Simulation of UML StateChart

IETA MICHAËL RIGAUD

Stage chief: Prof. Hans Vangheluwe

Tutor: Simon Van Mierlo

Contents

Introduction

Part I

Presentation

CHAPTER 1

Presentation of the project

1.1 The goal

The goal of this project is to create a simulator of Statechart which can be use with UMLDesigner. This simulator should permit to visualize and debug a model of a state machine. Moreover, UMLDesigner is a modeling software for UML model and Statechart, so we could create the model and simulate it on the same tools. The picture 1.1 represent the aim of this project.

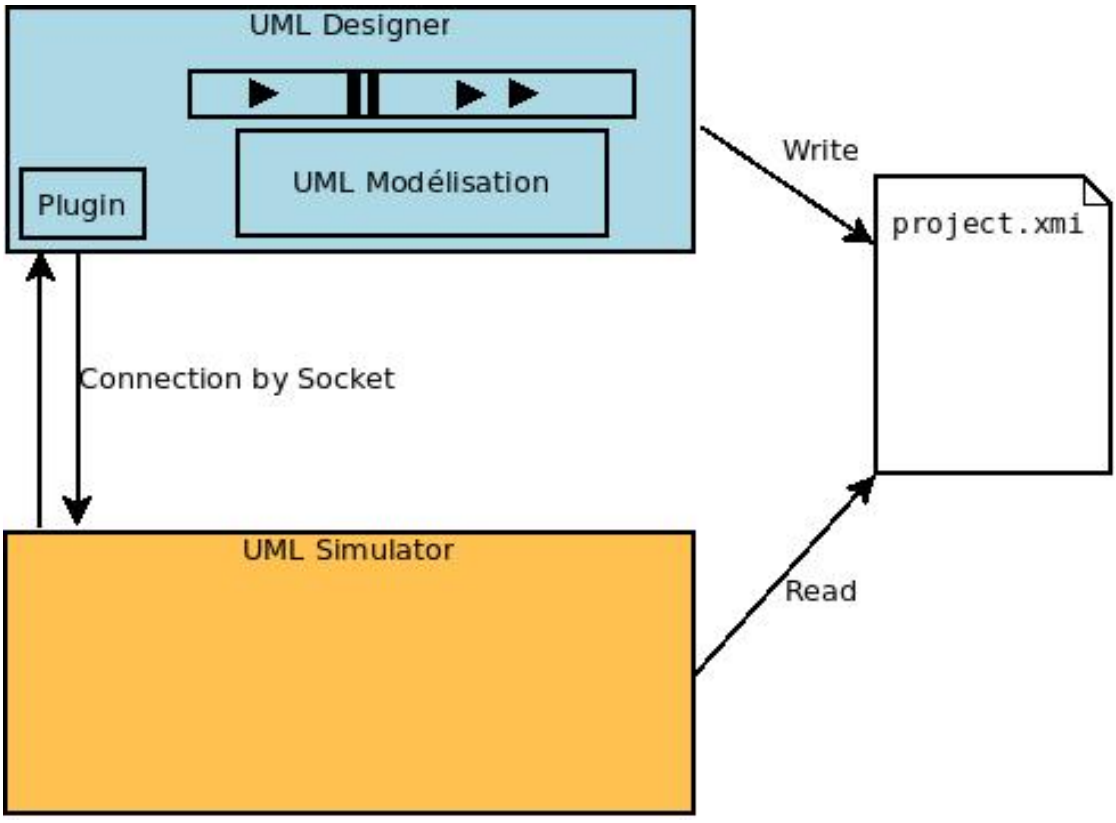

Figure 1.1: Description of the project

1.2 Tools at the disposal

At the begin of this project, some of the tools, which were needed, existed. In fact, ULMDesigner is a UML modeling tool develop by *Obeo*. However, it didn't exist yet

a simulator for Statechart adapted for UMLDesigner. On the chapter 2, the running of UMLDesigner will be discuss.

Then, Mr Ciprian Theodorov, one of my professor, has developed a simulator for Statechart. This simulator needed to be improved, but it composed a good beginning for this project.

CHAPTER 2

UML Designer

UML Designer is an open-source tool to edit and visualize UML2 models created by the French company: *Obeo*. The project is licensed under the EPL[1]

Figure 2.1: UML Designer logo

2.1 Utilization

UML Designer is a graphical modeling tool for UML2 as defined by OMG[2]. As you can see on the figure 2.2, it permit to create diagram on which ones it is possible to add some elements. The type of the elements proposed depend on the types of the diagram chosen. For example, if you choose a *User case diagram* it is possible to add 'user' component that is impossible in *Class diagram*.

So with graphical action it is possible to create many UML diagram which have transverse elements.

To finish, it is possible to create the code of the application that you have develop from the model.

2.2 List of diagram supported

- Packages diagram
- Use case diagram
- Activity diagram
- Class diagram

[1] Eclipse public license
[2] Object Management Group[3]

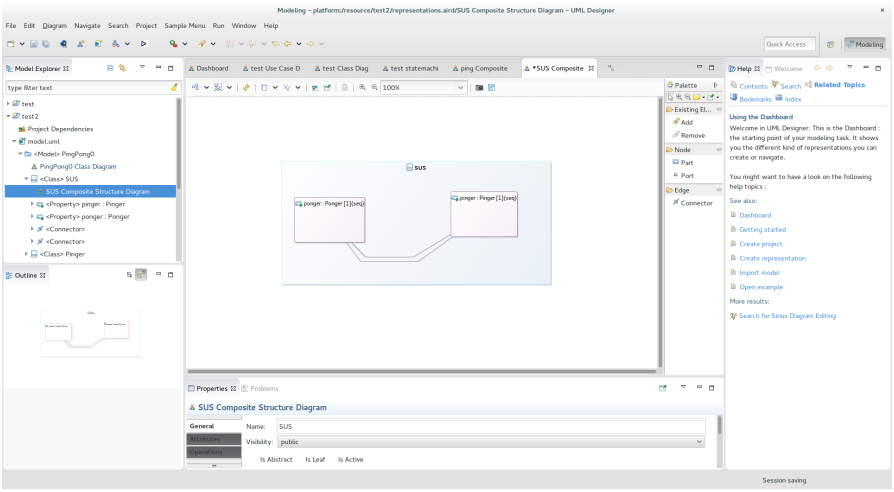

Figure 2.2: Screen shot of UML Designer

- Component diagram
- Composite Structure diagram
- Sequence diagram
- State Machine diagram
- Documentation table
- Use Case cross table
- Package containment diagram
- Profile diagram

2.3 Released

Version	Release Date
1.0.0	2012
2.0.0	17 January 2013
2.1.0	1 February 2013
2.2.0	12 April 2013
2.3.0	13 June 2013
2.4.0	13 September 2013
3.0.0	17 January 2014
4.0.0	8 July 2014
4.0.1	5 August 2014
5.0.0	29 May 2015
6.0.0	19 October 2015

Legend:
Latest stable release

2.4 Base on

UML Designer is based on a Eclipse and Sirius. It is a UML2 Eclipse plugin.

Sirius

Sirius is an open-source software project of the Eclipse Foundation. Sirius allows to create graphical modeling workbench. It include EMF[3] and GMF[4]. On the figure 2.3, it is possible to see the architecture of Sirius.

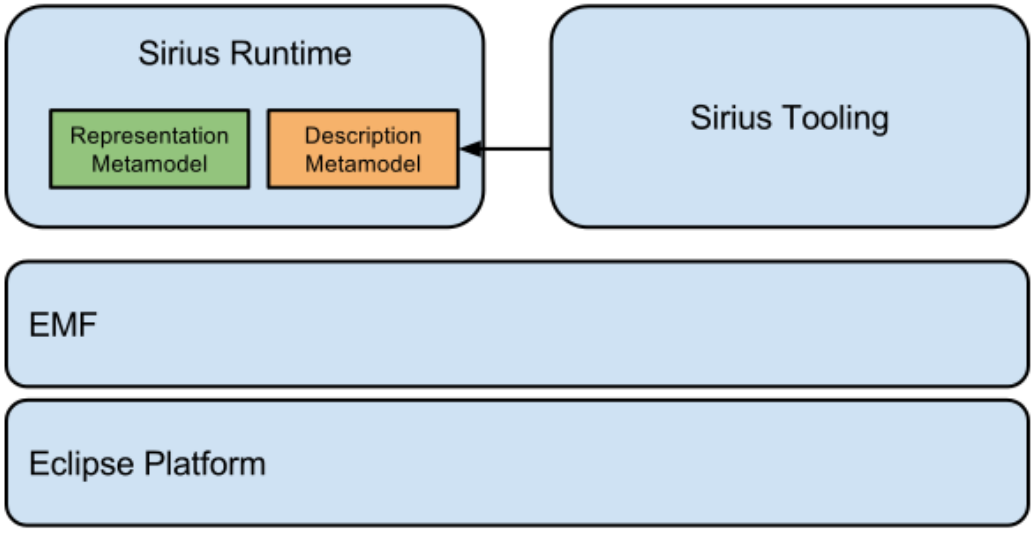

Figure 2.3: Sirius architecture[2]

Eclipse

UML Designer is base on Eclipse. The interface is the same as Eclipse. You can notice on figure 2.2 that the menu are the same in the both software.

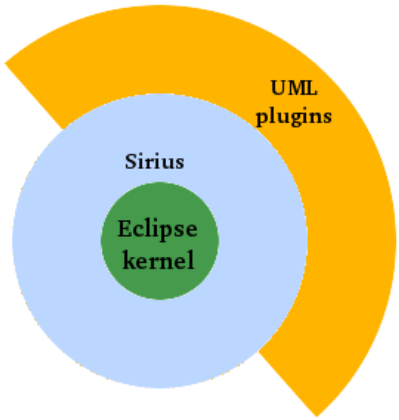

Figure 2.4: The UML Designer kernel

[3]Eclipse Modeling Framework
[4]Graphical Modeling Framework

CHAPTER 3

Simulator

3.1 Description

At the beginning of this project, we had at our disposal the simulator of Mr Teodorov (figure 3.1). This simulator have a graphic user interface as you can see on the figure 3.1.

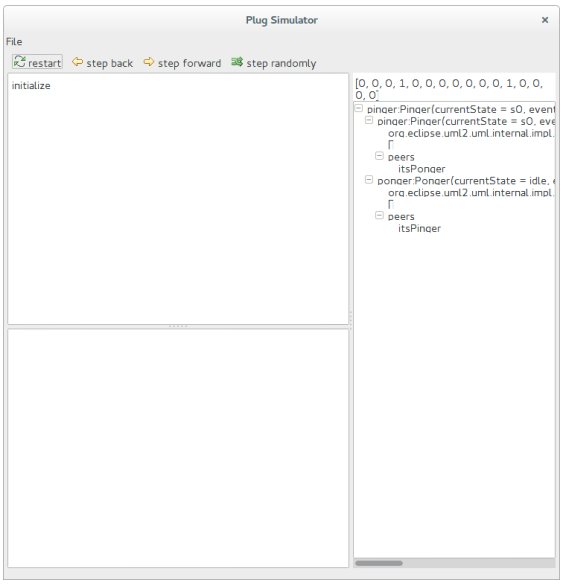

Figure 3.1: Mr Teodorov simulator

The simulator is compose on 4 part.

- On the top: some buttons to select an action
- On the top-left-corner: The list of the next step
- On the bottom-left-corner: The State Machine associated to the Current State.
- On the right: A visualization of the Statechart

3.2 Type of entry

This simulator simulate a uml file. The uml file need to have a particular architecture.

Part II

Study of the subject

CHAPTER 4

Communication inter process

4.1 Type of communication conceivable

A lot of type of communication inter process were suggested to create a discussion enter the plugin and the simulator. But we will present only the most consistent with their advantages and their drawbacks.

The communication is the most important part of this project, because that will implement the interface between the two software.

Socket

Advantages	Drawback
Work with every simulator type (python, java, ...)	Message need to be formatted
	Not very fast

File

Advantages	Drawback
Problem when two software want to change the same file at the same moment	Communication asynchronous

Named pipe

Advantages	Drawback
It is possible to use the Simulator outside the graphical modeling tool	

Shared Memory

Advantages	Drawback
It is possible to use the Simulator outside the graphical modeling tool	

Thread

Advantages	Drawback
	problem if the thread don't avance at the good speed

Heritage

Advantages	**Drawback**
Easy to implement	Need to add code in the simulator
	We can only use simulator in Java

Our solution

The solution was not in this list of common way to communicate inter process. In fact, we use the *Runtime* class which is in the java library.

Advantages	**Drawback**
It is possible to use the Simulator outside the graphical modeling tool	
Work with every type of simulator	

Conclusion

Annexe

Appendix A

Organisation of the work

A.1 Calendar

Tasks/weeks	1	2	3	4	5	6	7	8	9	10	11	12	13	14
State of the art	-	-												
Create a plugin			-											
Visualize the simulation				-	-	-								
Unit tests						-	-							
Integration tests								-						
Improve the simulator									-	-				
Try an other simulator											-	-		
Redaction		-	-	-	-	-	-	-	-	-	-	-	-	
Soutenance														-

A.2 Tools use for the project

The Framaboard application:

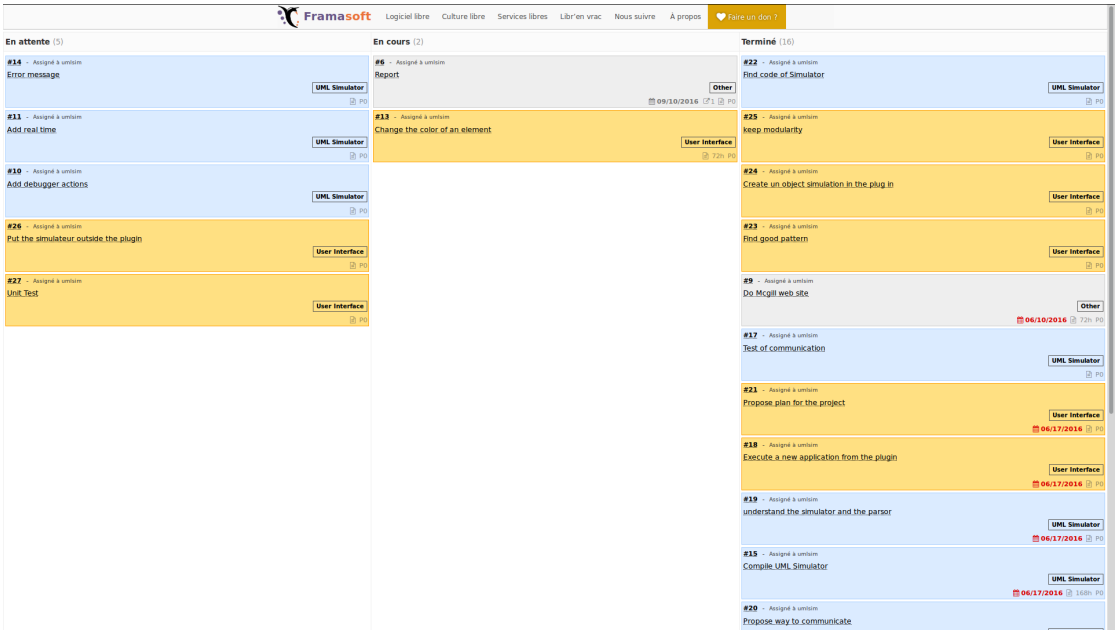

Figure A.1: Screen shot of the framaboard

The web site of MSDL researcher:

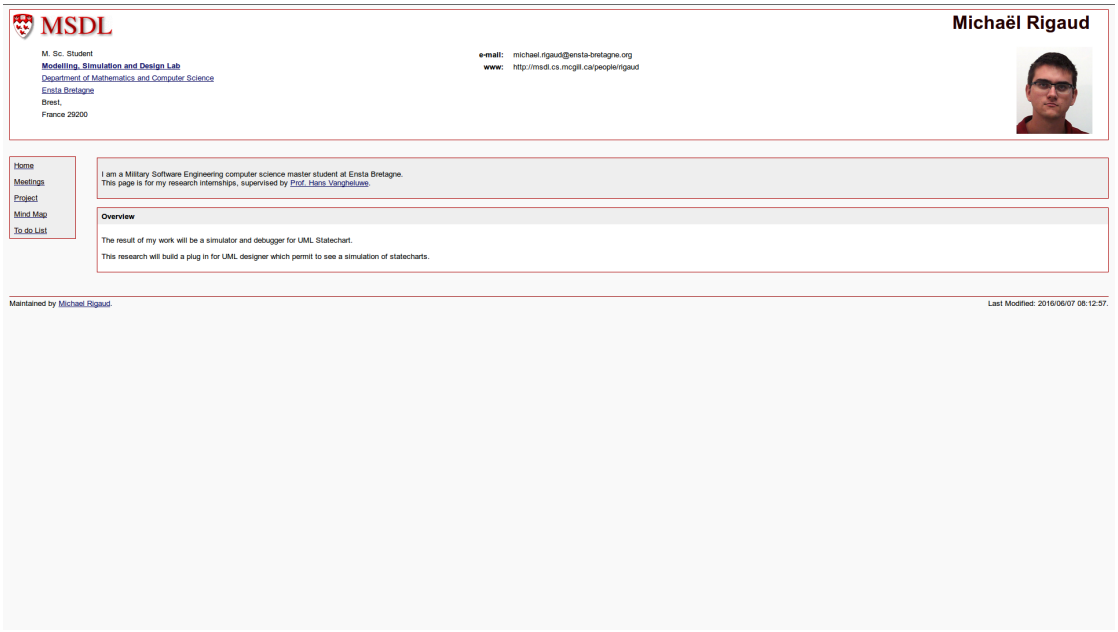

Figure A.2: MSDL web site

List of Figures

Bibliography

[1] Obeo. Contribute developer guide.

[2] Eclipse Obeo. Sirius documentation. https://www.eclipse.org/sirius/.

[3] OMG. http://www.omg.org/.