

Basic AToMPM usage

Yentl Van Tendeloo

November 7, 2014

This document will explain the basics on how to load, edit and save a model in AToMPM. While this introduction uses the SCCD example, the same general actions are used for other formalisms, such as DEVS, but just with different files to load (the DEVS model and DEVS exporter).

First of all, start Google Chrome and navigate to `http://localhost:8124/atomp`. The first time, you will be asked for a username and password. As you probably don't have an account yet, you have to create one. The server is running locally, so you can choose whatever username and password you want. After that, you should see the main AToMPM window as in Figure 1.

Now you can either create a new model, or open an existing model. For the SCCD assignment, you will be modifying an existing model, so open it by pressing the folder icon, as shown in Figure 2. Navigate to the folder `/Models/SCCD/` and load the `chatwindow.model` file, like in Figure 3. Afterwards, you should see approximately the same as in Figure 4. Notice that you now have the SCCD toolbar loaded too (Figure 5), which contains all Statechart constructs you will be using.

You can now start editing the model using the following commands (using the mouse):

- *left click* on an object in the SCCD toolbar and *right click* somewhere in the canvas to create the specified object.
- *left click* on an object in the canvas to select it. The following operations can be done using a selected object:
 - *drag* with the *left mouse button* to move the object.
 - *press the 'insert' key* to edit the selected object, this gives something similar to Figure 6.
 - *press the 'delete' key* to delete the selected object.
 - *press the 'shift' key* for lines to modify their route.
- *middle click* on an object in the canvas to edit its attributes. This is equivalent to selecting and pressing the *'insert'* key.
- *drag* with the *right mouse button* to create links between two objects, such as a transition. While dragging, you can press the *'control'* key to route the link differently.

You can always save the model. The browser will ask for confirmation if you want to leave AToMPM without having saved the current model. In the remainder of this document, code synthesis will be explained.

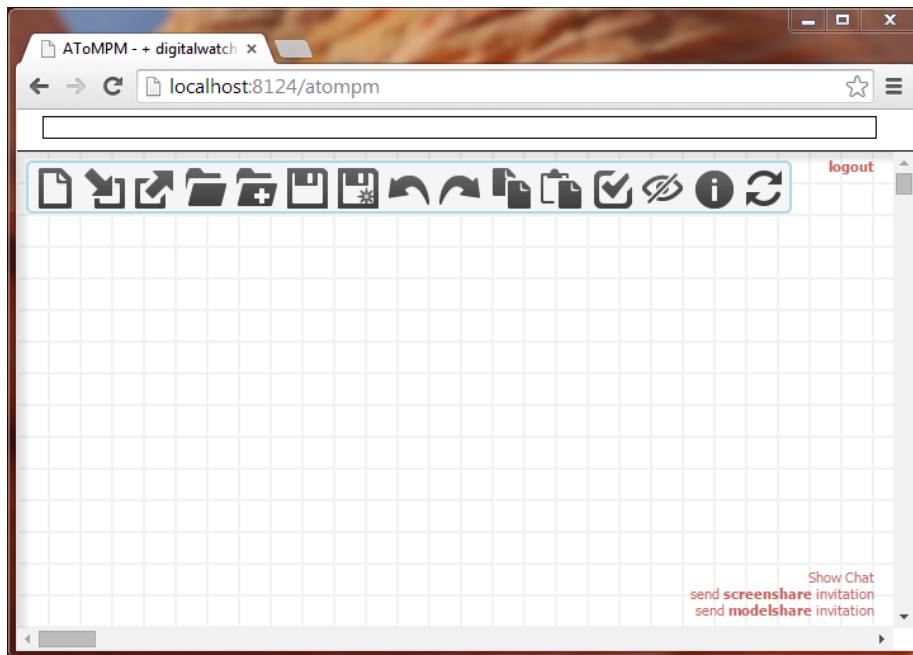


Figure 1: Main AToMPM window after startup

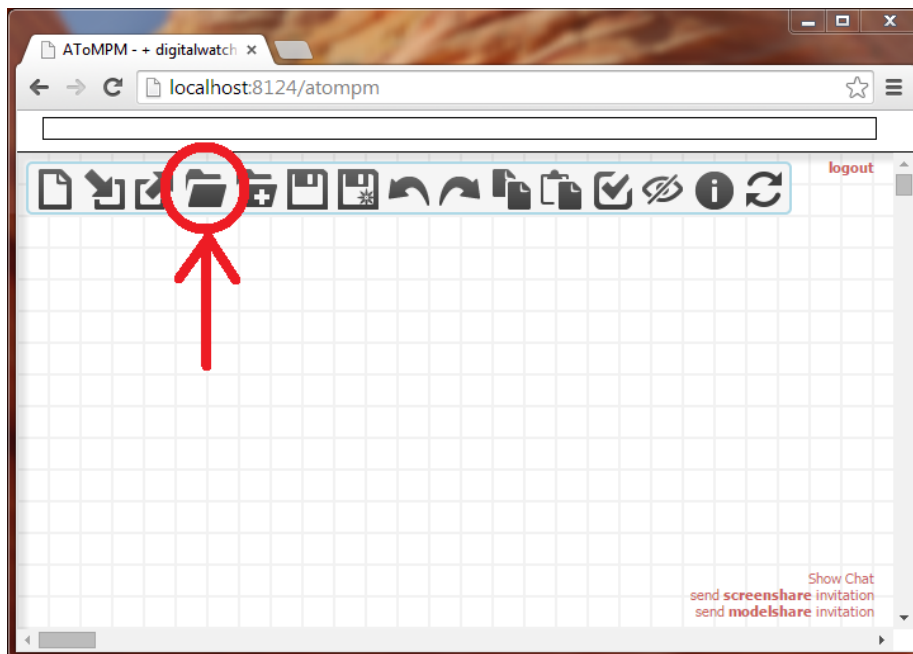


Figure 2: Open Model button

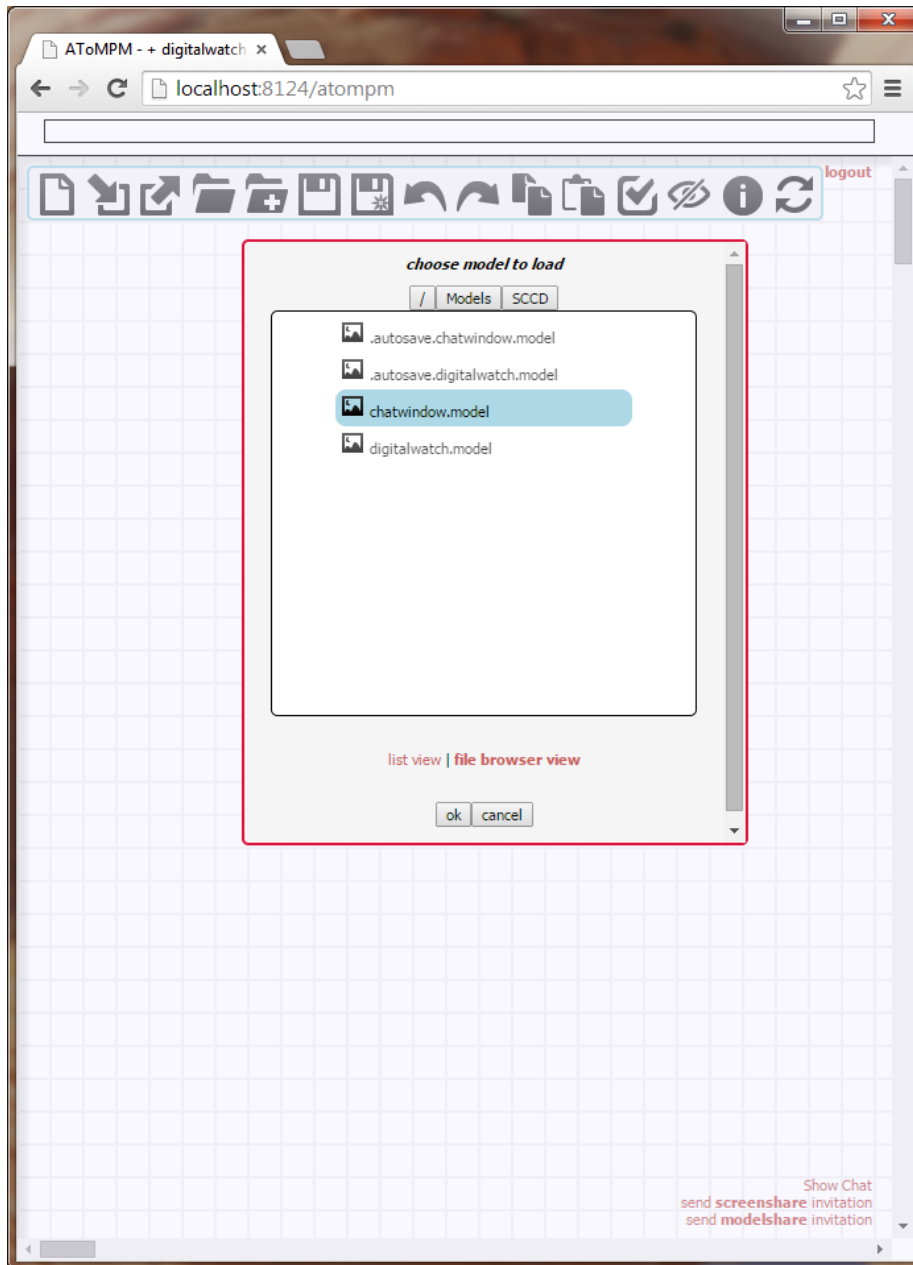


Figure 3: Open model /Models/SCCD/chatwindow.model

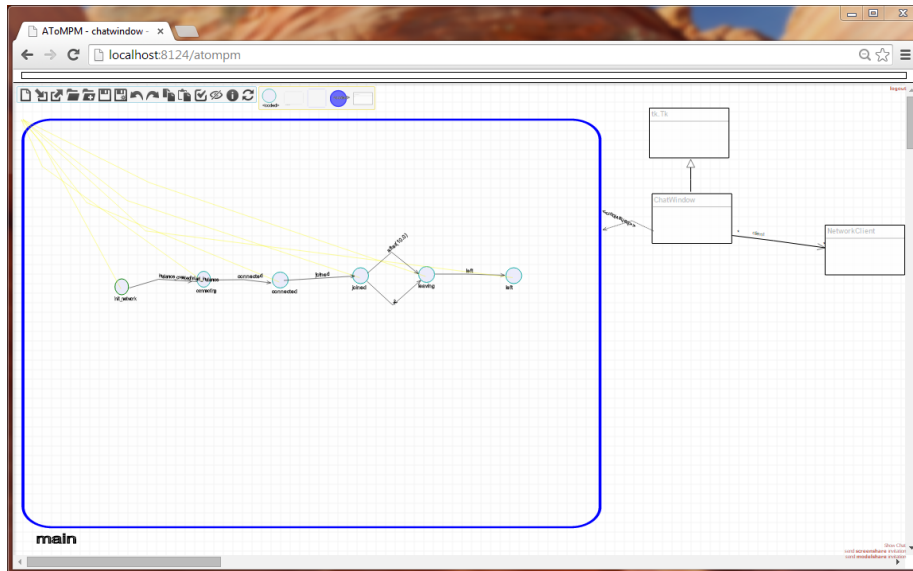


Figure 4: Model /Models/SCCD/chatwindow.model

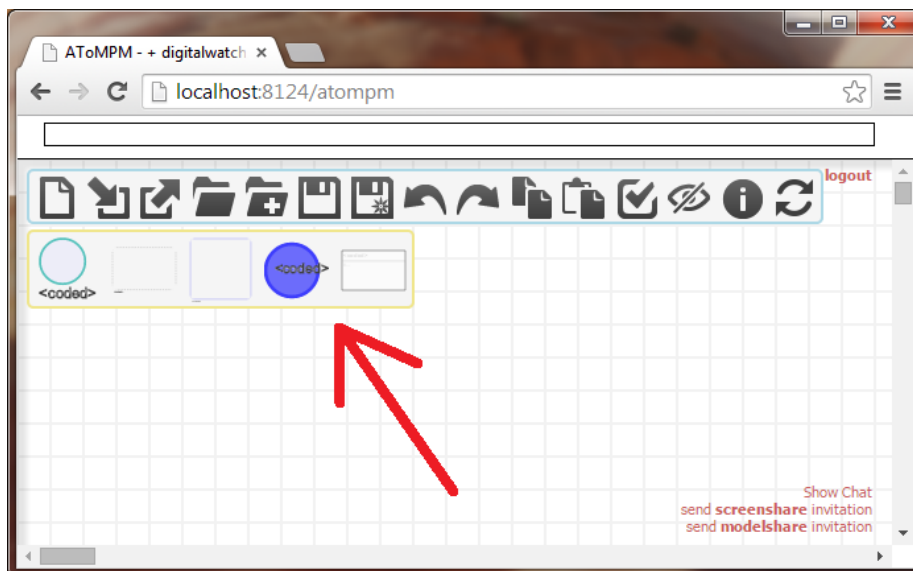


Figure 5: SCCD toolbar

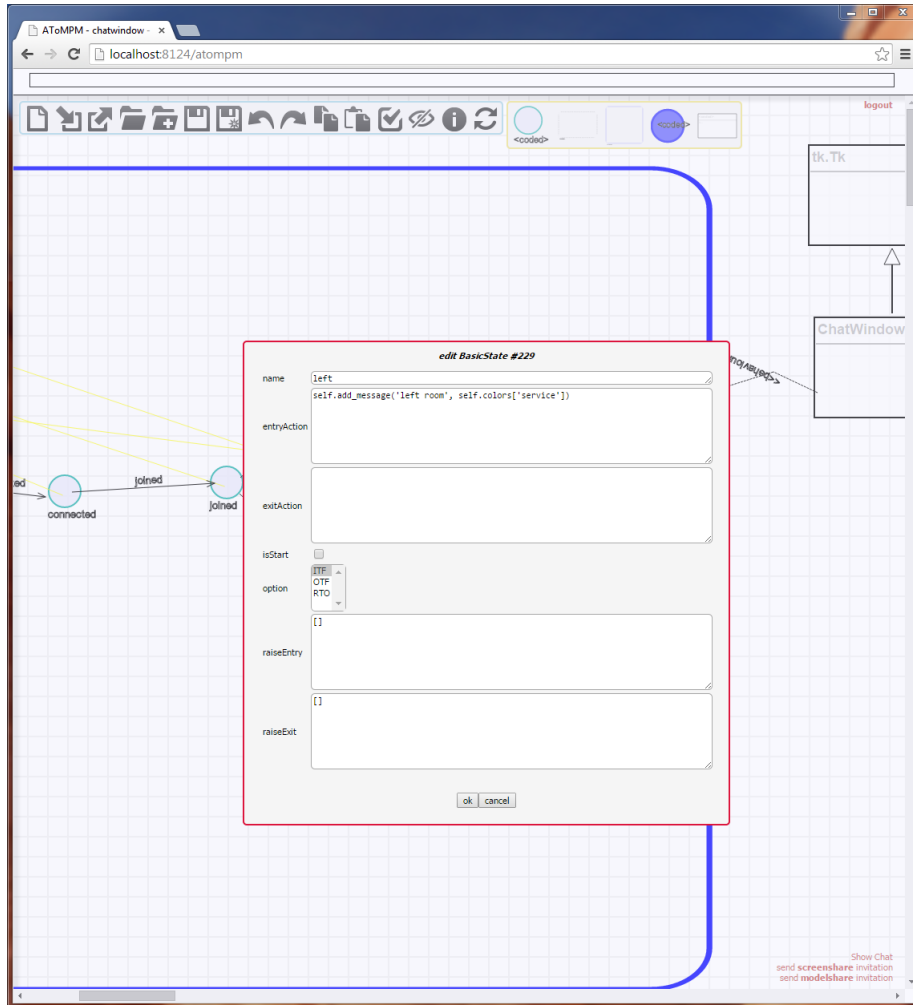


Figure 6: Edit an object

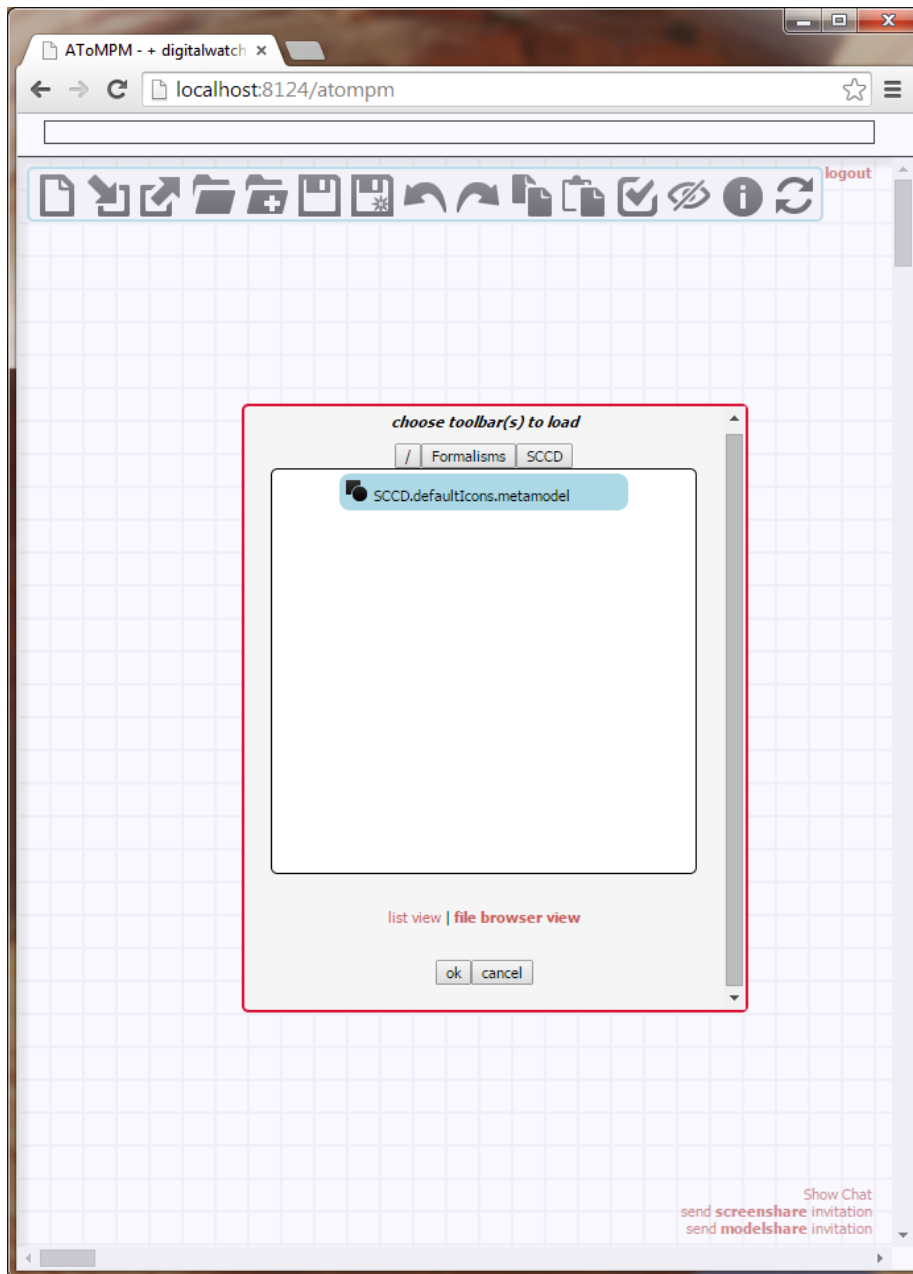


Figure 7: Open a toolbar

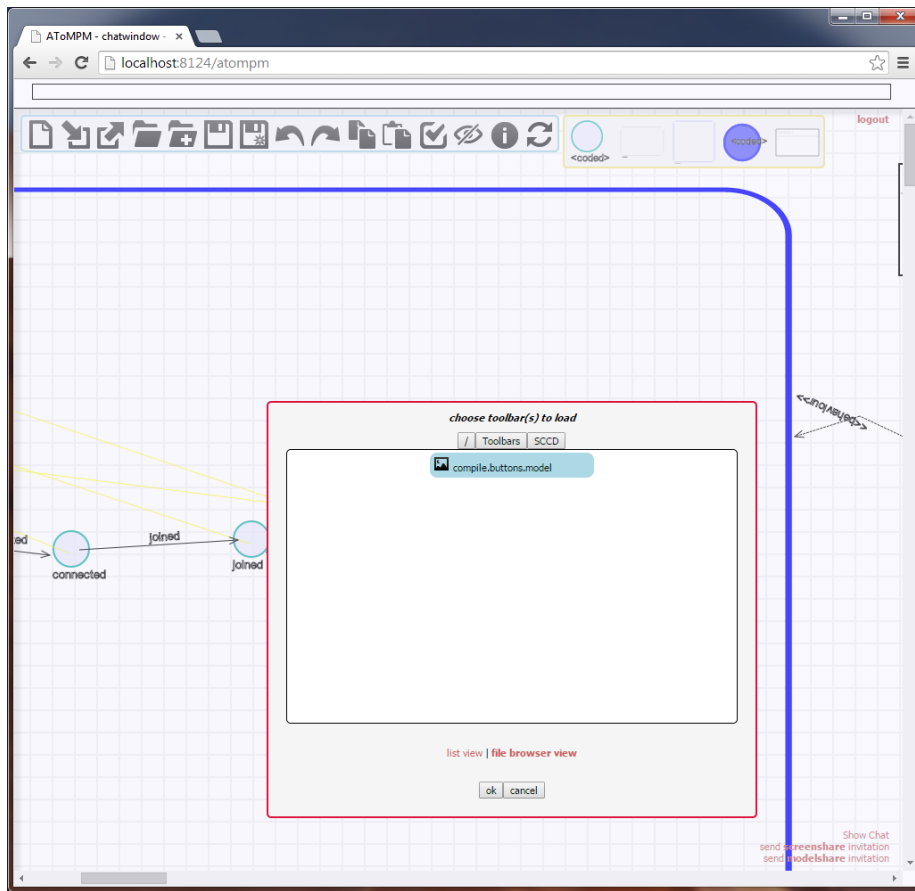


Figure 8: Toolbar /Toolbars/SCCD/compile.buttons.model

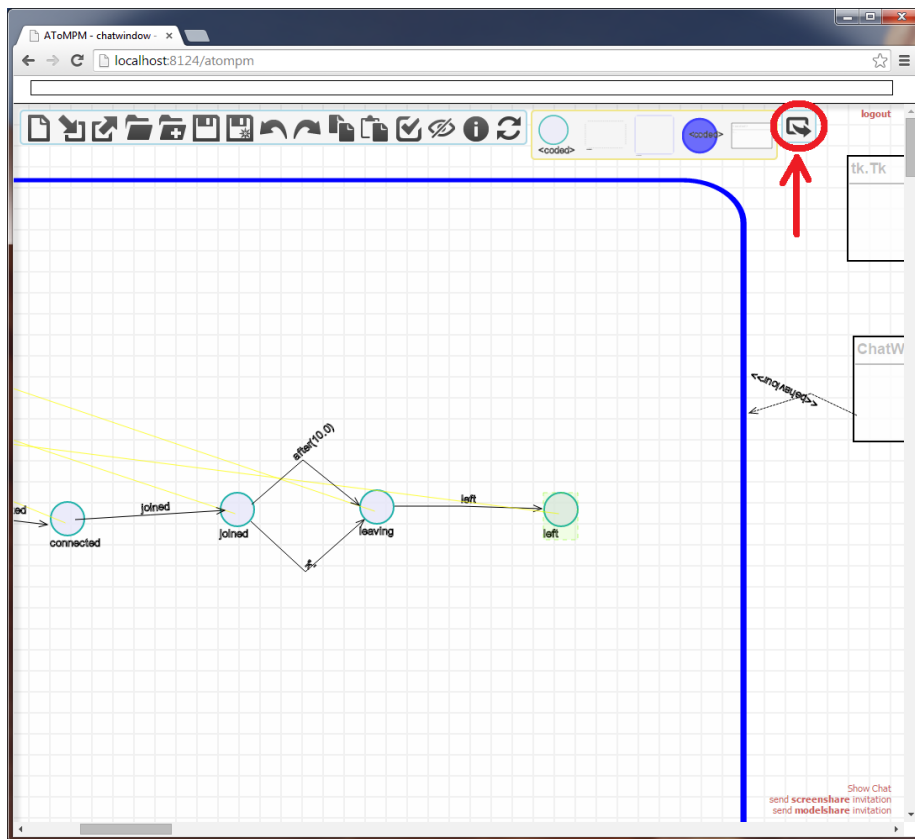


Figure 9: Export to SCCDXML

First, you will have to open an additional toolbar, by clicking the button circled in Figure 7. Now navigate to `/Toolbars/SCCD/` and select `compile.buttons.model` (Figure 8). This will give you a new (small) toolbar, as shown in Figure 9. Pressing this new button will generate the file `model.xml` in the folder `exported_to_sccdxml/classes/`.

Execute `python python_sccd_compiler/sccd.py -p gameloop chatclient.xml` to compile everything together in the file `chatclient.py`. Should the compiler generate some errors, you have probably generated an invalid model in AToMPM. You can now execute it by running `python run_client.py`.