

Umweltmodellierung

Prof. Dr.-Ing. Rüdiger Dillmann

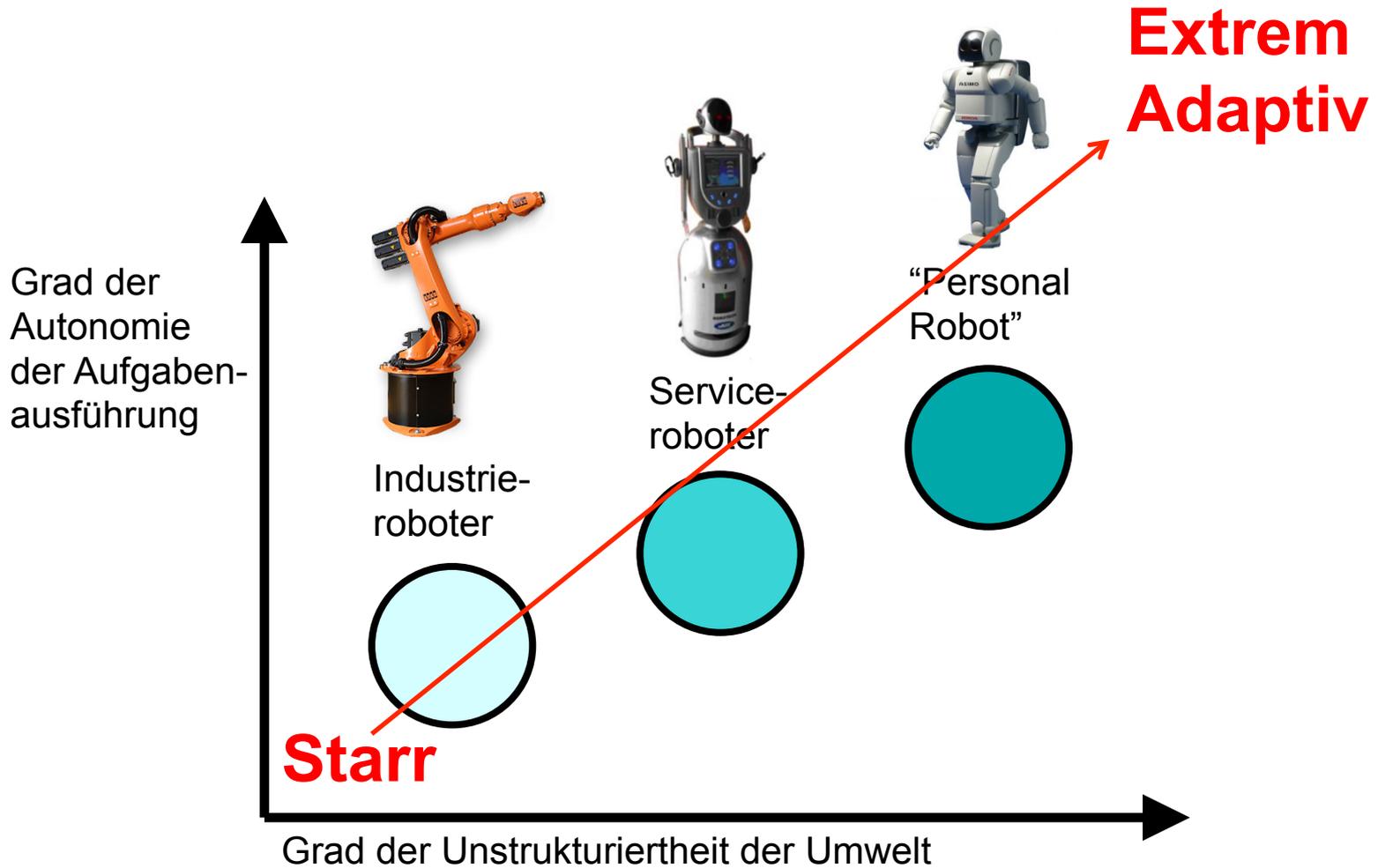
Dr.-Ing. Sven R. Schmidt-Rohr

Dr.-Ing. Rainer Jäkel

- **Motivation: Adaptive Roboterarbeiten**
- Objektmodelle
 - Geometrische Beschreibung
 - Zusätzliche Eigenschaften
- Szenenmodelle

Bisher in VL: un- oder marginal flexible Roboterarbeiten

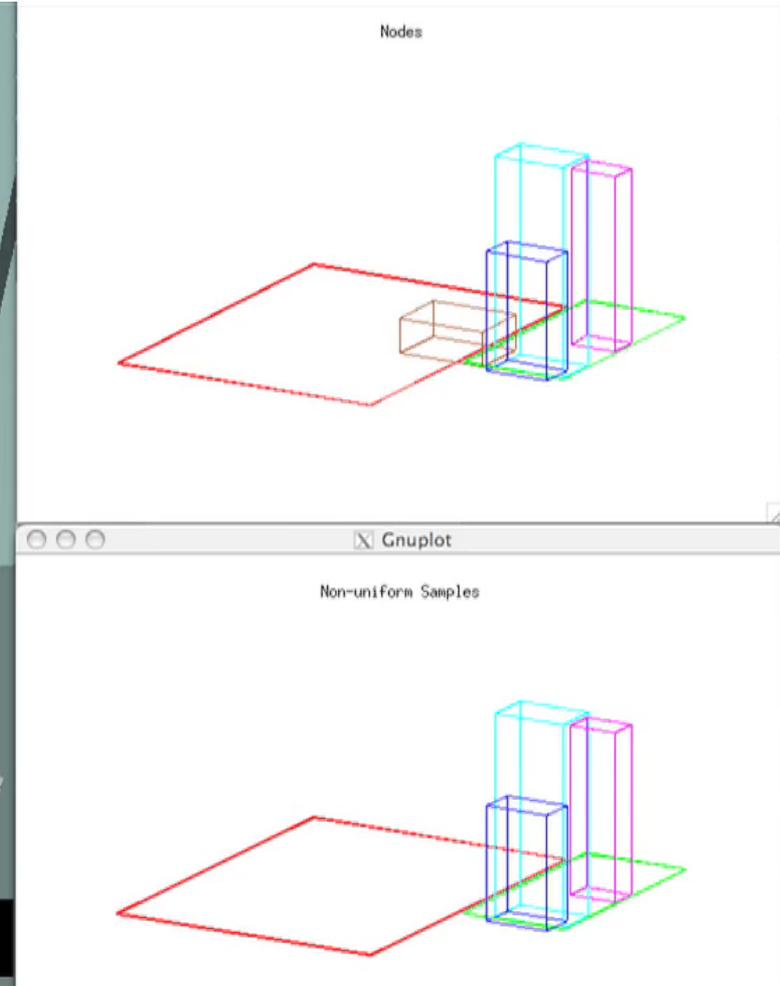
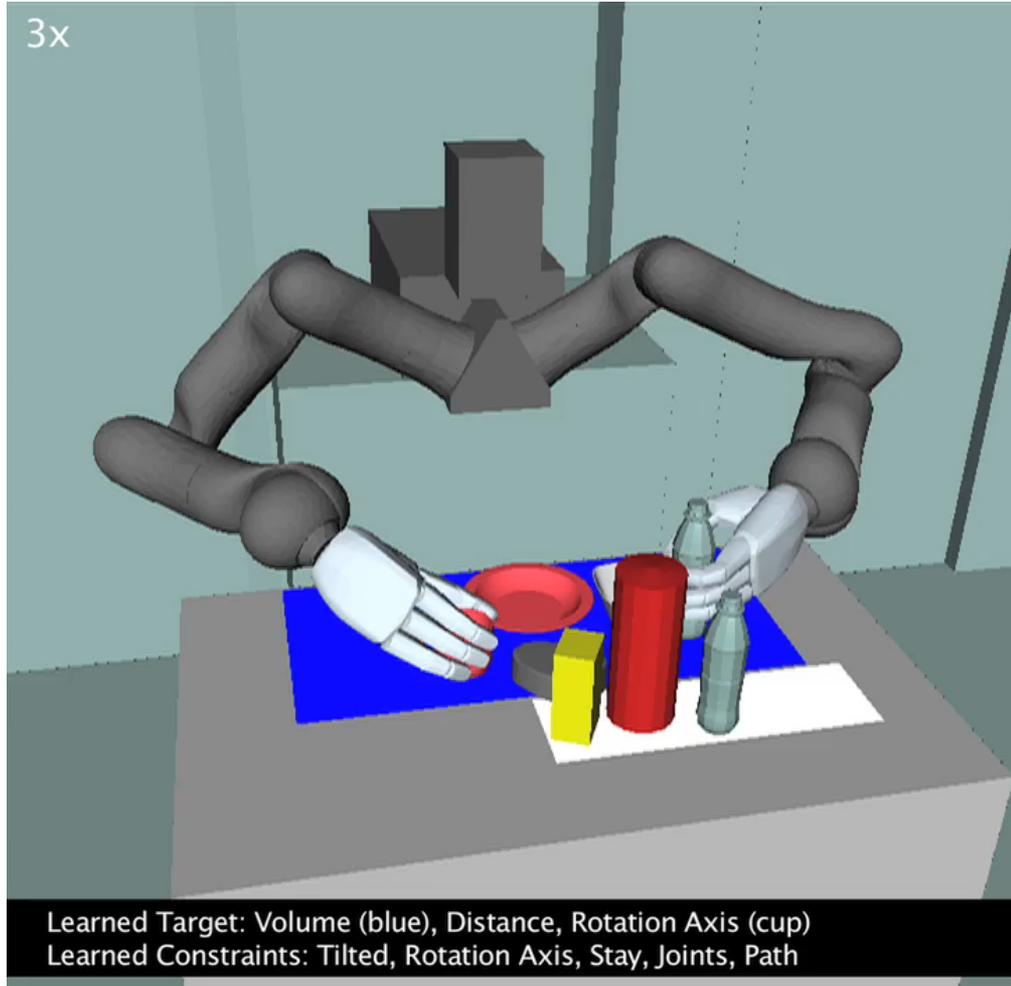
- Feste Trajektorie programmiert:
 - Positionsregelung
 - Eventuell lokale Kraftregelung
 - Benötigte Modelle für Programmierung und Regelung:
 - Kinematisches Modell
 - Dynamisches Modell
 - Benötigte Sensoren:
 - Gelenkencoder
 - (eventuell) Kraft-Momenten-Sensor
- Bahn absolut fest oder nur lokal-adaptiv, repetitiv ausgeführt



Bei adaptiven Roboterarbeiten andere Voraussetzungen!

- Keine feste Trajektorie programmierbar
 - Trajektorie muss flexibel vom Roboter bestimmt werden
 - Benötigte Modelle für Programmierung und Regelung:
 - Kinematisches Modell
 - Dynamisches Modell
 - **+ Umweltmodelle**
 - **+ Aufgaben/Planungsmodelle**
 - Benötigte Sensoren:
 - Gelenkencoder
 - (eventuell) Kraft-Momenten-Sensor
 - **+ visuelle und eventuell taktile Sensoren**
- Bahn völlig frei, flexibel, nicht-repetitiv ausgeführt

Bei adaptiven Roboterarbeiten andere Voraussetzungen!

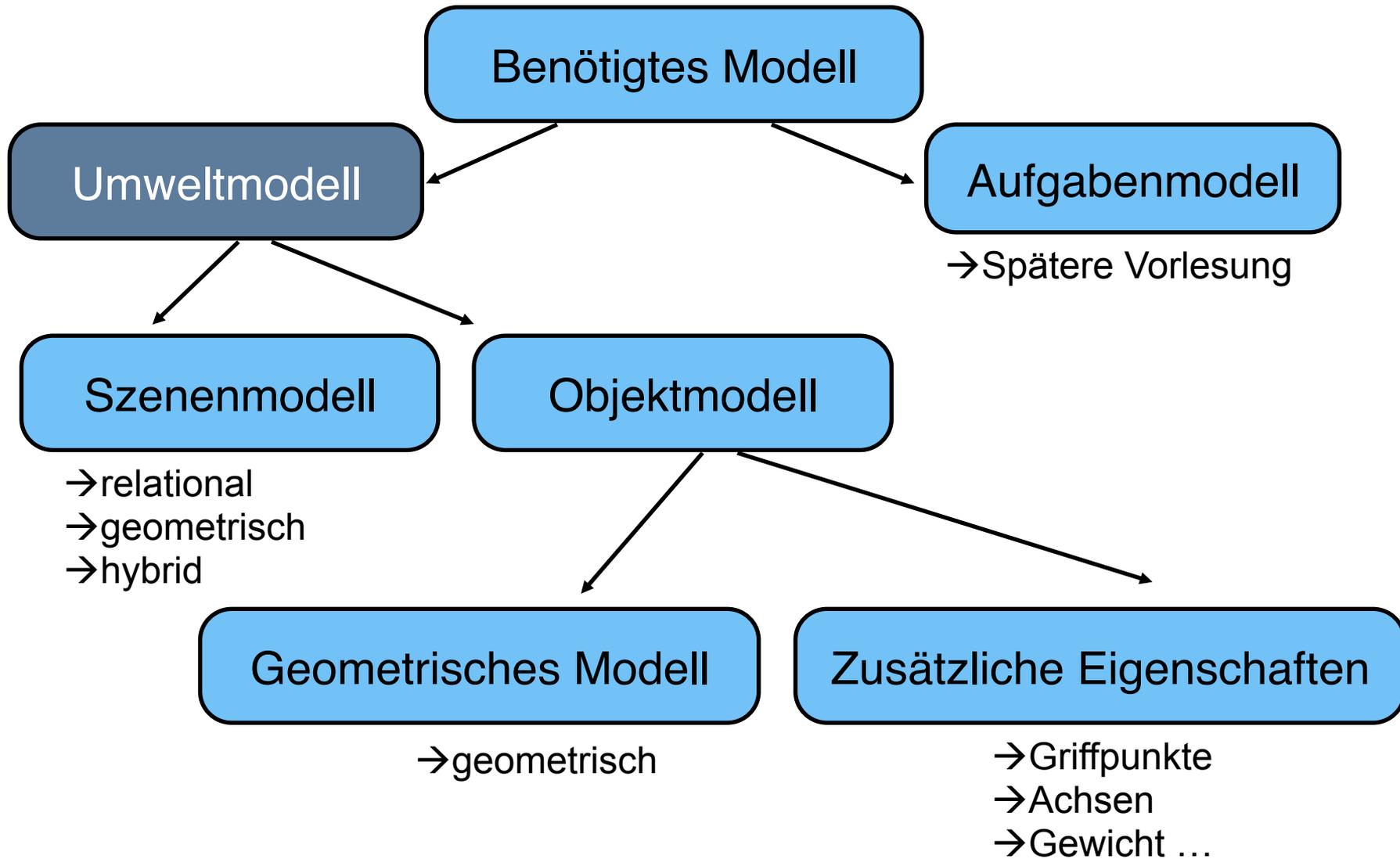


Zusätzliche Anforderungen gegenüber starren Aufgaben

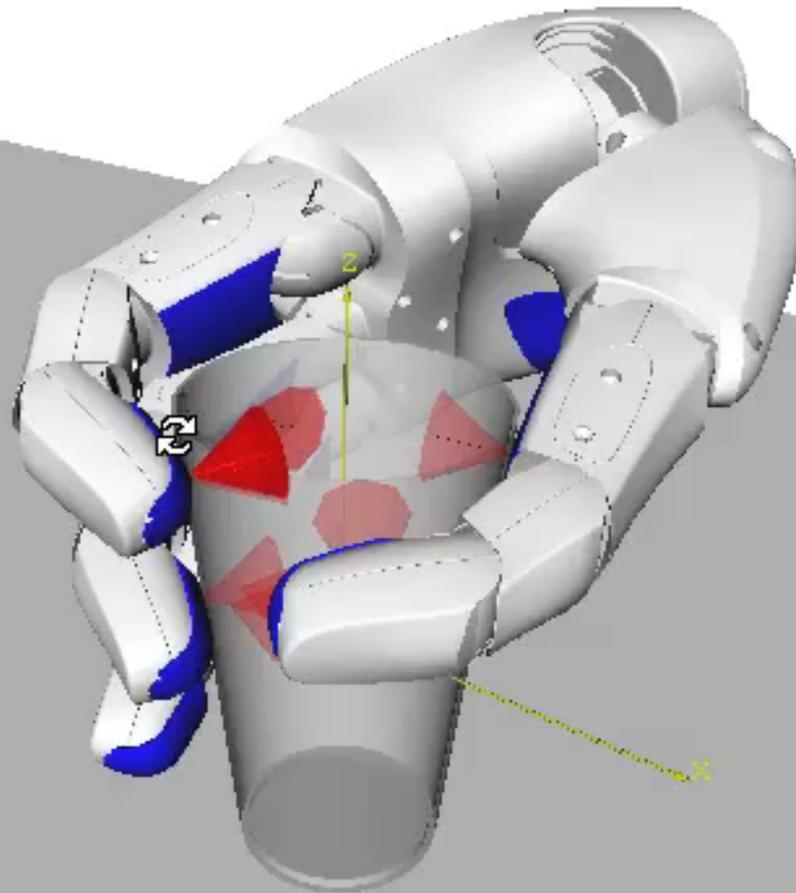
1. Visuelle und (event.) taktile Sensorik zur Umwelterfassung:
 - *Zentrales Thema der Vorlesung Robotik 3 im SoSe*
2. Datenrepräsentation der Umgebung/Umwelt:
 - *Diese Vorlesung*
3. Planungsmethoden zur Aufgaben/Bewegungsberechnung
 - *Folgende Vorlesungen*

Verschiedene Arten von Modellen in der Robotik

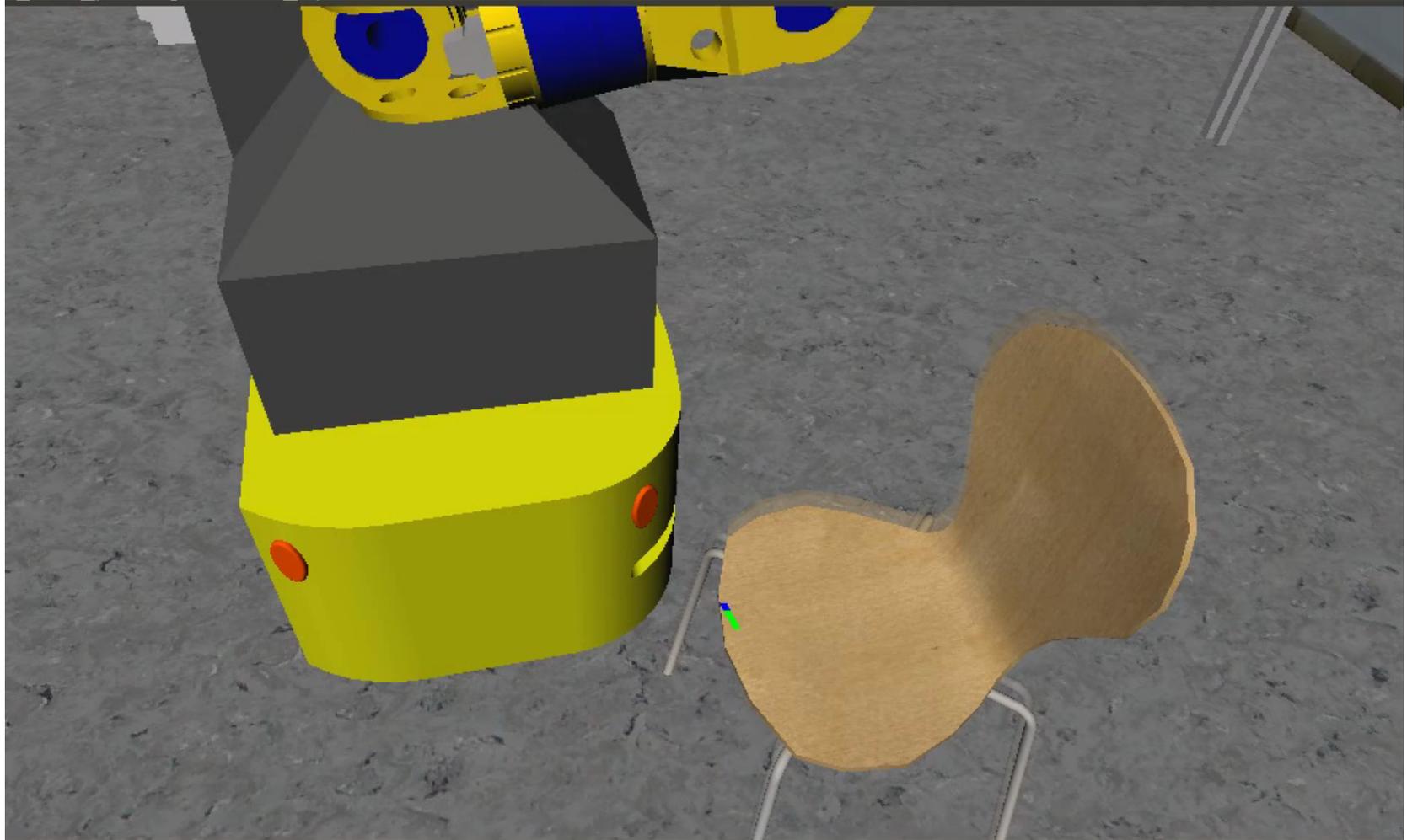
1. Modell der Kinematik (Manipulator, mobile Plattform)
2. Modell der Dynamik (Manipulator, mobile Plattform)
3. Modelle der Sensoren
4. Umweltmodell (Umgebung: Einzelobjekte und Szenen)
5. Aufgabenmodell (Bewegungen und abstrakte Aufgaben)



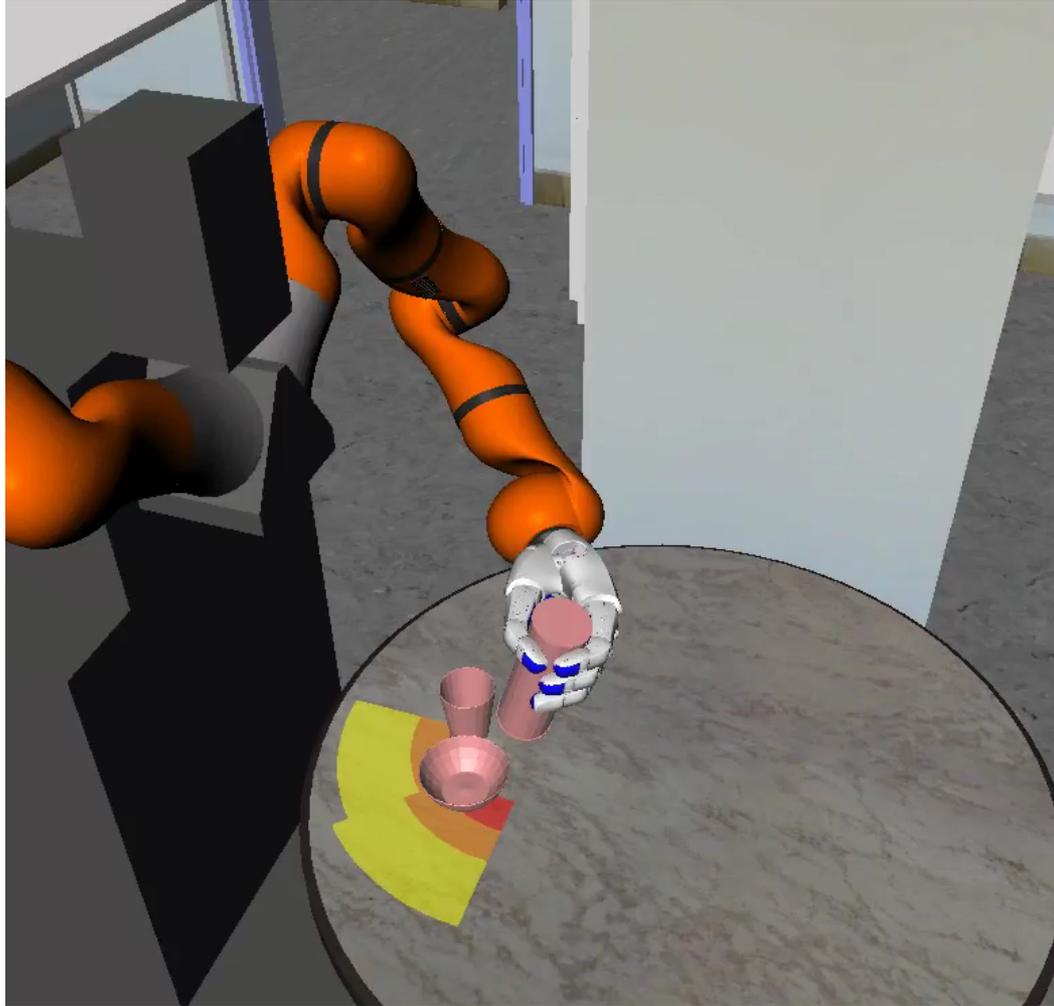
- Motivation: Adaptive Roboterarbeiten
- **Objektmodelle**
 - **Geometrische Beschreibung**
 - Zusätzliche Eigenschaften
- Szenenmodelle



Simulation der Effekte von Handlungen auf die Umwelt



Simulation der Effekte von Handlungen auf die Umwelt



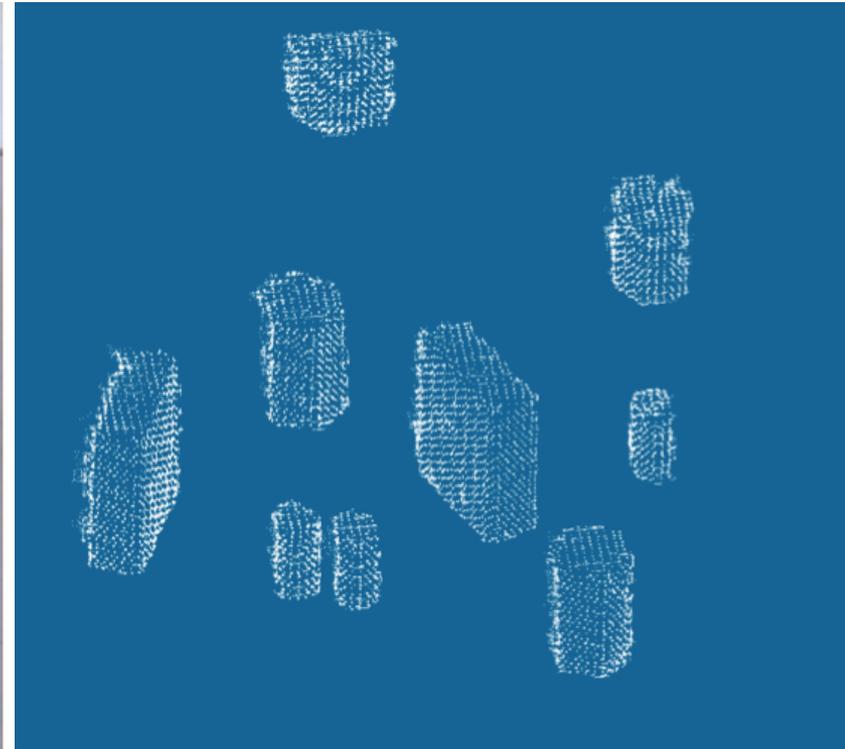
Anwendungsgebiete für das geometrische Modell

- Umweltwahrnehmung (Live)
 - Objektklassifizierung
 - Objektlokalisierung
 - Bewegungserkennung/klassifikation des Menschen
- Bewegungsplanung (Live, offline)
 - Pfadplanung (mobile Systeme)
 - Bewegungsplanung (Manipulatoren)
 - Greif- und Inhandmanipulationsplanung
- Abstrakte Aufgabenplanung (Live, offline)
- Dynamische Effektsimulation und –prädiktion (Live, offline)
- Arbeitsraum/Anlagenplanung (Offline)

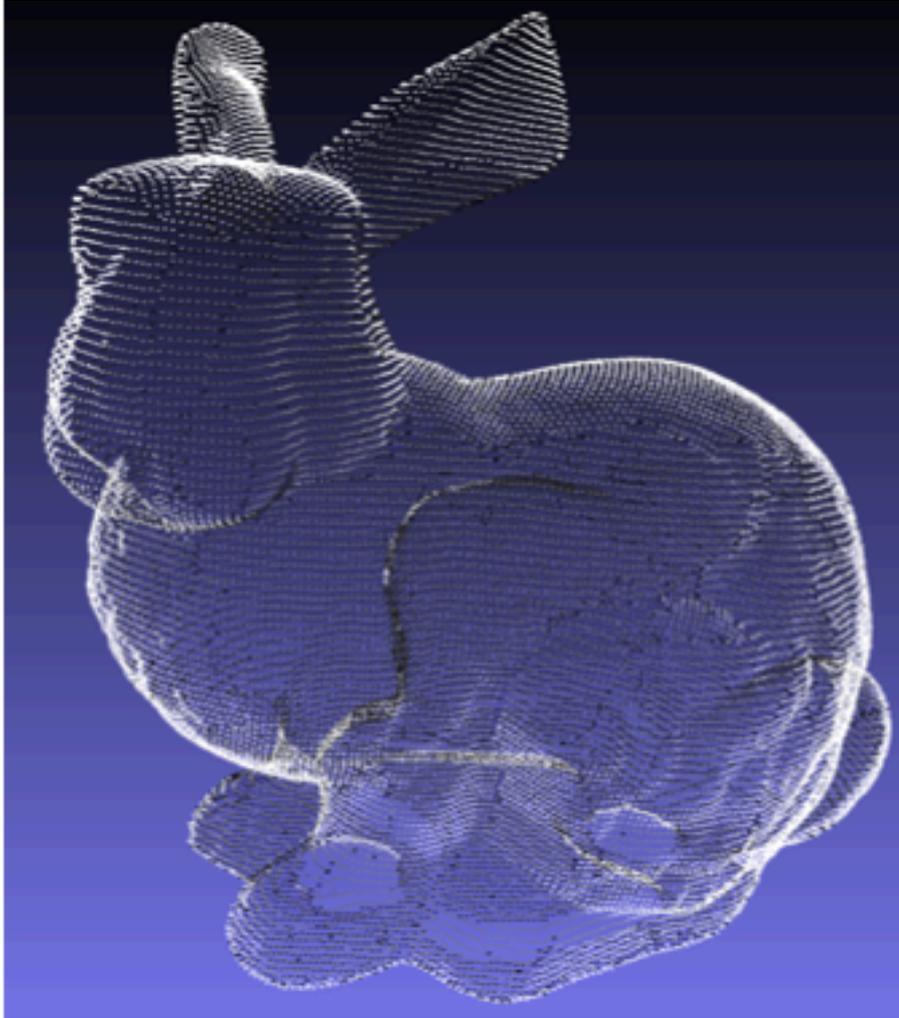
- **Punktwolken**
- Approximative **Oberflächenmodelle** (B-Rep)
 - Dreiecksflächen (Meshes)
 - Vierecksflächen (Pflaster)
 - Bezierflächen
- Approximative **Zellenbelegung**
 - Voxel
 - Octtree
- Analytisch-**parametrische Modelle**
 - Constructive Solid Geometry (CSG)

Punktwolken (Point Clouds) sind das einfachste Objektmodell

- Modell ist eine beliebige Menge aus Raumpunkten
- Daten direkt aus aktuellen Tiefensensoren (3D-Sensoren)
- Objekte nur implizit, indirekt modelliert
- Mehrdeutigkeiten
- Hochgradig approximativ
- Datenlücken schon durch die Modellform
- Grundlage für weitergehende Objektmodelle
- Es gibt aber auch Verfahren, um direkt auf Punktwolken z.B. Objekte oder relevante Frames im Raum zu lokalisieren



Punktwolken (Beispiele)



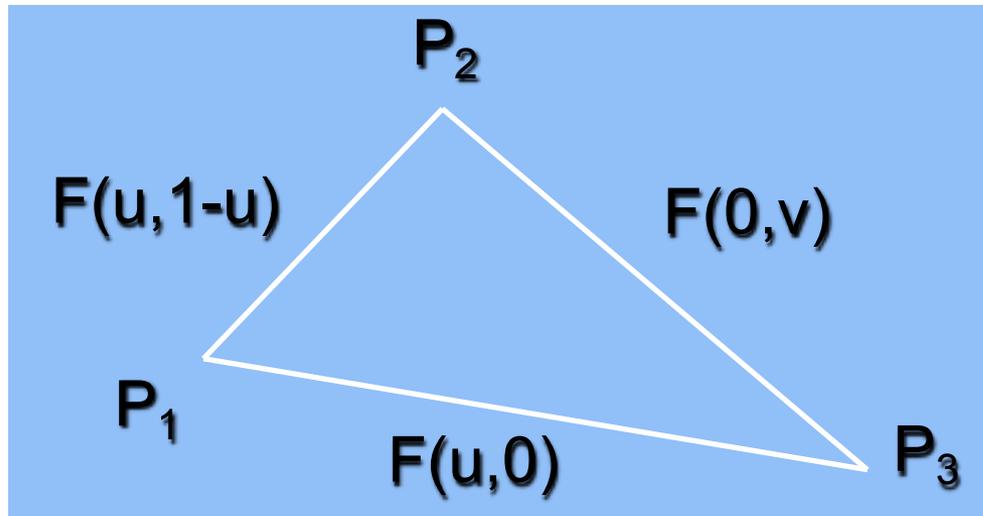
(a) Stanford Bunny dargestellt mit Punkten

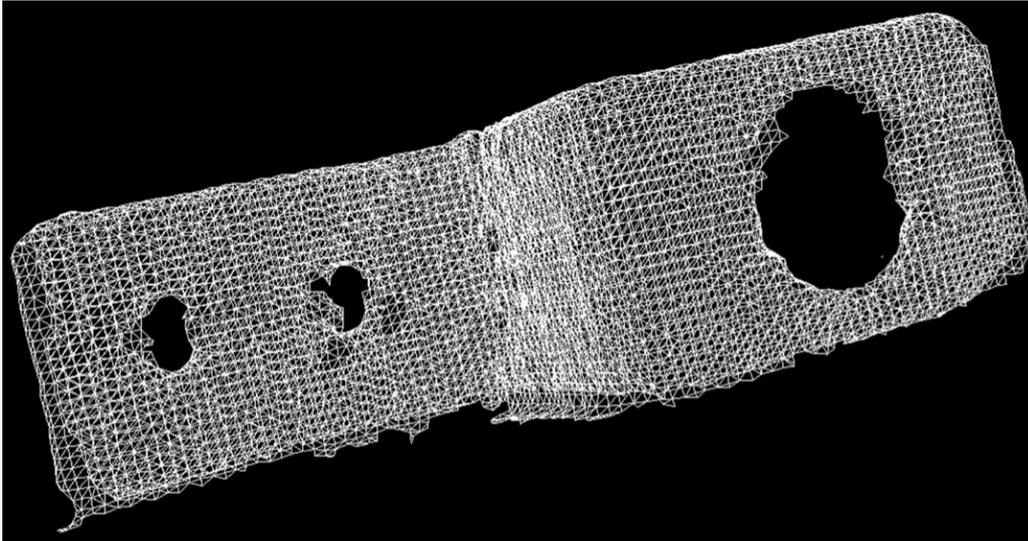


(b) Stanford Bunny dargestellt mit Flächen

- Bildung einer großen Fläche aus einem Netz („Mesh“) von einfachen Einzelflächen, z.B. Dreiecke, Vierecke
- **Vorteile:**
 - + Definition sehr einfach
 - + einfache Algorithmen
- **Nachteile:**
 - hoher Speicherbedarf
 - hoher Rechenaufwand

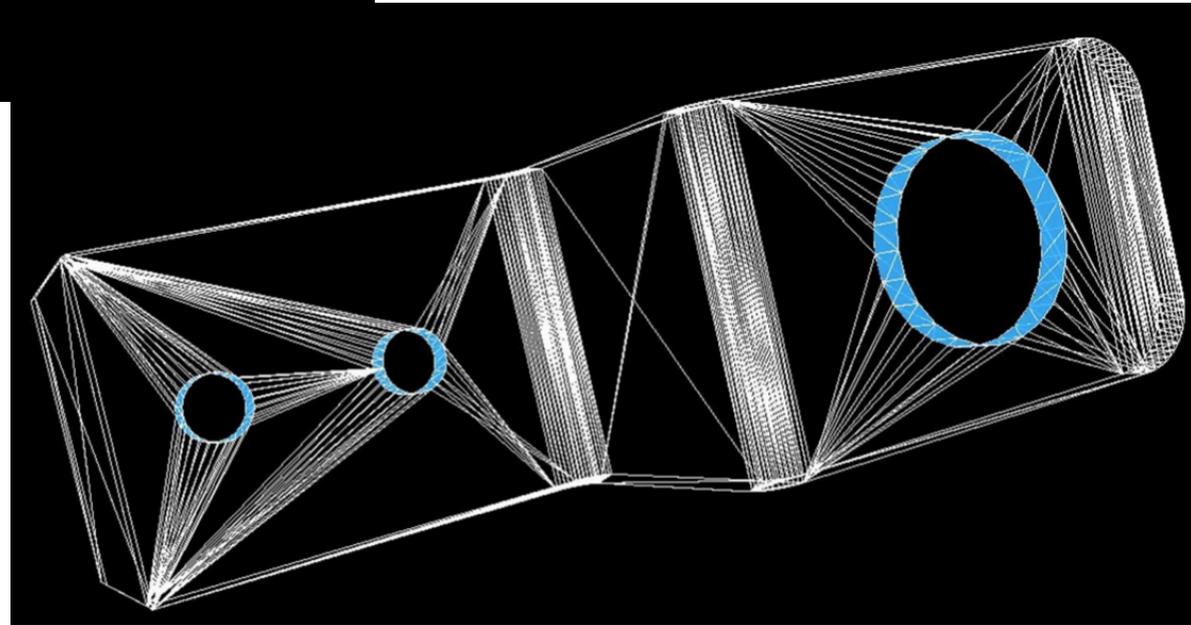
- Approximation von Freiformflächen: Die einfachste Fläche ist die Dreiecksfläche (Mesh)
- Definition: Gegeben seien 3 Punkte im Raum P_1, P_2, P_3
Damit hat die Fläche folgende Gleichung:
 $F(u,v) = u \cdot P_1 + v \cdot P_2 + (1-u-v) \cdot P_3$ mit $0 \leq u, v, u+v \leq 1$.

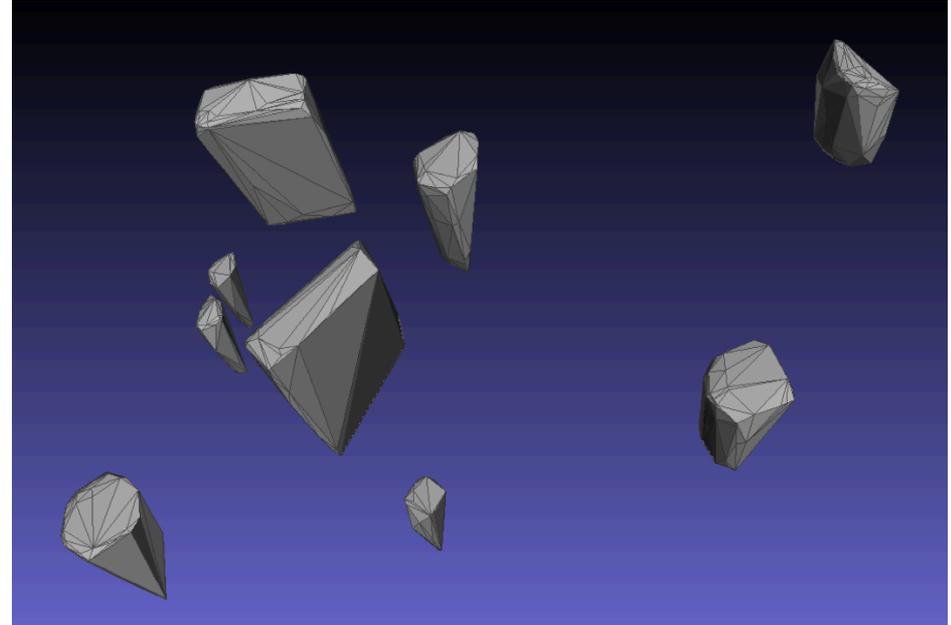




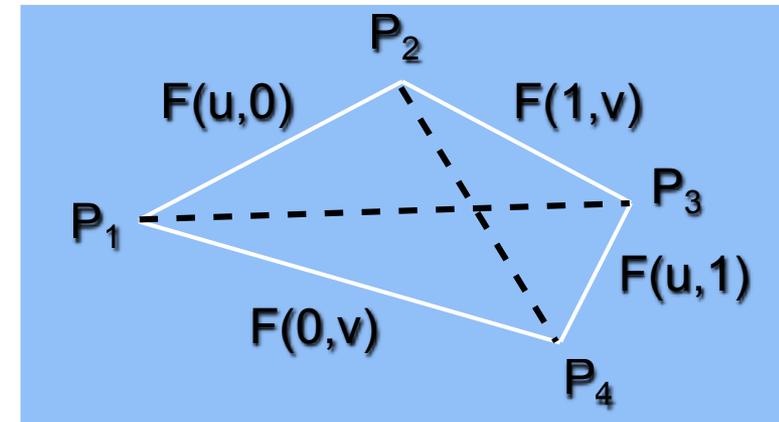
Mesh aus
Laserdaten
(Punktwolke)
rekonstruiert
(links oben)

Mesh aus
CAD-Modell
(analytisches Modell)
generiert
(rechts unten)





- Definition: Gegeben sind 4 Punkte im Raum P_1, P_2, P_3, P_4
Damit wird die Fläche definiert durch
$$F(u,v) = (1-u)(1-v) \cdot P_1 + (1-u)v \cdot P_2 + u(1-v) \cdot P_3 + uv \cdot P_4$$
mit $0 \leq u \leq 1, 0 \leq v \leq 1$.
- **Vorteil:**
 - + Flächenelemente können gekrümmt sein
⇒ weniger Gitterpunkte bei gleich guter Approximation
- **Nachteil:**
 - Rechnen mit gekrümmten Flächen ist aufwendig



Erweiterung des Ansatzes der Bezierkurven

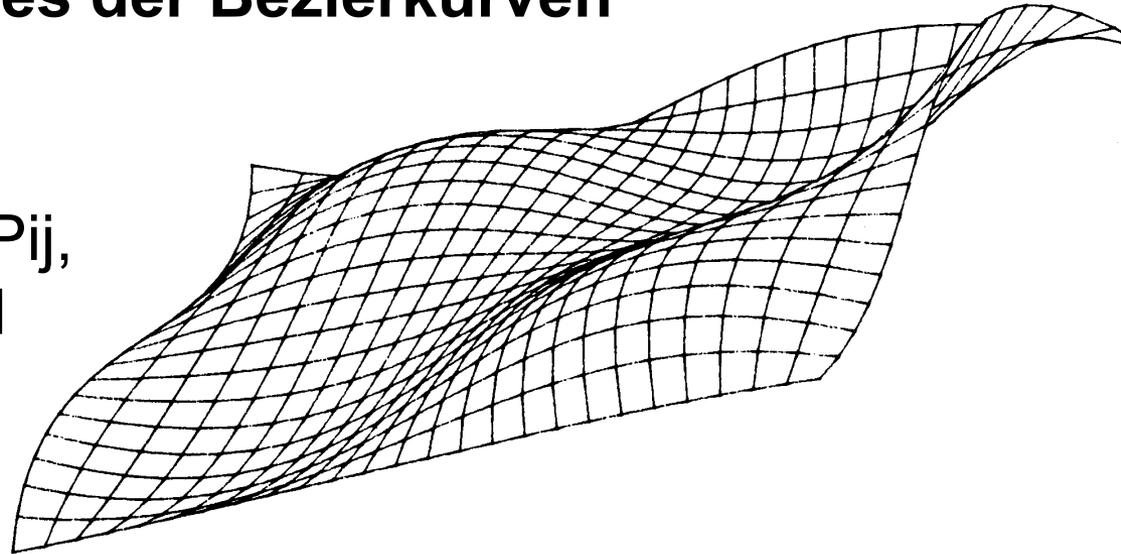
- Definition:
Gegeben ist ein Gitter von Führungspunkten P_{ij} ,
 $0 \leq i \leq N$ und $0 \leq j \leq M$
- Damit ist die Fläche beschrieben durch

$$F(u, v) = \sum_{i=0}^N \sum_{j=0}^M P_{ij} \cdot B_{i,N}(u) \cdot B_{j,M}(v)$$

mit $B_{i,N}(u) = (1-u)B_{i,N-1}(u) + uB_{i-1,N-1}(u)$

$B_{j,M}(v) = (1-v)B_{j,M-1}(v) + vB_{j-1,M-1}(v)$

Die $B_{i,N}$ bzw. $B_{j,M}$ heißen auch Bernsteinpolynome.



Geometrische Modellierung von Objekten

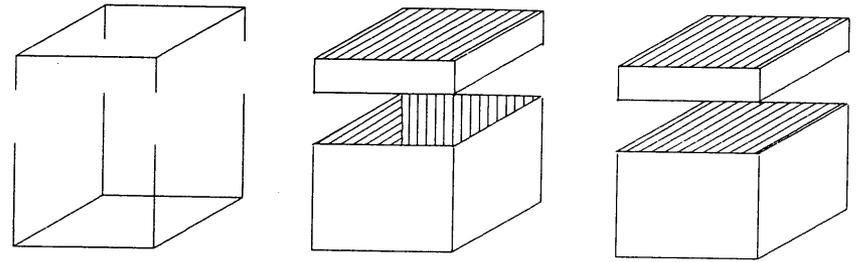
- Kantenmodell
- Flächenmodell
- Volumenmodell

Reines Kantenmodell ohne weitere Informationen

- Nur die Kanten werden gespeichert, d.h. Punkte und
- Verbindungen (Gerade, Polygonzug, Bezierkurve, ...)

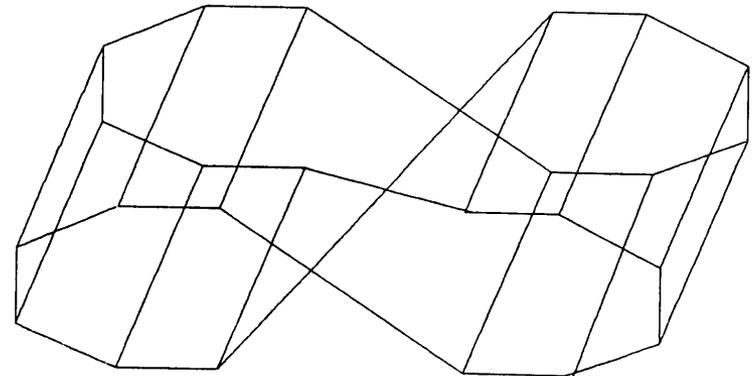
- **Vorteile:**

- + einfache Daten
- + wenige Daten

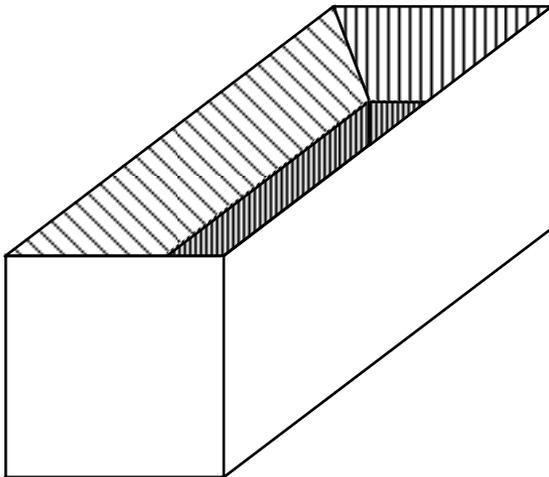
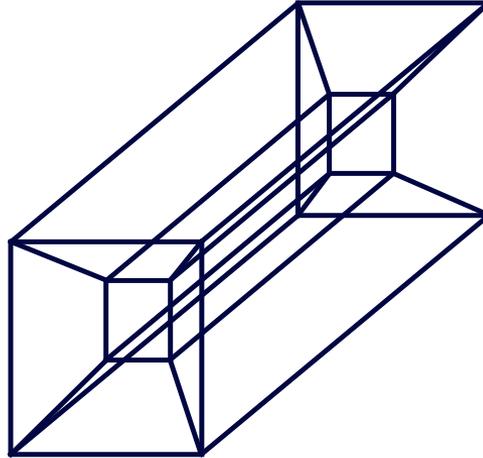


- **Nachteile:**

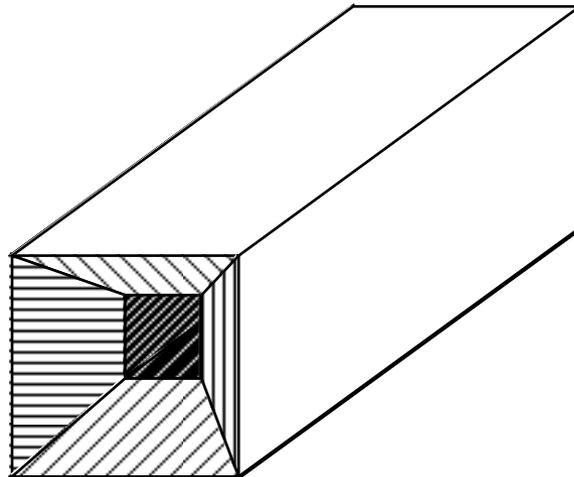
- Mehrdeutigkeiten
- hoher Eingabeaufwand
- keine Kollisionsberechnung
- kein Schnitt



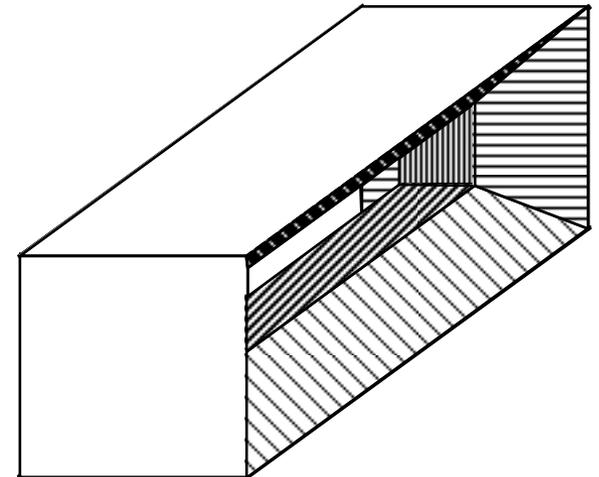
Kantenmodell



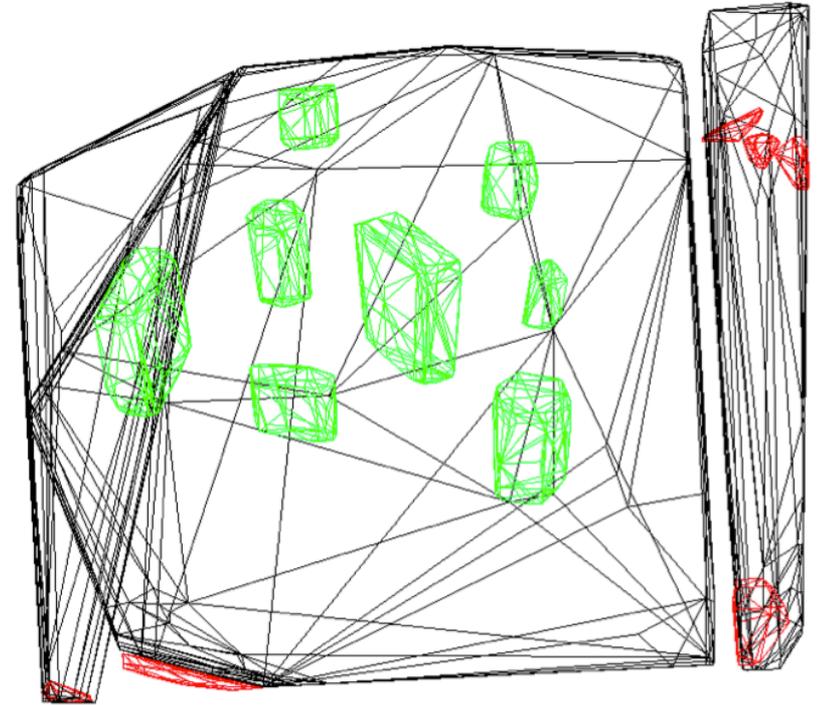
oben - unten



vorne - hinten



links - rechts



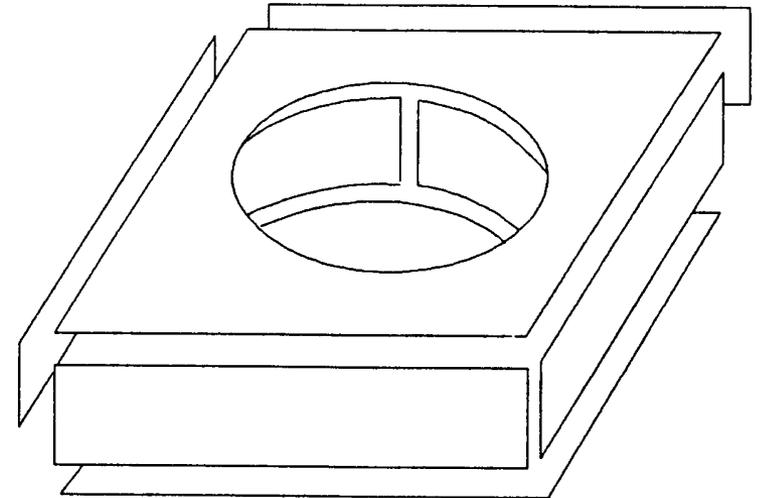
Speicherung von Kanten & Oberflächen (Dreiecke, Bezier)

- **Vorteile:**

- + effiziente Verfahren
- + entspricht dem Vorgehen während der Modellierung
- + schnelle Kollisions- und Abstandsberechnung

- **Nachteile:**

- hoher Eingabeaufwand
- Darstellung aufwendig
- Problem bei Schnittoperationen
- Inkonsistenzen möglich



Organisationsform der geometrischen Flächenmodelle

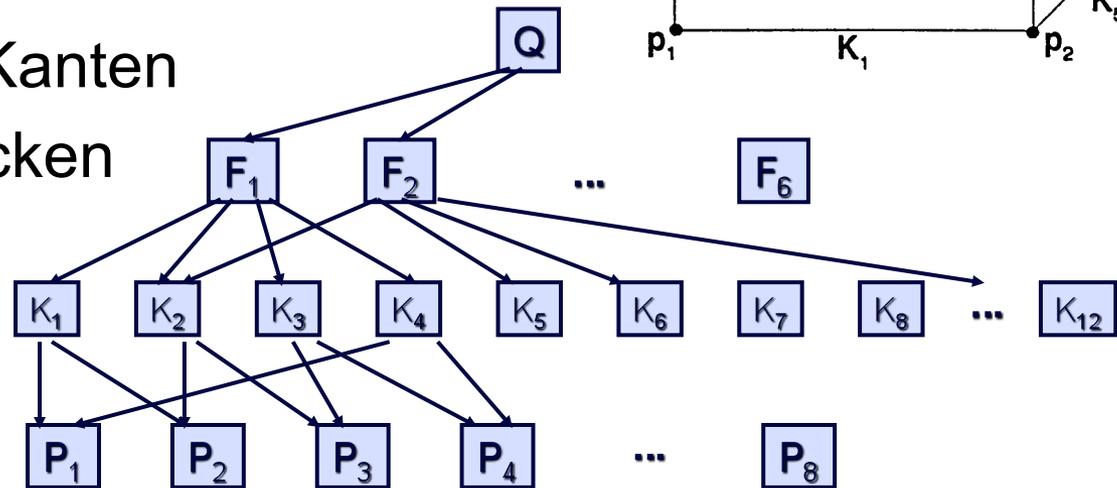
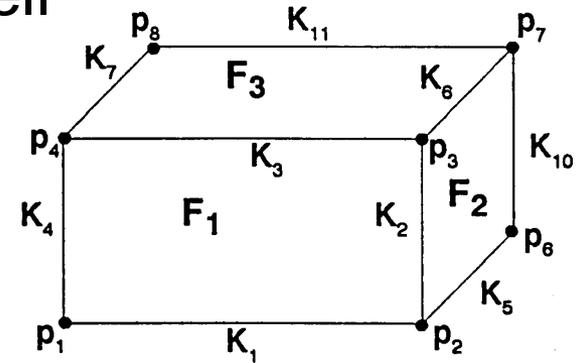
- Hierarchische Darstellung eines Objektes durch begrenzende Elemente, i.d.R. Kanten oder Flächen.
- Elemente eines Quaders im Flächenmodell
- Elemente:

Q : Quader

F_i : $i \in \{1, \dots, 6\}$: Flächen

K_i : $i \in \{1, \dots, 12\}$: Kanten

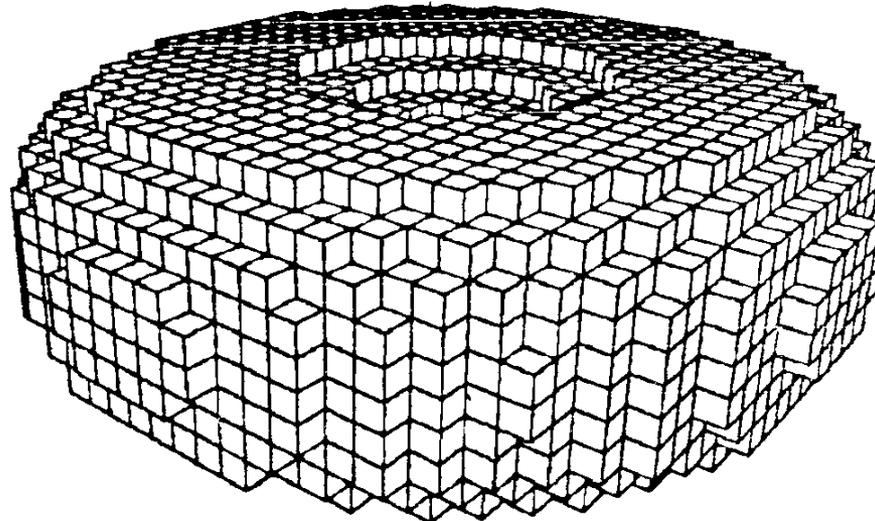
P_i : $i \in \{1, \dots, 8\}$: Ecken



Aus topologischer Struktur Informationen über

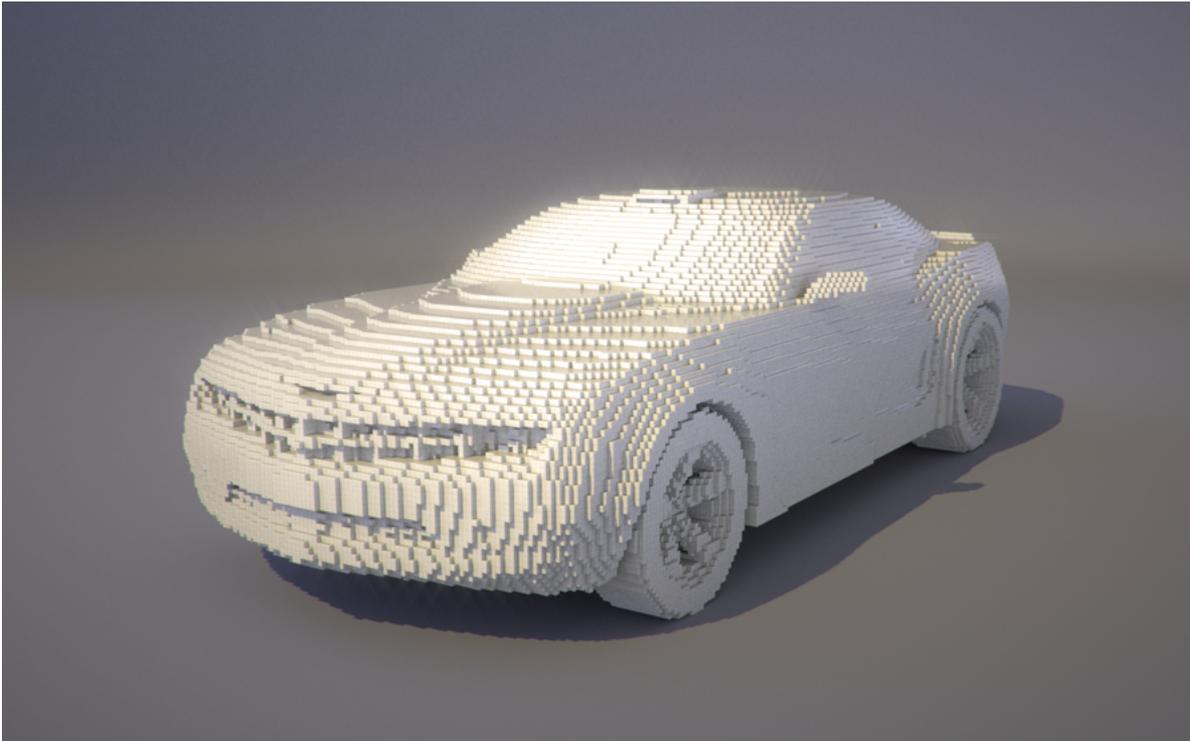
- Welche Flächen gehören zum Objekt?
 - Welche Kanten gehören zur Fläche?
 - Zu welchem Objekt gehört eine Fläche?
 - Zu welchem Objekt gehört eine Kante?
 - Welche Flächen stoßen aneinander?
- *kantenbasierte Objekterkennung*

- Objekte werden aus disjunkten Elementarzellen aufgebaut. Verwendung finden einfache geometrische Objekte z.B. Tetraeder, Quader, ...
- Benutzt in der Strukturanalyse mit Finite-Elemente-Methoden (FEM).



Äquidistante Raumunterteilung in 3D

- Speicherungsmöglichkeiten
 - 0: nicht-belegt / 1: belegt / 2: weiß-nicht (unbekannt)
 - Wahrscheinlichkeit belegt 0.0 bis 1.0



Äquidistante Raumunterteilung in 3D

- **Vorteile:**

- + Einfache Darstellung
- + Berechnungen homogen, parallel auf GPU
- + Raycasting u.ä. einfach parallelisierbar

- **Nachteile:**

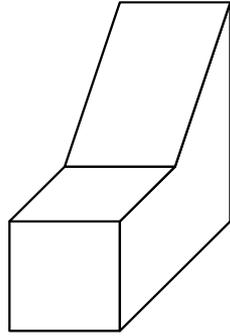
- Festes Gitter, gleich(mäßig) sowohl bei großflächigen als auch detaillierten Strukturen
- Kartengröße/Präzision durch Speicher begrenzt

Octtree: effizientere Topologie als Voxel

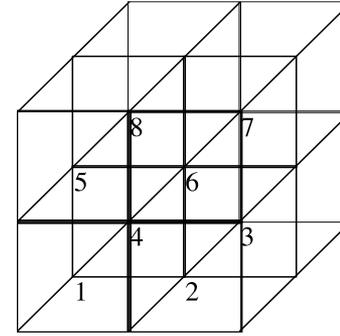
- Der Raum wird in mehrere Zellen unterteilt (i.d.R. 8 Zellen: „Octree“).
- Zelle komplett vom Objekt belegt → als „belegt“ markieren
- Wenn die Zelle nur teilweise belegt ist, dann wird auf diese Zelle das Verfahren rekursiv angewendet. Ansonsten ist die Zelle leer.
- Die Rekursion terminiert bei einer vorbestimmten minimalen Zellgröße.
- Teilbelegte kleinste Zellen werden als belegt markiert.

Octtree (Beispiel)

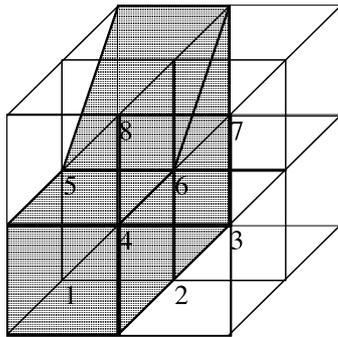
Körper



Zerlegung

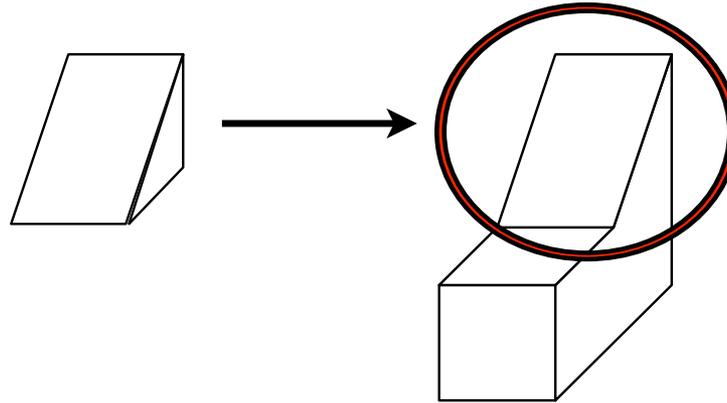


1. Zerlegung

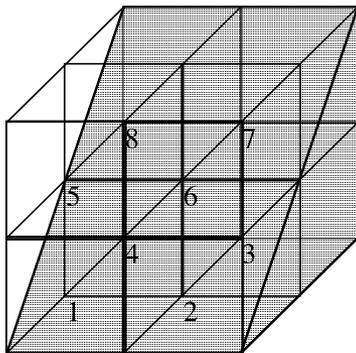


Zelle 1	belegt
Zelle 2	frei
Zelle 3	frei
Zelle 4	belegt
Zelle 5	frei
Zelle 6	frei
Zelle 7	frei
Zelle 8	teilbelegt

Restkörper

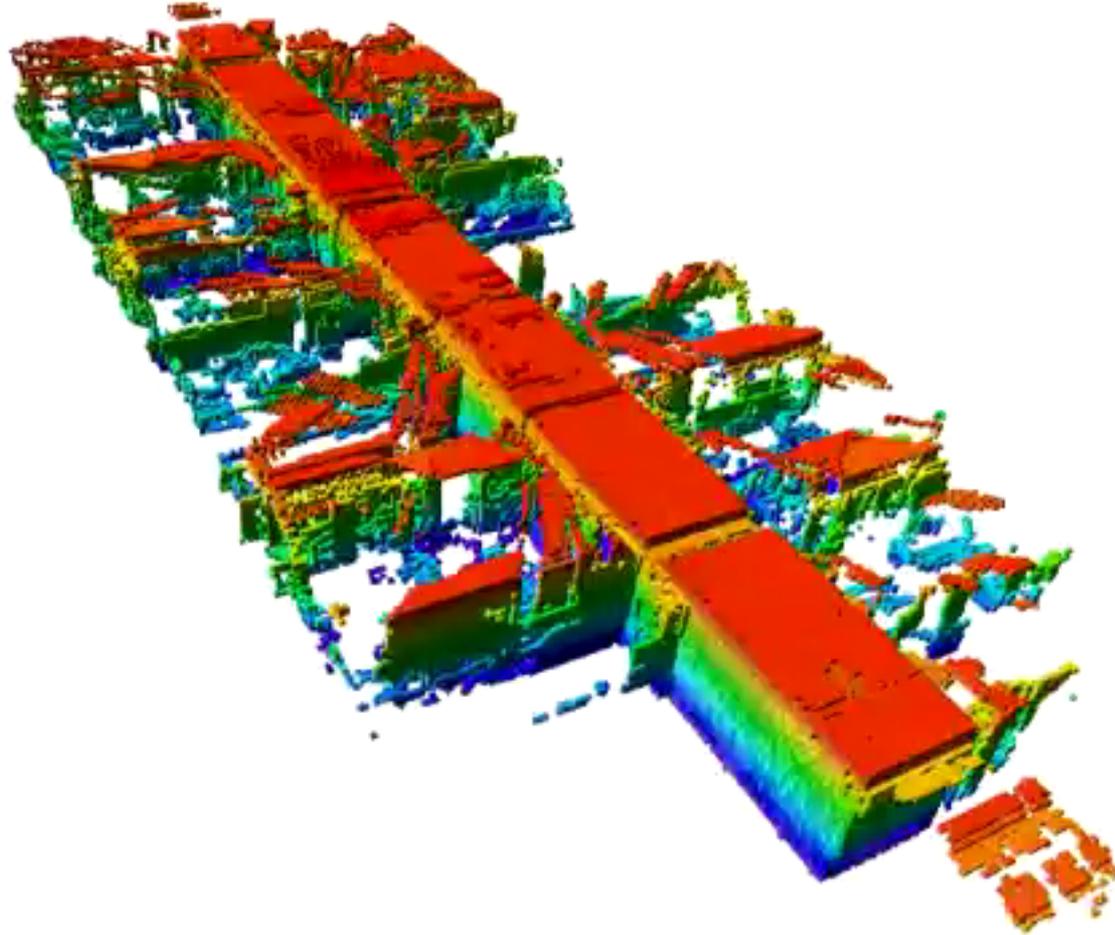


2. Zerlegung



Zelle 1	teilbelegt
Zelle 2	teilbelegt
Zelle 3	belegt
Zelle 4	belegt
Zelle 5	frei
Zelle 6	frei
Zelle 7	teilbelegt
Zelle 8	teilbelegt

Octomap (BSD Lizenz) für Kartierung mobile Roboter:



<http://octomap.github.io>

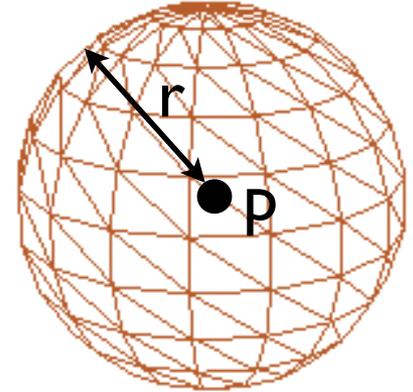
Octomap (BSD Lizenz) für Manipulation:



<http://octomap.github.io>

Beispiel analytisch gegebene Fläche

- Beispiel: 3D-Kugel
- $r = \|x - p\|$
- Exaktes Modellierungsverfahren
- **Vorteile:**
 - + Geschlossene Darstellung (wenig Speicherbedarf)
 - + Analytische Darstellung erlaubt einfache Rechenverfahren (z.B. Schnitt von Ebenen / Kugeln → schnelle Kollisionsberechnung)
- **Nachteil:**
 - Wenige Flächen sind analytisch darstellbar



- Grundkörper und topologische Operationen auf diesen (Schnitt, Vereinigung, ...) werden abgespeichert.

- **Vorteile:**

- + eindeutige Objektbeschreibung

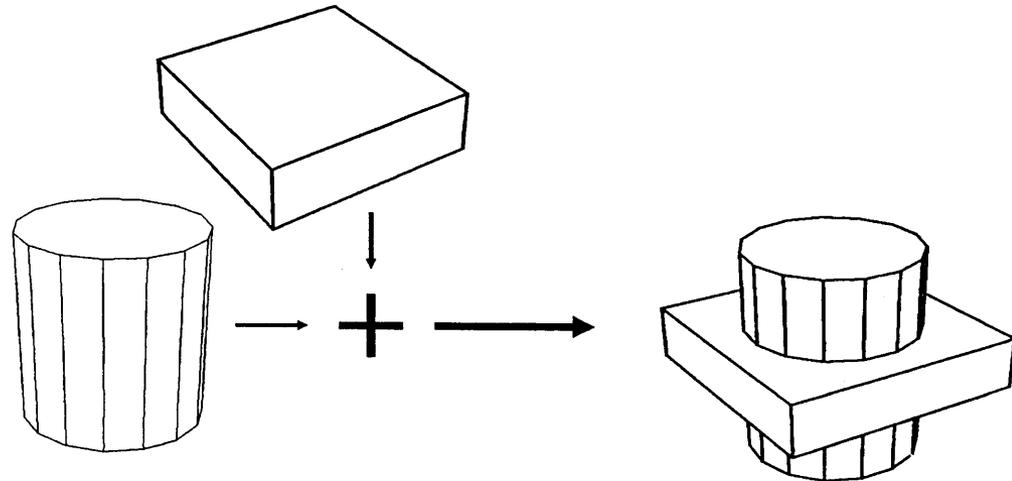
- + geringer Eingabeaufwand

- + Ergebnis von Operationen sind korrekte Objekte

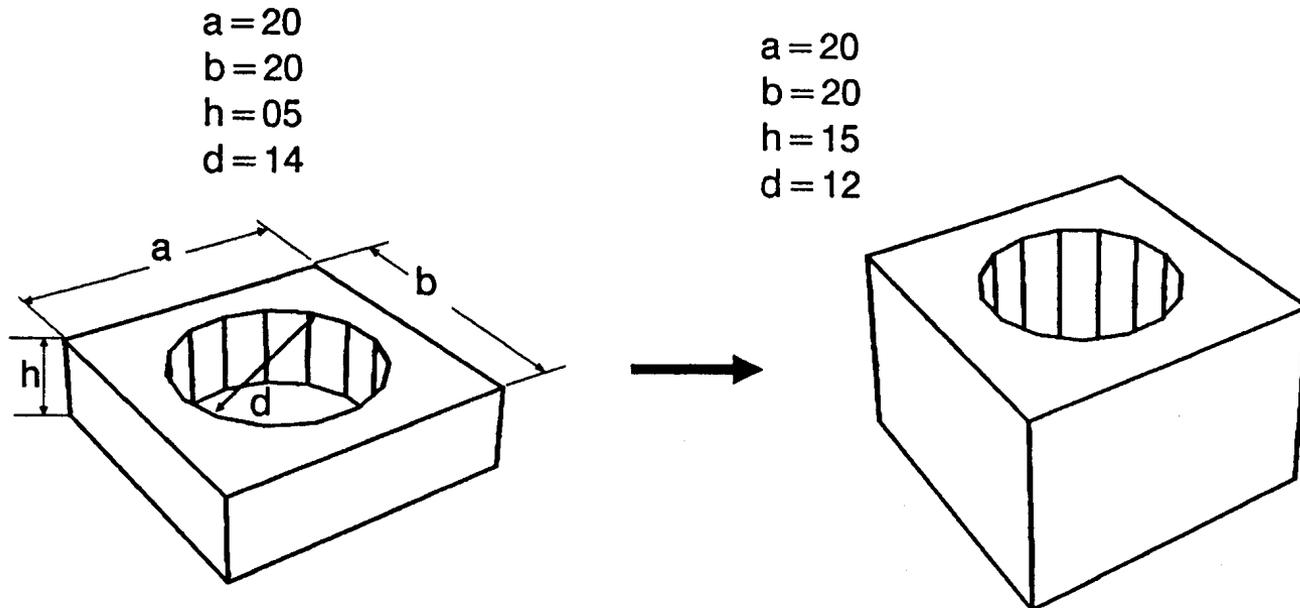
- **Nachteile:**

- hoher Implementierungsaufwand

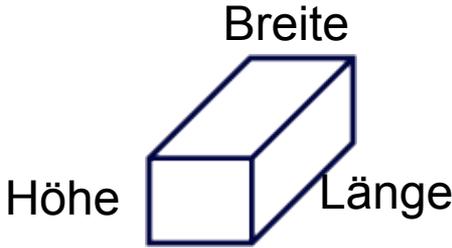
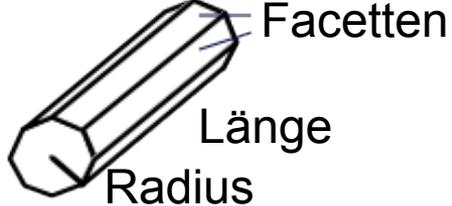
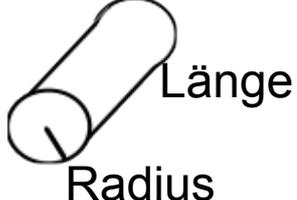
- Einbindung von Freiformflächen schwierig

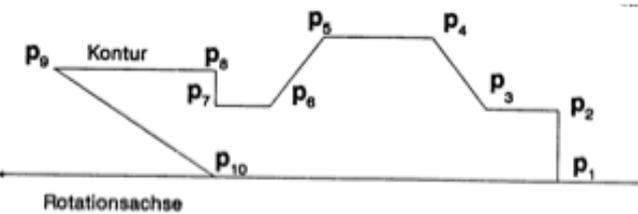
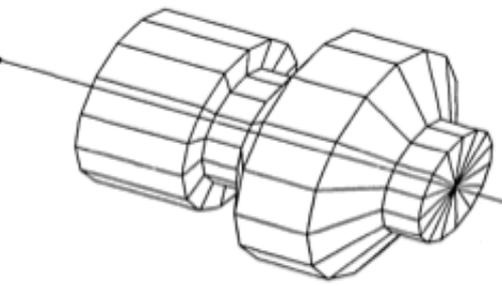


- Objekte sind bereits vorhanden und können durch Angabe von Parametern angepasst werden (Varianten).
- Konsistenzprüfungen sind notwendig ! ($d < a$)

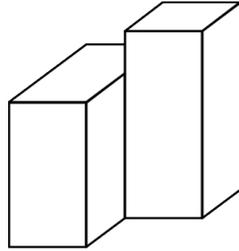


- Es gibt eine Menge von einfachen Grundkörpern, die parametrisiert werden können und auf denen verschiedene Operationen definiert sind.

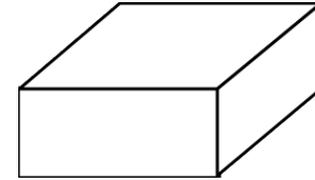
Grundkörper	Parameter	Skizze
Quader	Länge, Breite, Höhe	
Prisma	Länge, Radius, Facetten	
Zylinder	Länge, Radius	

Grundkörper	Parameter	Skizze
Kegel	Länge, Radius a, Radius b	 <p>Radius a Länge Radius b</p>
Ellipsoid	Radius a, Radius b, Radius c	 <p>Radius b Radius c Radius a</p>
Rotationskörper	Achse, Kontur 	 <p>Rotationsachse</p>

Objekt A

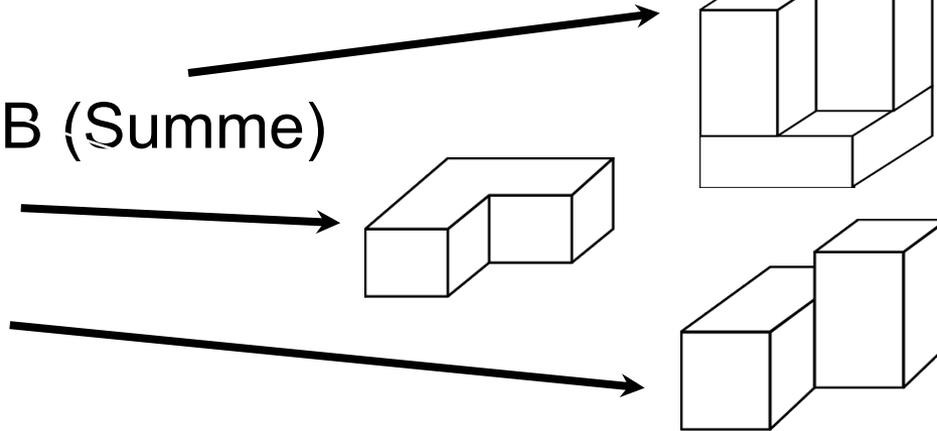


Objekt B



Operatoren:

- Vereinigung $A \cup B$ (Summe)
- Schnitt $A \cap B$
- Differenz A / B

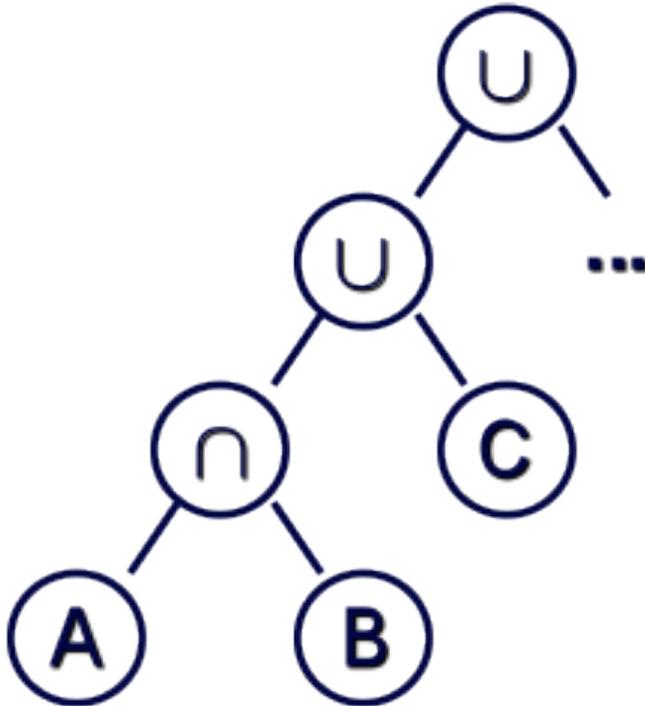


- Sweep:

Ein Grundelement (u.U. eine Fläche) wird entlang einer Raumkurve verschoben. Der durchdrungene Raum stellt das neue Objekt dar.

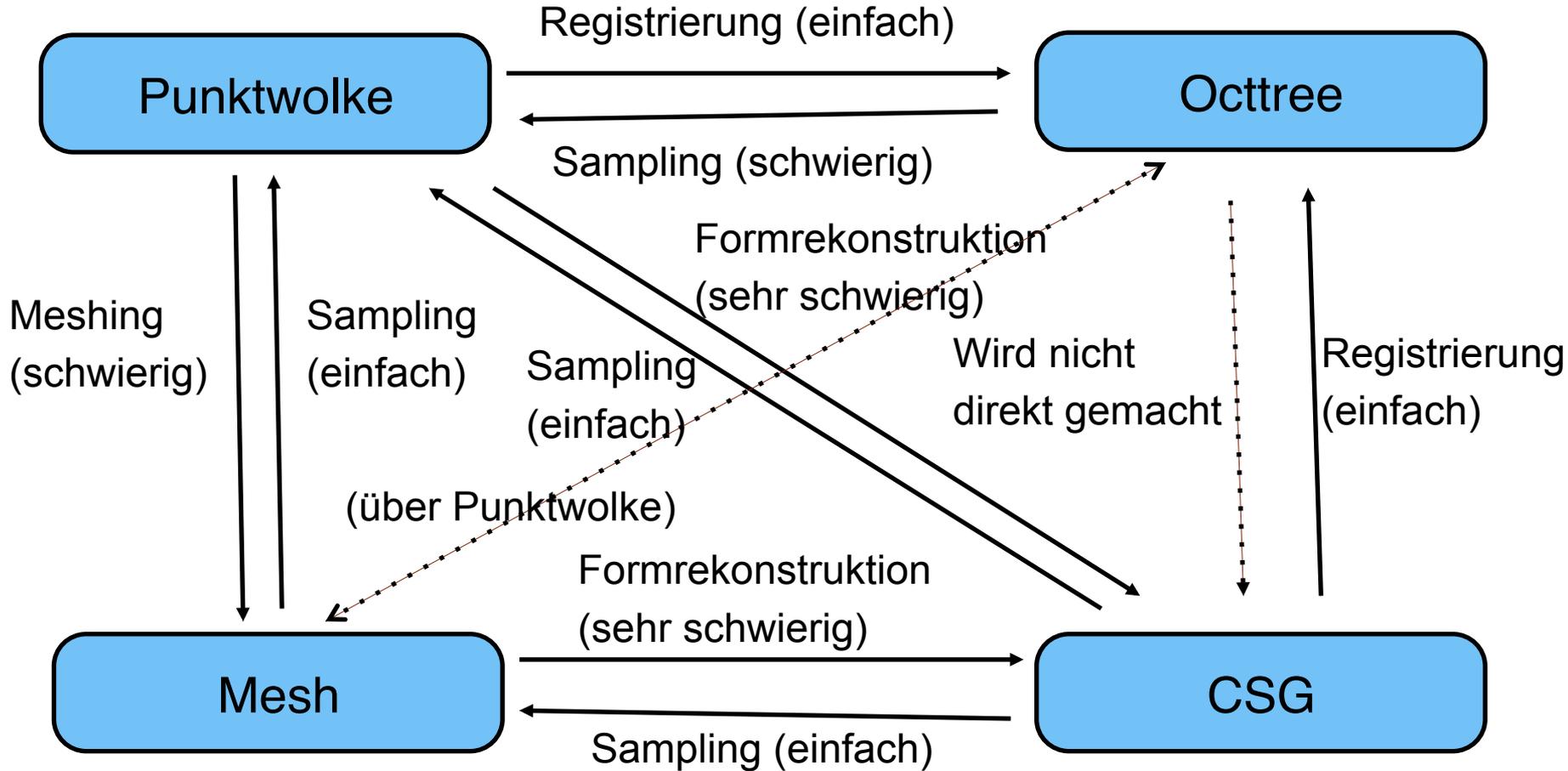
Speicherung erfolgt als Binär-Baum mit

- Knoten als Operation
- linker bzw. rechter Teilbaum als Teil-CSG oder Grundkörper

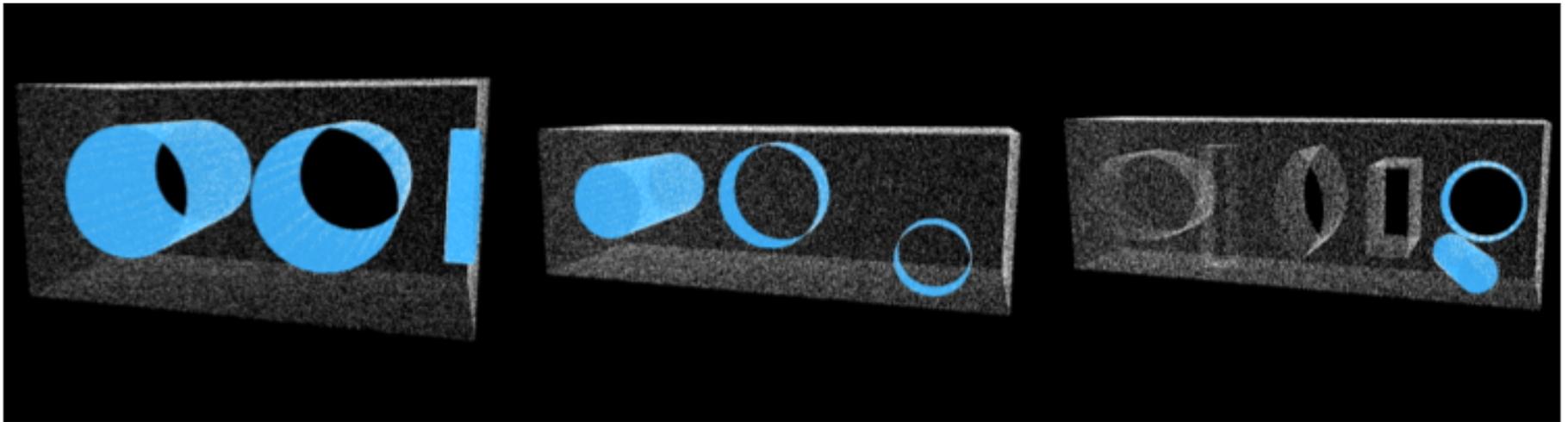
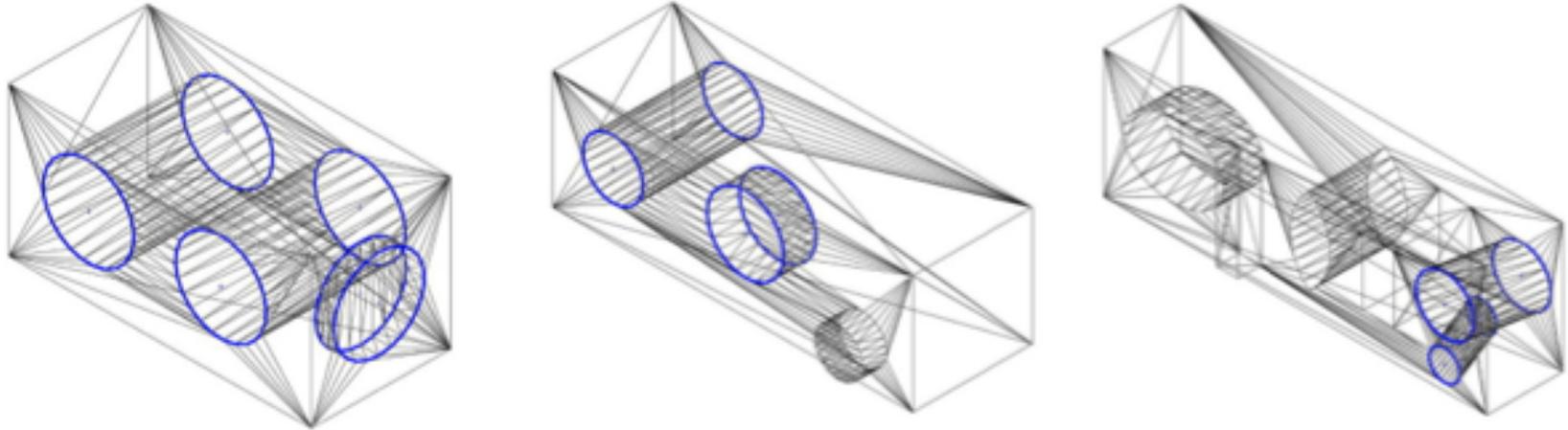


Vereinigung $A \cup B$ (Summe)

Schnitt $A \cap B$



Gradient der Informationskompression der Modelltypen klar zu erkennen



Anwendungsgebiete für das geometrische Modell

- Punktwolke:
 - Lokalisation, Klassifikation
 - Kartierung (mobile Systeme)
- Mesh:
 - Bewegungsplanung (v.a. Manipulatoren)
 - Dynamische Simulation (komplexe Starrkörper)
- Voxel/Octree:
 - Bewegungsplanung (v.a. mobile Systeme)
 - Dynamische Simulation elastische Materialien (FEM)
- CSG:
 - CAD/CAE/CAM
 - Dynamische Simulation (einfache Starrkörper)

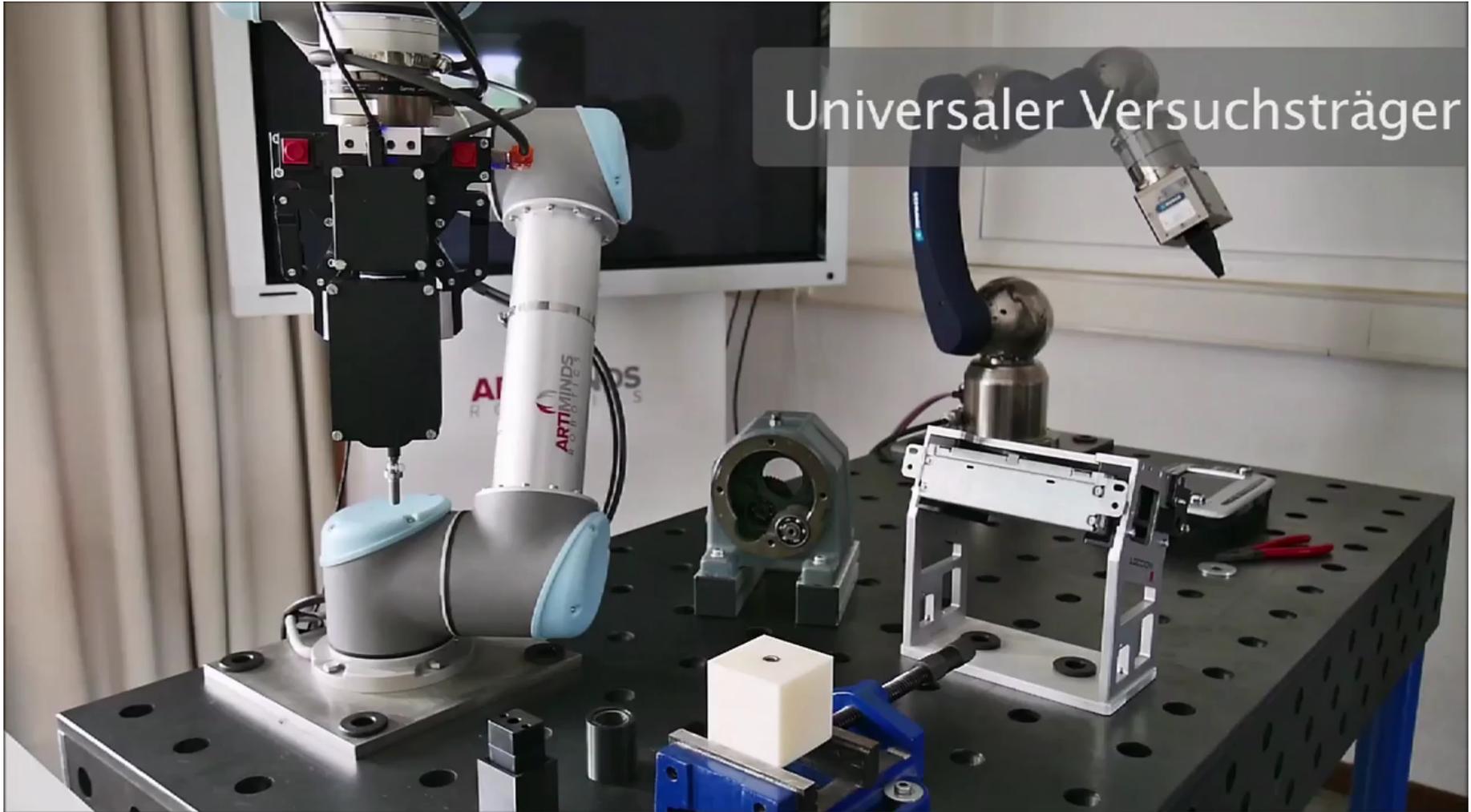
- Motivation: Adaptive Roboterarbeiten
- **Objektmodelle**
 - Geometrische Beschreibung
 - **Zusätzliche Eigenschaften**
- Szenenmodelle

Eigenschaften des Objektes mit Geometriebezug

- Masse
 - Oberflächeneigenschaften (z.B. Reibung)
 - Temperatur
 - Steifigkeit

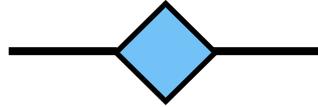
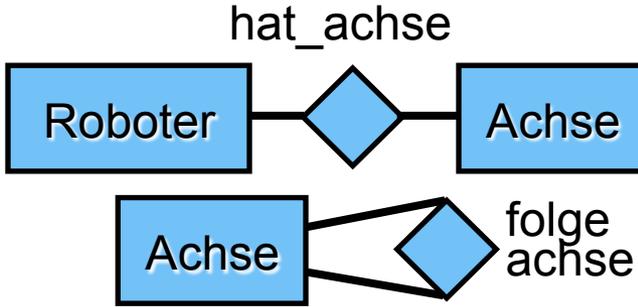
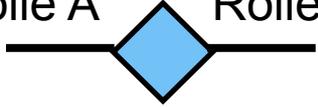
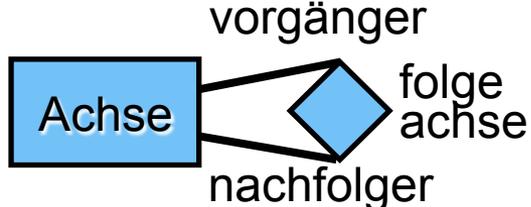
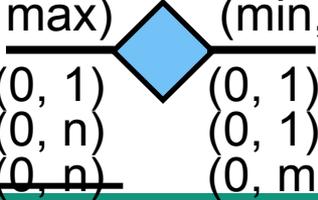
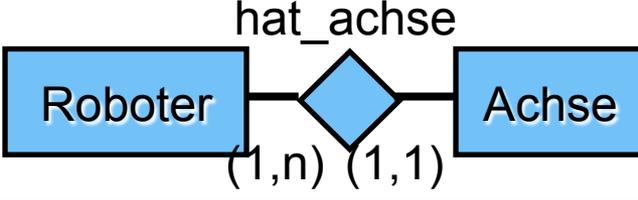
 - Greifpunkte
 - Verbindungspunkte zur Montage
 - Ablagepunkte bei Paletten
 - Füllmenge bei Paletten
 - Stellung bezüglich eines Referenzobjektes
- Stark aufgabenabhängig!**

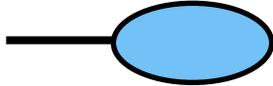
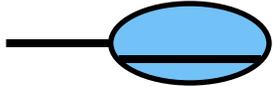
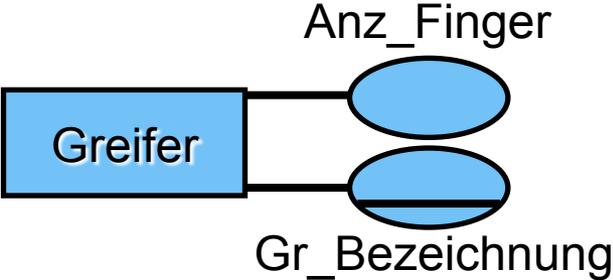
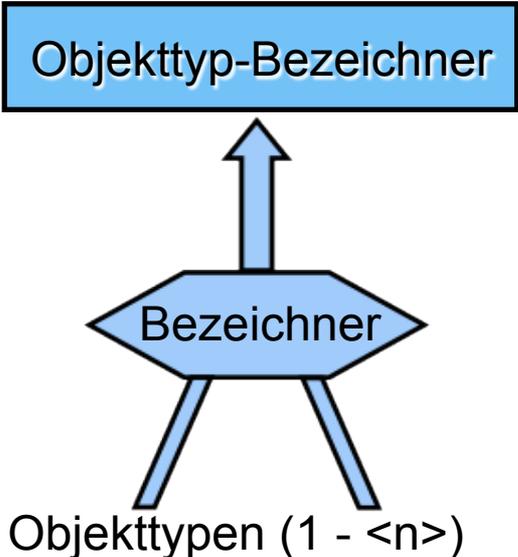
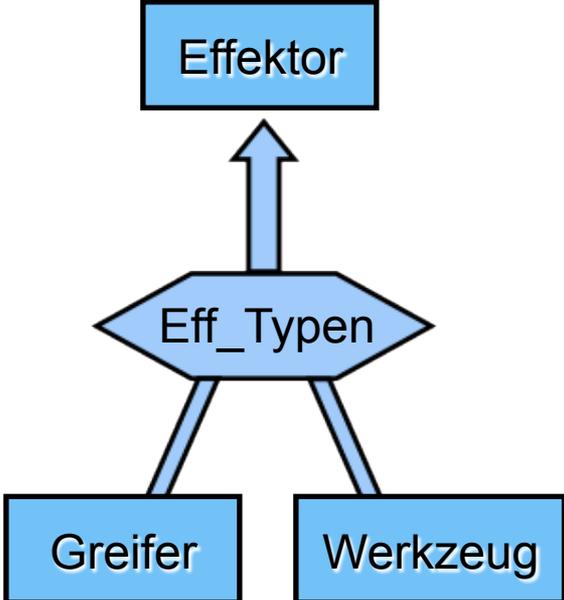
Beispiele zusätzliche Eigenschaften

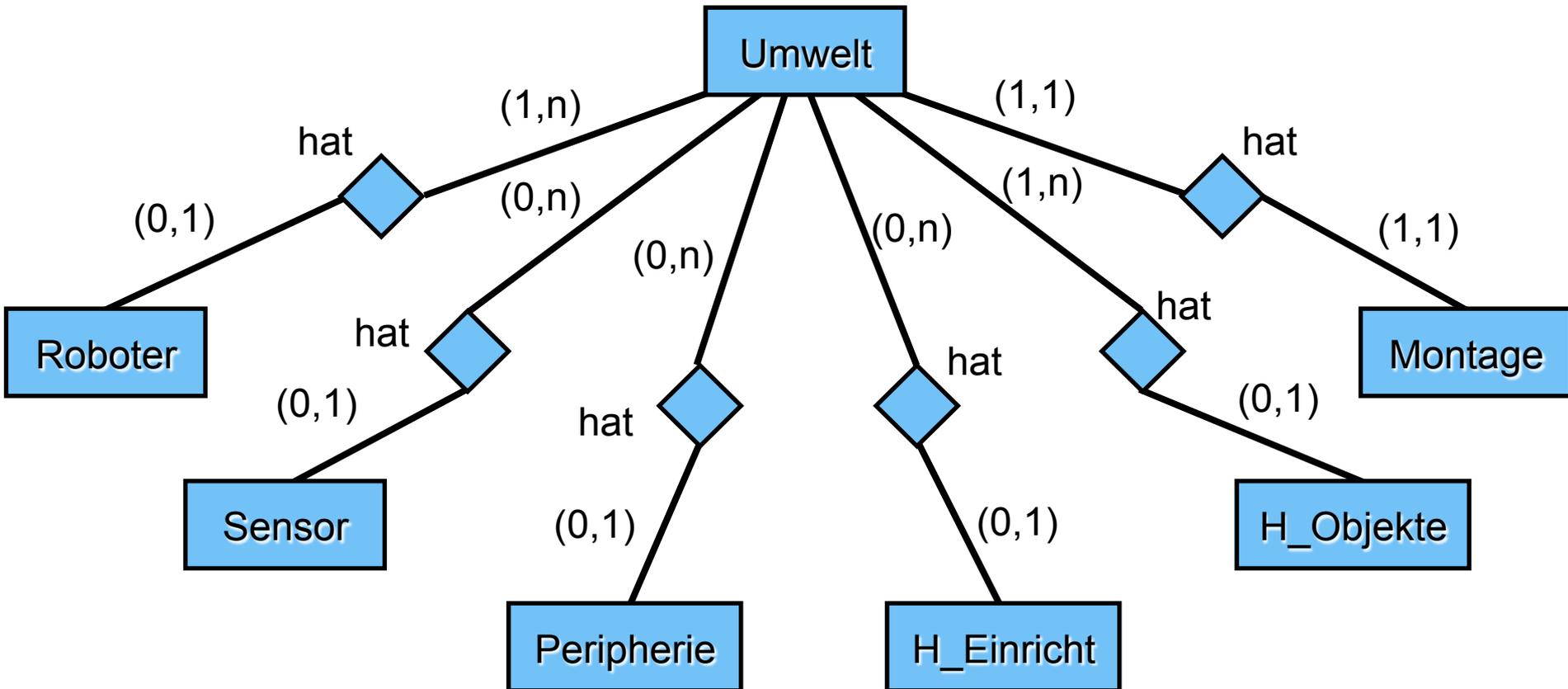


- Motivation: Adaptive Roboterarbeiten
- Objektmodelle
 - Geometrische Beschreibung
 - Zusätzliche Eigenschaften
- **Szenenmodelle**

- Entity-Relationship-Modelle
- Semantische Netze
- Frame Modelle nach Minsky
- Implicit Shape Models (ISMs)
- Probabilistische Object Constellation Models (OCMs)

Modellierkonstrukt	Graphisches Symbol	Beispiel
Objekttyp (Entity set)		
Beziehungstyp (Relationship type) <ul style="list-style-type: none"> ○ zweiseitig ○ rekursiv 	Beziehungstyp-bezeichner 	
Rollen	Rolle A Rolle B 	
Kardinalität		

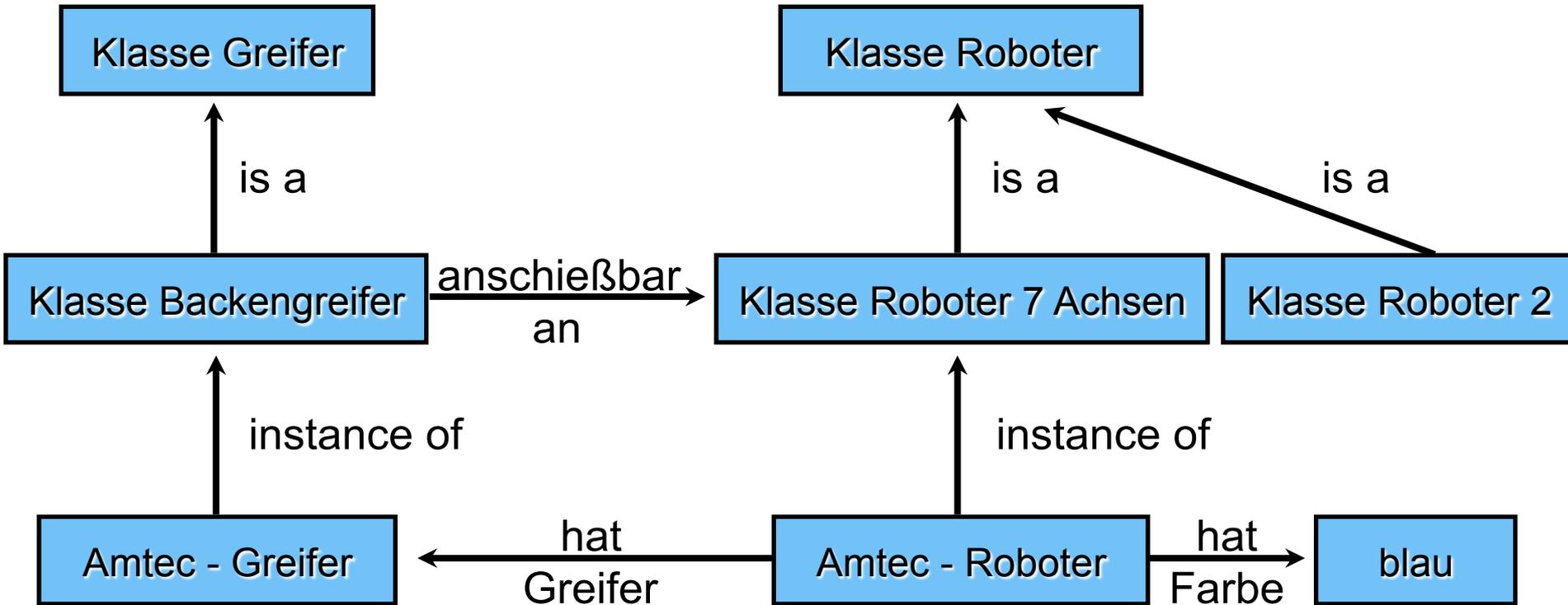
Modellierkonstrukt	Graphisches Symbol	Beispiel
<p>Attribute</p> <ul style="list-style-type: none"> ○ beschreibende ○ identifizierende 	<p>Attributbezeichner</p>  <p>Attributbezeichner</p> 	
<p>Generalisierungshierarchie</p>		



Konzept: Objekte und Beziehungen

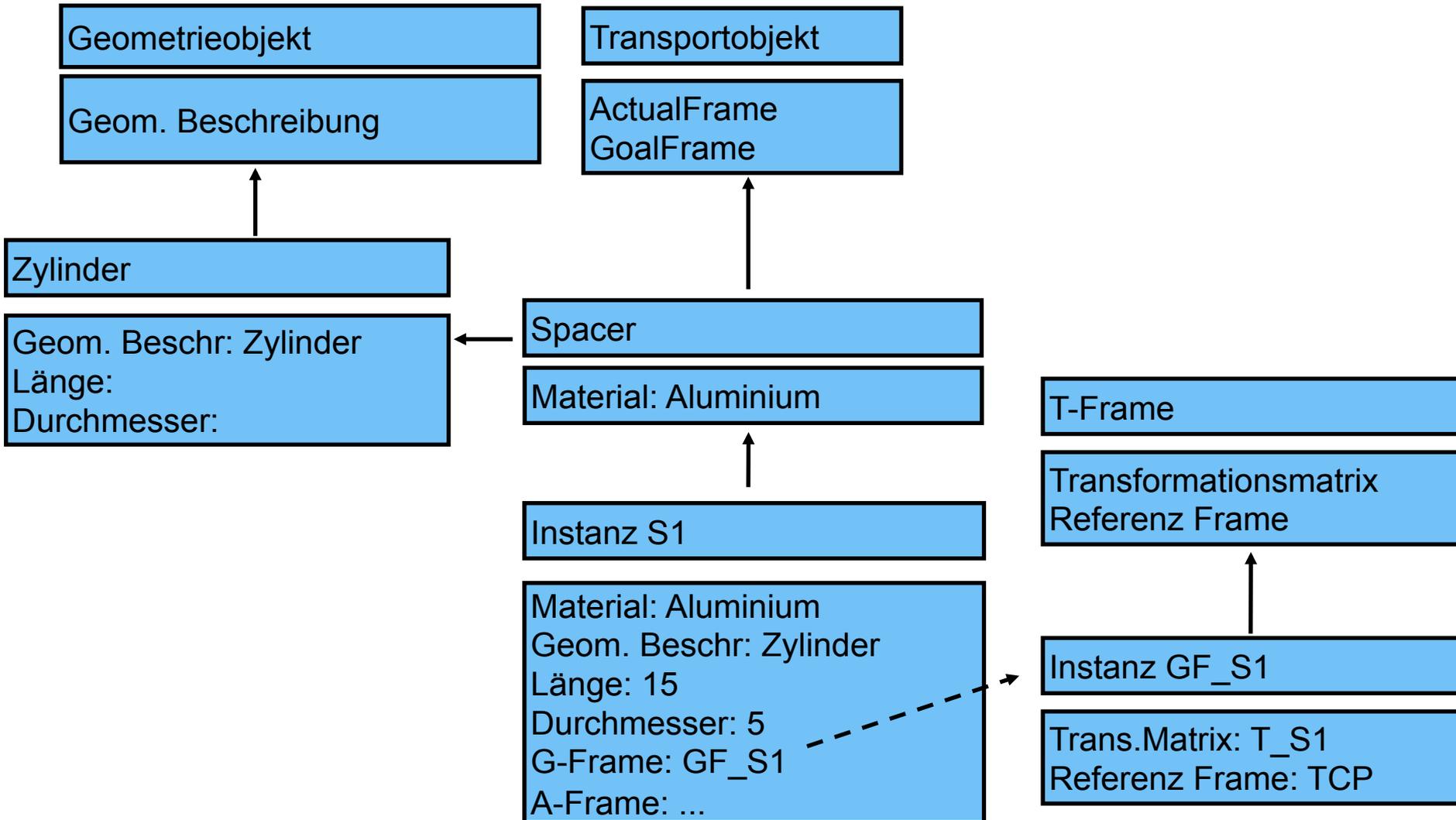
- Semantisches Netz = gerichteter Graph
- Knoten = Objektklasse oder Einzelobjekt
- gerichtete und benannte Kanten = Beziehungen

- nur zweistellige Beziehungen
- mehrstellige Beziehungen => als eigenes Objekt
- keine Attribute => Attribute als „Wertobjekt“



- Frame = Schablone (nicht mit Koordinaten-Frames verwechseln)
 - Erfassung der:
 - Eigenschaften von Objekten
 - Einordnung in Hierarchie von Objektklassen
 - Leichte Implementierung mit objektorientierten Sprachen
 - Frame:
 - beschreibt ein Objekt / Objektklasse
 - enthält eine Menge von Slots
 - Slots enthalten:
 - Attribute
 - Verweise auf andere Frames
- Instanzen, Vererbung

Beispiel eines Frame Modells

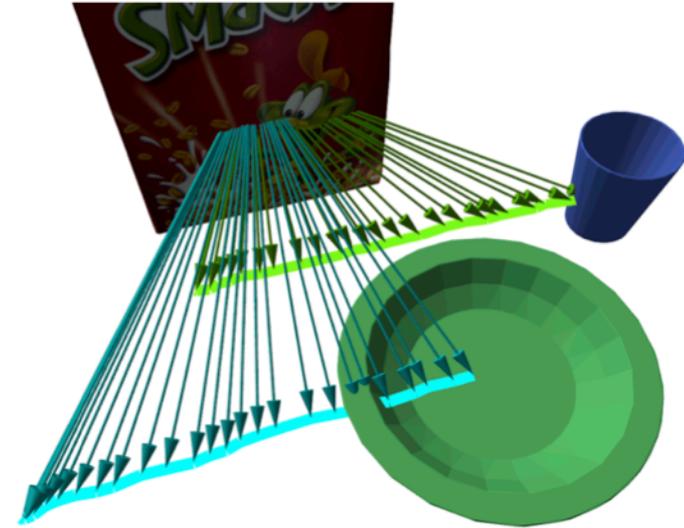


Moderne Szenenbeschreibung, robust bei Varianzen

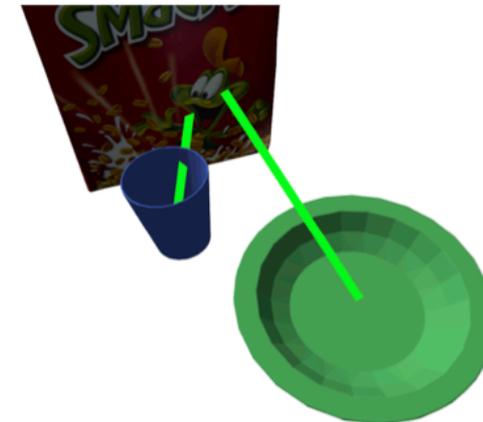
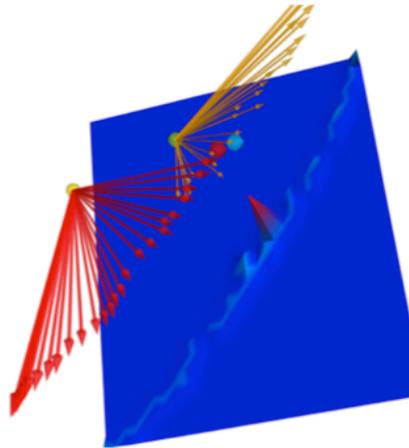
- Szene besteht aus relativen Transformationsrelationen zwischen Objekten
- ISM ist Variation der Generalisierten Hough Transformation (Maschinensehen)
- Daraus folgt, dass Instanzen der Relationen in *buckets* abstimmen (*voten*)
- Baumartige Topologie der relativen Relationen

Hough voting findet im 3D-Raum statt

- Votes werden in einem Voxel-Gitter akkumuliert



- Weitere Informationen:
Meißner & al.

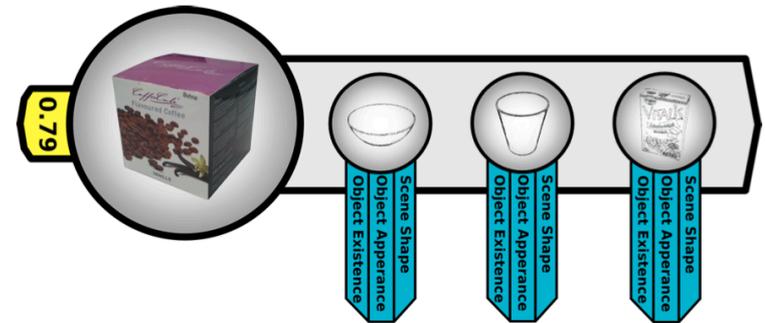
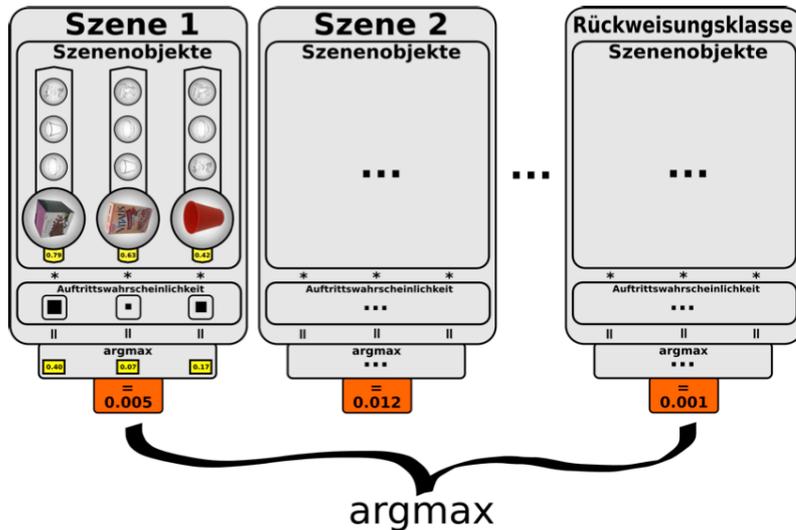


Active Scene Recognition for Programming by Demonstration using Next-Best-View Estimates from Hierarchical Implicit Shape Models

Pascal Meißner, Reno Reckling, Valerij Wittenbeck, Sven R. Schmidt-Rohr and Rüdiger Dillmann

Kurzvorstellung OCM: modernes probabilistisches Modell

- Probabilistische Szenenrelationen
- Basiert auf Gauss-Mixturen (GMMs) und probabilistischer Graph-Inferenz



5.6: Das Object Constellation Modell beschreibt ein einzelnes Szenenobjekt, also ein Objekt (hier die Kaffeebox) im Kontext einer Szene. Die Slots werden durch die kleinen Kreise rechts symbolisiert, jeder Slot beschreibt ein Objekt der Szene mit den in blau dargestellten Parametern. Der gelbe Kasten gibt an, mit welcher Wahrscheinlichkeit das Szenenobjekt vorliegt.

Weitere Infos: Gehring, Meißner & al.

5.4: Eine grafische Darstellung des Szenenmodells. Es wird die Wahrscheinlichkeit einer Reihe von Szenen inklusive der Rückweisungsklasse berechnet. Jede Szene basiert auf einer Reihe von Szenenobjekten, wobei das mit der höchsten gewichteten Wahrscheinlichkeit die Wahrscheinlichkeit der Szene ergibt.

- Bahn- und Greifplanung
- Interaktive Programmierung
- Autonome Roboter:
Serviceroboter, Softwarearchitekturen und
symbolische Aufgabenplanung
- Praxisvorlesung