

# GloveVM

## DOKUMENTATION

KARLSRUHER INSTITUT FÜR TECHNOLOGIE

25. November 2017

# Inhaltsverzeichnis

<b>1</b>	<b>Herunterladen der VM</b>	<b>2</b>
<b>2</b>	<b>Benutzung der VM</b>	<b>2</b>
<b>3</b>	<b>Allgemeines zur VM</b>	<b>3</b>
<b>4</b>	<b>Installierte Software</b>	<b>3</b>
<b>5</b>	<b>Funktionsweise der Datenübertragung</b>	<b>4</b>
<b>6</b>	<b>Anleitung: Starten aller Anwendungen</b>	<b>4</b>
6.1	Starten des Servers und Verbindungsaufbau . . . . .	4
<b>7</b>	<b>Sonstige Software</b>	<b>7</b>
7.1	Git . . . . .	7
7.1.1	Git von der Kommandozeile nutzen . . . . .	7
7.1.2	Git mit GitKraken verwenden . . . . .	8

# 1 Herunterladen der VM

Die virtuelle Maschine wird auf Google Drive gehostet und verfügbar unter:

<https://goo.gl/Pg1WCJ>

## 2 Benutzung der VM

Um die VM starten zu können, wird ein VM Manager Programm wie beispielsweise VirtualBox benötigt. (<https://www.virtualbox.org/wiki/Downloads>) Nachdem die VM von obigem Link heruntergeladen wurde, muss VirtualBox gestartet werden. Anschließend muss sie zu VirtualBox hinzugefügt werden (1). Dafür muss in der oberen Menüleiste der Menüpunkt “Maschine” -> “Hinzufügen” ausgewählt werden. Anschließend muss das Abbild ( .vdi Datei) ausgewählt werden. Danach erscheint die VM auf der linken Seite des Fensters (2). Zum Schluss kann die VM durch Klicken des Buttons “Starten” gestartet werden (3).

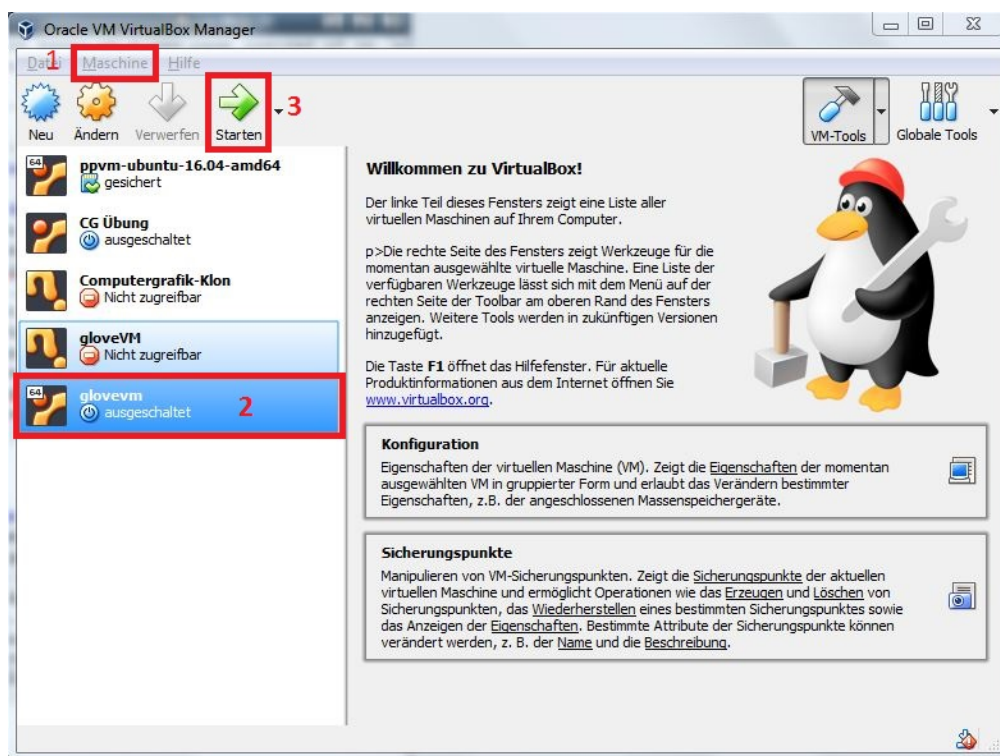


Abbildung 1: Übersicht VirtualBox Programm

### 3 Allgemeines zur VM

Die in dieser Dokumentation beschriebene virtuelle Maschine ist dafür gedacht, für das Entwickeln und Testen von Software für den virtuellen Handschuh genutzt zu werden. Alle dafür benötigten Programme sind bereits installiert. Für die Nutzung müssen keine zusätzlichen Anwendungen konfiguriert oder heruntergeladen werden.

Es ist möglich, den Glove zu simulieren, ohne, dass dieser benötigt wird. Die Vorgehensweise dazu wird in Kapitel .. näher erläutert. Dies bringt den Vorteil, standortunabhängig am Projekt arbeiten zu können. Außerdem entfällt der zeitaufwendige Prozess aller Installationen. Es ist lediglich nötig, die VM herunterzuladen und auszuführen.

Bei dem Betriebssystem der VM handelt es sich um Ubuntu 14.04.1 LTS. Der Benutzername ist `glovevm`, das Passwort ist ebenfalls `glovevm`.

### 4 Installierte Software

Alle für den Betrieb des Glove nötige Software ist bereits installiert. Dazu gehört:

Programm	Installationsverzeichnis
Code::Blocks	Menü > Entwicklung
PolyVR	/home/glovevm/polyvr
VRPN	/usr/include/vrpn
Serveranwendung	/home/glovevm/Rechner/Rechner/build
Gitkraken	Menü > Entwicklung

Code::Blocks ist die Umgebung, in der PolyVR gestartet wird.

PolyVR wird benötigt, um VR Anwendungen auszuführen, man braucht es also insbesondere, um die entwickelte Software zu testen.

VRPN ist das Übertragungsprotokoll, mit dem Daten zwischen einer Serverapplikation und PolyVR ausgetauscht werden, um die Signale des Handschuhs zu verarbeiten.

## 5 Funktionsweise der Datenübertragung

Der Handschuh überträgt mittels einfacher TCP-Pakete Daten an eine Serveranwendung, die auf dem Rechner läuft, auf dem später auch PolyVR ausgeführt wird. Die übermittelten Pakete haben dabei eine festgelegte Struktur. Jedes Paket besteht aus 10 Bits. Die Daten sind auf die Bits von links nach rechts wie folgt verteilt:

Bit 1: Button 1: Berührung Daumen, Zeigefinger

Bit 2: Button 2: Berührung Daumen, Mittelfinger

Bit 3: Button 3: Berührung Daumen, Ringfinger

Bit 4: Button 4: Berührung Daumen, kleiner Finger (??)

Bit 5: Channel 0: Flex Sensor 1 (Zeigefinger)

Bit 6: Channel 1: Flex Sensor 2 (Mittelfinger)

Bit 7: Channel 2: Flex Sensor 3 (Ringfinger)

Bit 8: Channel 3: Flex Sensor 4 (kleiner Finger)

Bit 9: Channel 4: Joystick X Koordinate

Bit 10: Channel 5: Joystick Y Koordinate

Dabei bedeutet jeweils ein Wert von 1, dass der Button gedrückt bzw. der Flexsensor gebeugt wurde.

Diese Pakete werden fortlaufend an die Serveranwendung geschickt, welche sie intern in VRPN konforme Pakete umwandelt und an PolyVR weiterleitet. Dort werden die Eingaben dann verarbeitet und visualisiert.

## 6 Anleitung: Starten aller Anwendungen

Im folgenden Abschnitt wird beschrieben, wie alle für den Betrieb des Handschuhs benötigten Anwendungen gestartet werden können.

### 6.1 Starten des Servers und Verbindungsaufbau

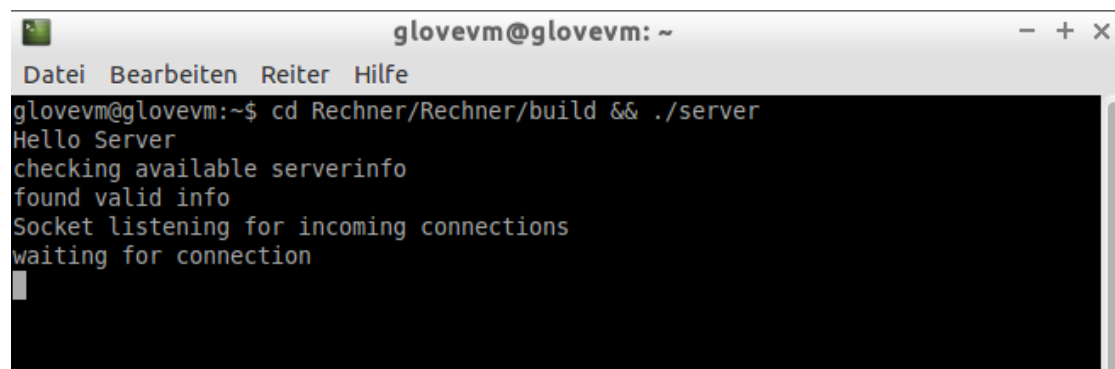
Nachdem ein Terminal geöffnet wurde, muss dort der Befehl `cd Rechner/Rechner/build && ./server` ausgeführt werden. Anschließend versucht der Server, eine Verbindung zum Handschuh aufzubauen. Um den Handschuh zu simulieren, muss nun

ein neues Terminal geöffnet werden. Dort wird der Befehl `netcat localhost 3490` ausgeführt. Nun denkt der Server, er würde mit dem Handschuh kommunizieren. Netcat baut eine Verbindung zu localhost auf Port 3490 auf.

Um den Verbindungsaufbau abzuschließen, muss die Zahl 69 eingegeben und mit Enter bestätigt werden. Dies nutzt der Server, um sicherzustellen, dass er mit dem richtigen Gerät kommuniziert.

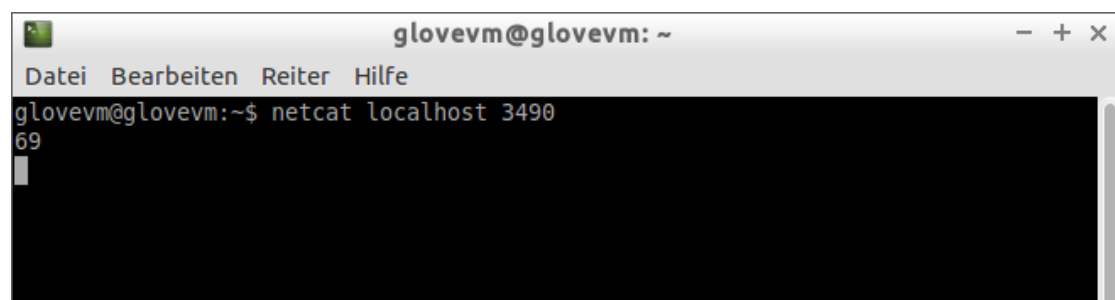
Danach ist der Verbindungsaufbau abgeschlossen. Der Ablauf ist nachfolgend nochmals visualisiert. Um nun Pakete vom simulierten Handschuh an den Server übermitteln zu können, wird einfach eine 10-stellige Sequenz von Ziffern in das Terminal, in dem `netcat` gestartet wurde, eingegeben und wieder mit Enter bestätigt. Eine 0 bedeutet, dass der jeweilige Button nicht gedrückt ist, eine 1 bedeutet, dass der Button gedrückt wurde. Für die Flexsensoren und Joysticks sind alle Ziffern von 0 bis 9 sowie Buchstaben möglich.

Zum Senden muss die Enter-Taste betätigt werden.



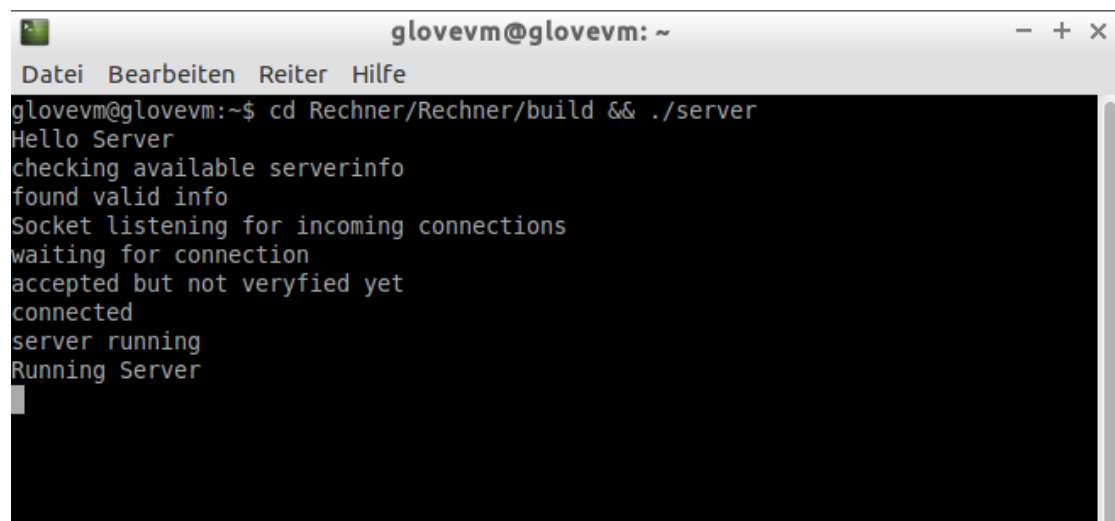
```
glovevm@glovevm: ~  
Datei Bearbeiten Reiter Hilfe  
glovevm@glovevm:~$ cd Rechner/Rechner/build && ./server  
Hello Server  
checking available serverinfo  
found valid info  
Socket listening for incoming connections  
waiting for connection  
█
```

Abbildung 2: Starten des Servers



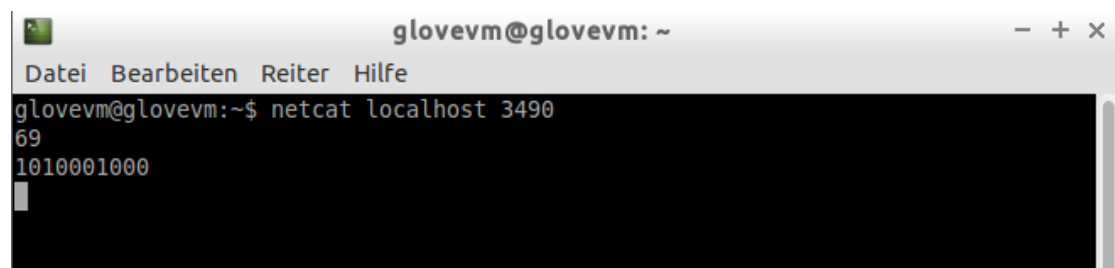
```
glovevm@glovevm: ~  
Datei Bearbeiten Reiter Hilfe  
glovevm@glovevm:~$ netcat localhost 3490  
69  
█
```

Abbildung 3: Simulieren des Handschuhs

A terminal window titled 'glovevm@glovevm: ~' with a menu bar containing 'Datei', 'Bearbeiten', 'Reiter', and 'Hilfe'. The terminal output shows the execution of 'cd Rechner/Rechner/build && ./server', followed by status messages: 'Hello Server', 'checking available serverinfo', 'found valid info', 'Socket listening for incoming connections', 'waiting for connection', 'accepted but not verified yet', 'connected', 'server running', and 'Running Server'.

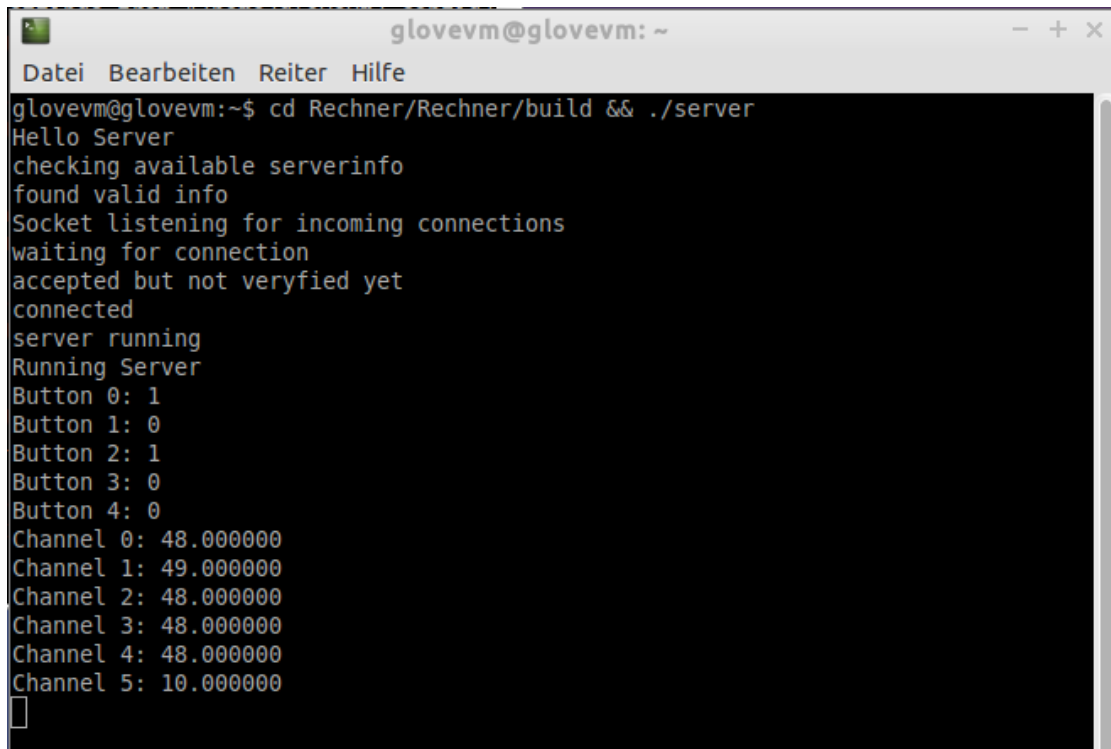
```
glovevm@glovevm: ~  
Datei Bearbeiten Reiter Hilfe  
glovevm@glovevm:~$ cd Rechner/Rechner/build && ./server  
Hello Server  
checking available serverinfo  
found valid info  
Socket listening for incoming connections  
waiting for connection  
accepted but not verified yet  
connected  
server running  
Running Server
```

Abbildung 4: Abgeschlossener Verbindungsaufbau

A terminal window titled 'glovevm@glovevm: ~' with a menu bar containing 'Datei', 'Bearbeiten', 'Reiter', and 'Hilfe'. The terminal output shows the execution of 'netcat localhost 3490', followed by the received data: '69' and '1010001000'.

```
glovevm@glovevm: ~  
Datei Bearbeiten Reiter Hilfe  
glovevm@glovevm:~$ netcat localhost 3490  
69  
1010001000
```

Abbildung 5: Beispielhaftes Senden eines Datenpakets



```
glovevm@glovevm:~$ cd Rechner/Rechner/build && ./server
Hello Server
checking available serverinfo
found valid info
Socket listening for incoming connections
waiting for connection
accepted but not verified yet
connected
server running
Running Server
Button 0: 1
Button 1: 0
Button 2: 1
Button 3: 0
Button 4: 0
Channel 0: 48.000000
Channel 1: 49.000000
Channel 2: 48.000000
Channel 3: 48.000000
Channel 4: 48.000000
Channel 5: 10.000000
```

Abbildung 6: Anzeigen des gesendeten Datenpakets beim Server

## 7 Sonstige Software

### 7.1 Git

Auf der VM befindet sich eine Installation der Versionsverwaltungssoftware Git. Diese kann über die Kommandozeile oder mithilfe der installierten Version von GitKraken verwendet werden.

#### 7.1.1 Git von der Kommandozeile nutzen

Zunächst ist wichtig dass alle Git-Befehle im Hauptverzeichnis des Repositories gemacht werden. Um das Cyberglove-Repository herunterzuladen und dort hinein zu wechseln reicht der Befehl `git clone http://imi-dev.imi.kit.edu:443/vrp1718/Cyberglove.git && cd Cyberglove`.

Wenn an dem Code Änderungen vorgenommen wurden, können diese mittels



`git add . && git commit -m "Beschreibung der Änderungen" && git pull && git push` gesichert werden. Dieser Befehl sichert zunächst die Änderungen lokal in einem neuen Commit, holt dann Änderungen aus dem Online-Repository und führt die Änderungen zusammen, und lädt schließlich die Änderungen hoch.

Ein Cheatsheet mit den wesentlichen Git-Befehlen findet sich hier:<sup>1</sup> Auf <https://try.github.io> findet man ein einsteigerfreundliches Tutorial zur Funktionsweise von Git, welches sich mit relativ wenig Aufwand durchlaufen lässt.

### 7.1.2 Git mit GitKraken verwenden

Eine Alternative bietet die grafische Benutzeroberfläche GitKraken, die als mächtige Gitoberfläche den Prozess der Versionsverwaltung zwar um einiges übersichtlicher gestaltet, aber für Git-Einsteiger nicht ganz intuitiv ist.

GitKraken kann über den Menüpunkt Entwicklung oder durch den Befehl `sudo gitkraken` gestartet werden.

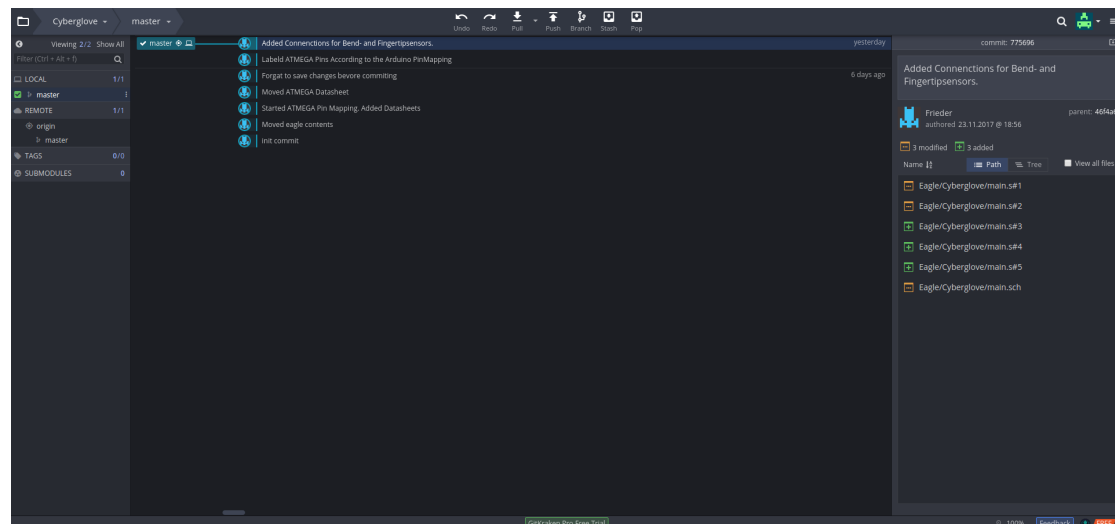


Abbildung 7: Benutzeroberfläche von GitKraken

Um ein neues Repository herunterzuladen, muss man in GitKraken in 8 auf das Ordnersymbol oben links (grüner Kreis) klicken, im geöffneten Fenster Repository Management dann zum Menüpunkt Clone (orangener Kreis) wechseln und dann

<sup>1</sup><http://lukasbach.com/downloads/vcs/git-cheatsheet.pdf>

bei URL `http://imi-dev.imi.kit.edu:443/vrp1718/Cyberglove.git` eingeben.  
Mit Clone the repo! kann man das Herunterladen dann starten.

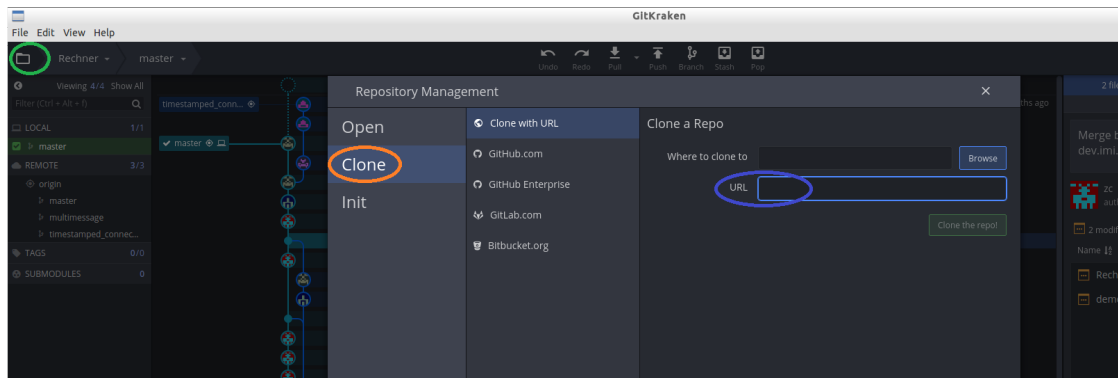


Abbildung 8: Einrichten eines neuen Repositories in GitKraken