Horst Lohnstein
Logik in der Linguistik*

Zur Theorie sprachlicher Bedeutung

1 Grundlegende Annahmen

Das Sprachsystem des Menschen besitzt die Fähigkeit, auf der Basis rekursiv organisierter Verkettungen im Prinzip unendlich viele diskrete sprachliche Objekte zu bilden. Diese besitzen sowohl formale wie auch inhaltliche Charakteristika, die in komplexen Strukturkomponenten organisiert sind. Die Bildungsprinzipien der möglichen Strukturen, ihre grammatische Basis und die jeweils erzeugten Laut- und Bedeutungsstrukturen konstituieren den Gegenstandsbereich moderner linguistischer Forschung.

Dabei stellen die Lexeme -- etwas vereinfacht die ‚Wörter' -- eine endliche Menge elementarer sprachlicher Zeichen dar, die ein Lautbild mit einer Vorstellung so eng verknüpfen, dass Ferdinand de Saussure (1916) ihre Zusammengehörigkeit mit den beiden Seiten einer Münze verglichen hat.[1] Das Verhältnis von Lautbild zur Vorstellung ist in dem Sinne *arbiträr*, dass weder das Lautbild aus der Vorstellung noch die Vorstellung aus dem Lautbild systematisch ableitbar ist. Innerhalb einer Sprachgemeinschaft liegen diese Zuordnungen jedoch weitgehend in gleicher Weise vor, so dass das Verhältnis *konventionalisiert* sein muss, d. h. dass diese Laut-Bedeutungszuordnungen in sprachlich vermittelten kulturellen Kontexten erworben werden. Gemäß F. de Saussures Konzeption lässt sich das sprachliche Zeichen wie in der folgenden Grafik darstellen:

(1)

Neuere Theorien der sprachlichen Beschreibung[2] schlagen als Struktur für Lexikoneinträge Tripel ⟨PF, GF, SF⟩ mit den Komponenten PF (phonologische

* In: Klimczak, Peter / Zoglauer, Thomas (2017). *Logik in den Wissenschaften*. Münster: mentis Verlag, 243-277.

1 Vgl. Saussure, Cours de linguistique generale.

2 Z. B. Haider, Bierwisch, *Steuerung kompositionaler Strukturen* und Bierwisch, *Thematic Roles*.

Form), GF (‚Grammatical Features', dt.: grammatische Merkmale) und SF (semantische Form) vor, wobei das Θ-Raster von einer hierarchisch geordneten Sequenz von λ-Operatoren gebildet wird, das einen propositionalen Ausdruck wie etwa *geben(x, y, z)*, das vom logischen Typ t ist, in ein dreistelliges Prädikat vom logischen Typ ⟨e⟨e⟨et⟩⟩⟩ überführt. Bereits derartige Bedeutungsrepräsentationen erfordern das Instrumentarium der Prädikaten- und Typenlogik mit λ-Operator, dem wir im weiteren Verlauf begegnen werden.

(2) illustriert die Form- und die Inhaltsseite einer Lexikoneinheit (LE) -- gemäß der Konzeption von Bierwisch[3] -- am Beispiel des Verbs *geben* :

(2) $\underbrace{\underbrace{/\text{geb}/}_{\text{PF}} : \underbrace{[+\text{V}, -\text{N}]}_{\text{GF}}}_{\text{Form(LE)}} \underbrace{\underbrace{\lambda z \lambda y \lambda x}_{\Theta\text{-Raster}} \underbrace{[\text{tun}(x,e) \wedge \text{verursach}(e, \text{werd}(\text{haben}(y,z)))]}_{\text{SF}}}_{\text{Inhalt(LE)}}$

Die semantische Form (SF) des Verbs *geben* lässt sich paraphrasieren als ‚x tut etwas (e), was die Ursache dafür ist, dass es wird, dass y z hat'. In der Bedeutung von ‚geben' ist damit die Bedeutung des inchoativen (Zustandswechsel-)Prädikats *erhalten* (werden, dass y z hat) und des Zustandsprädikats *haben* (y hat z) enthalten.

Die einzelnen λ-Operatoren im Θ-Raster sind mit Bündeln grammatischer Merkmale GF assoziiert, in denen die nicht prädiktable grammatische Information derjenigen Argumente notiert ist, welche die im Θ-Raster spezifizierten Variablen in der semantischen Form SF belegen:

(3) Argumentstruktur:

$$\begin{array}{cccc} \lambda x_n & \lambda x_{n-1} & \ldots & \lambda x_0 \\ | & | & & | \\ GF_n & GF_{n-1} & \ldots & GF_0 \end{array}$$

Damit sind die wesentlichen Aspekte und Informationskomponenten in der Struktur der lexikalischen Einheiten benannt. Die Zuordnung der Form zum Inhalt einer lexikalischen Einheit variiert mit den kulturellen Ausprägungen innerhalb von Sprachgemeinschaften, während die Komponenten des Inhalts(LE) in (2) aus universell angelegten Prinzipien des konzeptuellen Systems des Menschen ableitbar sind.

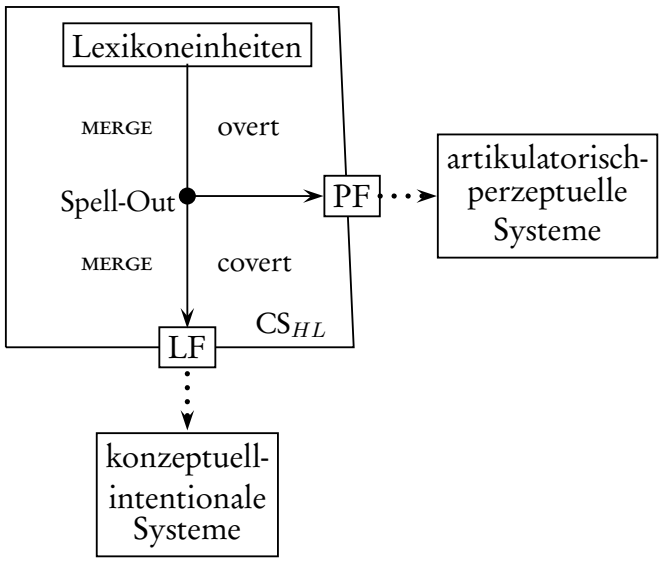
Die Anzahl der Lexikoneinheiten ist zwar groß, aber endlich. Unendlich viele diskrete (komplexe) Objekte zu bilden, wird erst durch die kombinatorischen Prinzipien der Syntax einer natürlichen Sprache erreicht. Neuere Theorien der syntaktischen Strukturbildung gehen von sehr einfachen Prinzipien aus, die durch iterative Anwendung rekursiv organisierter Operationen die syntaktische Strukturbildung festlegen. So wird im Rahmen der generativen Grammatik seit Chomsky (1995) ein Modell wie in (4) angenommen:[4]

---

[3] Vgl. Bierwisch, *Thematic Roles*.

[4] Vgl. Chomsky, *The Minimalist Program*.

(4) Syntaxmodell der generativen Grammatik:



Das System CS_{HL} (Computational System of Human Language) spezifiziert die Bedingungen, unter denen aus einer Kollektion von Lexikoneinheiten mit Hilfe der generellen Operation MERGE[5] (intern und extern) komplexe Lautstrukturen (PF = phonologische Form) und komplexe Bedeutungsstrukturen (LF = logische Form) gebildet werden, die die sprachexternen Systeme der artikulatorischen und perzeptuellen Berechnung der Lautstruktur einerseits sowie die konzeptuellen und intentionalen Systeme der Bedeutungsberechnung andererseits an das Sprachsystem stellen. Die in den Lexikoneinheiten verbundenen Informationen der Laut- und Bedeutungsstruktur werden am Punkt ‚Spell Out'[6] getrennt, wobei es möglich ist, dass hörbare (overte), aber auch nicht-hörbare (coverte) Operationen ausgeführt werden.

Diese syntaktische Subkomponente des grammatischen Systems erlaubt es, Lexikoneinheiten nach rekursiven Prinzipien so miteinander zu verbinden, dass Sätze gebildet werden können, für deren Länge keine obere Schranke existiert. Die Sequenz der Beispielsätze in (6) verdeutlicht diese Eigenschaft:

(6) i. Karl kommt.
ii. Erna hat gesagt, dass Karl kommt.
iii. Fritz glaubt, dass Erna gesagt hat, dass Karl kommt.

---

5 MERGE ist eine zweistellige Strukturbildungsoperation, die aus zwei Konstituenten K_1 und K_2 eine neue Konstituente K_3 bildet: MERGE(K_1, K_2) = K_3.

6 ‚Spell Out' ist ein Punkt, an dem bereits während der Satzderivation die phonologischen Informationen, die mit den Lexikoneinheiten verbunden sind, an die phonologische Komponente gesendet werden, so dass sie für die Berechnung der logischen Form nicht mehr verfügbar sind.

iv. ... Maria behauptet, dass Fritz glaubt, dass Erna gesagt hat, dass Karl kommt.

Eine semantische Theorie, die nur die Bedeutungen der lexikalischen Einheiten zu erfassen hätte, benötigt nicht notwendigerweise Prinzipien der Bedeutungskomposition, da der Wortschatz endlich ist und infolgedessen eine (endliche) Liste von Bedeutungen ausreichen würde, um die Semantik jeder lexikalischen Einheit anzugeben. Die prinzipielle Unbeschränktheit der syntaktischen Komplexbildung erfordert jedoch Kompositionsprinzipien, um allen syntaktisch wohlgeformten Ausdrücken die ihnen entsprechende Bedeutung zuzuweisen.

Damit ist eine zentrale Frage der modernen Semantiktheorie thematisiert: Nach welchen Prinzipien lassen sich Bedeutungen komponieren, so dass sie -- möglichst isomorph -- auf die Teilkomplexe syntaktischer Strukturen bezogen werden können. Diese generelle Eigenschaft der Kompositionalität lässt sich in dem folgenden -- auf Frege zurückgehenden -- Prinzip formulieren:[7]

(7) Freges Kompositionalistätsprinzip

Die Bedeutung eines komplexen sprachlichen Ausdrucks lässt sich aus der Bedeutung der beteiligten Teilausdrücke und der Art ihrer Zusammensetzung (Syntax) bestimmen.

Wenn Bedeutungen *kompositionell* rekonstruiert werden müssen, sind die *elementaren Bausteine* der Bedeutung und die *Regeln und Prinzipien ihrer Kombinatorik* zu bestimmen. Dies geschieht mit Hilfe *rekursiv definierter* Logiksprachen. Sie bestehen -- wie natürliche Sprachen auch -- aus:

(8) i. einem Lexikon, das die elementaren Einheiten der Sprache enthält,
ii. einer Syntax, welche die *formale Kombinatorik* von Ausdrücken (rekursiv) regelt,
iii. einer Semantik, die den einfachen und komplexen Ausdrücken (rekursiv) ihre *Bedeutung* zuweist.

Der amerikanische Mathematiker und Logiker Richard Montague sieht dabei keinen wesentlichen Unterschied in der theoretischen Untersuchung von natürlichen, menschlichen Sprachen und den formalen Sprachen der Logiker: „There is in my opinion no important theoretical difference between natural languages and the artificial languages of logicans; indeed I consider it possible to comprehend the syntax and semantics of both kinds of languages within a single natural and mathematically precise theory.“[8]

7 Siehe etwa Frege, *Der Gedanke*.
8 Montague, *Formal philosophy*, S. 222.

Der Gegenstand der Semantik natürlicher Sprachen ist die sog. *wörtliche Bedeutung* sprachlicher Ausdrücke. Darunter kann man in erster Näherung die Bedingungen verstehen, unter denen ein Deklarativsatz *wahr* ist. Dieser Betrachtungsweise liegt eine Sentenz aus Ludwig Wittgensteins ‚Tractatus' zugrunde:

(9) Tractatus Logico Philosophicus:[9]

> Die Bedeutung eines Satzes zu kennen, heißt zu wissen, was der Fall ist, damit er wahr ist. Man muss dazu nicht wissen, ob er wahr ist.

Der polnische Logiker und Mathematiker Alfred Tarski (1944) definiert den Begriff *Wahrheit* in der Metasprache mit Hilfe des Begriffs *Erfüllung* in der Objektsprache.[10] Für die Wahrheit eines objektsprachlichen Satzes müssen die ausgedrückten Bedingungen in (einem Modell von) der Welt erfüllt sein:

(10) Der Satz „Schnee ist weiß" ist wahr genau dann, wenn Schnee weiß ist.

Entsprechend ist dieser Satz wahr, wenn die Substanz, die wir mit „Schnee" bezeichnen, die Farbe hat, die wir mit „weiß" bezeichnen.

2 Prädikat-Argument-Strukturen

Lexikalische Einheiten haben i. d. R. prädikative Eigenschaften. Ein einstelliges Prädikat wie etwa *schlafen* führt zu einer Aussage, wenn es mit einem Eigennamen -- etwa *Fritz* -- verbunden wird: *Fritz schläft*, wobei wir hier von der Finitheit, die am Verb markiert ist, noch absehen. Die Situation, die mit dem Satz (11.i) zutreffend beschrieben wird, lässt sich hinsichtlich ihrer zeitlichen Struktur grafisch wie in (11.ii) darstellen:

(11) i. Fritz schläft.

ii. 

iii. schlafen(Fritz)

iv. $[\![\text{schlafen(Fritz)}]\!] = \begin{cases} \text{wahr} & \text{-- innerhalb des Intervalls} \\ \text{falsch} & \text{-- außerhalb des Intervalls} \end{cases}$

Die dem Satz (11.i) entsprechende Aussage in der prädiaktenlogischen Notation in (11.iii), also der metasprachliche Ausdruck für den Satz in (11.i), denotiert in dem angegebenen Intervall das Wahre und außerhalb dieses Intervalls das Falsche (11.iv). Dabei werden die Doppelklammern $[\![...]\!]$ als Denotatsfunktion aufgefasst, die einen Ausdruck der Sprache auf sein Denotat abbildet.

---

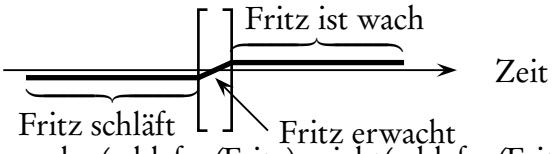
9 Wittgenstein, *Tractatus*, 4.024.

10 Vgl. Tarski, *Der Wahrheitsbegriff*.

Während *schlafen* ein Zustandsprädikat ist, ist das Verb *erwachen* ein (inchoatives) Zustandswechsel-Prädikat, das gerade dasjenige Zeitintervall bezeichnet, in dem der Zustand des Schlafens in den Zustand des Wachseins übergeht:

(12) i. Fritz erwacht.

ii.



iii. werden(schlafen(Fritz), nicht(schlafen(Fritz))), wobei das Wach-sein von Fritz als nicht(schlafen(Fritz)) charakterisiert ist.

iv. ⟦erwachen(Fritz)⟧ = ⟦werden(schlafen(Fritz), nicht(schlafen(Fritz)))⟧ =

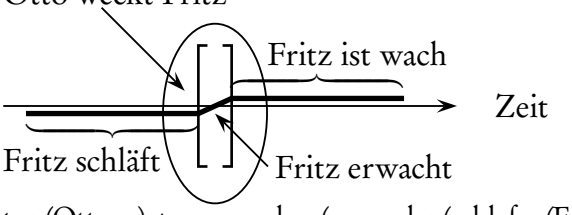
$$= \begin{cases} \text{wahr} & \text{-- innerhalb des Intervalls} \\ \text{falsch} & \text{-- außerhalb des Intervalls} \end{cases}$$

Die beiden Prädikate *schlafen* und *erwachen* sind einstellig (intransitiv), d. h. sie verbinden sich mit einem Argument, das hier durch den Eigennamen *Fritz* geliefert wird.

Das kausative Verb *wecken* ist demgegenüber zweistellig (transitiv). Die zusätzliche Bedeutungskomponente liefert die kausale Relation, die zwischen dem Erwachen von Fritz und einer Aktivität von Otto besteht:

(13) i. Otto weckt Fritz.

ii. Otto weckt Fritz



iii. tun(Otto, e) ∧ verursachen(e, werden(schlafen(Fritz), nicht(schlafen(Fritz))))

iv. ⟦wecken(Otto, Fritz)⟧ =
⟦tun(Otto, e) ∧ verursachen(e, erwachen(Fritz))⟧ =
⟦tun(Otto, e) ∧ verursachen(e, werden(schlafen(Fritz), nicht(schlafen(Fritz))))⟧ =

$$= \begin{cases} \text{wahr} & \text{-- innerhalb der Ellipse} \\ \text{falsch} & \text{-- außerhalb der Ellipse} \end{cases}$$

In der Sprache PL_1 der Prädikatenlogik lässt sich die Verbindung von mehrstelligen Prädikaten mit der entsprechenden Anzahl von Argumenten ausdrücken. Dabei basiert PL_1 auf einem Lexikon, in dem sowohl n-stellige Prädikate als auch Terme (Individuenkonstanten (Eigennamen) und -variablen (x, y, z, ...)) enthalten sind.

Die Syntax von PL_1 enthält die folgenden Regeln:

(14) Syntax von PL_1:

i. Wenn p ein n-stelliges Prädikat ist, und t_1, t_2, ..., t_n Terme sind, dann ist $p(t_1, t_2, ..., t_n)$ eine Aussage.
ii. Wenn p und q Aussagen sind, so sind auch $p \wedge q$ (Konjunktion), $p \vee q$ (Disjunktion), $p \rightarrow q$ (Konditional), $p \leftrightarrow q$ (Bikonditional) und $\neg p$ (Negation) Aussagen.
iii. Wenn $\varphi(x)$ eine Aussage ist, in der die Variable x frei vorkommt, so ist $\forall x[\varphi(x)]$ eine Aussage. ($\forall$: *Allquantor*)
iv. Wenn $\varphi(x)$ eine Aussage ist, in der die Variable x frei vorkommt, so ist $\exists x[\varphi(x)]$ eine Aussage. ($\exists$: *Existenzquantor*)

Regel (14.i) bestimmt, wie Prädikate mit Argumenten verbunden werden. Regel (14.ii) steuert gemäß der Syntax der Aussagenlogik die Verknüpfung von Aussagen, und die Regeln (14.iii) und (14.iv) legen die Struktur von quantifizierten Aussagen fest. Dabei nennt man den Bereich $[\varphi(x)]$ den *Skopus* des jeweiligen Quantors.

In der Sprache PL_1 sind nur diejenigen Ausdrücke wohlgeformt, die gemäß der Regeln (14.i) bis (14.iv) gebildet sind, wobei bisher noch nichts über die Bedeutung der jeweiligen Ausdrücke gesagt wurde. Die Syntax von PL_1 ist rekursiv definiert. Außer bei Regel (14.i) dienen stets Aussagen als Eingabe für die Regel, und es ergibt sich durch die Anwendung der Regel wieder eine Aussage. Aussagen können daher beliebig komplex werden. Um sicherzustellen, dass jede syntaktisch erzeugbare Aussage auch semantisch interpretiert werden kann, benötigt man zu jeder syntaktischen Regel eine semantische Regel, so dass immer dann, wenn eine syntaktische Regel einen Ausdruck erweitert, die semantische Regel die Bedeutung dieses Ausdrucks bestimmen kann. Diese Verfahrensweise entspricht aber genau Freges Kompositionalitätsprinzip, denn die Bedeutung eines komplexen sprachlichen Ausdrucks lässt sich auf diese Art stets aus der Bedeutung der beteiligten Teilausdrücke und der Art ihrer Zusammensetzung bestimmen.

Will man die Bedeutung eines sprachlichen Ausdrucks angeben, so benötigt man eine Vorstellung von (einem Modell von) der Welt, relativ zu der dieser Ausdruck interpretiert werden soll. Dies entspricht Wittgensteins Tractatus-Sentenz in (9). Entsprechendes gilt für sprachliche Ausdrücke, die ‚kleiner' sind als der Satz.

Die Modelltheorie liefert für die Interpretation dieser Ausdrücke das geeignete theoretische Instrumentarium. Ein *Modell* $M = \langle D, F \rangle$ besteht aus zwei Komponenten D und F, wobei D die Diskursdomäne ist, die aus einer nicht-leeren Menge von Individuen besteht, und F die Denotatsfunktion, die jeder nicht-logischen Konstanten von PL_1 ein Denotat zuweist. Da Variablen kein Denotat besitzen, wird ihnen mittels der sog. Variablenbelegungsfunktion g ein Denotat zugewiesen:[11]

[11] Zu Details siehe Lohnstein, *Formale Semantik*, S. 99ff.

(15) $g: \begin{bmatrix} x & \rightarrow & \text{Fritz} \\ y & \rightarrow & \text{Maria} \\ z & \rightarrow & \text{Otto} \end{bmatrix}$

Dann lassen sich die Denotate der elementaren (Lexikon-)Ausdrücke von PL_1 in der folgenden Weise angeben:

(16) i. Terme
  a. zu jeder Individuenkonstante ein Element aus D,
  b. zu jeder Individuenvariable x den Wert der Variablenbelegungsfunktion g(x).

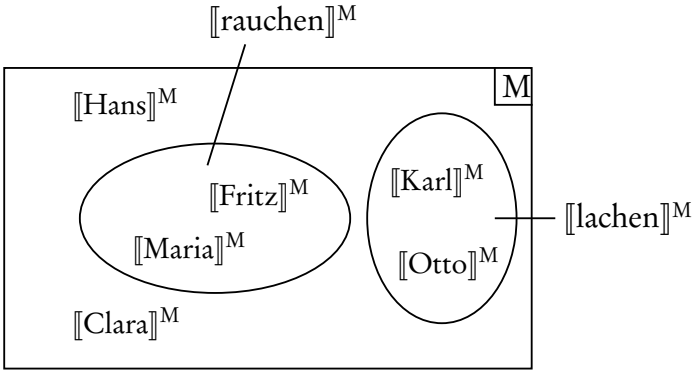
ii. Prädikate
  a. zu jedem einstelligen Prädikat eine Teilmenge von D,
  b. zu jedem zweistelligen Prädikat eine Teilmenge von D x D -- dem cartesischen Produkt über D,
  c. zu jedem n-stelligen Prädikat eine Teilmenge von $\underbrace{D \times D \times \ldots \times D}_{\text{n-mal}}$.

Die Denotatsfunktion F wird auch mit Hilfe der semantischen Doppelklammer $[\![...]\!]$ geschrieben, so dass sich die Denotate notieren lassen wie in (17):

(17) i. Wenn α eine Variable enthält, dann ist $[\![\alpha]\!]^{M,g}$ das Denotat von α relativ zum Modell M und der Variablenbelegung g.
ii. Wenn α keine Variable enthält, dann ist $[\![\alpha]\!]^{M}$ das Denotat von α relativ zum Modell M.

Die folgenden Beispiele illustrieren die Konzeption:

(18) i. Das Denotat des PL_1-Terms ‚Fritz' ist in M das Individuum $[\![\text{Fritz}]\!]^M$.
ii. Das Denotat des 1-stelligen PL_1-Prädikats ‚lachen' ist in M die Menge der Individuen, die in M lachen:
$[\![\text{lachen}]\!]^M = \{x/\ x \text{ lacht in M}\} = \{[\![\text{Karl}]\!]^M, [\![\text{Otto}]\!]^M\}$.



Nachdem die modelltheoretischen Konzepte dargelegt sind, lassen sich die semantischen Regeln von PL_1 formulieren. Dabei ist es wichtig, dass jeder syntaktischen

Regel genau eine semantische Regel entspricht, so dass immer dann, wenn die Syntax einen Ausdruck erweitert, die Semantik die Bedeutung des erweiterten Ausdrucks angeben kann.

(19) Semantik von PL_1:

i. $[\![p(t_1, t_2, ..., t_n)]\!]^{M,g}$ = w, gdw. $\langle [\![t_1]\!]^{M,g}, [\![t_2]\!]^{M,g}, ..., [\![t_n]\!]^{M,g} \rangle \in [\![p]\!]^{M,g}$

ii.

p	q	p ∧ q
w	w	w
w	f	f
f	w	f
f	f	f

p	q	p ∨ q
w	w	w
w	f	w
f	w	w
f	f	f

p	q	p → q
w	w	w
w	f	f
f	w	w
f	f	w

p	q	p ↔ q
w	w	w
w	f	f
f	w	f
f	f	w

p	¬p
w	f
f	w

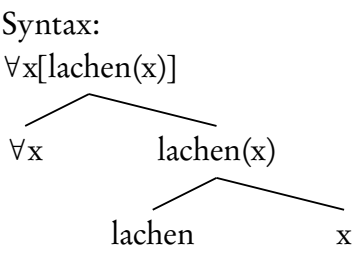
iii. $[\![\forall x[\varphi(x)]]\!]^{M,g}$ = w gdw. für *alle* Individuen a ∈ D gilt: $[\![\varphi(a)]\!]^{M,g}$ = w.

iv. $[\![\exists x[\varphi(x)]]\!]^{M,g}$ = w gdw. für *mind. ein* Individuum a ∈ D gilt: $[\![\varphi(a)]\!]^{M,g}$ = w.

Die semantische Regel (19.i) gibt die Bedingung an, unter der die Verbindung von n Termen mit einem n-stelligen Prädikat wahr ist. Die in (19.ii) gezeigten Wahrheitstabellen legen die Wahrheit konjunktiver Aussagen im Sinne der Aussagenlogik fest. Eine allquantifizierte Aussage ist gemäß (19.iii) genau dann wahr, wenn der Skopus für alle Individuen in der Diskursdomäne wahr ist, und die semantische Regel (19.iv) legt fest, dass eine existenzquantifizierte Aussage genau dann wahr ist, wenn es mindestens ein Individuum in der Diskursdomäne gibt, für das der Skopus wahr ist. Die folgenden Beispiele illustrieren die syntaktische Strukturbildung und die entsprechende semantische Interpretation:

(20) Alle lachen.

Syntax:

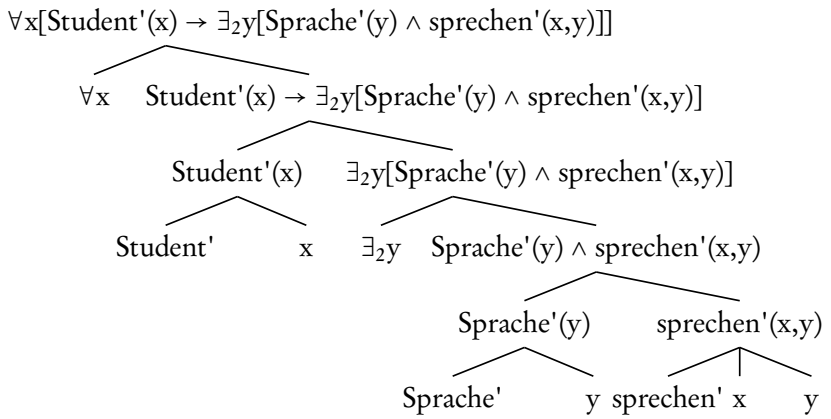


Semantik:
$[\![\forall x[\text{lachen}(x)]]\!]^{M,g}$ = wahr gdw.
für alle a ∈ D gilt: $[\![\text{lachen}(a)]\!]^{M,g}$ = wahr,
d. h. wenn alle Individuen a in D lachen.

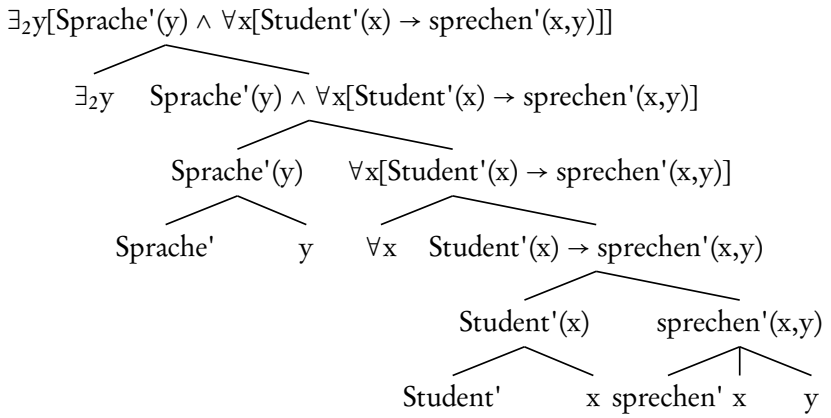
Wenn mehrere Quantorenausdrücke in einem Satz auftreten, können sog. Skopusambiguitäten auftreten. Bezogen auf das Grammatikmodell in (4) können einer phonetischen Form (PF) systematisch zwei logische Formen (LF) zugeordnet werden:

(22) PF: weil jeder Student zwei Sprachen spricht.

i. LF_1: Zu jedem Studenten x gibt es zwei Sprachen y, so dass gilt: x spricht y.



ii. LF_2: Es gibt zwei Sprachen y, so dass für jeden Studenten x gilt: x spricht y.

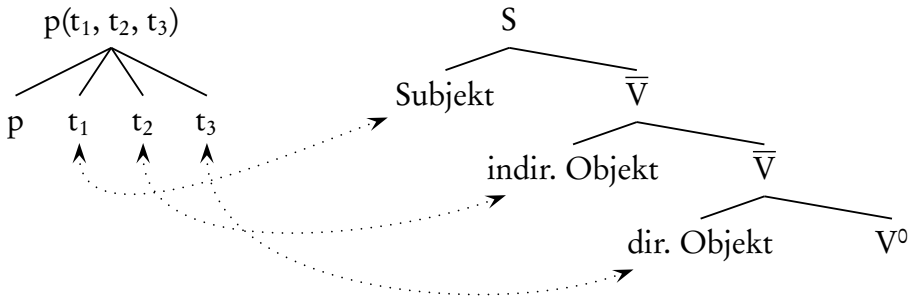


Die Skopusambibiguität wird hier durch die unterschiedliche Reihenfolge der Anwendung von Existenzquantor und Allquantor rekonstruiert. In Abschnitt 5 wird die Operation *quantifier raising (QR)* vorgestellt, die derartige Lesarten systematisch ableitet.

3 Theorie der logischen Typen

Vergleicht man die Strukturen, die in PL_1 mit Hilfe der Regel 1 erzeugt werden, mit der Struktur der Verbalphrase im Deutschen, so stellt man leicht fest, dass die Hierarchie, die VPen im Deutschen aufweisen, in der PL_1-Struktur nicht reflektiert ist. Die syntaktische Struktur der Prädikat-Argument-Verbindung in der prädikatenlogischen Aussage $p(t_1, t_2, t_3)$ ist flach, während die entsprechende Struktur des deutschen Satzes hierarchisch strukturiert ist:

(23) PL_1-Struktur: Konstituentenstruktur im Deutschen:



In der PL_1-Form sind die Argumente strukturell nicht unterschieden, denn sie liegen auf einer hierarchischen Ebene. Im Deutschen lassen sich jedoch ganz klar strukturelle Unterschiede zwischen den Objekten untereinander und den Objekten und dem Subjekt feststellen. Wir wollen erreichen, dass die prädikatenlogische Form ebenfalls hierarchisch ist, und damit der syntaktischen Struktur des Deutschen entspricht.

Die klassische Typentheorie[12] unterscheidet die beiden elementaren Typen e (= entity) und t (= truth value). Auf der Basis der rekursiven Vorschrift in (24) können aus diesen beiden Typen beliebig komplexe neue Typen erzeugt werden:

(24) Wenn a und b Typen sind, dann ist ⟨a,b⟩ ein Typ.

Die möglichen Denotate für die jeweiligen Klassen von logischen Typen bestimmen sich dann gemäß (25):

(25) Mögliche Denotate für Typen:

i. Wenn α vom logischen Typ e ist, so ist das mögliche Denotat von α ein Individuum aus D.

ii. Wenn α vom logischen Typ t ist, so ist das mögliche Denotat von α ein Wahrheitswert, d. h. ein Element aus {wahr, falsch}.

iii. Wenn φ vom logischen Typ ⟨a,b⟩ ist, so ist das mögliche Denotat von φ eine *Funktion aus* $D_b^{D_a}$ *von möglichen Denotaten vom Typ a in mögliche Denotate vom Typ b.*

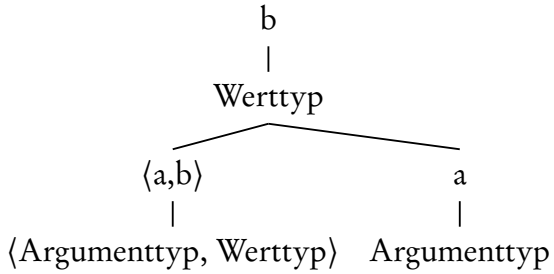
Während (25.i) und (25.ii) die möglichen Denotate für die elementaren Typen angeben, spezifiziert (25.iii) in allgemeiner Weise den logischen Typ einer komplexen Funktorkategorie φ. Die Funktion φ bildet Denotate vom Typ a auf Denotate vom Typ b ab:

(26) φ: $\boxed{D_a} \longrightarrow \boxed{D_b}$

Die komplexen Funktortypen⟨a,b⟩ stehen daher für Funktionen, die auf ein Argument (vom Typ a) angewendet einen Wert (vom Typ b) ergeben, so dass ein komplexer Typ immer ein geordnetes Paar darstellt, das aus dem Argumenttyp a und einem Werttyp b besteht: ⟨a,b⟩ = ⟨Argumenttyp, Werttyp⟩:

[12] Vgl. Russell, *Mathematical Logic*.

(27)

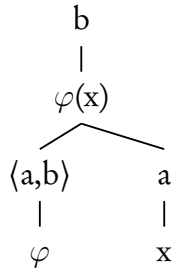


Die Anwendung eines Ausdrucks mit einem (komplexen) Funktortyp auf einen Ausdruck mit einem Argumenttyp ist die *Funktionale Applikation (FA)*, die sich wie in (28.i) charakterisieren lässt und zu einer Darstellung wie in (28.ii) führt:

(28) Funktionale Applikation (FA):

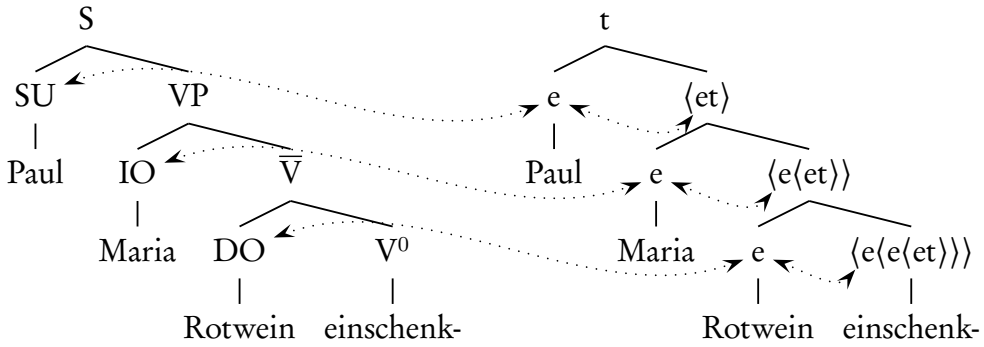
i. Wenn φ ein Ausdruck vom Typ ⟨a,b⟩ und x ein Ausdruck vom Typ a ist, dann ist φ(x) ein Ausdruck vom Typ b.

ii.



Auf diese Weise sorgen die logischen Typen dafür, dass mehrstellige Prädikate ‚geschönfinkelt'[13] werden, d. h. als Sukzession von Funktionen in einer Variablen geschrieben werden können. Schreibt man die Typen ohne Komma, so ergibt sich die folgende Struktur:

(29)



Gegenüber der flachen Struktur, welche die syntaktische PL_1-Regel in (14.i) erzeugt, spiegelt die getypte Struktur in (29) bereits die Hierarchie der syntaktischen Argumente im Deutschen wider.

[13] Moses Schönfinkel (1889--1942) war ein sowjetischer Mathematiker, der u. a. auch an der Universität Göttingen lehrte.

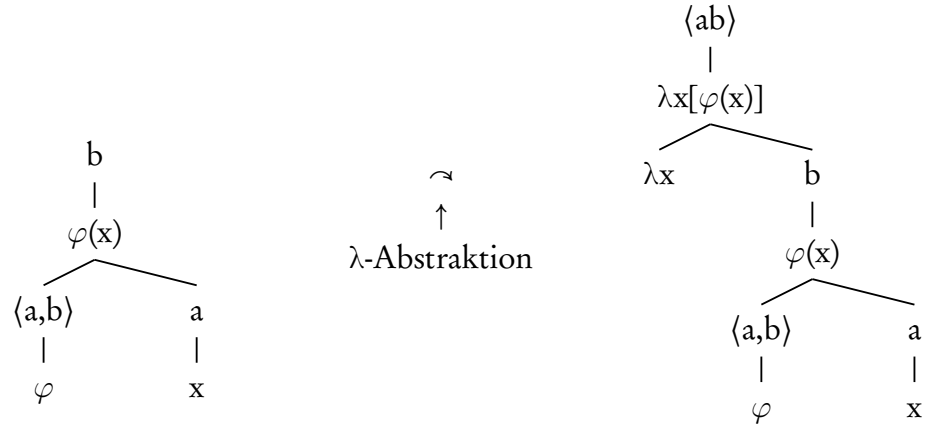
4 Der λ-Operator

Die semantische Analyse natürlich-sprachlicher Ausdrücke basiert wesentlich auf dem Begriff *Funktion*. Die damit verbundene Konzeption geht auf Freges Analyse des *Begriffs* zurück,[14] derzufolge ein Satz mit einem einstelligen Prädikat genau dann wahr ist, wenn das Denotat des Arguments unter den Begriff fällt, oder -- anders ausgedrückt -- wenn der Begriff -- als Funktion aufgefasst -- das Argument auf das Wahre abbildet. Ein solcher Satz ist falsch, wenn das Denotat des Arguments nicht unter den Begriff fällt und entsprechend auf das Falsche abgebildet wird.

Der zentrale Stellenwert des Begriffs *Funktion* führte Alonzo Church zur Entwicklung des λ-Kalküls (Lambda-Kalküls).[15] Dieser Kalkül stellt ein System von abstrakten Prinzipien dar, mit deren Hilfe sich einerseits *Funktionen* systematisch konstruieren, andererseits aber auch Argumentstellen gegen Argumente konvertieren lassen. λ-Abstraktion einer Variablen x vom Typ a überführt einen Ausdruck vom Typ b in einen Ausdruck vom Typ ⟨ab⟩:

(30) λ-Abstraktion:

Wenn x eine Variable vom Typ a und $\varphi(x)$ ein Ausdruck vom Typ b ist, in dem die Variable x frei vorkommt, dann ist der Ausdruck $\lambda x[\varphi(x)]$ vom Typ ⟨a,b⟩:



Das Pedant zur λ-Abstraktion ist die λ-Konversion, bei der ein Argument -- nach Funktionaler Applikation (FA) -- wieder in den λ-Ausdruck eingesetzt wird:

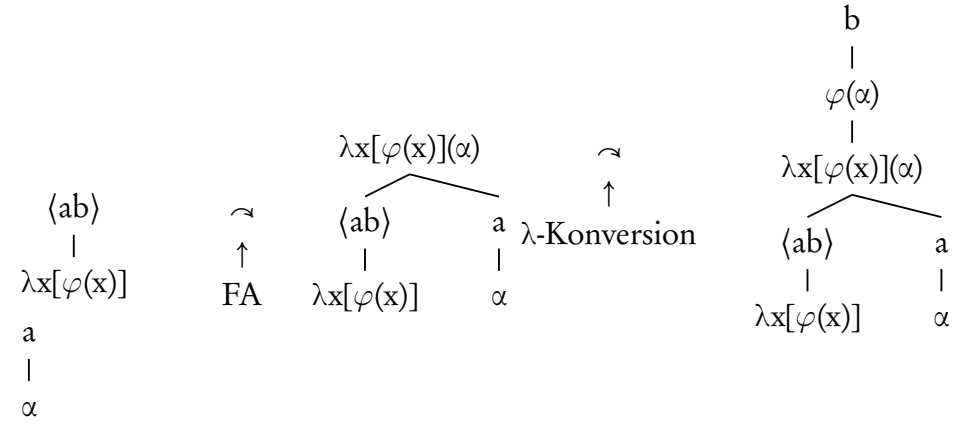
(32) λ-Konversion:

Sei φ ein Ausdruck vom Typ b und x eine Variable vom Typ a. Wenn α vom Typ a ist, dann gilt: $\lambda x[\varphi(x)](\alpha) \equiv \varphi(\alpha)$.

---

14 Vgl. Frege, *Über Begriff und Gegenstand*.

15 Vgl. Church, *The Calculi of Lambda-conversion*.



Während λ-Abstraktion eine (freie) Variable von beliebigem Typ a aus einem Ausdruck abstrahiert und damit eine Funktion über Ausdrücke vom Typ a bildet, bewirkt nach Funktionaler Applikation die λ-Konversion gerade, dass die Argumentausdrücke die Variablen im Funktionsausdruck belegen:

(33) Semantik des λ-Operators:

Wenn x eine Variable vom Typ a und $\varphi(x)$ ein Ausdruck vom Typ b, in dem x frei vorkommt, dann ist $[\![\lambda x[\varphi(x)]]\!]^{M,g}$ diejenige Funktion h von D^a nach D^b, für die gilt: $h(u) = [\![\varphi(x)]\!]^{M,g^u_x}$, für alle $u \in D^a$, wobei g^u_x die Variablenbelegung g, die an der Stelle x den Wert u hat.

Die Semantik des λ-Operators ist damit genau so formuliert, dass λ-Ausdrücke genau diejenige Funktion als Denotat haben, die alle Werte des Prädikatsausdrucks im jeweiligen Modell M ergibt.

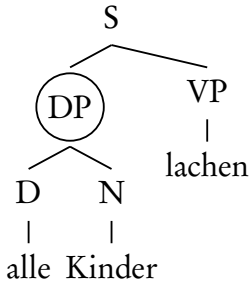
Angewendet auf die Struktur der Lexikoneinträge, lässt sich der prädikative Charakter und die Argumentstellen der lexikalischen Einheiten mit Hilfe der λ-Operatoren repräsentieren, wie dies in der Darstellung des Lexems *geben* in (2) bereits vorwegnehmend dargestellt wurde.

5 Quantifikation

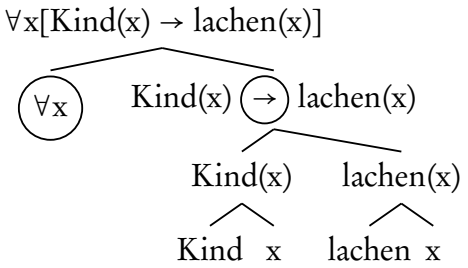
Natürliche Sprachen verfügen über mannigfaltige Möglichkeiten, sich auf Mengen von Objekten in der konzeptuellen Ontologie von Menschen zu beziehen. Dazu zählen neben Individuen auch Zeiten, Welten (Möglichkeiten), Ereignisse usw. Betrachtet man etwa einen Satz wie (34.i), der die syntaktische Struktur (34.ii) hat, und stellt man die Frage, wie das Denotat der Konstituente *alle Kinder* zu repräsentieren sei, so zeigt sich, dass der entsprechende PL_1-Ausdruck die Konstituenz überhaupt nicht ausdrückt, denn die mit dem Allquantor verbundenen Komponenten gehören strukturell nicht zu einer Konstituente, wie (34.iii) zeigt:

(34) i. (weil) alle Kinder lachen.

ii. Syntax:



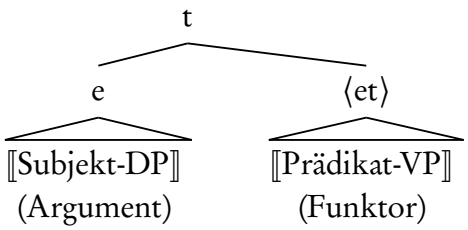
iii. Semantik:



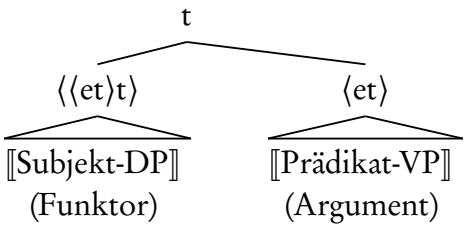
Das Denotat ⟦alle Kinder⟧ der syntaktischen Subjekts-DP[16] ‚alle Kinder' findet in dem PL_1-Ausdruck also keine Entsprechung, so dass eine andere Repräsentation gefunden werden muss, um die syntaktische Konstituenz auch auf der Bedeutungsseite auszudrücken.

Einen anderen Problemfall stellen Quantorenausdrücke wie etwa ‚die meisten' dar, die in der Sprache PL_1 nicht ausgedrückt werden können. Einen Weg, um diese Probleme zu lösen, stellt die Theorie der generalisierten Quantoren dar.[17] Folgt man der klassischen Auffassung, so nimmt das Prädikat in einem Satz das Subjekt als Argument. Das Prädikat ist dabei ein Funktorausdruck vom logischen Typ ⟨et⟩, der das Subjekt vom logischen Typ e als Argument nimmt (35.i). In der Theorie der generalisierten Quantoren wird dieses Verhältnis gerade umgekehrt. Das Subjekt wird dabei nicht als eine Entität vom Typ e aufgefasst, sondern als die Menge aller Eigenschaften vom Typ ⟨e⟨et⟩⟩, die diese Entität hat (35.ii). Um die Wahrheit eines Satzes zu ermitteln, muss folglich festgestellt werden, ob die vom Prädikat ausgedrückte Eigenschaft zu den Eigenschaften des Subjekts gehört. Dabei wird das Subjekt zum Funktorausdruck und das Prädikat zum Argument:

(35) i. klassisch:



ii. generalisierte Quantifikation:



Einen generalisierten Quantor konstruiert man aus der Wahrheitsbedingung für einen Satz, in dem eine quantifizierende Subjekt-DP enthalten ist, indem man von der Prädikats-VP[18] mithilfe des λ-Operators abstrahiert. Dies ergibt sich aus

[16] DP steht für Determinatorphrase, eine etwas neuere Kategorienbezeichnung für die klassische NP (= Nominalphrase).

[17] Siehe Barwise, Cooper: *Generalized Quantifiers*.

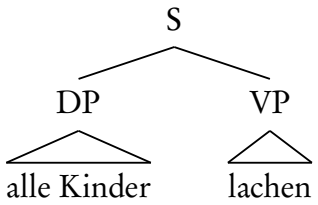
[18] VP ist die Kategorienbezeichnung der Verbalphrase.

Freges Kompositionalitätsprinzip, denn wenn sich ⟦S⟧ in (36.iii) aus der Bedeutung von ⟦DP⟧ und ⟦VP⟧ und der Art ihrer Kombinatorik ergibt, so ergibt sich ⟦DP⟧, indem man in ⟦S⟧ von ⟦VP⟧ abstrahiert. Genau dies leistet die λ-Abstraktion in (36.v). Dabei wird derjenige Teil, welcher der VP (hier: $[_{VP}$ lachen]) entspricht, durch eine Variable vom gleichen Typ (hier: Q) ersetzt und λ-abstrahiert, wie man es beim Übergang von (36.iv) zu (36.v) sieht:

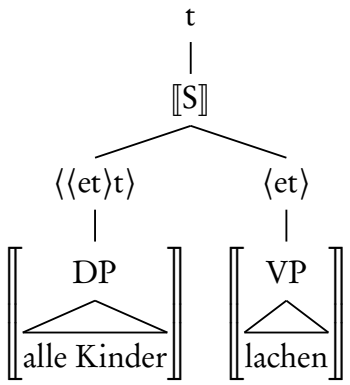
(36)

i. (weil) alle Kinder lachen.

ii. Syntax:



iii. Semantik:



iv. ⟦S⟧ = ∀x[Kind(x) → lachen(x)], TYP(⟦S⟧) = t

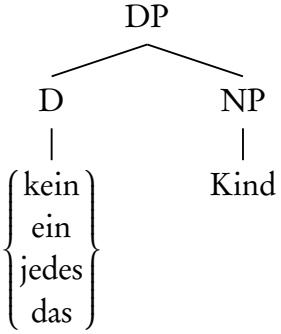
v. ⟦DP⟧ = λQ[∀x[Kind(x) → Q(x)]], TYP(⟦DP⟧) = ⟨⟨et⟩t⟩

Mithilfe dieses Verfahrens (λ-Abstraktion des VP-Denotats) lassen sich auch andere generalisierte Quantoren aus den jeweiligen PL_1-Übersetzungen gewinnen:

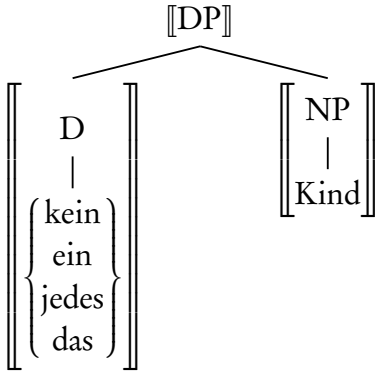
(37) a. ⟦kein Kind⟧: λQ[¬(∃x[Kind(x) ∧ Q(x)])]
b. ⟦ein Kind⟧: λQ[∃x[Kind(x) ∧ Q(x)]]
c. ⟦jedes Kind⟧: λQ[∀x[Kind(x) → Q(x)]]
d. ⟦das Kind⟧: λQ[∃x[Kind(x) ∧ ∀y[Kind(y) → x=y]] ∧ Q(x)]

Man macht sich leicht klar, dass sich -- folgt man wiederum Freges Kompositionalitätsprinzip -- die Bedeutung der ‚reinen' Quantoren ergibt, wenn man von dem Denotat ⟦DP⟧ der DP das Denotat ⟦NP⟧ der NP ebenfalls λ-abstrahiert. Die syntaktische Struktur der DP in (38.i) besteht aus dem Kopfelement D und seinem Komplement NP:

(38) i.



ii.

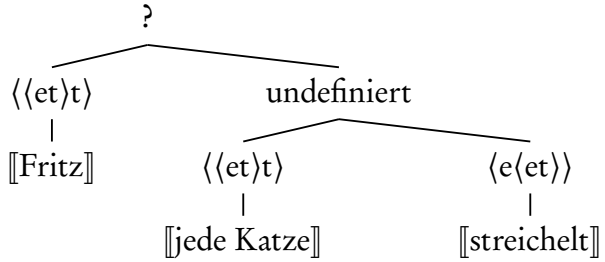


λ-abstrahiert man nun auch das NP-Denotat, indem man dafür eine Variable (etwa P) vom logischen Typ ⟨et⟩ einsetzt und λ-abstrahiert, so stellen die jeweils resultierenden Ausdrücke Denotate für die reinen Quantoren dar:

(39) a. ⟦kein⟧: λP[λQ[¬(∃x[P(x) ∧ Q(x)])]]
b. ⟦ein⟧: λP[λQ[∃x[P(x) ∧ Q(x)]]]
c. ⟦jedes⟧: λP[λQ[∀x[P(x) → Q(x)]]]
d. ⟦das⟧: λP[λQ[∃x[P(x) ∧ ∀y[P(y) → x=y]] ∧ Q(x)]]

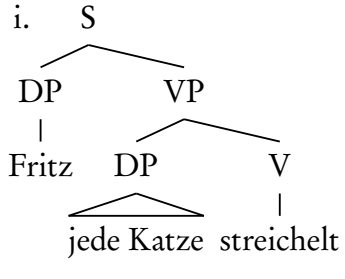
Für transitive Verben ist die Analyse als generalisierter Quantor aber nicht problemlos möglich, weil die Objekt-NP die Bedingung für die Kombinatorik der logischen Typen nicht erfüllt:

(40) i. (weil) Fritz jede Katze streichelt.
ii.

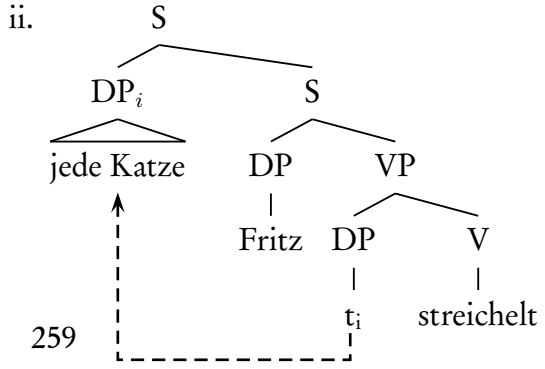


Behandelt man nämlich sowohl Subjekt als auch Objekt als generalisierte Quantoren vom logischen Typ ⟨⟨et⟩t⟩, so kann zwischen Objekt und Verb keine funktionale Applikation stattfinden, da die logischen Typen nicht passen. Ein Ausweg aus dieser Schwierigkeit lässt sich mit Hilfe einer angemessenen Interpretation der syntaktischen Operation *Quantoren-Anhebung* (Quantifier Raising (QR)) finden. Dieser Operation zufolge kann eine DP_i aus der sie umgebenden XP unter Hinterlassung einer Spur t_i extrahiert und an diese Phrase adjungiert werden:

(41) i.

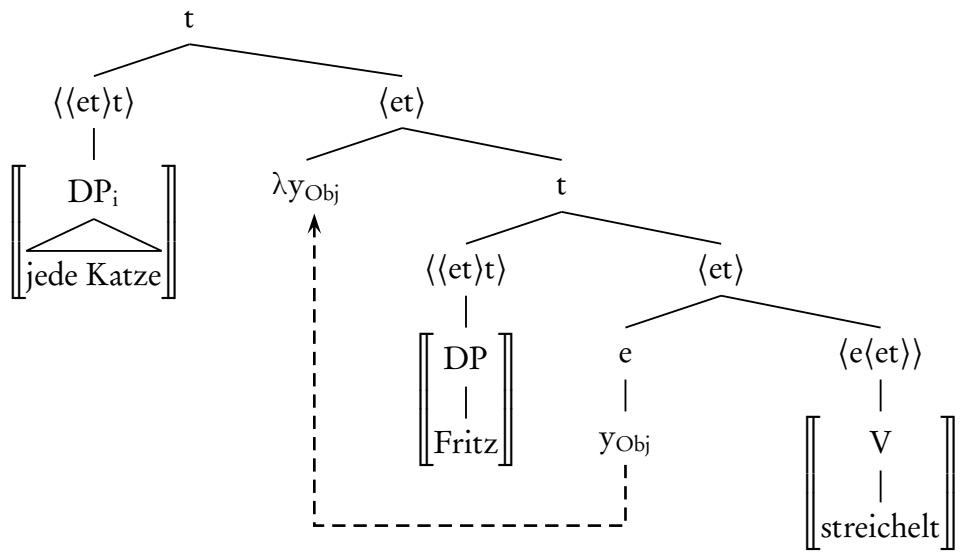


ii.



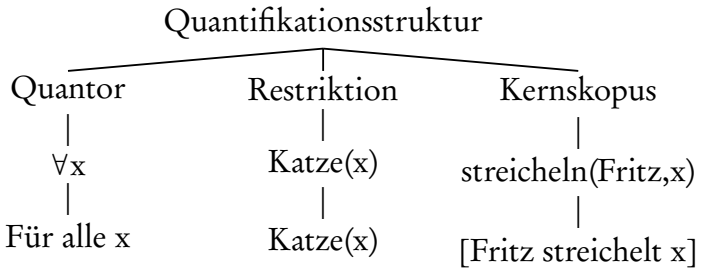
Damit wird die quantifizierende DP in die für die logische Form (LF) nötige Skopusposition versetzt. Interpretiert man die Spur t_i als Variable vom Typ e, und behandelt man den Index i an der versetzten Konstituente ‚jede Katze' als λ-Abstrakt der Variablen, so können sowohl Subjekt als auch Objekt als generalisierter Quantor realisiert werden und der Typenkonflikt verschwindet[19]:

(42)



Die Operation *Quantoren-Anhebung (QR)* vermeidet aber nicht nur den Typenkonflikt, sondern ist unabhängig davon erforderlich, um Skopusambiguitäten abzuleiten, die in Beispiel (22) schon betrachtet wurden. Dort fand sog. unrestringierte Quantifikation statt, da die semantischen PL_1-Regeln in (19.iii) und (19.iv) für die *gesamte* Diskursdomäne formuliert sind. Um einen Quantorenausdruck wie *jede Katze* zu evaluieren, muss man aber nicht die Hunde, Hamster und Meerschweinchen konsultieren, sondern eben nur die Katzen, d. h. die Individuen der Diskursdomäne können auf die Menge der Katzen eingeschränkt (restringiert) werden. Dies führt zum Konzept der *restringierten Quantifikation*, die eine dreigeteilte Quantifikationsstruktur annimmt, die aus dem Quantor, einer Restriktion und einem (Kern-)Skopus besteht, wie es in der folgenden Struktur dargestellt ist:[20]

(43)



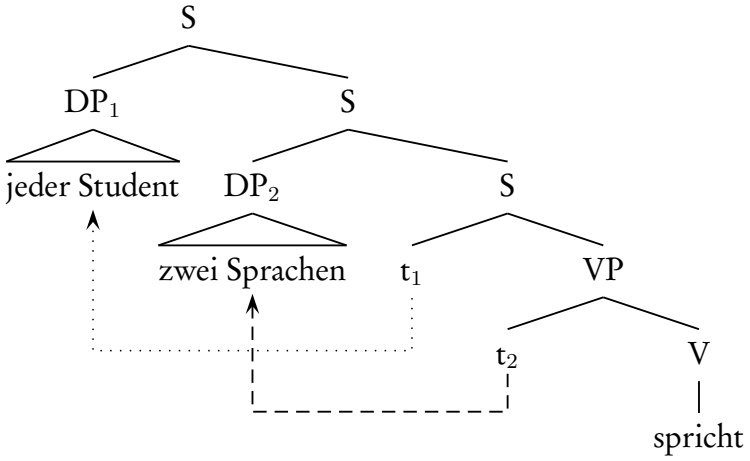
[19] Vgl. Heim, Kratzer, *Semantics in Generative Grammar*.
[20] Vgl. von Fintel, *Restrictions on Quantifier Domains*.

Der Quantor ∀x steht dabei in linker c-kommandierender Skopus-Position vor dem (Kern-)Skopus. Die Operation QR versetzt eine Quantorenphrase in genau diese Position, so dass auf der Ebene der logischen Form (LF) im Syntaxmodell in (4) Quantorenphrasen in der richtigen (Skopus-)Position stehen.

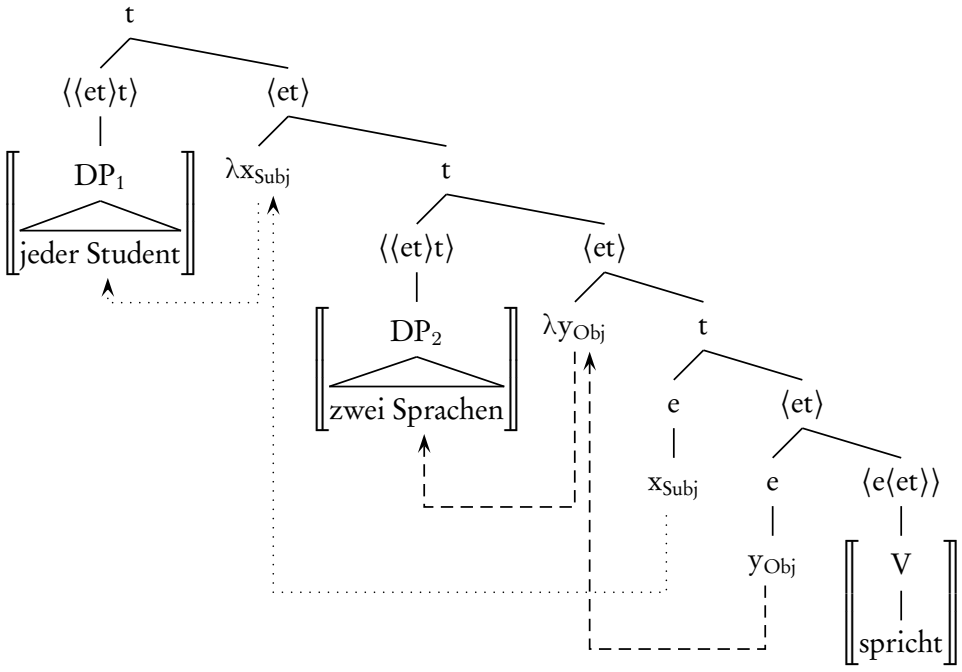
QR kann auf mehrere Quantorenphrasen angewendet werden, wie das folgende Beispiel (44) zeigt, bei dem zunächst das Objekt über das Subjekt und dann das Subjekt über das Objekt angehoben wurde, so dass Quantorenphrasen generell in eine linksperiphere Skopusposition des Satzes versetzt werden:

(44) i. (weil) jeder Student zwei Sprachen spricht.

ii. Syntax:

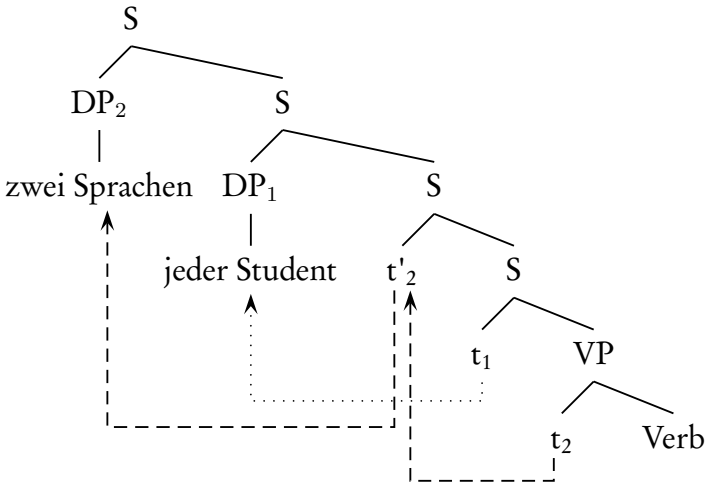


iii. typentheoretische Semantik:

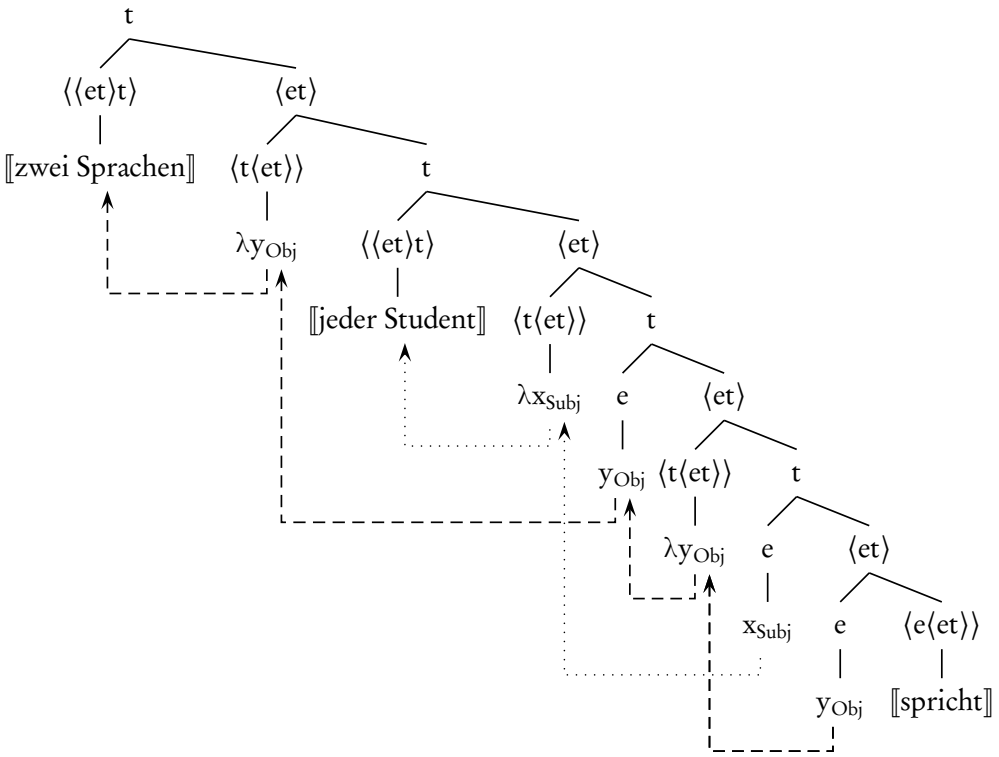


QR kann auf eine Quantorenphrase mehrfach angewendet werden. Außer den Skopusverhältnissen ändert sich dabei nichts. Auf diese Weise lassen sich Skopusambiguitäten wie in (22) systematisch ableiten, wie es in (45.i) dargestellt wird:

(45) i. Syntax:



ii. Semantik:

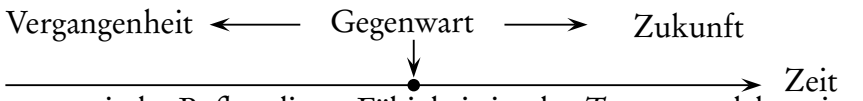


Indem das Objekt einmal mittels QR versetzt wurde, befindet es sich in der linksperipheren Skopusposition. Da auch die Subjekts-Quantorenphrase mittels QR nach vorne versetzt wird, hat der Subjekts-Quantor Skopus über den Objekts-Quantor. Wendet man QR erneut auf den Objekts-Quantor an, so ergibt sich die

inverse Skopus-Lesart mit der Paraphrase: *Es gibt zwei Sprachen (etwa Deutsch und Englisch), die jeder Student spricht.*

6 Tempus

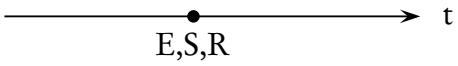
Zeit ist ein *Ordnungsprinzip* des menschlichen Geistes, welches die im Gedächtnis gespeicherten Ereignisse in ihrer zeitlichen Abfolge ordnet:

(46) 

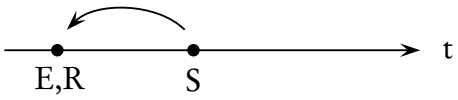
Der grammatische Reflex dieser Fähigkeit ist das *Tempus*, welches eine *grammatische Kategorie* ist, die bestimmte zeitliche Bezüge zwischen der Sprechzeit und der Zeit des Ereignisses ausdrückt. Die Tempora lassen sich nach der Theorie von Hans Reichenbach (1947) mit Hilfe von drei Zeitparametern recht gut charakterisieren:[21] (1) Die Sprechzeit S, (2) die Ereigniszeit E, (3) die Referenzzeit R:

(47) Interpretation der Tempora nach der Theorie von Reichenbach:

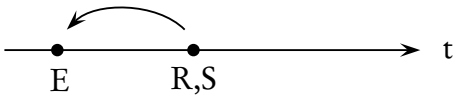
Präsens: Fritz lacht.
E = S = R:



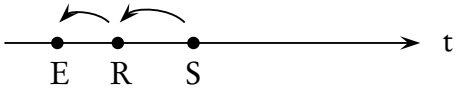
Präteritum: Fritz lachte.
E = R < S:



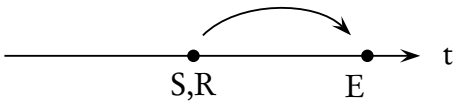
Perfekt: Fritz hat gelacht.
E < R = S:



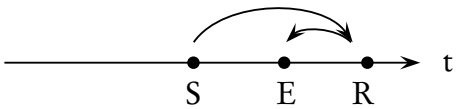
Plusquamperfekt: Fritz hatte gelacht.
E < R < S:



Futur 1: Fritz wird lachen.
S = R < E:



Futur 2: Fritz wird gelacht haben.
S < E < R:



Erweitert man das PL_1-Modell um eine -- in Analogie zur Menge $\mathbb{R}$ der reellen Zahlen unendlich großen -- Menge von Zeitpunkten und erweitert man das System

[21] Vgl. Reichenbach, *Elements of Symbolic Logic*.

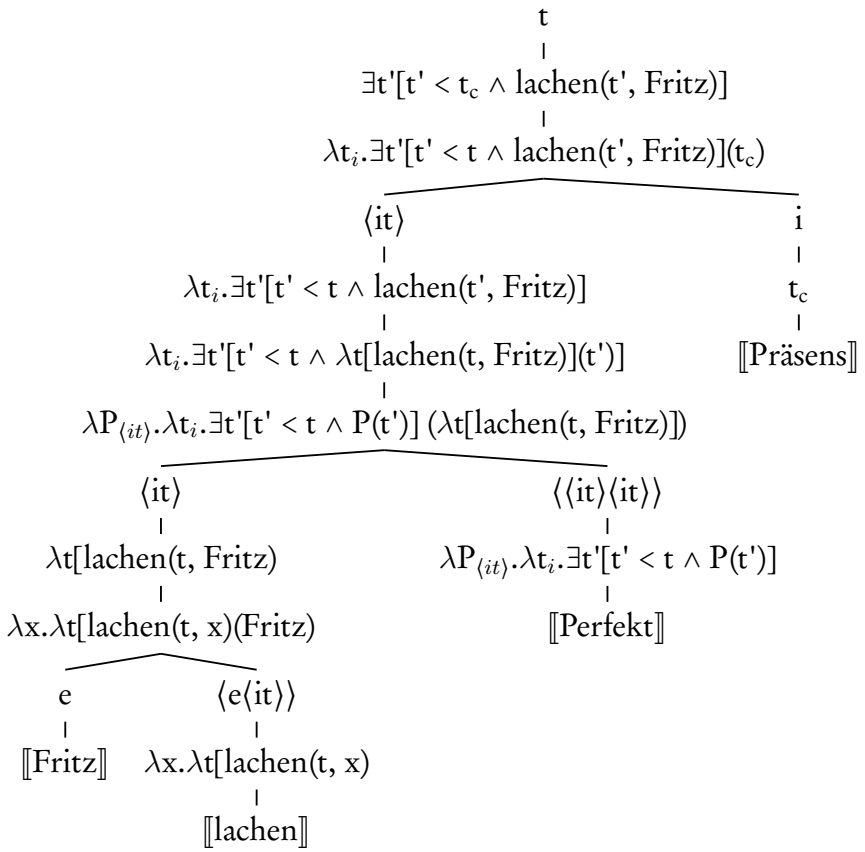
der logischen Typen um den Typ i für Zeitintervalle, so lassen sich die Tempora in das bisher entwickelte Sprachsystem integrieren.

(48) i ist der (Basis-)Typ für Zeitintervalle.
  i. Erweiterung der Basistypen: i ist ein Typ.
  ii. Semantik: Wenn α vom logischen Typ i ist, so ist das mögliche Denotat von α ein Zeitintervall.

Zur Illustration betrachten wir einen Satz des Deutschen im Perfekt. Das Perfekt -- wie alle Tempora des Deutschen außer Präsens und Präteritum -- wird im Deutschen mittels einer periphrastischen Konstruktion gebildet und aus der Form des Partizips II und dem Hilfsverb im Präsens zusammengesetzt.[22]

Setzt man als Denotat des Präsens die aktuelle Sprechzeit t_c an (c für context) wie in (49.i) und für das Perfekt ein Denotat wie in (49.ii),[23] so lässt sich der Perfekt-Satz in (49.iii) kompositionell herleiten wie in (49.iv):

(49) i. $[\![\text{Präsens}]\!] = t_c$
  ii. $[\![\text{Perfekt}]\!] = \lambda P_{\langle it \rangle}.\lambda t_i.\exists t'[t' < t \wedge P(t')]$
  iii. (weil) Fritz gelacht hat
  iv.



[22] Vgl. etwa Ballweg, *Tempusformen* für eine kompositionelle Analyse der Tempora im Deutschen.

[23] Vgl. von Stechow, *Schritte zur Satzsemantik*.

Dabei sichert das Denotat des Perfekts die Existenz eines vergangenen Zeitintervalls t' und das Präsens stellt den Bezug zur aktuellen Sprechzeit t_c her.

7 Modalität

Neben der *Wirklichkeit*, in der wir leben, gibt es unendlich viele *Möglichkeiten*, wie die Welt beschaffen sein könnte, de facto aber nicht ist. So lassen sich etwa mit Satzadverbien wie *möglicherweise* oder *notwendigerweise* Sätze *modalisieren*:

(50) i. Möglicherweise ist Fritz in Paris.
ii. Notwendigerweise gilt der Satz von Pythagoras.

Für die Wahrheit von (50.i) ist es irrelevant, ob Fritz in Paris ist oder nicht. Für die Wahrheit von (50.ii) ist es unerheblich, ob das Hypothenusenquadrat *einiger* konkreter rechtwinkliger Dreiecke gleich der Summe der Kathetenquadrate ist -- wenn (50.ii) wahr ist, so gilt dies für *alle* rechtwinkligen Dreiecke.

Der Satz in (50.i) ist also wahr, wenn es irgendeine mögliche Situation (Welt) gibt, in welcher der nicht modalisierte Satz wahr ist, und (50.ii) ist wahr, wenn für alle rechtwinkligen Dreiecke in allen möglichen Situationen die Summe der Kathetenquadrate gleich dem Hypothenusenquadrat ist.

Um die Wahrheitsbedingungen von notwendiger- und möglicherweise wahren Sätzen korrekt darstellen zu können, erweitern wir das Inventar der logischen Typen um einen Typ s für die Elemente in der Menge S der möglichen Situationen:[24]

(51) s ist der (Basis-)Typ für mögliche Situationen.
i. Erweiterung der Basistypen: s ist ein Typ.
ii. Semantik: Wenn α vom logischen Typ s ist, so ist das mögliche Denotat von α eine *Situation*.

Wir erweitern die PL_1-Regeln um den Möglichkeits-Operator **M** und den Notwendigkeits-Operator **N**:

(52) i. Syntaktische Regeln:[25]
i. Wenn φ eine Aussage ist, dann ist $\mathbf{M}\varphi$ eine Aussage. (Möglichkeit)
ii. Wenn φ eine Aussage ist, dann ist $\mathbf{N}\varphi$ eine Aussage. (Notwendigkeit)
ii. Semantische Regeln:
i. $[\![\mathbf{M}\varphi]\!]$ = wahr, gdw. mindestens eine Situation $s \in S$ gibt, so dass $[\![\varphi]\!]$ = wahr.
ii. $[\![\mathbf{N}\varphi]\!]$ = wahr, gdw. für alle Situationen $s \in S$ gilt: $[\![\varphi]\!]$ = wahr.

[24] In Montagues (1974) intensionalem System ist s kein logischer Typ, sondern kennzeichnet nur in Kombination mit einem logischen Typ dessen intensionales Denotat.

[25] Wir verwenden hier aus Gründen der Einheitlichkeit der Notation in diesem Band für den Notwendigkeits-Operator das Zeichen **N**, das dem Zeichen □ entspricht und das Zeichen **M** für den Möglichkeits-Operator, der auch als ◇ geschrieben wird.

(52.ii-i) entspricht einer Existenzquantifikation und (52.ii-ii) einer Allquantifikation über mögliche Situationen.[26]

Modalverben sind andere sprachliche Mittel, mit denen Sätze modalisiert werden können. Die Modalverben im Deutschen: *müssen*, *sollen*, *wollen*, *können*, *dürfen*, *brauchen*, *möchten* lassen sich auf verschiedene Weisen interpretieren: epistemisch, deontisch, disponentiell, ..., so dass ihre theoretische Behandlung einerseits darin bestehen könnte, für jede Interpretation ein separates Lexem mit je eigener Semantik anzusetzen, oder andererseits -- wie in Kratzer[27] entwickelt -- eine Relativierung auf verschiedene Redehintergründe. Verfolgt man die erste Strategie, so bleiben die Gemeinsamkeiten bei den verschiedenen Interpretationen der Modalverben zufällig. Mit der zweiten Strategie lassen sich hingegen sowohl die Unterschiede wie auch die Gemeinsamkeiten der Modalverben angemessen erfassen. Die Sätze in (53.i) und (53.ii) mit den Modalverben *müssen* und *können* illustrieren die verschiedenen Interpretationsmöglichkeiten:

(53) i. Paul *muss* kommen.
Hinsichtlich dessen, ...
... was das Gesetz verlangt, muss es so sein, dass Paul kommt.
(deontisch)
... was wir wissen, muss es so sein, dass Paul kommt.
(epistemisch)
... was wir uns wünschen, muss es so sein, dass Paul kommt.
(buletisch)
...
ii. Paul *kann* kommen.
Hinsichtlich dessen, ...
... was das Gesetz verlangt, kann es so sein, dass Paul kommt.
(deontisch)
... was wir wissen, kann es so sein, dass Paul kommt.
(epistemisch)
... wozu Paul in der Lage ist, kann es so sein, dass er kommt.
(disponentiell)
...

Kratzer unterscheidet drei Komponenten, die für die Interpretation von Modalverb-Konstruktionen relevant sind:[28]

(54) i. modale Kraft (modal force)
ii. ein Redehintergrund HG (modal base)
iii. Ordnungsquelle I (Ideal) (ordering source)

---

[26] S bezeichne die Menge aller *möglichen Situationen s*. Die Bezeichnung und Charakterisierung geht auf Kratzer (1989) zurück; bei Leibniz: *mögliche Welten*, bei Wittgenstein: *states of affairs*, bei Carnap: *state descriptions*, bei Kripke: *mögliche Welten*.

[27] Vgl. Kratzer, *Was ‚können' und ‚müssen' bedeuten können müssen* und Kratzer, *Modality*.

[28] Vgl. Kratzer, *Modality*.

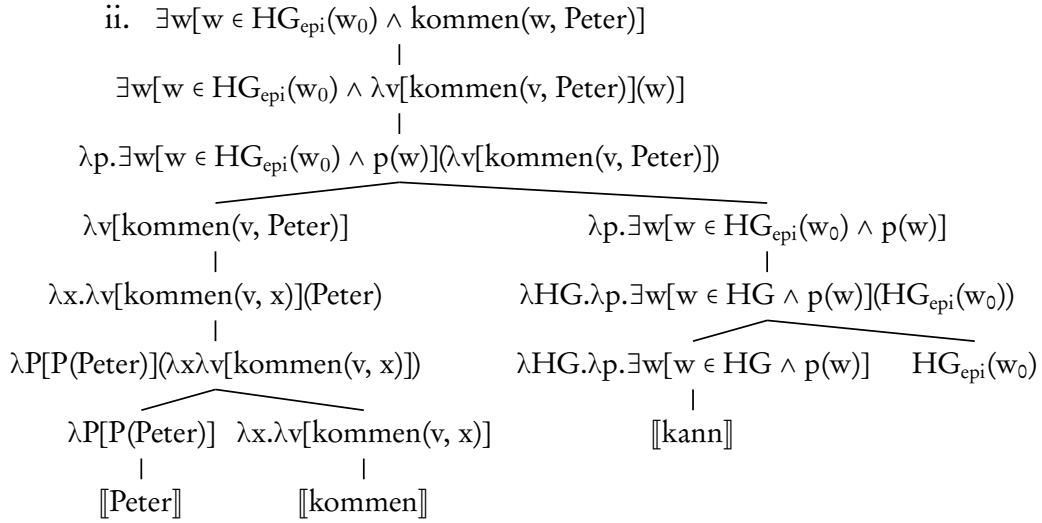
Beide Modalverben quantifizieren über mögliche Situationen, entweder -- wie im Falle von *können* -- als Existenzquantifikation oder -- wie im Falle von *müssen* -- als Allquantifikation. Diese Art der Quantifikation bestimmt die modale Kraft und stellt eine jeweils invariante Eigenschaft eines Modalverbs dar. Ein Redehintergrund kann aufgefasst werden als eine Funktion, die jeder Situation s diejenige Menge von Propositionen zuweist, die in s gewusst, gehofft, geglaubt, vom Gesetz vorgeschrieben, ... werden. Die modale Kraft bezieht sich dann nur auf die Situationen, die von dem jeweiligen Redehintergund in einer Situation s bestimmt sind. Eine Ordnungsquelle -- gelegentlich auch als *Ideal* bezeichnet -- erlaubt es, die möglichen Situationen im Hinblick auf ihre Entsprechung zu der vom Ideal bestimmten Propositionen zu ordnen.[29] Die semantische Form von *können* und *müssen* zeigt (55):

(55) i. $[\![\text{können}]\!]$: $\lambda HG_v.\lambda p.\exists w[w \in [\![HG_v]\!] \wedge p(w)]$, oder kurz: $HG_v \cap p \neq \varnothing$.
ii. $[\![\text{müssen}]\!]$: $\lambda HG_v.\lambda p.\forall w[w \in [\![HG_v]\!] \rightarrow p(w)]$, oder kurz: $HG_v \subseteq p$.

Modalverben sind demzufolge als zweistellige Prädikate zu analysieren, die einen (implizit gegebenen) Redehintergrund HG und eine Proposition p als Argumente nehmen. Gemäß (55.i) gilt dann für eine mit *können* modalisierte Proposition, dass sie mit dem jeweiligen Redehintergrund HG *kompatibel* sein muss, während eine mit *müssen* modalisierte Proposition aus ihrem jeweiligen HG zu *folgen* hat.

Die kompositionelle Analyse der Modalverb-Konstruktion in (56.i) zeigt die Struktur in (56.ii):

(56) i. (weil) Peter kommen kann.
ii.



[29] Zur Ordnungssemantik vgl. Lewis, *Ordering Semantics*.

8 Intensionalität

Gottlob Frege hat den Begriff *Bedeutung* in die zwei Konzepte *Bedeutung* und *Sinn* differenziert:

(57) i. Die *Bedeutung* eines Satzes ist sein *Wahrheitswert*. (das *Wahre* oder das *Falsche*).
ii. Der *Sinn* eines Satzes ist der in ihm enthaltene *Gedanke*.

Dabei ist die *Bedeutung* eines Ausdrucks α der *Begriffsumfang* von α und der *Sinn* von α ist die *Art des Gegebenseins* von α. Der deutsche Philosoph Rudolf Carnap hat diese Unterscheidung vor dem Hintergrund *möglicher Situationen* und der Distinktion zwischen *Äquivalenz* und *Logischer Äquivalenz* mit Hilfe der Konzepte *Extension* und *Intension* auf die korrespondierenden Fregeschen Begriffe *Bedeutung* und *Sinn* bezogen und weiter expliziert.[30] Demzufolge stellt die *Intension* eines Ausdrucks α eine Funktion von möglichen Situationen in die *Extension* von α in den jeweiligen Situationen dar. Mit Hilfe der Intension eines Ausdrucks wie etwa *rot*, lässt sich also in jeder Situation s die Extension von *rot* -- also die Menge derjenigen Objekte, die in s rot sind -- bestimmen.

Ein Beispiel für eine intensionale Interpretation stellt das Verb *suchen* dar. So kann man sich etwa auf der Suche nach dem heiligen Gral befinden, obwohl dieser gar nicht existieren muss. Im Gegensatz dazu ist das Verb *finden* in seiner Objektposition extensional, denn man kann nichts finden, was es nicht gibt:

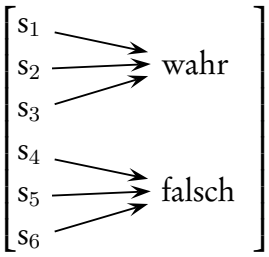
(58) i. weil Artus den heiligen Gral sucht.
suchen(s, Artus, λs'[HGral(s', x)])
Artus sucht in der Situation s nach der Intension des heiligen Grals.
ii. weil Parzival den heiligen Gral findet.
∃x[HGral(s, x) ∧ finden(s, Parzival, x)]
Es gibt ein x, x ist der Heilige Gral in s, und Parzival findet x in s.

(58.i) drückt aus, dass Artus in s in der Relation des Suchens zum Konzept des heiligen Grals steht, ohne dass dieser in s existieren müsste. (58.ii) drückt hingegen aus, dass es in s ein x gibt, welches der heilige Gral ist, und Parzival in der Relation des Findens zu diesem x in s steht.

Der Intension eines (Deklarativ-)Satzes entspricht eine Proposition. Eine Proposition p ist in einer Situation s ∈ S genau dann *wahr*, wenn s ein Element von ⟦p⟧ ist, wenn also gilt: s ∈ ⟦p⟧ oder ⟦p(s)⟧ = wahr; anderenfalls ist p *falsch* in s.

(59) i. ⟦Fritz raucht⟧ = Menge aller Situationen, in denen Fritz raucht.
ii. ⟦Fritz raucht⟧ = {s / Fritz raucht in s}

[30] Vgl. Carnap, *Meaning and Necessity*.

iii. $[\![\text{Fritz raucht}]\!] =$ 

Eine Proposition p führt damit zu einer *Bipartition der möglichen Situationen* in solche, in denen p wahr ist, und solche, in denen p falsch (bzw. ¬p wahr) ist:

(60)

p ist wahr	p ist falsch
s_1, s_2, s_3	s_4, s_5, s_6

Die Menge der möglichen Situationen, die von einer Proposition zutreffend beschrieben werden, kann mit dem Denotat dieser Proposition -- gemäß gängiger Praxis -- identifiziert werden. Entsprechend ist das Denotat des Satzes *Fritz raucht* die Menge derjenigen Situationen, in denen Fritz raucht, also: $[\![\text{Fritz raucht}]\!] = \{s_1, s_2, s_3\}$.

9 Mengen von Propositionen

Propositionen lassen sich zu Mengen von Propositionen zusammenfassen. Wenn eine Proposition eine Menge von Situationen denotiert, so denotiert eine Menge von Propositionen eine Menge von Mengen von Situationen. Für Propositionenmengen und ihr Verhältnis zu Propositionen lassen sich verschiedene Eigenschaften formulieren:

(61) Sei P eine Menge von Propositionen:

i. P ist *konsistent*, gdw. $\bigcap_{\forall p \in P} [\![p]\!] \neq \varnothing$.

ii. q ist mit P *kompatibel*, gdw. $P \cup \{q\}$ konsistent ist.

iii. q *folgt aus* P, gdw. $\forall s \in \bigcap_{\forall p \in P}$ gilt: q(s) = wahr, oder kurz: $[\![P]\!] \subseteq [\![q]\!]$.

Gemäß dieser Festlegungen ist eine Propositionenmenge P konsistent, wenn es mindestens eine Situation gibt, in der alle Propositionen wahr sind -- (61.i). Eine Proposition q ist mit einer Menge von Propositionen P kompatibel, wenn die um q ergänzte Propositionenmenge konsistent ist -- (61.ii), und eine Proposition q folgt aus einer Propositionenmenge P, gdw. die Menge derjenigen Situationen, in denen q wahr ist, die Menge der Situationen, in denen alle Propositionen aus P wahr sind, enthält.

So lässt sich etwa eine wissenschaftliche Theorie als eine Menge von Propositionen bestimmen, die bzgl. der aktuellen Situation konsistent sein muss und aus der weitere Theoreme gefolgert werden können. Aber auch das *epistemische System* Epi_s^i eines Individuums i in s kann als Propositionenmenge aufgefasst werden. In

diesem Fall ist dies die Menge aller Propositionen, die das Individuum i in s weiß: Epi_s^i = {p / i weiß p in s}.

Aber auch die Bedeutung von Fragesätzen kann als Menge von Propositionen rekonstruiert werden, wie zuerst von Charles Leonhard Hamblin (1973) im Rahmen der Montague-Grammatik vorgeschlagen wurde. Dieser Konzeption entsprechend lässt sich das Denotat einer Frage als die Menge der möglichen (propositionalen) Antworten auf sie charakterisieren:

(62) $[\![$Welcher Baum trägt Früchte?$]\!]$ = $\lambda p.\exists x[\text{Baum}(x) \wedge p = x \text{ trägt Früchte}]$
= Menge aller Propositionen p, für die gilt: Es gibt ein x, x ist ein Baum, und p ist die Proposition, dass x Früchte trägt.

10 Semantik von Fragen

Zur kompositionellen Rekonstruktion von Fragen betrachtet Hamblin die Denotationstypen von Personalpronomina im Kontrast zu Fragepronomina:

(63)

TYP(Personalpronomina)	=	e	(Individuen)
TYP(Fragepronomina)	=	⟨et⟩	(Menge von Individuen)

Demnach denotieren Personalpronomina *Individuen*, während Fragepronomina *Mengen von Individuen* denotieren. Obwohl die syntaktische Position eines Fragepronomens (bis auf wh-Bewegung) analog zu Personalpronomina distribuiert ist -- so Hamblin --, weisen Fragepronomina doch einen anderen Denotationstyp auf. Diese Erweiterung des Denotationstyps zieht eine Erweiterung der semantischen Kompositionsprinzipien nach sich, welche die Verbindung zwischen Prädikaten und Fragepronomina steuern.

Das Prinzip der semantischen Kombinatorik ist die Anwendung einer Funktion auf ein Argument, die in (28) definierte ‚Funktionale Applikation (FA)'. Wenn Fragepronomina aber *Mengen* von Individuen denotieren, muss diese einfache Funktionale Applikation erweitert werden, zu einer *Funktionalen Applikation für Denotations**mengen***:

(64) Funktionale Applikation für Denotationsmengen:
Sei A eine Menge von Funktorkategorien und B eine Menge von (passenden) Argumentkategorien, dann ist die funktionale Applikation A(B) für Denotationsmengen definiert wie folgt: A(B) := { a(b) / a ∈ A, b ∈ B }.

Jedes Element der Funktormenge A wird auf jedes Element der Argumentmenge B funktional appliziert, so dass die Funktionale Applikation für Denotationsmengen nichts anderes ist, als die wiederholte Anwendung der einfachen Funktionalen Applikation.

Der Fragessatz in (65) denotiert der Hamblinschen Theorie zufolge die Menge seiner möglichen Antworten. Diese Menge lässt sich nun einfach aus dem Denotat

des Frageprononens *wer* in (65.i), dem Denotat von *lachen* (die Funktion in (65.ii)) und mit Hilfe der Funktionalen Applikation für Denotationsmengen bestimmen wie in (65.iii) bzw. (65.iv):

(65) Wer lacht?

i. ⟦wer⟧ = {Peter, Maria, Clara}
ii. ⟦lachen⟧ = λx[lachen(x)]
iii. ⟦lachen(wer)⟧ = ⟦lachen⟧(⟦wer⟧) =
={λx[lachen(x)](Peter), λx[lachen(x)](Maria), λx[lachen(x)](Clara)}
={lachen(Peter), lachen(Maria), lachen(Clara)}
iv. {Peter lacht, Maria lacht, Clara lacht}

Direkte Fragen denotieren damit die Menge derjenigen Propositionen, die eine mögliche Antwort auf sie sein können.
Für Entscheidungsfragen wie (66) ist dies die zweielementige Propositionenmenge in (66.i) bzw. (66.ii):

(66) Regnet es?

i. λp[p = regnen ∨ p = ¬regnen]
ii. {Es regnet, Es regnet nicht}

Verben, die zwei [+w]-Argumentsätze verlangen, wie etwa *abhängen von* stellen eine Relation zwischen dem Denotat des Subjektsatzes und dem Denotat des Objektsatzes dar:

(67) Wer gewählt wird, hängt davon ab, wer beliebt ist.

Falls die [+w]-Argumentsätze jeweils alle möglichen Antworten denotieren -- wie Hamblin annimmt --, so könnte der Satz bedeuten, dass diejenigen gewählt werden, die nicht beliebt sind. Das ist natürlich keine mögliche Bedeutung von (67).

Dies liefert ein Argument für Lauri Karttunen, nur die *wahren Antworten* als Denotat der Frage zuzulassen.[31] Eine mögliche Darstellung dieses Sachverhalts gibt (68) an, wobei das Zeichen '$^\vee$' der *Extensor* und das Zeichen '$^\wedge$' der *Intensor* ist:

(68) i. λp[$^\vee$p ∧ ∃x[p = $^\wedge$lachen(x)]]
ii. λp[$^\vee$p ∧ [p = $^\wedge$regnen ∨ p = $^\wedge$¬(regnen)]]

Intensor und Extensor können auch so aufgefasst werden, dass der Intensor ein λ-Operator für mögliche Situationen ist, während der Extensor die Funktionale Applikation einer Intension auf die reale Situation repräsentiert. Die indirekte Frage in (69.i) hat demzufolge das Denotat (69.ii):

(69) i. (Peter weiß,) wer lacht.
ii. p ist wahr, und es gibt ein x, so dass p die Proposition ist, dass x in s lacht.

[31] Vgl. Karttunen, *Syntax and semantics of questions*.

Entsprechend denotiert der mit *ob* eingeleitete indirekte Entscheidungsfragesatz in (70.i) die in (70.ii) charakterisierte Menge von Propositionen:

(70) i. (Peter weiß,) ob es regnet.
ii. p ist wahr, und p ist die Proposition, dass es in s regnet, oder p ist die Proposition, dass es in s nicht regnet.

Die Auffassung, dass das Denotat einer Frage eine Menge von Propositionen ist, ist in der gegenwärtigen semantischen Forschung eine Standardannahme.

Karttunen (1977) schlägt vor, die Semantik der indirekten Fragesätze aus Deklarativsätzen abzuleiten. Dazu verwendet er die sog. Proto-Frage-Regel, die dafür sorgt, dass die von einem Deklarativsatz ausgedrückte Proposition zu einer Menge von wahren Propositionen wird, welche die wahren Antworten auf die Frage darstellt:

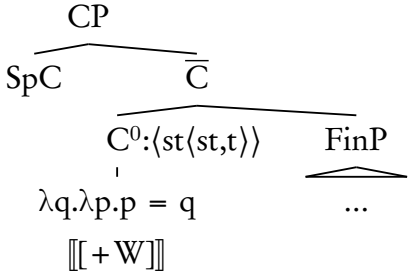
(71) Proto-Frage-Regel[32]
Wenn $\phi \in P_t$ (d. h., ϕ eine Phrase der Kategorie t) ist, dann ist $\ulcorner ?\phi \urcorner \in P_Q$ (d. h. $\ulcorner ?\phi \urcorner$ ist eine Phrase der Kategorie Q).
Wenn ϕ in ϕ' übersetzt wird, dann wird $\ulcorner ?\phi \urcorner$ in $\hat{p}\,[\check{}p \wedge p = \hat{}\phi']$ übersetzt.

Von Fintel & Heim[33] legen den Ansatz von Karttunen zugrunde und entwickeln eine sog. typengetriebene Semantik für multiple Fragen, welche die Voranstellung aller W-Phrasen an den linken Satzrand auf der Ebene der logischen Form theoretisch erzwingt. Wenn es in einer Situation s vier Bäume gibt, die Früchte tragen, so ist das Denotat der Frage in (72.i) die Menge der Propositionen in (72.ii), die hier als Funktion vom Typ ⟨st,t⟩ dargestellt ist:

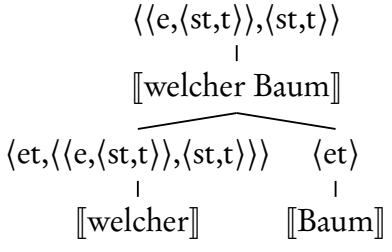
(72) i. Welcher Baum trägt Früchte?
ii. $\lambda p_{\langle st \rangle}.\exists x[x \in \{\text{Kirschbaum, Apfelbaum, Birnbaum, Quittenbaum}\} \wedge p = \text{tragen}(x, \text{Früchte})]$

Um die Bedeutung in (72.ii) abzuleiten, muss das W-Satztyp-Merkmal mit den vorangestellten W-Phrasen typentheoretisch kombiniert werden. Von Fintel & Heim (2001) wählen dazu den folgenden Ansatz, wobei das W-Satztyp-Merkmal wie in Karttunens Proto-Frage-Regel für die Erzeugung der Propositionsmenge zuständig ist, auf welche die W-Phrase funktional appliziert:

(73) i. W-Satztyp-Merkmal:



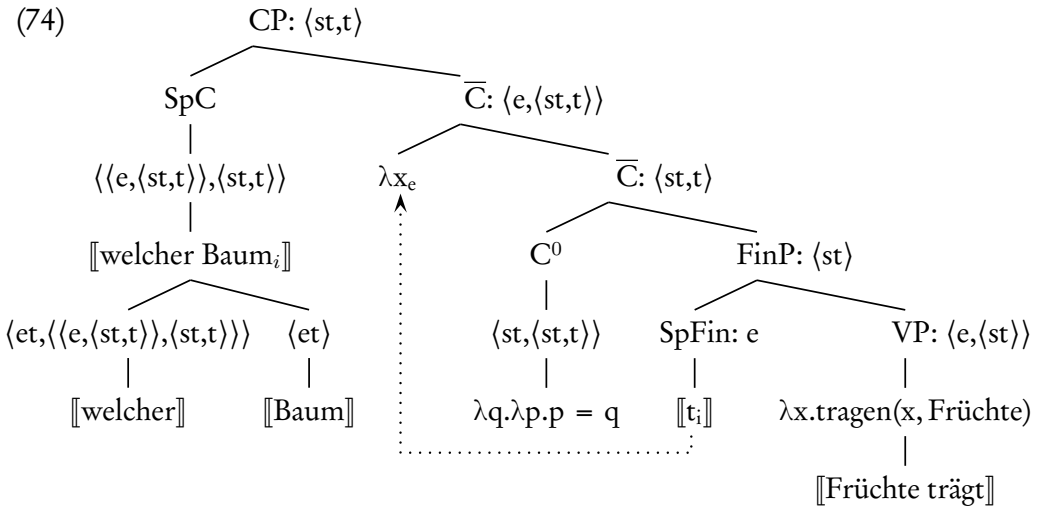
ii. W-Phrasen-Merkmal:



32 Karttunen, *Syntax and semantics of questions*, S.13.
33 Vgl. Fintel, Heim, *Karttunen on the interpretation and LF of interrogative clauses*.

Analog zur Quantorenanhebung bei generalisierten Quantoren in (45.ii) ergibt sich die syntaktische Struktur mit den passenden logischen Typen für (multiple) Fragekonstruktionen:

(74)



Ersetzt man in dem logischen Typ eines generalisierten Quantors ⟨et,t⟩ für Eigenschaftsmengen den Typ t durch den logischen Typ ⟨st,t⟩ für Propositionsmengen, so lassen sich die iterierten Quantorenanhebungen für generalisierte Quantoren, die in (45.ii) dargestellt wurden, in analoger Weise für W-Phrasen durchführen.

Aufgrund des spezifischen Denotats des W-Satztyp-Merkmals sind die W-Phrasen nur in einer Position zu interpretieren, die mit der Position des Satztyp-Merkmals komponiert wird, in situ jedoch nicht, weil ihr logischer Typ nicht passt. Da nur in dieser Position die interpretative Integration der W-Phrasen in die Gesamtstruktur möglich ist, wird (auch coverte) LF-Bewegung für W-Phrasen typentheoretisch erzwungen. Dieses Ergebnis ist theoretisch wünschenswert, weil die Distribution von W-Phrasen typologisch hinsichtlich der overten Realisierung zwar eine gewisse Variation zeigt, auf der Ebene LF befinden sich aber sprachübergreifend alle W-Phrasen in der linken Satzperipherie.

11 Fazit

Dieser kurze Ausflug in die Methoden zur Charakterisierung der Semantik natürlicher Sprachen sollte deutlich machen, wie die Verfahrensweisen der modernen Logik Wege aufzeigen können, um das Konzept der Bedeutung eines komplexen sprachlichen Ausdrucks einer Explikation zuzuführen und zugleich mit Hilfe der Kompositions- und Abstraktionsprinzipien die elementaren Einheiten der Bedeutungsstruktur zu bestimmen. Der Weg, den die moderne Semantiktheorie beschreitet, bringt weitere Klarheit in das alte Problem des Zusammenhangs zwischen Sprache und Denken und macht die Prinzipien sichtbar, nach denen die

beteiligten Systeme in ihrer je eigenen Spezifik organisiert sind und miteinander interagieren. Sie dient dem Zweck, diese wunderbare Fähigkeit von Menschen etwas besser zu verstehen.

Literatur

Ballweg, J.: *Die Semantik der deutschen Tempusformen*, Düsseldorf: Schwann 1988.

Barwise, J., Cooper, R.: Generalized Quantifiers and Natural Language. *Linguistics and Philosophy* 4.2, 1981, S. 159--219.

Bierwisch, M.: Thematic Roles -- Universal, Particular, and Idiosyncratic Aspects. In: *Semantic Role Universals and Argument Linking: Theoretical, Typological, and Psycholinguistic Perspectives*, hg. von I. Bornkessel, M. Schlesewsky, B. Comrie, A. D. Friederici. Berlin 2006, S. 89--126.

Chomsky, N.: *The Minimalist Program*. Cambridge, Mass. 1995.

Carnap, R.: *Meaning and Necessity*, Chicago 1956.

Church, A.: An Unsolvable Problem of Elementary Number Theory. In: *American Journal of Mathematics 58,2*, 1936, S. 345--363.

Church, A.: *The Calculi of Lambda-conversion*, Princeton 1941.

von Fintel, K.: *Restrictons on Quantifier Domains*. MIT-Diss. 1994.

von Fintel, K., Heim, I.: *Topics in Semantics -- Karttunen on the interpretation and LF of interrogative clauses*. Ms. 24.979 MIT 2001.

Frege, G.: Über Begriff und Gegenstand. In: *Vierteljahresschrift für wissenschaftliche Philosophie 16*, 1892, S. 192--205.

Frege, G.: Der Gedanke. In: *Logische Untersuchungen*, hg. von G. Patzig. Göttingen 1918/1993.

Haider, H., Bierwisch, M.: Steuerung kompositionaler Strukturen durch thematische Informationen. Finanzierungsantrag, Sonderforschungsbereich 340: Sprachtheoretische Grundlagen für die Computerlinguistik. Stuttgart 1989, S. 65--90.

Hamblin, C. L.: Questions in Montague Grammar. In: *Foundations of Language 10*, 1973, S. 41--53.

Heim, I., Kratzer, A.: *Semantics in Generative Grammar*, Oxford 1998.

Hintikka, J.: *Knowledge and Belief. An Introduction to the Logic of the Two Notions*, Ithaca, NY 1962.

Karttunen, L.: Syntax and semantics of questions. In: *Linguistics and Philosophy 1*, 1977, S. 3--44.

Kratzer, A.: Was „können" und „müssen" bedeuten können müssen. In: *Linguistische Berichte 42*, 1976, S. 1--28.

Kratzer, A.: An Investigation of the Lumps of Thought. In: *Linguistics and Philosophy* 12, 1989, S. 607--653.

Kratzer, A.: Modality. In: *Semantik. Ein internationales Handbuch der zeitgenössischen Forschung*, hg. von A. von Stechow & D. Wunderlich. Berlin, New York 1991, S. 639--650.

Kripke, S.: *Naming and necessity*, Cambridge, Mass. 1980.

Leibniz, G. W.: *Essais de Théodicée sur la Bonté de Dieu, la Liberté de l'Homme et l'Origine du Mal*. Amsterdam 1710.
(Dt. Übers.: (1996): Die Theodizee von der Güte Gottes, der Freiheit des Menschen und dem Ursprung des Übels. In: *Philosophische Schriften, Bd. 2*, 1996.

Lewis, D.: Ordering Semantics and Premise Semantics for Counterfactuals. In: *Journal of Philosophical Logic 10*, 1981, S. 217--234.

Lohnstein, H.: *Formale Semantik und natürliche Sprache*, Berlin, New York 2011.

Montague, R.: Formal philosophy. In: *Formal Philosophy. Selected Papers by Richard Montague*, hg. von R. Thomason. New Haven 1974.

Reichenbach, H.: *Elements of Symbolic Logic*, New York, London 1947.

Russell, B.: Mathematical Logic as Based on the Theory of Types. In: *American Journal of Mathematics, Vol. 30, No. 3*, 1908, S. 222--262.

de Saussure, Ferdinand, Bally, Hugo v. Ch., Sechehaye, A.: *Cours de linguistique gènèrale*. Èdition critique. (dt.: *Grundfragen der allgemeinen Sprachwissenschaft*) Wiesbaden, 1916/1971.

von Stechow, A.: *Schritte zur Satzsemantik*, Tübingen 2011.

Tarski, A.: Der Wahrheitsbegriff in den formalisierten Sprachen. In: *Logik-Texte. Kommentierte Auswahl zur Geschichte der modernen Logik*, hg. von K. Berka & L. Kreiser. Berlin 1944.

Wittgenstein, L.: *Tractatus logico-philosophicus, Logisch-philosophische Abhandlung*, Frankfurt am Main 1921/2003.