

SPINOV

COMMENT LIRE LES DIAGRAMMES UML
GUIDE POUR DES NON INFORMATIENS

Auteur	Céline Décosse	CRP Henri Tudor
Date	NOVEMBRE 2002	
Référence	H:\Projets\Projets_Kirchberg\spinov\WP5_Mise en oeuvre Serv Q\WP5.3_Formalisation boites outils\Langage UML\GU_SPINOV_CommentLireLesDiagrammesUML_1.0.doc	
Validation	CRP Henri Tudor	
Diffusion	Toute personne non informaticienne qui est amenée à lire des diagrammes UML.	
Annexes	Plus d'informations sur la lecture des diagrammes d'activité	
	Informations sur l'écriture des diagrammes d'activités	

Contenu	
Comment lire un diagramme de classes ?	2
Comment lire un diagramme d'activité ?.....	5
Comment lire un diagramme d'état ?.....	6

Mises à jour :

- Août 2003 : Comment lire un diagramme de classes : ajout de la notion de rôle dans les diagrammes de classes, d'un exemple.
- Mai 2004 : tableau de légende d'un diagramme d'activité
- Août 2004 : corrections de style.

Présentation de UML

Les diagrammes ont été réalisés avec le langage de modélisation orienté objet UML : Unified Modeling Language. Ce langage a été normalisé par l'Object Management Group.

Les structures de données sont faites avec des Diagrammes de Classes.

Les états sont dessinés avec des Diagrammes d'états.

Comment lire un diagramme de classes (UML 1.4) ?

1.1 Les classes

Un rectangle modélise une classe. Une classe est un concept métier, c'est-à-dire un élément du domaine, qui a des relations avec d'autres éléments du domaine. Par ailleurs une classe peut avoir des attributs, par exemple le Nom de la personne. Attention, une classe ne représente pas un élément, mais une description générique de tous les éléments d'un ensemble. Les éléments décrits par les classes s'appellent objets ou instances, ils ne sont pas représentés sur un diagramme de classes. Par exemple, on dit que « Monsieur Jean » est une instance de la classe "Personne".

1.2 Les relations entre classes

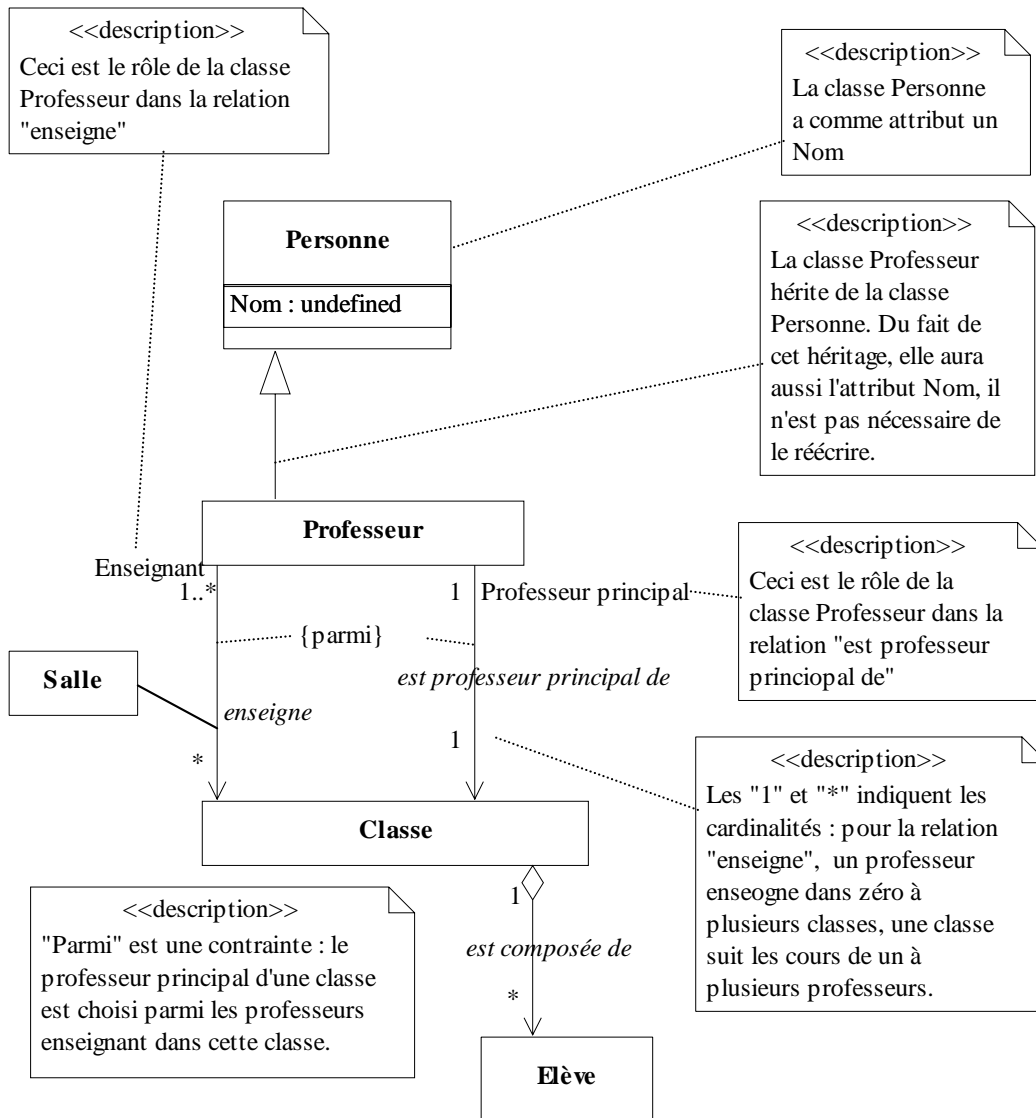
Il y a plusieurs sortes de relations entre les classes.

- Flèches pleines de blanc : elles illustrent l'héritage. Se lit "est une sorte de ". Exemple : un Professeur est une sorte de Personne. On dit que la classe "Professeur" est une sous-classe de la classe "Personne". Les sous-classes ont toutes les propriétés de leurs super-classes, plus d'autres plus spécifiques. Les classes qui servent seulement à factoriser des propriétés et à faire de l'héritage, mais n'ont pas d'instances directes, ont leur nom en italiques non gras.
- Trait simple (avec ou sans nom dessus) : c'est une association. Une classe est associée à une autre classe pour une raison précise. On peut mettre cette raison en toutes lettres. La flèche vide indique simplement un sens d'utilisation de la relation. Si un nom est ajouté à la fin de la flèche vide, il s'agit du rôle que joue la classe dans la relation. Par exemple, si un professeur enseigne dans des classes d'élèves, on aura une relation entre la classe « Professeur » et la classe « Classe ».
- Losange blanc : ajouté au bout d'une association, il signifie que cette association est une relation d'agrégation. Une instance de la classe proche de ce losange "contient" une ou plusieurs instances de la classe en relation.
- Losange plein : il indique une relation de composition, la composition est une agrégation forte dans laquelle la destruction du composé entraîne la destruction des composants.
- Cardinalités : caractérise les associations. Comme les classes sont des descriptions génériques d'instances (aussi appelés "objets"), on veut pouvoir représenter sur le schéma le fait qu'une instance donnée puisse être en relation avec un certain nombre d'instances d'une autre classe. Un élément donné de la classe A peut être associé à x éléments de la classe B. Dans la modélisation UML, on remplace ce x par :
 - * pour signifier "0 à plusieurs". Exemple : une catégorie peut être améliorée par zéro à plusieurs formations.
 - 1..n pour signifier "de un à plusieurs".
 - rien ou 1 pour signifier "exactement 1"

- Traits pointillés : pour lier des commentaires à un élément. Il n'y a pas de sémantique précise pour les pointillés, on les utilise quand on a besoin de mettre ensemble des éléments et que les associations et l'héritage ne suffisent pas. Ils servent aussi à lier des classes d'association aux associations qu'elles concernent (mais cela sort mal sur le graphique ci-dessous). Par ailleurs, on peut mettre des contraintes entre relations à l'aide de pointillés.
- Une classe d'association est une classe qui existe uniquement parce que une association entre deux classes est porteuse d'information. Cette information est mise dans une classe d'association, sous forme d'attributs de la classe d'association.
- "Post-it" = commentaires. Exemple : le titre des diagrammes a été mis sous cette forme ci-dessous.

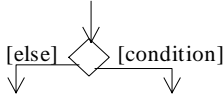
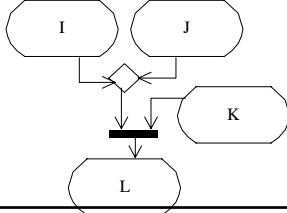
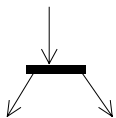
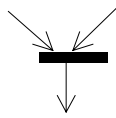

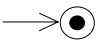
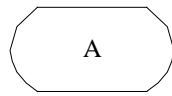

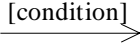
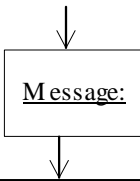
1.3 Illustration

- Dans l'illustration ci-dessous, un professeur est professeur principal d'une des classes dans lesquelles il enseigne.



Vous pouvez consulter cette adresse pour voir d'autres exemples : <http://uml.free.fr/cours/i-p14.html>

Comment lire un diagramme d'activité (UML 1.4) ?

<p>Choix / Condition (XOR split) </p> 	<p>Rassemblement de flèches (XOR join) </p> 	<p>Swimlanes (rôles)</p> <table border="1" style="width: 100%; height: 60px; border-collapse: collapse;"> <tr> <td style="width: 50%; padding: 5px;">Développeur:</td> <td style="width: 50%; padding: 5px;">Big chef:</td> </tr> </table>	Développeur:	Big chef:
Développeur:	Big chef:			
<p>Synchronisation (AND split)</p> 	<p>Synchronisation (AND join)</p> 			
<p>Début du processus</p> 	<p>Fin du processus</p> 	<p>Activité</p> 		
<p>Transition</p> 	<p>Transition gardée</p> 	<p>Flux d'information (messages, produits, docs)</p> 		

Les diagrammes d'activité représentent les processus. Un processus est un enchaînement d'activités. Une activité peut souvent être résumée par un verbe. Une activité est représentée par un rectangle aux coins très arrondis. Les flèches représentent des transitions d'une activité à une autre. Ces transitions ont lieu lorsque les activités sont finies. Parfois, ces transitions n'ont lieu que si, en plus de la terminaison d'une activité, une certaine condition est remplie ; cette condition

est alors mise entre crochets. Les activités peuvent utiliser ou générer des informations qui peuvent être représentées par des rectangles. Ces informations peuvent être concrètes (document, papier), informatiques (ce sera indiqué sur le schéma) ou verbales. Elles sont attachées à des activités par des flèches en pointillés.

Les disques pointent sur le début du processus, les disques encadrés signifient la fin du processus. S'il n'y en a pas, le processus est à lire de haut en bas.

Comment lire un diagramme d'état (UML 1.4) ?

Un diagramme d'état montre les états potentiels d'un objet, et les transitions d'un état à un autre. Une transition est déclenchée par un événement, il peut y avoir une condition de garde, et une action associée à la transition. Un état peut avoir des sous-états. Un état a une durée, une transition est instantanée.

Les diagrammes d'activité concernent des enchaînements d'actions réalisées par des agents, qui peuvent avoir une influence sur de nombreux éléments du système. Les diagrammes d'état ne concernent qu'une partie du système (un objet, un ensemble d'objets), et montre comment les événements (le plus souvent générés lors des activités des diagrammes d'activité) font changer l'objet d'état.

!Obsolète : Le diagramme d'activité est un cas particulier du diagramme d'état, où les états seraient des activités. Pour les diagrammes d'activité, les transitions ont lieu automatiquement lorsque l'activité est terminée alors que pour les digrammes d'états, elles doivent être explicites.