

Mitteilung Nr. 135

**WISSENSREPRÄSENTATION
FÜR FORTGESCHRITTENE COMPUTER-ANWENDUNGEN**

BERND NEUMANN

IFI-HH-M-135/85

Oktober 1985

Fachbereich Informatik
Universität Hamburg
Schlüterstr. 70
D-2000 Hamburg 13

Dieser Bericht basiert auf einem Beitrag zur PROMED-Tagung, Düsseldorf,
16.-19. September, 1985

WISSENSREPRÄSENTATION FÜR FORTGESCHRITTENE COMPUTER-ANWENDUNGEN

Bernd Neumann
Fachbereich Informatik
Universität Hamburg

Kurzfassung

Der Beitrag führt zunächst in das Forschungsgebiet "Künstliche Intelligenz" ein. Es wird gezeigt, welche Rolle Wissensrepräsentation bei der Lösung komplexer Aufgaben spielt. Der Hauptteil des Beitrages befaßt sich mit verschiedenen Techniken der Wissensrepräsentation und zeigt dabei Vor- und Nachteile der einzelnen Verfahren auf. Der Unterschied zwischen konventioneller und wissensbasierter Systemarchitektur wird verdeutlicht.

1. Was ist "Künstliche Intelligenz"?

Wissensrepräsentation ist ein zentraler Forschungsgegenstand der Künstlichen Intelligenz (KI). Da sich dieser Beitrag auch an fachfremde Leser wendet, wird im folgenden kurz dargestellt, um was es in der KI geht und an welcher Stelle Wissensrepräsentation zum Tragen kommt.

Die Bezeichnung des Fachgebietes ist eine Übersetzung des amerikanischen "Artificial Intelligence". Dieser schillernde Titel ist mittlerweile 20 Jahre alt und kann wohl - sehr zum Bedauern vieler KI-Forscher - nicht mehr durch einen seriösen Namen ersetzt werden. Denn kaum jemand aus dem KI-Bereich hat eine klare Definition des Begriffs 'Intelligenz' parat. In der Tat ist 'Intelligenz' aus der Sicht der KI eine wissenschaftliche Leerformel - im Gegensatz beispielsweise zur Experimentellen Psychologie, in der 'Intelligenz' aufgrund anderer wissenschaftlicher Methoden sehr wohl definiert ist.

KI umschreibt ein Forschungsgebiet, in dem es um Computerlösungen für Probleme geht, die, falls sie ein Mensch lösen müßte, Intelligenz erfordern. Dabei wird der Mensch in ganz unterschiedlicher Weise zum Vorbild genommen. In einer erst kürzlich erschienenen KI-Einführung (CHARNIAK und McDERMOTT 85) wird KI beispielsweise als "Erforschung mentaler Prozesse mithilfe von Berechnungsmodellen" definiert. Hier steht also eine Erklärung menschlicher Intelligenz im Vordergrund. Aus der Sicht vieler anderer KI-Forscher geht es jedoch in erster Linie um Rechnerlösungen für gewisse komplexe Probleme - eben solche, die gemeinhin beim Menschen Intelligenz erfordern.,

Wie man auch zur Definition von KI steht - das Forschungsgebiet KI existiert und ist darüberhinaus derzeit in einer lebhaften Expansion begriffen. Eine erste Vorstellung von den in der KI behandelten Problemen vermitteln die primären Anwendungsgebiete:

- Natürlichsprachliche Systeme
(z.B. Befragung von Datenbanken in natürlicher Sprache)

- Bildverstehen
(z.B. Auswertung von Kamerabildern zur Lenkung eines Fahrzeugs)

- Robotik
(z.B. selbsttätige Planung bei einer Montageaufgabe)

- Expertensysteme
(z.B. Beratung bei der medizinischen Diagnose)

- Deduktionssysteme
(z.B. Korrektheitsprüfungen für Rechnerprogramme)

- Kognitive Psychologie
(Modelle für kognitive Prozesse, z.B. Lernen)

Obwohl jedes Teilgebiet seine eigene Problematik hat und auch spezifische, anwendungsbezogene Lösungen kennt, gibt es dennoch gemeinsame methodische Grundlagen, die den eigentlichen Kern der KI ausmachen. Die wichtigsten

Methoden sind

- Heuristische Suche
- Deduktion
- Wissensrepräsentation
- Informelles Schließen
- Wissenserwerb

sowie - auf niedrigerer Ebene - spezielle KI-Programmiermethoden.

Die Bedeutung von Suchverfahren, speziell der Heuristischen Suche (bei der man Wissen zur Reduktion des Suchaufwandes heranzieht), kommt daher, daß es für typische KI-Probleme im allgemeinen keine Lösungsverfahren gibt, die direkt zum Ziel führen. Vielmehr ist man meist auf geschicktes Probieren, auf das Verfolgen und Verwerfen von unsicheren Hypothesen und das Generieren vieler Alternativen angewiesen.

Deduktionstechniken benutzen logische Zusammenhänge zur Lösung von Problemen. Hierfür wird weiter unten im Zusammenhang mit logischer Wissensrepräsentation ein ausführliches Beispiel gebracht.

Wissensrepräsentation - das engere Thema dieses Beitrages - spielt in allen Anwendungsgebieten eine fundamentale Rolle. Dies war durchaus nicht von Anfang an klar. Wesentliche Forschungsarbeiten der sechziger Jahre hatten zum Ziel, allgemeine und anwendungsunabhängige Problemlösungstechniken zu entwickeln, wie z.B. der 'General Problem Solver' in ERNST und NEWELL 69. Erst später wurde deutlich, daß die Leistungsfähigkeit eines Problemlösers entscheidend von Wissen über das Problemgebiet abhängt.

Die letztgenannten Methoden - Informelles Schließen und Wissenserwerb - sind aktueller Forschungsgegenstand. Informelles Schließen bezeichnet die für Menschen charakteristische Fähigkeit, aus unsicherem oder unvollständigen Fakten vernünftige Folgerungen abzuleiten. Die Bedeutung derartiger Verfahren hängt eng damit zusammen, daß Alltags- und Erfahrungswissen bei vielen fortgeschrittenen KI-Anwendungen mit einbezogen werden muß.

Wissenserwerb schließlich ist eine Aufgabe, die derzeit als der Flaschenhals vieler KI-Anwendungen angesehen wird. Hier geht es um systematische Verfahren, mit denen das für ein KI-System erforderlichen Wissen - z.B. menschliches Expertenwissen - aufbereitet und in eine rechneradäquate Form gebracht werden kann. Menschliche Spezialisten, die dies beherrschen, werden gerne 'Wissensingenieure' genannt. Es gibt allerdings bisher kaum Techniken für eine ingenieurmäßige Lösung dieser Problematik.

2. Was sind "wissensbasierte Systeme"?

Die Bedeutung von Wissen für Problemlösungen in der KI wurde bereits hervorgehoben. In diesem Abschnitt soll nun in größerem Detail erläutert werden, welche Arten von Wissen eine Rolle spielen und was unter einem wissensbasierten System zu verstehen ist.

Eine Gliederung in verschiedene Arten von Wissen ergibt sich, wenn man danach fragt, welche Informationen für die Bearbeitung eines Problems herangezogen werden müssen. Als erstes ist Wissen über den Aufgabenbereich ("Domänenwissen") zu nennen. Hierzu gehören die Gesetzmäßigkeiten, typische Probleme und Lösungsmethoden, aber auch die charakteristischen Eigenschaften und Beziehungen von Objekten des Aufgabenbereiches. Bei einem Expertensystem für medizinische Diagnose, beispielsweise, ist unter Domänenwissen medizinisches Fachwissen zu verstehen. Als 'Objekte' des Aufgabenbereichs sind in diesem Fall der Mensch und seine biologischen Komponenten anzusehen, aber auch Zellstrukturen, bakterioide Organismen, etc. Domänenwissen wird in Expertensystemen häufig in Form von WENN-DANN-Regeln gespeichert. Die folgende Regel ist Bestandteil der Wissensbasis von MYCIN, einem vielbeachteten Expertensystem zur Diagnose bakterieller Infektionen und zur Therapieberatung (BUCHANAN und SHORTLIFFE 84).

- WENN
- 1) die Gram-Färbung eines Organismus Gram-negativ ist, und
 - 2) die Form des Organismus stäbchenförmig ist, und
 - 3) die Aerobizität des Organismus anaerobisch ist,

DANN: ist die Identität des Organismus wahrscheinlich (0.6) bakteroid.

Dies ist eine Paraphrase der weniger leicht verständlichen Originalform:

```
(PREMISE:      ($AND (SAME CNTXT GRAM GRAMNEG)
                   (SAME CNTXT MORPH ROD)
                   (SAME CNTXT AIR ANAEROBIC))
ACTION:      (CONCLUDE CNTXT IDENTITY BACTEROIDS TALLY .6))
```

Ein zweiter Wissensbereich bezieht sich auf die konkrete Aufgabe. Hierunter fallen alle Angaben zur Problemstellung, z.B. die Symptome eines Patienten und der Befund von Labortests. Weiterhin gehören Anforderungen an die Lösung dazu, z.B. die zur Verfügung stehende Rechenzeit oder die erforderliche Zuverlässigkeit. Problemwissen entspricht wohl nicht unserem intuitiven Wissensbegriff, wird allerdings in der KI wegen seiner engen Beziehungen zu Domänenwissen auch als 'Wissen' bezeichnet.

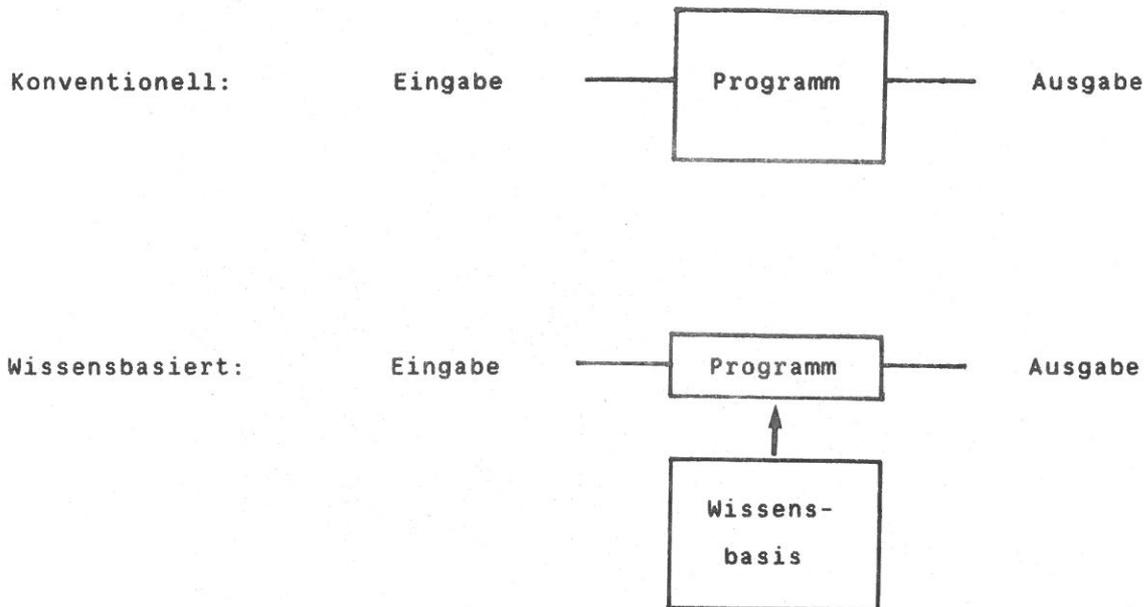
Ein dritter, häufig übersehener Wissensbereich bezieht sich auf den menschlichen Benutzer des technischen Systems. Zu einer Problemlösung gehört zweifellos auch Wissen über die Erwartungen und Möglichkeiten des Benutzers, z.B. ob er ein Laie oder Experte ist. Dieses Wissen kann auch sukzessive während eines Dialogs aufgebaut werden. Die rechnerinterne Repräsentation dieses Wissens wird 'Benutzermodell' genannt.

Die Tatsache, daß vielfältiges Wissen zur Problemlösung herangezogen wird, macht ein System noch nicht wissensbasiert im Sinne der KI-Terminologie. Die folgenden Forderungen müssen erfüllt sein:

- Das für die Lösung einer Aufgabe erforderliche Wissen ist explizit und strukturiert repräsentiert.
- Das Verhalten des Systems wird durch dieses Wissen gesteuert.

Der erste Punkt zeigt den entscheidenden Unterschied zu konventionellen Systemen auf. Wissen darf nicht implizit in einem komplexen Programm versteckt sein, sondern muß in wohlstrukturierter Form als durchschaubarer Datenbestand eingebracht werden. Dadurch ergeben sich die folgenden

unterschiedlichen Systemarchitekturen:



Der Programmteil des wissensbasierten Systems ist mit Bedacht kleiner gezeichnet. Dadurch, daß alles domänenabhängige Wissen in die Wissensbasis verlagert ist, besteht das eigentliche Programm im wesentlichen aus einem allgemeinen Interpreter, der im Idealfall für die verschiedensten Aufgabenbereiche identisch ist und deshalb keinen wesentlichen Programmieraufwand erfordert. Der softwaretechnische Vorteil der wissensbasierten Systemarchitektur liegt somit auf der Hand.

3. Semantische Netze

In diesem Abschnitt wird ein erster Repräsentationsformalismus vorgestellt, an dem sich wichtige Aspekte der Wissensrepräsentation erläutern lassen. In den folgenden Abschnitten werden dann Alternativen und Weiterentwicklungen aufgezeigt.

Semantische Netze wurden zunächst von Forschern aus dem Grenzgebiet zwischen KI und Kognitiver Psychologie als ein biologisch plausibler Repräsentationsformalismus vorgeschlagen (QUILLIAN 68), später jedoch innerhalb der KI stark erweitert und ausgebaut (HENDRIX 79). Ein semantisches Netz besteht aus einem Netz, das sich aus Knoten und Kanten

zusammensetzt, sowie einer Vorschrift, die Knoten und Kanten mit Objekten und Beziehungen der realen Welt in Verbindung bringt. Diese Vorschrift heißt 'Semantik', sie gibt der formalen Struktur des Netzes Bedeutung.

Abb. 1 zeigt ein einfaches semantisches Netz, dessen Knoten vier verschiedene Konzepte repräsentieren. Die Kanten drücken die Oberbegriffsbeziehung zwischen jeweils zwei Konzepten aus (AKO = 'a-kind-of').

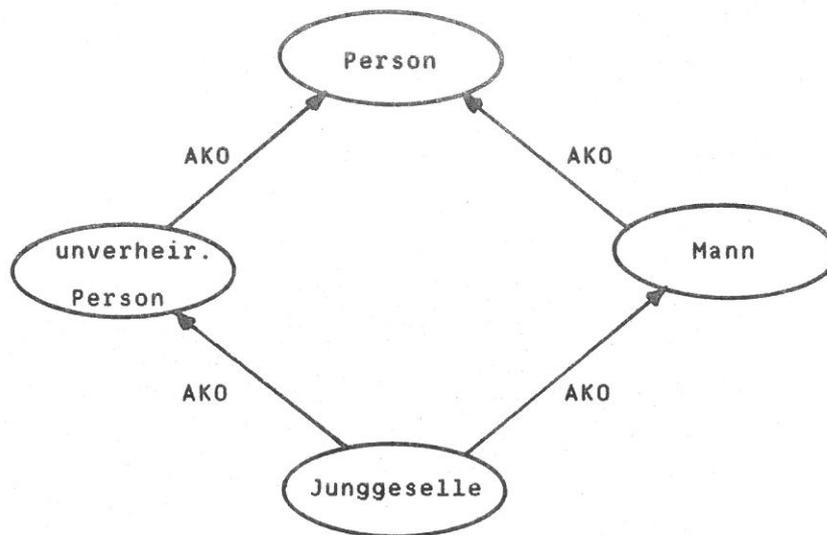


Abb. 1: Einfaches semantisches Netz

Diese Beziehung ist für die Strukturierung von Wissen außerordentlich wichtig, denn sie erlaubt es, Taxonomien aufzubauen.

Abb. 2 zeigt einen Ausschnitt aus einer Taxonomie für physikalische Objekte (aus SCHUBERT et al. 83). Hier sind aus Gründen der Übersichtlichkeit mehrfache Kanten zusammengezogen worden. Die Knoten sind zusätzlich mit einem Indexpaar zur Unterstützung von Evaluierungsprogrammen versehen.

Hierarchische Strukturen können auch durch andere Beziehungen entstehen. Abb. 3 (aus SCHUBERT et al. 83) zeigt eine mehrfache TEIL-VON-Hierarchie. Die Knoten entsprechen verschiedenen Körperteilen oder Gruppen von Körperteilen. Die Kanten markieren die TEIL-VON-Beziehung (P = part-of).

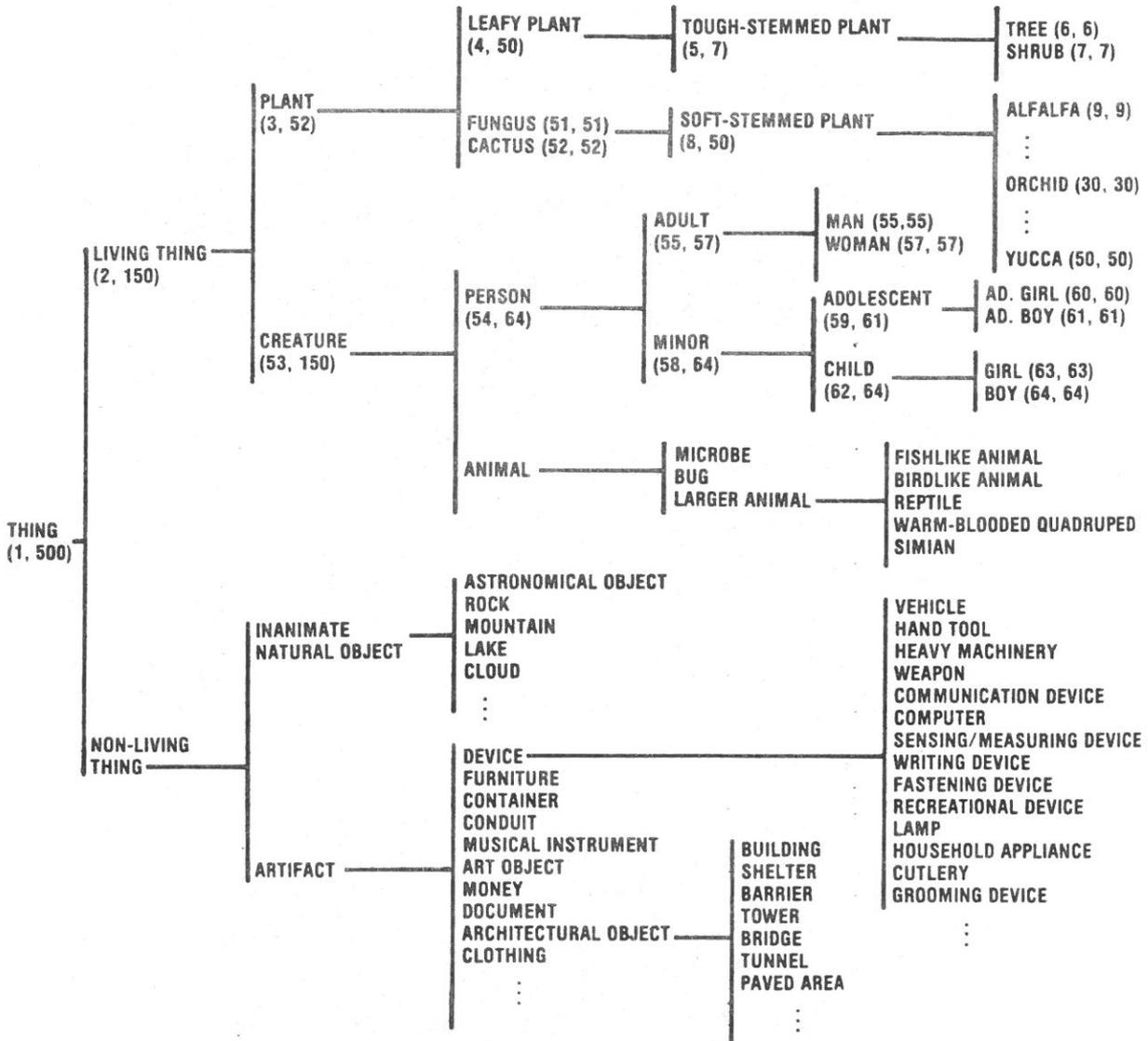


Abb. 2: Taxonomie von physikalischen Objekten

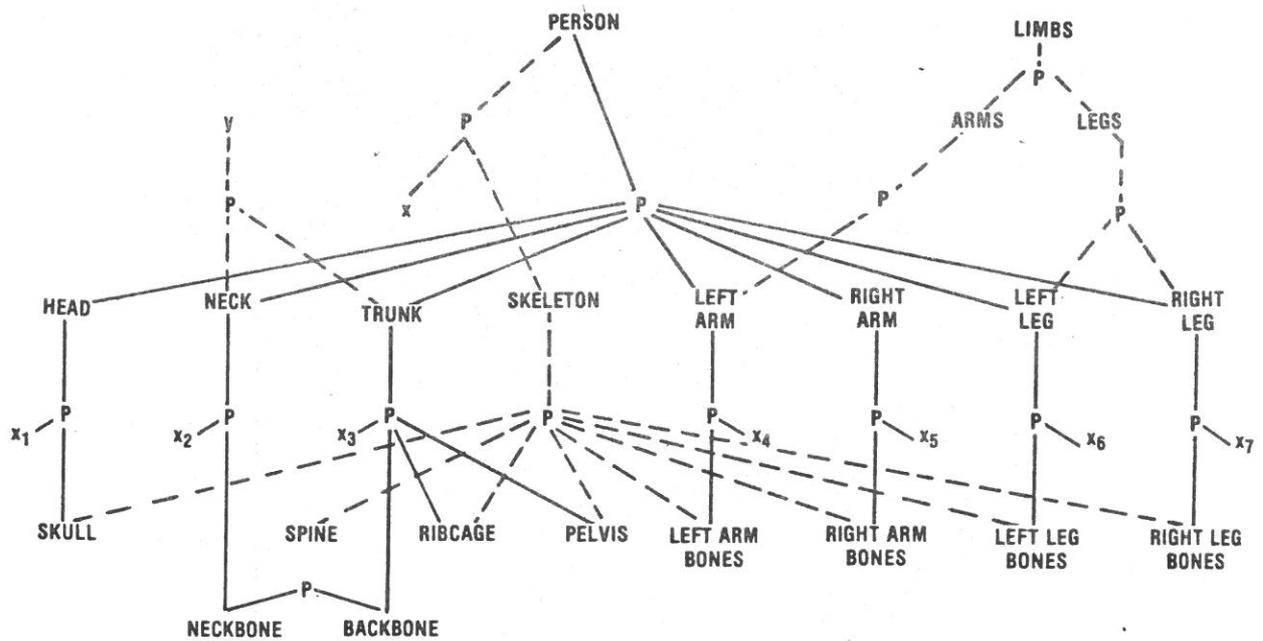
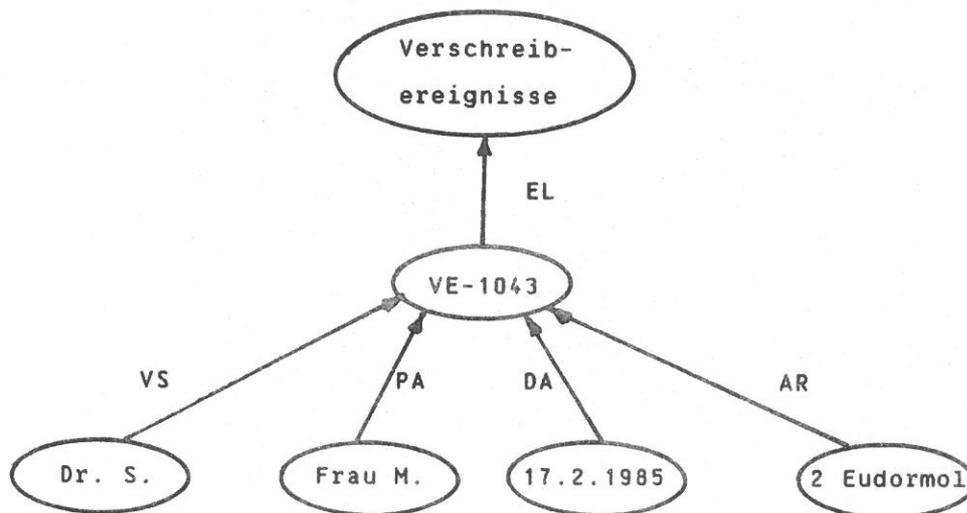


Abb. 3: Mehrfache TEIL-VON-Hierarchie.

In einem Semantischen Netz können jeweils nur zwei Knoten mit einer Kante verbunden werden. Dadurch werden formal zweistellige Beziehungen repräsentiert. Mehrstellige Beziehungen können jedoch grundsätzlich in mehrere zweistellige Beziehungen transformiert werden. Dies soll an einem Beispiel gezeigt werden. Zu repräsentieren sei:

"Dr. Schmidt hat Frau Müller am 17.2.1985 zwei Tabletten Eudormol verschrieben"

Offenbar geht es hier um eine "Verschreibbeziehung" zwischen den vier Komponenten "Dr. S.", "Frau M.", "17.2.1985" und "zwei Eudormol". Anstelle dieser vierstelligen Relation wird der Sachverhalt mit fünf zweistelligen Relationen repräsentiert:



Hier bedeuten die Kantenmarkierungen: EL = ELEMENT-VON, VS = VERSCHREIBER, PA = PATIENT, DA = DATUM und AR = ARZNEI.

Semantische Netze sind im Rechner leicht symbolisch zu repräsentieren und auch wegen ihrer graphischen Anschaulichkeit sehr attraktiv. Diese geht jedoch schnell verloren, wenn der Wissensbestand umfangreicher wird. Abb. 4 zeigt ein solches Beispiel, an dem auch noch andere Probleme sichtbar werden. Zum einen ist festzustellen, daß die inhaltlich zusammengehörigen Datenobjekte weder formal noch in der graphischen Wiedergabe zusammengeklammert werden. Dies ist der Hauptgrund für die Unübersichtlichkeit. Zum zweiten erweist sich die einfache Zuordnung von Knoten zu den Elementen einer Beziehung als simplizistisch. In Abb. 4 haben beispielsweise die Knoten "Dr. S." und "2 Eudormal" formal denselben Status. Aber offenbar bezeichnet der erste ein konkretes Objekt, während der zweite wegen seiner Beziehungen zu mehreren Verschreibereignissen nicht zwei konkrete Tabletten sondern eine abstrakte Arznei wiedergibt. Die Knoten haben also einen unterschiedlichen existenziellen Status. Zudem liegt es nahe, den Knoten "2 Eudormal" weiter zu strukturieren, etwa in Dosierung "2" und Arzneytyp "Eudormal".

Diese Beispiele berühren nur einige der Feinheiten, die bei einer sorgfältigen und schlüssigen Wissensrepräsentation zu beachten sind. Sie sollen an dieser Stelle lediglich verständlich machen, daß semantische Netze - naiv angewandt - nicht in jedem Fall ein geeignetes Repräsentationswerkzeug sind.

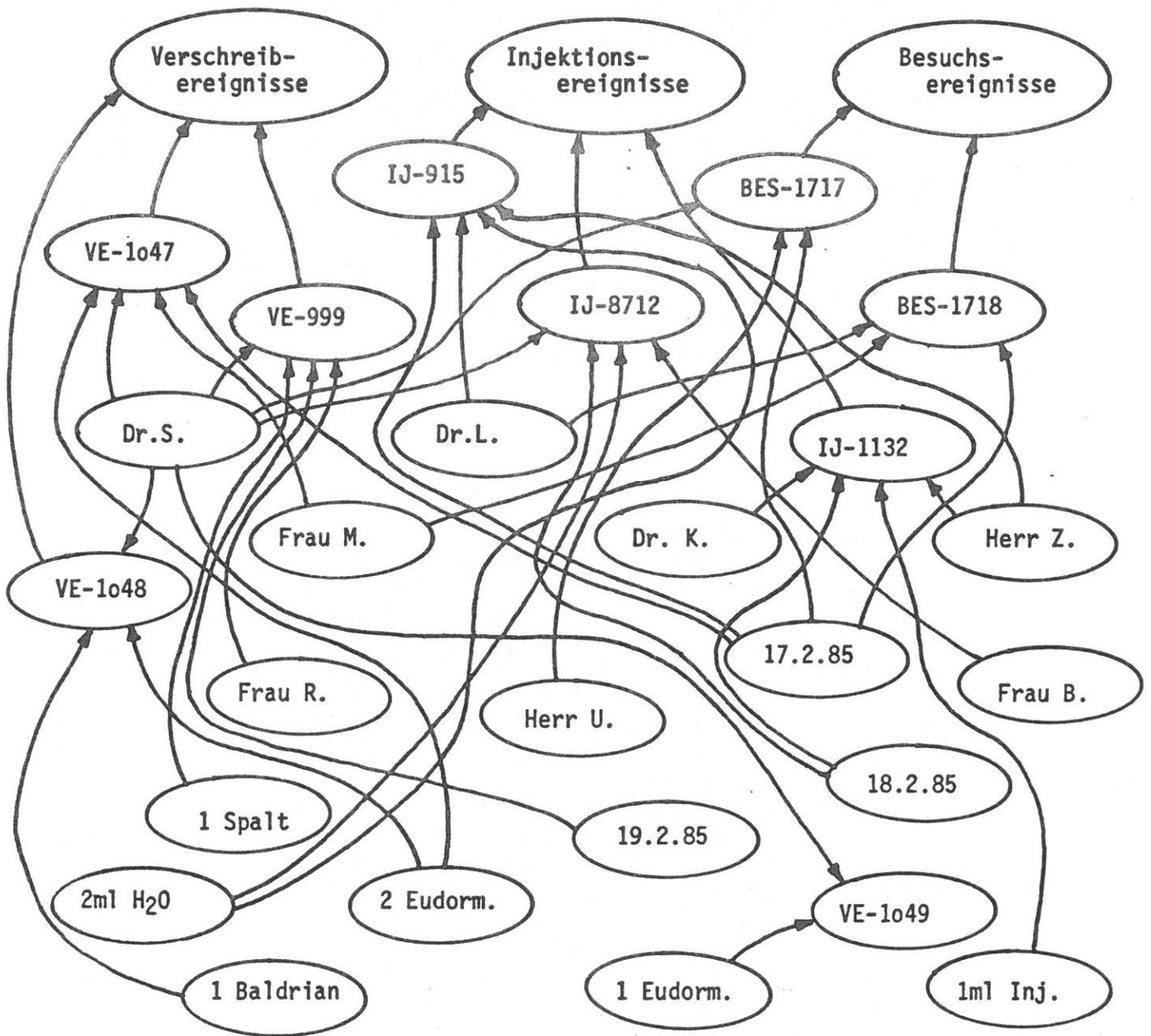


Abb. 4: Probleme mit semantischen Netzen

4. Schemata

Eine wesentlich besser strukturierte Wissensrepräsentation ergibt sich, wenn die Objekte nicht als atomare Knoten sondern als in sich gegliederte

Einheiten repräsentiert werden. Eine solche Einheit stellt dann gewissermaßen den Rahmen für objektinterne Eigenschaften und Beziehungen dar, deshalb die Bezeichnung 'frame' im englischen Sprachtum, hier mit 'Schema' übersetzt. Abb. 5 zeigt zwei einfache Schemata, an denen der grundsätzliche Aufbau zu erkennen ist.

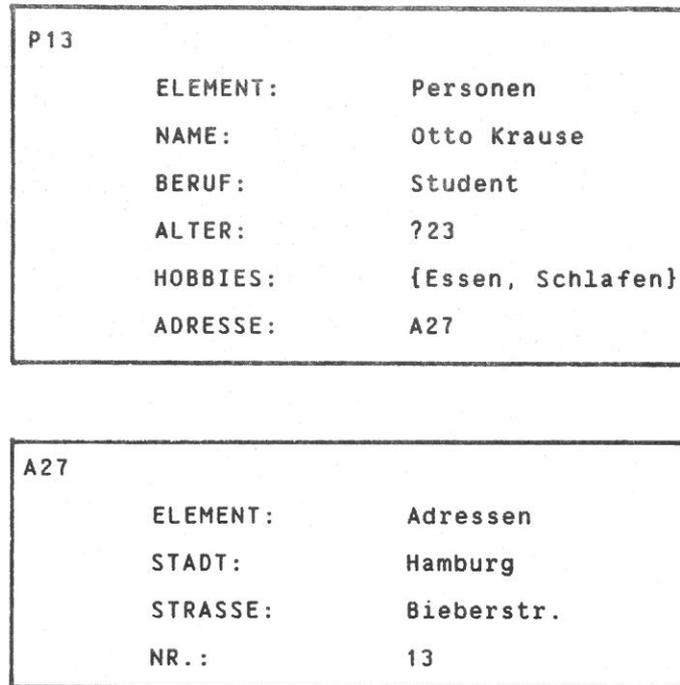


Abb. 5: Zwei einfache Schemata (frames)

Ein Schema besteht aus einer beliebigen Zahl von Feldern (slots), die sich jeweils aus einem Attributbezeichner (z.B. BERUF) und dem dazugehörigen Datum zusammensetzen. Das Datum kann dabei auf andere Schemata verweisen (z.B. A27 bei der Adreßangabe). Schemata ermöglichen in erster Linie eine objektzentrierte Bündelung von Wissen. Sie erlauben es aber auch, weiterführende Konzepte zu realisieren. Beispielsweise kann dafür gesorgt werden, daß bei fehlender Altersangabe eine vernünftige Vorbesetzung (default) verwendet wird, die dann aber auch als solche gekennzeichnet werden muß (hier: ?23).

In der Wissensrepräsentationssprache FRL (frame representation language, ROBERTS und GOLDSTEIN 77) sind einige interessante weiterführende Konzepte realisiert. Ein Schema in FRL hat die folgende formale Struktur:

| | | | | |
|--------|------------|-------------|-------------|------------|
| (frame | (slot1 | (facet1 | (datum1 | (message1) |
| | (slot2...) | (facet2...) | (datum2...) | (message2) |
| | . | . | . | . |
| | . | . | . | . |
| | . | . | . | . |
| | (slotN...) | (facetN...) | (datumN...) | (messageN) |

Jedes Feld besteht aus einem Attributbezeichner und einem mehrfach untergliederten Datenfeld. Die 'Facetten' eines Datenfeldes erlauben es, zwischen Angaben mit unterschiedlicher Modalität zu differenzieren. Das folgende Schema exemplifiziert einige dieser Möglichkeiten.

```
(S1044 (ISA ($VALUE (STUDENT)))
      (NAME ($VALUE (KLAUS MEIER)))
      (FACH ($VALUE (MEDIZIN)
                  (INFORMATIK (SEIT 1985))))
      (NATIONALITÄT ($DEFAULT (DEUTSCH)))
      (HOBBIES ($VALUE (MUSIK)
                      ($IF-ADDED (COND ((EQ :VALUE SF)
                                         (NOTIFY :FRAME SF-KLUB))))))
      (IQ ($VALUE (145 (NACH EIGENEN ANGABEN))
                ($REQUIRE (GREATERP :VALUE 100))))
```

Die Facettenbezeichner \$VALUE und \$DEFAULT kennzeichnen ein konventionelles Datum bzw. seine Vorbesetzung. Ein \$IF-ADDED hat den Effekt, daß die angeschlossene Prozedur bei Eintragung eines Datums abläuft. Im Beispiel wird der Science-Fiction-Klub benachrichtigt, sobald das entsprechende Hobby eingetragen wird. Einen ähnlichen Effekt hat \$REQUIRE, das ein Datum geeigneten Tests unterwirft, bevor es eingetragen wird.

Erwähnenswert ist auch noch ein Vererbungsmechanismus, der fehlende Daten bei Bedarf automatisch aus übergeordneten Schemata (angegeben im ISA-Feld) beschafft. Auf diese Weise kann z.B. die individuelle Eintragung eines zusätzlichen Attributes GEHALT unterbleiben, da das Gehalt für alle Studenten gemeinsam im übergeordneten Schema STUDENT mit Null verzeichnet wäre.

5. Prädikatenkalkül

Logik in der Notation des Prädikatenkalküls eignet sich vorzüglich zur Repräsentation von Wissen, auch für die Fälle, wo der Gegenstand auf den ersten Blick nicht viel mit Logik zu tun hat. Logische Repräsentationen haben die folgenden hauptsächlichen Vorteile:

- Logik erlaubt Deduktionen
- Logik hat eine präzise Semantik
- Logik wird durch eine Programmiersprache (PROLOG) unterstützt

Vielleicht überrascht es, daß Deduktionen nicht nur beim Problemlösen, sondern gerade auch bei der Wissensrepräsentation eine Rolle spielen. Der Grund liegt darin, daß Wissen nur in einfachen Fällen so repräsentiert werden kann, daß alle relevanten Fakten unmittelbar abgerufen werden können. Effektivitätsüberlegungen erfordern es im allgemeinen, daß ein (begrenzter) Bestand von nackten Fakten durch Deduktionsregeln angereichert wird, die dann die Beantwortung einer Fülle von Anfragen erlauben. Dies wird im folgenden anhand eines ausführlichen Beispiels illustriert.

Es sei der Bestand einer Bibliothek zu repräsentieren, die u.a. folgende Bücher enthält:

- 1 Buch über Fische
- 1 Buch über Säugetiere (ausgeliehen)
- 1 Buch über Tiere

Wie kann man die Frage

"Kann ich ein Buch über Affen und Löwen einsehen?"

automatisch beantworten? Zunächst müssen einige Abkürzungen vereinbart werden.

- BF das Fischbuch
- BS das Säugetierbuch
- BT das Tierbuch
- A Affen

L Löwen
 F Fische
 S Säugetiere
 T Tiere
 A(x) x ist ausgeliehen
 E(x) man kann x einsehen
 Ü(x,y) Buch x ist über y
 O(x,y) x ist Oberbegriff von y

Der Zustand der Bibliothek kann damit folgendermaßen repräsentiert werden (~ ist die Negation):

$\sim A(BF)$ $\dot{U}(BF, F)$
 $A(BS)$ $\dot{U}(BS, S)$
 $\sim A(BT)$ $\dot{U}(BT, T)$

Die Anfrage lautet:

$(\exists x) (E(x) \ \& \ \dot{U}(x, A) \ \& \ \dot{U}(x, L))$

("Gibt es ein x derart, daß x einsehbar und über Affen und über Löwen ist?")

Um die Anfrage beantworten zu können, ist noch beträchtliches zusätzliches Wissen erforderlich.

$O(T, F)$ $O(S, A)$
 $O(T, S)$ $O(S, L)$

("Tiere ist Oberbegriff von Fische", "Tiere ist Oberbegriff von Säugetiere", etc.)

$(Ax) (Ay) (Az) (O(x, y) \ \& \ O(y, x) \Rightarrow O(x, z))$

("Für alle x, y und z gilt: Wenn x Oberbegriff von y und y Oberbegriff von z ist, so ist x Oberbegriff von z")

$(Ax) (A(x) \Rightarrow \sim E(x))$

("Für alle x gilt: Wenn x ausgeliehen ist, so ist x nicht einsehbar".)

$(Ax) (Ay) (Az) (\dot{U}(x, y) \ \& \ O(y, z) \Rightarrow \dot{U}(x, z))$

("Für alle x, y und z gilt: Wenn x über y ist und y Oberbegriff von z ist,

dann ist x über z".)

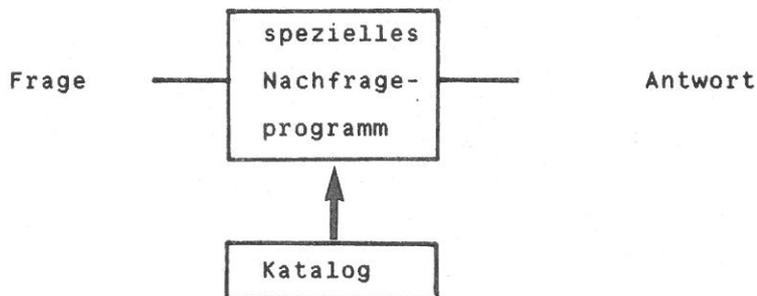
Auf diese Formeln kann jetzt ein allgemeingültiges Verfahren angewendet werden, das tatsächlich das gewünschte Ergebnis liefert:

"Ja, Sie können ein Buch über Affen und Löwen einsehen".

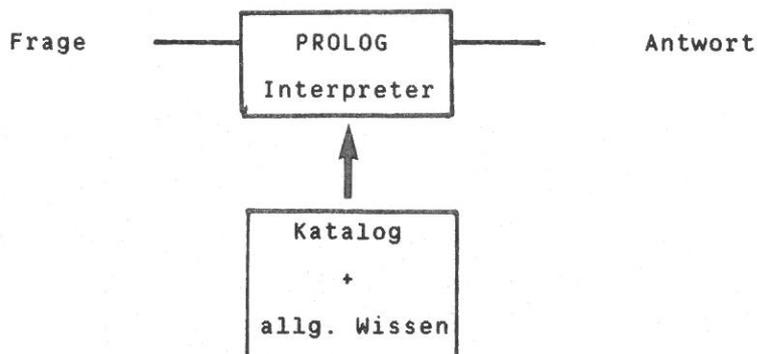
Auf Einzelheiten des Verfahrens kann hier nicht näher eingegangen werden. Wichtig ist jedoch die Feststellung, daß es sich dabei lediglich um eine syntaktische Manipulation der Formeln handelt, die für alle im Prädikatenkalkül notierten Probleme nach denselben Regeln abläuft. In der Tat ist das Verfahren Bestandteil der Programmiersprache PROLOG, muß also im Einzelfall gar nicht programmiert werden.

Hier wird der Unterschied zu einer konventionellen Lösung besonders deutlich: Die Formulierung einer Wissensbasis ist an die Stelle von problembezogener Programmierung getreten.

Konventionell:



Wissensbasiert/logisch:



Wissensrepräsentation durch logische Formeln kann jedoch auch Nachteile haben und auf Grenzen stoßen. Die am häufigsten geäußerten Kritikpunkte sind:

- Logische Formeln sind manchmal unintuitiv
- Unsicheres oder unscharfes Wissen kann nicht ohne weiteres repräsentiert und verarbeitet werden
- Die Repräsentation von Metawissen ist nicht ohne weiteres möglich
- Der logische Formalismus ist zwar sehr mächtig, kann aber wegen seiner Allgemeinheit ineffektiv sein.

Wissensrepräsentation und Programmierung in Logik stellen aktuelle Forschungsgebiete dar.

6. Produktionen

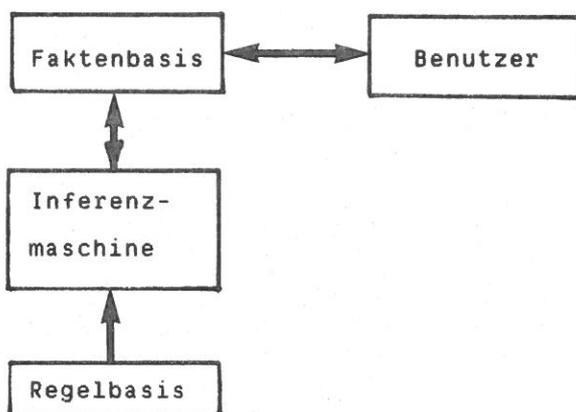
Bei bestimmten KI-Anwendungen, insbesondere bei der Entwicklung von Expertensystemen, kommt es darauf an, Wissen in einer Form zu repräsentieren, die der Denkweise von Menschen möglichst nahe kommt. Dafür eignen sich Produktionen, denn sie erlauben es, Wissen in Form von einfachen WENN-DANN-Regeln wiederzugeben. Eine Produktion hat den folgenden formalen Aufbau:

<Bedingung> ---> <Aktionen>

Der Bedingungsteil ist meistens ein logisches Prädikat, mit dem eine Faktenbasis geprüft wird. Der Aktionsteil kann z.B. Folgerungen produzieren, die der Faktenbasis zugefügt werden, kann aber auch andere geeignete Aktionen vorschreiben, z.B. eine Frage an den Benutzer. Die in Abschnitt 2 wiedergegebene MYCIN-Regel ist ein typisches Beispiel.

Produktionen haben das charakteristische Format von Regelwissen, sind aber für andere Repräsentationsaufgaben, etwa die Beschreibung komplexer Objekte, weniger geeignet. Produktionensysteme stellen eine besondere Form von wissensbasierten Systemen dar, bei denen Probleme durch sukzessive

Anwendung von Produktionen auf eine Faktenbasis gelöst werden (eine umfassende Darstellung findet sich z.B. in WATERMAN und HAYES-ROTH 78). Die Systemkomponente, mit der Produktionen zunächst auf Anwendbarkeit geprüft und dann ggf. angewendet werden, wird häufig 'Inferenzmaschine' genannt. Sie ist meist aufgabenunabhängig konzipiert und muß deshalb - ähnlich wie der PROLOG-Interpreter - nur einmal programmiert werden. Ein Produktionensystem hat damit den folgenden schematischen Aufbau:



7. Zusammenfassung

Wissensrepräsentation hat in wissensbasierten Systemen und generell in der Künstlichen Intelligenz eine zentrale Bedeutung. Während konventionelle Programme das für eine Aufgabe erforderliche Wissen meist implizit enthalten, geht die Entwicklung von wissensbasierten Systemen dahin, daß das problemspezifische Wissen explizit als ein transparenter Datenbestand repräsentiert wird und das verarbeitende Programm problemunabhängig wird. Dadurch können Aufgaben von großer Komplexität, insbesondere auch solche, die bei Menschen gemeinhin Intelligenz erfordern, in systematischer und durchschaubarer Weise von Rechnern gelöst werden. Um diese Aussage zu verdeutlichen, wurden verschiedene Techniken der Wissensrepräsentation anhand von Beispielen vorgestellt. Dabei zeigten sich die folgenden wesentlichen Merkmale:

- Semantische Netze repräsentieren Objekte und ihre Beziehungen
- Schemata sind objektorientiert und bieten zusätzliche Möglichkeiten zur Wissensrepräsentation
- Logik hat eine präzise Semantik und bietet umfassende Deduktionsmechanismen
- Produktionen entsprechen menschlichem Regelwissen, bieten aber wenig Strukturierungshilfen.

8. Literaturhinweise

Buchanan and Shortliffe 84

Rule-Based Expert Systems

B.G. Buchanan, E.H. Shortliffe

Addison-Wesley, 1984

Charniak and McDermott 85

Introduction to Artificial Intelligence

E. Charniak, D. McDermott

McGraw-Hill, 1985

Ernst and Newell 69

GPS: A Case Study in Generality and Problem Solving

G.W. Ernst, A. Newell

Academic Press, 1969

Hendrix 79

Encoding Knowledge in Partitioned Networks

G.G. Hendrix

in: N.V. Findler (ed.), Associative Networks,

Academic Press, 1979, 51-92

Quillian 68

Semantic Memory

M.R. Quillian

in: M.L. Minsky (ed.), Semantic Information Processing,

The MIT Press, 1968, 216-270

Roberts and Goldstein 77

The FRL Primer

R.B. Roberts, I.P. Goldstein

Report AIM-408, MIT AI-Lab, Cambridge, MA, 1977

Schubert et al. 83

Determining Type, Part, Color, and Time Relationships

L.K. Schubert, M.A. Papalaskaris, J. Taugher

IEEE-Computer, October 1983, 53-60

Waterman and Hayes-Roth 78

Pattern-Directed Inference Systems

D.A. Waterman, F. Hayes-Roth

Academic Press 78