

# User Guide für das ProKlaue Maya Plugin - ab Maya 2019

Installation unter Windows

Kai Hainke

13. Mai 2019

## INSTALLIEREN DES PLUGINS

Zur Installation des Plugins benötigt man den vollständigen Projektordner (dies beinhaltet die Dateien aus dem Git und die Packages numpy, scipy und sklearn jeweils in ihrem Unterordner in `proKlaueInstall\site-packages`, sowie ggf. die Executables in `bin`. Aus lizenz-rechtlichen Gründen müssen diese Dateien aus anderen Quellen besorgt werden, für die Packages bspw. hier. Dann verfährt man wie folgt:

0. Wenn nicht bereits dort zu finden, verschiebt man die Packages, so dass numpy, scipy und sklearn in ihren jeweiligen Unterordnern unter `proKlaueInstall\site-packages` liegen
1. Man speichert alle Projektdateien in Form des `ProKlaue` Ordners an einer beliebigen Stelle, bspw. unter `C:\projects\ProKlaue`
2. Dann führt man `proKlaueInstall\install.bat` aus. Für eine Standard-Maya-Installation im Standard-Pfad muss man hier nichts weiter tun. Andernfalls wird man gebeten den Pfad zum Maya-Python-Interpreter zu spezifizieren, standardmäßig zu finden unter `C:\Program Files\Autodesk\Maya2019\bin\mayapy.exe`.
3. Nun fügt man den Pfad zum `ProKlaue` Ordner aus Schritt 1, der `MAYA_MODULE_PATH` Umgebungsvariable hinzu (bzw. erstellt diese, falls sie noch nicht bereits existiert). Dies geschieht unter Systemsteuerung, System, Erweiterte Systemeinstellungen, Erweitert, Umgebungsvariablen (Systemvariablen für eine Installation für alle Nutzer, Benutzervariablen für eine lokale Installation).
4. Nach einem erneuten Start von Maya wird nun das `ProKlaue` Plugin im Plugin-Manager-Window angezeigt. Dies muss nun geladen werden und kann dann verwendet werden:
  - a) In Maya unter Windows → Settings/Preferences → Plugin Manager → Refresh.
  - b) Nach dem Eintrag `proKlaue.py` suchen und `Loaded` und `Auto load` auswählen.
  - c) Zuletzt Windows → Settings/Preferences → Preferences → Settings → Selection → `Track selection order` aktivieren.

## ZU DEN PYTHON POSTPROCESSING SCRIPTS

Für das Ausführen der Python Scripts zum Postprocessing (Druck Statistiken) wird zusätzlich ein Python Interpreter  $\geq 2.7.11$  (nicht Python 3) benötigt. Mit folgenden packages:

- numpy
- scipy
- scikit-learn (sklearn) (momentan nur benutzt bei den Achsen und Druck Scripts)
- pyclicker

Dafür sind nach der Installation von Python sowohl das Python-Basis-Verzeichnis (`C:\Python27`) als auch das Scripts-Verzeichnis (`C:\Python27\Scripts`) zur PATH Variable hinzuzufügen (unter Systemsteuerung, System, Erweiterte Systemeinstellungen, Erweitert, Umgebungsvariablen, Systemvariable, falls für alle Benutzer oder Benutzervariable, falls nur für den derzeitigen Nutzer). Die packages sind am einfachsten über pip zu Installieren, mittels folgender Befehle in einer Eingabeaufforderung:

- `pip install intel-numpy`
- `pip install intel-scipy`
- `pip install intel-scikit-learn`
- `pip install pyclicker`

## INFORMATIONEN ZUM PROJEKT

### 3.1 Struktur

Einen groben Überblick gibt die folgende Liste:

- `bin` - Executables für die VHACD Berechnung
- `doc` - Sphinx Dokumentation
- `doc\_user` - Guides, Latex+pdf-Dateien
- `postprocessing` - Scripts für die Nachbearbeitung von Daten aus Maya (R/Python):
  - `pressureStatistics.py` - Python Skript für die Zusammenführung von Fußungsfläche und Druckdaten (mit `pressureStatisticsUtilFunctions.py` für Funktionsdefinitionen)
  - `druck.R` - R Skript für die Zusammenführung von Fußungsfläche und Druckdaten
  - `Zones_plots_and_data.R` - R Skript für die statistische Auswertung der Druckdaten (auf den vorher festgelegten Fußungszonen)
  - `JCS_plots_and_data.R` - R Skript für die Auswertung der Joint-Coordinate-Systems-Daten
  - `plotHeatmaps.R` - R Skript zum Plotten von Heatmaps mit Hilfe vorher erstellter Daten aus dem Maya-Command `cmds.altitudeMap`
- `processing` - Scripts für das Bearbeiten und Auslesen von Daten in Maya (R/Python):
  - `angles.py` - Script zum Tracken der Winkel in einer animierten Szene mit JCS (nach Groot und Suntay)
  - `calculateJointCS.py` - Python Skript für die Berechnung von Joint Coordinate Systems
  - `axesToAnimated.py` - Python Skript für die Überführung von berechneten JCS von einer Szene in eine andere (bspw. von neutral zu animiert)
- `proKlaueInstall` - Ordner mit Installationsdateien
- `proKlaueModule` - Ordner mit den Dateien des Maya-Modules
  - `scripts\pk_src` - Ordner mit Maya Commands des Plugins, insbesondere
    - \* `overlapStatistics.py` zum Erstellen von Statistiken zur Überdeckung von zwei oder mehreren Objekten (relativ robuste Berechnung von Schnittvolumen), siehe auch `intersection.py` und `vhacd.py`

- \* `altitudeMap.py` zum Erstellen von Höhenmaps
- \* `frontVertices.py` zum Abspeichern aller Vorderseiten-Segmente/Punkte von einer gegebenen Plane aus
- \* `normalize.py` zum Ausrichten von Objekten anhand der Eigenvektoren ihrer Kovarianz-Matrix
- \* `misc.py` mit verschiedenen, nützlichen, all-purpose Funktionen
- `plug-ins` - Ordner mit den Plugins für das ProKlaueModul, hier lediglich das ProKlaue-Plugin, mit `proKlaue.py` für das Registrieren des Plugins in Maya

### 3.2 Nützliche Dev-Tools

Für die Entwicklung der Scripts/Plugins bieten sich einige Tools an, lediglich eine Empfehlung, aber vlt. hilft es ja:

- RStudio für R
- PyCharm für Python
- MayaCharm (PyCharm Plugin zur Interaktion mit Maya)
- Sublime für das Anzeigen großer Textdateien
- Sphinx für Dokumentation
- git zur Versionierung

## WEITERE INFORMATIONEN ZUR INSTALLATION VON PYTHON PACKAGES FÜR DEN MAYA PYTHON INTERPRETER

Dieses Kapitel soll weiterführende Informationen festhalten zur Installation von packages für den Maya-Python-Interpreter. Für eine einfache Installation sind diese Informationen (zur Zeit) nicht relevant, sie könnten jedoch in Zukunft relevant werden, für neuere Maya Versionen.

Grundsätzlich gibt es drei Varianten, die benötigten python libs zu installieren. Die erste Möglichkeit ist einen möglichst exakt gleichen Python-Interpreter (Version, Compiler-Version, Bit-Architektur, etc.) zu bauen und die Packages dafür zu installieren und dann zu kopieren. Die zweite Variante ist die Installation mittels pip aus der Maya-Installation (`C:\Program Files\Autodesk\Maya2019\Python\Scripts`). Diese Möglichkeit führt unter Windows jedoch für Pakete, die auf compilierte Algebra Bibliotheken zurückgreifen, zu Fehlern, da standardmäßig numpy ohne Linkage zur Intel Math Kernel Library installiert wird. Deshalb benötigt man Installation Wheels wie von z. Bsp. hier. Das Problem hierbei ist, dass der Maya-Python-Interpreter NICHT mit den Standard Microsoft Visual Studio Versionen gebaut wurde. Es bleibt demnach lediglich das selbst bauen von Python und numpy, scipy und scikit-learn. Das Bauen von Python lässt sich erleichtern durch ein cmake Buildsystem wie hier. Dies gestaltet sich jedoch ebenfalls sehr schwierig, weshalb die momentan verwendeten Packages aus dieser Quelle stammen und nach `C:\Program Files\Autodesk\Maya2019\Python\Lib\site-packages` verschoben werden. Die zur Zeit benötigten packages sind:

- numpy
- scipy
- scikit-learn (sklearn) (momentan nur benutzt bei den Achsen und Druck Scripts)

Für mehr Informationen ist die Dokumentation zu konsultieren. Zu finden unter `ProKlaue\doc\_build\html\index.html`.