Meta-Mutation Operators for Ecore Modelling Concepts

0.1 Example of Applying the Systematic Approach

In this example, we illustrate the application of the systematic approach to one of Ecore modelling concept - *EPackage*. Since this particular concept is an EClass, we define the following:

1. Name: EPackage
2. Super-type: ENamedElement
3. EStructuralFeatures (whether direct or inherited): EString nsURI, EString nsPrefix, EClassifier eClassifiers, EPackage eSubpackages, EString name, EAnnotation eAnnotations

In this example, we only show the mutation operators for *nsURI* and *eClassifiers* features with explanation of some preconditions. The mutation operators for other features are not listed because of the limit restrictions and they are quite similar. For example, the features *nsPrefix* and *name* have similar datatype and complexity of the feature *nsURI*. Likewise, the feature *eSubpackages* and *eAnnotations* have identical complexity (e.g. same lowerBound and upperBound) with the feature *eClassifiers*.

The list of features (whether direct or inherited) are given below. For each feature, we define some preconditions that must be met before applying the designed mutation operators.

- **EAttribute[EString nsURI], lowerBound=0, upperBound=1**: this feature can take the following mutation operators defined for *EString* datatype.
  ADD(EString nsURI, EString toAdd): Appends to the value of *nsURI* the value specified by *toAdd*.
  Preconditions
  - $nsURI.isDefined()$ & $toAdd.isDefined()$
  - $toAdd.length \geq 1$

  DEL(EString nsURI, Integer toRemove): Removes randomly a number of *toRemove* characters from *nsURI*.
  Preconditions
  - $nsURI.isDefined()$ & $toRemove.isDefined()$
  - $nsURI.length \geq toRemove \geq 1$

  REP(EString nsURI, EString newValue): Replaces the value of *nsURI* with the value of *newValue*.
  Preconditions

- $nsURI.isDefined()$ & $newValue.isDefined()$
- $nsURI \neq newValue$

- **EAttribute[EString nsPrefix], lowerBound=0, upperBound=1**: this feature can take the following mutation operators defined for *EString* data-type.
  ADD(EString nsPrefix, EString toAdd): Appends to the value of *nsPrefix* the value specified by *toAdd*.
  Preconditions
  - $nsPrefix.isDefined()$ & $toAdd.isDefined()$
  - $toAdd.length \geq 1$

  DEL(EString nsPrefix, Integer toRemove): Removes randomly a number of *toRemove* characters from *nsPrefix*.
  Preconditions
  - $nsPrefix.isDefined()$ & $toRemove.isDefined()$
  - $nsPrefix.length \geq toRemove \geq 1$

  REP(EString nsPrefix, EString newValue): Replaces the value of *nsPrefix* with the value of *newValue*.
  Preconditions
  - $nsPrefix.isDefined()$ & $newValue.isDefined()$
  - $nsPrefix \neq newValue$

- **EReference[EClassifier eClassifiers], lowerBound=0, upperBound=***
  *ADD(EClassifier eClassifiers, Integer index, EClassifier extra):*Inserts *extra* at the specific position in the list *eClassifiers*.
  Preconditions:
  - $eClassifiers.isDefined()$ & $extra.isDefined()$
  - $0 \leq index < eClassifiers.size()$
  - $extra.isKindOf(eClassifiers.getType())$
  - $lowerBound \leq eClassifiers.size() + 1 \leq upperBound$

  DEL(EClassifier eClassifiers, Integer index): Deletes the element at the specific position in the list *eClassifiers*
  Preconditions:
  - $eClassifiers.isDefined()$
  - $0 \leq index < eClassifiers.size()$
  - $lowerBound \leq eClassifiers.size() - 1 \leq upperBound$

  REP(EClassifier eClassifiers, Integer index, EClassifier newEClassifier): Replaces the value at the specific position in *eClassifiers* with *newEClassifier*.
  Preconditions:

- $eClassifiers.isDefined()$ & $newEClassifier.isDefined()$
- $0 \leq index < eClassifiers.size()$
- $newEClassifier.isNotTypeOf(eClassifiers(index).getType())$

- **EReference[EPackage eSubpackages], lowerBound=0, upperBound=***

  ADD(EPackage eSubpackages, Integer index,EPackage extra): Inserts *extra* at the specific position in the list *eSubpackages*.
  Preconditions:

  - $eSubpackages.isDefined()$ & $extra.isDefined()$
  - $0 \leq index < eSubpackages.size()$
  - $extra.isKindOf(eSubpackages.getType())$
  - $lowerBound \leq eSubpackages.size() + 1 \leq upperBound$

  DEL(EPackage eSubpackages, Integer index): Deletes the element of the specific position in the list *eSubpackages*.
  Preconditions:

  - $eSubpackages.isDefined()$
  - $0 \leq index < eSubpackages.size()$
  - $lowerBound \leq eSubpackages.size() - 1 \leq upperBound$

  REP(EPackage eSubpackages, Integer index, EPackage newEPackage): Replaces the value at the specific position in *eSubpackages* with *newEPackage*.
  Preconditions:

  - $eSubpackages.isDefined()$ & $newEPackage.isDefined()$
  - $0 \leq index < eSubpackages.size()$
  - $newEPackage.isNotTypeOf(eSubpackages(index).getType())$

- **EAttribute[EString name], lowerBound=0, upperBound=1:** this feature can take the following mutation operators defined for *EString* datatype.
  ADD(EString name, EString toAdd): Appends to the value of *name* the value specified by *toAdd*.
  Preconditions

  - $name.isDefined()$ & $toAdd.isDefined()$
  - $toAdd.length \geq 1$

  DEL(EString name, Integer toRemove): Removes randomly a number of *toRemove* characters from *name*.
  Preconditions

- $name.isDefined()$ & $toRemove.isDefined()$
- $name.length \geq toRemove \geq 1$

REP(EString name, EString newValue): Replaces the value of *name* with the value of *newValue*.
Preconditions

- $name.isDefined()$ & $newValue.isDefined()$
- $name \neq newValue$

- **EReference[EAnnotation eAnnotations], lowerBound=0, upperBound=***
ADD(EAnnotation eAnnotations, Integer index ,EAnnotation extra): Inserts *extra* at the specific position in *eAnnotations*.
Preconditions:
  - $eAnnotations.isDefined()$ & $extra.isDefined()$
  - $0 \leq index < eAnnotations.size()$
  - $extra.isKindOf(eAnnotations.getType())$
  - $lowerBound \leq eAnnotations.size() + 1 \leq upperBound$

  DEL(EAnnotation eAnnotations, Integer index): Deletes the element at the specific position in the list *eAnnotations*
  Preconditions:
  - $eAnnotations.isDefined()$
  - $0 \leq index < eAnnotations.size()$
  - $lowerBound \leq eAnnotations.size() - 1 \leq upperBound$

  REP(EAnnotation eAnnotations, Integer index, EAnnotation newEAnnotation): Replaces the value at the specific position in *eAnnotations* with *newEAnnotation*.
  Preconditions:
  - $eAnnotations.isDefined()$ & $newEAnnotation.isDefined()$
  - $0 \leq index < eAnnotations.size()$
  - $newEAnnotation.isNotTypeOf($ $eAnnotations(index).getType())$