4. Logik – und ihre Bedeutung für WBS

WBS verwenden formale Sprache L um Wissen zu repräsentieren

Grundidee: Problemlösen = Folgern aus in L dargestelltem Wissen

Folgern = implizites Wissen explizit machen

> **Syntax: legt fest, welche Zeichenketten (Formeln) zu L gehören**
>
> **Semantik: legt fest, was sie bedeuten**

Erst exakt definierte Semantik sagt, welche Folgerungen zulässig sind!!

Problem: Semantik unabhängig von (nichtlogischen) Symbolen zu definieren

Und was hat das alles mit Logik zu tun?

Logik

- **Folgerungsbegriff der zentrale Begriff der Logik**
- **Formale Logik = formale Theorie des Folgerns**
- **Klassische (Aussagen-, Prädikaten-) Logik Standard/Referenzsprache**
- **Gut verstanden, gut untersucht, allgemein akzeptiert, viele Resultate**
- **Viele Wissensrepräsentationssprachen sind Varianten von AL und PL**
- **Zu beachten: es gibt nicht die Logik, sondern viele verschiedene**
- **Beispiele: Modallogik, deontische Logik, nichtmonotone Logik, ...**

Was Sie noch über Logik wissen sollten: AL

Syntax:
Junktoren, Atome, Formelaufbau

Semantik:
Belegungen (Interpretationen), Modelle, Folgerungsbegriff, Kontradiktion, Tautologie, Erfüllbarkeit, Äquivalenz

Beweisverfahren:
Hilfsmittel: Normalformen, KNF, DNF,
Resolution, Widerspruchsbeweise, einige weitere Verfahren in Kürze

Wichtig: Unterscheidung Formelebene/Metaebene:

„Aus A folgt B“ ≠ „$A \rightarrow B$“!!

Aber:

„Aus A folgt B“ = „Die Formel $A \rightarrow B$ ist eine Tautologie“

Was Sie noch über Logik wissen sollten: PL

Syntax:

Junktoren, Quantoren, Variablen, Prädikanten-/Funktionssymbole

Formelaufbau

Semantik:

Strukturen, Interpretationen, Modelle, Folgerungsbegriff, Kontradiktion, Tautologie, Erfüllbarkeit, Äquivalenz

Beweisverfahren:

Normalformen, Skolemisierung, KNF, DNF, Klauselmengen, Unifikation, Resolution, Widerspruchsbeweise

Wichtig: im Gegensatz zu AL ist PL nicht entscheidbar!!

Resolution ist aber widerlegungsvollständig:

Klauselmenge inkonsistent genau dann wenn leere Klausel herleitbar

Von informeller Beschreibung zu formaler Repräsentation

Bsp.: ***Die dümmsten Bauern haben (= produzieren) die dicksten Kartoffeln***

Satz lässt (mindestens) folgende verschiedene Interpretationen zu:

***Wenn P zu den dümmsten Bauern gehört und K produziert,
dann ist K eine der dicksten Kartoffeln.***

***Wenn P zu den dümmsten Bauern gehört und eine Kartoffel K produziert,
dann ist K eine der dicksten Kartoffeln.***

***Wenn P zu den dümmsten Bauern gehört und K eine der dicksten Kartoffeln ist,
dann produziert P K.***

***Wenn K eine der dicksten Kartoffeln ist und von P produziert wird,
dann ist P einer der dümmsten Bauern.***

Welche Interpretation die richtige ist, ergibt sich oft nur aus dem Kontext.

Formalisierung

Abk.: $\forall x\ [\text{DümmsterBauer}(x) \leftrightarrow \text{Bauer}(x) \wedge \neg\exists z\ (\text{Bauer}(z) \wedge \text{Dümmer}(z,x))]$

$\forall x\ [\text{DicksteKartoffel}(x) \leftrightarrow \text{Kartoffel}(x) \wedge \neg\exists z\ (\text{Kartoffel}(z) \wedge \text{Dicker}(z,x))]$

Wenn P zu den dümmsten Bauern gehört und K produziert, dann ist K eine der dicksten Kartoffeln.

$$\forall x,y\ [\text{DümmsterBauer}(x) \wedge \text{erzeugt}(x, y) \rightarrow \text{DicksteKartoffel}(y)]$$

Wenn P zu den dümmsten Bauern gehört und eine Kartoffel K produziert, dann ist K eine der dicksten Kartoffeln.

$$\forall x,y\ [\text{DümmsterBauer}(x) \wedge \text{Kartoffel}(y) \wedge \text{erzeugt}(x, y) \rightarrow \text{DicksteKartoffel}(y)]$$

Wenn P zu den dümmsten Bauern gehört und K eine der dicksten Kartoffeln ist, dann produziert P K.

$$\forall x,y\ [\text{DümmsterBauer}(x) \wedge \text{DicksteKartoffel}(y) \rightarrow \text{erzeugt}(x, y)]$$

Wenn K eine der dicksten Kartoffeln ist und von P produziert wird, dann ist P einer der dümmsten Bauern.

$$\forall x,y\ [\text{DicksteKartoffel}\ (y) \wedge \text{erzeugt}(x,y) \rightarrow \text{DümmsterBauer}(x)]$$

Davis Putnam (Logemann Loveland) Verfahren für AL

Eingabe: **Klauselmenge C**

Ausgabe: **YES falls C erfüllbar, NO sonst**

Notation: **L Literal, C[L] entsteht aus C durch Streichen aller Klauseln mit L, Streichen des Komplements von L aus allen verbleibenden Klauseln**

(SAT)	if C={} then YES (erfüllbar)
(Empty)	if empty clause in C then NO (unerfüllbar)
(Unit)	if unit clause {*L*} in C then DPLL(C[*L*])
(Split)	select arbitrary L in C; if DPLL(C[*L*]) then YES else DPLL(C[-*L*])

Systematische Konstruktion eines Modells. Modell entspricht gewählten Literalen.

Tableauverfahren für AL

**Grundidee: erzeuge einen mit Formeln markierten Baum, so dass jeweils gilt:
F erfüllbar gdw. Formeln an einem Pfad von Wurzel zu Blatt erfüllbar**

Erzeugungsregeln (für Formeln ohne $\rightarrow$, $\leftrightarrow$):

$$\frac{\neg\neg H}{H} \qquad \frac{G \wedge H}{\begin{matrix} G \\ H \end{matrix}} \qquad \frac{\neg(G \wedge H)}{\neg G \mid \neg H} \qquad \frac{G \vee H}{G \mid H} \qquad \frac{\neg(G \vee H)}{\begin{matrix} \neg G \\ \neg H \end{matrix}}$$

Lies:

Wenn in Pfad $\neg\neg H$ vorkommt, hänge Knoten H an (falls es den noch nicht gibt).

Wenn in Pfad $G \wedge H$ vorkommt, erweitere ihn um Knoten G und Knoten H.

Wenn in Pfad $\neg(G \wedge H)$ vorkommt, verzweige mit Knoten G und Knoten H.

...

Offensichtlich widersprüchliche Pfade mit P, $\neg$P nicht weiterverfolgen.

Falls keine Regel mehr anwendbar und kein Widerspruch: erfüllbar.

AL Erfüllbarkeit als Constraint Satisfaction Problem

Variablen = aussagenlogische Variablen, Domäne jeweils {0,1}

Constraints = Klauseln (Bedingungen für Wertzuweisung an ihre Variablen)

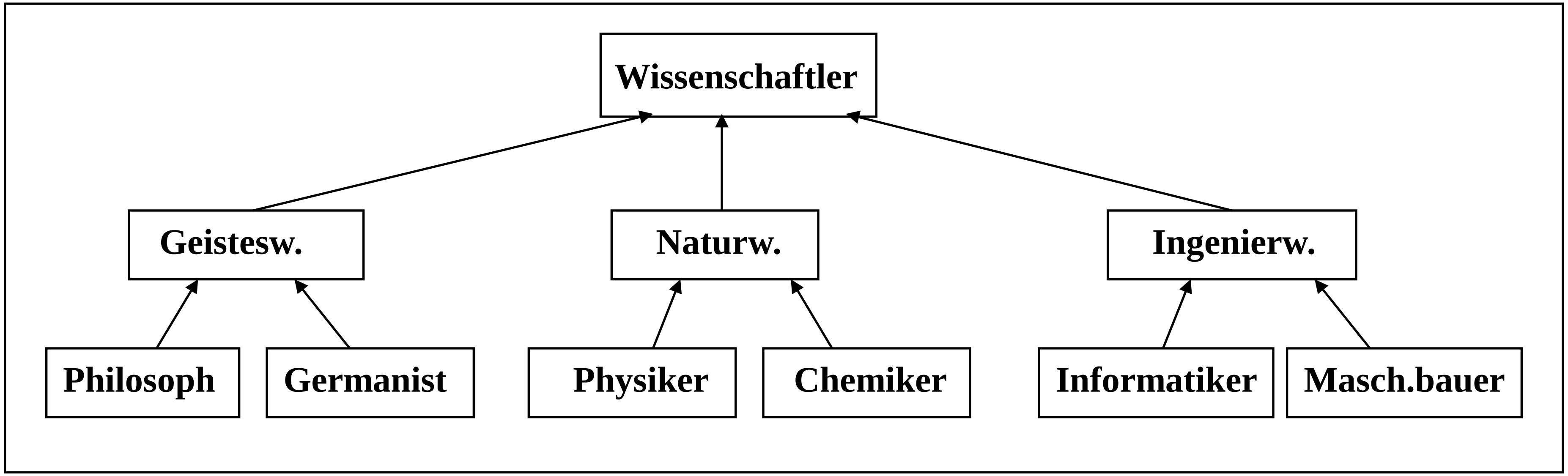
Standard-Constraint-Verfahren können angewendet werden

Besonders erfolgreich: lokale Verbesserungsverfahren:

1. **Beginne mit beliebiger Wertzuweisung (Variablenbelegung, Interpretation).**
2. **Falls kein Constraint (keine Klausel) verletzt, fertig.**
3. **Wähle Variable, für die Flippen des Wertes zu Erfüllen einer maximalen Anzahl von Klausen führt, und ändere ihren Wert.**
4. **Gehe zu 2.**

Logik und Ontologien

- **Logik repräsentiert das gesamte Wissen in Menge von Formeln**
- **Weitere Strukturierung der Formelmenge hilfreich**
- **Häufig 2 Bereiche zu unterscheiden:**
  1. **Terminologische Festlegungen (T-Box))**
  2. **Anwendung der Terminologie auf Objekte (A-Box)**



Peter ist Physiker, Fritz ist Naturwissenschaftler, Anna ist Informatiker(in)

Repräsentation in Logik

T-Box: $\forall x\, [\text{Philosoph}(x) \rightarrow \text{Geisteswissenschaftler}(x)]$

$\forall x\, [\text{Geisteswissenschaftler}(x) \rightarrow \text{Wissenschaftler}(x)]$

...

A-Box: **Physiker(Peter), Naturw(Fritz), Informatiker(Anna)**

Weitere mögliche terminologische Festlegungen:

$\forall x\, [\text{Geisteswissenschaftler}(x) \rightarrow \text{Philosoph}(x) \vee \ldots \vee \text{Germanist}(x)]$

$\forall x\, [\text{Philosoph}(x) \wedge \text{Informatiker}(x) \rightarrow \text{false}]$

$\forall x\, [\text{Genie}(x) \rightarrow \text{Philosoph}(x) \wedge \text{Informatiker}(x)]$

Grundidee der Beschreibungslogik: Stelle einfache Sprachkonstrukte für in Terminologien vorkommende Formeln zur Verfügung