

Programmierung von CAX-Systemen

Übung 2: B-Rep Topologie

David Straub

Lernziele

Nach dieser Übung können Sie:

- Die topologische Struktur eines B-Rep-Körpers inspizieren und beschreiben
- Sub-Shapes gezielt nach Position, Achsausrichtung und Geometrietyp selektieren
- Operationen (Fillet, Chamfer, Aufbauen) auf topologisch selektierte Elemente anwenden
- Robuste Selektionskriterien formulieren, die nach Modifikationen stabil bleiben

Startcode

```
import build123d as bd
from build123d import GeomType

# Unser Ausgangskörper: Montageplatte mit vier Bohrungen
platte = bd.Box(80, 60, 8)

for x, y in [(-27, -20), (27, -20), (-27, 20), (27, 20)]:
    platte = platte - bd.Pos(x, y) * bd.Cylinder(radius=4, height=8)

platte
```

Aufgabe 1: Topologie erkunden

1.1 – Strukturübersicht

Untersuchen Sie die Topologie der Montageplatte:

1. Rufen Sie `show_topology()` auf – was sehen Sie?
2. Zählen Sie die Elemente: Faces, Edges, Vertices, ...
3. Wie viele Flächen haben Sie erwartet? Was trägt jede Bohrung zur Topologie bei?

Hinweise: `.show_topology()`, `.faces()`, `.edges()`, `.vertices()`, `len()`

1.2 – Geometrietypen

Geben Sie für jede Fläche den Geometrietyp und die Fläche aus. Dasselbe für jede Kante mit Länge.

1. Welche Geometrietypen kommen bei Flächen vor? Und bei Kanten?
2. Wie viele Kanten sind Kreise? Passt das zur Anzahl der Bohrungen?

Hinweise: `.geom_type`, `.area`, `.length`, `for f in platte.faces():`

Aufgabe 2: Gezielte Selektion

2.1 – Flächen selektieren

1. Selektieren Sie die **Oberseite** der Platte. Geben Sie Typ und Flächeninhalt aus.
2. Selektieren Sie alle **zylindrischen Flächen** (Bohrungswände). Wie viele sind es?

Hinweise: `.faces()`, `.sort_by(Axis.Z)`, `.last`, `.filter_by(GeomType.CYLINDER)`, `.geom_type`, `.area`

2.2 – Kanten selektieren und verrunden

1. Selektieren Sie alle **kreisförmigen Kanten** und verrunden Sie sie mit `radius=1`.
2. Verrunden Sie nur die **oberen** Kreiskanten (Bohrungseintritt oben). Was ist der visuelle Unterschied?

Hinweise: `.edges()`, `.filter_by(GeomType.CIRCLE)`, `.filter_by_position(Axis.Z, min, max)`, `bd.fillet(..., radius=...)`

2.3 – Selektion nach Flächengröße

Selektieren Sie die **kleinste** und die **größte** Fläche der Platte. Geben Sie jeweils Typ und Flächeninhalt aus.

Welche Flächen sind das geometrisch?

Hinweise: `.sort_by(SortBy.AREA)`, `.first`, `.last`, `.area`, `.geom_type`

Aufgabe 3: Modell erweitern

3.1 – Zapfen auf der Oberseite

Bauen Sie einen Zylinder (`radius=12`, `height=15`) mittig auf der Oberseite der Platte auf.

Frage: Warum `bd.Plane(oberseite)` statt `bd.Plane.XY.offset(8)`? Was wäre der Unterschied in einem komplexeren Modell?

Hinweise: `.faces().sort_by(Axis.Z).last`, `bd.Plane(face)`, `ebene * zapfen`, `basis + ...`

3.2 – Kanten des Zapfens verrunden

1. Verrunden Sie die **Oberkante** des Zapfens mit `radius=2`.
2. Verrunden Sie zusätzlich die **Übergangskante** zwischen Platte und Zapfen.

Hinweise: `.edges().filter_by(GeomType.CIRCLE)`, `.sort_by(Axis.Z)`, `.last`, List-Slicing `[-2:]`, `bd.fillet()`

Zusatzaufgabe 1: Flansch mit Lochkreis

Flansch konstruieren

Konstruieren Sie einen Flansch (Ringscheibe mit Bohrungen auf einem Lochkreis):

```
import math

# Ringscheibe: großer Zylinder minus kleiner Zylinder
flansch = bd.Cylinder(radius=40, height=12) - bd.Cylinder(radius=18, height=12)

# Lochkreis: 6 Bohrungen im Abstand von 60°
for winkel in range(0, 360, 60):
    x = 30 * math.cos(math.radians(winkel))
    y = 30 * math.sin(math.radians(winkel))
    flansch = flansch - bd.Pos(x, y) * bd.Cylinder(radius=4, height=12)

flansch
```

Flansch analysieren und bearbeiten

1. Wie viele Flächen hat der Flansch? Wie viele davon sind zylindrisch?
2. Verrunden Sie alle **oberen Kreiskanten** (Bohrungen + Innen-/Außenring) mit `radius=1`.
3. Fügen Sie eine **Fase** an der Unterkante des Innenrings hinzu.

Hinweise: `.filter_by(GeomType.CYLINDER)`, `.filter_by(GeomType.CIRCLE)`,
`.filter_by_position(Axis.Z, ...)`, `bd.fillet()`, `bd.chamfer()`

Zusatzaufgabe 2: Topologie eines fremden Modells

STEP-Import

Laden Sie eine STEP-Datei (z.B. aus FreeCAD, GrabCAD oder dem Kursordner):

```
teil = bd.import_step("mein_teil.step")
```

STEP-Analyse und Bearbeitung

1. Wie viele Faces, Edges, Vertices hat das Teil?
2. Welche Geometrietypen kommen bei den Flächen vor und wie häufig?
3. Führen Sie eine sinnvolle Operation durch (z.B. Kreiskanten verrunden, auf einer Fläche aufbauen).

Hinweise: `.show_topology()`, `.faces()`, `.geom_type`, `collections.Counter`,
`.filter_by(GeomType.CIRCLE)`, `bd.fillet()`