Kapitel 12

3D-Repräsentation

Die Repräsentation dreidimensionaler Objekte in der Computergrafik teilt sich in verschiedene Repräsentationsklassen, die hierarchisch aufgebaut sind.

12.1 Elementarobjekte

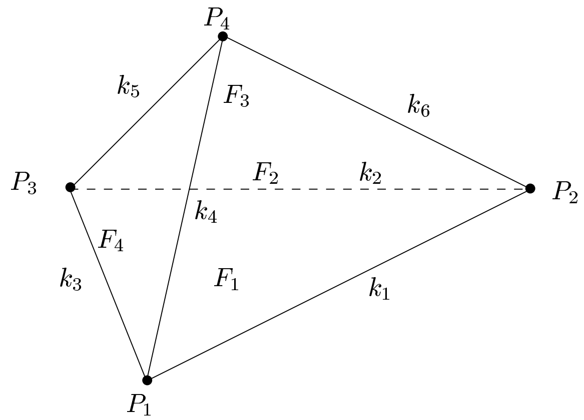
Für den Benutzer sollte die Beschreibung einer Szene durch *Elementarobjekte* erfolgen. Diese können unterschiedlich kompliziert sein, sollten aber durch wenige Parameter beschrieben werden können. Eine Kugel z.B. ist bereits durch Mittelpunkt und Radius eindeutig im Raum plaziert. Es ist sinnvoll, jedes Objekt in seinem eigenen, lokalen Modellkoordinatensystem zu definieren. Dessen Ursprung wird im Inneren des Objekts gewählt und das gesamte Objekt in ein Einheitsvolumen (z.B. $-1 \leq x, y, z \leq +1$) eingeschlossen. Für Orts- und Größenveränderungen sind Transformationen zuständig.

12.2 Drahtmodell

In der nächsten Repräsentationsklasse wird das Elementarobjekt als *Drahtmodell* durch eine Liste von Kanten dargestellt. Jede Kante besteht aus zwei Punkten im kartesischen Koordinatensystem.
Beim Würfel verbinden die Kanten die Eckpunkte, bei einer Kugel werden die Längen- und Breitenkreise durch n-Ecke angenähert, wobei mit n die Güte der Approximation steigt. Das Drahtmodell skizziert nur die Umrisse eines Objekts und enthält keine zusätzlichen Flächen- oder Volumeninformationen.

12.3 Flächenmodell

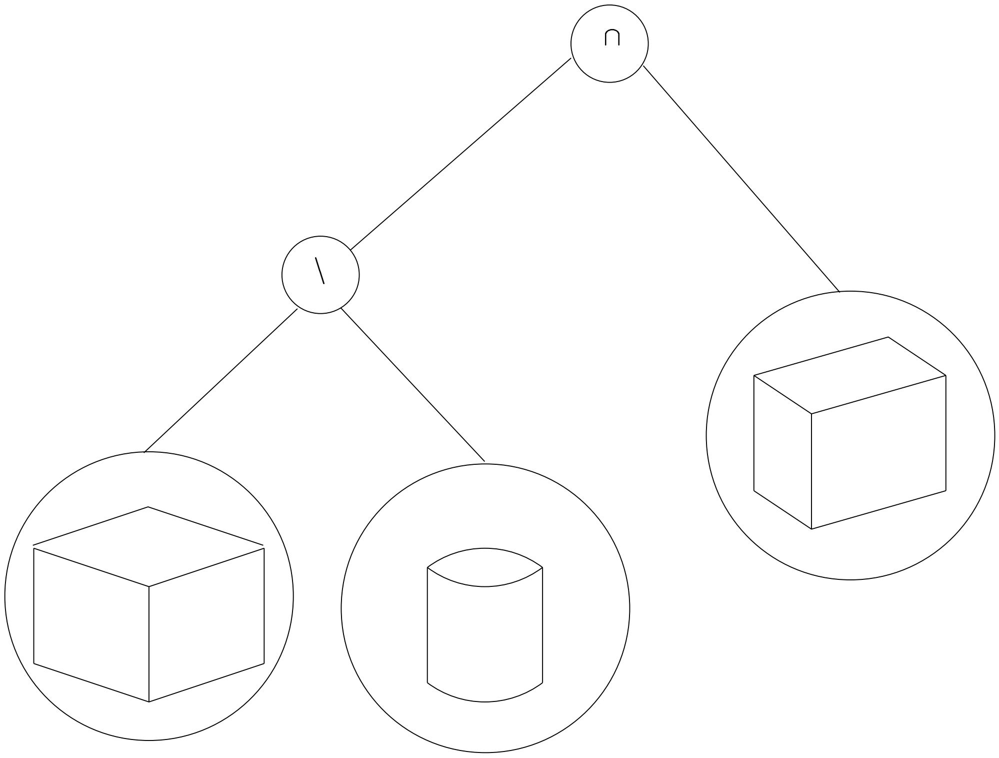
Beim *Flächenmodell* werden Objekte durch approximierte oder analytische Flächen, z.B. durch eine Liste von konvexen Polygonen, dargestellt. Ein solches Polygon wird beschrieben durch seine Eckpunkte, die durch Kanten verbunden sind.
Zur vollständigen Beschreibung einer Fläche gehört noch die Angabe, welche Seite "innen" und welche Seite "außen" liegt. Dies geschieht durch Angabe des Normalenvektors: Er steht senkrecht auf der Fläche und zeigt von innen nach außen. Für die Approximation gekrümmter Flächen wird häufig pro Eckpunkt eine Normale verwendet.
Beispiel: Tetraeder als Flächenmodell



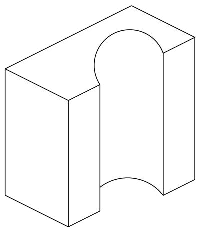
Punkteliste	Kantenliste	Flächenliste
$P_1 : (x_1, y_1, z_1)$	$k_1 : P_1, P_2$	$F_1 : k_1, k_2, k_3$
$P_2 : (x_2, y_2, z_2)$	$k_2 : P_2, P_3$	$F_2 : k_1, k_6, k_4$
$P_3 : (x_3, y_3, z_3)$	$k_3 : P_3, P_1$	$F_3 : k_2, k_6, k_5$
$P_4 : (x_4, y_4, z_4)$	$k_4 : P_1, P_4$	$F_4 : k_3, k_4, k_5$
	$k_5 : P_4, P_3$	
	$k_6 : P_4, P_2$	

12.4 CSG (constructive solid geometry)

Jedes Objekt wird beschrieben durch einen binären Baum, dessen Blätter beschriftet sind mit Elementarobjekten und dessen innere Knoten beschriftet sind mit den Mengenoperationen $\cup$ (Vereinigung), $\cap$ (Durchschnitt), $\setminus$ (Differenz).

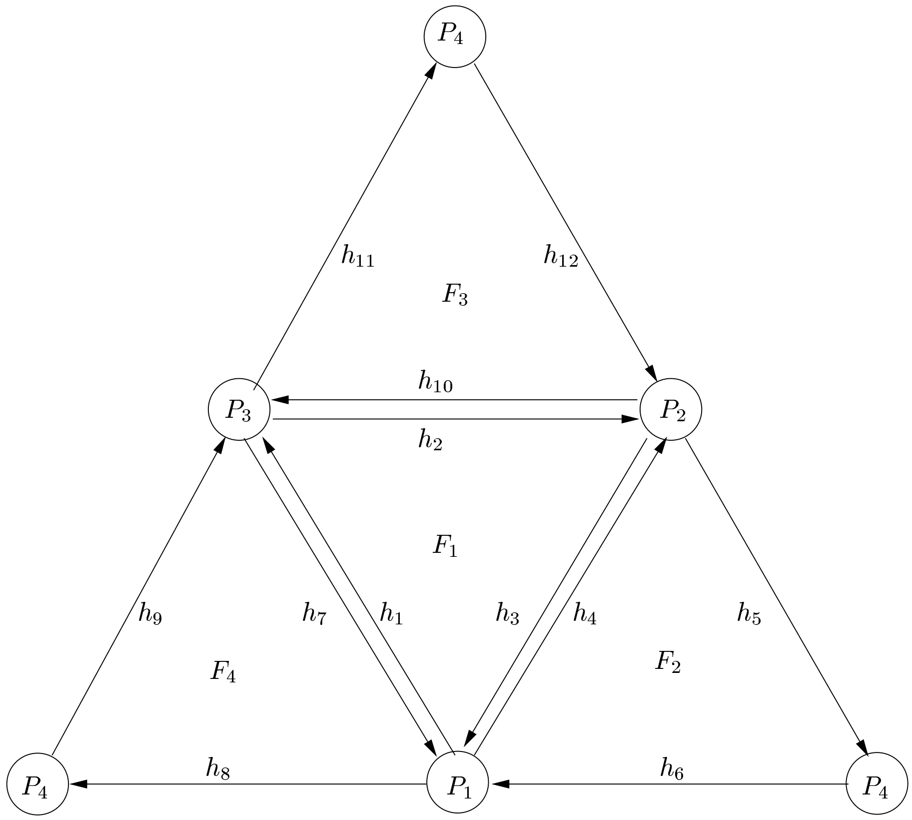


bezeichnet das folgende Objekt:

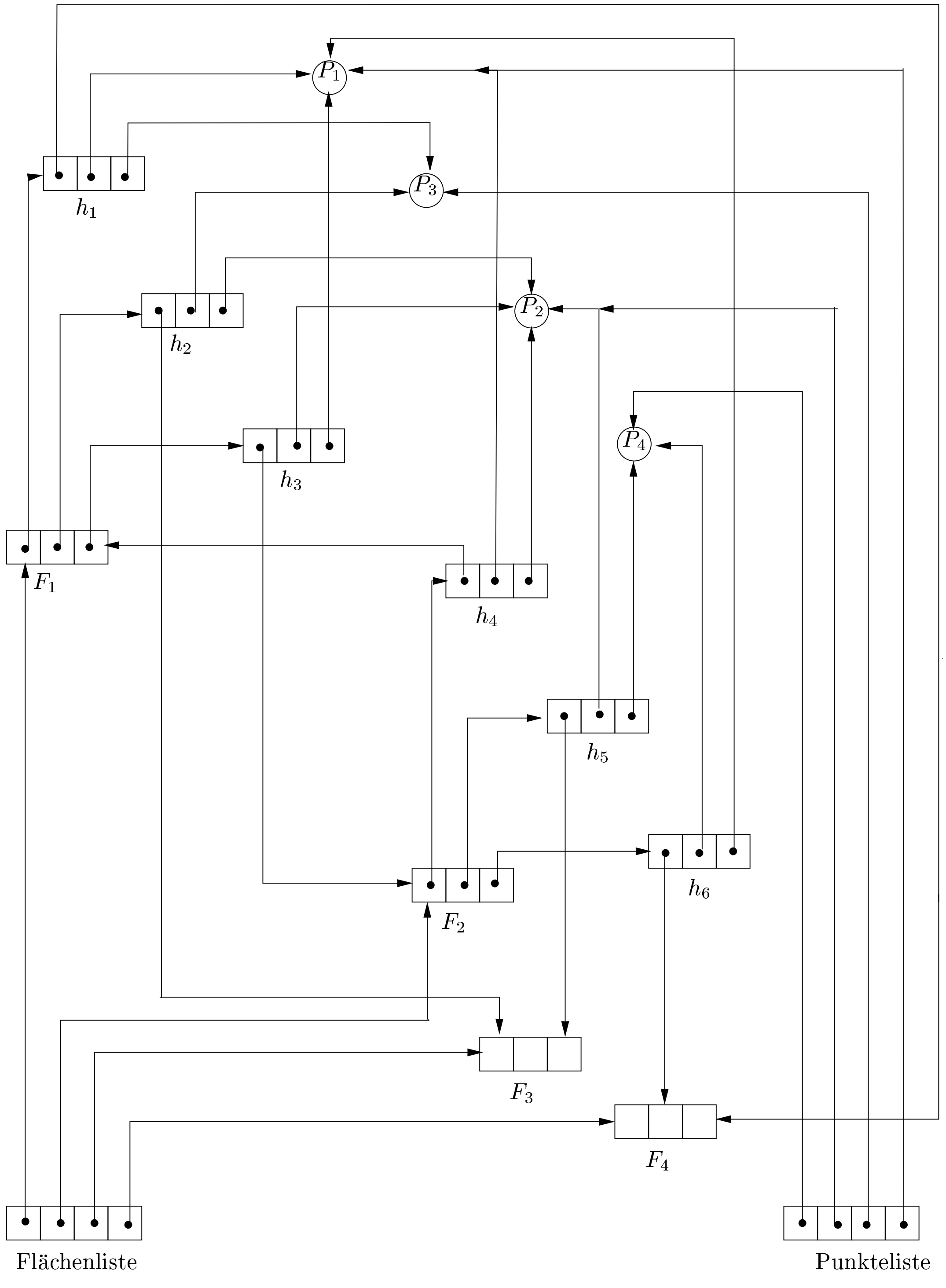


12.5 Flächenmodell mit Halbkantendarstellung

Jedes Objekt enthält eine Liste von Flächen, die von Halbkanten begrenzt werden. Die Halbkanten sind von außen betrachtet im Uhrzeigersinn orientiert und zeigen auf ihre jeweils linke Nachbarfläche sowie ihre Anfangs- und Endpunkte.
Die Halbkantendarstellung eignet sich zur effizienten Entfernung von verdeckten Kanten und Flächen. Eine Kante zwischen Punkt P_1 und Punkt P_2, welche die Flächen F_1 und F_2 trennt, taucht einmal als Halbkante (P_1, P_2) in der Kantenliste zu F_2 auf mit einem Verweis auf die Nachbarfläche F_1 und ein weiteres Mal als (P_2, P_1) in der Kantenliste zu F_1 mit einem Verweis auf die Nachbarfläche F_2. Werden nun alle Halbkanten einer Fläche F_1 bearbeitet, so regelt die Sichtbarkeit von F_1 und die Sichtbarkeit der jeweils anstoßenden Fläche die Sichtbarkeit der jeweiligen Halbkante. Das doppelte Zeichnen einer Kante läßt sich vermeiden, indem bei jeder Fläche vermerkt wird, ob ihre Halbkanten bereits bearbeitet wurden.
Auch die Flächennormalen können in der Datenstruktur gespeichert werden.



Tetraeder mit 4 Knoten, 12 Halbkanten, 4 Flächen



Objektstruktur für Tetraeder. Nicht gezeichnet sind die Halbkanten der Flächen F_3 und F_4.