



Gesellschaft für Informatik e. V.

Fachtagung

**CAD '96**

Verteilte und intelligente  
CAD-Systeme

7.-8. März 1996



Deutsches Forschungszentrum für  
Künstliche Intelligenz (DFKI) GmbH  
Kaiserslautern

In D. Ruland (ed.): *Verteilte und intelligente CAD-Systeme* (pp. 107-119), Bonn: Ges. für Informatik; Kaiserslautern/Saarbr.: DFKI, 1996, BEST PAPER AWARD.

## Ein wissensbasiertes System für die 3D-computergraphische Montage-Simulation

*Bernhard Jung und Ipke Wachsmuth*  
*Technische Fakultät, AG WBS (KI)*  
*Universität Bielefeld*  
*{jung, ipke}@techfak.uni-bielefeld.de*

### Zusammenfassung

Der CODY *Virtuelle Konstrukteur* ist ein wissensbasiertes System, das die interaktive Montage von 3D-computergraphisch dargestellten Bauteilen zu komplexen Aggregaten ermöglicht. Eine Besonderheit des Systems ist, daß es grob spezifizierte natürlichsprachliche Eingaben des Benutzers verarbeitet und teilautonom in technische Kommandos zur Aktualisierung der Graphikszene umsetzt. Darüber hinaus wird der jeweilige Montagestand schritthaltend mit dem strukturierten Modell eines Zielaggregats abgeglichen. Dynamische Wissensrepräsentationen werden erzeugt und modifiziert, wenn entstandene Aggregate als Baugruppen des Zielaggregats erkannt werden. Außerdem werden, entsprechend ihrer Verwendung in diesem Zielaggregat, die spezifischen Funktionen einzelner Bauteile erkannt. Verbale Eingaben können sich so immer auf den aktuellen Stand der Montage beziehen.

### Abstract

The CODY *Virtual Constructor* is a knowledge-based system that enables the interactive assembly of 3D-graphically visualized mechanical parts to complex aggregates. A novelty of the system is that it processes vaguely specified input in natural language which is translated semi-autonomously into technical commands for updating the graphics scene. Furthermore, the current state of the assembly is step-keepingly matched against the structured model of a target aggregate. Dynamic knowledge representations are created and modified when constructed aggregates are recognized as assembly groups of the target aggregate. Also, according to their use, the specific functions of single parts in the target aggregate are determined. Therefore, verbal instructions can always refer to the current state of the assembly.

# 1 Einleitung

Im Konstruktionsbereich finden dreidimensional computergraphisch visualisierte „virtuelle Szenen“ zunehmendes Interesse. Dabei wird aus Geometrie- und Materialbeschreibungen von Szenenobjekten (Position, Orientierung, Gittermodell, Farbe und Textur), sowie aus Kamera- und Beleuchtungsmodellen ein annähernd fotorealistisches Bild der darzustellenden Szene erzeugt, so daß Entwürfe bereits vor ihrer Realisierung betrachtet werden können.

Während Graphiksysteme zunächst darauf ausgelegt waren, die generierten Konstruktionsmodelle dem eher passiven Benutzer nur zur Betrachtung zu präsentieren, werden in jüngerer Zeit *interaktive Graphiksysteme* erprobt, die eine intuitive Manipulation der Objektmodelle in Analogie zu ihren realen Gegenständen und dadurch ein weitergehendes Einbeziehen des Benutzers in den Erstellungsprozeß ermöglichen. Im Bereich der „Virtuellen Realität“ können Benutzer mittels spezieller Ausstattungen in die virtuelle Welt „eintauchen“ und sie mit einem Datenhandschuh auf intuitive Weise direkt manipulieren (siehe auch [WKC94]).

Die möglichen Interaktionen sind allerdings noch recht beschränkt. So sind Montagen komplexer Aggregate mit direkter Manipulation bislang schwierig zu bewerkstelligen. Dabei stellen nicht nur mangelnde Taktilität und fehlendes Kraft-Feedback, mithin die Erfahrbarkeit physikalisch bedingter Eigenschaften Hemmnisse dar, sondern auch Defizite der Datenmodelle. Da Visualisierungen von Objektmodellen zunächst nur Bilder „zum Anschauen“ liefern, ist eine Rückkopplung in eine interne, sinnkonstruierende Repräsentation notwendig. In der Simulation einer Montage kann zum Beispiel das wirklichkeitsgetreue Fügen von Bauelementen nur gelingen, wenn neben den zur Visualisierung benötigten Geometriemodellen der Bauteile auch ihre Verbindungsmöglichkeiten sowie Eigenschaften der Baugruppen als Ganzes berücksichtigt werden.

Der im folgenden beschriebene CODY<sup>1</sup> *Virtuelle Konstrukteur* ist ein wissensbasiertes System, das die interaktive Montage von 3D-computergraphisch dargestellten Bauteilen zu komplexen Aggregaten ermöglicht. Die Interaktion erfolgt für einfache Manipulationen wie Verändern der Betrachterperspektive und Bewegen von Bauteilen über direkte Maussteuerung. Komplexere Manipulationen, wie das Fügen, Auseinandernehmen und Drehen von Teilaggregaten, werden über verbale Anweisungen vermittelt und vom System selbst teilautonom ausgeführt. Hierdurch kann auf umständliche beziehungsweise teure Eingabegeräte, wie zum Beispiel den Datenhandschuh, verzichtet werden.

Um die Umsetzung der Benutzeranweisungen und den paßgenauen Zusammenbau der Bauteile gewährleisten zu können, ist der Virtuelle Konstrukteur mit Wissen über die Fügемöglichkeiten der Bauteile versehen. Ferner werden Modelle sinnvoller Baugruppen in einer statischen Wissensbasis vorgehalten. Darüber hinaus wird die aktuelle Szene mit den Bauteilen und entstandenen Aggregaten in einer dynamischen Wissensbasis semantisch repräsentiert („konzeptualisiert“). Diese Konzeptualisierung wird dynamisch ange-

---

<sup>1</sup>Das Projekt CODY („Concept Dynamics“) ist ein Teilprojekt des DFG-Sonderforschungsbereichs „Situierete Künstliche Kommunikatoren“ (SFB 360) an der Universität Bielefeld. Der Virtuelle Konstrukteur ist als Demonstrator für die in diesem Projekt entwickelten dynamischen Wissensrepräsentationen entstanden. Die Autoren danken Martin Hoffhenke, Christian Pütter und Britta Lenzmann für ihre Mithilfe. Teile dieses Beitrags basieren auf [JLW95].

paßt, um ein intern verarbeitbares logisches Modell der Szene mitzuführen. Durch die Kombination von Auswertungen der Geometriebeschreibung und wissensbasierten Techniken wird die interaktive Manipulation der graphischen Umgebung ermöglicht.

In den folgenden Abschnitten wird der CODY Virtuelle Konstrukteur genauer erläutert (Abschnitt 2), die zugrundeliegenden wissensbasierten Methoden beschrieben (Abschnitt 3) und im Hinblick auf ihre Übertragbarkeit diskutiert (Abschnitt 4).

## 2 Der CODY Virtuelle Konstrukteur

Der Virtuelle Konstrukteur kann als situierter, in die virtuelle Umgebung gekoppelter "Agent" verstanden werden, der die Graphikszene wahrnehmen und in dieser handeln kann [JW95]. Um Bauteile auf der virtuellen Montagefläche zusammensetzen, kommuniziert der Benutzer seine Änderungswünsche mittels natürlichsprachlicher Anweisungen an den Virtuellen Konstrukteur. Dieser fungiert als *Interface-Agent* [Lau90], der die - oft vagen - qualitativen Anweisungen des Benutzers in technische Kommandos zur quantitativen Veränderung der Objektszene übersetzt. Als Erprobungsdomäne dient die Konstruktion eines einfachen Flugzeugmodells aus Holzbauteilen, deren Graphikmodelle auf einer „virtuellen Montagefläche“ assembliert werden (siehe Abb. 1).

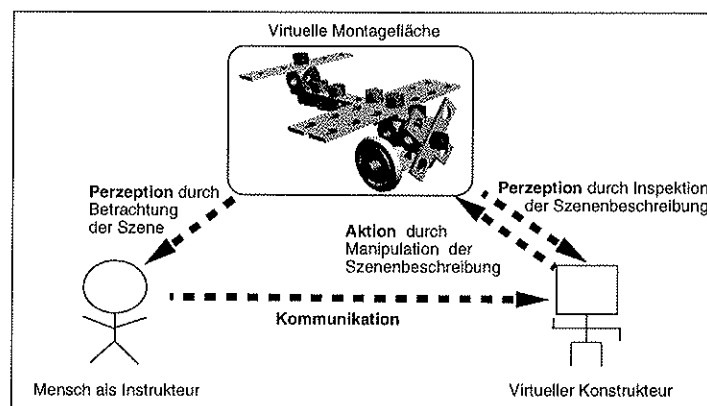


Abbildung 1: Instrukteur-Konstrukteur-Kooperation auf der virtuellen Montagefläche

Zu Beginn einer Sitzung werden dem Benutzer die Bauteile auf der virtuellen Montagefläche präsentiert. Im Verlauf einer Sitzung können zusätzliche Bauteile angefordert werden. Zur Manipulation der Grafik stehen dem Benutzer folgende Möglichkeiten zur Verfügung:

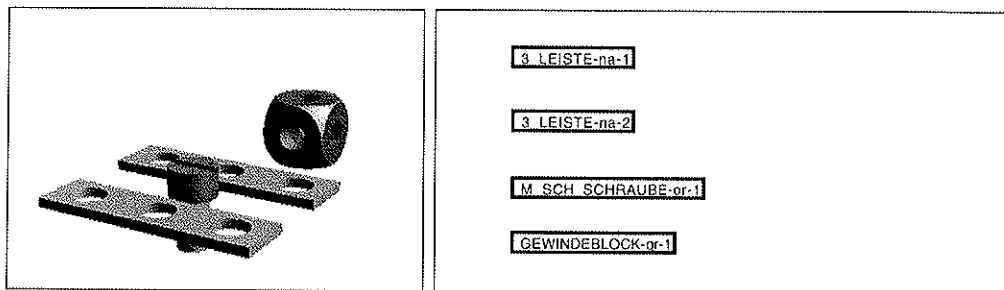
1. Verändern der Betrachterperspektive durch Bewegen des Betrachterstandorts und Ändern der Blickrichtung
2. Bewegen und Drehen von Bauteilen und Aggregaten
3. Fügen und Auseinandernehmen von Bauteilen und Aggregaten
4. Drehen von einzelnen Bauteilen in Aggregaten

Während 1. und 2. durch direkte (mausgesteuerte) Manipulation vorgenommen werden, werden 3. und 4. über verbale Anweisungen – derzeit über Tastatur – vermittelt.

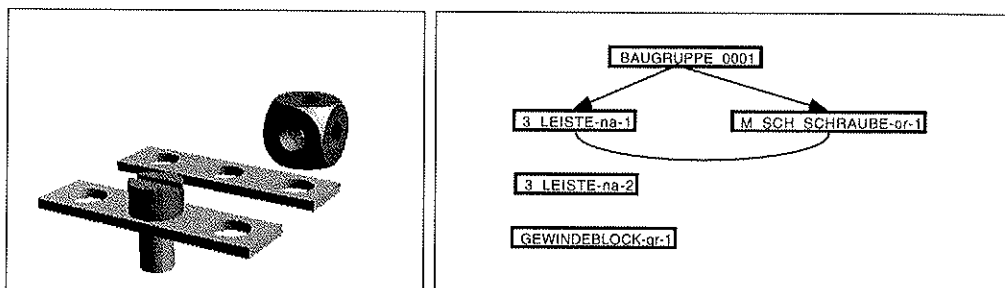
In den verbalen Anweisungen kann Bezug genommen werden auf den Typ der Bauteile (Schraube, Leiste etc.), auf verschiedene Objekteigenschaften (zum Beispiel Farbe, Form, Größe und Funktion) sowie auf die räumliche Lage der Konstruktionselemente (zum Beispiel *links*, *rechts* oder Kombinationen wie *links oben*). Neben einzelnen Bauteilen können auch bereits gebaute Aggregate angesprochen werden. Da die Objekte anhand ihrer wahrnehmbaren Eigenschaften benannt werden können, ist es nicht notwendig, daß dem Benutzer interne Objektbezeichner (z.B. PROPELLER\_0001) bekannt sind. Anstelle exakter Typbezeichnungen können Bauteile und Aggregate auch allgemein als *Ding* angesprochen werden. Bei der Auswertung solcher Anweisungen berücksichtigt der Virtuelle Konstrukteur die aktuelle Anordnung der Bauteile auf der Montagefläche, wie sie vom Benutzer aus seiner aktuellen Perspektive wahrgenommen wird, aber auch dynamisch aufgebautes Wissen über die jeweils vorhandenen Bauteile und Aggregate.

### Beispiel

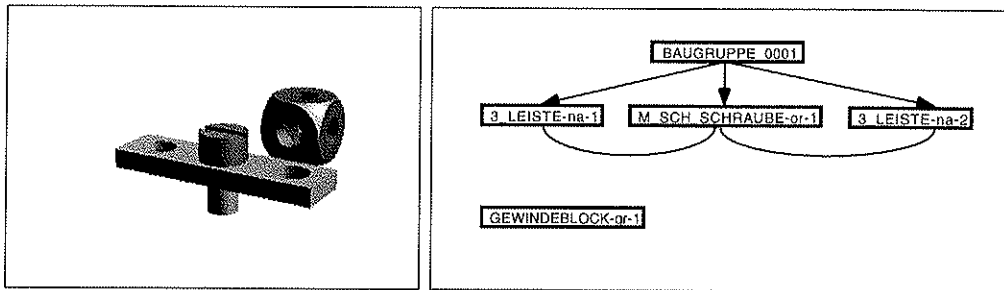
In den folgenden Beispielinteraktionen wird der Propeller des Flugzeugmodells konstruiert und an einen Gewindeblock montiert. Neben der graphischen Objektszene werden auch die dynamisch aktualisierten internen Objektrepräsentationen und -hierarchien dargestellt. Teil-von-Beziehungen zwischen gefügten Aggregaten und ihren Teilen werden durch Pfeile dargestellt, hergestellte Verbindungen durch einfache Linien.



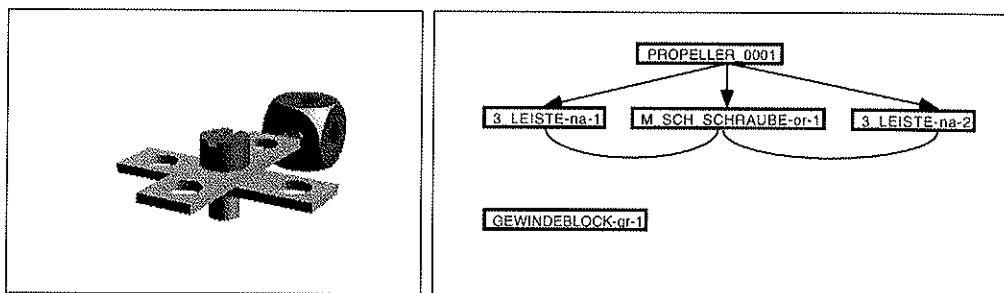
EINGABE: stecke die Schraube in die Mitte der vorderen Dreierleiste



EINGABE: stecke die Baugruppe in die Mitte der anderen Leiste

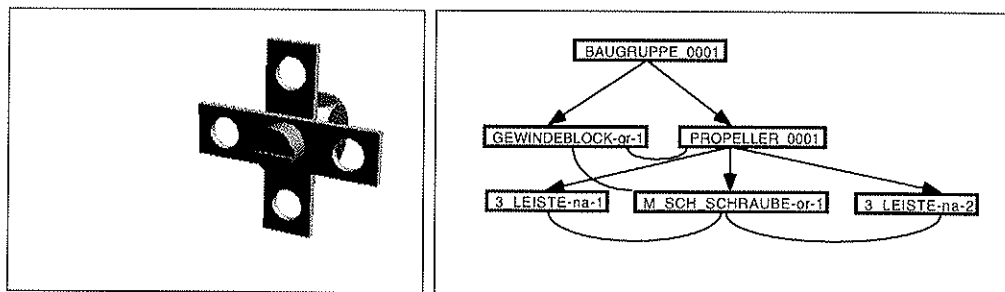


EINGABE: aber quer

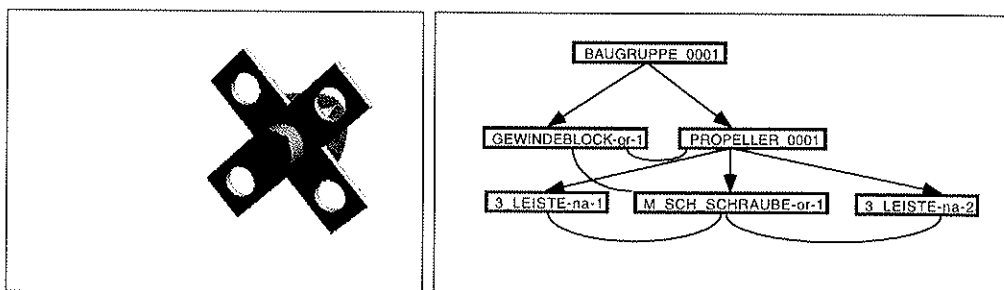


Das System hat erkannt, daß ein Propeller entstanden ist, der als logisches Modell im Hintergrundwissen generisch vordefiniert ist.

EINGABE: stecke den Propeller vorne an den Block



EINGABE: drehe den Propeller schraeg zum Block



Auf ähnliche Weise können mit dem CODY Virtuellen Konstrukteur beliebige Aggregate aus den Teilen des Beispielszenarios konstruiert werden. Routinen zur Kollisionserkennung verhindern den Zusammenbau physikalisch unmöglicher Aggregate.

### 3 Wissensverarbeitung

Die Systemfunktionalität des Virtuellen Konstrukteurs beruht wesentlich auf dem Einsatz wissensbasierter Methoden zur semantischen Repräsentation der Objektszene. Die Systemarchitektur ist schematisch in Abb. 2 dargestellt. Das statische Vorwissen des Virtuellen Konstrukteurs wird in vier Wissensbasen definiert. Eine Wissensbasis enthält die Geometriemodelle der Bauteile, die um Beschreibungen ihrer Kopplungsstellen erweitert werden. In der zweiten Wissensbasis werden verschiedene qualitative räumliche Relationen über den Geometriemodellen definiert. Die beiden anderen Wissensbasen definieren konzeptuelles Wissen über die multifunktionalen Bauteile (in einer Wissensbasis **Bauteilwelt**) einerseits, sowie das Konstruktionsziel (in einer Wissensbasis **Flugzeugwelt**) andererseits. Das dynamische Wissen des Virtuellen Konstrukteurs über die aktuelle Objektszene umfasst neben den visualisierten Geometriemodellen auch logische Objektrepräsentationen, die die Bauteile bezüglich der beiden Wissensbasen dynamisch konzeptualisieren. Bei der Auswertung und Umsetzung von Benutzeranweisungen wird sowohl auf die geometrische, als auch auf logische Objektbeschreibungen zurückgegriffen.

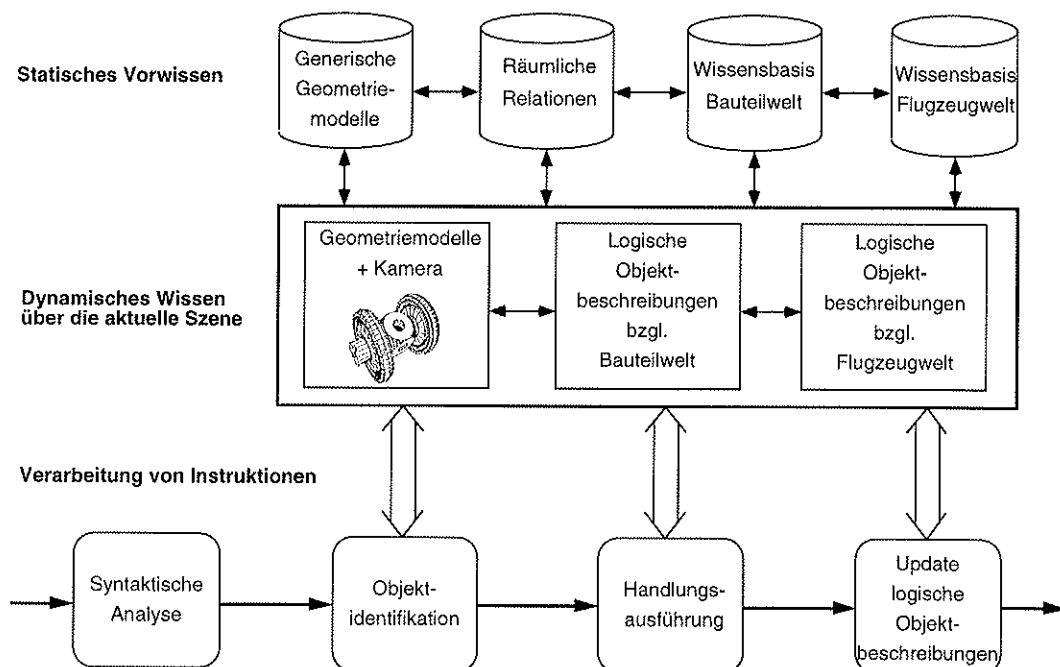


Abbildung 2: Systemwissen und Verarbeitung von Instruktionen

Im Rest dieses Kapitels wird zunächst der Repräsentationsformalismus dargestellt, in dem die strukturierten Beschreibungen der Bauteile und Baugruppen in den Wissensbasen Bauteilwelt und Flugzeugwelt definiert werden. Dann werden die insgesamt vier statischen Wissensbasen und die entsprechenden dynamischen Wissensbasen einzeln beschrieben. Schließlich wird auf den Gebrauch der logischen Objektbeschreibungen bei der Auswertung von Benutzerinstruktionen sowie deren Aktualisierung nach Montageschritten eingegangen.

### 3.1 Wissensrepräsentationssprache

Die strukturierten Beschreibungen der Bauteile und Baugruppen des Zielaggregats in den Wissensbasen Bauteilwelt und Flugzeugwelt werden in einer speziell entwickelten Beschreibungssprache definiert, die auf dem Semantischen-Netzwerk-Formalismus ERNEST [Sag90] und der in [PL94] vorgestellten terminologischen Sprache für Bestandteilsbeziehungen aufbaut. Der Formalismus unterstützt speziell die getrennte Modellierung der multifunktionalen Bauteile und ihrer spezifischen Funktionen in größeren Baugruppen. Dadurch können kontextabhängige Merkmale und Benennungen definiert und vom System zur Laufzeit berechnet werden.

Die Konzepte können über eine Spezialisierungs-, sowie eine Bestandteilsrelation hierarchisch angeordnet werden. Die erlaubten Fügungen zwischen Konstruktionselementen werden mittels einer Verbindungsrelation definiert. Attribute beschreiben objektintrinsische Eigenschaften, wie z.B. Farben oder Schaftkapazitäten. Konzepte auf unterschiedlichen Abstraktionsebenen werden durch eine Konkretisierungsrelation zueinander in Beziehung gesetzt. Hierbei werden die Konzepte der Bauteilwelt konkreter als die Konzepte der Flugzeugwelt angenommen. Zum Beispiel ist das Konzept 3-LEISTE konkreter als das Konzept ROTORBLATT (Dreilochleisten fungieren, wenn sie im Propeller verwendet werden, als Rotorblatt).

Beziehungen zwischen den Bestandteilen von Baugruppen werden durch strukturelle Beschreibungen von Randbedingungen (Constraints) ausgedrückt. Da hier nur Beziehungen zwischen genau zwei Bestandteilen betrachtet werden, werden diese, der Notation von [PL94] folgend, als part-part-constraints (PP-Constraints) bezeichnet. Dabei wird zwischen zwei Arten von Constraints unterschieden: *Logische* Constraints verlangen die Existenz logischer Relationen, z.B. Verbindung, zwischen Bestandteilen (nur diese werden in [PL94] betrachtet). *Geometrische* Constraints, z.B.  $\text{ortogonal}_x$ , verlangen dagegen, daß entsprechende räumliche Relationen zwischen den zugehörigen Geometriemodellen gelten. Diese werden bei Bedarf durch spezielle Testprozeduren zur Laufzeit ermittelt.

Die Beschreibungsmittel für die konzeptuellen Objektrepräsentationen können wie folgt in abstrakter Syntax zusammengefaßt werden:

Konzept: *Konzeptname*

Spezialisierung-von: *Konzeptname*\*

(Bestandteil *Relationenname* #anz: *Konzeptname*)\*

(Verbindung *Relationenname*: *Konzeptname*)\*

(Konkretisierung *Relationenname*: *Konzeptname*)\*

(Attribut *Attributname*: *Wertebereich*)\*

(PP-Constraint *Relationentyp*

<Pfad von Relationennamen> <Pfad von Relationennamen>)\*

### 3.2 Geometriemodelle

Die Geometriemodelle der Bauteile dienen dem System einerseits zur Visualisierung der aktuellen Objektszene, werden darüber hinaus aber auch als Informationsquelle über die räumliche Anordnung und hergestellten Verbindungen der Bauteile eingesetzt.

Zur Visualisierung der Objektszene enthalten die Geometriemodelle u.a. die Gittermodelle der Bauteile. Die Gittermodelle werden mit geeigneter 3D-Modellierungssoftware, z.B. CAD-Systemen, erstellt und können dann in das System importiert werden. In den Geometriemodellen wird auch die prototypische Orientierung der Bauteile festgelegt, welche als Referenz bei der Bestimmung der Orientierung von konkreten Bauteilen dient. Bei der Modellierung der prototypischen Orientierung von, zum Beispiel, verschieden langen Leisten ist darauf zu achten, daß diese dieselbe Ausrichtung besitzen: andernfalls könnte nicht bestimmt werden, ob zwei (verschieden lange) Leisten auf der virtuellen Montagefläche z.B. parallel zueinander sind. Ein umschreibender Quader (*"Bounding Box"*) dient dem System zum einen bei der Durchführung von Montageschritten in der Objektszene zur Erkennung von Objektkollisionen; zum anderen wird der umschreibende Quader aber auch bei der Berechnung von räumlichen Objektmerkmalen, zum Beispiel Größe und Form von Bauteilen und insbesondere von dynamisch erzeugten Aggregaten verwendet.

Darüber hinaus enthalten die Geometriemodelle auch Beschreibungen der Verbindungsstellen (*"Ports"*) der Bauteile, deren Position und Orientierung relativ zu dem zugehörigen Bauteil angegeben werden (eine ähnliche Erweiterung von CAD-basierten Bauteilbeschreibungen um ihre Verbindungsstellen wird zum Beispiel in [BB95] vorgeschlagen). Abb. 3 zeigt die geometrische Beschreibung einer individuellen Dreilochleiste. Die Dreilochleiste enthält sechs Verbindungsports, nämlich je zwei Lochseiten für die drei Löcher. Durch die Modellierung der Lochseiten kann unterschieden werden, von welcher Seite z.B. eine Schraube durch ein Loch gesteckt wird. Die geometrischen Beschreibungen konkreter Bauteile enthalten weitere für ihre Visualisierung relevante Eigenschaften, wie zum Beispiel deren absolute Position und Orientierung.

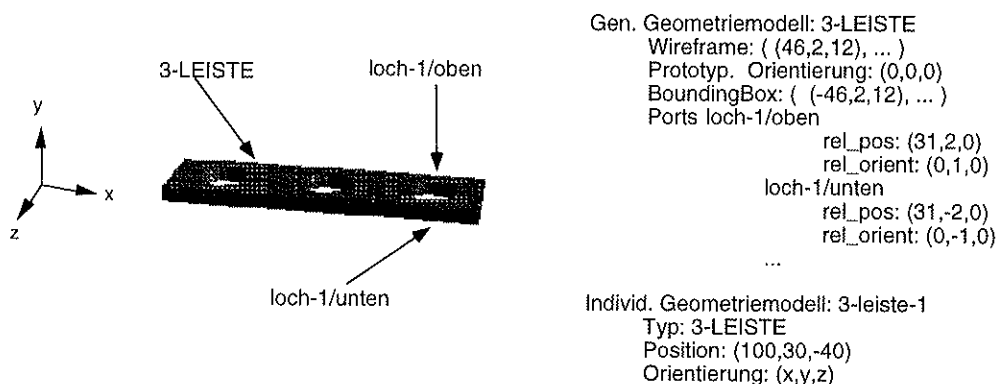


Abbildung 3: Geometrische Beschreibung einer individuellen Dreilochleiste als Instanz ihres generischen Modells (Ausschnitt)

Geometrische Repräsentationen von entstandenen Aggregaten bestehen aus einer Liste ihrer Bestandteile sowie einem umschreibenden Quader, der in der Visualisierung optional angezeigt werden kann, um die Gruppierung von Baugruppen zu verdeutlichen. Das geometrische Wissen im Arbeitsspeicher enthält zusätzlich ein Modell einer Kamera, die das aktuelle Blickfeld des Benutzers festlegt. Standort, Ausrichtung und Zoomeinstellung der Kamera können interaktiv verändert werden.



### 3.3 Räumliche Relationen

Über den Geometriemodellen von Bauteilen und Aggregaten werden verschiedene qualitative räumliche Relationen definiert. Die derzeit realisierten räumlichen Relationen sind  $\text{parallel}_x$ ,  $\text{parallel}_y$ ,  $\text{parallel}_z$ ,  $\text{orthogonal}_x$ ,  $\text{orthogonal}_y$ ,  $\text{orthogonal}_z$ ,  $\text{berührt}$  und  $\text{neben}$ . Dazu kommt eine Relation  $\text{verbindung}$ , für deren Definition geometrisches und konzeptuelles Wissen benötigt wird. Die Verbindungsrelation ist nur zwischen den Portbeschreibungen der Geometriemodelle definiert (genauer: zwischen je einem "Nehmerport" und einem "Geberport", s.u.).

Die räumlichen Relationen können bei Bedarf jederzeit aus der geometrischen Szenenbeschreibung berechnet werden. Indem nach jeder Fügeoperation auf der Montagefläche die entstandenen Objektverbindungen inferiert werden, können logische Repräsentationen für entstandene Aggregate erzeugt werden. Die räumlichen Relationen finden auch in Konzeptdefinitionen für die Flugzeugbaugruppen zur Beschreibung der räumlichen Anordnung ihrer Bestandteile Verwendung. Zum Beispiel müssen die beiden Rotorblätter des Propellers  $\text{orthogonal}_x$  zueinander stehen (s.u.).

### 3.4 Konzeptuelles Wissen über die Bauteile

Konzeptuelles Wissen über die Bauteile wird in einer Wissensbasis *Bauteilwelt* in der oben beschriebenen Beschreibungssprache modelliert. Neben Konzepten für die einzelnen Bauteile des Szenarios ist ein allgemeines, unstrukturiertes AGGREGAT-Konzept definiert, das mindestens zwei miteinander verbundene Bauteile als Bestandteil hat. Zur Laufzeit des Systems werden miteinander verbundene Bauteile durch Instanziierung dieses allgemeinen Konzepts zu Aggregaten zusammengefaßt. Darüber hinaus sind die PORTs der Bauteile definiert, wobei zwischen Geberports, z.B. dem Schraubschaft, und Nehmerports, z.B. den aufnehmenden Lochseiten, unterschieden wird. Schließlich wird ein LOCH-Konzept definiert (modelliert als Paar gegenüberliegender Lochseiten), das Löcher mit und ohne Innengewinde umfaßt. Im Gegensatz zu den zweidimensionalen Lochseiten sind Löcher dreidimensionale Entitäten, die die Berechnung von konsumierten Kapazitäten von Steckverbindungen ermöglichen.

Abbildung 4 zeigt die logische Beschreibung einer Dreilochleiste im Arbeitsspeicher. Die Dreilochleiste hat drei Löcher als "Bestandteile", die wiederum jeweils zwei Lochseiten als Bestandteile besitzen. Die Leiste und die Lochseiten haben Entsprechungen in der geometrischen Beschreibung aus Abb. 3 (die Löcher der Leiste haben keine entsprechenden Beschreibungen in den Geometriemodellen).

### 3.5 Konzeptuelles Wissen über das Konstruktionsziel

Wissen über das Zielaggregat (hier: Flugzeugmodell) wird in einer weiteren Wissensbasis *Flugzeugwelt* modelliert. Die Konzepte dieser Wissensbasis definieren die Baugruppen des Flugzeugmodells sowie die spezifischen funktionalen Eigenschaften einzelner Bauteile in diesen Baugruppen. Dazu kann in den Konzeptbeschreibungen auf die räumlichen Eigenschaften der Bauteile, die aus deren Geometriemodellen abgeleitet werden, und auf die Konzeptbeschreibungen aus der Wissensbasis *Bauteilwelt* Bezug genommen werden.

Objektbeschreibung: leiste-1		Objektbeschreibung: leiste-1/loch-1/s1	
<u>Instanz-von</u>	3-LOCHLEISTE	<u>Instanz-von</u>	LOCHSEITE
<u>Attribut</u>	farbe: natur	<u>Attribut</u>	belegt: false
<u>Attribut</u>	form: eckig	<u>Bestandteil</u>	-
<u>Bestandteil</u>	hat-loch-1: leiste-1/loch-1	<u>Verbindung</u>	-
<u>Bestandteil</u>	hat-loch-2: leiste-1/loch-2		
<u>Bestandteil</u>	hat-loch-3: leiste-1/loch-3	Objektbeschreibung: leiste-1/loch-1/s2	
<u>Verbindung</u>	-	<u>Instanz-von</u>	LOCHSEITE
		<u>Attribut</u>	belegt: false
Objektbeschreibung: leiste-1/loch-1		<u>Bestandteil</u>	-
<u>Instanz-von</u>	LOCH	<u>Verbindung</u>	-
<u>Attribut</u>	gesamt_kap: 4		
<u>Attribut</u>	aktuelle_kap: 4		
<u>Bestandteil</u>	hat-seite-1: leiste-1/loch-1/s1		
<u>Bestandteil</u>	hat-seite-2: leiste-1/loch-1/s2		
<u>Verbindung</u>	-		

Abbildung 4: Logische Beschreibung einer Dreilochleiste bezüglich der Wissensbasis Bauteilwelt (Ausschnitt)

Ein Konzept BAUGRUPPE umfaßt die komplexen Komponenten des Flugzeugmodells, z.B. das Fahrwerk, das Leitwerk und das Flugzeug selbst. Ein Konzept FLUGZEUGTEIL umfaßt funktionale Beschreibungen einzelner Bauteile im Flugzeugmodell. Beispiele sind Konzepte für Achsen, Felgen, Rotorblätter etc.

Konzept: PROPELLER		Konzept: ROTORBLATT	
<u>Spezialisierung-von</u>	BAUGRUPPE	<u>Spezialisierung-von</u>	FLUGZEUGTEIL
<u>Bestandteil</u>	"hat-rotorblatt-1" 1..1: ROTORBLATT	<u>Konkretisierung</u>	3-LEISTE
	"hat-rotorblatt-2" 1..1: ROTORBLATT		
	"hat-propschraube" 1..1: PROPELLERSCHRAUBE		
<u>PP-Constraint</u>	Verbindung "hat-rotorblatt-1" "hat-propschraube"		
	Verbindung "hat-rotorblatt-2" "hat-propschraube"		
	orthogonal <sub>x</sub> "hat-rotorblatt-1" "hat-rotorblatt-2"		


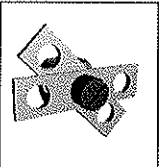



Abbildung 5: Definition des Propellers mit geometrischem Constraint (oben). Darstellung: Propeller (unten links), kein Propeller (unten rechts)

Abb. 5 zeigt die Definition des Konzepts PROPELLER. Der Propeller besteht aus einer Propellerschraube sowie zwei Rotorblättern. Die Bedingung, daß die beiden Rotorblätter des Propellers quer zueinander stehen müssen, wird durch einen geometrischen Constraint  $orthogonal_x$  ausgedrückt. In der unteren Hälfte der Abbildung ist links ein Propeller zu sehen und rechts ein komplexes Aggregat, das aus denselben Bestandteilen besteht, bei dem aber das genannte geometrische Constraint verletzt ist.

Im Arbeitsspeicher werden dynamisch aktualisierte logische Objektbeschreibungen mitgeführt, die die geometrische Objektszene bezüglich der Wissensbasis Flugzeugwelt konzeptualisieren. Die Aktualisierungen dieser logischen Objektbeschreibungen beinhalten die Erkennung konstruierter Baugruppen des Flugzeugmodells mittels einer Va-

riante von *Compositional Inferencing* (vgl. [PL94]), sowie die Zuschreibung von kontextabhängigen funktionalen Merkmalen an die Bauteile, wenn sie in Baugruppen verwendet werden.

### 3.6 Verarbeitung von Instruktionen

Konzeptuelle und geometrische Szenenbeschreibungen werden herangezogen, um verbale Montageanweisungen auszuwerten und auszuführen. Durch dynamisch aktualisierte Konzeptualisierung der geometrischen Szenenbeschreibung können sich die Montageanweisungen zu jedem Zeitpunkt begrifflich auf die aktuell für den Benutzer sichtbare Szene beziehen.

Bei der Verarbeitung von Instruktionen werden zunächst die verbalen Eingaben syntaktisch analysiert und dann die benannten Objekte identifiziert. Bei Füge- und Trennanweisungen sind auch die jeweiligen Ports der benannten Objekte eindeutig zu bestimmen. Bei der Auswertung der oft unterspezifizierten Objektbenennungen werden verschiedene Heuristiken, wie zum Beispiel die Nähe der Bauteile zum Betrachter und der bisherige Montageverlauf, berücksichtigt.

Danach wird die Handlung auf der virtuellen Montagefläche durchgeführt, indem die Zielpositionen der betroffenen Bauteil-Geometriemodelle berechnet werden. Da bei der Handlungsausführung physikalische Randbedingungen berücksichtigt werden, z.B. sind miteinander verbundene Bauteile gemeinsam zu bewegen und Objektüberlagerungen auszuschließen, können die spezifizierten Handlungen fehlschlagen oder Seiteneffekten unterliegen.

Bei der anschließenden Aktualisierung der logischen Objektrepräsentationen wird daher in einem Fokusgebiet überprüft, welche Veränderungen tatsächlich aufgetreten sind. Dabei wird zunächst nur überprüft, ob neue Verbindungen zwischen Bauteilports entstanden sind bzw. bisherige Verbindungen zerstört wurden, und ob – nach Rotationen – die geometrischen Constraints von Baugruppen noch erfüllt sind. Neue Portverbindungen werden in den logischen Objektrepräsentationen in Form von qualitativen Relationen gespeichert und entlang von Bestandteil-von- und Konkretisierung-von-Relationen propagiert. Es werden unstrukturierte Aggregate eingeführt und mit den Modellen der Flugzeugbaugruppen abgeglichen. Für entstandene Flugzeugbaugruppen werden neue logische Objektrepräsentationen eingeführt. Die Bestandteile von Baugruppen werden dynamisch entsprechend ihrer jeweiligen Funktion konzeptualisiert, indem sie bezüglich den Konzepten der Flugzeugwelt reklassifiziert werden. Schließlich wird die veränderte Objektszene visualisiert.

## 4 Diskussion

Mit dem Virtuellen Konstrukteur wurde ein System entwickelt, das eine interaktive Simulation einfacher Montagen durch Einbezug wissensbasierter Methoden ermöglicht. Eine Besonderheit des Systems ist die Verwendung natürlicher Sprache an der Mensch-Maschine-Schnittstelle. Das System umfaßt neben der 3D-computergraphischen Oberfläche und den beschriebenen Wissensrepräsentationsmöglichkeiten einen Browser für die

Inspektion der dynamisch aktualisierten Objektmodelle und -hierarchien. Mit einem Skriptrekorder können durchgeführte Montageabläufe als Sequenz der anstoßenden Instruktionen aufgezeichnet und später als Animation wiederholt werden.

Der vorgestellte Ansatz basiert auf einer engen Kopplung geometrischer und konzeptueller Objektbeschreibungen. Die interaktive Montage der Bauteile wird ermöglicht, indem die geometrischen Bauteilbeschreibungen um logische Modelle ihrer Verbindungsstellen angereichert werden. Andererseits ist zur Beschreibung von komplexen Baugruppen i.a. sehr spezifisches Wissen über die räumliche Anordnung ihrer Teile notwendig. Dazu können räumliche Relationen, die aus den Geometriemodellen der Bauteile abgeleitet werden, in die konzeptuellen Beschreibungen aufgenommen werden. Des Weiteren wird in der Modellierung strikt zwischen den multifunktionalen Bauteilen und deren spezifischen Funktionen in größeren Aggregaten unterschieden. Beziehungen zwischen Bauteilen und ihren möglichen Funktionen werden im Hintergrundwissen über eine besondere semantische Relation ("Konkretisierung") modelliert. Die logischen Objektbeschreibungen im Arbeitsspeicher werden fortlaufend aktualisiert, indem Aggregatrepräsentationen entsprechend dem jeweiligen Montagestand erzeugt und gelöscht werden und die Funktion von Bauteilen in Abhängigkeit von ihrem Aggregatkontext berechnet wird. Schließlich wird das Wissen über die Bauteile und das Zielaggregat in getrennten Wissensbasen modelliert. Diese Modularität des Hintergrundwissens unterstützt die Wiederverwendbarkeit der Wissensbasen. Zum Beispiel kann die Wissensbasis Bauteilwelt bei der Modellierung beliebiger Zielaggregate verwendet werden.

Die Übertragbarkeit der beschriebenen Verfahren auf CAD-basierte Konstruktionsdomänen erscheint möglich. Hierzu müßten die CAD-Modelle gemäß der Fügемöglichkeiten durch eine logische Beschreibung vervollständigt und entsprechendes Konstruktionswissen modelliert werden. Das System ist in C++ implementiert und derzeit auf SGI-Workstations einsetzbar.

## Literatur

- [BB95] K. Büttner and H. Birkhofer. Design of modular products using a CAD-based configuration system. In *Proc. International Conference on Engineering Design ICED-95*, 1995.
- [CJW95] Y. Cao, B. Jung, and I. Wachsmuth. Situated verbal interaction in virtual design and assembly, IJCAI-95 Videotape Program. In *Proc. International Joint Conference on Artificial Intelligence*, volume 2, pages 2061–2062. IJCAI, Morgan Kaufman, 1995.
- [JLW95] B. Jung, B. Lenzmann, and I. Wachsmuth. Interaktive Montage-Simulation mit wissensbasierter Grafik. In G. Lux and U. Schroeder, editors, *8. Workshop Simulation und Künstliche Intelligenz*, 1995. ASIM-Mitteilungen 49. Auch als SFB 360 Report 95/6, Universität Bielefeld.
- [JW95] B. Jung and I. Wachsmuth. Situierete Instruktionsverarbeitung im CODY Virtuellen Konstrukteur. In L. Dreschler-Fischer and S. Pribbenow, editors, *KI-95 Activities: Workshops, Posters, Demos*, pages 332–334. GI, 1995.
- [Lau90] B. Laurel. *The art of human-computer interface design*. Addison-Wesley, New York, 1990.

- [PL94] L. Padgham and P. Lambrix. A framework for part-of hierarchies in terminological logics. In *Principles of Knowledge Representation and Reasoning*, pages 485–496. Morgan Kaufmann, San Francisco, CA, 1994.
- [Sag90] G. Sagerer. *Automatisches Verstehen gesprochener Sprache*. Bibliographisches Institut, Mannheim, 1990.
- [WKC94] I. Wachsmuth, W. Krüger, and Y. Cao. Virtuelle Räume. *KI-94*, 1:26–33, 1994.