Glossaire

Acteur

Un acteur représente un utilisateur d'un cas d'utilisation dans son rôle vis-à-vis du système. Le nom de l'acteur est alors celui du rôle.

Deux catégories d'acteurs doivent être distinguées :

- Les acteurs primaires, pour lesquels l'objectif du cas d'utilisation est essentiel et constitue un objectif de l'acteur.
- Les acteurs secondaires, pour lesquels l'objectif du cas d'utilisation n'est pas essentiel bien qu'ils interagissent avec lui.

Activité

Une activité est une série d'actions. Une action consiste à affecter une valeur à un attribut, à créer ou à détruire un objet, à effectuer une opération, à invoquer une méthode d'un autre objet ou de l'objet lui-même, etc.

Activité composée

Le contenu d'une activité composée est formé d'autres activités.

Agrégation ou composition faible

L'agrégation est l'association qui relie un objet composé à ses composants. Elle est faible pour deux raisons : les composants peuvent appartenir à d'autres objets composés et la destruction de l'objet composé n'entraîne pas la destruction de ses composants.

Alternative

Dans un diagramme de séquence, l'alternative est l'un des opérateurs d'un fragment combiné. Elle est associée à une condition. Si la condition est vérifiée, le contenu du cadre est exécuté.

Dans un diagramme d'activités, l'alternative sert à sélectionner l'activité suivante. Les branches de l'alternative sont dotées de conditions de garde qui s'excluent.

Artefact

Un artefact est la forme physique d'un élément logiciel. Un fichier exécutable, une bibliothèque partagée ou un script sont des exemples de formes physiques d'un élément logiciel.

Association entre objets

Une association entre objets est un ensemble de liens entre les instances de deux ou plusieurs classes. Elle est décrite dans le diagramme des classes.

Association réflexive

Une association réflexive relie les instances d'une classe entre elles.

Attribut calculé

La valeur d'un attribut calculé est donnée par une fonction basée sur la valeur d'autres attributs.

Attribut de classe

Un attribut de classe est lié à la classe elle-même, et non à chaque instance. Un tel attribut est partagé par l'ensemble des instances de la classe.

Boucle

Dans un diagramme de séquence, la boucle est l'un des opérateurs d'un fragment combiné. Elle consiste en une exécution répétée du contenu du cadre tant que la condition de fin n'est pas remplie ou que le nombre maximal de répétitions n'est pas atteint.

Cadre d'interaction

Un cadre d'interaction décrit, à l'aide d'un diagramme de séquence, une partie de la dynamique d'un système. Les cadres d'interaction ont été introduits dans un objectif de description modulaire de la dynamique globale d'un système.

Cardinalité minimale ou cardinalité maximale

Une cardinalité est fixée à une extrémité d'une association. Une cardinalité minimale, respectivement maximale, permet de fixer le nombre minimal, respectivement maximal, d'instances auxquelles une instance de la classe située à l'autre extrémité de l'association est reliée.

Cas d'utilisation

Un cas d'utilisation décrit les interactions entre un utilisateur et le système.

Dans un cas d'utilisation avec objectif de l'utilisateur, cette suite d'interactions est liée à un objectif fonctionnel de l'utilisateur.

Dans un cas d'utilisation de sous-fonction, cette suite d'interactions est destinée à être incluse dans un autre cas d'utilisation.

Classe

Une classe est constituée par un ensemble d'objets similaires possédant les mêmes attributs et méthodes. La représentation commune à ces objets est définie au niveau de la classe.

Une classe concrète définit un modèle complet. Elle possède des instances directes.

Une classe abstraite définit un modèle abstrait. Elle ne possède pas d'instances directes. Une telle classe sert à factoriser, en tant que surclasse, des attributs et des méthodes communes de plusieurs classes concrètes.

Une classe-association est à la fois une association et une classe dont les instances sont les occurrences de l'association. Ainsi, ces occurrences peuvent être dotées d'attributs ou d'opérations.

Une classe template est un modèle de classes relié par une relation de liaison (binding) à ses classes instanciées.

Composant

Un composant est une unité logicielle offrant des services au travers d'une ou de plusieurs interfaces. C'est une boîte noire dont le contenu n'intéresse pas ses clients.

Composition

La composition (ou composition forte) est l'association qui relie un objet à ses composants. Elle est forte pour deux raisons : les composants ne peuvent pas appartenir à d'autres objets composés et la destruction de l'objet composé entraîne la destruction de ses composants.

Condition de garde

Une condition de garde est utilisée dans les diagrammes de communication, d'états-

transitions et d'activités. Elle constitue une condition pour respectivement envoyer le message, franchir la transition ou enchaîner les activités.

Contrainte sur la relation d'héritage

Il existe quatre contraintes sur la relation d'héritage entre une surclasse et ses sous-classes :

- `{incomplete}` : l'ensemble des sous-classes est incomplet et ne couvre pas la surclasse, c'est-à-dire que l'ensemble des instances des sous-classes est un sous-ensemble de l'ensemble des instances de la surclasse.
- `{complete}` : l'ensemble des sous-classes est complet et couvre la surclasse.
- `{disjoint}` : les sous-classes n'ont aucune instance en commun.
- `{overlapping}` : les sous-classes peuvent avoir une ou plusieurs instances en commun.

Couloir

Un couloir regroupe toutes les activités dont un même objet est le responsable.

Cycle de vie

Le cycle de vie d'un objet est l'ensemble de ses états et des transitions les reliant.

Diagramme d'activités

Ce diagramme décrit les activités d'un ou de plusieurs objets ainsi que leurs enchaînements.

Diagramme de cas d'utilisation

Ce diagramme décrit l'ensemble (ou un sous-ensemble) des cas d'utilisation et des acteurs d'un système ainsi que les associations les reliant.

Diagramme de classes

Ce diagramme décrit l'ensemble (ou un sous-ensemble) des classes et interfaces d'un système ainsi que les associations qui les relient.

Diagramme de communication

Ce diagramme décrit les interactions entre un ensemble d'objets en montrant, de façon spatiale, les envois de message intervenant entre eux.

Diagramme de déploiement

Ce diagramme décrit l'architecture matérielle d'un système.

Diagramme de composants

Ce diagramme montre la structuration en composants logiciels d'un système.

Diagramme d'états-transitions

Ce diagramme illustre l'ensemble des états du cycle de vie d'un objet séparés par des transitions.

Diagramme d'objets

Ce diagramme montre, à un moment donné, les instances créées et leurs liens lorsque le système est actif.

Diagramme de paquetage

Ce diagramme est un regroupement d'éléments de modélisation.

Diagramme de profil

Ce diagramme regroupe les éléments d'extension du métamodèle comme les stéréotypes, les tagged values, les contraintes, etc.

Diagramme de séquence

Ce diagramme décrit les interactions entre un ensemble d'objets en montrant, de façon séquentielle, les envois de message qui interviennent entre eux.

Diagramme de timing

Le diagramme de timing montre les changements d'état d'un objet en fonction du temps.

Diagramme de vue d'ensemble des interactions

Le diagramme de vue d'ensemble des interactions est un diagramme d'activités où chaque activité peut être décrite par un diagramme de séquence.

Encapsulation

L'encapsulation consiste à masquer la structure et le comportement internes et propres au fonctionnement de l'objet. Ce masquage peut être complet (encapsulation privée), ne pas s'appliquer aux sous-classes (encapsulation protégée) ou ne pas s'appliquer aux classes du même paquetage (encapsulation de paquetage).

Enchaînement d'activités

Un enchaînement d'activités est un lien depuis une activité d'origine vers une activité de destination. Il est franchi lorsque l'activité d'origine est terminée.

Envoi de message

Voir *Message*.

État

L'état d'un objet correspond à un moment de son cycle de vie. Pendant qu'il se trouve dans un état, un objet peut réaliser une activité ou attendre un signal provenant d'autres objets.

Fragment combiné

Un fragment combiné est une partie du diagramme de séquence associée à un opérateur qui en détermine la modalité d'exécution. Les principales modalités sont l'option, l'alternative et la boucle.

Généralisation

La généralisation est la relation qui lie une sous-classe à sa surclasse (ou à l'une des surclasses en cas d'héritage multiple).

La généralisation s'applique également aux cas d'utilisation.

Granularité

La granularité d'un objet représente sa taille. Un objet de petite taille est dit de granularité fine ou de petit grain. Un objet volumineux est dit de granularité importante ou de gros grain.

Héritage

L'héritage est la propriété qui fait bénéficier une sous-classe de la structure et du

comportement de sa surclasse.

L'héritage est multiple quand une sous-classe possède plusieurs surclasses.

Instance

Une instance d'une classe est un élément de l'ensemble des objets de cette classe.

Interface

Une interface est une classe abstraite ne contenant que des signatures de méthodes. La signature d'une méthode est composée de son nom et de ses paramètres.

Une interface fournie décrit les services offerts par un composant. Une interface requise décrit les services qu'un composant attend d'un autre composant dont il est le client.

Ligne de vie

Au sein d'un diagramme de séquence, une ligne de vie montre les actions et réactions d'une instance, ainsi que les périodes pendant lesquelles elle est active.

MDA

MDA (*Model-Driven Architecture* ou architecture guidée par les modèles) est une proposition de l'OMG dont l'objectif est la conception de systèmes basée sur la seule modélisation du domaine, indépendamment de la plateforme.

Message

Un message est envoyé à un objet pour l'activer et provoquer l'exécution de la méthode de même nom. Un envoi de message est un appel de méthode.

Un message peut être envoyé de façon asynchrone. Dans ce cas, l'appelant atteint la fin de l'exécution de la méthode de l'objet récepteur avant de continuer son exécution.

Un message peut être aussi envoyé de façon synchrone. Dans ce cas, l'appelant continue son exécution immédiatement après l'envoi du message.

Métamodèle

Un métamodèle est un modèle décrivant l'ensemble des éléments d'un langage. Par exemple, le métamodèle d'UML décrit l'ensemble des éléments d'UML.

Méthode

Une méthode est un ensemble d'instructions prenant des valeurs en entrée et modifiant les valeurs des attributs ou produisant un résultat.

L'ensemble des méthodes d'une classe décrit le comportement des instances de cette classe.

Méthode de classe

Une méthode de classe est liée à la classe elle-même, et non à une instance. Son invocation se fait au travers de la classe, et non de l'une de ses instances.

MOF

Le MOF (*Meta Object Facility*) est une architecture standardisée pour décrire les éléments d'un métamodèle comme le métamodèle d'UML.

Navigation

La navigation d'une association en détermine le sens de parcours.

Nœud

Un nœud est une unité matérielle capable de recevoir et d'exécuter des éléments logiciels.

Objet

Un objet est une entité identifiable du monde réel. Dans le modèle UML, un objet est une instance d'une classe.

Occurrence d'une association

Une occurrence d'une association est un lien entre des instances des classes situées aux extrémités de cette association.

OCL

OCL (*Object Constraint Language* ou langage de contraintes objet) est un langage destiné à exprimer les contraintes au sein d'un diagramme de classes sous forme de conditions logiques.

OMG

L'OMG (*Object Management Group*) est un consortium formé de plus de 800 sociétés et universités qui a pour but de promouvoir les technologies de l'objet.

Paquetage

Un paquetage est un regroupement d'éléments de modélisation : classes, composants, cas d'utilisation, autres paquetages.

Un paquetage template est un modèle de paquetages relié par une relation de liaison (binding) à ses paquetages instanciés.

PIM

Le PIM (*Platform-Independent Model* ou modèle indépendant de la plateforme) est le modèle de conception de l'architecture MDA.

Polymorphisme

Le polymorphisme est la différence de comportement qui existe entre des sous-classes d'une même surclasse pour les méthodes de même nom.

Processus

Un processus est un ensemble d'opérations prenant en entrée des données et produisant de nouvelles données.

Processus Unifié

Le Processus Unifié est un processus de conception et d'évolution de logiciels basé sur UML.

PSM

Le PSM (*Platform-Specific Model* ou modèle spécifique de la plateforme) est le modèle cible de l'architecture MDA.

Qualification

Une association peut être qualifiée à une extrémité afin de réduire à l'autre extrémité la cardinalité maximale. En effet, la valeur du qualificateur est alors prise en compte pour déterminer le nombre de liens.

Relation de réalisation

La réalisation d'une interface, c'est-à-dire l'implantation de ses méthodes, est confiée à une ou plusieurs classes concrètes, sous-classes de l'interface. La relation d'héritage qui existe entre une interface et une sous-classe d'implantation est appelée relation de réalisation.

Cette relation existe également entre une interface et un composant qui implante ses méthodes.

Relation d'extension

La relation d'extension permet d'enrichir un cas d'utilisation par un cas d'utilisation de sous-fonction. Cet enrichissement est optionnel.

Relation d'inclusion

La relation d'inclusion permet d'enrichir un cas d'utilisation par un cas d'utilisation de sous-fonction. Cet enrichissement est obligatoire.

Relation entre paquetages

Il existe trois relations entre paquetages :

- La relation d'importation consiste à amener dans le paquetage de destination un élément du paquetage d'origine. L'élément fait alors partie des éléments visibles du paquetage de destination.
- La relation d'accès consiste à accéder depuis le paquetage de destination à un élément du paquetage d'origine. L'élément ne fait alors pas partie des éléments visibles du paquetage de destination.
- La relation de fusion entre un paquetage d'origine et un paquetage de destination exprime que le contenu du paquetage d'origine est issu de la fusion entre son contenu initial et le contenu du paquetage de destination.

Scénario

Un scénario est une instance d'un cas d'utilisation dont toutes les alternatives ont été fixées.

Spécialisation

La spécialisation est la relation qui lie une surclasse à l'une de ses sous-classes.

La spécialisation s'applique également aux cas d'utilisation.

Stéréotype

Un stéréotype est un mot-clé utilisé pour expliciter la spécialisation d'un élément. Un stéréotype est noté entre guillemets.

Transition

Une transition est un lien orienté entre deux états qui exprime le fait que l'objet a la possibilité de passer de l'état d'origine de la transition à son état de destination.

Type

Le type peut être une classe ou un type standard. Les types standards sont respectivement désignés ainsi :

- `Integer` pour le type des entiers.
- `String` pour le type des chaînes de caractères.
- `Boolean` pour le type des booléens.
- `Real` pour le type des réels.

UML

UML (*Unified Modeling Language* ou langage unifié de modélisation) est un langage graphique destiné à la modélisation de systèmes et de processus.