

DIAGRAMME DE SÉQUENCE

Ilhem Boussaïd

ilhem_boussaid@yahoo.fr

Université des Sciences et de la Technologie Houari Boumediene
Licence 3 Académique

<http://sites.google.com/site/ilhemboussaid>

6 décembre 2010

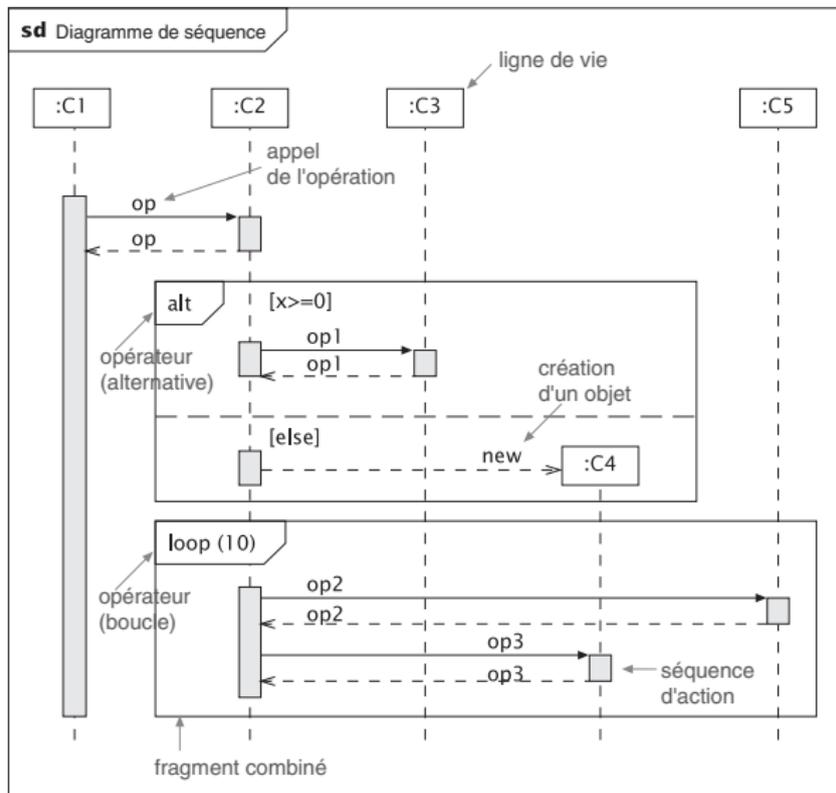
DIAGRAMMES DE SÉQUENCE

- Les diagrammes de séquences permettent de décrire **COMMENT** les éléments du système interagissent entre eux et avec les acteurs.
 - Montrent les interactions entre objets selon un point de vue temporel
 - Description de scénarios types et des exceptions
- Deux utilisations principales :
 - 1 **Documentation des CU** (point de vue Fonctionnel)
 - 2 Représentation précise des **interactions** (point de vue Dynamique)
 - identification des messages, des envois, réceptions, etc.

DIAGRAMMES DE SÉQUENCE

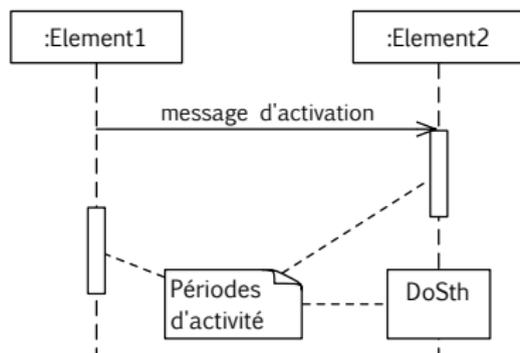
- Concepts principaux :
 - 1 Les participants (le plus souvent des objets)
 - Une ligne de vie : représente un participant à une interaction (objet ou acteur).
 - Des zones d'activation
 - 2 Les messages :
 - L'opération et éventuellement ses paramètres
 - Éventuellement son résultat
 - 3 Des Fragments combinés
 - Alt : conditionnelle
 - Loop : boucle
 - Ref : reference a un autre diagramme de sequence (=appel de fonction)
 - Etc.

REPRÉSENTATION

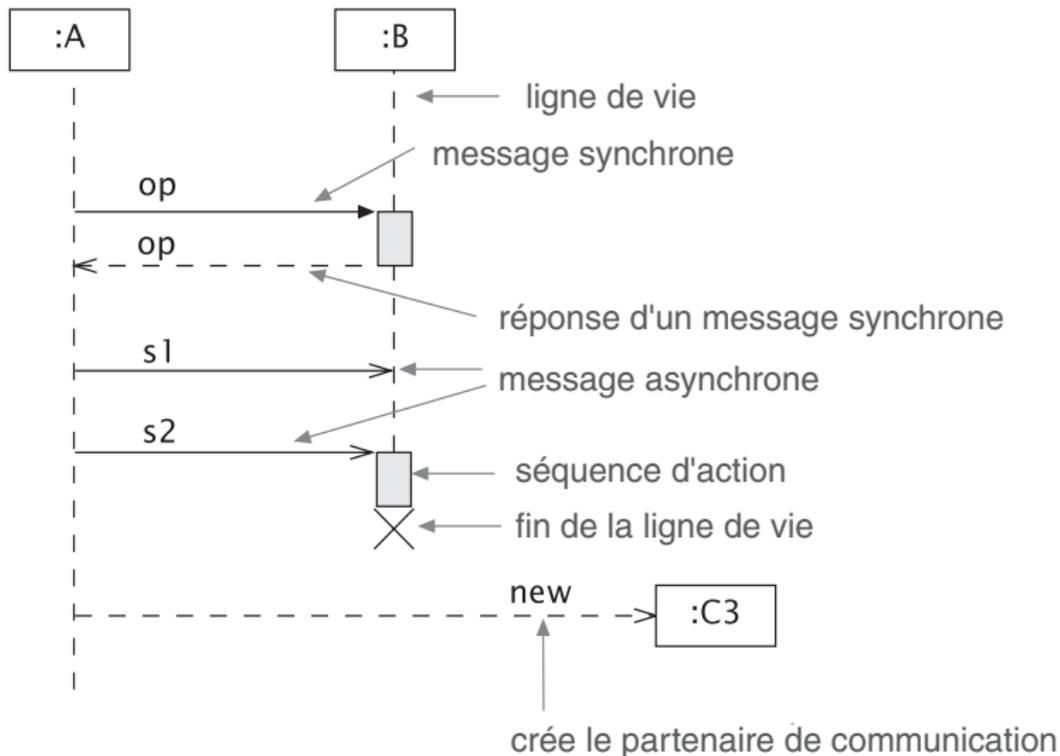


PÉRIODE D'ACTIVITÉ

- La réception des messages provoque une période d'activité (rectangle vertical sur la ligne de vie) marquant le traitement du message.
- Période durant laquelle un objet effectue une action
- Etat **actif** (\neq durée de vie)
- un objet peut être actif plusieurs fois



CATÉGORIES DE MESSAGES



CATÉGORIES DE MESSAGES

Principales catégories de messages :

- **Message synchrone** : émetteur bloqué pendant le traitement du message par le récepteur (appel)
 - Typiquement : appel de méthode (Si un objet A invoque une méthode d'un objet B, A reste bloqué tant que B n'a pas terminé.



- **Message asynchrone** : non bloquant.
 - Le message envoyé peut être pris en compte par le récepteur à tout moment ou ignoré.



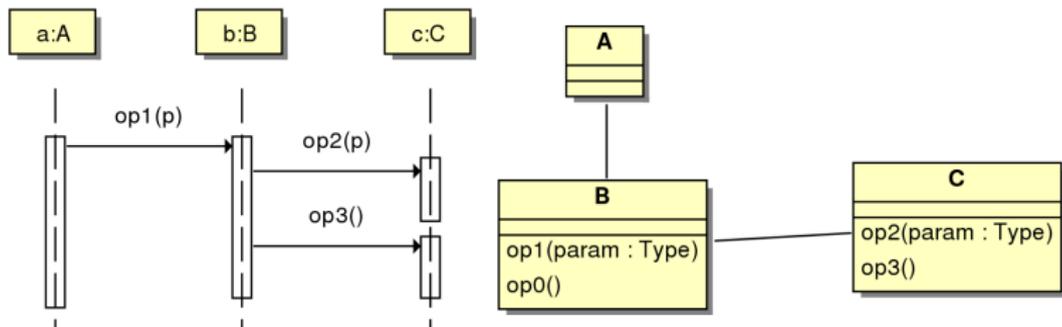
- **Message de retour** : On peut associer aux messages d'appel de méthode un message de retour (en pointillés) marquant la reprise du contrôle par l'objet émetteur du message synchrone.



CORRESPONDANCE MESSAGES / OPÉRATIONS

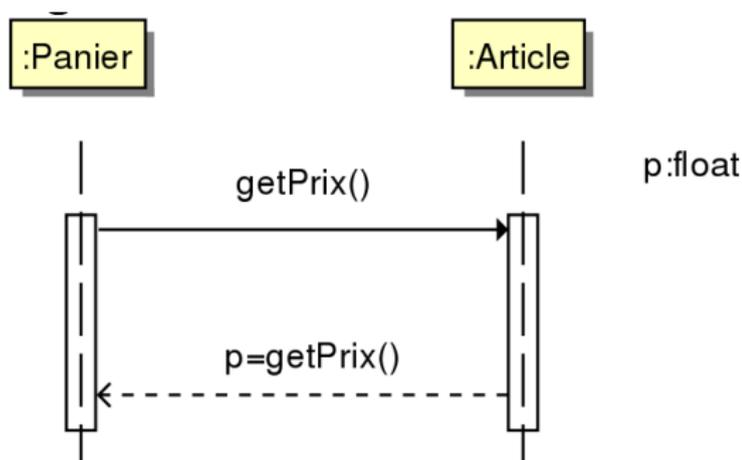
- Les messages synchrones correspondent à des opérations dans le diagramme de classes.

Envoyer un message et attendre la réponse pour poursuivre son activité revient à invoquer une méthode et attendre le retour pour poursuivre ses traitements.



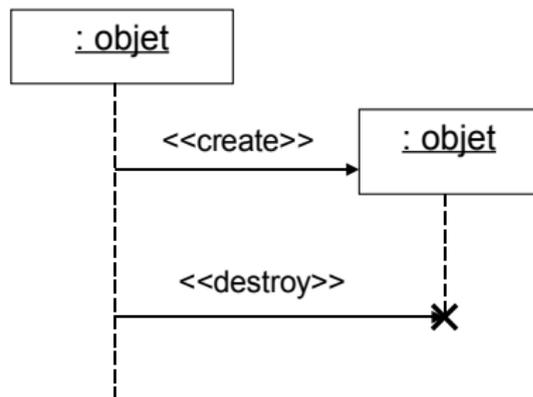
MESSAGES DE RETOUR

- Le récepteur d'un message synchrone rend la main à l'émetteur du message en lui envoyant un message de retour
- Les messages de retour sont optionnels : la fin de la période d'activité marque également la fin de l'exécution d'une méthode.
- Ils sont utilisés pour spécifier le résultat de la méthode invoquée.



MESSAGES DE CRÉATION/DESTRUCTION

- Un message peut entraîner la création ou la destruction d'objets
 - La **création** d'un objet est matérialisée par une flèche qui pointe sur le sommet d'une ligne de vie (On peut aussi utiliser un message asynchrone ordinaire portant le nom «**create**»).
 - La **destruction** d'un objet est matérialisée par une croix qui marque la fin de la ligne de vie de l'objet.



CATÉGORIES DE MESSAGES

- message **trouvé** (Found message) :
 - message dont on ignore la provenance
 - en dehors du cadre décrit par le Diagramme de Séquence
- message **perdu** (Lost message) :
 - message envoyé, mais jamais reçu

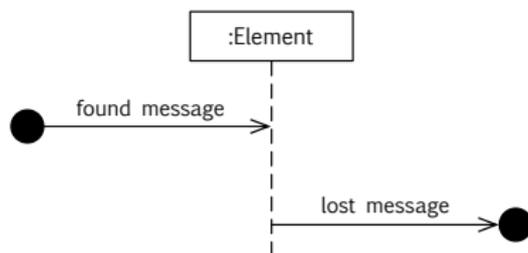
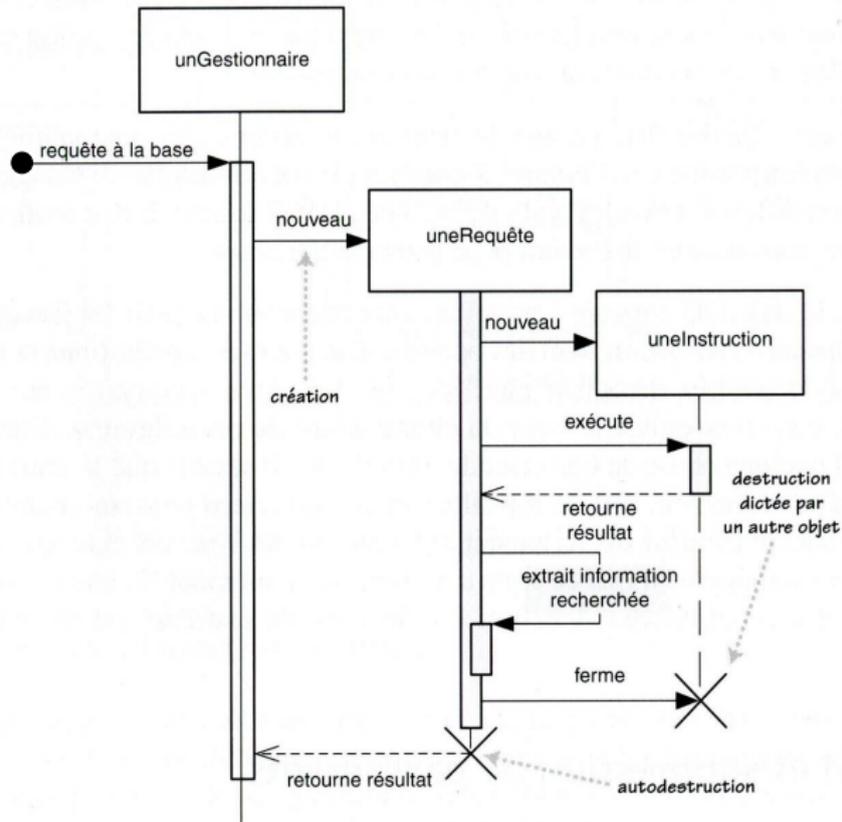
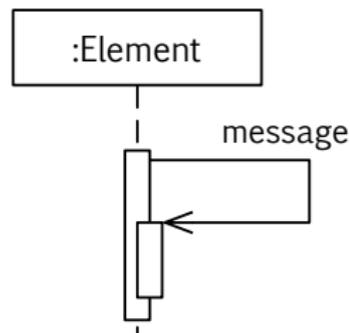
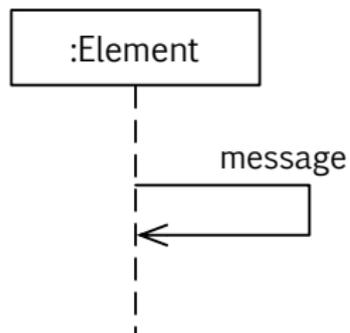


DIAGRAMME DE SÉQUENCE - EXEMPLE



MESSAGE REFLEXIF

- Un objet peut s'envoyer des messages
 - appel à une autre méthode de l'objet
 - appel récursif



DURÉES ET CONTRAINTES TEMPORELLES

Représentation des délais de transmission :

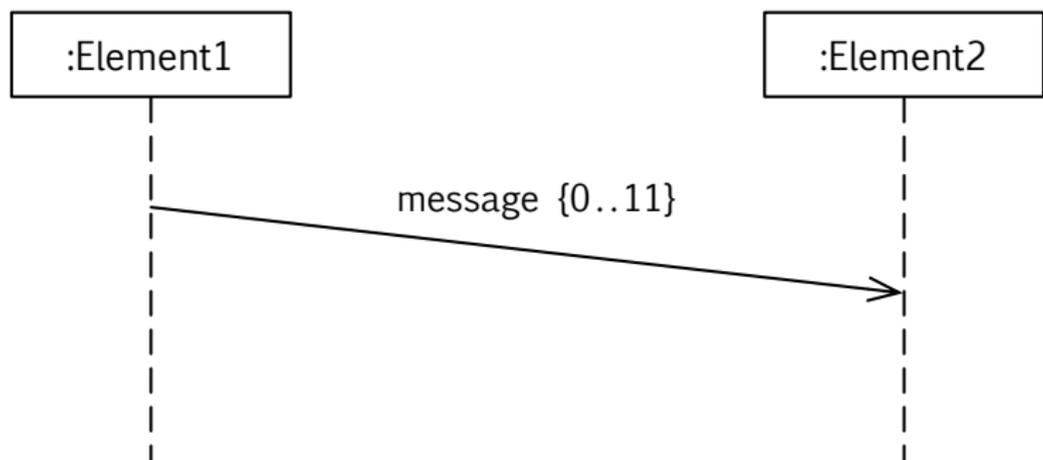
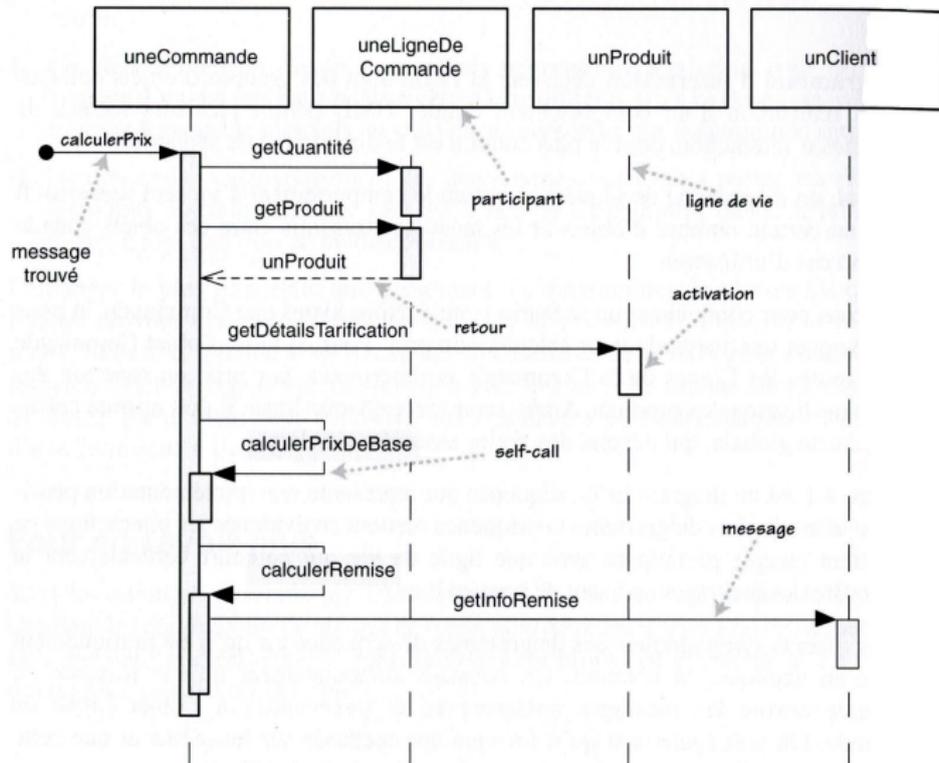


DIAGRAMME DE SÉQUENCE - EXEMPLE



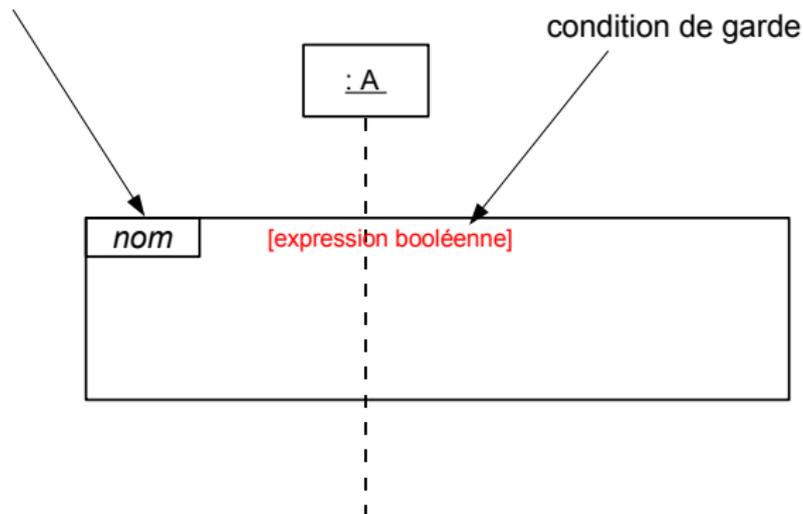
FRAGMENTS COMBINÉS

- Un fragment combiné permet de décomposer une interaction complexe en fragments suffisamment simples pour être compris.

Fragments combinés

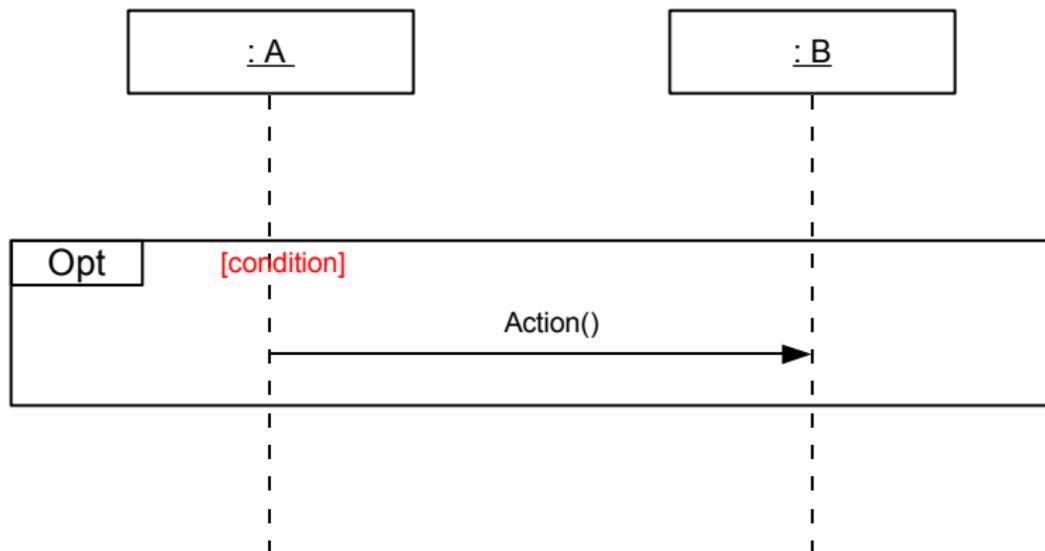
Types

- 1 Opt
- 2 Loop
- 3 Alt
- 4 Break
- 5 Critique
- 6 Ref
- 7 ...



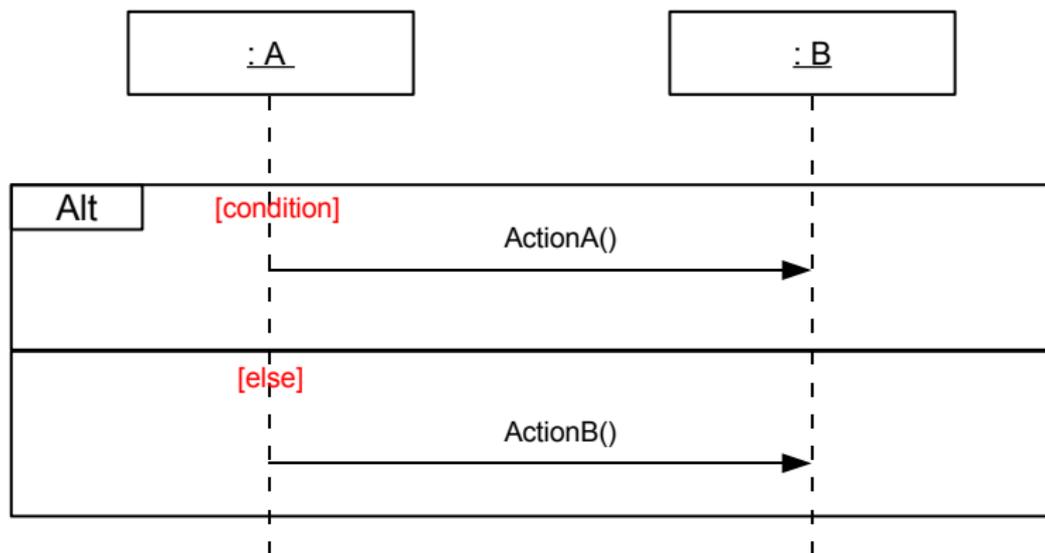
FRAGMENTS COMBINÉS

- **Opt** : Fragment parcouru si une condition est vérifiée



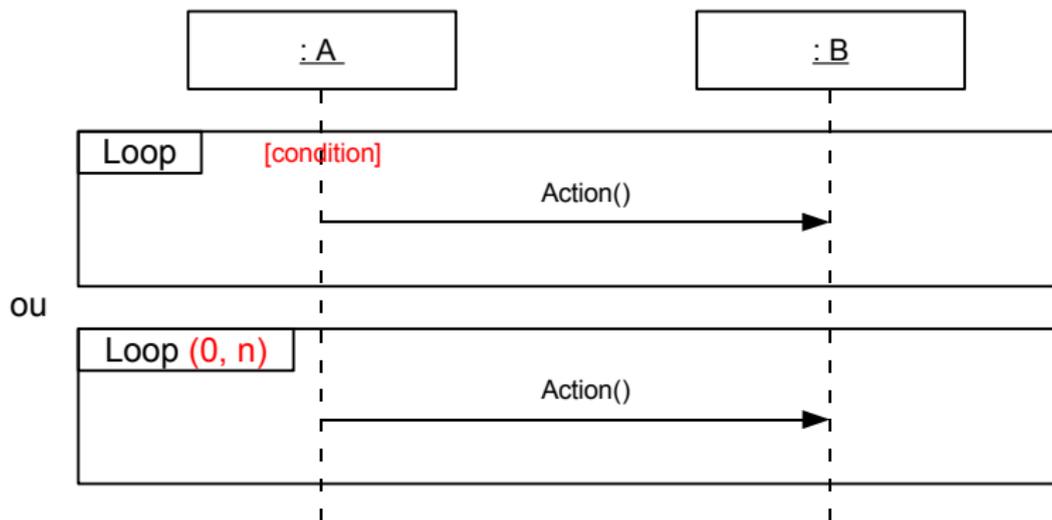
FRAGMENTS COMBINÉS

- **Alt** : Equivalent à la structure de contrôle "si .. alors .. sinon".

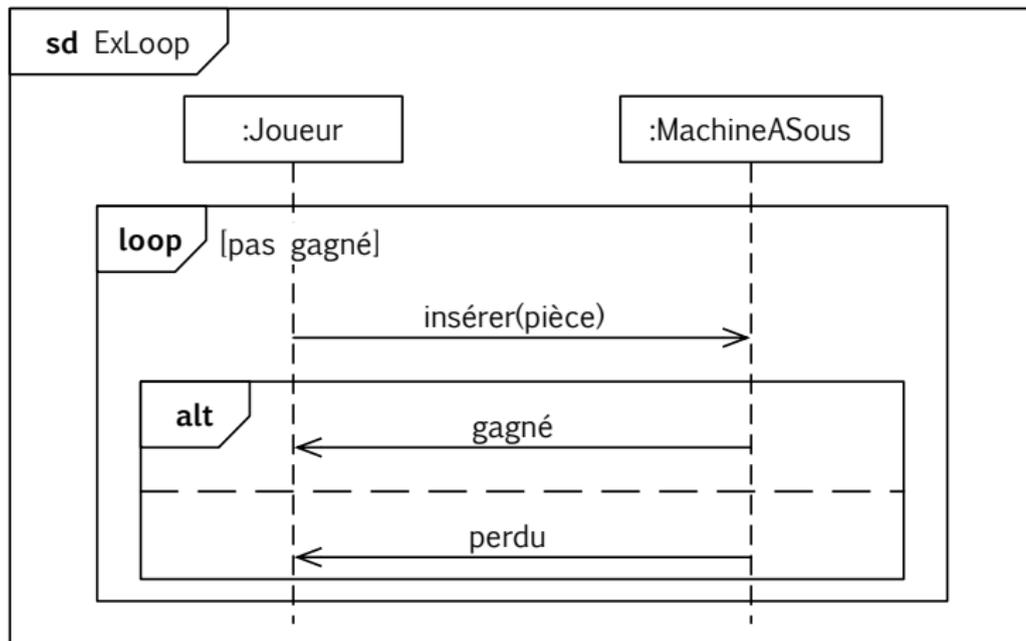


FRAGMENTS COMBINÉS

- **Loop** : Répétition du fragment tant que la condition est vérifiée

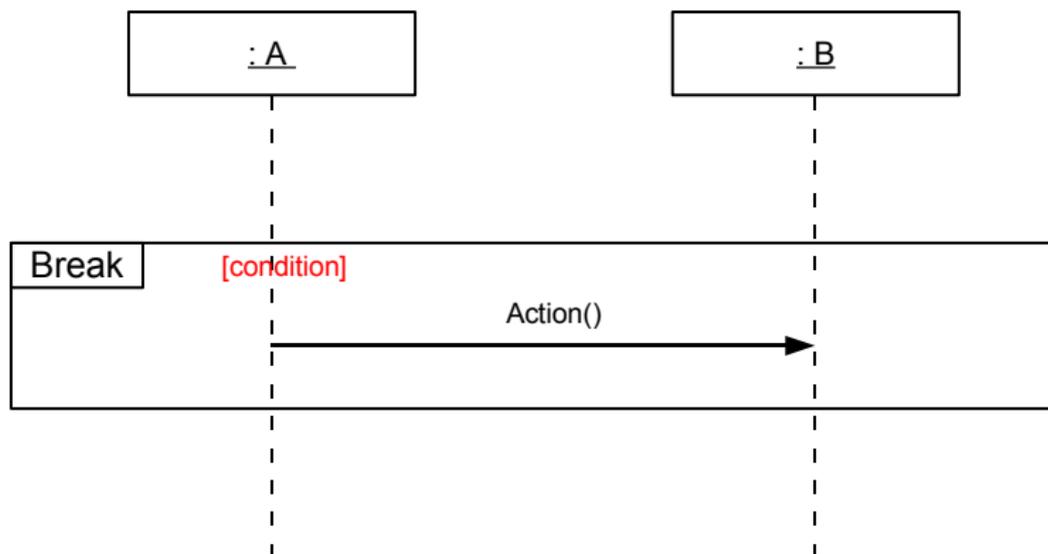


FRAGMENTS COMBINÉS - EXEMPLE



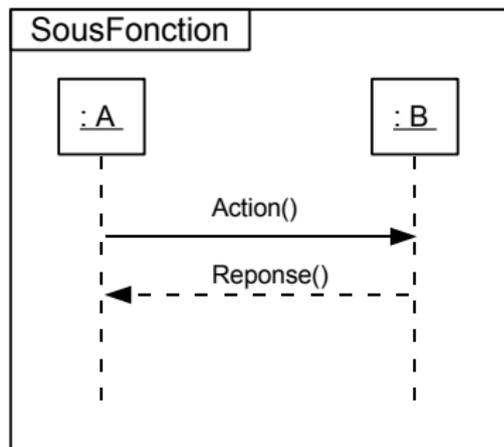
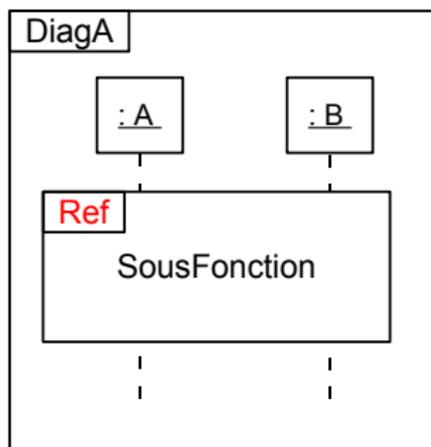
FRAGMENTS COMBINÉS

- **Break** : Fragment exécuté et met fin au fragment englobant



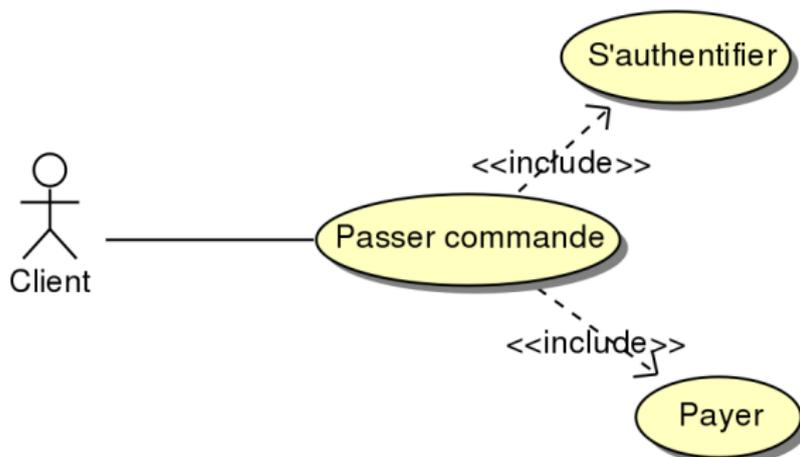
FRAGMENTS COMBINÉS

- **Décomposition** : mot clef "ref"



UTILISATION D'UN DS POUR MODÉLISER UN CAS D'UTILISATION

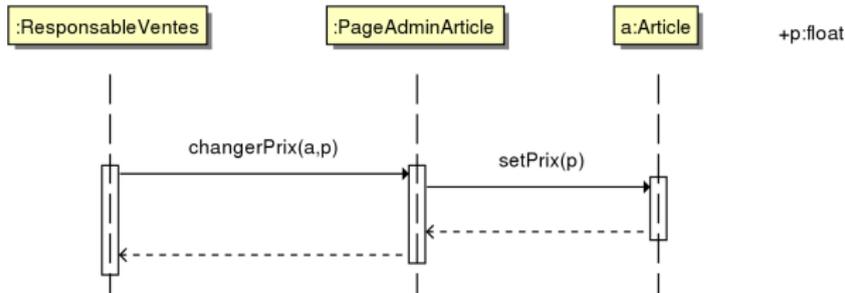
- Chaque cas d'utilisation donne lieu à un diagramme de séquences



- Les inclusions et les extensions sont des cas typiques d'utilisation de la réutilisation par référencement

DIAGRAMME DE SÉQUENCE - EXEMPLE D'INTERACTION

- Diagramme de séquences correspondant :



- Opérations nécessaires dans le diagramme de classes :

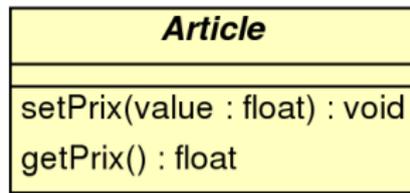


DIAGRAMME DE SÉQUENCE - RETOUR SUR L'ÉTUDE DE CAS - LE DAB (DISTRIBUTEUR AUTOMATIQUE DE BILLETS)

Le système offre les services suivants :

- 1 Distribution d'argent à tout porteur de carte de crédit, via un lecteur de carte et un distributeur de billet
- 2 Consultation de solde de compte, dépôt d'espèces et de chèques pour les clients porteurs d'une carte de crédit de la banque associée au DAB

plus. . .

- les transactions sont sécurisées
- il faut parfois recharger le distributeur, etc.

DIAGRAMME DE SÉQUENCE - RETOUR SUR L'ÉTUDE DE CAS - LE DAB

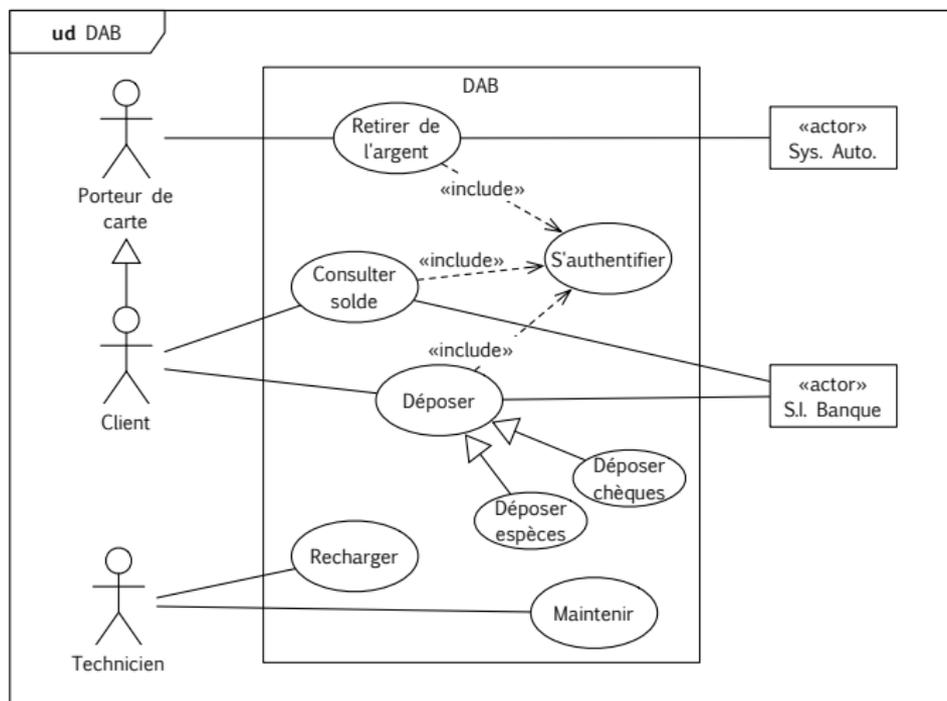


DIAGRAMME DE SÉQUENCE - RETOUR SUR L'ÉTUDE DE CAS - LE DAB

- Travail à effectuer :
 - décrire les cas d'utilisation **S'authentifier** et **Retirer de l'argent** à l'aide de diagrammes de séquence



Chenu, Emmanuel *Cours - Génie Logiciel orienté Objet*, ESISAR