

Analyse et Conception avec UML

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

1

UML au travail

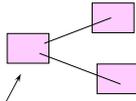
- L'université ESU (Pennsylvanie) désire automatiser son système d'inscription
 - Le chef du service des inscriptions établit le programme des cours pour un semestre
 - Un cours peut être offert plusieurs fois
 - Les étudiants doivent sélectionner 4 cours primaires et 2 cours secondaires
 - Dès qu'un étudiant s'est inscrit pour un semestre, le système de facturation est notifié
 - Les étudiants peuvent utiliser le système pour modifier leurs choix pendant une certaine période de temps après leur inscription
 - Les enseignants utilisent le système pour consulter leur emploi du temps (tableau d'activités en fonction des cours qui tournent)
 - Les utilisateurs du système d'inscription reçoivent des mots de passe qui sont nécessaire à la procédure d'identification

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

2

Quatre objectifs

- ★ Se comprendre 
- ① Représenter le système 
- ② Exprimer le service rendu 
- ③ Décrire la manière dont le système est perçu 

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

3

Les moyens

- Utilisation d'un dictionnaire du domaine
- Les acteurs UML
- Les *use-cases* UML

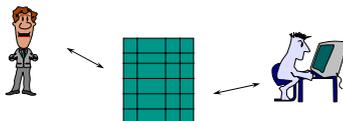
05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

4

Intérêt du dictionnaire

- Outil de dialogue
- Informel, évolutif, simple à réaliser
- Etablir et figer la terminologie
 - Permet de figer la terminologie du domaine d'application.
 - Constitue le point d'entrée et le référentiel initial de l'application ou du système.



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

5

Exemple de dictionnaire

- Dictionnaire d'un simulateur de vol

Notion	Définition	Traduit en ...	Nom informatique
Pilotage	Action de piloter un avion en enchaînant des manoeuvres élémentaires	Package	Pilotage
Instrument	Organe d'interaction entre le pilote et l'avion ou entre l'avion et le pilote	Classe abstraite	Instrument
Manette des gaz	Instrument qui permet d'agir sur la quantité de carburant injectée dans le moteur	Classe	Manette_gaz
Action	Définition	Traduit en ...	Nom informatique
Mettre les gaz à fond	Action qui permet d'injecter le maximum de carburant pour atteindre la vitesse maximale	Opération	Mettre_a_fond

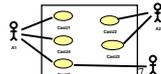
05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

6

Les diagrammes de cas d'utilisation

- Une des notations d'UML (use-cases)
- But :
 - définir le système du point de vue des utilisateurs
 - définir les limites précises du système
- Notation très simple, compréhensible par tous
- Permet de structurer :
 - les besoins (cahier des charges)
 - le reste du développement
 - la progression d'un cycle en spirale
- Les cas d'utilisation sont nommés en utilisant la terminologie décrite dans le dictionnaire



Jean-Marie Favre

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

UML au travail

- L'université ESU (Pennsylvanie) désire automatiser son système d'inscription
 - Le **chef du service des inscriptions** établit le programme des cours pour un semestre
 - Un cours peut être offert plusieurs fois
 - Les **étudiants** doivent sélectionner 4 cours primaires et 2 cours secondaires
 - Dès qu'un étudiant s'est inscrit pour un semestre, le **système de facturation** est notifié
 - Les étudiants peuvent utiliser le système pour modifier leurs choix pendant une certaine période de temps après leur inscription
 - Les **enseignants** utilisent le système pour consulter leur emploi du temps (tableau d'activités en fonction des cours qui tournent)
 - Les utilisateurs du système d'inscription reçoivent des mots de passe qui sont nécessaire à la procédure d'identification

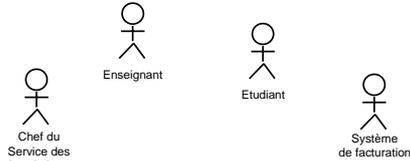
05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

8

Définir le périmètre du SI : Acteurs

- Définir les acteurs **externes**
 - physiques et logiques
 - rôle et entité concrète
- « Un acteur est une personne ou une chose qui va interagir avec le système »



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

9

Acteurs



- Un Acteur =
 - élément externe qui interagit avec le système
 - rôle qu'un utilisateur joue par rapport au système
ex: un enseignant, un guichetier
- Une même personne peut jouer plusieurs rôles
ex: Marie est enseignante et étudiante
Maurice est directeur mais peut faire le guichetier
- Plusieurs personnes peuvent jouer un même rôle
ex: Paul et Pierre sont deux clients
- Un acteur n'est pas forcément un être humain
ex: un distributeur de billet peut être vu comme un acteur; un gestionnaire de mot de passes

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

10

Description des acteurs



- Pour chaque acteur :
 - choisir un identificateur représentatif de son rôle
 - donner une brève description textuelle



Guichetier

Un guichetier est un employé de la banque chargé de faire l'interface entre le système informatique et les clients qu'il reçoit au comptoir. Le guichetier peut réaliser les opérations courantes : création d'un compte, dépôt et retrait d'argent, etc.

Jean-Marie Favre

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

11

UML au travail

- L'université ESU (Pennsylvanie) désire automatiser son système d'inscription
 - Le **chef du service des inscriptions** établit le **programme des cours** pour un semestre
 - Un cours peut être offert plusieurs fois
 - Les **étudiants** doivent **sélectionner** 4 **cours** primaires et 2 cours secondaires
 - Dès qu'un étudiant s'est inscrit pour un semestre, le **système de facturation** est notifié et la facture éditée, prête à être envoyée
 - Les étudiants peuvent utiliser le système pour modifier leurs choix pendant une certaine période de temps après leur inscription
 - Les **enseignants** utilisent le système pour **consulter leur emploi** du temps (tableau d'activités en fonction des cours qui tournent)
 - Les utilisateurs du système d'inscription **reçoivent des mots** de passe qui sont nécessaire à la procédure d'identification

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

12

Cas d'utilisation

- Un cas d'utilisation est un motif de comportement intrinsèque au système
 - Chaque cas d'utilisation est une séquence de transactions connectées, effectuées par un dialogue entre un acteur et le système
- Identification des besoins des acteurs
 - Chef du service des inscriptions – maintenir le programme des études
 - Enseignant – demander un tableau de service
 - Etudiant – s'établir un emploi du temps
 - Système de facturation – recevoir les informations de facturation du système d'inscription



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

13

Le système

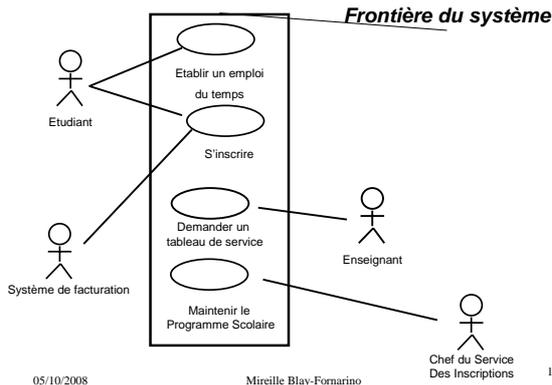
- Le système est un ensemble de cas d'utilisation
- Le système contient :
 - les cas d'utilisation,
 - mais pas les acteurs.
- Un modèle de cas d'utilisation permet de définir :
 - les fonctions essentielles du système,
 - les limites du système,
 - le système par rapport à son environnement.

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

14

Diagrammes de cas d'utilisation

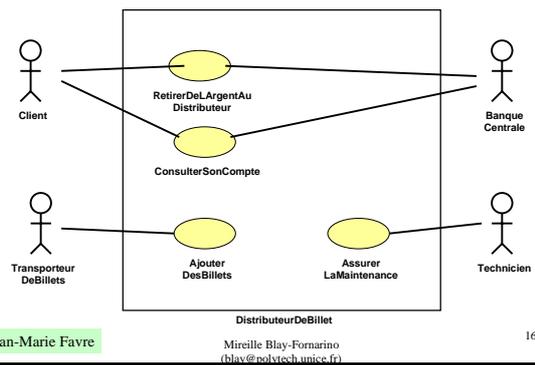


05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

15

Exemple de diagrammes de cas d'utilisation



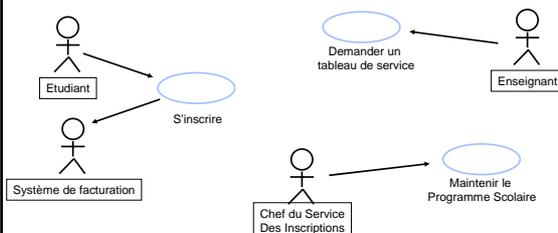
Jean-Marie Favre

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

16

Diagramme des cas d'utilisation

- Objectif visualiser les relations entre acteurs et cas d'utilisation



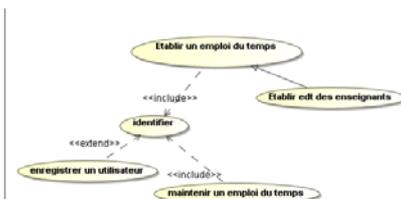
05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

17

Relations entre use cases Uses et Extends

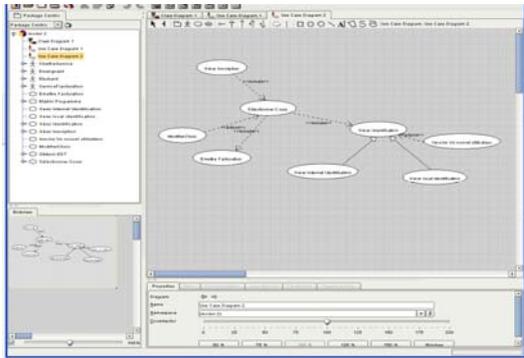
- Au fur et à mesure que les cas d'utilisation sont documentés, des relations peuvent apparaître
 - Une relation **includes** utilisation systématique
 - Une relation **extends** dénote un comportement optionnel :



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.umice.fr)

Use case et Généralisation



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

19

Identifier les Cas d'utilisation

- 1 cas d'utilisation
 - ensemble d'actions fournissant un résultat observable pour un acteur particulier.
 - Une instance de l'utilisation du système
 - Quoi pas Comment (ni IHM, ni erreur)
 - Décrire exhaustivement les exigences fonctionnelles
- Pour chaque cas d'utilisation candidat
 - vérifier qu'il fournit une valeur ajoutée notable
 - contrôler qu'un événement externe au système déclenche son exécution
 - décrire le cas d'utilisation succinctement
- Acteur principal
 - celui pour lequel le cas d'utilisation produit la plus value métier

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

20

Identifier les cas d'utilisation

- **Attention**
 - à ne pas descendre trop bas
 - un cas d'utilisation est une séquence d'actions du système concrétisant une intention de l'acteur
 - à ne pas réinventer la décomposition fonctionnelle
 - limiter à **20** le nombre de cas d'utilisations
 - ne pas mélanger IHM et fonctionnel

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

21

Documenter les cas d'utilisation

- Pour chaque cas d'utilisation, documenter le flux des événements
 - Le faire du point de vue de l'acteur
- Détailler ce que le système doit fournir à l'acteur quand le cas d'utilisation est exécuté
- Contenu typique
 - Description des enchaînements
 - Flux normal des événements
 - Exceptions dans le flux des événements
 - Pré-conditions et post-conditions
 - Compléter avec des diagrammes dynamiques simples (vus plus loin)
 - diagrammes d'activité global et scénarios
 - Besoins d'IHM
 - Contraintes non fonctionnelles
 - servira à évaluer les contraintes techniques

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

22

Flux des événements : Maintenir le Programme scolaire

- Ce cas d'utilisation démarre lorsque le Chef du service des inscriptions s'identifie auprès du système. Le système vérifie la validité de son mot de passe (E-1) et demande à l'acteur de sélectionner le semestre courant ou un semestre futur (E-2). L'acteur entre le semestre choisi. Le système offre alors le choix entre 4 activités : AJOUT, SUPPRESSION, VERIFICATION, SORTIE.
- Si l'activité sélectionnée est AJOUTER, le flux S-1: Ajouter un Cours est exécuté.
- Si l'activité sélectionnée est SUPPRESSION, le flux S-2: Supprimer un Cours est exécutée.
- Si l'activité sélectionnée est VERIFICATION, le flux S-3: Vérifier le Programme.
- Si l'activité sélectionnée est QUITTER, le cas d'utilisation s'arrête.
- ...

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

23

Exemple de description détaillée d'un CU


Retirer
DeL'Argent
AuDistributeur

Précondition :

Le distributeur contient des billets, il est en attente d'une opération, il n'est ni en panne, ni en maintenance

Début : lorsqu'un client introduit sa carte bancaire dans le distributeur.

Fin : lorsque la carte bancaire et les billets sont sortis.

Postcondition :

Si de l'argent a pu être retiré la somme d'argent sur le compte est égale à la somme d'argent qu'il y avait avant, moins le montant du retrait. Sinon la somme d'argent sur le compte est la même qu'avant.

Jean-Marie Favre

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

24

Exemple de description détaillée d'un CU



Retirer DeL'Argent AuDistributeur

Déroulement normal :

- (1) le *client* introduit sa carte bancaire
- (2) le *système* lit la carte et vérifie si la carte est valide
- (3) le *système* demande au client de taper son code
- (4) le *client* tape son code confidentiel
- (5) le *système* vérifie que le code correspond à la carte
- (6) le *client* choisi une opération de retrait
- (7) le *système* demande le montant à retirer
- ...

Variantes :

- (A) *Carte invalide* : au cours de l'étape (2) si la carte est jugée invalide, le système affiche un message d'erreur, rejète la carte et le cas d'utilisation se termine.
- (B) *Code erroné* : au cours de l'étape (5) ...

Jean-Marie Favre

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

25

Exemple de description détaillée d'un CU



Retirer DeL'Argent AuDistributeur

Contraintes non fonctionnelles :

- (A) *Performance* : le système doit réagir dans un délai inférieur à 4 secondes, quelque soit l'action de l'utilisateur.
- (B) *Résistance aux pannes* : si une coupure de courant ou une autre défaillance survient au cours du cas d'utilisation, la transaction sera annulée, l'argent ne sera pas distribué. Le système doit pouvoir redémarrer automatiquement dans un état cohérent et sans intervention humaine.
- (C) *Résistance à la charge* : le système doit pouvoir gérer plus de 1000 retraits d'argent simultanément
- ...

Jean-Marie Favre

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

26

Cas d'utilisation : résumé

- Se servir des Cas d'Utilisation UML pour identifier les exigences fonctionnelles.

Construct	Description	Syntax
use case	A sequence of actions, including variants, that a system (or other entity) can perform, interacting with actors of the system.	
actor	A coherent set of roles that users of use cases play when interacting with these use cases.	
system boundary	Represents the boundary between the physical system and the actors who interact with the physical system.	

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

27

Cas d'utilisation : résumé

Construct	Description	Syntax
association	The participation of an actor in a use case. i.e., instance of an actor and instances of a use case communicate with each other.	
extend	A relationship from an extension use case to a base use case, specifying how the behavior for the extension use case can be inserted into the behavior defined for the base use case.	
generalization	A taxonomic relationship between a more general use case and a more specific use case.	

Construct	Description	Syntax
include	An relationship from a base use case to an inclusion use case, specifying how the behavior for the inclusion use case is inserted into the behavior defined for the base use case.	

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

28

Réalisation des Cas d'Utilisation « Diagrammes d'interactions »

- Le cas d'utilisation présente une vue externe du système
- Les **diagrammes d'interaction** montrent comment des sociétés d'objets peuvent collaborer pour réaliser les cas d'utilisation
- **Interaction**: a collection of communications between instances, including all ways to affect instances, like operation invocation, as well as creation and destruction of instances
- The communications are partially ordered (in time)

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

29

Les diagrammes d'interactions

- 2 notations :
 - les diagrammes de séquences
 - les diagrammes de collaboration
 - Diagrammes de séquences :
 - mise en évidence des objets et des messages échangés
 - correspondent à la structure temporelle
 - Diagrammes de collaboration :
 - établissement des responsabilités - qui coopère avec qui pour l'obtention d'un but
 - correspondent à la structure spatiale
- Nota : les 2 notations sont duales, et souvent on se contente de l'une ou de l'autre !

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

30

Concepts fournis par les diagrammes de séquence

- **Objet :**
 - objet dédié : une instance particulière d'une classe
 - objet anonyme : n'importe quelle instance d'une classe
- **Stimulus :**
 - une instance de message i.e. représentation de l'échange d'information entre objets
- **Supporte des flots de données et divers types de synchronisation**
- **Déroulement temporelle :**
 - **vertical :** représente la ligne de vie des objets et les périodes d'activité des objets
 - **horizontal :** représente l'enchaînement des stimuli entre 1 objet émetteur et 1 objet récepteur i.e. les flots de contrôle (séquence, répétition, alternative)

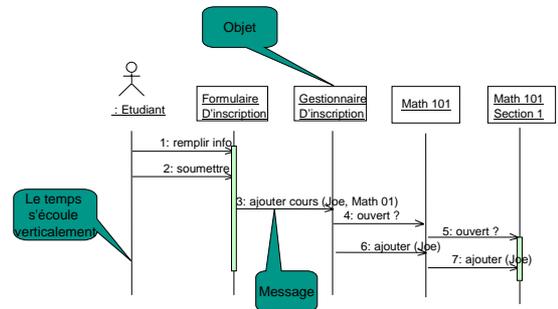
05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

31

Diagramme de Séquence

- Vue temporelle de l'interaction entre objets

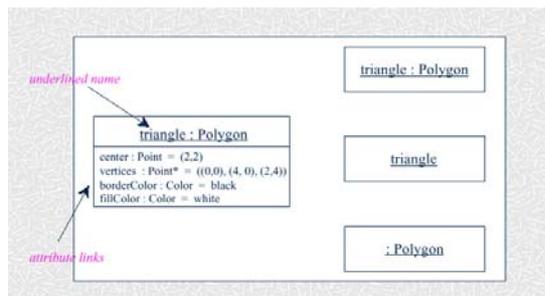


05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

32

Exemple d'instances

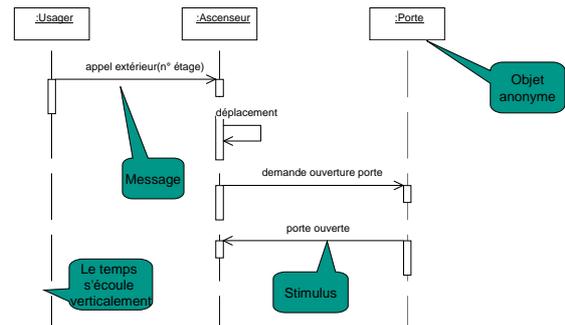


05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

33

Notations pour les diagrammes de séquences

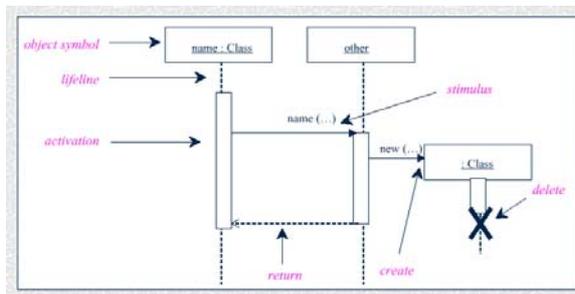


05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

34

Notations pour les diagrammes de séquences



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

35

Activations et envois de messages

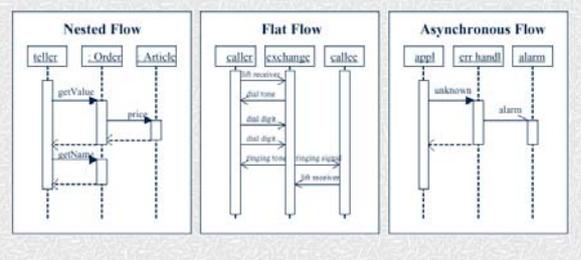
- Chaque réception de message donne lieu à une durée d'activation : le temps de traitement du message
- La durée d'activation de l'émetteur recouvre celle du récepteur
- Type de messages :
 - flot de contrôle à plat :
 - message synchrone
 - message asynchrone
 - flot de contrôle emboîté ou appel de procédure (avec attente implicite du retour)
 - retour d'un appel de procédure, avec ou sans paramètre de retour

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

36

Des communications différentes



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

37

Concepts fournis par les diagrammes de collaboration

- 2 niveaux de représentation :
 - niveau spécification : classes et rôles
 - niveau instance : objets et messages

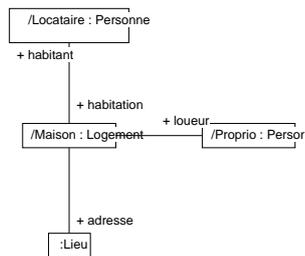
05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

38

Niveau Spécification

- Instance de classe
- Représente un graphe de rôles (ex. /Maison: Logement)



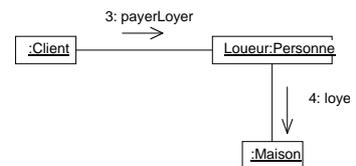
05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

39

Niveau Instance

- Objets = instances de classe
- Stimuli = instances de message
- Liens = instances de relations
- Acteurs
- Messages :
 - data flow
 - contrôle et synchronisation



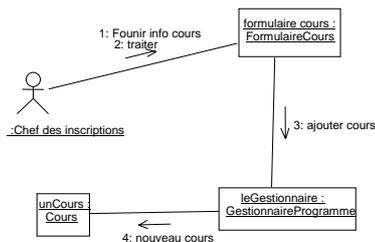
05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

40

Diagramme de Collaboration

- Vue spatiale des objets et de leurs liens

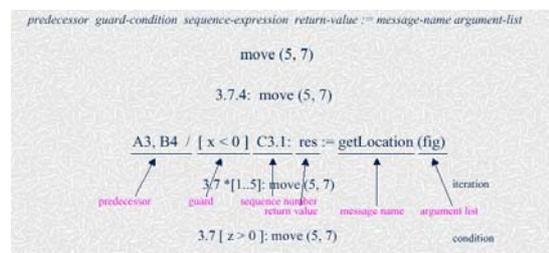


05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

41

Informations sur les flèches

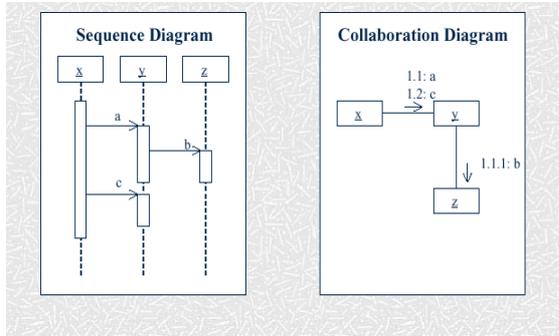


05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

42

Diagrammes de Séquences et Collaboration



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

43

Interactions : Résumé (1)

Construct	Description	Syntax
Instance (object, data value, component instance etc.)	An entity with a unique identity and to which a set of operations can be applied (signals be sent) and which has a state that stores the effects of the operations (the signals).	
Action	A specification of an executable statement. A few different kinds of actions are predefined, e.g. CreateAction, CallAction, DestroyAction, and UninterpretedAction.	textual

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

44

Interactions : Résumé (2)

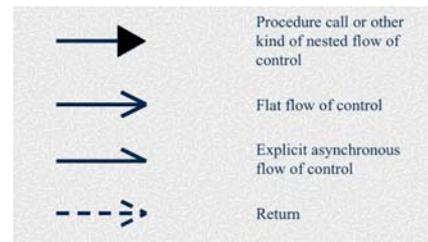
Construct	Description	Syntax
Stimulus	A communication between two instances.	
Operation	A declaration of a service that can be requested from an instance to effect behavior.	textual
Signal	A specification of an asynchronous stimulus communicated between instances.	
Construct	Description	Syntax
Link	A connection between instances.	
Attribute Link	A named slot in an instance, which holds the value of an attribute.	textual

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

45

Interactions : Résumé (3)



05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

46

Importance des diagrammes d'interactions

Essentiel :

- un scénario = un diagramme de séquence
 - un scénario = 1 et 1 seule suite d'actions (seule la séquence est autorisée)
 - les scénarios doivent décrire TOUS les cas de traitements de TOUS les événements possibles
- | | |
|---|--|
| <ul style="list-style-type: none"> ■ Diagrammes de collaboration <ul style="list-style-type: none"> - identification des responsabilités - identification des rôles | <ul style="list-style-type: none"> ■ Diagrammes de séquence <ul style="list-style-type: none"> - identification des objets, et donc des classes potentielles - identification des échanges entre objets, issus des scénarios |
|---|--|

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

47

Contraintes temporelles

- Flèche oblique vers le bas : traduit le fait que l'a transmission du contrôle et des données prend un certain temps
- Une contrainte temporelle peut s'exprimer :
 - sur une durée d'activation par une annotation ou un texte entre { }
 - sur les instants d'émission/réception de messages, par une annotation ou un texte entre { }
 - sur les différences entre temps instants d'émission/réception de messages
- Une contrainte temporelle est toujours une expression relationnelle !

05/10/2008

Mireille Blay-Fornarino
(blay@polytech.unice.fr)

48

Interactions : Quand les utiliser

- Pour spécifier comment les instances interagissent entre elles.
- Pour identifier les interfaces des classes.
- Pour montrer l'ordre explicite entre les stimuli
 - when modeling real- time
- Use collaboration diagrams
 - when structure is important
 - to concentrate on the effects on the instances