

## Teil 1: Modellierung

### Objekte und ihre Beschreibung

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Einleitung

### Computergraphik: 3D sehr wichtig

- ◆ photo-realistic rendering
- ◆ Computer-Animation, Modellierung
- ◆ Visualisierung, Virtual Reality

#### Ansatz:

- ◆ per **rendering** wird eine **3D-Szene** dargestellt

#### Frage: Beschreibung der Szene?

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## 3D Szene – Inhalt

### Virtuelle Welt:

- ◆ Objekte als Abbild wirklicher Objekte: Architektur, Objekte des tägl. Lebens, ...
- ◆ Oberflächen: eben, uneben, fraktal
- ◆ Volumensobjekte: innere Struktur
- ◆ real-3D objects: Fraktale wie Wolken, ...

#### Aber wie beschreiben?

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## 3D Szene – Beschreibung

### Modellierung:

- ◆ Virtuelle 3D Welt – soll dargestellt werden
- ◆ Szene-Beschreibung (Graph-Form)
- ◆ Teile:
  - ◆ Objekte – 0D, 1D, 2D, 3D
  - ◆ Objekt-Attribute
  - ◆ Transformationen
  - ◆ Beleuchtung
  - ◆ Kamera

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## In der Szene: Objekte

### Einfache Objekte

- ◆ **Primitiva:** Punkt (0D), Liniensegment (1D), Bézier-Kurve (1D), Dreieck (2D), Patch (2D), Voxel (3D), ...

### Komplexe Objekte: z.B.:

- ◆ **Sammel-Objekte:** Poly-Linie (1D), Mesh (2D), Volumensdatensatz (3D), ...
- ◆ **Transformierte Objekte:** sweeps, ...
- ◆ **Kombinierte Objekte:** CSG, ...

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Objekte – Überblick

**Basis:** Punkte, Liniensegmente, etc.

**Diskrete Approximation:** meshes

**Erweiterung:** terrains, fraktale Gebirge

**Modellierung durch sweeps**

**Modellierung durch soft objects**

**Modellierung:** Partikelsysteme

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Basis: Punkte im 3D

### Start im 3D: Punkt:

- ◆ Darstellung: vektoriell  $\rightarrow \mathbf{x} = (x_1, x_2, x_3)^T$

### Unterschied Punkt $\leftrightarrow$ Vektor

- ◆ Vektor = translationsinvariant
- ◆ Vektor = (End-)Punkt  $\mathbf{q}$  – (Start-)Punkt  $\mathbf{p}$   
 $= \mathbf{q} - \mathbf{p} = (q_1 - p_1, q_2 - p_2, q_3 - p_3)^T$   
 $= \mathbf{v} = (v_1, v_2, v_3)^T$
- ◆ Liniensegment: durch  $\mathbf{p}$  und  $\mathbf{q}$   
oder  $\mathbf{p}$  und  $\mathbf{v}$  gegeben

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## WH (CGR3): Meshes

### Basis:

- ◆ Dreieck, Viereck
- ◆ Infos: Eckpunkte, Normale(n), Nachbarn

### Sehr oft in Verwendung: Meshes:

- ◆ Liste von Dreiecken bzw.  $\Delta$ -strips
- ◆ BRep (WH): Punkte  $\rightarrow$  Kanten  $\rightarrow$  Flächen
- ◆ Winged-Edge DS (WH): Kante++

### Aspekte: Speicherplatz, Berechnungen

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Datenstruktur: $\Delta$ -strip

13 Punkte:  
11 Dreiecke

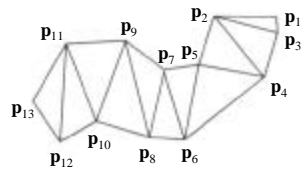


Figure 10-6  
A triangle strip formed with  
11 triangles connecting 13  
vertices.

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

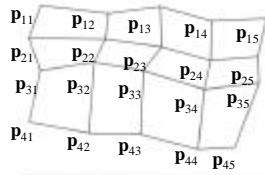
---

---

---

## Quadrilateral Mesh

**5\*4 Punkte:  
12 Vierecke!**



*Figure 10-7*  
A quadrilateral mesh containing 12 quadrilaterals constructed from a 5 by 4 input vertex array.

Helwig Hauer

Teil 1: Modellierung




---

---

---

---

---

---

---

---

---

---

## BRep: Beispiel

**Mesh:**

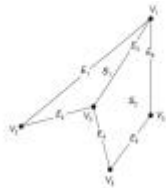
- ◆ Dreieck  $S_1$
- ◆ Viereck  $S_2$

**6 Kanten:**

- ◆ 1 gemeinsam

**5 Eckpunkte**

- ◆ 2 gemeinsam



VERTER TABLE	EDGE TABLE	POLYGON SURFACE TABLE
$V_1 = p_1, p_2, p_3$	$E_1 = V_1, V_2$	$S_1 = E_1, E_2, E_3$
$V_2 = p_2, p_3, p_4$	$E_2 = V_2, V_3$	$S_2 = E_1, E_3, E_4, E_5$
$V_3 = p_3, p_4, p_5$	$E_3 = V_3, V_4$	
$V_4 = p_4, p_5, p_6$	$E_4 = V_4, V_5$	
$V_5 = p_5, p_6, p_1$	$E_5 = V_5, V_1$	

Helwig Hauer

Teil 1: Modellierung




---

---

---

---

---

---

---

---

---

---

## Objekte – Überblick

**Basis: Punkte, Liniensegmente, etc.**

**Diskrete Approximation: meshes**

**Erweiterung: terrains, fraktale Gebirge**

**Modellierung durch sweeps**

**Modellierung durch soft objects**

**Modellierung: Partikelsysteme**

Helwig Hauer

Teil 1: Modellierung




---

---

---

---

---

---

---

---

---

---

## Terrains

### Definition:

- ◆ Basis-Gitter (2D),  
pro Gitter-Punkt: 1 Höhenwert

### Mögliche Erweiterung:

- ◆ Farbinformationen
- ◆ Texturen

Hilwig Hausar

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## VRML-Beispiel: terrain (1)

```
geometry ElevationGrid {  
  xDimension 5  
  zDimension 5  
  xSpacing 2  
  zSpacing 2  
  height [ 2, 2, 3, 2, 2,  
          1, 1, 2, 1, 1,  
          1, 1, 2, 1, 1,  
          2, 2, 3, 2, 2,  
          2, 2, 3, 2, 2 ]  
}
```

Hilwig Hausar

Teil 1: Modellierung



---

---

---

---

---

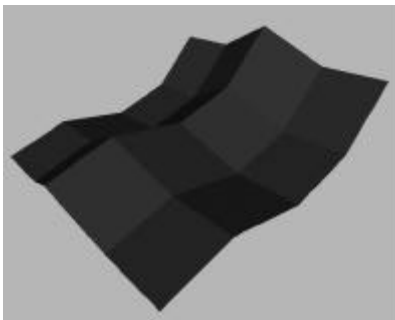
---

---

---

## VRML-Beispiel: terrain (2)

5\*5  
Terrain



Hilwig Hausar

Teil 1: Modellierung



---

---

---

---

---

---

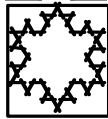
---

---

## Fraktale Gebirge (1)

### 1D-Beispiel: Koch-Kurve

- ◆ Initiator: Startpolygon
- ◆ Generator: Ersetzungsregel



### Fraktales Gebirge:

- ◆ Generator + Zufallszahlen
- ◆ Initiator: 1 Dreieck oder 2
- ◆ Generator: jede Kante i. d. Mitte teilen, Mittelpunkt per Zufallszahl verschieben

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

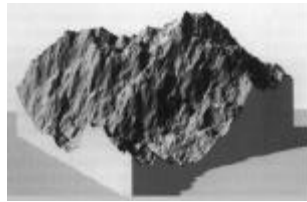
---

---

## Fraktales Gebirge (2)

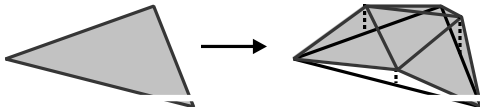
### Pro Schritt:

- ◆ 1 Dreieck → 4 Dreiecke



### Stopp wenn:

- ◆ Unterteilung fein genug



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

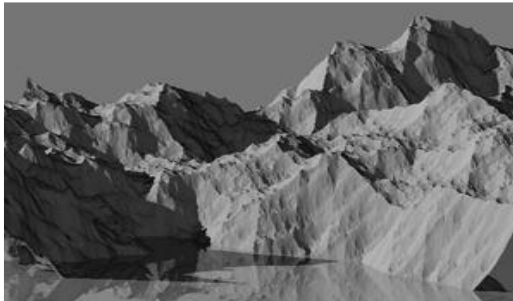
---

---

---

---

## Beispiel: Fraktales Gebirge



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Objekte – Überblick

**Basis:** Punkte, Liniensegmente, etc.  
**Diskrete Approximation:** meshes  
**Erweiterung:** terrains, fraktale Gebirge  
**Modellierung durch sweeps**  
**Modellierung durch soft objects**  
**Modellierung:** Partikelsysteme

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Sweeps

### Idee:

- ◆ 2D Kontur + Transformation

### Formen:

- ◆ translational sweep
- ◆ rotational sweep
- ◆ conical sweep
- ◆ spherical sweep
- ◆ general cylinder

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Translational Sweep

### Definition:

- ◆ 2D-Kontur + Translation entlang Pfad



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

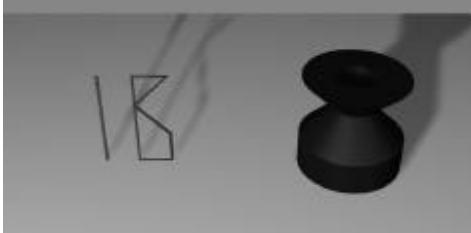
---

---

## Rotational Sweep

**Definition:**

- ◆ 2D-Kontur + Rotation um Achse



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

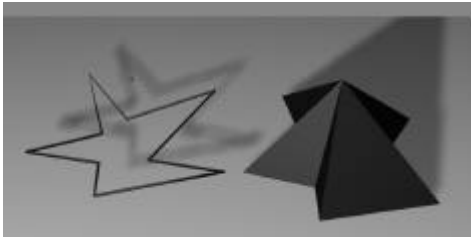
---

---

## Conical Sweep

**Definition:**

- ◆ 2D-Kontur + Verjüngung zu Punkt hin



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

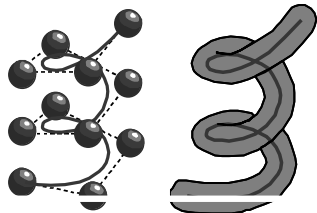
---

---

## Spherical Sweep

**Definition:**

- ◆ Kugel (o.Ä.) + sweep entlang Pfad



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

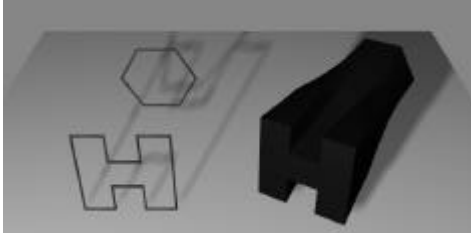
---



## General Cylinder

### Definition:

- ◆ 2 2D-Konturen + Verbindung dazwischen



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Sweeps – Repräsentation

Idee = nur Modellierung —

### Analytische Form:

- ◆ abhängig von Definition (evtl. nicht-trivial)
- ◆ abhängig von notwendigen Operationen

### Approximation:

- ◆ Mesh = tessellation = Zerteilung in kleine Flächenteile → Approximation durch  $\Delta$

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Objekte – Überblick

Basis: Punkte, Liniensegmente, etc.

Diskrete Approximation: meshes

Erweiterung: terrains, fraktale Gebirge

Modellierung durch sweeps

Modellierung durch soft objects

Modellierung: Partikelsysteme

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

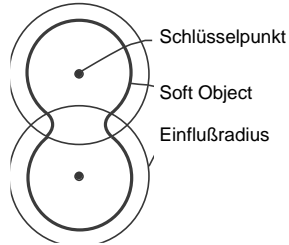
---

---

## Soft Objects (1)

### Definition:

- ◆ Schlüsselpunkte, Einflußfunktion sowie Iso-Wert → Iso-Fläche



Helmig Hausler

Teil 1: Modellierung




---

---

---

---

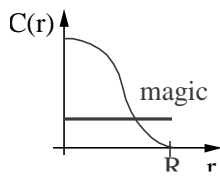
---

---

---

---

## Soft Objects (2)



Einfluß-Funktion fällt mit Entfernung  $r$   
 Iso-Wert **magic** definiert Objekt

Helmig Hausler

Teil 1: Modellierung




---

---

---

---

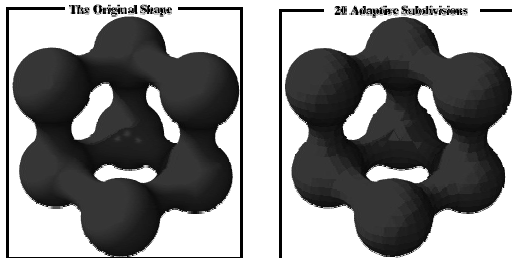
---

---

---

---

## Beispiel: Soft Objects



Repräsentation? Analytisch? Dreiecke?

Helmig Hausler

Teil 1: Modellierung




---

---

---

---

---

---

---

---

## Objekte – Überblick

**Basis:** Punkte, Liniensegmente, etc.  
**Diskrete Approximation:** meshes  
**Erweiterung:** terrains, fraktale Gebirge  
**Modellierung durch sweeps**  
**Modellierung durch soft objects**  
**Modellierung:** Partikelsysteme

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Partikelsysteme (1)

### Definition:

- ◆ Meist große Menge von Punkten
- ◆ charakteristisches Aussehen pro Partikel
- ◆ Verhalten von Partikel

### Anwendungen:

- ◆ Feuer, Rauch, etc.
- ◆ Partikel-sweeps: natürliche Objekte

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Partikelsysteme (2)

### Pro Partikel:

- ◆ Eigenschaften:
  - ◆ Geometrische Repräsentation (einfach!)
  - ◆ Farbe, Transparenz
- ◆ Bewegung:
  - ◆ Richtung, Geschwindigkeit
  - ◆ Lebensdauer

### Global:

- ◆ Anwendung von Zufallszahlen

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---

## Partikelsysteme: Ablauf

### Pseudo-Code:

- ◆ Wenn Lebensdauer abgelaufen: löschen,
- ◆ Sonst Partikeldaten auf neuen Stand.
- ◆ Evtl. neue Partikel erzeugen
- ◆ Alle aktuellen Partikel darstellen

### Auch hierarchisch möglich:

- ◆ Wald-System → Baum-System → Blätter

Helwig Hauser

Teil 1: Modellierung



---

---

---

---

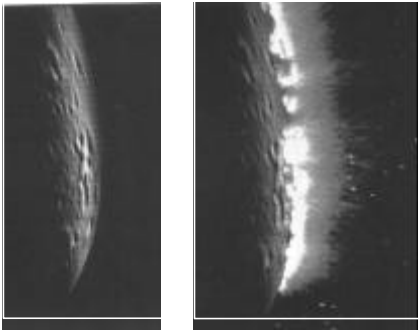
---

---

---

---

## Beispiel: Partikelsysteme



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

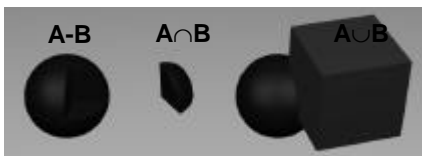
---

---

## WH (CGR3): CSG

### Per booleschen Operationen: Primitiva → komplexe Objekte:

- ◆ logisches UND<sub>2</sub>: Durchschnitt
- ◆ logisches ODER<sub>2</sub>: Vereinigung
- ◆ MINUS<sub>2</sub>: „Wegschneiden“



Helwig Hauser

Teil 1: Modellierung



---

---

---

---

---

---

---

---