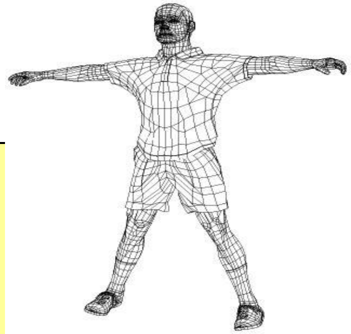
Vorlesung Computergrafik

Modellierung

P.A.Henning
media::lab
Karlsruhe University of Applied Sciences

Modelle

Ein Modell ist eine abstrakte Darstellung eines konkreten Sachverhaltes



- Modelle sind an einen Zweck gebunden
  - Beispiel: Architekturmodelle dienen der Visualisierung
- Modelle sind einfacher als der zugrundeliegende Sachverhalt
  - Beispiel: Modell eines Roboter-Arms ist ein Satz von Koordinaten
- Modelle erlauben je nach Zweckbindung und Vereinfachungsgrad Vorhersagen über das Verhalten des zugrundeliegenden realen Sachverhalts, die daran u.U. nur unter wesentlich größerem Aufwand überprüfbar oder realisierbar wären.
  - Beispiel: Wettermodell
  - Beispiel: Reise durch den menschlichen Körper

Modelle in der Computergrafik

In der grafischen Datenverarbeitung

- abstrahieren Modelle die geometrischen, materiellen und optischen Eigenschaften realer Körper.

- dienen Modelle dem Zweck, dem Betrachter eines Bildes die Illusion zu vermitteln, er oder sie sähe den realen Körper.

- erlauben Modelle die Vorhersage, wie der Betrachter den realen Körper unter
    - veränderten geometrischen Umgebungsparametern (Standpunkt, Orientierung)
    - veränderten materiellen Gegebenheiten (Oberfläche, Farbe, Textur)
    - veränderten optischen Umgebungsparametern (Beleuchtung)

  sähe.

Modelle in der GDV II

- Anwendungsgebiete
  - interaktiver Entwurf, Computer Aided Design
  - Simulation und Visualisierung von Prozessen und Vorgängen
  - Kunst

- Probleme
  - angemessene Beschreibung der realen Körper (Objekte)
  - Generierung der Darstellung aus dem Modell (Detailtreue, Automatisierungsgrad, Geschwindigkeit)
  - Positionierung und Orientierung von Teilmodellen
  - Kollisionen und Durchdringung von Objekten
  - Deformation und Morphing von Objekten

Geometrische Begriffe

- Vektoren
  - Vektorraum, Basis, Koordinatensystem
  - Addition von Vektoren, Skalarprodukt, äußeres Produkt

- Punkte, Geraden, Ebenen
  - definierende Gleichungen
  - definierende geometrische Relationen
  - Abstandsberechnung
  - Projektion
  - Schnittmengen

- Lineare Algebra
  - Lineare Abbildungen
  - Matrixmultiplikation
  - Matrixinversion

Recherchieren Sie dies !

Parameter grafischer Modelle

Geometrie
- Ausdehnung
- Form
- grafische Primitive

Topologie

Relative Anordnung mehrerer Objekte

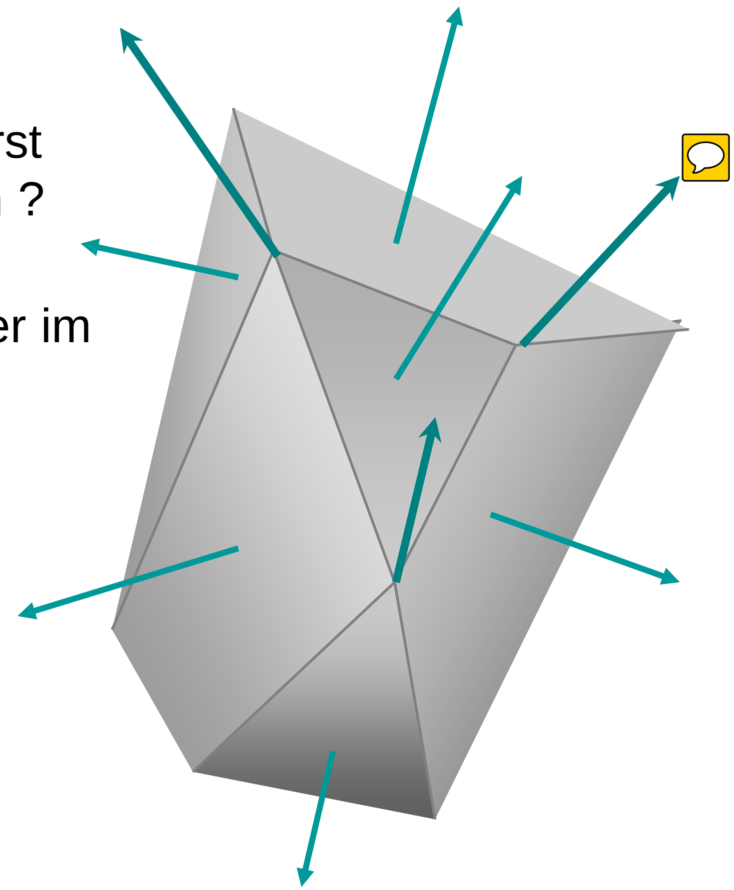
Optik
- Lichtquellen
- Lichtstärken
- Lichtfarben
- Standpunkt

Material
- Transparenz
- Reflektivität
- Emissivität
- Rauhigkeit

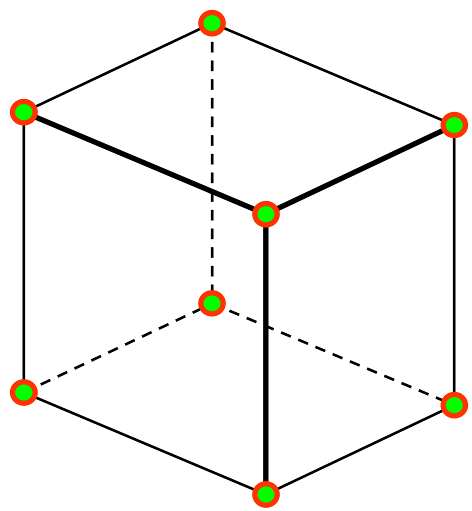
Sekundärdaten
- Masse
- Ladung
- Härte
- Teilenummern
- Beschriftung
- .....

Geschwindigkeit vs. Speicherplatz

- „Time-Space Tradeoff“
- Wieviel Information soll im Modell direkt gespeichert werden, wieviel Information soll erst für die konkrete Darstellung berechnet werden ?
  – Beispiel: Schattierte Darstellung benötigt Normalenvektoren. Diese können entweder im Modell gespeichert werden, oder bei der Darstellung berechnet werden.



Parameter eines Würfels



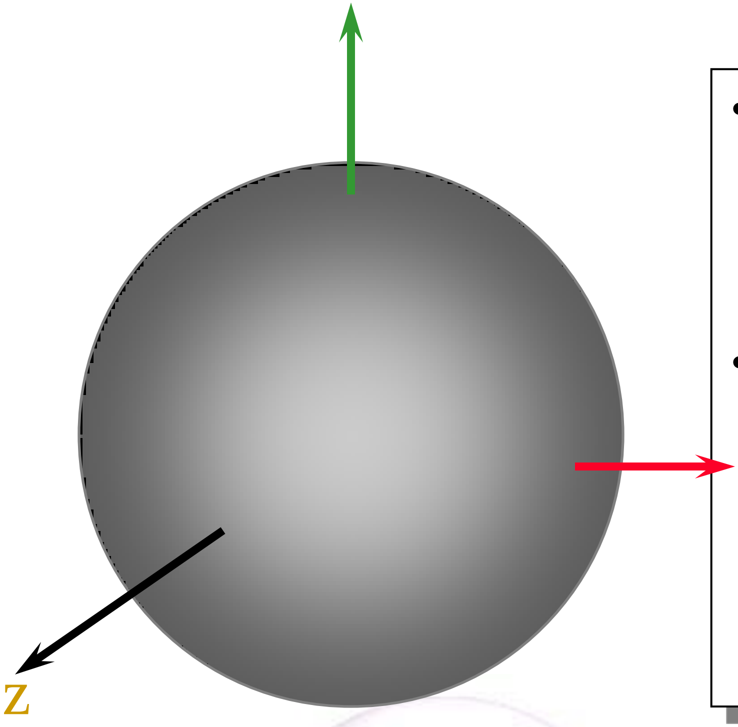
- Geometrie:

  8 Punkte, Abstände
- Topologie

  12 Kanten - wer mit wem ?

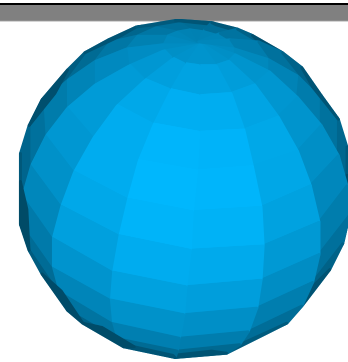
  6 Flächen

Parameter einer Kugel



- Explizite Beschreibung durch einen Satz von Polygonen (Geometrie) und ihren Zusammenhang (Topologie)

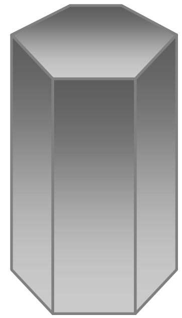
- Implizite Beschreibung durch eine Quadrik:
  - Mittelpunkt (x_0,y_0,z_0)
  - Radius r
  - $(x-x_0)^2+(y-y_0)^2+(z-z_0)^2=r^2$



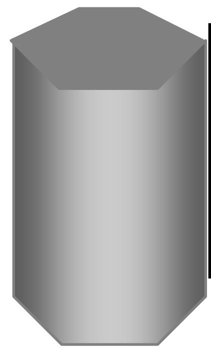
Parameter eines Zylinders

Objekt

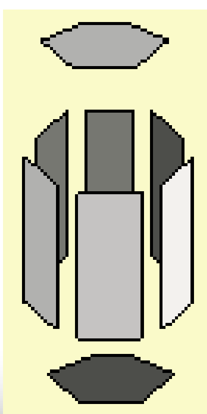
Zerlegung = Abstraktion

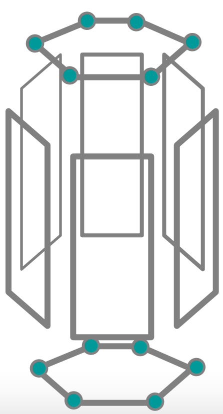


Darstellung mit notwendiger Effizienz (Anzahl Kanten)



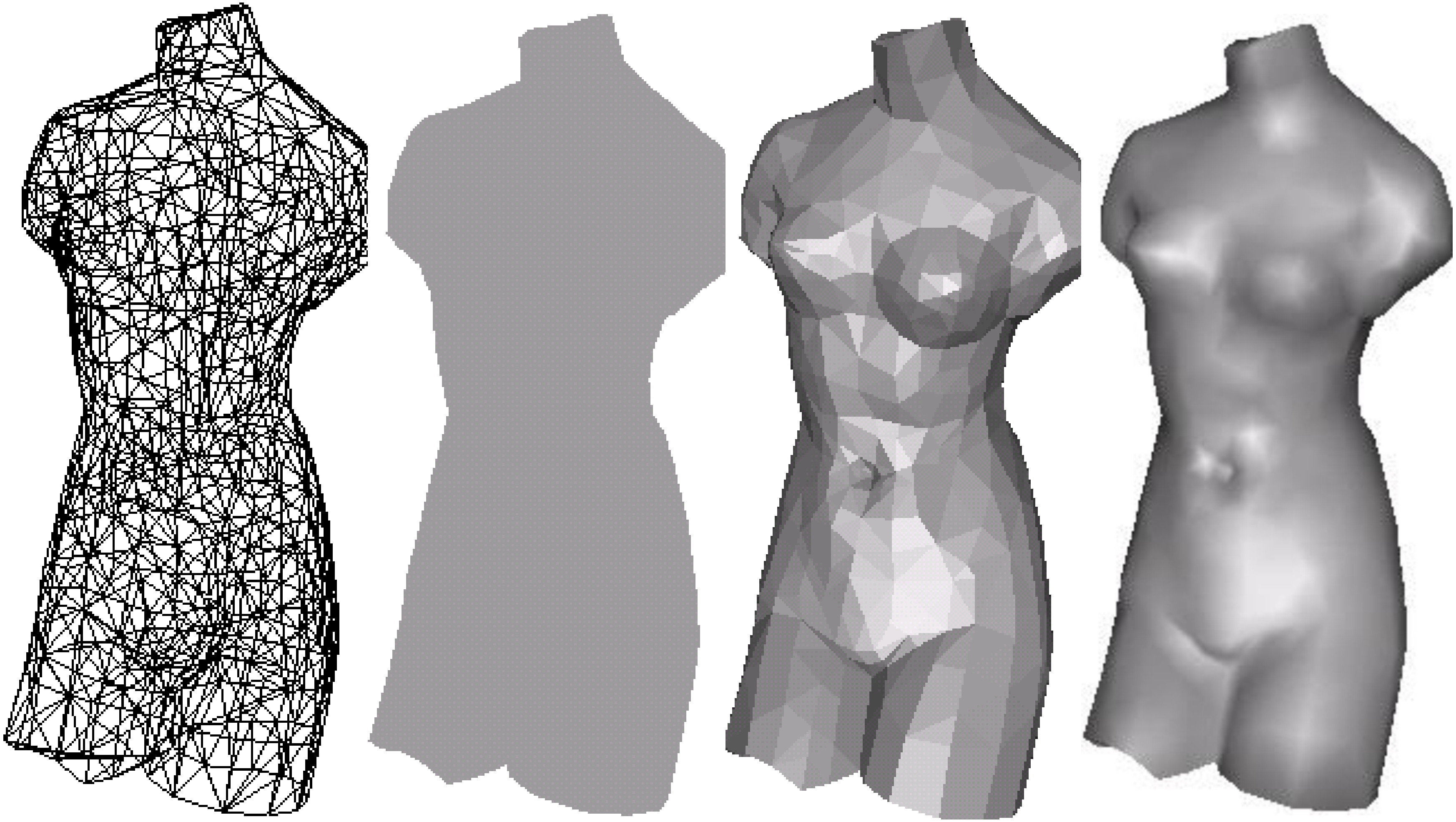
Verbesserung durch Erhöhung der Datenmenge (Anzahl Kanten) oder des Darstellungsaufwands (Shading, hier: Gouraud)





Modell = Geometrie und Topologie

Modellierungsbeispiel



Modellierungsverfahren

- Flexibilität:
  - Möglichst viele verschiedene Formen darstellbar
- Eindeutigkeit und Vollständigkeit
- Genauigkeit
  - Darstellung möglichst auf vielen Skalen korrekt
- Unmöglichkeit der Darstellung unmöglicher Objekte (z.B. Kanten ohne Flächen)
- Abgeschlossenheit gegenüber Transformationen
  - Formen ändern sich nicht unter Translation/Drehung
- Kompaktheit
  - Speicherbedarf möglichst klein, kompakte Dateiformate
  - Gegenbeispiel PBM = Public Bitmap Format

Recherchieren Sie dies !

- Effizienz
  - Möglichst schnelle Bilderzeugung

Koordinatensysteme

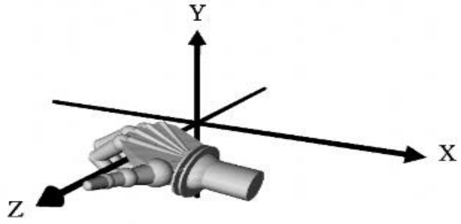
- Koordinatensysteme abstrakter Räume
  - Beispiel: Farbkoordinaten
- 2D-Koordinatensysteme: Lokation von Punkten in einer Ebene
  - Beispiel: Display-Koordinaten = Pixel-Koordinaten
  - Beispiel: Länge und Breite auf einer Kugeloberfläche => nicht-euklidische Geometrie
- 3D-Koordinatensysteme: Lokation von Punkten im Ortsraum.
  - Beispiel: Kartesische Koordinaten
  - Beispiel: Zylinderkoordinaten
- 4D-Koordinatensysteme: Bewegung von Punkten im Ortsraum
  - Beispiel: Animationspfade, Storyboards

Kartesische Koordinaten

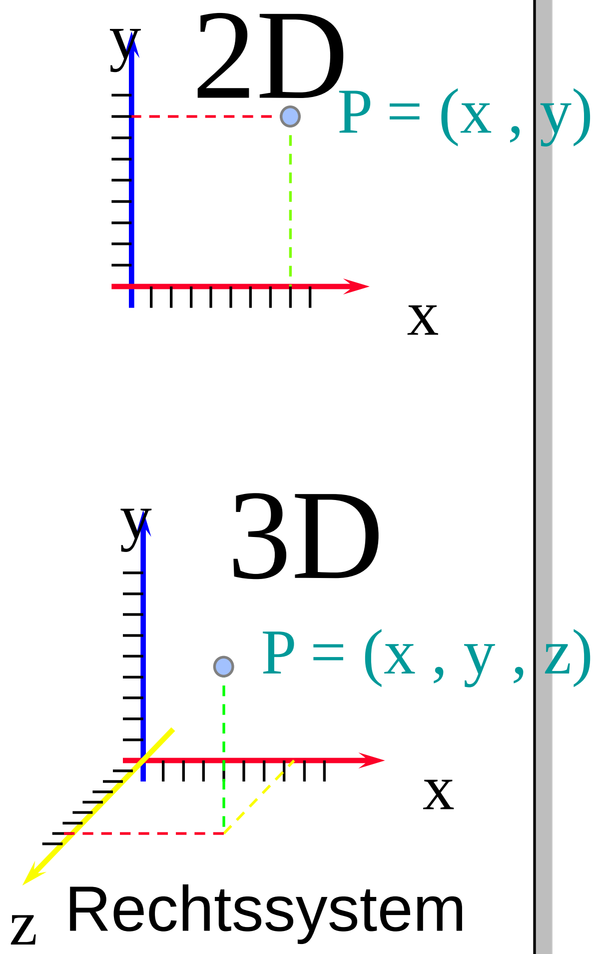
- Rechtwinklige Koordinatenachsen
- konstante Skala =
  äquidistante Einheitsmarkierungen

René Descartes

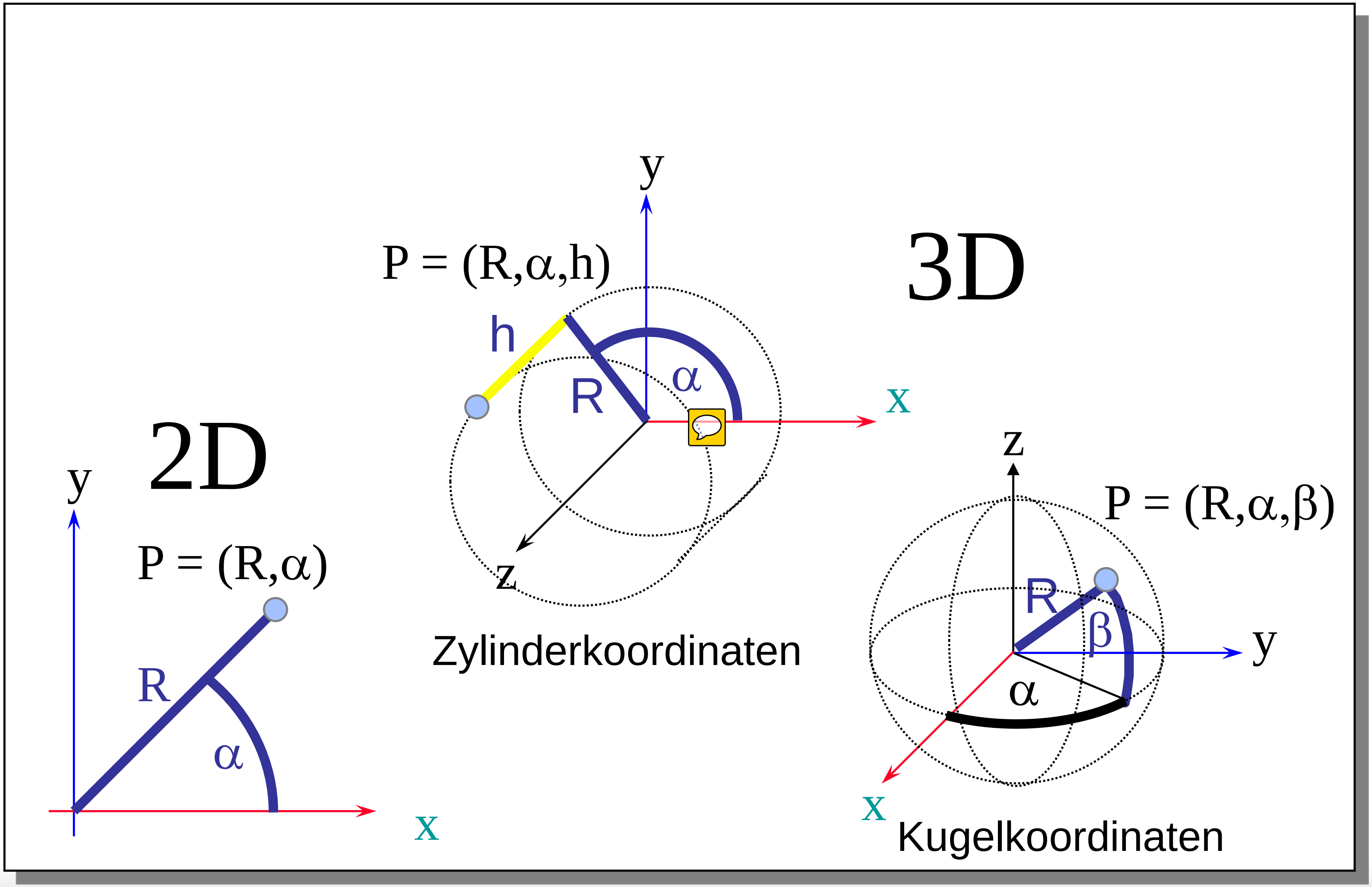
Recherchieren Sie dies !



3D: Rechte-Hand-Regel



Polarkoordinaten

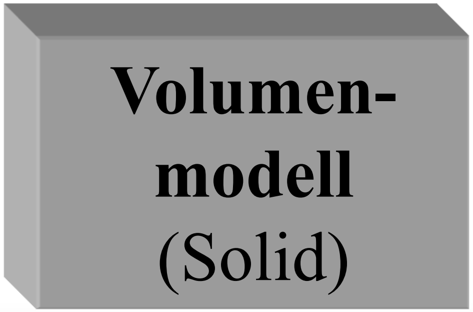


Klassifikation

Draht(gitter)modell
(Wireframe)
Objekt ist nur durch Punkte und Kanten beschrieben
Kanten sind Geradenstücke oder Kurvenstücke

Flächenmodell
(Surface Model)
Objekt ist durch Oberflächen oder umschreibende Flächen definiert
Polygonale Modelle, Freiformflächen, Parameterflächen, Rotationskörper

Volumenmodell
(Solid Model)
Modell enthält Volumeninformation, wird beschrieben durch Zusammensetzung von Primitivobjekten (CSG, Voxel, Octrees)



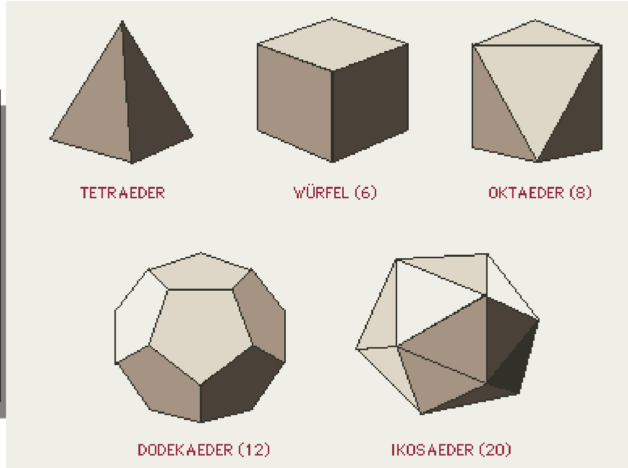
Partialmodell

Geometrie

- **Punkte**
- **Linien**
  - Gerade,Strecke
  - Kreisbogen
  - Kegelschnitt
  - Freiformkurve
- **Flächen**
  - Polygone
  - analytische: Ellipsoid, ...
  - Platonische Körper
  - Freiformflächen

Topologie

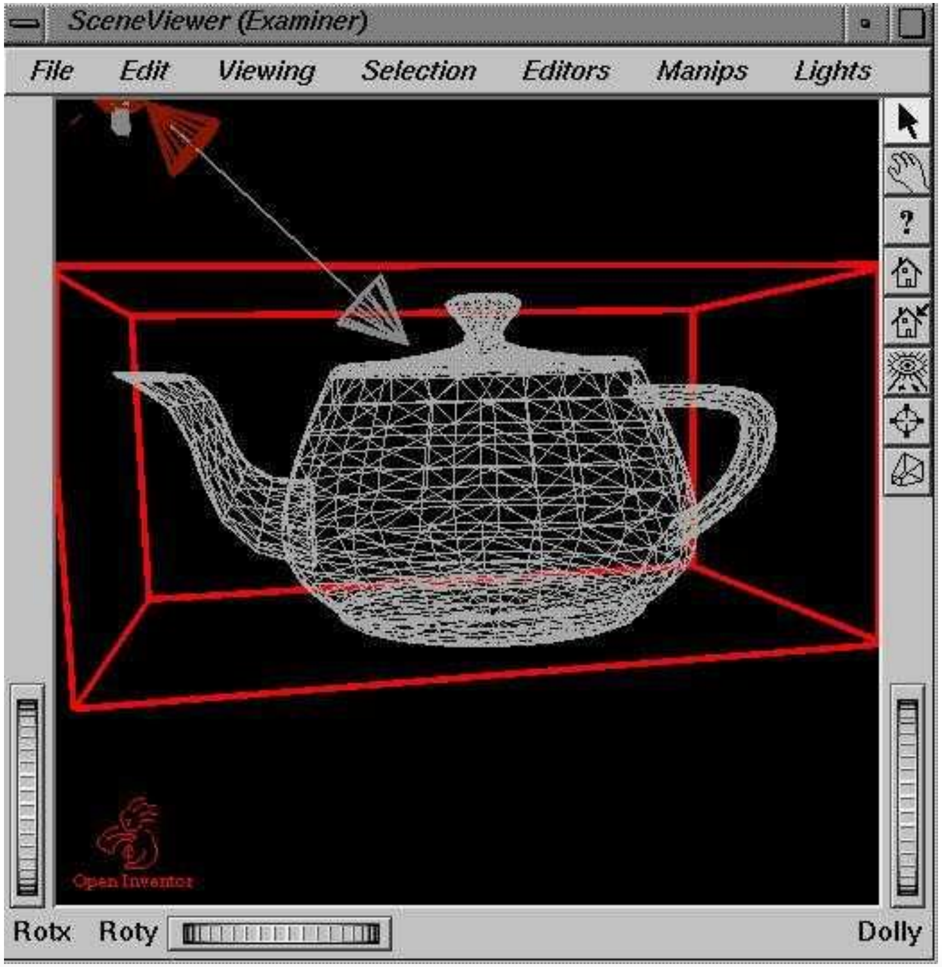
Zusammen-hänge, Kanten



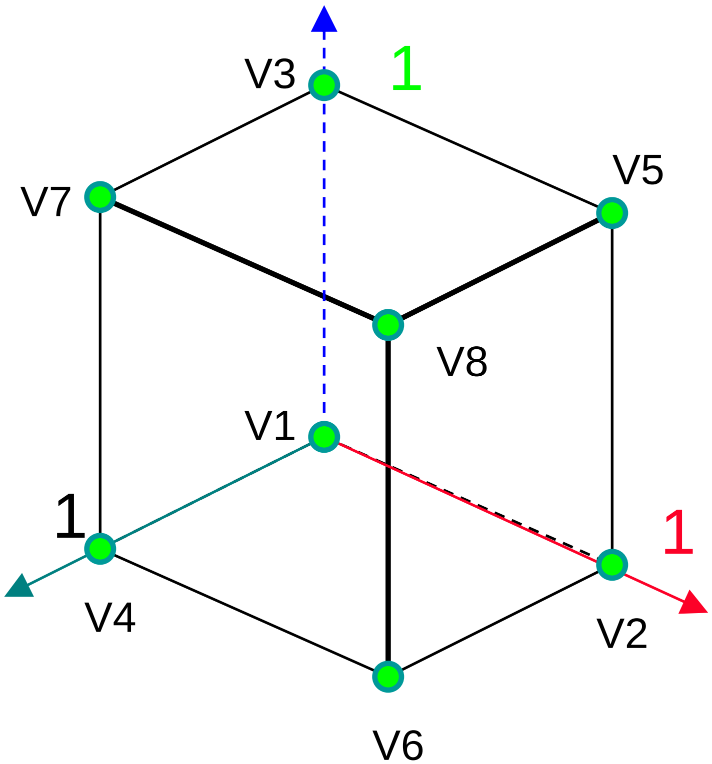
Recherchieren Sie den Zusammenhang zwischen der Anzahl der platonischen Körper, einem Duell im Morgengrauen und der Auflösbarkeit algebraischer Gleichungen

Draht(gitter)modell

- Einfachstes Modell
- Geometrie= Koordinaten der Knotenpunkte
- Topologie=Kantenmenge
- Hidden-Line/Hidden-Surface Removal möglich, aber selten wegen unrealistischen Eindrucks
- Schnitt durch ein Drahtmodell ergibt Menge von Punkten



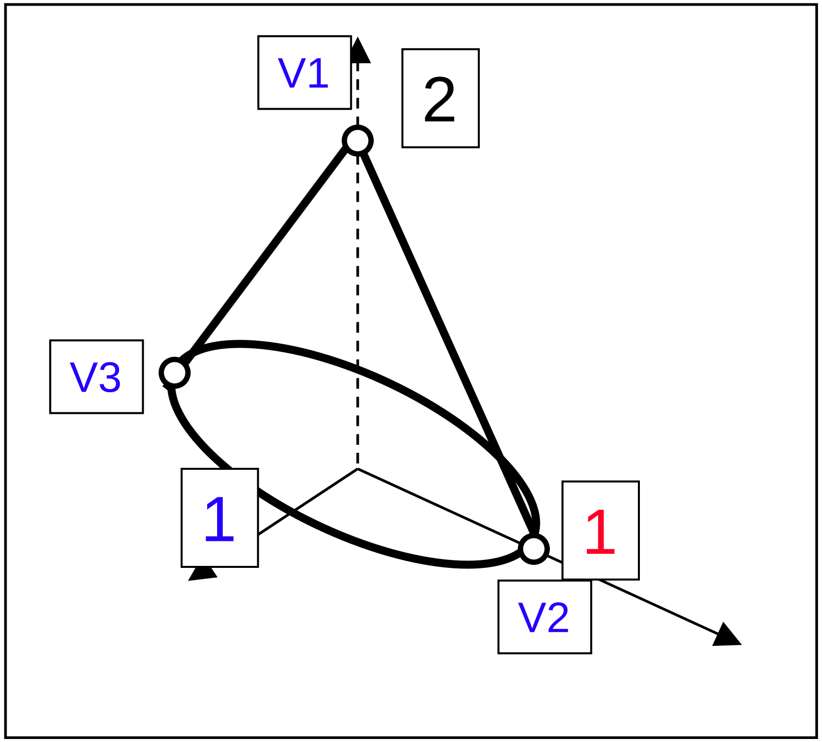
Drahtmodell mit geraden Kanten



Knoten
V1(0,0,0)
V2(0,1,0)
V3(0,0,1)
V4(1,0,0)
V5(0,1,1)
V6(1,1,0)
V7(1,0,1)
V8(1,1,1)

Kanten
E1(V1,V2)
E2(V1,V3)
E3(V1,V4)
E4(V2,V5)
E5(V5,V3)
E6(V3,V7)
E7(V7,V4)
E8(V7,V8)
E9(V6,V8)
E10(V4,V6)
E11(V6,V2)
E12(V8,V5)

Drahtmodell mit gekrümmten Kanten



Knoten
V1(0,0,2)
V2(0,1,0)
V3(0,-1,0)

Kanten

E1(V1,V2)	Gerade
E2(V1,V3)	Gerade
E3(V2,V3)	Halbkreis
E4(V3,V2)	Halbkreis

Darstellung gekrümmter Kanten:

- Exakt, d.h. durch Angabe einer formalen Beschreibung
  Hier: $(x-x_0)^2+(y-y_0)^2=1$, $z=0$, $x>0$
- Angenähert, z.B. durch Freiformkurve oder Stützstellen

Flächenmodell

- Beschreibung eines Objektes
  - explizit durch Punkte, Kanten, d.h. Angabe der Flächen durch ihre Berandung
  - implizit durch analytische Angabe der Flächen
- Schnitt durch ein Flächenmodell liefert Punkte (Schnittpunkte mit Kanten) und Verbindungen (Schnitt Fläche/Fläche)
- Das Innere eines Körpers ist im Flächenmodell hohl, die Wände sind unendlich dünn

Darstellung von Flächen
•analytisch, nur für wenige Zwecke geeignet
•meist als Polygon-Gitter, gut geeignet für Hardware-Beschleunigung
•als Freiformfläche (Stützpunkte)

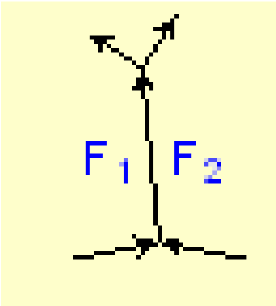
Polygonnetze

- Objekte oft schon mit wenigen Eckpunkten eines Polygonnetzes gut darstellbar
- Leicht zu bearbeiten: Clipping, Durchdringung,...
- Gekrümmte Flächen nur näherungsweise darstellbar, Darstellung kann systematisch verfeinert werden.
- Anforderungen an Polygonnetze
  - Polygone dürfen sich nicht gegenseitig durchdringen
  - Polygone sollen konvex sein
  - Polygone sollten Dreiecke sein (-> garantiert flach), höchstens Vierecke
  - Maximal zwei Polygone teilen sich eine Kante

Polygonnetze II

- Probleme
  - Löcher, frei schwebende Kanten
  - numerische Ungenauigkeiten
  - Konsistenzprüfungen notwendig
- Speicherung
  - explizit: Jedes Polygon als Kette aller seiner Eckpunkte => Eckpunkte werden mehrfach gespeichert
  - Zeiger in Punkteliste: Seiten durch Indizes definiert
  - Zeiger in Kantenliste: Kanten durch Indizes in Punkteliste, Seiten durch Indizes in Kantenliste
  - kompliziertere Strukturen denkbar, z.B: Winged Edge:

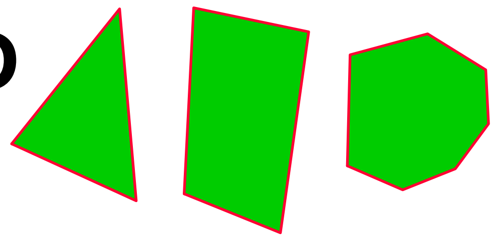
  Jede Kante „kennt“ ihre beiden benachbarten Seiten und die von den Eckpunkten ausgehenden Kanten



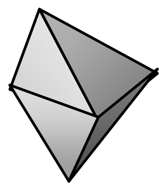
Konvexität von Polygonen und Polyedern

konvex

2D

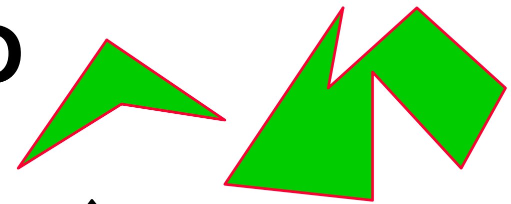


3D

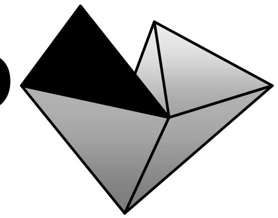


Nicht-konvex = „konkav“

2D



3D

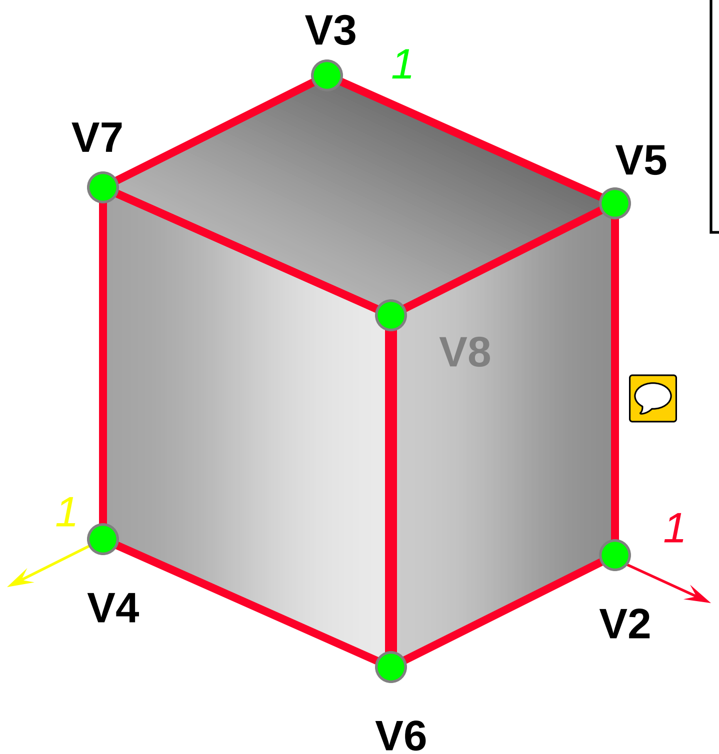


Viele Algorithmen der Computergrafik sind für konvexe Objekte sehr viel einfacher

Eigene Problemklasse:
Erzeugung konvexer Hüllen
Zerlegung konkaver Objekte in konvexe Teile

Recherchieren Sie dies !

Flächenmodell eines Würfels



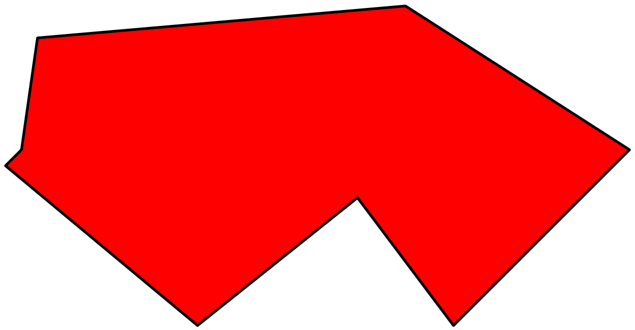
Knoten
V1(0,0,0)
V2(0,1,0)
V3(0,0,1)
V4(1,0,0)
V5(0,1,1)
V6(1,1,0)
V7(1,0,1)
V8(1,1,1)

Kanten
E1(V1,V2)
E2(V1,V3)
E3(V1,V4)
E4(V2,V5)
E5(V5,V3)
E6(V3,V7)
E7(V7,V4)
E8(V7,V8)
E9(V6,V8)
E10(V4,V6)
E11(V6,V2)
E12(V8,V5)

Flächen
F1(V1,V2,V6,V4)
F2(V4,V6,V8,V7)
F3(V6,V2,V5,V8)
F4(V2,V5,V3,V1)
F5(V1,V3,V7,V4)
F6(V4,V1,V3,V7)

Reguläres Polygon

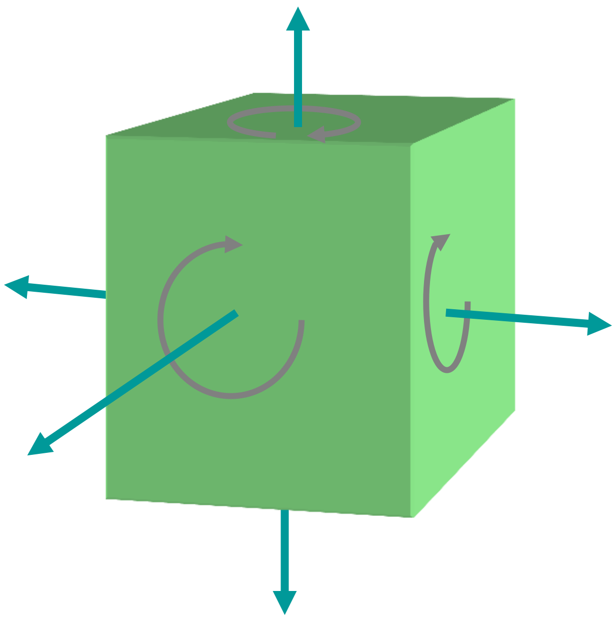
- Ein reguläres Polygon ist ein 2D-Objekt, bestimmt durch einen geschlossenen, sich nicht selbst schneidenden Zug von Geradenstücken



P=(E1,E2,E3,E4...En)

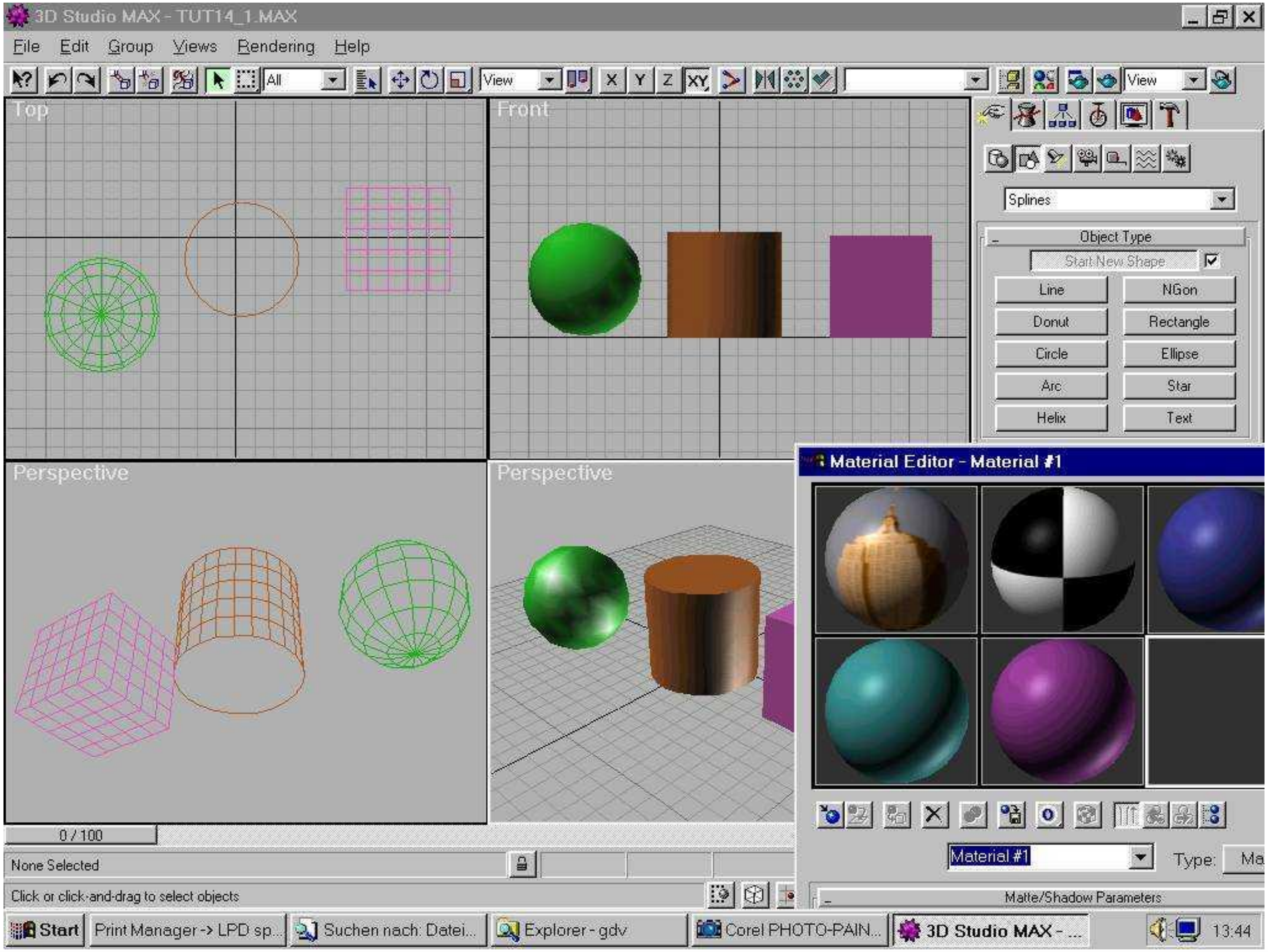
- Jede Kante besitzt genau zwei Eckpunkte

Orientierung von Polygonen



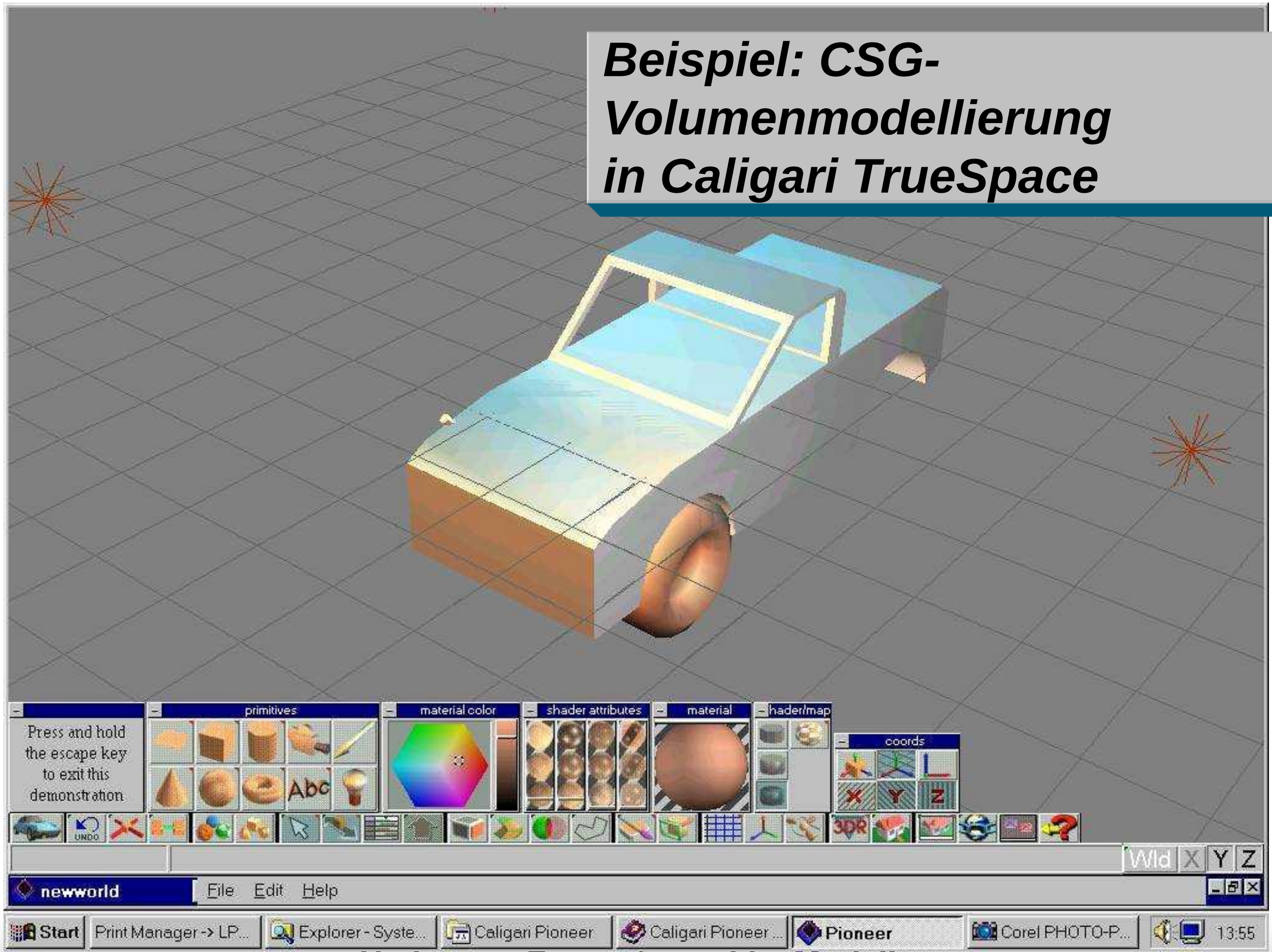
- Eckpunkte der Seitenflächen werden in Datenstrukturen der Computergrafik i.A. so sortiert, dass sie von außen gesehen im Uhrzeigersinn verlaufen. (Linke-Hand-Regel)
- Die mathematische Orientierung von Flächen wird i.A. andersherum gezählt (Rechte-Hand-Regel)
- Alle Normalenvektoren zeigen nach außen

3D-Studio Max

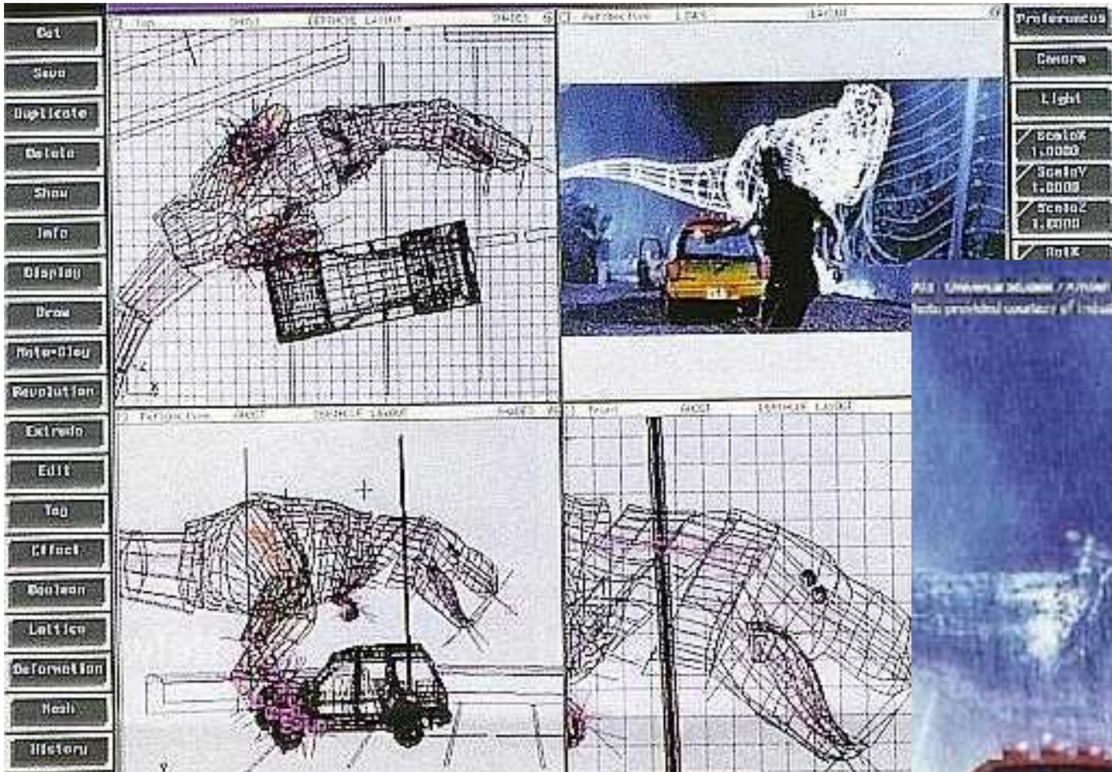


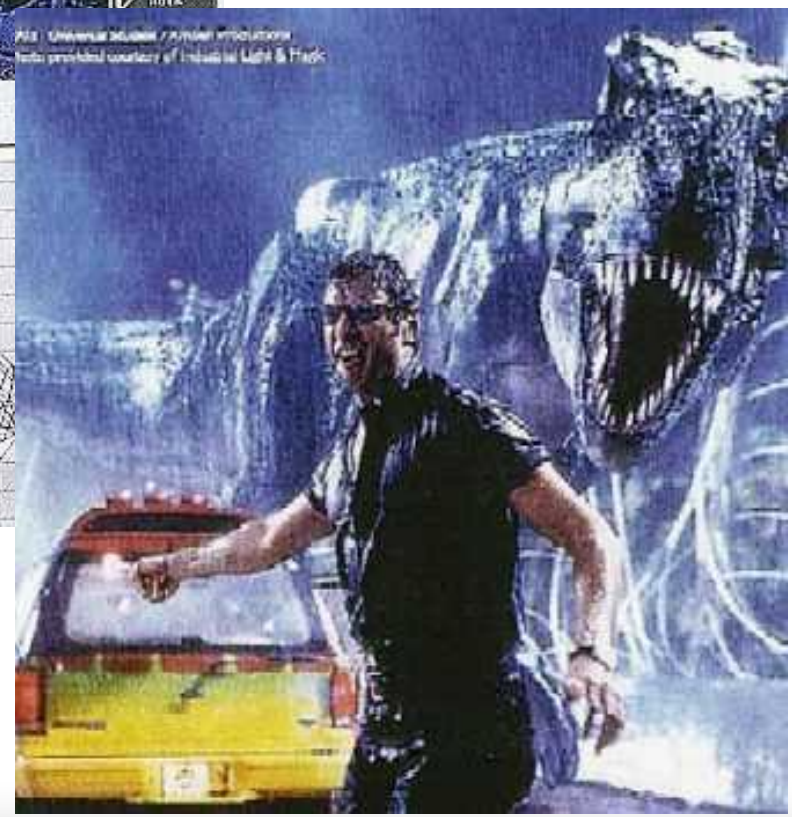
Caligari TrueSpace

Beispiel: CSG-Volumenmodellierung in Caligari TrueSpace

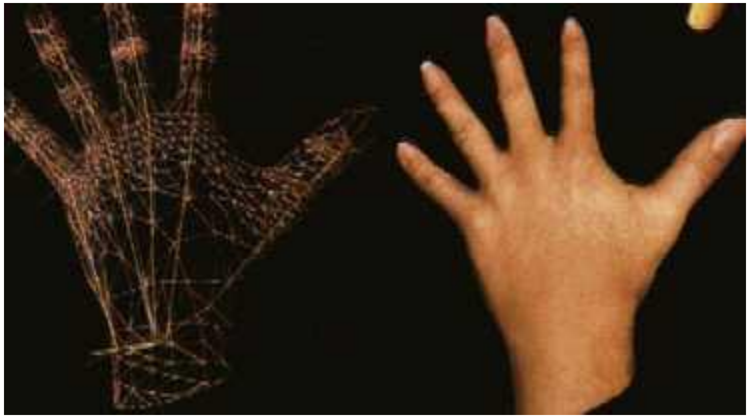


Beispiele Polygonnetze 1993



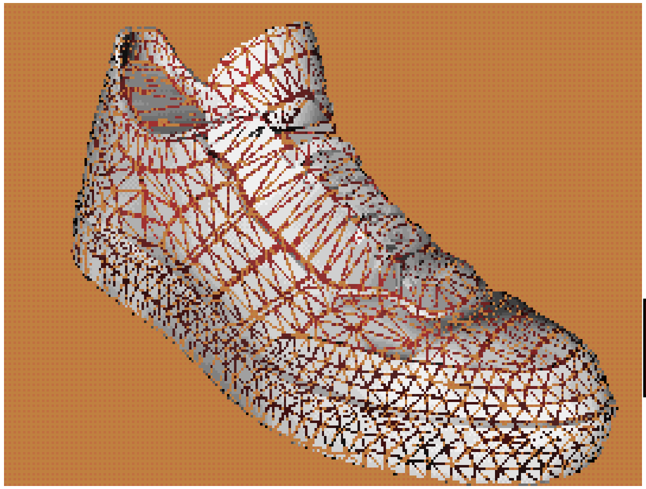


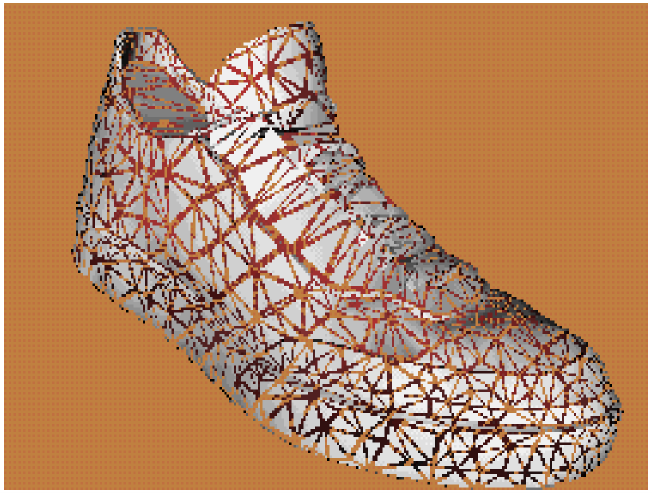
Polygonnetze für komplexe Oberflächen



Problemklasse:
Reduktion der Anzahl der Polygone
bei gleicher subjektiver Form

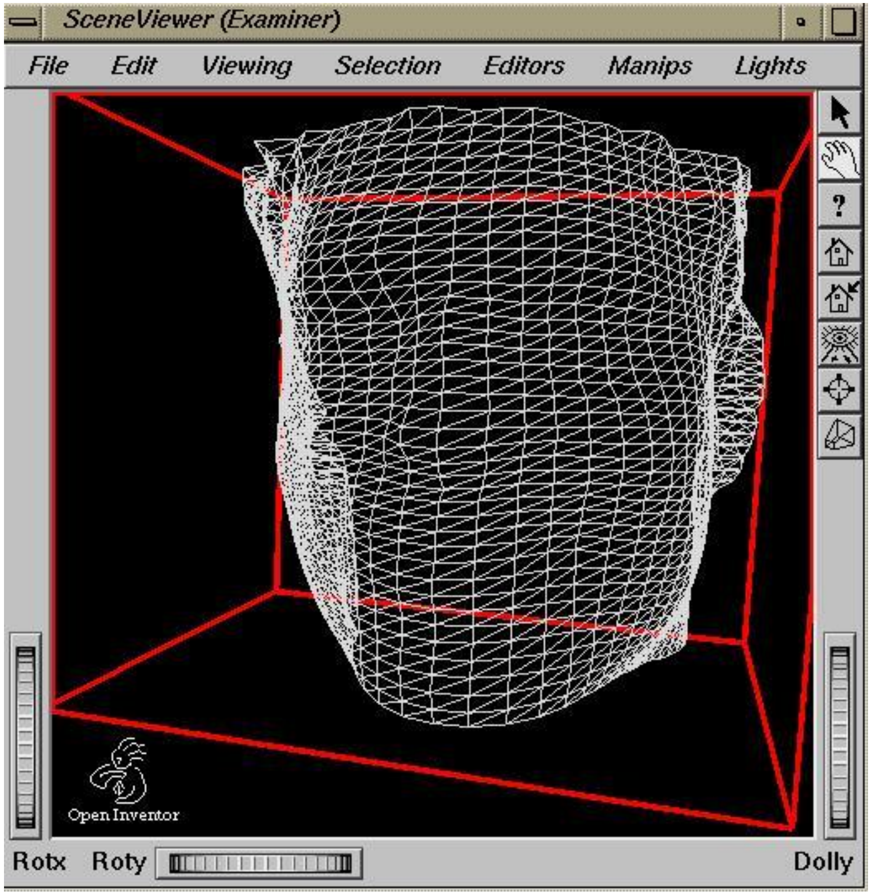
Mesh-Decimation

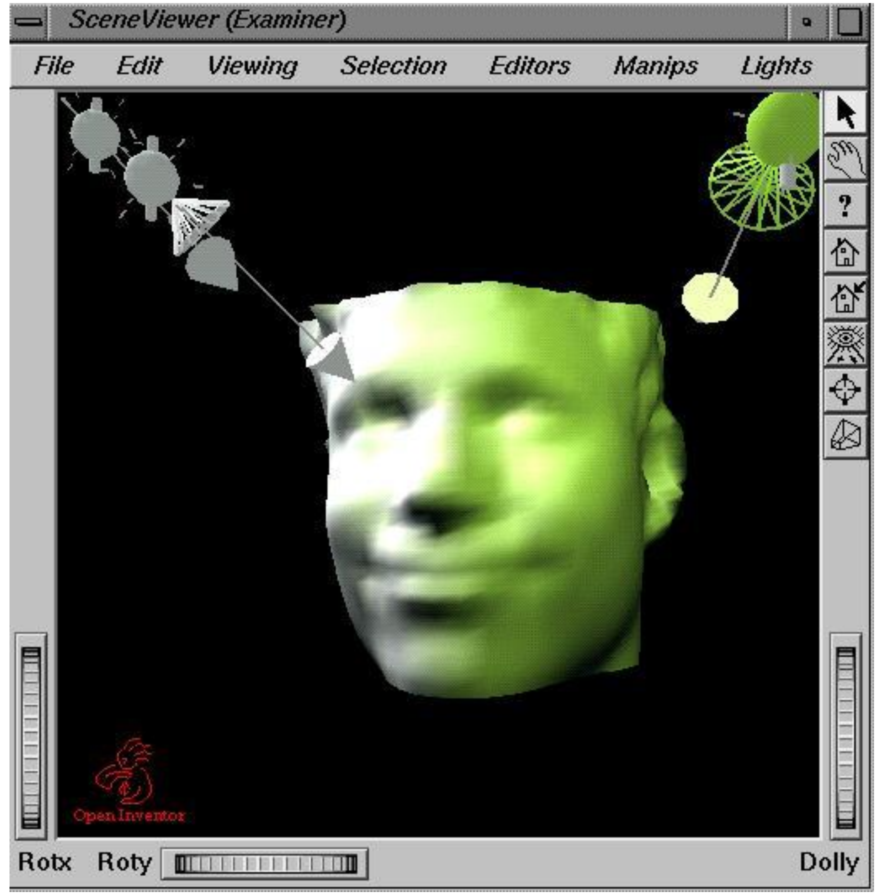




Polygonales Flächenmodell

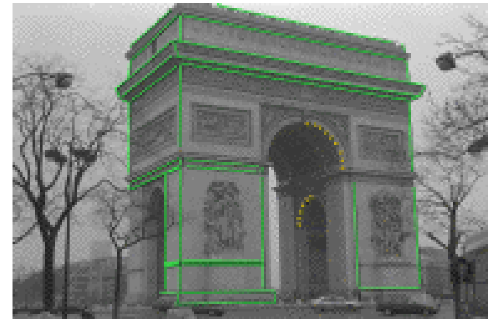
schattiertes Flächenmodell mit 2 lokalen Lichtquellen

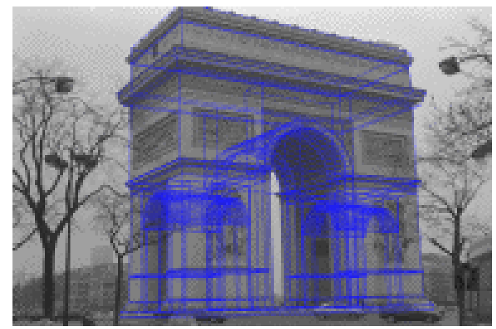




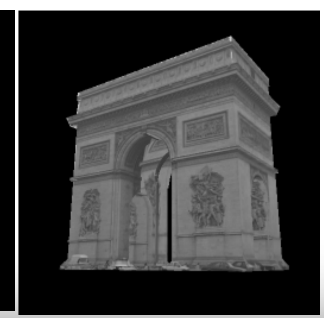
Gebäude-Rekonstruktion

Original photograph with marked edges — Recovered model — Model edges projected onto photograph — Synthetic rendering





Bessere (schnellere) Lösung:
Texturen auf einfachen geometrischen Objekten



Gekrümmte Flächen

- Viele Darstellungen benötigen gekrümmte Flächen
- Optimierungsproblem:
  - Eignung für die Anwendung
  - Komplexität handhabbar
- Drei Darstellungsmöglichkeiten
  - implizit: $f(x,y,z)=0$
  - explizit: $z=f(x,y)$
  - parametrisiert (meistverwendet):

    $x=X(u,v) \quad y=Y(u,v) \quad z=Z(u,v)$
    - Keine Mehrdeutigkeit
    - Leicht in Hardware zu realisieren

Stetigkeit von Kurven und Flächen

•Kurven heißen stetig, wenn sie sich an beliebigen „Anknüpfungspunkten“ ohne „Sprung“ fortsetzen lassen. Die Menge der stetigen Funktionen bezeichnet man als $C^{0.}$

•Kurven heißen einmal stetig differenzierbar, wenn sie an „Anknüpfungspunkten“ auch mit der gleichen Steigung fortgesetzt werden. Die Menge der einmal stetig differenzierbaren Funktionen heißt $C^{1.}$

...

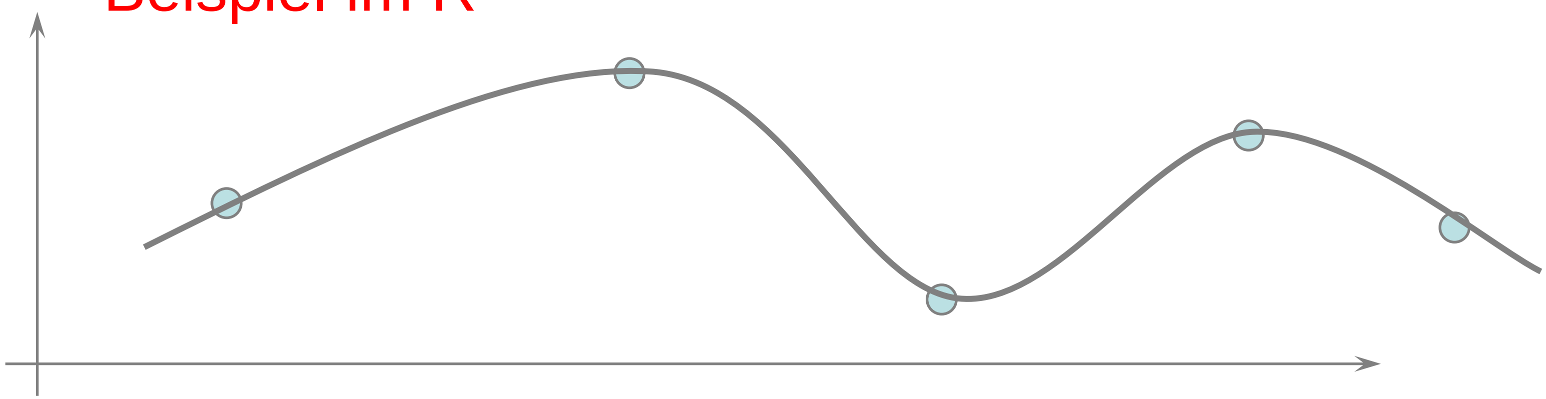
•Kurven heißen n-mal stetig differenzierbar, wenn sie an „Anknüpfungspunkten“ auch mit der gleichen n-ten Ableitung fortgesetzt werden können. Die Menge der n-mal stetig differenzierbaren Funktionen heißt $C^{n.}$

- Die Zusammensetzung von gekrümmten Oberflächen aus Teilpolygonen verlangt mindestens C^0-Stetigkeit.
- C^1-Stetigkeit wird oft in hinreichender Qualität durch das Gehirn „ergänzt“.

Problemklasse: Interpolation

Bei der **Interpolation** wird eine (möglicherweise unbekannte) Funktion **g(x)** durch eine zu ermittelnde Funktion **f(x)** so angenähert, daß zumindest an gegebenen **Stützstellen** x_i, $i=0,\ldots,n$ gilt: $\mathbf{f(x_i) = g(x_i)}$

Beispiel im R^2



Interpolationstechniken

- **Lagrange**
  - einfache Bestimmung durch Angabe von Punkten
  - Aneinanderfügung nur mit Stetigkeit 0-ter Ordnung C^0 (Knicke)
  - Grad der Polynome = Anzahl der Stützstellen-1
  - Oszillation ab Grad 5
  - Kann durch Neville´s Algorithmus verbessert werden, hat aber nur theoretische Bedeutung

- **Hermite**
  - An Stützstellen müssen auch Ableitungen angegeben werden
  - Nachteile wie Lagrange, aber Aneinanderfügung C^1-stetig
- **Bikubische Splines**
  - Ableitungen in Stützstellen werden automatisch ermittelt
  - Spezialfall von Hermite
  - Polynome immer vom Grad 3, Aneinanderfügung C^2-stetig

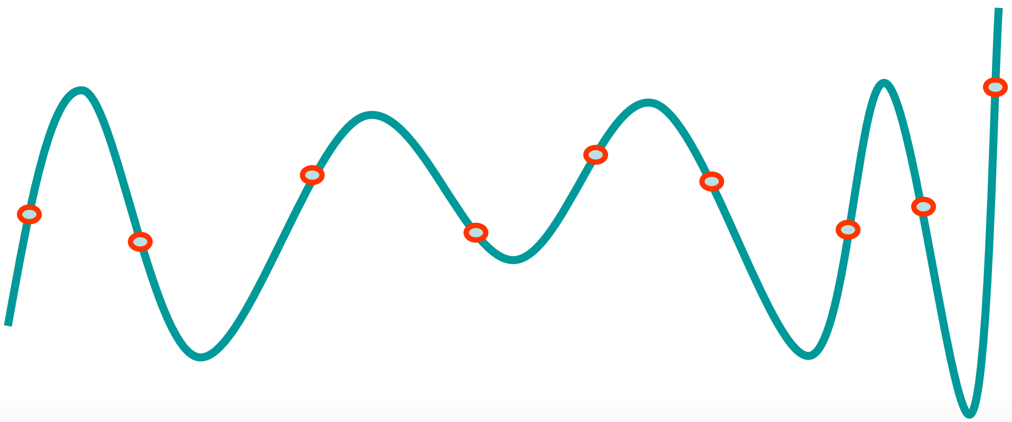
Lagrange-Interpolation

Geg.: n+1 Punkte (Stützpunkte)
Ges.: Ein Polynom n-ten Grades, das diese Punkte durchläuft

$$P_n(x) = \sum_{i=0}^{n} y_i \frac{(x-x_0)\ldots \quad (x-x_{i-1})(x-x_{i+1})\ldots \quad (x-x_n)}{(x_i-x_0)\ldots \quad (x_i-x_{i-1})(x_i-x_{i+1})\ldots \quad (x_i-x_n)}$$

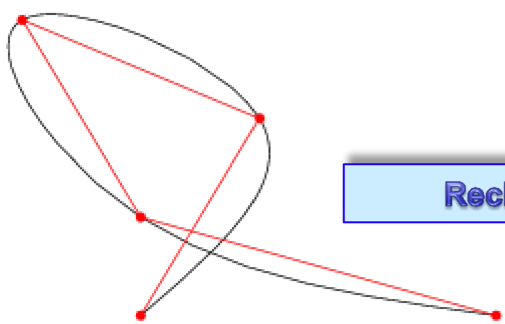
Problem: Bei n>4 fängt Lösung an zu oszillieren.

Was hat das mit dem Duell im Morgengrauen zu tun ?



Spline-Interpolation

- Idee: Angabe der Ableitungen in Stützstellen nicht erforderlich
- Aber **Spline-Bedingung**:
  Die ersten d Ableitungen in den Stützstellen müssen übereinstimmen
- Bei Splines vom Grad n (Ordnung n+1) müssen d-1 Ableitungen übereinstimmen
- Bei **kubischen Splines** (Grad 3) stimmen die ersten beiden Ableitungen überein.
  - Näheres siehe http://mathworld.wolfram.com/CubicSpline.html
  - http://www.alglib.net/interpolation/spline3.php



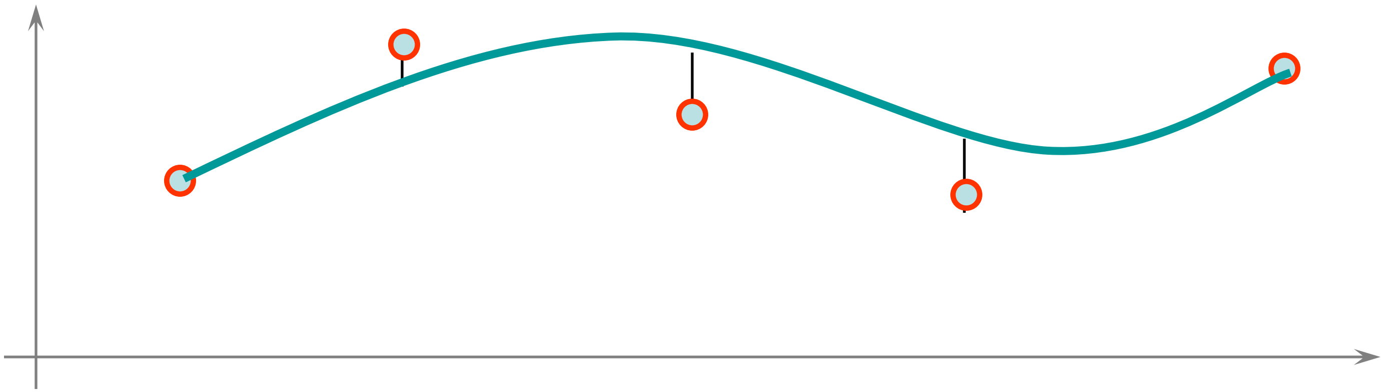
Recherchieren Sie dies !

Problemklasse: Approximation

- Eine Funktion **g(x)** soll durch **f(x)** *angenähert* werden. Dabei wird *nicht* verlangt, daß sie gegebene **Kontrollpunkte** p_i (i=1,...n) durchquert.

- Es soll u.a. das Ziel $\sum_{i=1}^{n} \left| f(p_i) - g(p_i) \right| \longrightarrow 0$ verfolgt werden.

Beispiel im R^2



Bézier-Approximation I

- entwickelt von **Pierre Bézier**, Renault
  Entwurf elegant gekrümmter und paßgenauer Oberflächen im Automobilbau/Karosseriedesign
- geg: n+1 Kontrollpunkte $p_i(x_i, y_i, z_i)$

$B_{i,n}(u)$ Blendfunktionen
(Bernstein-Polynome)

$$
\begin{aligned}
P(u) &= \sum_{i=0}^{n} p_i B_{i,n}(u) \\
&= \sum_{i=0}^{n} p_i \cdot C_{n,i} \cdot u^i \cdot (1-u)^{n-i} \\
&= \sum_{i=0}^{n} p_i \cdot \frac{n!}{i! \cdot (n-i)!} \cdot u^i \cdot (1-u)^{n-i}
\end{aligned}
$$

Bézier-Approximation II

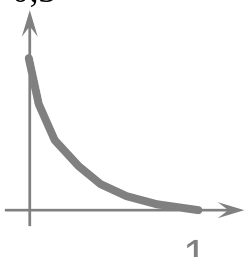
Eigenschaften von Bézier-Kurven:

- Die Einhüllende ist konvex
- Sie durchqueren ersten und letzten Kontrollpunkt.
- Gilt $P_0 = P_n$, so ist die Kurve geschlossen.
- unabhängig vom Koordinatensystem, d.h. invariant unter Transformationen
- Kurven sind überall beliebig oft stetig differenzierbar.
- Verschiebung eines Kontrollpunktes verändert ganze Kurve, aber langsam
- tangiert Verbindungslinien $P_0\, P_1$ und $P_{n-1}\, P_n$

Bézier-Approximation III

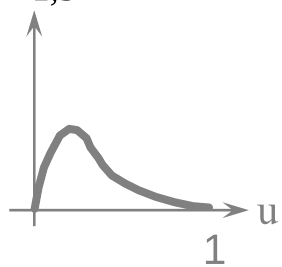
- **üblicherweise verwendet man n=4 Kontrollpunkte**
  (bikubische Parameterdarstellung)

$B_{0,3}(u) = (1-u)^3$



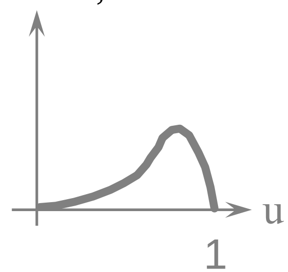
Gewichtsfunktion für den 1. Kontrollpunkt

$B_{1,3}(u) = 3u(1-u)^2$



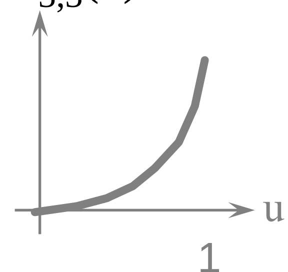
Gewichtsfunktion für den 2. Kontrollpunkt

$B_{2,3}(u) = 3u^2(1-u)$



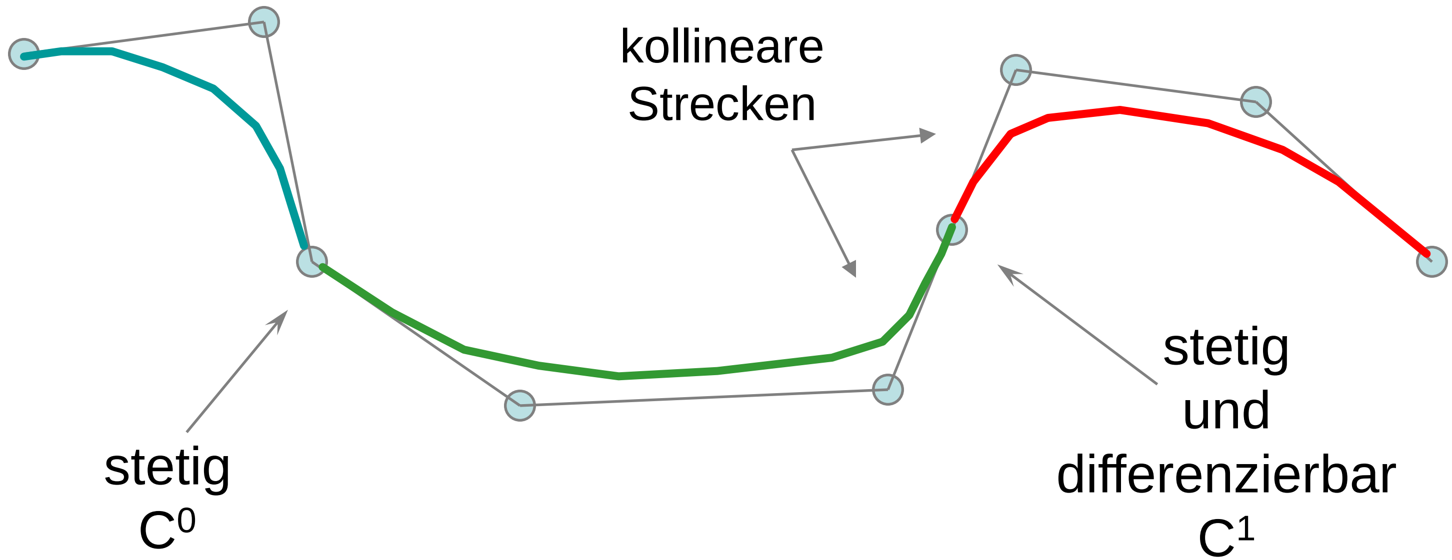
Gewichtsfunktion für den 3. Kontrollpunkt

$B_{3,3}(u) = u^3$



Gewichtsfunktion für den 4. Kontrollpunkt

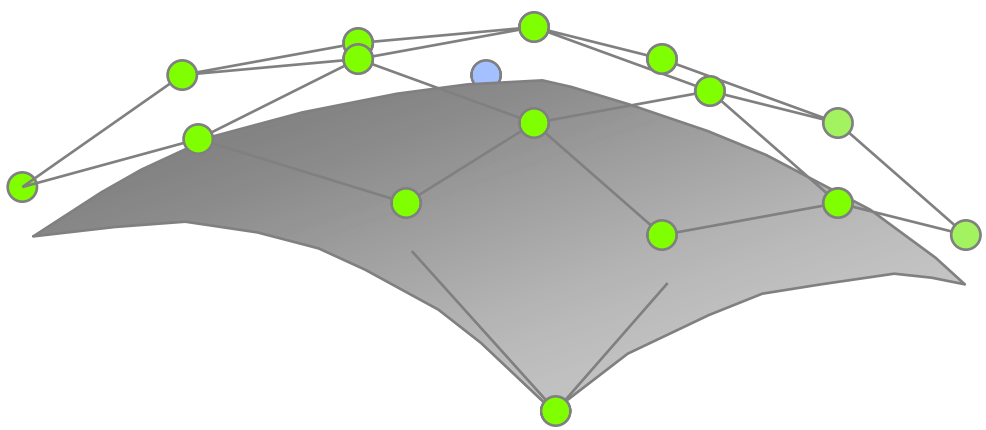
Aneinanderfügen von Béziersegmenten



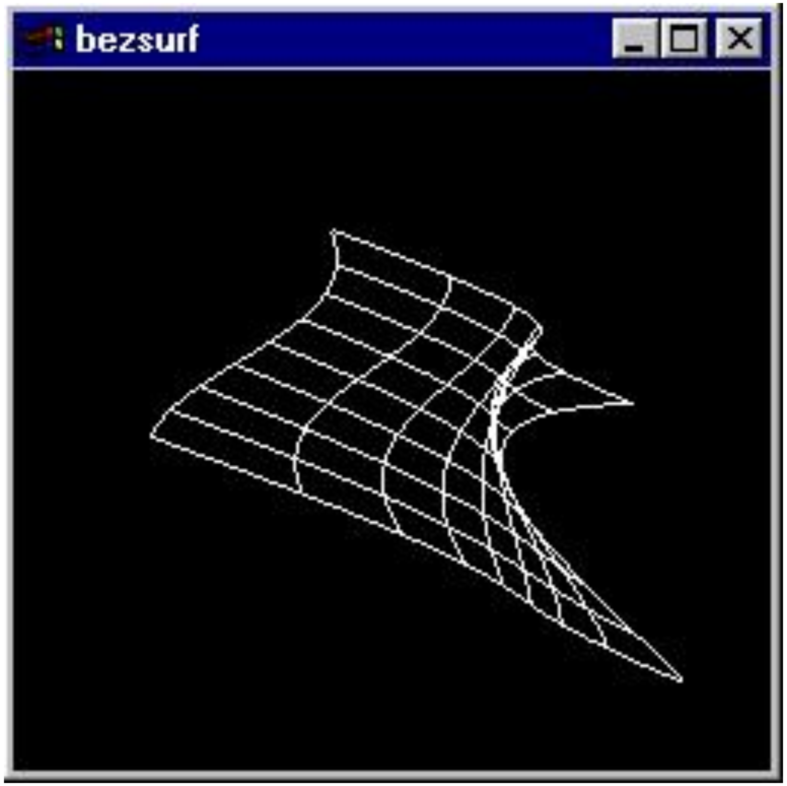
Bézier-Flächen

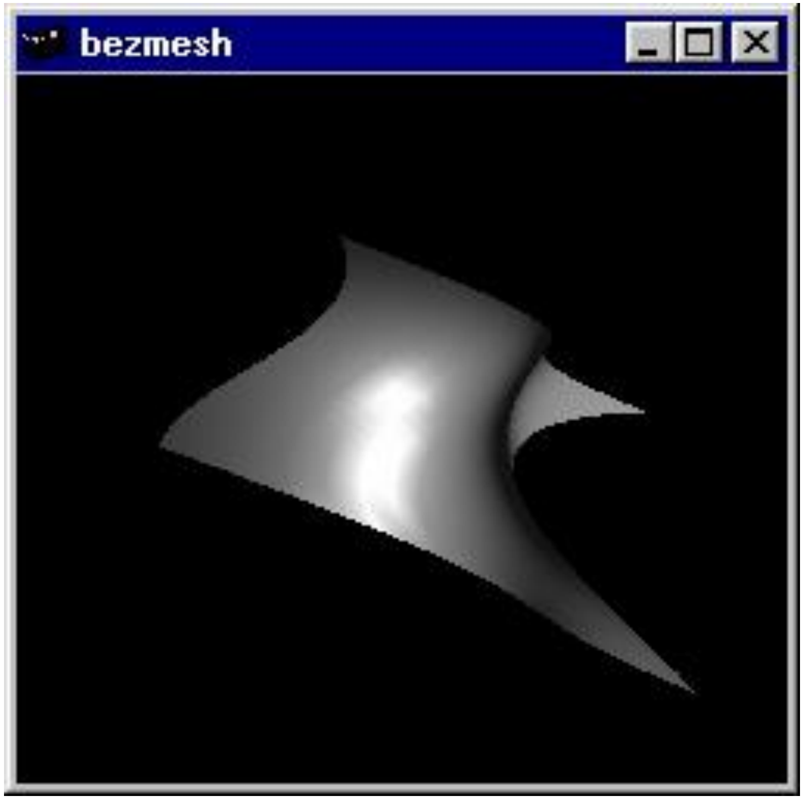
- Durch separate Anwendung des Bézier-Prinzips auf 2 Parameter entsteht eine Fläche (**Bézier-Patch**)
- Querschnitte durch die Fläche in u- oder v-Richtung liefern genau die Bézierkurven der entsprechenden Kontrollpunkte.

$$P(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} \vec{p}_{i,j} B_{i,n}(u) B_{j,m}(v)$$



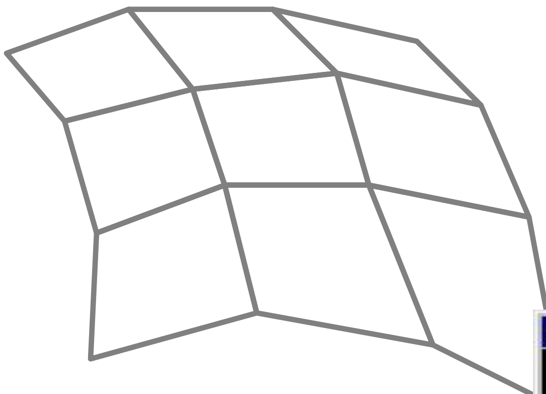
Bézier-Flächen

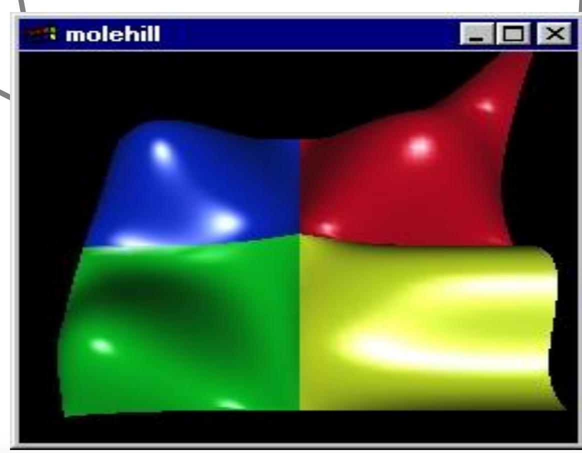


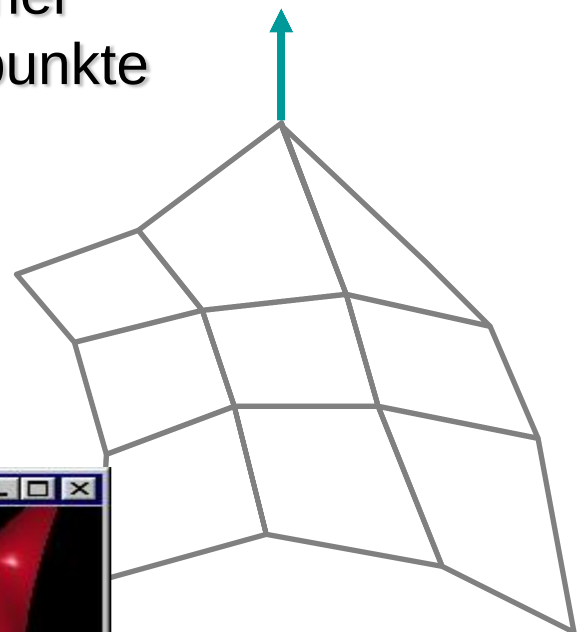


Die Deformation von Bézier-Flächen

Verschiebung einzelner
oder mehrerer Kontrollpunkte







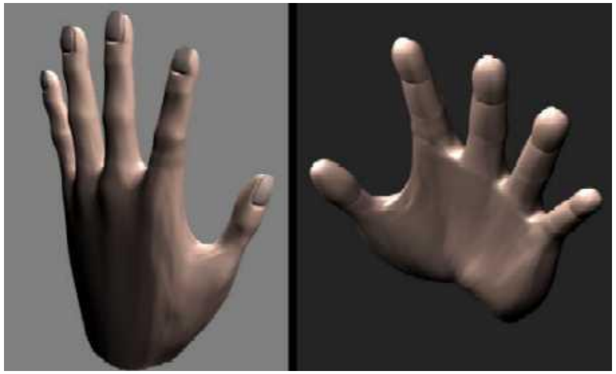
Andere Approximationstechniken

B-Splines

- \+ bessere **lokale** Kontrolle
- \+ **C²-Stetigkeit** beim Aneinanderfügen
- \+ **Knotenvektor** enthält Parameter (Anzahl der Parameter abh. von Anzahl der Kontrollpunkte, quadratisch, kubisch, quartisch, etc.)
- \+ **Gewichtsverstärkung** von Kontrollpunkten (Spitzen, Knicke) durch Knoten-Wiederholung.

Non-Uniform-Rational-B-Splines

•Knoten **ungleichmäßig** verteilt
•**rationale Form**
(Brüche von B-Splines)
•in Graphikworkstations **Hardware-Unterstützung** für NURBS
•als Darstellungsprimitiv NURBS direkt
•**Trimmingmöglichkeiten** durch Parameterintervalle



NURBS I

Non-Rational Uniform Bi-Splines

- gegeben (m+1) * (n+1) Kontrollpunkte $P_{ij}(x_{ij}, y_{ij}, z_{ij})$
  - i läuft von 0 ... m, j läuft von 0 ... n, k sei die Ordnung der NURBS-Kurve
  - Jeder Punkt erhält einen Gewichtsfaktor $w_{ij} \in (0,1)$

$$P(u,v) = \frac{\sum_{i=0}^{m} \sum_{j=0}^{n} B_{i,k}(u) B_{j,k}(v) w_{ij} P_{ij}}{\sum_{i=0}^{m} \sum_{j=0}^{n} B_{i,k}(u) B_{j,k}(v) w_{ij}}$$

$B_{i,k}(u)$ Blendfunktionen
(Bernstein-Polynome)

NURBS II

- Basisfunktionen: Gegeben sei eine nicht-fallende Sequenz (Knotenvektor) U aus m+1 reellen Zahlen

$$U = \{u_0, u_1, \ldots u_m\}$$

$$B_{i,1}(u) = \begin{cases} 1, \text{wenn } u_i \le u < u_{i+1} \\ 0, \text{sonst} \end{cases}$$

$$B_{i,r}(u) = \frac{u - u_i}{u_{i+r-1} - u_i} B_{i,r-1}(u) + \frac{u_{i+r} - u}{u_{i+r} - u_{i+1}} B_{i+1,r-1}(u)$$

- Eigenschaften:
  - Stetigkeit in den Kontrollpunkten ist C^{k-1-x}, dabei ist x die Zahl der gleichen Knotenpunkte
  - Die Kurve (Fläche) liegt vollständig innerhalb der konvexen Hülle ihrer Kontrollpunkte

NURBS III

- Beispiel: k=3, m=4
- U={0, 0.25, 0.5, 0.75, 1}

$$B_{0,1} = 1, \text{wenn } u < 0.25$$

$$B_{1,1} = 1, \text{wenn } 0.25 \leq u < 0.5$$

$$B_{2,1} = 1, \text{wenn } 0.5 \leq u < 0.75$$

$$B_{3,1} = 1, \text{wenn } 0.75 \leq u < 1$$

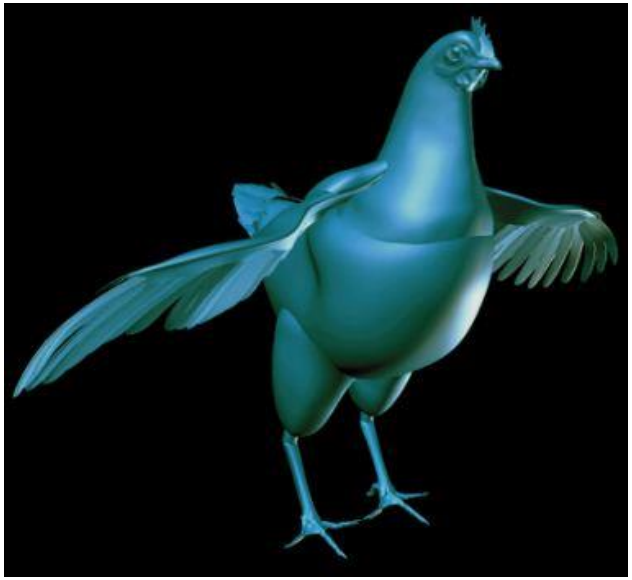
$$B_{0,2} = \begin{cases} 4u, \text{wenn } u < 0.25 \\ 4(0.5\text{-}u), \text{wenn } 0.25 \leq u < 0.5 \end{cases}$$

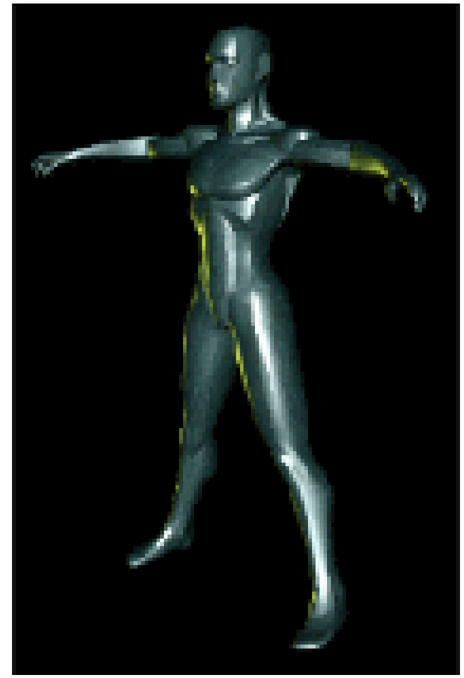
$$B_{1,2} = \begin{cases} 4(u - 0.25), \text{wenn } 0.25 \leq u < 0.5 \\ 4(0.75\text{-}u), \text{wenn } 0.5 \leq u < 0.75 \end{cases}$$

$$B_{2,2} = \begin{cases} 4(u - 0.5), \text{wenn } 0.5 \leq u < 0.75 \\ 4(1\text{-}u), \text{wenn } 0.75 \leq u < 1 \end{cases}$$

$$B_{3,2} = 4(u - 0.75), \text{wenn } 0.75 \leq u < 1$$

NURBS-Modell





Konvertierung von Flächendarstellungen

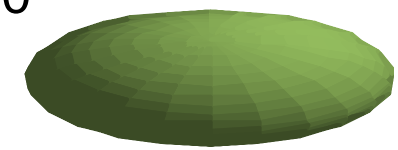
- Alle **Tensorproduktflächen** lassen sich in die **Bézier-Basis** (Bernsteinpolynome) konvertieren.
- Für Bézierflächen gibt es elegante **Unterteilungsalgorithmen**, die eine gekrümmte Fläche in beliebig viele ebene polygonale Flächen zerlegen.
- Zur Bilderzeugung werden letzten Endes in der Regel **polygonale Modelle** verwendet: Freiformflächen werden per Software oder Hardware wieder in Dreiecke zerlegt.

Quadriken

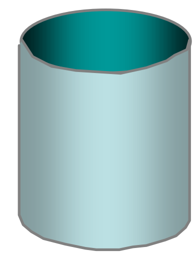
- **Quadriken** sind Flächen im Raum, die sich analytisch durch eine Gleichung zweiter Ordnung definieren lassen.
- **Allgemeine Form:**
  $Ax^2 + By^2 + Cz^2$
  $+ Dxy + Exz + Fyz$
  $+ Gx + Hy + Iz + K = 0$
- Liegen die Hauptachsen parallel zu den Koordinatenachsen, so entfallen die gemischten Glieder (D = E = F = 0)
  - Schnittprüfungen sind mathematisch einfach
  - Bestimmung der Koeffizienten ist schwierig

Spezialfälle:

- **Kugel**:
  $x^2 + y^2 + z^2 - r^2 = 0$

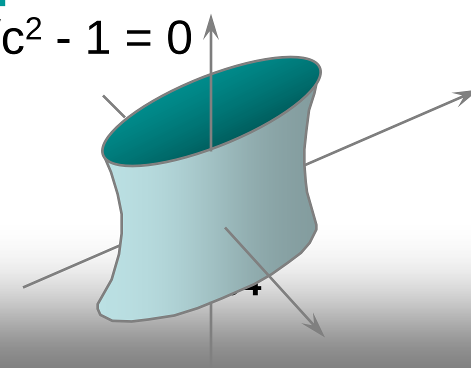


- **Ellipsoid:**
  $x^2/a^2 + y^2/b^2 + z^2/c^2 - 1 = 0$

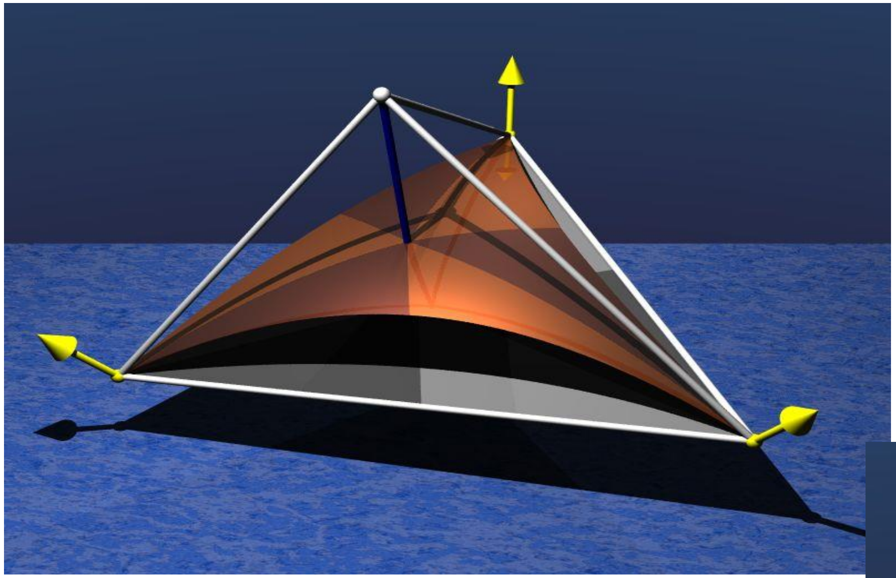


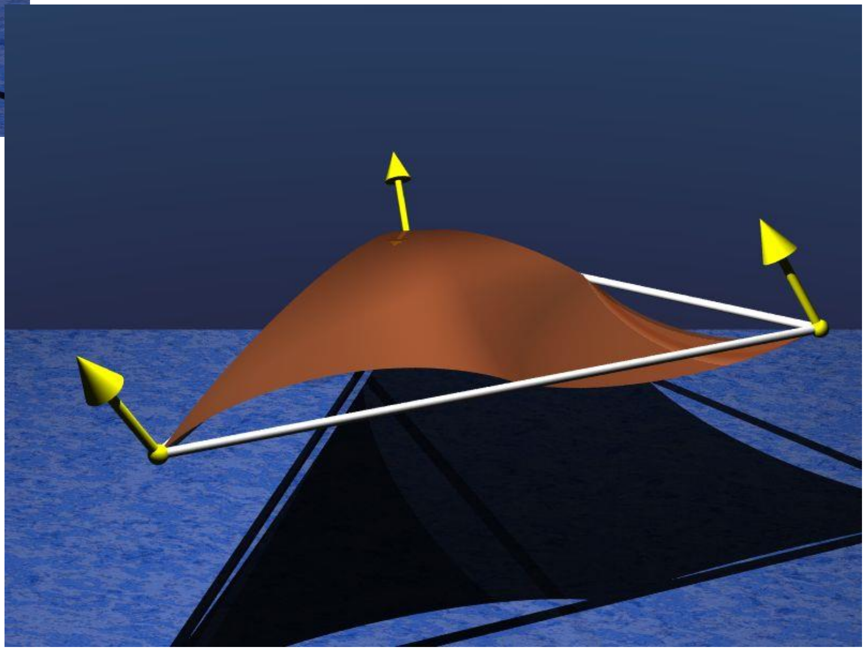
- **Zylindermantel** um z-Achse
  $x^2 + y^2 - r^2 = 0$
- **Kegel** (um die z-Achse)
  $x^2/r^2 + y^2/r^2 - z^2/d^2 = 0$

- **Einschaliges Hyperboloid**
  $x^2/a^2 + y^2/b^2 - z^2/c^2 - 1 = 0$



Quadrikensplines





gegeben

Punkte mit Tangentialebenen

Triangulation der Punkte

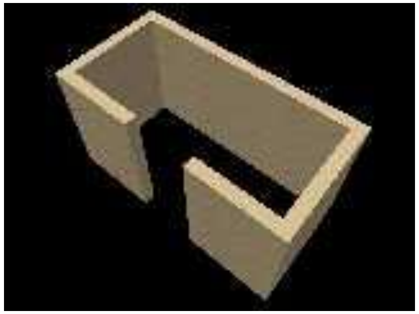
gesucht

C^1-Interpolant aus Quadriken

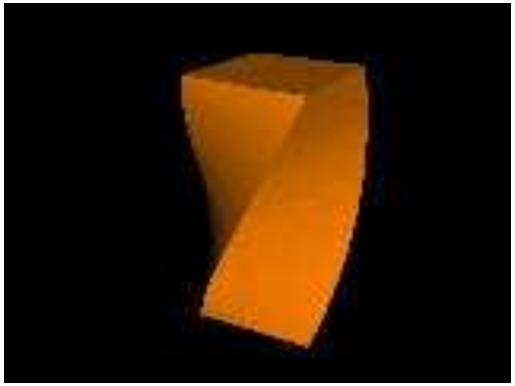
Extrusions- und Schiebekörper

- entstehen durch die Bewegung einer Kurve, Fläche oder eines Körpers entlang einer vorgegebenen Raumkurve.
  Translationskörper

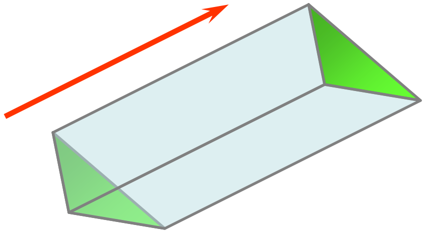
- Ein 2D-Flächenstück wird entlang eines Vektors / einer Kurve im Raum verschoben
  - stückweise gerade Kurve: Spine



Ein Kreis als Querschnittsfläche mit variierendem Radius

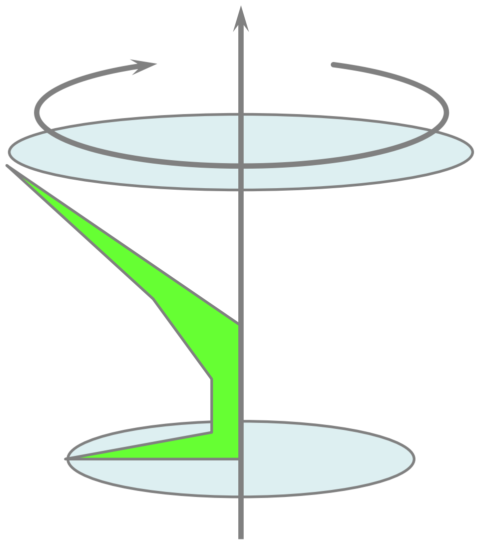


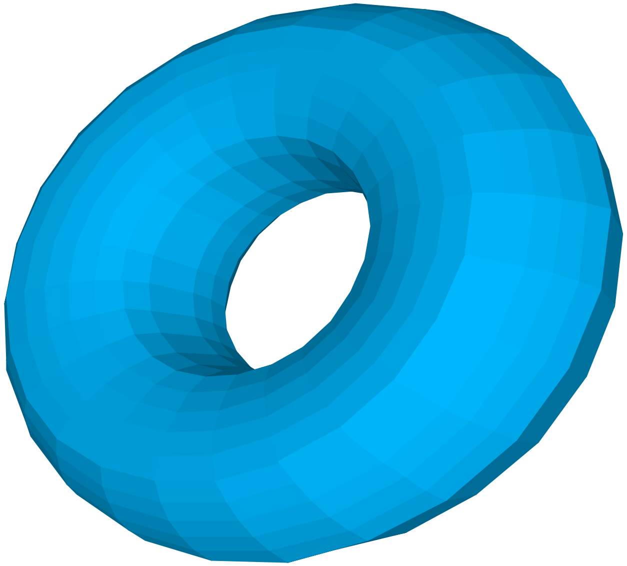
Querschnittsfläche wird während der Verschiebung gedreht.

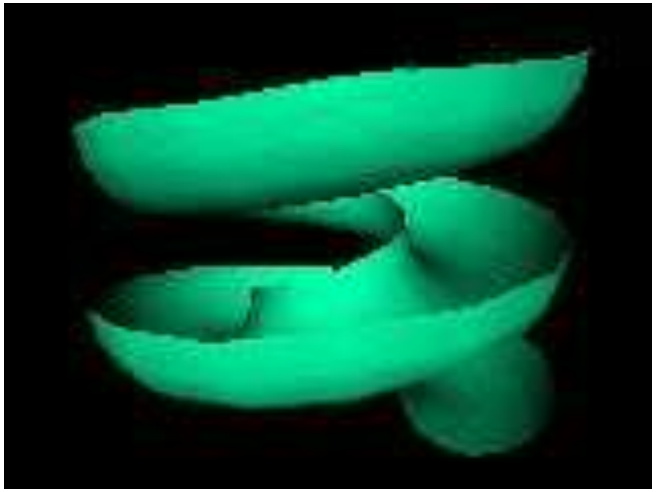


Rotationskörper

- Ein 2D-Flächenstück wird um eine Achse im Raum gedreht.
- Beispiel:

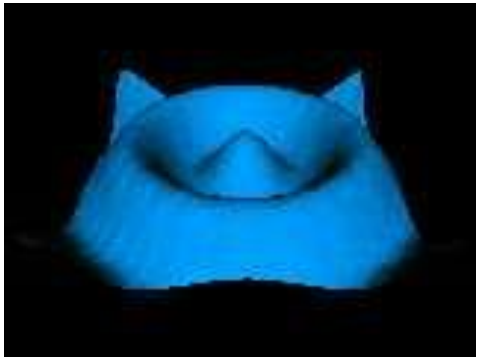


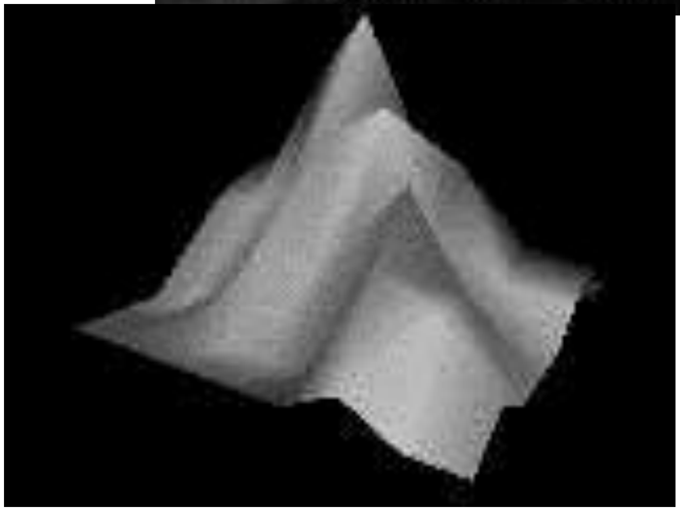




Elevation Grids

- In einem Raster der Größe nxm in der x-z-Ebene werden jedem Punkt ein **Höhenwert** sowie **Farb- und Texturwerte** zugewiesen.
- Geeignet zur Modellierung von Terrain, Landschaften, 2D-Funktionen, etc.





Volumenmodellierung

- Objekte werden repräsentiert als Menge von **Eckpunkten**, **Kanten**, **Seitenflächen** und/oder inneren **Volumenelementen**
- zusätzliche Information:
  - Wo befindet sich Material?
  - Wo ist innen, wo ist außen?
- Schnitt durch Volumenmodelle: Kantenpunkte, Seitenlinien und innere Schnittebenenstücke

Volumenmodelle

Erzeugung des Volumenmodells

akkumulativ

räumliche Unterteilung
regulär (Voxel)
hierarchisch (Octrees)

Hauptsächlich wissenschaftlich-medizinische Visualisierungs-anwendungen (Scientific Visualization)

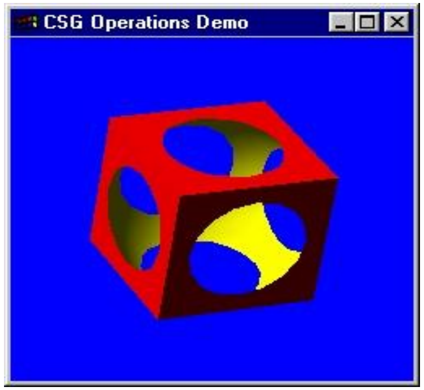
generativ

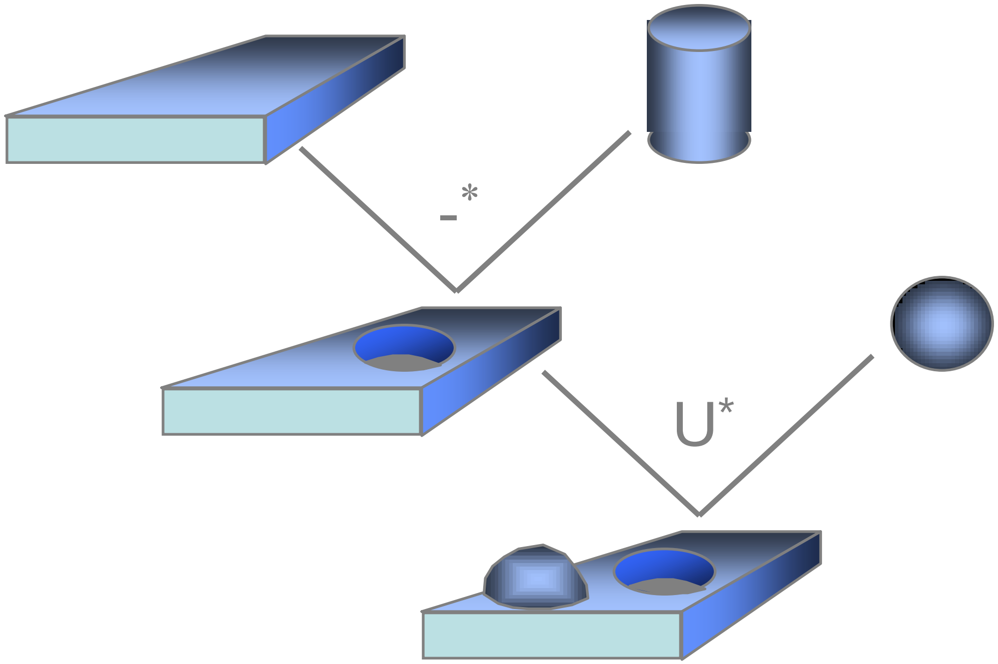
mengentheoretisches Modellieren
CSG = Constructive Solid Geometry

Objekte aus begrenzter Menge von Grundobjekten werden mit booleschen Operatoren verknüpft

Constructive Solid Geometry

- Verknüpfung von Grundobjekten mit booleschen Operatoren
- typische Grundobjekte:
- Quader, Zylinder, Kegel, Kugel, Torus



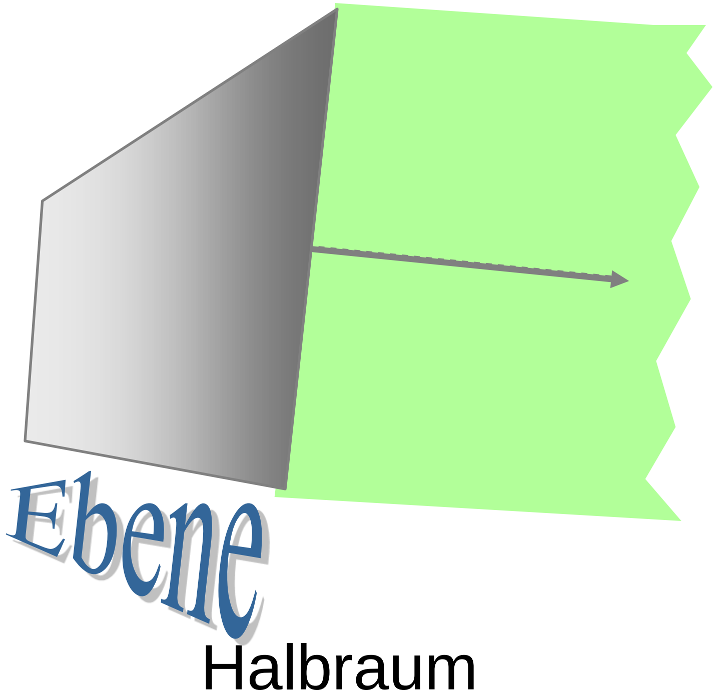


Halbräume

Halbraum allgemein: $\mathbf{ax + by + cz + d > 0}$

Kurzform: **[a b c d]**

[	0	0	1	-1	]
[	-1	0	0	-1	]
[	0	0	-1	-1	]
[	1	0	0	-1	]
[	0	-1	0	-1	]
[	0	1	0	-1	]

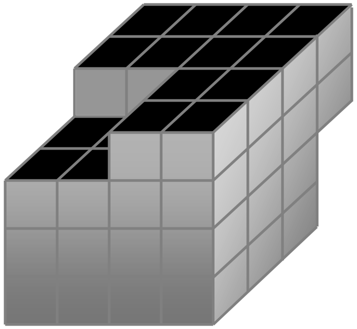


Voxeltechnik

- Ein Objekt wird durch Auflistung von Raumzellen (Volume element = **voxel**) beschrieben
  (auch: Raumumterteilungstechnik)

- einfache (auch parallelisierbare) Realisierung
  von Mengenoperationen
  (Vereinigung, Schnitt, Subtraktion, Enthaltensein)

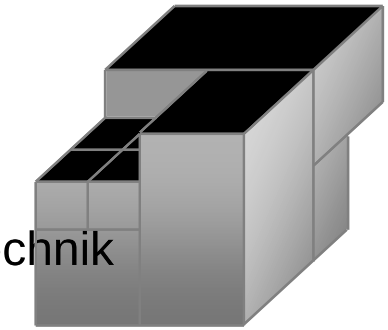
- Approximative Darstellung, die $O(n^3)$ Voxel erfordert



Octrees

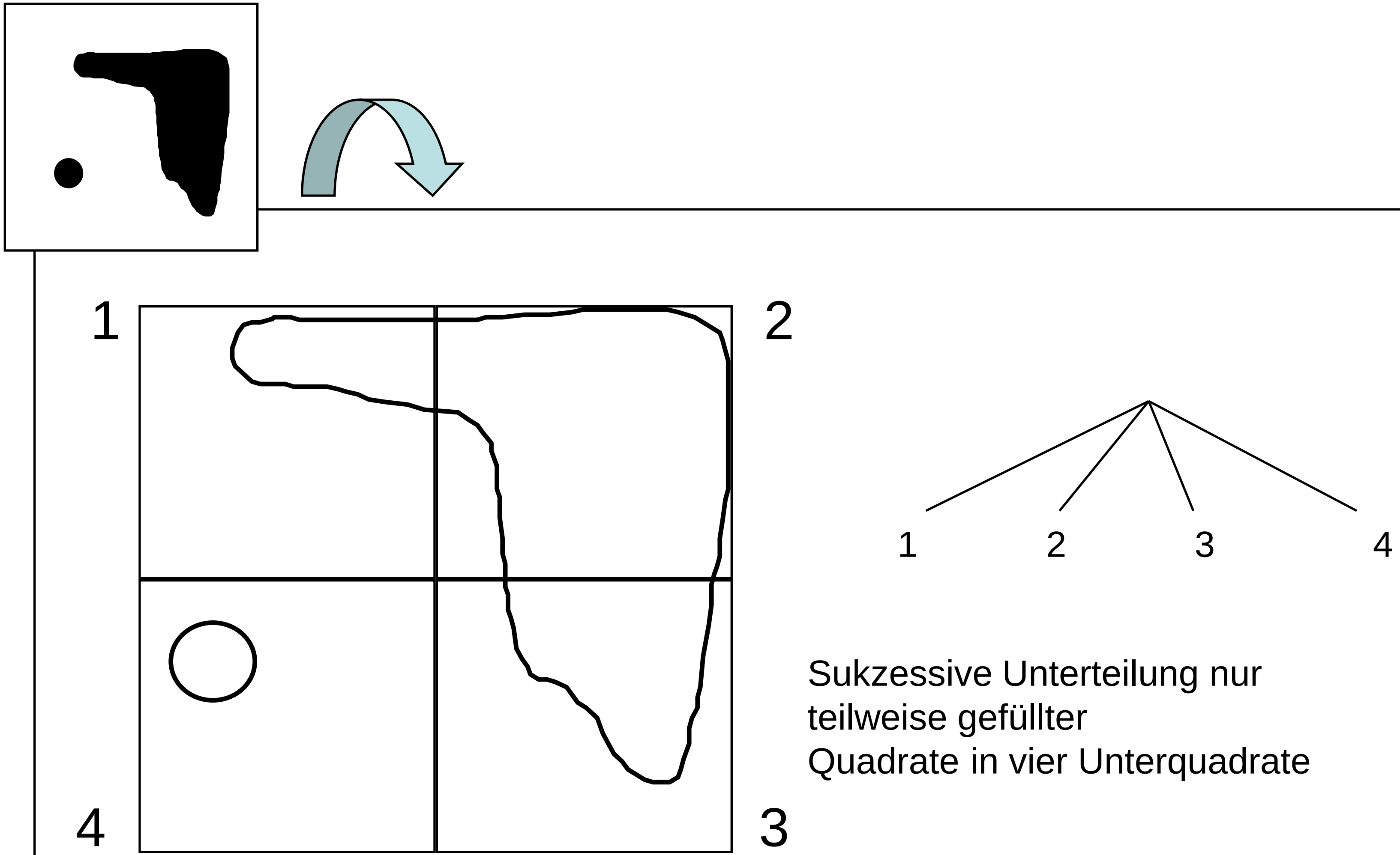
Octree: Hierarchische Variante der Raumunterteilungstechnik.

- einfache (auch parallelisierbare) Realisierung von Mengenoperationen (Vereinigung, Schnitt,
  Subtraktion, Enthaltensein)
- **geringerer Speicherbedarf** als bei regulärer Voxeltechnik



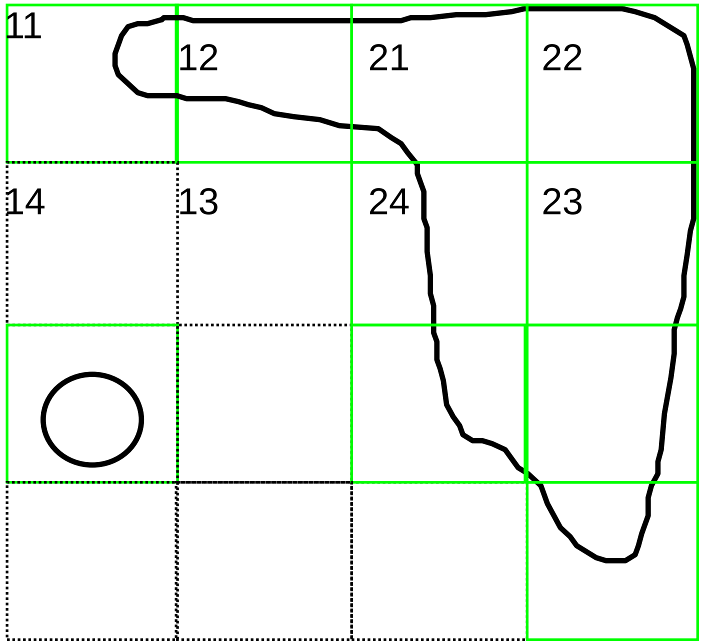
- Approximative Darstellung
- Zugriff auf bestimmte Zellen algorithmisch komplexer als bei Voxeltechnik

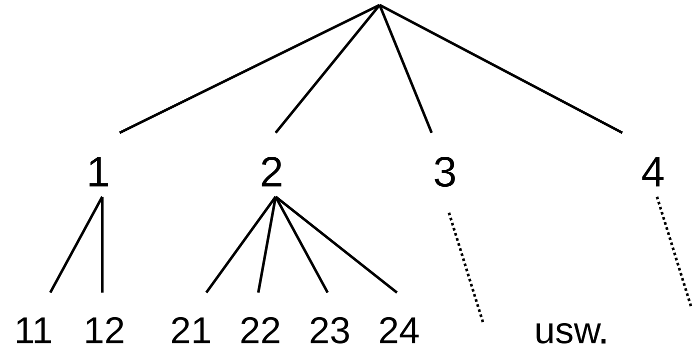
Quadtrees I

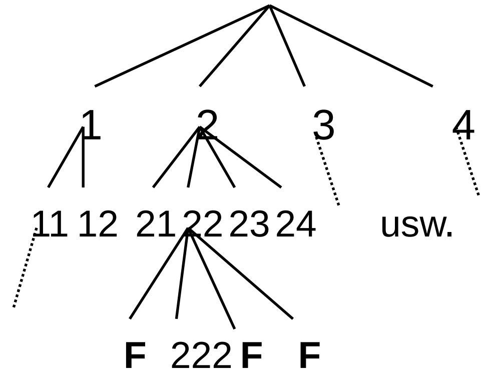


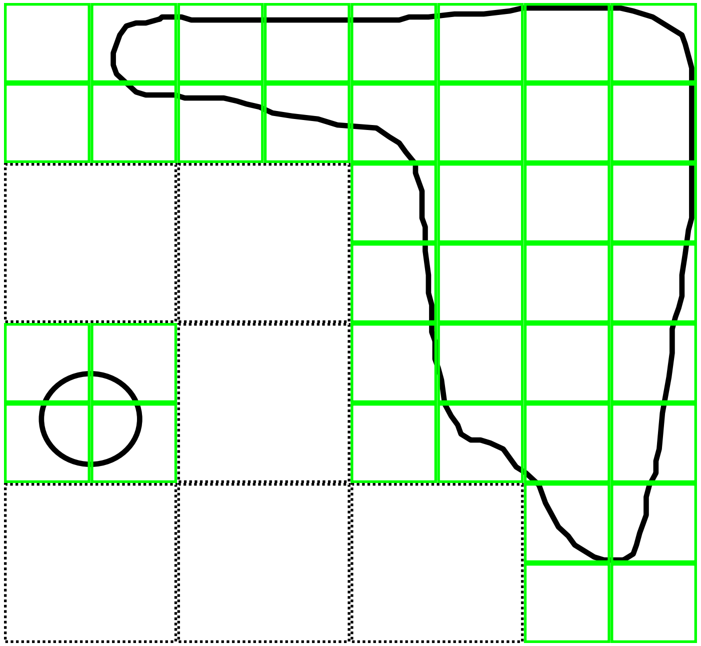
Sukzessive Unterteilung nur
teilweise gefüllter
Quadrate in vier Unterquadrate

Quadtrees II

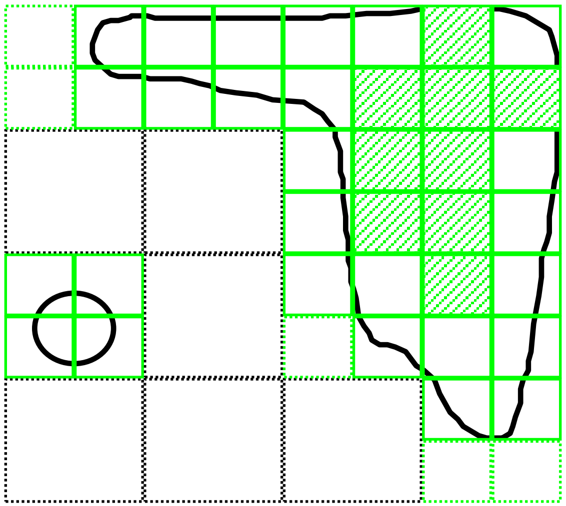


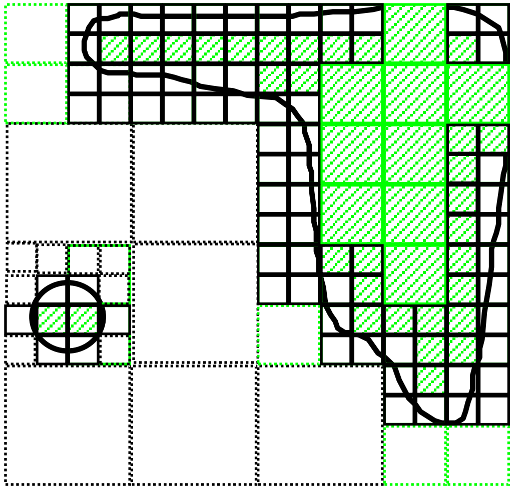


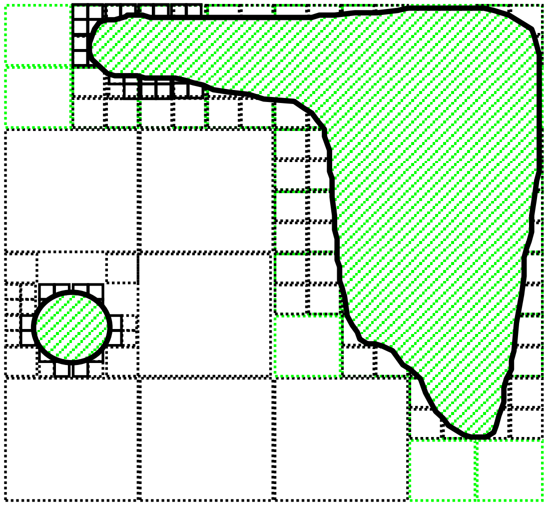




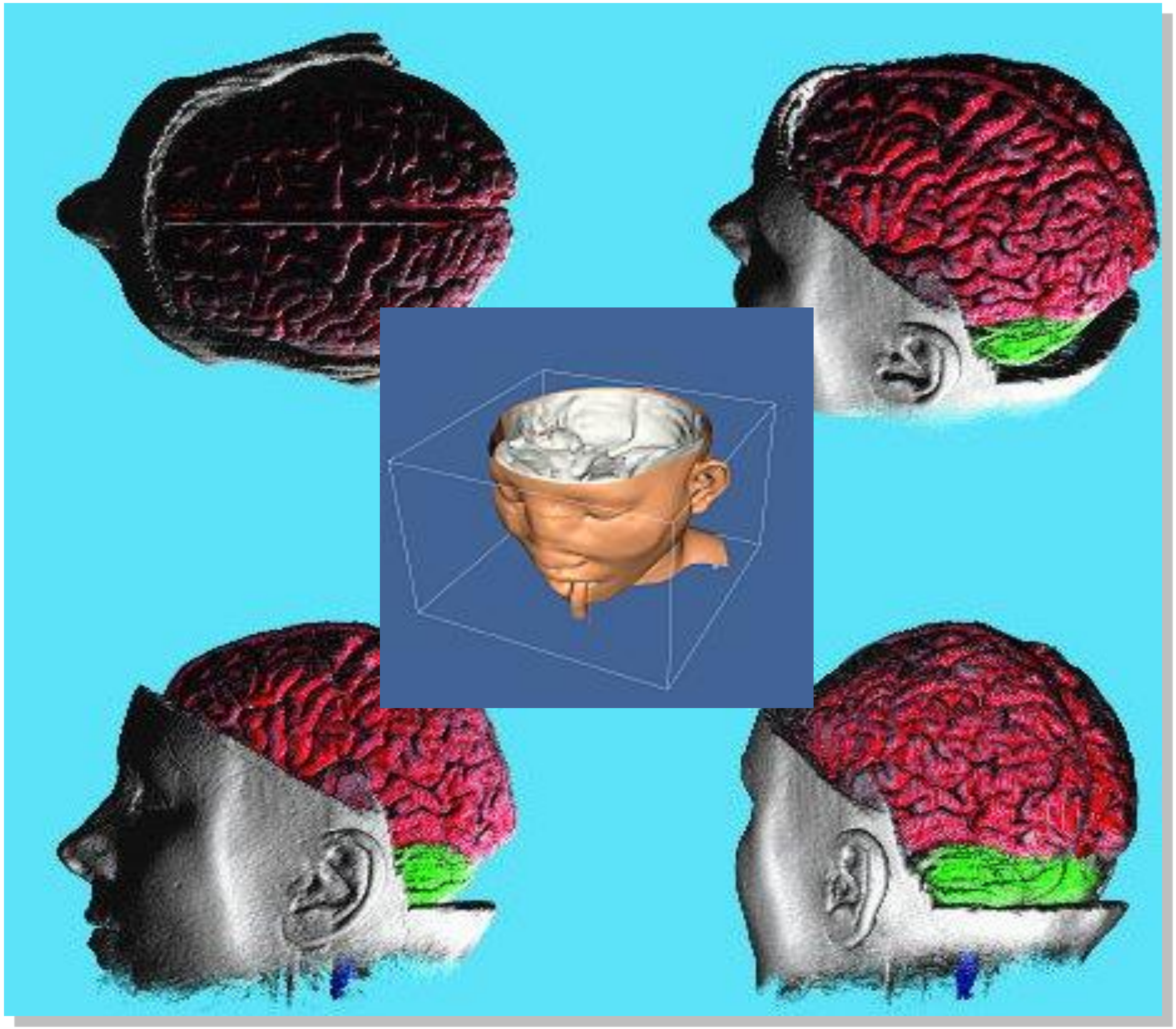
Quadtrees III



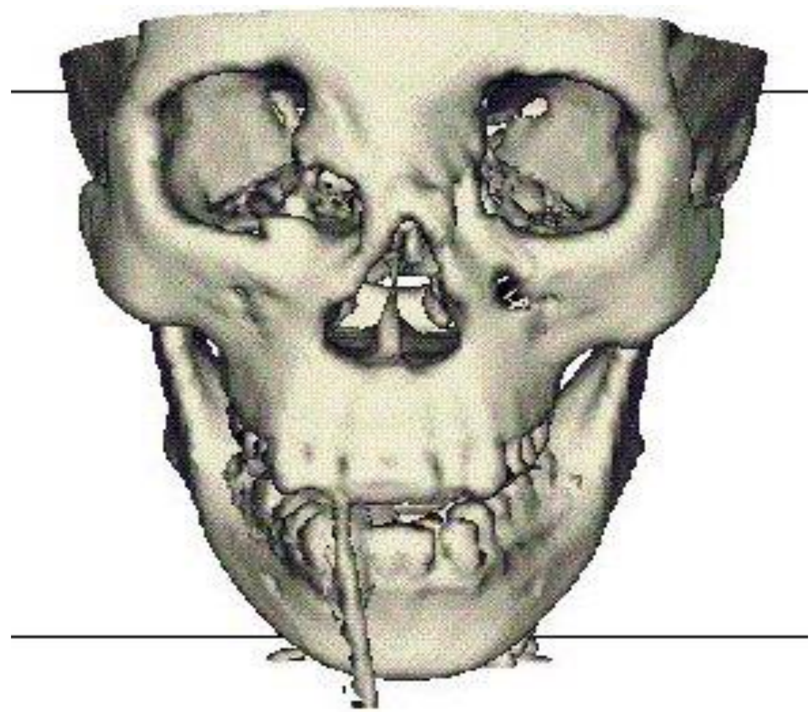




Volumenmodellierung



häufigstes Anwendungsfeld der Volumenmodellierung: Medizin



Vergleich der Modell-Repräsentationen

Genauigkeit
(Voxel, Octree, Polygonmesh: abhängig von Auflösung, CSG: exakt)

Flexibilität
(Raumraster, Freiformen > Extrusion > CSG)

Eindeutigkeit
(Octrees eindeutig, andere nicht)

Validierbarkeit
(Raumraster > CSG > Polygonmeshes)

Abgeschlossenheit
(CSG: ja, Polygonmesh: mit Aufwand)

Kompaktheit
(CSG > Polygonmesh > Raumraster)

Effizienz
(anwendungsabhängig, meist gilt Polygonmesh, Raumraster > CSG)

Andere Modellierungstechniken

- Modelle, die sich durch “externe Ereignisse” gesteuert selbst modifizieren können
  - **Fraktale**: Funktionen mit gebrochener Dimension, Selbstähnlichkeit, (Julia-Fatou, Mandelbrotmenge),
    IFS (Sierpinski-Dreieck), Midpoint-Displacement-Algorithmus
  - **Soft Objects:** Beschreibung über Skalarfelder, Äquipotentialflächen, etc. (Blobs)

    Recherchieren Sie dies !

- **Partikelsysteme:**
  Teilchen mit Position, Größe (auch < Pixel), Geschwindigkeit, Beschleunigung, Farbe, Alter, Lebensdauer, Schwarmverhalten

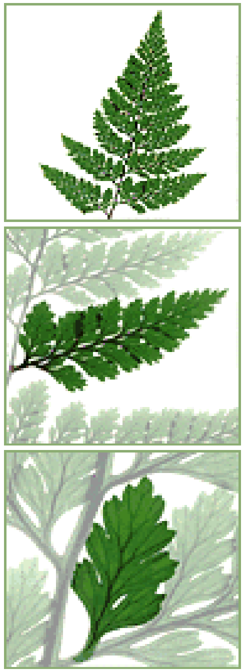
  Recherchieren Sie dies !

- **Wachstumsmodelle:**
  Grammatiken, Pflanzenwachstum, L-Systeme, String rewriting system

Pflanzen im Computer

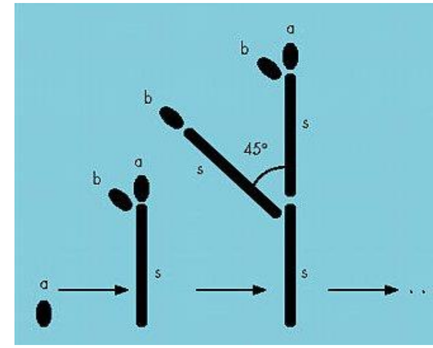
- Viele Pflanzen bestehen aus ähnlichen (nicht gleichen) Strukturen auf mehreren Skalen

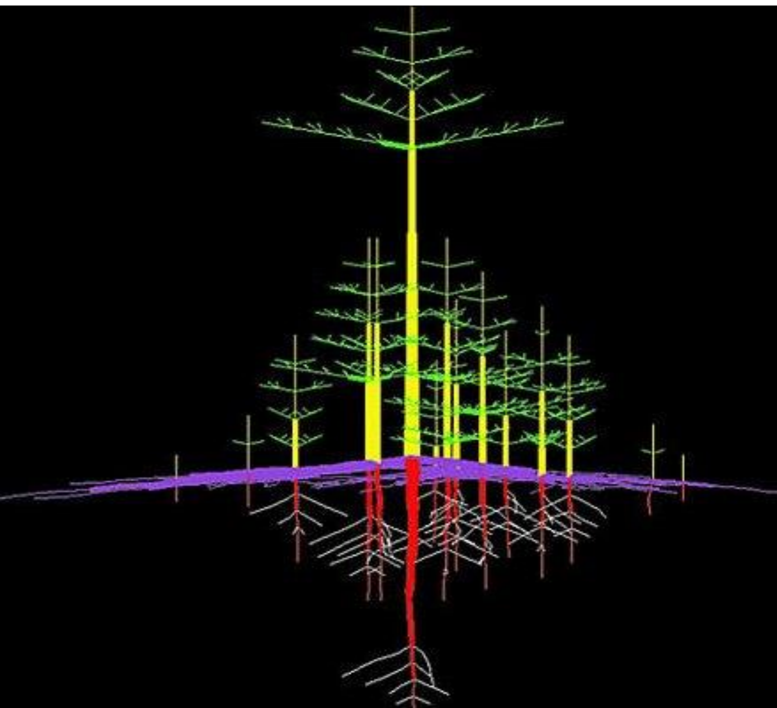
  - Pflanzen sind also durch Wiederholung einfacher Ersetzungsalgorithmen modellierbar (Lindenmayer 1967).

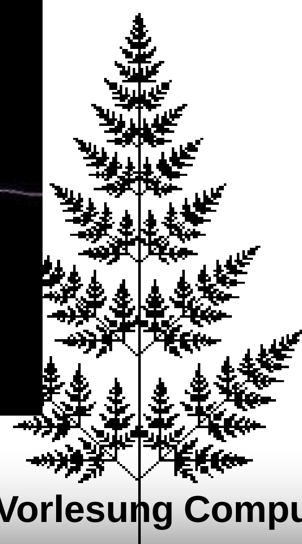


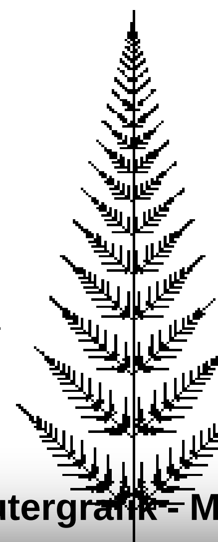
Lindenmayer-Systeme

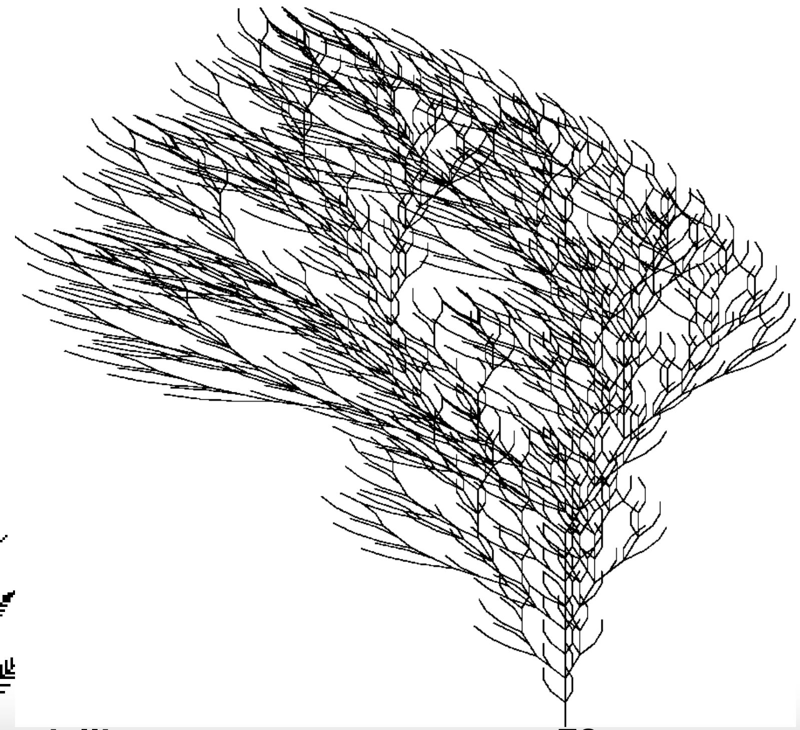
- Lindenmayer-Systeme bestehen aus einem Alphabet, einem Anfangs-“Wort“ und einem Ersetzungs-Algorithmus.
- Grafische Interpretation durch Hogeweg und Hesper 1974



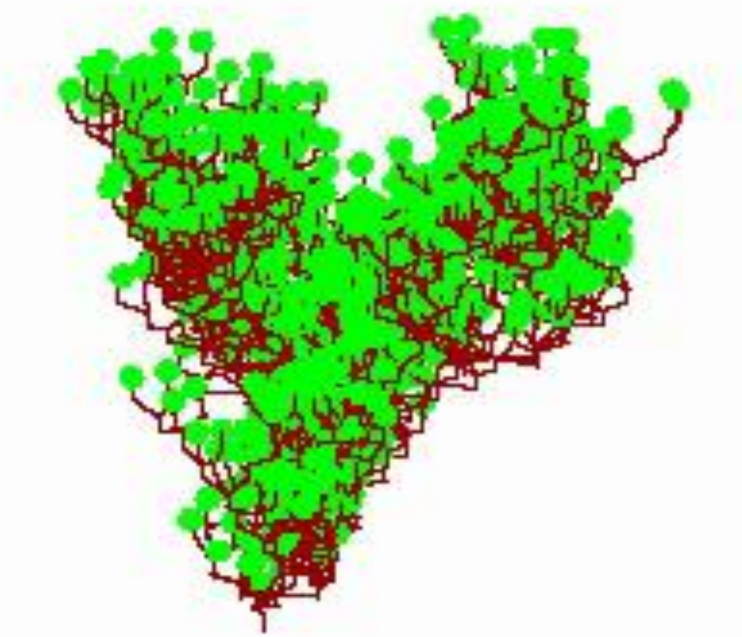








Resultate



Funktionale Ergebnisse
durch Wachstumssimulation:
Phototrophismus,
Wasserhaushalt,
Alterungsprozesse, etc.

Bäume sind *Fraktale*, um eine in allen Richtungen gleiche Blattflächenprojektion mit möglichst geringem Materialaufwand zu erzielen.

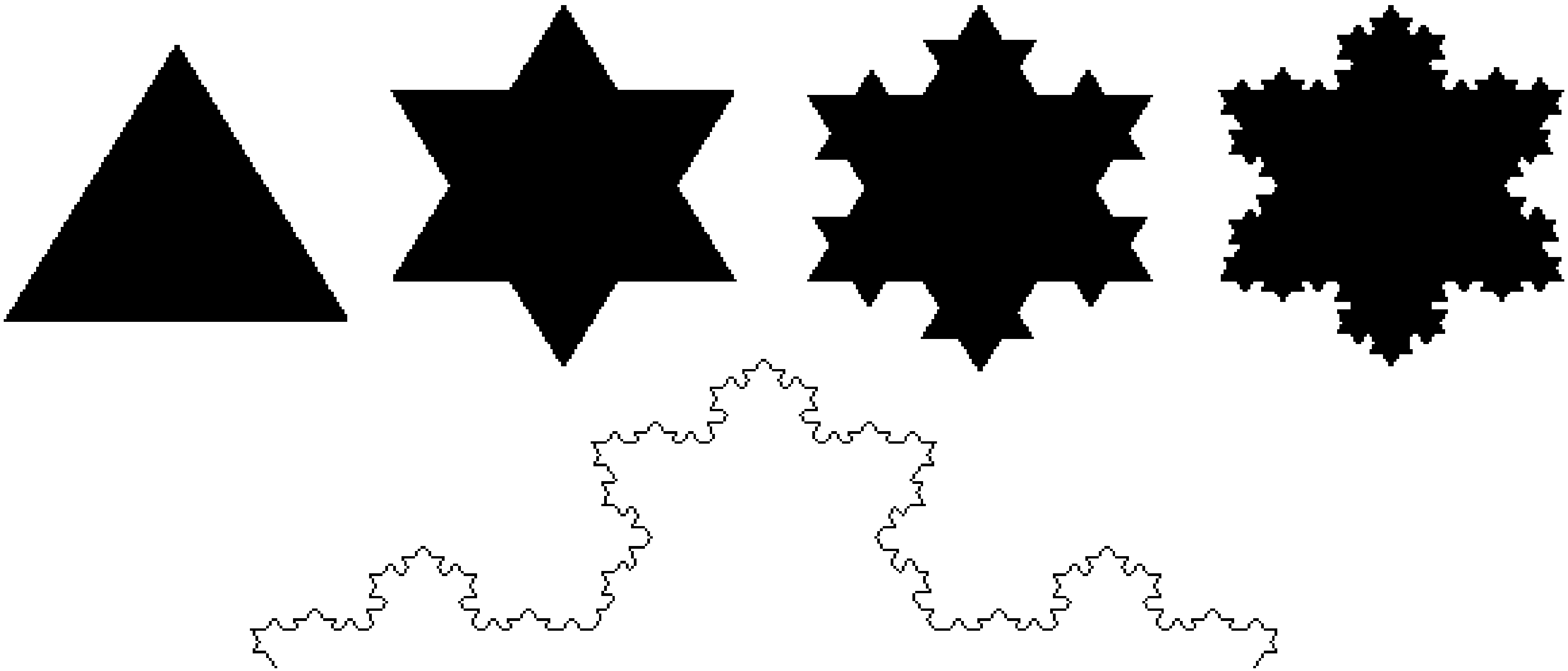
Eigenschaften Prozeduraler Modellierung

Vor- und Nachteile prozeduraler Modellierung:

- **platzsparend** beim Entwurf und in der geometrischen Repräsentation
- adaptiv durch beliebige **Parametrisierbarkeit** der generierenden Prozeduren

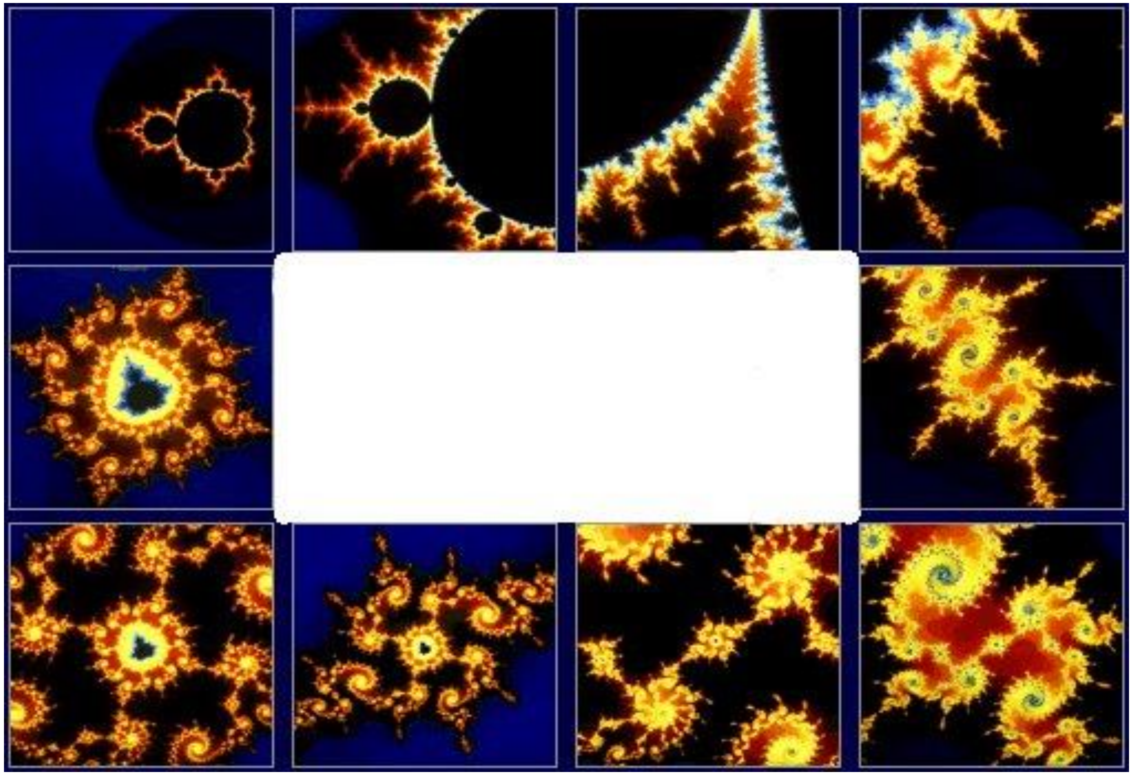
- u.U. höhere **Laufzeit**
- U.U. höherer **Implementierungsaufwand**

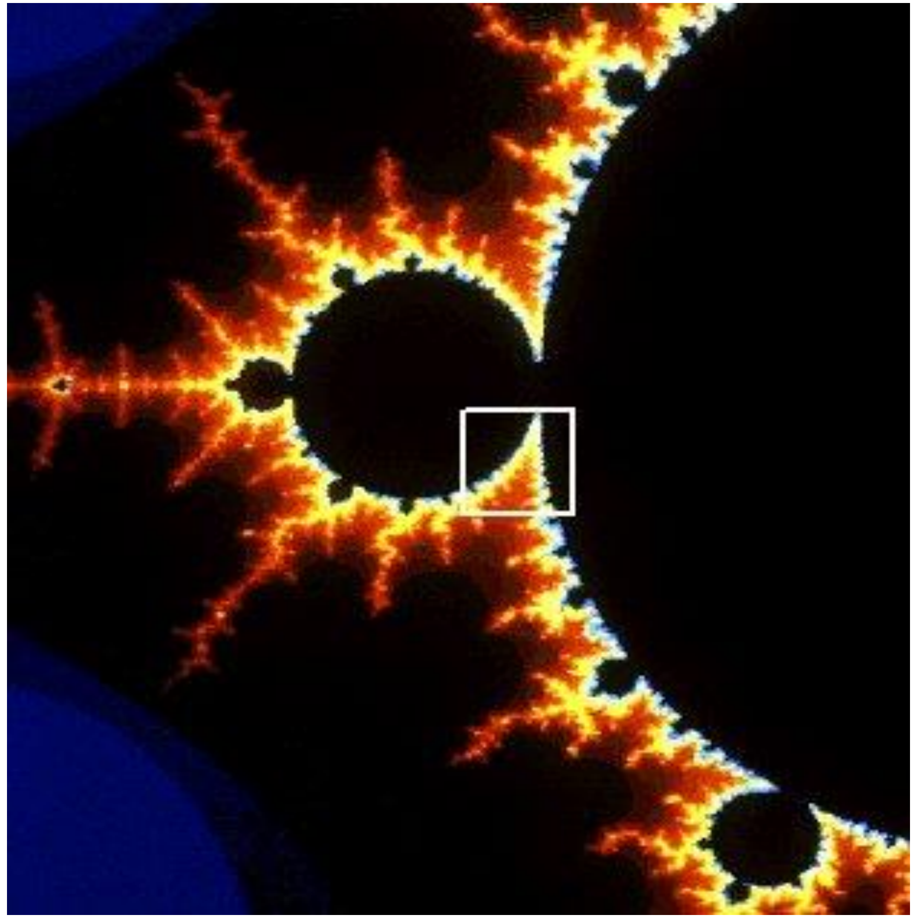
Fraktale



Schneeflockenkurve

Fraktale





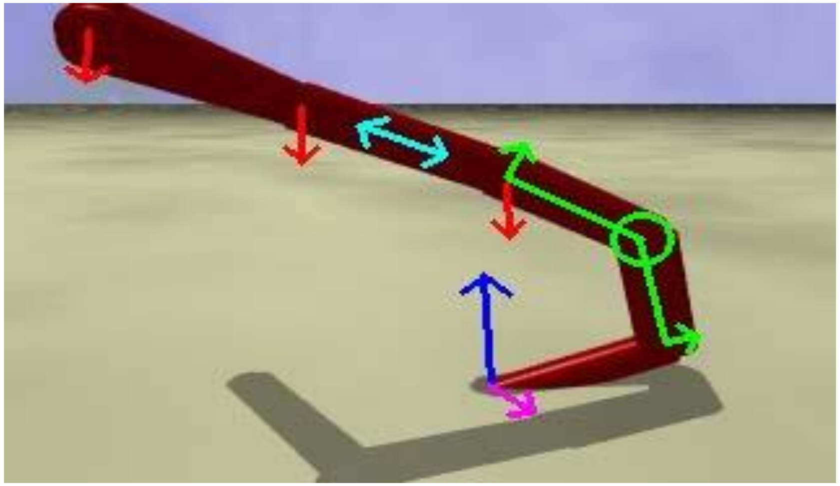
Schreiben Sie ein Programm zur Berechnung von Apfelmännchen !
Challlenge: nur auf der GPU ...

Artificial Life

Wie bewegen sich einfache Lebenwesen ?
Wie reagieren sie auf Reize – Licht, Schall, Nahrung ?
Wie interagieren Lebenwesen mit einfachen Überlebensstrategien ?

Frühe wichtige Beispiele:
- Conways „Game of Life“
- Braitenbergsche Vehikel

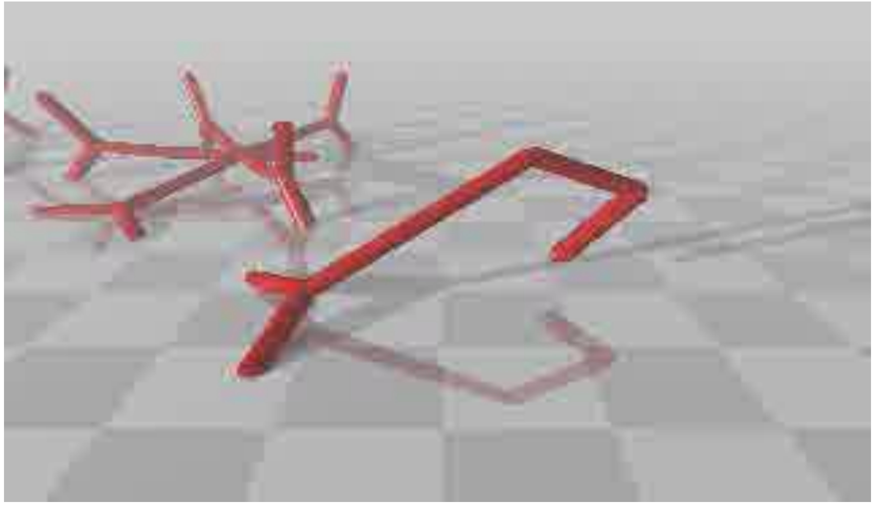
Recherchieren Sie dies !

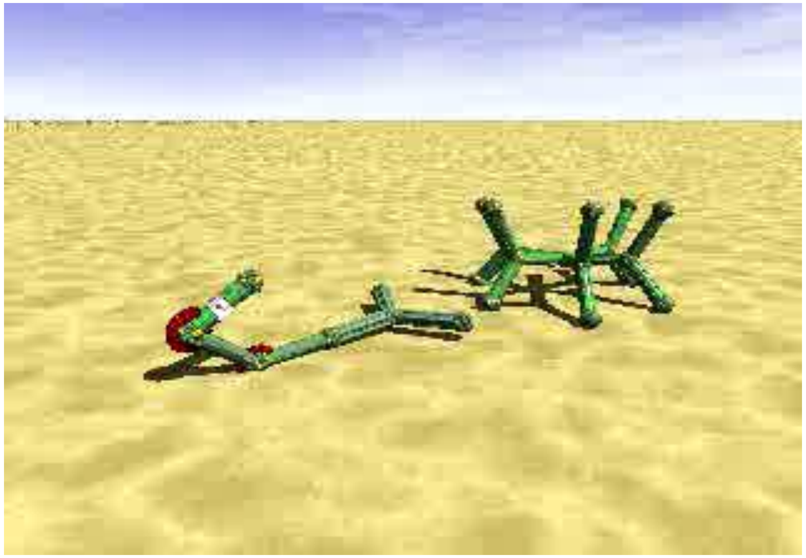


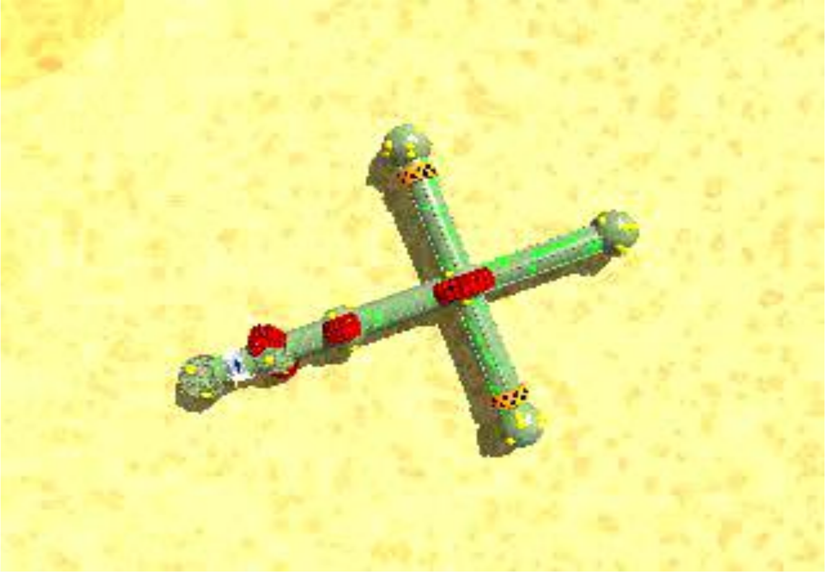
Framsticks von Maciej Komosinski (1999)
„Lebewesen“ aus gelenkig miteinander verbundenen Stäben

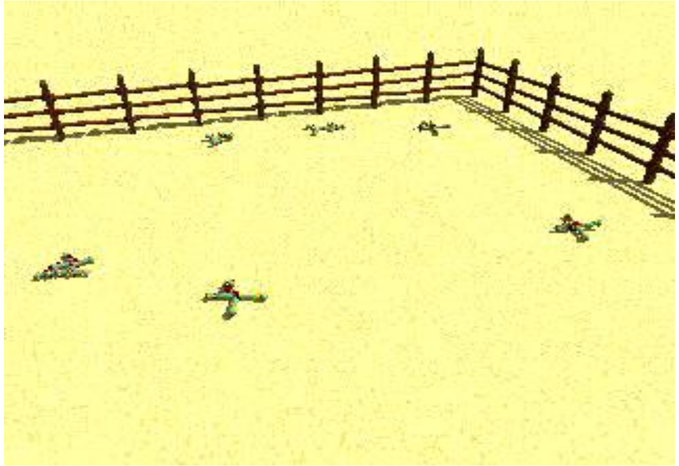
Rot: Gravitation
Magenta: Reibung
Grün: Muskelkräfte
Cyan: Dämpfung
Blau: Boden-Reaktion

Resultate







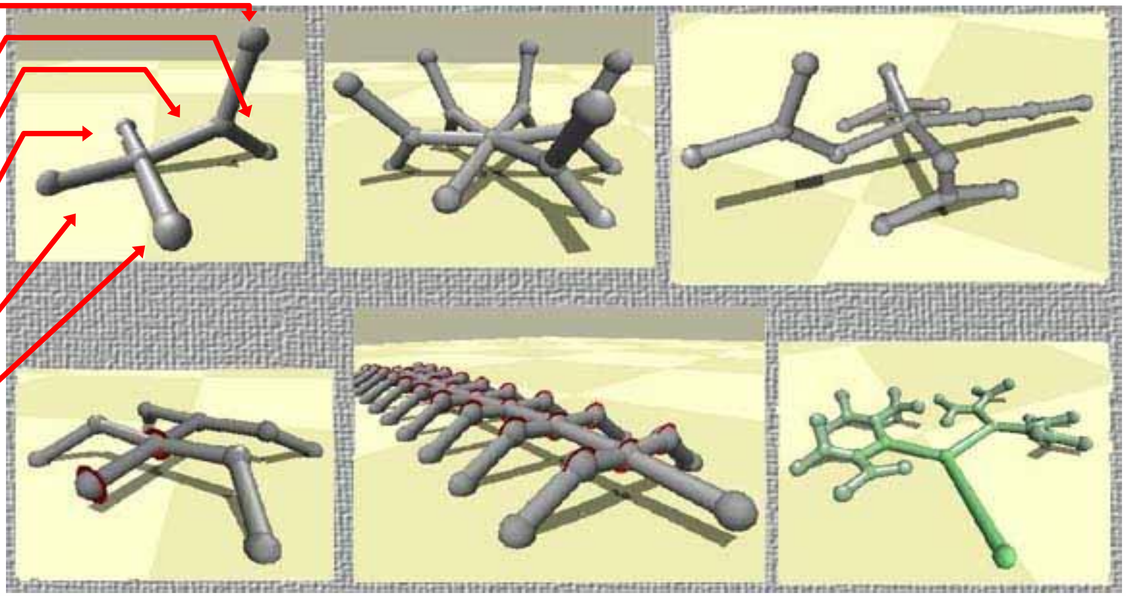


Genetik und Kybernetik der Framsticks

Framsticks haben einfache „Neuronen“, die mit verschiedenen Aktoren und Sensoren zusammenwirken:
•Gelenke haben Muskeln
•G-Sensor für die Orientierung in Bezug auf die Senkrechte
•T-Sensor für die Mitteilung einer Berührung
•S-Sensor für den „Geruch“ nach Energie

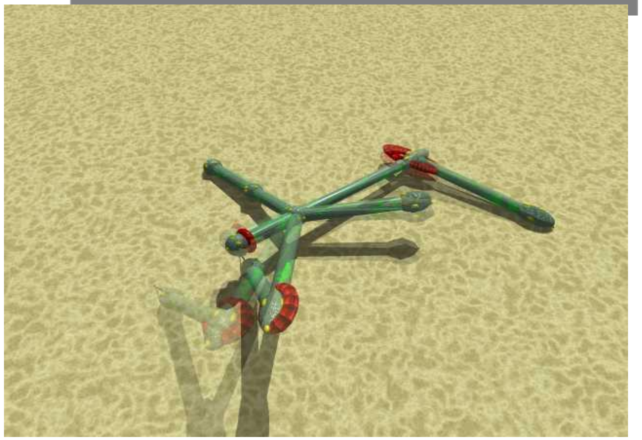
Spezifikation der Framsticks:
X = Stick, () = Gelenk
Modifikatoren, z.B.
R = Rotation 45 Grad

X(X,X,RRX(X,X))



Künstliche Evolution mit Framsticks

Virtuelle Genetik, erlaubt Evolution...



z.B. Zielfunktion: maximale horizontale Geschwindigkeit

Genotyp
(,m((X),LLMXRlc(,,,lMMMMMMqqqXLLLLLLLLQQX
[@0:1.357,=:0.728][0:4.301]
[|1:-0.321,1:478.220,/-0.701,-1:-1.739,1:2.454][@G:876.564))))

Das wars zur Modellierung !