

CNI-OS³

The Open Source Satellite Simulator

Installation Guide

Marian P. Felder, B. Sc.
Dennis Kaulbars, B. Sc.
cand.-Ing. Thomas Klute
Sarah Lehnhausen, B. Sc.
Daniel Merget, B. Sc.
Florian Schweikowski, B. Sc.
Lisa Underberg, B. Sc.

December 2, 2012

Version 1.0

TU Dortmund University
Communication Networks Institute
Prof. Dr.-Ing. C. Wietfeld
Otto-Hahn-Strasse 6
44227 Dortmund
Germany

Contents

1	Supported Operating Systems	3
2	Requirements	3
3	Windows Operating Systems	3
3.1	Windows - Checklist	15
4	Linux (Ubuntu/Debian)	16
4.1	Linux - Checklist	23
5	Mac OS X	24
5.1	Mac OS X - Checklist	32
6	Registration process for live data integration	33
6.1	Weather data	33
6.2	Altitude data	33

1 Supported Operating Systems

- Windows (Windows XP, Windows 7)
- Linux
- Mac OS X Lion (or newer)

2 Requirements

- OMNeT++ (tested with version 4.2.2) - <http://www.omnetpp.org>
- INET Framework for OMNeT++ (version 20111118) - <http://www.inet.omnetpp.org>

Important:

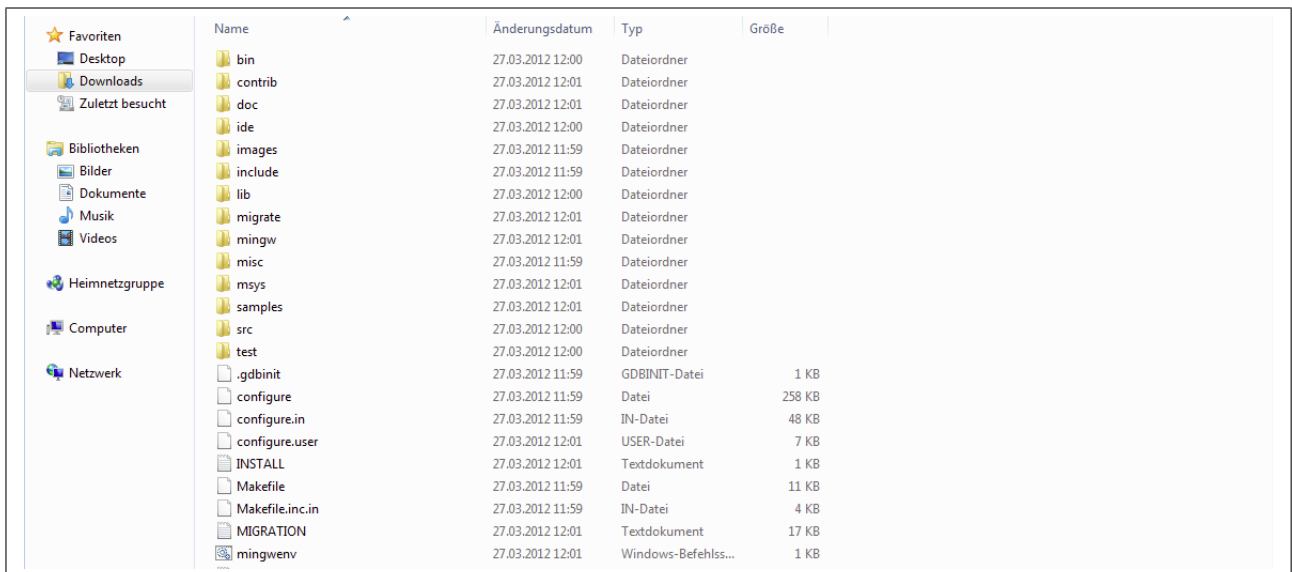
OS³ only works with this version of the INET framework! A port is scheduled to be released in December 2012.

- cURL (tested with version 7.27) - <http://curl.haxx.se>
Important: Take care of choosing the sources instead of the binaries for downloading!

3 Windows Operating Systems

1. Installation of OMNeT++

- Download *OMNeT++* (tested with version 4.2.2) from <http://www.omnetpp.org>. Be sure to choose the version for *Windows* operating systems **including** *MinGW32* environment.
- Extract the archive (e.g. with *WinRAR*) into a folder on your hard disk. After the extraction, your folder structure in the destination directory should look like in Figure 1.



Name	Änderungsdatum	Typ	Größe
bin	27.03.2012 12:00	Dateiordner	
contrib	27.03.2012 12:01	Dateiordner	
doc	27.03.2012 12:01	Dateiordner	
ide	27.03.2012 12:00	Dateiordner	
images	27.03.2012 11:59	Dateiordner	
include	27.03.2012 11:59	Dateiordner	
lib	27.03.2012 12:00	Dateiordner	
migrate	27.03.2012 12:01	Dateiordner	
mingw	27.03.2012 12:01	Dateiordner	
misc	27.03.2012 11:59	Dateiordner	
msys	27.03.2012 12:01	Dateiordner	
samples	27.03.2012 12:01	Dateiordner	
src	27.03.2012 12:00	Dateiordner	
test	27.03.2012 12:00	Dateiordner	
.gdbinit	27.03.2012 11:59	GDBINIT-Datei	1 KB
configure	27.03.2012 11:59	Datei	258 KB
configure.in	27.03.2012 11:59	IN-Datei	48 KB
configure.user	27.03.2012 12:01	USER-Datei	7 KB
INSTALL	27.03.2012 12:01	Textdokument	1 KB
Makefile	27.03.2012 11:59	Datei	11 KB
Makefile.inc.in	27.03.2012 11:59	IN-Datei	4 KB
MIGRATION	27.03.2012 12:01	Textdokument	17 KB
mingwenv	27.03.2012 12:01	Windows-Befehls...	1 KB

Figure 1: Structure of the OMNeT++ folder

- Open the **MINGW32** environment by executing the *mingwenv.exe* file. After that, a command shell window will open (see Figure 2).

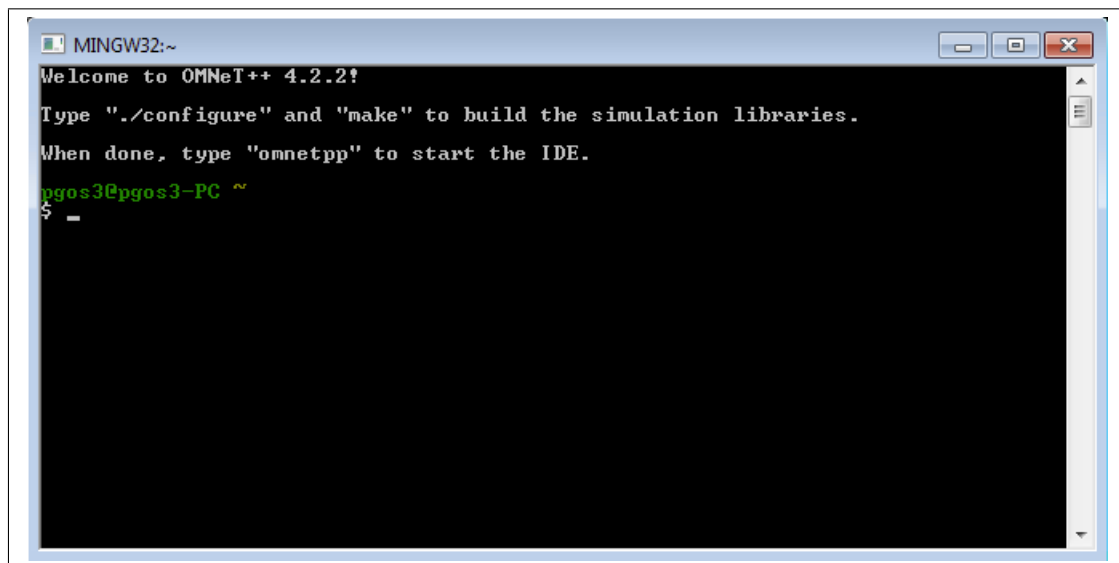


Figure 2: MinGW32 command shell

- Type in the following commands (press **RETURN** after each command):
`. setenv (take care of the space character after the dot)`
`./configure`
`make`
- Start *OMNeT++* by typing `omnetpp` into the *MinGW32* console. Now *OMNeT++* should start.

- Create a *workspace* by selecting a new folder on your hard disk where your projects have to be located to.
- If you want to create a shortcut on your desktop for starting the IDE, select the *omnetpp.cmd* located in the *omnetpp-4.2.2/ide* directory in Windows Explorer, **right-click** it, and choose **Send To** → **Desktop (create shortcut)** from the menu. On Windows 7, you can **right-click** the taskbar icon while the IDE is running, and select **Pin this program to taskbar** from the context menu.

2. Import of the INET framework into OMNeT++ workspace

Due to the OS³ simulator bases on the *INET* framework, you have to import the actual version of *INET* into your *OMNeT++* workspace.

- Minimize *OMNeT++ IDE* window.
- Download the INET project from <http://inet.omnetpp.org>. Choose a version which fits to your OMNeT++ version (in this How-To we select version 20111118 for OMNeT++ version 4.22).

Important: Do not extract the archive.

- Switch back to the *OMNeT++ IDE* window, click on *Workbench*, click right on the *Project Explorer* section and choose the **Import** option (see Figure 3) to launch the *Import Wizzard*.

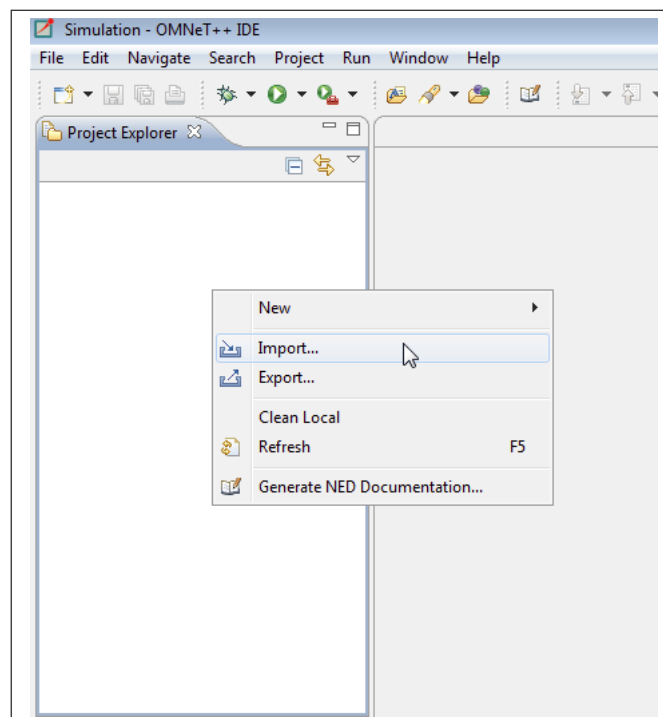


Figure 3: OMNeT++ project explorer - Import

- Double click on **General** and choose **Existing Projects into Workspace**.

- Choose **Select Archive file** and select the *INET project archive* you have downloaded before. Ensure that **all file types** are displayed in the file manager (select *.* as file types).
- Click on **finish**, the *INET-project* is now imported into your workspace (see Figure 4).

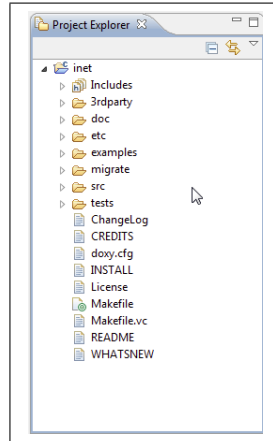


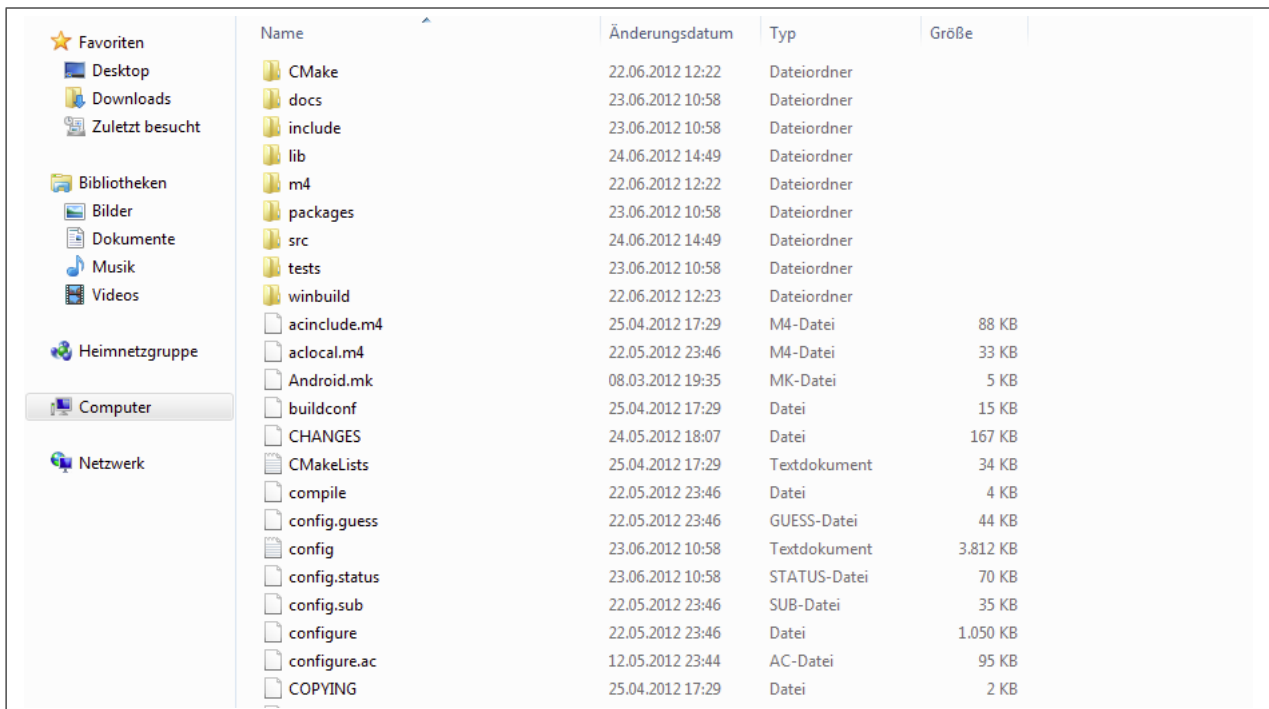
Figure 4: OMNeT++ project explorer after INET project import

- Press **CTRL + B** on your keyboard to build the *INET project*.

3. Installation of cURL on MINGW32 environment

The OS³ simulator uses the *cURL* library to fetch data from web services. If you want to use current weather and altitude data for individual points in the world, you have to install the *cURL* environment on your system.

- Minimize the *OMNeT++ IDE* window.
- Download the *source archive* of *cURL* from <http://curl.haxx.se/libcurl> (this How-To is tested with version 7.27).
- Extract the download archive (e.g. with *WinRAR*) into a folder on your hard disk. It is recommended to choose a folder directly on *c:* (e.g. *c:\curl-7.27.0*). After the extraction is completed, folder structure in the destination directory should look like in Figure 5.



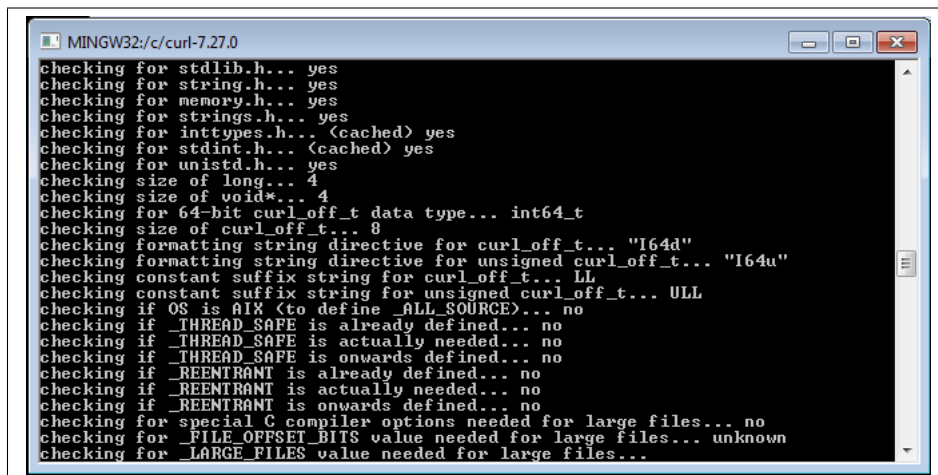
Name	Änderungsdatum	Typ	Größe
CMake	22.06.2012 12:22	Dateiordner	
docs	23.06.2012 10:58	Dateiordner	
include	23.06.2012 10:58	Dateiordner	
lib	24.06.2012 14:49	Dateiordner	
m4	22.06.2012 12:22	Dateiordner	
packages	23.06.2012 10:58	Dateiordner	
src	24.06.2012 14:49	Dateiordner	
tests	23.06.2012 10:58	Dateiordner	
winbuild	22.06.2012 12:23	Dateiordner	
acinclude.m4	25.04.2012 17:29	M4-Datei	88 KB
aclocal.m4	22.05.2012 23:46	M4-Datei	33 KB
Android.mk	08.03.2012 19:35	MK-Datei	5 KB
buildconf	25.04.2012 17:29	Datei	15 KB
CHANGES	24.05.2012 18:07	Datei	167 KB
CMakeLists	25.04.2012 17:29	Textdokument	34 KB
compile	22.05.2012 23:46	Datei	4 KB
config.guess	22.05.2012 23:46	GUESS-Datei	44 KB
config	23.06.2012 10:58	Textdokument	3.812 KB
config.status	23.06.2012 10:58	STATUS-Datei	70 KB
config.sub	22.05.2012 23:46	SUB-Datei	35 KB
configure	22.05.2012 23:46	Datei	1.050 KB
configure.ac	12.05.2012 23:44	AC-Datei	95 KB
COPYING	25.04.2012 17:29	Datei	2 KB

Figure 5: Structure of the cURL folder

- Go into *MinGW32 shell* window and change folder to the *cURL* folder you have created before. For example, if you have extracted *cURL* to *c:\curl-7.27.0*, you have to type in `cd /c/curl-7.27.0/` into the *MinGW console*. After typing in the *cd-command*, press **RETURN**.
- Type in the following commands into the *MINGW32 console* (Press **RETURN** after each command):

```
./configure
make
make install
make clean
```

Note: This procedure may take a while. During the installation routine, your *MINGW32 shell* window should look like in Figure 6.



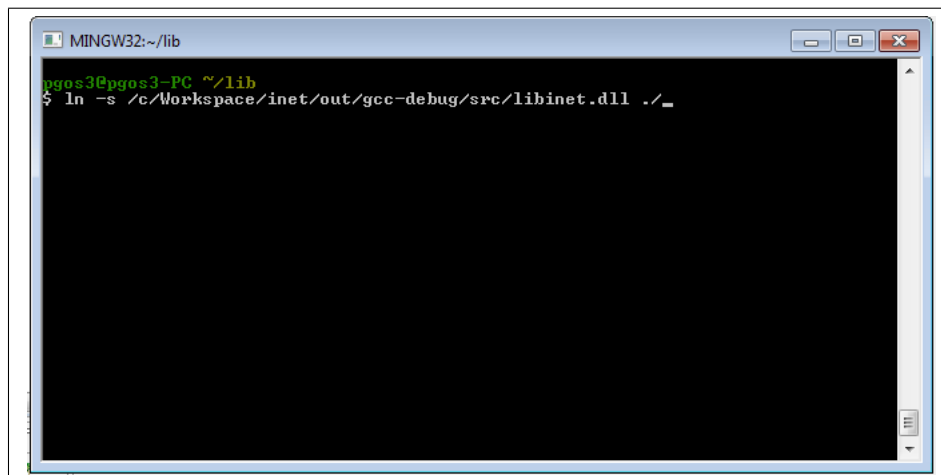
```

MINGW32: c/curl-7.27.0
checking for stdlib.h... yes
checking for string.h... yes
checking for memory.h... yes
checking for strings.h... yes
checking for inttypes.h... <cached> yes
checking for stdint.h... <cached> yes
checking for unistd.h... yes
checking size of long... 4
checking size of void*... 4
checking for 64-bit curl_off_t data type... int64_t
checking size of curl_off_t... 8
checking formatting string directive for curl_off_t... "I64d"
checking formatting string directive for unsigned curl_off_t... "I64u"
checking constant suffix string for curl_off_t... LL
checking constant suffix string for unsigned curl_off_t... ULL
checking if OS is AIX (to define _ALL_SOURCE)... no
checking if _THREAD_SAFE is already defined... no
checking if _THREAD_SAFE is actually needed... no
checking if _THREAD_SAFE is onwards defined... no
checking if _REENTRANT is already defined... no
checking if _REENTRANT is actually needed... no
checking if _REENTRANT is onwards defined... no
checking for special C compiler options needed for large files... no
checking for _FILE_OFFSET_BITS value needed for large files... unknown
checking for _LARGE_FILES value needed for large files...
  
```

Figure 6: MINGW32 shell during installation of cURL

4. Linking of LibINET to OMNeT++ simulation environment

- Go to the *MINGW32 console* and change the directory to the *OMNeT++ library* folder. For example, if you have installed *OMNeT++* in *c:\omnetpp-4.2.2*, you have to type in `cd /c/omnetpp-4.2.2/lib` into the *MINGW32 shell*.
- Link the *INET library* to the *OMNeT++ library* folder. The *INET library* path is in the *INET project* folder in your workspace. For example, if your workspace is located on *c:\workspace*, you have to type in `ln -s /c/workspace/inet/out/gcc-debug/src/libinet.dll ./` (see figure 7).



```

MINGW32: ~/lib
pgos3@pgos3-PC ~/lib
$ ln -s /c/Workspace/inet/out/gcc-debug/src/libinet.dll ./
  
```

Figure 7: Symbolic link in MINGW32 shell

5. Installation of the OS³ simulator

- Download the latest version of the OS³ from <http://www-os3.kn.e-technik.tu-dortmund.de>.

Important: Do not extract the archive.

- Switch back to the *OMNeT++ IDE* window, click on *Workbench*, click right on the *Project Explorer* section and choose the **Import** option (see Figure 3) to launch the *Import Wizzard*.
- Double click on **General** and choose **Existing Projects into Workspace**.
- Choose **Select Archive file** and select the *OS³ project archive* you have downloaded before.
- Click on **finish**, the *OS³-project* is now imported into your workspace.
- Now the *OS³* has to be configured to your simulation environment. Click right on the *OS³ project folder* on the left side of the *OMNeT++ IDE* and select **Properties** (see Figure 8). Now the *project properties* will appear.

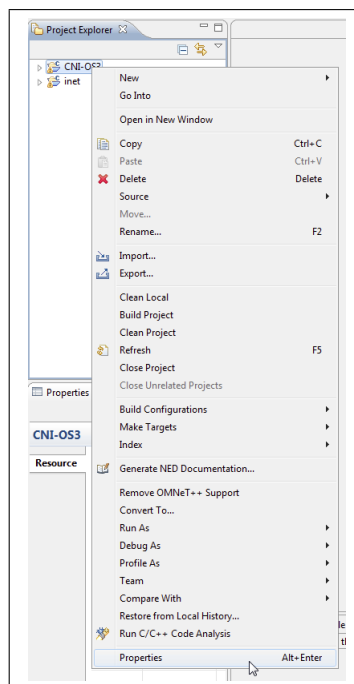


Figure 8: Selecting the OS³ project properties

- Select **C/C++ General**, then select **Path and Symbols**. The window should now look like in Figure 9.

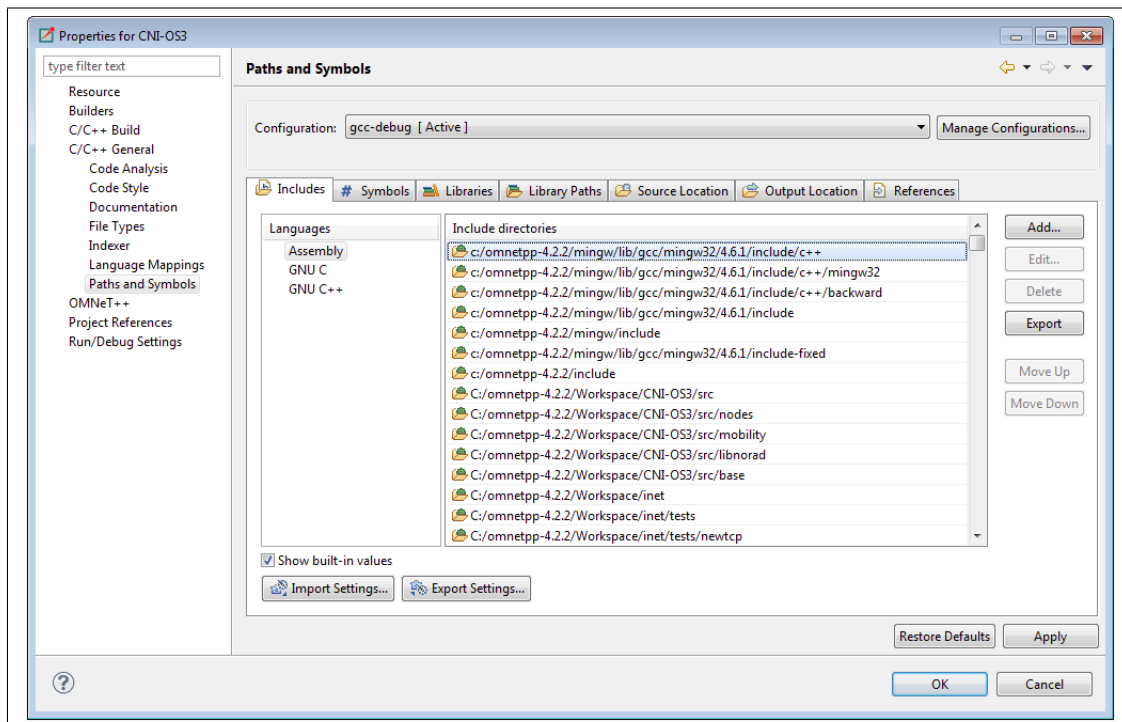


Figure 9: Property window for C/C++ - Path and Symbols

- Press the **Add** button on window's right side to open a window for entering a new directory path.
- Type `/usr/local/include` into the *directory* field and select the options **Add to all configurations** and **Add to all languages** (see Figure 10). Then press **OK**.

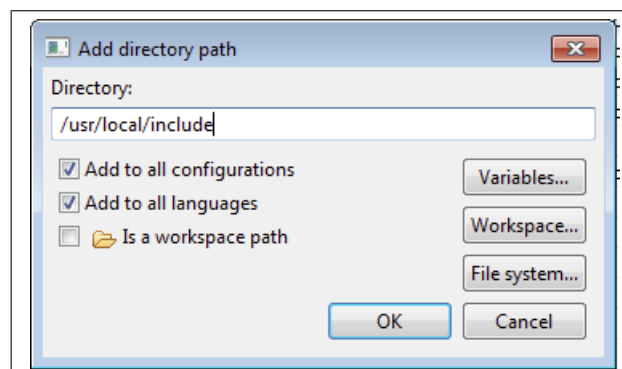


Figure 10: Window for adding an include folder

- Press the **Add** button again and enter the path of your *cURL include* folder on your hard disk into the *directory* field. For example, if you have installed *cURL* in `c:\curl-7.27.0`, you have to type `c:\curl-7.27.0\include` into the *directory* field. It is also possible to select the include folder via the **File System** button.

Note: This step is necessary to overcome the *File not found* IDE bug.

- In the *Path and Symbols* window, select the **Library path** section (see Figure 11).

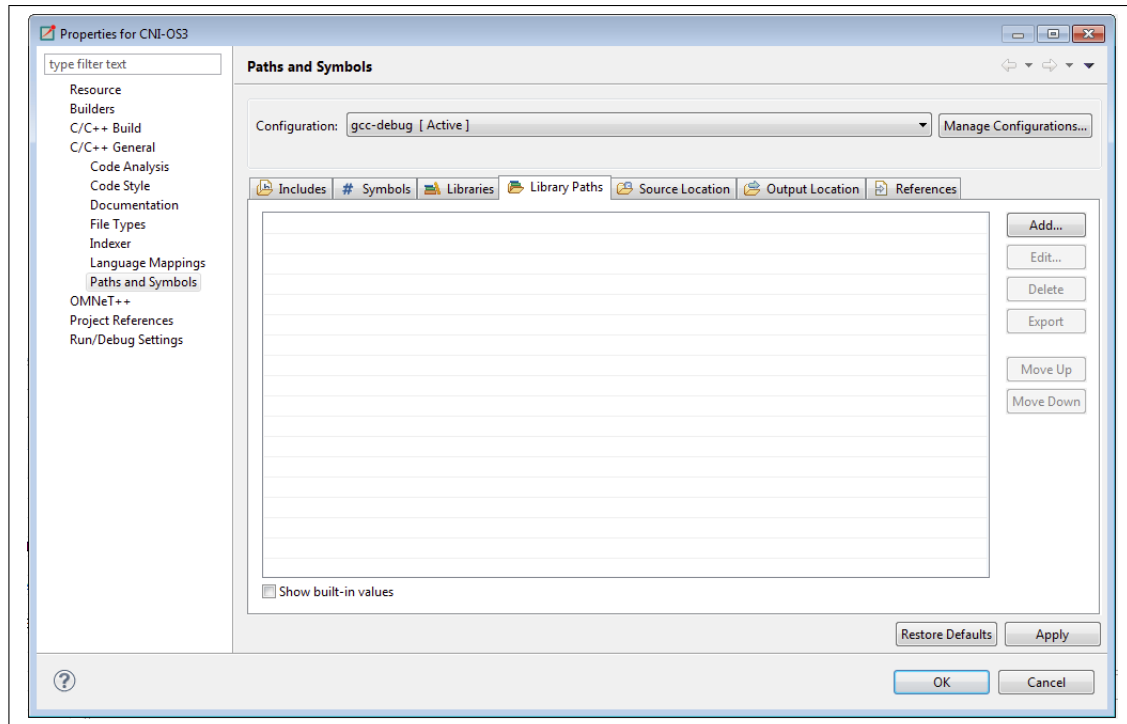


Figure 11: Library Path Section

- Press the **Add** button and type in `/usr/local/lib` into the *directory field* and select the options **Add to all configurations** and **Add to all languages** (see Figure 12). Then press the **OK** button.

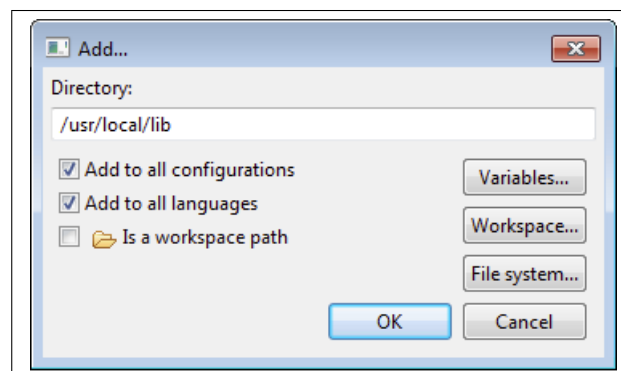


Figure 12: Window for adding a library folder

- Press the **Apply** button.

- Now select the **OMNeT++** section on window's left side, then select **MakeMake** part. Now the *MakeMake* window will open (see Figure 13).

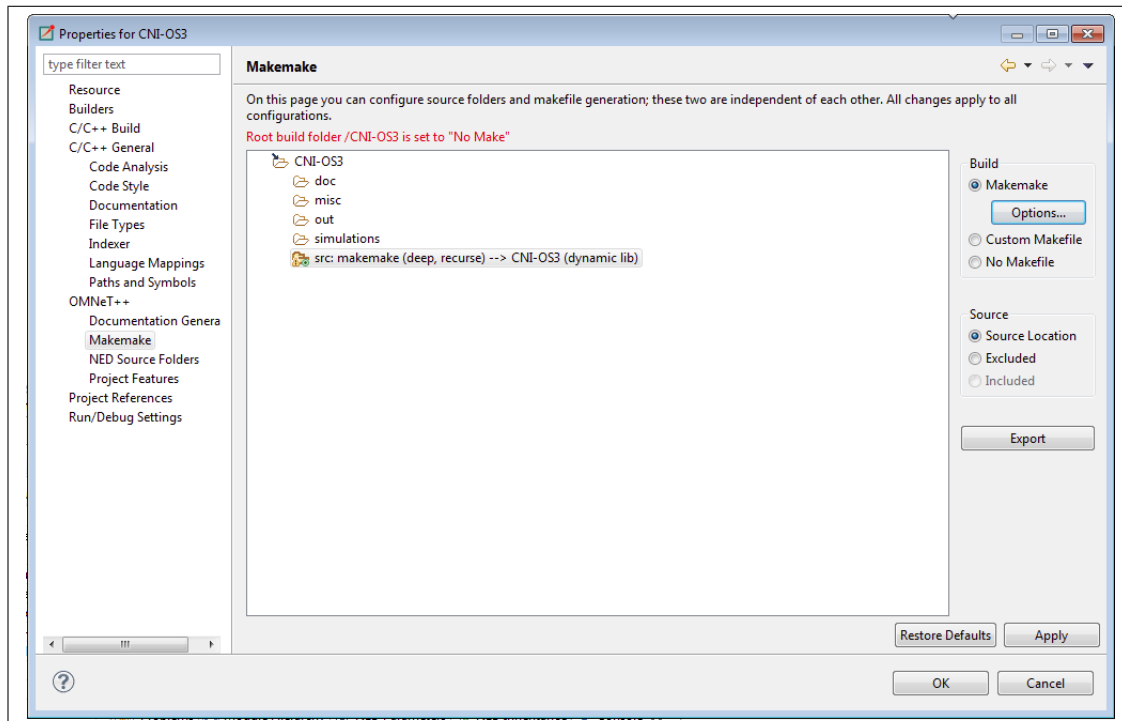


Figure 13: MakeMake section

- Select the **src** folder, then select the **MakeMake** bullet item, afterwards click on the **Options** button on the right side of the window. Now a *MakeMake options* window will open, which looks like in Figure 14.

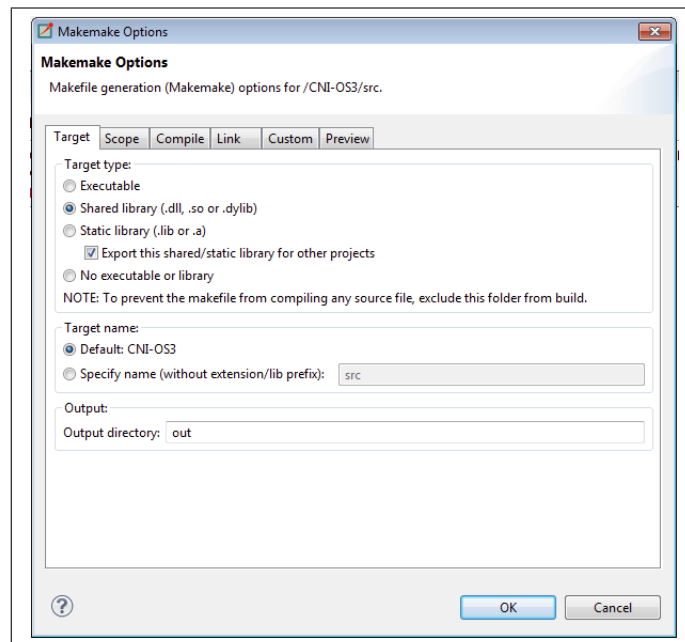


Figure 14: MakeMake options window

- Select the **Shared Library** bullet item as the *target type*, then go the **Link** section.
- Select **Link with libraries from referenced projects** and click on **More**. Now the options window should look like in Figure 15.

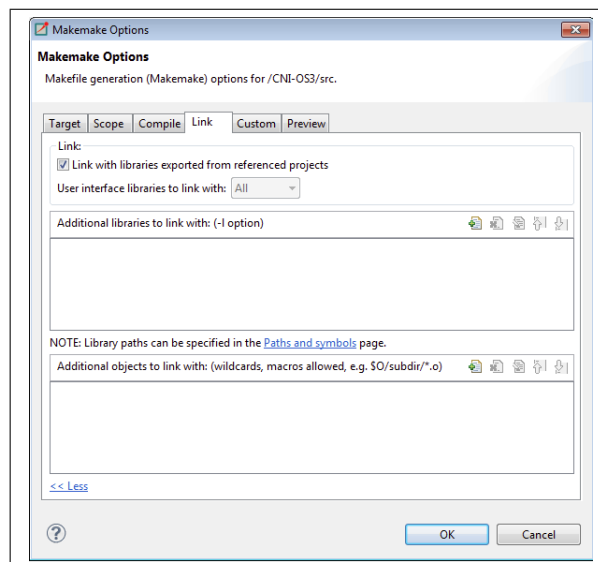



Figure 15: MakeMake options window - Link section

- Click on the **Plus symbol**  and type `curl` into the text field (see Figure 16). Afterwards, press the **OK** button.

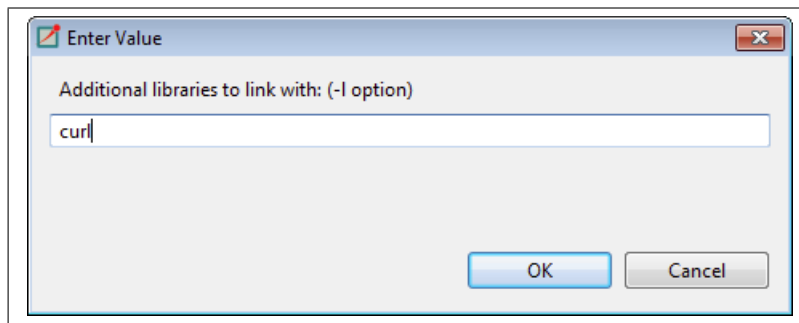


Figure 16: Adding cURL library

- Press **Apply** and **OK** button to close the *OS³ project settings* window. The configuration of the *OS³ simulator* has been finished now.
- Press **CTRL+B** on your keyboard to compile all related projects. If no errors occur, *OS³* has been installed successfully on your system.

Now the *OS³* simulator is ready for further action. Now you have to **register** to some websites (see chapter 6) in order to use live data from web service module in your simulation. After this, you can **test** your configuration by running an example located in the *example* folder, which is part of the *OS³* project.

3.1 Windows - Checklist

- ☐ **Download** OMNeT++ (Version 4.2.2) [<http://www.omnetpp.org>]
- ☐ **Install** OMNeT++ via the MinGW environment
- ☐ **Download** INET framework (**Version 20111118**) [<http://inet.omnetpp.org>]
- ☐ **Import** INET framework into workspace and build the INET project
- ☐ **Download** cURL sourcecode (Version 7.27) [<http://curl.haxx.se>]
- ☐ Open **MinGW32 console**, change to cURL directory and type in the following **commands**:
./configure
make
make install
make clean
- ☐ Create **symbolic link** of *libinet.dll* from the out folder of the INET project to the *lib* directory of OMNeT++ (e.g. /c/omnet-4.2.2/lib)
- ☐ **Download** the OS³ simulator (Latest Version)
[<http://www-os3.kn.e-technik.tu-dortmund.de>]
- ☐ **Import** the OS³ project into workspace
- ☐ Open **Project Settings**
Go to **Path and Symbols** section and add two folders to **all configurations** and **all languages**:
/usr/local/include
include directory of cURL (e.g. c:\curl-7.27\include)
- ☐ Go to **Library Paths** and add the following folder to **all configurations** and **all languages**:
/usr/local/lib
- ☐ Go to **OMNeT++** menu, choose **MakeMake** section, select the **src** folder and click on **Options**. Select **Shared Library** bullet item as *target type*
- ☐ Go to **Link** section and add the curl library
- ☐ **Build** the OS³ project

Installation is now finished!

4 Linux (Ubuntu/Debian)

1. Installation of OMNeT++

- Open a **terminal** window.
- Update your package library via typing `apt-get update` into the terminal.
- Install several packages via the `apt-get` command:

```
sudo apt-get install build-essential gcc g++ bison flex perl \
tcl-dev tk-dev blt libxml2-dev zlib1g-dev openjdk-6-jre \
doxygen graphviz openmpi-bin libopenmpi-dev libpcap-dev
```
- **Important:** Make sure that you have sufficient permissions to execute *substitute user* (*sudo*) commands.
- Download *OMNeT++* (tested with version 4.2.2) from <http://www.omnetpp.org>. Be sure that you download the sources instead of binaries.
- Extract the archive (e.g. with the terminal command `tar -xvzf omnetpp-4.2.2-src.tar.gz`) into a folder on your hard disk. After the extraction, your folder structure in the destination directory should look like in Figure 17.

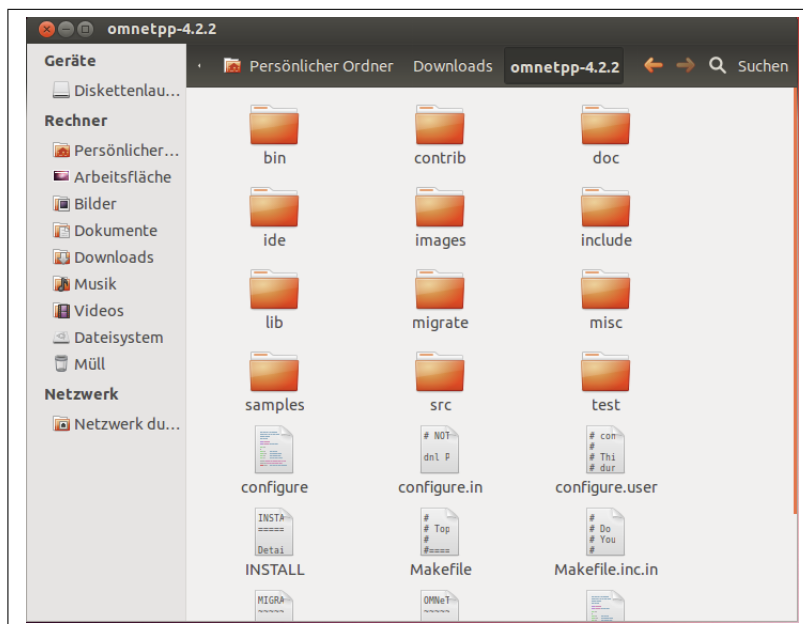


Figure 17: Structure of the OMNeT++ folder

- Open the text file **.bash_rc**, which is located in your *home* directory. For example, if your username is *os3user*, the *.bash_rc* file is located in `/home/os3user/.bash_rc`. After opening the text file, add the following two lines to the end of the file:

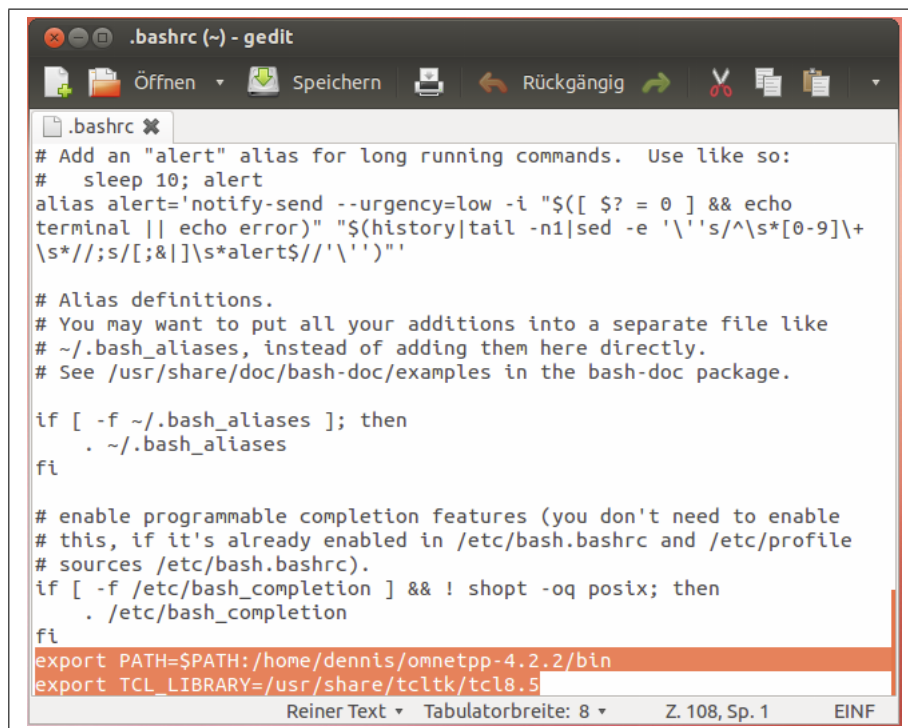

```
export PATH=$PATH:/home/%username%/path_to_omnet/bin
export TCL_LIBRARY=/usr/share/tcltk/tcl8.5
```

The variables %username and %path_to_omnet have to be replaced with your username and the path to the extracted OMNeT++ installation files on your hard disk (e.g. /home/os3user/omnetpp-4.22/bin). Now the text file should look like in Figure 18. After you have done this, restart the terminal.

- Go back to the terminal and change the dir to the extracted OMNeT++ folder (e.g. cd /home/os3user/omnetpp-4.22/). Then type in the following commands:
./configure
sudo make (the sudo in the beginning is optional)

Note: This procedure may take some time.

- Start OMNeT++ by typing omnetpp into the terminal. Now OMNeT++ should start.
- Create a *workspace* by selecting a new folder on your hard disk where your projects have to be located to.



```
.bashrc (~) - gedit
# Add an "alert" alias for long running commands. Use like so:
# sleep 10; alert
alias alert='notify-send --urgency=low -i "${?} = 0 ] && echo
terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+
\s*//;s/[:&|]\s*alert$/'\''')'

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if [ -f /etc/bash_completion ] && ! shopt -oq posix; then
    . /etc/bash_completion
fi

export PATH=$PATH:/home/dennis/omnetpp-4.2.2/bin
export TCL_LIBRARY=/usr/share/tcltk/tcl8.5
```

Figure 18: Modified .bash_rc text file

2. Import of the INET framework into OMNeT++ workspace

Due to the OS³ simulator bases on the *INET* framework, you have to import the actual version of *INET* into your OMNeT++ workspace.

- Minimize OMNeT++ IDE window.

- Download the INET project from <http://inet.omnetpp.org>. Choose a version which fits to your OMNeT++ version (in this How-To we select version 20111118 for OMNeT++ version 4.22).

Important: Do not extract the archive.

- Switch back to the *OMNeT++ IDE* window, click on *Workbench*, click right on the *Project Explorer* section and choose the **Import** option (see Figure 19) to launch the *Import Wizzard*.

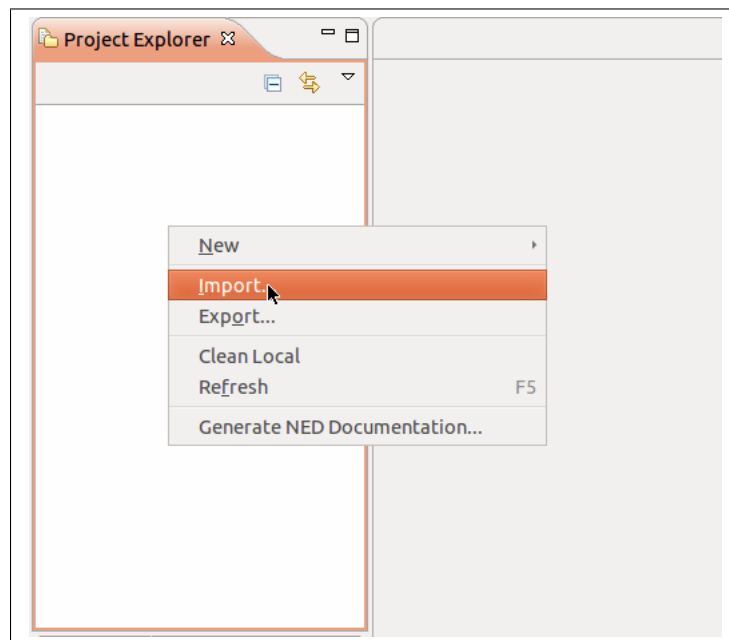


Figure 19: OMNeT++ project explorer - Import

- Double click on **General** and choose **Existing Projects into Workspace**.
- Choose **Select Archive file** and select the *INET project archive* you have downloaded before.
- Click on **finish**, the *INET-project* is now imported into your workspace (see Figure 20).

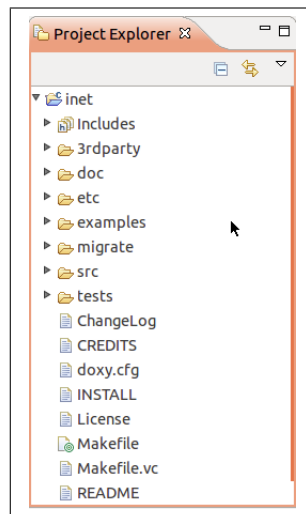


Figure 20: OMNeT++ project explorer after INET project import

- Press **CTRL + B** on your keyboard to build the *INET* project.

3. Installation of cURL

The OS³ simulator uses the *cURL* library to fetch data from web services. If you want to use current weather and altitude data for individual points in the world, you have to install the *cURL* environment on your system.

- Minimize the *OMNeT++ IDE* window.
- Open a **terminal** window.
- Install the cURL package via the command `sudo apt-get install libcurl4-nss-dev`

Important: Make sure that you have sufficient permissions to execute *substitute user (sudo)* commands.

4. Installation of the OS³ simulator

- Download the latest version of the OS³ from <http://www-os3.kn.e-technik.tu-dortmund.de>.

Important: Do not extract the archive.

- Switch back to the *OMNeT++ IDE* window, click on *Workbench*, click right on the *Project Explorer* section and choose the **Import** option (see Figure 19) to launch the *Import Wizzard*.
- Double click on **General** and choose **Existing Projects into Workspace**.
- Choose **Select Archive file** and select the OS³ project archive you have downloaded before.
- Click on **finish**, the OS³-project is now imported into your workspace.

- Now the OS³ has to be configured to your simulation environment. Click right on the OS³ *project folder* on the left side of the OMNeT++ IDE and select **Properties** (see Figure 21). Now the *project properties* will appear.

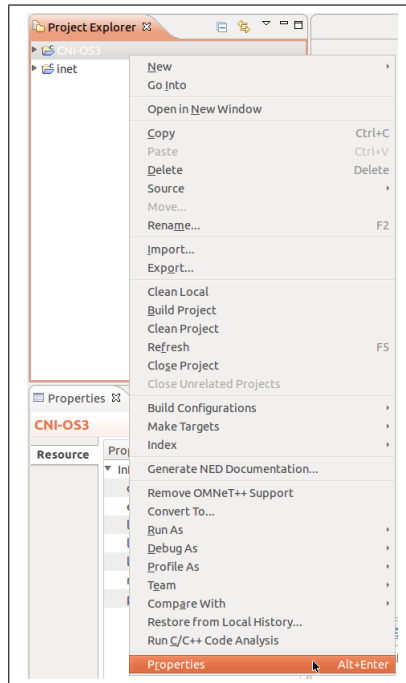


Figure 21: Selecting the OS³ project properties

- Now select the **OMNeT++** section on window's left side, then select **MakeMake** part. Now the *MakeMake* window will appear (see Figure 22).

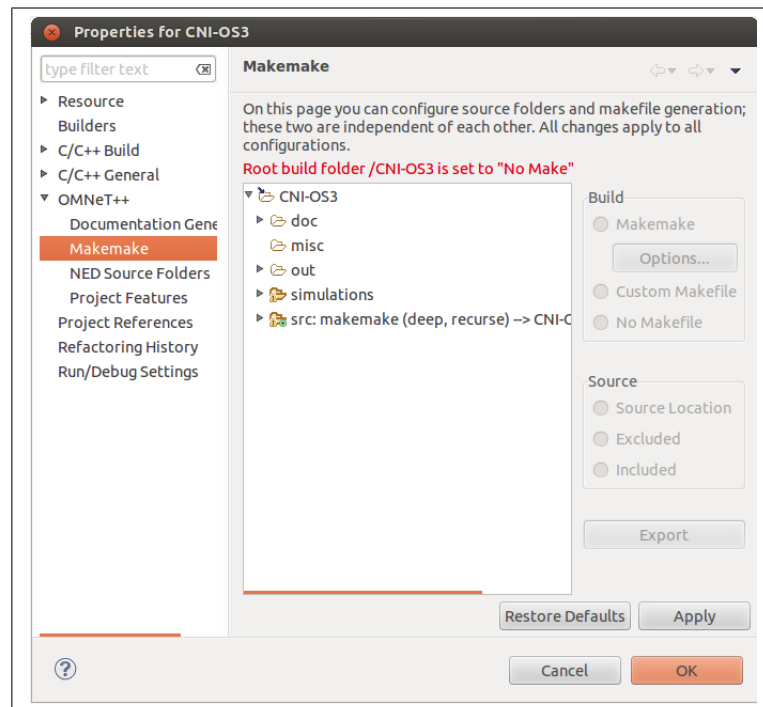


Figure 22: MakeMake section

- Select the **src** folder, then select the **MakeMake** radio button, afterwards click on the **Options** button on the right side of the window. Now a *MakeMake options window* will open, which looks like in Figure 23.

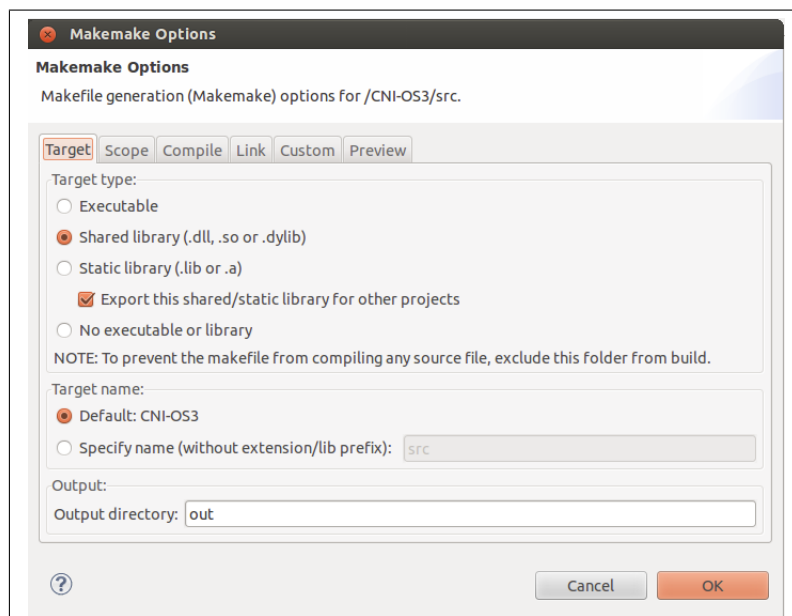


Figure 23: MakeMake options window

- Select the **Shared Library** bullet item as the *target type*, then go the **Link** section.
- Select **Link with libraries from referenced projects** and click on **More**. Now the options window should look like in Figure 24.

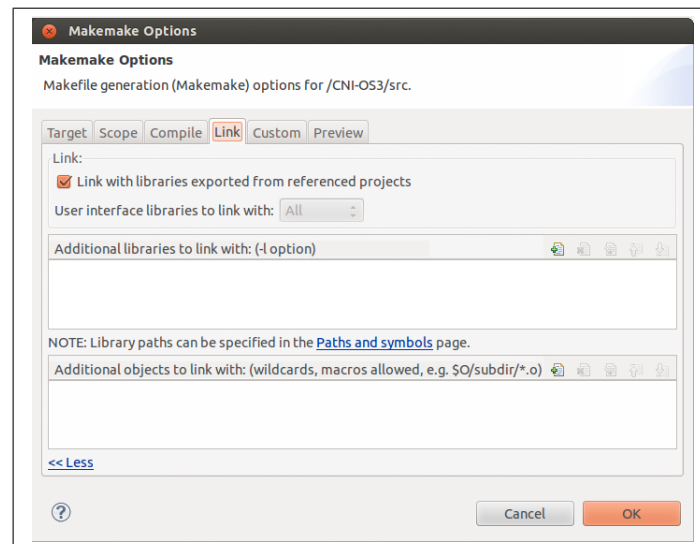



Figure 24: MakeMake options window - Link section

- Click on the **Plus symbol**  and type `curl` into the text field (see Figure 25). Afterwards, press the **OK** button.

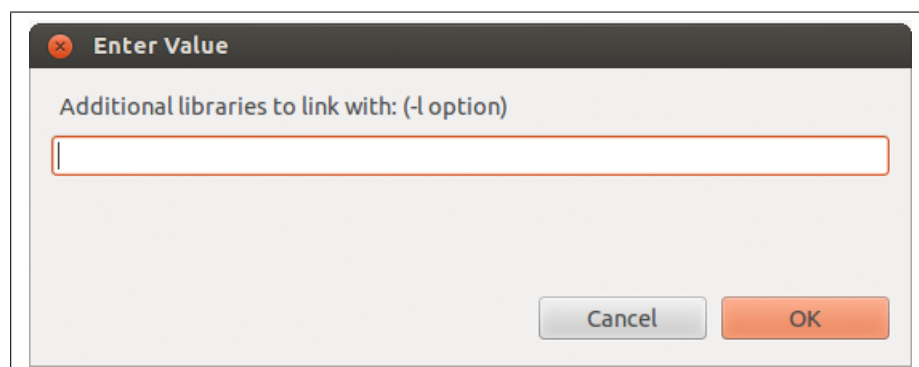


Figure 25: Adding cURL library

- Press **Apply** and **OK** button to close the *OS³ project settings* window. The configuration of the *OS³ simulator* has been finished now.
- Press **CTRL+B** on your keyboard to compile all related projects. If no errors occur, *OS³* has been installed successfully on your system.

Now the *OS³* simulator is ready for further action. Now you have to **register** to some websites (see chapter 6) in order to use live data from web service module in your simulation. After this, you can **test** your configuration by running an example located in the *example* folder, which is part of the *OS³* project.

4.1 Linux - Checklist

- ☐ **Download** OMNeT++ (Version 4.2.2) [<http://www.omnetpp.org>]
- ☐ **Install** OMNeT++ via the MinGW environment
- ☐ **Download** INET framework (**Version 20111118**) [<http://inet.omnetpp.org>]
- ☐ **Import** INET framework into workspace and build the INET project
- ☐ **Install** cURL via *apt-get* command (be sure you have sufficient privileges):
`apt-get install libcurl4-nss-dev`
- ☐ **Download** the OS³ simulator (Latest Version)
[<http://www-os3.kn.e-technik.tu-dortmund.de>]
- ☐ **Import** the OS³ project into workspace
- ☐ Open **Project Settings**. Go to **OMNeT++** menu, choose **MakeMake** section,
select the **src** folder and click on **Options**. Select **Shared Library** bullet item as *target type*
- ☐ Go to **Link** section and add the `curl` library
- ☐ **Build** the OS³ project

Installation is now finished!

5 Mac OS X

1. Installation of OMNeT++

- Open a **terminal** window.
- Download *OMNeT++* (tested with version 4.2.2) from <http://www.omnetpp.org>. Be sure that you download the sources instead of binaries.
- Extract the archive (e.g. with the terminal command `tar -xvzf omnetpp-4.2.2-src.tar.gz`) into a folder on your hard disk. After the extraction, your folder structure in the destination directory should look like in Figure 26.

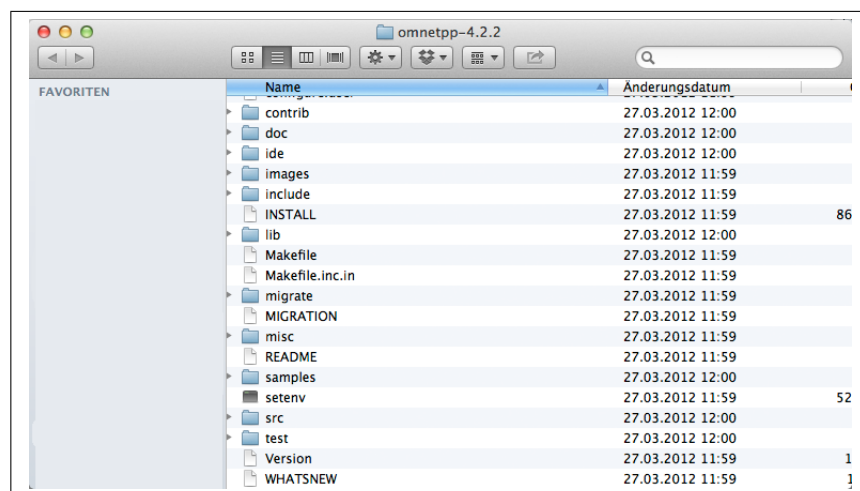


Figure 26: Structure of the OMNeT++ folder

- Open the text file **.bash_profile**, which is located in your *home* directory. For example, if your username is *os3user*, the *.bash_profile* file is located in */Users/os3user/.bash_profile*. After opening the text file, add the following line to the end of the file:

```
export PATH=$PATH:/Users/%username/%path_to_omnet/bin
```

The variables *%username* and *%path_to_omnet* have to be replaced with your username and the path to the extracted OMNeT++ installation files on your hard disk (e.g. */Users/os3user/omnetpp-4.22/bin*). The text file should now look like in Figure 27.

After you have done this, restart the terminal.

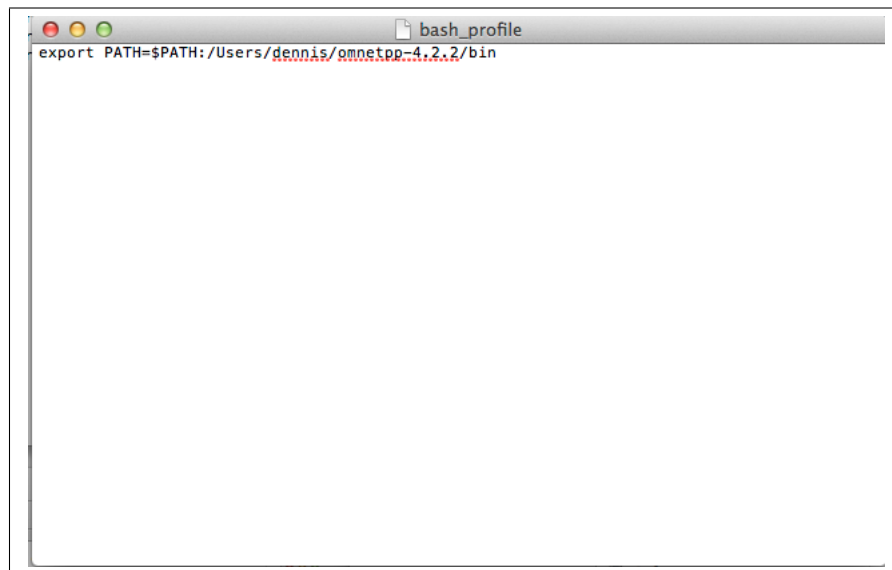


Figure 27: Modified .bash_rc text file

- Go back to the terminal and change the dir to the extracted *OMNeT++* folder (e.g. `cd /Users/os3user/omnetpp-4.22/`). Then type in the following commands:
`./configure`
`sudo make` (the *sudo* in the beginning is optional)
- Note:** This procedure may take some time.
- Start OMNeT++ by typing `omnetpp` into the terminal. Now *OMNeT++* should start.
- Create a *workspace* by selecting a new folder on your hard disk where your projects have to be located to.

2. Import of the INET framework into OMNeT++ workspace

Due to the OS³ simulator bases on the *INET* framework, you have to import the actual version of *INET* into your *OMNeT++* workspace.

- Minimize *OMNeT++ IDE* window.
- Download the INET project from <http://inet.omnetpp.org>. Choose a version which fits to your OMNeT++ version (in this How-To we select version 20111118 for OMNeT++ version 4.22).
- Switch back to the *OMNeT++ IDE* window, click on *Workbench*, click right on the *Project Explorer* section and choose the **Import** option (see Figure 28) to launch the *Import Wizzard*.

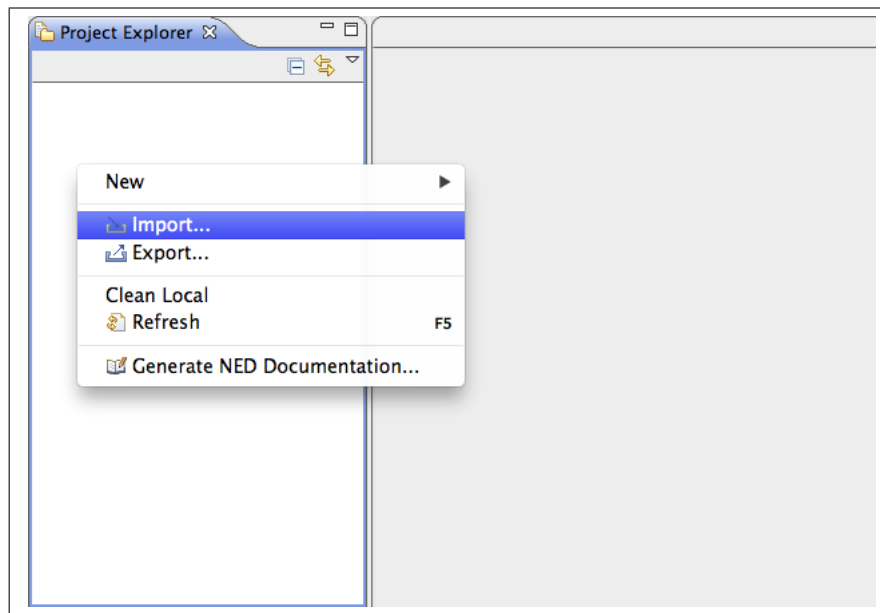


Figure 28: OMNeT++ project explorer - Import

- Double click on **General** and choose **Existing Projects into Workspace**.
- Choose **Select Archive file** and select the *INET project archive* you have downloaded before.
- Click on **finish**, the *INET-project* is now imported into your workspace (see Figure 29).

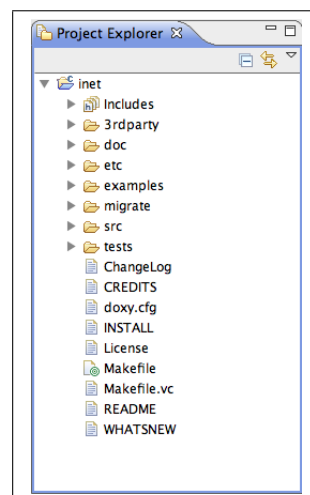


Figure 29: OMNeT++ project explorer after INET project import

- Press **CTRL + B** on your keyboard to build the *INET project*.

3. Installation of cURL

The OS³ simulator uses the *cURL* library to fetch data from web services. If you want to use

current weather and altitude data for individual points in the world, you have to install the *cURL* environment on your system.

- Minimize the *OMNeT++ IDE* window.
- Download the *source archive* of *cURL* from <http://curl.haxx.se/libcurl> (this How-To is tested with version 7.27).
- Extract the download archive into a folder on your hard disk.
After the extraction is completed, folder structure in the destination directory should look like in figure 30.

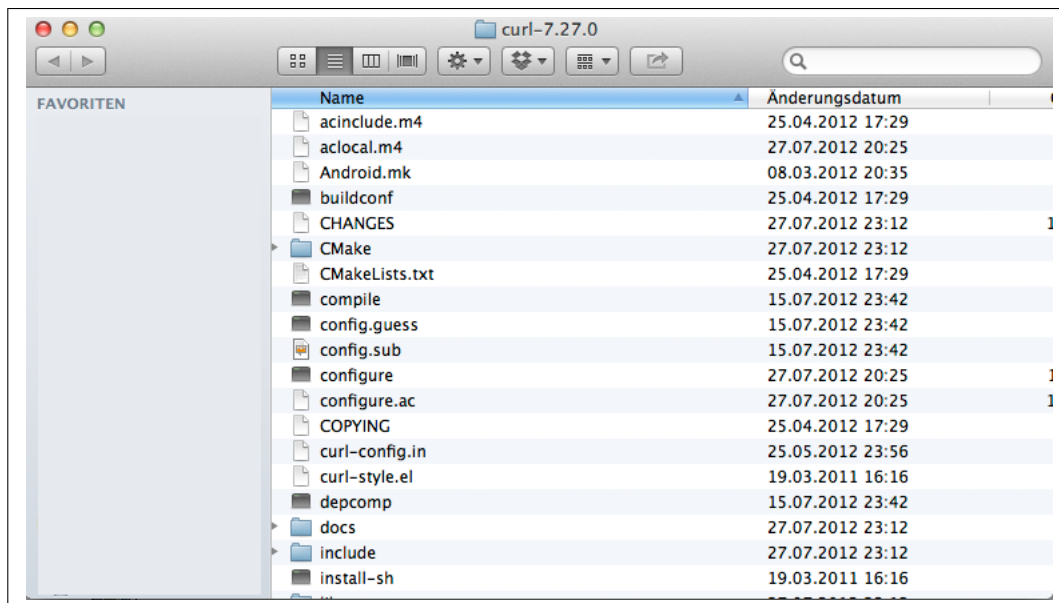
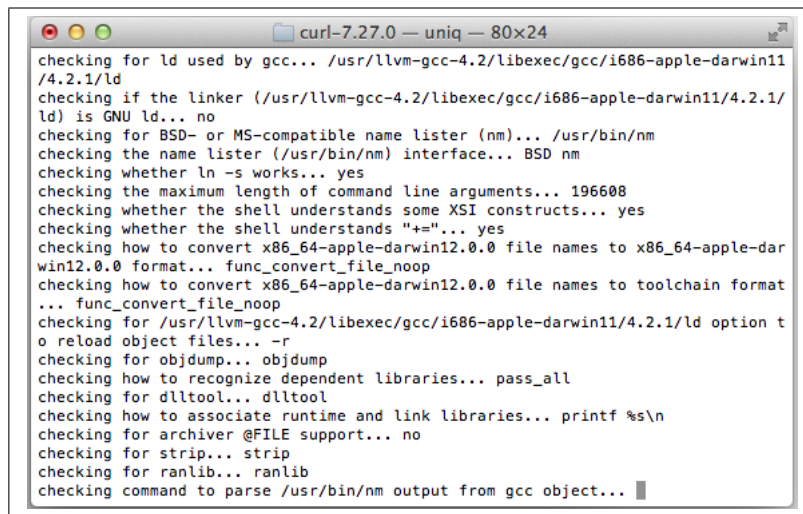


Figure 30: Structure of the *cURL* folder

- Open a **terminal**.
- Change folder to the *cURL* folder you have created before. For example, if you have extracted *cURL* to `/Users/os3user/curl-7.27.0`, you have to type in `cd /Users/os3user/curl-7.27.0/` into the shell.
- Type in the following commands into the terminal (Press **RETURN** after each command):

```
./configure
make
sudo make install
sudo make clean
```

Note: This procedure may take a while. During the installation routine, the terminal window should look like in Figure 31.



```

checking for ld used by gcc... /usr/llvm-gcc-4.2/libexec/gcc/i686-apple-darwin11/4.2.1/ld
checking if the linker (/usr/llvm-gcc-4.2/libexec/gcc/i686-apple-darwin11/4.2.1/ld) is GNU ld... no
checking for BSD- or MS-compatible name lister (nm)... /usr/bin/nm
checking the name lister (/usr/bin/nm) interface... BSD nm
checking whether ln -s works... yes
checking the maximum length of command line arguments... 196608
checking whether the shell understands some XSI constructs... yes
checking whether the shell understands "+="... yes
checking how to convert x86_64-apple-darwin12.0.0 file names to x86_64-apple-darwin12.0.0 format... func_convert_file_noop
checking how to convert x86_64-apple-darwin12.0.0 file names to toolchain format... func_convert_file_noop
checking for /usr/llvm-gcc-4.2/libexec/gcc/i686-apple-darwin11/4.2.1/ld option to reload object files... -r
checking for objdump... objdump
checking how to recognize dependent libraries... pass_all
checking for dlltool... dlltool
checking how to associate runtime and link libraries... printf %s\n
checking for archiver @FILE support... no
checking for strip... strip
checking for ranlib... ranlib
checking command to parse /usr/bin/nm output from gcc object...

```

Figure 31: terminal window during installation of cURL

4. Installation of the OS³ simulator

- Download the latest version of the OS³ from <http://www-os3.kn.e-technik.tu-dortmund.de>.

Important: Do not extract the archive.

- Switch back to the *OMNeT++ IDE* window, click on *Workbench*, click right on the *Project Explorer* section and choose the **Import** option (see Figure 28) to launch the *Import Wizzard*.
- Double click on **General** and choose **Existing Projects into Workspace**.
- Choose **Select Archive file** and select the OS³ *project archive* you have downloaded before.
- Click on **finish**, the OS³-*project* is now imported into your workspace.
- Now the OS³ has to be configured to your simulation environment. Click right on the OS³ *project folder* on the left side of the *OMNeT++ IDE* and select **Properties** (see Figure 32). Now the *project properties* will appear.

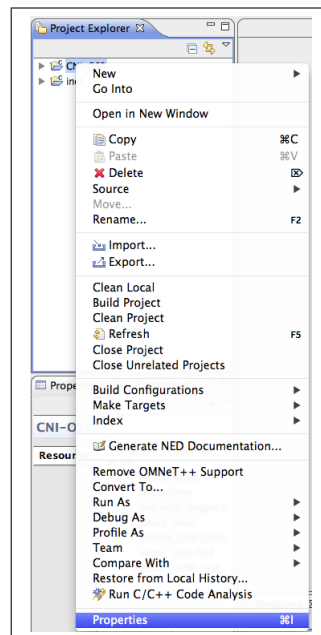


Figure 32: Selecting the OS³ project properties

- Now select the **OMNeT++** section on window's left side, then select **MakeMake** part. Now the *MakeMake* window will appear (see Figure 33).

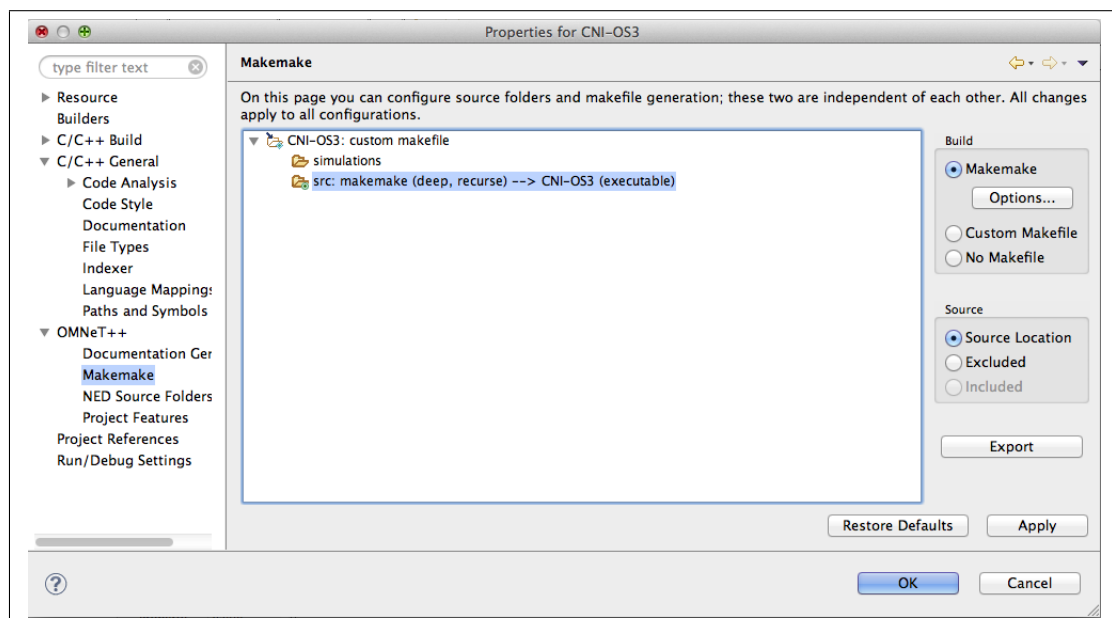


Figure 33: MakeMake section

- Select the **CNI-OS3** folder, then select the **MakeMake** bullet item, afterwards click on the **Options** button on the right side of the window. Now a *MakeMake options window* will open, which looks like in Figure 34.

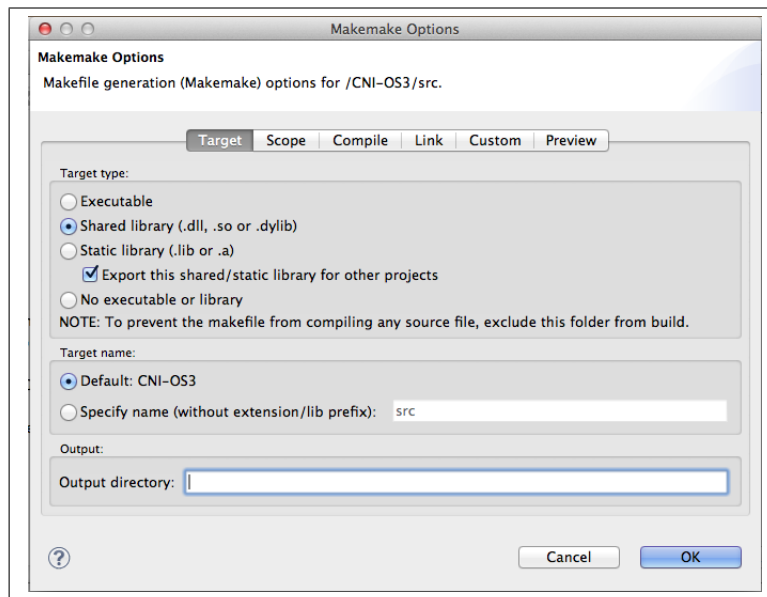


Figure 34: MakeMake options window

- Select the **Shared Library** bullet item as the *target type*, then go the **Link** section.
- Select **Link with libraries from referenced projects** and click on **More**. Now the options window should look like in Figure 35.

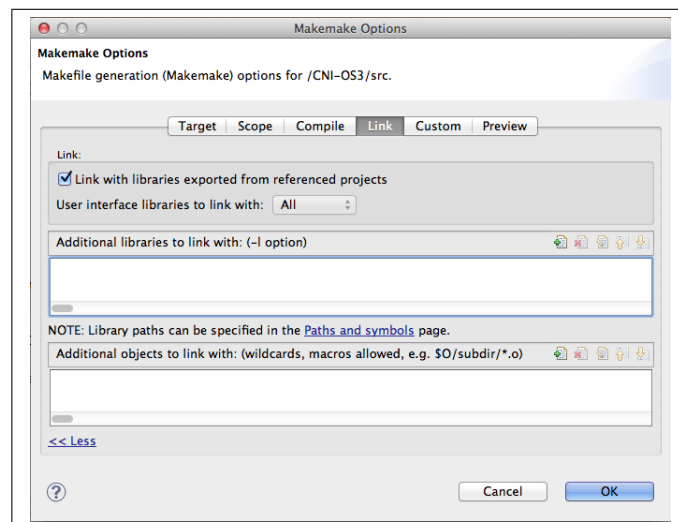



Figure 35: MakeMake options window - Link section

- Click on the **Plus symbol**  and type curl into the text field (see Figure 36). Afterwards, press the **OK** button.

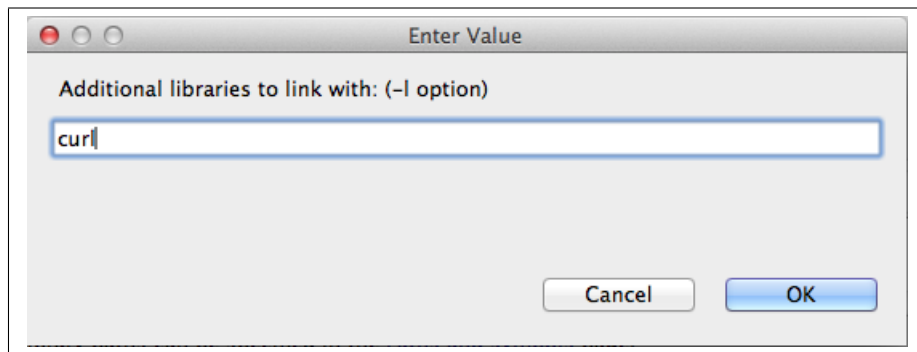


Figure 36: Adding cURL library

- Press **Apply** and **OK** button to close the *OS³ project settings* window. The configuration of the *OS³ simulator* has been finished now.
- Press **CTRL+B** on your keyboard to compile all related projects. If no errors occur, *OS³* has been installed successfully on your system.

Now the *OS³* simulator is ready for further action. Now you have to **register** to some websites (see chapter 6) in order to use live data from web service module in your simulation. After this, you can **test** your configuration by running an example located in the *example* folder, which is part of the *OS³* project.

5.1 Mac OS X - Checklist

- ☐ **Download** OMNeT++ (Version 4.2.2) [<http://www.omnetpp.org>]
- ☐ **Install** OMNeT++ via the MinGW environment
- ☐ **Download** INET framework (**Version 20111118**) [<http://inet.omnetpp.org>]
- ☐ **Import** INET framework into workspace and build the INET project
- ☐ **Download** cURL source code (Version 7.27) [<http://curl.haxx.se>]
- ☐ Open a **terminal**, change to cURL directory and type in the following **commands** (be sure you have sufficient privileges):
 ./configure
 make
 make install
 make clean
- ☐ **Download** the OS³ simulator (Latest Version)
 [<http://www-os3.kn.e-technik.tu-dortmund.de>]
- ☐ **Import** the OS³ project into workspace
- ☐ Open **Project Settings**. Go to **OMNeT++** menu, choose **MakeMake** section, select the src folder and click on **Options**. Select **Shared Library** bullet item as *target type*
- ☐ Go to **Link** section and add the curl library
- ☐ **Build** the OS³ project

Installation is now finished!

6 Registration process for live data integration

6.1 Weather data

In order to be able to use current weather data for an individual latitude/longitude coordinate pair, you have to create an account on <http://www.worldweatheronline.com>, where the OS³ simulator gets its weather data from.

Only a few steps are necessary to create an account. There are no additional costs for the creation process, everything is free of charge. After registration is complete, you are able to use live weather data for any place in the world.

- Go to <http://www.worldweatheronline.com/free-weather.aspx>.
- Click on the **Free Signup** Button at the bottom of the page (see figure 37(a)).
- Fill out the formular with your data. Finish the process with a click on the **Generate API Key** Button (see figure 37(b)).
- You will receive a *Activation mail*. This mail contains a link which you have to open in order to activate your account. After you have done this, your account is enabled.
- Insert your personal *API key*, which is also included in the *Activation Mail*, into the OS³ simulation environment. Therefore, you can find a field called `**.webServiceControl.apiKeyWeather` in the *omnetpp.ini* in your OMNeT++ project where you can enter your *API key*.

Afterwards, you have successfully finished the registration process for live weather data.

6.2 Altitude data

In order to be able to use current altitude data for an individual latitude/longitude coordinate pair, you have to create an account on <http://www.geonames.org>, where the OS³ simulator gets its altitude data from.

Only a few steps are necessary to create an account. There are no additional costs for the creation process, everything is free of charge. After registration is complete, you are able to use high resolution altitude data (30m rastered) for any place in the world.

- Go to <http://www.geonames.org/login>.
- Fill up the formular underneath the login form to create a new account (see figure 38(a)). To complete the registration, click on the **Create account** button.
- You will receive a *Activation mail*. This mail contains a link which you have to open in order to activate your account. After you have done this, your account is enabled.
- Go to <http://www.geonames.org/manageaccount> in order to activate your account for the usage of web service. Therefore, click on the linked text **Click here to enable**. Afterwards, your account has been enabled for the usage of Geonames' web services (see figure 38(b)).

Free Weather API Overview

We offer two types of free weather API for developers and programmers.

- Local Weather API**

The local weather API offers weather forecast data in XML, JSON and CSV format and returns current weather and next 5 days of weather data for worldwide locations.

The user can query via. UK Postcode, Canada Postal code, US Zipcode, City and town name (filter by country) or Latitude and Longitude.
- Marine/Surfing Weather API**

The Marine or Surfing weather API offers weather forecast data in XML, JSON and CSV format and returns todays weather data for worldwide locations.

The user can query via. Latitude and Longitude.
- API Usage REST API**

The API Usage API allows you to get the last 7 days data of your API request to our server. The data is returned in XML and JSON format.

Free sign up

If you are looking for more extended weather forecast then please do visit our [Premium Weather API](#) section to learn more.

Weather API Sign Up

Register to get an API Key to access our **Free Global Weather API** and **Free Marine / Surfing Weather API** for accurate and reliable weather forecast.

Having Problems?

- [Resend Verification Email](#)
- [Forgotten your API Key?](#)

Note: You only need to register once and use the same API Key to access both **Global API** and **Marine API** weather data.

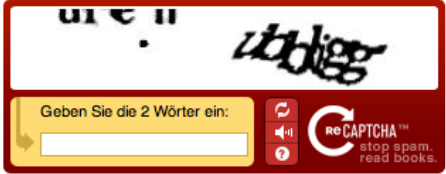
Note: All the fields are required!

Your Name:

Email:

Confirm Email:

Are you human?



Geben Sie die 2 Wörter ein:

☐ I have read and agree to [T&C's](#) and [Usage Policy](#).

(a) Introduction

(b) Registration forumular

Figure 37: WorldWeatherOnline.com - Free Weather API

GeoNames user account

Login

Username

Password

☒ remember me on this computer

[\[I forgot my password \]](#)

or create a new user account

Username

Email

Confirm Email

Password

Confirm password

Free Web Services

the account is not yet enabled to use the free web services. [Click here to enable.](#)

(a) Registration formular
(b) Web service activation

Figure 38: Geonames.org - Free Web Service API for altitude data

- Insert your personal *username*, which you have choosen during registration process, into the OS³ simulation environment. Therefore, you can find a field called `**WebServiceControl.usernameAltitude` in the *omnetpp.ini* in your OMNeT++ project where you can enter your *username*.

Afterwards, you have successfully finished the registration process for live altitude data.