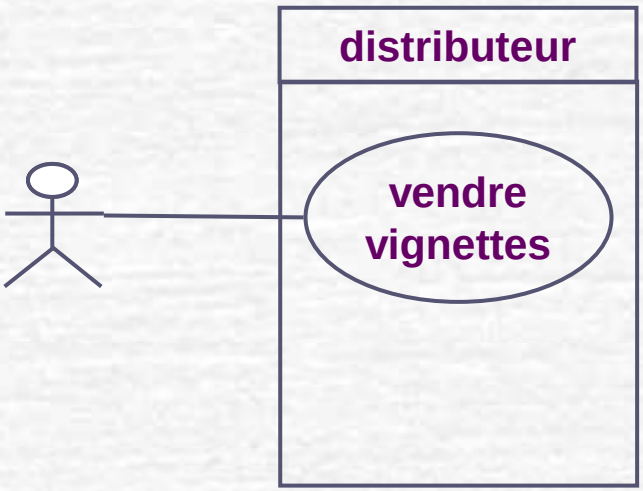
EXERCICE

Au lieu de faire la queue pour affranchir vos lettres, vous préférez utiliser le distributeur automatique, il faut :

- Initialiser le distributeur (p. ex. un bouton sur l'écran tactile)
- Poser une lettre sur la balance,
- Choisir le tarif d'expédition sur l'écran tactile,
- L'écran affiche alors le prix et demande si d'autres lettres sont à affranchir.
- Si oui, le même scénario se répète (à partir de poser une lettre),
- Sinon il faut payer : le montant total s'affiche et vous devez introduire les pièces.
- La monnaie est rendue et les vignettes sont délivrées.

NB : Représenter le distributeur sous forme de plusieurs objets.


Distributeur
1
sert >
*
Client
achète
v
1..*
Vignette
Balance
Clavier
Ecran
Monnayeur


distributeur
vendre
vignettes