Graph Master Interface: a DEVS-Graph Web-Application

User Manual

Nirmal Patel, Raneem Abouseta, Prof. Cristina Ruiz-Martin, Prof. Gabriel Wainer

Introduction

Cadmium is a specialized tool designed for modeling and executing DEVS (Discrete Event System Specification) models. This software is particularly useful for presenting simulation outputs to users. Cadmium manages the execution of simulations, working with a user-friendly frontend application known as the Graph Master interface [1]. Cadmium processes the underlying calculations, while the Graph Master interface translates these results into an easily understandable format for users. This helps with some common challenges encountered in modeling and simulation such as adapting to model scalability, facilitating interaction between users and models, and ensuring access to high-quality data [1]. Such communication between components is essential, as misalignment can lead to models that do not fulfill the intended requirements [2].

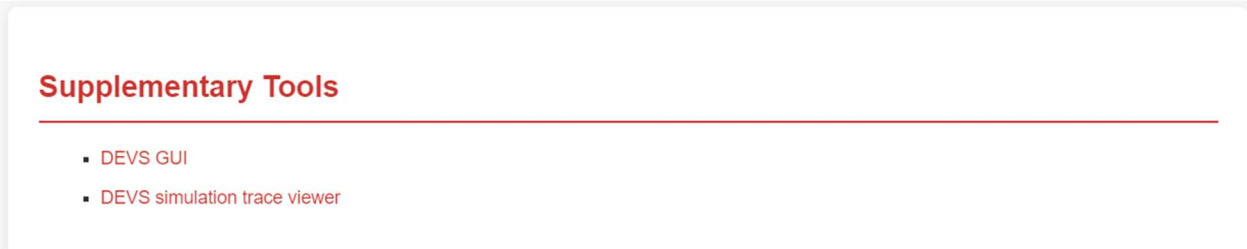
The DEVS Graph Master is a front end web application designed to convert C++ code into a state diagram, and a state diagram into C++ code. It provides remote design and execution although you can use the tool to generate your models locally. It is built using VanillaJS and HTML and is currently found on port 8000, served through an HTTP connection. DEVS-Graph is a graphical formalism to represent DEVS models. With the DEVS-Graph web-application users can draw the DEVS model using graphical user-interface, generate the JSON for the model so that it can be re-used in future. The users can also generate cadmium code for DEVS models from the DEVS-Graph.

To use the tool follow these steps:

1. Open this URL

https://devssim.carleton.ca/DEVS-Graph/

and look for:



The first item will bring you to the following URL:

https://devssim.carleton.ca/DEVS-Graph/

2. Use the three main components of the application

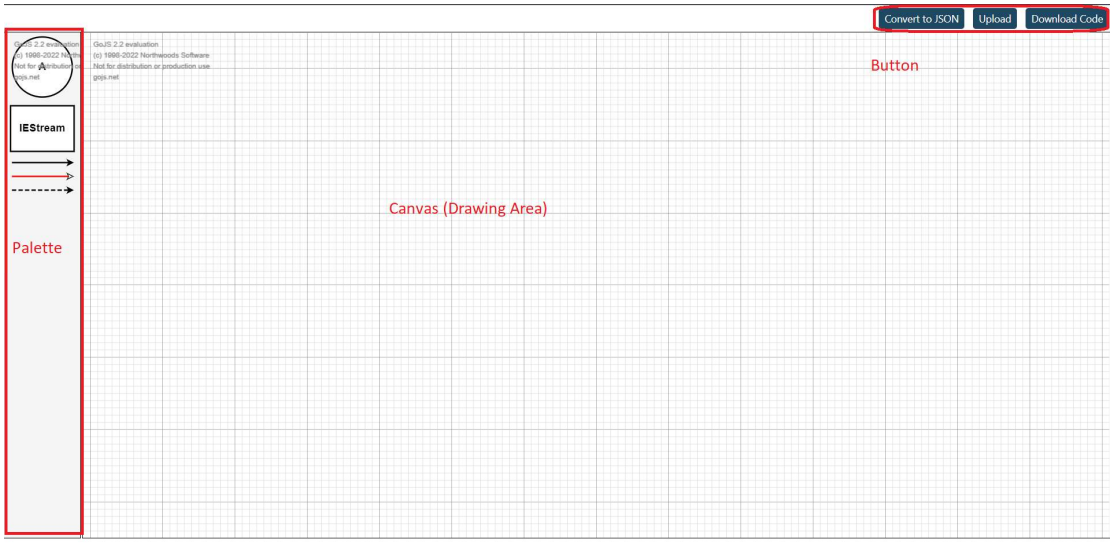


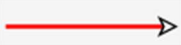
Figure 1: Overview of DEVS-Graph web-application

Palette: On left-hand side of the window, the palette is positioned. There are various elements in the palette which can be used to draw the DEVS model.

: The circle is used to define the state. The state name can be edited by clicking on the text inside the circle. By default, the circle has label “A”. This text can be multi-lined label as well.

: The square is used to indicate external input source to the model. To define the type of the input, click on the text and press enter to type the label.

: The solid line arrow is used for defining external transition in the atomic models. The transition condition can be edited by clicking on the label of the link.

 : The red arrow with white arrowhead is used for couplings between atomic and coupled models.

 : The dashed line arrow is used for defining internal transition in the atomic models. The transition output can be edited by clicking on the label of the link.

Canvas: The canvas or the drawing area is used for drawing the models. To draw the model, drag and drop different components from the palette window. Some of the short-cuts for the canvas are as follows,

Ctrl + A: Select all the components of the canvas

Ctrl + - : Zoom-out the canvas layout (makes the components look smaller)

Ctrl + +: Zoom-in the canvas layout (makes the components look bigger)

Ctrl + C: Copy the selected components

Ctrl + V: Paste the copied components

Ctrl + G: Group the selected component, puts the selected components in a square box.

Ctrl + Shift + G: Ungroup the selected components.

Buttons: There are different buttons on the right-top part of the webpage. The buttons functions are as follows,

- Convert to JSON: Converts the drawn model in canvas to a raw JSON, which will be downloaded in the user's machine. This JSON can be re-used to load the same model again in future.

- Upload: Upload JSON from the user's machine to draw the model defined in the JSON file.

- *FUTURE* Download Code: To generate cadmium code for the DEVS model click on the download code button which will prompt for a name for the project and upon filling the name and clicking on download will download a zip file containing all the source files needed to run the model.

- Export Screenshot: it allows to generate a graphic from the system to be used in documentation

- View Trace: it allows viewing execution results of a model

3. Executing an example

In the Appendix you will find a sample model; open any text editor and copy-paste the content of the file and save it as model.json

Open the tool, and click on the “load” button to upload the file created in step 1 (model.json).

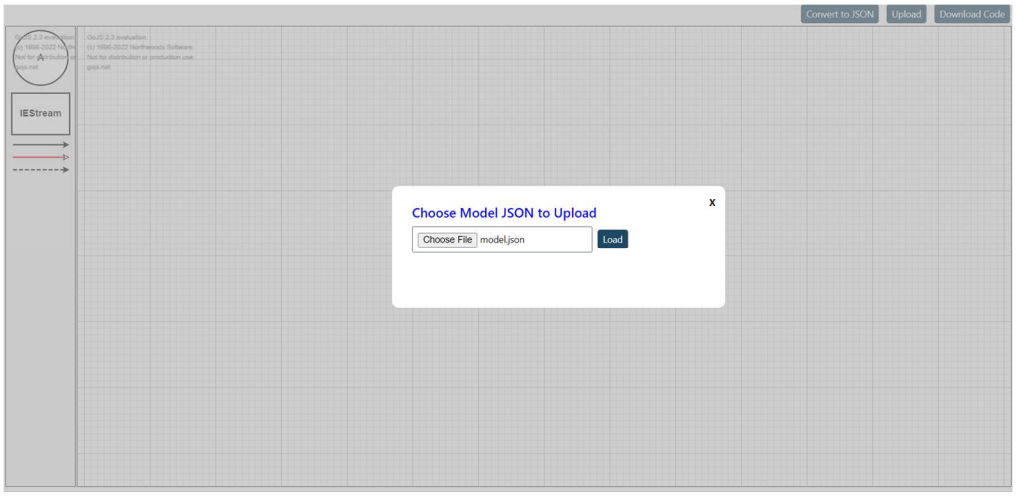


Figure 2: Upload file model.json to the web-application

Click on load and the drawn model will be loaded in the canvas.

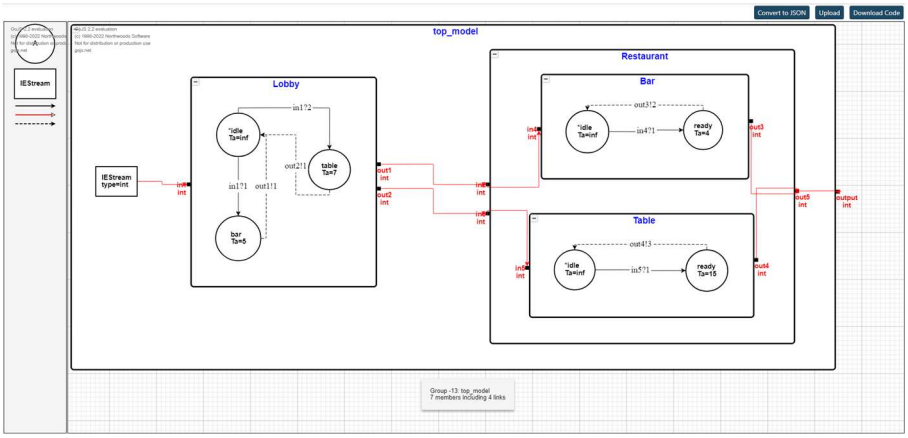


Figure 3: Canvas after loading model.json

To save the model, click on the “Convert to JSON” button, and save the file. Now it can be reloaded when needed.

References

[1] https://link.springer.com/article/10.1007/s11831-022-09794-9

[2] https://link.springer.com/article/10.1007/s10270-019-00773-6

Appendix: JSON example for the Hotel Lobby Coupled Model

{ "class": "GraphLinksModel",

"copiesArrays": true,

"copiesArrayObjects": true,

"linkFromPortIdProperty": "fromPort",

"linkToPortIdProperty": "toPort",

"modelData": {"position":"2112.238576250846 852.238728752538"},

"nodeDataArray": [

{"text":"*idle\nTa=inf","figure":"Ellipse","size":"94 94","fill":"white","key":-1,"loc":"2474.000000000001 1092.5058499991082","group":-3},

{"text":"bar\nTa=5","figure":"Ellipse","size":"98 97","fill":"white","key":-2,"loc":"2474.000000000001 1312.5058499991082","group":-3},

{"text":"Lobby","isGroup":true,"color":"blue","I":[{"id":"I1","text":"in1\nint","color":"red"}],"O":[{"id":"O1","text":"out1\nint","color":"red"},{"id":"O2","text":"out2\nint","color":"red"}],"key":-3,"group":-13},

{"text":"*idle\nTa=inf","figure":"Ellipse","size":"93 92","fill":"white","key":-4,"loc":"3216.500000000001 1085.8025975772794","group":-6},

{"text":"ready\nTa=4","figure":"Ellipse","size":"85 85","fill":"white","key":-5,"loc":"3462.500000000001 1082.3025975772794","group":-6},

{"text":"Bar","isGroup":true,"color":"blue","I":[{"id":"I1","text":"in4\nint","color":"red"}],"O":[{"id":"O1","text":"out3\nint","color":"red"}],"key":-6,"group":-11},

{"text":"*idle\nTa=inf","figure":"Ellipse","size":"89 87","fill":"white","key":-7,"loc":"3189.5 1379.417097577279","group":-9},

{"text":"ready\nTa=15","figure":"Ellipse","size":"91 90","fill":"white","key":-8,"loc":"3470.5 1380.917097577279","group":-9},

{"text":"Table","isGroup":true,"color":"blue","I":[{"id":"I1","text":"in5\nint","color":"red"}],"O":[{"id":"O1","text":"out4\nint","color":"red"}],"key":-9,"group":-11},

{"text":"table\nTa=7","figure":"Ellipse","size":"91 87","fill":"white","key":-10,"loc":"2668.000000000001 1166.5058499991087","group":-3},

{"text":"Restaurant","isGroup":true,"color":"blue","I":[{"id":"I1","text":"in2\nint","color":"red"},{"id":"I2","text":"in3\nint","color":"red"}],"O":[{"id":"O1","text":"out5\nint","color":"red"}],"key":-11,"group":-13},

{"text":"IEStream\ntype=int","figure":"Rectangle","size":"90 65","fill":"white","key":-12,"loc":"2215 1191.55","group":-13},

{"text":"top_model","isGroup":true,"color":"blue","I":[],"O":[{"id":"O1","text":"output\nint","color":"red"}],"key":-13}

],

"linkDataArray": [

{"points":[2474,1139.5058499991085,2474,1149.5058499991085,2474,1201.7558499991085,2474,1201.7558499991085,2474,1254.0058499991085,2474,1264.0058499991085],"color":"black","from":-1,"to":-2,"fromPort":"B","toPort":"T","label":"in1?1"},

{"points":[2523,1312.5058499991085,2533,1312.5058499991085,2533,1092.5058499991085,2532,1092.5058499991085,2531,1092.5058499991085,2521,1092.5058499991085],"dash_array":[6,3],"from":-2,"to":-1,"fromPort":"R","toPort":"R","label":"out1!1"},

{"points":[3263,1085.802597577279,3273,1085.802597577279,3341.5,1085.802597577279,3341.5,1082.302597577279,3410,1082.302597577279,3420,1082.302597577279],"color":"black","from":-4,"to":-5,"fromPort":"R","toPort":"L","label":"in4?1"},

{"points":[3462.5,1039.802597577279,3462.5,1029.802597577279,3216.5,1029.802597577279,3216.5,1029.802597577279,3216.5,1029.802597577279,3216.5,1039.802597577279],"dash_array":[6,3],"from":-5,"to":-4,"fromPort":"T","toPort":"T","label":"out3!2"},

{"points":[3234,1379.417097577279,3244,1379.417097577279,3329.5,1379.417097577279,3329.5,1380.917097577279,3415,1380.917097577279,3425,1380.917097577279],"color":"black","from":-7,"to":-8,"fromPort":"R","toPort":"L","label":"in5?1"},

{"points":[3470.5,1335.917097577279,3470.5,1325.917097577279,3189.5,1325.917097577279,3189.5,1325.917097577279,3189.5,1325.917097577279,3189.5,1335.917097577279],"dash_array":[6,3],"from":-8,"to":-7,"fromPort":"T","toPort":"T","label":"out4!3"},

{"points":[2474,1045.5058499991085,2474,1035.5058499991085,2668,1035.5058499991085,2668,1074.2558499991085,2668,1113.0058499991085,2668,1123.0058499991085],"color":"black","from":-1,"to":-10,"fromPort":"T","toPort":"T","label":"in1?2"},

{"points":[2668,1210.0058499991082,2668,1220.0058499991082,2596.4874954223633,1220.0058499991082,2596.4874954223633,1092.5058499991082,2531,1092.5058499991082,2521,1092.5058499991082],"dash_array":[6,3],"from":-10,"to":-1,"fromPort":"B","toPort":"R","label":"out2!1"},

{"points":[2777.261423749155,1155.0270277827572,2787.261423749155,1155.0270277827572,2888.7787348417323,1155.0270277827572,2888.7787348417323,1197.844600180241,2990.2960459343094,1197.844600180241,3000.2960459343094,1197.844600180241],"color":"red","arrow_color":"red","fill_arrow":"red","from":-3,"to":-11,"fromPort":"O1","toPort":"I1","label":""},

{"points":[2777.261423749155,1206.777924999554,2787.261423749155,1206.777924999554,2888.7787348417323,1206.777924999554,2888.7787348417323,1259.595497397038,2990.2960459343094,125

9.595497397038,3000.2960459343094,1259.595497397038],"color":"red","arrow_color":"red","fill_arrow":"red","from":-3,"to":-11,"fromPort":"O2","toPort":"I2","label":""},

{"points":[3004.2960459343094,1193.844600180241,3004.2960459343094,1203.844600180241,3113.238576250847,1203.844600180241,3113.238576250847,1149.0036994844404,3113.238576250847,1094.1627987886397,3113.238576250847,1084.1627987886397],"color":"red","arrow_color":"red","fill_arrow":"red","from":-11,"to":-6,"fromPort":"I1","toPort":"I1","label":""},

{"points":[3004.2960459343094,1263.595497397038,3004.2960459343094,1253.595497397038,3088.238576250846,1253.595497397038,3088.238576250846,1307.4145230928389,3088.238576250846,1361.2335487886396,3088.238576250846,1371.2335487886396],"color":"red","arrow_color":"red","fill_arrow":"red","from":-11,"to":-9,"fromPort":"I2","toPort":"I1","label":""},

{"points":[3564.761423749155,1067.2873501802412,3564.761423749155,1077.2873501802412,3564.761423749155,1217.844600180241,3613.6145782506846,1217.844600180241,3662.4677327522145,1217.844600180241,3662.4677327522145,1207.844600180241],"color":"red","arrow_color":"red","fill_arrow":"red","from":-6,"to":-11,"fromPort":"O1","toPort":"O1","label":""},

{"points":[3575.761423749154,1354.358100180241,3575.761423749154,1344.358100180241,3575.761423749154,1205.844600180241,3619.114578250684,1205.844600180241,3662.4677327522145,1205.844600180241,3662.4677327522145,1215.844600180241],"color":"red","arrow_color":"red","fill_arrow":"red","from":-9,"to":-11,"fromPort":"O1","toPort":"O1","label":""},

{"points":[2260.000000000001,1191.5499999999993,2270.000000000001,1191.5499999999993,2312.1192881254237,1191.5499999999993,2312.1192881254237,1197.7779249995544,2354.238576250846,1197.7779249995544,2364.238576250846,1197.7779249995544],"color":"red","arrow_color":"red","fill_arrow":"red","from":-12,"to":-3,"fromPort":"R","toPort":"I1","label":""},

{"points":[3666.4677327522145,1211.844600180241,3676.4677327522145,1211.844600180241,3755.1740417552746,1211.844600180241,3755.1740417552746,1212.594600180241,3755.1740417552746,1213.344600180241,3745.1740417552746,1213.344600180241],"color":"red","arrow_color":"red","fill_arrow":"red","from":-11,"to":-13,"fromPort":"O1","toPort":"O1","label":""}

]}