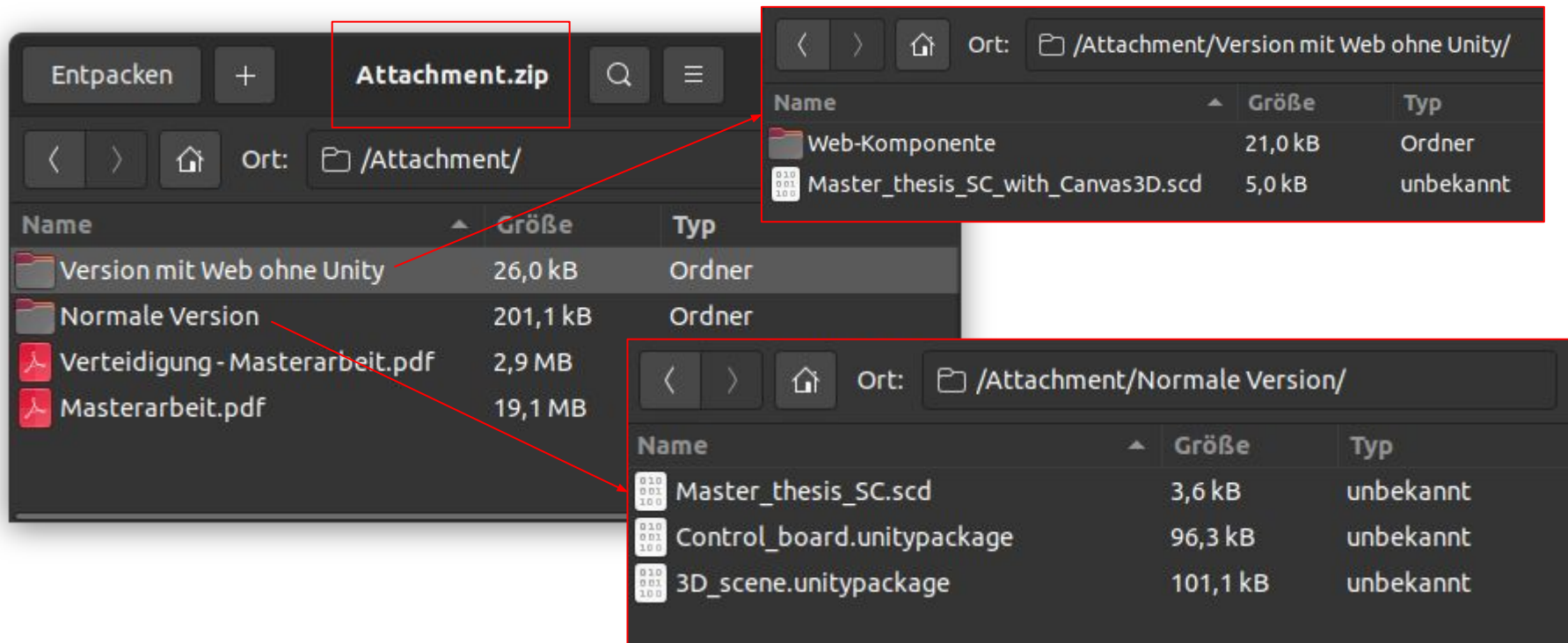


Hinweis

Inhalt von "Attachment.zip"



Code für die "normale" Version mit Unity und die Alternative ohne Unity (mit Webkomponente). **scd**-Dateien sind Code in SuperCollider.

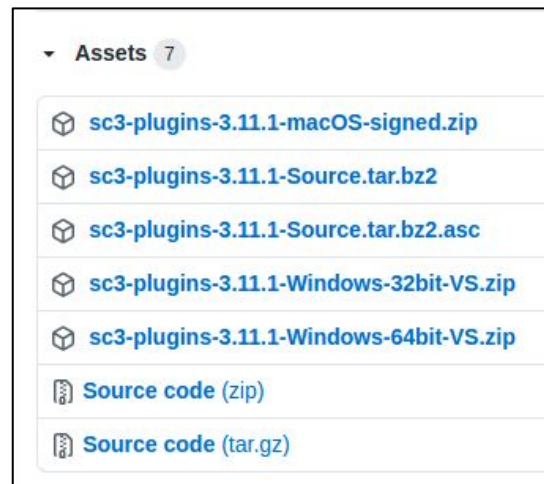
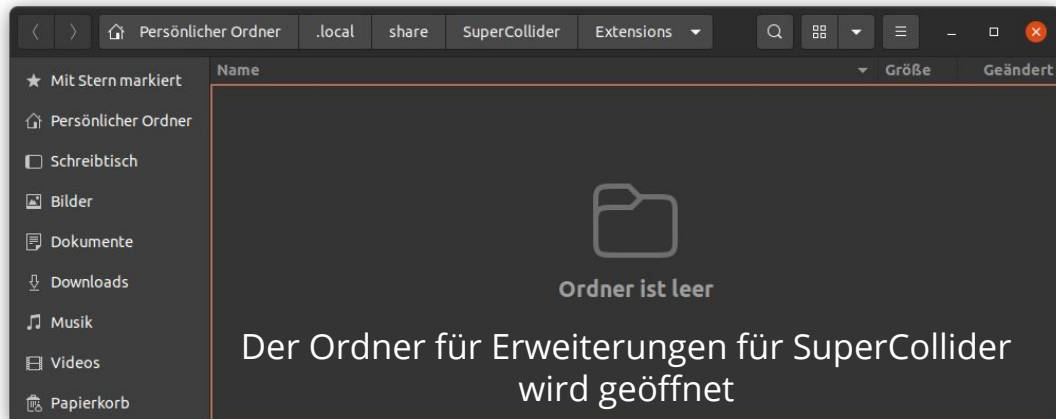
Zusätzliche Abhängigkeiten in SuperCollider

- SuperCollider Plugins
 - <https://github.com/supercollider/sc3-plugins/releases/latest>
- Ambisonic Toolkit (ATK)
 - <https://github.com/ambisonictoolkit/atk-sc3>
- Canvas3D (nur für die Version mit Web-Komponente und ohne Unity)
 - <https://github.com/supercollider-quarks/Canvas3D>
- Mosca (optional - ein schönes Plugin zum Test von Ambisonics mit GUI direkt in SuperCollider)
 - <https://github.com/escuta/mosca#begginer-tips-to-install-from-scratch>

SuperCollider-Plugins

```
File Session Edit View Language Server Help  
1 Platform.userExtensionDir.openOS
```

Das Kommando **Platform.userExtensionDir.openOS** in SuperCollider ausführen

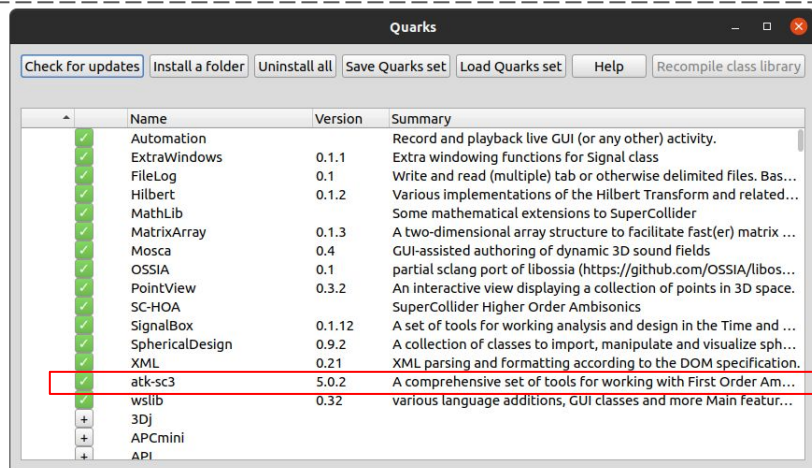


SC-Plugins von: <https://github.com/supercollider/sc3-plugins/releases/latest> je nach Betriebssystem herunterladen und in diesem Ordner entpacken (Für Ubuntu/Linux muss man selbst vom Sourcecode bauen).

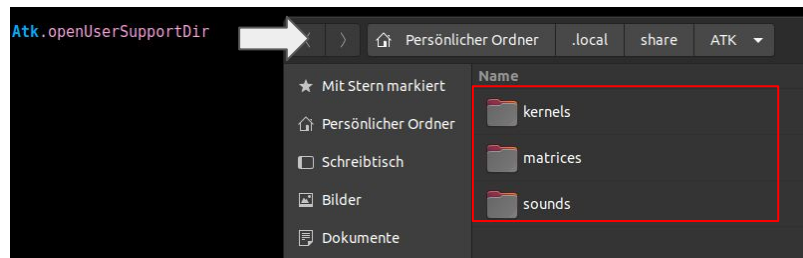
Ambisonic Toolkit

- Befehl zur Installation in SuperCollider:
 - `Quarks.install("https://github.com/ambisonictoolkit/atk-sc3.git");`
 - Evtl. muss auch **git** aufm Rechner installiert werden
- Auch die Kernels, Matrizen und Sounds-Komponente von ATK müssen installiert/ heruntergeladen werden.
- Mehr Info unter: <https://github.com/ambisonictoolkit/atk-sc3#installing>
- Falls Fehler auftritt (z.B. Class "ATK" not defined), sollen die gesamte Bibliothek-Klassen in SuperCollider neu kompiliert werden (Unter Menü **Language** => **Recompile Class Library**)

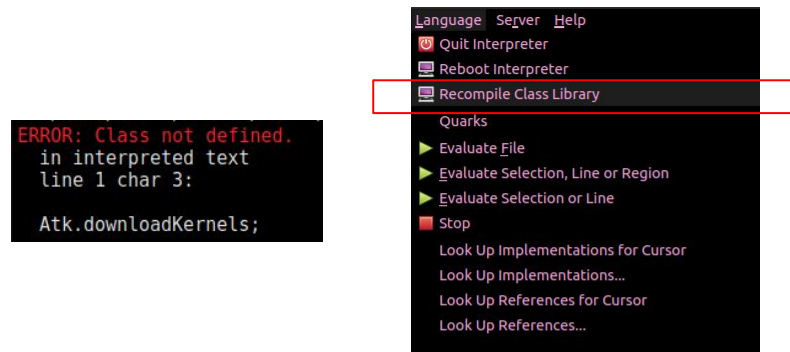
Ambisonic Toolkit (vielleicht hilfreiche Anmerkungen)



Unter **Language => Quarks** kann man sehen, ob **atk-sc3** installiert wurde oder nicht.



Mit **Atk.openUserSupportDir** kann geprüft werden, dass die 3 Komponente tatsächlich jetzt vorhanden sind.



Eine neue Kompilierung aller Bibliotheksklassen für SuperCollider kann bei solchen Fehlern helfen.

Canvas3D

```
File Session Edit View Language Server Help
Canvas3D.scd x Untitled x Post window
1 Quarks.install("Canvas3D");
Installing Canvas3D
Canvas3D installed
-> Quark: Canvas3D[0.9]
```



Ist die Hilfsdatei für Canvas3D nicht mit "Search" im "Help Window" von SC zu finden, kann man unter dem Menü **Language => Quarks => "Canvas 3D"** wählen => Unten auf **"Open help"** klicken => Hilfsdatei wird geöffnet

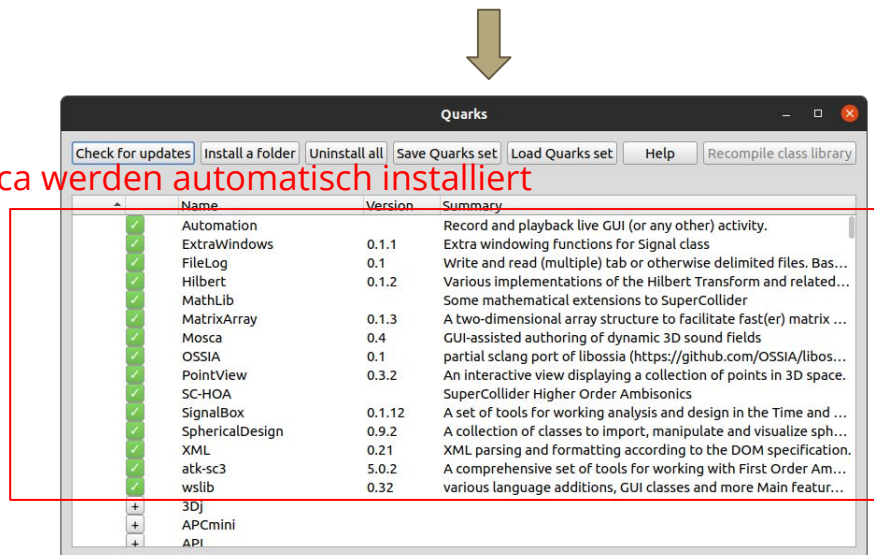
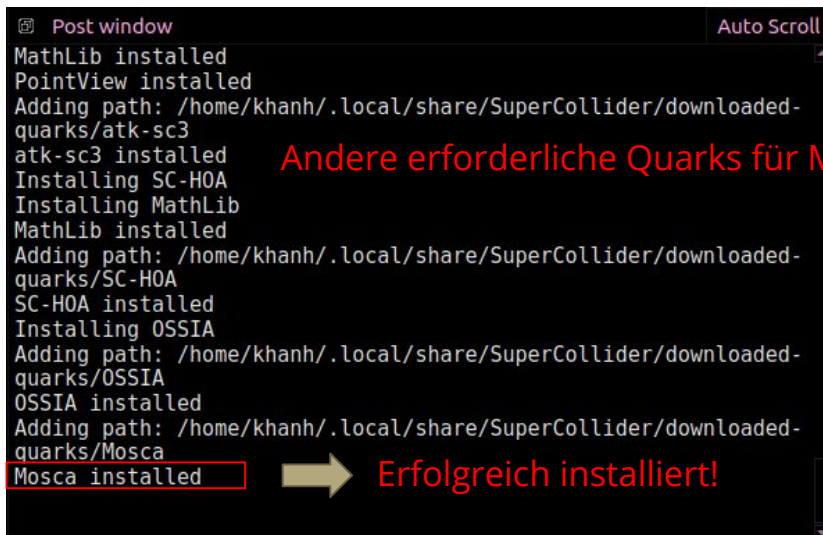
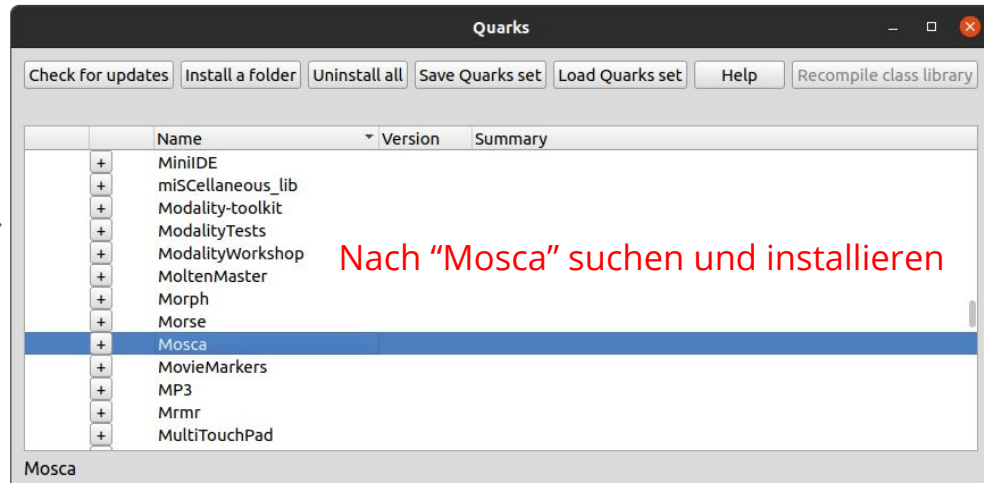
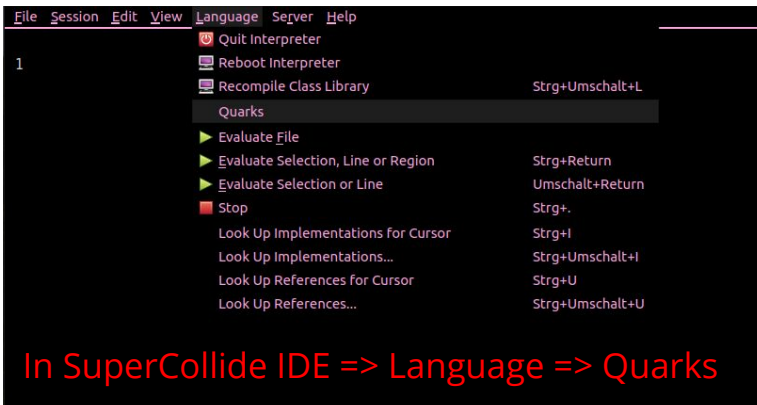
The top part of the image shows the SuperCollider web interface for the Canvas3D extension. It includes a search bar, a table of contents, and a description of the extension. A red arrow points from the 'Canvas3D' entry in the table of contents to the 'Open help' button in the Quarks menu.

The bottom part of the image shows the Quarks menu. It has a table with columns for Name, Version, and Summary. The 'Canvas3D' entry is highlighted. Below the table, there are buttons for 'Help', 'Classes', 'Extensions', 'schelp', 'since', and 'author'. The 'Help' button is highlighted, and a red arrow points to it.

Name	Version	Summary
Automation		Record and playback live GUI (or any oth
Canvas3D	0.9	3D Wire-frame vector graphics canvas

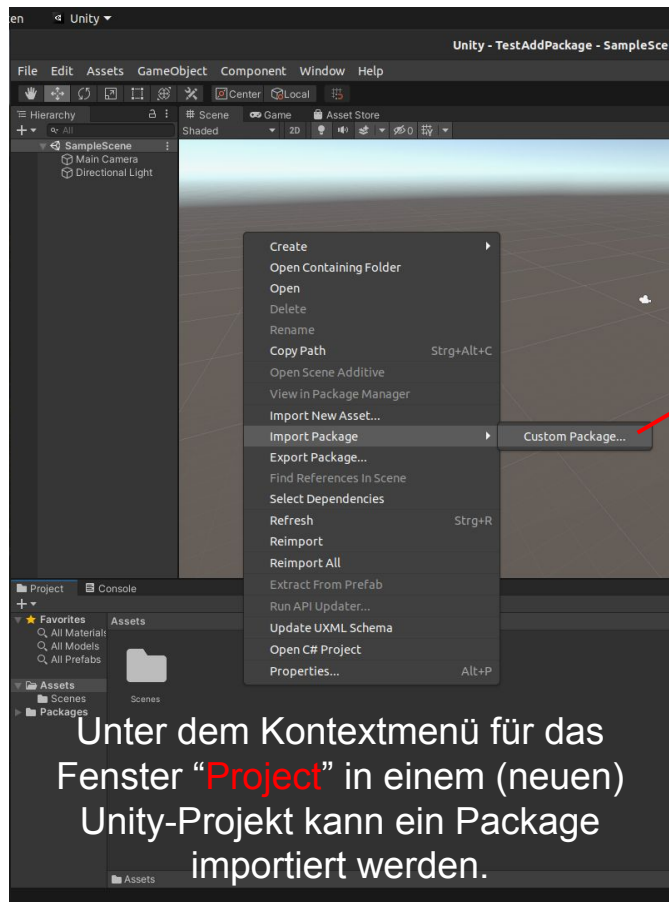
Canvas3D	
Help	Open help
Classes	Show classes
Extensions	Show extension methods
schelp	
since	Classes/Canvas3D
author	2011
	Jonatan Liljedahl

Mosca (optional)

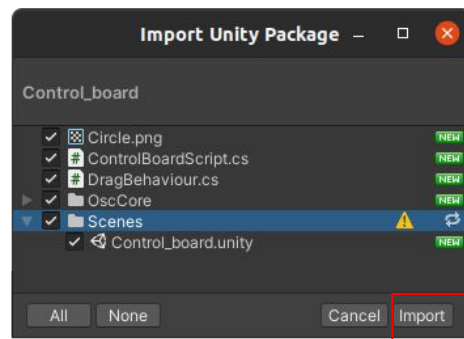


Unity-Package in einem Projekt importieren (1)

Hier z.B. das Package für die Steuerung-GUI wählen

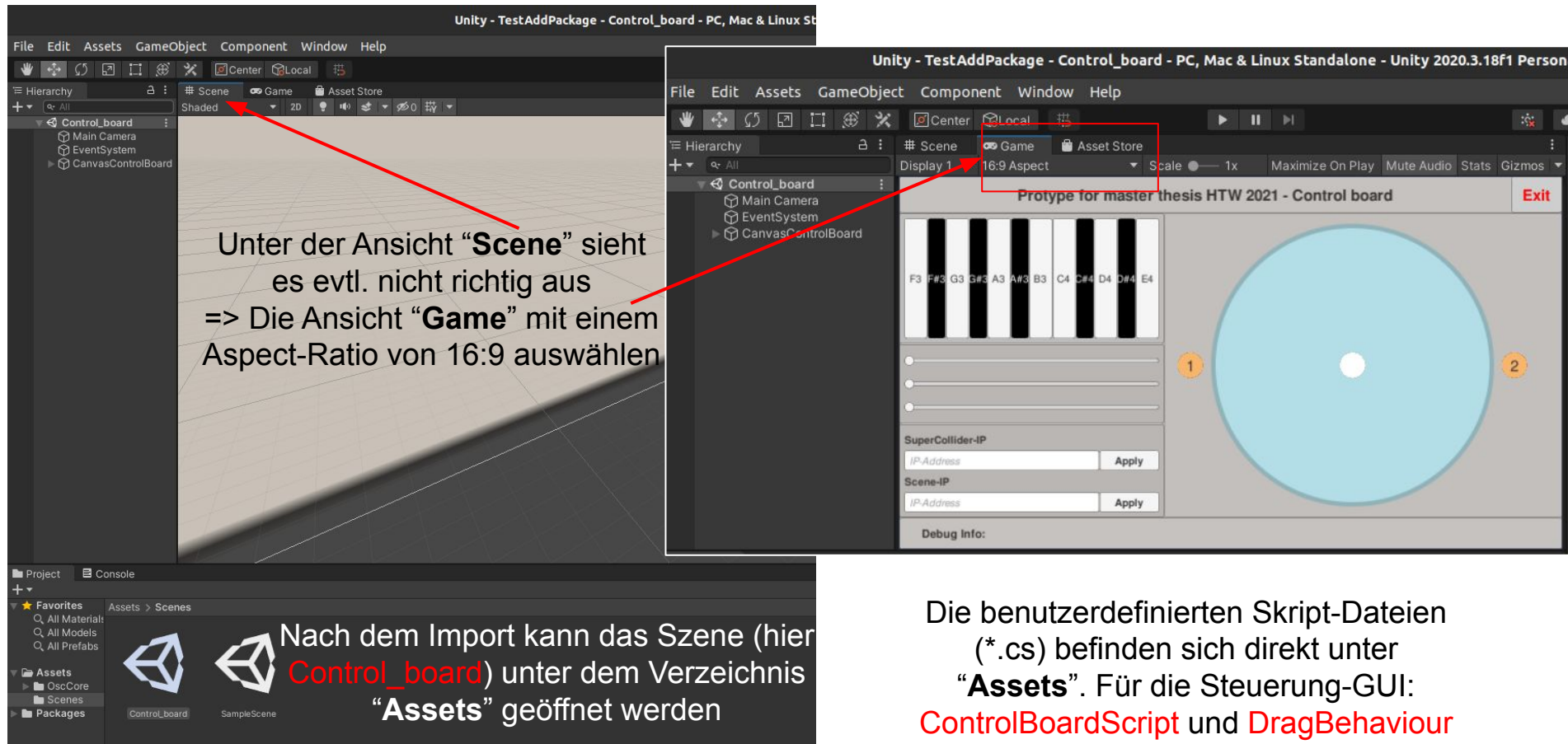


Name	Größe	Typ
Master_thesis_SC.scd	3,6 kB	unbekannt
Control_board.unitypackage	96,3 kB	unbekannt
3D_scene.unitypackage	101,1 kB	unbekannt



Die Bibliothek **OscCore** ist auch schon drin.
(<https://github.com/stella3d/OscCore>)

Unity-Package in einem Projekt importieren (2)

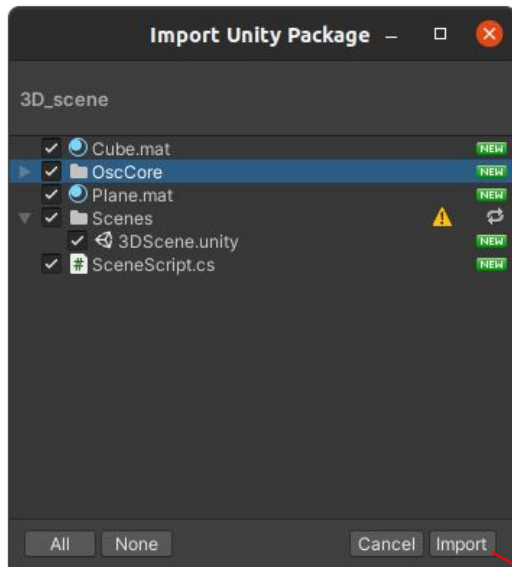


Unter der Ansicht “**Scene**” sieht es evtl. nicht richtig aus
=> Die Ansicht “**Game**” mit einem Aspect-Ratio von 16:9 auswählen

Nach dem Import kann das Szene (hier **Control_board**) unter dem Verzeichnis “**Assets**” geöffnet werden

Die benutzerdefinierten Skript-Dateien (*.cs) befinden sich direkt unter “**Assets**”. Für die Steuerung-GUI: **ControlBoardScript** und **DragBehaviour**

Unity-Package in einem Projekt importieren (3)



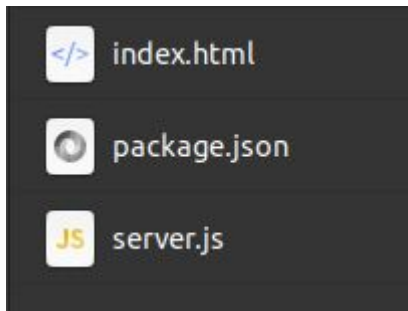
Das gleiche Prinzip für den Import gilt auch für das 3D-Szene (in einem anderen (neuen) Unity-Projekt). Für das 3D-Szene gibt es nur eine benutzerdefinierte Skript-Datei:

SceneScript.cs

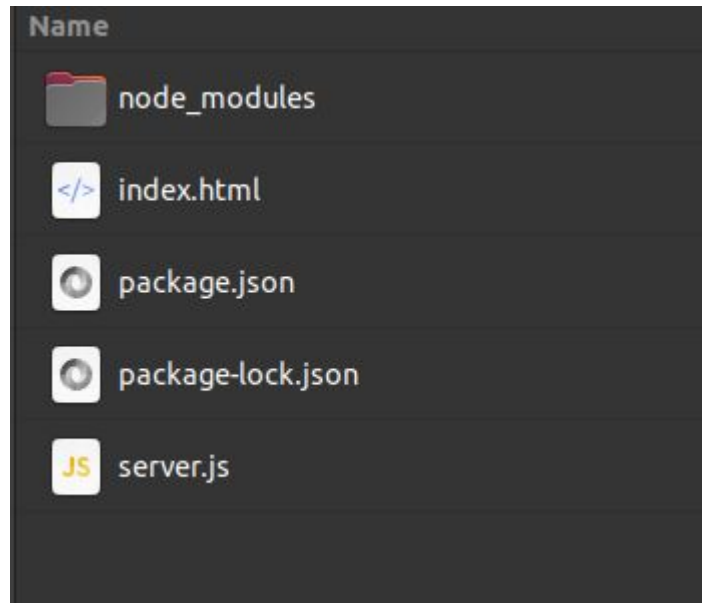


Alternative Herangehensweise ohne Unity (1)

- In SuperCollider ist Canvas3D in diesem Fall nötig (und natürlich auch ATK)
- Für die Webkomponente ist node.js nötig
- Installation von node.js: <https://nodejs.org/en/download/>



Diese 3 Dateien sollen unter gewünschten Ordner kopiert. Dort per Kommandozeilen mit “**npm install**” lassen alle Abhängigkeiten installieren (hier sind **express** und [osc.js](#) nötig). Danach soll der Ordnerinhalt ungefähr so aussehen.



Alternative.... (2)

```
Kommandozeile

Computer name $ node server.js
Express server started at port 3000
Sending message [ { type: 's', value: 'Hello from control board (web)!'} ] to
Sending message undefined to 127.0.0.1:57120 on /stopAll
```

Mit dem Befehl “**node server.js**” soll die Webkomponente auf
“**localhost:3000**” in einem Webbrowser abrufbar sein.
Die Quelltextdatei in SuperCollider unter Verwendung von
Canvas3D heißt “**Master_thesis_SC_with_Canvas3D.scd**”

