A. Userguide

A.1 Introduction

The Subtraction prototype contains the implementations of the subtraction algorithms and classes to test them. The prototype reads a UML model from a given XMI-File. The user can choose the following options:

- specify a set of model elements for
  - identifying dependent model elements or
  - deleting them from the model[1];
- specify a set of OCL constraints on the UML model and
  - remove undefined parts,
  - normalize them to Conjunctive Normal Form or
  - duplicate them.

For any kind of comment, hint or bug report please contact vera.kiessling@stud.uni-karlsruhe.de.

[1] The elements are only removed from the model in memory, the XMI-File remains untouched!

A.2 Installation

1. Please make sure the version of your installed jdk is at least 1.4 and the java command is available at command line.

2. Additionally to the file Subtraction.zip you need the third-party jar packages:

   - **jmi.jar** from Sun Microsystems,
     available at http://java.sun.com/products/jmi/index.jsp,
   - **openide.jar** from the project Openide from Netbeans,
     available at http://openide.netbeans.org, and
   - **jmiutils.jar, mdrapi.jar, mof.jar, nbmdr.jar** from the project MDR from Netbeans,
     available at http://mdr.netbeans.org.

3. Extract Subtraction.zip to a new folder `testfolder`.

   `testfolder` contains now

   - folder `constraints` containing all OCL constraints to create,
   - a folder `repository` with files for the repository,
   - `EmployeeModel.xmi`, the example UML model,
   - `elementsToRemove.txt`, a text file with the model elements to delete,
   - `userguide.pdf`,
   - `CreateConstraint.java`, the interface all constraints have to implement and
   - `howto.txt`, a text file containing the command text to start the example.

4. Copy the third-party jar packages to `testfolder`.

A.3 How to execute the tests

1. Make sure the classpath contains the path to `testfolder` and all contained JAR-Files.

2. Open a shell and change the directory to `testfolder`.

3. Execute the following command in the commandline:

`java subtraction.test.`*<nameOfTest>* *<xmi-file>* *(<elementsToRemove.txt>)* `[-v]`

- *<nameOfTest>* can be any <xxx>Test of Figure A.1.
- *<xmi-file>* is the name of the UML model XMI-File.
- *elementsToRemove.txt* is the file containing all elements to remove from the UML model. This parameter is only needed with *SubtractionTest, RemoveUndefinedTest* and *DeletionTest*.
- The -v stands for verbose output and is optional.

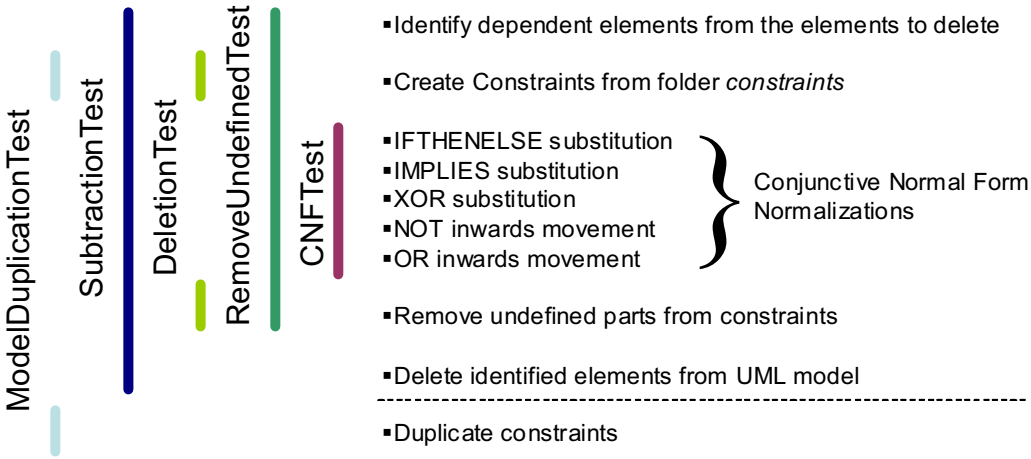

Figure A.1: Overview of available tests

For example the following command executes *SubtractionTest* with the EmployeeModel example and verbose output:

```
java -cp .;Subtraction.jar;
mof.jar;jmi.jar;mdrapi.jar;openide.jar;jmiutils.jar;mof.jar;nbmdr.jar
subtraction.test.SubtractionTest EmployeeModel.xmi file.txt -v
```

A.4 How to create own examples

1. Create a XMI-File of your UML model (e.g. with Poseidon for UML, available at http://www.gentleware.com).

2. Copy the XMI-File to `testfolder`.

3. Modify the entries in elementsToRemove.txt. The format of the entries is: (Please not that the additional <Classifier> entries were added for higher performance within the search of the element)

element	file entry
Classifier	Classifier <name>
AssociationClass	AssociationClass <name> <DependentClassifier>
Package	Package <name>
Association	Association <name> <DependentClassifier>
AssociationEnd	AssociationEnd <name> <CorrespondingClassifier>
Attribute	Attribute <name> <CorrespondingClassifier>

Table A.1: Possible entries of ElementsToRemove.txt

4. Write your own constraints with the Dresden OCL20 package. Sources and documentation are available at http://sourceforge.net/projects/dresden-ocl (cvs modul "OCL20"). The constraints have to implement the interface `CreateConstraint.java`. Copy the compiled classes to the subfolder `constraints`.

A.5 Example class diagram

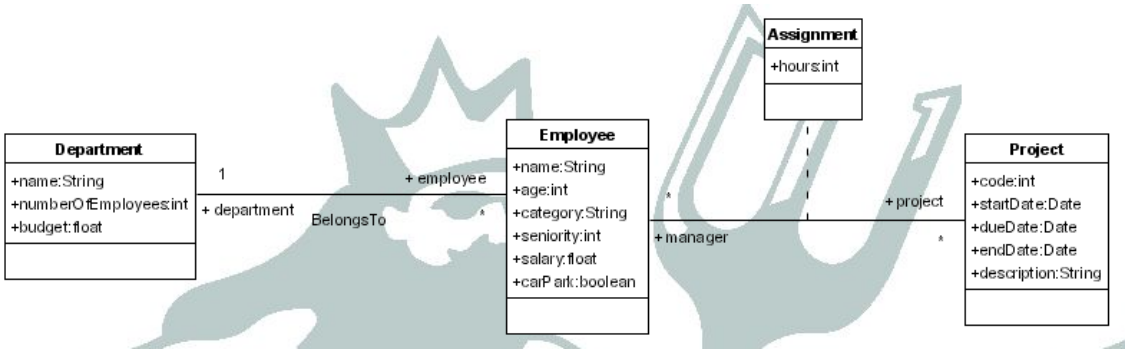

Figure A.2: EmployeeModel class diagram