



DEVS Training Course

Class 03 - DEVS Programming - Part 3

Prof. Dr. Valdemar Vicente Graciano Neto

In the previous class...



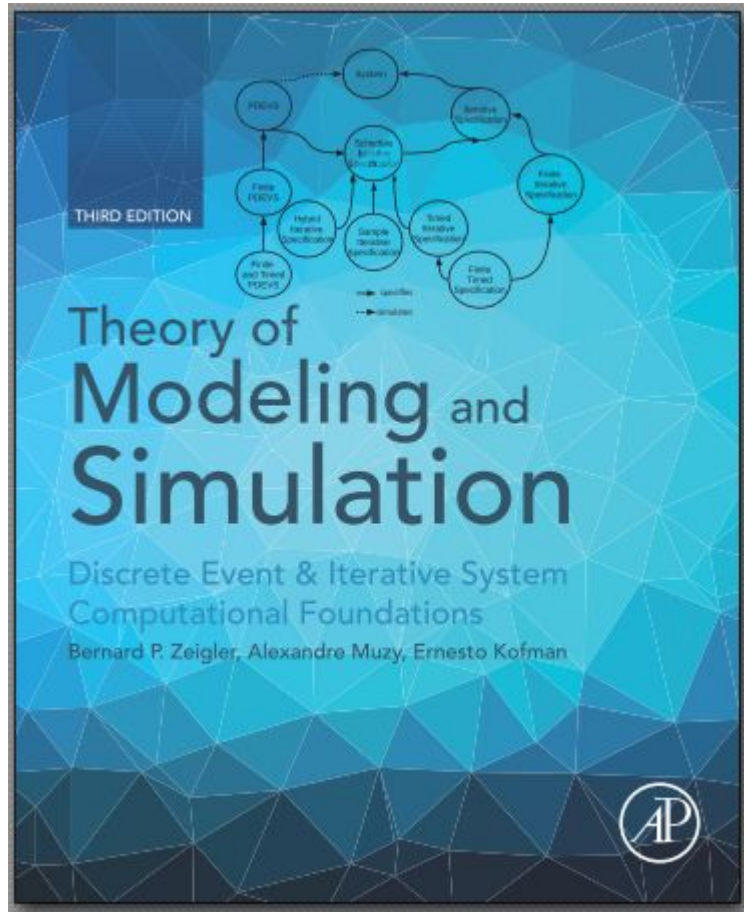
- Fundamentals of DEVS Theory
- DEVS and DEVSNL
- Some good programming practices to mention
- Changing perspectives and composing models.

In this class



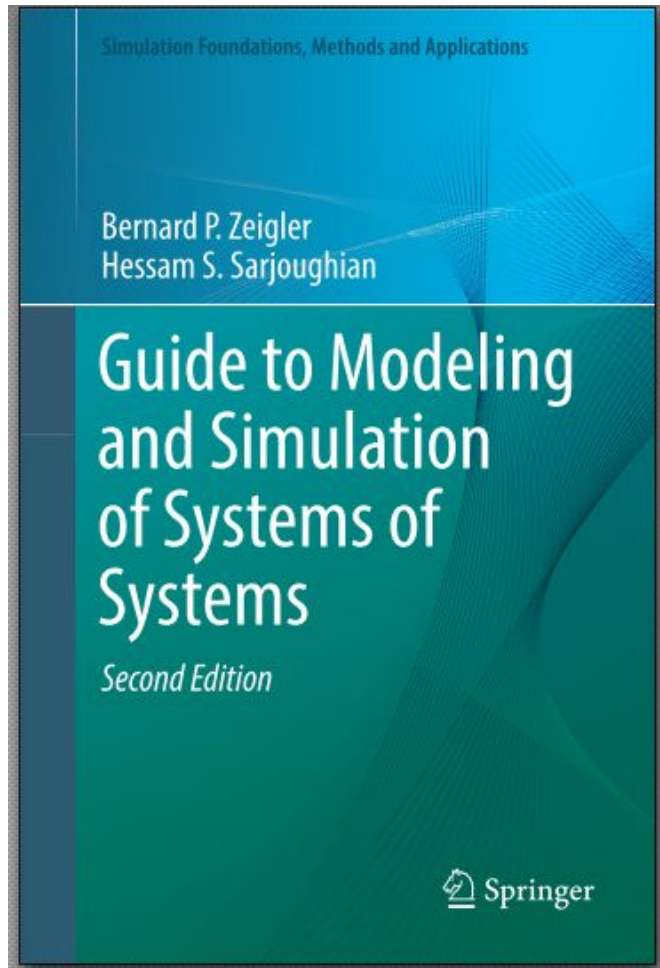
- ▣ Library Import
- ▣ State Diagram
- ▣ Working with files
- ▣ Parallel DEVS (P-DEVS)

Foundations for this class



Bernard Zeigler; Alexandre Muzy;
Ernesto Kofman. Theory of Modeling and
Simulation - Discrete Event & Iterative
System Computational Foundations. 3rd
Edition
Academic Press. 2018. 692 p.

Foundations for this class



Bernard Zeigler; Hessam Sarjoughian.
Guide to Modeling and Simulation of
Systems of Systems. Springer.
2013/2017. 400 p.



DEVS Practice

DEVS Practice



□ Importing Libraries into Atomic Models

```
add library
<%
import java.io.FileReader;
import java.io.BufferedReader;
%> !
```

DEVS Practice



□ Read file on event in Atomic Models

```
BufferedReader br = new BufferedReader(new FileReader("c:/file.html"));
while(br.ready()){
    String line = br.readLine();
    System.out.println(line);
}
br.close();
```


DEVS Practice

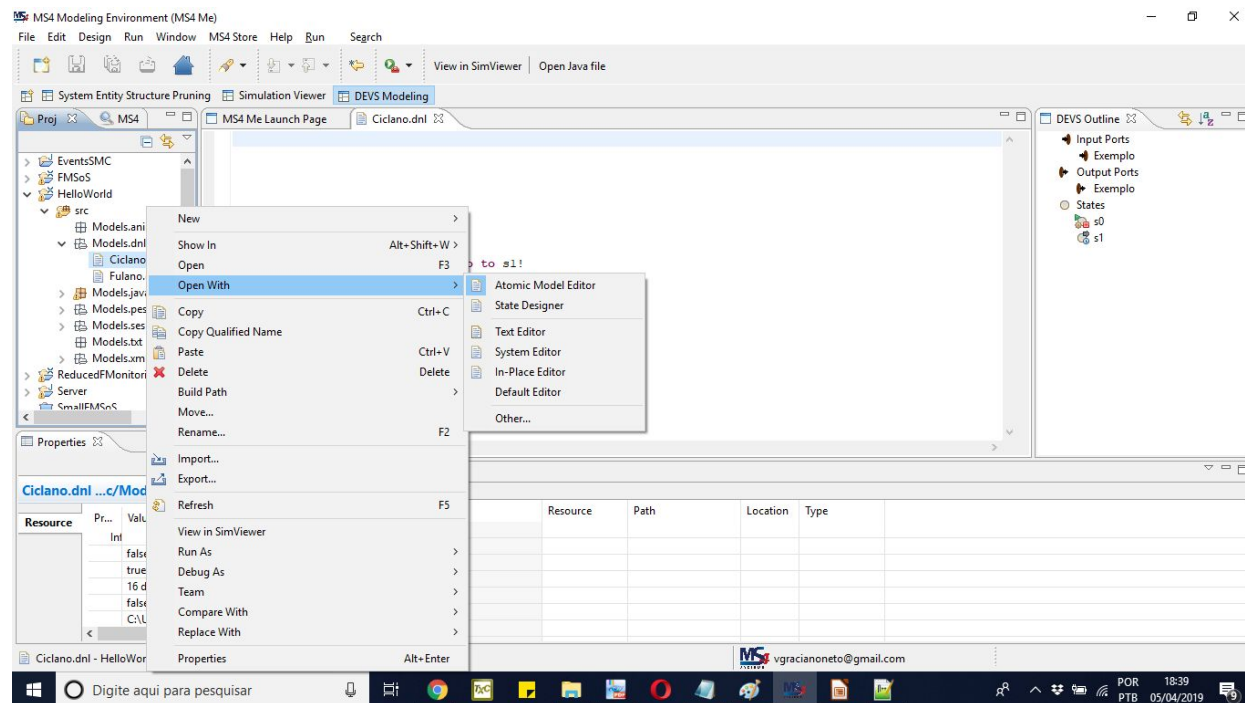
□ Write file on event in Atomic Models

```
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Date;
public class WriteFiles {
    private static final String path = "File path";
    public static void main(String[] args) throws IOException {
        File file = new File(path);
        long begin = System.currentTimeMillis();
        BufferedWriter writer = new BufferedWriter(new FileWriter(file));
        writer.write("Arquivo gravado em : " + new SimpleDateFormat("dd/MM/yyyy hh:mm:ss").format(new Date()));
        writer.newLine();
        writer.write("Path: " + path);
        writer.newLine();
        long end = System.currentTimeMillis();
        writer.write("Recording time: " + (end - begin) + "ms.");
        //Creating file content
        writer.flush();
        //Closing connection and file.
        writer.close();
        System.out.println("File stored in: " + path);
    }
}
```

Source:
<https://www.guj.com.br/t/er-e-escrever-arquivo-txt/84858/4>

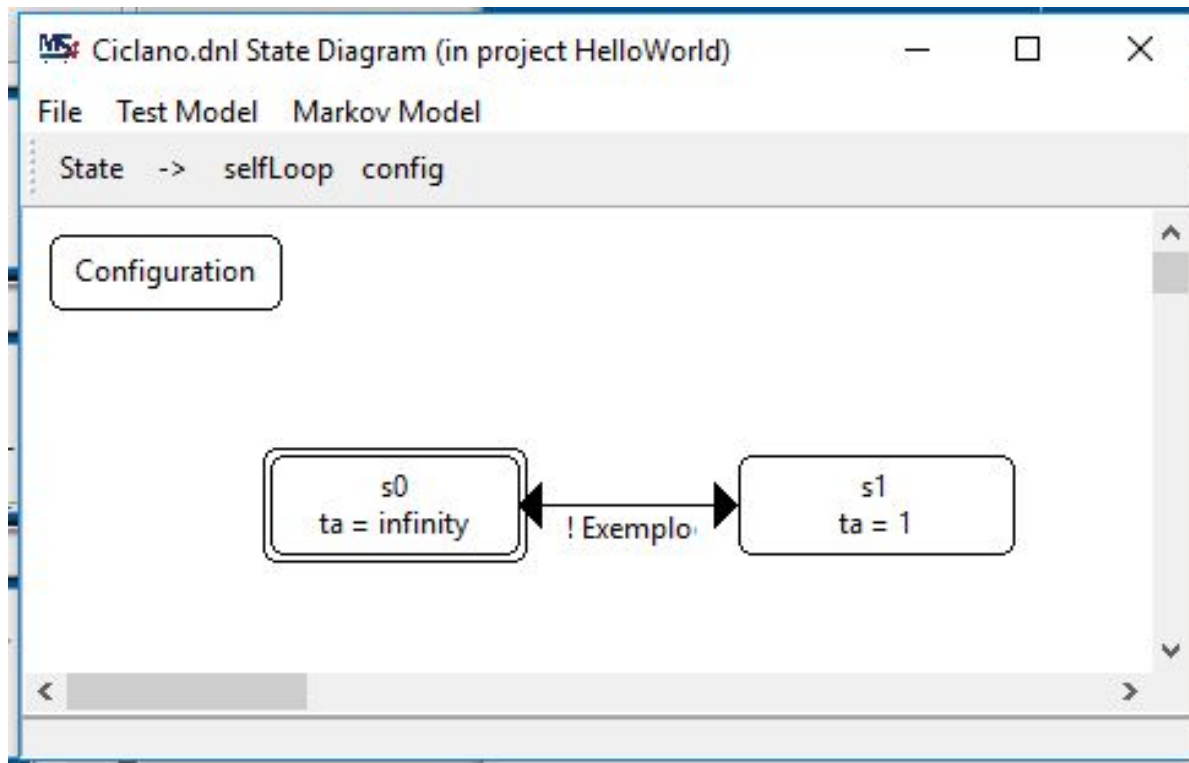
DEVS Practice

- See the Atomic Model state diagram
 - Right click on the .dnl and select Open With >> State Designer



DEVS Practice

- See the Atomic Model state diagram





DEVS Practice

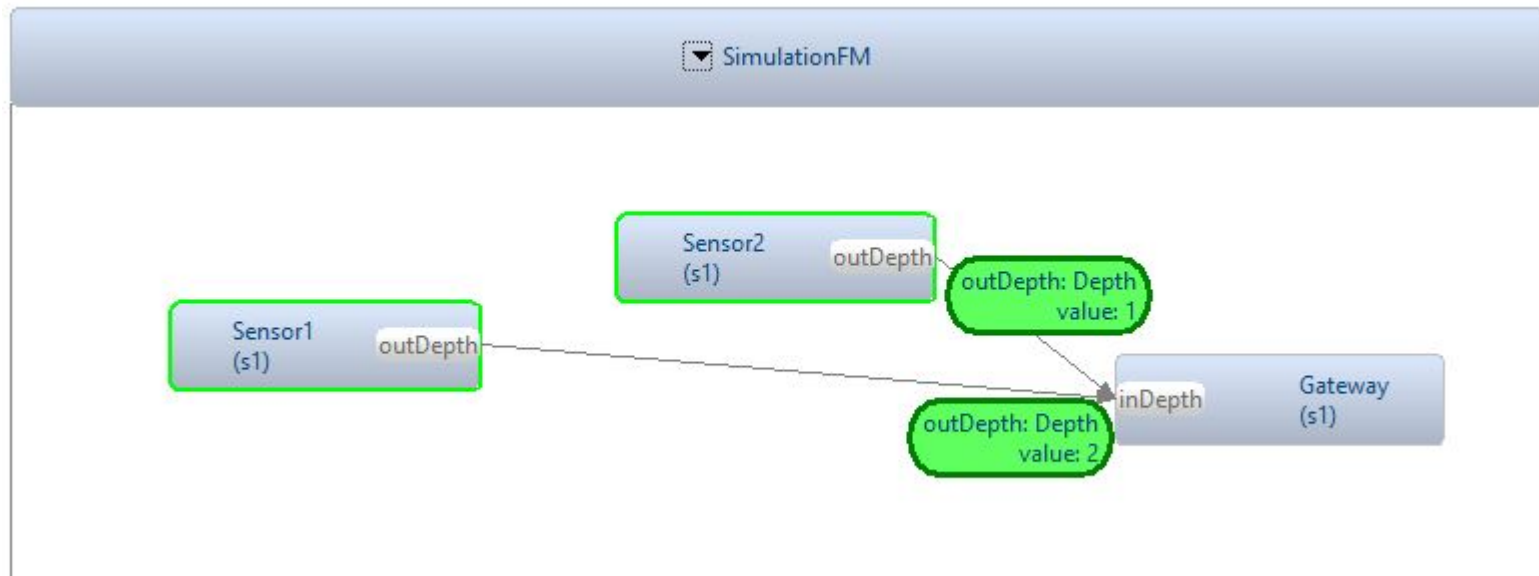
DEVS Practice



- Receive two data at the same time (Parallel DEVS or P-DEVS)
- It's actually trivial!
- Just for each external transition that receives data, iterate over the messageList and save/use the values!

DEVS Practice

Example:



DEVS Practice

□ Code of each Sensor (identical, except for value):

A Depth has a value!
the range of Depth's value is Integer!
use measureData with type Depth!

generates output on Depth with type Depth!

Initialize variables

```
<%  
java.util.Random gerador = new java.util.Random();  
measureData = new Depth(new Integer(1));  
%>!
```

to start hold in s0 for time 1!
after s0 output Depth!
from s0 go to s1!
hold in s1 for time 1!
from s1 go to s0!

output event for s0

```
<%  
output.add(outDepth, measureData);  
%>!
```

DEVS Practice



Gateway:

add library

```
<%  
import java.util.HashSet;  
%>!
```

A Depth has a value!
the range of Depth's value is Integer!

use measureData with type Depth!

accepts input on Depth with type Depth! //environment //sense

to start passivate in s0!
when in s0 and receive Depth go to s1!
hold in s1 for time 1!
from s1 go to s0!

external event for s0 with Depth

```
<%  
HashSet<Integer> set = new HashSet<Integer>();  
for(int i = 0; i < messageList.size(); i++){  
  
    Depth valueReceived = (Depth)messageList.get(i).getData();  
    int value = (int) valueReceived.getValue();  
    set.add(value);  
    System.out.println("Value received: " + value);  
}
```


DEVS Practice

□



.ses

From the top perspective, SimulationFM is made of
Sensor1, Sensor2, and Gateway!

From the top perspective, Sensor1 sends Depth to
Gateway!

From the top perspective, Sensor2 sends Depth to
Gateway!

Exercise



- 1) Replicate more sensors and make the uploaded content come from a file with multiple data. The output at the gateway must be an aggregate of the received data collections and must be written to a text file as well.