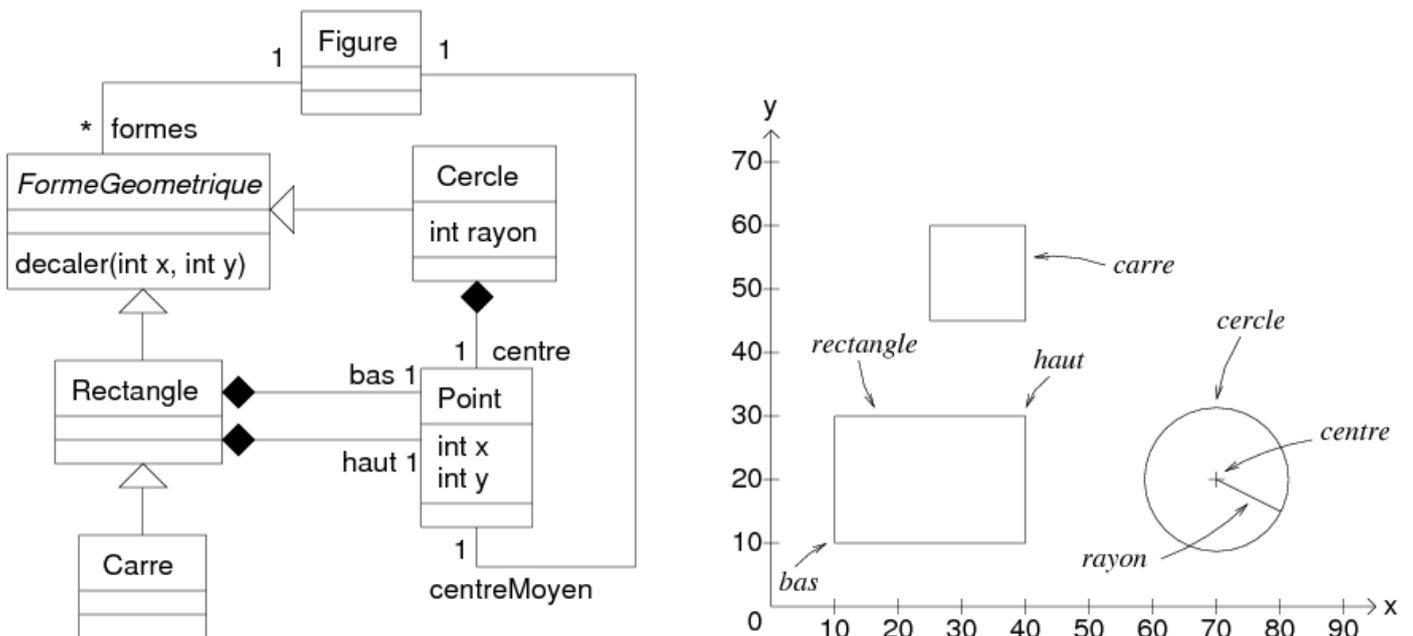


# TD OCL et méta-modélisation

Programmation Générative – partie Ingénierie des Modèles

Master TIIL-A 2<sup>ème</sup> année – UBO – Eric Cariou

## Exercice 1



La figure ci-dessus représente, dans sa partie gauche, un diagramme de classes UML spécifiant des figures qui contiennent des formes géométriques dans un repère géométrique dont les abscisses et ordonnées sont uniquement positives. La partie droite de la figure donne des exemples de ces formes géométriques.

Ces formes sont le cercle, le rectangle et le carré. Un cercle est spécifié par un point central et un rayon. Un carré et un rectangle par deux points ; le point bas étant le coin en bas à gauche et le point haut le coin en haut à droite. Un point est défini par un couple de coordonnées entières  $x$  et  $y$  selon l'axe des abscisses et des ordonnées. Par exemple, le rectangle de la figure a pour point bas celui de valeur (10, 10) et pour point haut celui de valeur (30, 40).

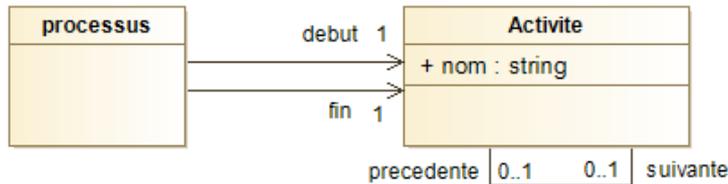
**Question 1** : donner tous les invariants OCL requis pour compléter le diagramme de classes afin de définir des formes géométriques valides pour le repère défini.

**Question 2** : spécifier la pré et la post-condition de l'opération `decaler(int x, int y)` définie dans la classe *FormeGeometrique*. Cette opération décale une forme géométrique des valeurs précisées en  $x$  et en  $y$ . Au besoin, vous pouvez spécifier cette opération dans chacune des classes concrètes *Cercle*, *Rectangle* et *Carre*.

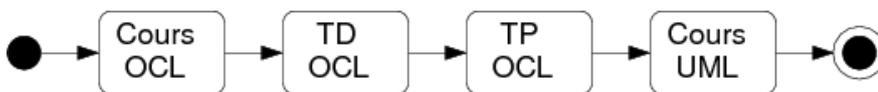
**Question 3** : une figure référence un point particulier, le centre moyen. Il s'agit de la moyenne des centres des cercles se trouvant dans la figure. Spécifier l'invariant associé à ce centre moyen et définissant sa valeur. Si aucun cercle n'est présent dans la figure, ce centre moyen a pour valeur (0,0).

## Exercice 2 : spécification d'une séquence d'activités

Le diagramme de classes UML ci-dessous permet de spécifier un processus composé d'une séquence ordonnée d'activités. Un processus possède une activité de début ainsi qu'une de fin et ensuite la séquence est définie par les relations précédente / suivante entre les activités. Définissez les invariants OCL pour que la séquence soit bien formée.



Voici un exemple de séquence bien formée :



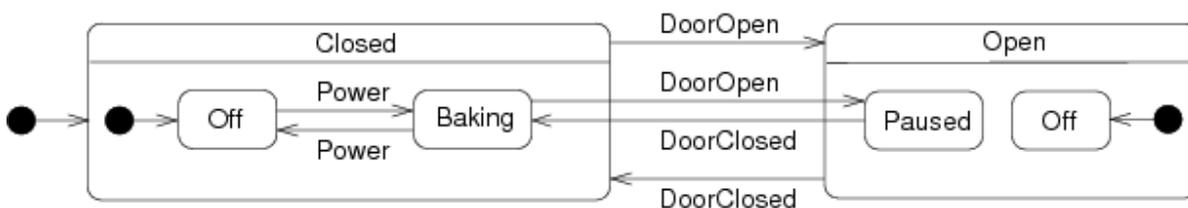
## Exercice 3 : méta-modèle Ecore du diagramme de machines à états UML

Nous allons définir le méta-modèle Ecore de la structure des machines à états UML (en les simplifiant) :

- Un état est défini par un nom.
- Un composite est un état particulier qui contient des sous-états. Tout composite possède un état initial. Il n'y a pas de limite dans la hiérarchie de composition.
- Une machine à états possède également un état initial.
- Une transition relie deux états en étant associée à un événement. Il n'existe pas deux transitions partant du même état et associées au même événement.
- On pourra ajouter à un état des opérations métiers (en do, entry et exit) définies par leur signature. De même, on pourra rajouter une opération métier à une transition.
- On pourra également rajouter une garde à un état : ce sera une opération métier qui retourne un booléen.
- Si on exécute une machine à états, il faut être capable de savoir quels sont les états actifs. Si un état « racine » est actif alors le composite qui le contient le sera également et ainsi de suite de manière récursive.

La contrainte d'un méta-modèle en Ecore est qu'il doit définir un élément racine à partir duquel tous les méta-éléments concrets sont accessibles directement ou transitivement par des relations de composition.

Voici pour rappel un exemple de machine à états hiérarchique (mais sans opérations métiers ou gardes) qui spécifie le comportement d'un four à micro-ondes.



**Question 1 :** définissez le méta-modèle Ecore définissant la structure d'une machine à états UML

**Question 2 :** spécifier les invariants OCL requis pour compléter le méta-modèle

**Question 3 :** définissez en OCL la cohérence d'activation des états pendant l'exécution d'une machine à états