

# Einführung in die KI

Prof. Dr. sc. Hans-Dieter Burkhard  
Vorlesung Winter-Semester 2005/06

Wissensrepräsentation:  
Grundbegriffe  
Allgemeine Probleme

## 1. Grundbegriffe: Wissen

1. allg. verfügbare Orientierungen im Rahmen alltäg. Handlungs- und Sachzusammenhänge (Alltagswissen);
2. im engeren, philosoph. und wiss. Sinne im Unterschied zu Meinung und Glauben die auf Begründungen bezogene und strengen Überprüfungspostulaten unterliegende Kenntnis, institutionalisiert in den Wissenschaften.  
(Meyers großes Taschenlexikon, 4.Auflage, 1992)

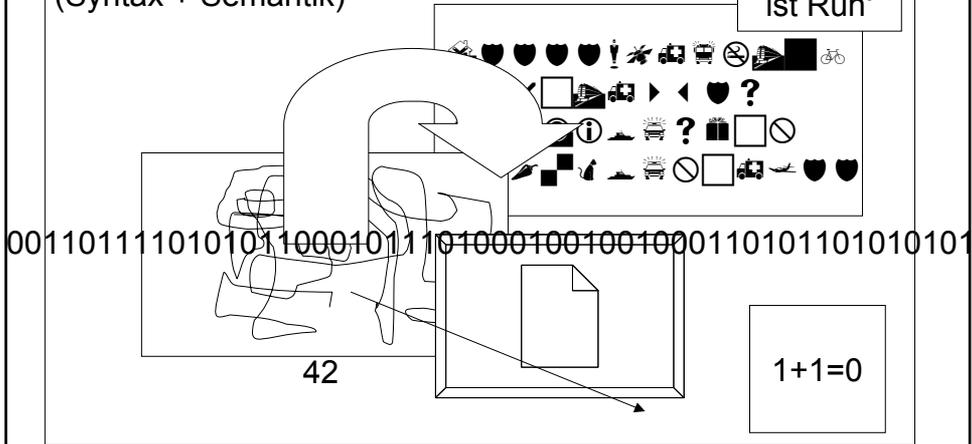
Wissen, Kenntnisse, Fähigkeiten

“Knowledge” vs. “Belief” (=Annahmen)

# Zeichen/Symbole + Interpretation

Formen und Bedeutungen?  
(Syntax + Semantik)

Über allen  
Wipfeln  
ist Ruh'



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/06

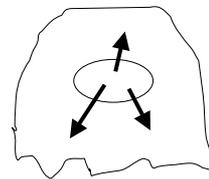
Vorlesung Einführung in die KI  
Wissensrepräsentation-Grundlagen

3

# Explizites vs. Implizites Wissen

explizit: z.B. Fakten, Regeln

implizit: ableitbares Wissen



Inferenz: Verfahren zur Herleitung von implizitem  
Wissen aus explizitem Wissen  
(z.B. Regelinterpreter)

Andere Bedeutung:

Implizit = nicht verbalisierbar („tacit knowledge“)

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/06

Vorlesung Einführung in die KI  
Wissensrepräsentation-Grundlagen

4

# Repräsentation von „Wissen“

Lexikon

Datenbank

Briefmarkensammlung

Programme

Landkarten

Bilder

Gesetze

Regeln

...



## UNTERSCHIEDE:

Darstellung und Interpretation

- Menschlicher Nutzer
- Maschinelle Auswertung

# Interpretation

Lexika

Datenbanken

Faktensammlungen

Programme

Landkarten

Bilder

Gesetze

Regeln

...

(Systematische) Zuordnung  
Zeichen/Struktur → Bedeutung  
Syntax → Semantik  
Vollständigkeit, Korrektheit, Konsistenz

## Semantik

Woher stammt Bedeutung?

- Konventionen
- Gemeinsames Referenzsystem  
(symbol grounding)

Inferenzprozess  
muss im Einklang mit Bedeutungen stehen.

# Inferenz

## Maschinelle Methoden

Lexika  
Datenbanken  
Faktensammlungen  
Programme  
Landkarten  
Bilder  
Gesetze  
Regeln  
...

Programmierte Abläufe  
- Suchverfahren  
- Constraints  
- Klassifikationsverfahren  
- Bildverarbeitung  
- Sprachverarbeitung  
-usw.

Zusätzliche „Meta“-Annahmen,  
z.B. über Umgang mit nicht-bekanntem Daten:  
• „Closed World Assumption“ (CWA)  
• Umgang mit Negation  
• Default-Annahmen usw.

# Negation

Was ist die Bedeutung der Negation von:

„Die Kinder schlafen ruhig im Nachbarzimmer.“

# Fallbeispiel PROLOG

Negation definieren über CWA.

Idee: Was nicht beweisbar ist, gilt nicht.

Wie weit ist das realisiert?

Was bedeutet das praktisch?

männlich(X) :- not (weiblich(X)).

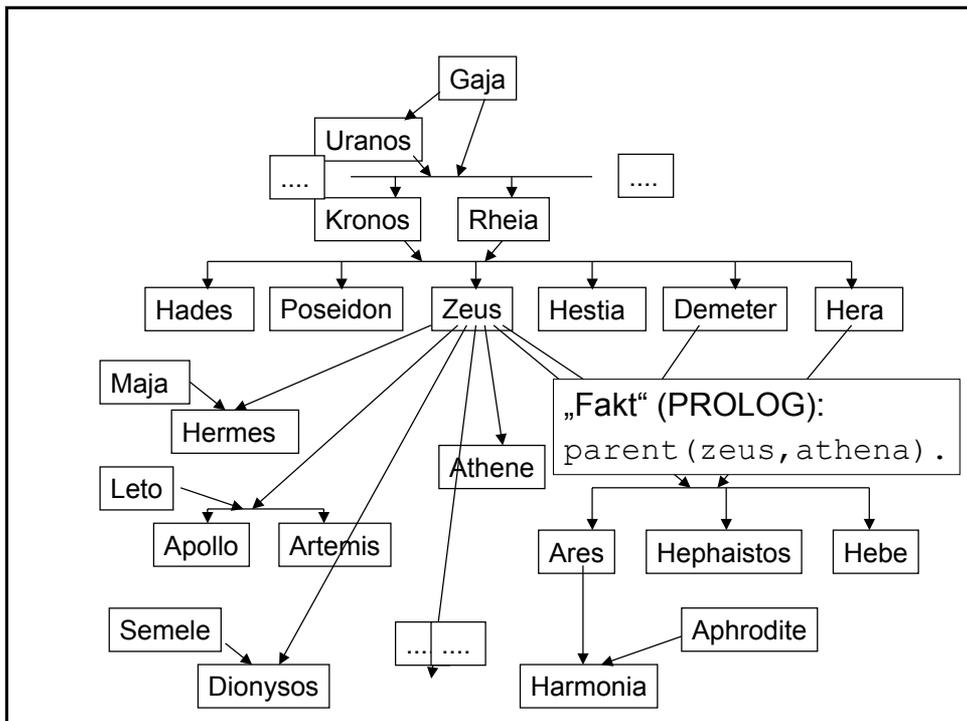
weiblich(frieda).

? – männlich(frieda).

? – männlich(fritz).

? – männlich(anna).

? – männlich(X).



## Definition einer Relation durch Aufzählung

Aufzählung: Die Beziehungen werden **explizit** aufgezählt  
(**Faktenmenge, Datenbank**)

parent(uranus, cronus).	female(gaea).
parent(gaea, cronus).	female(rhea).
parent(gaea, rhea).	female(hera).
parent(rhea, z male(uranus	female(hestia).
parent(cronus, z male(cronus	female(demeter).
parent(rhea, h male(zeus).	female(athena).
parent(cronus, h male(hades)	female(metis).
parent(cronus, h male(hermes	female(maia).
parent(rhea, h male(apollo	female(persephone).
parent(cronus, h male(dionys	female(aphrodite).
parent(rhea, h male(hephae	female(artemis).
...	female(lete).
male(poseidon).	

11

## Fakten in Prolog

Ein Fakt hat die Form

`funktor(argumente).`

funktor ist der Name einer n-stelligen Relation (Prädikat).  
n ist die Anzahl der Argumente.

```
male(zeus).
parent(zeus,athena).
parent(zeus,hera,ares).
plus(3,4,7).
kleiner(6,9).
```

```
male/1
parent/2
parent/3
plus/3
true/0
fail/0
```

# Anfragen in PROLOG

Eine Anfrage hat die Form `?-funktork(argumente).`

funktork ist der Name einer n-stelligen Relation (Prädikat).

```
parent(uranus, cronus).
?- male(zeus).
yes.
?- parent(zeus, athena).
yes.
?-parent(hera, athena).
no.
parent(ineas, hades).
parent(cronus, hestia).
parent(rhea, hestia).
parent(zeus, hermes).
parent(maia, hermes).
...
female(gaea).
female(rhea).
female(hera).
female(hestia).
female(demeter).
female(athena).
male(uranus).
male(cronus).
male(zeus).
male(hades).
male(hermes).
male(apollo).
male(dionysius).
male(hephaestus).
...
)
```

H.D.Burkhard  
Winter-Semester

13

# CWA: Closed World Assumption (1a)

Bedeutung der Antwort „no“ in PROLOG als Datenbank:  
Der Fakt ist in der Datenbank nicht enthalten.

```
parent(uranus, cronus).
parent(gaea, cronus).
female(gaea).
female(rhea).
female(hera).
female(hestia).
female(demeter).
female(athena).
male(uranus).
male(cronus).
male(zeus).
male(hades).
male(hermes).
male(apollo).
male(dionysius).
male(hephaestus).
...
?-parent(hera, athena).
no.
?-parent(zeus, hera, ares).
no.
?-kleiner(3, 7).
no.
parent(ineas, hestia).
parent(zeus, hermes).
parent(maia, hermes).
...
male(poseidon)
```

H.D.Burkhard  
Winter-Semester

14

# Anfragen nach Existenz

Anfrage enthält Variable      ?-funkt( argumente ) .

Variablen-Bezeichner beginnen mit Großbuchstaben

Antworten können alle passenden Fakten liefern.

```
?- parent(zeus, X) .  
X=hermes?  
;  
X=athena?  
;  
X=ares?  
.  
yes.
```

```
parent(gaea, cronus) .  
parent(gaea, rhea) .  
parent(rhea, zeus) .  
parent(cronus, zeus) .  
parent(rhea, hera) .  
parent(cronus, hera) .  
parent(cronus, hades) .  
parent(rhea, hades) .  
parent(cronus, hestia) .  
parent(rhea, hestia) .  
parent(zeus, hermes) .  
parent(maia, hermes) .  
...  
female(hera) .  
female(hestia) .  
female(demeter) .  
female(athena) .  
female(metis) .  
female(maia) .  
female(persephone) .  
female(aphrodite) .  
male(uranus) .  
male(cronus) .  
male(zeus) .  
male(hades) .  
male(hermes) .  
male(apollo) .  
male(dionysius) .  
male(hephaestus) .  
male(poseidon) .
```

# CWA: Closed World Assumption (1b)

Bedeutung der Antwort „no“ bei PROLOG als Datenbank:  
Es gibt keinen Fakt in der Datenbank, der mit der Anfrage  
unifizierbar ist („matcht“).

```
?-father(X, Y) .  
no.  
?-kleiner(3, N) .  
no.
```

```
parent(uranus, cronus) .  
parent(gaea, cronus) .  
parent(gaea, rhea) .  
parent(rhea, zeus) .  
parent(cronus, zeus) .  
parent(rhea, hera) .  
parent(cronus, hera) .  
parent(cronus, hades) .  
parent(rhea, hades) .  
parent(cronus, hestia) .  
parent(rhea, hestia) .  
parent(zeus, hermes) .  
parent(maia, hermes) .  
...  
female(gaea) .  
female(rhea) .  
female(hera) .  
female(hestia) .  
female(demeter) .  
female(athena) .  
female(metis) .  
female(maia) .  
female(persephone) .  
female(aphrodite) .  
male(uranus) .  
male(cronus) .  
male(zeus) .  
male(hades) .  
male(hermes) .  
male(apollo) .  
male(dionysius) .  
male(hephaestus) .  
male(poseidon) .
```

## Relationen definieren durch Regeln

Eine Regel hat die Form

```
kopf :- körper .
```

oder

```
goal :- subgoals .
```

Mit der intuitiven Bedeutung:

„goal“ gilt (ist beweisbar),

falls alle „subgoals“ gelten (beweisbar sind).

Definition der Relation „Vater-Kind“:

```
father (Vater, Kind)
```

```
:- parent (Vater, Kind) , male (Vater) .
```

## Relationen definieren durch Regeln

Die Variablen in Regeln sind universell quantifiziert.

Die Relationen der rechten Seite sind konjunktiv verknüpft.

```
father (X, Y) :- parent (X, Y) , male (X) .
```

```
mother (X, Y) :- parent (X, Y) , female (X) .
```

```
parent (X, Y, Z) :- father (X, Z) , mother (Y, Z) .
```

```
son (X, Y) :- parent (X, Y) , male (Y) .
```

```
grandfather (X, Z) :- father (X, Y) , parent (Y, Z) .
```

```
grandmother (X, Z) :- mother (X, Y) , parent (Y, Z) .
```

```
grandchild (X, Y) :- grandfather (Y, X) .
```

```
grandchild (X, Y) :- grandmother (Y, Z) .
```

## Regel als logische Formel

$goal(X_1, \dots, X_n) :- subgoal_1(X_1, \dots, X_n), \dots, subgoal_m(X_1, \dots, X_n).$

wird aufgefasst als

$\forall X_1 \dots \forall X_n$

$[ subgoal_1(X_1, \dots, X_n) \wedge \dots \wedge subgoal_m(X_1, \dots, X_n) \rightarrow goal(X_1, \dots, X_n) ]$

oder

$\forall X_1 \dots \forall X_n$

$[ \neg subgoal_1(X_1, \dots, X_n) \vee \dots \vee \neg subgoal_m(X_1, \dots, X_n) \vee goal(X_1, \dots, X_n) ]$

(Horn-Klausel)

## Regel als logische Formel

Variable, die nur im Regelkörper auftreten, können auch als existentiell quantifizierte Variable **innerhalb des Regelkörpers** betrachtet werden

$\forall X_1 \dots \forall X_n \forall Y_1 \dots \forall Y_k [$

$subgoal_1(X_1, \dots, X_n, Y_1, \dots, Y_k) \wedge \dots \wedge subgoal_m(X_1, \dots, X_n, Y_1, \dots, Y_k) \rightarrow goal(X_1, \dots, X_n) ]$

ist logisch äquivalent zu

$\forall X_1 \dots \forall X_n [$

$\exists Y_1 \dots \exists Y_k [ subgoal_1(X_1, \dots, X_n, Y_1, \dots, Y_k) \wedge \dots \wedge subgoal_m(X_1, \dots, X_n, Y_1, \dots, Y_k) ] \rightarrow goal(X_1, \dots, X_n) ]$

$grandfather(X, Z) :- father(X, Y), father(Y, Z).$



## Relationen/Prozeduren

Klauseln mit gleichem Kopf-Funktor (Name, Stelligkeit) definieren eine ... **Prozedur**

Redundanzen sind für **PROLOG** problematisch:

- evtl. unerwünschte Auswirkungen
- Softwaretechnologie: Minimalitätsprinzip

```
grandfather(X, Z) :- father(X, Y), father(Y, Z).  
grandfather(X, Z) :- father(X, Y), mother(Y, Z).  
grandfather(X, Z) :- father(X, Y), parent(Y, Z).  
grandfather(cronus, athena).  
grandfather(cronus, athena).
```

## Anfragen an Prolog-Programme

Anfragen haben die Form

? - goal( $X_1, \dots, X_n$ ).

im Sinne von „gilt ...?“ („ist ... beweisbar?“):

$\exists X_1 \dots \exists X_k [ \text{goal}(X_1, \dots, X_n) ]$

oder allgemeiner

? - goal<sub>1</sub>( $X_1, \dots, X_n$ ), ..., goal<sub>m</sub>( $X_1, \dots, X_n$ ).

im Sinne von „gilt ...?“ („ist ... beweisbar?“):

$\exists X_1 \dots \exists X_k [ \text{goal}_1(X_1, \dots, X_n) \wedge \dots \wedge \text{goal}_m(X_1, \dots, X_n) ]$

## Interpreter für (Standard-)PROLOG

Systematisches Probieren aller Beweismöglichkeiten

Als Suchverfahren „Tiefe-Zuerst“ im Und-Oder-Baum mit:

Reihenfolge innerhalb einer Prozedur

(Alternativen für Beweis)

oben vor unten

Reihenfolge innerhalb einer Klausel

(alle subgoals müssen erfüllt werden)

links vor rechts

## Interpreter für Standard-Prolog

**Backtracking:**

Alternativen für den Beweis eines Teilziel werden markiert („Backtrack-Punkte“).

Beim Fehlschlagen eines Beweisversuchs wird am jüngsten Backtrack-Punkt ein alternativer Beweis gestartet („chronologisches Backtracking“). Dabei werden zwischenzeitliche Variablenbindungen zurückgenommen.

**Eingabe von „;“ bei Antworten auf existentielle Anfragen wirkt wie Fehlschlag (löst Backtracking aus) .**

## Deklarative vs. Prozedurale Semantik

PROLOG-Programm: Menge von Klauseln (Fakten, Regeln)  
Anfrage: ? - **funktor(argumente)**.

Deklarative Semantik:  
„Folgt **funktor(argumente)**  
logisch (PK1) aus dem Programm?“

Prozedurale Semantik:  
„Findet der PROLOG Interpreter  
einen Beweis für **funktor(argumente)**  
anhand des Programms?“

Auswirkungen ?  
Zusammenhang?

## Situation im PK1

Q folgt aus Formelmenge  $\{P_1, \dots, P_n\}$   
gdw. Q ist aus Formelmenge  $\{P_1, \dots, P_n\}$  syntaktisch ableitbar  
gdw.  $P_1 \wedge \dots \wedge P_n \rightarrow Q$  ist allgemeingültig

Allgemeingültigkeit ist axiomatisierbar/aufzählbar:

Falls ein Ausdruck H allgemeingültig ist,  
so ist das in endlich vielen Schritten feststellbar.

Genauer: Es gibt dafür ein universelles Verfahren.

## Situation im PK1

Q folgt aus Formelmenge  $\{P_1, \dots, P_n\}$

gdw. Q ist aus Formelmenge  $\{P_1, \dots, P_n\}$  syntaktisch ableitbar

gdw.  $P_1 \wedge \dots \wedge P_n \rightarrow Q$  ist allgemeingültig

Allgemeingültigkeit ist nicht entscheidbar:

Es gibt **kein universelles** Verfahren, das für beliebige H **entscheidet**, ob H allgemeingültig ist.

Falls ein Ausdruck H **nicht allgemeingültig** ist,  
so ist das eventuell nicht feststellbar  
(Verfahren kommt evtl. nicht zum Abbruch).

Genauer: Es gibt dafür **kein** universelles Verfahren.

## Situation im PK1

Q folgt aus Formelmenge  $\{P_1, \dots, P_n\}$

gdw. Q ist aus Formelmenge  $\{P_1, \dots, P_n\}$  syntaktisch ableitbar

gdw.  $P_1 \wedge \dots \wedge P_n \rightarrow Q$  ist allgemeingültig

Falls ein Ausdruck H **allgemeingültig** ist,  
so ist das in endlich vielen Schritten feststellbar.

Genauer: Es gibt dafür ein **universelles Verfahren**.

Falls ein Ausdruck H **nicht allgemeingültig** ist,  
so ist das nicht allgemein feststellbar.

Genauer: Es gibt dafür **kein universelles Verfahren**.

## Konsequenzen aus Situation im PK1

Kein Entscheidungsprogramm im PK1 möglich.

Spezielle Situation für Prolog-Interpreter:

- eingeschränkt durch Horn-Klauseln
- eingeschränkt durch Beweisstrategie im Interpreter (Frage nach Vollständigkeit/Korrektheit)

Entscheidungsverfahren existieren zumindest prinzipiell für

- aussagenlogische Programme oder
- Programme über endlichen Relationen

## Suche nach Beweis in PROLOG

**Ausgangspunkt und Zwischenzustände:**

Menge von zu beweisenden Teilzielen:

$$\text{subgoals} = \{ \text{subgoal}_1(\dots), \dots, \text{subgoal}_m(\dots) \}$$

Die Teilziele haben die Form  $\text{funkt}(\text{t}_1, \dots, \text{t}_n)$  .

Dabei bezeichnet  $\text{funkt}$  ein  $n$ -stelliges Prädikat, dessen Variablen  $X_i$  jeweils an Terme  $t_i$  gebunden sind.

## Suche nach Beweis in PROLOG

### Ziel (Ende des Verfahrens):

- subgoals = {}, d.h. alle Teilziele sind bewiesen
  - dabei Antwort „yes“ bzw.
  - Angabe der Terme, an die die Variablen der Anfrage gebunden wurden
- oder: kein weiterer Beweisversuch möglich
  - dabei Antwort „no“

## Such-Schritte für Beweis in PROLOG

- **Nächstes** zu beweisendes Teilziel:

$\text{funktork}(t_1, \dots, t_n) \in \text{subgoals}$

- **Nächste** Klausel der zugehörigen Prozedur:

$\text{funktork}(X_1, \dots, X_n) \text{ :- funktork}^1(X_1^1, \dots, X_{n1}^1), \dots, \text{funktork}^m(X_1^m, \dots, X_{nm}^m)$

- **Unifikation**

des Kopfes  $\text{funktork}(X_1, \dots, X_n)$  mit Teilziel  $\text{funktork}(t_1, \dots, t_n)$   
ergibt eine Variablensubstitution  $\sigma$   
(Ersetzung von Variablen durch Terme)

### Neuer Zwischenzustand:

$\text{subgoals} := \sigma( (\text{subgoals} - \{ \text{funktork}(t_1, \dots, t_n) \} ) \cup \{ \text{funktork}^1(X_1^1, \dots, X_{n1}^1), \dots, \text{funktork}^m(X_1^m, \dots, X_{nm}^m) \} )$

## Suche nach Beweis in PROLOG

Die Suche ist erfolgreich,  
wenn am Ende  $\text{subgoals}=\{\}$  gilt und  
in jedem Zwischenschritt  $i$  eine passende Klausel existiert,  
bei der die Unifikation mit einer Substitution  $\sigma_i$  gelingt.

Die Substitutionen ergeben in ihrer Gesamtheit die Terme,  
an die die Variablen  $X$  der Anfrage gebunden werden.

„Antwort-Substitution“:

$$\sigma(\mathbf{X}) = \sigma_k(\sigma_{k-1}(\dots \sigma_1(\mathbf{X}) \dots))$$

?- parent(zeus, X) .

X=hermes?

...

## Closed World Assumption - CWA (2)

Wann soll Interpreter Antwort „no“ auf Anfrage  $Q$  liefern?

*Varianten:* Dabei jeweils implizite Annahmen bzgl. Negation.

Wenn  $\neg Q$  bewiesen wurde.

Wenn  $Q$  nachweisbar nicht bewiesen werden kann.

Wenn alle Beweisversuche für  $Q$  fehlgeschlagen sind.

In Prolog:

*Variante 3* „Negation by (finite) failure“

„Wenn alle Beweisversuche für  $Q$  fehlgeschlagen sind“.



## CWA: Negation by failure

Wenn H nicht aus Programm P beweisbar ist,  
wird angenommen, dass  $\neg H$  beweisbar ist.

Inhaltlich oft sinnvoll

Allerdings: Veränderte Semantik  
Probleme mit „not“ in PROLOG

## CWA: Fallbeispiel PROLOG

Anfrage der Form ? - **query(argumente).**

Antwort „no“ bedeutet:

Bzgl. Fakten bei Faktenbasis als Datenbank:

–Anfrage mit konstantem Argument: Der konkret erfragte  
Fakt ist nicht in der Datenbank enthalten.

–Anfrage mit variablem Argument: Es ist kein  
entsprechender Fakt in der Datenbank enthalten.

Bzgl. Beweisversuch in Prolog-Programmen:

–Nach endlich vielen Schritten konnte gezeigt werden:  
Das Prädikat ist im gegebenen Programm nicht  
beweisbar („finite failure“).

# CWA: Fallbeispiel PROLOG

1) Anfragen der Form **? - query(argumente)**.  
mit variablen Argumenten sind Fragen nach der Existenz geeigneter Variablenwerte.

Falls sie existieren: Angabe der Werte.

Falls sie nicht existieren: Antwort „no“.

2) Anfragen der Form **? - not(query(argumente))**.

mit variablen Argumenten sind verneinte Fragen nach der Existenz geeigneter Variablenwerte.

Falls sie existieren: Antwort „no“.

Falls sie nicht existieren: Antwort „yes“.

3) Bei Definitionen der Art **männlich(X):- not(weiblich(X))**.

wird die Anfrage **? - männlich(X)** aber wie unter (2) beantwortet.

später mehr zu Theorembeweisern

H.D.Burkhard Winter-Semester 2005/06  
sowie zu Negation/CWA/Default-Annahmen  
Wissensrepräsentation-Grundlagen

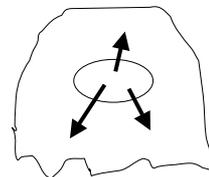
41

# Explizites vs. Implizites Wissen

PROLOG

explizit: Programm (Klauseln)

implizit: Beweisbare Fakten



Inferenz: Verfahren zur Herleitung von implizitem Wissen aus explizitem Wissen

PROLOG-Interpreter

## Deklaratives vs. Algorithmisches Wissen

### *Deklaratives Programm*

PROLOG („logisch“)  
LISP („funktional“)  
...

Ausführungswissen (Inferenz)  
im Interpreter fest kodiert:  
PROLOG: Tiefe-Zuerst-Suche  
Variablenbindung

### *Prozedurales/imperatives Progr.*

PASCAL  
C++  
JAVA, ...

Ausführungswissen  
(Steueranweisungen) im Code  
festzulegen

## Externalisiertes vs. internes Wissen

- externalisierbar = sprachlich kommunizierbares Wissen
- intern = „tacit knowledge“ (auch als „implizit“ bezeichnet)

Übergänge in beiden Richtungen:

- automatisieren
- externalisieren

Knowledge engineering  
Wissensakquisition



## Interpretation, Kommunikation

1. αλλγ. περφ} γβαρε Οριεντιερυνγεν ιμ Ραημεν  
αλλτ<sup>TM</sup>γλ. Ηανδλυνγσ– υνδ Σαχηζυσαμμενη<sup>TM</sup>νγε  
(Αλλταγσωισσεν);
2. ιμ ενγερεν, πηιλοσοπη. υνδ ωισσ. Σιννε ιμ Υντερ–  
σχηιεδ ζυ Μεινυγ υνδ Γλαυβεν διε αυφ  
Βεγρ} νδυγγεν βεζογενε υνδ στρενγεν  
←βερπρ} φυγγσποστυλατεν υντερλιεγενδε Κενντινισ,  
ινστιτυτιοναλισιερτ ιν δεν Ωισσενσχηαφτεν.

(Μεγερσ γροϋεσ Τασχηενλεξικον, 4.Αυφλαγε, 1992)

Ωισσεν, Κενντινισσε, Φ<sup>TM</sup>ηιγκειτεν

Κνοωλεδγε  πσ. Βελιεφ  (=Ανναημεν)

## Interpretation, Kommunikation

1. allg. verfügbare Orientierungen im Rahmen alltäggl.  
Handlungs- und Sachzusammenhänge  
(Alltagswissen);
2. im engeren, philosoph. und wiss. Sinne im Unter-  
schied zu Meinung und Glauben die auf  
Begründungen bezogene und strengen  
Überprüfungspostulaten unterliegende Kenntnis,  
institutionalisiert in den Wissenschaften.

(Meyers großes Taschenlexikon, 4.Auflage, 1992)

Wissen, Kenntnisse, Fähigkeiten

“Knowledge” vs. “Belief” (=Annahmen)

# Kommunizierbares Wissen

Voraussetzung:

- gemeinsames Wissen
- gemeinsamer Kontext
- gemeinsame Sprachkonventionen (Syntax/Semantik)

natürliche Sprache :

- vieldeutig, irregulär, flexibel, adaptiv

Sie sah die Fliege, die die Mutter schlug.  
Sie entfernte sich von ihr.

künstliche Sprache: Formalismen

- eindeutig, regelmäßig
- (hoffentlich) klare Semantik

# „gemeinsame Sprachkonventionen“

Statt um 15 Prozent dürfen die Mieten bei Wohnungen nur um 10 Prozent anziehen, die „nicht mit einer Zentralheizung und einem Bad ausgestattet sind“.

	mit Bad	ohne Bad
mit FHZ	15%	?
ohne FHZ	?	10%

# Gemeinsame Konventionen

rot  
255,0,0

- Etwas steht für etwas anderes
- Qualia: individuelles Empfinden
- *Symbol Grounding*

Roter Rauch bedeutet Feuer: gemeinsame Erfahrung  
Rotes Licht bedeutet „Halt“: Konvention

## Kontexte von Symbolen: Constraints

- Wärme, Aggression, Blut, Arbeiterklasse,...

## Assoziation:

- Ähnlichkeit von Kontexten

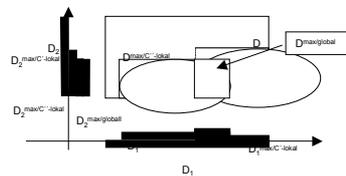
Notwendig für

- Begriffsbestimmung
- Wiedererkennung

**Ich sehe rot!**

# Bildliche Repräsentation

## Bilder/Grafiken zur Kommunikation



## Bildliche Sprache (räumliche Begriffe u.a.)

Hinter seinem hintergründigen Lächeln verbarg sich ...  
Unter Rücksicht auf ...

Herausforderung: Übertragung auf Computer?

# Zweckmäßigkeit von Repräsentationen

## Ökonomie: Rekonstruieren statt Abspeichern

Inferenz:

Herleitung von implizitem Wissen aus explizitem Wissen

## Experimente (Antwortzeiten für „ja“/„nein“)

Ein Kanarienvogel kann singen.	1.31 sec
Ein Kanarienvogel kann fliegen.	1.38 sec
Ein Kanarienvogel hat eine Haut.	1.47 sec
Ein Kanarienvogel ist ein Kanarienvogel.	1.00 sec
Ein Kanarienvogel ist ein Vogel.	1.17 sec
Ein Kanarienvogel ist ein Tier.	

Speicherökonomie  
vor Zeitökonomie

# Kontexte

Zur allgemeinen Begriffsbestimmung:

- Einordnen „Intelligenz“
- Beziehungen herstellen

Im konkreten Gebrauch

- Beziehungen klären Vampire
- Dereferenzierung Sie schenkte es ihm.

Allgemeiner („unsichtbarer“) Kontext:

- Alltagswissen Vampire sind ...

# Modellierung/Strukturierung

## Beschreibung von Zusammenhängen

- „ist ein ...“,
- „ist Teil von ...“
- „ist Ursache von ...“
- ...

Vgl. Ontologien

## Strukturierung

- Objektorientierte Modellierung
- Datenbank-Modellierung
- Logische Modellierung
- Terminologische Modellierung
- ...

# Modellierung/Strukturierung

Ontologie

eigentlich: Lehre vom Seienden

Ontologische Systeme = Begriffssysteme

- Botanisches System
- DIN-Bezeichnung
- Wissensrepräsentationssprachen

Dokumentenverwaltung:

- Strukturierung
- (künstliche) Ontologien
- Gegensatz: „Freie Texte“

Festlegung von Begriffen  
und ihren Beziehungen  
(Hierarchie, Vernetzung)

ants

liquids

solids

## Domänen-Wissen vs. Alltagswissen Wissen

Alltagswissen Wissen: „common sense knowledge“

**Warum** kann ein Schwein unter Wasser nicht Fahrrad fahren?

(Künstliche) Intelligenz: Turing-Test  
Mathematik vs. Bildzeitung

Modellierung von Zeit, Raum,....  
Beispiel: Allen-Kalkül

„Naive Theorien“

Ich kann mir ein Glas Wasser einschenken,  
ohne dafür Physik studiert zu haben.

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/06

## Wissen: Alltagswelt

Darstellbar als

- Sammlung von Fakten (Datenbank) + Inferenz ?
- Hätte viele Anwendungen

– Problem: Umgang mit „Inkonsistenzen“  
„Es gibt schreckliche Vampire.“

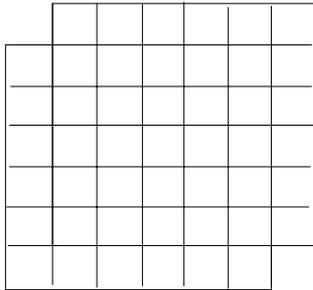
Projekt Cyc:  
Strukturierung in Mikrowelten  
Märchen  
Biologie  
Finanzen

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/06

Vorlesung Einführung in die KI  
Wissensrepräsentation-Grundlagen

# Zweckmäßigkeit von Repräsentationen

## Gute Definitionen ersparen Theoreme



Überdeckung durch  
Steine der Form   
Möglich?

# Unterscheidungen

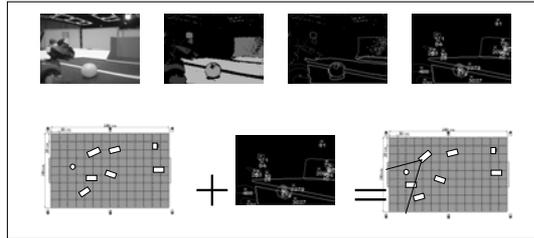
- Art des Wissens
  - sicher/unsicher (vielleicht morgen)
  - bestimmt/vage (billig)
  - genau/ungenau (ungefähr 70 cm)
- Gegenstand
  - allgemein: Zeit, Raum, ...
  - speziell: Diagnose, Konstruktion, ...
- Formalismus der Repräsentation
  - Regeln
  - Logik
  - Objekt-orientiert
  - . . .

# Nachbildung im Computer

- Interpretation
- Inferenz bzgl. Wissen / Annahmen
- Inferenz bzgl. Handlungsmöglichkeiten

Existiert ein Weg zum Knoten x?

Ich möchte eine Reise in den Süden.



# Formalisierung

Fakten

Beziehungen

Verfahren

Gegenstandsbereich (Domäne)

Gedankliche  
Aufarbeitung

Formalismus (Theorie, ...)

Maschinengerechte  
Umsetzung

Implementation (Programm,...)

„KI-Lücke“:

Abstand zwischen

Repräsentation

Implementation

# Formalisierung einer Domäne

## *Rekonstruktion eines Weltausschnitts*

Beschränkung auf *wesentliche* Inhalte  
Komplexität der Modellierung



Wieweit ist Alltagswissen  
wesentlich?

## Anforderungen

Kommunizierbarkeit

Ausdrucksfähigkeit

Vollständigkeit, Korrektheit

Operationalisierbarkeit:

Implementation

Entscheidbarkeit, Komplexität

Ich möchte eine Reise in den Süden

**Adäquatheit:**

**Wesentliches wird korrekt erfasst**

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/06

Vorles  
Wissel