Installation Guide

The following programs are required for the setup:

- Eclipse Modeling Tools 2021-06 R
  - Plugins used in the MOMoT Framework (loaded by target platform + Xtext)
- Java 8
- Git
- Python (optional for plot generation)
  - Package manager for installing required packages

1 Installation of Eclipse Modeling Tools

- The Eclipse Installer 2021-06 R can be downloaded from here: https://www.eclipse.org/downloads/packages/

→ Download and run the Eclipse Installer for your operating system

→ Select Eclipse Modeling Tools (cf. Figure 1)

→ Choose any of the Java Runtime Environments (Java 8 is required in our project and will be downloaded and setup in 3.), a destination for installation, and continue to install (cf. Figure 2). The Eclipse Foundation Software User Agreement may need to be accepted in this course.

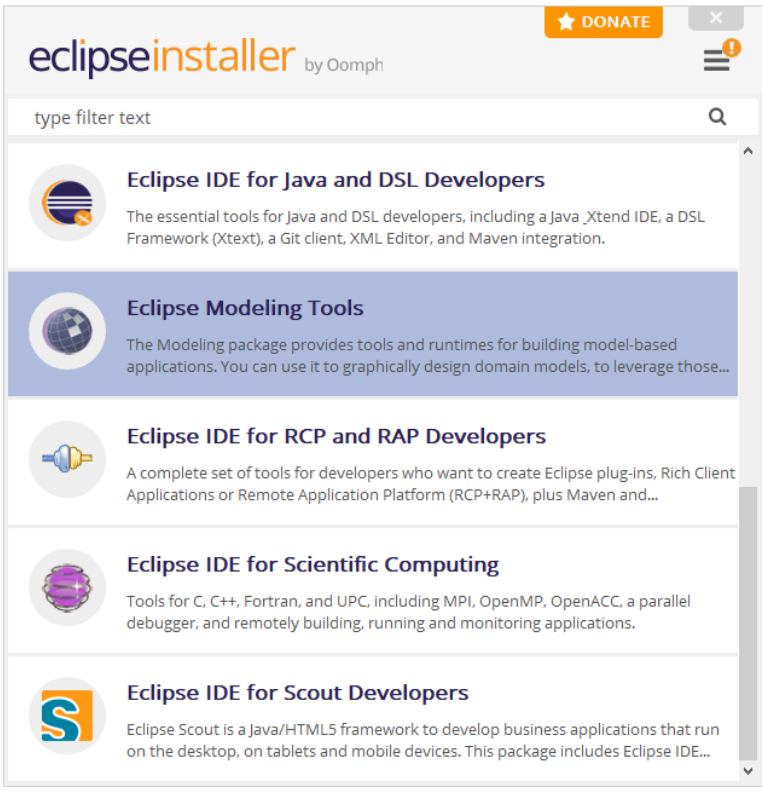

Figure 1

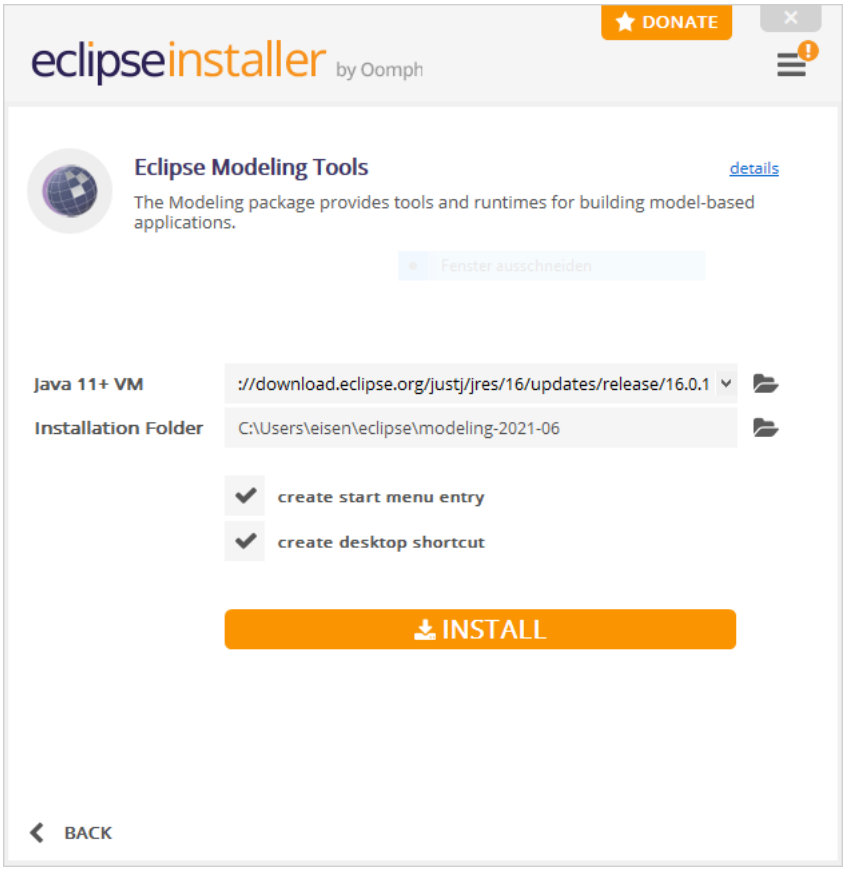

Figure 2

2 Installation of Java Development Kit 8

The MOMoT framework prerequisites Java 8 in some parts, e.g., to generate the grammar for defining search configurations.

- The Java Development Kit 8 can be downloaded here (requires free oracle account): https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html

→ Download the JDK 8 for your OS

→ Run the .exe and follow the installation of JDK 8

  - In the starting window, press “Next” to start the installation process
  - If required, change the destination for installing JDK 8 (cf. Figure 3). Continue with “Next”.
  - When finished, the setup will prompt for installing the Java Runtime Environment. Change the destination for installing by desire and continue.

→ Alternatively, on Linux OS, download as Compressed Archive and unzip at a desired location.

→ By default, Eclipse Modeling Tools uses the Java Runtime Environment selected during the installation of Eclipse for new projects. This will be changed before buildings MOMoTs language artifacts in Section 3.

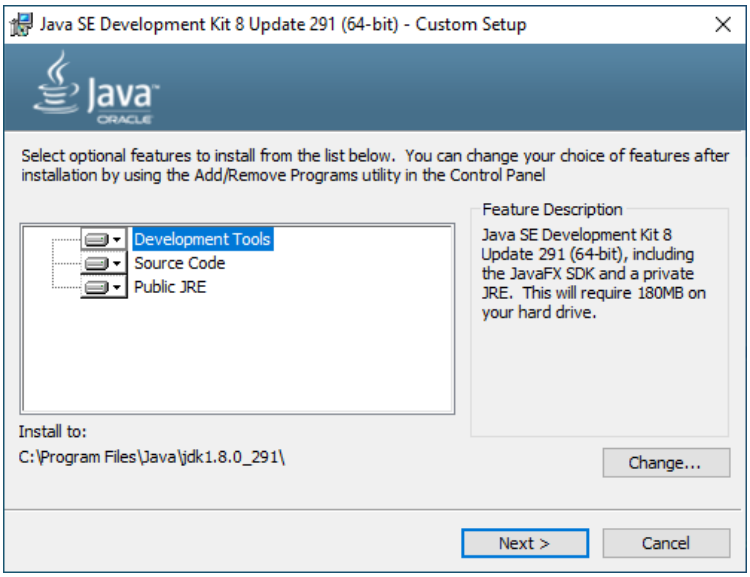

Figure 3

3 Importing and setting up the project into Eclipse workspace

The following steps describe the import process such that the dependencies of all plugins are installed and setup in the correct way.

- The project is publicly available at https://github.com/rl4mt/rl4mt

→ Clone the project
→ Launch Eclipse Modeling Tools and launch a new workspace
→ Import the project into your workspace:
  - In Eclipse top menu, select “File” -> ”Import...”. Choose the import wizard “Projects from Git” (cf. Figure 4).
  - As Repository Source, choose “Existing local repository”.
  - Click “Add...”. In the popup select “Browse...”, navigate to the project directory (project clone destination) and press “Search”. Then check the project entry, press “Finish” and “Next” in the import wizard.

- Select “Import existing Eclipse projects” and “Next” (cf. Figure 5).

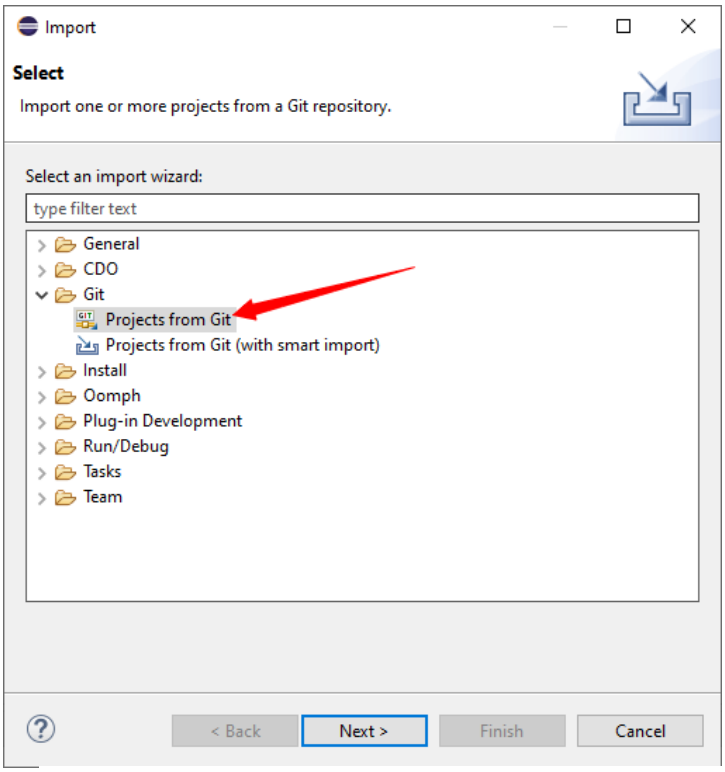

Figure 5

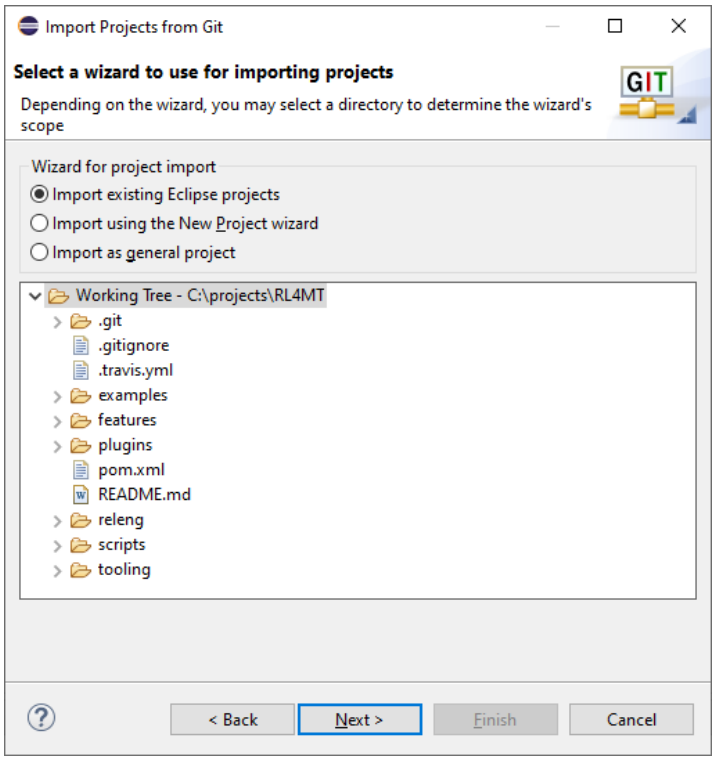

Figure 4

- Check “Search for nested projects” and select all projects in the list and press “Finish” (cf. Figure 6)

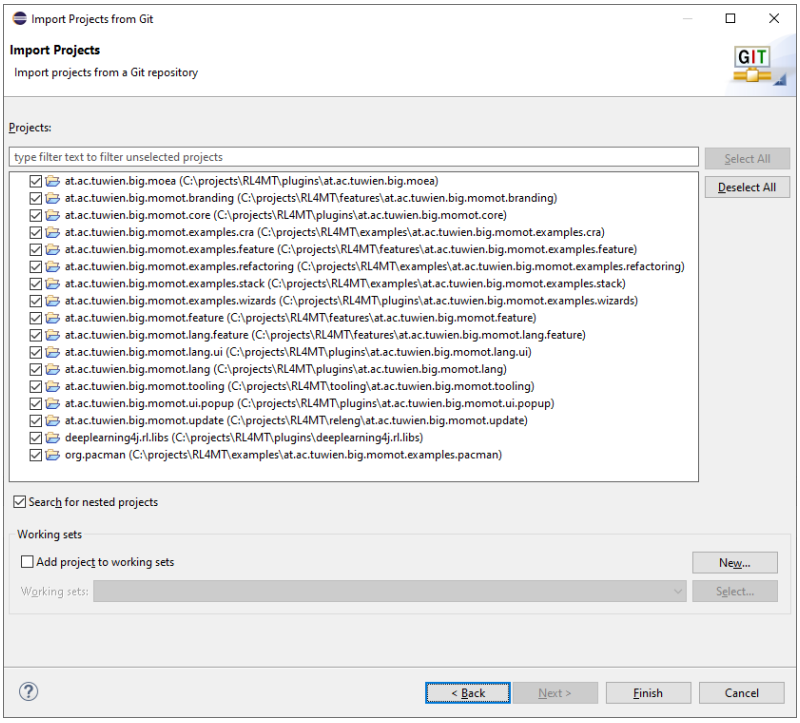

Figure 6

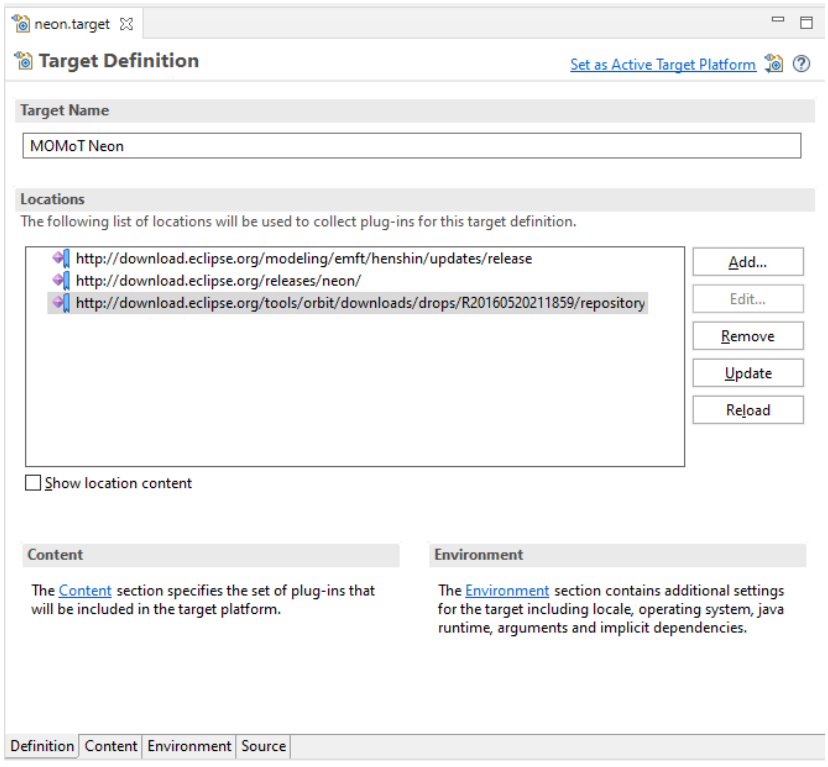

Figure 7

→ After the import is finished, the active target platform needs to be set:
- In project “at.ac.tuwien.big.momot.tooling” open “neon.target” in the folder “targetplatform” with the target editor.
- Press “Set as Active Target Platform” (cf. Figure 7). Wait for Eclipse to build the project as the target dependencies are loaded and updated. Note that the progress is shown in the right lower corner in Eclipse.

→ Installing XText framework:
- In Eclipse top menu, go to “Help”->”Eclipse Marketplace”. In the search box enter “xtext” and install Eclipse Xtext (cf. Figure 8).
- Accept the user agreement and press “Finish”.
- Upon finish, restart Eclipse as will suggested by the IDE.
- Errors may remain in plugins at this point, e.g., in “at.ac.tuwien.big.momot.lang” and “at.ac.tuwien.big.momot.lang.ui”. Select “Project” -> “Clean” in Eclipse top menu, check “Clean all projects” option and press “Clean”. Wait until the cleaning process is finished as indicated in lower right corner of Eclipse.

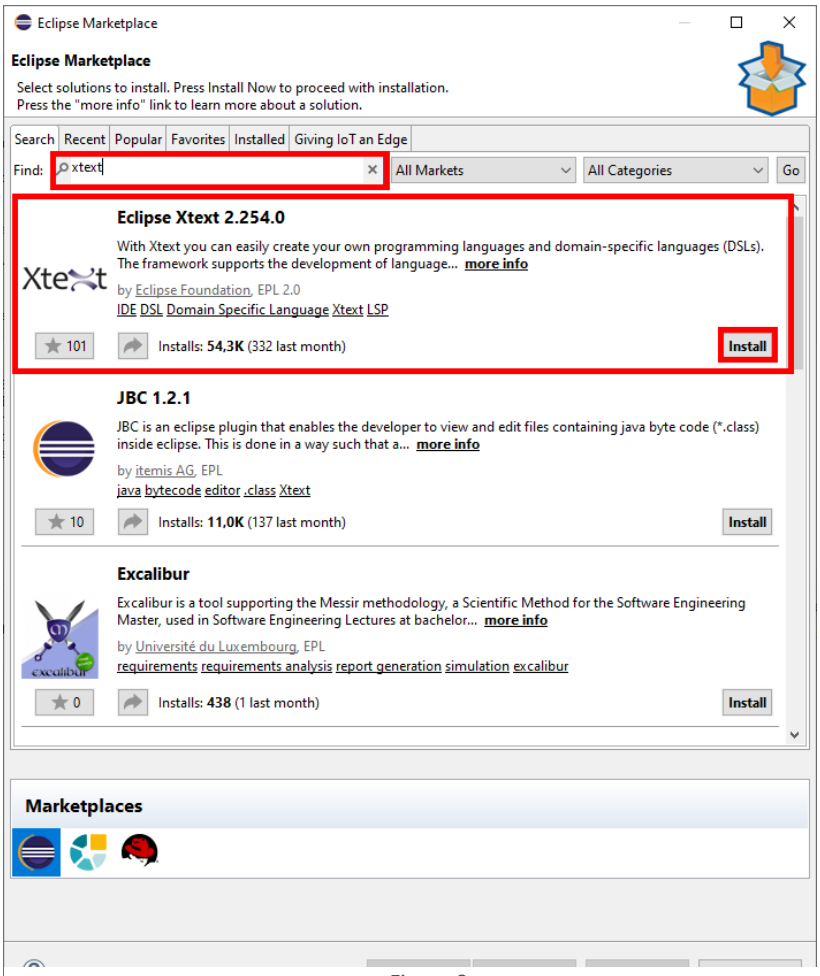

Figure 8

→ Further steps such as the generation of language artifacts require JRE 8. Therefore, the default Runtime Environment used by the IDE is set to version 8 as follows:

- In Eclipse, go to “Window” -> “Preferences”. Open the “Java”-tree and click on “Installed JREs”.
- Beside the list of Installed JREs, click “Add…” and select “Standard VM” as type and “Next”
- For “JRE home”, click “Directory” and navigate to the destination chosen when installing Java 8 and select the “bin”-subfolder, e.g., “C:\Program Files\Java\jre1.8.0_291\bin”. The corresponding sources should be listed as “JRE system libraries” (cf. Figure 9). Click on “Finish”.
- The just added JRE should be listed now with the “Installed JREs”. Check the JRE and press “Apply and Close” (cf. Figure 10).

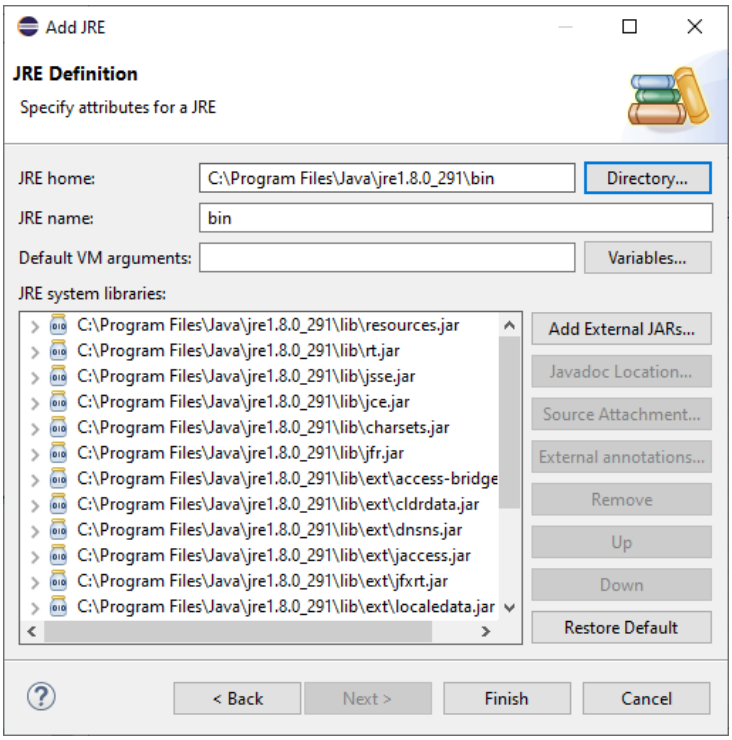

Figure 9

→ Building MOMoTs language artifacts:

→ Open “at.ac.tuwien.big.momot.lang” plugin, navigate into the source package “at.ac.tuwien.big.momot.lang” and right click on “MOMoT.xtext”. Select “Run As” and choose “Generate Xtext Artifacts” (cf. Figure 11). Note that this requires Java 8 to be used for successful execution. A successful execution results in info output on the console saying “[main] INFO .emf.mwe2.runtime.workflow.Workflow – Done.”

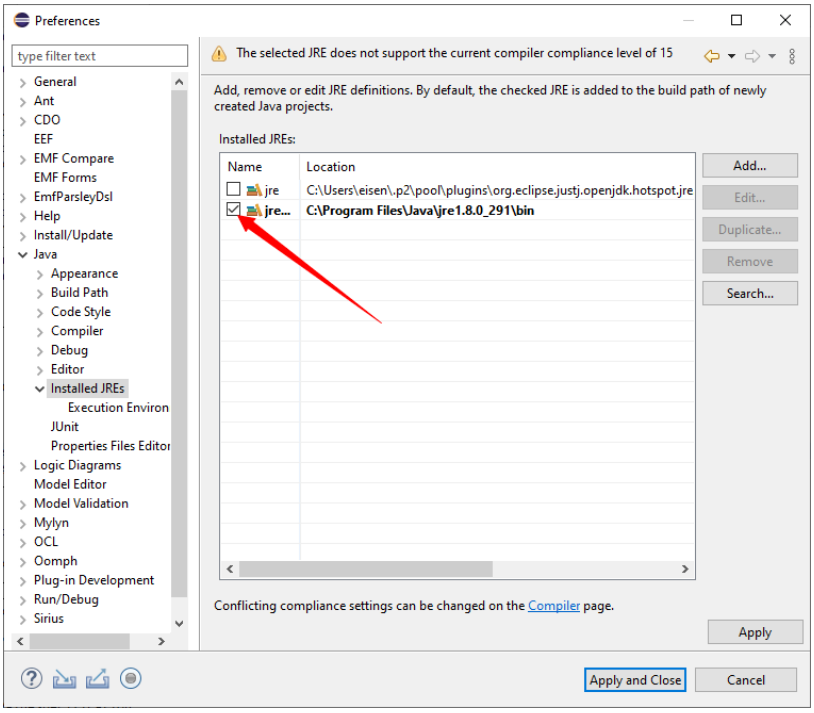

Figure 10

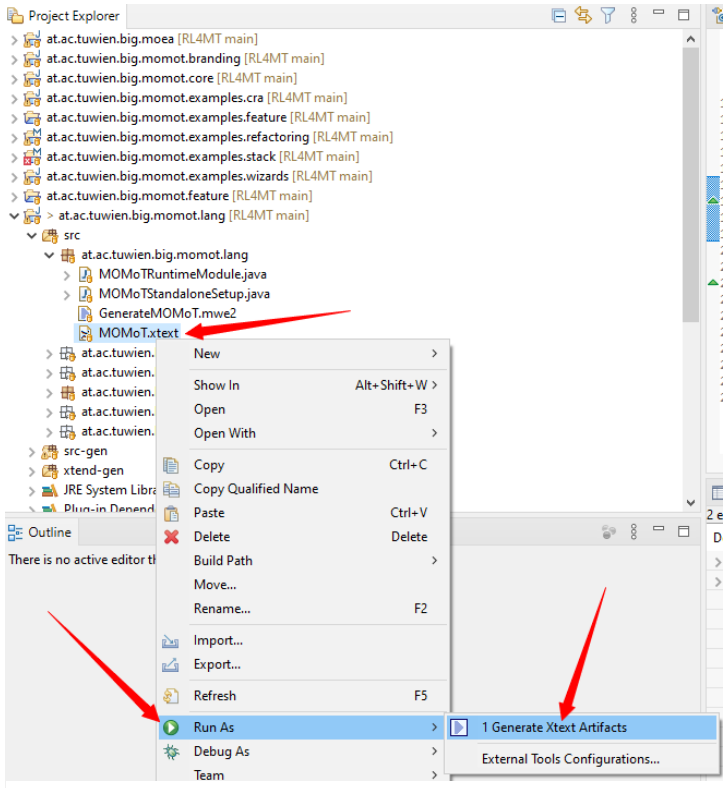

Figure 11

→ The project setup is finished and should be usable to run the case studies. Therefore, execution details are given in file “readme.md”. To test the installation, e.g., refer to the quick example given in section “Example: Stack Load Balancing”.

4 Installing Python and setup for plot generation (optional)

To generate a plot of average rewards after training RL agents in the Pacman example, Python is required. The steps for generating the plot can be found in “readme.md”. Note that this is not required for running any of the case studies but to recreate the plot of average rewards as depicted in the paper.

→ The latest version of Python can be downloaded at http://python.org/downloads

4.1 Installing Python

The following steps guide through the python installation process. If you have Python already installed and added to the Environment Variables, proceed with 4.2.

→ Download Python for your OS
- Start the installer. In the setup window, choose “Customize Installation” (cf. Figure 12)
- Uncheck all features except for “pip” (python package installer) and go “Next”
- Check options “Add Python to environment variables” and “Precompile standard library” (cf. Figure 13), adjust the installation destination by desire and “Install”.
- Upon finish, python should be installed correctly. This can be checked by opening the command line and executing “python” command which will start the python runtime environment.

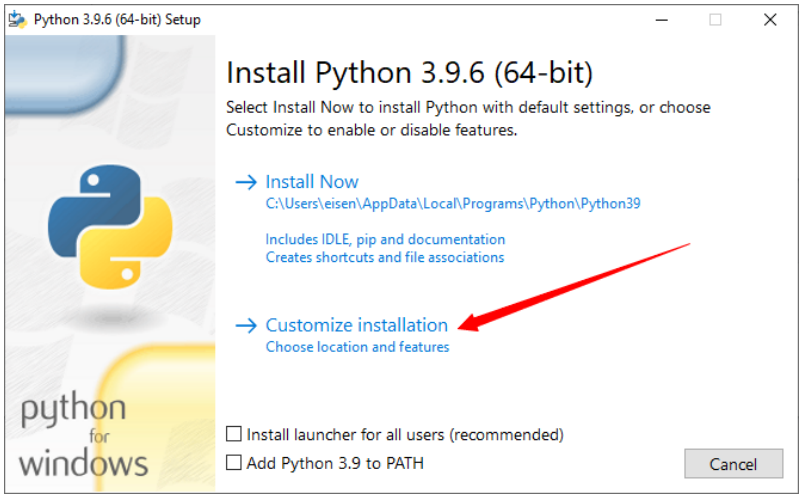

Figure 12

- Alternatively, on Linux OS, most distributions naturally include python 3. This can be checked by starting the python runtime environment running “python3” in bash. To install the package manager, the command “sudo apt install python3-pip” (Debian) can be used.

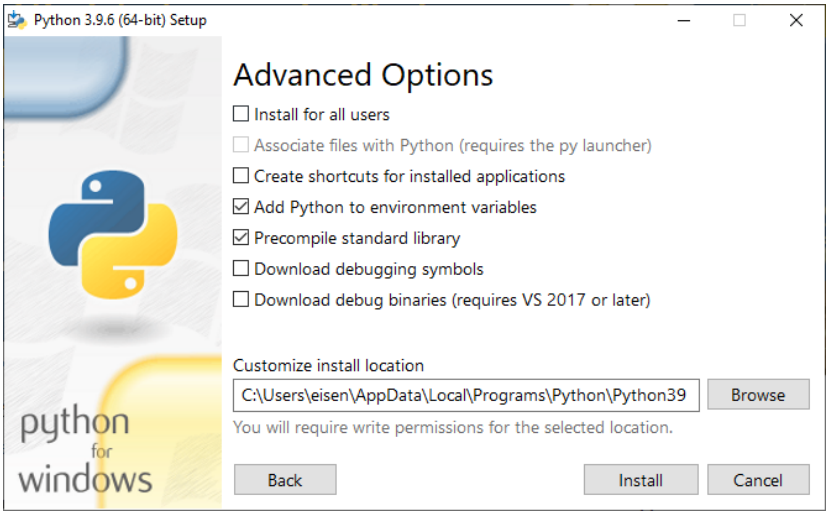

Figure 13

4.2 Installing required packages

Required packages for requiring plots are listed in “requirements.txt” in the Pacman example project in subfolder “plot_util”. They can be installed with the following command in the command line tool, assuming “plot_util” to be the current working directory: “pip install -r requirements.txt”. Necessary packages for plotting will be installed and enable using the “generate_plot.py” script to generate plots of the average returned rewards as described in “readme.md”