

Installationsanleitung¹

ProtSim über die WebIDE

Sie können die Aufgaben des Projektseminars in unserer WebIDE bearbeiten. Neben einer im Browser ausführbaren Entwicklungsumgebung, die auf Visual Studio Code basiert, erhalten Sie in der WebIDE Zugriff auf eine grafische Desktopumgebung via VNC.

Ersteinrichtung

1. Rufen Sie <https://webide.prakinf.tu-ilmenau.de> in einem Browser Ihrer Wahl auf und melden Sie sich mit Ihrem Uni-Login an. Ihr Passwort wird von Diensten des UniRZs geprüft, und nicht an die WebIDE selbst weitergeleitet.
2. Zuerst müssen Sie den Zugriff auf den ProtSim-Container erhalten. Hierzu finden Sie im Moodle-Kurs einen sogenannten Container-Token. Diesen kopieren Sie in das entsprechende Feld auf der Startseite und klicken anschließend auf 'Container hinzufügen'. Dieser Schritt ist nur für die erste Einrichtung notwendig.
3. Sie sehen jetzt einen neuen Eintrag in der Tabelle der verfügbaren Container. Klicken Sie rechts auf den Start-Knopf und warten Sie kurz. Nachdem der Container gestartet ist, können Sie auf die WebIDE mit einem Klick auf `Open IDE` zugreifen.
4. Die benötigten Dateien für ProtSim befinden sich zu Beginn noch im Ordner `/referenceCode`, auf den Sie nur lesend zugreifen können. Um eigene Änderungen vorzunehmen, müssen Sie die Dateien in Ihren `projectCode`-Ordner kopieren. Alle im `projectCode`-Ordner gespeicherten Dateien sind persistent und bleiben auch nach einem Neustart des Containers erhalten. Zum Kopieren der Daten können Sie die Kommandozeile verwenden. Öffnen Sie hierzu ein Terminal, indem Sie in dem oberen Menü `Terminal` ⇒ `New Terminal` auswählen. Bei der darauf folgenden Abfrage nach dem sogenannten Working Directory des Terminals geben Sie am besten immer `projectCode` an. Für die Ersteinrichtung ist das Working Directory allerdings egal.

Führen Sie in der nun geöffneten Kommandozeile folgenden Befehl aus:

- ```
cp -r /referenceCode/code/ /home/student/projectCode/protsim
```
  - Dieser Befehl kopiert den Ordner `code` in `referenceCode` in Ihr `projectCode`-Verzeichnis und benennt ihn zu `protsim` um.
5. Damit die IDE wie erwartet funktioniert, sollten Sie `initial` alle Projekte kompilieren. Führen Sie daher im Anschluss folgenden Befehl aus
    - ```
/home/student/projectCode/protsim/buildAll.sh
```
 - Die Ausführung dieses Befehls wird `initial` etwas Zeit in Anspruch nehmen und ist nur für die Ersteinrichtung notwendig. Einige Projekte werden erst nach dem Bearbeiten der Aufgaben erfolgreich kompilieren, daher können Sie eventuelle Fehlermeldungen ignorieren. Nachdem der Befehl ausgeführt wurde, ist die initiale Einrichtung abgeschlossen.

¹Stand: 12. Oktober 2023

Allgemeine Nutzung anhand des Projekts protsim01

1. Öffnen Sie den Ordner `projectCode/protsim/protsim01` in der IDE. Hierfür klicken Sie oben links auf das Menü, und anschließend auf `File ⇒ Open Folder`. Navigieren Sie anschließend zu dem Ordner und drücken Sie OK.
2. Öffnen Sie wie zuvor beschrieben ein Terminal und führen Sie jetzt folgenden Befehl aus:
 - `./run.sh`
3. Das Projekt startet mit einer grafischen Oberfläche. Damit Sie auf diese zugreifen können, benötigen Sie VNC. Klicken Sie in der Containerübersicht der WebIDE auf VNC. Sie sollten jetzt einen Desktop mit der geöffneten OMNeT-Simulationsumgebung sehen.
4. Sie können weiterhin eine Debug-Session mit der F5-Taste starten.

Weitere Hinweise

1. Bitte stoppen Sie nach Ihrer Arbeit den Container mit dem roten `Stop`-Button in der Containerübersicht. Sie können den Container nicht über VNC stoppen.
2. Beachten Sie, dass der Container automatisch etwa zwei Stunden nach dem letzten Öffnen der IDE- oder VNC-Ansicht gestoppt wird. Um dies zu verhindern, können Sie vor dem Ablauf dieser Zeit entweder ein neues IDE- oder VNC-Fenster über die Containeransicht öffnen. Ein Neustart der VNC-Session hat keinen Einfluss auf geöffnete Programme im Container.
3. Der Container hat keinen Zugriff zum Internet. Firefox kann aber via VNC verwendet werden, um lokale html-Dateien (z.B. für Codedokumentation) anzuschauen.
4. Der Ordner mit den Aufgaben ist gleichzeitig als lokales git-Repository konfiguriert. Dies erlaubt es Ihnen, Änderungen am Quelltext in der WebIDE nachzuverfolgen.
5. Sie können mit `./build.sh` auch Builds zum Debuggen oder mit Address Sanitizer erstellen und diese anschließend mit `./run.sh` ausführen. Dafür setzen Sie den gewünschten Modus als Environment-Variable, z.B. indem sie vorher `export MODE=debug` bzw. `export MODE=sanitize` ausführen.

Lokale Installation

Im folgenden Abschnitt finden Sie Hinweise für eine lokale Installation des Simulationsframeworks OMNeT++ und der Aufgabenstellungen auf einem Linux-Rechner (vorzugsweise Debian/Ubuntu). Die benötigten Dateien werden Ihnen in Form eines zip-Archivs im Moodle-Kurs bereitgestellt.

Anleitung

1. Das zip-Archiv an einem Ort Ihrer Wahl entpacken. Wir gehen in der weiteren Anleitung davon aus, dass der entpackte Ordner `protsim` heißt.
2. Ein Terminal öffnen und das Arbeitsverzeichnis mittels `cd` auf den entpackten `protsim`-Ordner setzen.
3. Für OMNeT++ benötigte Paket-Abhängigkeiten installieren (hierzu werden Administratorrechte benötigt, auf den Rechnern des Fachgebietes sind diese Pakete bereits vorhanden):
 - für Ubuntu (und ähnliche):
 - `cd binaries`
 - `./requiredPackets.sh`
 - andere Distributionen: Installation der Pakete (oder äquivalente):

```
- build-essential gcc g++ gdb bear bison flex perl python3
python3-numpy python3-scipy python3-pandas python3-matplotlib
python3-posix-ipc qt5-default libqt5opengl5-dev tcl-dev
tk-dev libxml2-dev zlib1g-dev default-jre doxygen graphviz
libwebkitgtk-1.0 moreutils xdg-utils gnuplot wget
```
4. OMNeT++ Installation
 - Wechseln in den Ordner mit den OMNeT++ Quellen (`cd binaries`)
 - Installationsskript aufrufen mit `./localInstall.sh` (Dies wird OMNeT++ in den Ordner `protsim/binaries/omnetpp` installieren)
5. nun kann man die jeweiligen Aufgabenstellungen, welche sich in den Verzeichnissen `protsim/protsim??` befinden mit
 - `./build.sh` kompilieren (für Debugging mit `gdb`: `MODE=debug ./build.sh` und
 - `./run.sh` starten (bzw. mit `MODE=debug ./run.sh` debuggen)
6. zum Bearbeiten der praktischen Aufgaben können Sie z.B. die OMNeT IDE verwenden (`protsim/binaries/omnetpp/bin/omnetpp` bzw. Desktop-Shortcut)
 - Installieren Sie INET *nicht*, falls Sie gefragt werden!
 - die einzelnen Projekte können über das Menü `File` → `Import` → `General` → `Existing Projects into Workspace` importiert werden (`root directory: protsim??`)

7. es gibt außerdem noch die Möglichkeit mit dem Skript `./updateQtProject.sh` im jeweiligen Protsim Ordner ein Projekt für den QtCreator zu erstellen
8. die API Dokumentation kann mit dem Befehl `doxygen` nach dem Wechsel in das jeweilige `protsim??/doc` Verzeichnis erstellt werden. Anschließend kann die Dokumentation (`protsim??/doc/api/index.html`) mit einem Browser geöffnet werden. Unter Umständen muss das Projekt vorher einmal neu kompiliert werden, um für alle Klassen eine Dokumentation zu erhalten. Insbesondere werden die Klassen zu Nachrichtentypen erst beim kompilieren aus den `msg` Dateien erstellt und können erst danach in die Dokumentation aufgenommen werden.

Hinweis: Eventuell muss man den Skripten vorher noch mit `chmod +x <scriptname.sh>` die benötigten Ausführrechte geben. Das Skript `run.sh` akzeptiert zudem weitere Parameter, die Sie im Zuge des Seminars kennenlernen und benutzen werden.