Chapter 6

Xtext Laboratory

6.1 Project Initialisation

- Create a new Xtext project by New | Other. . . | Xtext | Xtext project with the following parameters:

Project name:	`hu.bme.mit.inf.socialnetwork`
Language name:	`hu.bme.mit.inf.socialnetwork.SocialNetworkLanguage`
Extensions:	`sn`
Create SDK feature project:	no

- Three projects are generated: a language definition project (`hu.bme.mit.inf.socialnetwork`), an editor (`hu.bme.mit.inf.socialnetwork.ui`) and a project for testing (`hu.bme.mit.inf.socialnetwork.tests`

- The `hu.bme.mit.inf.socialnetwork` project contains two source files: a grammar specification (`SocialNetworkLanguage.xtext`) and a workflow (`GenerateSocialNetworkLanguage.mwe2`) which defines the editor-specific details of the code generator.

- Generate the example language by Right click on the `.mwe2` or the `.xtext` file | | Run As MWE2 Workflow.

- A message appears in the console which asks for permission to download a necessary component. Press `Y` and enter.

- A complete editor is generated.

6.2 Grammar Specification

- Go to the `SocialNetworkLanguage.xtext` file. Create a community rule:

```
Community: ’community’ ID (’{’ (Community)+ ’}’)?;
```

- The previous rule is purely syntactic. In order to map the abstract syntax to an instance model to the parsing process metamodelling-specific information should be added to the grammar. Modify the rule to the following:

```
Community: ’community’ name = ID (’{’ (subcommunities += Community)+ ’}’)? ;
```

- Generate the language, then check the model folder.

- Create the following code:

```
SocialNetwork: ’social’ ’network’ {SocialNetwork} ’{’
  ( entities += Entity )*
’}’;

Entity: Community | Person;
```

```
Community: ’community’ name = ID
  (’{’
    (subcommunities += Community)*
  ’}’)?;

Person: ’person’ name=ID ’{’
  ’communities’ ’{’
    (community += [Community] (’,’ community += [Community])*)?
  ’}’
  ( acquaintaces += Acquaintace )*
’}’;

Acquaintace:
  type = AcqType ’of’ target = [Person]
;

enum AcqType: friend=’friend’ |
              relative = ’relative’ |
              college = ’college’;
;
```

6.3 Auxiliary Language Services

- Validator

```
@Check
def checkAcquaintanceEnds(Acquaintance acq) {
  if (acq.target == acq.getEParent()) {
  error("Acquaintance should be defined between two different persons.",
    SocialNetworkLanguagePackage.Literals.ACQUAINTANCE__TARGET);
  }
}
```

- Formatter

```
override protected void configureFormatting(FormattingConfig c) {
  val access = grammarAccess as SocialNetworkLanguageGrammarAccess
  // It’s usually a good idea to activate the following three statements.
  // They will add and preserve newlines around comments
  c.setLinewrap(0, 1, 2).before(access.getSL_COMMENTRule)
  c.setLinewrap(0, 1, 2).before(access.getML_COMMENTRule)
  c.setLinewrap(0, 1, 1).after(access.getML_COMMENTRule)

  val sne = access.getSocialNetworkAccess()
  c.setLinewrap(1, 1, 1).after(sne.getLeftCurlyBracketKeyword_1)
  c.setLinewrap(1, 1, 1).before(sne.rightCurlyBracketKeyword_3)
  c.setLinewrap(1, 1, 1).after(sne.getEntitiesAssignment_2_0)
  c.setLinewrap(2, 2, 2).before(sne.getAcquaintancesAcquaintanceParserRuleCall_2_1_0)
  c.setIndentationIncrement().after(sne.getLeftCurlyBracketKeyword_1)
  c.setIndentationDecrement().before(sne.getRightCurlyBracketKeyword_3)

  val ca = access.getCommunityAccess()
  c.setLinewrap(1, 1, 1).after(ca.getLeftCurlyBracketKeyword_2_0)
  c.setLinewrap(1, 1, 2).before(ca.getRightCurlyBracketKeyword_2_2)
  c.setIndentationIncrement().after(ca.getLeftCurlyBracketKeyword_2_0)
  c.setIndentationDecrement().before(ca.getRightCurlyBracketKeyword_2_2)

  val pa = access.getPersonAccess()
  c.setLinewrap(1, 1, 2).before(pa.getMembershipsKeyword_4_0)
  c.setLinewrap(1, 1, 1).before(pa.getSexAssignment_3)
  c.setLinewrap(1, 1, 1).after(pa.getSexAssignment_3)
  c.setLinewrap(1, 1, 1).after(pa.getLeftCurlyBracketKeyword_2)
  c.setLinewrap(1, 1, 1).before(pa.getRightCurlyBracketKeyword_5)
  c.setNoSpace().before(pa.getCommaKeyword_4_2_0)
  c.setIndentationIncrement().after(pa.getLeftCurlyBracketKeyword_2)
  c.setIndentationDecrement().before(pa.getRightCurlyBracketKeyword_5)
  c.setLinewrap(1, 1, 1).after(access.getAcquaintanceRule)
}
```