

Scripting interface for Plaxis 3D

Background

Plaxis 3D is an application for creating, calculating and evaluating 3D models of soil and structures, such as tunnels, dikes and skyscraper foundations. The model creation is essentially command-driven, but there is also a rich, interactive GUI that generates and executes commands based on the user's manipulation of the model.

Problem definition

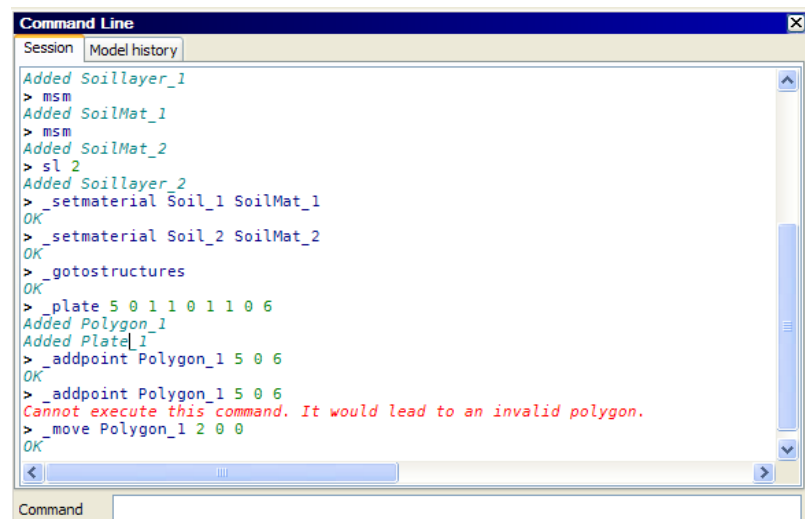
The built-in command language is simple and not Turing-complete, which limits its usefulness for tasks like macros and parametrized model creation wizards. Advanced users would benefit from the ability to use a full-featured, dynamically typed programming language when dealing with Plaxis 3D. Rather than extending the built-in custom language, Plaxis would prefer to utilize an existing language.

Project goals

This project entails the choosing a target environment (e.g. Python, .Net or Active Scripting) and the implementation of a layer that makes the internal object system and commands available for use with that external environment. The implementation should be flexible, in that new commands or objects available in Plaxis 3D are automatically available or can be made available with little effort to the bindings. There must be unit tests available that demonstrate the proper functioning of the interface. Additionally, a small library of commonly used functions for e.g. user input can be created for use by the external programming language.

How to apply

Plaxis bv • Attn: Andrei Chesaru • Delftechpark 53 • P.O. Box 572 • 2600 AN Delft • The Netherlands
Tel +31 (0)15 2517 720 • Fax +31 (0)15 2573 107 • E-mail a.chesaru@plaxis.nl • www.plaxis.nl



```
Command Line
Session  Model history
Added Soillayer_1
> msm
Added SoilMat_1
> msm
Added SoilMat_2
> sl 2
Added Soillayer_2
> _setmaterial Soil_1 SoilMat_1
OK
> _setmaterial Soil_2 SoilMat_2
OK
> _gotostructures
OK
> _plate 5 0 1 1 0 1 1 0 6
Added Polygon_1
Added Plate_1
> _addpoint Polygon_1 5 0 6
OK
> _addpoint Polygon_1 5 0 6
Cannot execute this command. It would lead to an invalid polygon.
> _move Polygon_1 2 0 0
OK
Command
```