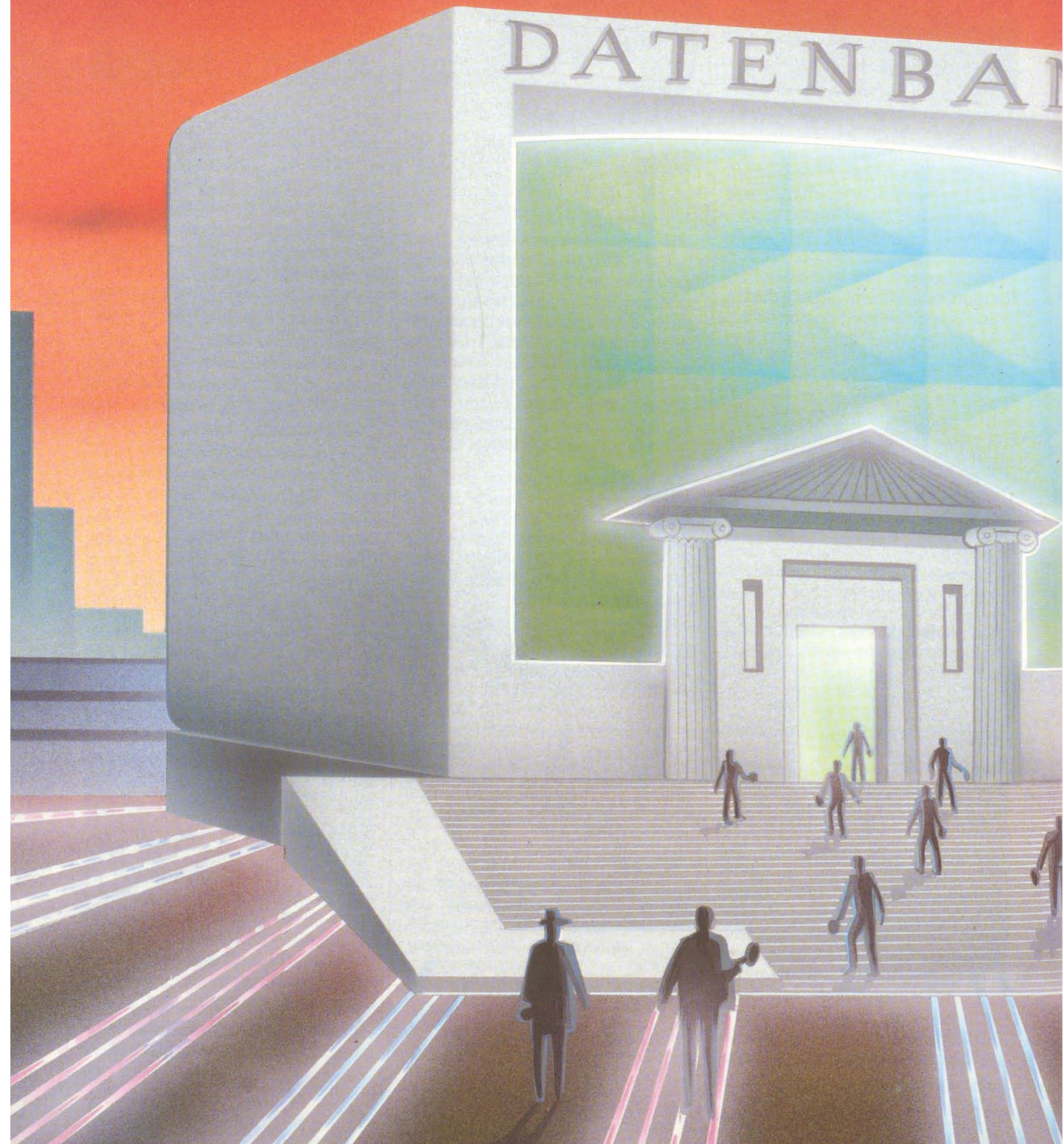


# Struktur und Verwaltung grafischer Daten

Dipl.-Inform. Christoph Hübel  
Dipl.-Inform. Bernhard Mitschang  
Universität Kaiserslautern



**D**ie Arbeitsgruppe Datenverwaltungssysteme am Fachbereich Informatik der Universität Kaiserslautern befaßt sich schon seit längerem im Rahmen des Sonderforschungsbereichs »VLSI-Entwurfsstrukturen und Parallelität« mit der Thematik des Datenbankeinsatzes in sogenannten »Nicht-Standard«-Datenbankanwendungen. Seit kurzem werden diese Forschungsaktivitäten noch ergänzt durch das neugeschaffene »Zentrum für Rechnergestützte Ingenieurssysteme«. Innerhalb dieser Projekte wurden bereits verschiedene datenbankbasierende Software-Prototypen aus den Bereichen VLSI-Entwurf, geografische Informationssysteme, Expertensysteme und rechnergestützte Konstruktion entwickelt. Dabei trat überall die Problematik der Strukturierung und Verwaltung grafischer Daten zum Vorschein. Dies gab Anlaß zu einer systematischen Untersuchung des Zusammenspiels von Grafik- und Datenbanksystem, deren Ergebnisse in der Folge dargelegt werden.

Bevor im folgenden auf die Probleme bei der Strukturierung und Verwaltung grafischer Daten näher eingegangen wird, sollen zunächst einige wichtige Gründe aufgeführt werden, die die Notwendigkeit einer effizienten Zusammenarbeit gerade zwischen Grafik- und Datenbanksystem untermauern.

Betrachtet man die verschiedenen Bereiche, in denen Grafiksysteme zum Einsatz gelangen, zum Beispiel die Prozeßdatenverarbeitung, viele Simulationsanwendungen, die medizinische Datenverarbeitung und vor allem das große Feld der rechnerunterstützten Ingenieur Anwendungen (unter anderem CAD/CAM), so lassen sich eine Reihe von Gemeinsamkeiten erkennen. In den beispielhaft aufgeführten Disziplinen spielt die Schnittstelle zu einem menschlichen Benutzer eine wesentliche,



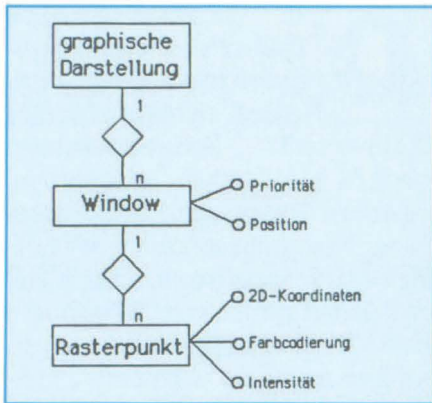


Bild 1: Rasterdaten im ER-Modell (Entity/Relationship) zur Daten-Strukturierung.

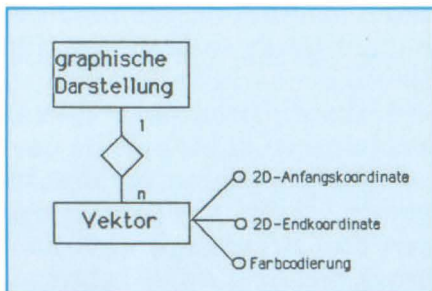


Bild 2: Eine grafische Darstellung ist durch eine Vektormenge bestimmt.

wenn nicht eine für die Akzeptanz des Systems entscheidende Rolle. Die hohe Informationsdichte, die dem Benutzer angeboten wird, kann nur noch mit grafischen Mitteln dargestellt werden. Der Vielzahl der bereitgestellten Informationen entspricht ein großes, systeminternes Datenvolumen, so daß die meisten Datenanwendungen als *datenintensiv* bezeichnet werden können.

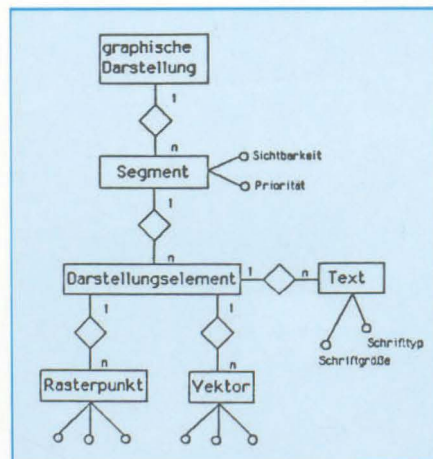
Datenbankbasierende Anwendungen sind bislang durch Aufgaben aus dem administrativ-betriebswirtschaftlichen Bereich geprägt. Beispielhaft hierfür stehen viele erfolgreich eingesetzte Datenbanksysteme zur Verwaltung von Lager, Kundenstamm und Finanzen, aber auch eine Vielzahl von Buchungs- und Reservierungssystemen. Neue Anforderungen an Datenbanksysteme kommen überwiegend aus dem Bereich der Ingenieuraufgaben. Gerade hier gibt es ein breites Spektrum von sehr heterogenen Anwendungen. Aus Sicht der Datenbankforschung handelt es sich hierbei um sogenannte »Nicht-Standard«-Anwendungen, deren optimale Unterstützung durch neu zu entwickelnde Datenbankkonzepte erreicht werden soll /HR85/. Eine Gemeinsamkeit der Nicht-Standard-Daten-

bankanwendungen scheint darin zu bestehen, daß sie ebenfalls über eine hohe Informationsdichte an der Benutzerschnittstelle verfügen. Man kann sie daher auch als *darstellungsintensive* Anwendungen einstufen.

Anhand dieser knappen Skizze typischer Grafik- und neuerer Datenbankanwendungen gewinnt die Forderung nach einer effizienten Kooperation zwischen beiden Systemen zunehmend an Bedeutung. Zukünftige (daten- und darstellungsintensive) Anwendungssysteme benötigen zur Erfüllung ihrer Aufgaben sowohl Grafik- als auch Datenbanksysteme.

Typische Vertreter solcher Systeme bilden Anwendungen aus dem CAD-Umfeld. Dies verwundert nicht, wenn man bedenkt, daß hauptsächlich die Anforderungen aus dem CAD-Bereich sowohl zur Standardisierung grafischer Software geführt als auch die Ausgangsbasis bestimmt haben. Eine der dringlichsten CAD-Forderungen, die bislang nicht oder nur unzureichend erfüllt werden kann, besteht in der Austauschbarkeit systeminterner Informationen zwischen den unterschiedlichen CAD-Anwendungen. Der grafikorientierte Metafile-Ansatz (GKS-Metafile, /GKS83/) kann als nützlich angesehen werden, er ist allerdings nicht befriedigend, da nur grafische Daten ausgetauscht werden können. Ein zweiter Ansatz, der Datenaustausch über IGES-Dateien /IGES83/, erlaubt zwar eine gewisse Produktorientierung wie auch die Übermittlung von teilweise nicht-grafischen Daten. Aber auch IGES kann nicht als Antwort auf alle CAD-Anforderungen verstanden

Bild 3: GKS-Informationsstruktur anhand eines Entity/Relationship-Diagramms.



werden. Vielmehr ist die Strukturierung und die zentrale Verwaltung aller produktbezogenen Daten anzustreben, um somit eine Integration verschiedener CAD-Dienste zu ermöglichen. Die Domäne der Datenbanksysteme liegt aber gerade bei der anwendungsunabhängigen Strukturierung und Verwaltung anwendungsabhängiger Daten.

**GKS-Metafile-Ansatz ist nicht befriedigend**

Unter Berücksichtigung all dieser Aspekte bilden CAD-Anwendungen ein ausgezeichnetes Experimentier- und Prüffeld, auf dem das Zusammenspiel zwischen Grafik- und Datenbanksystem untersucht werden kann. Eine solche Betrachtung soll an einem Prototypsystem, einem 3D-Bauteilmodellierer für

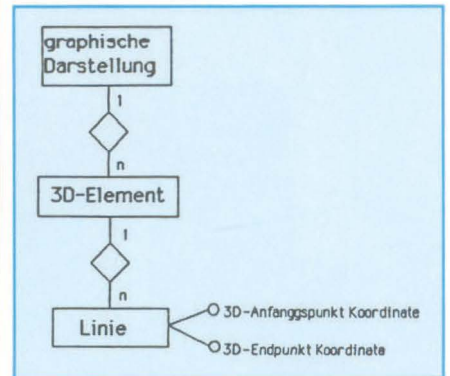


Bild 4: ER-Diagramm einer linienorientierten Datenstruktur.

polyederförmige Werkstücke, erläutert werden, nachdem durch allgemeine Betrachtungen zur Strukturierung grafischer Daten eine Verständnisgrundlage geschaffen wurde.

**Struktur grafischer Daten:** Unter grafischen Daten sollen alle diejenigen Daten verstanden werden, aus denen eine grafische Darstellung abzuleiten ist. Prinzipiell sind damit alle Daten auch als grafische Daten aufzufassen, jedoch muß in einer konkreten Anwendung die Darstellung eines Datums nicht notwendigerweise eine Grafik sein. Anwendungsspezifisch lassen sich daher sehr wohl grafische und nicht-grafische Daten unterscheiden.

Zur Beschreibung der grafischen Datenstruktur wird im folgenden eine Darstellung nach dem Entity-Relationship-Modell (ER-Modell) verwendet. Das ER-Modell /Ch80/

dient zur Strukturierung der in einem konkreten Weltausschnitt vorherrschenden Information. Die Gegenstände dieses Weltausschnittes werden auf »Entities«, ihre Beziehungen auf »Relationships« abgebildet. Die Eigenschaften dieser Gegenstände und ihrer Beziehungen können durch Entity- und Relationship-Attribute dargestellt werden. Gleichartige Gegenstände beziehungsweise damit assoziierte Entities werden zu sogenannten »Entity-Mengen« (Typen) zusammengefaßt. Entsprechendes gilt für die Beziehungen (Relationships). Bei den Beziehungen unterschei-

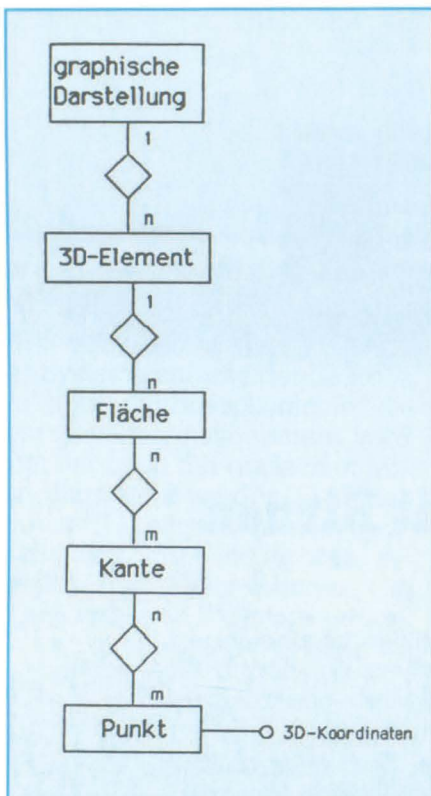


Bild 5: Entity/Relationship-Diagramm für flächenorientierte Datenstrukturen.

det man drei Arten: (1:1), (1:n) und (n:m). Damit werden eindeutige, funktionale und komplexe Beziehungen zwischen Entities charakterisiert. Zur grafischen Darstellung eines ER-Modells verwendet man im allgemeinen ER-Diagramme, die aus Rechtecken, Rauten und Verbindungslinien bestehen. Hierbei stehen die Rechtecke für einen Entity-Typ, die Rauten für einen Relationship-Typ. Die Rechtecke der an einer bestimmten Relationship beteiligten Entity-Typen sind durch Linien mit der entsprechenden Relationship-Raute verbunden. An den Linien befindet sich eine Beschriftung, die die entsprechen-

de Beziehungsart ((1:1), (1:n) (n:m)) bestimmt. Die in den Bildern 1 bis 5 dargestellten Diagramme sollen keineswegs die vollständige Information, sondern lediglich die prinzipielle Struktur grafischer Daten aufzeigen.

Die einfachste Art grafischer Daten sind Rasterdaten. Ihre Ableitung zu einer entsprechenden grafischen Darstellung ist extrem einfach: jedem Bildpunkt einer grafischen Darstellung ist ein Rasterpunkt zugeordnet. Die Attribute der Rasterpunkte, wie Farbcodierung oder Intensitätsstufe, bestimmen das gesamte Bild. Die Ableitung gestaltet sich etwas komplizierter, wenn es sich um strukturierte Rasterdaten handelt, wobei jede Teilstruktur zu einem Bildausschnitt abgeleitet wird. Da sich verschiedene Teilstrukturen und damit verschiedene Bildausschnitte überlappen können, müssen bei der Herleitung einer grafischen Darstellung Teilstrukturprioritäten berücksichtigt werden. Bild 1 zeigt die schematische Darstellung von einstufig strukturierten Rasterdaten nach dem Entity-Relationship-Modell. Die Teilstrukturen sind hierbei durch Entities vom Typ »Window« repräsentiert.

### CAD-Anforderungen erzwingen neue Datenbankkonzepte

Eine weitere Art grafischer Daten ist durch die Vektordaten gegeben. Ihre grafische Darstellung kann abgeleitet werden, indem ein Anfangspunkt mit einem Endpunkt durch eine Linie verbunden wird. Können diese Vektordaten direkt vom grafischen Ein-/Ausgabegerät interpretiert werden (Vektorgeräte), so ist die Ableitung entsprechend einfach. Ist dies nicht der Fall, so müssen die Vektordaten zunächst auf Rasterdaten abgebildet werden. Wie an dem Entity-Relationship-Modell für Vektordaten zu erkennen ist (Bild 2), ist eine grafische Darstellung durch eine Vektormenge bestimmt. Eine Überlappung, ähnlich wie bei den Rasterdaten, erscheint hier nicht sinnvoll, da es sich bei den Linien prinzipiell um nicht-flächenhafte Objekte handelt.

Eine dritte Art grafischer Daten stellt eine Kombination der beiden

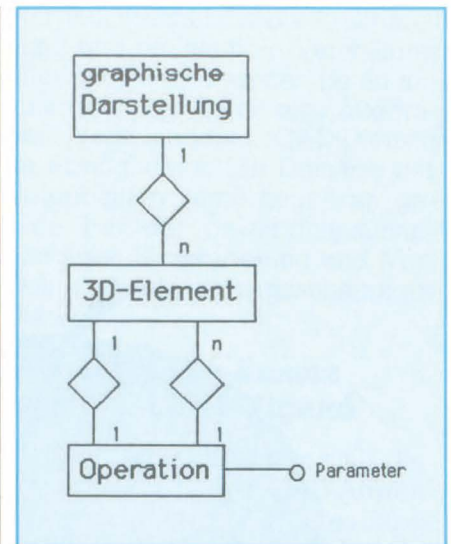


Bild 6: Körperorientierte Strukturen bieten eine natürliche Objekt-Beschreibung.

ersten Typen dar. Die Grundlage einer grafischen Darstellung besteht in sogenannten Darstellungselementen. Diese können sich entweder wiederum aus Rasterbeziehungsweise Vektordaten zusammensetzen oder aus impliziten Ableitungsvorschriften bestehen, durch die dann Raster- oder Vektordaten erzeugt werden. Darstellungselemente können zu komplexeren Strukturen zusammengefaßt werden, wobei sich die Attribute der Teilstrukturen, ähnlich wie bei den strukturierten Rasterdaten, auf die Eigenschaften der zugehörigen Darstellungselemente auswirken. Das standardisierte grafische Kernsystem (GKS) verfügt über solche Darstellungsprimitive /GKS83/. GKS erlaubt eine einstufige Strukturierung durch sogenannte Segmente. Bild 3 zeigt die GKS-Informationsstruktur anhand eines Entity-Relationship-Diagramms.

### Prinzipiell sind alle Daten als »grafische« Daten aufzufassen

Die Unterschiede der drei Arten grafischer Daten liegen im zugehörigen Ableitungsprozeß, das heißt, in den Verfahren, mit denen eine entsprechende grafische Darstellung abgeleitet werden kann. Je »entfernter« ein grafisches Datum von der Bilddarstellung einer Grafik ist, desto aufwendiger gestaltet sich das entsprechende Ableitungsverfahren. Hingegen fällt es oft leichter, in einer abstrakteren Beschreibung komplizierte grafische Elemente darzustellen. So

wird zum Beispiel aus einem Kreis, der in Rasterdarstellung durch unzählige, einzeln berechnete Punkte beschrieben wird, ein einziges Darstellungselement, das durch Mittelpunkt und Radius bestimmt wird.

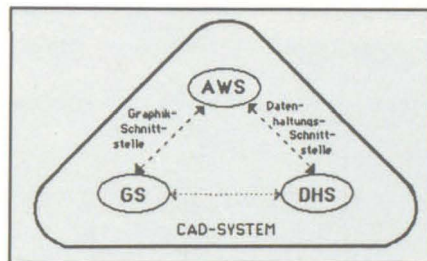
Bei der bisherigen Beschreibung grafischer Daten wurde implizit eine 2D-Orientierung vorausgesetzt. Dies erscheint in einem gewissen Sinne natürlich, da man im allgemeinen mit einer Grafik eine 2D-Darstellung assoziiert. Nach der vorherigen Begriffsbildung für grafische Daten ist diese Einschränkung allerdings unnötig. So sind zum Beispiel 3D-Raster- und Vektordaten beziehungsweise 3D-Darstellungselemente denkbar (die 3D-GKS-Schnittstelle besitzt solche Beschreibungselemente). Aus Sicht der notwendigen Ableitungsverfahren kommt hiermit ein zusätzlicher Komplexitätsgrad ins Spiel, denn zur Darstellung auf einer 2D-Bildebene ist stets eine Projektion anzuwenden. Waren die 2D-Daten an den geometrischen Objekten Punkt, Linie und Fläche orientiert, so erweitert sich das Darstellungsspektrum um körperhafte Elemente. Dies macht komplizierte Algorithmen zur Ermittlung von »Überlappungen«, sogenannte Hidden-Line-/Hidden-Surface-Verfahren, erforderlich. Die Orientierung zu 3D-Grafik-Daten begründet das Interesse an einer Reihe von körperorientierten grafischen Datenstrukturen /RV82/. Im folgenden sollen die wichtigsten kurz vorgestellt werden.

Linienorientierte Datenstrukturen stellen 3D-Objekte durch Linien (Kanten) dar. Ein Netz von Kanten auf der Körperoberfläche dient zur Beschreibung des Körpers. Eine solche Struktur ist einfach, bietet allerdings nur wenige Vorteile gegenüber den strukturierten 3D-Vektordaten, da bei der Darstellung die Flächen des Körpers unbekannt sind und so auch nicht die Überdeckungen ermittelt werden können. *Bild 4* zeigt ein Entity-Relationship-Diagramm einer linienorientierten Datenstruktur

Flächenorientierte Datenstrukturen benutzen zur Beschreibung von 3D-Objekten deren begrenzende Flächen. Die Flächen selbst werden durch 3D-Vektorzüge (Polygone) dargestellt. Aufgrund der

expliziten Beschreibung der Körperflächen können bei der Ableitung einer entsprechenden grafischen Darstellung Hidden-Line-/Hidden-Surface-Algorithmen verwendet werden. Eine flächenorientierte Datenstruktur wird in *Bild 5* durch ein zugehöriges Entity-Relationship-Diagramm dargestellt.

Eine etwas abstraktere Sicht bieten die körperorientierten Strukturen. Primitive Basiskörper und eine Anzahl von definierten Operationen (Vereinigung, Durchschnitt, Differenzen, etc.) bilden die Beschreibungsgrundlage. Komplexere 3D-Objekte werden durch eine baumartige Struktur bestimmt, deren Blätter aus Basiskörpern und deren innere Kanten aus den oben aufgeführten Operatoren bestehen. Der einfachen und natürlichen Objektbeschreibung steht ein äußerst komplexes Verfahren zur Ableitung der entsprechenden grafischen Darstellung gegenüber. *Bild 6* zeigt das Entity-Relationship-Diagramm einer körperorientierten Datenstruktur.



*Bild 7: Grobarchitektur eines CAD-Systems, nach der alle Systeme aufgebaut sind.*

Im folgenden werden die hier kurz vorgestellten, verschiedenen grafischen Datenstrukturen vergleichend bewertet und charakterisiert. Zunächst kann man eine Gruppierung nach der 2D-beziehungsweise 3D-Orientierung der Daten durchführen. Bezogen auf die Komplexität der Ableitungsverfahren ist hier eine klare Trennungslinie zu ziehen. Aber auch hinsichtlich des Anwendungsbezugs können Unterschiede festgestellt werden. Während Raster- und Vektordaten beziehungsweise die kombinierten Darstellungselemente als anwendungsneutral einzustufen sind, kommen die körperorientierten grafischen Datenstrukturen nur für körperorientierte Anwendungen in Betracht. Ebenso ist festzustellen, daß immer mehr »nicht-grafische« Informationen, Informationen also, die nicht grafisch dar-

gestellt werden, Eingang in die Datenstrukturen finden. So wird zum Beispiel die Kenntnis von der Zusammensetzung eines komplexen Körpers, die ja in der körperorientierten Datenstruktur vorhanden ist, nicht explizit grafisch dargestellt. Allerdings kann diese Information zur algorithmischen Herleitung wichtiger Körpereigenschaften, etwa dem Schwerpunkt, ausgenutzt werden.

**Körperorientierte Datenstruktur ermöglicht weitergehende Berechnungen**

Die aufgeführten grafischen Datenstrukturen sind keineswegs vollständig, denn es sind prinzipiell viele weitere Strukturen denkbar, zu denen Ableitungsverfahren angegeben werden können, die eine entsprechende grafische Darstellung ermitteln. Solche Strukturen zeichnen sich durch einen immer stärkeren Anwendungsbezug aus. Ein Beispiel hierfür bilden grafische Statistikdaten. Hier ist eine Datenstruktur denkbar, die aus Beschrei-

bungsprimitiven, wie Balkendiagramm, Torte oder Funktionskennlinie, die Darstellung statistischer Daten bestimmt.

Betrachtet man diese verschiedenen Datenstrukturen im Hinblick auf ihre Eignung zur langfristigen Speicherung oder zur Verwaltung durch ein Datenbanksystem, so stößt man auf zweierlei wichtige Faktoren: die grafische Wiedergabe und den Speicherplatzbedarf. Die grafische Wiedergabe ist in vielen Anwendungsfällen zeitkritisch, so daß grafische Daten, deren Ableitung zu einer entsprechenden grafischen Darstellung einfach ist, als besonders geeignet erscheinen. Bezüglich der Speicherung sind natürlich solche grafischen Daten zu bevorzugen, für die eine möglichst kompakte Repräsentation möglich ist. Aus Sicht eines Anwendungssystems ist ein weiterer Punkt von Interesse: wie gut unterstützt die grafische Datenstruktur und das mit ihr assoziierte Ableitungsverfahren die Bedürfnisse der Anwendung? Wünschenswert wäre eine Datenstruk-

tur, die allen drei Gesichtspunkten gerecht wird, sie existiert allerdings nicht. Dennoch kann versucht werden, durch Kombination verschiedener Datenstrukturen die gewünschten Eigenschaften zu erlangen. Ein einziges Objekt kann gleichzeitig in verschiedenen Strukturen repräsentiert sein. In diesem Zusammenhang spricht man auch von einer Mehrfachrepräsentation. Hierbei ist allerdings auf die Übereinstimmung der verschiedenen Repräsentationen zu achten.

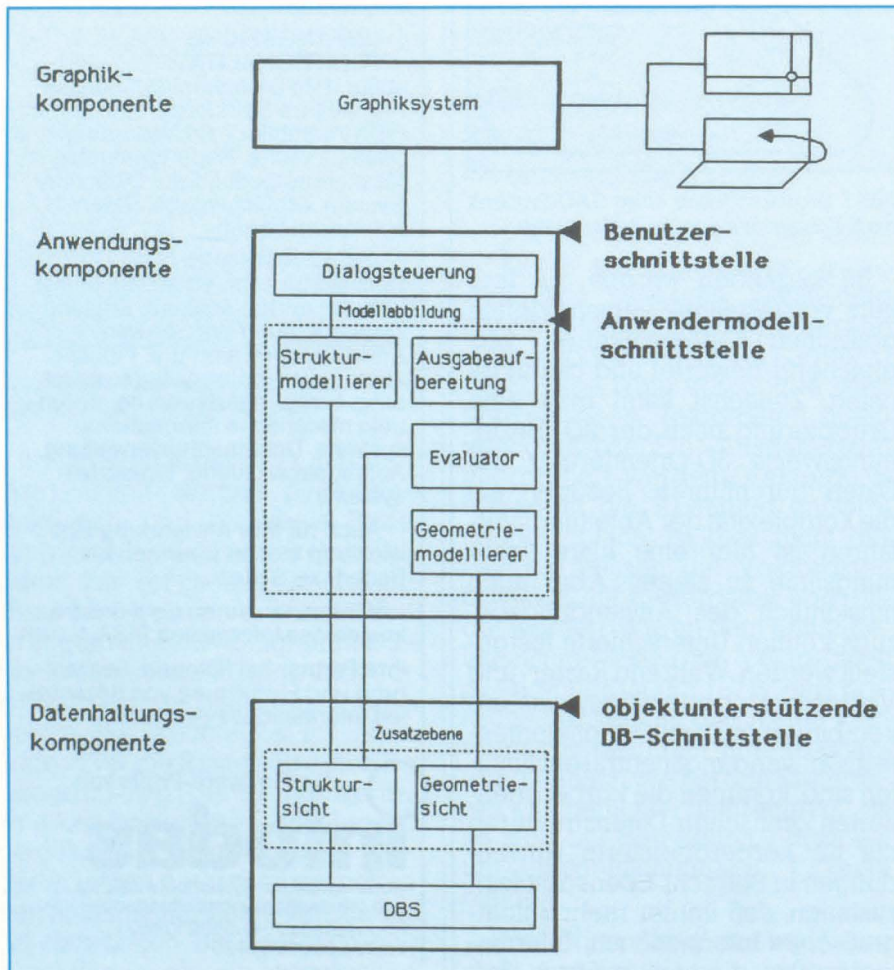
Am Beispiel einer nicht trivialen Grafik- und Datenbanksystemanwendung aus dem CAD-Bereich, einem 3D-Modellierer für polyederartige Werkstücke, sollen nun die aufgezeigten Problempunkte diskutiert werden. Neben einer Betrachtung der relevanten grafischen Daten interessiert vor allem das konkrete Zusammenwirken von Grafik- und Datenbanksystem.

**Architektur von CAD-Systemen:**

Die zentrale Aufgabe eines CAD-Systems ist die adäquate Unterstützung des Konstruktionsvorganges. Dazu müssen die Entwurfsobjekte sowohl grafisch dargestellt als auch in einer rechnerinternen Darstellung, der grafischen Datenstruktur, verwaltet werden. Die wichtigsten Eigenschaften eines CAD-Systems sind daher im wesentlichen hohe Funktionalität, leichte Handhabbarkeit und ein hohes Maß an Benutzerfreundlichkeit. Zunehmend werden weitere Merkmale als relevant erkannt. Sie erstrecken sich von der Portabilität über die leichte Erweiterbarkeit um benutzerspezifische Funktionen bis hin zu einer möglichen Kopplung mit anderen CAD- oder CAM-Systemen. Das wachsende Bedürfnis nach diesen Systemeigenschaften führt zu der Forderung nach einer Strukturierung der CAD-Software durch eine klare Abgrenzung in Anwendungs-, Datenhaltungs- und Grafikaspekte sowie zur Standardisierung der Schnittstellen zwischen den entsprechenden Subsystemen.

In Bild 7 ist die Grobarchitektur eines CAD-Systems skizziert, wie sie in allen existierenden Systemen wiederzufinden ist. Die drei wesentlichen Komponenten werden repräsentiert durch das konkrete Anwendungssystem (AWS), das Da-

Bild 8: Das System KUNICAD besteht aus der Datenhaltungs-, der Anwendungs- und der Grafikkomponente.



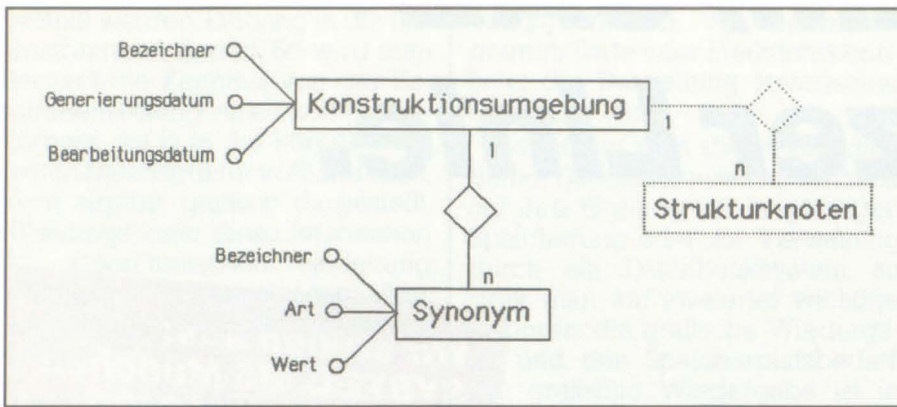


Bild 9: Für jede definierte Umgebung existiert genau ein Entity vom Typ »Konstruktionsumgebung«.

tenhaltungssystem (DHS) und das Grafiksystem (GS). Durch diese Komponentenbildung werden gleichzeitig zwei Subsystemschnittstellen definiert. Die Datenhaltungsschnittstelle ermöglicht die Verwaltung der Entwurfsobjekte in der rechnerinternen Darstellung. Kontrolliert von dem Anwendungssystem, werden die Entwurfsobjekte gespeichert, gelöscht, geändert oder auf Verlangen bereitgestellt. Die Grafikschnittstelle erlaubt deren grafische Darstellung, ebenfalls gesteuert durch das Anwendersystem. (Zwischen Grafik- und Datenhaltungskomponente existiert — zumindest bei den herkömmlichen CAD-Systemen — keine direkte Verbindung. Eine solche Schnittstelle zwischen GS und DHS würde allerdings den Aufwand für eine grafische Ausgabe der Entwurfsobjekte durch eine direkte Umsetzung von der rechnerinternen zur grafischen Darstellung, das heißt, für den Ableitungsprozeß, entscheidend verringern.) Bezüglich der Standardisierung der Subsystemschnittstellen sind auf dem Grafiksektor mit der Normierung durch GKS (siehe /GKS83/) gewisse Vorgaben zu erkennen. Dies wird durch die wachsende Vermarktung von GKS-orientierten Grafiksystemen zusätzlich untermauert. Im Bereich der Datenhaltung gibt es bereits seit langem einheitliche Schnittstellennormen in Form von Datenmodell-Definitionen. Dies und die ständig fortschreitende Expansion und Integration der Anwendungen legt den Einsatz von Datenbanksystemen nahe. Damit können viele der Nachteile von dateiorientierten CAD-Systemen, wie übermäßige Datenredundanz, fehlende Integritätskontrolle, mangelnde Daten-

strukturierung vermieden sowie Standardisierung, Austausch und Übertragung von komplexen Datenstrukturen verbessert werden. Zusätzlich werden ebenfalls wichtige Aufgaben (Kontrolle des Mehrbenutzerbetriebs, Sicherung der Daten, Strukturierung der Verarbeitung mittels des Transaktionskonzeptes) vom Datenbanksystem automatisch übernommen. **Das KUNICAD-System:** Das für Lehr- und Forschungsaufgaben entwickelte, experimentelle KUNICAD-System /HHL86/ ist ein datenunterstütztes Modellierungssystem für Werkstücke, das als Kernalgorithmus für die geometrische Modellierung ein volumenorientiertes Verfahren (CSG, Constructive Solid Geometry /RV82/) einsetzt. Dabei werden Bauteile als Polyeder dargestellt, die aus Basiskörpern unter Anwendung der zugelassenen Operationen gewonnen werden. Intern werden zusätzlich Strukturen nach dem Begrenzungsflächenmodell (BREP: Boundary Representation /RV82/) gehalten, die automatisch aus den CSG-Strukturen abgeleitet und, falls erforderlich, nachgeführt werden. Das KUNICAD-System stellt ein sogenanntes körperorientiertes »Dual-Repräsentation«-Modellierungssystem dar, da die rechnerinterne Darstellung des Konstruktionkörpers einer CSG/BREP-Kombination entspricht. Die primäre, CSG-artige Darstellung erlaubt die direkte und einfache Abbildung des Anwendermodells, während sich die optionale BREP-Darstellung wegen ihrer Nähe zur Grafik-Repräsentation sehr gut für Ausgabezwecke eignet. Damit wird die zeitkritische grafische Wiedergabe entsprechend unterstützt. Durch kontrolliertes Nach-

führen der sekundären grafischen Datenstruktur kann zudem der benötigte Speicherplatz den Erfordernissen dynamisch angepaßt werden.

**Die Systemarchitektur:** Gemäß Bild 7 läßt sich der Bauteilmodellierer in drei Hauptkomponenten unterteilen:

- Die Grafikkomponente stellt Hilfsmittel zur Realisierung ergonomischer Benutzerschnittstellen zur Verfügung.
- Die Datenhaltungskomponente verwaltet die rechnerinterne Darstellung der Konstruktionsdaten.
- Die Anwendungskomponente bildet die »Anwendersicht« auf die rechnerinterne »Datensicht« ab.

Im folgenden sollen diese Aufgaben anhand der in Bild 8 dargestellten Bausteine näher erläutert werden.

Die Grafikkomponente erlaubt dem Anwendungssystem die Ein- und Ausgabe grafischer und textueller Informationen. Dabei entspricht die Funktionalität des Grafiksystems der von GKS. Der Baustein »Dialogführung« ist Bestandteil der Anwendungskomponente. Seine Hauptaufgabe besteht in der Vorverarbeitung der Ein- und Ausgabe sowie in der Entkopplung der implementierten Benutzerschnittstelle von der darunterliegenden Anwendermodell-schnittstelle (auf der Ebene des Anwendermodells spielt es keine Rolle, ob die Benutzerschnittstelle menü- oder kommandogesteuert arbeitet, beziehungsweise, ob eine Selektion durch eine »Pick«-Eingabe möglich ist oder nicht). Es werden die notwendigen Aktivitäten innerhalb der Modellabbildung angestoßen und kontrolliert. Die weiteren Bausteine der Anwenderkomponente dienen der Modellabbildung, das heißt, der Projektion der Anwendermodell-Objekte auf die Datenstrukturen und die damit assoziierten Operationen einer im System darunterliegenden Schnittstelle (Geometrie-/Struktursicht).

Der Strukturmodellierer verwaltet und aktualisiert die organisatorischen und strukturellen Konstruktionsdaten. Hierzu verwendet er Funktionen des Bausteins »Struktursicht« an der Schnittstelle zur Da-

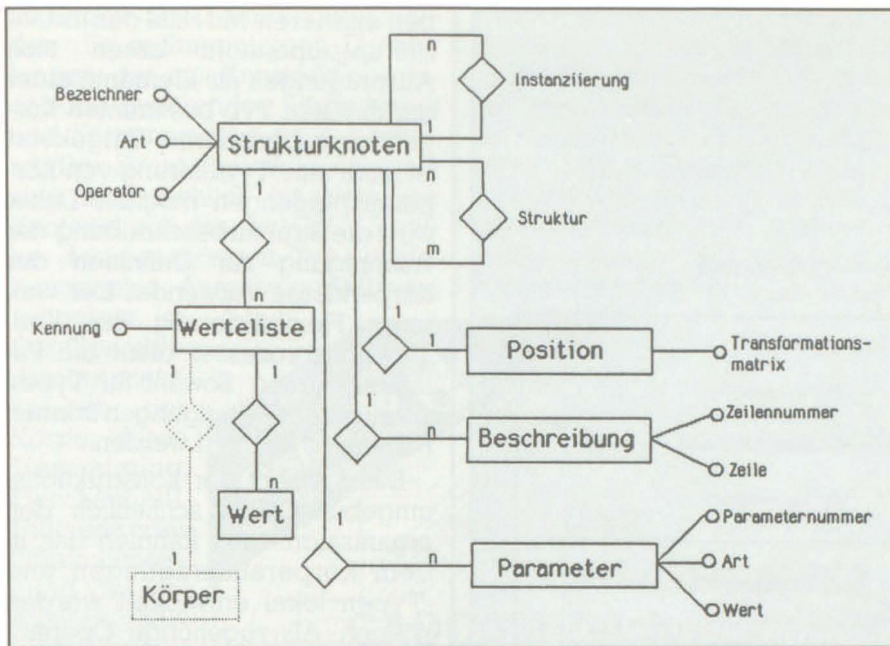


Bild 10: Der Strukturteil realisiert hauptsächlich ein Netz von Knoten.

tenhaltungskomponente. Der Baustein »Ausgabenaufbereitung« übernimmt vorwiegend die Transformation der gespeicherten 3D-Information in eine 2D-Darstellung, die dann unmittelbar auf einem Grafik-Terminal sichtbar gemacht

werden kann. Im Geometriemodellierer wird die eigentliche Verarbeitung der Geometriedaten durchgeführt. Die Datenhaltungskomponente stellt innerhalb der Geometriesicht die hierzu erforderlichen Grundfunktionen bereit. Der

Evaluator schließlich repräsentiert das Bindeglied zwischen der strukturellen (CSG) und der geometrischen (BREP) Interdarstellung eines Werkstückes. Dieser Programmbaustein steuert die Berechnung der BREP-Darstellung eines Körpers aus seiner CSG-Repräsentation und garantiert damit die Konsistenz dieser Mehrfachrepräsentation. Hiermit ist es möglich — vom Aufbau des CSG-Baumes ausgehend — eine geometrische Beschreibung erst dann herzuleiten, wenn sie tatsächlich gebraucht wird. Das verwendete Datenbanksystem bietet ein netzwerkorientiertes Datenmodell mit einer einfachen navigierenden Datenmanipulationssprache an /Da81/. Der mangelnde Anwendungsbezug bei der Datenmodellierung und die für die Konstruktionsaufgaben überhaupt nicht geeigneten Datenbank-Operationen erfordern die Bereitstellung einer mehr anwendungsbezogenen und stärker objektorientierten Datenhaltungsschnittstelle. Die Bausteine »Struktur- und Geometriesicht« der Datenhaltungskomponente repräsentieren



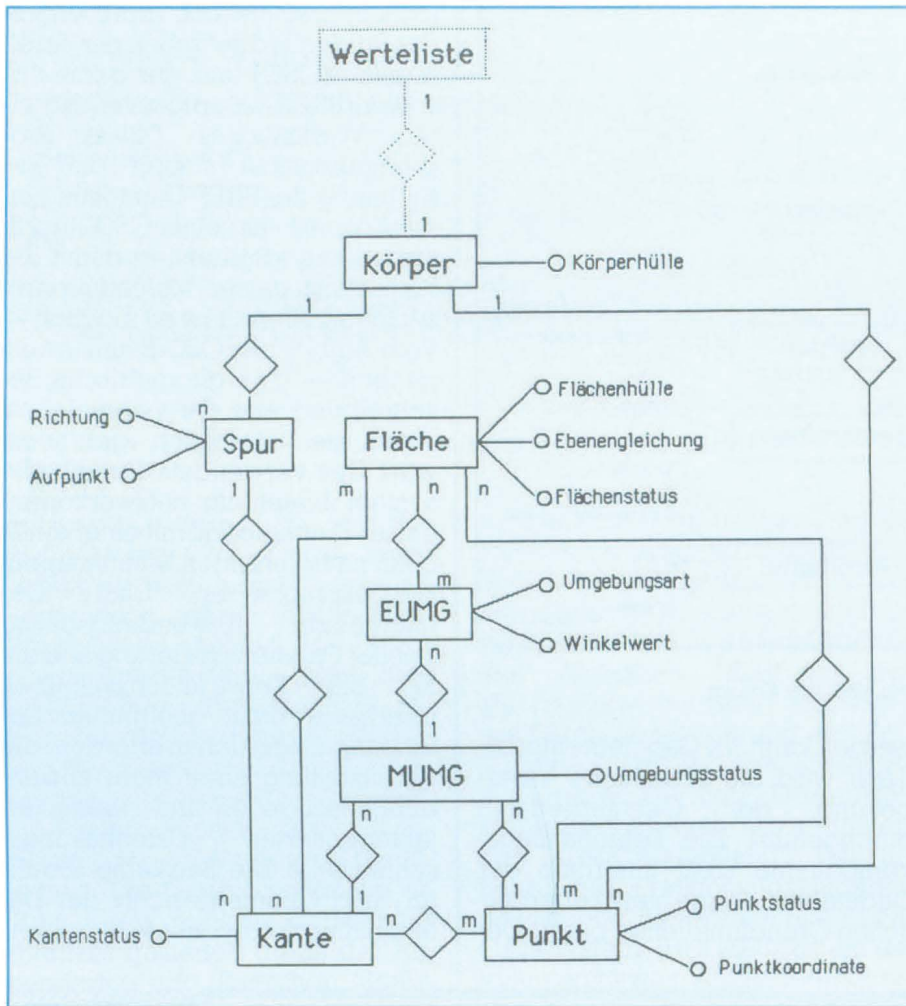


Bild 11: Der Geometrieteil benutzt zur Darstellung der Körperdaten das Begrenzungsflächenmodell.

tieren diese sogenannte objektunterstützende Datenbank-Schnittstelle. Sie stellen Objekte bereit, die von den Bausteinen der Modellabbildung auf vorteilhafte Art und Weise zu nutzen sind.

**Das Anwendermodell:** In dem hier konzipierten Anwendermodell werden konstruktive, organisatorische und geometrische Informationen über die Werkstücke modelliert und verwaltet.

Durch die Unterscheidung von drei Körperarten wird dabei eine geeignete Konstruktionssemantik ermöglicht. Die Basiskörper entsprechen den Grundbausteinen des CSW-Modells, das heißt, sie stellen die Primitiven dar, aus denen komplexere Körper konstruiert werden können. In diesem Modell sind das unter anderem Quader und zylinderförmige Polyeder. Die möglichen Körperaggregationen werden durch die binären Modelloperatoren »Vereinigung«, »Durchschnitt« und »Differenz« sowie durch die unären Operatoren »Translation«, »Rotation« und »Ska-

lierung« bestimmt. Zusätzlich unterstützt das Modell noch einen Operator »Zusammensetzung«. Dies ist ein Spezialfall der Vereinigung, bei dem sich die Operanden nicht überlappen dürfen. Unter einem Bauteilkörper versteht man eine Verknüpfung aus Basisbeziehungsweise wiederum Bauteilkörpern mittels Vereinigung, Durchschnitt oder Differenz. Hingegen sind Baugruppenkörper als reine Zusammensetzung von Basis-, Bauteilbeziehungsweise Baugruppenkörpern definiert. Sie lassen sich in ihre Bestandteile zerlegen und beschreiben damit mehr den Aufbau eines Körpers. Dagegen repräsentiert der Bereich der Bauteile die Konstruktionsgeschichte von Körpern.

Die Typisierung erlaubt den Übergang von der Ausprägung zum Typ und umgekehrt vom Typ zur Ausprägung. Damit ist die Einführung von neuen und komplexeren Körperklassen möglich. Von allen drei Körperarten können sowohl Ausprägungen als auch Ty-

pen existieren. Mit Hilfe des Instanzierungsoperators lassen sich Ausprägungen als Elemente einer durch einen Typ bestimmten Körperklasse generieren. Umgekehrt ist auch eine Typisierung von Körperausprägungen möglich. Dabei wird die Strukturbeschreibung der Ausprägung zur Definition der Körperklasse verwendet. Der weiteren Flexibilisierung des Konstruktionsprozesses dient die Parametrisierung. Sowohl für Typen als auch für Ausprägungen können Parameter definiert werden.

Das Konzept der Konstruktionsumgebung stellt schließlich den organisatorischen Rahmen dar, in dem Körperausprägungen und -Typen lokal entwickelt werden können. Als zugehörige Operatoren sind definiert: Beginnen, Beenden, Öffnen und Schließen. Das Öffnen und Schließen einer Konstruktionsumgebung bestimmt Bearbeitungsphasen, in deren Rahmen ein vom Benutzer veranlaßtes Zurücksetzen möglich ist. Dagegen wird beim Systemzusammenbruch ein Überleben der zuletzt erfolgreich durchgeführten Modelloperation garantiert. Bei Beendigung der Konstruktion wird vom System eine Globalisierung durchgeführt. Dabei werden lokale Körpertypen global bekannt gemacht, lokale Körperausprägungen dagegen verlieren ihre Bedeutung und werden gelöscht.

**Das Datenbankschema:** Im Datenbank-Schema wird die Struktur der vom DBS zu verwaltenden Daten beschrieben. Analog zur Dreiteilung im Anwendermodell werden hier ebenfalls drei Arten von Anwendungsinformation unterschieden: organisatorische, strukturelle und geometrische Information.

Der Organisationsteil, dargestellt als Entity-Relationship-Diagramm in Bild 10 beschreibt den Zustand der einzelnen Konstruktionsumgebungen. Für jede definierte Umgebung existiert genau ein Entity vom Typ »Konstruktionsumgebung«. Jedem solchen Umgebungs-Entity ist eine Menge von Synonym-Entities zugeordnet, die die Variablen- und Parameterinformation enthalten. Der Organisationsteil ist mit dem Strukturteil verbunden über einen Relationship-Typ zwischen den Entity-Typen »Konstruktionsumgebung« und »Strukturknoten«.

Im Entitytyp »Strukturknoten« werden die rechnerinternen Darstellungen der zur Konstruktionsumgebung gehörenden Körpertypen und Körperausprägungen verwaltet. Der Strukturteil (*Bild 11*) realisiert hauptsächlich ein Netz von Knoten durch das die »CSG«-Struktur des zugehörigen Typ- beziehungsweise Ausprägungskörpers festgelegt ist.

Das Operator-Attribut zeigt an, ob der Körper durch Vereinigung, Durchschnitt, Differenz oder Zusammensetzung der assoziierten Körper entstanden ist. Die zugeordneten Entities vom Typ »Parameter« repräsentieren die aktuellen Parameter der Instanziierung. Ergänzend können noch Entities vom Typ »Beschreibung« für erläuternden Text sowie vom Typ »Position« zugeordnet werden.

Das Attribut »Transformationsmatrix« vom Positionstyp nimmt alle Informationen bezüglich Verschiebung, Rotation und Skalierung des Körpers auf. Die Entities vom Typ »Werteliste« verbinden den Strukturteil mit dem optionalen Geometrieteil. Vergleicht man *Bild 11* mit *Bild 6*, so er-

kennt man sehr deutlich, daß eine Anreicherung um nichtgrafische und eine enge Verflechtung mit der grafischen Information stattgefunden hat. Die in *Bild 6* beschriebene, körperorientierte grafische Datenstruktur ist auch in *Bild 11* wiederzufinden: der Entity-Typ »Strukturknoten« und der reflexive Relationship-Typ »Struktur« model-

lieren die Körperstruktur. Alle restlichen Informationen sind nicht-grafisch und dienen der genaueren Abbildung des Anwendermodells. Der Geometrieteil (*Bild 12*) benutzt zur Darstellung der geometrischen und topologischen Körperdaten das Begrenzungsflächenmodell, in dem die Körpergeometrie durch die Körpermantelflächen charakte-

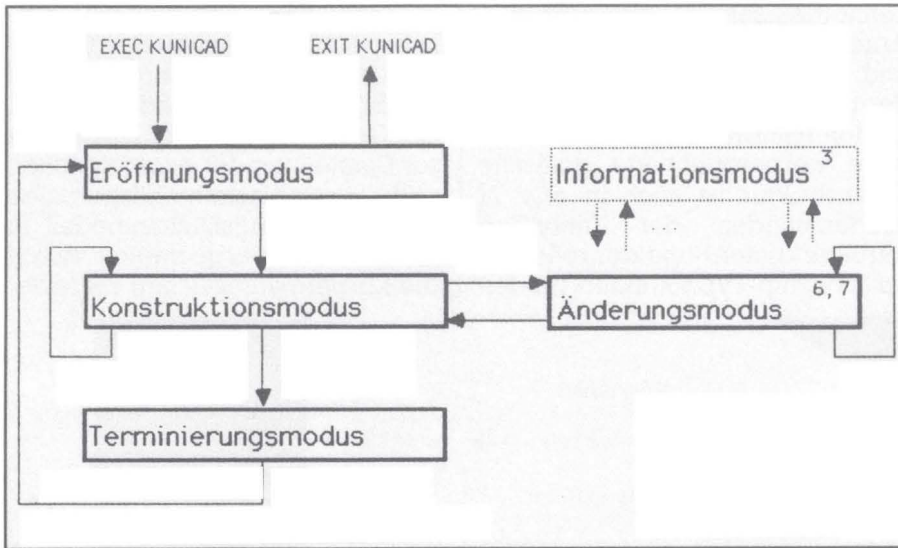


Bild 12: Zustandsübergangsdiagramm der Benutzerschnittstelle des experimentellen CAD-Systems KUNICAD.

risiert wird. Die Flächen selbst sind durch die begrenzenden Kanten und diese wiederum durch deren Randpunkte bestimmt. Der Entity-Typ »Körper« enthält als beschreibendes Attribut unter anderem die sogenannte Körperhülle. Sie besteht aus den Endpunkten der Raumdiagonalen eines Quaders, der den eigentlichen Körper umhüllt. Diese Information wird zur Erkennung von etwaigen Körperüberlappungen benötigt. Der Entity-Typ »Spur« dient der Partitionierung der Kantenmenge eines Körpers. Einer Ausprägung sind alle jeweils auf einer Geraden liegenden Kanten zugeordnet (dargestellt durch die Attribute »Richtung« und »Aufpunkt«). Die Attribute des Flächen-Typs bestehen unter anderem aus der Flächenhülle in Form eines umhüllenden Quaders, einer Ebenengleichung definiert zusätzlich zur assoziierten Ebene noch die Ausrichtung des Körperinneren relativ zur Fläche. Der Entity-Typ »Punkt« besitzt die Punktkoordinaten, die die eigentliche geometrische Information beinhalten sowie ein Statusattribut. Die Bedeutung des Entity-Typs »Kante« beruht allein auf den mit ihm assoziierten Entity-Typen. Das Attribut »Kantenstatus« erfüllt, wie auch der Flächen- und Punktstatus, nur einen organisatorischen Zweck während der Verknüpfungsphase zweier Begrenzungsflächendarstellungen. Die Entitytypen MUMG (Mehrfachumgebung) und EUMG (Einfachumgebung) werden aufgrund des hier verwendeten Verfahrens zur Körperverarbeitung benötigt.

Sie dienen der genaueren Beschreibung der geometrischen Umgebung eines Punktes beziehungsweise einer Kante bezogen auf einen Körper. Diese Information ist insbesondere bei einer Körperberührung von Bedeutung. Da sich Körper quasi selbst berühren können, sind »Mehrfachumgebungen«, bestehend aus mehreren »Einfachumgebungen« denkbar. Das Attribut »Umgebungsart« erlaubt die Unterscheidung Flächen-, Kanten- und Eckpunktumgebungen. Die Kernbeschreibung der Begrenzungsflächendarstellung des Geometrieteils ist auch in Bild 5 als flächenorientierte grafische Datenstruktur wiederzufinden. Die restlichen Informationen des Geometrieteils stellen wiederum nicht-grafische Daten dar, die das hier gewählte Verarbeitungsverfahren unterstützen.

### CSG-artige Darstellung erlaubt die einfache Abbildung des Modells

**Die Objektunterstützende Datenbank-Schnittstelle:** Aus Gründen einer besseren Verständlichkeit wurde die Schemainformation in Form von Entity-Relationship-Diagrammen visualisiert. Im KUNICAD-System wird die entsprechende netzwerkartige Form verwendet, da das dort benutzte Datenbanksystem auf dem Netzwerkdatenmodell basiert. Unabhängig von der Darstellung der Schemainformation bleibt allerdings folgendes festzustellen: die im Anwen-

dermodell vorhandenen Objekte werden auf Schemaebene nicht direkt als »Schemaobjekte«, sondern jeweils als »Schemaausschnitt«, bestehend aus mehreren »Schemaobjekten« dargestellt. Das bedeutet, daß der »Gegenstand der Verarbeitung« an der Schnittstelle zum Anwendersystem meist von komplexerer Natur ist als herkömmliche Datenbankobjekte. Aus diesem Grund wird mit Hilfe der Geometrie- und Struktursicht, die sogenannte objektunterstützende Schnittstelle gebildet, die die benötigten Verarbeitungseinheiten (VE) bereitstellt.

Alle Verarbeitungseinheiten der objektunterstützenden Schnittstelle korrespondieren mit dem zugrundeliegenden Datenbankschema. So beinhaltet die Geometriesicht die Körper-VE, die sich auf der Schemaebene aus dem gesamten Geometrieteil zusammensetzt; auf Ausprägungsebene sind mit ihr alle diejenigen Satzausprägungen assoziiert, die mit einem einzigen Körper-Ankersatz über entsprechende Beziehungen verbunden sind. Weitere Verarbeitungseinheiten sind Flächen-, Kanten-, Punkt- und die Umgebungs-Verarbeitungseinheiten. Sie bestehen aus Satzstrukturen, die im wesentlichen durch eine Reihe konstituierender Satztypen sowie einen Ankersatztyp bestimmt sind. Die Verarbeitungseinheiten der Struktursicht umfassen jeweils größere Teile des Schemas. Instanz- und Typ-VE sind repräsentiert durch die zugehörige Strukturbeschreibung und den assoziierten Geometrieteil. Die Verarbeitungseinheit »Konstruktionsumgebung« enthält zusätzlich noch die Satzausprägung von Konstruktionsumgebung als Anker sowie die zugehörigen Synonym-Sätze. Lediglich die Synonym-VE ist einfach aufgebaut und besteht nur aus dem Ankersatz.

An der objektunterstützenden Schnittstelle sind drei unterschiedliche Operationstypen zu erkennen, die zur Charakterisierung der Verarbeitung an dieser Schnittstelle dienen. Für die satzorientierten Operationen ist der Gegenstand der Verarbeitung genau ein Datenbankobjekt. Die strukturorientierten Operationen verarbeiten jeweils eine strukturierte Menge heterogener Datenbankobjekte. Die Verarbeitung ist geprägt durch

das »Herausziehen« solcher Satz- mengen aus der und ihr »Einbringen« in die Datenbank. Für die objektorientierten Operationen ist der Verarbeitungsgegenstand ebenfalls eine strukturierte Menge heterogener Datenbankobjekte, deren Verarbeitung allerdings direkt durch eine »semantisch höherwertige« Funktion erfolgt.

### Instanziierungsoperator erlaubt die Typisierung von Körperausprägungen

**Die Benutzerschnittstelle:** Aufbauend auf dem oben vorgestellten Anwendermodell lassen sich nun ergonomische Benutzerschnittstellen realisieren, die sich dem Arbeitsablauf und den Denkgewohnheiten des Anwenders im CAD-Bereich anpassen und damit die Akzeptanz solcher Systeme für kreative Entwurfsarbeiten beeinflussen. Der Baustein »Dialogführung« implementiert mit Hilfe der Grafikkomponente die Benutzerschnittstelle von KUNICAD, die letztendlich nur eine syntaktische Aufbereitung der abstrakteren Anwendermodellschnittstelle mittels der konkret vorhandenen grafischen Hard- und Software darstellt.

Die Benutzerschnittstelle des CAD-Systems /Stö86/ läßt sich mit dem in *Bild 13* dargestellten Zu-

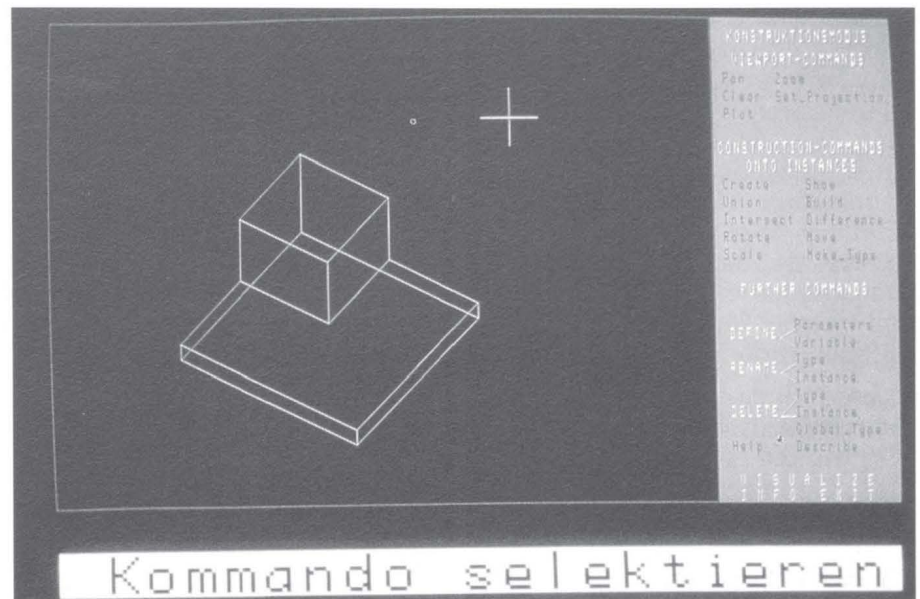


Bild 13: Konstruktionsfenster nach einer CSG-Operation.

standsübergangsdiagramm gut beschreiben, man unterscheidet fünf verschiedene Zustände. Die Zustände werden durch Rechtecke und die möglichen Zustandsübergänge durch Pfeile repräsentiert. Der Informationsmodus ist außer vom Terminierungsmodus von jedem anderen Modus aus zu erreichen. Durch das Starten von KUNICAD erreicht man den Eröffnungsmodus. Unter Angabe eines Konstruktionsumgebungsnamens gelangt man in den Konstruktionsmodus. Von dort aus wird Information über die vom System momen-

tan verwalteten Körperausprägungen eingeholt. Dazu wird kurzzeitig in den Informationsmodus gewechselt. Anschließend werden die zwei benötigten Basiskörper im Konstruktionsfenster gezeigt. Das Ergebnis der »Verschneide«-Operation dieser Basiskörper erscheint als nächstes im Konstruktionsfenster (*Bild 14*). Ausgehend von diesem aktuellen Bauteil wird der Änderungsmodus aktiviert. Zusätzlich vom Konstruktionsfenster erscheint nun noch das CSG-Fenster (CSG, Konstruktive Solid Geometrie), das zur Navigation in der CSG-

Struktur des jeweils aktuellen Körpers dient (Bild 15). In einem weiteren Fenster, dem sogenannten Baumfenster, kann die Visualisierung der aktuellen CSG-Struktur zur Spezifikation von Änderungsoperationen dargestellt werden (Bild 16). Über den Konstruktionsmodus ist dann der Terminierungsmodus erreichbar, von dort aus gelangt man wieder in den Eröffnungsmodus.

Anhand dieser skizzierten Benutzerschnittstelle ist gut zu erkennen, wie die grafischen Daten, das heißt, die rechnerinternen Darstellungen der Körper, von KUNICAD verarbeitet und verwaltet werden. Für die grafische Darstellung eines

Körpers im Konstruktionsfenster wird die zugehörige Begrenzungsflächendarstellung verwendet. Hingegen wird der Inhalt von CSG- und Baumfenster aus der CSG-Struktur abgeleitet. Alle im Informationsmodus erfragbaren Daten

## Realisierte Datenbank basiert auf dem Netzwerk-Datenmodell

werden direkt aus der Datenbank geholt, ebenso die Informationen über eine bestimmte Konstruktionsumgebung. Die unterschiedliche Verwendung der CSG- und Begrenzungsflächendaten ist ebenfalls deutlich zu erkennen. Die Be-

grenzungsflächendaten dienen momentan ausschließlich der grafischen Repräsentation der Konstruktionskörper. Im Unterschied dazu wird die CSG-Information zur Visualisierung des Körperaufbaus, also der »Konstruktionsgeschichte« verwendet.

**Zusammenfassung:** Durch die Betrachtung typischer Anwendungen von Grafik- und Datenbanksystemen konnten deutliche Gemeinsamkeiten und Berührungspunkte erkannt werden. Als wesentliche Anforderung kristallisierte sich die effiziente Verwaltung vielfältig strukturierter (grafischer) Daten und deren (grafische) Darstellung heraus. Zum einen löste dies die Überlegungen zu grafischen Datenstrukturen aus, zum anderen war es Ausgangspunkt für die Untersuchungen an einer realistischen Datenbank- und Grafiksystemanwendung.

## Wesentliche Anforderung: effiziente Verwaltung von strukturierten Daten

Aus einem allgemeinen Blickwinkel heraus wurden Daten in einer konkreten Anwendung dann als grafische Daten definiert, wenn ein Ableitungsprozeß zur Transformation in eine zugehörige grafische Darstellung existiert. Die Tatsache, daß verschiedene grafische Datenstrukturen durch verschiedene Ableitungsprozesse in eine gemeinsame grafische Darstellung zu transformieren sind, bestimmt den Begriff der Mehrfachrepräsentation. Hierbei existieren nebeneinander unterschiedliche grafische Datenstrukturen mit unterschiedlichen wünschenswerten Eigenschaften, beispielsweise geringem Speicherplatzbedarf, schnellem Ableitungsvorgang und starkem Anwendungsbezug.

Die Betrachtungen an dem Prototypensystem KUNICAD brachten weitere Erkenntnisse bezüglich der Struktur und der Verarbeitung grafischer Daten. So wurde zum einen die enge Verflechtung und der fließende Übergang von grafischen und nichtgrafischen Daten deutlich. Zum anderen wurden die Probleme einer Mehrfachrepräsentation am Beispiel von körper- und flächenorientierten Datenstrukturen (Strukturteil/Geometrieteil des Da-

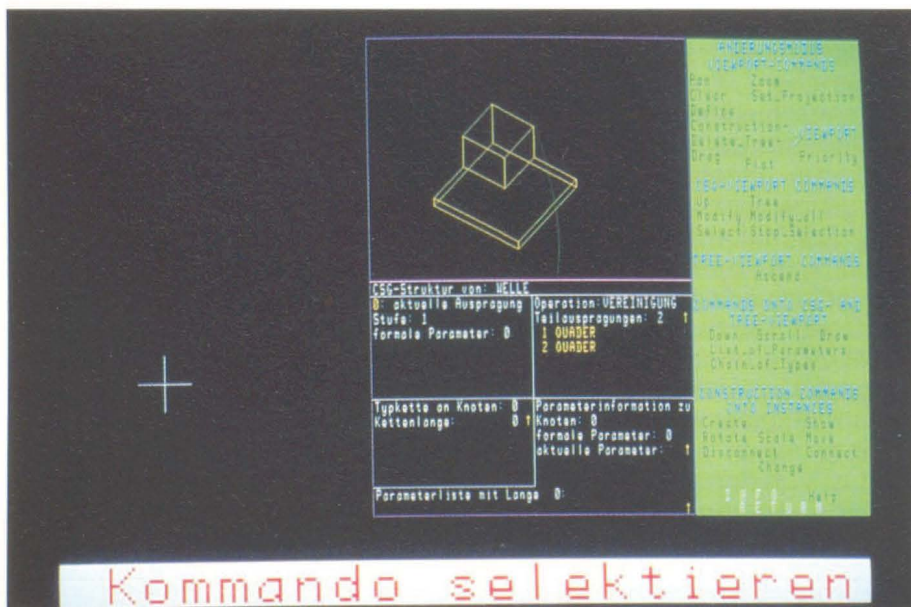
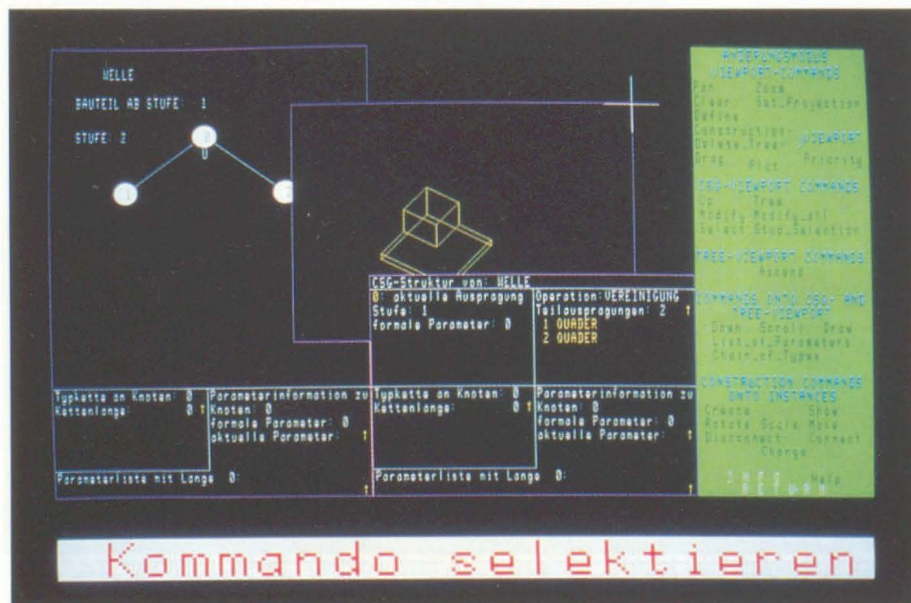


Bild 14: Änderungsmodus mit Konstruktions und CSG-Fenster.

Bild 15: Das »Baumfenster« zeigt die aktuelle CSG-Struktur.



tenbankschemas) erläutert. Die Wahrung der inhaltlichen Übereinstimmung zwischen den beiden grafischen Datenstrukturen, beziehungsweise die Ableitung der einen Repräsentation aus der anderen bildeten die Aufgaben wichtiger Systemkomponenten. Als ein weiterer wichtiger Punkt wurden Mängel an bestehenden Grafik-

### Verschiedene Ableitungsprozesse erzeugen Mehrfachrepräsentation

und Datenbanksystem-Schnittstellen aufgedeckt. Das bestehende Datenbanksystem mit einer an dem Netzwerk-Datenmodell orientierten Schnittstelle mußte um eine zusätzliche Programmschicht erweitert werden, durch die erst eine angemessene Verarbeitung anwendungsbezogener Objekte ermöglicht wurde. Ebenso wurde das Grafiksystem um Komponenten zur Behandlung der 3D-Aspekte erweitert.

Ein Ansatz zur Behebung der

aufgezeigten Schwächen scheint darin zu bestehen, daß neue, problemangepaßte Modelle zur Strukturierung und zur Verwaltung grafischer und nicht-grafischer Daten geschaffen werden, die die Verarbeitung anwendungsbezogener Objekte direkt unterstützen. Im Bereich der Datenbanksystemforschung führte dies bereits zur Entwicklung von sogenannten Nicht-Standard-Datenbanksystemen /HR85/. Darüberhinaus sollte durch weitestgehende Integration mit Grafiksystemeigenschaften, also durch die Schaffung integrierter grafischer Datenbanksysteme, dem inhärenten Zusammenhang zwischen daten- und darstellungsintensiven Anwendungen Rechnung getragen werden /HM86/.

#### Literatur

- /Ch76/ Chen, P.P.: The Entity-Relationship-Model - Toward a Unified View of Data, in: ACM TODS, Vol. 1, No. 1, 1976, pp. 9-36.  
 /Da81/ Date, C.J.: An Introduction to Database Systems, 3rd ed. Addison-Wesley Publ. Co., 1981.  
 /GKS83/ mehrere Artikel in: Informatik

Spektrum, Band 6, Heft 2, Springer-Verlag, April 1983.

/HHL86/ Härder, T., Hübel, Ch., Langenfeld, S., Mitschang, B.: KUNICAD -Ein datenbankgestütztes geometrisches Modellierungssystem für Werkstücke, Forschungsbericht Nr. 22/86, SFB 124, Universität Kaiserslautern, erscheint in Informatik -Forschung und Entwicklung, 1986.

/HM86/ Hübel, Ch., Mitschang, B.: Ein Ansatz zu graphischen Datenbanken und deren Verwendung in CAD-Systemen, in Proc. der Fachtagung für Graphische Datenverarbeitung »Austrographics 86«, Wien, 1986.

/HR85/ Härder, T., Reuter, A.: Architektur von Datenbanksystemen für Non-Standard-Anwendungen, in: Proc. GI-Fachtagung »Datenbanksysteme in Büro, Technik und Wissenschaft«, März 1985, IFB 94, Springer-Verlag, Berlin, Seite 253-286 (eingeladener Vortrag).

/IGES83/ Initial Graphics Exchange Specification (IGES), Version 2.0, U.S. Department of Commerce, NBS, Washington, DC 20234, USA, Feb. 83.

/Stö86/ Stölben, H.: Konzeption und Implementierung einer operationalen Benutzerschnittstelle für das KUNICAD-System, Diplomarbeit, Universität Kaiserslautern, 1986.

/RV82/ Requicha, A.A.G., Voelcker, H.B.: Solid Modelling: A Historical Summary and Contemporary Assessment, in: IEEE Computer Graphics and Applications, Vol. 2, No. 2, March, 1982, pp. 9-24 ■