AIFB

# Intelligente Systeme im World Wide Web

## Logische Grundlagen

Folien zur Vorlesung im Sommersemester 2006

Pascal Hitzler

Institut für Angewandte Informatik und Formale Beschreibungsverfahren (AIFB)

Universität Karlsruhe (TH)

AIFB

***„Logic is the Calculus of Computer Science“***

in etwa

„Die zentrale Rolle der Logik in der Informatik
ist vergleichbar mit der Rolle der Differential- und
Integralrechnung in den Naturwissenschaften.“

# Anwendungen

**Calculus**

Physik
Ingenieurswissenschaften
Chemie
Biologie
Via Statistik auch in
- Sozialwissenschaften
- Medizin

Etc.

**Logik**

Wissensrepräsentation
Automatisches Beweisen
Kognitive Robotik
Programmverifikation
Programmierung
Datenbanken
Elektronik
Komplexitätstheorie
Etc.

**Supported by**

- European Network of Excellence for Computational Logic (CoLogNET)
- Erasmus Mundus (EU)
- German Academic Exchange Service (DAAD)

**Double Degree** Between any two participating universities

**Start** Each winter term

**Duration** Two years

**Language** English

**Lecturers** Top international experts

**Scholarships** Available

**For more details see**
www.computational-logic.org

**Participating Universities**

- Libera Universitá di Bolzano, Italy

*"Logic is everywhere ..."*

**Contact**

**Prof. Steffen Hölldobler**
International Center for
Computational Logic

AIFB

# Inhalt der nächsten 5 Sitzungen

I. OWL – Syntax und allgemeines Verständnis
II. Logik (Wiederholung)
   a. Syntax
   b. Semantik
   c. Beweistheorie
III. OWL – Semantische Grundlagen
   a. Beschreibungslogiken
   b. Beweistheorie
IV. Ontologiesprache F-Logik

AIFB

## Zielsetzungen der nächsten 2 Vorlesungen

- Logik
  - Wiederholung der Grundlagen
  - Verständnis für automatisches Beweisen
  - Grundlage für die Kenntnis etablierter Ontologiesprachen und ihrer formalen Hintergründe

# Literatur

Wir wiederholen *kurz* und *informell* Grundlagen aus:

- Grundlagen der Informatik I
- Angewandte Informatik I

Wir setzen dann eine solide Kenntnis der Details voraus! Bitte selber wiederholen!

- Skript Studer: Grundlagen der Informatik I
- Skript Stucky/Studer: Angewandte Informatik I
- Uwe Schöning, Logik für Informatiker, Spektrum Akademischer Verlag, 5. Auflage 2000
- Steffen Hölldobler, Logik und Logikprogrammierung, Synchron Verlag, 3. Auflage 2003

AIFB

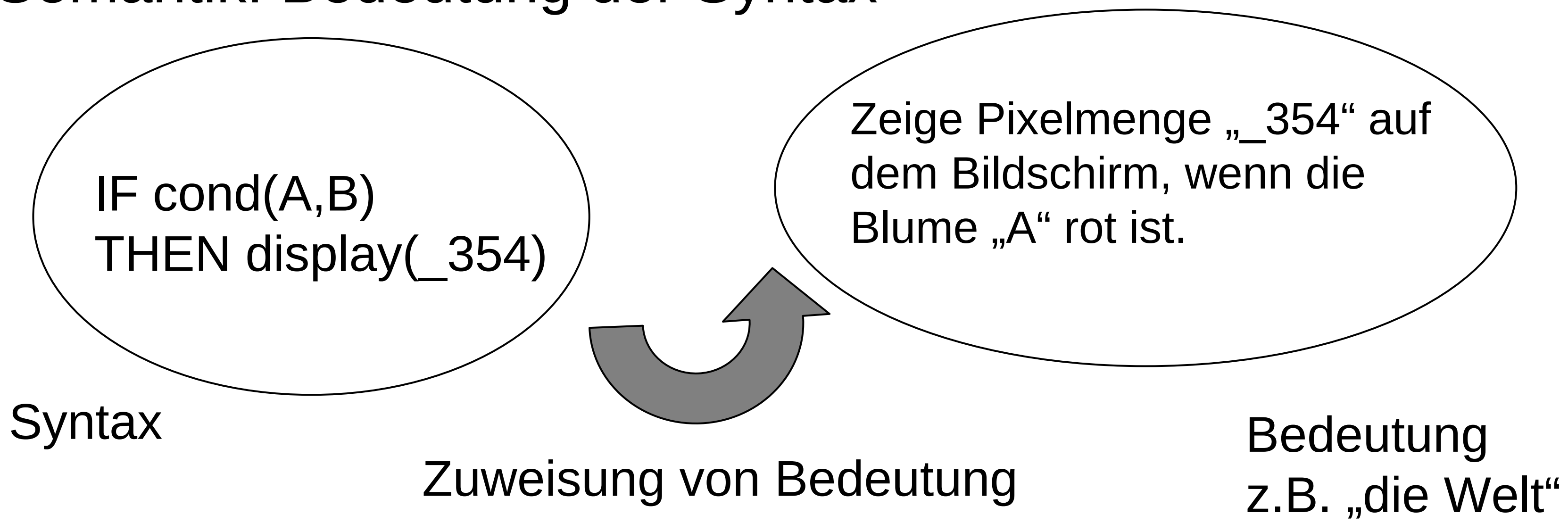
## Inhalt Logik

1. **Was ist Semantik – allgemein**
2. Syntax Aussagenlogik + FOL
3. Modelltheoretische Semantik
4. Beweistheoretische Semantik
   Vorbereitung: Normalformen
   a) Resolution
   b) Tableauverfahren
5. Eigenschaften der Logiken

# Syntax und Semantik

Syntax: Bedeutungsleere Zeichenfolgen

Semantik: Bedeutung der Syntax




Erfolgsbasis für Logik in der Informatik:

Semantik lässt sich auf Syntax zurückführen!

AIFB

# Was ist Semantik? Beispiel Programmiersprache

Berechnung der Fakultät

**Intendierte Semantik**

**Syntax**

```
FUNCTION f(n:natural):natural;
            BEGIN
            IF n=0 THEN f:=1
            ELSE f:=n*f(n-1);
            END;
```

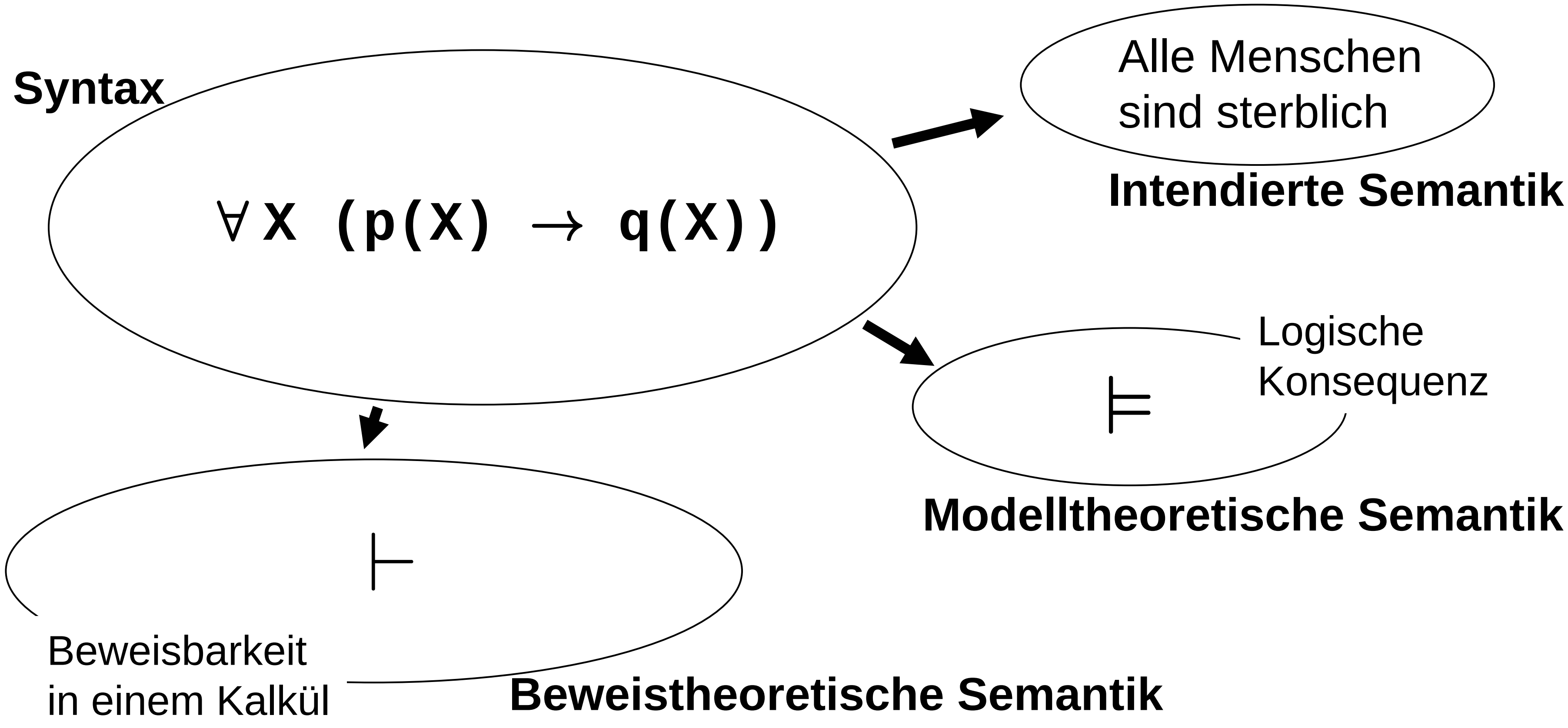
$f : n \mapsto n!$

**Formale Semantik**

Verhalten des Programms
bei Ausführung

**Prozedurale Semantik**

# Semantik von Logik/Wissensrepräsentationssprachen



# Abstraktere Formen der Semantik

Spieltheoretisch

Argumentationsbasiert

Algebraisch

Kategorientheoretisch

Geometrisch

Automatentheoretisch

Denotational

Fixpunktsemantik

…

AIFB

# Inhalt Logik

1. Was ist Semantik – allgemein
2. **Syntax Aussagenlogik + FOL**
3. Modelltheoretische Semantik
4. Beweistheoretische Semantik
   Vorbereitung: Normalformen
   a) Resolution
   b) Tableauverfahren
5. Eigenschaften der Logiken

AIFB

# Aussagenlogik: Syntax

| *Junktor* | *Name* | *Intuitive Bedeutung* |
|---|---|---|
| $\neg$ | Negation | „nicht“ |
| $\wedge$ | Konjunktion | „und“ |
| $\vee$ | Disjunktion | „oder“ |
| $\rightarrow$ | Implikation | „wenn – dann“ |
| $\leftrightarrow$ | Äquivalenz | „genau dann, wenn“ |

Prädikatssymbole/aussagenlogische Variablen, z.B. p, q, r, s, …

„richtiges“ Formen von Formeln – im Zweifelsfall klammern:

$((p \wedge \neg q) \vee s) \rightarrow \neg p$

$(p \leftrightarrow \neg q) \leftrightarrow (q \vee \neg p)$

Präzedenzen (bei uns): $\neg$ vor $\wedge,\vee$ vor $\rightarrow, \leftrightarrow$

Zusätzliche Klammern machen es trotzem oft lesbarer ☺

# Aussagenlogik: Beispiel

| *Einfache Aussagen* | *Modellierung* |
|---|---|
| Es regnet. | r |
| Die Straße wird nass. | n |
| Die Sonne ist grün | g |

| Zusammengesetzte Aussagen | Modellierung |
|---|---|
| Wenn es regnet, dann wird die Straße nass. | $r \rightarrow n$ |
| Wenn es regnet, und die Straße nicht nass wird, dann ist die Sonne grün. | $(r \wedge \neg n) \rightarrow g$ |

AIFB

# Prädikatenlogik erster Stufe (FOL): Syntax: Sprachelemente

| *Quantor* | *Name* | *Intuitive Bedeutung* |
|---|---|---|
| $\forall$ | Allquantor, universeller Quantor | „für alle“ |
| $\exists$ | Existenzquantor | „es existiert“ |

- Junktoren wie in der Aussagenlogik
- Variablen, z.B. X,Y,Z,…
- Konstantensymbole, z.B. a, b, c, …
- Funktionssymbole, z.B. f, g, h, … (mit Stelligkeit)
- Relations-/Prädikatssymbole, z.B. p, q, r, … (mit Stelligkeit)

$$(\forall X)(\exists Y)\ ((p(X) \vee \neg q(f(X),Y)) \rightarrow r(X))$$

AIFB

## FOL: Syntax

„richtiges“ Formen von *Termen* aus Variablen, Konstanten- und Funktionssymbolen:

f(X), g(a,f(Y)), s(a), .(H,T), x_location(Pixel)

„richtiges“ Formen von *Atomen* aus Relationssymbolen, deren Argumente Terme sind:

p(f(X)), q(s(a),g(a,f(Y))), add(a,s(a),s(a))

greater_than(x_location(Pixel),128)

„richtiges“ Formen von *Formeln* aus Atomen, Junktoren und Quantoren:

(∀Pixel)( greater_than( x_location(Pixel),128 ) $\rightarrow$ red(Pixel) )

Im Zweifelsfall klammern! Alle Variablen quantifizieren!

AIFB

# FOL Syntax: Beispiel *Addition*

$$(\forall X)(\forall Y)(\forall Z)\Big(\ add(a,X,X) \land (\ add(X,Y,Z) \rightarrow add(s(X),Y,s(Z))\ )\ \Big)$$

## Intendierte Semantik:

a … 0 (natürliche Zahl Null)

s … Nachfolgerfunktion/Addition von Eins

add(x,y,z) … „z ist die Summe von x und y“

AIFB

## FOL Syntax: Beispiel *Listen*

$(\forall H)(\forall T)(\ \mathrm{list}([]) \wedge (\mathrm{list}(T) \rightarrow \mathrm{list}(.(H,T))\ ))$

Informell: [] … leere Liste

.(H,T) … H ist Kopf, T Restliste

schreibe auch .(H,T) als [H|T]

$(\forall H)(\forall T)$

$(\ \mathrm{member}(a,[a|T])$

$\wedge\ (\ \mathrm{member}(a,T) \rightarrow \mathrm{member}(a,[H|T])\ )$

$)$

Intendierte Semantik:

member(x,liste) … “x ist Element von liste”

# FOL Syntax: Beispiel *Verwandtschaften*

$(\forall X)$ ( parent(X) $\leftrightarrow$ ( human(X) $\wedge$ $(\exists Y)$ parent_of(X,Y) ))

$(\forall X)$ ( human(X) $\rightarrow$ $(\exists Y)$ parent_of(Y,X) )

$(\forall X)$ (orphan(X) $\leftrightarrow$ (human(X) $\wedge$
$\neg(\exists Y)$ (parent_of(Y,X)$\wedge$ alive(Y))))

$(\forall X)(\forall Y)(\forall Z)$
( uncle_of(X,Z) $\leftrightarrow$ (brother_of(X,Y) $\wedge$ parent_of(Y,Z)) )

Intendierte Semantik: klar!

AIFB

# FOL Syntax: Beispiel *Pinguine*


$$
\begin{array}{ll}
( & (\forall X)(\ penguin(X) \rightarrow blackandwhite(X)\ ) \\
 & \wedge\ (\exists X)(\ oldTVshow(X) \wedge blackandwhite(X)\ ) \\
) & \rightarrow (\exists X)(\ penguin(X) \wedge oldTVshow(X)\ )
\end{array}
$$

**Intendierte Semantik?**

**Mit Hilfe der Logik können wir ganz formal zeigen, dass der Pinguin unlogisch ist.**

AIFB

# Inhalt Logik

1. Was ist Semantik – allgemein
2. Syntax Aussagenlogik + FOL
3. **Modelltheoretische Semantik**
4. Beweistheoretische Semantik
   Vorbereitung: Normalformen
   a) Resolution
   b) Tableauverfahren
5. Eigenschaften der Logiken

AIFB

# Aussagenlogik: Modelltheoretische Semantik

## Interpretation:

Abbildung aller Prädikatssymbole nach {w,f}.

Ist F eine Formel und I eine Interpretation, dann ist I(**F**) ein Wahrheitswert, der aus F und I mittels **Wahrheitstafeln** ermittelt wird.

| I(p) | I(q) | I($\neg$p) | I(p $\wedge$ q) | I(p $\vee$ q) | I(p $\rightarrow$ q) | I(p $\leftrightarrow$ q) |
|---|---|---|---|---|---|---|
| t | t | f | t | t | t | t |
| t | f | f | f | t | f | f |
| f | t | t | f | t | t | f |
| f | f | t | f | f | t | t |

## Aussagenlogik: Modelltheoretische Semantik

Wir schreiben I $\vDash$ F, wenn I(F)=w ist, und nennen dann die Interpretation I ein *Modell* der Formel F.

Zentrale Begriffe:

- allgemeingültig (Tautologie)
- erfüllbar
- widerlegbar
- widerspruchsvoll/widersprüchlich/**unerfüllbar**

# Prädikatenlogik: Modelltheoretische Semantik

Struktur:

- Festlegung eines Grundbereichs D.
- Konstantensymbole werden auf Elemente von D abgebildet.
- Funktionssymbole auf Funktionen auf D.
- Relationssymbole auf Relationen über D.

Dann:

- Terme werden zu Elementen von D.
- Relationssymbole mit Argumenten werden wahr oder falsch.
- Entsprechende Behandlung der Junktoren/Quantoren.

AIFB

# Prädikatenlogik: Modelltheoretische Semantik: Beispiel

$(\forall X)(\forall Y)(\forall Z)$

$($ $\quad add(a,X,X)$

$\quad \wedge ( add(X,Y,Z) \rightarrow add(s(X),Y,s(Z)) )$

$)$

Modell I:

Grundbereich: natürliche Zahlen $\mathbb{N}$

I(a) = 0

I(s): $n \mapsto n+1$

I(add(k,m,n))=w genau dann, wenn k+m=n ist.

I ist Modell der Formel.

# Prädikatenlogik: Modelltheoretische Semantik: Beispiel

$$F = \Big( (\forall X)(\text{ penguin}(X) \rightarrow \text{blackandwhite}(X) ) \wedge (\exists X)(\text{ oldTVshow}(X) \wedge \text{blackandwhite}(X) ) \Big) \rightarrow (\exists X)(\text{ penguin}(X) \wedge \text{oldTVshow}(X) )$$

Interpretation I:

Grundbereich:

eine Menge M, die Elemente a,b,c enthält.

… keine Konstanten- oder Funktionssymbole …

Wir zeigen: Die Formel ist widerlegbar (d.h. sie ist nicht allgemeingültig:

Sind I(penguin)(a), I(blackandwhite)(a), I(oldTVshow)(b), I(blackandwhite)(b) wahr, I(oldTVshow)(a) jedoch falsch, dann ist die Formel unter I falsch, d.h. $I \not\models F$.

**Mit Hilfe der Logik können wir ganz formal zeigen, dass der Pinguin unlogisch ist.**

# Prädikatenlogik: Modelltheoretische Semantik

Wir schreiben I $\models$ F, wenn I(F)=w ist, und nennen dann die Interpretation I ein *Modell* der Formel F.

Zentrale Begriffe:

allgemeingültig (Tautologie)

erfüllbar

widerlegbar

widerspruchsvoll/widersprüchlich/**unerfüllbar**

AIFB

## Der Begriff der logischen Konsequenz

Eine *Theorie* T ist eine Menge von Formeln.

Eine Interpretation I ist ein Modell für T, wenn I $\models$ G für jede Formel G in T gilt.

Eine Formel F ist eine *logische Konsequenz* aus T, wenn jedes Modell von T auch Modell von F ist. Wir schreiben dann T $\models$ F.

Zwei Formeln F,G heißen *logisch* (auch *semantisch*) *äquivalent*, wenn {F} $\models$ G und {G} $\models$ F gelten. Wir schreiben dann F $\equiv$ G.

AIFB

# Einige logische Äquivalenzen

$F \wedge G \equiv G \wedge F$

$F \vee G \equiv G \vee F$

$F \rightarrow G \equiv \neg F \vee G$

$F \leftrightarrow G \equiv (F \rightarrow G) \wedge (G \rightarrow F)$

$\neg(F \wedge G) \equiv \neg F \vee \neg G$

$\neg(F \vee G) \equiv \neg F \wedge \neg G$

$\neg\neg F \equiv F$

$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$

$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$

$\neg(\forall X)\ F \equiv (\exists X)\ \neg F$

$\neg(\exists X)\ F \equiv (\forall X)\ \neg F$

$(\forall X)(\forall Y)\ F \equiv (\forall Y)(\forall X)\ F$

$(\exists X)(\exists Y)\ F \equiv (\exists Y)(\exists X)\ F$

$(\forall X)\ (F \wedge G) \equiv (\forall X)\ F \wedge (\forall X)\ G$

$(\exists X)\ (F \vee G) \equiv (\exists X)\ F \vee (\exists X)\ G$

**DeMorgan’sche Regeln**

AIFB

# Inhalt Logik

1. Was ist Semantik – allgemein
2. Syntax Aussagenlogik + FOL
3. Modelltheoretische Semantik
4. **Beweistheoretische Semantik**
   **Vorbereitung: Normalformen**
   a) Resolution
   b) Tableauverfahren
5. Eigenschaften der Logiken

AIFB

## Vorbereitung: Normalformen

Fakt: Zu jeder Formel gibt es unendlich viele logisch äquivalente Formeln.

Für jede solche *Äquivalenzklasse* sucht man nun einfache *Repräsentanten*.
Diese heißen *Normalformen*.

Einfaches Beispiel:
schreibe $\neg F$ statt $\neg\neg\neg\neg\neg F$

AIFB

## Vorbereitung: Normalformen

Ziel: Umwandlung von Formeln in *Klauselform.*

Zwischenschritte:

1. Negationsnormalform
   alle Negationen stehen ganz innen
2. Pränexnormalform
   alle Quantoren stehen ganz vorne
3. Skolemisierte Pränexnormalform
   Eliminierung der Existenzquantoren
4. konjunktive Normalform (CNF) = Klauselform
   Darstellung als Konjunktion von Disjunktionen

AIFB

# Normalformen: 1. Negationsnormalform

Alle Negationszeichen nach innen ziehen
durch Verwendung der folgenden Äquivalenzen:

$F \leftrightarrow G \equiv (F \rightarrow G) \wedge (G \rightarrow F)$

$F \rightarrow G \equiv \neg F \vee G$

$\neg(F \wedge G) \equiv \neg F \vee \neg G$

$\neg(F \vee G) \equiv \neg F \wedge \neg G$

$\neg(\forall X)\ F \equiv (\exists X)\ \neg F$

$\neg(\exists X)\ F \equiv (\forall X)\ \neg F$

$\neg\neg F \equiv F$

Ergebnis:

- Keine $\rightarrow$ und $\leftrightarrow$ mehr.
- Keine mehrfachen Negationen mehr.
- Alle Negationszeichen stehen direkt vor Atomen.

AIFB

# Normalformen: 1. Negationsnormalform

Beispiel

$($ $\quad (\forall X)(\ penguin(X) \rightarrow blackandwhite(X)\ )$

$\quad \wedge (\exists X)(\ oldTVshow(X) \wedge blackandwhite(X)\ )$

$) \quad \rightarrow (\exists X)(\ penguin(X) \wedge oldTVshow(X)\ )$

wird zu

$\neg(\ (\forall X)(\ \neg penguin(X) \vee blackandwhite(X)\ )$

$\quad \wedge (\exists X)(\ oldTVshow(X) \wedge blackandwhite(X)\ )$

$) \vee (\exists X)(\ penguin(X) \wedge oldTVshow(X)\ )$

und dann zu

$( \quad (\exists X)(\ penguin(X) \wedge \neg blackandwhite(X)\ )$

$\quad \vee (\forall X)(\neg oldTVshow(X) \vee \neg blackandwhite(X)\ )$

$) \vee (\exists X)(\ penguin(X) \wedge oldTVshow(X)\ )$

# Vorbereitung: Normalformen

Ziel: Umwandlung von Formeln in *Klauselform.*

Zwischenschritte:

1. Negationsnormalform
   alle Negationen stehen ganz innen
2. **Pränexnormalform**
   alle Quantoren stehen ganz vorne
3. Skolemisierte Pränexnormalform
   Eliminierung der Existenzquantoren
4. konkunktive Normalform (CNF) = Klauselform
   Darstellung als Konjunktion von Disjunktionen

AIFB

## Normalformen: 2. Pränexnormalform

Erst Formel bereinigen
(Quantoren binden verschiedene Variablen).

( (∃X)( penguin(X) ∧ ¬blackandwhite(X) )
∨ (∀X)( ¬oldTVshow(X) ∨ ¬blackandwhite(X) )
) ∨ (∃X)( penguin(X) ∧ oldTVshow(X) )

wird zu

( (∃X)( penguin(X) ∧ ¬blackandwhite(X) )
∨ (∀Y)( ¬oldTVshow(Y) ∨ ¬blackandwhite(Y) )
) ∨ (∃Z)( penguin(Z) ∧ oldTVshow(Z) )

# Normalformen: 2. Pränexnormalform

Dann aus der Negationsnormalform einfach alle Quantoren in derselben Reihenfolge nach vorne ziehen.

$$( \quad (\exists X)(\ \mathrm{penguin}(X) \land \neg \mathrm{blackandwhite}(X)\ ) \\ \lor (\forall Y)(\ \neg \mathrm{oldTVshow}(Y) \lor \neg \mathrm{blackandwhite}(Y)\ ) \\ ) \lor (\exists Z)(\ \mathrm{penguin}(Z) \land \mathrm{oldTVshow}(Z)\ )$$

wird zu

$$(\exists X)(\forall Y)(\exists Z)(\ \ (\ \mathrm{penguin}(X) \land \neg \mathrm{blackandwhite}(X)\ ) \\ \lor (\ \neg \mathrm{oldTVshow}(Y) \lor \neg \mathrm{blackandwhite}(Y)\ )\ \ ) \\ \lor (\ \mathrm{penguin}(Z) \land \mathrm{oldTVshow}(Z)\ )$$

## Vorbereitung: Normalformen

Ziel: Umwandlung von Formeln in *Klauselform.*

Zwischenschritte:

1. Negationsnormalform
   alle Negationen stehen ganz innen
2. Pränexnormalform
   alle Quantoren stehen ganz vorne
3. **Skolemisierte Pränexnormalform**
   Eliminierung der Existenzquantoren
4. konkunktive Normalform (CNF) = Klauselform
   Darstellung als Konjunktion von Disjunktionen

AIFB

# Normalformen: 3. Skolemisierung

"Existenzquantoren entfernen"

$(\exists X)(\forall Y)(\exists Z)(\ \ (\text{penguin}(X) \land \neg\text{blackandwhite}(X))$

$\quad \lor (\neg\text{oldTVshow}(Y) \lor \neg\text{blackandwhite}(Y))\ \ )$

$\quad \lor (\text{penguin}(Z) \land \text{oldTVshow}(Z))$

wird zu …

$(\forall Y)(\ \ (\text{penguin}(a) \land \neg\text{blackandwhite}(a))$

$\quad \lor (\neg\text{oldTVshow}(Y) \lor \neg\text{blackandwhite}(Y))\ \ )$

$\quad \lor (\text{penguin}(f(Y)) \land \text{oldTVshow}(f(Y)))$

wobei a und f *neue* Symbole sind
(sog. *Skolemkonstanten* bzw. *–funktionen*).

# Normalformen: 3. Skolemisierung

Vorgehensweise:

- Entfernen der Existenzquantoren von links nach rechts.
- Gibt es keinen Allquantor links des zu entfernenden Existenzquantors, so wird die entsprechende Variable durch ein neues Konstantensymbol ersetzt.
- Gibt es *n* Allquantoren links des zu entfernenden Existenzquantors, so wird die entsprechende Variable durch ein neues Funktionssymbol mit Stelligkeit *n* ersetzt, dessen Argumente genau die Variablen der *n* Allquantoren sind.

# Vorbereitung: Normalformen

Ziel: Umwandlung von Formeln in *Klauselform.*

Zwischenschritte:

1. Negationsnormalform
   alle Negationen stehen ganz innen
2. Pränexnormalform
   alle Quantoren stehen ganz vorne
3. Skolemisierte Pränexnormalform
   Eliminierung der Existenzquantoren
4. **konjunktive Normalform (CNF) = Klauselform**
   Darstellung als Konjunktion von Disjunktionen

AIFB

# Normalformen: 4. Klauselform

Es gibt nur noch Allquantoren, also lassen wir sie weg:

$( \text{penguin}(a) \wedge \neg\text{blackandwhite}(a) )$

$\quad \vee ( \neg\text{oldTVshow}(Y) \vee \neg\text{blackandwhite}(Y) ) )$

$\quad \vee ( \text{penguin}(f(Y)) \wedge \text{oldTVshow}(f(Y))$

Mit Hilfe semantischer Äquivalenzen wird die Formel nun als Konjunktion von Disjunktionen geschrieben.

$$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$$

$$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$$

AIFB

# Normalformen: 4. Klauselform

( (penguin(a) ∧ ¬blackandwhite(a) )
∨ ( ¬oldTVshow(Y) ∨ ¬blackandwhite(Y) ) )
∨ ( penguin(f(Y)) ∧ oldTVshow(f(Y))

wird zu …

( ( penguin(a) ∨ ¬oldTVshow(Y) ∨ ¬blackandwhite(Y))
∧ ( ¬blackandwhite(a) ∨ ¬oldTVshow(Y) ∨ ¬blackandwhite(Y))
∨ ( penguin(f(Y)) ∧ oldTVshow(f(Y))

wird zu …

( penguin(a) ∨ ¬oldTVshow(Y) ∨ ¬blackandwhite(Y) ∨ penguin(f(Y)))
∧ ( penguin(a) ∨ ¬oldTVshow(Y) ∨ ¬blackandwhite(Y)
∨ oldTVshow(f(Y)))
∧ ( ¬blackandwhite(a) ∨ oldTVshow(Y) ∨ ¬blackandwhite(Y)
∨ penguin(f(Y)))
∧ ( ¬blackandwhite(a) ∨ oldTVshow(Y) ∨ ¬blackandwhite(Y)
∨ oldTVshow(f(Y)))

**Besser lesbar?**

**Für Maschinen schon!**

AIFB

## Normalformen: Eigenschaften

**Idee: Rückführung des Schließens auf die Suche nach Widersprüchen.**

Sei F eine Formel,

G die Pränexnormalform von F,

H die skolemisierte Pränexnormalform von G,

K die Klauselform von H.

Dann ist $F \equiv G$ und $H \equiv K$ aber i.A. $F \not\equiv K$.

Es gilt jedoch:

F ist unerfüllbar genau dann, wenn K unerfüllbar ist.

(Grundlage des Resolutionsverfahrens)

AIFB

## Skolemisierung ist keine Äquivalenztransformation

Die Formel $(\exists x)p(x)\lor\neg(\exists x)p(x)$ ist eine Tautologie.

- Negationsnormalform: $(\exists x)p(x)\lor(\forall y)\neg p(y)$
- Pränexnormalform: $(\exists x)(\forall y)(p(x)\lor\neg p(y))$
- Skolemnormalform: $(\forall y)(p(a)\lor\neg p(y))$
- Äquivalent dazu: $p(a)\lor\neg(\exists y)p(y)$

Die resultierende Formel ist keine Tautologie!

z.B. Interpretation I mit

$I(p(a))=f$

$I(p(b))=w$

# Inhalt Logik

1. Was ist Semantik – allgemein
2. Syntax Aussagenlogik + FOL
3. Modelltheoretische Semantik
4. **Beweistheoretische Semantik**
   Vorbereitung: Normalformen
   **a) Resolution**
   b) Tableauverfahren
5. Eigenschaften der Logiken

AIFB

# Automatisierung: Resolution – Vorbereitungen

Diese Aussagen sind äquivalent!

Theorie

- $\{F_1,\ldots,F_n\}$ hat $F_0$ als logische Konsequenz
- $\{F_1,\ldots,F_n\} \vDash F_0$
- $F_1 \wedge \ldots \wedge F_n \rightarrow F_0$ ist allgemeingültig
- $\neg(F_1 \wedge \ldots \wedge F_n \rightarrow F_0)$ ist unerfüllbar

Transformation in Klauselform

- $G_1 \wedge \ldots \wedge G_k$ ist unerfüllbar
- Das Resolutionsverfahren erlaubt die Ableitung eines Widerspruchs aus $G_1 \wedge \ldots \wedge G_k$.

AIFB

## Automatisierung: Resolution (Aussagenlogik)

Ist

$(p_1 \lor \ldots \lor p_k \lor \mathbf{p} \lor \neg q_1 \lor \ldots \lor \neg q_l) \land (r_1 \lor \ldots \lor r_m \lor \neg \mathbf{p} \lor \neg s_1 \lor \ldots \lor \neg s_n)$

wahr, dann:

Eins von **p**, **¬p** muss falsch sein. Also:

Eins der anderen muss wahr sein. D.h.

$p_1 \lor \ldots \lor p_k \lor \neg q_1 \lor \ldots \lor \neg q_l \lor r_1 \lor \ldots \lor r_m \lor \neg s_1 \lor \ldots \lor \neg s_n$

muss wahr sein.

Ergo: Ist $p_1 \lor \ldots \lor p_k \lor \neg q_1 \lor \ldots \lor \neg q_l \lor r_1 \lor \ldots \lor r_m \lor \neg s_1 \lor \ldots \lor \neg s_n$

**unerfüllbar**, dann auch

$(p_1 \lor \ldots \lor p_k \lor \mathbf{p} \lor \neg q_1 \lor \ldots \lor \neg q_l) \land (r_1 \lor \ldots \lor r_m \lor \neg \mathbf{p} \lor \neg s_1 \lor \ldots \lor \neg s_n)$

# Automatisierung: Resolution (Aussagenlogik)

$$\overset{K_1}{(p_1 \vee \ldots \vee p_k \vee \mathbf{p} \vee \neg q_1 \vee \ldots \vee \neg q_l)} \quad \overset{K_2}{(r_1 \vee \ldots \vee r_m \vee \mathbf{\neg p} \vee \neg s_1 \vee \ldots \vee \neg s_n)}$$

**Resolutionsschritt**

$$\{K_1, K_2\} \models K_3$$

$$\overset{K_3}{p_1 \vee \ldots \vee p_k \vee \neg q_1 \vee \ldots \vee \neg q_l \vee r_1 \vee \ldots \vee r_m \vee \neg s_1 \vee \ldots \vee \neg s_n}$$

Aus zwei Klauseln wird eine neue.

Werden Klauseln resolviert, die nur noch aus je einem Atom bzw. negierten Atom bestehen, dann entsteht eine „leere Klausel“, bezeichnet mit $\perp$.

## Automatisierung: Resolution (Aussagenlogik)

Vorgehensweise, um einen Widerspruch aus einer Menge M von Klauseln abzuleiten:

1. Wähle zwei Klauseln aus M und erzeuge aus ihnen eine neue Klausel K durch einen Resolutionsschritt.
2. Ist $K=\perp$, dann ist ein Widerspruch gefunden.
3. Falls $K\neq\perp$, füge K zur Menge M hinzu und gehe zu 1.

AIFB

# Automatisierung: Resolution (Prädikatenlogik)

In der Prädikatenlogik müssen bei der Resolution zusätzlich Variablenbindungen mit Hilfe von Substitutionen berücksichtigt werden.

Z.B. $(p(X,f(Y)) \vee q(f(X),Y))$ $(\neg p(a,Z) \vee r(Z))$

Resolution mit [X/a, Z/f(Y)] ergibt

$(q(f(a),Y) \vee r(f(Y)))$.

AIFB

## Automatisierung: Resolution (Prädikatenlogik) – Beispiel

Terminologisches Wissen (DL: *TBox*):

$(\forall X)\ (\ human(X) \rightarrow (\exists Y)\ parent\_of(Y,X)\ )$

$(\forall X)\ (\ orphan(X) \leftrightarrow$

$(human(X) \wedge \neg(\exists Y)\ (parent\_of(Y,X) \wedge alive(Y)))$

Wissen um Individuen (DL: *ABox*):

orphan(harrypotter)

parent_of(jamespotter,harrypotter)

Können wir folgern: $\neg$alive(jamespotter)?

(DL = Beschreibungslogik (Description Logic))

## Automatisierung: Resolution (Prädikatenlogik) – Beispiel

Zu zeigen:

$((\forall X)\ (\ \text{human}(X) \rightarrow (\exists Y)\ \text{parent\_of}(Y,X)\ )$

$\land (\forall X)\ (\text{orphan}(X) \leftrightarrow$

$\quad (\text{human}(X) \land \neg(\exists Y)\ (\text{parent\_of}(Y,X) \land \text{alive}(Y)))$

$\land\ \text{orphan}(\text{harrypotter})$

$\land\ \text{parent\_of}(\text{jamespotter},\text{harrypotter})$

$) \rightarrow \neg\text{alive}(\text{jamespotter}))$

ist allgemeingültig.

AIFB

# Automatisierung: Resolution (Prädikatenlogik) – Beispiel

Zu zeigen:

$\neg((\forall X)$ ( human(X) $\rightarrow$ $(\exists Y)$ parent_of(Y,X) )

$\wedge$ $(\forall X)$ (orphan(X) $\leftrightarrow$

(human(X) $\wedge$ $\neg(\exists Y)$ (parent_of(Y,X) $\wedge$ alive(Y)))

$\wedge$ orphan(harrypotter)

$\wedge$ parent_of(jamespotter,harrypotter)

) $\rightarrow$ $\neg$alive(jamespotter))

ist unerfüllbar.

# Automatisierung: Resolution (Prädikatenlogik) – Beispiel

Pränexnormalform:

$(\forall X)(\exists Y)(\forall X_1)(\forall Y_1)(\forall X_2)(\exists Y_2)$

$((\neg human(X) \vee parent\_of(Y,X))$

$\wedge (\neg orphan(X_1)$

$\quad \vee (human(X_1) \wedge (\neg parent\_of(Y_1,X_1) \vee \neg alive(Y_1)))$

$\wedge (orphan(X_2)$

$\quad \vee (\neg human(X_2) \vee (parent\_of(Y_2,X_2) \wedge alive(Y_2)))$

$\wedge\ orphan(harrypotter)$

$\wedge\ parent\_of(jamespotter,harrypotter))$

$\wedge\ alive(jamespotter))$

# Automatisierung: Resolution (Prädikatenlogik) – Beispiel

Klauselform:

$( \neg human(X) \lor parent\_of(f(X),X) )$

$\land (\neg orphan(X_1) \lor human(X_1))$

$\land (\neg orphan(X_1) \lor \neg parent\_of(Y_1,X_1) \lor \neg alive(Y_1))$

$\land (orphan(X_2) \lor \neg human(X_2) \lor$
$parent\_of(g(X,X_1,Y_1,X_2),X_2))$

$\land (orphan(X_2) \lor \neg human(X_2) \lor alive(g(X,X_1,Y_1,X_2)))$

$\land\ orphan(harrypotter)$

$\land\ parent\_of(jamespotter,harrypotter))$

$\land\ alive(jamespotter)$

AIFB

# Automatisierung: Resolution (Prädikatenlogik) – Beispiel

Wissen:

1. $( \neg human(X) \vee parent\_of(f(X),X) )$
2. $(\neg orphan(X_1) \vee human(X_1))$
3. $(\neg orphan(X_1) \vee \neg parent\_of(Y_1,X_1) \vee \neg alive(Y_1))$
4. $(orphan(X_2) \vee \neg human(X_2) \vee parent\_of(g(X,X_1,Y_1,X_2),X_2))$
5. $(orphan(X_2) \vee \neg human(X_2) \vee alive(g(X,X_1,Y_1,X_2)))$
6. orphan(harrypotter)
7. parent_of(jamespotter,harrypotter))
8. alive(jamespotter)

Abgeleitete Klauseln:

9. $\neg orphan(harrypotter) \vee \neg alive(jamespotter)$ (3,7)
10. $\neg orphan(harrypotter)$ (8,9)
11. $\perp$ (6,10)

AIFB

# Automatisierung: Resolution (Prädikatenlogik)

## Ein technisches Detail:

Damit das prädikatenlogische Resolutionsverfahren vollständig ist, wird neben der Resolutionsregel noch eine Faktorisierungsregel benötigt. Sie besagt:

Einthält eine Klausel K zwei Literale $L_1$ und $L_2$, die unifizierbar mit mgu $\sigma$ sind, dann kann $K\sigma$ abgeleitet werden.

AIFB

# Automatisierung: SLD-Resolution (Prädikatenlogik): pures Prolog

- Implementierung der prädikatenlogischen Resolution.
- Einschränkung: Klauseln dürfen nur je ein nichtnegiertes Atom enthalten.
  Dadurch schnelleres Verfahren: SLD-Resolution.
  (Terminiert aber nicht immer.)
- Darstellung: $(p \vee \neg q_1 \vee \ldots \vee \neg q_n)$ als $p \leftarrow q_1,\ldots,q_n$.
- "Aufsammeln" der Substitutionen beim Resolvieren zur Fragebeantwortung.

evtl. Demo

AIFB

# Inhalt Logik

1. Was ist Semantik – allgemein
2. Syntax Aussagenlogik + FOL
3. Modelltheoretische Semantik
4. **Beweistheoretische Semantik**
   Vorbereitung: Normalformen
   a) Resolution
   **b) Tableauverfahren**
5. Eigenschaften der Logiken

# Automatisierung: Tableauverfahren (Prädikatenlogik)

## Vorgehensweise:

- Zeige Allgemeingültigkeit von Formel F.
- Keine Umwandlung in Normalform nötig.
  (hier: nur Elimination von $\rightarrow$ und $\leftrightarrow$)
- Negation der Formel: $\neg$F
- Konstruiere Tableau für $\neg$F.
  Kann das Tableau *abgeschlossen* werden, dann ist F allgemeingültig.

# Tableauverfahren: Vorgehensweise

- schreibe ¬F ins Tableau
- bis das Tableau abgeschlossen ist tue folgendes
  - wähle eine Formel in einem nicht abgeschlossenen Zweig
  - füge nach den Regeln auf der nächsten Folie Formeln zum Tableau hinzu

Kann das Tableau abgeschlossen werden, so ist ¬F unerfüllbar d.h. F allgemeingültig.

AIFB

## Regeln für Tableauerstellung:

$$\frac{\neg\neg F}{F} \qquad \frac{F_1 \wedge F_2}{\begin{array}{c}F_1\\ F_2\end{array}} \qquad \frac{F_1 \vee F_2}{F_1 | F_2} \qquad \frac{\neg(F_1 \vee F_2)}{\begin{array}{c}\neg F_1\\ \neg F_2\end{array}} \qquad \frac{\neg(F_1 \wedge F_2)}{\neg F_1 | \neg F_2}$$

---

$$\frac{(\forall X)F}{F\{X \mapsto Y\}} \qquad \frac{\neg(\exists X)F}{\neg F\{X \mapsto Y\}}$$

mit $Y$ als neue freie Variable

---

$$\frac{(\exists X)F}{F\{X \mapsto f(X_1,\ldots,X_n)\}} \qquad \frac{\neg(\forall X)F}{\neg F\{X \mapsto f(X_1,\ldots,X_n)\}}$$

mit $f$ als neuem Skolemfunktionszeichen und
$X_1,\ldots,X_n$ als alle in $F$ frei vorkommenden Variablen

AIFB

# Anwendung der Tableauregeln

- Die Regeln für Disjunktion ($F_1 \vee F_2$) und für negierte Konjunktion ($\neg(F_1 \wedge F_2)$) *teilen* das Tableau in zwei Zweige, wie im nachfolgenden Beispiel.
- Ein Zweig ist abgeschlossen, wenn in ihm eine Formel F und deren Negation $\neg$F vorkommt.
- Ein Tableau ist abgeschlossen, wenn alle seine Zweige abgeschlossen sind.

AIFB

## Automatisierung: Tableaus (Prädikatenlogik) – Beispiel

$\neg(\neg((\forall X)\ (\neg \mathrm{orphan}(X) \vee$

$(\mathrm{human}(X) \wedge \neg(\exists Y)\ (\mathrm{parent\_of}(Y,X) \wedge \mathrm{alive}(Y))))$

$\wedge\ \mathrm{orphan}(\mathrm{harrypotter})$

$\wedge\ \mathrm{parent\_of}(\mathrm{jamespotter},\mathrm{harrypotter})$

$)\ \vee\ \neg \mathrm{alive}(\mathrm{jamespotter}))$

Kürzer:

$\neg(\neg(\quad (\forall X)\ (o(X) \rightarrow (h(X) \wedge \neg(\exists Y)\ (\mathrm{pof}(Y,X) \wedge \mathrm{al}(Y))))$

$\wedge\ o(\mathrm{harry}) \wedge \mathrm{pof}(\mathrm{james},\mathrm{harry}))$

$\vee\ \neg \mathrm{al}(\mathrm{james}))$

Beispieltableau auf nächster Folie!

- $\neg(\neg( \quad (\forall X)\ (\neg o(X) \lor (h(X) \land \neg(\exists Y)\ (pof(Y,X) \land al(Y))))$
  $\land\ o(harry) \land pof(james,harry)) \quad \lor \neg al(james))$
- $\neg\neg( \ (\forall X)\ (\neg o(X) \lor (h(X) \land \neg(\exists Y)\ (pof(Y,X) \land al(Y))))$
  $\land\ o(harry) \land pof(james,harry))$
- $\neg\neg al(james)$
- $(\forall X)\ (\neg o(X) \lor (h(X) \land \neg(\exists Y)\ (pof(Y,X) \land al(Y))))$
  $\land\ o(harry) \land pof(james,harry)$
- $al(james)$
- $(\forall X)\ (\neg o(X) \lor (h(X) \land \neg(\exists Y)\ (pof(Y,X) \land al(Y))))$
- $o(harry)$
- $pof(james,harry)$
- $\neg o(Z) \lor (h(Z) \land \neg(\exists Y)\ (pof(Y,Z) \land al(Y)))$ (Disjunktion! Teile Tableau!)

| | |
|---|---|
| • $\neg o(Z)$ mit [Z/harry] | • $h(Z) \land \neg(\exists Y)\ (pof(Y,Z) \land al(Y)))$ <br> • $h(Z)$ <br> • $\neg(\exists Y)\ (pof(Y,Z) \land al(Y)))$ <br> • $\neg(pof(W,Z) \land al(W))$ |

globale Bindung!

| | |
|---|---|
| • $\neg pof(W,Z)$ mit [W/james] | • $\neg al(W)$ |

globale Bindung!

AIFB

# Inhalt Logik

1. Was ist Semantik – allgemein
2. Syntax Aussagenlogik + FOL
3. Modelltheoretische Semantik
4. Beweistheoretische Semantik
   Vorbereitung: Normalformen
   a) Resolution
   b) Tableauverfahren
5. **Eigenschaften der Logiken**

# Eigenschaften der Prädikatenlogik

- Monotonie
  Bei Vergrößerung des Wissens gehen keine Schlussfolgerungen verloren.
- Kompaktheit
  Für jede Schlußfolgerung aus einer Theorie genügt eine endliche Teilmenge der Theorie.
- Semientscheidbarkeit
  Alle *wahren* Schlüsse lassen sich finden, wenn man lange genug sucht.

# Eigenschaften der Aussagenlogik

- Alle genannten Eigenschaften der Prädikatenlogik.
- Entscheidbarkeit
  Alle *wahren* Schlüsse lassen sich finden, und alle *falschen* Schlüsse lassen sich widerlegen, wenn man lange genug sucht.
  D.h. es gibt *immer terminierende* automatische Beweiser.

AIFB

# Wichtige Fragmente von FOL

- Aussagenlogik
- Datalog (Wie pures/reines Prolog, aber ohne Funktionssymbole)
  entscheidbar
- Disjunktives Datalog (Klauseln ohne Funktionssymbole)
  entscheidbar
- Hornklauseln (pures/reines Prolog)
  seminentscheidbar
- Beschreibungslogiken
  entscheidbar (manche)

z.B. OWL $\rightarrow$ nächster Teil der Vorlesung

# Inhalt der nächsten 3 Sitzungen

I. OWL – Syntax und allgemeines Verständnis

II. Logik (Wiederholung)

   a. Syntax

   b. Semantik

   c. Beweistheorie

III. OWL – Semantische Grundlagen

   a. Beschreibungslogiken

   b. Beweistheorie

IV. Ontologiesprache F-Logik