Textual AND graphical modeling: Xtext editors within Sirius Diagrams
Niko Stotz, niko.stotz@altran.com

EclipseCon Europe, 2018-10-24
Goals
Xtext within Sirius Diagram
Xtext within Properties View

<p>A <i>Room</i> is <b>not</b> available!</p>
<p>Please try at another place.</p>

<mismatched_input> mismatched input '<i>' expecting '</b>'</mismatched_input>

Press 'F2' for focus
Configured in regular Odesign Model
Without Changes to Xtext

```plaintext
grammar com.example.Classes with org.eclipse.xtext.common.Terminals

generate classes "http://www.example.com/Classes"

ClassModel:
   content+=Content*

Content:
   Constant
   Class
   Association

Constant:
   Description?

module com.example.GenerateClasses

import org.eclipse.xtext.xtext.generator.*
import org.eclipse.xtext.xtext.generator.model.project.*

var rootPath = ".."
Workflow {
    component = XtextGenerator {
        configuration = {
            project = StandardProjectConfig {
                baseName = "com.example.classes"
                rootPath = rootPath
                eclipsePlugin = {
                    enabled = true
                }
```
Demo
Config
Xtext Language Injector

```java
public class ClassesLanguageInjector implements IXtextLanguageInjector {
    @Override
    public Injector getInjector() {
        return ClassesActivator.getInstance().getInjector(
            ClassesActivator.COM_EXAMPLE_CLASSES);
    }
}
```
<plugin>
<extension point="com.altran.general.integration.xtextsirius.runtime.xtextLanguageInjector">
  <injector
    id="com.example.classes.design.ClassesLanguageInjectorId"
    class="com.example.classes.design.ClassesLanguageInjector"/>
</extension>
</plugin>
Odesign Tool / Widget
Odesign Details
Features
constant MAX_ROOMS = 23
constant MAX_PEOPLE = 42

class House {
    + inhabitants:
        string[1..MAX_PEOPLE]
}

association rooms House --> Room [1..MAX_ROOMS]

[[[<p>rooms are important</p>]]]

class Room {
    [[[
        <p>size doesn't really matter</p>]]]
    # size: integer[2..2]
}
Example Model as Tree

```
platform:/resource/classes/test.classes
  - Class Model
    • Constant MAX_ROOMS
    • Constant MAX_people
    • Constant MORE_CONSTANT
  - Class House
    • Attribute inhabitants
      • String Type
      • Integer Literal 1
      • Constant Ref
  - Class Room
    • Attribute size
    • Association rooms
```

Properties:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>[[[p] A &lt;i&gt;Room&lt;/i&gt; is &lt;b&gt;not&lt;/b&gt; available! /p] &lt;p&gt; Please try at another place. &lt;/p&gt; ]]</td>
</tr>
<tr>
<td>Name</td>
<td>Room</td>
</tr>
</tbody>
</table>
Xtext Editing Model String Property

Description:

- A Room is not available!
- Please try at another place.

Error message:

-mismatched input '<i>' expecting '</b>'

Press 'F2' for focus.
A Room is not available!
Please try at another place.
Document Completed from Config

```html
<html>
<head>
<title>dummy</title>
</head>
<body>

Prefix Text Expression: <html> <head> <title>dummy</title> </head> <body>

Suffix Text Expression: </body> </html>

</body>
</html>
```
A Room is not available!

Please try at another place.
Xtext Editing Model Itself
Pre-selecting Feature Text
Editing Only Subset of Features

- Attribute size
  - Integer Type
  - Integer Literal 2
  - Integer Literal 3

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td><code>&lt;p&gt;size doesn't really matter&lt;/p&gt;</code></td>
</tr>
<tr>
<td>Name</td>
<td>size</td>
</tr>
<tr>
<td>Visibility</td>
<td>#</td>
</tr>
</tbody>
</table>

<Room>
# size: integer[2..3]
</Room>

Xtext Model Direct Edit Label editAttribute

- General
  - Id*: editAttribute
  - Injector Id: com.example.classes.design.ClassesLanguageInjectorId
- Editable Features:
  - Attribute.name, Attribute.visibility, Attribute.lowerBound, Attribute.upperBound, Attribute.type
Editing Only Subset of Features

- Attribute size
  - Integer Type
  - Integer Literal 2
  - Integer Literal 3

Room

1. # size: integer[2..3]

1. Visibility

1. Xtext Model Direct Edit Label editAttribute

- General
  - Id*: editAttribute
  - Injector Id: com.example.classes.di.InjectionClassesLanguageInjector
  - Editable Features: Attribute.name, Attribute.visibility, Attribute.lowerBound, Attribute.upperBound, Attribute.type
Editing Only Subset of Features
Editing Only Subset of Features

1. Name: size
2. Visibility:
3. Attribute size

- Integer Type
- Integer Literal 2
- Integer Literal 3

Room
- size: integer

Xtext Model Direct Edit Label editAttribute
- Editable Features: Attribute.name, Attribute.visibility, Attribute.lowerBound, Attribute.upperBound, Attribute.type

Description: <p>size doesn't really matter</p>
Editing Only Subset of Features
Editing Only Subset of Features
Editing Only Subset of Features

- Description: <p>size doesn't really matter</p>
- Name: size
- Visibility: #

### Xtext Model Direct Edit Label editAttribute

**General**
- Id: editAttribute

**Documentation**
- Injector Id: com.example.classes.design.ClassesLanguageInjectorId
- Editable Features: Attribute.name, Attribute.visibility, Attribute.lowerBound, Attribute.upperBound, Attribute.type

### Room

```plaintext
# size: integer[2..3]
```
Different Grammar to Cut Middle Features

**association** rooms **House --> Room** [1..MAX_ROOMS]

```
Property | Value
---|---
Description | <p>rooms are important</p>
Name | rooms
Source | Class House
Target | Class Room
```
Gotchas

• Editing identifying property
• Working with erroneous models
• Heuristics to find terminals adjacent to features
• Size of Xtext edit box
Extensive Documentation

Source Code & Samples on GitHub

https://github.com/altran-mde/xtext-sirius-integration
Future: Contribute to Sirius?
Lifecycle of Ecore Model

- File on Disk
- Load
- Text including Whitespaces, Comments
- Parse
- Ecore Model
- Render
- Update
- Sirius Diagram
Lifecycle of Ecore Model

File on Disk → Load → Text including Whitespaces, Comments → Parse → Ecore Model

Ecore Model → render → update → Sirius Diagram

Graphical User Edit

Dirty Ecore Model → render → update → Dirty Sirius Diagram

Changed File on Disk → Save → Changed Text including Whitespaces, Comments → Serialize

Changed Text including Whitespaces, Comments → Graphical User Edit → Changed Text including Whitespaces, Comments
Lifecycle of Textual User Edit

File on Disk → Load → Parse → Ecore Model → Render

Text including Whitespaces, Comments

Changed File on Disk → Save → Serialize → Dirty Ecore Model → Render

Changed Text including Whitespaces, Comments

Dirty Ecore Model

Graphical User Edit

Sirius Diagram

Dirty Sirius Diagram

Clone
Lifecycle of Textual User Edit

File on Disk → Load → Text including Whitespaces, Comments → Parse → Ecore Model → render update → Sirius Diagram

Changed File on Disk → Save → Changed Text including Whitespaces, Comments → Serialize → Dirty Ecore Model → render update → Dirty Sirius Diagram

Textual User Edit

Dirty Ecore Model Clone
Lifecycle of Textual User Edit

File on Disk

Load

Text including Whitespaces, Comments

Parse

Ecore Model

render update

Sirius Diagram

Graphical User Edit

Changed File on Disk

Save

Changed Text including Whitespaces, Comments

Serialize

Dirty Ecore Model

render update

Dirty Sirius Diagram

Xtext editor with Text Part

Display Part to User

Dirty Text

Serial ize

Clone

Dirty Ecore Model Clone

Textual User Edit
Lifecycle of Textual User Edit

File on Disk

Load

Text including Whitespaces, Comments

Parse

Ecore Model

Render update

Sirius Diagram

Changed File on Disk

Save

Changed Text including Whitespaces, Comments

Serialize

Dirty Ecore Model

Render update

Dirty Sirius Diagram

Textual User Edit

Dirty Ecore Model Clone

Clone

Xtext editor with Text Part

Display Part to User

Dirty Text

Serialize

Xtext editor with changed Text Part

Get complete Text

Changed Text

User Edits Text
Lifecycle of Textual User Edit

File on Disk → Load → Text including Whitespaces, Comments → Parse → Ecore Model

Changed File on Disk → Save → Changed Text including Whitespaces, Comments → Serialize → Dirty Ecore Model

Textual User Edit

Dirty Ecore Model Clone

Dirty Sirius Diagram

Graphical User Edit

Changed EObjects

Extract

Clone

Dirty Ecore Model

Changed Ecore Model Clone

Changed Text

Parse

Get complete Text

Xtext editor with changed Text Part

Display Part to User

User Edits Text

Xtext editor with Text Part

Parse

Render

Update
Lifecycle of Textual User Edit

1. **File on Disk**
   - Load
   - Text including Whitespaces, Comments
   - Parse
   - Ecore Model
   - Render update
   - Sirius Diagram

2. **Changed File on Disk**
   - Save
   - Changed Text including Whitespaces, Comments
   - Serialize
   - Dirty Ecore Model
   - Render update
   - Dirty Sirius Diagram

3. **Textual User Edit**
   - Xtext editor with Text Part
   - Display Part to User
   - Dirty Text
   - Serialize
   - Dirty Ecore Model Clone

4. **Graphical User Edit**
   - Clone
   - Replace Original EObjects
   - Extract
   - Changed EObjects

5. **User Edits Text**
   - Xtext editor with changed Text Part
   - Get complete Text
   - Changed Text
   - Parse
   - Changed Ecore Model Clone
References from Edited Model Part

Original Model

```plaintext
events
  event1 [10 .. pi]

constants
  e 271
  pi 314
```
References from Edited Model Part

Original Model

- events
  - event1 [10 .. pi]
- constants
  - e 2.71
  - pi 3.14

Cloned Model

- events
  - event1 [10 .. pi]
- constants
  - e 2.71
  - pi 3.14

Textual User Edit

---

For the Edited Model Part, the constants e and pi remain unchanged. Only the events section is adjusted to include the event1 with a range from 10 to pi.
References from Edited Model Part

Original Model

events
  event1 [10 .. pi]

constants
  e 271
  pi 314

Cloned Model

Textual User Edit

events
  event1 [10 .. pi]

constants
  e 271
  pi 314

Changed Model

User Edits Text

events
  event1 [e .. pi]

constants
  e 271
  pi 314

Changed EObjects

events
  event1 [e .. pi]

constants
  e 271
  pi 314
References from Edited Model Part

Original Model

- events
  - event1 $[10 \ldots \pi]$
- constants
  - e 2.71
  - pi 3.14

Cloned Model

- events
  - event1 $[10 \ldots \pi]$
- constants
  - e 2.71
  - pi 3.14

Changed Model

- events
  - event1 $[e \ldots \pi]$
- constants
  - e 2.71
  - pi 3.14

Changed EObjects

- event1 $[e \ldots \pi]$
References from Edited Model Part

Original Model

- **events**
  - event1 \([10..\pi]\)
- **constants**
  - e 271
  - pi 314

Cloned Model

- **events**
  - event1 \([10..\pi]\)
- **constants**
  - e 271
  - pi 314

Changed Model

- **events**
  - event1 \([e..\pi]\)
- **constants**
  - e 271
  - pi 314

Proxified

- **events**
  - event1 \([e..\pi]\)
- **constants**
  - e 271
  - pi 314

Changed EObjects

- **events**
  - event1 \([e..\pi]\)
- **constants**
  - e 271
  - pi 314
References from Edited Model Part

Original Model

- **Events**: event1 \([10 .. \pi]\)
- **Constants**: e 271, \(\pi 314\)

Cloned Model

- **Events**: event1 \([10 .. \pi]\)
- **Constants**: e 271, \(\pi 314\)

User Edits

- **Events**: event1 \([\text{replaced]} \ e .. \pi]\)
- **Constants**: e 271, \(\pi 314\)

Proxified

- **Events**: event1 \([\text{proxified]} \ e .. \pi]\)
- **Constants**: e 271, \(\pi 314\)

Updated Model

- **Events**: event1 \([\text{updated]} \ e .. \pi]\)
- **Constants**: e 271, \(\pi 314\)

Proxified EObjects

- **Events**: event1 \([\text{proxified]} \ e .. \pi]\)
- **Constants**: e 271, \(\pi 314\)

Changed EObjects

- **Events**: event1 \([\text{proxified]} \ e .. \pi]\)
- **Constants**: e 271, \(\pi 314\)
Evaluate the Sessions

Sign in and vote at eclipsecon.org

-1  0  +1