

Université Mohammed Premier
École Nationale des Sciences Appliquées d'Oujda
GI3



Unified Modeling Language
Chapitre 6 :
Diagramme d'activité
Diagramme de communication

Plan du cours

1 Diagramme d'activité

2 Diagramme de Communication

Plan du cours

- 1 Diagramme d'activité
- 2 Diagramme de Communication

2 Diagramme de Communication

Downloaded from <http://ajphaphysocpharm.sagepub.com/> at 11:01 11 November 2014

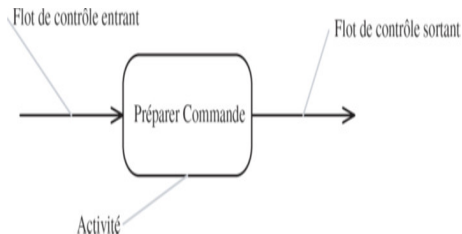
- Le diagramme d'activité est un diagramme états-transitions simplifié pour lequel les états se réduisent à de simples actions ou activités et dont les transitions se déclenchent automatiquement avec éventuellement des gardes
- Sert à analyser et décrire les use case et comprendre un flux de données les traversant.
- On parle généralement dans ce cas de modélisation de workflow (graphe d'activité).
- Les diagrammes d'activités sont utilisés pour la modélisation des aspects dynamiques du système.

Composants

- Contenu d 'un diagramme d 'activités :
 - des états d'activités
 - des transitions
 - des objets : (si nécessaire)
 - des commentaires et des contraintes.

états d'activités

- Les activités décrivent un traitement.
- Le flot de contrôle reste dans l'activité jusqu'à ce que les traitements soient terminés.
- Les activités peuvent être imbriquées hiérarchiquement : on parle alors d'activités composites.



Composants

Transition

- Les transitions connectent les activités entre elles, elles sont déclenchées dès que l'activité source est terminée et déterminent la prochaine activité à déclencher.
- Les transitions sont représentées par des flèches pleines qui connectent les activités entre elles.
 - Elles sont déclenchées dès que l'activité source est terminée.
 - Elles provoquent automatiquement le début immédiat de la prochaine activité à déclencher (l'activité cible).



Composants

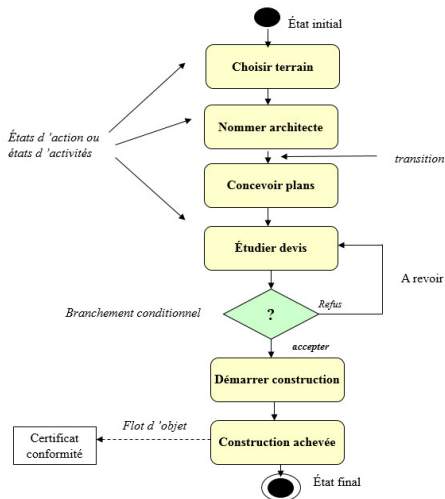
Les conditions (point de choix)

Un noeud de décision permet de faire un choix entre plusieurs flots sortants en fonction des conditions de garde de chaque flot.

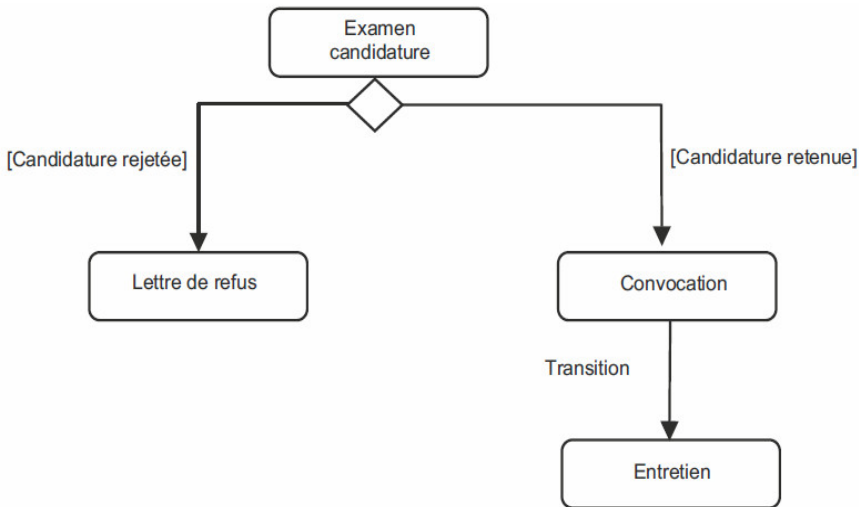
On peut aussi utiliser seulement deux flots de sortie : le premier correspondant à la condition vérifiée et l'autre traitant le cas sinon.



Exemple : Construction d'une maison



Exemple



Composants

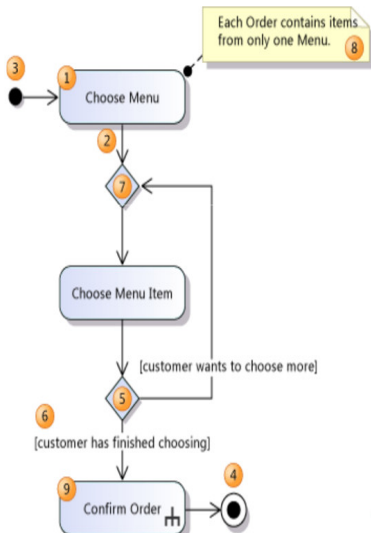
Noeud de fusion

Un noeud de fusion permet d'avoir plusieurs flots entrants possibles et un seul flot sortant. Le flot sortant est donc exécuté dès qu'un des flots entrants est activé.



Composants

présentation

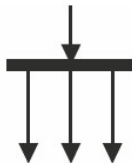


1	Action : single atomic step où l'utilisateur ou le logiciel exécute une tâche avec un nœud de décision
2	Control flow : connecteur de flux entre actions
3	Initial node : première (s) action (s), début du flux
4	Activity final node : fin d'activité
5	Decision node : branche conditionnée dans le flux
6	Guard : la condition pour prendre le connecteur associé
7	Merge node : joindre les flux séparés
8	Comment
9	Call behavior action : activité appelée dans un autre diagramme

News d'UML 2

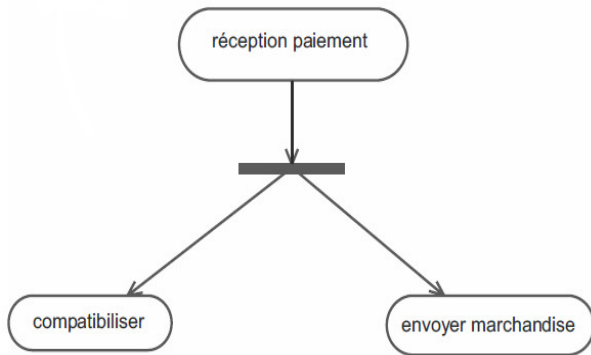
nœud de bifurcation (Fork)

Un noeud de bifurcation (fourche) permet à partir d'un flot unique entrant de créer plusieurs flots concurrents en sortie de la barre de synchronisation.



News d'UML 2

nœud de bifurcation (Fork)

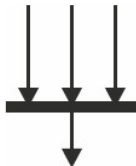


News d'UML 2

nœud de jonction (Join)

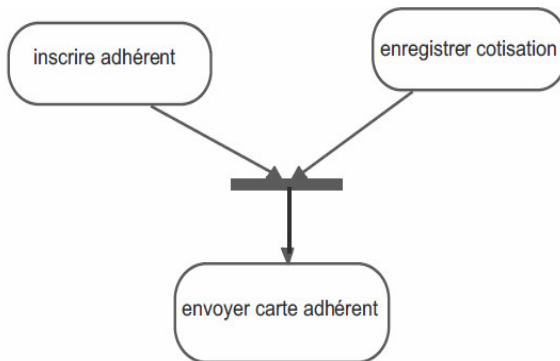
Un nœud de jonction (synchronisation) permet, à partir de plusieurs flots concurrents en entrée de la synchronisation, de produire un flot unique sortant.

Le nœud de jonction est le symétrique du nœud de bifurcation.



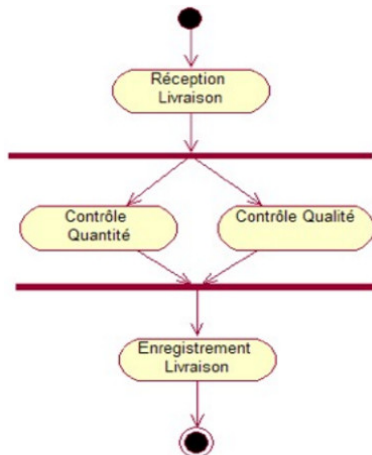
News d'UML 2

nœud de jonction (Join)



News d'UML 2

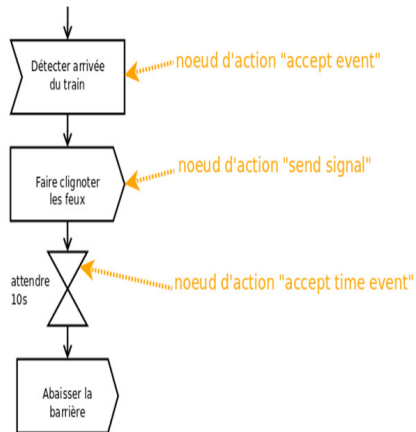
Fork & Join : exemple



News d'UML 2

accept, send and time signal

- Accept signal : Une action qui attend l'apparition d'une réunion des conditions d'événement spécifié
- Send signal : Une action qui crée un exemple de signal à partir de ses entrées, et le transmet à l'objet cible
- Time signal qui déclenche une activité



News d'UML 2

- Un flot de contrôle qui atteint un nœud de contrôle de type « flow final » est détruit. Les autres flots de contrôle ne sont pas affectés.



Nœud de terminaison qui définit la fin du flux de l'activité

Final activity node

- Ce type de nœud est moins fort qu'un nœud de contrôle final.
- Dans ce cas, tous les autres flots de contrôle de l'activité sont interrompus et détruits.



Nœud de terminaison qui définit la fin du flux

Final node

News d'UML 2

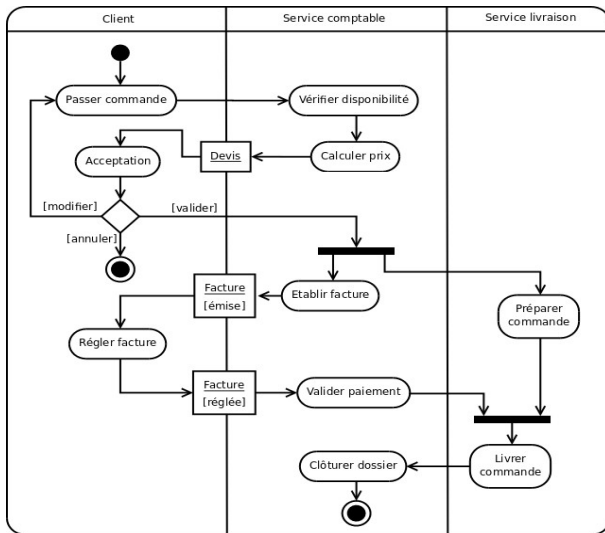
Les couloirs

- Pour modéliser un traitement mettant en œuvre plusieurs classeurs, on peut spécifier le classeur responsable de chaque activité.
- Les partitions permettent d'attribuer les activités à des éléments particuliers du modèle.
- Une partition peut elle-même être décomposée en sous-partitions.
- Pour spécifier qu'une activité est effectuée par un classeur particulier, on la positionne dans la partition correspondante.
- Chaque couloir correspond à un domaine de responsabilité d'un certain nombre d'actions.

«external»	«attribute» libelléService : Service	
Client	<u>Service comptable</u>	<u>Service livraison</u>

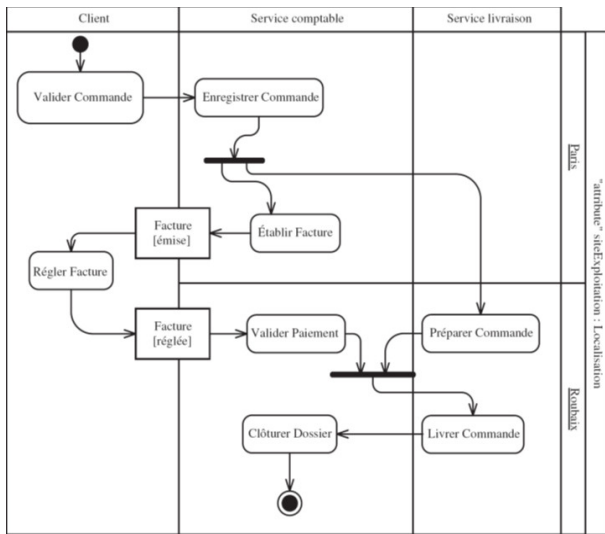
News d'UML 2

Exemple



News d'UML 2

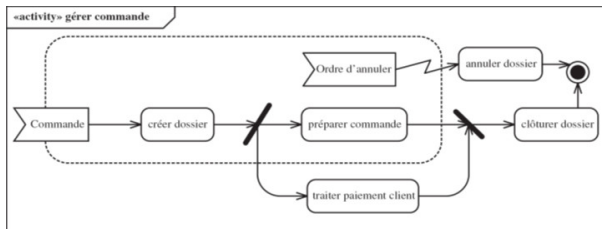
Couloirs : exemple



News d'UML 2

Les régions interruptible

- Une région interruptible est représentée par un cadre arrondi en pointillés.
- Si l'événement d'interruption se produit, toutes les activités en cours dans la région interruptible sont stoppées
- Le flot de contrôle suit la flèche en zigzag qui quitte la région.



News d'UML 2

Exception

- Les exceptions permettent d'interrompre un traitement quand une situation qui dévie du traitement normal se produit. Elles assurent une gestion plus propre des erreurs qui peuvent se produire au cours d'un traitement.
- Un flot de données correspondant à une exception est matérialisé par une flèche en zigzag .

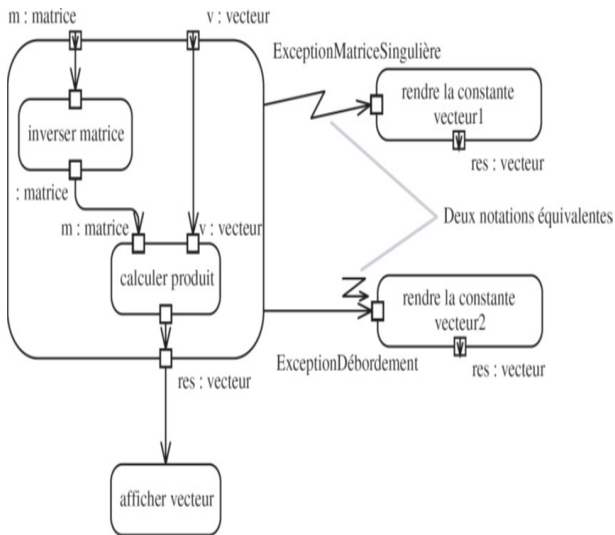
Rappel :

Le processus d'exception se compose de deux phases :

- Throw : lancer l'exception à un certain point dans le code.
- Catch : attraper l'exception et exécuter une action.

L'exception peut survenir à plusieurs endroits pour une seule section du code
=> regroupement dans un bloc (en java try)

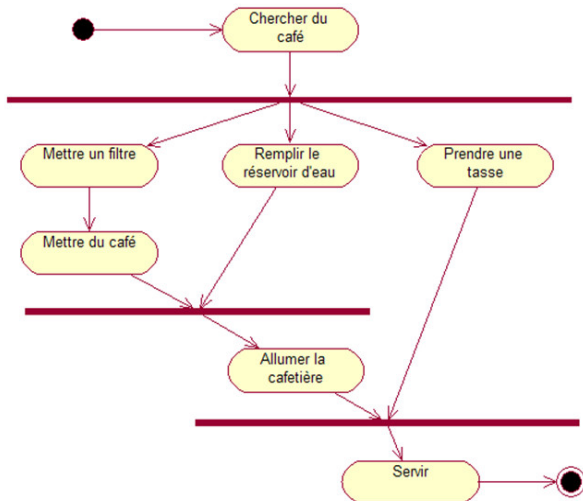
Exception



Exercice

- Construire un diagramme d'activité représentant l'utilisation d'une cafetière électrique :
 - premier état : chercher du café
 - dernier état : Servir du café

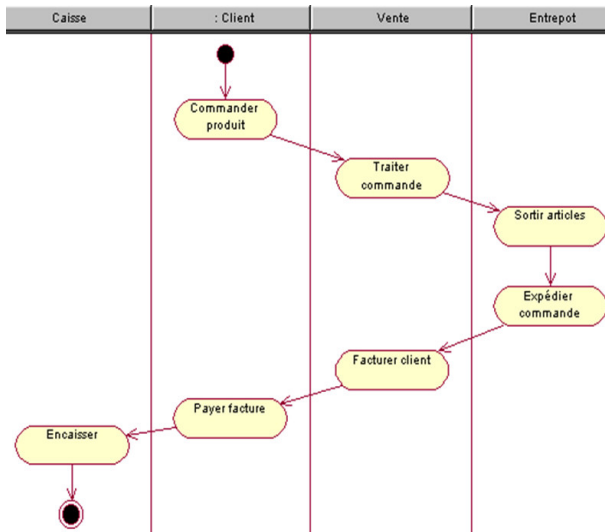
Solution



Exercice

- Construire un diagramme d'activité pour modéliser le processus de commander d'un produit. Le processus concerne les acteurs suivants :
 - Client : qui commande un produit et qui paie la facture
 - Caisse : qui encaisse l'argent du client
 - Vente : qui s'occupe de traiter et de facturer la commande du client
 - Entrepôt : qui est responsable de sortir les articles et d'expédier la commande.

Solution



Avant propos

utilité

- Un diagramme de communication est un diagramme d'interactions UML 2.0 (appelé diagramme de collaboration en UML 1), représentation simplifiée d'un diagramme de séquence se concentrant sur les échanges de messages entre les objets.
- Modélise la dynamique d'un comportement relié à un use case en se focalisant sur la collaboration des objets impliqués.

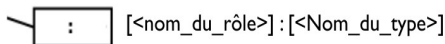
Diagramme de séquence Vs diagramme de communication :

- Un diagramme de séquence montre des interactions sous un angle temporel, en mettant l'accent sur le séquençage temporel de messages échangés entre des lignes de vie
- Un diagramme de communication montre une représentation spatiale des lignes de vie.
- Ils représentent la même chose, mais sous des formes différentes.

Composants

Rôle et connecteur

- Le rôle permet de définir le contexte d'utilisation de l'interaction.
- Un rôle dans un diagramme de communication correspond à une ligne de vie dans un diagramme de séquences.
- Les relations entre les lignes de vie sont appelées connecteurs .



lifeline



Lifeline anonymous User Lifeline data from class Stock

Composants

Message

- Comme dans les diagrammes de séquence, on distingue deux types de messages :

- Un message synchrone bloque l'expéditeur jusqu'à la réponse du destinataire. Le flot de contrôle passe de l'émetteur au récepteur.



- Un message asynchrone n'est pas bloquant pour l'expéditeur. Le message envoyé peut être pris en compte par le récepteur à tout moment ou ignoré.



- Pour représenter les aspects temporels, on adjoint des numéros de séquence aux messages.

- Des messages successifs sont ordonnés selon un numéro de séquence croissant (1, 2, 3, ... ou encore 3.1, 3.2, 3.3, ...).
 - Des messages envoyés en cascade (ex : appel de méthode à l'intérieur d'une méthode) portent un numéro d'emboîtement avec une notation pointée
 - 1.1, 1.2, ... pour des messages appelés par une méthode dont l'appel portait le numéro 1
 - 2.a.1, 2.a.2, ... pour des messages appelés par une méthode dont l'appel portait le numéro 2.a

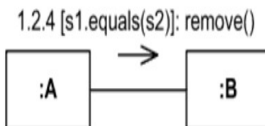
Composants

Message

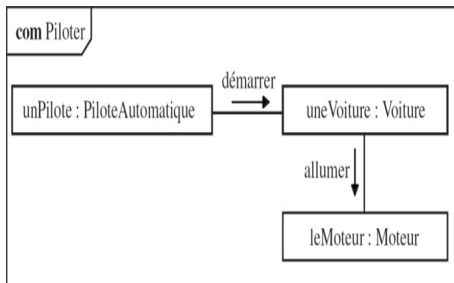
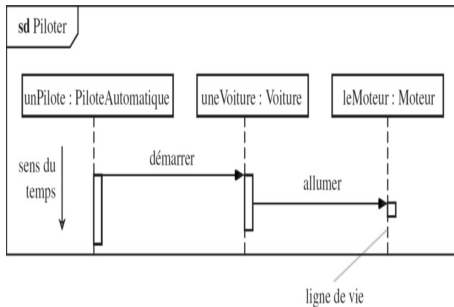
syntaxe de message

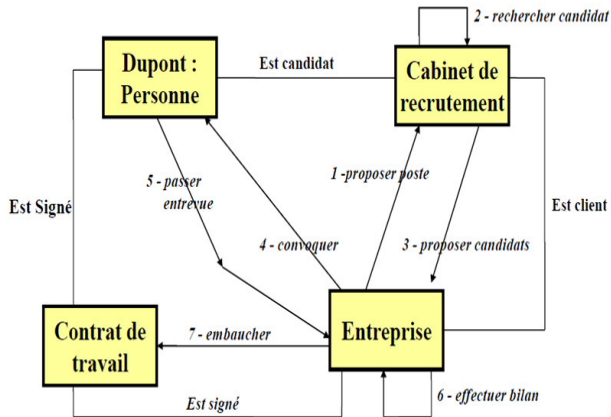
[[séq] '['cond']' [*||| '['iter']'] :] [var :=] msg([par])

- cond : condition expression booléenne
- séq : Numero de séquence du message
- iter spécifie l'envoi séquentiel (ou en parallèle, avec ||) de plusieurs messages.
- *|| : type d'envoi (parallèle, séquentiel)
- var : valeur de retour du message
- msg : nom du message
- par : paramètres

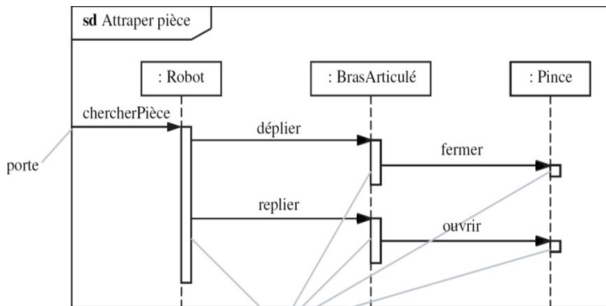


Exemple

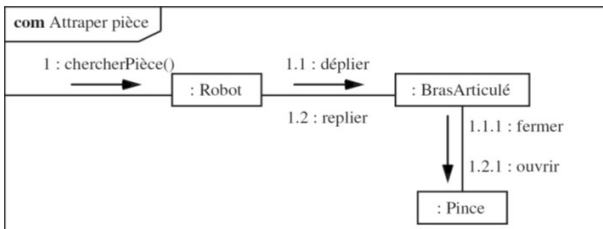




Exemple



spécifications de l'exécution des méthodes



Exercice

Un objet nommé B747 de classe Avion et en état détresse est en relation avec Luna, une tour de contrôle. Un ensemble d'autres avions anonymes dont l'état est à terre sont aussi liés à Luna. La tour de contrôle communique avec P123, une caserne de pompiers. Considérons le scénario suivant :

- L'avion en détresse envoie un SOS à la tour de contrôle Simultanément :
 - La tour de contrôle ordonne à l'ensemble des avions en piste de se diriger vers le parking
 - Elle envoie aux pompiers l'ordre de venir sur la piste P1
- La tour de contrôle demande à l'avion en détresse d'atterrir sur la piste P1

Question : Donnez le diagramme de communication correspondant à ce scénario

Solution

