

GDLnucleus für MAC OS

Task für Visual Studio Code für MAC OS X zum parallelen Bearbeiten von GDL-Objekten in ArchiCAD und Visual Studio Code

Für den freien Texteditor Visual Studio Code von Microsoft gibt es ein Syntax-Paket von Graphisoft für GDL-Entwickler.

Visual Studio Code (VSC) gilt bei vielen freien und professionellen Entwicklern aller möglichen Programmiersprachen, in erster Linie aber wohl für Webapplikationen, als ultimativer Editor.

VSC ist extrem vielseitig und extrem nutzeranpassbar, besitzt zudem die Möglichkeit Extensions automatisch zu installieren und up to date zu halten.

Das Paket von Graphisoft (<https://marketplace.visualstudio.com/items?itemName=GRAPHISOFT.gdl>) enthält das komplette Syntaxhighlighting für alle GDL-Befehle und globalen Variablen. Außerdem gibt es Code-Snippets und Autovervollständigen von Code.

Damit VSC mit dem vorliegenden Paket von Graphisoft effektiv zur GDL-Programmierung genutzt werden kann, sollte es in irgendeiner Form eine direkte Anbindung an ArchiCAD geben. Da externe Editoren theoretisch nur über die API-Schnittstelle direkt mit ArchiCAD in Verbindung treten können und uns diese Lösung zu aufwändig und vor allem zu teuer erschien, haben wir eine indirekte Verknüpfung mit ArchiCAD hergestellt, wobei die GDL-Objekte über das LP_XML-Converter-Tool (im folgenden XML-Converter genannt) von ArchiCAD nach VSC und visa versa übertragen werden.

Die Automatisierung dieses Prozesses übernimmt das hier beschriebene Task-System, welches alle notwendigen Konvertierungsschritte auf Knopfdruck durchführt und so eine reibungslose Arbeit in der Weise erlaubt, dass man jederzeit verlustfrei von ArchiCAD nach VSC und zurück wechseln kann.

In diesem Manual wird ausschließlich die Funktionsweise des Task-Systems für VSC und MAC OS X beschrieben.

Funktionen und Eigenschaften

- Ausgelöst durch eine Task in VSC werden GSM-Dateien, welche in ArchiCAD erzeugt oder bearbeitet wurden per XML-Converter in XML-Dateien umgewandelt und anschließend in

Einzelscriptdateien (3D-Script, 2D-Script, Master-Script etc.) aufgesplittet, so dass sie direkt in VSC weiter bearbeitet werden können. Alte XML-Dateien werden als Backup gesichert, so dass zu jedem angestoßenen Task-Prozess immer ein Backup vorhanden ist.

- Ausgelöst durch eine Task in VSC werden sämtliche Einzelscriptpakete incl. Parameter, Header usw. wieder zu einer einzigen XML-Datei zusammengesetzt und per XM_-Converter zurück in eine GSM-Datei konvertiert. Alte GSM-Dateien werden als Backup gesichert, so dass zu jedem angestoßenen Task-Prozess immer ein Backup vorhanden ist.
- Ausgelöst durch eine Task in VSC wird eine vollständige Parameter-Liste im CSV-Format ausgegeben
- Ausgelöst durch eine Task in VSC werden die in speziellen Ordnern abgelegten Grafiken automatisch im Binärformat in die GSM-Dateien eingebettet.
- Gleichzeitiges paralleles Bearbeiten von mehreren GSM-Dateien innerhalb eines VSC-Projektes

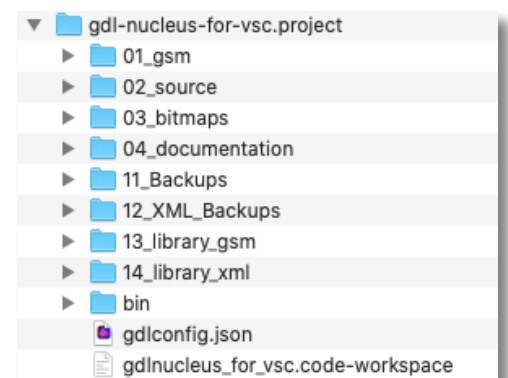
Installation

Als erstes müssen Sie VSC installieren, falls noch nicht geschehen. Laden Sie sich die neueste Version von VSC herunter unter <https://code.visualstudio.com/> und installieren das Programm entsprechend der Anleitung des Herstellers.

Als zweites müssen Sie das GDL-VSC-Package von Graphisoft installieren. Gehen Sie dazu in VSC auf Extensions >> Suchen >> GDL und wählen und installieren die GDL Extension von Graphisoft.

Weiterhin wird empfohlen, die Extension „Taskrunner“ oder „Task Explorer“ für VSC zu installieren, sowie die Extension „Trigger Task on Save“.

Von uns haben Sie eine Zip-Datei bekommen, die entpackt die folgend abgebildete Verzeichnis-Struktur enthält.



Kopieren Sie den entpackten Ordner als Template für jedes neue GDL-Objekt, an dem Sie arbeiten. Alle notwendigen globalen Anpassungen, die Sie selber vornehmen, spielen Sie bitte am Ende der Anpassungen wieder auf dieses Template zurück, damit Sie immer auf dem aktuellen Stand sind.

Folgende Anpassungsschritte sind erforderlich, damit Sie Ihr Template an Ihre Bedürfnisse anpassen und immer mit dieser Grundlage Projekte starten:

1. Öffnen Sie die Datei `gdlconfig.json` im Hauptverzeichnis. Ändern Sie hier den Pfad Ihres bevorzugten XML-Converters. Achtung: Grundeingestellt ist der XML-Converter für ArchiCAD 22. Wenn Sie Objekte in ArchiCAD 23 oder höher erstellen, müssen Sie den XML-Converter 23 oder höher benutzen. In dieser Datei können Sie auch ein Passwort für die Verschlüsselung des GDL-Objektes entsprechend dem vorgegebenen Beispiel vergeben. Falls das Objekt bereits ein Passwort besitzt, muss dies zwingend hier eingegeben werden, da andernfalls die Konvertierung fehlschlägt. Außerdem sind in dieser Datei die Namen zusätzlicher Backup-Verzeichnissen angegeben.

2. Sie können bei Bedarf einige Vorgaben in der `gdlconverter.py` im Order bin ändern, müssen das aber nicht. Zeile 15 müssen Sie in jedem Projekt ohnehin jedesmal neu anpassen. Das ist der Pfad der GSM-Dateien und deren Makros. Bitte bei jedem Projekt andere Namen des Unterordners „`gdlnucleus`“ wählen, „`01_gsm`“ bitte belassen. Zeile 16 - 24: Hier können Sie bei Bedarf die Ordernamen in Hochkommas anpassen, müssen dann aber auch die Ordernamen selber umbenennen. Zeile 33 - 40: Im vorderen Teil stehen die Namen der Source-Dateien, die in der Reihenfolge 0-9, A-Z im Source-Ordner angeordnet werden; Empfehlung: wichtige Scripte nach oben, unwichtige nach unten. Im hinteren Teil befinden Sie die Script-Namen: diese bitte NICHT ändern.

3. Nach diesem Schritt sollten Sie alles sichern und Ihre Template-Datei aktualisieren. An dem Template sind dann nur Änderungen erforderlich, falls Sie die Version des XML-Converters wechseln wollen oder andere

globale Anpassungen vornehmen möchten.

Folgende Anpassungsschritte sind erforderlich, bevor Sie mit einem GDL-Objekt starten können:

1. Benennen Sie den kopierten Ordner „`gdl-nucleus-for-vsc.project`“ um in den Namen Ihres Projektes, z.B. „`Flachdachhauben.project`“
2. Den Unterordner „`gdlnucleus`“ im Ordner „`1_gsm`“ umbenennen. Dieser Ordner ist später samt Unterordner in die geladene ArchiCAD-Bibliothek zu laden, z.B. in „`Flachdachhauben`“
3. Öffnen Sie die Datei `gdlconverter.py` im Order bin: Ändern Sie Zeile 15 entsprechend Punkt 2 der Allgemeinen Einstellungen.
4. mind. eine GSM-Datei in den Ordner `01_gsm/Flachdachhauben` ziehen; Makros in den Makros-Ordner ziehen. (Sämtliche GSM-Files sind vor Nutzung in VSC vorher in ArchiCAD anzulegen)
5. VSC starten. In VSC: Datei -> Arbeitsbereich aus Datei öffnen -> „`Flachdachhauben.project`“
6. Wählen Sie im TaskRunner oder Task Explorer die Task --> `GSM2XML (all Files)`; der erste Task-Prozess startet und konvertiert die GSMs zu XML-Dateien und legt im Source-Ordner die notwendigen Einzelscripte an, je Objekt in einem eigenen Unterordner.

Trigger Task on Save

Mit der Extension „Trigger Task on Save“ können Sie das Konvertieren von XML zu GSM automatisch anstoßen lassen, jedesmal, wenn Sie eine der `.gdl`-Dateien im Ordner „`02_source`“ speichern. Damit ersparen Sie sich den Click auf den Taskrunner.

Nach der Installation von „Trigger Task on Save“ gehen Sie auf „Code >> Einstellungen >> Einstellungen“, wählen dort „Erweiterungen >> Trigger Task on Save“, scrollen dann etwas herunter und klicken auf in „`settings.json`“ bearbeiten.

Dort fügen Sie bitte folgendes ein:

```
"triggerTaskOnSave.tasks": {
  "XML2GSM (all files)": [
    „02_source/**/*.*.gdl“
  ]
},
```

und speichern die Datei, welche dann am Ende wie folgt aussieht:

```

28 "security.workspace.trust.untrustedFiles": "open",
29 "editor.accessibilitySupport": "on",
30 "liveServer.settings.donotVerifyTags": true,
31 "open-php-html-js-in-browser.selectedBrowser": "Chrome",
32 {
33   "[python]": {
34     "editor.formatOnType": true
35   },
36   "triggerTaskOnSave.tasks": {
37     "XML2GSM (all files)": [
38       "source/**/*.gdl"
39     ]
40   },
41   "VsCodeTaskButtons.tasks": [
42     "taskExplorer.includeAnt": ""
43   ]

```

Benutzung

Das erste Projekt ist nun fertig zur Nutzung.

Starten Sie ArchiCAD. Laden Sie den Ordner 1_gsm/Flachdachhauben zu Ihrer geladenen Bibliothek hinzu.

Erzeugen in ArchiCAD neben der bereits oben unter Punkt 4 erzeugten Datei eine oder mehrere GSMs jeweils mit einer eigenen GUID und legen dort die Parameter an. Sichern Sie diese GSM-Datei in den Ordner 1_gsm/Flachdachhauben bzw. in den Makros-Ordner (wenn es sich um Makros handelt).

Wechseln Sie zu VSC und wählen dort im TaskRunner —> GSM2XML aus.

Jetzt sollten alle in ArchiCAD erzeugten GSMs in VSC mit allen Scripten und der Parameterliste verfügbar sein, je Datei in einem eigenen Unterordner des Source-Ordners. Sie können jetzt in VSC sämtliche Scripte editieren, theoretisch auch Parameter manuell ergänzen (was aber in ArchiCAD einfacher ist; Parameter von einer in eine andere Datei übertragen, geht natürlich in den Parameters.xml deutlich schneller).

Wechsel zu ArchiCAD

Sie haben einige Scripte in VSC editiert und möchten nun die Scripte in ArchiCAD prüfen lassen, das 3D-Modell testen oder das User-Inteface überprüfen.

Speichern Sie zunächst alle geänderten .gdl-Dateien. Falls Sie „Trigger Task on Save“ installiert und konfiguriert haben, müsste jetzt der Prozess „XML2GSM“ automatisch angestoßen werden.

Falls nicht, wählen Sie unter Taskrunner den Befehl >> XML2GSM (all Files). Hierdurch schreibt der Task-Prozess sämtliche Einzeldateien wieder in eine einzige XML-Datei und startet den XML-Converter, welcher die XML-Datei in eine GSM-Datei konvertiert, wobei

von der alten GSM-Datei ein Backup mit Zeitstempel im Namen angelegt wird.

In ArchiCAD klicken Sie jetzt auf Ihren Shortcut für „Bibliotheken neu laden“ und sämtliche Änderungen, die Sie in VSC vorgenommen haben, sind in ArchiCAD verfügbar.

Erneuter Wechsel zu VSC

Sie haben einige neue Parameter in ArchiCAD angelegt oder Scriptänderungen vorgenommen, die in der 3D-Vorschau häufiger kontrolliert werden mussten. Jetzt möchten Sie weiter in VSC arbeiten.

W I C H T I G : Sichern Sie jetzt unbedingt Ihre GSM-Datei in ArchiCAD: ungesicherte GSM-Dateien werden vom Backup des Task-Systems nicht erfasst.

Wählen Sie nun TaskRunner —> GSM2XML aus. Hierdurch startet der Task-Prozess den XML-Converter, welcher die GSM-Datei(en) in eine (oder mehrere) XML-Datei(en) konvertiert, wobei von der alten XML-Datei ein Backup mit Zeitstempel im Namen angelegt wird.

In VSC können Sie dann sofort weiterarbeiten.

Einbinden von Grafiken

- Zuerst ein Vorschaubild in ArchiCAD manuell in die GSMs einbauen
- Dann GSM2XML (all Files)
- Jetzt befindet sich das Vorschaubild in einem Unterordner von „bitmaps“ (bzw. die Vorschaubilder)
 - In den selben Ordner packen Sie alle Bilder, die außerdem ins Objekt bzw. in die Objekte sollen.
 - Wenn Sie jetzt das Picture-Task (Update Picture.xml) anstoßen, werden die Bitmap-Scripte ins GDLPict.xml geschrieben
 - und beim nächsten XML2GSM werden die Bitmaps eingebunden.

Erzeugen einer Parameterliste als CSV

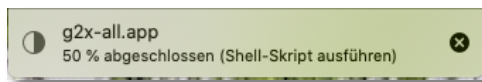
- Verwenden Sie das Task „Parameter.csv“. Die Ausgabe landet im Ordner documentation. (Open Office stellt die CSV bei mir besser dar als Excel).

Automatisches Konvertierung der GSMs

Es ist auch möglich, die GSMs automatisch in XMLs konvertieren zu lassen, in dem Moment, in welchem sie gespeichert werden.

Gehen Sie hierzu wie folgt vor:

- Speichern Sie die Datei „entr-install.sh“ aus dem Unterordner „21_auto“ im Downloads-Ordner
- Terminal öffnen
- dort das hier ausführen: `bash ~/Downloads/entr-install.sh`
- Legen Sie die Datei „g2x-all.app“ auf Ihrem Desktop ab oder eine Verknüpfung davon. Sie können die Datei in Automator öffnen und editieren.
- Doppelklick auf die Datei oder deren Verknüpfung startet einen Prozess, welcher in der Taskleiste angezeigt wird.



- Es öffnet sich ein Dialogfeld, in welchem Sie den Hauptprojektordner Ihrer GDL-Objekte angeben müssen (in unserem Fall „gdl-nucleus-for-vsc“ bzw. dessen Umbenennung)
- Solange es dort einen Ordner „01_gsm“ gibt, in dessen Unterordner sich gsm-Dateien befinden, wird der Prozess GDL2XML in der „gdlconverter.py“ angestoßen.

System Voraussetzungen

Das vorliegende Tool funktioniert in MAC OS X, auf welchem Python 3 installiert ist. Sie benötigen außerdem eine Version von ArchiCAD, welche das Speichern von Dateien erlaubt.

Fragen und Anregungen

Fragen und Anregungen sind überaus willkommen, da ich gerne erfahren würde, wie die Anwender das Tool nutzen, welche Nachteile, Einschränkungen, Verbesserungswünsche etc. vorliegen.

Ausschlüsse und Lizenzen

Diese Software wird geliefert "wie sie ist". Das Tool wurde umfassend getestet, jedoch übernimmt der Programmierer keine Gewährleistung für möglichen Datenverlust, Datenbeschädigung, oder Hardwarebeschädigung und sonstiger Schäden einschließlich Schäden aus entgangenem Gewinn, Betriebsunterbrechung, Verlust von geschäftlichen Informationen oder von Daten oder aus anderem finanziellen Verlust.

Dieses Objekt ist keine Freeware. Sämtliche Urheberrechte liegen beim Auto Joachim Sühlo.

Seien Sie bitte so fair und geben es nicht an Dritte weiter. Mit einer erworbenen symbolischen Lizenz können Sie aber an beliebig vielen Rechnern in Ihrem Büro oder zu Hause arbeiten.