Prototype Handbook
Extended Petri net Simulator

Group A

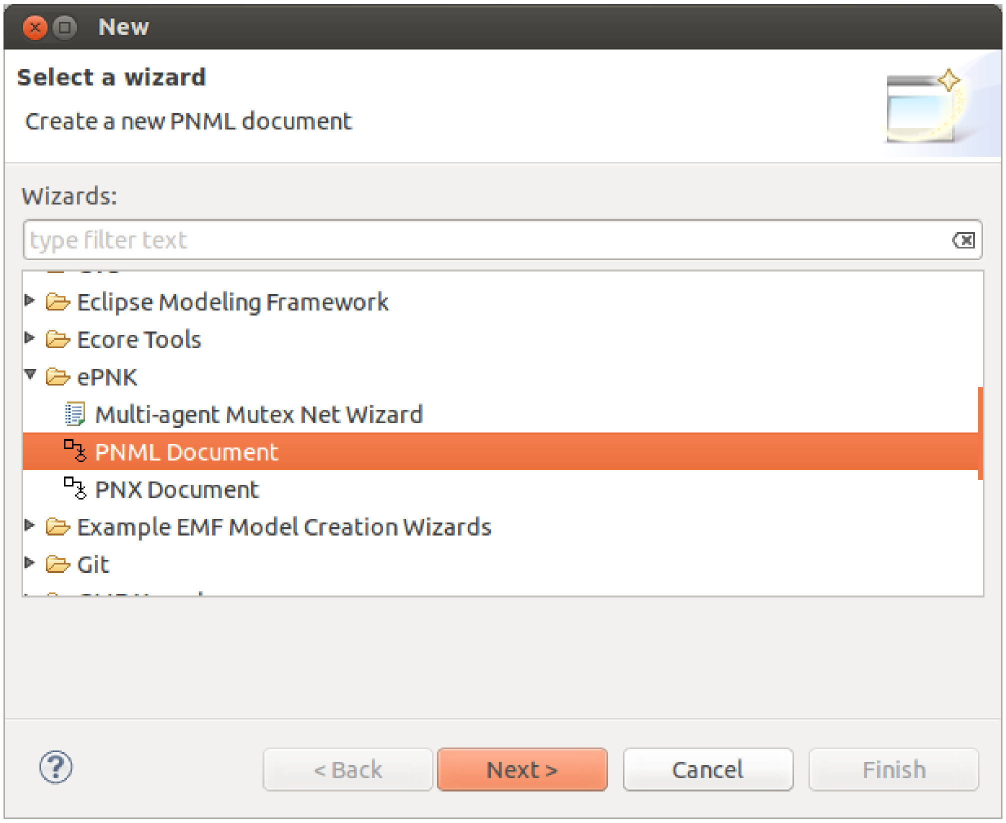
October 24, 2013

Abstract

This document includes a Handbook for the first prototype of the project for Software Engineering 2.

Contents

1 Introduction



Select a wizard

Create a new PNML document

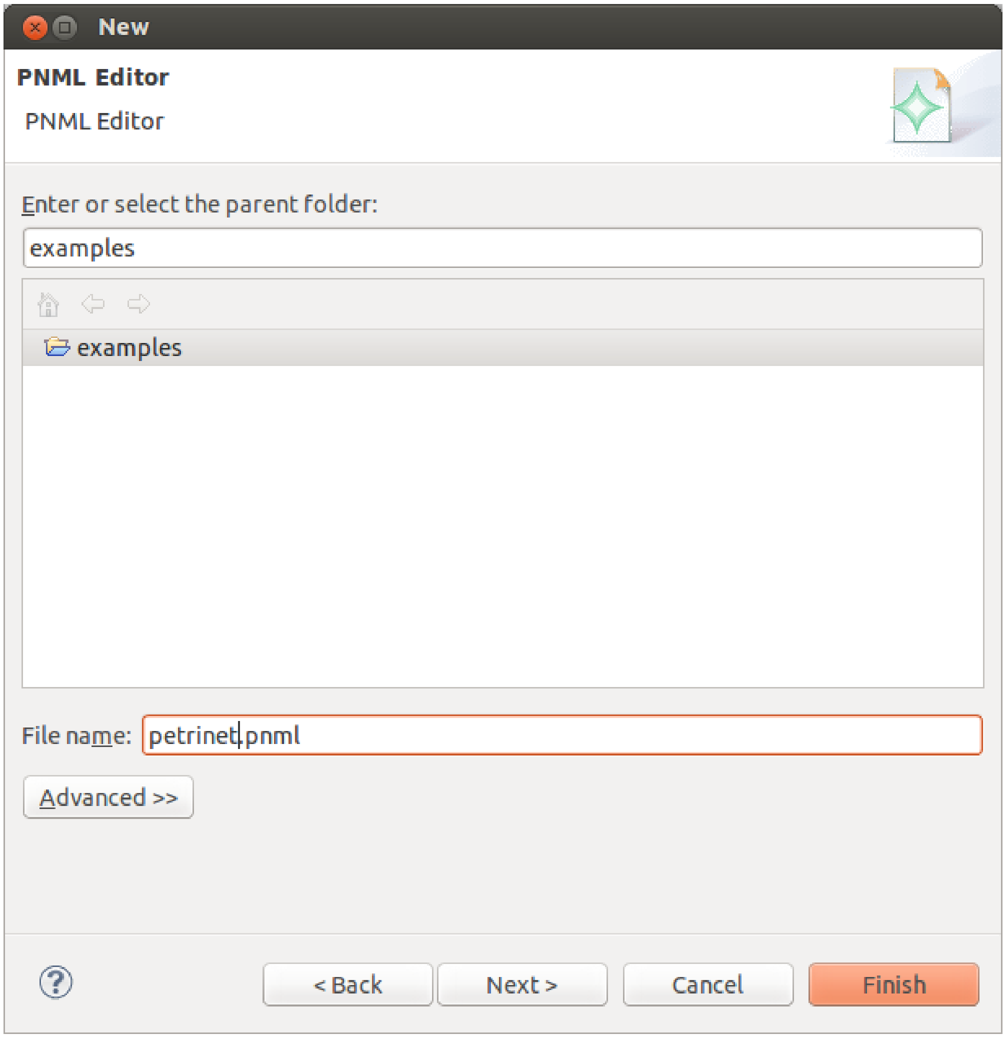
Figure 1: Use cases for the Petri net Editor

To use geometry editor in Eclipse first thing needed to do is to create a geometry file. To create it, a new General Project can be created by $General- > Project$ which is found in $File- > New- > Project$. Give the project some name and click finish. Right click the project folder $New- > Other$. Find Geometry diagram from the list, choose it and click next.

Name it and click finish. On the project explorer leaf click on the file that you just generated. Geometry editor should open if it wasn't open already.

Available tools are presented on right of the canvas. Clicking InputPoint, Connector or BendPoint allows user to create different points on the canvas. Line allows connecting Connector with each other. Following picture illustrates one example created with these tools.

Furthermore InputPoint (solid circle), Connector (box) and Bendpoints (dashed circle) can be created when user hovers the mouse on the screen for a while. A view opens which allows picking point type. Additionally hovering the mouse on a Connector presents arrows. By clicking them user can draw line to another connector the same way with clicking Line from tool palette. Clicking on some geometry object show a property view. If this is not visible right click on a geometry object on canvas and click Show Properties View. InputPoint has following properties to edit: Id, In, Label, Out, XLocation and YLocation. Id is a number, Label is name of the object if user wants to name



New

PNML Editor

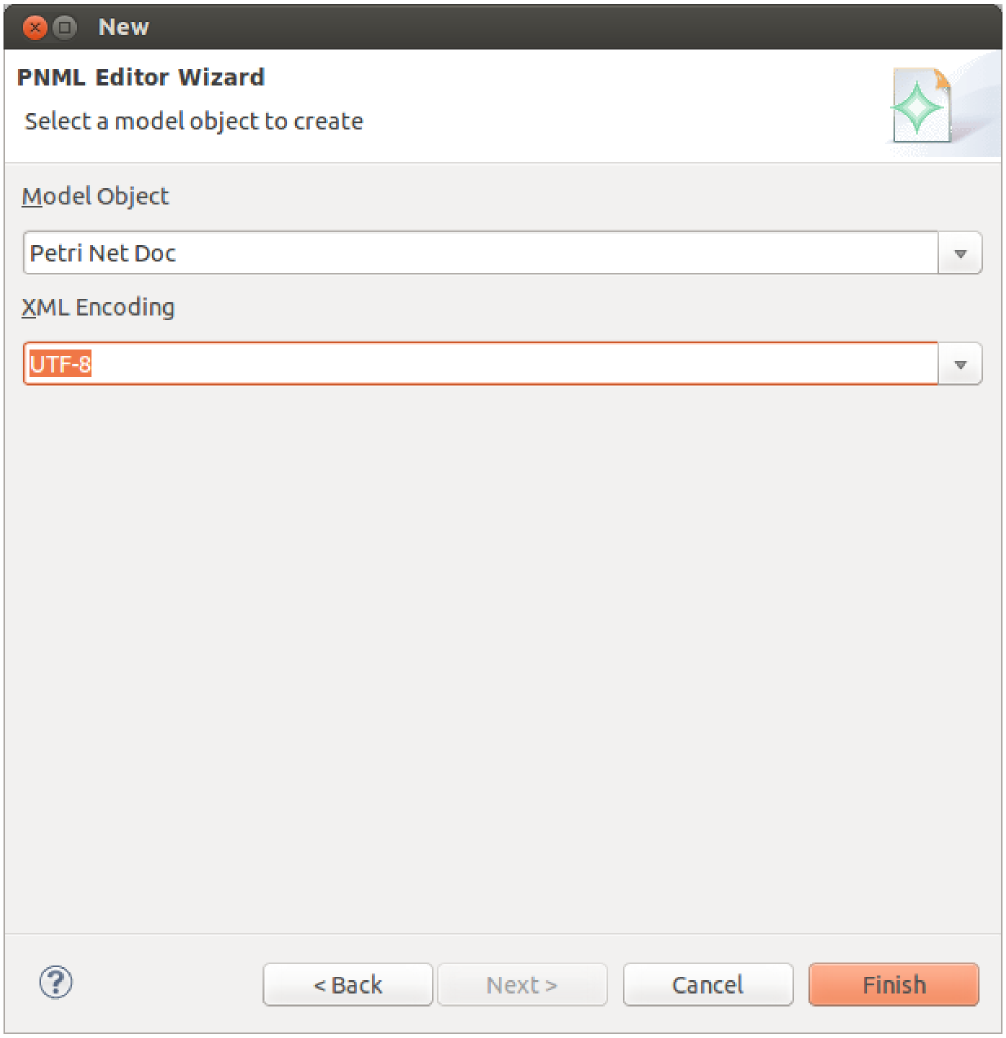
PNML Editor

Enter or select the parent folder: examples

examples

File name: petrinet.pnml

Figure 2: Use cases for the Petri net Editor



PNML Editor Wizard

Select a model object to create

Model Object

Petri Net Doc

XML Encoding

UTF-8

Figure 3: Use cases for the Petri net Editor

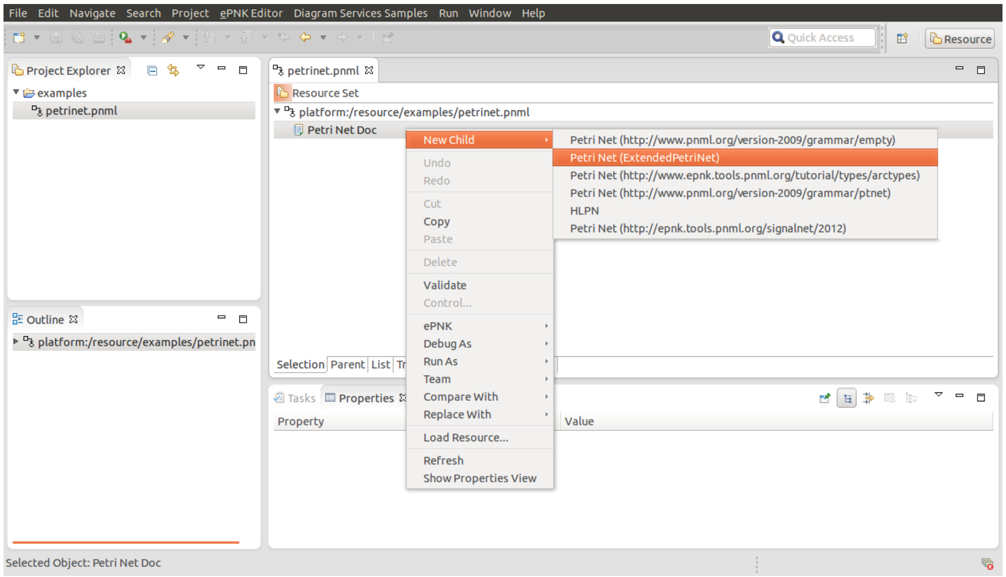

File Edit Navigate Search Project ePNK Editor Diagram Services Samples Run Window Help
Quick Access
Resource
Project Explorer
examples
petrinet.pnml
petrinet.pnml
Resource Set
platform:/resource/examples/petrinet.pnml
Petri Net Doc
New Child
Undo
Redo
Cut
Copy
Paste
Delete
Validate
Control...
ePNK
Debug As
Run As
Team
Compare With
Replace With
Load Resource...
Refresh
Show Properties View
Petri Net (http://www.pnml.org/version-2009/grammar/empty)
Petri Net (ExtendedPetriNet)
Petri Net (http://www.epnk.tools.pnml.org/tutorial/types/arctypes)
Petri Net (http://www.pnml.org/version-2009/grammar/ptnet)
HLPN
Petri Net (http://epnk.tools.pnml.org/signalnet/2012)
Outline
platform:/resource/examples/petrinet.pn
Selection Parent List Tr
Tasks
Properties
Property
Value
Selected Object: Petri Net Doc

Figure 4: Use cases for the Petri net Editor

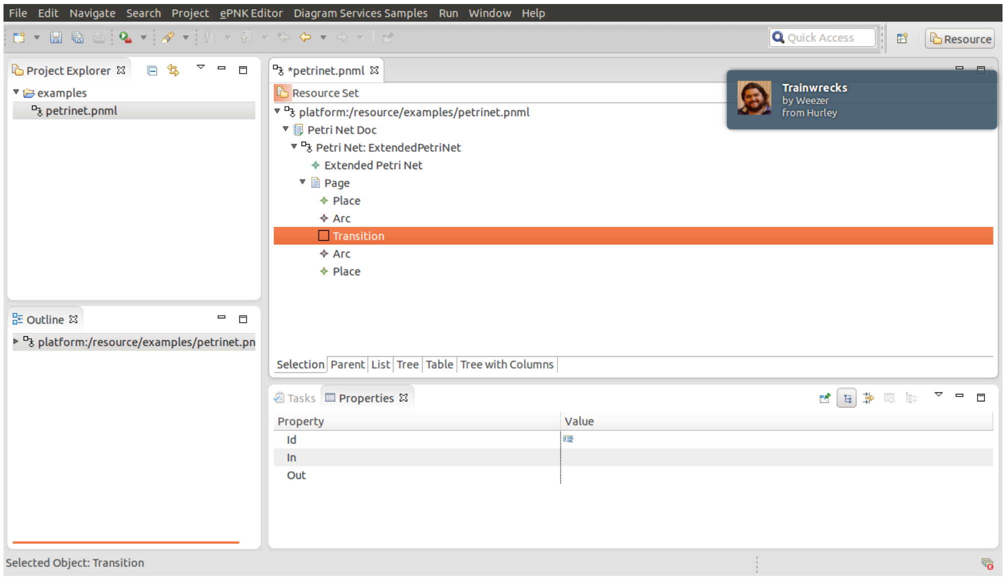
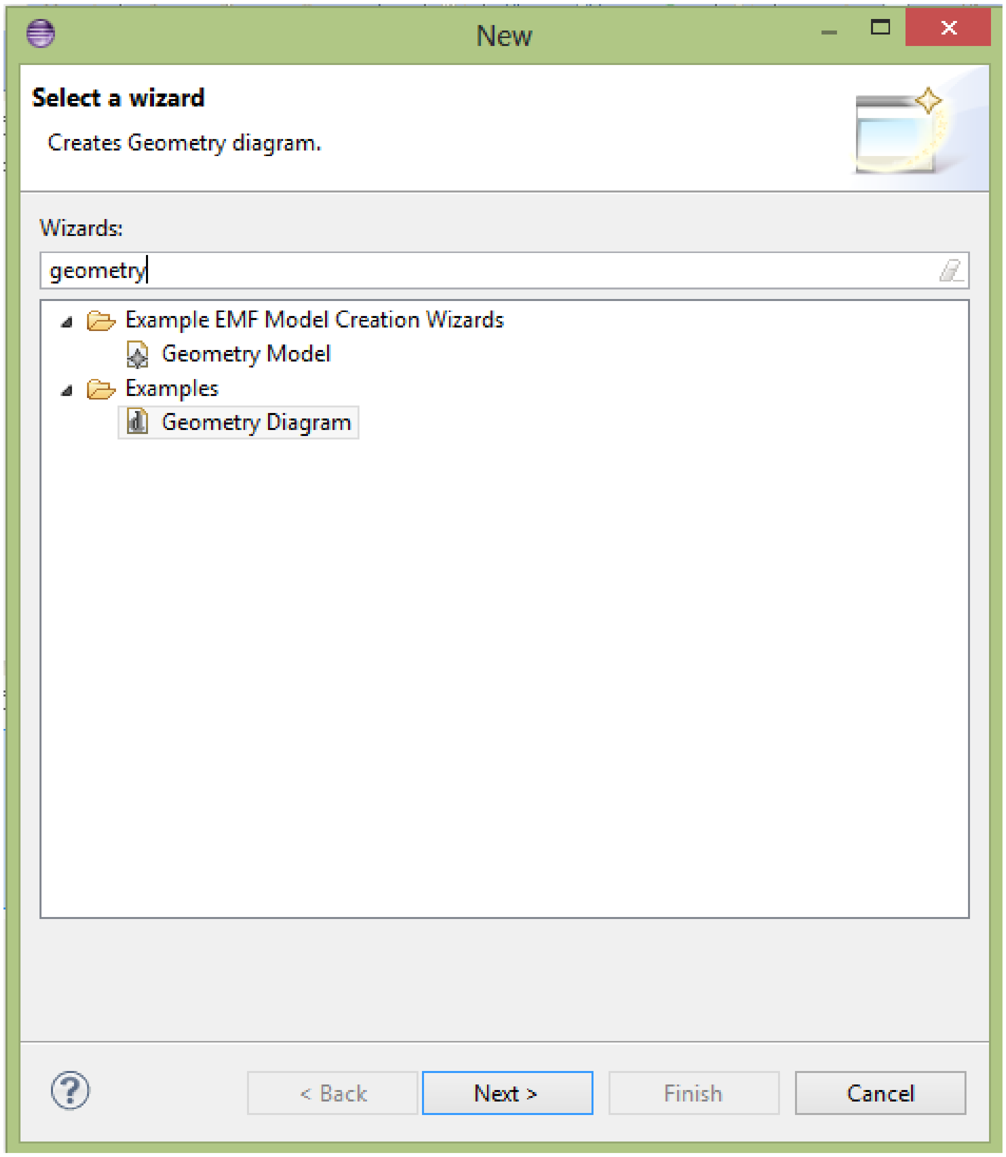

File Edit Navigate Search Project ePNK Editor Diagram Services Samples Run Window Help
Quick Access
Resource
Project Explorer
examples
petrinet.pnml
*petrinet.pnml
Trainwrecks
by Weezer
from Hurley
Resource Set
platform:/resource/examples/petrinet.pnml
Petri Net Doc
Petri Net: ExtendedPetriNet
Extended Petri Net
Page
Place
Arc
Transition
Arc
Place
Outline
platform:/resource/examples/petrinet.pn
Selection Parent List Tree Table Tree with Columns
Tasks
Properties
Property
Value
Id
In
Out
Selected Object: Transition

Figure 5: Use cases for the Petri net Editor



New

Select a wizard

Creates Geometry diagram.

Wizards: geometry

- Example EMF Model Creation Wizards
  - Geometry Model
- Examples
  - Geometry Diagram

Figure 6: Use cases for the Petri net Editor

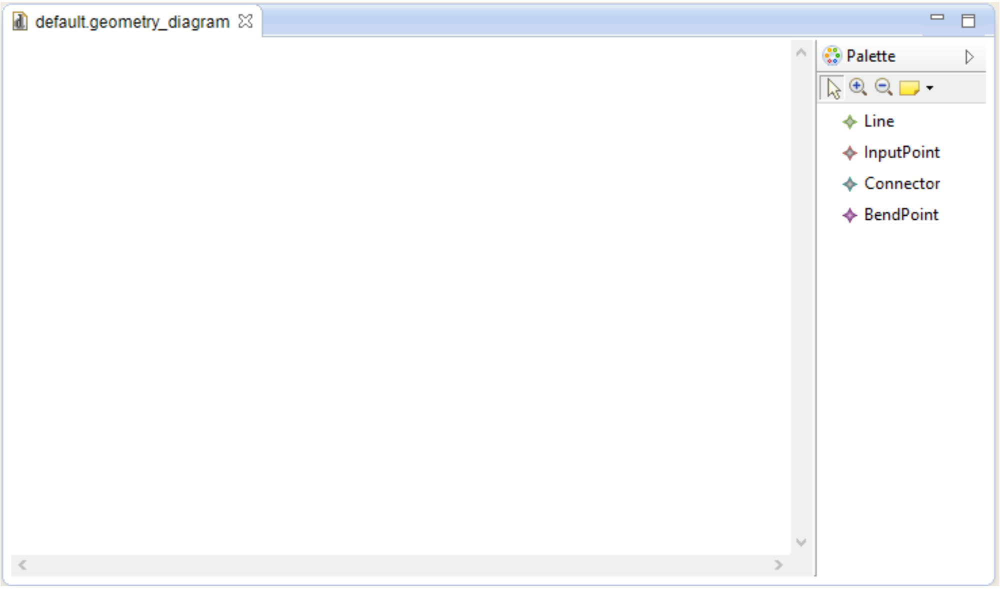


Figure 7: Use cases for the Petri net Editor

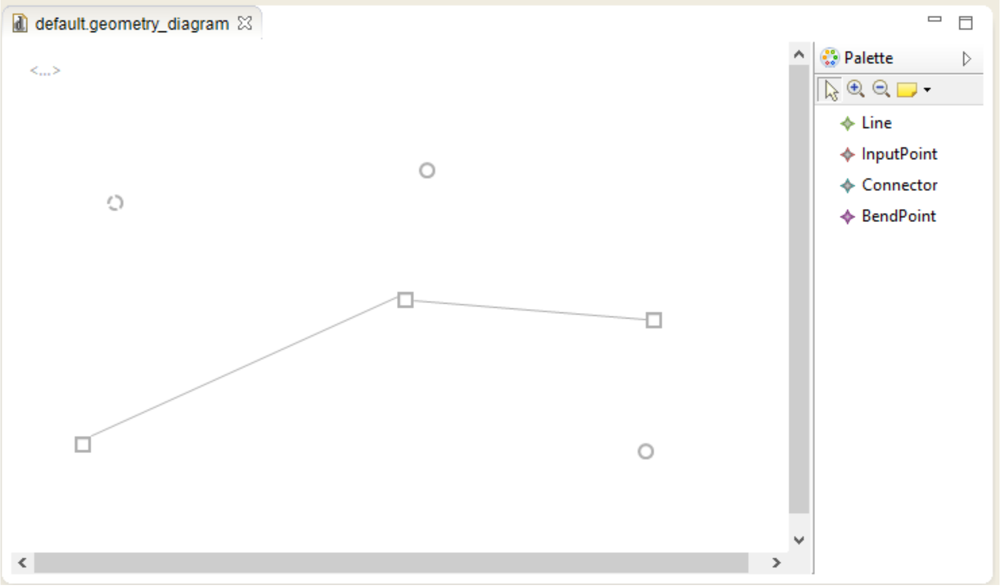
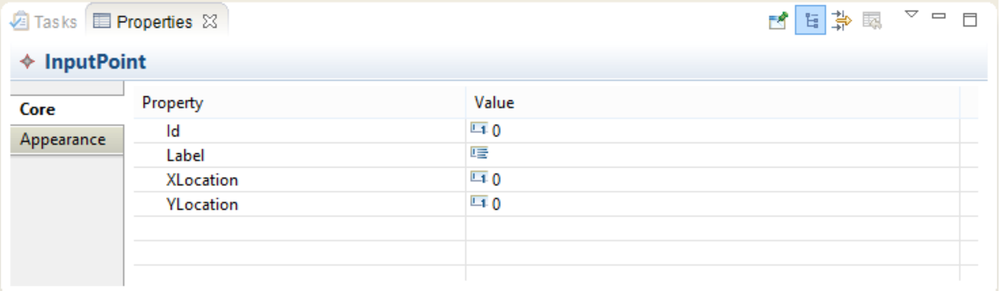
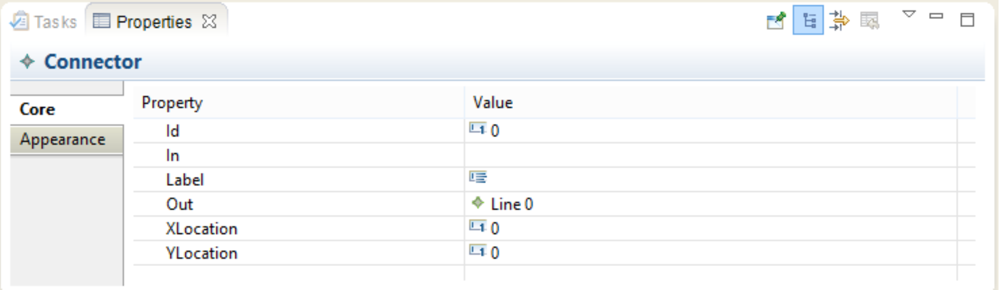


Figure 8: Use cases for the Petri net Editor



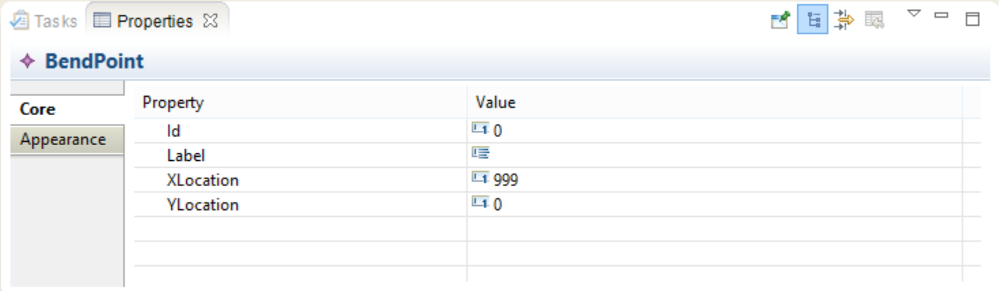
Property	Value
Id	0
Label	
XLocation	0
YLocation	0

Figure 9: Use cases for the Petri net Editor



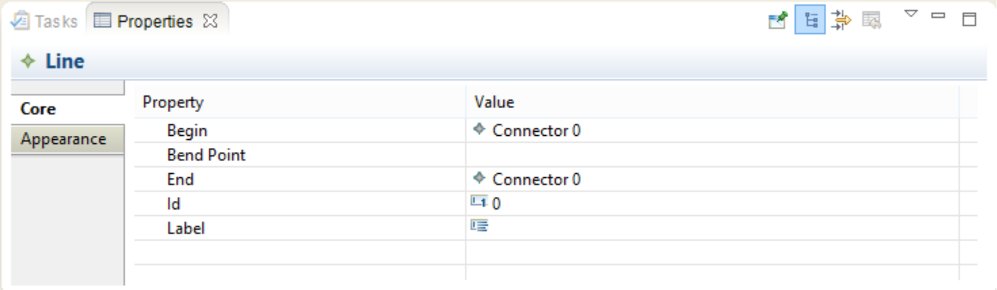
Property	Value
Id	0
In	
Label	
Out	Line 0
XLocation	0
YLocation	0

Figure 10: Use cases for the Petri net Editor



Property	Value
Id	0
Label	
XLocation	999
YLocation	0

Figure 11: Use cases for the Petri net Editor



Property	Value
Begin	Connector 0
Bend Point	
End	Connector 0
Id	0
Label	

Figure 12: Use cases for the Petri net Editor

it and XLocation and YLocation are the locations to show where the object is. As of currently location of canvas hasn't anything to do with the location in the canvas. This will be adjusted on later versions of this project.

Connector has following properties to edit: Id, Label, XLocation and YLocation. Besides Id, Label and XLocation and YLocation Connector has In and Out selector. For these user can choose a Line to be connected. The number at the end of Line denotes Id.

BendPoint has the same properties as InputPoint has. The XLocation of this BendPoint has been changed to some other value.

Line has following properties to edit: Begin, BendPoint, End, Id and Label. Begin denotes the starting connector of the line. Bend Point is the parameter point on how the line is curved. The geometry canvas doesn't show this curvature on current version. End is the ending connector of the line.