

3D-Modellierung

Florian Buchegger, Michael Haberleitner

11. Dez. 2015

Programm

- **Einführung in das VTK-Datenformat**
- Kleinere Objekte direkt im VTK-Format erzeugen
- Einführung in Python
- Einfache Python-Programme schreiben
- Kompliziertere Objekte mit Python erzeugen
- 3D-Objekte animieren

Programm

- Einführung in das VTK-Datenformat
- Kleinere Objekte direkt im VTK-Format erzeugen
- Einführung in Python
- Einfache Python-Programme schreiben
- Kompliziertere Objekte mit Python erzeugen
- 3D-Objekte animieren

Programm

- Einführung in das VTK-Datenformat
- Kleinere Objekte direkt im VTK-Format erzeugen
- Einführung in Python
- Einfache Python-Programme schreiben
- Kompliziertere Objekte mit Python erzeugen
- 3D-Objekte animieren

Programm

- Einführung in das VTK-Datenformat
- Kleinere Objekte direkt im VTK-Format erzeugen
- Einführung in Python
- Einfache Python-Programme schreiben
- Kompliziertere Objekte mit Python erzeugen
- 3D-Objekte animieren

Programm

- Einführung in das VTK-Datenformat
- Kleinere Objekte direkt im VTK-Format erzeugen
- Einführung in Python
- Einfache Python-Programme schreiben
- Kompliziertere Objekte mit Python erzeugen
- 3D-Objekte animieren

Programm

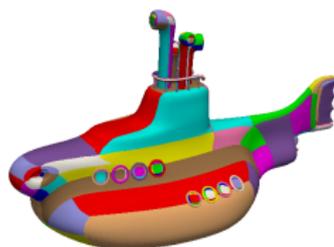
- Einführung in das VTK-Datenformat
- Kleinere Objekte direkt im VTK-Format erzeugen
- Einführung in Python
- Einfache Python-Programme schreiben
- Kompliziertere Objekte mit Python erzeugen
- 3D-Objekte animieren

Das VTK-Datenformat



Einleitung

- Konstrukteure erzeugen **aufwändige Geometrien**
- Wir benötigen **kompakte Darstellung** dieser Objekte
- Zahlreiche Datenformate
- VTK ist **einfach und effektiv**



Datei-Header

Die ersten drei Zeilen jeder Datei sind:

- Header # vtk DataFile Version 2.0
- Titel ein beliebiger Name (max 256 Zeichen)
- Datentyp ASCII oder BINARY

Beispiel:

```
#vtk DataFile Version 2.0  
Meine coolen Daten  
ASCII
```

Datei-Body

Danach kommen Blöcke von Daten, beginnend mit Codewörtern:

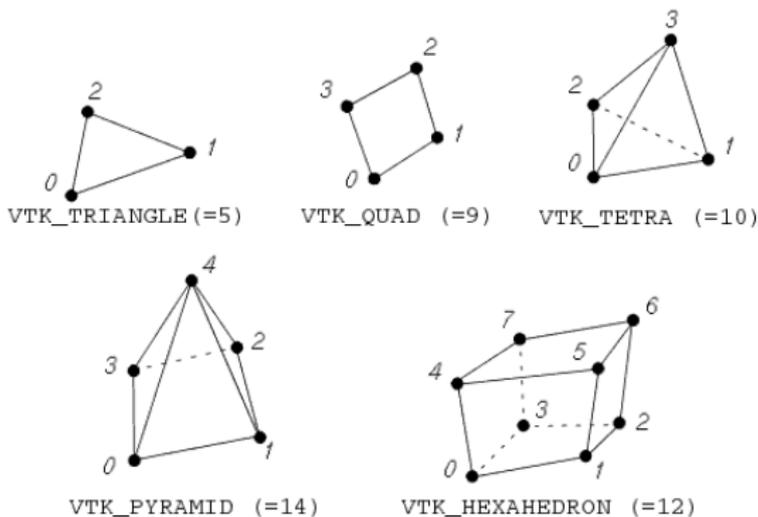
- `DATASET type` beschreibt den nun kommenden Geometrie-Typ

Es gibt verschiedene Geometrie-Typen. Wir verwenden nur `UNSTRUCTURED_GRID`.

- `POINTS n dataType` nun kommen *n* Zeilen mit Punkt-Koordinaten
- `CELLS m1 m2` nun kommen *m1* Zeilen mit Objekten und insgesamt *m2* Werten
- `CELL_TYPES m1` nun kommen *m1* Zeilen, welche die Objekt-Typen beschreiben

Typen von Objekten

- Es gibt viele verschiedene Objekt-Typen
- Wir verwenden heute nur 5

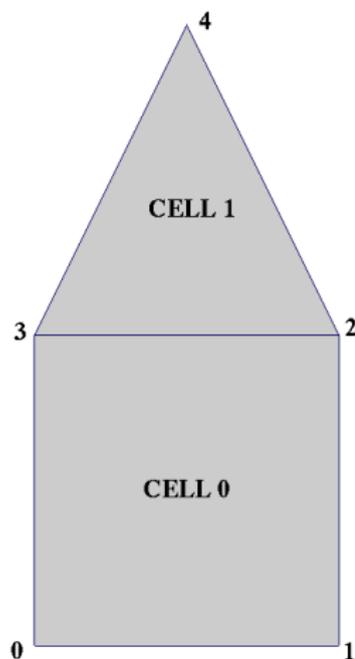


Beispiel Haus

```
# vtk DataFile Version 2.0
Voll cooles Haus
ASCII
DATASET UNSTRUCTURED_GRID
POINTS 5 float
0 0 0
1 0 0
1 1 0
0 1 0
0.5 2 0

CELLS 2 9
4 0 1 2 3
3 2 3 4

CELL_TYPES 2
9
5
```



Python



Python

Was ist Python?

- universelle Programmiersprache
- 1991 erschienen
- oft als Skriptsprache genutzt
- einfach zu erlernen



Python

Wir arbeiten mit der Programmierumgebung 'Idle' für Python.

Figure: Die Python Programmierumgebung 'Idle'

Python

'Idle' verfügt über:

- eine integrierte Shell
- Syntax-highlighting
- und vieles mehr...



```
Python 2.7.6 Shell
File Edit Shell Debug Options Windows Help
Python 2.7.6 (default, Jun 22 2015, 17:58:13)
IOCC 4.8.21 on linux2
Type "copyright", "credits" or "license()" for more information.
>>>
```

Python

Um Python näher kennenzulernen starten wir 'Idle'.

Python

Um Python näher kennenzulernen starten wir 'Idle'.

Beispiel: Punkte auf Gerade:

Erstelle die Funktion 'PointsOfLine(n,p1,p2)', die auf einer Gerade eine gegebene Anzahl an Punkten in gleichem Abstand erzeugt und sie in einer Liste am Bildschirm ausgibt.

Python - wichtige Funktionen

import math

- `math.cos(x)`
- `math.sin(x)`
- `math.radians(x)`
- `math.degrees(x)`
- `math.pi`

import numpy

- `numpy.array(list)`
- `x.tolist()`

import helpfunctions as hf

- `hf.addTriangle(objects,pointList)`
- `hf.addQuad(objects,pointList)`
- `hf.addTetraeder(objects,pointList)`
- `hf.addHexaeder(objects,pointList)`
- `hf.addPyramid(objects,pointList)`
- `hf.printToFile(filename,objects)`

```
a=[]  
hf.addTriangle(a, [0,0,1,1,2,1])  
hf.printToFile('myFile',a)
```