

OCLE 2 : Quelques contraintes simples (sans navigation de collections) dans le méta-modèle UML 2

Module *Ingénierie des modèles*

Définissez en OCL les contraintes et les opérations suivantes. Elles font référence aux diagrammes du méta-modèle UML 2.0 dont les noms vous sont rappelés. Notez que l'absence de valeur pour une propriété `p` se contrôle par `p->isEmpty()`.

Namespace Diagram

- La visibilité d'un `ElementImport` est public ou privé.
- Un `importedElement` est public ou bien n'a pas de visibilité.
- La requête `getName()` retourne le nom sous lequel l'élément importé est connu : c'est soit l'alias (s'il est valué), soit le nom de l'élément (si l'alias n'est pas valué).
- Si un `NamedElement` n'est pas possédé par un `NamedSpace`, il n'a pas de visibilité.

Multiplicities Diagram

La requête `lowerBound()` retourne la borne inférieure de multiplicité sous forme d'un entier si elle est spécifiée, 1 sinon.

```
context MultiplicityElement
  def : lowerBound() : Integer =
    if lowerValue->isEmpty() then 1 else lowerValue.integerValue()
```

La requête `upperBound()` retourne la borne supérieure de multiplicité sous forme d'un entier (éventuellement * pour un nombre indéfini) si elle est spécifiée, 1 sinon.

```
context MultiplicityElement
  def : upperBound() : UnlimitedNatural =
    if upperValue->isEmpty() then 1 else upperValue.unlimitedValue()
```

- La borne inférieure doit être positive ou nulle.
- La borne supérieure doit être supérieure à la borne inférieure.
- `lower` doit être égal à la borne inférieure.
- `upper` doit être égal à la borne supérieure.
- La requête `isMultivalued()` retourne vrai si la propriété peut prendre plus d'une valeur ; elle ne s'applique que lorsqu'une borne supérieure a été spécifiée.
- La requête `includesCardinality(C : Integer)` retourne vrai si la propriété peut prendre `C` valeurs ; elle ne s'applique que lorsqu'une borne inférieure et une borne supérieure ont été spécifiées.
- La requête `includesMultiplicity(M : MultiplicityElement)` retourne vrai si la multiplicité de l'élément inclut `M`. Vous devez déterminer les conditions d'application.

Operation Diagram

- Une `bodyCondition` ne peut être spécifiée que pour une opération de type requête.

Classes Diagram

- Dans une composition, la multiplicité n'est pas précisée ou n'est pas supérieure à 1.
- Une union dérivée est une propriété dérivée (particulière).
- Une union dérivée est accessible seulement en consultation.
- Définir la valeur dérivée de `isComposite`.
- Définir la valeur dérivée de `default` en utilisant l'opération `stringValue` de `ValueSpecification`.