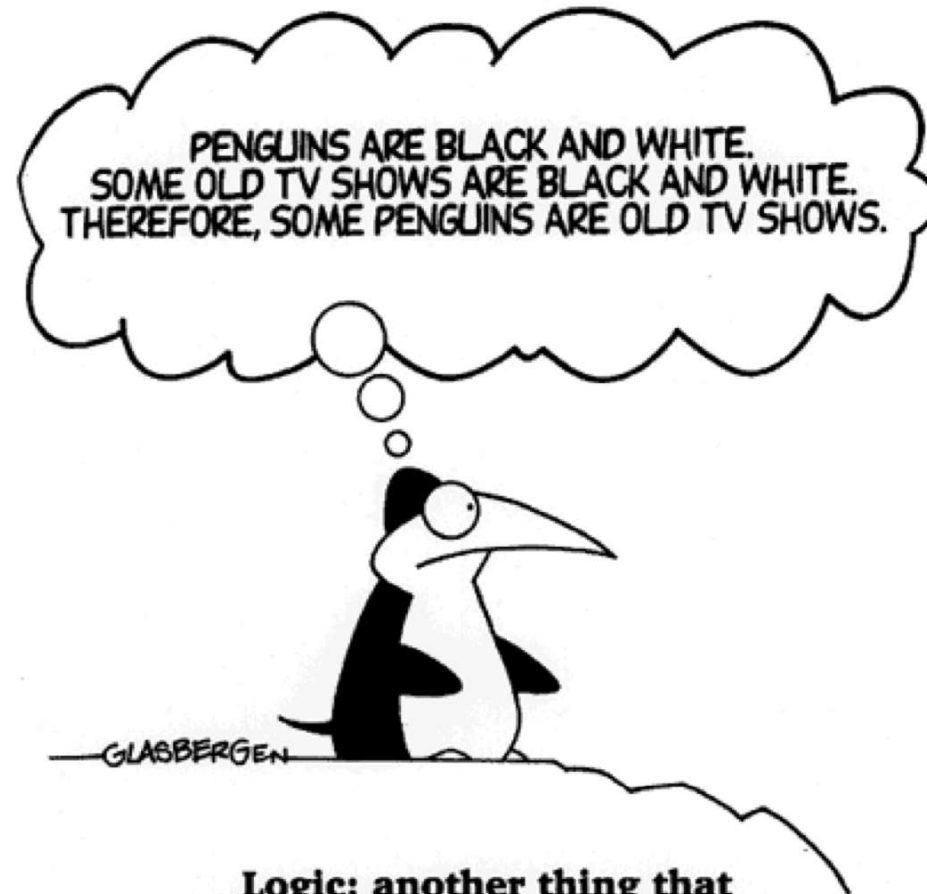


## Teil 7



**Logic: another thing that penguins aren't very good at.**

# Grundlagen Logik

# Was ist Logik?

---

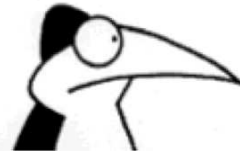
- etymologische Herkunft: griechisch λογος bedeutet „Wort, Rede, Lehre“ (s.a. Faust I...)
- Logik als Argumentation:



Alle Menschen sind sterblich.  
Sokrates ist ein Mensch.  
 Also ist Sokrates sterblich.



Warum?



Alle Pinguine sind schwarz-weiß.  
Einige alte TV-Shows sind schwarz-weiß.  
 Einige Pinguine sind alte TV-Shows.



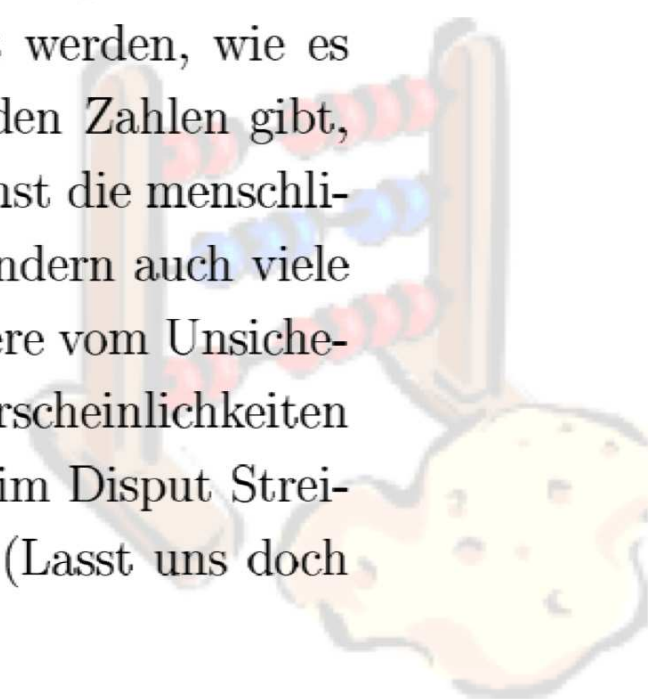
- Definition für diese Vorlesung:  
*Logik ist die Lehre vom formal korrekten Schließen.*

## Warum formal?



- **Automatisierbarkeit! Eine „Rechenmaschine“ für Logik!!**  
**G. W. Leibniz (1646-1716):**

„alle menschlichen Schlussfolgerungen müssten auf irgendeine mit Zeichen arbeitende Rechnungsart zurückgeführt werden, wie es sie in der Algebra und Kombinatorik und mit den Zahlen gibt, wodurch nicht nur mit einer unzweifelhaften Kunst die menschliche Erfindungsgabe gefördert werden könnte, sondern auch viele Streitigkeiten beendet werden könnten, das Sichere vom Unsicheren unterschieden und selbst die Grade der Wahrscheinlichkeiten abgeschätzt werden könnten, da ja der eine der im Disput Streitenden zum anderen sagen könnte: *Calculemus* (Lasst uns doch nachrechnen).“



# Grundbegriffe der Logik

---

Interpretation  
Modell  
Erfüllbarkeit  
Ableitungsregel  
Folgerung  
Term  
Proposition  
Satz  
Domäne  
Atom  
Formel  
Entscheidbarkeit  
Deduktionskalkül  
Individuum  
Syntax  
Semantik  
Diskursuniversum  
Tautologie  
Modelltheorie  
Widerspruch

## Wie funktioniert Logik?

---

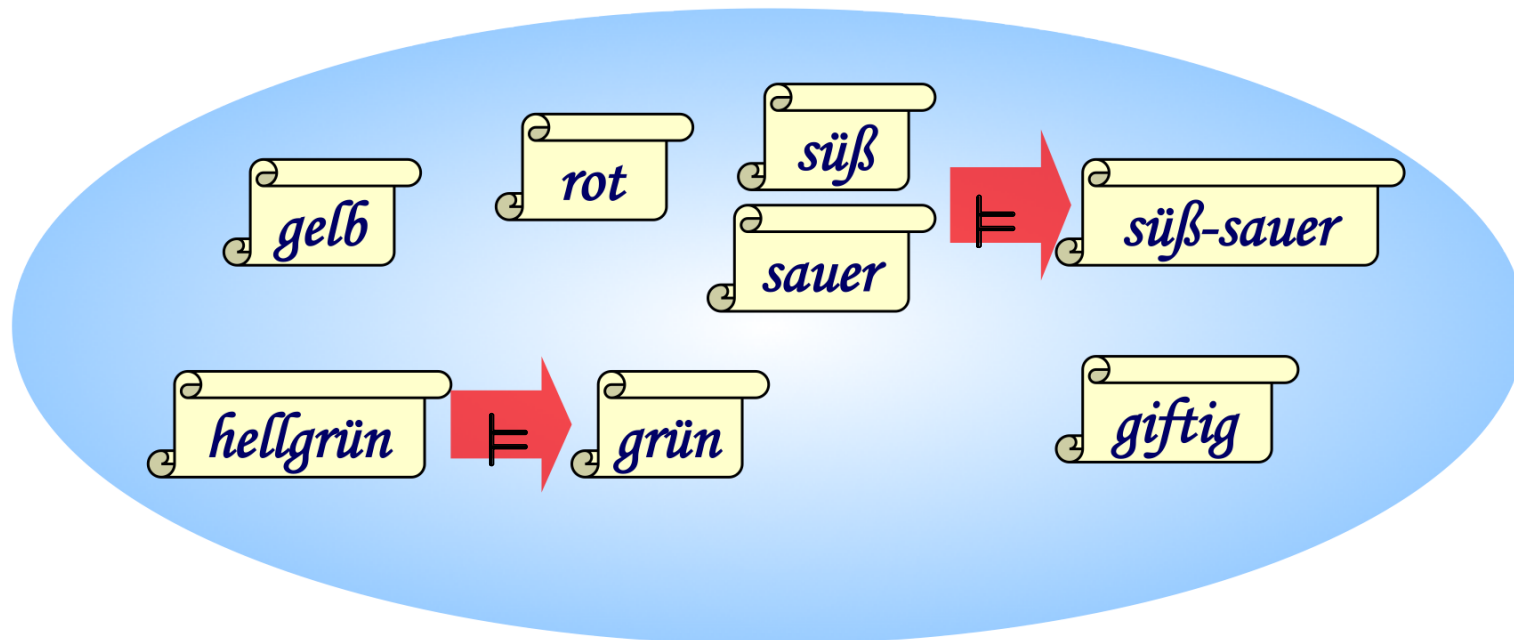
Alle Menschen sind sterblich.  
Sokrates ist ein Mensch.  
Also ist Sokrates sterblich.

*Logik ist die Lehre vom formal korrekten Schließen.*

- Was schließen wir *woraus*?
- Beschreibende Grundelemente der Logik nennen wir **Sätze**.

# Wie funktioniert Logik? Sätze und Schlussfolgerungen

- Jede Logik besteht aus einer Menge von **Sätzen** zusammen mit einer **Schlussfolgerungsrelation** (entailment relation). Letztere liefert die Semantik (grch. σημαντικός – *zum Zeichen gehörend*).



# Folgerung und Äquivalenz von Sätzen

---

Formal:  $L := (S, \models)$  mit  $\models \in 2^S \times S$

Dabei bedeutet für

⇒ eine Menge  $\Phi \subseteq S$  von Sätzen und

⇒ einen Satz  $\varphi \in S$

$$\Phi \models \varphi$$

„Aus den Sätzen  $\Phi$  folgt der Satz  $\varphi$ “ oder auch

„ $\varphi$  ist eine logische Konsequenz aus  $\Phi$ .“

Gilt für zwei Sätze  $\varphi$  und  $\psi$ , dass sowohl  $\{\varphi\} \models \psi$  als auch  $\{\psi\} \models \varphi$ , dann sind diese Sätze (*logisch*) *äquivalent* und man schreibt auch  $\psi \equiv \varphi$ .

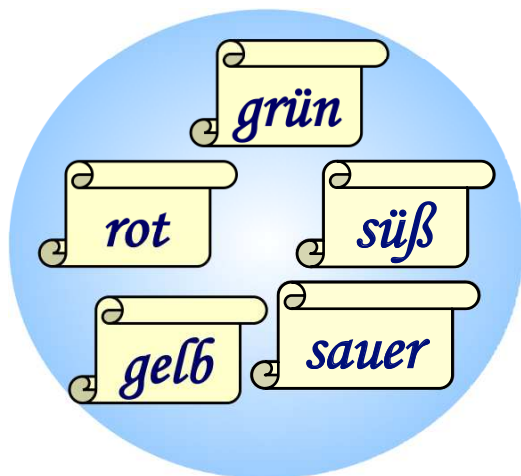
## Wie funktioniert Logik? Syntax.

Syntax (von grch. συνταξις – *Zusammenstellung, Satzbau*)  
erschließt sich über die Frage

- Was ist ein „richtiger“ Satz? D.h. wie wird die Menge der Sätze einer Logik definiert?

Nutzung von „Erzeugungsregeln“ zur Definition (Konstruktion)  
von wohlgeformten Sätzen, z.B.:

### Grundelemente:



**Syntax-Regel:** „Wenn  $\varphi$  und  $\psi$  Sätze sind, dann auch  $\varphi$ - $\psi$ “



*Konstruktor oder Junktor*



## Wie funktioniert Logik? Ausdrucksstärke.

---

Tradeoff: Logiken mit vielen Ausdrucksmitteln (Konstruktoren /Junktoren) sind:

- komfortabler in der Verwendung (verschiedene und komplexe Sachverhalte sind einfach auszudrücken), aber
- schwieriger (meta)mathematisch zu handhaben (Beweisen von Eigenschaften der Logik umständlicher).

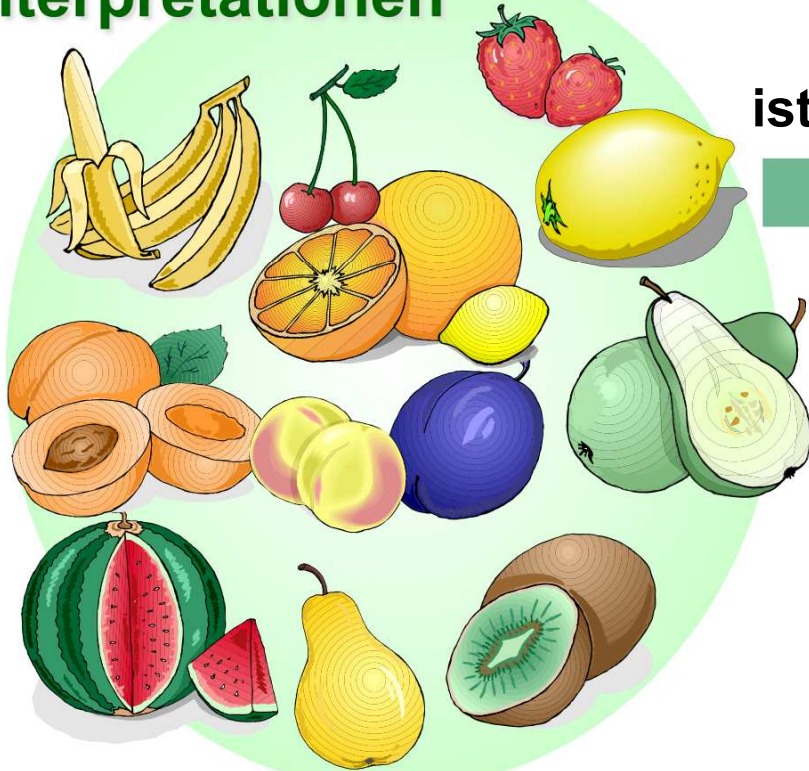
Möglicher Ausweg: Einschränkung der Sätze auf Teilmenge, die für jeden Satz der Logik einen logisch äquivalenten Vertreter enthält (vgl. Normalformen, minimale Junktorenmengen...) und Definition der anderen Sätze/Junktoren als „syntactic sugar“.

Wird eine Logik über dieses Maß hinaus eingeschränkt, erhält man ein *Fragment* der ursprünglichen Logik mit geringerer *Ausdrucksstärke*.

# Wie funktioniert Logik? - Modelltheorie

Eine Möglichkeit, die **Schlussfolgerungsrelation** zu definieren besteht über **Interpretationen** bzw. **Modelle**.

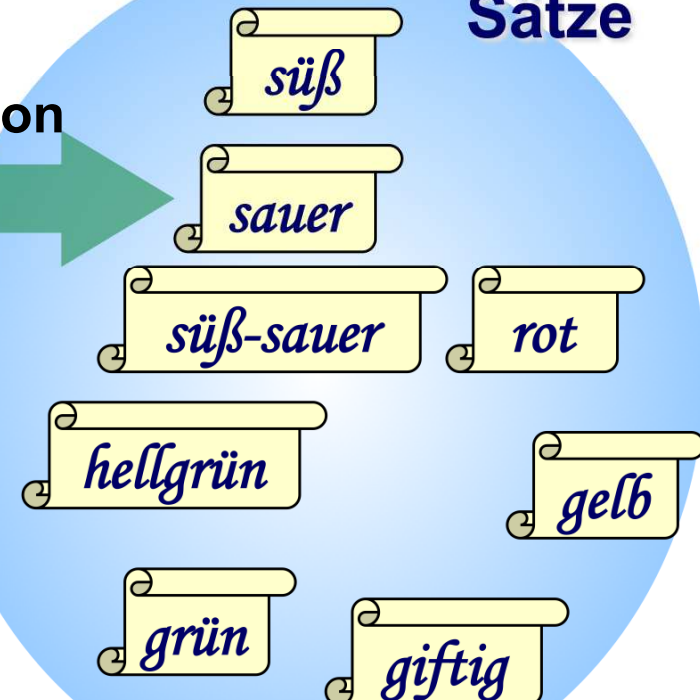
## Interpretationen



ist Modell von

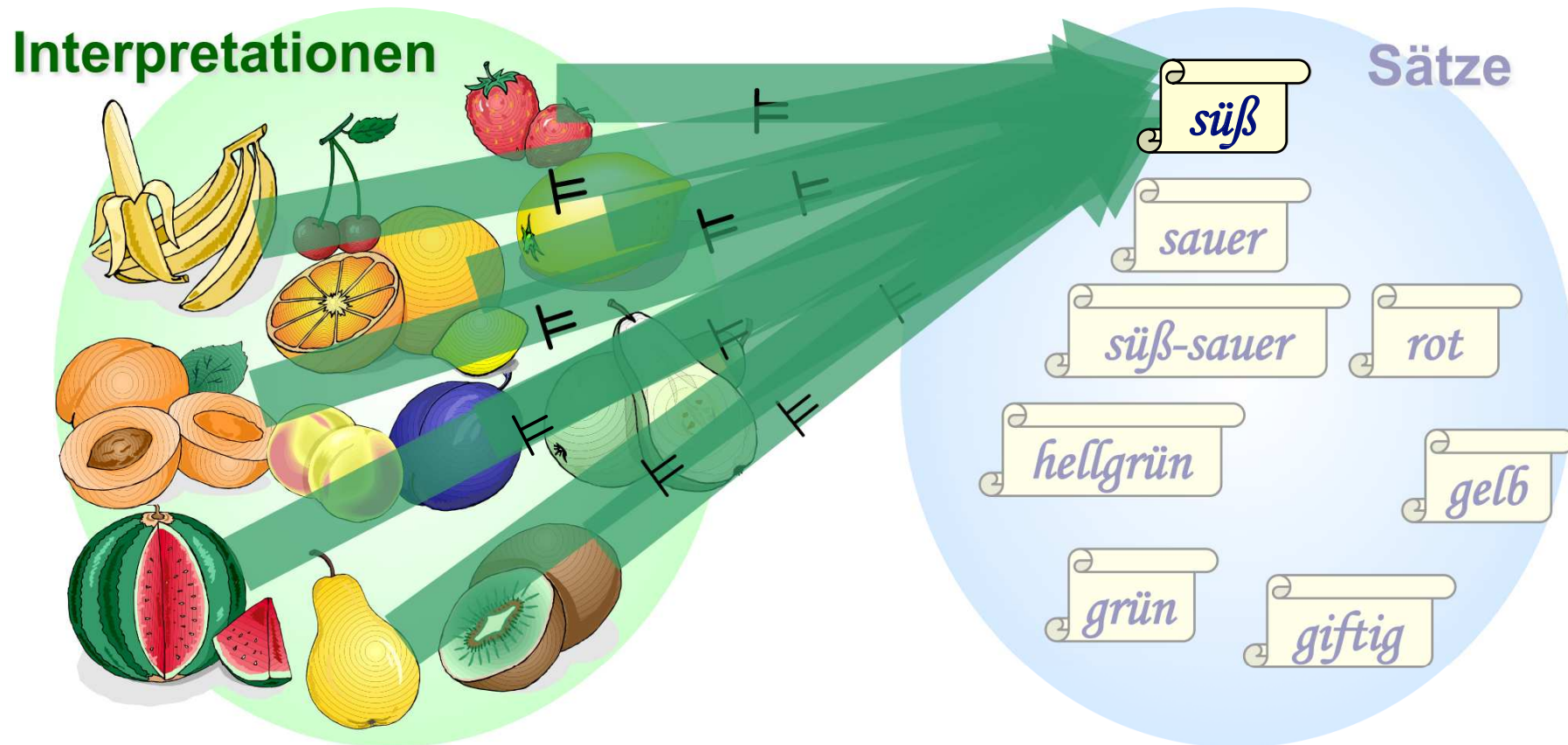


## Sätze



# Wie funktioniert Logik? - Modelltheorie

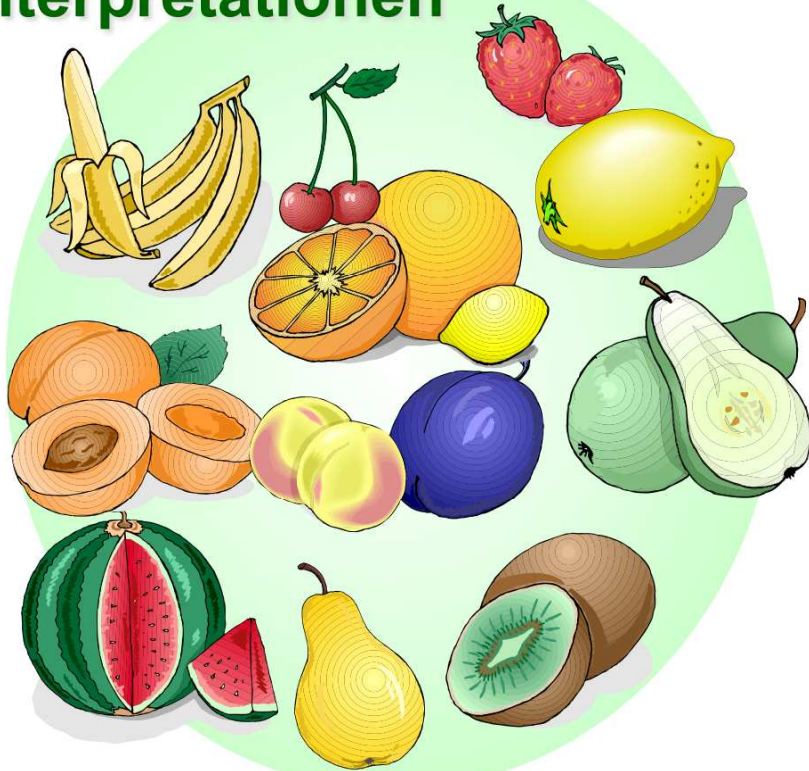
Sätze, für die **jede** Interpretation ein Modell ist, heißen *allgemeingültig* oder *Tautologien* (grch. ταυτολογία).



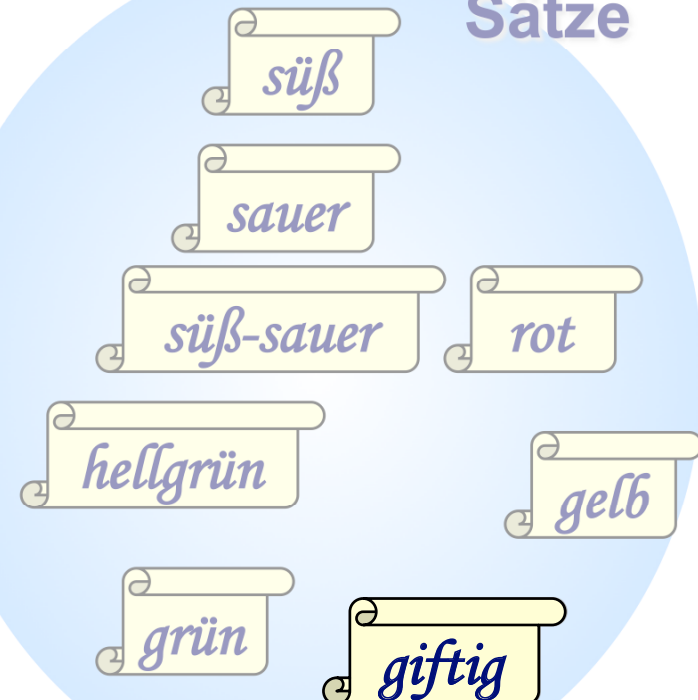
# Wie funktioniert Logik? - Modelltheorie

Sätze, für die **keine** Interpretation ein Modell ist, heißen *widersprüchlich* oder *unerfüllbar* (auch: *kontradiktorisch*).

## Interpretationen

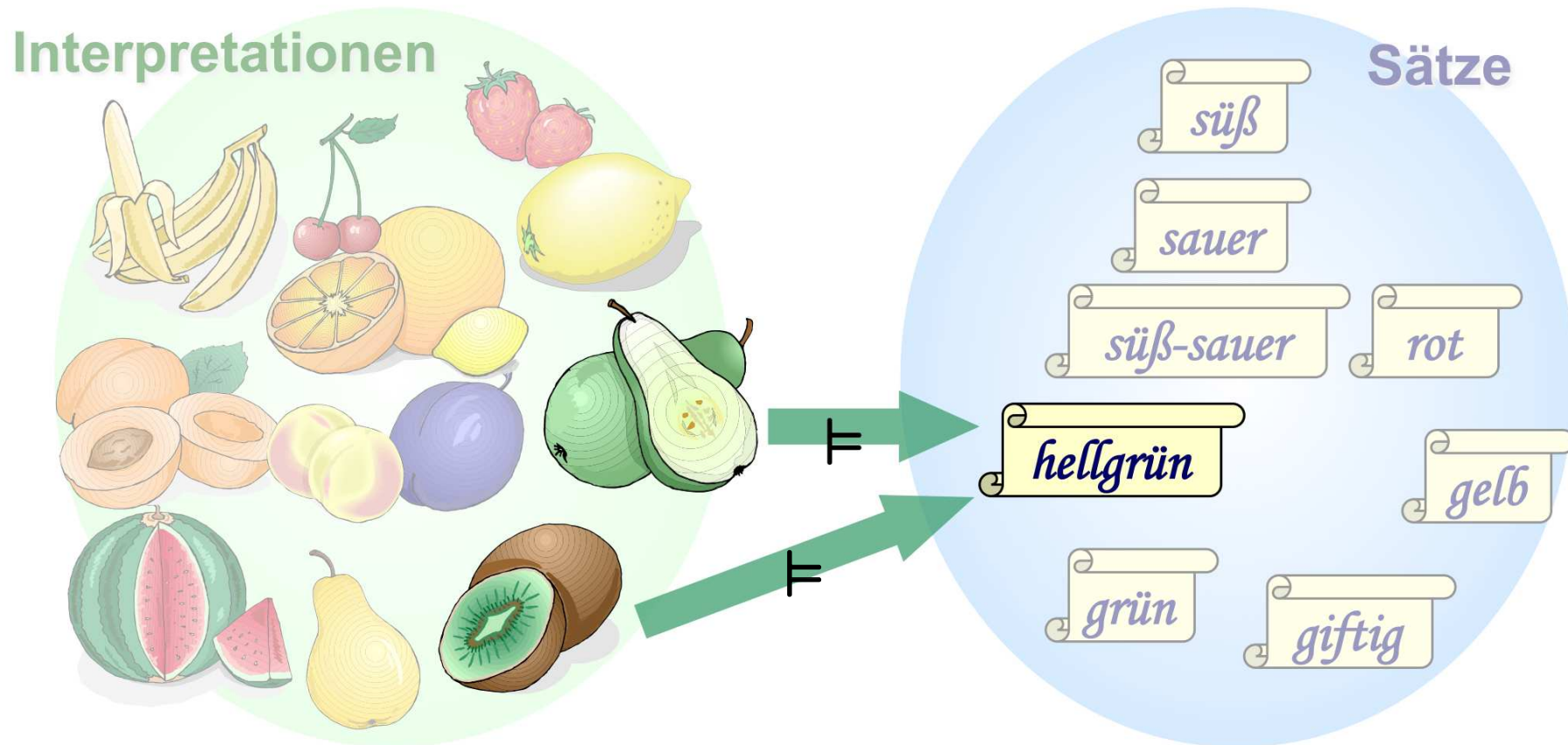


## Sätze



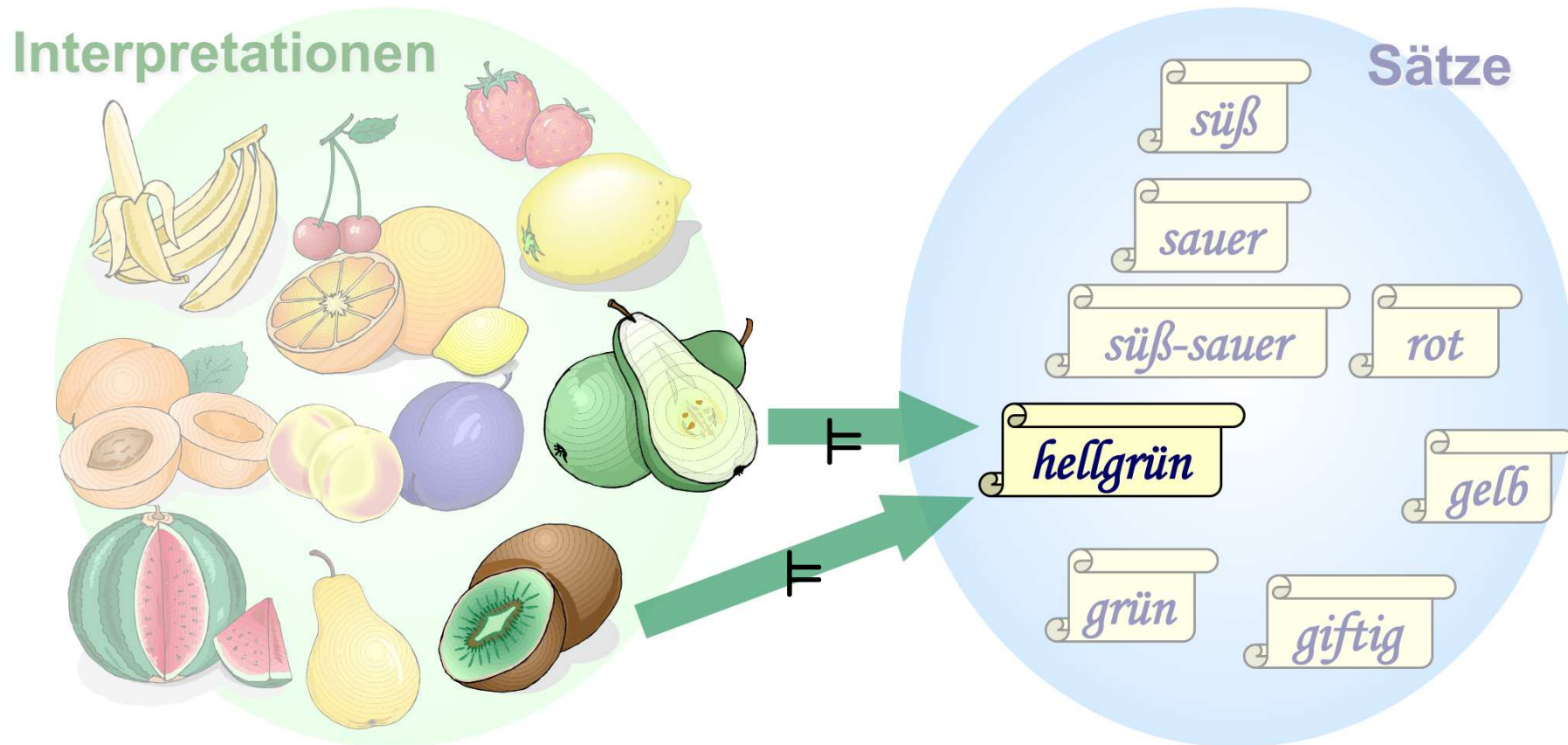
# Wie funktioniert Logik? - Modelltheorie

Eine Sätze, die (mindestens ein) Modell haben, heißen *erfüllbar* (auch: *kontingent*).



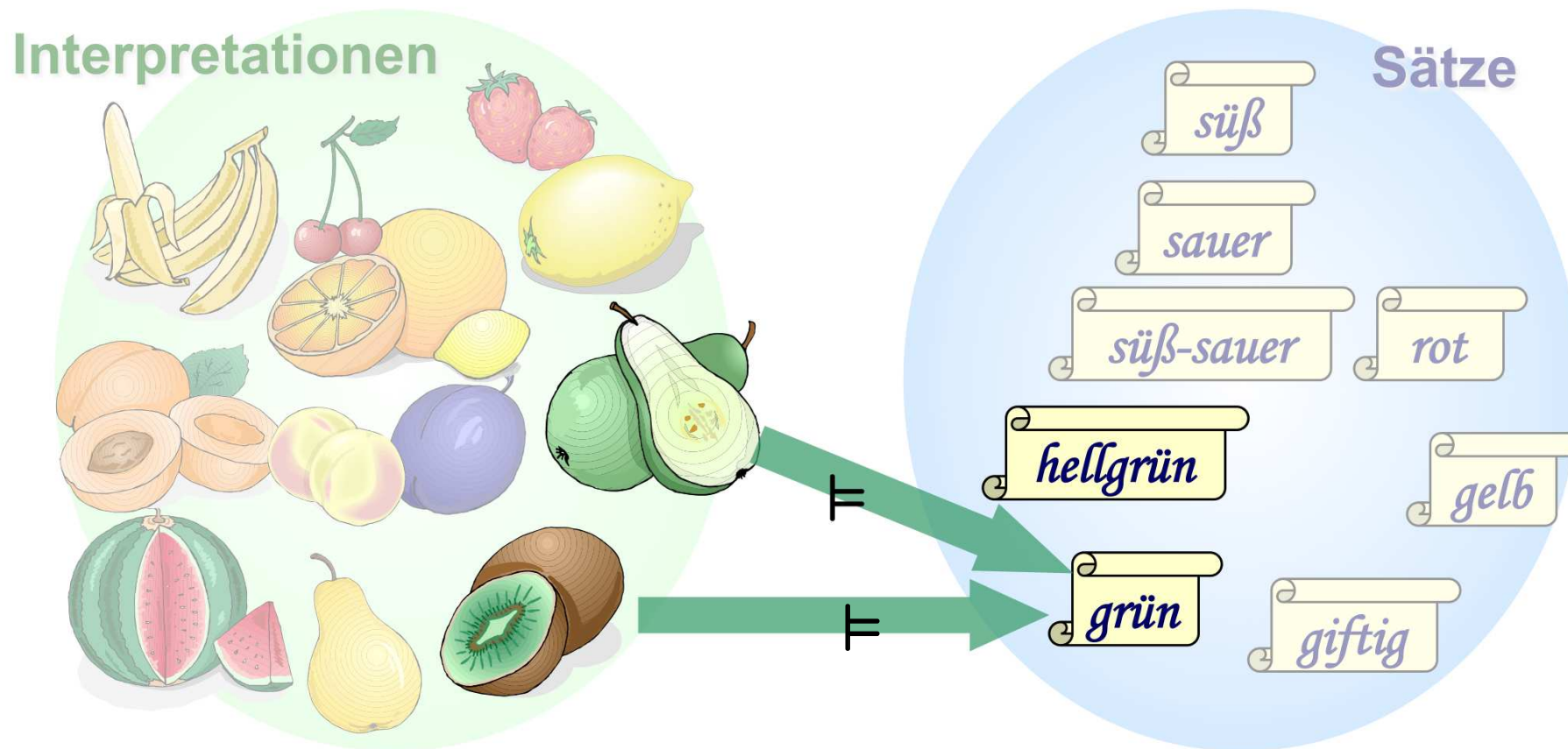
# Wie funktioniert Logik? - Modelltheorie

Eine Möglichkeit, die **Schlussfolgerungsrelation** zu definieren besteht über **Interpretationen** bzw. **Modelle**.



# Wie funktioniert Logik? - Modelltheorie

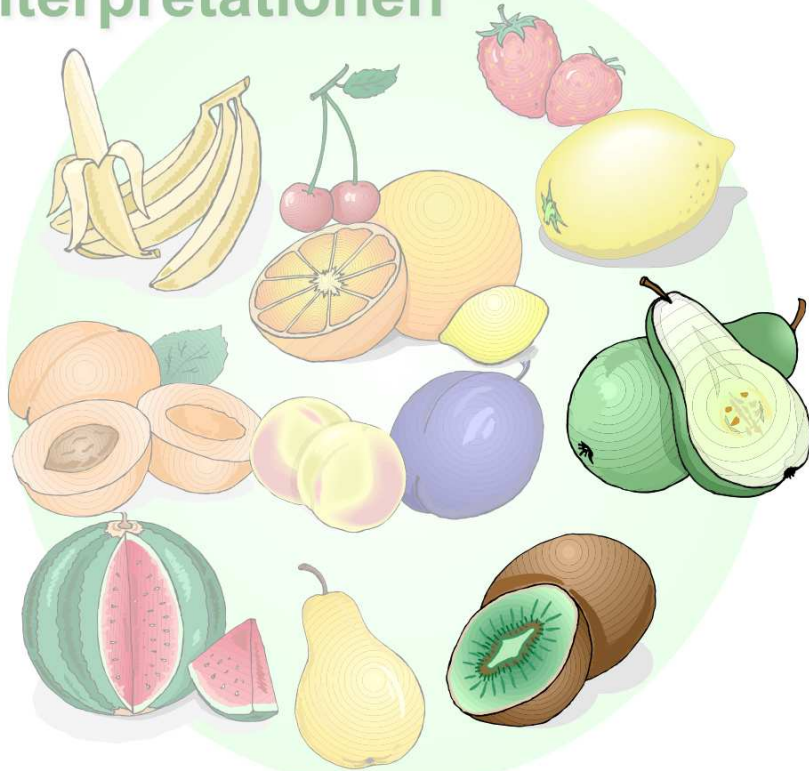
Eine Möglichkeit, die **Schlussfolgerungsrelation** zu definieren besteht über **Interpretationen** bzw. **Modelle**.



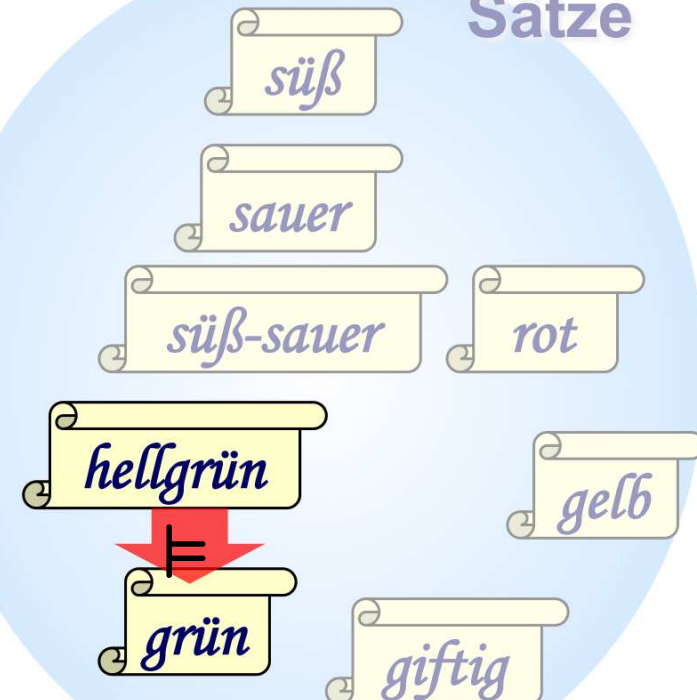
# Wie funktioniert Logik? - Modelltheorie

Eine Möglichkeit, die **Schlussfolgerungsrelation** zu definieren besteht über **Interpretationen** bzw. **Modelle**.

## Interpretationen



## Sätze



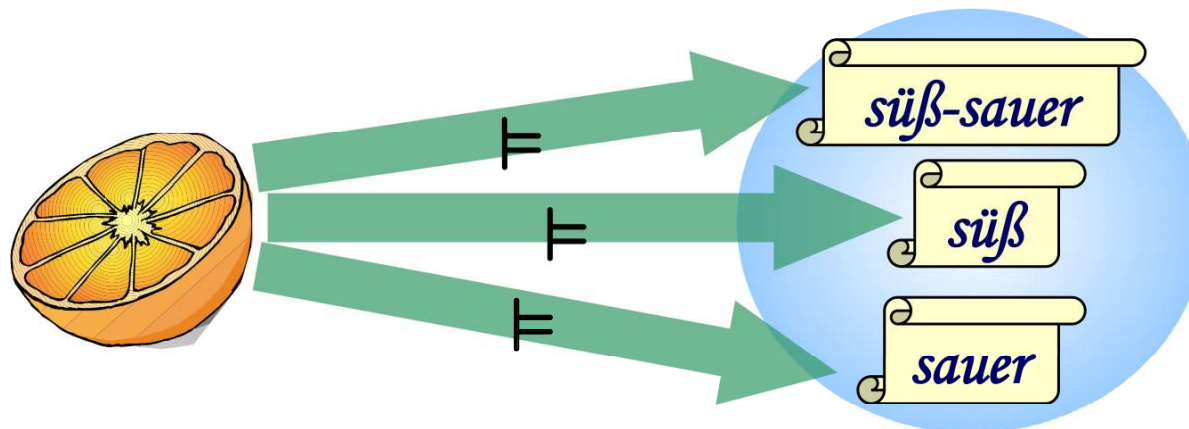


## Wie funktioniert Logik? Semantik entlang der Syntax

Häufiges Prinzip bei Definition von Interpretationen:

- Interpretation von Grundelementen wird festgelegt
- Interpretation von zusammengesetzten (konstruierten) Sätzen wird auf die Interpretation der Teile zurückgeführt, z.B.:

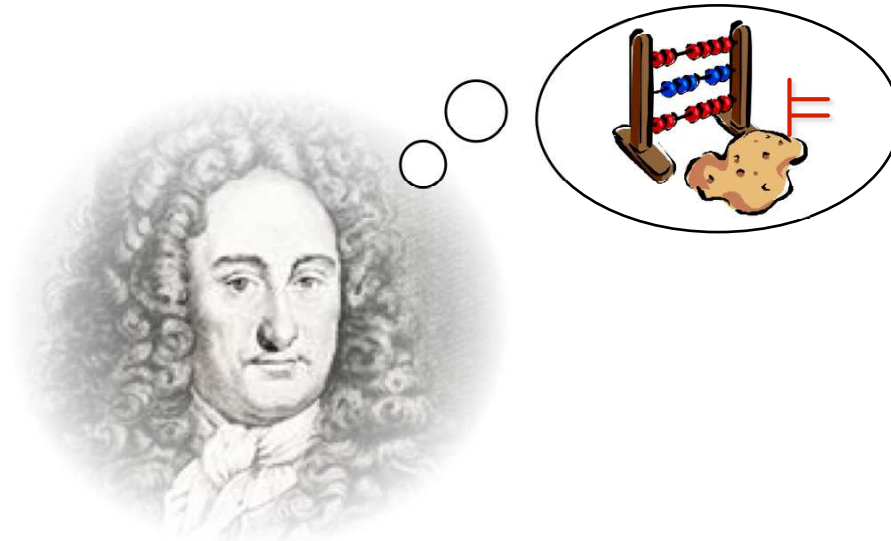
**Semantik-Regel:** „Die Modelle von  $\varphi$ - $\psi$  sind genau die Interpretationen, die Modelle sowohl von  $\varphi$  als auch von  $\psi$  sind.“



# Beweistheorie

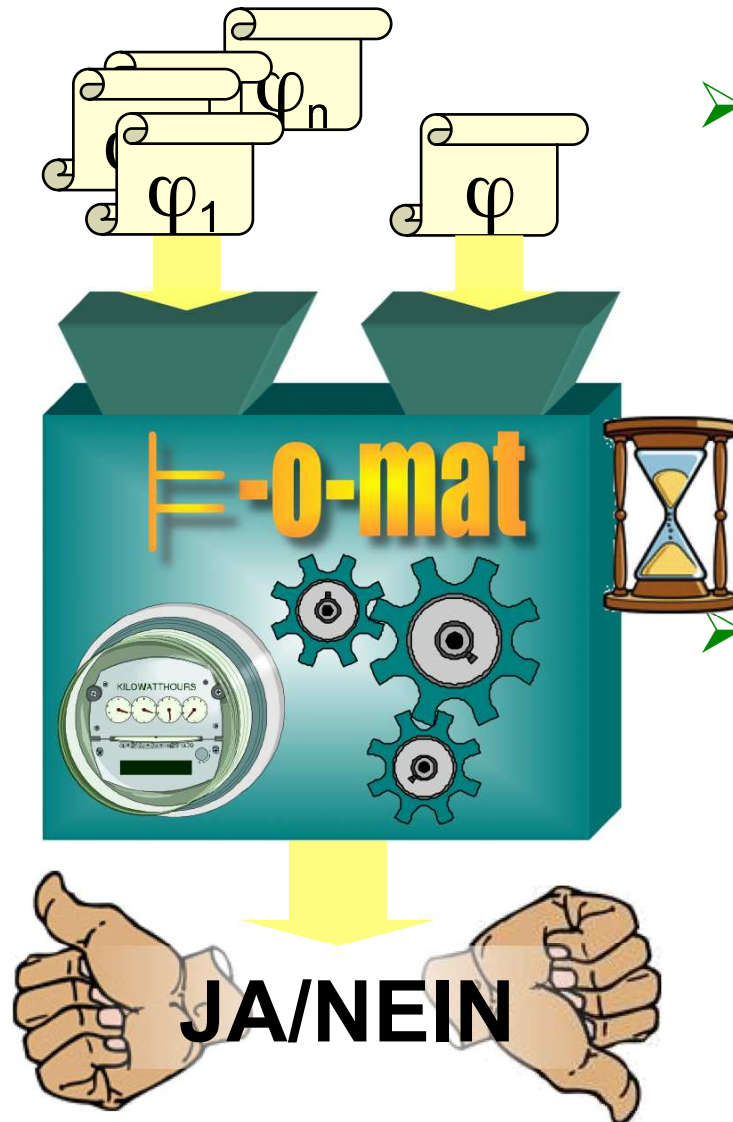
---

- Zurück zu Leibniz:  
Rechenmaschine  
für Logik



- Aber: Möglichkeit, direkt mit allen möglichen Interpretationen zu arbeiten, oft eingeschränkt
- Daher: Versuch, Schlussfolgerungsrelation durch rein syntaktische Verfahren zu beschreiben/berechnen

# Entscheidungsverfahren/Entscheidbarkeit

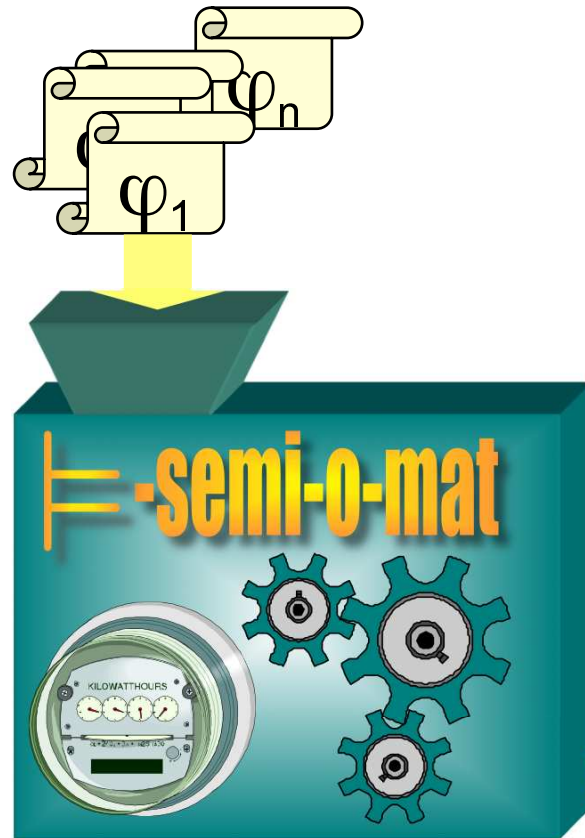


➤ Entscheidungsalgorithmus:

- ⇒ input: Menge  $\{\varphi_1, \dots, \varphi_n\}$  von Sätzen und Satz  $\varphi$
- ⇒ terminiert nach endlicher Zeit
- ⇒ output:
  - ✧ „Ja“, falls  $\{\varphi_1, \dots, \varphi_n\} \models \varphi$
  - ✧ „Nein“ sonst

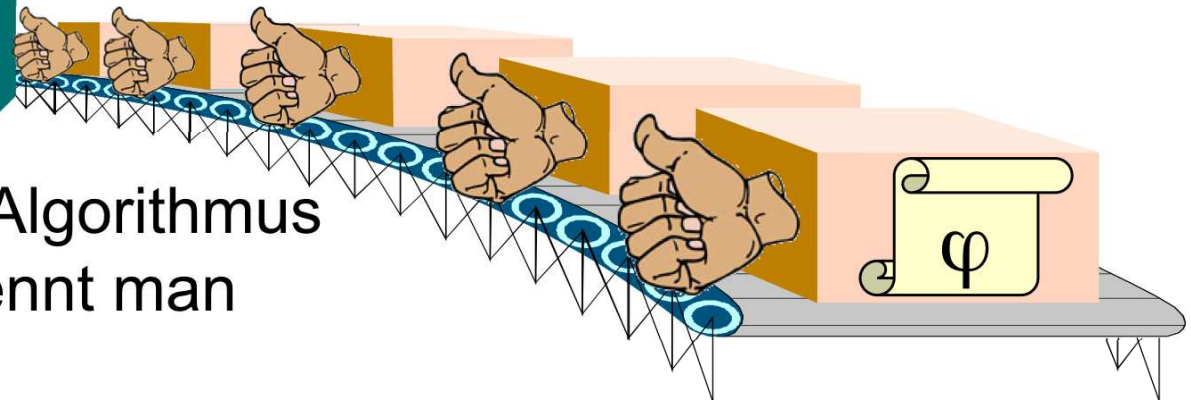
➤ Gibt es einen solchen Algorithmus für eine Logik, dann nennt man sie *entscheidbar*.

# Aufzählungsverfahren/Semientscheidbarkeit



- Aufzählungsverfahren:
  - ⇒ input: Sätze  $\{\varphi_1, \dots, \varphi_n\}$
  - ⇒ output: Sätze  $\varphi$ , für die gilt  $\{\varphi_1, \dots, \varphi_n\} \models \varphi$
  - ⇒ jeder solche Satz wird (irgendwann) ausgegeben

- Gibt es einen solchen Algorithmus für eine Logik, dann nennt man sie *semientscheidbar*.

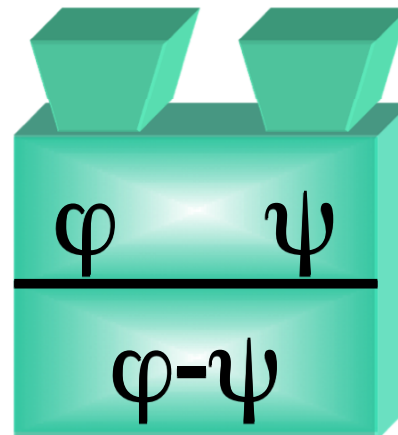


# Deduktionskalkül

---

- kann gesehen werden als spezielle Form eines Aufzählungsverfahrens
- besteht aus *Ableitungsregeln*, z.B.:

$\{\varphi, \psi, \omega, \dots\dots\}$



# Deduktionskalkül

---

Ein Satz  $\varphi$  ist aus einer Menge  $\Phi$  von Sätzen *ableitbar* (geschrieben:  $\Phi \vdash \varphi$ ), wenn sich  $\varphi$  durch wiederholtes Anwenden der Ableitungsregeln eines Deduktionskalküls aus  $\Phi$  „erzeugen“ lässt.

Deduktionskalkül ist *korrekt* (engl. *sound*), wenn aus  $\Phi \vdash \varphi$  immer  $\Phi \models \varphi$  folgt, d.h. alle ableitbaren Schlüsse auch wirklich logisch folgen.

Deduktionskalkül ist *vollständig* (engl. *complete*), wenn aus  $\Phi \models \varphi$  immer  $\Phi \vdash \varphi$  folgt, d.h. alle logischen Konsequenzen auch abgeleitet werden können.

In einem korrekten und vollständigen Deduktionskalkül gilt:

$$\models = \vdash$$

und man kann es als Aufzählungsverfahren verwenden.  
 Achtung! Es gibt Logiken, für die nachweislich kein solches Deduktionskalkül existiert (Gödel 1931).




## Weitere interessante Eigenschaften von Logiken:

---

- Monotonie
- Kompaktheit
- Algorithmische Komplexität für Entscheidungsverfahren
- ...und jede Menge anderes...

# Aussagenlogik

---

- auch: *propositionale* Logik  
*boolesche* Logik
  - schon bei den Stoikern voll  
ausgearbeitete Junktorenlogik
  - George Boole (1815 – 1864)  
„*An Investigation of the Laws of Thought*“ (1854)
- 
 A portrait of George Boole, a man with dark hair, wearing a dark suit and a white shirt with a dark bow tie. He is looking slightly to the right of the viewer.
- syntaktische Grundelemente:  
atomare Sätze / Propositionen / Aussagen  
( $p, q, \dots, p_1, p_2, \dots$ )
  - Können als natürlichsprachliche Aussagen gedacht  
werden: „Es regnet.“ ...



# Aussagenlogik – Syntax

---

- Erzeugungsregeln für Sätze:
  - ⇒ alle atomaren Propositionen sind Sätze ( $p, q, \dots$ )
  - ⇒ ist  $\varphi$  ein Satz, dann auch  $\neg\varphi$
  - ⇒ sind  $\varphi$  und  $\psi$  Sätze, dann auch  $(\varphi \wedge \psi)$ ,  $(\varphi \vee \psi)$ ,  $(\varphi \rightarrow \psi)$  und  $(\varphi \leftrightarrow \psi)$
- Klammern können ggf. weggelassen werden; Präzedenzen (bei uns):  $\neg$  vor  $\wedge, \vee$  vor  $\rightarrow, \leftrightarrow$ .
- Zusätzliche Klammern machen es trotzdem oft lesbarer...

# Aussagenlogik – Syntax

<i>Junktor</i>	<i>Name</i>	<i>Intuitive Bedeutung</i>
$\neg$	Negation	„nicht“
$\wedge$	Konjunktion	„und“
$\vee$	Disjunktion	„oder“
$\rightarrow$	Implikation	„wenn – dann“
$\leftrightarrow$	Äquivalenz	„genau dann, wenn“

## *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.

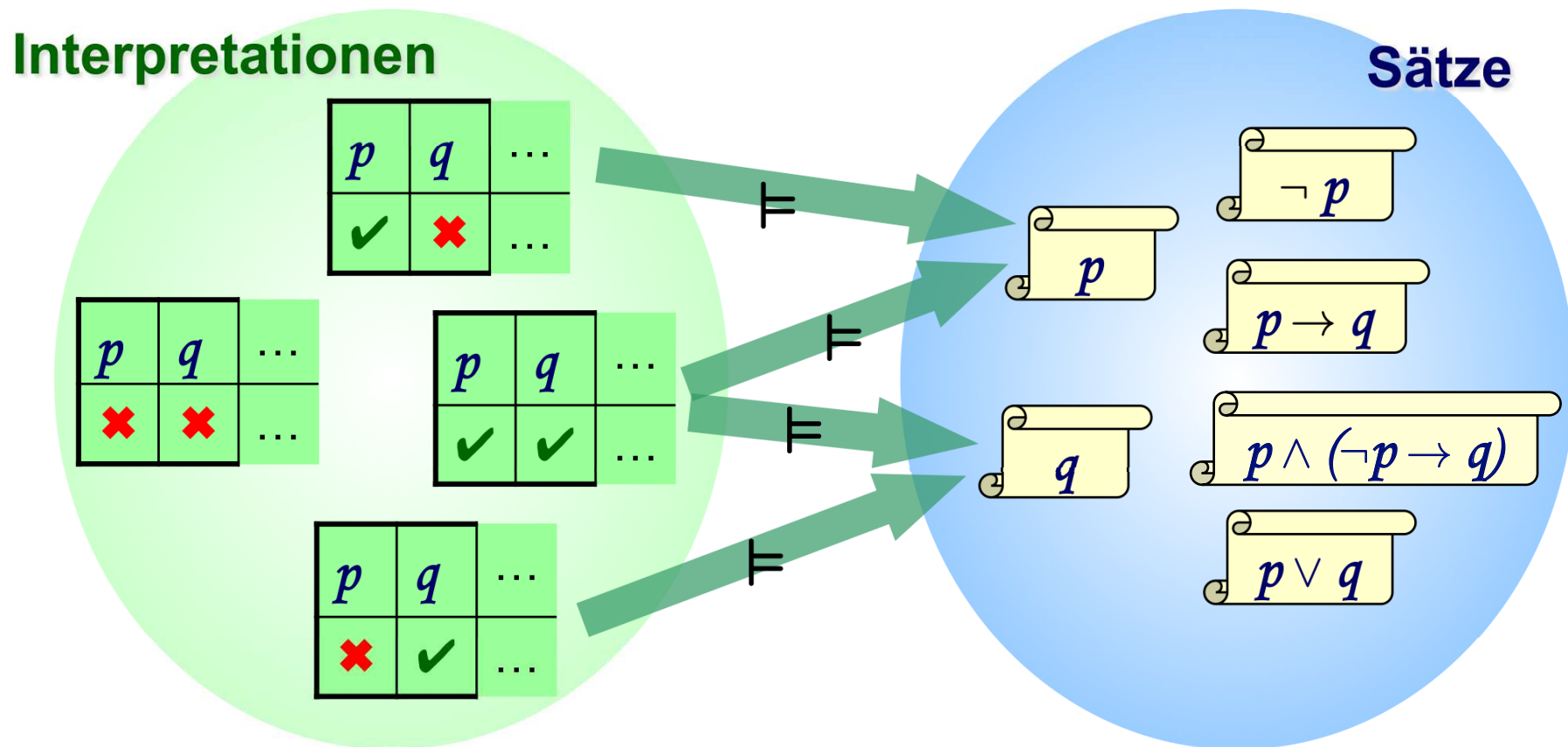
$r \rightarrow n$

Wenn es regnet, und die Straße nicht nass wird,  
dann ist die Sonne grün.

$(r \wedge \neg n) \rightarrow g$

# Aussagenlogik – Modelltheoretische Semantik

- Was sind die Modelle der Aussagenlogik?



# Aussagenlogik – Modelltheoretische Semantik

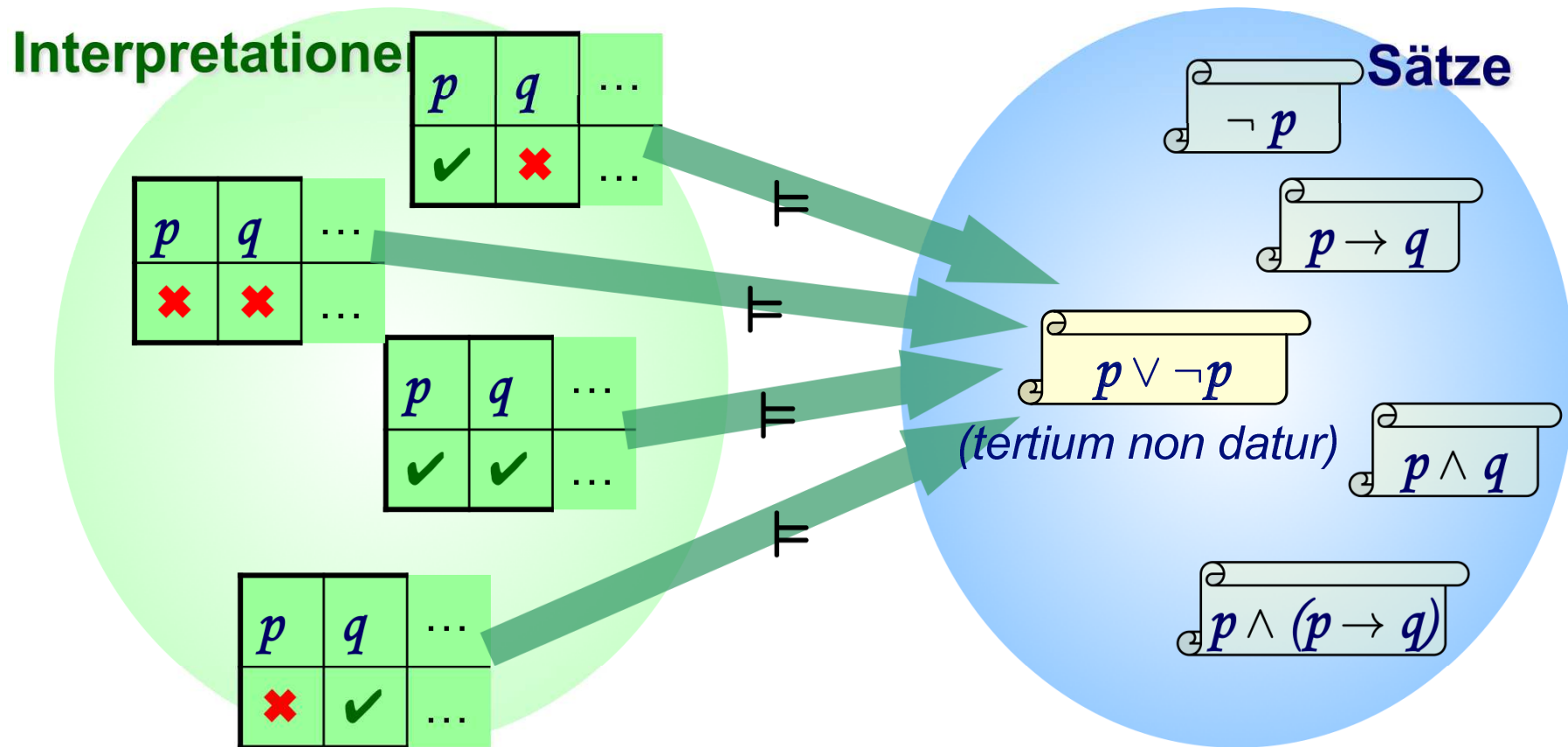
- Formal: Interpretationen  $I$  sind Abbildungen von der Menge der atomaren Propositionen in die Menge {wahr, falsch}, d.h. jeder dieser Propositionen  $p$  wird ein Wahrheitswert  $WW_I(p)$  zugeordnet.
- Daraus bestimmt man Modelle für zusammengesetzte Sätze über

## Semantik-Regeln

- ⇒  $I$  Modell von  $\neg\varphi$  genau dann, wenn  $I$  **kein** Modell von  $\varphi$
- ⇒  $I$  Modell von  $(\varphi \wedge \psi)$  genau dann, wenn  $I$  Modell von  $\varphi$  **und** von  $\psi$
- ⇒  $I$  Modell von  $(\varphi \vee \psi)$  genau dann, wenn  $I$  Modell von  $\varphi$  **oder** von  $\psi$
- ⇒  $I$  Modell von  $(\varphi \rightarrow \psi)$  genau dann, wenn  $I$  **kein** Modell von  $\varphi$  **oder**  $I$  Modell von  $\psi$
- ⇒  $I$  Modell von  $(\varphi \leftrightarrow \psi)$  genau dann, wenn  $I$  Modell für **jeden oder keinen** der beiden Sätze ist.

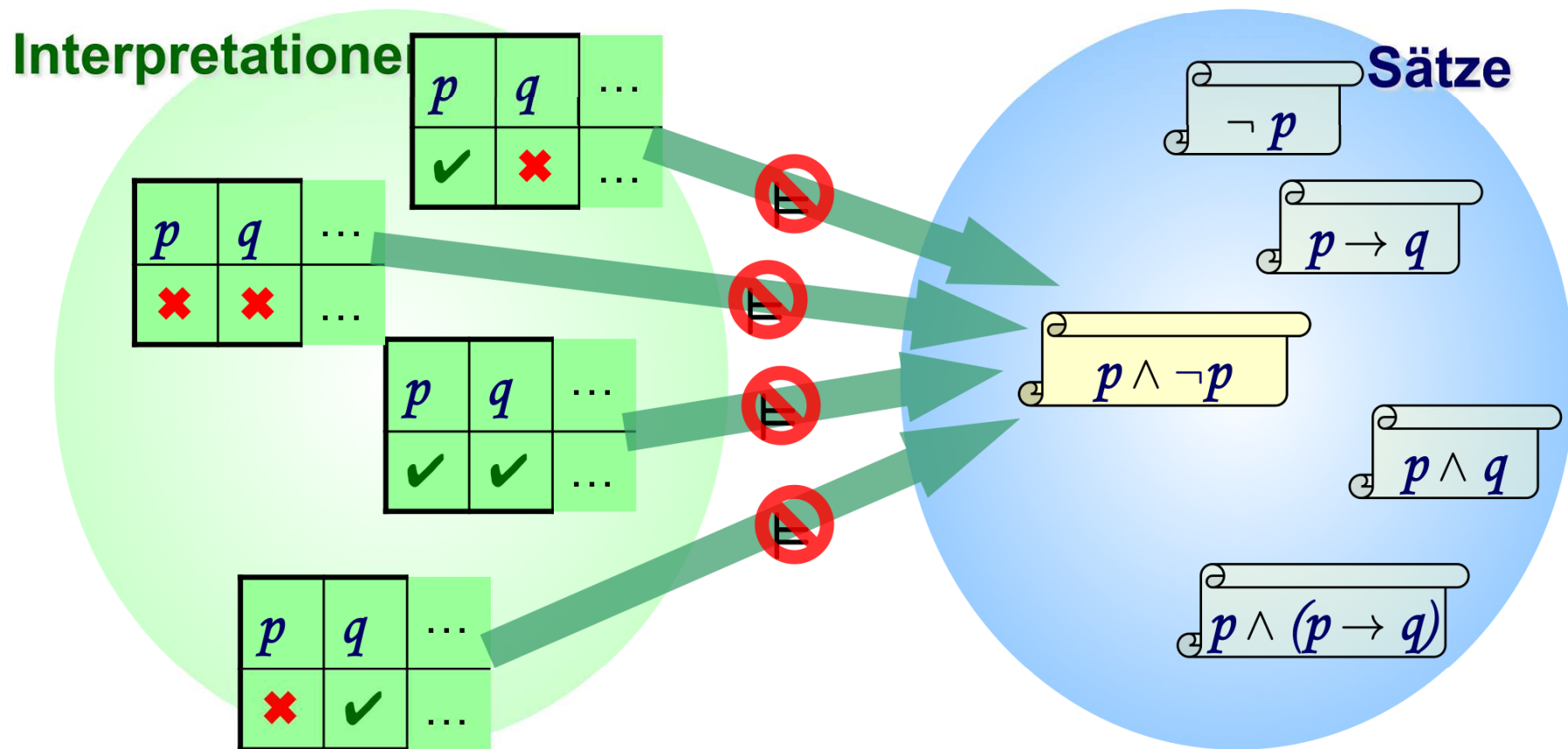
# Aussagenlogik – Modelltheoretische Semantik

- Beispiel für Tautologie in der Aussagenlogik.



# Aussagenlogik – Modelltheoretische Semantik

- Beispiel für Kontradiktion in der Aussagenlogik.



# Aussagenlogik – einige logische Äquivalenzen

---

$$\varphi \wedge \psi \equiv \psi \wedge \varphi$$

$$\varphi \vee \psi \equiv \psi \vee \varphi$$

$$\varphi \wedge (\psi \wedge \omega) \equiv (\varphi \wedge \psi) \wedge \omega$$

$$\varphi \vee (\psi \vee \omega) \equiv (\varphi \vee \psi) \vee \omega$$

$$\varphi \wedge \varphi \equiv \varphi$$

$$\varphi \vee \varphi \equiv \varphi$$

$$\varphi \wedge (\psi \vee \varphi) \equiv \varphi$$

$$\varphi \vee (\psi \wedge \varphi) \equiv \varphi$$

$$\varphi \rightarrow \psi \equiv \neg \varphi \vee \psi$$

$$\varphi \leftrightarrow \psi \equiv (\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$$

$$\neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi$$

$$\neg(\varphi \vee \psi) \equiv \neg\varphi \wedge \neg\psi$$

$$\neg\neg\varphi \equiv \varphi$$

$$\varphi \vee (\psi \wedge \omega) \equiv (\varphi \vee \psi) \wedge (\varphi \vee \omega)$$

$$\varphi \wedge (\psi \vee \omega) \equiv (\varphi \wedge \psi) \vee (\varphi \wedge \omega)$$

# Aussagenlogik – Normalformen & vollständige Junktoren

---

aus diesen Äquivalenzen folgt:

- zu jeder Formel gibt es eine logisch äquivalente Formel, die nur die Junktoren  $\wedge$  und  $\neg$  enthält.
- zu jeder Formel gibt es eine Formel in konjunktiver Normalform, d.h.
  - ⇒ nur einfache Negation direkt vor atomaren Propositionen (sog. Literale)
  - ⇒ Formel ist Konjunktion von Disjunktionen von Literalen
  - ⇒ Bsp.:  $(p \vee \neg q \vee r \vee \neg s) \wedge (\neg p \vee q \vee s) \wedge (q \vee \neg r \vee s)$



# Aussagenlogik – Entscheidungsalgorithmus

---

- Aussagenlogik ist entscheidbar
- nützliche Eigenschaft dabei:  
 $\{\varphi_1, \dots, \varphi_n\} \models \varphi$  gilt genau dann, wenn  $(\varphi_1 \wedge \dots \wedge \varphi_n) \rightarrow \varphi$   
eine Tautologie ist
- Entscheidung, ob Satz Tautologie ist, über  
Wahrheitstabelle
- im Prinzip: Überprüfung aller Interpretationen (nur die  
Wahrheitswerte der vorkommenden atomaren  
Propositionen fallen ins Gewicht)

# Aussagenlogik – Entscheidungsalgorithmus



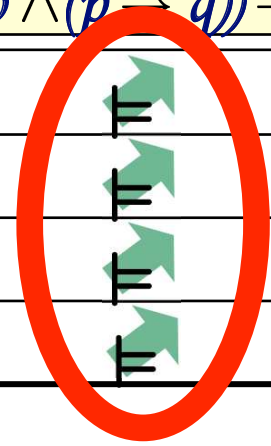
Modus Ponens:

$$\{ \underbrace{p}, \underbrace{p \rightarrow q} \} \models \underbrace{q}$$

$$\models \underbrace{(p \wedge (p \rightarrow q)) \rightarrow q}$$



$p$	$q$	...	$p \rightarrow q$	$p \wedge (p \rightarrow q)$	$(p \wedge (p \rightarrow q)) \rightarrow q$
×	×	...	⇒	⊘	⇒
×	✓	...	⇒	⊘	⇒
✓	×	...	⊘	⊘	⇒
✓	✓	...	⇒	⇒	⇒



## was hier noch kommen muss:

---

- Deduktionskalkül?
- Tableau?
- Resolution?

# Überleitung zur Prädikatenlogik

---

# Inhalt

---

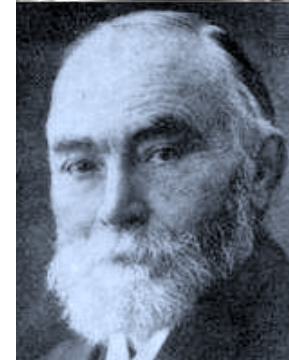
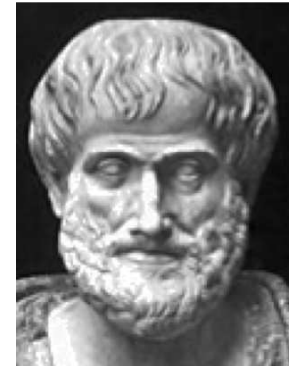
- Was ist Logik?  
*Eine etwas andere Einführung*
- Aussagenlogik  
*Logik auf der Ebene der Propositionen*
- Prädikatenlogik erster Stufe  
*Logik auf der Ebene der Objekte*

# Prädikatenlogik erster Stufe

---

auch: *first order logic* (FOL)

- erste Ansätze bei Aristoteles  
(Syllogismen, s. *Analytica Posteriora*)
- J. G. Frege (1848 – 1925):  
*Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens* (1879)
- C. S. Peirce (1839 – 1914)  
(Einführung der heute gebräuchlichen Notation)



# Prädikatenlogik erster Stufe – Syntax

---

- syntaktische Grundelemente:  
Prädikate, Eigennamen (und Variablen)

## was hier noch kommen muss:

---

- Entscheidungsalgorithmus (wahrheitstabellen)
- Deduktionskalkül