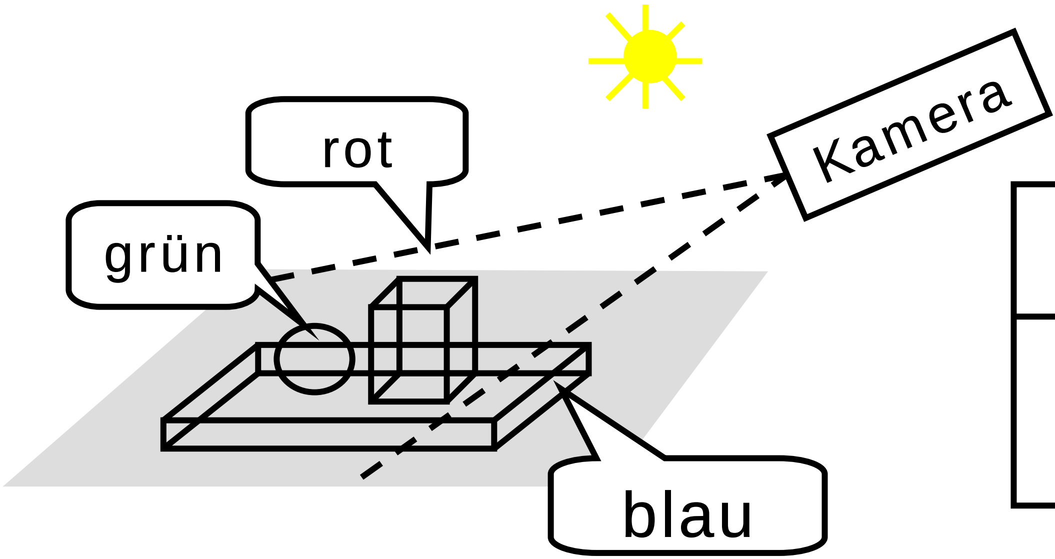
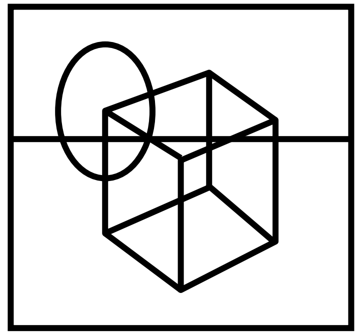
Computergrafik 2016
Oliver Vornberger

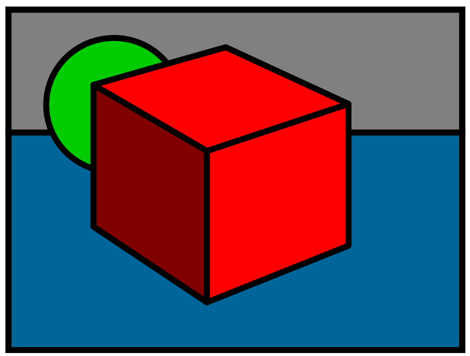
Vorlesung vom 24.05.2016

Kapitel 16:
3D-Repräsentation

Sequenz von Transformationen







- Modeling
- View Orientation
- View Mapping
- Device Mapping

Repräsentation + Darstellung

- Datenstruktur zur Beschreibung der Szene
- Lichtquellen
- Synthetische Kamera

Repäsentation

- das daraus berechnete 2D-Pixelbild

Darstellung

Repräsentationshierarchie

- Elementarobjekt
  (Kugel, Würfel, Kegel, Pyramide,Zylinder)

- Drahtmodell
  (Punkte, Kanten)

- Flächenmodell
  (Punkte, Kanten, Flächen, Normalen)

Darstellungshierarchie

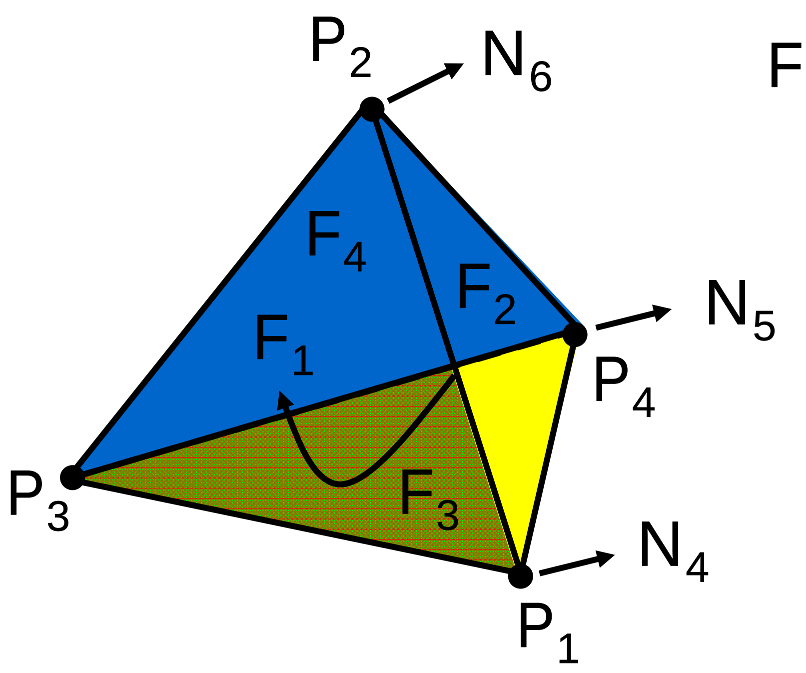
- Punktmodell
- Drahtmodell
- Drahtmodell, verdeckte Kanten entfernt
- Flächenmodell, ohne abgewandte Flächen
- Flächenmodell, verdeckte Flächen entfernt
- Flächenmodell mit Beleuchtung
- Flächenmodell mit Beleuchtung + Schatten
- Flächenmodell mit Spiegelung + Brechung

Datenstruktur für Polyeder

- Punkte
- Kanten
- Flächen
- Normale
- Farbe
- Materialeigenschaften
- Textur
- Bump Map

Beispiel: Tetraeder

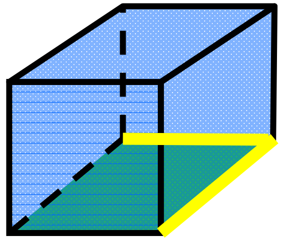
Farbe, Punkte, Kanten, Flächen, Normalen



F_2

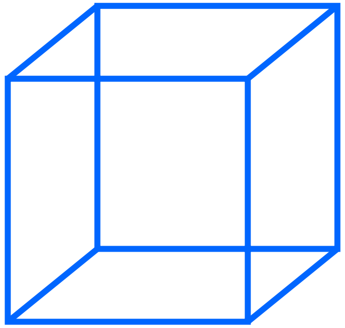
- (255,255,0)
- P_2, P_1, P_4
- N_4, N_5, N_6
- $P_2 \rightarrow P_1 : F_1$
- $P_1 \rightarrow P_4 : F_3$
- $P_4 \rightarrow P_2 : F_4$

Drahtgitterdarstellung



```
für jede Fläche F tue {
  für jede Kante E von F tue {
    falls !bearbeitet(E) {
      falls sichtbar(F)
        zeichne E solide;
      falls !sichtbar(F) {
        falls sichtbar(Nachbarfläche(F))
          zeichne E solide;
        falls !sichtbar(Nachbarfläche(F))
          zeichne E gestrichelt;
      }
      markiere E als bearbeitet;
    }
  }
}
```

Würfel



Kantenlänge 1
Schwerpunkt im Ursprung

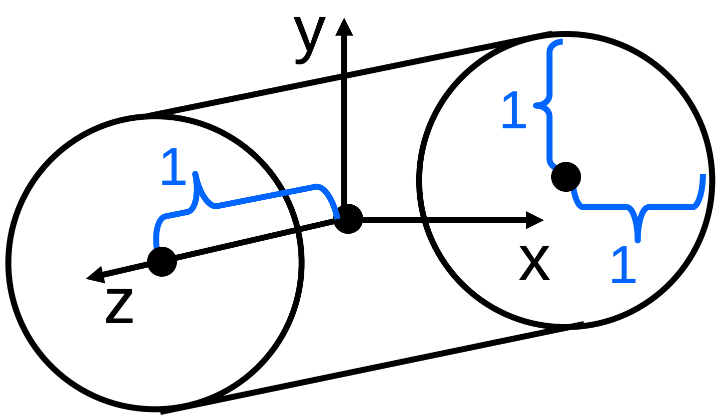
```
(+0.5,  +0.5,  +0.5,  1.0)
(+0.5,  -0.5,  +0.5,  1.0)
(-0.5,  +0.5,  +0.5,  1.0)
(-0.5,  -0.5,  +0.5,  1.0)
(+0.5,  +0.5,  -0.5,  1.0)
(+0.5,  -0.5,  -0.5,  1.0)
(-0.5,  +0.5,  -0.5,  1.0)
(-0.5,  -0.5,  -0.5,  1.0)
```

Zylinder

Schwerpunkt im Ursprung

Zwei Deckflächen mit Radius 1

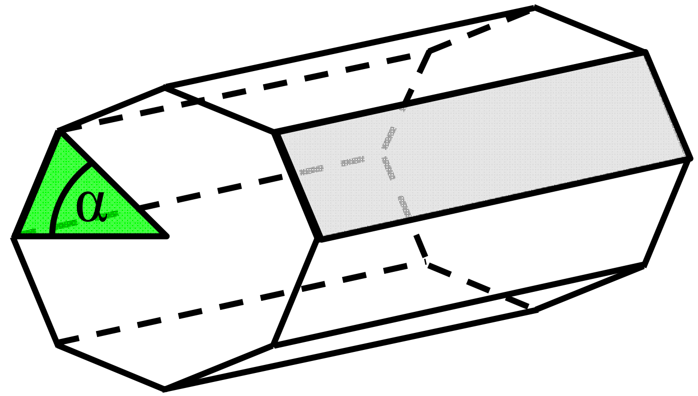
Eine Mantelfläche der Länge 2



Approximation eines Zylinders

$\alpha = 2 \cdot \pi / n$

$\varphi = i \cdot \alpha,\ i = 0, \ldots, n\text{-}1$



Eckpunkte:

$[\cos(\varphi), \quad \sin(\varphi), \quad +1,1]$

$[\cos(\varphi+\alpha), \sin(\varphi+\alpha), +1,1]$

$[\cos(\varphi+\alpha), \sin(\varphi+\alpha), -1,1]$

$[\cos(\varphi), \quad \sin(\varphi), \quad -1,1]$

Normalen:

$[\cos(\varphi), \quad \sin(\varphi), \quad 0,0]$

$[\cos(\varphi+\alpha), \sin(\varphi+\alpha), 0,0]$

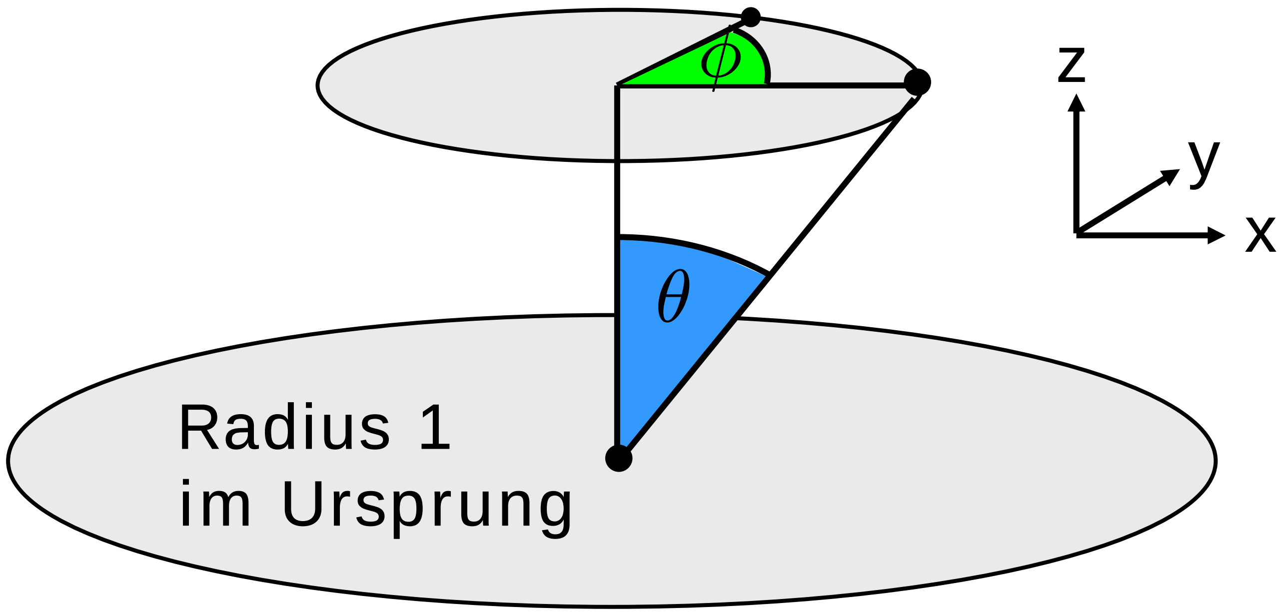
$[\cos(\varphi+\alpha), \sin(\varphi+\alpha), 0,0]$

$[\cos(\varphi), \quad \sin(\varphi), \quad 0,0]$

Kugel

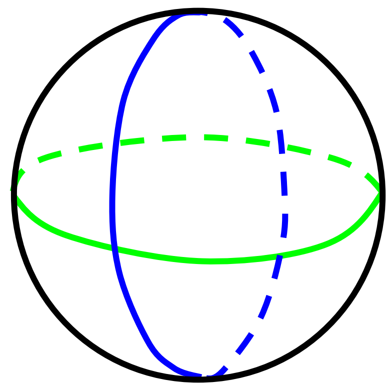
$$\left(\ \sin(\theta)\cdot\cos(\phi),\quad \sin(\theta)\cdot\sin(\phi),\quad \cos(\theta)\ ,1\right)$$

$$0 \leq \phi \leq 2 \cdot \pi \qquad \bullet \qquad 0 \leq \theta \leq \pi$$



Approximation einer Kugel

- wähle n gerade
- bilde n/2 Längenkreise
  (alle gleich groß)
- bilde n/2-1 Breitenkreise
  (verschieden groß)



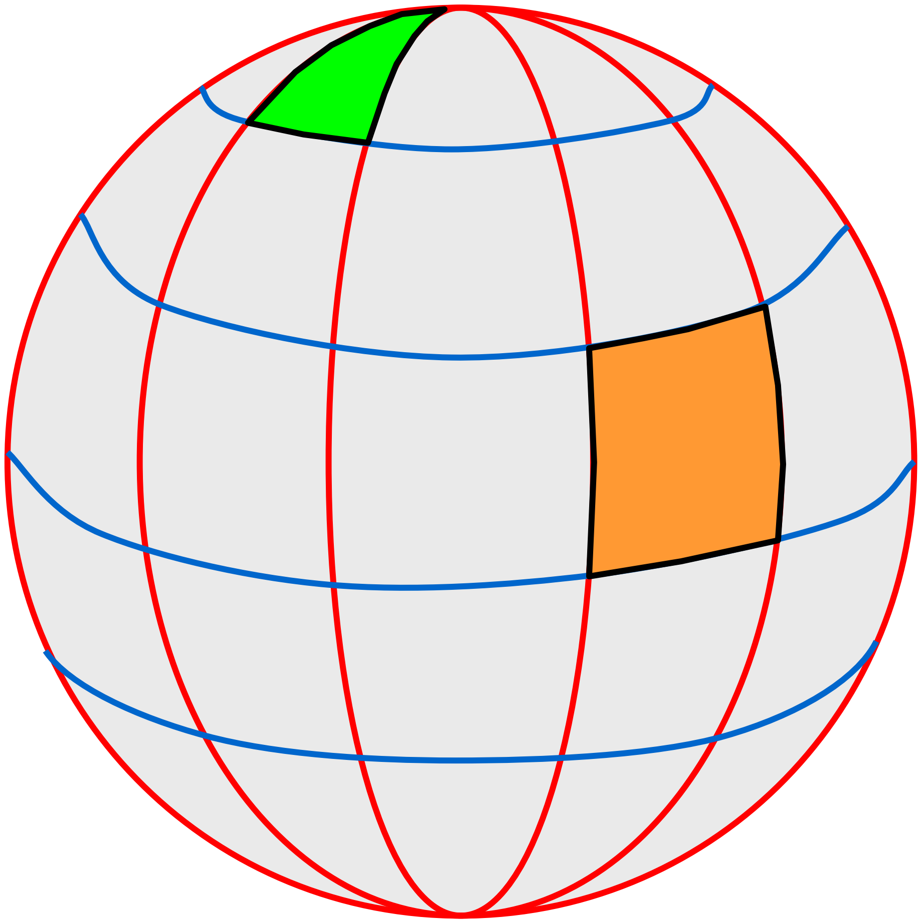
Kugel für n=10

n/2 = 5
Längenkreise

n/2-1 = 4
Breitenkreise

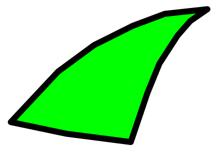
2n=20
Dreiecke

n·(n/2-2)=30
Rechtecke



Dreiecke von Kugel

$$\alpha = (2\pi)/n, \quad n \in \mathbb{N} \text{ gerade}$$



$$\phi = k \cdot \alpha,\ k \in \{0, ..., n-1\}$$

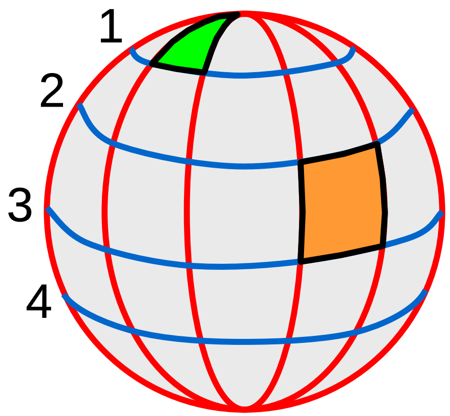
$$(0, 0, +1, 1),$$

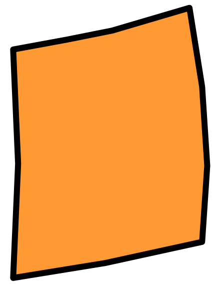
$$(\sin(\alpha)\cos(\phi), \sin(\alpha)\sin(\phi), \cos(\alpha), 1),$$

$$(\sin(\alpha)\cos(\phi+\alpha), \sin(\alpha)\sin(\phi+\alpha), \cos(\alpha), 1)$$

Rechtecke von Kugel

n=10





$$\alpha = (2\pi)/n, \quad n \in \mathbb{N} \text{ gerade}$$

$$\phi = k \cdot \alpha, k \in \mathbb{N}, k < n$$

l=1,2,3

$$\theta = l \cdot \alpha, l \in \mathbb{N}, 0 < l < (n/2 - 1)$$

$$(\sin(\theta)\cos(\phi), \sin(\theta)\sin(\phi), \cos(\theta), 1)$$

$$(\sin(\theta)\cos(\phi + \alpha), \sin(\theta)\sin(\phi + \alpha), \cos(\theta), 1)$$

$$(\sin(\theta + \alpha)\cos(\phi + \alpha), \sin(\theta + \alpha)\sin(\phi + \alpha), \cos(\theta + \alpha), 1)$$

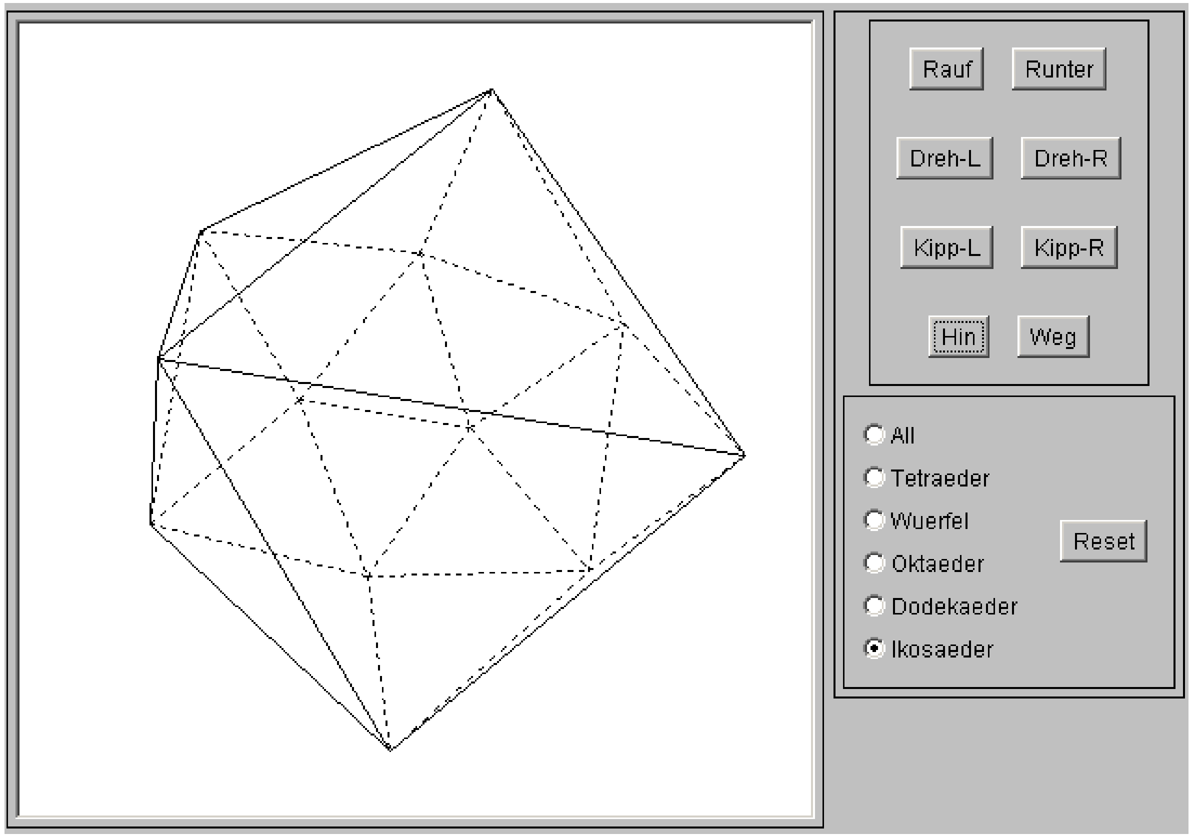
$$(\sin(\theta + \alpha)\cos(\phi), \sin(\theta + \alpha)\sin(\phi), \cos(\theta + \alpha), 1)$$

Normalen von Kugel

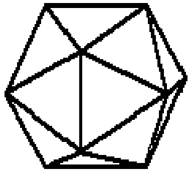
$$\alpha = (2\pi)/n, \quad n \in \mathbb{N} \text{ gerade}$$
$$\phi = k \cdot \alpha, k \in \mathbb{N}, k < n$$
$$\theta = l \cdot \alpha, l \in \mathbb{N}, 0 < l < (n/2 - 1)$$

$$(\sin(\theta)\cos(\phi), \sin(\theta)\sin(\phi), \cos(\theta), 0)$$
$$(\sin(\theta)\cos(\phi + \alpha), \sin(\theta)\sin(\phi + \alpha), \cos(\theta), 0)$$
$$(\sin(\theta + \alpha)\cos(\phi + \alpha), \sin(\theta + \alpha)\sin(\phi + \alpha), \cos(\theta + \alpha), 0)$$
$$(\sin(\theta + \alpha)\cos(\phi), \sin(\theta + \alpha)\sin(\phi), \cos(\theta + \alpha), 0)$$

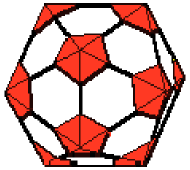
Java-Applet zur Wireframe-Projektion

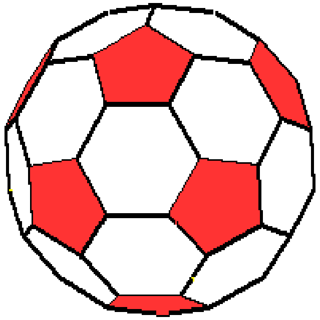


Platonische Körper



4 Dreiecke
6 Vierecke
8 Dreiecke
12 Fünfecke
20 Dreiecke



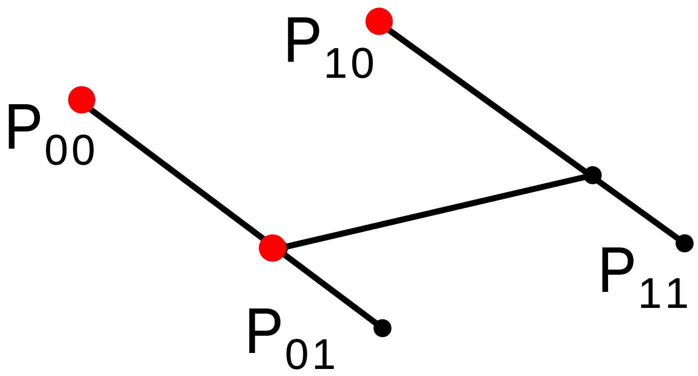


~cg/2016/skript/Applets/3D-wire/App.html

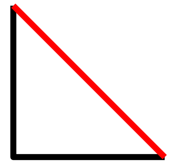
https://de.wikipedia.org/wiki/Platonischer_K%C3%B6rper

http://www.mathematische-basteleien.de/fussball.htm

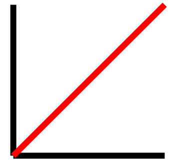
parametrisierte Fläche



$$f_0(t) = 1 - t$$



$$f_1(t) = t$$



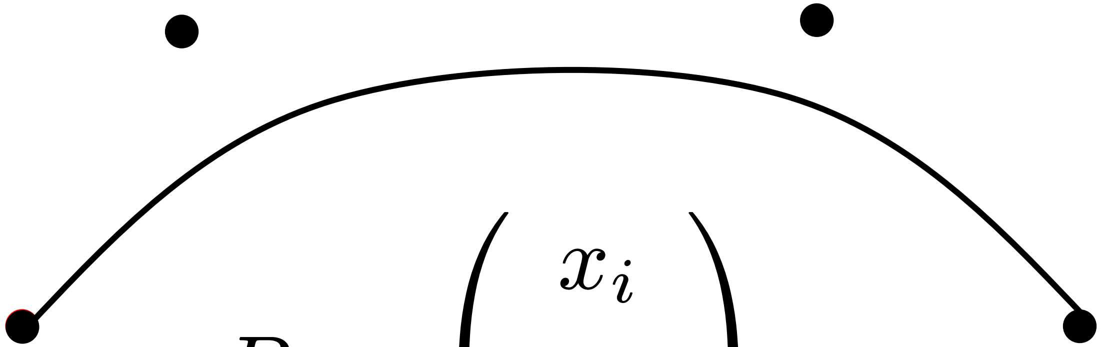
$$(1-u)\cdot[(1-v)\cdot P_{00} + v\cdot P_{01}] + u\cdot[(1-v)\cdot P_{10} + v\cdot P_{11}]$$

$$f_0(u)f_0(v)P_{00} + f_0(u)f_1(v)P_{01}$$
$$+ f_1(u)f_0(v)P_{10} + f_1(u)f_1(v)P_{11}$$

$$P(u,v) = \sum_{i=0}^{1}\sum_{j=0}^{1} f_i(u)\cdot f_j(v)\cdot P_{i,j}$$

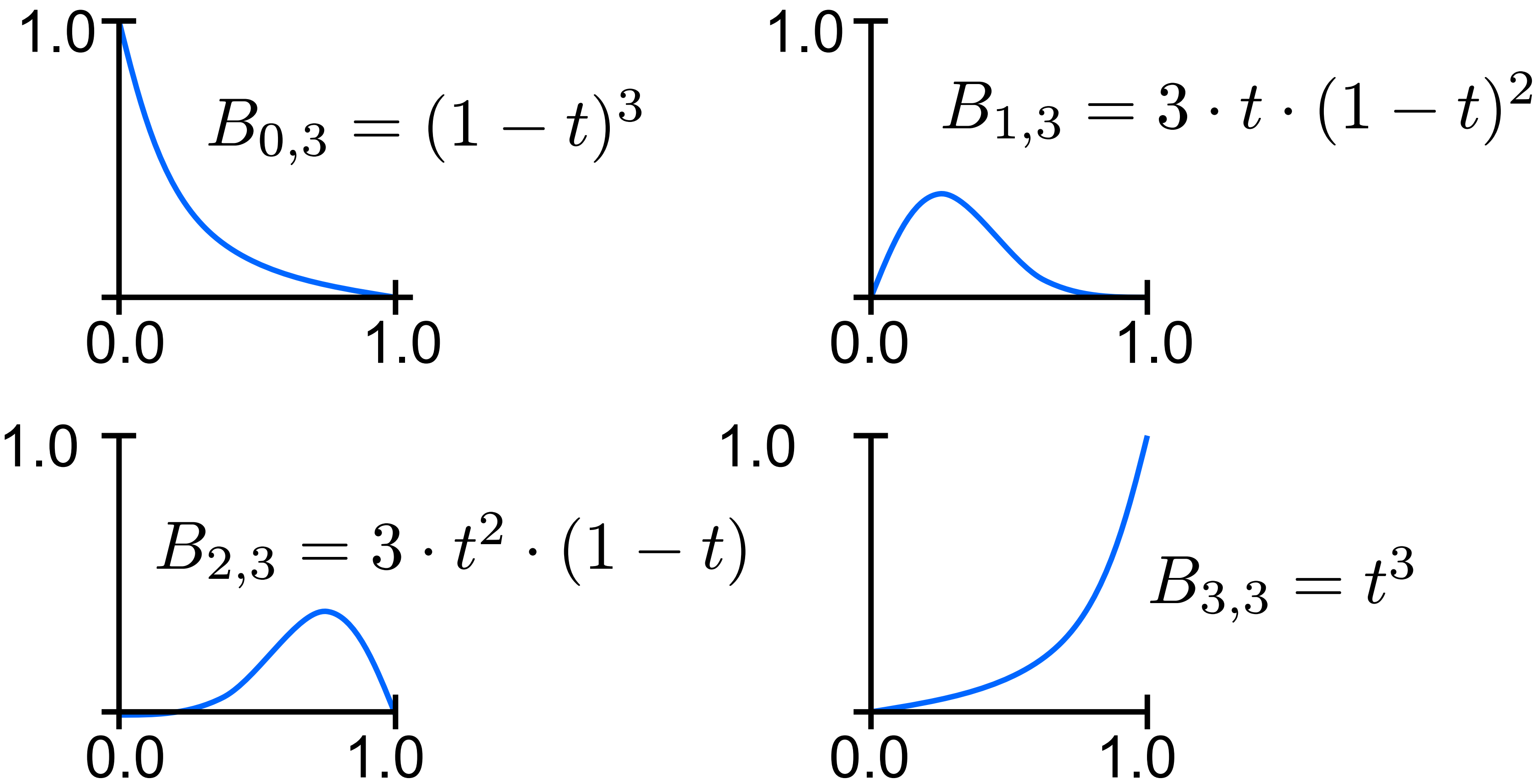
Bezier-Kurve

$$P(t) = \sum_{i=0}^{3} B_{i,3}(t) \cdot P_i$$

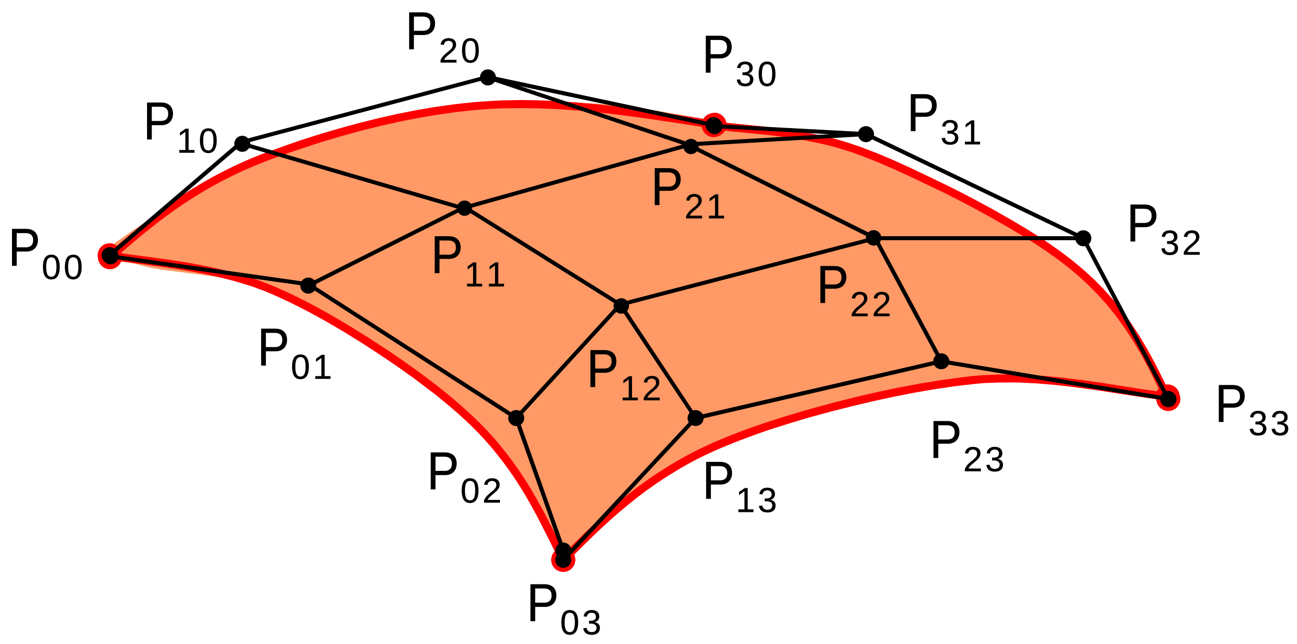


$$P_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}$$

Kubische Bernstein-Polynome



Gekrümmte Fläche

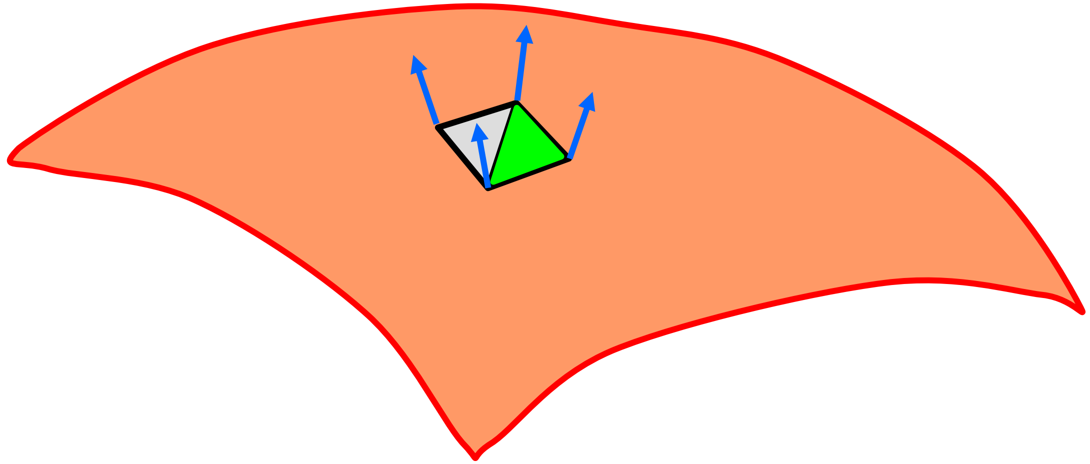


$$P(u,v) = \sum_{i=0}^{3} \sum_{j=0}^{3} \cdot B_{i,3}(u) \cdot B_{j,3}(v) \cdot P_{i,j}$$

Drahtgitterdarstellung

```
for (u=0.0; u<=1.0; u=u+0.1){
  for (v=0.0; v<=1.0; v=v+0.1){
    p = new Punkt();
    for (i=0; i<=3; i++){
      for (j=0; j<=3; j++){
        p = add(p,B_{i,3}(u)·B_{j,3}(v)·P_{i,j});
      }
    }
    // Punkt p verarbeiten
  }
}
```

Flächendarstellung

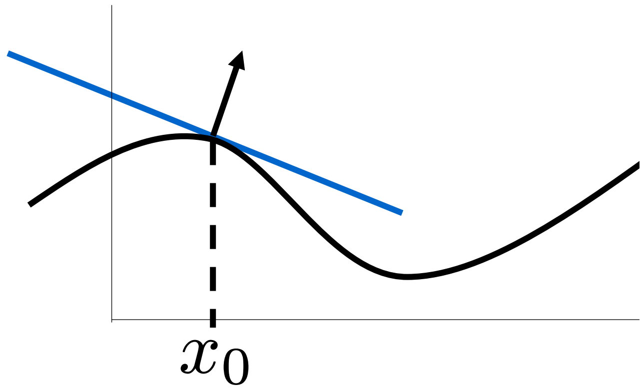


Zerlege Rechteck in 2 Dreiecke

berechne Normalenvektoren

färbe Dreiecke ein

Normalen berechnen



- im Approximationspunkt Bezierkurve nach u ableiten

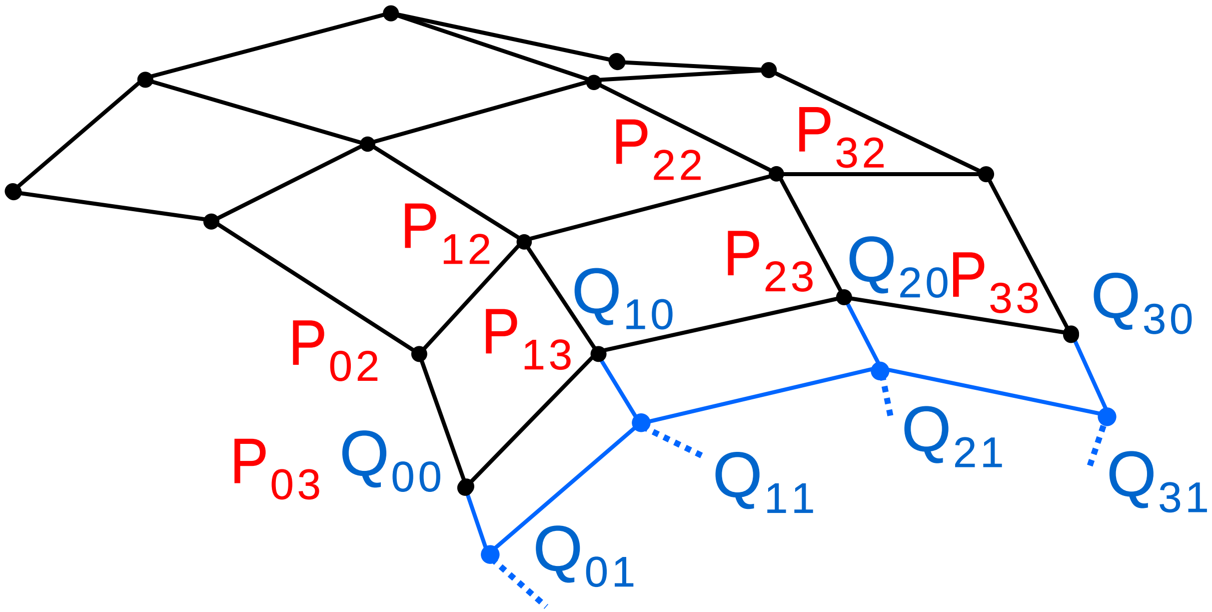
$$\frac{\partial P(u,v)}{\partial u} = \sum_{i=0}^{3} \sum_{j=0}^{3} B'_{i,3}(u) \cdot B_{j,3}(v) \cdot P_{i,j}$$

- im Approximationspunkt Bezierkurve nach v ableiten

$$\frac{\partial P(u,v)}{\partial v} = \sum_{i=0}^{3} \sum_{j=0}^{3} B_{i,3}(u) \cdot B'_{j,3}(v) \cdot P_{i,j}$$

- Kreuzprodukt beider Tangenten berechnen

Bezier-Flächen anstückeln

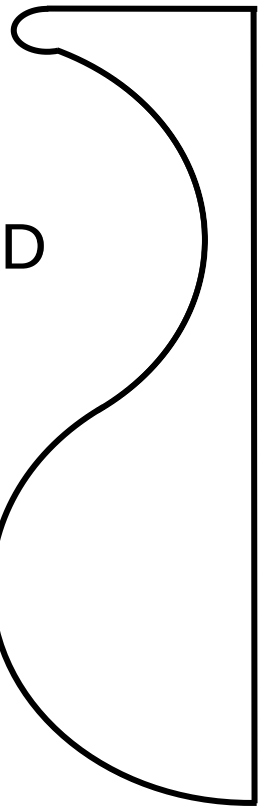


Anschlusspunkte $P_{i2}, P_{i3} = Q_{i0}, Q_{i1}$ collinear

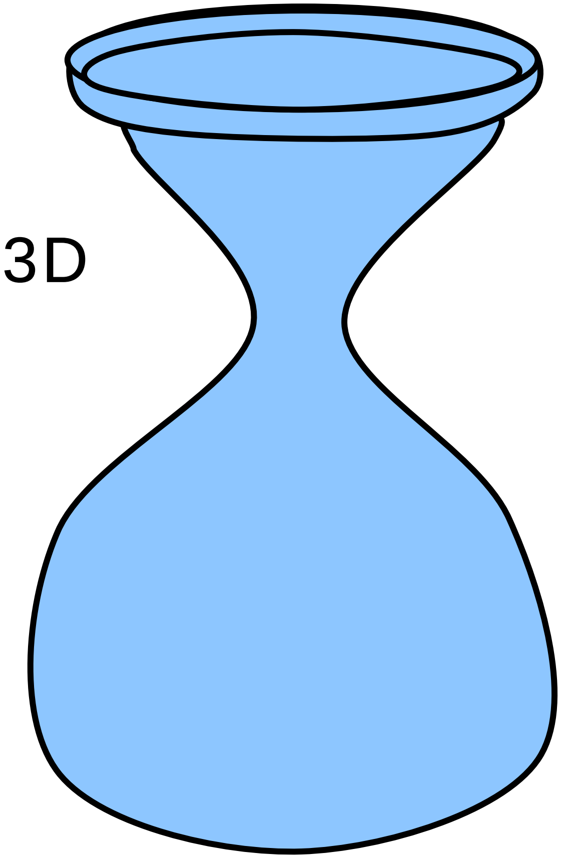
Verhältnis der Abstände $\frac{|P_{i3} - P_{i2}|}{|Q_{i1} - Q_{i0}|}$ konstant

Rotationskörper

2D



3D



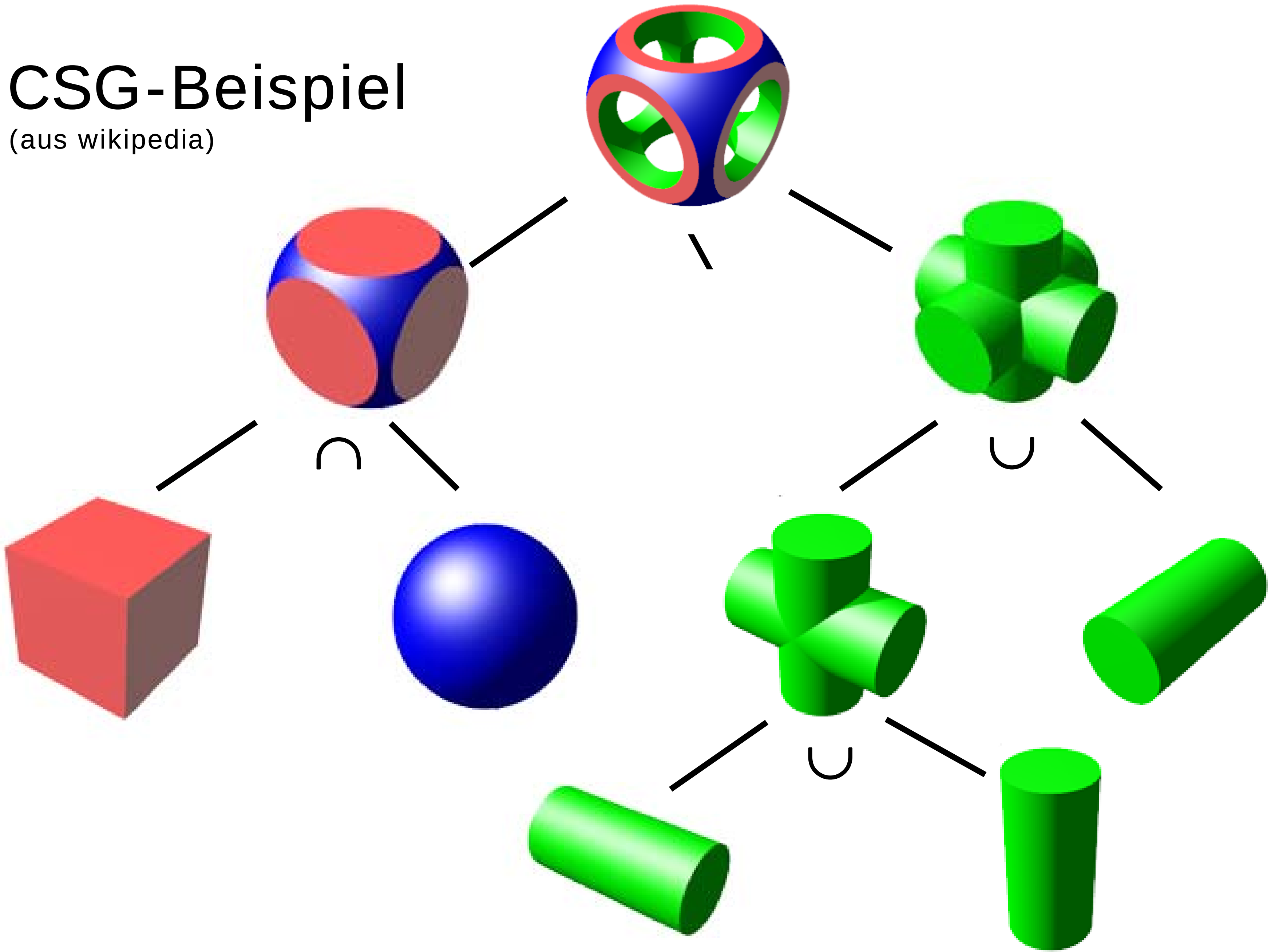
CSG

Constructive Solid Geometry

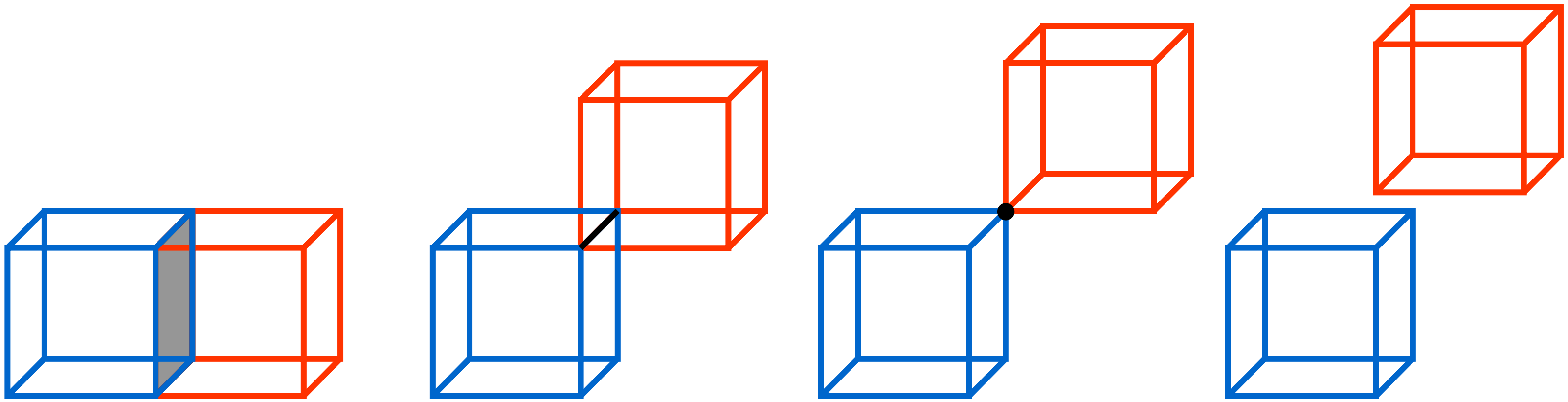
- Objekte sollen physikalisch realisierbar sein
- Objekte haben Volumen, Gewicht
- Objekte lassen sich maschinell herstellen

CSG-Beispiel

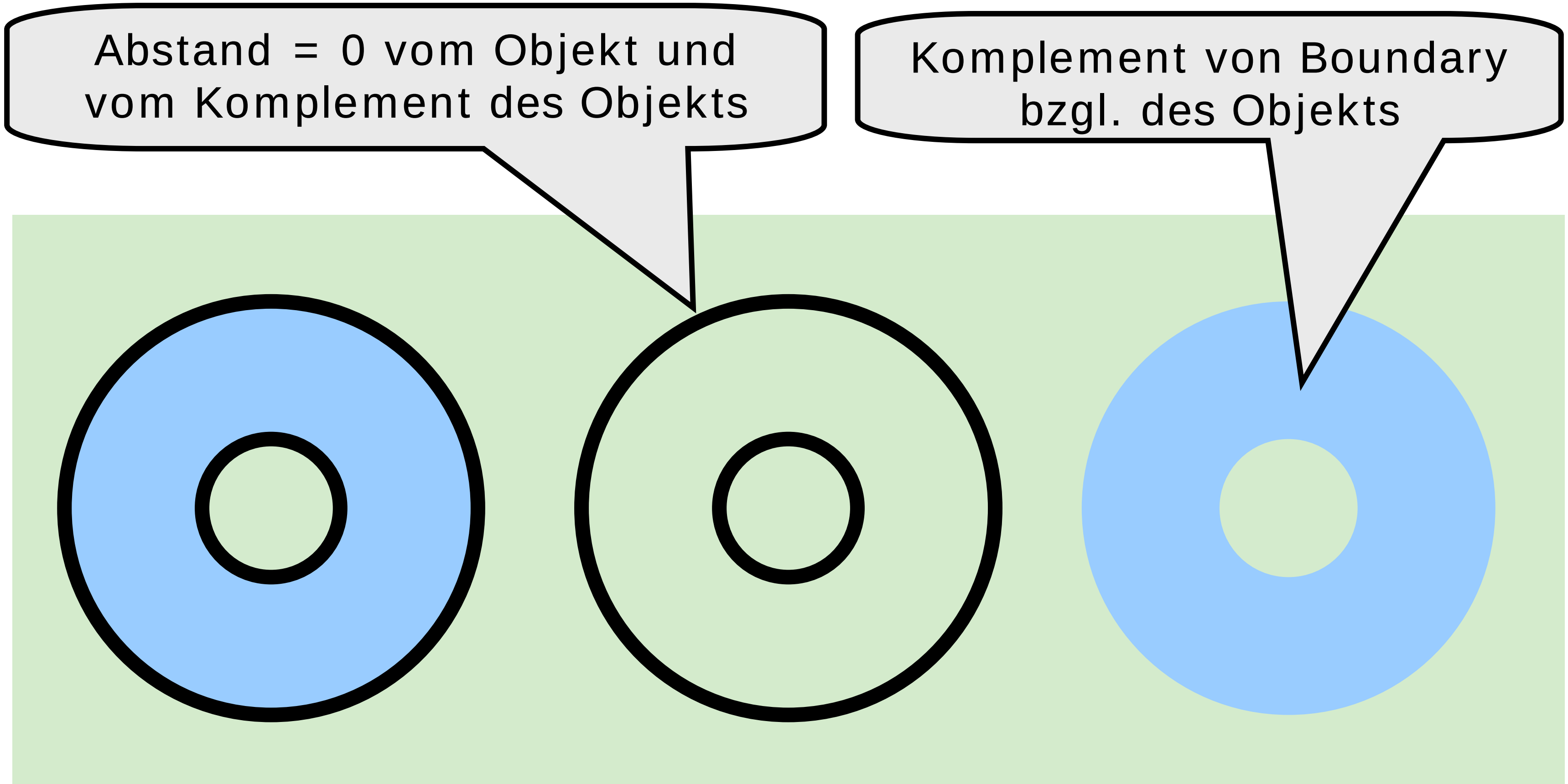
(aus wikipedia)



Schnittmengen



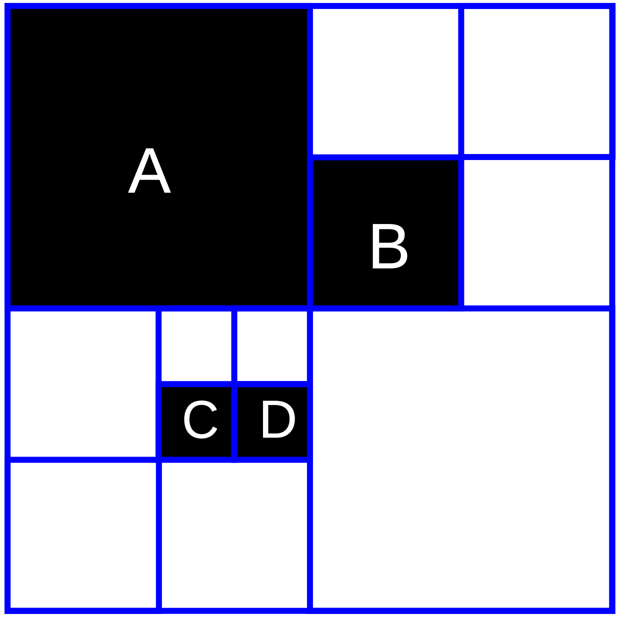
Innen und außen

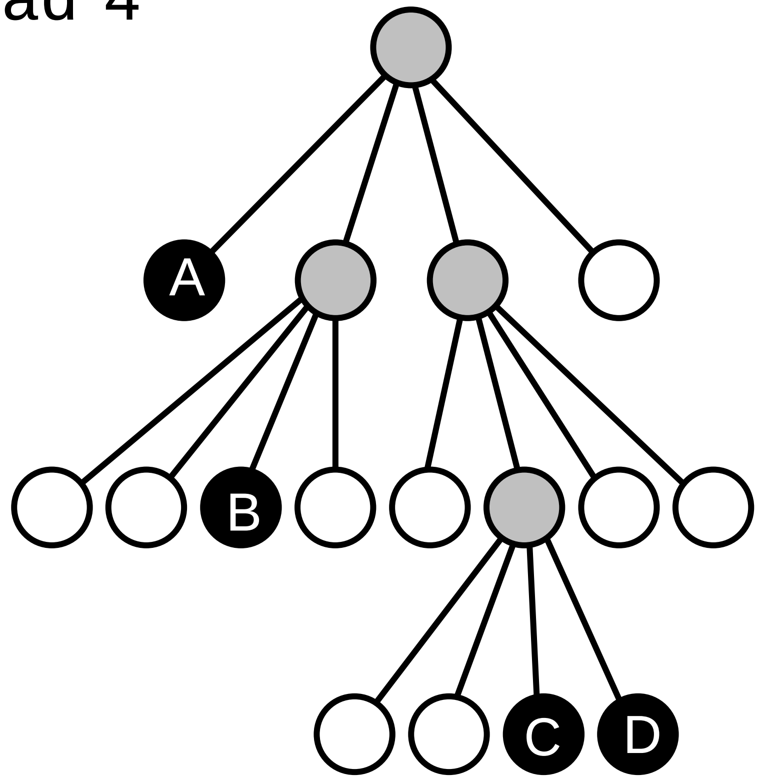


Objekt Boundary Interior

Quadtree

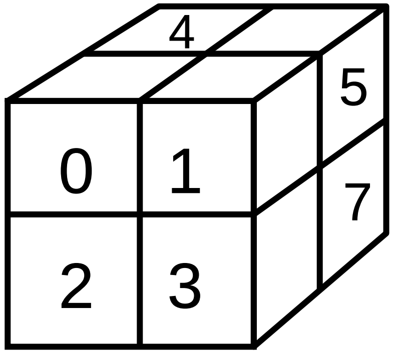
strukturiere 2D-Fläche
durch Baum vom Grad 4



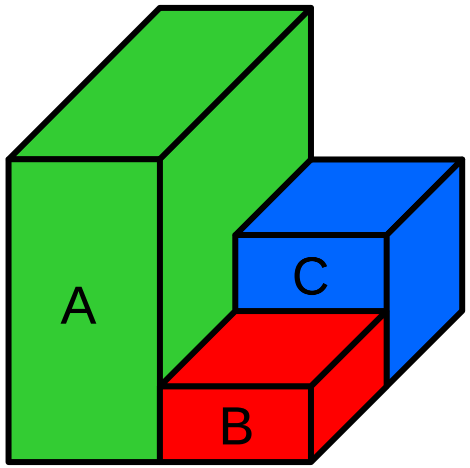


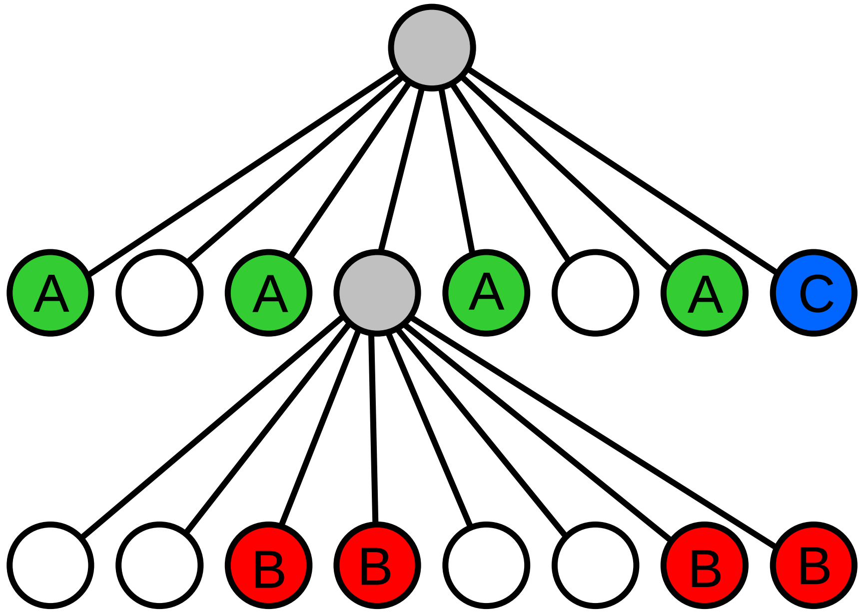
Effizientes ∪ ∩ \ durch Traversieren

Octree



strukturiere 3D-Raum
durch Baum vom Grad 8





Effizientes $\cup$ $\cap$ $\setminus$ durch Traversieren