

AToMPM installation

Yentl Van Tendeloo

October 26, 2015

The following sections will guide you to install AToMPM on a variety of platforms. These instructions deviate from those at <http://www-ens.iro.umontreal.ca/~syriani/atomp/atomp.htm> and should therefore not be combined. All referred links are local files, as AToMPM is quite fragile with different versions.

1 AToMPM

This section will install a basic AToMPM, as used in the course *Modeling of Software-Intensive Systems*.

1.1 Windows

1. Install node.js version 0.10.31 and make sure to select the ‘Add to path’ option during installation. An installer can be found <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/node-win.msi>
2. Download and unzip AToMPM from <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/atomp.zip>
3. Go into the extracted folder and open a terminal here. From your file manager, this can be done by right-clicking the *atomp* folder while holding the shift key. There will now be an option to open a command window there.
4. In the command window, execute the command `node httpwsd.js`. If all went well, this should say that socket.io has started. If it gives an error about `node` not being found, it probably isn’t added to your path correctly. In this case, you have to fully specify `node` using its complete path. This can be `"C:\Program Files\node.js\node.exe" httpwsd.js`, though the actual path might vary on your system.

1.2 Mac

1. Install node.js version 0.10.31 from <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/node-mac.pkg>
2. Download and unzip AToMPM from <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/atomp.zip>

3. Open a terminal in the unzipped folder and execute `node httpwsd.js` in this folder.

1.3 Linux

1. Install node.js version 0.10.31 using your package manager.
2. Download and unzip AToMPM from <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/atomp.zip>
3. Open a terminal in the unzipped folder and execute `node httpwsd.js` in this folder.

2 Transformation server

For model transformations, you will need to start the transformation server in addition to the node server. The previous section is thus a mandatory precondition.

2.1 Windows

1. Install Python 2.7 from <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/python-win.msi>
2. Install Python-igraph from <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/python-igraph-win.msi>.
3. Open a command window in the atomp folder (as previously) and execute `python mt\main.py`
4. If that command did not produce an error, the transformation server should be running.

2.2 Mac

1. If Python 2.7 is not pre-installed on your machine (it most likely is), install it from <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/python-mac.dmg>. Note that Python 3 is **NOT** the same. You can check your version by executing the command `python`.
2. Install Python-igraph from <http://msdl.cs.mcgill.ca/people/yentl/AToMPM/python-igraph-mac.dmg>
3. Open a terminal in the AToMPM folder (as previously) and execute `python mt/main.py`
4. If that command did not produce an error, the transformation server should be running.

2.3 Linux

Sadly, there are no binaries readily available of Python-igraph, making installation on linux slightly more difficult. If your package manager has python-igraph version 0.6.5 (**exactly that version!**), you might want to try installing that one first before following these steps.

1. Download the installation and compilation script from http://msdl.cs.mcgill.ca/people/yentl/AToMPM/run_mt.sh and place it in your AToMPM folder.
2. If you do not have a complete version of GCC (as by default on e.g. Ubuntu), you have to install that first. On Ubuntu, you will need the package `build-essential`.
3. Execute `./run_mt.sh`. This command will compile Zlib, Python, igraph and python-igraph. The first time it is executed, it will compile all this in `~/AToMPM`. When the script has finished (without an error), you can run the command again to start the transformation server. Should the script have failed for some reason, please manually remove the `~/AToMPM` folder. Note that you can **NOT** run the transformation server directly, as on Windows and Mac, because you have to start it with the compiled Python from the previous steps.