Arbeitspapiere des Sonderforschungsbereichs 340
"Sprachtheoretische Grundlagen der Computerlinguistik"
Bericht Nr. 95

Ein HPSG-Fragment des Deutschen
Teil 1: Theorie

Erhard Hinrichs, Detmar Meurers, Frank Richter,
Manfred Sailer, Heike Winhart

April 1997

Universität Stuttgart • Universität Tübingen
IBM Deutschland

Inhaltsverzeichnis

Kapitel 1. Erhard Hinrichs: "Ein HPSG-Fragment des Deutschen" . 7

Kapitel 2. Frank Richter: "Die Satzstruktur des Deutschen und die Behandlung langer Abhängigkeiten in einer Linearisierungsgrammatik. Formale Grundlagen und Implementierung in einem HPSG-Fragment" 13
1. Einleitung 14
2. Formale Grundlagen 18
2.1. RSRL 22
2.2. Einige weitere Definitionen 37
2.3. Anmerkungen zur AVM-Notation 42
3. Die Satzstruktur des Deutschen 46
3.1. Topologische Felder und Theorie der Wortstellung 48
3.1.1. VL bei einfachen Komplementierern 52
3.1.2. V1-Sätze 54
3.1.3. V2-Sätze 55
3.1.4. w-Phrasen als Komplementierer 57
3.1.5. Überblick 59
3.2. Die syntaktische Baumstruktur von Sätzen 64
3.3. Generalisierungen 74
4. Das Linearisierungsmodul der Grammatik 80
4.1. Zur Permutationsdomäne finiter Sätze 82
4.2. Die Phonologie von Phrasen 95
4.3. Die Gesamtstruktur des Linearisierungsmoduls 99
4.4. Die Openerregel 112
5. Die Theorie der langen Abhängigkeiten 117
5.1. Topikalisierung und w-Bewegung in einfachen Sätzen 120
5.2. Schnittstelle zu einer Illokutionstheorie 134
5.3. Verberstparenthesen statt Extraktion aus V2 140
5.4. Inseldomänen nichtlokaler Abhängigkeiten 164
5.4.1. Topikalisierungsdomänen 168

5.4.2. w-Bewegungsdomänen ... 175
5.4.3. Fazit ... 182
6. Literaturverzeichnis ... 185

Kapitel 3. Detmar Meurers: "Statusrektion und Wortstellung in kohärenten Infinitkonstruktionen des Deutschen" ... 189
1. Motivation ... 190
2. Kopf-Komplement Konstruktionen ... 191
2.1. Erwartete beobachtbare Eigenschaften ... 191
2.2. Der Verbkomplex als Kopf-Komplement Konstruktion ... 192
2.2.1. Syntax ... 192
2.2.2. Semantik ... 197
2.2.3. Zusammenfassung ... 197
3. Irreguläre Eigenschaften von kohärenten Verbkomplexen ... 198
3.1. Statusrektion ... 198
3.1.1. Ersatzinfinitiv ... 199
3.1.2. Ersatz-zu-infinitiv ... 207
3.1.3. Schlüsse aus der Statusrektion ... 208
3.2. Wortstellung ... 209
3.2.1. Oberfeldumstellung ... 209
3.2.2. Zwischenstellung ... 216
3.2.3. "Linksstellung" ... 218
3.2.4. Schlüsse aus der Wortstellung ... 220
3.3. Beziehungen zwischen Statuszuweisung und Wortstellung ... 220
3.4. Zur Semantik ... 221
4. Theoretische Deutung ... 223
4.1. Anmerkungen zur Literatur ... 223
4.2. Ein alternativer Vorschlag ... 226
4.2.1. Idee ... 226
4.2.2. Durchführung ... 231
4.2.3. Beispielanalysen ... 239
4.3. Offene Fragen ... 242
5. Literaturverzeichnis ... 244

Kapitel 4. Manfred Sailer: "Adjunkte in einer Linearisierungsgrammatik: Syntax, Semantik und Wortstellung" ... 249
1. Bemerkungen zu Syntax, Semantik und Wortstellung ... 250
1.1. Einführung ... 250
1.2. Unterspezifikation ... 252
1.3. Unterspezifikation in der Semantik ... 254
2. Motivation für eine andere Adjunktanalyse ... 257

3. Grundlagen ... 263
3.1. Einfache Modifikation ... 263
3.2. Mehrfachmodifikation ... 269
3.3. Bemerkung zur syntaktischen Repräsentation ... 274
4. Syntax verbaler Modifikation ... 274
4.1. Einführung ... 274
4.2. Modifikatoren als Vorfeldkonstituente ... 276
4.2.1. Empirische Motivation ... 276
4.2.2. Formale Definition ... 282
4.3. Modifikatoren im Mittelfeld ... 285
4.3.1. Unterspezifizierte lexikalische Semantik ... 286
4.3.2. Dritte Konstruktion ... 289
4.3.3. Neg-Raising ... 291
4.4. Modifikatoren als Teil des Vorfelds ... 292
4.5. Zusammenfassung ... 294
5. Syntax nominaler Modifikation ... 295
6. Chunkstruktur in HPSG ... 299
6.1. Chunkstruktur ... 299
6.2. Chunks und UPMs ... 304
6.3. Chunks und Extraposition ... 307
6.4. Zusammenfassung ... 312
7. Schlußbemerkung ... 313
8. Literaturverzeichnis ... 315

Kapitel 5. Heike Winhart: "Die Nominalphrase in einem HPSG-Fragment des Deutschen" ... 319
1. Zur Analyse deutscher Nominalphrasen ... 320
1.1. Einleitung ... 320
1.2. Syntaktischer Aufbau der DP nach Gallmann ... 321
1.2.1. Kategoriell komplexe Wortformen ... 322
1.2.2. Prinzipien ... 322
1.2.3. Zu den Wortformen D/N und D/A ... 324
1.2.4. Nomina des Typs N und D/N ... 325
1.2.5. Adjektive des Typs A und D/A ... 325
1.2.6. Artikelwörter des Typs A und D/A ... 327
1.2.7. Mögliche DP-Konfigurationen ... 329
1.3. Flexion in der DP ... 332
1.3.1. Pränominale Genitive ... 332
1.3.2. Mehrteilige Artikelwörter ... 333
1.3.3. Possessiva ... 335

1.4. Nominalisierungen ... 338
1.4.1. Nominalisierung attributiver Adjektive ... 338
1.4.2. Nominalisierung von Artikelwörtern ... 342
1.5. Eigennamen ... 343
2. Analyse in HPSG ... 346
2.1. Ontologie und relevante Prinzipien ... 346
2.1.1. Die Sorten ***nominal*** und ***def*** ... 346
2.1.2. Einige grundlegende Prinzipien ... 349
2.2. Beispiele und Lexikoneinträge ... 351
2.2.1. Nomina der Sorte ***noun*** ... 351
2.2.2. Nomina der Sorte ***sg-count*** ... 354
2.2.3. Nomina der Sorte ***mass-pl*** ... 356
2.2.4. Nomina der Sorte ***en*** ... 358
2.2.5. Nomina der Sorte ***pro*** ... 363
2.3. Anwendungen für Lexikalische Regeln ... 364
2.3.1. Nominalisierte Adjektive ... 364
2.3.2. Eigennamen ... 365
3. Präpositionalphrasen - Ein Analysevorschlag ... 367
3.1. Präpositionen als funktionale Kategorien ... 367
3.2. DP-Adverbiale ... 369
3.3. Die komplexe Kategorie P/D ... 371
3.4. Pronominaladverbien ... 373
4. HPSG-Analysen im Vergleich ... 375
4.1. Stand der Forschung ... 375
4.2. Allgemeine Konzeption in Netter (1994) ... 377
4.3. Vergleich mit dem eigenen Ansatz ... 381
5. Literaturverzeichnis ... 382

KAPITEL 2

Die Satzstruktur des Deutschen und die Behandlung langer Abhängigkeiten in einer Linearisierungsgrammatik. Formale Grundlagen und Implementierung in einem HPSG-Fragment

Frank Richter*

Zusammenfassung. Die vorliegende Arbeit beschäftigt sich mit den formalen Grundlagen der HPSG und ihrer Anwendung auf eine Linearisierungsgrammatik des Deutschen. Im Anfangsteil wird eine in Zusammenarbeit mit Manfred Sailer entwickelte erweiterte formale Sprache für HPSG vorgestellt, welche auf Arbeiten von Paul King aufbaut und dessen Formalismus um Variablen, Quantoren und Relationen anreichert. Im Anschluß daran wird auf dieser Grundlage ein Kernfragment des Deutschen definiert. Hauptgegenstand der Untersuchung sind finite Sätze und die Frage, wie sich die zentralen empirischen Generalisierungen in dieser Datendomäne in einer Linearisierungsgrammatik rekonstruieren lassen. Besondere Aufmerksamkeit wird dem Zusammenhang zwischen Komplementierern und finiten Verben in Verberst- und Verbzweitstellung geschenkt. Im letzten Kapitel wird eine relational formulierte Theorie der langen Abhängigkeiten in diesen Rahmen eingeordnet und gezeigt, wie barrierentheoretische Konzepte in die HPSG integriert werden können.

* Email: fr@sfs.nphil.uni-tuebingen.de
URL: http://www.sfs.nphil.uni-tuebingen.de/~fr/

Für Unterstützung in formalen Fragen danke ich Thilo Götz; für seinen niemals endenden Enthusiasmus für alle Fragen der HPSG und Unterstütung im Kampf mit Latex Detmar Meurers. Tilman Höhle hat das Manuskript freundlicherweise mehrfach gelesen und wichtige Verbesserungsvorschläge gemacht. Ohne Manfred Sailers beständige Anregungen und Beiträge sähe alles ganz anders aus.

1. Einleitung

In diesem Papier wird eine Linearisierungsgrammatik für ein Kernfragment des Deutschen vorgestellt. Dabei soll das linearisierungsgrammatische Konzept der strikten Trennung von phonologischer Wortfolge und syntaktischer Phrasenstruktur auf seine Tragfähigkeit in einem umfangreichen Fragment untersucht werden. Im Zentrum steht der Versuch einer vollständigen Formalisierung der betrachteten empirischen Domäne. Diese Vorgehensweise wird es erlauben, den bei der Umsetzung einer Grammatik unter der gewählten Perspektive zu treibenden technischen Aufwand genau zu bestimmen und festzustellen, welche Art von Generalisierungen zur deutschen Syntax in einer Linearisierungsgrammatik naheliegen und welche gegebenenfalls problematisch sind.

Grundlage der Formalisierung ist das Framework der Head-Driven Phrase Structure Grammar (HPSG), die in (Pollard und Sag 1994) anhand eines Fragments des Englischen vorgestellt und für die in (King 1994) ein mathematischer Formalismus vorgelegt wurde. Ausgehend von der Arbeit von King werden wir eine um Relationen und Quantoren erweiterte formale Sprache für HPSG vorstellen, in der das Fragment formalisiert wird.

Ein Kernfragment des Deutschen hat sich notwendigerweise mit den grundlegenden Eigenheiten des Satzbaus auseinanderzusetzen. Das heißt, es ist die Frage zu beantworten, wie Verberst-, Verbzweit- und Verbletztsätze strukturiert sind, welcher Zusammenhang zwischen ihnen besteht und welchen gegebenenfalls gemeinsamen Regularitäten sie unterworfen sind. Eine hinlänglich umfassende Theorie sollte außerdem in der Lage sein, die Distribution der drei Satztypen auf gemeinsame, allgemeine Gesetzmäßigkeiten zurückzuführen. Die Antworten, die eine Linearisierungsgrammatik auf diese Fragen gibt, unterscheiden sich in mehrfacher Hinsicht grundlegend von denen, die traditionelle, phrasenstrukturorientierte Ansätze gegeben haben. Insofern wird es interessant sein zu untersuchen, auf welche Weise in dem neuen Ansatz typische empirische Generalisierungen erfaßt werden.

Eine erste umfangreiche Linearisierungsgrammatik des Deutschen hat (Kathol 1995) im Rahmen der HPSG vorgestellt. Während wir in der vorliegenden Arbeit auf Erkenntnisse dieser ersten Studie zurückgreifen, werden wir sowohl hinsichtlich der syntaktischen Analyse als auch hinsichtlich der technischen Realisierung der Linearisierungskomponente der Grammatik signifikant andere Wege gehen. (Kathol 1995) orientiert sich hinsichtlich der Linearisierungstheorie eng an der klassischen Theorie des topologischen Feldermodells finiter Sätze des Deutschen (vgl. (Höhle 1986)).

Kathol integriert das Feldermodell ausdrücklich in die Grammatik, indem er den Feldern korrespondierende sprachliche Objekte einführt und diese dazu benutzt, die beobachteten Wortstellungsverhältnisse und andere wichtige Gesetzmäßigkeiten der Syntax finiter Sätze zu beschreiben. Im Gegensatz dazu werden wir argumentieren, daß den topologischen Feldern in einer Linearisierungsgrammatik des Deutschen keine Realität in Gestalt eigener sprachlicher Objekte zukommt. Vielmehr ergibt sich die empirisch beobachtete Oberflächenreihenfolge der Wörter in finiten Sätzen aus der Interaktion von Wortstellungsregeln mit charakteristischen lexikalischen Eigenschaften von bestimmten Klassen von Wörtern. Desgleichen gibt es keine ausgezeichnete Repräsentationsebene in sprachlichen Zeichen, auf denen Wortstellungregeln operieren. Sie finden außer in der phonologischen und damit empirisch beobachtbaren Wortfolge keinen Ausdruck in Form von Eigenschaften von Merkmalstrukturen.

Kernstück dieses alternativen Linearisierungsansatzes ist eine allgemeine Linearisierungstheorie, die in ihrer Anlage auf (Richter und Sailer 1995) zurückgeht. (Richter und Sailer 1995) definiert ein generelles Format zur Formalisierung beliebiger Wortstellungsregeln aus der linguistischen Literatur in der HPSG. Einer der Vorteile dieses Systems besteht darin, daß Wortstellungsgesetzmäßigkeiten weitgehend unabhängig von Annahmen über syntaktische Baumstrukturen formuliert werden könnnen. Diese Eigenschaft wird es uns erlauben, eine uniforme, alle drei Satztypen umfassende syntaktische Satzstruktur des Deutschen zu definieren und dennoch deren charakteristische Wortstellungsunterschiede aus einer ihrerseits einheitlichen Wortstellungstheorie abzuleiten.

Die sogenannten langen Abhängigkeiten sind ein Thema, dem in der Tradition der transformationellen Grammatik große Aufmerksamkeit widerfahren ist. In der HPSG-Literatur zum Deutschen wurde es bislang dagegen weitgehend vernachlässigt. Lange w-Bewegung und lange Topikalisierung wurden in der transformationellen Literatur mittels einer zumeist informell umschriebenen Relation move-α beschrieben, deren wesentliche Aufgabe es ist, das in eine bestimmte syntaktische Baumposition bewegte Element zu seiner Basisposition, in der eine koindizierte Spur angenommen wird, in Beziehung zu setzen. Durch die allgemeine syntaktische Theorie motivierte universelle Beschränkungen auf move-α erklären dann die Distribution von langen Abhängigkeiten.

Während viele Elemente und Einsichten dieser Analyse Eingang in die HPSG gefunden haben, werden lange Abhängigkeiten technisch üblicherweise nicht mittels einer Relation, sondern mittels Strukturidentitäten

in lokalen Bäumen beschrieben. Das heißt, es besteht in diesen Analysen zwischen bewegtem Element und Spur keine Relation im technischen Sinne. Die Beziehung zwischen den beiden wird vielmehr über eine Serie lokaler Strukturidentitäten vermittelt. Die Prinzipien der Grammatik, die diese Strukturidentitäten regieren, sowie Prinzipien zur Verteilung von bewegten Elementen und Spuren in Baumstrukturen erklären die Distribution von langen Abhängigkeiten. Im Gegensatz dazu werden wir eine HPSG-Analyse von langen Abhängigkeiten mit den traditionellen transformationellen Ideen zu einer Relation zwischen Antezedens und Spur verbinden. Die erweiterte formale Sprache für HPSG wird es uns erlauben, eine präzise Definition einer Relation move-α vorzuschlagen. Was die Distribution der empirisch beobachtbaren langen Abhängigkeiten anlangt, werden wir auf das Konzept der auf Barrierenbegriffen beruhenden Bewegungsdomänen zurückgreifen und zeigen, wie diese in HPSG formalisiert werden können.

Im Kontext der Analyse langer Abhängigkeiten verfolgen wir einen neueren Vorschlag aus der generativen Literatur zur sogenannten Extraktion aus Verbzweitsätzen. (Reis 1995) argumentiert in einer sorgfältigen empirischen Untersuchung überzeugend, daß es im Deutschen keine Bewegung aus Verbzweitsätzen gibt. Reis weist nach, daß es sich bei den Fällen, die in der generativen Literatur traditionell als Extraktion aus Verbzweitsätzen analysiert wurden, in Wirklichkeit um parenthetische Konstruktionen handelt. Durch die Adaptierung dieses Vorschlags und die Formalisierung einer Verberstparenthesenanalyse für die einschlägigen Konstruktionen ergibt sich eine erhebliche Vereinfachung der Theorie der langen Abhängigkeiten im Deutschen, die insbesondere ohne Zwischenspuren für syntaktische Bewegung auskommt.

Das Papier ist folgendermaßen strukturiert. Kapitel 2 enthält eine kurze Zusammenfassung der formalen Grundlagen, auf denen unsere Grammatik basiert. Dabei folgen wir in den Grundzügen den Vorschlägen von (King 1994). Allerdings erweitern wir die dort vorgestellte formale Sprache um Quantoren und Relationen, wie sie in der linguistischen Literatur zur HPSG weitgehend implizit angenommen werden. Während die Lektüre von Kapitel 2 für ein intuitives Verständnis des folgenden nicht unbedingt erforderlich ist, legt es doch die notwendigen formalen Grundlagen, um die Details einer Linearisierungsgrammatik und relationalen Theorie der langen Abhängigkeiten zu präzisieren. Ohne eine Explikation der Frage, was Relationen in einer Grammatik bedeuten, die sprachliche Objekte mittels Merkmalstrukturen modelliert, müßte der theoretische Gehalt der Verwendung solcher Relationen notwendig mysteriös bleiben.

Kapitel 3 führt in die zentralen Annahmen zur Satzstruktur ein und enthält die Prinzipien und wichtigsten Lexikoneinträge, aus deren Interaktion die anvisierten Strukturen hervorgehen. Im Anschluß an ihre Präsentation werden die Generalisierungen betrachtet, die sich aus der Neuorganisation der Satzstrukturanalyse gewinnen lassen, und es wird argumentiert, daß sie wesentliche Einsichten in die Syntax finiter Sätze widerspiegelt.

In Kapitel 4 wenden wir uns schließlich der Frage zu, wie aus der Satzstruktur von Kapitel 3 die empirisch beobachtbare Wortstellung abgeleitet wird. Zu diesem Zweck wird eine allgemeine Linearisierungstheorie für HPSGrammatiken skizziert, mit deren Hilfe beliebige Wortstellungsregeln formalisiert werden können. Es wird gezeigt, daß unter diesen Voraussetzungen eine einzige Regel genügt, um die Wortstellungsgesetzmäßigkeiten an der linken Peripherie finiter Sätzen zu erfassen.

Im letzten Kapitel wenden wir uns schließlich der Theorie der langen Abhängigkeiten zu. move-α wird als Relation zwischen sprachlichen Zeichen definiert. Diese Formalisierung wird der traditionellen HPSG-Analyse gegenübergestellt, die sich für die Zwecke der Bewegungstheorie einer aufwendigen Merkmalsgeometrie und mehrerer grammatischer Prinzipien bedient. Im Anschluß daran wird die Verberstparenthesenanalyse aus (Reis 1995) in HPSG integriert und werden zwei Barrierenbegriffe für w-Bewegung und Topikalisierung definiert, aus denen sich entsprechende Domänenbegriffe gewinnen lassen. Diese Domänen beschränken im vorgestellten Kernfragment des Deutschen das Auftreten langer syntaktischer Bewegung.

2. Formale Grundlagen

Das Standardwerk zur HPSG, (Pollard und Sag 1994), definiert selbst keine formale Sprache für HPSGrammatiken, sondern begnügt sich mit einer ausführlichen informellen Beschreibung der Eigenschaften, die eine den Intentionen der Autoren entsprechende Beschreibungssprache aufzuweisen hätte. (King 1994) diskutiert die von Pollard und Sag gewünschten Anforderungen eingehend, vergleicht diese Eigenschaften mit bestehenden Formalismen ((Rounds und Kasper 1986), (Johnson 1988), (Carpenter 1992)) und entwickelt auf der Basis dieser Diskussion den K-Formalismus, der sich als logischer Grundlage bis auf einige terminologische Änderungen im wesentlichen der in (King 1989) vorgestellten, speziell auf die Bedürfnisse von HPSG zugeschnittenen Speciate Re-entrant Logic (SRL) bedient. (King 1994) verwendet SRL, um die Begriffe Type und Token und ihre Beziehung zueinander zu definieren und zu formalisieren, was es für eine Theorie bedeutet, einen Type zuzulassen. Ziel des K-Formalismus ist, Pollards und Sags informelle Diskussion der Begriffe Type und Token formal zu explizieren und mathematisch greifbar zu machen. Des weiteren konstruiert (King 1994) ein mathematisches Modell für den K-Formalismus und zeigt, daß die Elemente dieses Modells den als modellierende Strukturen konzipierten Merkmalstrukturen von (Pollard und Sag 1994) entsprechen. Damit ist gezeigt, daß der K-Formalismus weitgehend den Anforderungen von Pollard und Sag nachkommt.

Modelltheoretische Überlegungen führen allerdings zu der Erkenntnis, daß der K-Formalismus alleine den Bedürfnissen der linguistischen Theoriebildung nicht genügt. Linguisten sind unter anderem an den Grammatikalitätsvoraussagen ihrer Grammatik interessiert. Unter der Perspektive einer Beschreibungslogik beschreiben sie mit den Prinzipien der Grammatik die Menge der linguistischen Objekte, die als wohlgeformt gelten. Die generelle Idee des K-Formalismus ist dabei die, daß in den Modellen der Grammatik jedes Objekt sämtliche Prinzipien erfüllt; das gleiche gilt für alle Teilobjekte eines Objekts in den Modellen. Zum Beispiel muß bei Merkmalstrukturen in einem Modell, die Sätzen entsprechen, nicht nur der oberste Knoten im syntaktischen Baum die einschlägigen Prinzipien erfüllen, sondern sie müssen auch für alle Konstituenten, die Teil des Satzes sind, gelten.

Dies reicht für linguistische Zwecke jedoch noch nicht aus. Zwar ist aufgrund der genannten Bedingungen jedes Objekt in einem Modell ein zulässiges Objekt, da es den vom Linguisten formulierten Bedingungen

genügt. Insofern ist es durch die Betrachtung ausgewählter Modelle feststellbar, ob der Linguist etwa wohlgeformte Objekte der zu beschreibenden Sprache irrtümlich ausschließt, das heißt, ob die Grammatik untergeneriert. Indessen ist es nicht jedem beliebigen Modell der Theorie anzusehen, ob die Grammatik im transformationellen Sinne übergeneriert, das heißt, Objekte zuläßt, die nicht wohlgeformten Objekten der beschriebenen natürlichen Sprache entsprechen. Dies liegt daran, daß durch nichts sichergestellt wird, daß die verschiedenen Modelle der Theorie auch tatsächlich Instanzen *aller* Objekte enthalten, die von ihr potentiell lizenziert werden. Wenn der Linguist eine Grammatik schreibt, ist er jedoch vom sprachwissenschaftlichen Gesichtspunkt genau an denjenigen Modellen interessiert, die exakt die von ihm beschriebene Sprache enthalten, und diese sollen von der Grammatik eindeutig charakterisiert werden. Andererseits möchte er die Grammatik alleine mit den Mitteln der Beschreibungssprache formulieren und sich nicht um zusätzliche Bedingungen für die Interpretation der linguistischen Theorie kümmern müssen.

Um dieser Anforderung nachzukommen, ist es im Rahmen von SRL notwendig, auf der Basis von (King 1994) eine angemessene Modelltheorie für HPSGrammatiken zu entwickeln.[1] Dabei ist der Leitgedanke der, eine Klasse von Modellen zu definieren, so daß bei einer gegebenen Menge von linguistischen Prinzipien ein Modell aus dieser Klasse Instanzen aller wohlgeformten Objekte enthält, die von diesen Prinzipien beschrieben werden. (King 1995) nennt Modelle, die diese Bedingung erfüllen, *exhaustive Modelle* und zeigt, daß die exhaustiven Modelle einer gegebenen Menge von Prinzipien ununterscheidbar sind. Das bedeutet, daß diese Modelle zwar verschieden groß sein können und es im allgemeinen keinen Homomorphismus zwischen zwei exhaustiven Modellen gibt, sie sich jedoch bezüglich der linguistischen Fragen von Über- und Untergenerierung gleich verhalten.

Wenn der K-Formalismus zusammen mit der angedeuteten Modelltheorie den Anforderungen von (Pollard und Sag 1994) nachkommt, stellt sich die Frage, weshalb hier eine erweiterte formale Sprache für HPSG entwickelt werden soll. In verschiedenen neueren Arbeiten in HPSG, unter anderem etwa in (Kathol 1995), (Kasper 1994) und (Pollard, Levine, und Kasper 1994), zeigt sich, daß Linguisten in der standardmäßig verwendeten AVM-Beschreibungssprache zunehmend freien Gebrauch von Relationen machen. Während (Pollard und Sag 1994, S.21) bereits andeutet, daß einige Relationen wie unter anderem “append” Teil der Beschreibungssprache sein sollten, hat sich ihre Verwendung mittlerweile in der

[1] Erste Ansätze dazu finden sich in (King 1995).

Praxis erheblich ausgeweitet. Mit der zunehmenden Verwendung einer Vielzahl von Relationen kann es keine sinnvolle Strategie sein, ihre Bereitstellung an einen zugrundeliegenden, zumeist nur als Desideratum angemeldeten Formalismus zu delegieren.[2] Was vielmehr vonnöten scheint, ist eine Erweiterung des Formalismus, der es Linguisten erlaubt, generell all diejenigen Relationen selbst zu definieren, die für ihre jeweiligen Zwecke notwendig erscheinen. Dies leistet SRL nicht.

Ein weiterer Unterschied zwischen der linguistischen Praxis und den Werkzeugen, die SRL bereitstellt, besteht darin, daß Linguisten für ihre Generalisierungen lebhaften Gebrauch von einer partiellen Ordnung auf Sorten, auch Sortenhierarchie genannt, machen, die nicht direkt Bestandteil von SRL ist. Zwar weist bereits (King 1989, S.47–48) darauf hin, daß SRL implizit so strukturiert ist, daß sie so verwendet werden kann, als verfügte sie über eine Sortenhierarchie. Dies mag allerdings für den Linguisten, der explizit reichen Gebrauch von einer Sortenhierarchie macht, etwas unbefriedigend sein. Daher weichen wir in der folgenden Sprache insofern von SRL ab, als wir die Sortenhierarchie mit den in (Pollard und Sag 1994) geforderten Bedingungen in unsere Definitionen integrieren.[3] Während dies eine triviale Übung ist, verhält es sich bei der Frage der Relationen anders. Hier stellen sich insbesondere zwei Probleme:

Erstens ist es nicht damit getan, SRL um eine Syntax und Semantik von Relationen zu erweitern. Um den Bedürfnissen einer Grammatiktheorie gerecht zu werden, muß die Erweiterung in einer Weise geschehen, daß die zugehörige Modelltheorie erhalten bleibt oder für eine Entsprechung Sorge getragen wird. Es muß also weiterhin möglich sein, eine Klasse von Modellen anzugeben, welche die empirischen Vorhersagen der linguistischen Prinzipien widerspiegelt.

Zweitens zeigt sich in der Literatur, daß Linguisten oftmals Prinzipien angeben, bei denen eine Relation offenbar im Skopus von Negation steht. Ein typisches Beispiel dafür wäre eine Formulierung, die eine bestimmte Wohlgeformtheitsbedingung davon abhängig macht, daß eine Konstituente ein Pronomen enthält. Gewöhnlich wird man dies nicht durch die

[2] Vergleiche zur Strategie einer Logik, die eine begrenzte Anzahl von Relationen zur Verfügung stellt, zum Beispiel (Reape 1989).

[3] Eine alternative Möglichkeit, die mathematisch eleganter wäre, bestünde darin, die Allgemeinheit von SRL hinsichtlich der Sortenhierarchie beizubehalten und letztere dem Linguisten als 'syntaktischen Zucker' verfügbar zu machen. Da wir im vorliegenden Papier die Anwendung auf eine linguistische Grammatik im Auge haben, sparen wir uns den zusätzlichen definitorischen Aufwand und gehen den am Zweck orientierten Weg der in die formale Sprache integrierten Sortenhierarchie.

Einführung eines neuen Attributes und Perkolationsmechanismus handhaben mögen, die lediglich dem Zweck dienen, das Vorhandensein eines Pronomens innerhalb einer Konstituente zu signalisieren. Bei der Verwendung von Relationen im Antezedens von Prinzipien und damit im Skopus von Negation stellt sich in erster Linie die Frage, von welchen Intuitionen solche Prinzipien geleitet sind. Die formale Sprache und Modelltheorie sollten wenn möglich so gestaltet werden, daß sie diese Intuitionen erfassen. Kein uns bekannter Formalismus für HPSG ist bisher in der Lage, mit diesem Problem umzugehen.

Die Sprache, die wir im folgenden definieren, wollen wir *relationale SRL* (RSRL) nennen. Ausgangspunkt der RSRL ist die Feststellung, daß sich alle Relationen, die Linguisten verwenden, scheinbar generell in sogenannte Junk-Slots oder rekursive Sortendefinitionen übersetzen lassen,[4] eine Technik, die auf (Ait-Kaci 1984) zurückgeht, und zuerst von (Carpenter 1992) auf HPSG angewandt wurde. Ihre praktische Brauchbarkeit für die Definition der in einer Linearisierungsgrammatik benötigten Relationen wird in (Richter und Sailer 1995) erfolgreich exploriert. Die Idee dabei ist, eine Sortenhierarchie und rekursive Prinzipien zu definieren, welche die Relation simulieren. Der problematische Zug dieser Herangehensweise besteht darin, daß eine Vielzahl an Objekten in das linguistische Universum eingeführt wird, die lediglich rechnerischen Zwecken dienen und keiner linguistischen Interpretation im Sinne der übrigen Objekte des Modells der Grammatik zugänglich sind. Die grundsätzliche technische Möglichkeit, die Bedürfnisse des Linguisten mit rekursiven Sortendefinitionen zu befriedigen, weist jedoch auf den Gedanken, daß die zu definierenden Relationen in ihrer Ausdrucksstärke diesen rekursiven Sorten entsprechen sollten. Außerdem ist es naheliegend zu vermuten, daß eine Sprache in der Art und von der Ausdrucksstärke von SRL im Prinzip genau den Bedürfnissen des HPSG treibenden Linguisten entspricht.

Während im vorliegenden Zusammenhang nicht genügend Raum ist, um die Details dieser Überlegungen zu diskutieren, spiegelt die folgende Definition von RSRL ihre Konsequenzen wider. Die RSRL baut in ihrer Struktur direkt auf SRL auf. Der mit (King 1994) vertraute Leser wird ohne weiteres die Parallelität der Definitionen nachvollziehen können. Einige der Besonderheiten, die von Standarddefinitionen abweichen, haben ihre Ursache in der linguistischen Anwendung, die wir für RSRL im Auge haben. Das heißt, es wird versucht, mit möglichst sparsamen Mitteln dem in der Literatur zur HPSG zu beobachtenden Bedarf an Relationen nachzukommen.

[4] Eine Ausnahme hiervon bildet die Verwendung von Relationen in (Kathol 1995).

2.1. RSRL[5]. Aus der Perspektive des Linguisten besteht eine HPS-Grammatik im wesentlichen aus drei Teilen: Erstens aus einer Signatur, die das linguistische Universum mit Hilfe von Sorten, einer Sortenhierarchie, Attributnamen und Relationsnamen deklariert und strukturiert. Zweitens aus einer Menge von Klauseln, mit denen die Bedeutung der verwendeten Relationen definiert wird. Und drittens aus einer Menge von Prinzipien, welche Wohlgeformtheitsbedingungen für die Menge der zulässigen linguistischen Objekte im Universum darstellen. Vom technischen Standpunkt aus sind Klauseln nichts anderes als Prinzipien und werden daher genauso behandelt wie jene. Die Signatur stellt die syntaktischen Mittel für die Beschreibungen bereit, aus denen die Prinzipien bestehen. Daneben sind Variablen Teil der Beschreibungssprache.

1 Definition. Var *is a countably infinite set of symbols.*

Wir nennen jedes Element von Var eine Variable. Keines der Symbole $\exists$, $\forall$, : , $\sim$, $\approx$, $\neg$, [,], $\wedge$, $\vee$, $\rightarrow$ und $\leftrightarrow$ ist eine Variable.

2 Definition. Σ *is a signature iff*

Σ *is a septuple* $\langle \mathfrak{G}, \sqsubseteq, \mathfrak{S}, \mathfrak{A}, \mathfrak{F}, \mathfrak{R}, Ar \rangle$,

$\langle \mathfrak{G}, \sqsubseteq \rangle$ *is a finite partial order,*

$$\mathfrak{S} = \left\{ \sigma \in \mathfrak{G} \;\middle|\; \begin{array}{l} \text{for each } \sigma' \in \mathfrak{G}, \\ \text{if } \sigma' \sqsubseteq \sigma \text{ then } \sigma = \sigma' \end{array} \right\},$$

elist, list, nelist $\in \mathfrak{G}$, *elist, nelist* $\in \mathfrak{S}$, *elist* $\sqsubseteq$ *list, nelist* $\sqsubseteq$ *list, for each* $\sigma \in \mathfrak{G} \setminus \{$*elist, nelist, list*$\}, \sigma \not\sqsubseteq$ *list,*

$\mathfrak{A}$ *is a set,*

$\textsc{first}, \textsc{rest} \in \mathfrak{A}$,

$\mathfrak{F}$ *is a partial function from the Cartesian product of* $\mathfrak{G}$ *and* $\mathfrak{A}$ *to* $\mathfrak{G}$, *and*

for each $\sigma_1 \in \mathfrak{G}$, *for each* $\sigma_2 \in \mathfrak{G}$ *and for each* $\alpha \in \mathfrak{A}$,

if $\mathfrak{F}\langle \sigma_1, \alpha \rangle$ *is defined and* $\sigma_2 \sqsubseteq \sigma_1$

then $\mathfrak{F}\langle \sigma_2, \alpha \rangle$ *is defined and* $\mathfrak{F}\langle \sigma_2, \alpha \rangle \sqsubseteq \mathfrak{F}\langle \sigma_1, \alpha \rangle$,

for each $\alpha \in \mathfrak{A}$, $\mathfrak{F}\,\langle elist, \alpha \rangle$ *is undefined,*

$\mathfrak{F}\,\langle nelist, \textsc{rest} \rangle = list$,

$\mathfrak{F}\,\langle nelist, \textsc{first} \rangle$ *is defined,*

for each $\alpha \in \mathfrak{A} \setminus \{\textsc{first}, \textsc{rest}\}$, $\mathfrak{F}\,\langle nelist, \alpha \rangle$ *is undefined,*

$\mathfrak{R}$ *is a finite set, and*

Ar is a total function from $\mathfrak{R}$ *to* $\mathbb{N}^+$.

Wir nennen die Elemente von $\mathfrak{G}$ Sorten, $\langle \mathfrak{G}, \sqsubseteq \rangle$ Sortenhierarchie, die Elemente der Untermenge $\mathfrak{S}$ von $\mathfrak{G}$ Species, die Elemente von $\mathfrak{A}$ Attribute, die Elemente von $\mathfrak{R}$ Relationssymbole, $\mathfrak{F}$ die Angemessenheitsfunktion und *Ar* die Stelligkeitsfunktion. Die Symbole $\exists$, $\forall$, : , $\sim$, $\approx$, $\neg$, [,], $\wedge$, $\vee$, $\rightarrow$ und $\leftrightarrow$ sind Beschreibungen vorbehalten, und wir fordern, daß keines von ihnen und kein Element von Var eine Sorte, ein Attribut oder ein Relationssymbol ist. In der Literatur werden Sorten und Species oftmals auch Typen genannt. Species heißen oft auch maximal spezifische Sorten oder maximal spezifische Typen.

Die endliche partielle Ordnung $\langle \mathfrak{G}, \sqsubseteq \rangle$ stellt die Sortenhierarchie zur Verfügung. In Übereinstimmung mit (Pollard und Sag 1994, S.395) sagen wir, daß eine Sorte genau dann maximal spezifisch oder eine Species ist, wenn sie keine Untersorten hat. Ohne Beschränkung der Allgemeinheit nehmen wir ferner an, daß die Sortenhierarchie ein Topelement (eine Wurzel) hat. Eine Sorte ist die Wurzel der Sortenhierarchie genau dann, wenn sie Obersorte aller Sorten ist.

Mit der Angemessenheitsfunktion $\mathfrak{F}$ wird festgelegt, welche Attribute für welche Sorten definiert sind und zu welchen Sorten man mit den definierten Attributen gelangt. In der sprachwissenschaftlichen Literatur wird dies manchmal Attributdeklarationen und manchmal Angemessenheitsbedingungen genannt. Ist $\mathfrak{F}\langle \sigma, \alpha \rangle$ für eine Sorte σ und ein Attribut α definiert, dann sagen wir, daß α für σ angemessen ist. Die Bedingung auf die Definition von $\mathfrak{F}$ stellt sicher, daß die Angemessenheit von Attributen von Obersorten auf Untersorten vererbt wird, und die Werte von $\mathfrak{F}$ bei Vererbung von Attributen spezifischer werden. Das heißt, wenn α für eine Sorte σ angemessen ist, dann ist α auch für alle Untersorten von σ angemessen. Außerdem ist für ein gegebenes α, das für σ angemessen ist,

[5]Die in diesem Abschnitt vorgestellte logische Sprache wurde in Zusammenarbeit mit Manfred Sailer entwickelt. Für Anmerkungen, Anregungen und unendlich viel Geduld danken wir Mike Calcagno, Tom Cornell, Thilo Götz, Tilman Höhle, Hans-Peter Kolb, Detmar Meurers, Uwe Mönnich, Frank Morawietz und Carl Pollard. Unser ganz besonderer Dank gilt Paul King für wertvolle Hinweise und Unterstützung bei der endgültigen Formulierung von RSRL.

und eine Untersorte σ' von σ der Wert von $\mathfrak{F}$ auf $\langle\sigma', \alpha\rangle$ eine (eventuell unechte) Untersorte von $\mathfrak{F}\langle\sigma, \alpha\rangle$.

Die Stelligkeitsfunktion ordnet jedem Relationssymbol eine positive natürliche Zahl zu und gibt damit die Stelligkeit der Relation an.

Die Definition der Signatur schließt die Definition von Sorten und Attributen für die Behandlung von Listen mit ein. In traditioneller Weise sind dabei *elist* für die leere Liste und *nelist* für die nichtleere Liste maximal spezifische unmittelbare Untersorten der Listensorte *list*. Während für *nelist* genau die beiden Attribute FIRST für ein Element der Liste und REST für den Rest der Liste angemessen sind, ist für *elist* kein Attribut angemessen. Listen sind eine der ausgezeichneten rekursiven Datenstrukturen in Grammatiken und von zentraler Wichtigkeit bei der Definition der in HPSGrammatiken üblicherweise verwendeten Relationen. Aus diesem Grund wird ihnen hier ein besonderer Status eingeräumt, indem ihr Vorhandensein vorausgesetzt wird. Dies wird es uns später ermöglichen, besondere Mechanismen für die Verwendung von Listen zur Verfügung zu stellen.

3 Definition. *For each set* M, $\overline{M} = (M \cup M^*)$.

Wir nennen die Teilmenge M^* einer Menge $\overline{M}$ die Menge der Ketten von Elementen von M und notieren eine Kette der Länge n als $\langle m_1, \ldots, m_n\rangle$.

4 Definition. *For each signature* Σ, I *is a* Σ*-interpretation iff*

I *is a quadruple* $\langle U, S, A, P\rangle$,

U *is a set of objects*,

S *is a total function from* U *to* $\mathfrak{S}$,

A *is a total function from* $\mathfrak{A}$ *to the set of partial functions from* U *to* U,

for each $\alpha \in \mathfrak{A}$ *and each* $u \in U$,

> *if* $A(\alpha)(u)$ *is defined*
>
> *then* $\mathfrak{F}\langle S(u), \alpha\rangle$ *is defined, and* $S(A(\alpha)(u)) \sqsubseteq \mathfrak{F}\langle S(u), \alpha\rangle$, *and*

for each $\alpha \in \mathfrak{A}$ *and each* $u \in U$,

> *if* $\mathfrak{F}\langle S(u), \alpha\rangle$ *is defined then* $A(\alpha)(u)$ *is defined,*

P *is a total function from* $\mathfrak{R}$ *to the powerset of* $\overline{U}^+$, *and for each* $r \in \mathfrak{R}$,

$$P(r) \subseteq \overline{U}^{Ar(r)}.$$

U ist die Menge der Objekte des Universums. Wir nennen S die Specieszuweisungsfunktion, A die Attributinterpretationsfunktion und P die Relationsinterpretationsfunktion.

Die Specieszuweisungsfunktion stellt sicher, daß jedem Objekt im Universum genau eine Species zugewiesen wird. Wir sagen, daß ein Objekt von einer bestimmten Species ist oder eine bestimmte Species hat. Die Annahme, daß jedes Objekt von einer maximal spezifischen Sorte, also von einer Sorte aus $\mathfrak{S}$, ist und nicht etwa von einer beliebigen Sorte, also von einer beliebigen Sorte aus $\mathfrak{G}$, sein darf, wird in der Literatur zuweilen die 'closed world assumption' genannt. Das Universum wird von S in eine endliche Menge disjunkter Kategorien partitioniert.

Die Attributinterpretationsfunktion A gibt den Attributsnamen eine Bedeutung, indem sie ihnen partielle Funktionen von U in U zuweist. Dies läßt sich so verstehen, daß die Attribute die Menge der zulässigen Übergänge zwischen Objekten des Universums benennen. Die Zulässigkeit ist abhängig von der Species der betrachteten Objekte und der Signatur Σ. Dieser Zusammenhang kommt in den beiden Bedingungen auf die Definition von A zum Ausdruck, welche die erforderliche Korrelation zwischen Attributinterpretationsfunktion und Sortenhierarchie und Angemessenheitsfunktion erzwingen.

Die erste der Bedingungen besagt, daß, wenn die durch ein Attribut α benannte partielle Funktion auf einem Objekt u des Universums definiert ist, die Angemessenheitsfunktion auf der Species von u und dem Attribut α definiert sein muß. Außerdem muß das Objekt u', welches der Wert der von α benannten Funktion auf u ist, von einer Species sein, die gemäß $\mathfrak{F}$ für die Species von u und α angemessen ist. Mit anderen Worten muß u' von einer Species sein, die in der Sortenhierachie unterhalb von $\mathfrak{F}\langle S(u), \alpha\rangle$ liegt.

Umgekehrt fordert die zweite Bedingung, daß, falls ein Attribut α für die Species eines Objekts u angemessen ist, die partielle Funktion, die α in der Σ-Interpretation I modelliert, auch tatsächlich auf u definiert sein muß. Das bedeutet, daß jedes Attribut α eine einstellige partielle Funktion von Objekten auf Objekte benennt, die auf allen oder keinen Objekten einer maximal spezifischen Sorte definiert ist.

Die Relationsinterpretationsfunktion P ordnet jedem Relationssymbol, das durch die Stelligkeitsfunktion Ar als n-stellig ausgewiesen wird, eine Menge von n-Tupeln aus Objekten des Universums und Ketten von Elementen des Universums zu. Durch die Stelligkeitsfunktion Ar und die Wertemenge von P wird sichergestellt, daß es keine nullstelligen Relationen gibt.

5 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$, $\widehat{S}$ is the total function from $\overline{U}$ to $\mathfrak{S}$ such that*

for each $u \in U$, $\widehat{S}(u) = S(u)$,

for each $u \in U^*$,

$$\widehat{S}(u) = \begin{cases} \mathit{elist} & \text{if } u \in U^0, \\ \mathit{nelist} & \text{otherwise.} \end{cases}$$

$\widehat{S}$ ist eine Erweiterung der Specieszuweisungsfunktion S. Auf den Objekten des Universums entspricht $\widehat{S}$ genau S. Darüber hinaus enthält die Domäne von $\widehat{S}$ jedoch die Menge von Ketten von Elementen des Universums. Bei der Definition von $\widehat{S}$ auf Elementen von U^* machen wir Gebrauch von der Tatsache, daß Listensorten fester Bestandteil der Signatur sind, und verwenden sie, um über Ketten wie über Listen zu sprechen. Der leeren Kette ordnet $\widehat{S}$ daher *elist* zu, nichtleeren Ketten *nelist*.

6 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$, $\widehat{A}$ is the partial function from $\mathfrak{A}$ to the set of partial functions from $\overline{U}$ to $\overline{U}$ such that for each $\alpha \in \mathfrak{A}$*

for each $u \in U$, $\widehat{A}(\alpha)(u) = A(\alpha)(u)$,

for each $\langle u_0, \ldots, u_n\rangle \in U^+$,

$$\widehat{A}(\alpha)(\langle u_0, \ldots, u_n\rangle) = \begin{cases} u_0 & \text{if } \alpha = \textsc{first}, \\ \langle u_1, \ldots, u_n\rangle & \text{if } \alpha = \textsc{rest}, \\ \mathit{undefined} & \text{otherwise,} \end{cases}$$

$\widehat{A}(\alpha)(\langle\rangle)$ is undefined.

$\widehat{A}$ ist eine Erweiterung der Attributinterpretationsfunktion A. Im Wertebereich der Menge der partiellen Funktionen von U in U interpretiert $\widehat{A}$ jedes Attribut α genau wie die Attributinterpretationsfunktion. Abgesehen von den beiden in der Signatur explizit vorgegebenen Listenattributen FIRST und REST müssen alle Attribute als partielle Funktionen von U in U interpretiert werden.

Für FIRST und REST gibt es jeweils noch eine weitere Möglichkeit. Im gegenüber A erweiterten Wertebereich der Funktion $\widehat{A}$ wird FIRST als diejenige Funktion von den nichtleeren Ketten in U interpretiert, welche eine Kette auf ihr erstes Element abbildet. Parallel dazu wird REST im Wertebereich der Menge der partiellen Funktionen von nichtleeren Ketten in Ketten als die Funktion interpretiert, welche einer nichtleeren Kette die um das erste Element verkürzte Kette zuordnet. Parallel zu $\widehat{S}$ für den Fall von Sorten macht $\widehat{A}$ Gebrauch von der Tatsache, daß Listenattribute fester Bestandteil der Signatur sind, und nutzt sie aus, um mit diesen Attributen über Ketten wie über Listen zu sprechen.

7 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$,*

$$\mathit{Ass}_I = \overline{U}^{\mathrm{Var}} \text{ is the set of variable assignments in } I.$$

Wir definieren die Menge der Variablenzuweisungen in einer Σ-Interpretation so, daß Variablen sowohl Objekte des Universums als auch Ketten von Elementen des Universums zugewiesen werden können. Die Elemente von Ass_I nennen wir Belegungen in I.

8 Definition. *For each signature Σ, $\mathfrak{T}^\Sigma$ is the smallest set such that*

$: \in \mathfrak{T}^\Sigma$,

for each $v \in \mathit{Var}$, $v \in \mathfrak{T}^\Sigma$, and

for each $\alpha \in \mathfrak{A}$ and each $\tau \in \mathfrak{T}^\Sigma$, $\tau\alpha \in \mathfrak{T}^\Sigma$.

Wir nennen die Elemente von $\mathfrak{T}^\Sigma$ Σ-Terme. Ein Σ-Term besteht aus dem ausgezeichneten Symbol $:$ oder einer Variablen und einer (möglicherweise leeren) Folge von Attributen. Σ-Terme sind von links nach rechts zu lesen und lehnen sich somit an die in der linguistischen Literatur übliche Notation von Attribut-Wert-Matrizen an.

9 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$, for each $\mathit{ass} \in \mathit{Ass}_I$, T_I^{ass} is the total function from $\mathfrak{T}^\Sigma$ to the set of partial functions from $\overline{U}$ to $\overline{U}$ such that for each $u \in \overline{U}$*

$T_I^{ass}(:)(u)$ *is defined and* $T_I^{ass}(:)(u) = u$,

for each $v \in Var$, $T_I^{ass}(v)(u)$ *is defined and* $T_I^{ass}(v)(u) = ass(v)$,

for each $\tau \in \mathfrak{T}^\Sigma$, *and* $\alpha \in \mathfrak{A}$,

$T_I^{ass}(\tau\alpha)(u)$ *is defined*

iff $T_I^{ass}(\tau)(u)$ *is defined and* $\widehat{A}(\alpha)(T_I^{ass}(\tau)(u))$ *is defined, and*

if $T_I^{ass}(\tau\alpha)(u)$ *is defined*

then $T_I^{ass}(\tau\alpha)(u) = \widehat{A}(\alpha)(T_I^{ass}(\tau)(u))$.

Σ-Terme werden in einer Σ-Interpretation bezüglich einer Belegung ass in I als partielle Funktionen von $\overline{U}$ in $\overline{U}$ interpretiert. Wir nennen T_I^{ass} die Terminterpretation in I unter der Belegung ass. : denotiert die Identitätsfunktion auf $\overline{U}$. Eine Variable v wird für jedes $u \in \overline{U}$ interpretiert als Abbildung von u auf das $u' \in \overline{U}$, das v unter der Belegung ass in I zugewiesen wird. Die Bedeutung von komplexen Σ-Termen der Form $v\alpha_1 \dots \alpha_n$ (wobei an die Stelle der Variable v auch das Symbol : treten kann) erhält man durch funktionale Komposition der einstelligen partiellen Funktionen, als die die Attribute α_1 bis α_n interpretiert werden.

10 Definition. *For each signature* Σ, $\mathfrak{D}^\Sigma$ *is the smallest set such that*

for each $\sigma \in \mathfrak{G}$ *and each* $\tau \in \mathfrak{T}^\Sigma$, $\tau{\sim}\sigma \in \mathfrak{D}^\Sigma$,

for each $\tau_1 \in \mathfrak{T}^\Sigma$ *and each* $\tau_2 \in \mathfrak{T}^\Sigma$, $\tau_1{\approx}\tau_2 \in \mathfrak{D}^\Sigma$,

for each $r \in \mathfrak{R}$, *and each* $x_1, \dots, x_{Ar(r)} \in Var$, $r(x_1, \dots, x_{Ar(r)}) \in \mathfrak{D}^\Sigma$,

for each $x \in Var$, *and each* $\delta \in \mathfrak{D}^\Sigma$, $\exists x\, \delta \in \mathfrak{D}^\Sigma$,

for each $x \in Var$, *and each* $\delta \in \mathfrak{D}^\Sigma$, $\forall x\, \delta \in \mathfrak{D}^\Sigma$,

for each $\delta \in \mathfrak{D}^\Sigma$, $\neg\delta \in \mathfrak{D}^\Sigma$,

for each $\delta_1 \in \mathfrak{D}^\Sigma$ *and each* $\delta_2 \in \mathfrak{D}^\Sigma$, $[\delta_1 \wedge \delta_2] \in \mathfrak{D}^\Sigma$,

for each $\delta_1 \in \mathfrak{D}^\Sigma$ *and each* $\delta_2 \in \mathfrak{D}^\Sigma$, $[\delta_1 \vee \delta_2] \in \mathfrak{D}^\Sigma$,

for each $\delta_1 \in \mathfrak{D}^\Sigma$ *and each* $\delta_2 \in \mathfrak{D}^\Sigma$, $[\delta_1 \to \delta_2] \in \mathfrak{D}^\Sigma$, *and*

for each $\delta_1 \in \mathfrak{D}^\Sigma$ *and each* $\delta_2 \in \mathfrak{D}^\Sigma$, $[\delta_1 \leftrightarrow \delta_2] \in \mathfrak{D}^\Sigma$.

Wir nennen die Elemente von $\mathfrak{D}^\Sigma$ Σ-Beschreibungen. Σ-Beschreibungen der Form $\tau{\sim}\sigma$, $\tau_1{\approx}\tau_2$ und $r(x_1, \dots, x_{Ar(r)})$ nennen wir atomar. Atomare Σ-Beschreibungen der Form $\tau{\sim}\sigma$ heißen Sortenzuweisung, atomare Σ-Beschreibungen der Form $\tau_1{\approx}\tau_2$ heißen Pfadgleichung, und atomare Σ-Beschreibungen der Form $r(x_1, \dots, x_{Ar(r)})$ heißen Relationsaufruf. Aus atomaren Σ-Beschreibungen können mit dem Existenzquantor, dem Allquantor, der Negation und den zweistelligen logischen Junktoren Konjunktion, Disjunktion, Implikation und Äquivalenz komplexe Σ-Beschreibungen gebildet werden.

Die Menge der Variablen, die frei in einer Σ-Beschreibung vorkommen, können wir induktiv über den Aufbau von Σ-Beschreibungen definieren.

11 Definition. *For each* $\pi \in \mathfrak{A}^*$, *for each* $\tau, \tau_1, \tau_2 \in \mathfrak{T}^\Sigma$, *for each* $r \in \mathfrak{R}$, *for each* $v, x_1, \dots, x_{Ar(r)} \in Var$, *and for each* $\delta, \delta_1, \delta_2 \in \mathfrak{D}^\Sigma$,

$\mathrm{FV}(:\pi) = \emptyset$,

$\mathrm{FV}(v\pi) = \{v\}$,

$\mathrm{FV}(\tau \sim \sigma) = \mathrm{FV}(\tau)$,

$\mathrm{FV}(\tau_1 \approx \tau_2) = \mathrm{FV}(\tau_1) \cup \mathrm{FV}(\tau_2)$,

$\mathrm{FV}(r(x_1, \dots, x_{Ar(r)})) = \{x_1, \dots, x_{Ar(r)}\}$,

$\mathrm{FV}(\exists v\, \delta) = \mathrm{FV}(\delta) \backslash \{v\}$,

$\mathrm{FV}(\forall v\, \delta) = \mathrm{FV}(\delta) \backslash \{v\}$,

$\mathrm{FV}(\neg\delta) = \mathrm{FV}(\delta)$,

$\mathrm{FV}(\delta_1 \wedge \delta_2) = \mathrm{FV}(\delta_1) \cup \mathrm{FV}(\delta_2)$,

$\mathrm{FV}(\delta_1 \vee \delta_2) = \mathrm{FV}(\delta_1) \cup \mathrm{FV}(\delta_2)$,

$\mathrm{FV}(\delta_1 \to \delta_2) = \mathrm{FV}(\delta_1) \cup \mathrm{FV}(\delta_2)$, *and*

$\mathrm{FV}(\delta_1 \leftrightarrow \delta_2) = \mathrm{FV}(\delta_1) \cup \mathrm{FV}(\delta_2)$.

Ein Σ-Term, der mit : beginnt, enthält keine freie Variable. Für jede beliebige Variable, mit der ein Σ-Term τ beginnt, ist v die freie Variable in τ. Für Sortenzuweisungen und Pfadgleichungen ergibt sich die Menge der freien Variablen aus den freien Variablen der Σ-Terme, aus denen sie aufgebaut sind. Die Variablen in einem Relationsaufruf sind die freien Variablen eines Relationsaufrufs. Bei Existenz- und Allquantifikation ist die Menge der freien Variablen die Menge der freien Variablen in δ ohne die

Variable, über die quantifiziert wird. Bei den durch Junktoren gebildeten komplexen Σ-Beschreibungen ergibt sich die Menge der freien Variablen aus der Vereinigung der Mengen der freien Variablen in den durch den Junktor verbundenen Σ-Beschreibungen.

12 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$ and each $u \in U$, $Comp_I(u)$ is the set of components of u in I iff*

$$Comp_I(u) = \left\{ T_I^{ass}(:\pi)(u) \;\middle|\; : \pi \in \mathfrak{T}^\Sigma \right\}.$$

Wir nennen die Elemente der Menge $Comp_I(u)$ die Komponenten von u in I und sagen, daß u endlich ist, wenn u in I endlich viele Komponenten hat. Umgekehrt sagen wir, daß u unendlich ist, wenn $Comp_I(u)$ unendlich ist. Die Menge der Komponenten eines Objekts u des Universums bezüglich einer Σ-Interpretation ist die Menge all derjenigen Objekte von U, die man mit beliebigen mit : beginnenden Σ-Termen von u aus erreichen kann. Die Wahl der Σ-Terme, die mit : beginnen, stellt sicher, daß das erste Attribut in π eine Funktion denotiert, die auf u definiert ist, wenn $:\pi$ auf u definiert ist. Dies ist bei Variablen als Termanfang im allgemeinen nicht der Fall.

13 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$, for each $ass \in Ass_I$, for each $v \in Var$, for each $u \in \overline{U}$, for each $w \in Var$,*

$$ass\tfrac{u}{v}(w) = \begin{cases} u & \text{if } v = w \\ ass(w) & \text{otherwise.} \end{cases}$$

Wenn ass eine Belegung in I ist, dann ist $ass\frac{u}{v}$ diejenige Belegung in I, die v auf u abbildet und für alle von v verschiedenen Variablen mit ass übereinstimmt.

14 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$, for each $ass \in Ass_I$, D_I^{ass} is the total function from $\mathfrak{D}^\Sigma$ to the powerset of U such that*

for each $\tau \in \mathfrak{T}^\Sigma$, for each $\sigma \in \mathfrak{G}$,

$$D_I^{ass}(\tau \sim \sigma) = \left\{ u \;\middle|\; \begin{array}{l} T_I^{ass}(\tau)(u) \text{ is defined, and} \\ \widehat{S}(T_I^{ass}(\tau)(u)) \sqsubseteq \sigma \end{array} \right\},$$

for each $\tau \in \mathfrak{T}^\Sigma$, for each $\sigma \in \mathfrak{S}$,

$$D_I^{ass}(\neg\tau \sim \sigma) = U \backslash D_I^{ass}(\tau \sim \sigma),$$

for each $\tau_1 \in \mathfrak{T}^\Sigma$, for each $\tau_2 \in \mathfrak{T}^\Sigma$,

$$D_I^{ass}(\tau_1 \approx \tau_2) = \left\{ u \;\middle|\; \begin{array}{l} T_I^{ass}(\tau_1)(u) \text{ is defined,} \\ T_I^{ass}(\tau_2)(u) \text{ is defined, and} \\ T_I^{ass}(\tau_1)(u) = T_I^{ass}(\tau_2)(u) \end{array} \right\},$$

for each $\tau_1 \in \mathfrak{T}^\Sigma$, for each $\tau_2 \in \mathfrak{T}^\Sigma$,

$$D_I^{ass}(\neg\tau_1 \approx \tau_2) = U \backslash D_I^{ass}(\tau_1 \approx \tau_2),$$

for each $r \in \mathfrak{R}$, for each $x_1, \ldots, x_{Ar(r)} \in Var$,

$$\begin{array}{l} D_I^{ass}(r(x_1, \ldots, x_{Ar(r)})) \\ = \left\{ u \;\middle|\; \langle ass(x_1), \ldots, ass(x_{Ar(r)})\rangle \in P(r) \right\}, \end{array}$$

for each $r \in \mathfrak{R}$, for each $x_1, \ldots, x_{Ar(r)} \in Var$,

$$D_I^{ass}(\neg r(x_1, \ldots, x_{Ar(r)})) = U \backslash D_I^{ass}(r(x_1, \ldots, x_{Ar(r)})),$$

for each $v \in Var$, for each $\delta \in \mathfrak{D}^\Sigma$,

$$D_I^{ass}(\exists v\, \delta) = \left\{ u \;\middle|\; \begin{array}{l} \text{if } Comp_I(u) \text{ is finite,} \\ \text{then for some } u' \in \overline{Comp_I(u)}, \\ u \in D_I^{ass\frac{u'}{v}}(\delta), \\ \text{else true} \end{array} \right\},$$

for each $v \in Var$, for each $\delta \in \mathfrak{D}^\Sigma$,

$$D_I^{ass}(\neg\exists v\, \delta) = \left\{ u \;\middle|\; \begin{array}{l} \text{if } Comp_I(u) \text{ is finite,} \\ \text{then for each } u' \in \overline{Comp_I(u)}, \\ u \in D_I^{ass\frac{u'}{v}}(\neg\delta), \\ \text{else true} \end{array} \right\},$$

for each $v \in Var$, for each $\delta \in \mathfrak{D}^\Sigma$,

$$D_I^{ass}(\forall v\, \delta) = \left\{ u \;\middle|\; \begin{array}{l} \text{if } Comp_I(u) \text{ is finite,} \\ \text{then for each } u' \in \overline{Comp_I(u)}, \\ u \in D_I^{ass\frac{u'}{v}}(\delta), \\ \text{else true} \end{array} \right\},$$

for each $v \in Var$, *for each* $\delta \in \mathfrak{D}^{\Sigma}$,

$$D_I^{ass}(\neg\forall v\ \delta) = \left\{ u \;\middle|\; \begin{array}{l} \textit{if } Comp_I(u) \textit{ is finite,} \\ \textit{then for some } u' \in \overline{Comp_I(u)}, \\ u \in D_I^{ass\frac{u'}{v}}(\neg\delta), \\ \textit{else true} \end{array} \right\},$$

for each $\delta_1 \in \mathfrak{D}^{\Sigma}$, *for each* $\delta_2 \in \mathfrak{D}^{\Sigma}$,

$$D_I^{ass}([\delta_1 \wedge \delta_2]) = D_I^{ass}(\delta_1) \cap D_I^{ass}(\delta_2),$$

for each $\delta_1 \in \mathfrak{D}^{\Sigma}$, *for each* $\delta_2 \in \mathfrak{D}^{\Sigma}$,

$$D_I^{ass}(\neg[\delta_1 \wedge \delta_2]) = D_I^{ass}(\neg\delta_1) \cup D_I^{ass}(\neg\delta_2),$$

for each $\delta_1 \in \mathfrak{D}^{\Sigma}$, *for each* $\delta_2 \in \mathfrak{D}^{\Sigma}$,

$$D_I^{ass}([\delta_1 \vee \delta_2]) = D_I^{ass}(\delta_1) \cup D_I^{ass}(\delta_2),$$

for each $\delta_1 \in \mathfrak{D}^{\Sigma}$, *for each* $\delta_2 \in \mathfrak{D}^{\Sigma}$,

$$D_I^{ass}(\neg[\delta_1 \vee \delta_2]) = D_I^{ass}(\neg\delta_1) \cap D_I^{ass}(\neg\delta_2),$$

for each $\delta_1 \in \mathfrak{D}^{\Sigma}$, *for each* $\delta_2 \in \mathfrak{D}^{\Sigma}$,

$$D_I^{ass}([\delta_1 \rightarrow \delta_2]) = D_I^{ass}(\neg\delta_1) \cup D_I^{ass}(\delta_2),$$

for each $\delta_1 \in \mathfrak{D}^{\Sigma}$, *for each* $\delta_2 \in \mathfrak{D}^{\Sigma}$,

$$D_I^{ass}(\neg[\delta_1 \rightarrow \delta_2]) = D_I^{ass}(\delta_1) \cap D_I^{ass}(\neg\delta_2),$$

for each $\delta_1 \in \mathfrak{D}^{\Sigma}$, *for each* $\delta_2 \in \mathfrak{D}^{\Sigma}$,

$$\begin{array}{l} D_I^{ass}([\delta_1 \leftrightarrow \delta_2]) \\ = (D_I^{ass}(\neg\delta_1) \cap D_I^{ass}(\neg\delta_2)) \cup (D_I^{ass}(\delta_1) \cap D_I^{ass}(\delta_2)), \end{array}$$

for each $\delta_1 \in \mathfrak{D}^{\Sigma}$, *for each* $\delta_2 \in \mathfrak{D}^{\Sigma}$,

$$\begin{array}{l} D_I^{ass}(\neg[\delta_1 \leftrightarrow \delta_2]) \\ = (D_I^{ass}(\delta_1) \cup D_I^{ass}(\delta_2)) \cap (D_I^{ass}(\neg\delta_1) \cup D_I^{ass}(\neg\delta_2)). \end{array}$$

Für jede Σ-Interpretation I und jede Belegung ass in I gibt es eine Beschreibungsdenotation D_I^{ass}. Informell gesprochen denotiert eine Σ-Beschreibung δ die Menge der Objekte des Universums der Σ-Interpretation, für die δ wahr ist.

$D_I^{ass}(\tau \sim \sigma)$ ist die Menge aller Objekte des Universums, auf denen der Σ-Term τ definiert ist und auf die angewandt die von τ benannte partielle Funktion $T_I^{ass}(\tau)$ ein u' in $\overline{U}$ ergibt, dem von $\widehat{S}$ eine maximal spezifische Sorte zugeordnet wird, die in der Sortenhierarchie unter σ liegt. Die Negation $\neg\tau \sim \sigma$ einer Sortenzuweisung denotiert das Komplement der Sortenzuweisung $\tau \sim \sigma$.

$D_I^{ass}(\tau_1 \approx \tau_2)$ ist die Menge aller Objekte u des Universums, so daß sowohl die durch den Σ-Term τ_1 benannte partielle Funktion $T_I^{ass}(\tau_1)$ als auch die durch den Σ-Term τ_2 benannte partielle Funktion $T_I^{ass}(\tau_2)$ auf u definiert ist und die beiden partiellen Funktionen angewandt auf u das gleiche $u' \in \overline{U}$ als Wert ergeben. Wie bei Sortenzuweisung ist auch die Bedeutung der Negation von Pfadgleichungen klassisch. $D_I^{ass}(\neg\tau_1 \approx \tau_2)$ ist die Menge der Objekte, für die die Σ-Beschreibung $\tau_1 \approx \tau_2$ falsch ist.

Für jedes Relationssymbol r mit $Ar(r) = n$ ist $D_I^{ass}(r(x_1, \ldots, x_n))$ die Menge all der Objekte des Universums, so daß das n-Tupel $\langle u_1, \ldots, u_n\rangle$ bestehend aus dem $u_1 \in \overline{U}$, auf das die Variable x_1 von der Belegung ass in I abgebildet wird, bis zu dem $u_n \in \overline{U}$, auf das die Variable x_n von der Belegung ass in I abgebildet wird, in $P(r)$ liegt. $D_I^{ass}(\neg r(x_1, \ldots, x_{Ar(r)}))$ ist die Komplementmenge von $D_I^{ass}(r(x_1, \ldots, x_{Ar(r)}))$.

$D_I^{ass}(\exists v\ \delta)$ ist die Menge aller Objekte u des Universums, die unendlich viele Komponenten haben, vereinigt mit den u, die aus endlich vielen Komponenten bestehen, so daß es ein u' gibt, das eine Komponente oder eine Kette von Komponenten von u ist, und u ist in $D_I^{ass\frac{u'}{v}}(\delta)$. Das heißt, es gibt ein u' unter den Komponenten oder Ketten von Komponenten von u, so daß, wenn jedes freie Vorkommen von v in δ als u' interpretiert wird, δ auf u zutrifft. Die Negation einer existentiell quantifizierten Σ-Beschreibung erhält man im allgemeinen nicht durch Komplementbildung. $D_I^{ass}(\neg\exists v\ \delta)$ ist die Menge aller Objekte u des Universums, die unendlich viele Komponenten haben, vereinigt mit den u, die aus endlich vielen Komponenten bestehen und der Art sind, daß für jedes u', das eine Komponente oder Kette von Komponenten von u ist, u in $D_I^{ass\frac{u'}{v}}(\neg\delta)$ ist. Das heißt, für jedes u' unter den Komponenten oder Ketten von Komponenten von u gilt, daß δ u nicht beschreibt, wenn die freien Vorkommen von v in δ als u' interpretiert werden.

Allquantifizierte Σ-Beschreibungen werden entsprechend zum Fall der Existenzquantifikation behandelt. Folglich ist $D_I^{ass}(\forall v\,\delta)$ die Menge derjenigen Objekte u des Universums, die aus unendlich vielen Komponenten bestehen, vereinigt mit den endlichen u, für die gilt, daß δ bei der Interpretation der freien Vorkommen von v in δ als beliebige u', die Komponenten oder Ketten von Komponenten von u sind, auf sie zutrifft. Nach dem Schema für Existenzquantifikation ergibt sich für die Negation im Fall der Allquantifikation, daß $D_I^{ass}(\neg\forall v\,\delta)$ die Menge derjenigen Objekte u des Universums ist, die unendlich sind, vereinigt mit der Menge der endlichen u, für die gilt, daß δ auf sie für mindestens ein u' aus der Menge der Komponeneten und Ketten von Komponenten von u nicht zutrifft, wenn alle freien Vorkommen von v in δ als u' interpretiert werden.

Die zweistelligen Junktoren werden klassisch interpretiert. $D_I^{ass}([\delta_1 \wedge \delta_2])$ ist die Menge der Objekte in U, für die sowohl δ_1 als auch δ_2 wahr ist, und $D_I^{ass}(\neg[\delta_1 \wedge \delta_2])$ die Menge der $u \in U$, so daß die Negation von δ_1 oder die Negation von δ_2 wahr ist. $D_I^{ass}([\delta_1 \vee \delta_2])$ ist die Menge der $u \in U$, auf die δ_1 oder δ_2 zutrifft; $D_I^{ass}(\neg[\delta_1 \vee \delta_2])$ die Menge der u, für die sowohl $\neg\delta_1$ als auch $\neg\delta_2$ wahr ist. $D_I^{ass}([\delta_1 \to \delta_2])$ ist die Menge der Objekte in U, auf die δ_1 nicht zutrifft oder auf die δ_2 zutrifft, und seine Negation, $D_I^{ass}(\neg[\delta_1 \to \delta_2])$, ist folglich die Menge derjenigen Objekte in U, auf die δ_1 zutrifft und δ_2 nicht zutrifft. Schließlich ist $D_I^{ass}([\delta_1 \leftrightarrow \delta_2])$ die Menge derjenigen $u \in U$, für die sowohl δ_1 als auch δ_2 wahr ist oder auf die weder δ_1 noch δ_2 zutrifft; und $D_I^{ass}(\neg[\delta_1 \leftrightarrow \delta_2])$ sind diejenigen Objekte, für die erstens δ_1 oder δ_2 wahr ist und zweitens die Negation von δ_1 oder die Negation von δ_2 wahr ist.

15 Proposition. *For each $\delta \in \mathfrak{D}^\Sigma$, for each Σ-interpretation I, for each $ass_1 \in Ass_I$, for each $ass_2 \in Ass_I$,*

if for each $v \in \mathrm{FV}(\delta)$, $ass_1(v) = ass_2(v)$, then

$D_I^{ass_1} = D_I^{ass_2}$.

Satz (15) sagt, daß zwei Beschreibungsdenotationen in I unter den Belegungen ass_1 und ass_2 identisch sind, wenn für alle Σ-Beschreibungen δ gilt, daß ass_1 und ass_2 hinsichtlich der in δ frei vorkommenden Variablen übereinstimmen. Der Beweis von (15) erfolgt induktiv über den Aufbau von Σ-Beschreibungen.

16 Definition. *For each signature* Σ,

$\mathfrak{D}_0^\Sigma = \{\delta \in \mathfrak{D}^\Sigma \mid FV(\delta) = \emptyset\}$.

Wir nennen die Elemente von $\mathfrak{D}_0^\Sigma$ Σ-Sätze. Eine Menge von Σ-Sätzen nennen wir eine Σ-Theorie. Σ-Sätze sind Σ-Beschreibungen, in denen keine Variable frei vorkommt.

17 Corollary. *For each $\delta \in \mathfrak{D}_0^\Sigma$, for each Σ-interpretation I, for each $ass_1 \in Ass_I$, for each $ass_2 \in Ass_2$,*

$D_I^{ass_1} = D_I^{ass_2}$.

Da Σ-Sätze per Definition keine freien Variablen enthalten, folgt (17) unmittelbar aus (15). Korollar (17) besagt, daß die Denotation von Σ-Sätzen unabhängig ist von der Belegung ass in I, unter der sie interpretiert werden. Wir können folglich bei der Definition der Denotation von Σ-Sätzen von den Belegungen abstrahieren.

18 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$, D_I is the total function from $\mathfrak{D}_0^\Sigma$ to the powerset of U such that for each $\delta \in \mathfrak{D}_0^\Sigma$,*

$$D_I(\delta) = \left\{ u \in U \;\middle|\; \begin{array}{l} \text{for each (equivalently, some) } ass \in Ass_I, \\ u \in D_I^{ass}(\delta) \end{array} \right\}.$$

D_I heißt die Satzdenotation in I. Die Definition von D_I sagt intuitiv, daß ein Σ-Satz für die Objekte des Universums wahr ist, die unter mindestens einer Belegung ass in I in der Beschreibungsdenotation D_I^{ass} von δ liegen. Da Variablenbelegungen für die Bedeutung von Σ-Sätzen irrelevant sind, kann ein beliebiges ass in I gewählt werden.

19 Definition. *For each Σ-interpretation $I = \langle U, S, A, P\rangle$, Θ_I is the total function from the powerset of $\mathfrak{D}_0^\Sigma$ to the powerset of U such that for each $\theta \subseteq \mathfrak{D}_0^\Sigma$,*

$\Theta_I(\theta) = \bigcap \{ D_I(\delta) \mid \delta \in \theta \}$

Θ_I heißt die Theoriedenotation in I. Die Theoriedenotation in I einer Σ-Theorie θ ist die Menge der Objekte u des Universums U, für die alle Σ-Sätze in θ wahr sind.

20 Definition. Γ *is a grammar iff*

Γ *is a pair* $\langle \Sigma, \theta \rangle$,

Σ *is a signature, and*

$\theta \subseteq \mathfrak{D}_0^\Sigma$.

Eine Grammatik besteht aus einer Signatur Σ und einer Σ-Theorie. Wenn in einer Grammatik der Bezug auf eine bestimmte Signatur klar ist, sprechen wir auch einfach von der Theorie einer Grammatik. Im gleichen Sinne sprechen wir von den Termen, den Beschreibungen und den Sätzen einer Grammatik.

Grammatiken werden verwendet, um bestimmte Klassen von Modellen zu charakterisieren. Die beiden im folgenden definierten Begriffe des Γ-Modells und des exhaustiven Γ-Modells zeichnen die interessanten Σ-Interpretationen einer Grammatik aus.

21 Definition. *I is a Γ-model iff*

Γ is a grammar $\langle \Sigma, \theta \rangle$,

I is a Σ-interpretation $\langle U, S, A, P \rangle$, and

$\Theta_I(\theta) = U$.

Ein Γ-Modell einer Grammatik $\Gamma = \langle \Sigma, \theta \rangle$ ist eine Σ-Interpretation $I = \langle U, S, A, P \rangle$, die von der Gestalt ist, daß alle Objekte des Universums U in der Theoriedenotation in I der Theorie θ liegen. Mit anderen Worten trifft die Theorie auf das gesamte Universum zu. Alle Sätze in der Theorie sind für alle u in I wahr.

Wenn der Bezug auf eine bestimmte Grammatik klar ist, sprechen wir statt von einer Σ-Interpretation einfach von einer Interpretation.

22 Definition. *I is an exhaustive Γ-model iff*

Γ is a grammar $\langle \Sigma, \theta \rangle$,

I is a Γ-model, and

for each $\theta' \subseteq \mathfrak{D}_0^\Sigma$,

if for some Γ-model I',
$\Theta_{I'}(\theta') \neq \emptyset$,

then $\Theta_I(\theta') \neq \emptyset$.

Sei I ein Γ-Modell einer Grammatik $\Gamma = \langle \Sigma, \theta \rangle$. Dann ist I ein exhaustives Γ-Modell, wenn folgendes zutrifft: Für jede beliebige Σ-Theorie θ',

die in einem zweiten Γ-Modell I' in $\Theta_{I'}$ nicht die leere Menge denotiert, ist auch die Theoriedenotation im exhaustiven Γ-Modell I für θ' nicht leer. Anders ausgedrückt heißt das, daß, wenn eine Theorie θ' in einem Γ-Modell I' etwas beschreibt, θ' auch in einem exhaustiven Γ-Modell I etwas beschreibt. Da das Γ-Modell I' beliebig gewählt werden darf und wir für jedes I' alle denkbaren Σ-Theorien betrachten, muß ein exhaustives Γ-Modell in gewissem Sinne Instanzen aller Objekte enthalten, welche die Theorie θ der Grammatik Γ auch nur beschreiben kann. Wäre das nicht der Fall, müßte es ein Γ-Modell geben, das Objekte enthält, zu denen sich im exhaustiven Γ-Modell kein Gegenstück finden läßt. Die Definition sagt aber, daß es zu allen Objekten in jedem beliebigen Γ-Modell ein Gegenstück im exhaustiven Γ-Modell gibt. Wir können damit sicher sein, daß in einem exhaustiven Γ-Modell nichts fehlt, was von der Theorie der Grammatik zugelassen wird. Andererseits enthält es aufgrund der Definition von Γ-Modellen nur Objekte, auf die die gesamte Theorie der Grammatik zutrifft.

Der Begriff des exhaustiven Γ-Modells schränkt die Menge der interessanten Interpretationen einer Grammatik auf die Klasse der linguistisch relevanten Interpretationen ein. Wenn Linguisten eine Grammatik schreiben, dann sind sie für die empirischen Vorhersagen der Grammatik an ihren exhaustiven Γ-Modellen interessiert.

2.2. Einige weitere Definitionen. Bei der Verwendung von RSRL zum Schreiben einer Grammatik erweist es sich als vorteilhaft, von einigen notationellen Abkürzungen Gebrauch zu machen. Diese Abkürzungen betreffen vor allem den Gebrauch von Relationen und zielen dahin, ihre Notation den in der linguistischen Gemeinde gängigen Konventionen anzunähern. Die leitende Idee für die folgenden abkürzenden Schreibweisen ist die, die Angabe von Quantoren sowie Variablen und bestimmten Pfadgleichungen an einigen Stellen implizit zu lassen und so die Übersichtlichkeit der Ausdrücke zu erhöhen. Dies wird es vor allem ermöglichen, im nächsten Abschnitt eine an die hier definierte Notation angelehnte leicht lesbare Erweiterung der herkömmlichen AVM-Notation zu skizzieren, welche den Lesegewohnheiten von Linguisten entgegenkommt und somit leichter zugänglich ist als Ausdrücke von RSRL.

Zu diesem Zweck wird zunächst ein Begriff der simultanen Substitution von Variablen durch neue Variablen in Termen und Beschreibungen definiert. Die Idee ist die, in einem Term beziehungsweise in einer Beschreibung bestimmte frei darin vorkommende Variablen simultan durch neue Variablen zu ersetzen. Im folgenden sei stets eine feste Signatur Σ

zugrundegelegt. In Definition (23) definieren wir simultane Substitution in Termen und verwenden sie in Definition (24) zur Definition der simultanen Substitution von Variablen in Beschreibungen.

Eckige Klammern dienen im folgenden der besseren Lesbarkeit.

23 Definition. *Seien $x_1, \ldots, x_n, v_1, \ldots, v_n$ paarweise verschiedene Variablen und π eine beliebige Folge von Attributen. Dann gilt*

$$[:\pi]\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \stackrel{\text{def}}{=} :\pi,$$

$$[x\pi]\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \stackrel{\text{def}}{=} \begin{cases} x\pi, & \text{falls } x \neq x_1, \ldots, x \neq x_n, \\ v_i\pi, & \text{falls } x = x_i. \end{cases}$$

Ein Term, der mit dem Symbol : beginnt, enthält keine Variable, und es wird nichts substituiert. Beginnt der Term hingegen mit einer der Variablen x_1 bis x_n, dann wird sie durch die entsprechende Variable v_i ersetzt.

24 Definition. *Seien $x_1, \ldots, x_n, v_1, \ldots, v_n$ paarweise verschiedene Variablen, τ, τ_1 und τ_2 Terme, σ eine Sorte, $r(z_1, \ldots, z_{Ar(r)})$ ein Relationsaufruf und δ eine Beschreibung. Dann gilt*

$$[\tau \sim \sigma]\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \stackrel{\text{def}}{=} \tau\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \sim \sigma,$$

$$[\tau_1 \approx \tau_2]\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \stackrel{\text{def}}{=} \tau_1\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \approx \tau_2\tfrac{v_1\ldots v_n}{x_1\ldots x_n},$$

$$\left[r(z_1, \ldots, z_{Ar(r)})\right]\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \stackrel{\text{def}}{=} r\left(z_1\tfrac{v_1\ldots v_n}{x_1\ldots x_n}, \ldots, z_{Ar(r)}\tfrac{v_1\ldots v_n}{x_1\ldots x_n}\right),$$

$$[\neg\delta]\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \stackrel{\text{def}}{=} \neg\left[\delta\tfrac{v_1\ldots v_n}{x_1\ldots x_n}\right].$$

Seien $x_1, \ldots, x_n, v_1, \ldots, v_n$ paarweise verschiedene Variablen, δ_1 und δ_2 Beschreibungen, und $\circ$ stehe für die Junktoren Konjunktion, Disjunktion, Implikation und Äquivalenz. Dann gilt

$$[\delta_1 \circ \delta_2]\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \stackrel{\text{def}}{=} \delta_1\tfrac{v_1\ldots v_n}{x_1\ldots x_n} \circ \delta_2\tfrac{v_1\ldots v_n}{x_1\ldots x_n}.$$

Seien $x_1, \ldots, x_r, v_1, \ldots, v_r$ paarweise verschiedene Variablen, $x_{i_1}, \ldots x_{i_n}$ $(i_1 < \ldots < i_n)$ gerade die Variablen x_i unter $x_1, \ldots, x_r$ mit $x_i \in \mathrm{FV}(\exists x\,\delta)$ und $x \neq v_{i_1}, \ldots, x \neq v_{i_n}$. Dann setzen wir

$$[\exists x\ \delta]\tfrac{v_1\ldots v_r}{x_1\ldots x_r} \stackrel{\text{def}}{=} \exists x\ \left[\delta\tfrac{v_{i_1}\ldots v_{i_n}}{x_{i_1}\ldots x_{i_n}}\right].$$

Entsprechend wird bei der Allquantifikation verfahren.

Der interessante Fall bei der simultanen Ersetzung von Variablen in Beschreibungen ist der, welcher quantifizierende Beschreibungen betrifft. Hier muß beachtet werden, daß keine Variable, über die quantifiziert wird, durch eine neue Variable ersetzt wird, die nicht mehr gebunden wäre. Umgekehrt darf durch simultane Substitution eine substituierte Variable nicht in den Wirkungsbereich eines Quantors gelangen.

Indem gefordert wird, daß in δ nur die x_i simultan ersetzt werden, die frei in $\exists x\,\delta$ vorkommen, wird ausgeschlossen, daß das durch die Quantifikation gebundene x ersetzt werden kann. Auf der anderen Seite sorgt die Forderung, daß zugleich auch nur die x_i ersetzt werden dürfen, so daß x verschieden ist von dem jeweiligen v_i, das für x_i substituiert wird, dafür, daß kein v_i von der Quantifikation gebunden wird.

RSRL überläßt es dem Benutzer, die Bedeutung von Relationen in der Theorie durch die Angabe von Beschreibungen zu bestimmen. Dies wird in der Literatur vorzugsweise durch die Angabe von Klauseln notiert, wobei es für eine Relation mehrere Klauseln geben kann. Letztere Eigenschaft ist insbesondere dann nützlich, wenn im Laufe einer Diskussion mehrere Klauseln an verschiedenen Stellen nacheinander eingeführt werden. Die syntaktisch separate Notation von Klauseln ermöglicht es, die Definition einer Relation zu erweitern, ohne daß die bereits gegebenen Teile wiederholt werden müssen. Darüber hinaus läßt die Klauselschreibweise die Quantifikation über die Variablen der definierten Relation implizit.

Um die Vorteile der Klauselnotation nutzen zu können, führen wir sie als syntaktischen Zucker ein:

25 Definition. *Sei $r(x_1, \ldots, x_{Ar(r)})$ ein Relationsaufruf mit paarweise verschiedenen Variablen $x_1, \ldots, x_{Ar(r)}$ und δ eine Σ-Beschreibung mit $\mathrm{FV}(\delta) \subseteq \{x_1, \ldots, x_{Ar(r)}\}$. Dann ist*

$$r(x_1, \ldots, x_{Ar(r)}) := \delta$$

eine Klausel.

Eine Klausel besteht aus einem Relationsaufruf (ihrem Kopf), den Symbolen := und einer Beschreibung δ (ihrem Rumpf), in der genau die im Klauselaufruf vorkommenden Variablen frei auftreten dürfen. Alle anderen gegebenenfalls auftretenden Variablen müssen gebunden sein. Man beachte, daß bei Klauseln eine paarweise Verschiedenheit der Variablen des Relationsaufrufs gefordert wird, die in RSRL im allgemeinen nicht gegeben sein muß.

Für eine Relation in der Grammatik gibt der Linguist eine Menge von Klauseln an, mit denen er ihre Bedeutung fixiert. In Definition (26) wird

festgelegt, welche Beschreibung in der Theorie der Grammatik durch die Angabe einer Menge von Klauseln abgekürzt wird. Dabei machen wir Gebrauch von der Tatsache, daß die Menge der Variablen abzählbar unendlich ist und wir daher fordern könnnen, daß sie alphabetisch geordnet sind.

26 Definition. *Sei* $K = \{r(v_{1_1}, \ldots, v_{1_{Ar(r)}}) := \delta_1, \ldots, r(v_{n_1}, \ldots, v_{n_{Ar(r)}}) := \delta_n\}$ *eine Menge von Klauseln. Seien ferner* $x_1, \ldots, x_{Ar(r)}$ *die alphabetisch ersten paarweise verschiedenen Variablen, die in keiner der Klauseln in* K *vorkommen. Dann ist* K *eine abkürzende Schreibweise für die* Σ*-Beschreibung*

$$\forall x_1 \ldots \forall x_{Ar(r)} \left[r(x_1, \ldots, x_{Ar(r)}) \leftrightarrow \left(\delta_1 \frac{x_1 \ldots x_{Ar(r)}}{v_{1_1} \ldots v_{1_{Ar(r)}}} \vee \ldots \vee \delta_n \frac{x_1 \ldots x_{Ar(r)}}{v_{n_1} \ldots v_{n_{Ar(r)}}} \right) \right].$$

Zur Bestimmung der abgekürzten Beschreibung nimmt man die neuen Variablen x_1 bis $x_{Ar(r)}$ und ersetzt in den Klauselrümpfen jeweils die dort verwendeten freien Variablen durch die entsprechenden neu gewählten. Dadurch wird erreicht, daß in den nach Substitution aus δ_1 bis δ_n gewonnenen Beschreibungen δ'_1 bis δ'_n nun jeweils die gleichen Variablen frei auftreten. Die substituierten Variablen werden schließlich durch Allquantifikation von außen gebunden. Daher tritt in der abgekürzten Beschreibung keine Variable frei auf, und sie ist ein Satz. Aus diesem Grund kann sie Element der Theorie einer Grammatik sein.

Die durch eine Menge von Klauseln für eine Relation r abgekürzte Beschreibung sagt inhaltlich, daß für alle $u \in U$ gilt, daß jede Komponente (oder Kette von Komponenten) $x_1, \ldots, x_{Ar(r)}$ von u genau dann in der Relation r steht, wenn eine der Beschreibungen δ'_1 bis δ'_n auf u zutrifft; oder u hat unendlich viele Komponenten.

Eine weitere syntaktische Abkürzung im Zusammenhang mit der Verwendung von Relationen erlaubt es schließlich, in bestimmten Kontexten auf die Einführung neuer Variablen und ihre Abbindung durch Existenzquantifikation notationell zu verzichten. Die Idee besteht darin, als Argumente von Relationsaufrufen nicht nur Variablen, sondern allgemein Terme zuzulassen:

27 Definition. *Sei* r *ein Relationssymbol mit* $Ar(r) = n$, $\tau_1, \ldots, \tau_n$ *Terme, von denen mindestens einer keine Variable ist, und* $x_1, \ldots, x_n$ *die alphabetisch ersten Variablen, die nicht frei in* $\tau_1, \ldots, \tau_n$ *vorkommen. Dann ist* $r(\tau_1, \ldots, \tau_n)$ *eine abkürzende Schreibweise für die* Σ*-Beschreibung*

$$\exists x_1 \ldots \exists x_n [x_1 \approx \tau_1 \wedge \ldots \wedge x_n \approx \tau_n \wedge r(x_1, \ldots, x_n)].$$

Mit dieser abkürzenden Schreibweise spart man die Einführung neuer Variablen x_1 bis x_n mit der zugehörigen Existenzquantifikation, wo man stattdessen auch einfach Terme als Argumente des Relationsaufrufs angeben kann. In der Anwendung wird sich zeigen, daß diese Vorgehensweise in Verbindung mit einer entsprechenden AVM-Notation die Angabe von Existenzquantifikation bei Relationsaufrufen in linguistischen Prinzipien oftmals überflüssig macht und zu einfach lesbaren Prinzipien führt.[6]

Zuletzt soll eine Relation eingeführt werden, durch deren Benutzung in Grammatiken sich einige sonst notwendige und notationell aufwendige Fallunterscheidungen vermeiden lassen. Dabei handelt es sich um den Umgang mit Listen, die selbst Merkmalstrukturen sind, und Ketten, bei denen nur die Elemente Merkmalstrukturen sind, nicht jedoch sie selbst. Durch die Erweiterung der Specieszuweisungsfunktion in Definition (5) und die Erweiterung der Attributinterpretationsfunktion in Definition (6) wurde bereits die Möglichkeit geschaffen, mit den Attributen FIRST und REST und den Sorten *list, elist* und *nelist* über Ketten wie über Listen zu sprechen. Nun möchte man aber auch manchmal von Listen zu Ketten übergehen. In solchen Fällen will man im wesentlichen ausdrücken, daß eine Kette und eine Liste insofern identisch sind, als daß sie identische Elemente in identischer Reihenfolge aufweisen. In einer entsprechenden Aussage über zwei Listen x und y oder zwei Ketten x und y erzielt man dieses Resultat einfach dadurch, daß man beide mittels einer Pfadgleichung vollständig miteinander identifiziert und damit zugleich für die Identität ihrer Elemente sorgt. Dies kann man aber nicht tun, wenn es sich bei x um eine nichtleere Kette und bei y um eine nichtleere Liste handelt. Während x dann keine Merkmalstruktur ist, sondern nur seine Elemente, ist y selbst eine Merkmalstruktur, nämlich ein Objekt von der Sorte *nelist*. Also können sie nicht miteinander identisch sein.

Die Relation $\approx$ leistet das Gewünschte. In (2.1) wird ihre Beschreibung durch zwei Klauseln angegeben.[7]

[6] Als typisches Beispiel hierfür wäre etwa das Subkategorisierungsprinzip (3.36) zu nennen.

[7] Zur Illustration sei hier der durch die Klauselnotation in (2.1) abgekürzte RSRL-Satz wiedergegeben:

$\forall x \forall y [\approx\!(x, y) \leftrightarrow ((x \sim \mathit{elist} \wedge y \sim \mathit{elist}) \vee$
$(\exists z_1 \exists z_2\ x_{\mathrm{FIRST}} \approx y_{\mathrm{FIRST}} \wedge x_{\mathrm{REST}} \approx z_1 \wedge y_{\mathrm{REST}} \approx z_2 \wedge \dot{\approx}(z_1, z_2)))]$

(2.1) *Die Klauseln der Relation* $\dot{\approx}$

$$\begin{aligned}\dot{\approx}(x, y) :=& \\ & x \sim \mathit{elist} \\ & \wedge\ y \sim \mathit{elist} \\ \dot{\approx}(x, y) :=& \\ & x\textsc{first} \approx y\textsc{first} \\ & \wedge\ \dot{\approx}(x\textsc{rest}, y\textsc{rest})\end{aligned}$$

Zwei Ketten, zwei Listen oder je eine Kette und eine Liste x und y stehen in der $\dot{\approx}$-Relation, wenn sie beide leer sind (erste Klausel), oder ihre ersten Elemente sind identisch und ihr Rest steht in der $\dot{\approx}$-Relation (zweite Klausel); die Relation ist auch erfüllt, wenn die beschriebene Struktur unendlich viele Komponenten aufweist.

Wir nehmen an, daß die Signatur jeder Grammatik das Relationssymbol $\dot{\approx}$ und ihre Theorie den mit den Klauseln (2.1) angegebenen Satz enthält, $\dot{\approx}$ also zur Verfügung steht. $\dot{\approx}$ erweist sich als nützlich bei der Definition von Relationen, die sowohl Listen als auch Ketten in beliebiger Mischung zugleich als Argumente behandeln können. Durch die Benutzung von $\dot{\approx}$ können Relationen zwischen Listen oder Ketten definiert werden, die für beliebige Mischungen von Ketten und Listen in ihren Argumenten wohldefiniert sind, indem sie bei der Identifikation (von Teilen) ihrer Argumente von Ketten auf Listen übergehen können. Eine dieserart definierte Relation kann also in verschiedenen Kontexten verwendet werden und leistet in relevanter Hinsicht jeweils das gleiche.[8] Für die Relation $\dot{\approx}$ verwenden wir gewöhnlich die Infixnotation.

2.3. Anmerkungen zur AVM-Notation. Abgesehen von der Angabe von Klauseln für Relationen werden wir in den folgenden Kapiteln normalerweise nicht direkt die Syntax von RSRL oder die in Abschnitt 2.2 definierten abkürzenden Schreibweisen für Beschreibungen verwenden, sondern die in der linguistischen HPSG-Literatur übliche AVM-Notation. Dies sollte indessen zu keinen Unklarheiten führen, da eine durchsichtige Korrespondenz zwischen AVMs und formaler Notation besteht und erstere im allgemeinen besser überschaubar sind. Da RSRL allerdings mit Quantoren und Relationen neue syntaktische Entitäten in die HPSG einführt, die dem klassischen Gebrauch von AVMs fremd sind, ist es notwendig,

[8] Als Beispiel vgl. die Klauseln von append in Fußnote 11, Seite 56. Durch die Verwendung von $\dot{\approx}$ in der ersten Klausel von append kann es sich bei den drei Argumenten der Relation jeweils beliebig um Ketten oder Listen handeln. Von dieser Eigenschaft wird in der Linearisierungstheorie Gebrauch gemacht, um Fallunterscheidungen zu vermeiden.

die Konventionen der AVM-Schreibweise um diese zu erweitern und auf die im folgenden eingehaltenen Regeln kurz einzugehen. Dabei soll keine Abbildung von AVMs auf RSRL-Beschreibungen definiert werden, sondern wir begnügen uns mit einigen Beispielen, welche die Erweiterung illustrieren.

In AVMs werden Pfadgleichungen als Tags notiert. Die Zeilen innerhalb der Boxen eines AVM werden im allgemeinen konjunktiv interpretiert. Traditionell gibt es weder Variablen noch Relationsaufrufe oder Quantoren. Sofern eine Beschreibung keines dieser zusätzlichen syntaktischen Ausdrucksmittel von RSRL verwendet, behalten wir einfach die üblichen Konventionen bei:

(2.2) *Ein AVM ohne Quantoren, Variablen und Relationen*

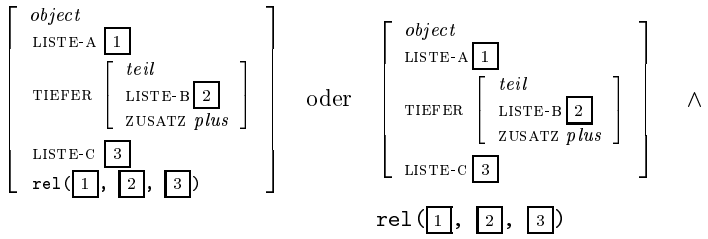
$$\left[\begin{array}{ll} \mathit{object} & \\ \textsc{liste-a}\ \mathit{list} & \\ \textsc{tiefer} & \left[\begin{array}{ll} \mathit{teil} & \\ \textsc{liste-b}\ \boxed{1} & \\ \textsc{zusatz}\ \mathit{plus} & \end{array}\right] \\ \textsc{liste-c}\ \boxed{1} & \end{array}\right]$$

ist eine AVM-Schreibweise für:

$$\begin{aligned} &: \sim \mathit{object} \\ \wedge\ &:\textsc{liste-a} \sim \mathit{list} \\ \wedge\ &:\textsc{tiefer} \sim \mathit{teil} \\ \wedge\ &:\textsc{tiefer liste-b} \approx :\textsc{liste-c} \\ \wedge\ &:\textsc{tiefer zusatz} \sim \mathit{plus} \end{aligned}$$

Relationsaufrufe notieren wir in einem von Attributen und Sorten verschiedenen Schrifttyp. Sie werden entweder in den Klammern des AVMs selbst angegeben und sind damit wie alle übrigen Zeilen als Konjunkt der Beschreibung zu verstehen; oder sie werden, wo dies übersichtlicher erscheint, gesondert neben den AVM geschrieben, wobei der Junktor ausdrücklich angegeben wird. Durch den syntaktischen Zucker für die Notation von Relationen, der es erlaubt, statt Variablen auch Terme als Argumente von Relationen aufzuschreiben, müssen bei der Verwendung von Relationen nicht notwendigerweise Variablen eingeführt werden, sondern es können in den meisten Fällen einfach Pfade als Argumente der Relationsaufrufe hingeschrieben werden. Der Übersichtlichkeit wegen erlauben wir uns, Pfade an den Argumentstellen von Relationen durch Tags abzukürzen:

(2.3) *AVMs mit Relationen*



ist eine AVM-Schreibweise für:

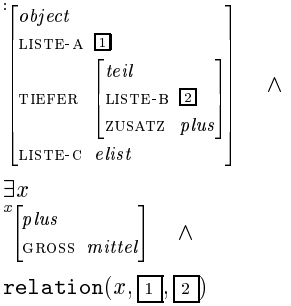
: ∼ *object*
∧ :TIEFER ∼ *teil*
∧ :TIEFER ZUSATZ ∼ *plus*
∧ **rel**(:LISTE-A, :TIEFER LISTE-B, :LISTE-C)

Zum Verständnis der Schreibweise in (2.3) ist es also essentiell, daß man darauf achtet, daß Tags niemals Variablen abkürzen. Insbesondere stehen sie als Argumente von Relationsaufrufen für Pfade und nicht für Variablen.

Wenn Variablen verwendet werden, wird das Bild sofort komplexer. Da Beschreibungen in der Theorie einer Grammatik immer Sätze sind, müssen die Variablen in Prinzipien durch Quantoren gebunden werden. Diese können nicht implizit gelassen werden, da zwischen Existenz- und Allquantifikation zu unterscheiden ist. Um in die AVM-Notation nicht komplizierte Klammerungsstrukturen einzuführen, werden wir davon ausgehen, daß Quantoren immer Skopus über den gesamten Ausdruck nehmen, der auf sie folgt. Nur in Ausnahmefällen wird eine Klammerung verwendet.

Aufgrund der Definitionen von RSRL bezeichnen Variablen semantisch eine Komponente (oder eine Kette von Komponenten) der durch das Symbol : denotierten Objekte der jeweiligen Beschreibung. Dabei kann die durch die Variable bezeichnete (Teil-) Merkmalstruktur ihrerseits näher beschrieben werden. Infolgedessen erhalten wir in einer komplexen Beschreibung mit Variablen gegebenenfalls mehrere AVM-Boxen, deren Beschreibungen sich auf verschiedene Merkmalstrukturen beziehen.

(2.4) *AVMs mit Variablen und Quantoren*



ist eine AVM-Schreibweise für:

: ∼ *object*
∧ :TIEFER ∼ *teil*
∧ :TIEFER ZUSATZ ∼ *plus*
∧ ∃*x* [*x* ∼ *plus*
∧ *x*GROSS ∼ *mittel*
∧ **relation**(*x*, :LISTE-A, :TIEFER LISTE-B)]

An der linken oberen Ecke der AVMs wird das Symbol : oder die Variable angegeben, mit denen die Terme des jeweiligen AVM beginnen. Man bemerke, daß bei der Verwendung von Tags als Argumenten von Relationsaufrufen das annotierte Zeichen an demjenigen AVM, in dem der Tag außer im Relationsaufruf vorkommt, angibt, mit welchem Zeichen der Term im Relationsaufruf beginnt. In Beispiel (2.4) ist dies sowohl bei Tag 1 als auch bei Tag 2 das : -Symbol. Wenn in einer Beschreibung Variablen auftreten, müssen alle AVMs dieser Beschreibung mit Variablen beziehungsweise dem Symbol : annotiert werden. Wenn keine Variablen aufteten, lassen wir hingegen in der Regel auch das Symbol : fort.

3. Die Satzstruktur des Deutschen

Während es im Prinzip ohne weiteres denkbar ist, HPSGrammatiken unabhängig von einem konkreten zugrundeliegenden Formalismus zu schreiben, werden wir im folgenden die Vorteile einer tatsächlich vorhandenen formalen Sprache ausnutzen. Da wir über eine präzise Semantik der spachwissenschaftlichen Beschreibungen verfügen und somit die zugelassenen Merkmalstrukturen konkret betrachten können, wird es einfacher, die Konsistenz der gegebenen Annahmen zu überprüfen. Außerdem vermittelt diese Vorgehensweise ein Gefühl für den tatsächlich notwendigen technischen Aufwand und die Umsetzbarkeit von informell gegebenen Prinzipien. Letzten Endes sollte diese Strategie zu einer leichteren Vergleichbarkeit konkurrierender Vorschläge führen. In diese Richtung zielt vor allem der Versuch, den Gebrauch von Relationen in einer sprachwissenschaftlichen Theorie auf eine formale Grundlage zu stellen und somit die Probleme und Grenzen dieses Werkzeuges abzuschätzen. Nicht zuletzt das Konzept der Linearisierung wird uns ausführlich Gelegenheit bieten, die Verwendung von Relationen zu explorieren.

Unter dem Aspekt der sprachwissenschaftlichen Verwendung unterteilen sich die von RSRL bereitgestellten Werkzeuge in drei Hauptgruppen, die es bei der Formalisierung einer Grammatik im Auge zu behalten gilt. Die Signatur bestimmt die Art der existierenden linguistischen Objekte und ihre Strukturierung. Die Prinzipien der Theorie geben an, welche der durch die Signatur gegliederten Merkmalstrukturen wohlgeformt und somit Element des exhaustiven Modells der Grammatik sind. Und die ebenfalls als Teil der Theorie formulierten Klauseln bestimmen die Denotation der Relationen und damit die Art der Beziehungen, die Relationsaufrufe in linguistischen Prinzipien zwischen Bestandteilen von Merkmalstrukturen induzieren.

Auf dem Hintergrund von RSRL wird es möglich sein, den empirischen Gehalt der Arbeitsteilung dieser drei Komponenten zu deuten und zu sehen, was sie jeweils zu einer kompakten und sprachwissenschaftlich intuitiven Beschreibung beitragen. Was die Signatur anlangt, werden wir weitgehend den Vorschlägen in (Pollard und Sag 1994) mit der dort im neunten Kapitel vorgenommenen Modifikation der Behandlung von Subkategorisierungslisten als COMPS- beziehungsweise SUBJ- und SPR-Listen folgen und nur dort davon abweichen, wo wir ausdrücklich darauf hinweisen. Anders verhält es sich mit den Prinzipien. Die Annahme einer Linearisierungsgrammatik im Unterschied zu einer im wesentlichen phrasenstrukturorientierten Architektur wird zu einer leichten Modifikation

der Merkmalsgeometrie führen, die eine Anpassung der meisten zentralen Prinzipien notwendig macht. Darüber hinaus werden einige Aufgaben der Grammatik an die Linearisierungskomponente delegiert oder doch so eng mit ihr verzahnt, daß sich eine neue Aufgabenverteilung ergibt. Explizite Definitionen tragen dazu bei, diese Verschiebungen transparent zu machen. Relationen werden in der Grammatik von (Pollard und Sag 1994) nicht ausdrücklich definiert, obwohl insbesondere für die Zwecke der Bindungstheorie eine ganze Reihe angenommen werden muß. Insofern betritt die vorliegende Grammatik weitgehend Neuland und kann sich nicht auf vorgegebene Definitionen beziehen. Gerade in der Wortstellungstheorie wird sich eine Syntax und Semantik von Relationen als äußerst nützlich erweisen, um zunächst informell präsentierte Prinzipien formal auszubuchstabieren. Erst durch diese Vorgehensweise wird es überhaupt möglich, den tatsächlichen Aufwand auf den ersten Blick zuweilen harmlos aussehender Prinzipien abzuschätzen. Das erscheint insbesondere im Hinblick darauf wichtig zu sein, daß ihrem Charakter nach relationale Prinzipien (wie zum Beispiel die Bindungstheorie oder LP-Regeln) in Papieren zur HPSG oft nur informell behandelt werden, wohingegen Prinzipien, die unmittelbar Attributwerte und deren Perkolation in Strukturen betreffen, explizit angegeben werden. Somit entsteht hinsichtlich letzterer oftmals der Eindruck größerer Komplexität im Vergleich zur relationalen Ausdrucksweise, die nicht in ihren technischen Details ausgearbeitet wird. Ziel der vorliegenden Exposition ist es unter anderem, diesen irreführenden Eindruck zu korrigieren und zugleich die Unterscheidung zwischen Relationen und Attributen mit Gehalt zu füllen.

Die Idee der im folgenden vorgestellten Grammatik ist es, einen breiten Kernbereich der deutschen Syntax in möglichst kompakter Form und einem modular erweiterbaren Format zu beschreiben. Dabei werden die Konzeption der linguistischen Objekte als Merkmalstrukturen im Gegensatz zu Phrasen- oder Baumstrukturen und die Implikationen der Verwendung von Relationen bei der Beschreibung von Merkmalstrukturen im Mittelpunkt stehen. Der empirische Gegenstandsbereich umfaßt einfache Verberst- (V1), Verbzweit- (V2) und Verbletztsätze (VL), sowie die Behandlung einfacher, vornehmlich adverbialer Adjunkte einschließlich von Adverbialsätzen. Eine einfache Schnittstelle zu einer Illokutionstheorie für Assertionen, Fragen und Exklamationen bildet die Voraussetzung zur Behandlung des Unterschiedes zwischen eingebetteten und uneingebetteten finiten Sätzen der drei Verbstellungstypen. Im Zusammenhang mit komplexen Sätzen werden schließlich kurze und lange Topikalisierung diskutiert sowie der Themenbereich von w-Interrogativsätzen mit kurzer

und langer w-Bewegung angeschnitten. Dies gibt Anlaß, eine Theorie der langen Abhängigkeiten vorzustellen und Inselbeschränkungen für lange Bewegung zu untersuchen. Daneben wird ein Linearisierungsmodul zur Behandlung der Wortstellung im Mittelfeld und von einfachen Extrapositionsdaten vorgestellt. Im Rahmen einer Implementierung (Courbet, De Kuthy, Meurers, Richter, und Sailer 1997) wurden an dieses Grundfragment eine umfassende Analyse des Verbkomplexes von (Meurers 1997) und eine reiche DP-Analyse (Winhart 1997) angebunden und damit seine modulare Erweiterbarkeit demonstriert.

Zu den Grundideen des linearisierungsbasierten Ansatzes in der HPSG gehört es, die syntaktische Baumstruktur von der Wortreihenfolge des Satzes zu dissoziieren und Wortstellungsregeln damit einen eigenständigen anstelle eines von Phrasenstrukturen abgeleiteten Status einzuräumen. Formale Voraussetzung hierfür ist die Betrachtung von linguistischen Objekten als informationsreiche Merkmalstrukturen, deren inhärente Struktur und kombinatorische Eigenschaften für die Formulierung von Wortstellungsregeln zur Verfügung steht. In der HPSG ist diese Bewegung fort von der für Phrasenstrukturgrammatiken typischen Betrachtungsweise insofern natürlich, als daß die traditionelle Baumstruktur in Form von DTRS-Attributen realisiert wird, welche die syntaktische Kombinatorik von Phrasen beziehungsweise Wörtern kodieren, wobei sie sich jedoch logisch betrachtet in keiner Weise von anderen Attributen unterscheiden (etwa im Sinne eines phrasenstrukturellen Rückgrates). Die in den DTRS-Attributen kodierte kombinatorische Information enthält insbesondere genuin keine Information über die lineare Reihenfolge der Wörter, die formal zunächst völlig unabhängig von der Kombinatorik in einem eigenständigen PHON-Merkmal von Zeichen ausgedrückt ist. Die Herstellung eines Zusammenhangs zwischen der Kombinatorik und den linearen Verhältnissen auf der PHON-Liste wird also vom technischen Ansatz her offengelassen, und seine Etablierung im Sinne einer traditionellen Phrasenstrukturgrammatik oder einer Lineariserungsgrammatik ist zunächst eine reine Architekturentscheidung, für die es in keine Richtung ein zwingendes Natürlichkeitsargument gibt. Insofern ist die Entscheidung für die eine oder andere Option letzten Endes im Licht ihrer Konsequenzen für die resultierende Gesamtgrammatik zu bewerten.

3.1. Topologische Felder und Theorie der Wortstellung. Die Herangehensweise bei einer linearisierungsbasierten Grammatik ist unmittelbar oberflächenorientiert. Im Zentrum steht die direkte Beschreibung der

empirisch beobachtbaren Daten der Wortstellung in der Domäne von einfachen Sätzen und nicht ihre Ableitung aus der Beschreibung anderer Strukturen. Da es sich gleichzeitig darum handeln muß, generelle Gesetzmäßigkeiten der Wortstellung zu erfassen, liegt der Rückgriff auf Konzepte der traditionellen deskriptiven Grammatiktheorie nahe. Für den Bereich der Wortstellung des finiten Verbs in den drei Satztypen V1, V2 und VL ist hier das topologische Feldermodell des Deutschen einschlägig, dessen Ursprünge bis ins frühe neunzehnte Jahrhundert zurückreichen (vgl. (Höhle 1986)). Da wir uns lediglich mit einem Kerndatenbereich der deutschen Syntax befassen, schränken wir die Betrachtung auf finite Sätze ein. Außerdem ignorieren wir das K_L-Feld, das für Linksversetzungsdaten und den Nominativus pendens verwendet wird, und die alternativ auftretenden Felder PARORD und KOORD, in denen nicht-koordinierende Partikeln beziehungsweise koordinierende Partikeln stehen. Die in den nachstehenden Tabellen zitierten Sätze sind weitgehend aus (Höhle 1986, S. 330) übernommen.

(3.1) *Topologische Felder in VL-Sätzen*

C	X	VK	Y
ob	wohl jemand den Hund	füttern würde	morgen abend?
daß	du mir ja nicht den Hund	fütterst!	
in dessen Hütte	eine Katze	gefunden wurde	
wer		kommt	

Finite Verbletztsätze müssen obligatorisch von einem als Komplementierer fungierenden Element eingeleitet werden, das dem C-Feld zugewiesen wird. Dabei kann es sich unter anderem entweder um lexikalische Komplementierer (*daß*, *ob*) oder aber auch um eine möglicherweise komplexe Interrogativ- oder Relativphrase (*wer*, *in dessen Hütte*) handeln. Im weiteren werden wir von Relativsätzen und VL-Sätzen mit *je* und Komparativ-Phrase beziehungsweise *so* und Positiv-Phrase im C-Feld absehen. Auf das C-Feld folgt der mit X gekennzeichnete Bereich, der üblicherweise Mittelfeld genannt wird. VK bezeichnet den Verbalkomplex, der bei finiten Verbletztsätzen immer besetzt ist und das finite Verb und eventuell von ihm abhängige infinite Verbformen enthält.[9] Der mit Y gekennzeichnete Bereich wird Nachfeld genannt und kann mit einigen Einschränkungen alle Konstituenten enthalten, die auch im Mittelfeld stehen

[9] Aus Gründen der Einfachheit abstrahieren wir hier von der Einordnung von Verbalkomplexen in das Feldermodell, die ein nichtleeres Oberfeld und gegebenenfalls Nominalphrasen, Präpositionalphrasen oder Adverbiale im Unterfeld aufweisen.

können. Darüber hinaus gibt es einige obligatorische Nachfeldbesetzer. Wie man (3.1) entnehmen kann, können sowohl das Mittelfeld als auch das Nachfeld unbesetzt bleiben.

(3.2) *Topologische Felder in V1-Sätzen*

FINIT	X	VK	Y
würde	denn jemand den Hund	füttern	morgen abend?
kommen	die Leute denn auch,		die du eingeladen hast?
hättest	du mir doch	zugehört	damals!
gib	mir mal einer den Hammer!		

Der Unterschied zwischen dem Feldermodell von V1- und VL-Sätzen ist das FINIT-Feld, das bei V1-Sätzen das C-Feld von VL-Sätzen an der linken Satzperipherie ersetzt. In FINIT steht, wie der Name nahelegt, das finite Verb. Die Gesetzmäßigkeiten für Mittelfeld, Verbalkomplex und Nachfeld bleiben bis auf die Tatsache erhalten, daß nun auch der Verbalkomplex gegebenenfalls leer sein kann.

(3.3) *Topologische Felder in V2-Sätzen*

K	FINIT	X	VK	Y
die Frau	hat	das Buch	gelesen	gestern abend
es	hört	schon	auf	
wer	kommt?			
dann	hör	doch	zu!	

Das topologische Feldermodell für V2-Sätze weist gegenüber (3.1) und (3.2) ein zusätzliches Feld K, das Vorfeld, auf. Das Vorhandensein eines Vorfeldes ist somit eine spezifische Eigenschaft von V2-Sätzen. Das Vorfeld wird im allgemeinen von einer einzigen Konstituente besetzt, für die spezifische Bedingungen gelten. Was die übrigen Felder anlangt, gelten für V1-Sätze und für V2-Sätze die gleichen Gesetzmäßigkeiten. Im besonderen verfügen V2-Sätze ebenfalls über ein FINIT-Feld und weisen kein C-Feld auf.

Anhand des topologischen Feldermodells lassen sich eine Reihe von wichtigen Generalisierungen über finite Sätze im Deutschen aufzeigen. Alle drei Satztypen verfügen über ein Mittelfeld, einen Verbalkomplex und ein Nachfeld, für die jeweils die gleichen Gesetzmäßigkeiten gelten. In allen drei Fällen gibt es Interrogativ-, Deklarativ- und Exklamativsätze. Interessante Unterschiede treten an der linken Satzperipherie auf. Sie besteht aus obligatorisch besetzten Feldern, bei denen es sich alternativ um das C-Feld (3.1), das FINIT-Feld (3.2) oder das K-Feld und das FINIT-Feld (3.3) handelt.

Im ersten, breit angelegten linearisierungsbasierten Ansatz zur deutschen Syntax innerhalb der HPSG, (Kathol 1995), wird das hier skizzierte Feldermodell als Ausgangspunkt der Analyse genommen. (Kathol 1995) weist dem Feldermodell einen realen Status in der Grammatik zu, indem Sorten für topologische Felder eingeführt werden und somit Objekte im linguistischen Universum, welche dem topologischen Felderbegriff in direkter Weise entsprechen. Die Wortstellungsregeln in (Kathol 1995) referieren auf diese topologischen Objekte, und wesentliche Prinzipien der Syntax werden unter unmittelbarer Bezugnahme auf sie formuliert. Wir werden im folgenden einen anderen Weg gehen und topologischen Feldern nicht den Status von linguistischen Objekten zubilligen. Vielmehr nehmen wir an, daß die beobachteten topologischen Felder aus anderen Eigenschaften sprachwissenschaftlicher Objekte abzuleiten sind. Das heißt, wir beginnen bei der Annahme, daß die in den verschiedenen Satztypen involvierten Wörter und Phrasen Eigenschaften, also Attribute und Werte, aufweisen, für die allgemeine Prinzipien der Grammatik gelten, die zu dem beobachteten Wortstellungsmuster führen. Topologische Felder sind damit keine originären Entitäten des linguistischen Universums. Andererseits geben wir in Übereinstimmung mit (Kathol 1995) die Annahme auf, daß Wortstellungsregeln auf lokalen syntaktischen Baumstrukturen operieren. An die Stelle von lokalen Bäumen als Definitionsbereich von Wortstellungsregeln tritt der Begriff der *Permutationsdomäne* (Richter und Sailer 1995). Permutationsdomänen werden in Kapitel 4 im Zuge der Diskussion des Linearisierungsmoduls der Grammatik mit Hilfe von Relationen definiert werden. Einstweilen reicht eine informelle Charakterisierung des Begriffes für finite Sätze aus.

Die Permutationsdomäne oder Wortstellungsdomäne eines finiten Satzes ist die Menge derjenigen Zeichen, deren phonologischer Wert als unverändertes Ganzes in die Phonologie des betrachteten Satzes eingeht. Mit anderen Worten heißt das, daß die Phonologie eines Elementes in der Permutationsdomäne in der durch den PHON-Wert des Zeichens gegebenen Reihenfolge und ohne einen Einschub anderen phonologischen Materials im Satz auftreten muß. Elemente der Permutationsdomäne eines Satzes sind abgeschlossene syntaktische Einheiten wie Nominalphrasen, Präpositionalphrasen, Eigennamen und eingebettete Sätze. Die nachstehenden Beispiele sollen ein Gefühl für den Begriff der Permutationsdomäne vermitteln. Die Klammerung gibt die Phonologie der Elemente der Permutationsdomäne des jeweiligen Satzes an. Die Elemente sind in der Reihenfolge angeordnet, in der ihre Phonologie im Satz erscheint.

(3.4) a. [ob] [wohl] [jemand] [den Hund] [füttern würde] [morgen abend]

b. [in dessen Hütte] [eine Katze] [gefunden wurde]
c. [hat] [die Frau] [das Buch] [gelesen] [gestern abend]

Die in (3.4) durch die Klammerung angedeuteten Zeichen sind als Elemente der Permutationsdomäne Gegenstand von Wortstellungsregularitäten. Diese Wortstellungsregularitäten beziehen sich auf Eigenschaften genau dieser Zeichen und auf Eigenschaften des Gesamtsatzes wie zum Beispiel Skopusverhältnisse. Was uns im folgenden interessiert, ist die Frage, welche Eigenschaften der Zeichen zu den beobachteten topologischen Verhältnissen an der linken Peripherie finiter Sätze führen. Sie sollen im Rahmen unserer HPSG-Analyse aus allgemein gültigen Eigenschaften der beteiligten Elemente der Permutationsdomäne abgeleitet werden.

3.1.1. *VL bei einfachen Komplementierern.* Als erstes betrachten wir von den Komplementierern *daß* und *ob* eingeleitete Verbletztsätze:

(3.5) a. daß du mir ja nicht den Hund fütterst!
b. ob wohl jemand den Hund füttern würde morgen abend?

daß und *ob* müssen als Wörter vom Lexikon der Grammatik lizenziert werden. Unter der Perspektive einer Beschreibungslogik, wie sie in Kapitel 2 vorgestellt wurde, wird das Lexikon einer HPSGrammatik in gleicher Weise wie die übrigen Prinzipien als eine Beschreibung der Theorie behandelt. In der Literatur ist diese Beschreibung unter dem Namen *Word Principle* bekannt (z.B. (Pollard 1993), (Höhle 1994)).

(3.6) *Das Wortprinzip*

$$\left[\begin{array}{l} word \\ \end{array}\right] \quad \Longrightarrow \quad \mathrm{Lex}_1 \vee \mathrm{Lex}_2 \vee \ldots \vee \mathrm{Lex}_n \vee \mathrm{LR}$$

Die Beschreibungen Lex_1 bis Lex_n werden traditionell Lexikoneinträge genannt. Außer durch Lexikoneinträge können Wörter in der Grammatik durch lexikalische Regeln lizenziert werden. Dies wird in (3.6) durch das letzte Disjunkt LR des Konsequens angedeutet. Traditionell werden lexikalische Regeln als Beziehungen zwischen Beschreibungen von Wörtern aufgefaßt und sind damit formal als Metabeschreibungen zu verstehen. (Meurers 1995) zeigt, daß lexikalische Regeln, wie sie in HPSG in der Praxis verwendet werden, im Gegensatz dazu auch einfach als Beschreibungen von Objekten aufgefaßt werden können, die Wörter zueinander in Beziehung setzen. Die beiden Ansätze unterscheiden sich dadurch, daß der eine auf einer Metaebene, der andere aber auf der Beschreibungsebene angesiedelt ist, formal in signifikanter Weise. Da indessen der Vorschlag von Meurers bislang der einzige ist, der speziell für die HPSG tatsächlich

präzise ausgearbeitet wurde, ist ein detaillierter Vergleich ihrer unterschiedlichen Eigenschaften bislang nicht möglich. Wenn wir im folgenden von lexikalischen Regeln sprechen, dann meinen wir lexikalische Regeln im Sinne von (Meurers 1995). Ihr Effekt ist der, daß zu allen Wörtern einer durch den Linguisten in der lexikalischen Regel angegebenen Form weitere Wörter einer anderen dort spezifizierten Form in der Grammatik existieren. Durch das rekursive Format der lexikalischen Regeln kann das Wortprinzip (3.6) potentiell unendlich viele voneinander verschiedene Wörter lizenzieren, obwohl es nur endlich viele Basislexikoneinträge Lex_1 bis Lex_n geben kann.

Einstweilen interessieren wir uns nicht für den semantischen Beitrag der Wörter an der linken Satzperipherie, da er für die syntaktische Wortstellung nicht relevant ist. Daher wird auf die Angabe einer CONTENT-Beschreibung verzichtet. Die im folgenden angegebenen Lexikoneinträge sind in dieser Hinsicht jeweils als unvollständig anzusehen. In einer vollständig ausgeführten Grammatik sind Beschreibungen der semantischen Repräsentation zu ergänzen.

(3.7) *Lexikoneintrag für daß*

$$\left[\begin{array}{l} word \\ \textsc{phon}\ \langle daß \rangle \\ \\ \textsc{ss loc cat} \left[\begin{array}{l} \textsc{head} \left[\begin{array}{l} marker \\ \textsc{fronted}\ empty \\ \textsc{spec} \left[\begin{array}{l} synsem \\ \textsc{loc cat head vform}\ finite \end{array}\right] \end{array}\right] \\ \textsc{marking}\ daß \end{array}\right] \end{array}\right]$$

In Übereinstimmung mit (Pollard und Sag 1994) werden Komplementierer als Marker analysiert. Mittels des SPEC-Attributes selegiert *daß* eine finite Verbprojektion als Kopftochter, und der MARKING-Wert des Satzes wird durch (3.7) in Interaktion mit dem klassischen Markingprinzip (3.40) auf *daß* gesetzt. Die Neuerung in (3.7) besteht in dem für den Kopfwert *marker* in der Signatur definierten neuen Attribut FRONTED, das hier den Wert *empty* trägt. Wir nennen dasjenige Element in der Permutationsdomäne eines Satzes, auf dessen Kopf das FRONTED-Attribut definiert ist, den *Opener* des Satzes. In der Linearisierungskomponente der Grammatik gibt es genau eine Wortstellungsregel, die sich auf das FRONTED-Attribut und damit auf den Opener bezieht:[10]

[10] Zur Formalisierung von (3.8) siehe Kapitel 4, (4.31).

(3.8) *Die Openerregel*

> In einer Phrase steht die Phonologie des Openers vor der Phonologie aller anderen Elemente der Permutationsdomäne.

Mit dem Lexikoneintrag von *daß* (3.7) und der Openerregel (3.8) ergibt sich die Wortstellung des Komplementierers in (3.5a) in Übereinstimmung mit dem topologischen Feldermodell für VL-Sätze.

Der Lexikoneintrag von *ob* ist in wesentlichen Zügen mit dem von *daß* identisch. Da es sich bei *ob* nach den Grundannahmen unserer Analyse um einen Marker handelt, muß sein Kopf das Attribut FRONTED tragen. Lexikalisch wird sein Wert wie im Falle von *daß* mit *empty* spezifiziert. Außerdem trägt *ob* den MARKING-Wert *ob*.

(3.9) *Lexikoneintrag für ob*

$$\left[\begin{array}{l} \textit{word} \\ \text{PHON}\ \langle \textit{ob} \rangle \\ \\ \text{SS LOC CAT} \left[\begin{array}{l} \text{HEAD} \left[\begin{array}{l} \textit{marker} \\ \text{FRONTED}\ \textit{empty} \\ \text{SPEC} \left[\begin{array}{l} \textit{synsem} \\ \text{LOC CAT HEAD VFORM}\ \textit{finite} \end{array}\right] \end{array}\right] \\ \\ \text{MARKING}\ \textit{ob} \end{array}\right] \end{array}\right]$$

Die Überlegungen zur Wortstellung von *ob* sind parallel zu denen von *daß*. In (3.10) werden die Beispiele aus (3.5) wiederholt und mit Klammerung zur Angabe ihrer Permutationsdomäne auf Satzebene versehen.

(3.10) a. [daß] [du] [mir] [ja] [nicht] [den Hund] [fütterst]!
b. [ob] [wohl] [jemand] [den Hund] [füttern würde] [morgen abend]?

Aufgrund der lexikalisch spezifizierten MARKING-Werte der Komplementierer können die von *daß* und *ob* eingeleiteten Sätze von subkategorisierenden Verben unterschieden werden (vgl. die Diskussion in 3.2).

3.1.2. *V1-Sätze.* Verberstsätze beginnen nach dem topologischen Feldermodell mit dem FINIT-Feld, in dem das finite Verb steht. Gemäß der Strategie, die Wortstellung von inhärenten Eigenschaften der beteiligten Elemente der Permutationsdomäne oder Eigenschaften von Sätzen abzuleiten, führen wir eine neue Untersorte von *head* für Verben ein, die in V1-Stellung stehen. Sie verfügen über den HEAD-Wert *verb1*. Die Signatur legt fest, daß *verb1*-Objekte das FRONTED-Attribut tragen und dieses den Wert *empty* hat. Für das Verb *geht* wie in Satz (3.11) ergibt sich damit vorläufig der Lexikoneintrag (3.12).

(3.11) geht der neue Hausmeister schon wieder so früh nach Hause?

(3.12) *Vorläufiger Lexikoneintrag für V1: geht*

$$\left[\begin{array}{l} \textit{word} \\ \text{PHON}\ \langle \textit{geht} \rangle \\ \\ \text{SS LOC CAT} \left[\begin{array}{l} \textit{cat} \\ \\ \text{HEAD} \left[\begin{array}{ll} \textit{verb1} & \\ \text{FRONTED} & \textit{empty} \\ \text{VFORM} & \textit{finite} \\ \text{MOD} & \textit{minus} \end{array}\right] \\ \\ \text{MARKING}\ \ \textit{unmarked} \\ \text{VAL COMPS}\ \langle \text{DP}[\textit{complete}] \rangle \end{array}\right] \end{array}\right]$$

Standardannahmen zur Syntax finiter Verben in HPSG folgend (z.B. (Hinrichs und Nakazawa 1994)) subkategorisiert *geht* sein Subjekt durch die COMPS-Liste. Angaben zum Kasus und Agreementmerkmale werden einstweilen ignoriert.

Unter der Annahme von (3.12) ist zur Wortstellung finiter V1-Verben wie in (3.11) und zur Ableitung der Parallelität des topologischen Feldermodells von VL-Sätzen und V1-Sätzen nichts weiter zu sagen. Die Wortstellung an der linken Satzperipherie folgt unmittelbar aus der Openerregel (3.8). Zwar handelt es sich bei *geht* nicht um einen Marker, sondern um ein *verb1*-Verb, aber die definierende Eigenschaft eines Openers ist das für den Kopfwert definierte FRONTED-Attribut, das hier vorliegt. Da die Openerregel außerdem impliziert, daß ein Satz höchstens einen Opener hat, wird das gemeinsame auftreten eines V1-Verbs und eines Komplementierers ausgeschlossen.

(3.13) a. *daß geht der neue Hausmeister schon wieder so früh heim
b. *ob geht der neue Hausmeister schon wieder so früh heim

Obwohl die vorgestellte lexikalische Lösung auf den ersten Blick ad hoc erscheinen mag, wird sich später zeigen, daß die Unterscheidung von finiten Verben nach ihrem Stellungstyp ohne weitere konstruktionsspezifische Annahmen zu weitreichenden Generalisierungen führt. In Interaktion mit anderen Prinzipien wird sie einer Klassifizierung der Satztypen dienen, die Schnittstelle zu Illokutionsprinzipien bilden und Generalisierungen über die Eigenschaften der Elemente an der linken Peripherie finiter Sätze erlauben, die in anderen Ansätzen nur durch weitere Stipulationen abgeleitet werden konnten.

3.1.3. *V2-Sätze.* Am schwierigsten mit der bisher verfolgten Idee zu vereinbaren scheint auf den ersten Blick das topologische Feldermodell für Verbzweitsätze, wo vor dem Mittelfeld nicht ein einziges Feld positioniert

ist sondern mit dem Vorfeld und dem FINIT-Feld zwei. Es zeigt sich jedoch, daß die Flexibilät einer getypten Merkmalsarchitektur und das starke Ausdrucksmittel der Relationen eine einfache Lösung auch dieses Falles ermöglicht, die darüber hinaus sprachwissenschaftlich attraktive Züge trägt. Betrachten wir zunächst den folgenden vorläufigen Lexikoneintrag (3.15) für das Verb *geht* in V2-Stellung wie in (3.14).

(3.14) der neue Hausmeister geht schon wieder so früh heim

(3.15) *Vorläufiger Lexikoneintrag für V2: geht*

$$
:\left[\begin{array}{ll}
word & \\
\text{PHON} & \boxed{2} \\
\text{SS LOC CAT} & \left[\begin{array}{ll}
cat & \\
\text{HEAD} & \left[\begin{array}{ll}
verb2 & \\
\text{FRONTED} & \left[\begin{array}{ll} moved & \\ \text{CONST PHON} & \boxed{1}\ list \end{array}\right] \\
\text{VFORM} & finite \\
\text{MOD} & minus
\end{array}\right] \\
\text{MARKING} & unmarked \\
\text{VAL COMPS} & \langle \text{DP}[complete] \rangle
\end{array}\right]
\end{array}\right] \wedge
$$

$$
\overset{\exists}{\alpha} x \left[\langle geht \rangle\right] \wedge \texttt{append}(\boxed{1}, x, \boxed{2})
$$

Wie Marker und *verb1*-Verben trägt auch ein Verb der Kopfsorte *verb2* ein FRONTED-Attribut und ist damit ein Opener. Als solcher ist ein V2-Verb Gegenstand der Openerregel (3.8), und seine Phonologie muß am Satzanfang stehen. Damit stellt sich unmittelbar die Frage, wie die Analyse der Tatsache gerecht wird, daß V2-Sätze gerade dadurch charakterisiert sind, daß dem finiten Verb die Vorfeldkonstituente vorangeht. Die Antwort liegt in der Struktur von *verb2*-Verben, deren FRONTED-Wert sich von den bislang besprochenen Markern und von *verb1*-Verben signifikant unterscheidet. Während der FRONTED-Wert der letzteren durchweg *empty* ist, ist der FRONTED-Wert von V2-Verben von der Sorte *moved*. Die Intuition dahinter ist die, daß V2-Verben eine syntaktische Position für bewegte Elemente zur Verfügung stellen. Das heißt, innerhalb von *moved*-Objekten befindet sich eine Stelle, die eine bewegte Konstituente einnehmen kann. Der phonologische Wert von V2-Verben ergibt sich dann aus dem phonologischen Beitrag dieser Konstituente gefolgt vom Eigenbeitrag des Verbs selbst. Im obigen Lexikoneintrag (3.15) von *geht* wird dies durch die append-Relation ausgedrückt,[11] welche $\langle geht \rangle$ mit der Phonologie der bewegten Konstituente konkateniert und das Resultat

[11] Die Relation append erhält die standardmäßige Definition:

als Gesamtphonologie des Lexikoneintrags bestimmt. Wenn man nun annimmt, daß Permutationsdomänen niemals den phonologischen Beitrag von Wörtern aufspalten sondern diese minimale Bestandteile von Permutationsdomänen sind, dann ergibt sich für Satz (3.14) an der linken Peripherie notwendig die folgende Klammerung:

(3.16) [der neue Hausmeister geht] [schon wieder] [so früh] [heim]

In der Konsequenz bedeutet das, daß V2-Verben immer mit phonologisch vorangehender Konstituente auftreten und gleichzeitig dennoch die Openerregel alleine ausreicht, um die Verhältnisse an der linken Peripherie über alle drei Satztypen hinweg korrekt zu beschreiben. Insbesondere folgt wie im Falle von V1-Sätzen aus der in der Openerregel impliziten Einzigkeitsbedingung des Openers in der Permutationsdomäne eines finiten Satzes, daß Komplementierer niemals in Kombination mit V2-Verben auftreten können.

(3.17) a. *daß der neue Hausmeister geht schon wieder so früh heim
b. *ob der neue Hausmeister geht schon wieder so früh heim

Auf die Gesetzmäßigkeiten der Vorfeldbesetzung werden wir anhand des Vorfeld-es in Abschnitt 3.3 dieses Kapitels noch einmal kurz zurückkommen. Die allgemeinen Bedingungen, denen ein Zeichen innerhalb von *moved*-Objekten unterworfen ist, werden im Zusammenhang mit der Theorie der Bewegung ausführlich Gegenstand von Kapitel 5 sein.

3.1.4. *w-Phrasen als Komplementierer.* Bei der Besprechung von VL-Sätzen in Abschnitt 3.1.1 wurden nur die lexikalischen Komplementierer *daß* und *ob* diskutiert. Nun ist es aber eine bekannte Tatsache der deutschen Syntax, daß w-Pronomen und komplexe w-Phrasen im C-Feld von VL-Sätzen ebenfalls als Komplementierer fungieren, obwohl sie beim Auftreten in anderen topologischen Feldern Nominalphrasen beziehungsweise Präpositionalphrasen sind. Man vergleiche hierzu (3.18) und (3.19).

(3.18) a. wer morgen abend denn eigentlich den Hund füttert
b. welcher Hund morgen abend denn eigentlich von Karl gefüttert wird
c. von welcher Professur gestern auf der Party die Rede war

(3.19) a. wer füttert morgen abend denn eigentlich den Hund?

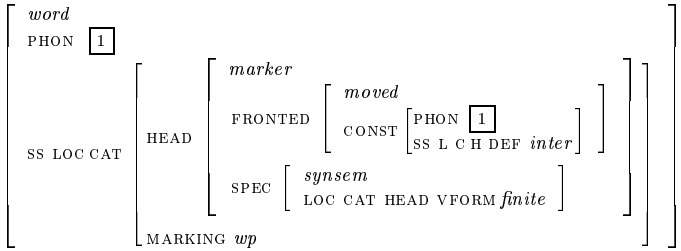
$$
\begin{array}{l}
\texttt{append}(x, y, z) := \\
\quad x \sim elist \\
\quad \wedge\ y \mathrel{\dot{\approx}} z \\
\texttt{append}(x, y, z) := \\
\quad x\text{FIRST} \approx z\text{FIRST} \\
\quad \wedge\ \texttt{append}(x\text{REST}, y, z\text{REST})
\end{array}
$$

b. welcher Hund wird morgen abend denn eigentlich von Karl gefüttert?
c. von welcher Professur war gestern auf der Party die Rede?

Interessanterweise kann es sich bei den als Komplementierer auftretenden w-Phrasen sowohl um einfache Wörter (3.18a) als auch um beliebig komplexe Nominal- beziehungsweise Präpositionalphrasen handeln ((3.18b) und (3.18c)).[12] Vor allem aufgrund der letztgenannten Daten wird man diese Zeichen nicht doppelt kategorisieren und bei einem Aufteten im C-Feld als Marker behandeln wollen. Je nach syntaktischem Kopf der Konstruktion würde das zu der Annahme führen, daß etwa die Präposition *von* oder die Nominalphrase *Hund* je nach syntaktischer Umgebung Marker sein können.

In dem hier verfolgten Wortstellungsansatz bietet sich eine Formalisierung der beobachteten Daten an, die sich eng an Generalisierungen der deutschen Syntaxtheorie anschließt. Zur Beschreibung der Daten in (3.18) führen wir genau einen neuen Lexikoneintrag in die Grammatik ein, den wir w-Marker nennen.

(3.20) *Der Lexikoneintrag des w-Markers*



Der w-Marker trägt das FRONTED-Attribut und ist somit ein Opener. Der FRONTED-Wert ist von der Sorte *moved*, was einem V2-Verb entspricht. Der w-Marker gleicht noch in einer weiteren Beziehung V2-Verben: Er erbt die Phonologie der Konstituente, die sich innerhalb seines *moved*-Objektes befindet. Da er allerdings keinen phonologischen Eigenbeitrag zu seiner Phonologie liefert, erscheint als seine Phonologie einzig und allein die Phonologie der eingebetteten Konstituente. Im Unterschied zu V2-Verben erlegt der w-Marker der Konstituente in *moved* lexikalisch eine Bedingung auf. Es muß sich um eine w-Konstituente handeln, wobei

[12] Von einer Diskussion der Pied-Piping Effekte bei infiniten Konstruktionen sehen wir in diesem Kontext ab. vgl. zu einer Diskussion im Rahmen der HPSG aber (De Kuthy 1996).

sie sowohl ein einfaches w-Pronomen als auch eine komplexe w-Phrase sein kann. Als syntaktisches Kennzeichen einer w-Konstituente wird der syntaktische Definitheitswert *inter(rogativ)* verwendet, der in (Winhart 1997, Kap.2) in die Analyse von Nominalphrasen eingeführt wird. Von seiner Gestalt her hat der w-Marker also große Ähnlichkeit mit V2-Verben und wird damit der oft konstatierten Beobachtung gerecht, daß im Deutschen Komplementierer und finite Verben eine natürliche Klasse bilden. Als Marker teilt der w-Marker zugleich Eigenschaften der Komplementierer *daß* und *ob*, indem er mit seinem SPEC-Attribut eine finite Verbprojektion selegiert. Sein MARKING-Wert is *wp*, was andeutet, daß es sich phonologisch um eine w-Konstituente handelt. Hinsichtlich der Tatsache, daß der w-Marker den FRONTED-Wert *moved* aufweist, verhält er sich zu den Komplementierern *daß* und *ob* wie V2-Verben zu V1-Verben. In Kapitel 5 werden wir außerdem sehen, daß sowohl bei den Konstituenten in V2-Verben wie bei den Konstituenten im w-Marker syntaktische Bewegung involviert ist.

Die Wortstellungsverhältnisse an der linken Satzperipherie folgen nun auch für w-Phrasen, die als Komplementierer fungieren, aus der für V1-Sätze, V2-Sätze und VL-Sätze mit einfachen Komplementierern verwendeten Openerregel (3.8). Der w-Marker kann weder mit anderen Komplementierern noch mit V2- oder V1-Verben zusammen auftreten, da die Permutationsdomäne eines finiten Satzes nach der Openerregel nur höchstens einen Opener enthalten darf. Aus dem gleichen Grund können nicht mehrer w-Marker in einem VL-Satz auftreten. Stehen in der Wortfolge eines Satzes mehrere phonologische Zeichenketten von w-Konstituenten, ist genau die w-Konstituente an der linken Satzperipherie in den w-Marker integriert, wohingegen die anderen ihren phonologischen Beitrag direkt in ihrer Funktion als Nominalphrasen oder Präpositionalphrasen leisten.

3.1.5. *Überblick.* In den vorangehenden Abschnitten wurde eine am topologischen Feldermodell orientierte Linearisierungsanalyse von V2-, V1- und finiten VL-Sätzen vorgestellt. Sie führt unter der Annahme, daß es sich bei sprachlichen Zeichen um komplexe Merkmalstrukturen handelt, die Gesetzmäßigkeiten an der linken Satzperipherie des Deutschen auf gemeinsame Eigenschaften der dort stehenden Wörter und eine einzige Wortstellungsregel der Grammatik zurück.

Die einschlägige Eigenschaft der Wörter an der linken Satzperipherie ist das Vorhandensein eines Attributes FRONTED an ihrem Kopf. Wörter mit diesem Attribut werden Opener genannt. Die relevante Wortstellungsregel besagt, daß der Opener in der Permutationsdomäne eines finiten Satzes

seinen phonologischen Beitrag am Satzanfang leistet. Unter dieser Perspektive läßt sich das topologische Feldermodell vereinfacht so darstellen:

(3.21) *Topologische Felder und Wortstellung*

Opener	Mittelfeld	Verbalkomplex	Extraponiertes Material
C	X	VK	Y
FINIT	X	VK	Y
K FINIT	X	VK	Y

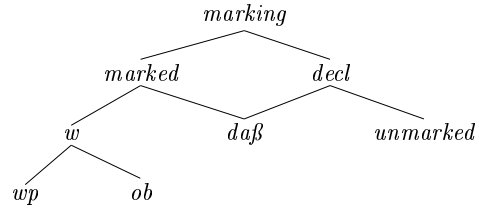
Das C-Feld, das FINIT-Feld bei V1-Sätzen und das K- und das FINIT-Feld bei V2-Sätzen werden als Opener zusammengefaßt. Über die übrigen Felder und die dort geltenden Wortstellungsregeln wurde nichts gesagt. Da sie jedoch für alle drei Satztypen gleich sind, ist klar, daß es sich in jedem Falle um generell für alle deutschen Sätze gültige Regularitäten handeln muß. Insofern kann die Einführung des für alle Satztypen einschlägigen Begriffes des Openers anstelle je nach Satztyp verschiedener Felder als eine Vereinheitlichung des Feldermodells im Rahmen einer Linearisierungsgrammatik angesehen werden.

Bevor wir uns in Abschnitt 3.2 der syntaktischen Baumstruktur und ihren Prinzipien zuwenden, sollen hier noch einige Beziehungen zwischen den oben definierten Openern diskutiert werden.

Wenden wir uns zunächst den Komplementierern zu. Wie bereits erwähnt folgen wir dem Vorschlag von (Pollard und Sag 1994) und analysieren Komplementierer als Marker. Nach (Pollard und Sag 1994) führen Marker mit ihrem auf *cat*-Objekten definierten MARKING-Attribut einen spezifischen *marking*-Wert ein, der Gegenstand des Markingprinzips ist. Die wesentliche Idee dabei ist, daß der Markingwert des Komplementierers auf die Satzebene perkoliert und somit die Art des gegenwärtigen Komplementierers lokal auf Satzebene zugänglich macht. Ein Markingwert *unmarked* auf Satzebene zeigt alternativ an, daß der infrage stehende Satz keinen Marker enthält. Auf diese Weise können Komplementsätze einbettende Verben durch Selektion des Markingwertes Anforderungen hinsichtlich Präsenz und Art eines Komplementierers stellen.

Aufgrund der bekannten Selektionsunterscheidungen zwischen eingebetteten V2-Sätzen, das heißt markerlosen Sätzen, eingebetteten daß-Sätzen und zumeist gemeinsam zulässigen eingebetteten *ob*- und w-VL-Sätzen ist die folgende Sortenhierarchie der Markingwerte naheliegend:

(3.22) *Die Sortenhierarchie unter marking*



Die Obersorte *marking* umfaßt sämtliche möglichen Markingwerte. Die Selektion des Wertes *marked* besagt, daß ein beliebiger Komplementierer gegenwärtig sein muß. *decl* faßt syntaktisch die Sätze zusammen, die semantisch keine Interrogativsätze sind.[13] Schließlich bedeutet eine Subkategorisierung für den Markingwert *w*, daß der eingebettete Satz entweder von *ob* oder von einer w-Konstituente eingeleitet wird. Mit Hilfe der Sortenhierarchie für Markingwerte wird somit eine hierarchische Beziehung zwischen den Komplementierern hergestellt.

Eine scheinbare Uneleganz des hier verfolgten Ansatzes besteht darin, daß es für Verben einer finiten Verbform offenbar drei Lexikoneinträge gibt. So scheint es, als müßte es für *geht* einen V1-Eintrag wie in (3.12), einen V2-Eintrag wie in (3.15) und einen VL-Eintrag geben, der folgendermaßen aussehen müßte:

(3.23) *Vorläufiger Lexikoneintrag für VL: geht*

$$\left[\begin{array}{ll} \textit{word} & \\ \textsc{phon} & \langle \textit{geht} \rangle \\ \textsc{ss loc cat} & \left[\begin{array}{ll} \textit{cat} & \\ \textsc{head} & \left[\begin{array}{ll} \textit{vfinal} & \\ \textsc{vform} & \textit{finite} \\ \textsc{mod} & \textit{minus} \end{array}\right] \\ \textsc{marking} & \textit{unmarked} \\ \textsc{val comps} & \langle \textsc{dp}[\textit{complete}] \rangle \end{array}\right] \end{array}\right]$$

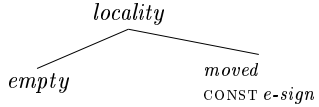
Für finite Verben in VL-Stellung gibt es in Entsprechung zu den beiden Sorten *verb1* und *verb2* eine neue Untersorte von *verb*, *vfinal*. Sie unterscheidet sich von den beiden anderen dadurch, daß das FRONTED-Attribut auf ihr nicht definiert ist. Damit ist für Verben dieser Sorte die Openerregel nicht einschlägig.

[13] Insofern ist diese Sorte (wie auch *w*) beim Hinzufügen einer artikulierten semantischen Struktur von Zeichen auf ihre Redundanz zu hinterfragen und gegebenenfalls zu eliminieren.

Es stellt sich nun bei näherer Betrachtung heraus, daß tatsächlich ein einziger Lexikoneintrag für alle drei finite Verbtypen ausreicht. Das Vorhandensein von Wörtern der drei Arten wird sich als Konsequenz des Aufbaus der Signatur und damit der zulässigen sprachlichen Zeichen ergeben. In diesem Zusammenhang wird zuweilen von (deskriptiver) Unterspezifikation von Lexikoneinträgen gesprochen (vgl. (Frank 1994)).

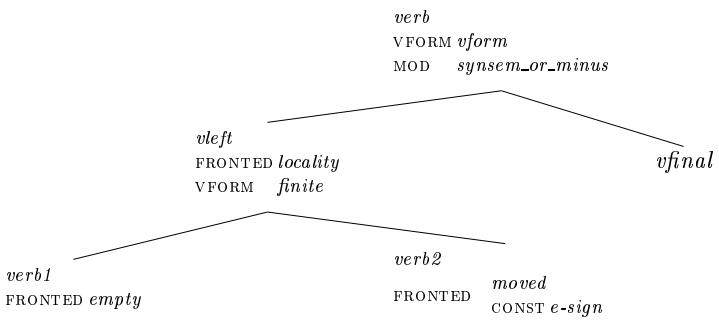
Um zu sehen, wie man dahin kommt, betrachten wir zunächst die Sortenhierarchie des Wertes des auf *marker*, *verb1* und *verb2* definierten FRONTED-Attributes. Die diskutierten Wörter haben zwei verschiedene FRONTED-Werte aufgewiesen: *empty* und *moved*. Aus der Perspektive der Signatur sind sie unter einer Obersorte *locality* angesiedelt.

(3.24) *Die locality-Hierarchie*



empty-Objekte sind atomar, für *moved*-Objekte gibt es ein Attribut CONST, das den Wert *e-sign* hat. Die Verwendung dieser Hierarchie wird sofort klar, betrachtet man die Hierarchie der Kopfsorte von Verben und die auf den darin vorkommenden Sorten definierten Attribute und deren Werte.

(3.25) *Die verb-Hierarchie*



Wie in (Pollard und Sag 1994) ist *verb* ein HEAD-Wert. Im Unterschied zur Signatur von (Pollard und Sag 1994) handelt es sich aber in der gegenwärtigen Grammatik bei *verb* nicht um eine maximal spezifische Sorte. Vielmehr gibt es vier Untersorten von *verb*. Die direkte Untersorte *vfinal* ist maximal spezifisch und erbt alle Attribute von *verb*, ohne weitere hinzuzufügen. Die zweite direkte Untersorte von *verb*, *vleft*, faßt die beiden maximal spezifischen Sorten *verb1* und *verb2* zusammen und zeichnet sich dadurch aus, daß sie den für *verb* angemessenen Attributen FRONTED hinzufügt. Intuitiv bedeutet dies, daß links stehende Verben in jedem Falle Opener sind. Darüber hinaus ist der Wert von VFORM auf *vleft* gemäß der Signatur *finite*, das heißt, linksstehende Verben sind immer finit. FRONTED hat als Wert auf *vleft* keine maximal spezifische Sorte, sondern die Obersorte der Hierarchie (3.24), *locality*.

Für V1- und V2-Wörter im linguistischen Universum ist hingegen der FRONTED-Wert in der Signatur bereits festgelegt. Bei V1-Verben ist er *empty*, bei V2-Verben *moved*. Diese Tatsache kann nunmehr bei der Formulierung eines einzigen Lexikoneintrages für alle drei finiten Verbtypen ausgenutzt werden.

(3.26) *Der Lexikoneintrag für eine finite Verbform: geht*

$$
\begin{bmatrix}
\textit{word} \\
\textsc{ss loc cat} \begin{bmatrix}
\textit{cat} \\
\textsc{head} \begin{bmatrix} \textit{verb} \\ \textsc{vform} \ \textit{finite} \\ \textsc{mod} \ \textit{minus} \end{bmatrix} \\
\textsc{marking} \ \textit{unmarked} \\
\textsc{val comps} \ \langle \textsc{dp}[\textit{complete}] \rangle
\end{bmatrix}
\end{bmatrix} \wedge
$$

$$
\exists x \begin{bmatrix} \langle \textit{geht} \rangle \end{bmatrix}_x \wedge \texttt{construct-phon}(:, x)
$$

Der endgültige Lexikoneintrag des finiten Verbs *geht* (3.26) faßt die vorläufigen Einträge (3.12), (3.15) und (3.23) zusammen. In (3.26) wird somit als HEAD-Wert einfach *verb* angegeben und damit zum Ausdruck gebracht, daß der Eintrag alle drei finite Verbtypen zugleich beschreibt. Das Interessante an (3.26) ist die Verwendung der zweistelligen Relation construct-phon. Sie stellt eine Beziehung her zwischen dem beschriebenen Wort und der Phonologie des Wortes, die im konkreten Beispiel den Bestandteil *geht* enthalten muß. In der Denotation von Beschreibungen befinden sich nur maximal spezifische Objekte. In der Denotation von (3.26) befindet sich also das V1-Verb *geht*, das V2-Verb *geht* und das VL-Verb *geht*. In Abhängigkeit davon, um welches es sich handelt, bestimmt construct-phon seinen PHON-Wert. Ihre definierenden Klauseln sind in (3.27) angegeben.

(3.27) *Die Klauseln für construct-phon*

$$\texttt{construct-phon}(x, y) :=$$
$$\neg\ x\textsc{ss loc cat hd fr const} \approx x\textsc{ss loc cat hd fr const}$$
$$\wedge\ x\textsc{phon} \approx y$$
$$\texttt{construct-phon}(x, y) :=$$
$$x\textsc{ss loc cat hd fr const} \approx x\textsc{ss loc cat hd fr const}$$
$$\wedge\ \texttt{append}(x\textsc{ss loc cat hd fr const phon}, y, x\textsc{phon})$$

Die erste Klausel in (3.27) sagt, daß, wenn auf dem Kopf des Zeichens x nicht die Attributfolge FR CONST definiert ist, seine Phonologie mit y identisch ist. FR CONST ist auf V1-Verben und auf VL-Verben nach dem in (3.25) angegebenen Ausschnitt der Signatur nicht definiert. Für den Lexikoneintrag (3.26) bedeutet das, daß das denotierte V1-Verb und das denotierte VL-Verb die Phonologie $\langle \textit{geht} \rangle$ aufweist.

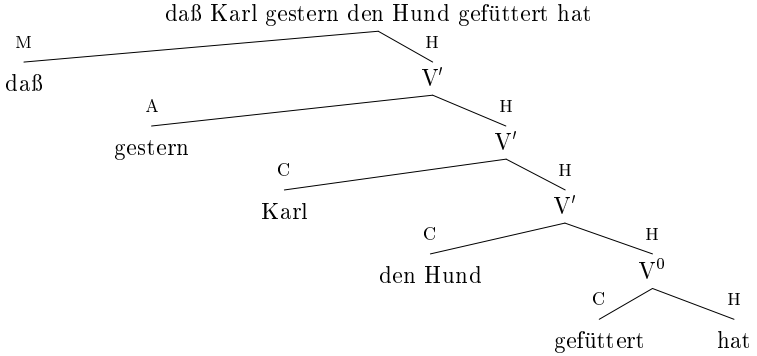
Die zweite Klausel in (3.27) sagt, daß, wenn auf dem Kopf des Zeichens x die Attributfolge FR CONST definiert ist, sich seine Phonologie aus y appendiert an die Phonologie der innerhalb des FRONTED-Wertes befindlichen Konstituente ergibt. Nach (3.25) ist die Attributfolge FR CONST auf dem Kopf von V2-Verben definiert. Somit besteht ihr phonologischer Wert aus der Phonologie der Vorfeldkonstituente plus ihrem phonologischen Eigenbeitrag, der in (3.26) *geht* ist. Damit ist gezeigt, daß aufgrund der Signatur ein einziger Lexikoneintrag ausreicht, um alle drei Typen eines finiten Verbs zu beschreiben. Infolgedessen ist durch die Unterscheidung von drei Stellungstypen von finiten Verben nichts an Eleganz der Grammatik eingebüßt worden.

3.2. Die syntaktische Baumstruktur von Sätzen. In Abschnitt 3.1 wurde die Stellung des finiten Verbs und des Komplementierers an der linken Satzperipherie aus Eigenschaften von Wörtern abgeleitet. Diese Wortstellungstheorie bezog sich auf den Begriff des Openers und auf den Begriff der Permutationsdomäne eines Satzes. Dagegen wurde kein Bezug zur Phrasenstruktur der Sätze hergestellt, wie das in klassischen Phrasenstrukturgrammatiken notwendig der Fall ist. Die vorliegende Grammatik unterscheidet also die drei Satztypen des Deutschen nicht durch Baumstrukturen, sondern in erster Linie durch inhärente Eigenschaften von Zeichen in der Permutationsdomäne von Sätzen. Insofern muß die folgende Definition der Baumstruktur nicht auf Bedürfnisse der Wortstellungstheorie hinsichtlich finiter Verben und Komplementierer Rücksicht nehmen, wie das traditionell der Fall ist. Stattdessen definieren wir eine einfache und einheitliche syntaktische Baumstruktur für Sätze aller

drei topologischen Feldertypen. Auf diese Weise werden die Probleme der Kopfbewegung des finiten Verbs und der verschiedenen Subkategorisierungseigenschaften des bewegten Verbs und seiner Spur bei gleichbleibendem semantischen Beitrag gegenstandslos.

(3.28) ist die Beschreibung der syntaktischen Stuktur eines finiten Verbletztsatzes.

(3.28) *Die Baumstruktur eines VL-Satzes mit Komplementierer*

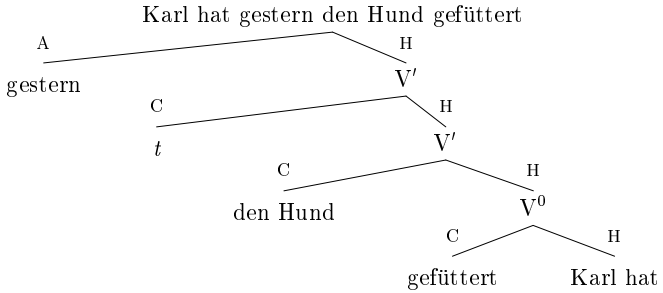


Was den Verbkomplex betrifft, gehen wir von einer Analyse in der Art von (Hinrichs und Nakazawa 1994) aus. Das heißt, das kohärent konstruierende Hilfsverb *hat* erbt von dem subkategorisierten infiniten Verb die nominalen Argumente (Argumentanhebung) und realisiert es als erstes in der syntaktischen Baumstruktur. Da wir aus Gründen der Einheitlichkeit eine universell binär verzweigende Baumstruktur annehmen, werden die angehobenen nominalen Argumente von *hat* oberhalb des Verbkomplexes jeweils einzeln in binären Bäumen realisiert. Das an die jeweiligen Äste annotierte C deutet dabei an, daß diese Nichtkopftöchter an der Kopfprojektion des finiten Verbs aufgrund der Anwendung des Kopf-Komplement-Schemas realisiert werden. Außerdem illustriert (3.28), daß die Komplemente der COMPS-Liste strikt von rechts nach links abgearbeitet werden. Dies wird sich als eine Konsequenz der Formulierung des Subkategorisierungsprinzips (3.36) herausstellen. Eine weitere Konsequenz dieses Prinzips ist es, daß sämtliche Adjunkte baumstrukturell oberhalb aller Komplemente eingeordnet werden. Ihre Wortstellung ergibt sich alleine aus ihrer semantischen Modifikationsbeziehung, ihren inhärenten Eigenschaften und den dafür einschlägigen Wortstellungsregeln und ist nicht von der Baumstruktur abhängig. Das annotierte A deutet in (3.28) die Anwendung des Kopf-Adjunkt-Schemas bei der Realisierung von *gestern* an.

Der Satz wird nach oben vom Marker abgeschlossen, der als Nichtkopftochter M durch das Kopf-Marker-Schema mit der finiten Verbprojektion kombiniert wird.

Der korrespondierende Verbzweitsatz (3.29) unterscheidet sich baumstrukturell betrachtet einzig und allein durch den fehlenden Marker von (3.28).

(3.29) *Die Baumstruktur eines V2-Satzes*



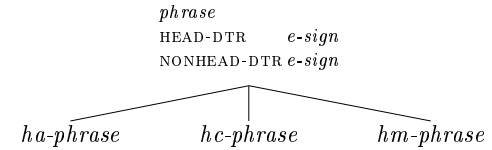
Ein weiterer, jedoch nicht baumstruktureller, Unterschied besteht darin, daß die Vorfeldkonstituente *Karl* in der Baumstruktur eine Spur hinterläßt, zu der sie in der move-α-Relation steht. Die Art und das Zustandekommen der hierbei involvierten Strukturidentitäten werden in Kapitel 5 diskutiert und definiert.

Um zu der illustrierten einheitlichen Satzstruktur zu gelangen, müssen einige zentrale Prinzipien der HPSG definiert und einige Annahmen zur Signatur expliziert werden. Im wesentlichen lehnen wir uns dabei eng an die Vorgaben von (Pollard und Sag 1994) fürs Englische an. Dies wird unter anderem dadurch erleichtert, daß die Behandlung der signifikant anderen Wortstellung des Deutschen im unabhängigen Linearisierungmodul der Grammatik geleistet wird. Somit werden die empirisch motivierten Unterschiede modular an einheitlicher Stelle in der Grammatik konzentriert und habe keine Seiteneffekte auf durch unabhängige Überlegungen motivierte Annahmen über sprachliche Strukturen.

Wie bereits erwähnt nehmen wir für den betrachteten Datenbereich eine einheitlich binär verzweigende Baumstruktur an. Im Zusammenhang damit wird das von (Pollard und Sag 1994) vorgeschlagene DTRS-Attribut auf Phrasen eliminiert und die unter der Sorte *const-struc* definierte Hierarchie auf die Ebene der Phrase angehoben. Als Resultat erhalten wir mehrere Species von unterschiedlichen Phrasen, wobei auf allen, die wir

hier betrachten wollen, die beiden Attribute HEAD-DTR und NONHEAD-DTR definiert sind. (3.30) gibt vereinfachend die Veränderungen an der Signatur wieder.

(3.30) *Die phrase-Hierarchie*



Zur Behandlung von finiten Sätzen werden Kopf-Adjunkt-Phrasen (*ha-phrase*), Kopf-Komplement-Phrasen (*hc-phrase*) und Kopf-Marker-Phrasen (*hm-phrase*) benötigt, die neben den — hier nicht angegebenen — üblichen Attributen für Phonologie (PHON) und syntaktisch-semantische Information (SYNSEM) Attribute für eine Kopftochter und für eine Nichtkopftochter aufweisen. Die Angemessenheitsfunktion legt weiterhin fest, daß sowohl Kopftochter als auch Nichtkopftochter von der Sorte *e-sign* sind, die für eingebettete Zeichen steht. Eingebettete Zeichen stehen uneingebetteten Zeichen gegenüber, die als unabhängige Äußerungen im Diskurs auftreten können und deren hervorragende Exponenten selbständige finite Sätze sind. Die einschlägige vollständige Sortenhierarchie von Zeichen unter *sign* wird Gegenstand der Diskussion der Schnittstelle zur Illokutionstheorie in Abschnitt 5.2 sein.

Die kombinatorischen Möglichkeiten von Zeichen werden in HPSG nicht durch Phrasenstrukturregeln festgelegt, die nicht Teil des allgemein vorausgesetzten und in Kapitel 2 konkretisierten Formalismus sind. Die Stelle von Phrasenstrukturregeln nehmen sogenannte ID-Schemata ein, welche im Gegensatz zu PS-Regeln jedoch keine kombinatorischen Möglichkeiten schaffen, sondern im Gegenteil als Konsequens einer implikativen Beschreibung in der Theorie die durch die Signatur generell erlaubte Kombinatorik auf die empirisch beobachteten Fälle einschränken. (Pollard und Sag 1994) nimmt insgesamt sechs ID-Schemata an, die jeweils ein Disjunkt im Konsequens einer Implikation der Theorie sind. Damit hat das ID-Prinzip folgende Gestalt.[14]

[14] Da (Pollard und Sag 1994) die Phrasentypen anhand der Sortenhierarchie unter ihrem DTRS-Attribut unterscheiden anstatt anhand von Untersorten von *phrase*, liegt dort ein komplexes Antezedens vor, das Phrasen mit dem DTRS-Wert *headed-struc* beschreibt.

(3.31) *Die logische Struktur des ID-Prinzips*

$$\left[\begin{array}{l} \textit{headed-phrase} \end{array}\right] \Longrightarrow$$

$$Desc_{ID_1} \vee Desc_{ID_2} \vee Desc_{ID_3} \vee Desc_{ID_4} \vee Desc_{ID_5} \vee Desc_{ID_6}$$

Wir nehmen an, daß *headed-phrase* eine Untersorte der in (3.30) angegebenen Sorte *phrase* und eine Obersorte der drei Species *ha-phrase*, *hc-phrase* und *hm-phrase* ist. Da Koordination nicht Gegenstand der gegenwärtigen Überlegungen ist, kann im übrigen angenommen werden, daß alle weiteren gegebenenfalls benötigten phrasalen Sorten Untersorten von *headed-phrase* sind. Einstweilen beschränken wir unsere Betrachtung auf die in (3.30) benannten drei Untersorten von *phrase* und geben zu jeder genau ein ID-Schemata an. Das ID-Prinzip besteht somit aus einer implikativen Beschreibung bestehend aus dem Antezedens *headed-phrase* und den drei im folgenden angegebenen Schemata im disjunktiven Konsequens. Da die drei Schemata Beschreibungen von Phrasen verschiedener Species sind und damit notwendig voneinander verschiedene Objekte beschreiben, wird jede zulässige Phrase von genau einem der Schemata beschrieben.

(3.32) *Das Kopf-Adjunkt-Schema*

$$\left[\begin{array}{l} \textit{ha-phrase} \\ \textsc{synsem loc cat val spr} \ \boxed{1} \\ \\ \textsc{head-dtr synsem} \ \boxed{2} \left[\begin{array}{l} \textsc{status} \ \textit{incomplete} \\ \textsc{loc cat val spr} \ \boxed{1} \end{array}\right] \\ \\ \textsc{nonhead-dtr synsem} \left[\begin{array}{l} \textsc{status} \ \textit{complete} \\ \textsc{loc cat head} \left[\begin{array}{l} \textit{adjunct} \\ \textsc{mod} \ \boxed{2} \end{array}\right] \end{array}\right] \end{array}\right]$$

Die Nichtkopftochter einer Kopf-Adjunktphrase muß von der Sorte *adjunct* sein, und sie selegiert mit ihrem MOD-Wert den SYNSEM-Wert der Kopftochter syntaktisch. Der SPR-Wert der Kopftochter perkoliert an der Kopflinie der Projektion nach oben. Im Unterschied zur Standardannahme ist diese Perkolation hier nicht Gegenstand des Subkategorisierungsprinzips, das lediglich die Verhältnisse auf der COMPS-Liste regelt. Das auf *synsem* definierte neue Merkmal STATUS verfügt über die beiden möglichen Werte *complete* und *incomplete*, welche anzeigen, ob ein Zeichen syntaktisch vollständig ist oder sich noch mit weiteren Zeichen zu einem vollständigen Zeichen kombinieren kann. Diese Unterscheidung wird weiter unten für finite Sätze wichtig werden und spielt in der Analyse von Nominalphrasen bei (Winhart 1997) eine entscheidende Rolle. Für den

Augenblick begnügen wir uns mit der Feststellung, daß die Adjunkttochter syntaktisch vollständig und damit zum Beispiel saturiert sein muß, wohingegen die Kopftocher aufgrund der Tatsache, daß sie sich hier mit einem weiteren Zeichen zu einer weiteren Projektion kombiniert, unvollständig ist. Über die syntaktische Vollständigkeit der resultierenden Kopf-Adjunktphrase wird nichts gesagt.

(3.33) *Das Kopf-Komplement-Schema*

$$\left[\begin{array}{l} \textit{hc-phrase} \\ \textsc{synsem loc cat val spr} \ \boxed{1} \\ \\ \textsc{head-dtr synsem loc cat val} \left[\begin{array}{l} \textit{val} \\ \textsc{comps} \ \textit{nelist} \\ \textsc{spr} \ \boxed{1} \end{array}\right] \end{array}\right]$$

Das Kopf-Komplement-Schema fordert von der Kopftochter, daß sie über eine nichtleere Komplementliste verfügt, und regelt die Perkolation des SPR-Wertes entlang der Kopflinie. Eine nichtleere Komplementliste ist Voraussetzung dafür, daß ein Komplement abgearbeitet werden kann. Das Subkategorisierungsprinzip in (3.36) sorgt dafür, daß in dieser Konstellation tatsächlich ein Komplement abgearbeitet werden muß.

(3.34) *Das Kopf-Marker-Schema*

$$\left[\begin{array}{l} \textit{hm-phrase} \\ \textsc{synsem status} \ \textit{complete} \\ \textsc{head-dtr synsem status} \ \textit{incomplete} \\ \textsc{nonhead-dtr synsem} \left[\begin{array}{l} \textsc{status} \ \textit{complete} \\ \textsc{loc cat head} \ \textit{marker} \end{array}\right] \end{array}\right]$$

Das Kopf-Marker-Schema schließlich verlangt, daß die Nichtkopftochter einer *hm-phrase* ein syntaktisch vollständiger Marker und ihre Kopftochter ein beliebiges, syntaktisch unvollständiges Zeichen ist, das durch die Verbindung mit dem Marker zu einem vollständigen Zeichen wird. Das bedeutet, daß die Kopfprojektion nicht über eine *hm-phrase* hinaus syntaktisch fortgesetzt werden kann. Der Marker schließt die Kopfprojektion damit nach oben hin ab. Da nur syntaktisch saturierte Zeichen vollständig sein können (vgl. (3.39)), ist darüber hinaus gewährleistet, daß sich ein Kopf erst dann mit einem Marker verbinden kann, wenn sämtliche Komplemente abgearbeitet sind.

Zur Behandlung einer Analyse weiterer syntaktischer Kombinationsmöglichkeiten in anderen Datenbereichen kann es sinnvoll sein, den betrachteten Untersorten von *headed-phrase* weitere hinzuzufügen und infolgedessen zusätzliche ID-Schemata zu formulieren. Ein Beispiel dafür ist die *hspr-phrase* bei der Analyse von Nominalphrasen in (Winhart 1997).

Das Head Feature Principle, welches die Perkolation des Kopfwertes einer syntaktischen Projektion entlang der Kopflinie sicherstellt, kann unter Berücksichtigung der Tatsache, daß die Kopftochter nun direkt auf der phrasalen Ebene mittels des Attributes HEAD-DTR zugänglich ist, aus der Literatur übernommen werden.

(3.35) *Das Kopfattributprinzip*

$$\left[\ \textit{headed-phrase}\ \right] \Longrightarrow \left[\begin{array}{l}\textsc{synsem loc cat head}\ \boxed{1}\\ \textsc{head-dtr synsem loc cat head}\ \boxed{1}\end{array}\right]$$

Das ursprüngliche Subkategorisierungsprinzip fordert, daß in einer Phrase mit syntaktischem Kopf die COMPS-Liste der Kopftochter besteht aus der Konkatenation der COMPS-Liste der Mutter mit den SYNSEM-Werten der Liste der Komplementtöchter. Im vorliegenden Ansatz sind nun aber sämtliche Strukturen in Phrasen mit Köpfen binärverzweigend, und es wird bei allen drei Species von Phrasen lediglich die Kopftochter von der einzigen Nichtkopftochter unterschieden. In (Pollard und Sag 1994) werden dagegen Adjunkt-, Marker,- Komplement- und Kopftöchter unter verschiedene Attribute eingeordnet. Als Konsequenz dieses Architekturunterschiedes kann unser Subkategorisierungsprinzip nicht analog zum ursprünglichen Subkategorisierungsprinzip Bezug nehmen auf eine von anderen Nichtkopftöchtern eindeutig abgegrenzte Komplementtochter, wenn es nicht in seiner Anwendung auf *hc-phrase* beschränkt wird.

Letzteres wäre allerdings insofern ein unschöner Zug, als daß dann die Tatsache, daß Adjunkte immer oberhalb von Komplementen in der Baumstruktur realisiert werden, in einem gesonderten Prinzip der Grammatik sichergestellt werden müßte. Daher behalten wir als Antezedens des Subkategorisierungsprinzips *headed-phrase* und damit den allgemeinen Fall bei. Die Vermittlung zwischen den COMPS-Listen der Kopftochter, ihrer Projektion und dem SYNSEM-Wert der Nichtkopftochter wird durch die Relation subcat geregelt.

(3.36) *Das Subkategorisierungsprinzip*

$$\left[\ \textit{headed-phrase}\ \right] \Longrightarrow \left[\begin{array}{ll}\textit{headed-phrase} & \\ \textsc{synsem loc cat val comps}\ \boxed{2} & \\ \textsc{head-dtr} & \left[\begin{array}{l}\textit{e-sign}\\ \textsc{synsem loc cat val comps}\ \boxed{1}\end{array}\right]\\ \textsc{nonhead-dtr} & \left[\begin{array}{l}\textit{e-sign}\\ \textsc{synsem}\ \boxed{3}\end{array}\right]\\ \texttt{subcat}(\boxed{1},\boxed{2},\boxed{3}) & \end{array}\right]$$

Intuitiv besagt das Subkategorisierungsprinzip (3.36), daß bei Vorliegen einer nichtleeren Komplementliste in einer *headed-phrase* das letzte Element dieser Komplementliste abgearbeitet werden muß, also mit dem SYNSEM-Wert der Nichtkopftochter identisch ist, und die übrigen Elemente der Komplementliste, sofern es welche gibt, an die Komplementliste der Mutter weitergegeben werden. In der Definition der Klauseln von subcat liegt die Substanz der Aussage des Subkategorisierungsprinzips.

(3.37) *Die Klauseln der Relation subcat*

$$\begin{array}{l}\texttt{subcat}(x,y,z) :=\\ \qquad x \sim \textit{elist}\\ \qquad \wedge\ y \approx x\\ \qquad \wedge\ z \sim \textit{synsem}\\ \texttt{subcat}(x,y,z) :=\\ \qquad x\textsc{rest} \sim \textit{elist}\\ \qquad \wedge\ z \approx x\textsc{first}\\ \qquad \wedge\ y \approx x\textsc{rest}\\ \texttt{subcat}(x,y,z) :=\\ \qquad x\textsc{rest} \sim \textit{nelist}\\ \qquad \wedge\ y\textsc{first} \approx x\textsc{first}\\ \qquad \wedge\ \texttt{subcat}(x\textsc{rest},\ y\textsc{rest},\ z)\end{array}$$

Die erste Klausel von subcat sagt in Verbindung mit dem Relationsaufruf im Subkategorisierungsprinzip (3.36), daß bei leerer Komplementliste der Kopftochter die Komplementliste ihrer Projektion ebenfalls leer und der SYNSEM-Wert der Nichtkopftochter beliebig ist. Es kann sich somit um einen Komplementierer oder ein Adjunkt handeln. Bei Vorliegen einer einelementigen Komplementliste kommt die zweite Klausel zur Anwendung, welche das *synsem*-Objekt auf der Komplementliste mit dem

synsem-Objekt der Nichtkopftochter identifiziert und die Komplementliste der Projektion auf die leere Liste setzt. Die dritte Klausel ist einschlägig, falls die Komplementliste der Kopftochter mehr als ein Element enthält. In diesem Fall wird rekursiv nach dem letzten Element gesucht und sichergestellt, daß alle übrigen Elemente an die Komplementliste der Kopfprojektion weitergegeben werden. Sobald die Rekursion beim letzten Element der Liste ankommt, wird die zweite Klausel angewendet und die Projektion der Elemente der Komplementliste abgeschlossen.

Das Specprinzip ist einschlägig für funktionale Elemente der Grammatik, die im Kopfattribut SPEC gewisse kombinatorischen Anforderungen stellen. Beispiele hierfür sind die Komplementierer *daß*, *ob* und der w-Marker. Die Tatsache, daß das Specattribut synsemwertig ist, bringt die Erwartung zum Ausdruck, daß funktionale Wörter genau ein Element selegieren und nicht optional mehrere oder keines, wie dies bei Nomen oder Verben der Fall sein kann.

(3.38) *Das Specprinzip*

$$\left[\begin{array}{l} \textit{phrase} \\ \text{NONHEAD-DTR SYNSEM LOC CAT HEAD SPEC } \textit{synsem} \end{array}\right] \Longrightarrow$$

$$\left[\begin{array}{l} \textit{phrase} \\ \text{HEAD-DTR SYNSEM } \boxed{1} \\ \text{NONHEAD-DTR SYNSEM LOC CAT HEAD SPEC } \boxed{1} \end{array}\right]$$

Das Specprinzip verlangt, daß die SPEC-Anforderungen einer Nichtkopftochter von der Kopftochter, mit der sie sich verbindet, respektiert werden müssen. Dies wird durch Strukturidentität des *synsem*-Objektes erzwungen. Auf diese Weise kann sich zum Beispiel der Komplementierer *daß* ausschließlich mit einer finiten Verbprojektion verbinden. Damit ist das Specprinzip instrumental, um die vorgestellte Satzstruktur für finite VL-Sätze zu erhalten.

Mit dem Specprinzip wurden alle Annahmen zur Signatur und alle Prinzipien vorgestellt, welche die in (3.28) and (3.29) wiedergegebene einheitliche syntaktische Baumstruktur finiter Sätze garantieren. Bevor wir uns in Abschnitt 3.3 einigen theoretisch interessanten Aspekten der Analyse zuwenden, sollen zwei Prinzipien kurz referiert werden, welche für die syntaktische Gestalt einfacher und komplexer Sätze zentral sind und auf die wir in Kapitel 5 die Analyse der Bewegung aufbauen können.

Das erste Prinzip betrifft den Begriff der syntaktischen Vollständigkeit, der im STATUS-Attribut kodiert wird und für die Theorie des Deutschen neu in die HPSG eingeführt wird. Das Vollständigkeitsprinzip besagt, daß

ein sprachliches Zeichen nur dann syntaktisch vollständig sein kann, wenn es saturiert ist.

(3.39) *Das Vollständigkeitsprinzip*

$$\left[\begin{array}{l} \textit{sign} \\ \text{SS LOC CAT VAL SPR } \textit{nelist} \end{array}\right] \vee \left[\begin{array}{l} \textit{sign} \\ \text{SS LOC CAT VAL COMPS } \textit{nelist} \end{array}\right] \Longrightarrow$$

$$\left[\begin{array}{l} \textit{sign} \\ \text{SS STATUS } \textit{incomplete} \end{array}\right]$$

Das Vollständigkeitsprinzip bezieht sich nur auf die Werte der beiden Valenzattribute COMPS und SPR, die für Komplemente und Spezifizierer verwendet werden. Es berücksichtigt dagegen das SUBJ-Attribut nicht, in dem üblicherweise die syntaktischen Anforderungen von Verben an ihr Subjekt enthalten sind. Diese Auslassung liegt darin begründet, daß mit (Hinrichs und Nakazawa 1994) angenommen wird, daß die Subjekte finiter Verben über die Komplementliste subkategorisiert und unter Anwendung des Subkategorisierungsprinzips für Komplemente wie Nichtsubjekte realisiert werden. Dagegen tragen infinite Verben die Subkategorisierungsinformation für Subjekte alleine auf der SUBJ-Liste. Dieses Element taucht jedoch nicht durch eine Strukturidentität als *synsem*-Objekt einer Tochter in der syntaktischen Baumstruktur auf. Damit folgt, daß die SUBJ-Liste kein Indikator der syntaktischen Vollständigkeit einer Projektion ist.

Das letzte in diesem Abschnitt vorzustellende Prinzip, das Markingprinzip, ist eine Anpassung des gleichnamigen Prinzips aus (Pollard und Sag 1994) an die gegenwärtige Annahmen zur Merkmalsgeometrie von Phrasen. Das Markingprinzip fordert, daß der Markingwert einer Phrase gleich dem Markingwert der Markertochter ist, falls eine solche vorliegt, und gleich dem Markingwert der Kopftochter sonst.

(3.40) *Das Markingprinzip*

$$\left[\begin{array}{l} \textit{hm-phrase} \end{array}\right] \Longrightarrow$$

$$\left[\begin{array}{l} \textit{hm-phrase} \\ \text{SYNSEM LOC CAT MARKING } \boxed{1} \\ \text{NONHEAD-DTR SYNSEM LOC CAT MARKING } \boxed{1} \end{array}\right]$$

$$\neg \left[\textit{hm-phrase} \right] \wedge \left[\textit{phrase} \right] \Longrightarrow$$

$$\left[\begin{array}{l} \textit{phrase} \\ \text{SYNSEM LOC CAT MARKING} \; \boxed{1} \\ \text{HEAD-DTR SYNSEM LOC CAT MARKING} \; \boxed{1} \end{array} \right]$$

Durch das Markingprinzip wird Information über das Vorhandensein und die Art oder das Nichtvorhandensein eines Komplementierers lokal an der obersten Projektion eines Satzes verfügbar gemacht, so daß sie für subkategorisierende Verben zugänglich ist.

3.3. Generalisierungen. In diesem Abschnitt soll kurz auf das Verhältnis der vorgestellten Analyse zu einigen bekannten Eigenschaften der Syntax finiter Sätze eingegangen werden. Ziel ist es anzudeuten, wie sich die neuen Strukturen und Mechanismen in einen größeren Zusammenhang einordnen. Hierzu werden das Vorfeld-es, der Verum-Fokus und Dialekte des Deutschen betrachtet, die in w-VL-Interrogativsätzen neben der w-Konstituente in der linken Peripherie ein *daß* aufweisen.

Während üblicherweise Konstituenten im K-Feld alternativ eine Position im Mittelfeld einnehmen können, ist es eine Besonderheit des Vorfeld-es, daß es ausschließlich im K-Feld von V2-Sätzen auftritt:

(3.41) a. Es kamen fünf Musiker auf die Bühne.
b. *Auf die Bühne kamen es fünf Musiker.
c. *Fünf Musiker kamen es auf die Bühne.
d. *Es kamen sie auf die Bühne.

Während das Vorfeld-es offenbar keine wahrheitsfunktionale Semantik zum Satz beisteuert, steht es in enger Beziehung zum in seiner Gegenwart notwendig rhematischen Subjekt des Satzes, sofern der Satz ein Subjekt enthält. Bei Vorliegen des Vorfeld-es kann das Subjekt nicht pronominal sein (3.41d) und ist vorzugsweise indefinit. Angesichts der Tatsache, daß das Vorfeld-es nicht im Mittelfeld stehen darf, möchte man aus Gründen der Ökonomie gerne annehmen, daß es nicht in einer Binder-Spur-Beziehung zu einer syntaktischen Position steht, die ihre Phonologie gewöhnlich im Mittelfeld realisieren kann.

Kathol (1995, S.200–201) schlägt daher vor, es als Marker zu behandeln, der lexikalisch durch ein entsprechendes topologisches Objekt in seinem DOM-Wert unmittelbar fordert, daß er ein Vorfeldbesetzer ist. Diese Analyse ist hier insofern nicht naheliegend, als daß Marker Opener finiter Sätze sind, welche die Präsenz eines zweiten Openers, wie er mit einem V2-Verb gegeben ist, ausschließen. Ein DOM-Attribut mit topologischen

Objekten, mittels derer die obligatorische Vorfeldbesetzung kodiert werden könnte, gibt es nicht. Auf der anderen Seite ist es eine offene Frage, wie attraktiv es ist, ein eindeutig im Vorfeld eines V2-Satzes stehendes Element als Marker oder, mit anderen Worten, als Komplementierer zu betrachten, wo diese doch im Deutschen in den unstrittigen Fällen gerade nicht zusammen mit linksstehenden finiten Verben auftreten.

Wie in Kapitel 5 erläutert wird, gehen wir davon aus, daß nur Komplemente in Vorfeldposition syntaktisch bewegt werden und damit in Beziehung zu einer Spur stehen. Dagegen sind Adjunkte im Vorfeld nicht bewegt. Es ist ferner klar, daß das Vorfeld-es Zugriff haben muß auf Information über das Subjekt des Satzes, da es dessen semantische Form restringiert. Folgender Lexikoneintrag entspricht diesen Überlegungen:

(3.42) *Der Lexikoneintrag des Vorfeld-es*

$$\boxed{1} \left[\begin{array}{l} \textit{word} \\ \text{PHON} \; \langle es \rangle \\ \text{SS LOC CAT HEAD} \left[\begin{array}{l} \textit{adverbial} \\ \text{MOD} \left[\begin{array}{l} \textit{synsem} \\ \text{LOC CAT HEAD} \left[\begin{array}{l} \textit{verb2} \\ \text{FR} \left[\begin{array}{l} \textit{moved} \\ \text{CONST} \; \boxed{1} \end{array} \right] \end{array} \right] \end{array} \right] \end{array} \right] \end{array} \right]$$

Das Vorfeld-es ist ein Adverbial. Als solches selegiert es mit seinem MOD-Attribut syntaktisch eine Projektion eines V2-Verbs. Die positionale Beschränkung des Vorfeld-es ergibt sich als lexikalische Eigenschaft dieses Wortes selbst, indem es durch die mit dem Tag $\boxed{1}$ ausgedrückte Strukturidentität vom selegierten Verb fordert, daß es die FRONTED-Konstituente eben dieses Verbs ist. Dies schließt die Möglichkeit aus, daß das Vorfeld-es an anderer Stelle als im Vorfeld in sprachlichen Objekten auftreten kann. Aus der Theorie der Wortstellung in Kapitel 4 wird folgen, daß es auch keine strukturidentische Kopie des Vorfeld-es als Adjunkttochter der Kopfprojektion des V2-Verbs geben kann, da es in diesem Fall seinen phonologischen Beitrag zum Satz zweimal leisten würde. Aus der obligatorischen Vorfeldposition folgt, daß das Vorfeld-es insbesondere das V2-Wort selbst selegieren kann, dessen FRONTED-Konstituente es ist. Das bedeutet aber, daß es über die strukturelle Information des Verbs über sein Subjekt seine eigenen Restriktionen hinsichtlich des Subjektes des Satzes geltend machen kann. In der klassischen HPSG-Architektur kann das beispielsweise über das auf der SUBJ-Liste des Verbs stehenden *synsem*-Objekt des Subjekts geschehen, das seine semantische Repräsentation umfaßt.

Im Effekt kommt (3.42) den empirisch beobachteten Erfordernissen nach. Weitere Untersuchungen haben zu klären, worin genau der semantische Beitrag des Vorfeld-es besteht und ob und gegebenenfalls wie er in der vorgeschlagenen Analyse repräsentiert werden kann.

Anhand von Verum-Fokus-Daten läßt sich beobachten, wie das linearisierungstheoretische Konzept des Openers es erlaubt, über Klassen von Daten zu generalisieren, die für phrasenstrukturbasierte Ansätze ein notorisches Problem darstellen. Unter Verum-Fokus versteht man die Hervorhebung der Wahrheit einer Proposition, die sich ergibt, wenn bestimmte Elemente finiter Sätze, deren nähere Charakterisierung Gegenstand der folgenden Diskussion ist, betont werden.[15] (3.43) illustriert Verum-Fokuseffekte in V1- und VL-Sätzen:

(3.43) a. ich weiß nicht, *ob* sie das Buch gelesen hat.
b. ich denke, *daß* er damit aufhört.
c. *hört* er denn damit auf?

In den VL-Sätzen in (3.43a–b) wird Verum-Fokus erreicht, indem jeweils der Komplementierer betont wird; in V1-Sätzen erfordert Verum-Fokus hingegen eine Betonung des Verbs (3.43c). Die Möglichkeit einer Verum-Fokus-Interpretation in VL-Sätzen verschwindet hingegen, wenn das finite Verb betont wird:

(3.44) a. #ich weiß nicht, ob sie das Buch gelesen *hat*.
b. #ich denke, daß er damit auf*hört*.

(3.44a–b) weisen keinen Verum-Fokus auf. Verum-Fokus ist also offensichtlich nicht an das finite Verb gebunden, sondern ein mit der Betonung tatsächlich linksperipherer Elemente finiter Sätze verbundenes Phänomen. In der vorliegenden Analyse der deutschen Satzstruktur ist daher die Generalisierung attraktiv, daß Verum-Fokus eine potentielle Eigenschaft der Opener von Sätzen ist. Mit anderen Worten ist Verum-Fokus dann möglich, wenn der Opener des Satzes durch Betonung hervorgehoben wird. In V1-Sätzen ist aber das finite Verb der Opener, wohingegen

[15] Für eine breitangelegte empirische Diskussion des Verum-Fokus siehe vor allem (Höhle 1992). (Brandt, Reis, Rosengren, und Zimmermann 1992) ordnet Verum-Fokus in einen GB-theoretischen Zusammenhang, (Kathol 1995) in einen von dem hiesigen sehr verschiedenen linearisierungsbasierten HPSG-Ansatz ein. Es ist wichtig zu bemerken, daß bei der Verwendung von Verum-Fokus bestimmte Kontextbedingungen erfüllt sein müssen, auf die hier nicht näher eingegangen werden kann. In den folgenden Beispielen ist mit der Betonung des kursiv gedruckten Wortes stets Verum-Fokus intendiert. Wo dieser auch bei entsprechendem Kontext mit der angegebenen Betonung unmöglich ist, ist dies mit einem # angegeben. In anderen Lesarten sind diese Sätze natürlich durchweg grammatisch.

in VL-Sätzen wie (3.43a–b) die Komplementierer *ob* und *daß* die Opener sind.

Die vorgeschlagene Generalisierung kann in der Tat auf w-VL- und V2-Sätze erweitert werden:

(3.45) a. ich möchte jetzt wissen, *wen* du getroffen hast.
b. ich möchte jetzt wissen, mit *wem* du getanzt hast.
c. du *hörst* jetzt damit auf!
d. # *du* hörst jetzt damit auf!

In w-VL-Sätzen mit einfacher (3.45a) oder komplexer (3.45b) w-Konstituente ist die Betonung dieser Konstituente gegebenenfalls mit Verum-Fokus assoziiert. Der für w-VL-Sätze benötigte w-Marker unterscheidet sich von den Komplementierern *daß* und *ob* dadurch, daß er lexikalisch keinen eigenen phonologischen Beitrag leistet, sondern die Phonologie der FRONTED-Konstituente übernimmt und diese zum Satz beisteuert[16]. Die Betonung der Phonologie seiner FRONTED-Konstituente stellt damit die einzige Möglichkeit dar, dem w-Marker Salienz zu verleihen. Höhle (1992, S.134) beobachtet, daß bei komplexen w-Phrasen zum Erzielen des Verum-Fokus immer das w-Pronomen betont werden muß, auch wenn dieses nicht der Fokusexponent der Phrase ist. In unserem Ansatz ließe sich das so interpretieren, daß auf diese Weise zwischen der Betonung der w-Konstituente und der Betonung des Openers unterschieden werden kann.

Eine verwandte Unterscheidung zwischen der Betonung des Openers an sich und der Betonung in ihn integrierter Konstituenten, die ihren phonologischen Beitrag zum Satz über den Opener leisten, findet sich bei V2-Verben. V2-Verben sind mit w-Markern verwandt, indem auch sie eine FRONTED-Konstituente integrieren. Im Unterschied zu ersteren verfügen sie jedoch über einen distinkten phonologischen Eigenbeitrag, nämlich über die Phonologie, die sie mit ihrer *verb1*- und finiten *vfinal*-Variante teilen und die lexikalisch spezifiziert ist (siehe (3.26)). Folgerichtig tritt Verum-Fokus in V2-Sätzen bei Betonung des finiten Verbs auf (3.45c), wohingegen die Betonung der Vorfeldkonstituente einen völlig anderen Effekt hat (3.45d).

Die Daten in (3.46) bestätigen die Unterscheidung zwischen der Betonung verschiedener phonologischer Bestandteile des Openers in Abhängigkeit von der Rolle, die sie jeweils im Opener und damit im Satz spielen. In w-V2-Interrogativsätzen ist die w-Konstituente zwar wie in w-VL-Interrogativsätzen in den Opener integriert. Der Opener von w-V2-Interrogativsätzen ist jedoch kein Marker sondern ein V2-Wort, das seinen

[16] vgl. hierzu den Lexikoneintrag des w-Markers in (3.20).

eigenen phonologischen Beitrag mitbringt. Folglich tritt Verum-Fokus bei Betonung des Verbs und damit des Openers auf, nicht jedoch bei Betonung der FRONTED-Konstituente, obgleich diese ein w-Element ist.

(3.46) a. wer *hat* denn jetzt gewonnen?
b. #*wer* hat denn jetzt gewonnen?

Die referierten Beobachtungen werden in (3.47) zusammengefaßt.

(3.47) *Generalisierung zum Verum-Fokus*

Verum-Fokus im Deutschen kann nur bei Fokussierung des Openers finiter Sätze auftreten. Der Opener wird fokussiert durch Betonung seines phonologischen Eigenbeitrags, sofern er einen solchen besitzt, und durch Betonung der Phonologie des w-Pronomens der FRONTED-Konstituente sonst.[17]

Damit wenden wir uns dem dritten und letzten Phänomenbereich dieses Abschnittes zu. Im Zusammenhang mit den Wortstellungsphänomenen und topologischen Gegebenheiten an der linken Peripherie finiter Sätze wird in der Literatur häufig die Verletzung des sogenannten Doubly Filled Comp Filters (DCF) in bestimmten deutschen Dialekten, insbesondere in Varianten des Bayrischen, angesprochen.[18] Damit ist gemeint, daß in den betroffenen Dialekten in w-VL-Sätzen eine w-Konstituente an der linken Peripherie gleichzeitig mit einem Element stehen kann, das phonologisch dem daß-Komplementierer gleicht.

(3.48) a. ich frage mich, wen daß du getroffen hast.
b. ich will wissen, mit wem daß du getanzt hast.
c. *ich glaube, mit wem daß du getanzt hast.

Dabei ist besonders aufschlußreich, daß diese Konstruktion unter Prädikaten grammatisch ist, die wie *fragen* obligatorisch einen Interrogativsatz, also zum Beispiel einen w-Interrogativsatz einbetten (3.48a), nicht jedoch unter Verben, die wie *glauben* keine Interrogativsätze erlauben (3.48c).

Diese Daten können durch die Annahme erklärt werden, daß in den einschlägigen Dialekten der w-Marker einen eigenen phonologischen Beitrag mitbringt. Er gleicht dem Standard-w-Marker (3.20) unter anderem darin, daß er nur w-Elemente als FRONTED-Konstituenten zuläßt, ist aber

[17] Von Verum-Fokus in Relativsätzen mit einer nicht-w-Relativphrase wird in (3.47) abgesehen. Eine sinngemäße Erfassung diese Fälle scheint indessen unproblematisch zu sein, sofern Relativpronomen und -konstituenten phonologisch einem am w-Marker orientierten Relativsatz-Opener zugeordnet werden.

[18] Zum Beispiel (Stechow und Sternefeld 1988, S.382), (Müller und Sternefeld 1993, S.485 und 489), (Kathol 1995, S.172–179)

V2-Verben dadurch noch ähnlicher als jener, daß er als seinen phonologischen Beitrag nicht einfach nur die Phonologie der FRONTED-Konstituente übernimmt, sondern sie genau wie V2-Verben vor seinem phonologischen Eigenbeitrag (*daß*) in seine Gesamtphonologie integriert:

(3.49) *Der bayrische w-Marker*

$$
:\left[\begin{array}{l}\mathit{word}\\ \textsc{ss loc cat}\ \left[\begin{array}{ll}\mathit{cat}\\ \textsc{head} & \left[\begin{array}{ll}\mathit{marker}\\ \textsc{fronted const ss loc cat head def} & \mathit{inter}\\ \textsc{spec loc cat head vform} & \mathit{finite}\end{array}\right]\\ \textsc{marking} & \mathit{wp}\end{array}\right]\end{array}\right]
\wedge
\exists x \left[{}^{x}\langle \mathit{daß}\rangle\right] \wedge \texttt{construct-phon}(:, x)
$$

Der MARKING-Wert *wp* des neuen w-Markers erklärt, warum von ihm phonologisch eingeleitete Sätze w-VL-Sätze sind. Als Brücke zwischen V2-Verben und dem standarddeutschen w-Marker kann der phonologisch selbst identifizierbare bayrische w-Marker als weiteres Argument für die Angemessenheit der hier vertretenen Opener-Analyse der linken Satzperipherie angesehen werden.

Die Tatsache, daß der w-Marker (3.49) einen phonologischen Eigenbeitrag besitzt und daher der Opener in Sätzen wie (3.48a–b) separat von der FRONTED-Konstituente fokussiert werden kann, stellt einen Testfall für die Generalisierung (3.47) zum Verum-Fokus bereit. (3.47) sagt vorher, daß die Betonung des phonologischen Eigenbeitrags *daß* des w-Markers zu Verum-Fokus führt, keinesfalls aber die Betonung der FRONTED-Konstituente. Dies wird durch die Daten bestätigt (zitiert nach (Höhle 1992, S.132–133)):

(3.50) a. jetzt möchte ich wissen, mit wem *daß* du getanzt hast.
b. #jetzt möchte ich wissen, mit *wem* daß du getanzt hast.

4. Das Linearisierungsmodul der Grammatik

Das Linearisierungsmodul der Grammatik folgt eng der Architektur, die zuerst in (Richter und Sailer 1995) vorgeschlagen wurde. Charakteristisch für diesen Ansatz ist die große Ausdrucksstärke der Linearisierungskomponente, die vollen Zugriff auf sämtliche Informationen eines sprachlichen Zeichens erhält. In Abschnitt 4.3 werden wir kurz einige Argumente zusammenfassen, welche diese Herangehensweise motivieren. Ein weiteres wichtiges Charakteristikum dieses Ansatzes ist die modulare Organisation der Linearisierung. Sowohl nach außen hin, was die Verwendung von Information aus sprachlichen Zeichen betrifft, als auch intern ist die Linearisierungskomponente modular strukturiert und damit weniger anfällig für Modifikationen der Annahmen über die Strukturierung sprachlicher Zeichen, die sich in Änderungen der Signatur und der Prinzipien der Grammatik manifestieren. Solange die für die Schnittstellen spezifizierten Informationen weiterhin zur Verfügung gestellt werden können, haben Änderungen etwa der Annahmen zur Phrasenstruktur oder zur Theorie des Verbkomplexes keinen Einfluß auf die Formalisierung der eigentlichen Wortstellungsregeln. Gerade für die Entwicklung größerer Grammatikfragmente ist dies von bedeutendem Vorteil.

Im Unterschied zur Linearisierungstheorie von (Richter und Sailer 1995) wird im vorliegenden Ansatz die Merkmalsgeometrie sprachlicher Zeichen nicht erweitert. Die Theorie der Linearisierung ist vielmehr, abgesehen von zwei Prinzipien, die sie an den Strukturen verankern, rein relational formuliert und damit in Klauseln formalisiert. (Richter und Sailer 1995) formalisiert die gesamte Linearisierung durch die Einführung einer großen Anzahl von Junk-Slots, das heißt durch eine Anreicherung der linguistischen Zeichen um reich strukturierte Objekte, für die rekursive Prinzipien geschrieben werden, welche die Linearisierung vollständig als Eigenschaft eigens postulierter Objekte kodiert. Das Abweichen von dieser Konzeption hat sowohl technische als auch intuitive Gründe.

Auf der technischen Seite steht mit den in Kapitel 2 vorgestellten Definitionen inzwischen eine formale Sprache zur Verfügung, die nicht nur den informellen Anforderungen von (Pollard und Sag 1994) nachkommt, sondern gleichzeitig auch einen freien Einsatz von Relationen in der Grammatik erlaubt. Damit sind erst die Voraussetzungen geschaffen, in präziser Weise über die Bedeutung von Relationen in einer HPS-Grammatik zu sprechen. Insofern besteht keine Veranlassung mehr, aus Gründen der technischen Explizitheit den Weg der rekursiven Sortendefinitionen zu gehen.

Was die sprachwissenschaftlichen Intuitionen anlangt, so erscheint es unplausibel, sprachliche Zeichen mit einer Struktur auszustatten, die etwa *category*-Objekten den gleichen Status einräumt wie *append*-Objekten. Während ersteren eine genuin linguistische Idee zum Wesen sprachlicher Objekte entspricht, sind letztere eindeutig durch die formale Notwendigkeit motiviert, eine Beziehung zwischen drei Listen zum Ausdruck zu bringen, nicht jedoch durch das Bedürfniss, *append*-Objekten eine linguistische Realität zuzubilligen. Insofern scheint es ein adäquater Zug zu sein, solche Objekte insgesamt aus dem linguistischen Universum zu verbannen, und die Funktionalität, die mit ihnen erzielt wurde, auf eine Weise zu erreichen, die sich formal von den Postulaten über Struktur und inhärente Eigenschaften von sprachwissenschaftlich interessanten Objekten unterscheidet.

Ein Anliegen der vorliegenden Arbeit zur HPSG fürs Deutsche ist die Reduzierung der Struktur sprachlicher Zeichen auf ausschließlich diejenigen Teile, für die auf der Basis von Intuitionen über jene argumentiert werden kann. Diese Sparsamkeit soll zum einen einer strukturellen Explosion vorbeugen und zum anderen helfen, technische Bedürfnisse von linguistischen zu scheiden. Wortstellungsregeln, die ihrer Natur nach Aussagen über die Reihenfolge von Wörtern und Konstituenten machen, sollen demnach strukturell auch alleine in der Reihenfolge von Wörtern auf der PHON-Liste und in Beziehungen zwischen ihnen zum Ausdruck kommen und nicht in zusätzlichen Objekten, sofern für diese nicht sprachwissenschaftliche Argumente angeführt werden.

Diese Überlegungen motivieren den Verzicht auf eine Entsprechung zum Attribut DOM, welches (Kathol 1995) als zentralen Bestandteil einer Linearisierungsgrammatik des Deutschen auf *sign*-Objekten definiert. Die Idee ist die, daß die Elemente der DOM-Liste, welche die wortstellungsrelevante Information der phonologisch zu ordnenden Zeichen enthalten, aufgrund dieser inhärenten Information linear geordnet werden, so daß ihre phonologische Information in der entsprechenden Reihenfolge auf die Phonologie des Satzes abgebildet werden kann. Ein wesentlicher Bestandteil dieser Strategie ist, daß die Elemente der DOM-Liste mit Information darüber versehen sind, welchem topologischen Feld sie angehören. Wortstellungsregeln, welche auf diese topologischen Objekte referieren, stellen ein Übereinstimmen der möglichen Wortfolgen mit dem topologischen Feldermodell sicher.

In Kapitel 3 haben wir am Beispiel der linken Satzperipherie dafür argumentiert, daß topologische Information kein struktureller Bestandteil deutscher Sätze ist. Das Stellungsmuster des topologischen Feldermodells

kommt vielmehr durch unabhängige Eigenschaften bestimmter Klassen von Wörtern, finiter Verben und Komplementierer, zustande, hinsichtlich derer Wortstellungsregeln sensitiv sind. Damit fallen topologische Objekte in der Grammatik dem Postulat der strukturellen Sparsamkeit zum Opfer. Dies gilt schließlich auch für die DOM-Liste selbst. Das Vorhandensein dieser Liste in einem *sign* drückt aus, daß es eine linguistische Eigenschaft des Zeichens ist, eine bestimmte Reihenfolge der Objekte auf der DOM-Liste aufzuweisen. Nach unserem Verständnis gibt es jedoch keine derartige Eigenschaft sprachlicher Zeichen. Vielmehr verfügen sie über die Eigenschaft einer bestimmten phonologischen Reihenfolge von Wörtern, und diese gilt es in der Grammatik direkt auszudrücken.

Das Kapitel ist folgendermaßen gegliedert. In Abschnitt 4.1 präzisieren wir den Begriff der Permutationsdomäne für finite Sätze, welcher in Kapitel 3 als Grundlage der Aussagen zur Wortstellung herangezogen wurde. Permutationsdomänen stellen die Verbindung her zwischen Phrasenstruktur und Wortstellung und fungieren als eine Schnittstelle zwischen diesen zwei Bereichen der Grammatik. Die formale Definition von Permutationsdomänen ermöglicht es, in Abschnitt 4.2 ein Prinzip anzugeben, welches festlegt, welche phonologischen Elemente in welcher Form Bestandteil einer Phrase sein können. Dieses Phonologieprinzip schafft damit die Rahmenbedingungen für die eigentlichen Wortstellungsregeln, deren Aufgabe es ist, den vom Phonologieprinzip generell zugelassenen Phonologien der Phrasen Beschränkungen aufzuerlegen. Wortstellungsregeln sind also nicht konstruktiv zu verstehen, wie es der Fall wäre, wenn sie im Sinne von Phrasenstrukturregeln bestimmte Strukturen erzeugten. Vielmehr passen sie ins Konzept der Verwendung von Constraints, denen der Raum der ontologisch durch die Signatur zugelassenen Zeichen unterworfen wird. Somit gilt, daß nicht explizit ausgeschlossene Wortreihenfolgen, die dem Phonologieprinzip entsprechen, tatsächlich existierende Wortreihenfolgen sind.

Abschnitt 4.3 gibt einen Überblick über den modularen Aufbau der Linearisierungskomponente, skizziert ihre Schnittstellen zu anderen Teilen der Grammatik und illustriert das allgemeine Format und einige zentrale Details von Wortstellungsregeln, darunter die Relation prec, welche in allen Präzedenzaussagen verwendet wird. Als konkretes Beispiel einer Wortstellungsregel wird in Abschnitt 4.4 die Openerregel (3.8) aus Kapitel 3 formalisiert.

4.1. Zur Permutationsdomäne finiter Sätze. Permutationsdomänen vermitteln zwischen Phrasenstruktur und empirisch beobachteter

Wortreihenfolge. Sie sind damit eines der zentralen Schnittstellenkonzepte zwischen Wortstellungsregeln und den linguistischen Objekten, auf die sie sich beziehen. In ihrer Schnittstellenfunktion führen Permutationsdomänen dazu, daß zum Beispiel eine Modifikation der phrasenstrukturellen Analyse nicht notwendigerweise die Formalisierung von Wortstellungsregeln affiziert. Letztere sind so lange unabhängig von solchen Änderungen, wie es möglich ist, die zur Verfügung gestellte Permutationsdomäne (und eventuell weitere Schnittstelleninformation) in dem für die Wortstellungsregeln spezifizierten Format zu definieren. Dies wird in der Regel nur durch massive Eingriffe in die Gesamtarchitektur unmöglich, die einen tiefgreifenderen theoretischen Wandel anzeigen. Gegenüber kleineren Modifikationen sind die Wortstellungsregeln jedoch immun.

Permutationsdomänen sind keine Eigenschaften von Phrasen und treten deshalb auch nicht als Bestandteile von ihnen auf. Sie werden durch eine zweistellige Relation perm-domain definiert, die zu jeder Phrase ihre Permutationsdomäne angibt. Dies läßt sich so verstehen, daß zu jeder Phrase ihre Permutationsdomäne bekannt ist und für die Formulierung von Beschränkungen verwendet werden kann. Damit ist die Permutationsdomäne ein durch die Struktur des Zeichens implizit bestimmter Bestandteil des Zeichens, ohne daß ihr eine eigenständige Realität in Form eines listenwertigen Attributs zukommt. Diese Unterscheidung füllt die Verwendung von Relationen anstelle von funktional definierten Attributen und *list*-Objekten dort, wo beides technisch möglich ist, mit sprachwissenschaftlichem Gehalt.

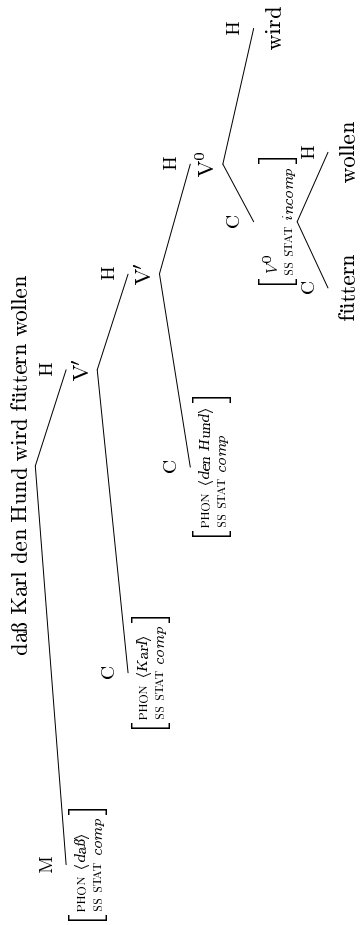
Die Phonologie eines Satzes enthält exakt das Material, das von Elementen seiner Permutationsdomäne stammt (vgl. das Phonologieprinzip (4.14)). Außerdem bestimmt die Permutationsdomäne einer Phrase genau die Zeichen, deren Phonologie als Ganzes ohne den Einschub anderen Materials in die Phonologie des Gesamtzeichens eingeht. Im Falle von finiten Sätzen sind das im allgemeinen Konstituenten wie Präpositionalphrasen, Nominalphrasen, eingebettete infinite und finite Sätze und — in Abhängigkeit von seiner theoretischen Analyse — gegebenenfalls der Verbkomplex, sowie der in Kapitel 3 eingeführte Opener.

Die syntaktische Baumstruktur finiter Verbprojektionen wurde in Abschnitt 3.2 vorgestellt. Für den dort nicht thematisierten Verbkomplex wollen wir für die hiesigen Zwecke eine binär verzweigende Struktur mit den die Argumente erbenden Verben als syntaktischen Köpfen im Sinne von (Hinrichs und Nakazawa 1994) annehmen. Adjunkte sind nach dem Kopf-Adjunkt-Schema (3.32) syntaktisch *complete*. Bei subkategorisierten Komplementen gehen wir davon aus, daß das subkategorisierende Verb die

Anforderung stellt, daß die selegierte Konstituente syntaktisch *complete* ist. Die einzige Ausnahme hiervon bilden subkategorisierte infinite Verben oder Verbprojektionen im Verbkomplex, von denen Argumente an syntaktisch höherstehende Verben vererbt werden und welche *incomplete* sind.

Unter den angeführten Voraussetzungen ergibt sich für einen finiten Verbletztsatz folgendes Bild:

(4.1) *Finiter Verbletztsatz mit Verbkomplex*



Wir nehmen an, daß die Permutationsdomäne von Nominalphrasen wie *den Hund* und vom Verbkomplexkomplement des finiten Verbs (*füttern wollen*) in angemessener Weise definiert ist, und interessieren uns im folgenden nur für die Permutationsdomänen der Projektionen des finiten

Verbs selbst, die zur Bestimmung der Permutationsdomäne des Gesamtsatzes bekannt sein müssen. Die Klauseln der Relation perm-domain sind so definiert, daß sie im ersten Argument eine Liste enthalten, welche die Permutationsdomäne ist, und im zweiten Argument die zugehörige Phrase. Die in (4.2) betrachtete Klausel ist für den Mutterknoten von *wird*, also den höchsten Knoten innerhalb des Verbkomplexes von (4.1), einschlägig.

(4.2) *Nichttrivialer Verbkomplex eines VL-Satzes*

$$\begin{aligned}&\texttt{perm-domain}(x, y) :=\\ &\quad \exists a\ [y\textsc{head-dtr ss loc cat head} \sim \mathit{vfinal}\\ &\quad \wedge\ y\textsc{head-dtr ss loc cat head vform} \sim \mathit{finit}\\ &\quad \wedge\ y\textsc{head-dtr} \sim \mathit{word}\\ &\quad \wedge\ y\textsc{nonhead-dtr ss status} \sim \mathit{incomplete}\\ &\quad \wedge\ y\textsc{nonhead-dtr} \sim \mathit{phrase}\\ &\quad \wedge\ \texttt{perm-domain}(a, y\textsc{nonhead-dtr})\\ &\quad \wedge\ \texttt{cons}(a, y\textsc{head-dtr}, x)]\end{aligned}$$

(4.2) verlangt, daß die Kopftochter der Phrase y ein finites Verb in Verbletztstellung ist. Die Forderung, daß die Nichtkopftochter syntaktisch unvollständig ist, besagt unter den oben angeführten Annahmen zu den Komplementen finiter Verben, daß es sich um ein infinites Verbkomplement handelt, von dem Argumente vererbt werden. In diesem Fall wird verlangt, daß die Permuationsdomäne des Verbkomplexes die Permutationsdomäne der Nichtkopftochter und das finite Verb selbst umfaßt. Die Hilfsrelation cons dient dazu, die Kopftochter an die Liste der Permutationsdomäne der Nichtkopftochter anzuhängen.[19]

Aufgrund von (4.2) enthält die Permutationsdomäne der Mutter V^0 des finiten Verbs *wird* die drei Wörter *füttern*, *wollen* und *wird*, und ihre lineare Abfolgemöglichkeiten sind abhängig von den Wortstellungsregeln, welche die Grammatik für sie enthält. Unter den gegebenen Voraussetzungen ist es grundsätzlich möglich, sie so zu formulieren, daß *wird* genau in Endstellung, im Oberfeld und in der Zwischenstellung auftreten kann:

[19] Die Definition von cons lautet folgendermaßen:

$$\begin{aligned}&\texttt{cons}(x, y, z) :=\\ &\quad x \sim \mathit{elist}\\ &\quad \wedge\ z\textsc{first} \approx y\\ &\quad \wedge\ z\textsc{rest} \sim \mathit{elist}\\ &\texttt{cons}(x, y, z) :=\\ &\quad x\textsc{first} \approx z\textsc{first}\\ &\quad \wedge\ \texttt{cons}(x\textsc{rest}, y, z\textsc{rest})\end{aligned}$$

An die leere Liste x wird ein Element y angehängt, indem eine Liste z mit genau dem Element y zurückgegeben wird. Um y an die nichtleere Liste anzuhängen, wird das erste Elment von x an z übergeben und y an den Rest von x angehängt.

(4.3) a. daß Karl den Hund füttern wollen wird
b. daß Karl den Hund wird füttern wollen
c. daß Karl den Hund füttern wird wollen

Die Frage, ob eine solche Analyse zufriedenstellende Generalisierungen über die Wortstellungsmöglichkeiten im Verbkomplex zuläßt, muß an dieser Stelle offen bleiben. Es sollte jedoch klar sein, daß die angeführten Daten zumindest deskriptiv erfaßt werden können.

Wenden wir uns nun den finiten Verbprojektionen oberhalb des Verbkomplexes zu. Der niedrigste dieser Knoten dominiert neben dem Verbkomplex genau eine Konstituente, deren Phonologie im Mittelfeld (oder Nachfeld) steht. Die Permutationsdomäne dieses Knotens soll weder die Permutationsdomäne des Verbkomplexes noch die der dominierten syntaktisch vollständigen Konstituente übernehmen, da deren Phonologien nicht diskontinuierlich in der Gesamtphonologie des Satzes auftreten können. Es ist also erforderlich, die Gesamtzeichen der beiden Töchter auf die Permutationsdomäne des V′-Knotens zu setzen, der *den Hund* und den Verbkomplex unmittelbar dominiert.

(4.4) *Mittelfeldprojektion, Kopf ist ein phrasaler Verbkomplex*

$$\begin{aligned}&\texttt{perm-domain}(x, y) :=\\ &\quad \exists a\ [y\textsc{head-dtr ss loc cat head} \sim \mathit{verb}\\ &\quad \wedge\ y\textsc{head-dtr} \sim \mathit{phrase}\\ &\quad \wedge\ y\textsc{nonhead-dtr ss status} \sim \mathit{complete}\\ &\quad \wedge\ \texttt{perm-domain}(a, y\textsc{head-dtr})\\ &\quad \wedge\ \texttt{vk}(a)\\ &\quad \wedge\ x\textsc{first} \approx y\textsc{head-dtr}\\ &\quad \wedge\ x\textsc{rest first} \approx y\textsc{nonhead-dtr}\\ &\quad \wedge\ x\textsc{rest rest} \sim \mathit{elist}]\end{aligned}$$

Die Kopftochter des in Frage stehenden Knotens ist eine phrasale Verbprojektion, und seine Nichtkopftochter ist syntaktisch vollständig, das heißt, der Knoten ist nicht Teil des Verbkomplexes. Die Relation vk überprüft, ob die Permutationsdomäne der Kopftochter von einer Form ist, die anzeigt, daß es sich bei ihr tatsächlich um denjenigen Knoten handelt, welcher die höchste Projektion des finiten Verbs innerhalb des Verbkomplexes ist. vk wird auf die Permutationsdomäne der Kopftochter angewandt. Es wird vorausgesetzt, daß infinite Verben den Kopfwert *vfinal* besitzen.

(4.5) *Die Klausel für vk*

$$\begin{aligned}&\texttt{vk}(x) :=\\ &\qquad x\textsc{first ss status} \sim \mathit{incomplete}\\ &\qquad \wedge\ x\textsc{first ss loc cat head} \sim \mathit{vfinal}\\ &\qquad \wedge\ (x\textsc{rest} \sim \mathit{elist} \vee \texttt{vk}(x\textsc{rest}))\end{aligned}$$

(4.5) trifft auf Listen beziehungsweise Ketten zu, deren Elemente alle syntaktisch unvollständige Zeichen mit dem Kopfwert *vfinal* sind. Dies trifft insbesondere auf die Permutationsdomäne des Verbkomplexes in (4.1) zu. Da Komplemente des finiten Verbs oberhalb des Verbkomplexes jedoch vollständig sein müssen, kann vk nicht auf Permutationsdomänen zutreffen, auf denen solche Komplemente von Verben oder Adjunkte beziehungsweise Komplementierer stehen.

Man bemerke, daß die Eigenschaft, der genau den Verbkomplex modellierende Knoten zu sein, nicht in einem neu einzuführenden Merkmal kodiert werden mußte, sondern aus den Elementen auf der Permutationsdomäne des Knotens abgeleitet werden konnte. Wie wir sogleich sehen werden, kann die Relation vk überdies herangezogen werden, um die Permutationsdomänen aller über dem gerade betrachteten Knoten liegenden Projektionen des finiten Verbs zu bestimmen.

(4.6) *Mittelfeldprojektion, Kopf ist eine Phrase aber kein Verbkomplex*

$$\begin{aligned}&\texttt{perm-domain}(x, y) :=\\ &\qquad \exists a\ [y\textsc{head-dtr ss loc cat head} \sim \mathit{verb}\\ &\qquad \wedge\ y\textsc{head-dtr} \sim \mathit{phrase}\\ &\qquad \wedge\ y\textsc{nonhead-dtr ss status} \sim \mathit{complete}\\ &\qquad \wedge\ \texttt{perm-domain}(a, y\textsc{head-dtr})\\ &\qquad \wedge\ \neg\ \texttt{vk}(a)\\ &\qquad \wedge\ \texttt{cons}(a, y\textsc{nonhead-dtr}, x)]\end{aligned}$$

Bei allen den Verbkomplex nicht unmittelbar dominierende Projektionen des finiten Verbs handelt es sich um solche, deren Kopftochter eine verbale Phrase und deren Nichtkopftochter syntaktisch vollständig ist. Darüber hinaus enthält die Permutationsdomäne der Kopftochter a mindestens ein Zeichen, das den STATUS-Wert *complete* aufweist. Daher trifft die Negation von vk(a) zu. Die Permutationsdomäne des betrachteten Knotens besteht dann aus der Nichtkopftochter angehängt an die Permutationsdomäne der Kopftochter.

Für den Beispielsatz (4.1) gelangen wir mit den vorgestellten Klauseln von perm-domain zu einer Permutationsdomäne, deren Gestalt in (4.7) durch Klammerung angegeben ist. Die Elemente der Permutationsdomäne sind in einer Reihenfolge angeordnet, in der ihre Phonologie im Satz erscheinen kann.

(4.7) [daß] [Karl] [den Hund] [wird füttern wollen]

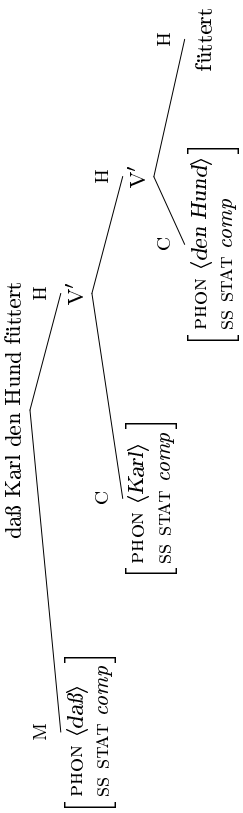
Die Anfangsstellung der Phonologie von *daß* im Satz wird durch die Openerregel sichergestellt. Die relative interne Abfolge der Phonologien des Verbkomplexes und der Nominalphrase *den Hund* ist Gegenstand von Wortstellungsregeln, die auf die jeweilige Domäne angewendet werden können. Da beide abgeschlossene Permutationsdomänen bilden, darf im Satz allerdings keine fremde Phonologie in diese Domänen verwoben werden.

(4.8) a. *daß Karl wird füttern den Hund wollen
b. *daß den Karl Hund füttern wollen wird
c. *daß Karl den wird Hund füttern wollen

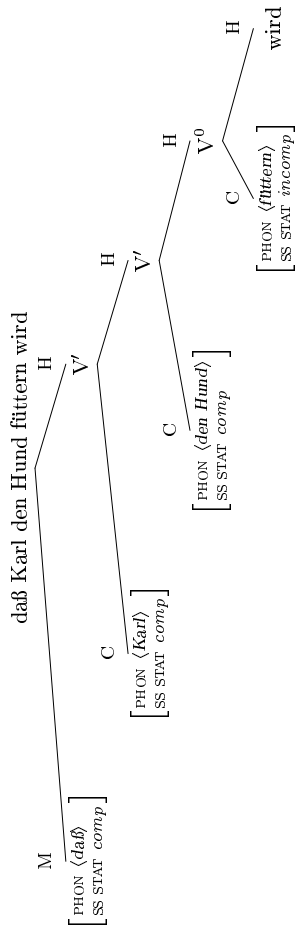
Mit dem Beispiel (4.1) wurden mögliche Baumstrukturen finiter Sätze ignoriert, deren Permutationsdomäne nicht mit den bisher angegebenen Klauseln beschrieben werden kann. Dabei handelt es sich um finite Verbletztsätze, deren Verbkomplex entweder nur ein einziges Verb oder genau zwei Verben enthält, und um Sätze, in denen die Phonologie des finiten Verbs in Anfangsstellung steht (V1- und V2-Sätze).

Der Fall der Verbletztsätze liegt in (4.9) beziehungsweise (4.10) vor.

(4.9) *Finiter Verbletztsatz mit einfachem Verbkomplex*



(4.10) *Finiter Verbletztsatz mit zwei Verben im Verbkomplex*



Bei einem einzigen finiten Verb im Verbkomplex wie in (4.9) werden dieses Verb selbst und die Nichtkopftochter in die Permutationsdomäne der Mutter übernommen. Das gleiche trifft auf die Mutter V^0 des finiten

Verbs in (4.10) zu, welche die höchste Projektion des finiten Verbs innerhalb des Verbkomplexes ist. Im Unterschied zur Klausel (4.4), die den Fall des einen phrasalen Verbkomplex unmittelbar dominierenden Knotens behandelt, kann sowohl in (4.9) (wo wir ebenfalls den den Verbkomplex unmittelbar dominierenden Knoten betrachten) als auch in (4.10) bei der Behandlung des V^0-Knotens nicht auf die Permutationsdomäne der Kopftochter referiert werden. Dies liegt daran, daß es sich hier jeweils um Wörter — nämlich um das finite Verbletztverb — handelt, welche intuitiv über keine Permutationsdomäne verfügen, so daß dieser Begriff nicht auf ihnen definiert ist. Technisch wäre eine solche Lösung durchaus denkbar und könnte dazu führen, einige Fallunterscheidungen bei der Definition der Klauseln von perm-domain zu eliminieren. Für eine linguistische Theorie scheint diese Option jedoch intuitiv unattraktiv zu sein.

Für den Mutterknoten von *wird* in (4.10) ist außerdem auch die Klausel (4.2) nicht einschlägig, welche die höchste Projektion des finiten Verbs innerhalb des Verbkomplexes von Beispiel (4.1) behandelt, da dort die Nichtkopftochter phrasal ist, in (4.10) hingegen nicht. Damit treffen die soeben angestellten Überlegungen zur Definition von Permutationsdomänen für Wörter auch auf die mögliche Anwendung von (4.2) auf den V^0-Knoten in (4.10) zu.

(4.11) *Mutterknoten des finiten vfinal Verbs bei einfachem Verbkomplex*

$$\begin{aligned}&\texttt{perm-domain}(x, y) := \\ &\quad y\textsc{head-dtr ss loc cat head} \sim \mathit{vfinal} \\ &\quad \wedge\ y\textsc{head-dtr ss loc cat head vform} \sim \mathit{finit} \\ &\quad \wedge\ y\textsc{head-dtr} \sim \mathit{word} \\ &\quad \wedge\ \neg\ (y\textsc{nh-dtr} \sim \mathit{phrase} \wedge y\textsc{nh-dtr ss status} \sim \mathit{incomplete}) \\ &\quad \wedge\ x\textsc{first} \approx y\textsc{head-dtr} \\ &\quad \wedge\ x\textsc{rest first} \approx y\textsc{nh-dtr} \\ &\quad \wedge\ x\textsc{rest rest} \sim \mathit{elist}\end{aligned}$$

Klausel (4.11) leistet die Beschreibung der zwei problematischen Fälle. Zum einen wird vorausgesetzt, daß die Kopftochter ein finites Verbletztverb ist. Für (4.9) ist weiterhin relevant, daß Verbkomplemente abgesehen vom Verbkomplex ebenso wie Adjunkte und Komplementierer immer *complete* sind. Damit stehen *füttert* und *den Hund* in der Permutationsdomäne des den nur aus *füttert* bestehenden Verbkomplex dominierenden Knotens V′. Für den V^0-Knoten in (4.10) ist entscheidend, daß bei einem Verbkomplex bestehend aus genau zwei Verben die Nichtkopftochter immer ein Wort ist. Somit gelangen *füttern* und *wird* auf die Permutationsdomäne von V^0.

Damit bleibt die Frage der Behandlung von V1- beziehungsweise V2-Sätzen. Diese ist tatsächlich einfach, da *vleft*-Verben niemals Teil des Verbkomplexes sind und daher keine Fallunterscheidung nötig ist. Die Permutationsdomäne ihrer Mutter besteht grundsätzlich aus ihnen selbst und der Nichtkopftochter:

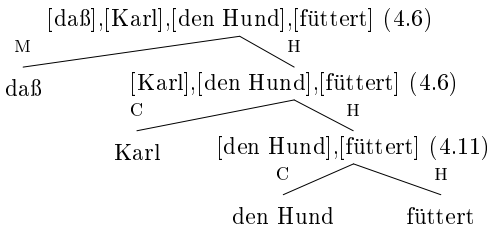
(4.12) *Verbkomplex: vleft-Satz*

$$\begin{aligned}&\texttt{perm-domain}(x, y) := \\ &\quad y\textsc{head-dtr ss loc cat head} \sim \mathit{vleft} \\ &\quad \wedge\ y\textsc{head-dtr} \sim \mathit{word} \\ &\quad \wedge\ x\textsc{first} \approx y\textsc{head-dtr} \\ &\quad \wedge\ x\textsc{rest first} \approx y\textsc{nonhead-dtr} \\ &\quad \wedge\ x\textsc{rest rest} \sim \mathit{elist}\end{aligned}$$

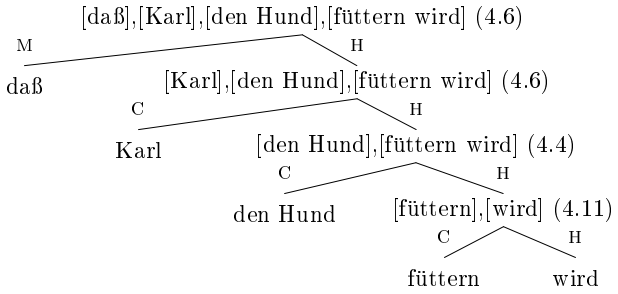
Bevor wir uns im folgenden Kapitel dem Phonologieprinzip zuwenden, geben wir in (4.13) eine abschließende Übersicht über die Permutationsdomänen der Projektionen finiter Verben in allen drei Satztypen, die in diesem Abschnitt definiert wurden. Hierzu geben wir die Beschreibungen der jeweiligen Baumstruktur und notieren an der Stelle der Knoten im Baum die Elemente der mit ihnen assoziierten Permutationsdomäne. Außerdem geben wir die Nummer derjenigen Klausel der Definition von perm-domain an, welche die jeweilige Permutationsdomäne lizenziert. Aus Gründen der Lesbarkeit werden die Elemente der Permutationsdomäne in einer Reihenfolge angeordnet, in der ihre Phonologie im Satz erscheinen kann. Der Leser sollte dabei aber im Auge behalten, daß die Reihenfolge der Elemente in Permutationsdomänen für die Wortstellung irrelevant ist und nicht notwendigerweise der angegebenen entspricht.

(4.13) *Zusammenfassung*

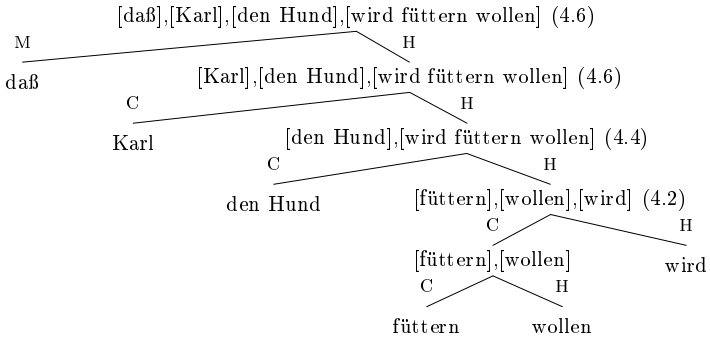
Finiter Verbletztsatz mit einelementigem Verbkomplex:



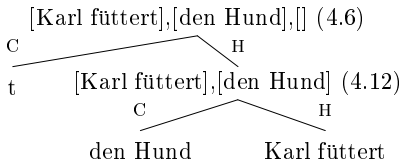
Finiter Verbletztsatz mit zwei Verben im Verbkomplex:



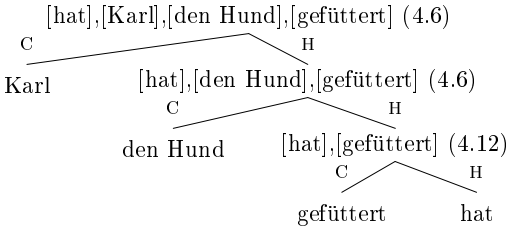
Finiter Verbletztsatz mit großem Verbkomplex:



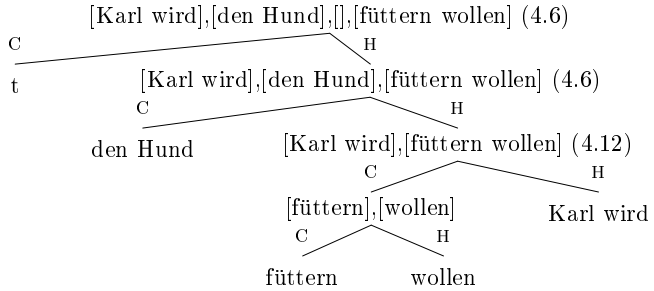
Verbzweitsatz mit leerem Verbkomplex:



Verberstsatz mit einem Verb im Verbkomplex:



Verbzweitsatz mit nichttrivialem Verbkomplex:



In einer vollständigen Grammatik muß zu jeder Phrase ihre Permutationsdomäne spezifiziert sein. In diesem Abschnitt haben wir nur die Permutationsdomänen der Projektionen des finiten Verbs betrachtet. Im folgenden werden wir voraussetzen, daß perm-domain für alle Phrasen adäquat definiert ist.

4.2. Die Phonologie von Phrasen. Solange die Theorie der Grammatik keine Beschreibungen enthält, die etwas zur Phonologie von Phrasen sagen, kann deren PHON-Liste einen beliebigen Inhalt haben. Er wird im allgemeinen nichts mit den Wörtern zu tun haben, die als Blätter der syntaktischen Baumstruktur des Zeichens auftreten.

Das Phonologieprinzip (4.14), das in diesem Abschnitt vorgestellt wird, schränkt die möglichen Phonologien von Phrasen beträchtlich ein. Es besagt, daß die Phonologie einer Phrase irgendeine Permutation der Phonologien der Elemente auf ihrer Permutationsdomäne ist. Aufgrund der Art und Weise, wie in Abschnitt 4.1 Permutationsdomänen definiert wurden, folgt daraus, daß eine Phrase nur phonologisches Material von Wörtern enthalten kann, die an irgendeiner Stelle Bestandteil dieser Phrase sind.

Eine Einschränkung wie das Phonologieprinzip ist selbst noch keine Wortstellungsregel. Sie gibt allerdings die Menge derjenigen Elemente vor, welche die eigentlichen Wortstellungsregeln zu ordnen haben. Gleichzeitig entbindet es die Wortstellungsregeln davon, Aussagen darüber zu machen, was mögliche Bestandteile der Phonologie sind. Indem das Phonologieprinzip die Menge der grundsätzlich sinnvollen PHON-Werte von Phrasen spezifiziert, wird es Wortstellungsregeln überlassen, die darunter enthaltenen nicht grammatischen Wortfolgen auszuschließen. Das heißt insbesondere auch, daß alle Wortfolgen, die nicht explizit durch Wortstellungsregeln ausgeschlossen werden, von der Grammatik zugelassen sind.

Das Phonologieprinzip trifft auf alle Phrasen in der Grammatik zu. Dagegen hat es zur Phonologie von Wörtern nichts zu sagen, da deren Phonologie durch Lexikoneinträge bestimmt wird. Dies trifft auch auf V2-Verben und w-Marker zu, die in ihre Phonologie fremde Phonologien integrieren, da sie dies in einer lexikalisch spezifizierten Weise tun. Diese Tatsache ist unabhängig davon, daß zum Beispiel die Vorfeldkonstituente, sofern sie phrasal ist, ihrerseits durchaus dem Phonologieprinzip unterworfen ist.

(4.14) *Das Phonologieprinzip*

$$:\left[\textit{phrase}\right] \quad \Longrightarrow \quad :\begin{bmatrix}\textit{phrase}\\ \textsc{phon}\ \boxed{1}\\ \texttt{build-phon}(:,\boxed{1})\end{bmatrix}$$

Das Phonologieprinzip macht Gebrauch von einer Relation build-phon, die auf der Grundlage einiger weiterer Relationen definiert ist. Alle im folgenden vorgestellten Relationen sind relationale Varianten von Junk-Slots, die in (Richter und Sailer 1995) zu genau den gleichen Zwecken definiert werden. Aufgrund des transparenten parallelen Aufbaus verzichten wir auf Korrektheitsbeweise unserer Behauptungen zu den Eigenschaften dieser Relationen. Sie werden in (Richter und Sailer 1995, S. 34–47) für die entsprechenden Junk-Slots geführt.

(4.14) sagt in der gegebenen Fassung, daß für jede Phrase die build-phon-Relation zwischen dieser Phrase und ihrer Phonologie besteht. Aufgrund der Definition der Klausel von build-phon heißt das, daß die Phonologie jeder Phrase eine Permutation der Phonologie der Elemente ihrer Permutationsdomäne ist.

(4.15) *Die Klausel der Relation build-phon*

$$\begin{aligned}&\texttt{build-phon}(v_1, v_2) :=\\ &\qquad \exists a \exists b\ [\texttt{perm-domain}(a, v_1)\\ &\qquad \wedge\ \texttt{perm}(b, a)\\ &\qquad \wedge\ \texttt{signphon}(v_2, b)]\end{aligned}$$

Beim Aufruf von build-phon in (4.14) enthält die Variable v_2 die Phonologie der Phrase und v_1 die Phrase selbst. Aufgrund der Definition von perm-domain wissen wir darüber hinaus, daß a die Permutationsdomäne der Phrase v_1 ist.[20] Die Relation perm ist so definiert, daß b eine beliebige Permutation von a ist. signphon nimmt diese Permutation b in der von perm gegebenen Reihenfolge und appendiert ihre Phonologien in v_2. Angenommen, b ist eine Liste von Zeichen mit zwei Elementen, deren erstes die Phonologie ⟨*die*, *Katze*⟩ und deren zweites die Phonologie ⟨*schläft*⟩ hat. Dann steht diese Liste in der signphon-Relation zu der phonologischen Liste ⟨*die*, *Katze*, *schläft*⟩.

Auf der Basis der Relationen perm-domain, perm und signphon sagt build-phon, daß die Phonologie v_2 einer endlichen Phrase v_1 mit endlicher Permutationsdomäne a eine beliebige Permutation der Gesamtphonologien der Elemente ihrer Permutationsdomäne a ist. Wir setzen voraus, daß perm-domain für jede Phrase in der Grammatik entsprechend definiert ist. Im folgenden wenden wir uns den Definitionen der Relationen perm und signphon zu, geben einige Erläuterungen zu technischen Details und motivieren informell, daß sie tatsächlich leisten, was wir für das Phonologieprinzip angenommen haben.

(4.16) *Die Klauseln der Relation perm*

$$\begin{aligned}&\texttt{perm}(v_1, v_2) :=\\ &\qquad v_2 \sim \textit{elist}\\ &\qquad \wedge\ v_1 \sim \textit{elist}\\ &\texttt{perm}(v_1, v_2) :=\\ &\qquad \exists x\ [\texttt{take-out}(v_2, v_1\textsc{first}, x)\\ &\qquad \wedge\ \texttt{perm}(v_1\textsc{rest}, x)]\end{aligned}$$

perm ist in zwei Klauseln definiert. Die erste sagt, daß die leere Liste v_1 eine Permutation der leeren Liste v_2 ist. Wirkliche Arbeit geschieht in der zweiten Klausel, die Gebrauch von einer weiteren Relation, take-out, macht. Auf der Basis von take-out besagt die zweite Klausel, daß das erste

[20] Diese Sprechweise geht davon aus, daß die Permutationsdomäne einer Phrase eindeutig bestimmt ist. Für manche Datenbereiche könnte sich diese Annahme als unangemessen herausstellen. Wir wollen uns an dieser Stelle auf keine Option festlegen. Insofern ist die implizierte Eindeutigkeit cum grano salis zu verstehen. Für das Phonologieprinzip spielt eine Entscheidung für die eine oder andere Variante keine Rolle.

Element von v_1 ein beliebiges Element von v_2 ist und der Rest von v_1 eine Permutation derjenigen Liste x, die man aus v_2 erhält, wenn man aus v_2 dasjenige Element herausschneidet, welches das erste Element von v_1 ist. Die Rekursion terminiert mit der ersten Klausel, sobald die leere Liste x erreicht wird. Das Herausnehmen eines beliebigen Elementes aus v_2, das in einer Liste x ohne das entfernte Element resultiert, geschieht mit Hilfe von take-out.

(4.17) *Die Klauseln der Relation take-out*

$$\begin{aligned}
&\texttt{take-out}(x, y, z) :=\\
&\qquad x\text{FIRST} \approx y\\
&\qquad \wedge\ x\text{REST} \mathrel{\dot{\approx}} z\\
&\texttt{take-out}(x, y, z) :=\\
&\qquad \neg\ x\text{FIRST} \approx y\\
&\qquad \wedge\ x\text{FIRST} \approx z\text{FIRST}\\
&\qquad \wedge\ \texttt{take-out}(x\text{REST},\ y,\ z\text{REST})
\end{aligned}$$

x und z sind Listen. Die erste Klausel von take-out ist der einfache Fall. y ist das erste Element der Liste x, und z ist gleich dem Rest der Liste x. z ist somit die Liste x ohne ihr erstes Element. Mit der zweiten Klausel kommt die Rekursion ins Spiel. In diesem Fall ist das erste Element von x ungleich y. Das Element y, das von x entfernt wird, so daß wir die verkürzte Liste z erhalten, muß also weiter hinten auf x zu finden sein. Zunächst übernehmen wir das erste Element von x als erstes Element von z. Mit der Rekursion wird im Rest von x nach y gesucht und die übrigen Elemente an den Rest von z übergeben. Die Suche terminiert mit der ersten Klausel, sobald auf x das erste Auftreten von y gefunden wird.

take-out sagt somit für jede endliche Liste x, daß y ein Element von x ist, und z ist diejenige Liste, die x bis auf die Tatsache gleicht, daß das erste Auftreten von y auf ihr entfernt wurde. Der Beweis läuft durch Induktion über die Länge von x.

Mit diesem Resultat über take-out kommen wir zurück auf perm. Es läßt sich jetzt zeigen, daß unter der Voraussetzung der Endlichkeit von v_2 die Liste v_1 eine beliebige Permutation von v_2 ist. Der Beweis läuft per Induktion über die Länge von v_2.

Für die Klausel von build-phon in (4.15) bedeutet das, daß wir inzwischen sichergestellt haben, daß jede Permutation der Elemente einer Permutationsdomäne zu einer Phrase zur Verfügung steht. Um zum angestrebten Resultat zu gelangen, bleibt noch zu zeigen, wie man zu einer beliebigen endlichen Liste endlicher Zeichen eine Phonologieliste erhält, auf der die Phonologien der Zeichen in der Reihenfolge der Zeichen auf der

Ursprungsliste stehen. Diese Konvertierung von einer Liste von Zeichen in eine Liste von Phonologien leistet die Relation signphon.

(4.18) *Die Klauseln der Relation signphon*

$$\begin{aligned}
&\texttt{signphon}(v_1, v_2) :=\\
&\qquad v_2 \sim \mathit{elist}\\
&\qquad \wedge\ v_1 \sim \mathit{elist}\\
&\texttt{signphon}(v_1, v_2) :=\\
&\qquad \exists x\ [\texttt{signphon}(x, v_2\text{REST})\\
&\qquad \wedge\ \texttt{append}(v_2\text{FIRST PHON}, x, v_1)]
\end{aligned}$$

v_1 und v_2 sind Listen. Einer leeren Liste v_2 von Zeichen entspricht nach der ersten Klausel eine leere Phonologieliste v_1. Wenn v_2 hingegen nicht leer ist, dann erhält man gemäß der zweiten Klausel v_1, indem man an die Phonologie des ersten Zeichens auf v_2 die Phonologien x anhängt, die man aus den restlichen Zeichen auf v_2 mit Hilfe von signphon extrahiert hat. Die Rekursion terminiert, sobald das Ende der Liste v_2 erreicht ist und somit die leere Phonologieliste angehängt wird.

signphon setzt die Standardrelation append voraus, die wir anläßlich der Definition von construct-phon (3.27) am Ende von Abschnitt 3.1.5 angegeben haben. Für append läßt sich zeigen, daß unter Voraussetzung der Endlichkeit der Argumente die Liste im dritten Argument eine Konkatenation der Listen im ersten und zweiten Argument ist. Mit diesem Resultat erhält man durch Induktion über die Länge von v_2, daß für jede endliche Liste von Zeichen v_2 im zweiten Argument der Relation signphon die Liste v_1 im ersten Argument eine Konkatenation der Phonologien der Zeichen in v_2 in der Reihenfolge der Elemente von v_2 ist. Daraus folgt mit den bisherigen Resultaten über die in (4.15) bei der Definition von build-pon verwendeten Relationen, daß das Phonologieprinzip (4.14) tatsächlich die gewünschten Eigenschaften hat: Die Phonologie jeder endlich großen Phrase ist eine beliebige Permutation der Gesamtphonologien der Elemente ihrer Permutationsdomäne.

4.3. Die Gesamtstruktur des Linearisierungsmoduls. In Abschnitt 4.1 wurde der Begriff der Permutationsdomäne einer Phrase konkretisiert, der daraufhin in 4.2 im Phonologieprinzip dazu verwendet wurde, um den phonologischen Wert von Phrasen auf Permutationen der Phonologie bestimmter Komponenten dieser Phrasen zu beschränken. Gegenstand des gegenwärtigen Abschnitts sind die eigentlichen Wortstellungsregeln, also die Generalisierungen, welche Linguisten zur Wortstellung ausdrücken möchten.

Da die Phonologieliste von Phrasen bereits durch das Phonologieprinzip eingeschränkt wird, besteht die Aufgabe des verbleibenden Teils der Linearisierungskomponente der Grammatik darin, die Wortstellung in Phrasen auf ihre Konsistenz mit den Wortstellungsregeln zu überprüfen. Dabei ist es möglich, daß manche Wortstellungsregeln nur auf bestimmte Arten von Phrasen zutreffen. Eine Phrase ist dann keine zulässige Phrase, wenn ihre Phonologie mindestens einer Wortstellungsregel nicht entspricht, die für sie einschlägig ist. Im folgenden werden wir uns konkret nur mit Wortstellungsregeln beschäftigen, die auf alle Phrasen zutreffen. Sie sind im Linearisierungsprinzip (4.19) zusammengefaßt.

(4.19) *Das Linearisierungsprinzip*

$$\left[\textit{phrase}\right] \quad \Longrightarrow \quad \left[\begin{array}{l}\textit{phrase}\\ \textsc{phon}\ \boxed{1}\\ \texttt{lp(:,}\boxed{1}\texttt{)}\end{array}\right]$$

Jede Phrase steht in der lp-Relation zu ihrer Phonologie. (4.19) ist das allgemeinste der Prinzipien der Grammatik, das eine Verbindung zwischen Phrasen und Wortstellungsregeln herstellt. Gesetzmäßigkeiten der Wortstellung werden Regeln statt Prinzipien genannt, da sie in der vorliegenden Architektur nicht als implikative Beschreibungen wie zum Beispiel das Kopfattributprinzip oder das Subkategorisierungsprinzip formuliert sind. Stattdessen werden Aussagen zur Wortstellung ausschließlich mit Hilfe von Relationen gemacht, die selbst nicht direkt in implikativen Beschreibungen auftreten. Wie wir in (4.20) sehen werden, werden diese Wortstellungsrelationen in der definierenden Klausel von lp aufgerufen. Damit ergibt sich der Zusammenhang zwischen Wortstellungsregeln und den Prinzipien der Grammatik indirekt dadurch, daß eine in einem Linearisierungsprinzip wie (4.19) verwendete Relation ihrerseits auf der Basis von Relationen definiert ist, welche die Kodierung der Wortstellungsregeln sind.

Während die Relation lp zwischen allen Phrasen und ihrer Phonologie besteht, können bei Bedarf speziellere Relationen, zum Beispiel in Entsprechung zur Sortenhierarchie unter *phrase*, geschrieben werden. Auf diese Weise erhält man die Linearisierungsrelationen lp-hm, die Wortstellungsregeln für Kopf-Marker-Phrasen, lp-ha, die Wortstellungsregeln für Kopf-Adjunkt-Phrasen, und lp-hc, die Wortstellungsregeln für Kopf-Komplement-Phrasen enthält. Sie gelten aufgrund des Antezedens des zugehörigen Prinzips jeweils für Phrasen der jeweiligen Sorte. Das heißt, daß zum Beispiel die Relation lp-hc in einem Prinzip verwendet wird, das (4.19) bis auf die Tatsache entspricht, daß die Sorte im Antezedens *hc-phrase* und die im Konsequens aufgerufene Relation lp-hc ist. Zusätzlich

gelten für die Phrasen all dieser Species die Wortstellungsregeln, die durch lp erzwungen werden. Sollte eine noch feinere Unterscheidung empirisch notwendig sein, wäre es auch möglich, lp-Relationen mittels komplexer Antezedentien auf Phrasen mit bestimmten syntaktischen oder semantischen Eigenschaften einzuschränken. Mittels eines zu (4.19) parallelen Prinzips, das jedoch im Antezedens Nominalphrasen beschreibt, könnte eine Relation lp-noun Wortstellungsregeln ausschließlich für Phrasen zur Verfügung stellen, die nominale Projektionen sind.

Während solche Unterscheidungen in Übereinstimmung mit den Möglichkeiten einer Linearisierungsgrammatik in HPSG stehen, wird sich die linguistische Theoriebildung darum bemühen, möglichst generelle Wortstellungsregeln zu formulieren und somit nicht notwendigerweise sämtliche durch die Architektur zur Verfügung gestellten Ausdrucksmittel ausschöpfen wollen. Bei der Betrachtung der Definition und Aufgliederung der Klausel für die Relation lp wird sich zeigen, daß die für die Wortstellungsregeln verwendeten Mechanismen so definiert sind, daß ihre allgemeine Anwendung auf jeden Phrasentyp nur dann unmöglich wäre, wenn sich Wortstellungsgegebenheiten in verschiedenen strukturellen Bereichen der Grammatik direkt widersprächen. Wäre dies tatsächlich der Fall, dann würde dies jedoch darauf hindeuten, daß eine Benutzung der verfügbaren Ausdrucksmittel empirisch motiviert wäre.

(4.20) *Die Struktur des Linearisierungsmoduls*

$$\begin{aligned}&\texttt{lp}(v_1, v_2) :=\\ &\qquad \exists aux_0 \ldots \exists aux_n\ [\texttt{perm-domain}(aux_0, v_1)\\ &\qquad \wedge\ \texttt{build-aux}_1(aux_1, v_1)\\ &\qquad \wedge\ \ldots\\ &\qquad \wedge\ \texttt{build-aux}_n(aux_n, v_1)\\ &\qquad \wedge\ \texttt{wor-0}_1(v_2, aux_0)\\ &\qquad \wedge\ \ldots\\ &\qquad \wedge\ \texttt{wor-0}_m(v_2, aux_0)\\ &\qquad \wedge\ \texttt{wor-1}_1(v_2, aux_1)\\ &\qquad \wedge\ \ldots\\ &\qquad \wedge\ \texttt{wor-1}_{m'}(v_2, aux_1)\\ &\qquad \wedge\ \ldots\\ &\qquad \wedge\ \texttt{wor-n}_1(v_2, aux_n)\\ &\qquad \wedge\ \ldots\\ &\qquad \wedge\ \texttt{wor-n}_{m''}(v_2, aux_n)]\end{aligned}$$

Die Struktur der Definition der Relation lp verdeutlicht den modularen Aufbau der eigentlichen Wortstellungskomponente der Grammatik.

Die Definition von lp zerfällt im wesentlichen zwei Teile. Der erste Abschnitt, der vom Aufruf der Relation perm-domain bis zur letzten build-aux-Relation reicht, hat die Aufgabe, bestimmte Informationen über die betrachtete Phrase, die zur Formulierung der Wortstellungsregeln essentiell sind, in einem bestimmten Schnittstellenformat zur Verfügung zu stellen. Eine der Konsequenzen dieser Architektur besteht darin, daß Änderungen in der Grammatik durch entsprechende Modifikationen in den zwischen der Struktur der Zeichen und den Wortstellungsregeln vermittelnden Schnittstellenrelationen aufgefangen werden können, ohne daß die eigentlichen Wortstellungsregeln umgeschrieben werden müssen. Diese Robustheit der Grammatik hinsichtlich von Modifikationen in Teilen einer Theorie, welche die Generalisierungen in einer anderen empirischen Domäne, hier der Wortstellung, nicht betreffen, ist im Prinzip wünschenswert und entspricht der Idee interagierender, aber in wesentlichen Zügen voneinander unabhängiger Module der Grammatik. Darüber hinaus erleichtert sie in offensichtlicher Weise die Entwicklung größerer Grammatikfragmente.

Im zweiten Abschnitt der Klausel von lp stehen die Relationen, welche die eigentlichen Wortstellungsregeln repräsentieren. Das heißt, jeder dieser Relationen entspricht eine intuitive linguistische Generalisierung zur Wortstellung. Diese Wortstellungsrelationen sind in Abhängigkeit von derjenigen Information über die Phrase angeordnet, die sie zu ihrer Formalisierung benötigen. Die einfachste Art von Wortstellungsregeln, bei denen es sich um jene handelt, die in der formalen Literatur gewöhnlich betrachtet werden, verwendet lediglich die Permutationsdomäne der Phrase. Das bedeutet, daß die Wortstellungrelationen wor-0_1 bis wor-0_m alleine Zugang zu einer Liste der Zeichen haben, deren Phonologie in die Phonologie der Phrase eingeht. Damit können diese Wortstellungsregeln nur auf inhärente Eigenschaften dieser Zeichen bezug nehmen, um ihre phonologischen Beiträge in der betrachteten Phrase zu ordnen. Mit ihnen ließe sich zum Beispiel ausdrücken, daß Pronomen vor nichtpronominalen Konstituenten stehen. Ein prominentes Beispiel dieses Typs von Wortstellungsregel ist die Openerregel, die im Abschnitt 4.4 als ein konkretes Beispiel näher betrachtet wird. Aufgrund ihrer Beschränkung auf inhärente Eigenschaften der Zeichen entspricht diese Art von Wortstellungsregel weitgehend derjenigen, die im Rahmen der GPSG durch sogenannte lineare Präzedenzregeln erfaßt wird. Allerdings sind sie bereits in zweifacher Hinsicht ausdrucksstärker. Lineare Präzedenzregeln in GPSG ordnen nur Schwestern eines lokalen Baumes, wohingegen hier eine

Permutationsdomäne betrachtet wird, die potentiell (und mit den Definitionen in Abschnitt 4.1 tatsächlich) über lokale Bäume hinausreicht. Zum anderen treffen GPSGartige LP-Regeln zwingend auf alle lokalen Bäume zu, während das implikative Format von HPSG-Prinzipien es erlaubt, in der oben diskutierten Weise für Phrasen mit bestimmten Eigenschaften spezielle lp-x-Relationen einzuführen. Wortstellungsregeln des Typs wor-0 innerhalb dieser lp-x-Relationen treffen dann nur auf die in den jeweiligen Phrasen auftretenden Permutationsdomänen zu.

Mit den build-aux-Relationen wird den Wortstellungsregeln wor-1 bis wor-n Information zur Verfügung gestellt, die den Zeichen, deren Phonologie geordnet wird, nicht inhärent ist, sondern die sich aus funktionalen Zusammenhängen ergeben. Beispiele hierfür sind die Skopusverhältnisse in Sätzen oder die Reihenfolge, in der die Komplemente des finiten Verbs auf dessen COMPS-Liste stehen. Beides sind wortstellungsrelevante Gegebenheiten innerhalb von Sätzen, die nicht Eigenschaft der betroffenen Konstituenten selbst sind. Vielmehr sind sie abhängig von der semantischen Interpretation des Satzes, also von seiner logischer Form, oder von der Organisation der COMPS-Liste des die Konstituenten subkategorisierenden Verbs.

Technisch wäre es nun durchaus denkbar, die Signatur und die Prinzipien der Grammatik so zu erweitern, daß jede quantifizierende Konstituente eines Satzes Information über ihren relativen Skopus zu anderen quantifizierenden Konstituenten enthält. Im Effekt bedeutete dies, die Skopusverhältnisse in inhärente Eigenschaften der involvierten Konstituenten umzukodieren. Während dieser Weg im Prinzip gangbar erscheint, ist gegen ihn einzuwenden, daß er technisch sehr aufwendig ist und zu einer unübersichtlichen Erweiterung der Merkmalsgeometrie führt. Darüber hinaus ist er überaus contraintuitiv, da neben dem technischen Erfordernis bei der Formulierung von Wortstellungsregeln einer beschränkten Ausdruckskraft keine linguistischen Argumente dafür sprechen, die Skopusverhältnisse von Sätzen in ihren quantifizierenden Ausdrücken selbst vollständig zu repräsentieren. Das gleiche gilt mutatis mutandis für die Kodierung der relativen Reihenfolge von Konstituenten auf der COMPS-Liste des sie subkategorisierenden Verbs in den Konstituenten.

Die build-aux-Relationen dienen dazu, eine inadäquate zeicheninhärente Kodierung zum Beispiel der Reihenfolge von Komplementen auf der COMPS-Liste des Verbs zu vermeiden. Dies wird erreicht, indem eine build-aux-Relation zu jeder Phrase die einschlägige nichtinhärente Information über die Elemente ihrer Permutationsdomäne zur Verfügung stellt. build-aux_1 kann so definiert werden, daß das erste Argument aux_1 der Relation

zu jeder Projektion eines Wortes eine Kette derjenigen Komplemente des Wortes enthält, die bis zu dieser Projektion bereits in der Baumstruktur realisiert wurden, wobei die Reihenfolge ihres Auftretens auf der Kette der Reihenfolge des Auftretens ihrer Synsemwerte auf der COMPS-Liste des Kopfes der Projektion entspricht. Damit besteht für die Wortstellungsregeln des Typs wor-1 die Möglichkeit, auf die Obliqueheitshierarchie zu referieren.

Es eine bekannte Tatsache der deutschen Syntax, daß die Wortstellung vom Fokus abhängig ist, der seinerseits durch die Betonungsverhältnisse der Konstituenten eingeschränkt wird (vgl. (Höhle 1982)). In Beispiel (4.21) wird dies anhand eines finiten Verbletztsatzes mit dem Verb *geben* illustriert:

(4.21) *Das Verb* geben*:* COMPS ⟨*dat*, *akk*⟩

1. weil Karl dem Kind das *Buch* gab
2. *weil Karl das *Buch* dem Kind gab
3. *weil Karl dem *Kind* das Buch gab
4. *weil Karl das Buch dem *Kind* gab
   Fk(S) = {dem Kind, das Buch, gab}

Wir nehmen an, daß bei der Äußerung eines Satzes S sich derjenige Teil von S im Fokus befindet, der nicht durch den Kontext von S gegeben ist. In (4.21) postulieren wir, daß das die Menge bestehend aus den Konstituenten *dem Kind*, *das Buch* und *gab* ist. Standardannahmen in der HPSG zur Obliqueheitshierarchie von *geben* folgend nehmen wir außerdem an, daß auf der COMPS-Liste von *geben* der Synsemwert des Dativobjektes vor dem Synsemwert des Akkusativobjektes steht. Betrachten wir nun die beiden relativen Wortstellungen des indirekten und des direkten Objekts unter verschiedenen Konstituentenbetonungen.

Wenn das direkte Objekt dem indirekten folgt und dem Verb direkt vorangeht, und das direkte Objekt ist betont, dann ist der Satz unter dem vorausgesetzten Fokus grammatisch (1). Das heißt, der Satz kann mit der angegebenen Betonung und Wortstellung als Antwort auf die Frage 'Warum ist Karl beliebt?' gebraucht werden. Vertauscht man hingegen die Reihenfolge des indirekten Objekts und direkten Objekts, dann ist der angenommene Fokus nicht mehr möglich, das heißt, der Satz ist als Antwort auf die gleiche Frage ungrammatisch (2). Eine Betonung des indirekten Objekts *dem Kind* ist bei dem angenommenen Fokus in keiner Stellung möglich ((3) und (4)).

Während man zunächst versucht sein könnte, diese Beobachtungen auf inhärente Eigenschaften der Konstituenten, etwa deren Kasus, zurückzuführen, zeigt ein interessantes Vergleichsdatum, daß dies nicht richtig sein kann. (Primus 1989) führt eine Reihe von Argumenten, darunter Bindungs- und Topikalisierungsdaten, an, die zeigen, daß sich die Grundstellung des Dativ- und Akkusativobjekts bei Verben wie *angleichen, anvertrauen, übergeben* und *überlassen* von derjenigen bei Verben wie *schenken, zeigen, empfehlen* und *geben* unterscheidet. Während bei der zweiten Gruppe die Reihenfolge Dativargument vor Akkusativargument zutrifft, gilt für die erste Gruppe Akkusativargument vor Dativargument. In der HPSG ließe sich diese Feststellung durch die Annahme repräsentieren, daß bei Verben wie *überlassen* auf der COMPS-Liste das Synsemobjekt mit dem Kasuswert Akkusativ vor dem Synsemobjekt mit dem Kasuswert Dativ steht.

Betrachten wir nun die möglichen Wortstellungen des Dativ- und Akkusativarguments von *überlassen* unter den verschiedenen Konstituentenbetonungen bei feststehendem Fokus.

(4.22) *Das Verb* überlassen*:* COMPS ⟨*akk*, *dat*⟩

1. *weil Karl dem Kind das *Buch* überließ
2. *weil Karl das *Buch* dem Kind überließ
3. *weil Karl dem *Kind* das Buch überließ
4. weil Karl das Buch dem *Kind* überließ
   Fk(S) = {dem Kind, das Buch, überließ}

Wie in (4.21) nehmen wir als Fokus und damit als kontextuell nicht gegeben neben dem finiten Verb die beiden Konstituenten *dem Kind* und *das Buch* an. Wir betrachten die Sätze in (4.22) also in einem Kontext, wie er durch die Frage 'Warum ist Karl beliebt?' gegeben ist.

Im Unterschied zu (4.21) stellt sich heraus, daß bei *überlassen* als einziges zulässiges Muster die Konstituentenbetonung von *dem Kind* in verbadjazenter Stellung hinter dem Akkusativobjekt grammatisch ist (4). Daher ist es nicht möglich, das beobachtete Wortstellungsverhalten von Akkusativ- und Dativobjekt bei den Verben *geben* und *überlassen* auf den Kasus der beiden nominalen Argumente zurückzuführen. Dagegen bietet es sich offensichtlich an, von den unterschiedlichen Subkategorisierungsverhältnissen auf den COMPS-Listen der zwei Verben Gebrauch zu machen.

(4.23) *Empirische Generalisierung*

wenn zwei nominale Argumente eines Verbs im Fokus sind,
und A steht vor B auf der COMPS-Liste des Verbs,

dann muß B betont werden
und A steht phonologisch vor B.

Die empirische Generalisierung (4.23) wird den betrachteten Daten gerecht und ist damit ein typisches Beispiel für eine Wortstellungsregel, die sich auf Information bezieht, die nicht in den zu ordnenden Konstituenten enthalten ist, sondern an anderer Stelle in der Merkmalstruktur repräsentiert wird. Die oben beschriebene Schnittstellenrelation build-aux$_1$ macht diese Information für die Wortstellungsregel lokal verfügbar.

Aus dem Beispiel (4.23) für eine Wortstellungsregel wird im Zusammenhang mit der allgemeinen Architektur der Linearisierungskomponente eine wichtige Anforderung an den Mechanismus ersichtlich, der die Wortstellungsregeln umsetzt. Da die Wortstellungsregeln möglichst allgemein für alle Phrasen gelten sollen, darf das Fehlen von Objekten, die im Antezedens der Wortstellungsgeneralisierung erwähnt werden, nicht zum Ausschluß der betrachteten Phrase führen. Dies wäre der Fall, wenn die Formalisierung der Regel (4.23) die Präsenz zweier fokusierter Argumente des Verbs erzwänge, was ersichtlich nicht intendiert ist. Die zu formalisierenden Wortstellungsregeln sind also in der Weise technisch umzusetzen, daß sie unter der Voraussetzung des Vorliegens der beschriebenen Eingangsbedingung die Ausgangsbedingung garantieren. Im andern Fall sind sie trivial erfüllt.

Der gleiche Sachverhalt gilt bereits für die Wortstellungsregeln des einfachsten Typs, bei dem die Elemente der Permutationsdomäne selbst die Information beinhalten, aufgrund derer die Wortstellung beschränkt wird. Diese Regeln sind von einem besonders einfachen, nichtrekursiven Format, dessen wesentlicher Bestandteil eine Relation find-x-y ist, deren Aufgabe es ist, diejenigen Elemente der Permutationsdomäne in zwei Ketten bereitzustellen, welche die Eigenschaft x beziehungsweise y aufweisen. Unter Voraussetzung des Schemas (4.24) ist es lediglich noch nötig, die Relation find-x-y in trivialer Weise für die jeweils benötigten Eigenschaften zu modifizieren, um eine beliebige Wortstellungsregel, die sich auf zeicheninhärente Information bezieht, zu formalisieren. In Abschnitt 4.4 geben wir für die Openerregel ein vollständiges Beispiel für eine konkrete Instanz von find-x-y.

(4.24) *Das Wortstellungsschema der Regeln wor-0 mit find-x-y*

$$\begin{aligned}&\texttt{wor-0}_i(p,a) :=\\ &\quad \exists x \exists y\ [\texttt{find-x-y}(a,x,y)\\ &\quad \wedge\ \texttt{prec}(p,x,y)]\end{aligned}$$

Eine Wortstellungsregel vom Typ 0 hat Zugriff auf die Phonologie p und die Permutationsdomäne a von Phrasen (vgl. (4.20)). Die Relation find-x-y macht Gebrauch von der Permutationsdomäne a, um in x eine Kette bestehend aus der Phonologie von Elementen aus a mit der jeweils festzulegenden Eigenschaft x und in y eine Kette bestehend aus der Phonologie von Elementen aus a mit der Eigenschaft y zu spezifizieren. Das eigentliche Kernstück einer jeden Wortstellungsregel, unabhängig von deren Typ, ist die Relation prec, die als eine Explikation des in der Linguistik weit verbreiteten Zeichens '$\prec$' in linearen Präzedenzregeln anzusehen ist. Die Relation prec bestimmt, daß in der Phonologie p der betrachteten Phrase jedes Element aus x jedem Element aus y vorangeht. Aus den obigen Überlegungen ergibt sich, daß diese Bedingung erfüllt ist, falls die Liste x, die Liste y oder beide leer sind.

Wortstellungsregeln, die sich wie (4.23) auf zeichenexterne Information beziehen, sind oftmals von einer komplizierteren Struktur. (Pollard und Sag 1987, S.174) formuliert in erster Annäherung eine Wortstellungsregel für das Englische, die ein neues Symbol '$\ll$' zur Kennzeichnung der phonologischen Präzedenz verwendet:

COMPLEMENT $\ll$ COMPLEMENT

Diese Regel soll ausdrücken, daß in der Projektionslinie eines Funktors die Phonologie jeder Komplementtochter der Phonologie jeder anderen Komplementtochter vorangeht, die obliquer ist als erstere. Mit anderen Worten entspricht die phonologische temporale Reihenfolge der Reihenfolge der zugehörigen Synsemobjekte auf der Subkategorisierungsliste des Funktors. Das besondere an diesem Typ von Wortstellungsregel ist ihre rekursive Struktur. Nehmen wir an, daß die Relation build-aux$_1$ in ihrem ersten Argument zu jeder Phrase die Kette von Komplementen des Kopfes der Projektion beinhaltet, die unterhalb dieser Phrase syntaktisch realisiert worden sind, wobei in der Kette ihre Reihenfolge auf der COMPS-Liste beibehalten wird. Dann ergibt sich für die Regel aus (Pollard und Sag 1987) folgende Formalisierung:

(4.25) *Eine Wortstellungsregel unter Berücksichtigung der Obliqueheitshierarchie*

$$\begin{aligned}&\texttt{wor-1}_1(p,c) :=\\ &\quad \exists x \exists y\ [x \approx c\textsc{first phon}\\ &\quad \wedge\ \texttt{signphon}(y, c\textsc{rest})\\ &\quad \wedge\ \texttt{prec}(p,x,y)\\ &\quad \wedge\ \texttt{wor-1}_1(p, c\textsc{rest})]\end{aligned}$$

$$\texttt{wor-1}_1(p, c) :=$$
$$c \sim \mathit{elist}$$

Die zweite Klausel stellt die Rekursionsverankerung dar. Wenn es innerhalb der Phrase v_1 kein syntaktisch realisiertes Komplement des Kopfes gibt, dann ist die Wortstellungsregel trivial erfüllt. Andernfalls tritt die erste Klausel in Kraft. Sie bestimmt, daß die Phonologie des ersten Elements der Kette syntaktisch realisierter Komplemente vor den phonologischen Beiträgen aller weiteren steht, sofern es weitere gibt. Für die Umwandlung einer Kette von Zeichen in eine Kette bestehend aus deren Phonologie wird die oben definierte Relation signphon (4.18) verwendet.

Ein Blick auf das Schema für einfache Wortstellungsregeln (4.24) und die Regel in (4.25) läßt vermuten, daß eine einzige Relation ausreicht, um lineare Präzedenzverhältnisse auf der Phonologieliste von Phrasen zu beschreiben, und keine voneinander verschiedenen Relationen benötigt werden, die den Symbolen '$\prec$' und '$\ll$' direkt entsprechen. Der wesentliche Unterschied zwischen diesen beiden Regeltypen liegt vielmehr in der Rekursivität und der Berücksichtigung von Information, die nicht in den hinsichtlich ihrer Phonologie zu ordnenden Zeichen selbst enthalten ist, im Fall von (4.25). In dieser Hinsicht entsprechen Wortstellungsregeln, die auf Skopusverhältnisse Bezug nehmen, genau jenen, welche die Obliqueheitshierarchie verwenden. Eine Wortstellungsregel des Deutschen, die forderte, daß eine Nominalphrase mit weitem Skopus einer Nominalphrase mit engem Skopus phonologisch vorangeht, wäre unter Verwendung einer die Skopusverhältnisse reflektierenden, durch build-aux$_2$ bereitgestellten Kette parallel zu (4.25) formalisieren.

Bislang sind wir stillschweigend davon ausgegangen, daß sich prec in der informell angedeuteten Weise in RSRL ausdrücken läßt. Im verbleibenden Teil dieses Abschnittes werden wir uns eine denkbare Formalisierung ansehen und ihre Voraussetzungen und Eigenschaften diskutieren. Wie auch bei den anderen Komponenten des Linearisierungsmoduls orientieren wir uns dabei an den Vorschlägen von (Richter und Sailer 1995). In den Beweisskizzen setzen wir stets stillschweigend die Endlichkeit der betrachteten Strukturen voraus.

(4.26) *Die Klausel der Relation prec*

$$\texttt{prec}(p, y, z) :=$$
$$\exists v\ [\texttt{get-prefix}(p, y, v)$$
$$\wedge\ \texttt{exclude}(v, z)]$$

Die Formalisierung von prec geht von zwei Vorbedingungen aus. Zum einen darf es auf der Phonologieliste p keine Strukturidentitäten geben.

Zum anderen müssen sämtliche Elemente der Listen y und z tatsächlich in p vorkommen.

Die erste Bedingung ist durch die Annahme motiviert, daß die Reihenfolge der Elemente der Phonologieliste der temporalen Reihenfolge der empirisch beobachtbaren Äußerung von Lauten entspricht. Insofern wäre es kurios anzunehmen, daß zu verschiedenen Zeitpunkten geäußerte Laute miteinander identisch sein können. Indessen ist es eine offene Frage, ob es sich bei der Annahme der Strukturverschiedenheit der Elemente der Phonologieliste um eine metagrammatische Voraussetzung handelt oder um eine Beschränkung, die im Linearisierungsmodul explizit ausgedrückt werden muß (wie das bei einer Implementierung der Fall ist). Wir wollen uns an dieser Stelle auf keine der beiden Möglichkeiten festlegen und halten fest, daß eine Relation, welche die Verschiedenheit aller Element auf der PHON-Liste von Phrasen erzwingt, der Definition der Klausel der Relation lp (4.20) leicht hinzugefügt werden kann.

Die zweite Bedingung ist für die Formalisierung der Schnittstellenrelationen build-aux und des Schemas von find-x-y wichtig. Sie dürfen in jeder Phrase nur solche Hilfsstrukturen zur Verfügung stellen, die für die betreffende Phrase auch tatsächlich von Belang sind. Von Belang sind genau die Informationen, die Elemente der Permutationsdomäne der Phrase betreffen. Nur für diese Elemente sind die relative Skopusinformation und ihre relative Reihenfolge auf der COMPS-Liste des Funktors für die Wortstellung einschlägig. Da find-x-y auf der Permutationsdomäne der Phrase selbst operiert (vgl. (4.24)), ist die zweite Bedingung für die Wortstellungsregeln vom Typ 0 notwendig erfüllt. Aber auch für die Ketten, welche zusätzliche Information über das Zeichen bereitstellen, wird dieser Bedingung auf natürliche Weise entsprochen. Da die COMPS-Liste des Funktors, die zur Rekonstruktion der Obliqueheitshierarchie benötigt wird, traditionell nur Synsemobjekte enthält, in einer Wortstellungsregel unter Verwendung der Relation prec jedoch die Phonologie geordnet wird, die erst in der Realisation einer Konstituente als Zeichen präsent ist, ergibt es sich von selbst, daß auf die Phonologie der Komplemente erst zugegriffen werden kann, sobald sie in der Baumstruktur realisiert wurden und nicht schon tiefer in der Struktur. Die gleiche Überlegung gilt für die Zugänglichkeit der Phonologie von skopustragenden Elementen und ist unabhängig davon, welche Art von semantischer Repräsentation gewählt wird.

Die Relation prec verwendet zwei weitere Relationen, get-prefix und exclude. Die Idee von prec ist, daß (unter der angeführten Vorbedingung)

alle Elemente in y allen Elementen in z auf p vorangehen. Dazu stellt get-prefix das kleinste Präfix v von p bereit, das alle Elemente von y enthält. exclude verwendet das Präfix v und stellt sicher, daß kein Element von z in v enthalten ist.

(4.27) *Die Klauseln der Relation get-prefix*

$$\begin{aligned}
&\texttt{get-prefix}(p, y, z) :=\\
&\qquad y \sim \mathit{elist}\\
&\qquad \wedge\ z \sim \mathit{elist}\\
&\texttt{get-prefix}(p, y, z) :=\\
&\qquad \exists v\ [p\textsc{first} \approx z\textsc{first}\\
&\qquad \wedge\ \texttt{in-and-out}(y, p\textsc{first}, v)\\
&\qquad \wedge\ \texttt{get-prefix}(p\textsc{rest}, v, z\textsc{rest})]
\end{aligned}$$

Nach der ersten Klausel ist die leere Liste z das kleinste Präfix von p, das die leere Liste y enthält. Der rekursive Fall in der zweiten Klausel macht Gebrauch von einer der Relation take-out (4.17) ähnlichen dreistelligen Relation in-and-out. take-out diente dazu, aus einer beliebigen Liste ein Element herauszunehmen und die daraus resultierende Liste anzugeben. in-and-out ist eine flexiblere Variante dieser Relation. Wenn das Element im zweiten Argument von in-and-out in der Liste im ersten Argument enthalten ist, verhält sich in-and-out wie take-out und stellt im dritten Argument die Liste zur Verfügung, die aus der ersten durch Herausnahme des ersten Vorkommens des Elementes resultiert. Wenn das Element im zweiten Argument in der Liste im ersten Argument nicht vorkommt, enthält das dritte Argument die gleiche Liste wie das erste.

Diese Eigenschaften werden in der zweiten Klausel von get-prefix zur Bestimmung des kleinsten Präfixes z von p, das alle Elemente von y enthält, ausgenutzt. Zunächst ist das erste Element auf p das erste Element des Präfixes, da dieses noch nicht vollständig ist. Wenn das erste Element von p auch in y ist, dann wurde ein Element von y auf p gefunden, und die Liste der zu findenden Elemente y wird um dieses Element zu v verkürzt. Wenn das erste Element von p nicht in y ist, wurde kein Element gefunden, und v übernimmt die Elemente von y unverändert. In beiden Fällen geht get-prefix mit dem Rest von p in die Rekursion, um im Rest von z das kleinste Präfix des Restes von p anzugeben, welches alle Elemente von v enthält. Die Rekursion terminiert mit der ersten Klausel, wenn v leer ist, das heißt, das ganze Präfix gefunden wurde, wobei dann der Rest von z durch *elist* abgeschlossen wird. Da wir gemäß Vorbedingung wissen, daß alle Elemente von y tatsächlich auf p vorhanden sind, terminiert die Rekursion bei der Verwendung von get-prefix in der Relation prec.

(4.28) *Die Klauseln der Relation in-and-out*

$$\begin{aligned}
&\texttt{in-and-out}(x, y, z) :=\\
&\qquad x \sim \mathit{elist}\\
&\qquad \wedge\ z \sim \mathit{elist}\\
&\texttt{in-and-out}(x, y, z) :=\\
&\qquad x\textsc{first} \approx y\\
&\qquad \wedge\ x\textsc{rest} \tilde{\approx} z\\
&\texttt{in-and-out}(x, y, z) :=\\
&\qquad \neg\ x\textsc{first} \approx y\\
&\qquad \wedge\ x\textsc{first} \approx z\textsc{first}\\
&\qquad \wedge\ \texttt{in-and-out}(x\textsc{rest}, y, z\textsc{rest})
\end{aligned}$$

in-and-out hat zwei Verankerungen, je nachdem ob das Element y auf x vorhanden ist oder nicht. Wir nehmen an, daß es auf x keine Strukturidentitäten gibt. Die erste Klausel besagt, daß y ein beliebiges Element sein kann, wenn x und z jeweils leer sind. Nach der zweiten Klausel stimmen der Rest von x und z hinsichtlich der Listenelemente und ihrer Reihenfolge überein, wenn y das erste Element auf x ist (das, nach Vorbedingung, das einzige Auftreten von y auf x ist).

Die dritte Klausel ist rekursiv. Wenn y nicht das erste Element auf x ist, wird dieses als erstes Element auf z übernommen und im Rest von x weiter nach y gesucht, wobei sich der Rest von z entsprechend dem Vorhandensein oder Fehlen von y auf x ergibt.

Damit ergibt sich, daß das dritte Argument z von in-and-out eine Liste ist, die hinsichtlich der Elemente und ihrer Reihenfolge der Liste x im ersten Argument genau entspricht, wenn das zweite Argument y ein Element von x ist; ansonsten ist y jene Liste, die man erhält, wenn y aus x entfernt wird. Der Beweis läuft durch Induktion über die Länge von x.

Mit dem Resultat über in-and-out kann man nun für die Relation get-prefix zeigen, daß unter der Annahme, daß die Liste p im ersten Argument keine Strukturidentitäten aufweist und alle Elemente der Liste y im zweiten Argument auch auf p vorkommen, das dritte Argument z das kleinste Präfix von p ist, das alle Elemente von y enthält. Der Beweis ergibt sich durch Induktion über die Länge von p.

Um die gewünschten Eigenschaften von prec zu zeigen, fehlt jetzt noch die Definition der Relation exclude.

(4.29) *Die Klauseln der Relation exclude*

$$\begin{aligned}
&\texttt{exclude}(x, y) :=\\
&\qquad y \sim \mathit{elist}
\end{aligned}$$

$$\texttt{exclude}(x, y) :=$$
$$\texttt{in-and-out}(x, y_{\text{FIRST}}, x)$$
$$\wedge\ \texttt{exclude}(x, y_{\text{REST}})$$

exclude verwendet wiederum die Relation in-and-out. exclude stellt sicher, daß kein Element der Liste y in x vorkommt. Das ist trivial erfüllt, wenn y die leere Liste ist (erste Klausel). Wenn y nicht leer ist, ist das erste Element von y nicht auf x vorhanden, und im Rest von y gibt es ebenfalls kein Element, das auf x vorkommt (zweite Klausel). Die zweite Klausel verwendet die Eigenschaft von in-and-out, daß bei Identität des ersten und dritten Arguments das Element im zweiten nicht auf jener Liste vorkommt. Durch Induktion über die Länge von y läßt sich zeigen, daß x und y genau dann in der exclude-Relation stehen, wenn kein Element von y auf x vorkommt.

Mit den Resultaten über die Relationen get-prefix und exclude kommen wir zurück auf die Relation prec in (4.26). Wir nehmen an, daß alle Elemente der beiden Listen y im zweiten und z im dritten Argument von prec in der Liste p des ersten Arguments auftreten und es auf den drei Listen jeweils keine strukturidentischen Elemente gibt. Dann läßt sich mit den Eigenschaften von get-prefix und exclude zeigen, daß alle Elemente in y allen Elementen in z auf p vorangehen. Diese Eigenschaft ist trivial erfüllt, wenn y, z oder beide leer sind (für die Details des entsprechenden Beweises in einer Junkslotkodierung siehe (Richter und Sailer 1995, S.169)).

4.4. Die Openerregel. Mit der Definition der Permutationsdomänen in finiten Sätzen in Abschnitt 4.1, dem Phonologieprinzip (4.14) und der im vorigen Abschnitt präsentierten Gesamtarchitektur des Linearisierungsmoduls stehen sämtliche Komponenten des Linearisierungsmoduls der Grammatik vollständig zur Verfügung. Es ist nunmehr ein leichtes, konkrete Wortstellungsregeln in dieses Modul einzufügen. Als konkretes Beispiel wird in diesem Abschnitt die Formalisierung der Openerregel (3.8) vorgestellt. Die Openerregel ist für die Wortstellung an der linken Peripherie finiter Sätze verantwortlich und faßt die Elemente zusammen, die in der Terminologie des topologischen Feldermodells das C-Feld, das FINIT-Feld sowie das Vorfeld und zugehörige FINIT-Feld besetzen (vgl. den Überblick in Abschnitt 3.1.5). Wir wiederholen sie in (4.30).

(4.30) *Die Openerregel (informell)*

> In einer Phrase steht die Phonologie des Openers vor der Phonologie aller anderen Elemente der Permutationsdomäne.

Da Opener als Wörter definiert sind, deren Kopf das Merkmal FRONTED aufweist, bezieht sich die Openerregel ausschließlich auf inhärente Eigenschaften der phonologisch zu ordnenden Elemente. Somit handelt es sich um eine Wortstellungsregel vom einfachsten Typ, der sich alleine der Permutationsdomäne bedient. Also ist sie eine Instanz des Schemas (4.24), wobei find-x-y parametrisiert ist auf die Eigenschaft x, ein Wort mit Kopfattribut FRONTED, beziehungsweise y, nicht ein Wort mit Kopfattribut FRONTED zu sein. Wir ordnen die Openerregel als Relation wor-0_1 in das Linearisierungsmodul (4.20) ein.

(4.31) *Die Openerregel (formal)*

$$\texttt{wor-0}_1(\mathit{phon}, \mathit{pd}) :=$$
$$\exists \mathit{fro}\ \exists \mathit{nfro}\ [\texttt{find-fro-nfro}(\mathit{pd}, \mathit{fro}, \mathit{nfro})$$
$$\wedge\ \texttt{prec}(\mathit{phon}, \mathit{fro}, \mathit{nfro})]$$

Durch den Aufruf von wor-0_1 in (4.20) ist sichergestellt, daß *phon* die Phonologie der Phrase und *pd* ihre Permutationdomäne ist. find-fro-nfro ist so zu definieren, daß *fro* die Phonologie des Wortes mit Kopfattribut FRONTED und *nfro* die Phonlogie aller übrigen Elemente der Permutationsdomäne enthält. Darüber hinaus ist zu beachten, daß die Openerregel verlangt, daß in einer Permutationsdomäne höchstens ein Wort mit Kopfattribut FRONTED vorkommen darf. Die im vorangegangenen Abschnitt diskutierte Relation prec stellt dann automatisch sicher, daß die Phonologie des Openers der Phonologie aller anderen Elemente der Permutationsdomäne in der betrachteten Phrase vorangeht. Weil wir durch das Phonologieprinzip wissen, daß die Phonologie einer Phrase genau aus den phonologischen Beiträgen der Elemente der Permutationsdomäne besteht, wird somit die intendierte Bedeutung der informellen Regel (4.30) erzielt. find-fro-nfro ist in vier Klauseln definiert:

(4.32) *Die Klauseln von find-fro-nfro*

$$\texttt{find-fro-nfro}(\mathit{pd}, \mathit{fro}, \mathit{nfro}) :=$$
$$\mathit{pd} \sim \mathit{elist}$$
$$\wedge\ \mathit{fro} \sim \mathit{elist}$$
$$\wedge\ \mathit{nfro} \sim \mathit{elist}$$

$$\texttt{find-fro-nfro}(\mathit{pd}, \mathit{fro}, \mathit{nfro}) :=$$
$$\exists x\ [\mathit{pd}_{\text{FIRST SS LC CT HD FR}} \approx \mathit{pd}_{\text{FIRST SS LC CT HD FR}}$$
$$\wedge\ \mathit{pd}_{\text{FIRST}} \sim \mathit{word}$$
$$\wedge\ \mathit{pd}_{\text{FIRST PHON}} \approx \mathit{fro}$$
$$\wedge\ x \sim \mathit{elist}$$
$$\wedge\ \texttt{find-fro-nfro}(\mathit{pd}_{\text{REST}}, x, \mathit{nfro})]$$

$$\begin{aligned}&\texttt{find-fro-nfro}(pd,\ fro,\ nfro) :=\\ &\quad \exists x\ [\neg\ pd\textsc{first ss lc ct hd fr} \approx pd\textsc{first ss lc ct hd fr}\\ &\quad \wedge\ \texttt{append}(x, pd\textsc{first phon}, nfro)\\ &\quad \wedge\ \texttt{find-fro-nfro}(pd\textsc{rest}, fro, x)]\end{aligned}$$

$$\begin{aligned}&\texttt{find-fro-nfro}(pd,\ fro,\ nfro) :=\\ &\quad \exists x\ [pd\textsc{first ss lc ct hd fr} \approx pd\textsc{first ss lc ct hd fr}\\ &\quad \wedge\ pd\textsc{first} \sim phrase\\ &\quad \wedge\ \texttt{append}(x, pd\textsc{first phon}, nfro)\\ &\quad \wedge\ \texttt{find-fro-nfro}(pd\textsc{rest}, fro, x)]\end{aligned}$$

Die leere Permutationsdomäne beinhaltet eine leere Kette von FRONTED-Wörtern und eine leere Kette von sonstigen Elementen (Klausel 1). Da find-fro-nfro die Permutationsdomäne nach Elementen mit den einschlägigen Eigenschaften durchsucht, bildet die erste Klausel die Rekursionsverankerung, und die übrigen sind rekursiv.

Klausel 2 ist der interessanteste Fall. Die Pfadgleichung in der ersten Zeile stellt fest, daß das erste Element der Permutationsdomäne von der Art ist, daß das FRONTED-Attribut auf seinem Kopf definiert ist. Aufgrund der zweiten Zeile handelt es sich außerdem um ein Wort. Also liegt hier ein Opener vor. Dann wird die Phonologie des Openers mit *fro* identifiziert. Im Rekursionsschritt werden die restlichen Elemente der Permutationsdomäne betrachtet. Dabei wird die Variable x für die Phonologie weiterer Opener auf *elist* gesetzt, das heißt, es darf kein weiterer Opener mehr auftreten. Alle weiteren Elemente (sofern noch welche folgen) müssen ihren phonologischen Beitrag in der Kette *nfro* leisten.

Die beiden verbleibenden Klauseln sind von einander ähnlicher Struktur und betreffen den Fall, wenn das erste Element auf *pd* kein Opener ist. Klausel 3 fordert in der ersten Zeile vom ersten Element auf *pd*, daß das FRONTED-Attribut auf seinem Kopf nicht definiert ist. Es kann folglich kein Opener sein. Seine Phonologie wird in die Kette *nfro* übernommen, indem sie an den phonologischen Beitrag x aller Nicht-Opener im Rest der Permutationsdomäne appendiert wird. Die Information über die Phonologie *fro* von Openern wird unverändert in den Rekursionsschritt übernommen.

Zuletzt behandelt auch die vierte Klausel den Fall, daß das erste Element auf *pd* kein Opener ist. Im Unterschied zu Klausel drei ist jedoch das FRONTED-Attribut auf seinem Kopf definiert. Da es allerdings kein Wort sondern eine Phrase ist, handelt es sich nicht um einen Opener, sondern um einen eingebetteten Satz, dessen Phonologie in *nfro* stehen muß. Zu diesem Zweck wird genau wie in Klausel 3 verfahren.

Durch Induktion über die Länge der Kette *pd* zeigt man, daß *fro* die leere Kette ist, wenn es auf *pd* keinen Opener gibt, oder eine Kette bestehend aus der Phonologie des einen Elements auf *pd* ist, das ein Opener ist; *nfro* ist eine Kette, die genau aus der Phonologie der übrigen Elemente von *pd* besteht. Auf der Grundlage dieser Eigenschaften von find-fro-nfro und dem Resultat über die Relation prec am Ende von Abschnitt 4.3 kann nun der Gehalt der Relation wor-0_1 (4.31) charakterisiert werden, welche die formale Umschreibung der Openerregel (4.30) darstellt. Unter der Voraussetzung, daß *pd* im zweiten Argument von wor-0_1 eine Kette von Zeichen ist und daß es zwischen den Phonologien der Elemente der Kette *pd* keine Strukturidentitäten gibt, beweist man durch Induktion über die Länge von *pd*, daß die Phonologie des einen Openers auf *pd* (sofern es einen gibt) die Phonologie aller anderen Elemente von *pd* (sofern es welche gibt) auf der Kette *phon* im ersten Argument von wor-0_1 präzediert. Dabei besteht *phon* genau aus einer Kette der Phonologien der Elemente von *pd*. Ist *pd* die leere Kette, dann ist auch *phon* die leere Kette.

Anhand der Einbettung der Openerregel in die Gesamtarchitektur des Linearisierungsmoduls der Grammatik lassen sich die Interaktion und der Beitrag der verschiedenen Komponenten beobachten. Außerdem wird deutlich, unter welchen Voraussetzungen die Wortstellungsregeln das leisten, was der Linguist mit ihnen ausdrücken möchte.

Mit der Entscheidung für eine relationale Architektur in RSRL nehmen wir den Standpunkt ein, daß die für Wortstellung interessanten Phrasen beliebig groß aber endlich (im Sinne von Definition (12) in Abschnitt 2.1) sind. In den folgenden Aussagen meinen wir diese linguistisch interessanten Phrasen, wenn wir einfach von den Phrasen in der Grammatik sprechen. Für jede solche Phrase gibt die zweistellige Relation perm-domain eine endliche Kette an, welche im sprachwissenschaftlich relevanten Sinne ihre Permutationsdomäne darstellt. Auf dieser Basis stellt das Phonologieprinzip (4.14) für jede Phrase sicher, daß ihre Phonologie exakt eine Permutation der Phonologien der Elemente ihrer Permutationsdomäne ist, wobei die Phonologien der Elemente der Permutationsdomäne nicht miteinander verschränkt werden dürfen. Die Wortstellungsregeln selbst kommen schließlich mit dem Linearisierungsprinzip (4.19) und der Struktur der Klausel der dafür zentralen Relation lp (4.20) ins Spiel. Das Linearisierungsprinzip sagt, daß für die Phonologie aller Phrasen die definierten Wortstellungsregeln gelten. Es setzt voraus, daß die PHON-Werte der Elemente jeder Permutationsdomäne untereinander und in sich keine Strukturindentitäten aufweisen.

Im konkreten Fall haben wir in Abschnitt 4.1 die Permutationsdomänen für die Projektionen finiter Verben vollständig definiert. Im gegenwärtigen Abschnitt wurde die Openerregel als Wortstellungsrelation wor-0_1 formalisiert. Daher können wir unter den angegebenen Voraussetzungen und auf der Basis der angegebenen Prinzipien und Relationen zeigen, daß in jeder finiten Verbprojektion der Opener allen anderen Elementen der Permutationsdomäne phonologisch vorausgeht.

5. Die Theorie der langen Abhängigkeiten

In Kapitel 3 wurden die für finite Sätze wichtigsten Lexikoneinträge diskutiert und die Prinzipien vorgestellt, welche die Struktur finiter Sätze determinieren; in Kapitel 4 das Konzept der linearisierungsbasierten Wortstellungsregeln ausgeführt, welche die satzgebundene Reihenfolge von Wörtern und Konstituenten regeln. Im vorliegenden Kapitel wenden wir uns nunmehr den sogenannten nichtlokalen Abhängigkeiten oder langer Bewegung zu. Damit ist Bewegung gemeint, die potentiell über die Grenzen finiter Sätze hinausreicht. Sie hebt sich in unserer Theorie bereits dadurch von strikt lokaler Bewegung, dem Scrambling und der Verbbewegung, ab, daß sie ein Phänomen darstellt, das mehr als eine Position in Merkmalstrukturen betrifft. Anstatt eine bloße Frage der Reihenfolge von Wörtern auf der PHON-Liste von Sätzen zu sein, ist lange Bewegung durch eine eigene strukturelle Position des bewegten Zeichens charakterisiert, von der aus es in einer Beziehung zu einer anderen Position steht. Unter dem Begriff der Bewegung fassen wir satzgebundene und lange Topikalisierung und w-Bewegung zusammen:

(5.1) a. Karl hat der Katze den Garaus gemacht.
b. Welcher moderne Hund liegt heute noch in einer Hundehütte?
c. Wem glaubst Du, daß der Kanzler auf die Schulter geklopft hat?

Bei Sätzen der Bauart (5.1a) spricht man traditionell davon, daß die im Vorfeld befindliche Konstituente topikalisiert ist. Handelt es sich bei der Konstituente im Vorfeld um eine w-Phrase (5.1b), so spricht man von w-Bewegung. Diese Unterscheidung ist insbesondere durch das unterschiedliche Verhalten von w-Bewegung und Topikalisierung bei langen Abhängigkeiten motiviert, auf das wir in Abschnitt 5.4 ausführlicher eingehen werden. (5.1c) ist ein Beispiel für lange w-Bewegung aus einem eingebetteten Satz. Anhand der Beispielsätze sieht man außerdem, daß kurze oder lange Bewegung in beziehungsweise aus finiten Sätzen in die linksperiphere Position des die bewegte Konstituente aufnehmenden Satzes erfolgt. Die naheliegende Vermutung, daß dies immer so ist, wird durch folgende Beispiele untermauert.

(5.2) a. Wem sie wohl meint, daß er das Buch gegeben hat?
b. *Sie hat dem Außenminister geglaubt, daß er auf die Schulter geklopft hat.
c. *Sie hat welchen gemeinsamen Bekannten gesagt, daß Maria haßt.

(5.2a) zeigt, daß der FRONTED-Wert des w-Markers eine ebensogute Landeposition darstellt wie die Vorfeldposition der Sätze in (5.1). Die

Sätze (5.2b) und (5.2c) hingegen illustrieren, daß es im Deutschen kein Long Distance Scrambling aus finiten Sätzen gibt. Mit anderen Worten ist es nicht möglich, eine lange Abhängigkeit zwischen phonologisch im Mittelfeld des Matrixsatzes realisierten Konstituenten und Argumentstellen eines eingebetteten Satzes zu konstruieren. In unserer Analyse, welche die linke Peripherie finiter Sätze in einer einheitlichen Theorie zusammenfaßt, bedeutet diese Datenlage, daß durchweg nur der FRONTED-Wert des Openers finiter Sätze für Bewegung zur Verfügung steht.

Die Theorie der langen Abhängigkeiten wird neue Domänenbegriffe einführen, die potentiell lange Bewegung von den auf der Basis von Permutationsdomänen lokal operierenden Wortstellungsregeln unterscheiden, welche Scrambling, Extraposition und das V2-Phänomen des Deutschen betreffen. In der Theorie der langen Bewegung selbst wird Topikalisierung von der w-Bewegung unterschieden, und beiden werden verschiedene Domänen zugewiesen. Orthogonal zu dieser Unterscheidung verläuft die Behandlung von bewegten Adjunkten und bewegten Argumenten, mit der die bekannten Extraktionsasymmetrien beschrieben werden.

Nichtlokale Abhängigkeiten wurden in der HPSG in der Nachfolge von (Pollard und Sag 1994) durch ein nichtlokales Merkmal SLASH beschrieben, dessen Wert in syntaktischen Baumstrukturen geleitet vom Nichtlokalen Merkmalsprinzip perkoliert. In der Nachfolge von Ideen aus der GPSG (Gazdar, Klein, Pullum, und Sag 1985) betrachtet dieses Prinzip die Verteilung von SLASH-Werten strikt in lokalen Bäumen. Ausgearbeitete Analysen der langen Bewegung speziell für das Deutsche liegen in der HPSG nicht vor. (Pollard und Sag 1994) ignoriert weitgehend die Diskussion von Barrieren für nichtlokale Bewegung, die in der GB eine prominente Rolle spielten. Auf der anderen Seite wurden in der GB-Theorie eine Vielzahl von Barrierentheorien für lange Abhängigkeiten diskutiert, und es liegen ausführliche Studien zur Datenlage im Deutschen vor. Typisch für GB-Ansätze ist die Vorstellung von globalen Beziehungen zwischen den bewegten Elementen und einer oder mehrerer Spuren, die von ihnen gebunden werden. Barrieren für Bewegung sind hier das wesentliche konzeptuelle Mittel, um lange Bewegung zu beschränken. Diese Ansätze wurden jedoch im Gegensatz zur HPSG nicht formalisiert sondern in der für die GB typischen Weise intuitiv charakterisiert.

Ziel der Analyse dieses Kapitels ist es, in einer ersten Annäherung konzeptuelle Ideen aus dem Bereich der GB nach HPSG zu übertragen. Die Verwendung von Relationen in der HPSG eröffnet die Möglichkeit, globale Relationen der in GB thematisierten Art zu formalisieren. Zugleich unterscheidet sich die Grammatikarchitektur der HPSG durch ihre

Annahme, daß linguistische Objekte getypte Merkmalstrukturen sind, so weit von den der GB zugrundeliegenden Baumstrukturen, daß eine direkte Übernahme der Ideen dieses Frameworks selbst dann nicht angemessen erscheint, wenn sich herausstellen sollte, daß sie im Prinzip nachgebildet werden können. Dieses Argument trifft umso mehr zu, als daß die unserem Ansatz zugrundeliegende Idee der Linearisierungsgrammatik stark von in der GB üblichen Annahmen über den Zusammenhang zwischen Baumstruktur und Wortreihenfolge abweicht.

In Abschnitt 5.1 beginnen wir mit der Behandlung von Instanzen von Topikalisierung und w-Bewegung, die nicht über den Bereich finiter Sätze hinausgehen. Adjunkte an der linken Peripherie finiter Sätze werden von Komplementen unterschieden und letztere durch eine Relation move-α mit einer Spur in Beziehung gesetzt. In einfachen Sätzen sind w-Bewegung und Topikalisierung keinen voneinander verschiedenenen Bedingungen unterworfen.

Abschnitt 5.2 behandelt die Frage nach dem Zusammenhang zwischen Syntax und Illokutionsprinzipien im Rahmen der HPSG. Es wird sich herausstellen, daß eine Unterscheidung zwischen eingebetteten und uneingebetteten sprachlichen Zeichen notwendig ist, um die Behandlung von wesentlichen sprachlichen Fakten ihres Status als Metaannahmen zu entheben und sie zum Gegenstand der Theorie selbst zu machen. Die Sortenhierarchie von Zeichen wird im Hinblick auf die Anbindung einer Illokutionstheorie erweitert und die Verwendung des Illokutionsmoduls anhand von einfachen Beispielen angedeutet.

In 5.3 schließen wir uns der Argumentation von (Reis 1995) an, indem wir annehmen, daß es im Deutschen keine Extraktion von Komplementen aus V2-Sätzen gibt. Die Daten, die in der Tradition von GB als Extraktion aus V2 angeführt wurden, behandeln wir als V1-Parenthesen und zeigen, wie diese Analyse in HPSG formalisiert werden kann.

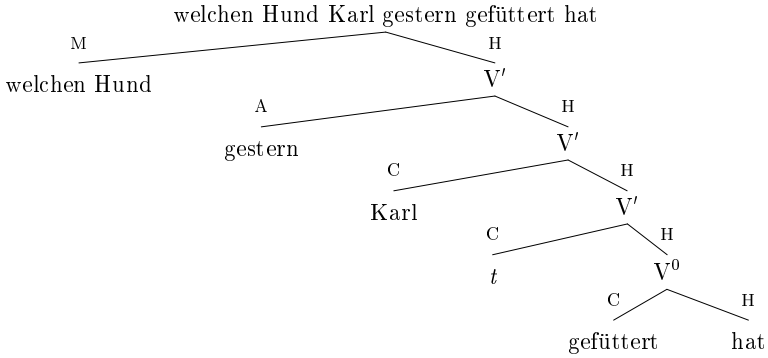
Abschnitt 5.4 enthält die eigentliche Theorie der langen Abhängigkeiten, die sich nach dem vorangegangenen nurmehr mit Extraktion aus finiten VL-Sätzen beschäftigen muß. Anschließend an die Resultate in Abschnitt 5.1 besteht auch bei langen Abhängigkeiten eine Relation move-α zwischen dem bewegten Element und einer Spur in dem Satz, aus dem bewegt wurde. Während wir keine Zwischenspuren postulieren, definieren wir Extraktionsdomänen für lange Topikalisierung und lange w-Bewegung, die hinsichtlich der Elemente, welche bewegt werden, sowie hinsichtlich der Sätze, aus denen bewegt wird, sensitiv sind. In dieses Bild wird die Unterscheidung zwischen Brückenprädikaten und Nichtbrückenprädikaten eingefügt.

5.1. Topikalisierung und w-Bewegung in einfachen Sätzen. Zentrales Ergebnis der Neukonzeption der Behandlung der deutschen Satzstruktur im Rahmen einer linearisierungsbasierten HPSG ist die in Abschnitt 3.1.5 zusammengefaßte Vereinheitlichung der linken Satzperipherie. Als Gegenstand der Wortstellungstheorie finiter Sätze handelt es sich beim phonologisch satzeinleitenden Element stets um den im Satz eindeutig bestimmten Opener, das heißt das Wort in der Permutationsdomäne des Satzes, auf dessen Kopf das Merkmal FRONTED definiert ist.

Während für die Satztypologie die Unterscheidung zwischen Sätzen mit einem Opener mit FRONTED-Wert *empty* und mit FRONTED-Wert *moved* relevant ist, generalisiert die Theorie der Bewegung über diejenigen Sätze, deren Opener in FRONTED ein Zeichen enthält, das seinen phonologischen Beitrag zum Satz über den Opener leistet. Dabei ergeben sich lexikalisch zwei Fälle, die zwei verschiedenen Satztypen mit verschiedenen Baumstrukturen entsprechen. Wörter mit FRONTED-Konstituenten sind im Standarddeutschen V2-Verben, wo es sich um die Vorfeldposition handelt, und der w-Marker, wo es sich bei der FRONTED-Konstituente um die bewegte w-Phrase handelt. Die Tatsache, daß in finiten w-VL-Sätzen im Gegensatz zu V2-Sätzen obligatorisch eine w-Phrase die FRONTED-Konstituente bildet, ergibt sich unmittelbar als lexikalische Eigenschaft des w-Markers (siehe (3.20)) und ist somit nicht Gegenstand der Bewegungstheorie selbst. Die Unterscheidung zwischen Vorfeld (in V2) und linksperipherer w-Konstituente (in VL) reduziert sich auf die lexikalische Unterscheidung zwischen w-Marker und V2-Verben und die Unzulässigkeit eines gleichzeitigen Auftretens von Wörtern dieser beiden Klassen in finiten Sätzen, welche aus der unabhängigen Forderung nach einem eindeutigem Opener hervorgeht.

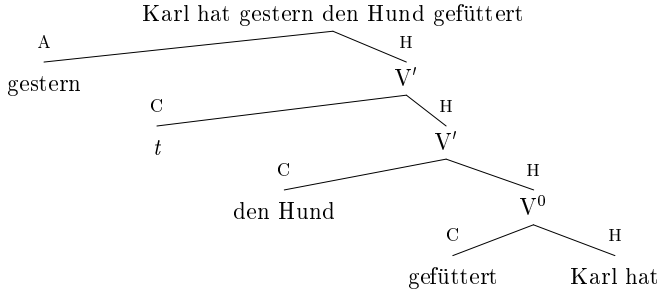
Interessant an dieser Analyse ist nun insbesondere, daß sich V2- und VL-Sätze strukturell in einer für die Bewegungstheorie relevanten Weise voneinander unterscheiden. Dabei erscheinen w-VL-Sätze strukturell einer traditionellen Bewegungsanalyse noch am ehesten zugänglich:

(5.3) *Ein einfacher w-VL-Satz*



In (5.3) hinterläßt die FRONTED-Konstituente *welchen Hund* in der Baumstruktur eine Spur, von der HPSG-theoretisch zu fordern ist, daß sie ihren LOCAL-Wert mit der FRONTED-Konstituente teilt. Dabei liegt in intuitiver Weise etwas Ähnliches wie das traditionelle c-commando-Verhältnis zwischen den beiden Positionen vor. Ein Blick auf einen V2-Satz hingegen zeigt sofort, daß dies hier nicht das einschlägige Kriterium sein kann. In (5.4) wiederholen wir die Beschreibung aus (3.29):

(5.4) *Ein einfacher V2-Satz*

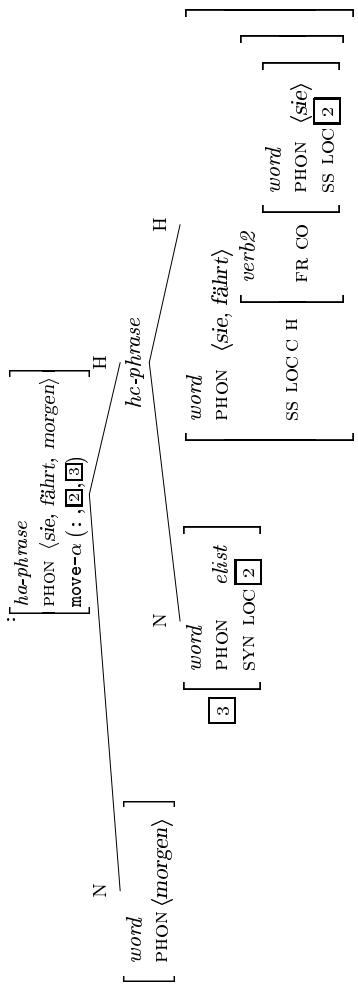


Baumstrukturell gesprochen besteht in (5.4) eindeutig nicht (eine Variante) des c-command-Verhältnisses zwischen der FRONTED-Konstituente *Karl* und der Spur, die sie bindet und mit der sie ihren LOCAL-Wert teilt. Dies hängt offensichtlich damit zusammen, daß der Phrasenstruktur in einer Linearisierungsgrammatik nicht die gleiche Funktion und der gleiche Stellenwert zukommt wie in einer Phrasenstrukturgrammatik beziehungsweise einer Grammatik, in der Bäume die bevorzugte Datenstruktur

darstellen. Ebenso evident ist es, daß die Theorie der Wortstellung und im gegebenen Fall insbesondere die Openerregel die Funktion einer (funktionalen) Phrasenstrukturprojektion übernommen hat, in der das finite Verb und die Vorfeldkonstituente jeweils bewegt würden, wobei zwischen Vorfeldkonstituente und ihrer Spur eine c-commando-Relation entstünde. Folglich muß dieser konzeptuellen Veränderung bei der Formulierung einer angemessenen Relation zwischen Antezedens und Spur in adäquater Weise Rechnung getragen werden.

Im einleitenden Abschnitt dieses Kapitels wurde bereits festgestellt, daß wir in der Bewegungstheorie Ansätze aus der GB in die HPSG integrieren. Statt eines auf lokalen Bäumen operierenden Merkmalsperkolationsmechanismus, der die Extraktion eines Elementes als Eigenschaft eines jeden Knotens zwischen dem Fuß der Extraktion und dem Filler behandelt, stellt die Relation move-α eine unmittelbare Verbindung zwischen Antezedens und Spur her. Die angestrebte Konfiguration stellt sich somit wie in (5.5) dar.

(5.5)



In einem finiten Satz mit einem Opener, der eine FRONTED-Konstituente enthält, steht der LOCAL-Wert der FRONTED-Konstituente in der move-α-Relation zur gebundenen Spur. In dem in (5.5) beschriebenen

Satz besteht somit eine dreistellige Relation move-α zwischen dem Satz, dem LOCAL-Wert der topikalisierten Konstituente und der Spur der topikalisierten Konstituente. Diese Relation ist dafür verantwortlich, daß der LOCAL-Wert von *sie* mit dem LOCAL-Wert seiner Spur identisch ist. Um das illustrierte Prinzip formal ausdrücken zu können, wird insbesondere eine formale Charakterisierung des Begriffs "Satz" benötigt. Bei seiner Bestimmung wiederum spielt der Begriff des Openers eine wichtige Rolle, der vorzüglich für die Theorie der Wortstellung relevant ist und nicht für die Theorie der Phrasenstruktur. Aufgrund dieser Verschiebungen im Aufbau der Analyse wird sich schließlich ergeben, daß bei der Bestimmung der Relation move-α zwischen Antezedens und Spur die c-command-Relation nicht vorkommt.

Die folgende Diskussion des Satzbegriffes konzentriert sich auf die im vorliegenden Fragment des Deutschen schwerpunktmäßig betrachteten finiten V1- und subkategorisierten und selbständigen V2- und VL-Sätze. Wenn also im folgenden von Sätzen die Rede ist, sind diese gemeint, sofern nicht ausdrücklich andere eingeschlossen werden. Insbesondere werden keine von adverbialen Konjunktionen eingeleiteten adverbialen Nebensätze betrachtet, für die ein gegenüber dem hier verwendeten erweiterter Satzbegriff notwendig wäre.

Bei Sätzen handelt es sich um Verbprojektionen. Außerdem wollen wir nur dann von einem Satz sprechen, wenn es sich um die Projektion eines finiten Verbs handelt, dessen sämtliche Argumente realisiert sind. Auch dies ist jedoch nur eine notwendige Bedingung: Nur diejenigen finiten Verbprojektionen sind Sätze, die alle Adjunkte und gegebenenfalls Marker dominieren, die in der fraglichen Verbprojektion als Nichtkopftöchter auftreten. Das heißt, daß im Satz (5.3) die Verbprojektion, welche die Nichtkopftochter *Karl* unmittelbar dominiert, nicht als Satz in Frage kommt, da sie weder *gestern* nocht *welchen Hund* dominiert. An dieser Stelle machen wir Gebrauch von dem Merkmal STATUS, dessen Wert die syntaktische Vollständigkeit eines Zeichens angibt. Das Vollständigkeitsprinzip (3.39) bestimmt, daß Saturiertheit eine notwendige Bedingung für Vollständigkeit ist. Die ID-Schemata für Kopf-Adjunkt-Phrasen und Kopf-Marker-Phrasen legen darüber hinaus fest, daß die jeweiligen Kopftöchter syntaktisch unvollständig sind. Für Sätze muß indessen die stärkere Bedingung gelten, daß sie erst dann vollständig sind, wenn die Verbprojektion einen Opener enthält. Bei VL-Sätzen wird dieser erst durch die oberste Phrase, die *hm-phrase*, eingeführt. Eine finite Verbprojektion eines VL-Verbs, welcher der Opener noch fehlt, kann nicht vollständig und damit kein Satz

sein. Im Satzprinzip (5.6) wird von einer vollständigen finiten Verbprojektion die Gegenwart eines Openers gefordert. (5.6) erlaubt es damit im folgenden, vollständige Projektionen finiter Verben als Sätze anzusprechen.

(5.6) *Das Satzprinzip*

Jedes vollständige finite verbale Zeichen enthält einen Opener.

$$\left[\begin{array}{l}\text{SS} \left[\begin{array}{l}\text{STATUS } \textit{complete}\\ \text{LOC CAT HEAD VFORM } \textit{finite}\end{array}\right]\end{array}\right]_{:} \Longrightarrow \exists x\, \texttt{contains-fr-word}(:, x)$$

Das Satzprinzip verwendet die zweistellige Relation contains-fr-word um sicherzustellen, daß alle (endlichen) vollständigen Projektionen finiter Verben einen Opener enthalten. contains-fr-word ist eine Relation zwischen der beschriebenen Phrase und dem Wort seiner Permutationsdomäne, auf dessen Kopf das Merkmal FRONTED definiert ist. Alternativ ist es auch möglich, daß es sich bei dem beschriebenen Zeichen selbst um den betreffenden Opener handelt. Diese Möglichkeit wird ins Auge gefaßt, um eine Analyse ohne leere Elemente der folgenden Daten zuzulassen:

(5.7) a. Komm!
b. Geh!

Sofern man davon ausgeht, daß die Sätze in (5.7) kein leeres Subjekt involvieren, ist es naheliegend, sie als satzwertige finite Verben (mit FRONTED-Wert *empty*) zu analysieren. Die folgende Definition von contains-fr-word erlaubt eine solche Analyse:

(5.8) *Die Klausel der Relation contains-fr-word*

$$\begin{aligned}&\texttt{contains-fr-word}(p, x) :=\\ &\quad x \sim \textit{word}\\ &\quad \wedge\ x\text{SS LOC CAT HEAD FRONTED} \sim \textit{locality}\\ &\quad \wedge\ (x \approx p\\ &\quad \vee\ \exists y\ [\texttt{perm-domain}(y, p) \wedge \texttt{member}(x, y)])\end{aligned}$$

Zwischen zwei Objekten p und x gilt die Relation contains-fr-word genau dann, wenn x ein Wort ist, auf dessen Kopf FRONTED definiert ist und x mit p identisch ist; oder x ist ein Wort, auf dessen Kopf FRONTED definiert ist, und die für p definierte Permutationsdomäne enthält x. Da Permutationsdomänen genau für die Phrasen der Grammatik definiert sind, wissen wir, daß es sich im zweiten Fall bei p um eine Phrase handelt. Die Enthaltenseinsbedingung wird in (5.8) mit Hilfe der Standardrelation member ausgedrückt:

(5.9) *Die Klauseln der Relation member*

$$\begin{aligned}&\texttt{member}(x, y) :=\\ &\qquad x \approx y_{\text{FIRST}}\\ &\texttt{member}(x, y) :=\\ &\qquad \neg\ x \approx y_{\text{FIRST}}\\ &\qquad \wedge\ \texttt{member}(x, y_{\text{REST}})\end{aligned}$$

Aufgrund der ersten Klausel steht x in der member-Relation zur Liste y, wenn x das erste Element von y ist. Wenn x nicht das erste Element von y ist (Klausel 2), dann muß es in der member-Relation zum Rest der Liste y stehen.

Mit der formalen Explikation der Relation contains-fr-word kann der Gehalt des Satzprinzips folgendermaßen beschrieben werden: Eine vollständige finite Verbprojektion ist entweder ein Satz bestehend aus einem finiten Verb wie in (5.7) illustriert; oder aber eine Phrase, deren Permutationsdomäne einen Opener enthält. Die Definition der Satzstruktur und das Linearisierungsmodul stellen sicher, daß zu jeder endlichen Phrase eine endliche Permutationsdomäne definiert ist und somit das Satzprinzip in der gegebenen Form in der Grammatik sinnvoll ist.

Mit der formalen Bestimmung des Satzbegriffes als syntaktisch vollständiger Projektion eines finiten Verbs sind die Voraussetzungen für eine relationale Theorie der langen Abhängigkeiten in der HPSG geschaffen. Da Opener in der vorliegenden Theorie die einzigen potentiellen Landepositionen für syntaktische Bewegung bereitstellen, finite Sätze nach dem Satzprinzip immer einen Opener haben und in einem syntaktisch vollständigen Zeichen jeder Opener, der eine FRONTED-Konstituente enthält, in der Permutationsdomäne eines Satzes vorkommt, sind Sätze diejenigen Phrasen, an denen die in (5.5) illustrierte Relation move-α gegebenenfalls ein Antezedens mit seiner Spur in Beziehung setzen muß.

Indessen analysieren wir nicht jede Konstituente in der linken Satzperipherie als syntaktisch bewegt und in der move-α-Relation zu einer Spur stehend. Motiviert wird diese Herangehensweise durch insbesondere im Englischen aber auch in anderen Sprachen oft beobachtete Asymmetrien zwischen dem Extraktionsverhalten von Adjunkten und Komplementen bei langer Bewegung (vgl. (Lutz 1993)), die sich auch im Deutschen finden. In der transformationellen Literatur wird allgemein angenommen, daß im Deutschen V2-Sätze mit overt besetztem Vorfeld strikte Inseln für Bewegung sind. Aus ihnen kann weder lang topikalisiert noch lang w-bewegt werden ((5.10a) und (5.10b)). Während diese Annahme bei den meisten Brückenverben zutrifft, läßt jedoch die (zu den Brückenprädikaten zählende) Klasse der sogenannten Präferenzprädikate Modifikation durch Adjunkte in V2-Sätze hinein zu ((5.10c), (Reis 1995, S.52–53)).

(5.10) a. *Dem Hans ist es besser, du gibst das Buch.
b. *Wem ist es besser, du gibst das Buch?
c. Dorthin ist es besser, du gehst zu Fuß.

Diesen Daten kann nur Rechnung getragen werden, wenn eine Unterscheidung hinsichtlich von Adjunkten gegenüber Komplementen in der linken Satzperipherie getroffen wird. In unserer Analyse unterliegen nur Komplemente in der linken Satzperipherie der syntaktischen Bewegungstheorie. Das heißt, Komplemente stehen in der move-α-Relation zu einer Spur. Zu Adjunkten in der linken Satzperipherie gibt es hingegen keine Spur, sondern sie stehen in einer direkten Modifikationsbeziehung zu ihrem Modifikanden, die eigenen Gesetzmäßigkeiten unterliegt, auf die wir hier nicht eingehen werden. Die Modifikationseigenschaften von Adjunkten als FRONTED-Konstituenten werden in (Sailer 1997) ausführlich beschrieben und im Rahmen der vorliegenden Architektur behandelt. Im folgenden behandeln wir ausschließlich FRONTED-Konstituenten, welche Komplemente sind.

Wie bereits oben in Beispiel (5.5) illustriert wurde, ist move-α eine dreistellige Relation zwischen einem Satz, dem LOCAL-Wert der bewegten Konstituente und der Spur der bewegten Konstituente. Eine wesentliche Aufgabe von move-α ist es, die Identität des LOCAL-Wertes von Antezedens und Spur sicherzustellen. Dies ist unmittelbar an der Definition seiner Klausel ersichtlich:

(5.11) *Die Klausel der Relation move-α*

$$\begin{aligned}&\texttt{move-}\alpha(p, a, t) :=\\ &\qquad t_{\text{PHON}} \sim \mathit{elist}\\ &\qquad \wedge\ a \approx t_{\text{SYNSEM LOC}}\\ &\qquad \wedge\ \texttt{movedom}(p, t)\end{aligned}$$

Aufgrund des Aufrufs von move-α im Frontedbesetzungsprinzip (5.13) ist gesichert, daß es sich bei der dortigen Verwendung von move-α beim ersten Argument p um einen Satz handelt. Die PHON-Liste von t ist leer, womit t als eine Spur ausgezeichnet ist, und der LOCAL-Wert von t ist identisch mit dem zweiten Argument a. Es ist wiederum eine Aufgabe des Frontedbesetzungsprinzips, die tatsächliche Identität von a mit dem LOCAL-Wert des Antezedens herzustellen. Die Relation movedom zwischen dem Satz p und der Spur t schließlich bestimmt, an welcher Stelle

innerhalb der Struktur von p sich die Spur befindet. Die wesentliche Einschränkung, die movedom der Position der Spur auferlegt, ist die, daß sie fordert, daß es sich um eine Nichtkopftochter handeln muß. Mit anderen Worten heißt das, daß Köpfe nicht syntaktisch bewegt werden können. Man bemerke, daß eine Voraussetzung dieser Beschränkung ist, daß wir Verbbewegung in V1- beziehungsweise V2-Sätzen, die in GB gewöhnlich eine Instanz von Kopfbewegung ist, in der Wortstellungstheorie und nicht in der Theorie der Bewegung behandeln.

(5.12) *Die Klausel der Relation movedom*

$$\begin{aligned}&\texttt{movedom}(p, t) :=\\ &\quad t \approx p_{\text{NONHEAD-DTR}}\\ &\quad \vee\ \texttt{movedom}(p_{\text{NONHEAD-DTR}}, t)\\ &\quad \vee\ \texttt{movedom}(p_{\text{HEAD-DTR}}, t)\end{aligned}$$

Die Definition von movedom bestimmt, daß t entweder die Nichtkopftochter von p ist oder in der movedom-Relation mit der Nichtkopftochter oder der Kopftochter von p steht. Mit dieser Definition ist movedom verwandt mit der üblichen Dominanzrelation.

Mit den Eigenschaften von movedom lassen sich die Eigenschaften der Relation move-α nun folgendermaßen bestimmen: move-α besteht zwischen einer Phrase p, dem LOCAL-Wert a einer Spur t und dieser Spur t, die eine Nichtkopftochter der von p dominierten Baumstruktur ist. Diese Eigenschaften von move-α können in einfacher Weise benutzt werden, um zum einen zu Komplementen in der linken Satzperipherie die Existenz von Spuren zu erzwingen und zum anderen deren LOCAL-Wert zu identifizieren. Dies geschieht im Frontedbesetzungsprinzip.

(5.13) *Das Frontedbesetzungsprinzip (FBP)*

Komplemente in der linken Satzperipherie stehen in der move-α-Relation zu ihrer Spur.

$$\begin{aligned}&\forall x\\ &:\left[\text{SS}\left[\begin{array}{l}\text{STATUS}\ \ \textit{complete}\\ \text{LOC CAT HEAD VFORM}\ \ \textit{finite}\end{array}\right]\right] \wedge\\ &^{x}\left[\text{SS LOC CAT HEAD FR CONST SS LOC } \boxed{1}\left[\text{CAT HEAD MOD}\ \ \textit{minus}\right]\right] \wedge\\ &\texttt{contains-fr-word(:, } x\texttt{)} \Longrightarrow\\ &\\ &\exists z\\ &:\left[\begin{array}{l}\textit{phrase}\\ \texttt{move-}\alpha\texttt{(:, }\boxed{1}\texttt{, } z\texttt{)}\end{array}\right]\end{aligned}$$

In verbaler Umschreibung sagt das FBP: Für jeden Satz S mit einem Opener O, dessen FRONTED-Konstituente kein Adjunkt ist, gilt, daß er an einer beliebigen Stelle seiner Baumstruktur als Nichtkopftochter eine Spur enthält, deren LOCAL-Wert mit dem LOCAL-Wert der FRONTED-Konstituente des Openers O von S identisch ist.

Aufgrund des Satzprinzips wissen wir, daß alle Objekte, welche die Beschreibung ein Zeichen mit STATUS-Wert *complete* und VFORM-Wert *finite* zu sein erfüllen, Sätze sind. contains-fr-word hat im zweiten Argument für jeden Satz seinen Opener. Das FBP spricht über alle Sätze, deren Opener eine FRONTED-Konstituente mit MOD-Wert *minus* enthält. Es muß sich also um eine nichtmodifizierende Konstituente und somit um ein Nicht-Adjunkt handeln.[21] Für die damit charakterisierten Sätze gilt, daß es eine Spur z gibt, so daß die move-α-Relation zwischen dem jeweiligen Satz, dem LOCAL-Wert der FRONTED-Konstituente seines Openers und z gilt.

(5.14) illustriert den Effekt des FBP anhand eines w-V2- und eines w-VL-Satzes.

[21] Im hier in Betracht gezogenen Ausschnitt des Deutschen folgt damit automatisch, daß es sich um ein Komplement handelt. Eine umfassendere Analyse müßte an dieser Stelle gegebenenfalls sorgfältiger differenzieren.

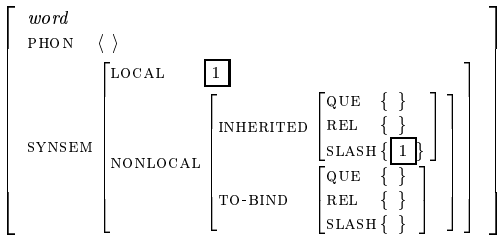
(5.14) *Satzgebundene w-Bewegung in V2 und VL*

Ein Vergleich der Beschreibung des w-V2- und des w-VL-Satzes zeigt, daß die Bezugnahme des FBP auf den Opener des Satzes die herkömmliche c-command Beziehung zwischen Antezedens und Spur irrelevant werden läßt.

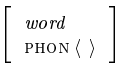
Das Frontedbesetzungsprinzip ersetzt die gesamte Theorie der langen Abhängigkeiten von (Pollard und Sag 1994) und eliminiert alle ihre Bestandteile aus unserer Analyse. Die Verkleinerung des technischen Inventars zur Behandlung von (langer) Bewegung ist dabei drastisch. Während die Annahme, daß Spur und Binder ihren LOCAL-Wert teilen, von Pollard und Sag übernommen wird, ist der Mechanismus, der diese Strukturidentität herstellt, ein völlig anderer. Am deutlichsten wird dieser Unterschied, betrachtet man den Lexikoneintrag der Spur bei (Pollard und Sag 1994, S.164) und in unserer Grammatik:[22]

(5.15) *Vergleich der Lexikoneinträge für Spuren*

Traditionelle HPSG-Bewegungstheorie:



In der vorliegenden Neuformulierung:



Während bei uns eine Spur einfach ein Wort mit leerer Phonologie ist, das in Sätzen seinen LOCAL-Wert aufgrund eines grammatischen Prinzips mit seinem Binder teilt, müssen traditionell am Fuß einer langen Abhängigkeit technische Vorkehrungen getroffen werden, um die Identifikation der LOCAL-Werte von Binder und Spur zu initiieren. Hierzu wird

[22] (Pollard und Sag 1994) geben nicht explizit an, daß es sich um ein Wort handelt. Diese Information ist auch tatsächlich redundant, da sie sagen, daß die Beschreibung der Spur ein Lexikoneintrag ist ("as it appears in the lexicon"), womit sie nur meinen können, daß die Spurbeschreibung ein Disjunkt im Konsequens des Wortprinzips (3.6) ist. Aus Gründen der Deutlichkeit wiederholt (5.15) diese Information ausdrücklich.

ein Merkmal NONLOCAL auf *synsem* definiert, dessen komplexer Wert dazu benutzt wird, um in einem SLASH-Merkmal den LOCAL-Wert des Binders perkolieren zu lassen. QUE und REL dienen der Behandlung weiterer nichtlokaler Abhängigkeitsverhältnisse, die wir hier nicht diskutieren, von denen wir aber implizit annehmen, daß sie ebenfalls relational reformuliert werden können. Ein Perkolationsmechanismus, das Nichtlokale Merkmalsprinzip, muß in der Theorie von Pollard und Sag an jeder Phrase der Baumstruktur die korrekte Verteilung der NONLOCAL-Werte sicherstellen. Zum Abbinden der langen Abhängigkeit an ihrem Binder wird ein spezielles ID-Schema, das Kopf-Filler-Schema benötigt, das den Binder als Filler in die Baumstruktur einführt und mithilfe des Rechenmerkmals TO-BIND in Zusammenarbeit mit dem Nichtlokalen Merkmalsprinzip dafür sorgt, daß sein LOCAL-Wert nicht im Baum weiter nach oben perkoliert.

Aus der in unserem Ansatz vertretenen Sicht auf Merkmale ist die Kodierung von Bewegung in SLASH deshalb suspekt, weil dabei das Fehlen einer Konstituente in einer Strukur qua explizitem Merkmal als Eigenschaft dieser Struktur ausgewiesen wird. Mit der relationalen Ausdrucksweise soll dagegen vielmehr gesagt werden, daß zwischen zwei Konstituenten eine direkte Beziehung besteht, welche die Identität ihrer *local*-Objekte involviert. Dazwischenliegende Projektionen haben aber nicht die als solche kodierte Eigenschaft, daß ihnen eine Konstituente fehlt, und sie sind damit auch nicht Gegenstand von Prinzipien, welche Bewegung steuern. Letzteres ist der Zweck des Nichtlokalen Merkmalsprinzips, dem bei Pollard und Sag alle Kopfphrasen unterworfen werden.

Gemäß dem Programm, die Struktur sprachlicher Zeichen auf das zu beschränken, was direkt sprachwissenschaftlichen Intuitionen zugänglich ist, und technische Bedürfnisse nicht durch die Kreation anderweitig unmotivierter Struktur linguistischer Objekte zu befriedigen, unterliegt die *nonlocal*-Struktur dem gleichen Argument, demzufolge sowohl die DOM-Liste als Hilfsmittel der Wortstellungstheorie als auch die Kodierung der topologischen Felder des Deutschen in speziell dafür geschaffenen Objekten verworfen wurde. Damit bleibt lediglich die Beziehung zwischen der bewegten Konstituente und ihrer Spur auszudrücken. Genau dies und nichts sonst geschieht im FBP durch die Relation move-α. Alle Bestandteile der Signatur, welche in (Pollard und Sag 1994) ausschließlich der Behandlung der langen Abhängigkeiten dienen (die Merkmale NONLOCAL, INHERITED, TO-BIND und ihre Werte, sowie die Merkmale QUE, REL und SLASH), kommen in der relationalen Analyse ebenso wie das Head-Filler-Schema nicht vor.

Man kann sich leicht überlegen, daß das FBP weit mehr Extraktion lizenziert, als sich empirisch beobachten läßt. Sämtliche Nichtköpfe innerhalb der in Frage stehenden Baumstruktur des Satzes kommen als Spurpositionen in Betracht. Angenommen, Präpositionalphrasen werden als Kopf-Komplement-Phrasen analysiert, deren Kopf die Präposition ist, so sagt dies die Möglichkeit von Preposition Stranding voraus. Dies ist jedoch im Deutschen im allgemeinen ungrammatisch:

(5.16) a. *Dem Kanzler ist der Außenminister heute bei gewesen.
b. *Dem Fernglas hat die Frau den Mann mit gesehen.

Eine ähnliche Frage erhebt sich zum Beispiel hinsichtlich von Komplementierern oder Adjektiven in Nominalphrasen. Es ist offensichtlich, daß hier adäquate Bewegungsbeschränkungen zu formulieren sind, welche der empirischen Datenlage gerecht werden. Auf der anderen Seite sollten diese Beschränkungen nicht Teil des FBP selbst sein, sondern im Sinne einer modularen Architektur in Verbindung mit den jeweils zu beobachtenden Generalisierungen formuliert werden. In Abschnitt 5.4 wenden wir uns dem damit angedeuteten Problemkomplex zu und formalisieren eine Domänentheorie für syntaktische Bewegung, die illustriert, wie man sich grammatische Beschränkungen in einer relationalen Theorie syntaktischer Bewegung in der HPSG vorstellen kann. Es wird sich zeigen, daß eine solche Barrieren- oder Domänentheorie das FBP miteinschließt und dieses schließlich vollständig ersetzt.

5.2. Schnittstelle zu einer Illokutionstheorie. Es ist eine bekannte Eigenschaft der deutschen Syntax, daß Verbletztsätze nicht nur eingebettet, sondern sehr wohl auch uneingebettet auftreten können (vgl. (Brandt, Reis, Rosengren, und Zimmermann 1992)):

(5.17) a. Ob Karl den Hund auch gefüttert hat?
b. Daß du mir bloß nicht wieder davonläufst!
c. Wen du wieder mal alles gesehen hast!

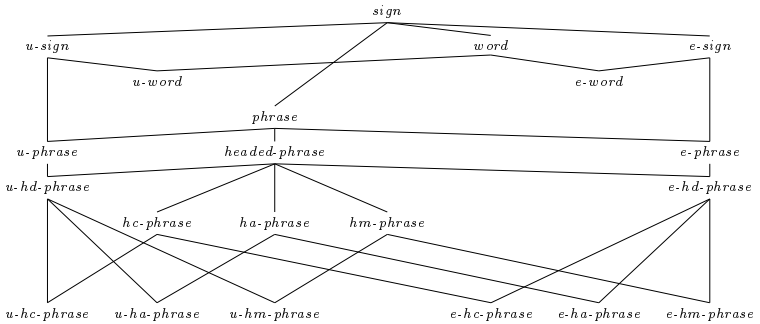
Es ist auch klar, daß die Sätze in (5.17) andere Eigenschaften haben als eingebettete VL-Sätze. Eine davon besteht darin, daß sie eine eigene illokutive Kraft aufweisen. Im Zusammenhang mit der Theorie der langen Bewegung und Überlegungen zu deren Beschränkung ist es außerdem bemerkenswert, daß sie im Gegensatz zu entsprechenden eingebetteten Sätzen keine innerhalb des Satzes ungebundenen Spuren enthalten dürfen:

(5.18) a. ?Den Hund weiß ich nicht, ob Karl *t* gefüttert hat.
b. *Ob Karl *t* gefüttert hat?
c. Den Hund habe ich geglaubt, daß Karl *t* füttert.
d. *Daß Karl *t* füttert!

Für eine HPSG-Theorie der (langen) Bewegung ergeben sich aus den angeführten Daten sofort einige Konsequenzen. Uneingebettete Zeichen müssen von eingebetteten Zeichen unterschieden werden. Uneingebettete Zeichen verfügen über eine illokutive Kraft, eingebettete hingegen nicht. Die Tatsache, daß ein Zeichen uneingebettet ist, impliziert darüber hinaus, daß innerhalb seiner keine unabgebundenen Spuren auftreten dürfen. Das bedeutet, daß (5.18b) und (5.18d) als eingebettete Zeichen von den Prinzipien der Grammatik sehr wohl lizenziert werden, da sonst die Sätze (5.18a) und (5.18c) nicht grammatisch sein könnten. Als selbständige Äußerungen (als uneingebettete Zeichen) hingegen sind sie ungrammatisch, da sie eine unabgebundene Spur enthalten. Mit anderen Worten könnte man sagen, daß selbständige Äußerungen Inseln für syntaktische Bewegung darstellen.

Zur Unterscheidung zwischen eingebetteten und uneingebetteten Zeichen wird die Sortenhierarchie unter *sign* um genau diese Differenzierung erweitert. Das heißt, die Sorte *sign* zerfällt in die Untersorten *e(mbedded)-sign* und *u(nembedded)-sign*, so wie sie in *phrase* und *word* unterteilt wird. *word* ist nicht mehr maximal spezifisch und *phrase* nicht mehr wie in (3.30) Obersorte von drei maximal spezifischen Sorten. Vielmehr hat *word* die neuen Untersorten *u-word* und *e-word*, und *phrase* wird in *headed-phrase*, *u-phrase* und *e-phrase* untergliedert, wobei die genannten Untersorten (mit Ausnahme von *headed-phrase*) zugleich auch jeweils Untersorten von *u-sign* beziehungsweise *e-sign* sind. Es ergibt sich damit hinsichtlich der in unserer Analyse betrachteten Untersorten von *sign* folgendes Gesamtbild:

(5.19) *Die erweiterte sign-Hierarchie*



Die maximal spezifischen Sorten in (5.19) sind *e-word* und *u-word*, sowie *e-hc-phrase*, *e-ha-phrase*, *e-hm-phrase*, *u-hc-phrase*, *u-ha-phrase* und

u-hm-phrase. Da es im betrachteten Fragment des Deutschen keine *non-headed-phrases* gibt, sind die Sorten *headed-phrase*, *u-hd-phrase* und *e-hd-phrase* in (5.19) eigentlich redundant; sie werden hier nur eingefügt, um deutlich zu machen, an welcher Stelle gegebenenfalls eine Erweiterung um Nichtkopfphrasen anzusetzen hätte. Man beachte, daß die erweiterte Sortenhierarchie von *sign* so aufgebaut ist, daß weder das Wortprinzip noch die reiche Anzahl von Prinzipien auf Phrasen mit ihrer Einführung modifiziert werden müssen. Dies rührt daher, daß die jeweiligen Beschränkungen ihrer Idee nach sowohl für eingebettete als auch für uneingebettete Zeichen gelten, die ihrerseits unter *word* und *phrase* vollständig subsumiert sind. Die Erweiterung der *sign*-Hierarchie dient dazu, zusätzliche Eigenschaften uneingebetteter Zeichen zu beschreiben.

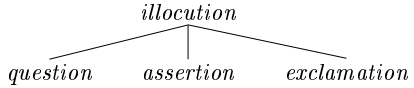
Die *u-signs* der Grammatik sind dadurch vor anderen ausgezeichnet, daß sie illokutive Kraft besitzen. Diese Eigenschaft uneingebetteter Zeichen wird durch ein Merkmal ILLOCUTION zum Ausdruck gebracht. (5.20) zeigt die in der Hierarchie (5.19) höchsten Sorten, auf denen die Attribute PHON, SYNSEM, ILLOCUTION, HEAD-DTR und NONHEAD-DTR definiert sind, sowie die jeweiligen Werte der Angemessenheitsfunktion $\mathfrak{F}$.

(5.20) *Die Angemessenheitsfunktion für die sign-Hierarchie*

sign	PHON	*list*
	SYNSEM	*synsem*
u-sign	ILLOCUTION	*illocution*
phrase	PHON	*nelist*
	HEAD-DTR	*e-sign*
	NONHEAD-DTR	*e-sign*

Durch die Merkmalsvererbung (siehe Abschnitt 2.1, Definition 2) ergibt sich, daß für *e-word* die Attribute PHON und SYNSEM definiert sind; für *u-word* zusätzlich ILLOCUTION. Auf Objekten der Untersorten von *e-phrase* PHON, SYNSEM, HEAD-DTR und NONHEAD-DTR; und auf *u-phrases* außerdem ILLOCUTION. Kopftöchter und Nichtkopftöchter sind aufgrund ihres Status als syntaktische Töchter größerer Strukturen *e-signs*, das heißt, sie können keine uneingebetteten Zeichen sein. Unter der Sorte *illocution*, welche der Wert der Angemessenheitsfunktion auf *u-sign* und ILLOCUTION ist, befindet sich die Hierarchie der Illokutionstypen.

(5.21) *Eine einfache Illokutionshierarchie*



Mit (5.21) wird keinesfalls der Anspruch verbunden, eine angemessene oder gar vollständige Sortenhierachie für Illokutionstypen einzuführen.

Vielmehr soll lediglich exemplarisch die Schnittstelle illustriert werden, an der eine tatsächliche Illokutionstheorie einzufügen wäre. Es steht zu erwarten, daß eine artikulierte Illokutionstheorie unter ILLOCUTION komplexe Strukturen mit eigenen, nur auf diesem Modul operierenden Prinzipien in die Grammatik einführt. Für unsere Zwecke genügt einstweilen die rudimentäre Struktur in (5.21), um Effekte der Interaktion zwischen Illokutionstheorie und Syntax zu skizzieren und das anhand der Daten in (5.17) und (5.18) angesprochene Problem zu lösen.

Auf der Basis der Definitionen in (5.19) – (5.21) lassen sich einfache Illokutionsprinzipien an der Schnittstelle von Syntax, Semantik und Pragmatik definieren. Semantisch sind die ob-VL-Sätze in (5.22) Interrogativsätze.

(5.22) a. Ob Maria heute abend zur Institutsfeier kommt?
b. Ich frage mich, ob Maria heute abend zur Institutsfeier kommt.

Ein einfaches Illokutionsprinzip könnte fordern, daß selbständige ob-Sätze Fragen sind, und damit zum Beispiel ausschließen, daß mit ihnen eine Assertionshandlung vollzogen werden kann.

(5.23) *Illokutionsprinzip 1*

Uneingebettete ob-VL-Sätze sind Fragen.

$$\left[\begin{array}{l} \textit{u-sign} \\ \text{SYNSEM} \left[\begin{array}{l}\text{STATUS}\ \ \textit{complete} \\ \text{LOC CAT HEAD MARKING}\ \ \textit{ob}\end{array}\right] \end{array}\right] \Longrightarrow$$

$$\left[\text{ILLOCUTION}\ \textit{question}\right]$$

Da der Satz (5.22a) eine uneingebettete Äußerung darstellt, muß es sich um eine *u-phrase* handeln. Mit (5.23) ist (5.22a) als ob-VL-Satz damit eine Frage. Als *e-phrase* hätte der Satz umgekehrt keine illokutive Kraft und könnte somit nicht als selbständige Äußerung auftreten. Man beachte, daß hier zwischen dem semantischen Begriff des Interrogativsatzes und dem pragmatischen Begriff der Frage unterschieden wird. So ist der eingebettete ob-VL-Satz in (5.22b) zwar semantisch ein Interrogativsatz. Pragmatisch handelt es sich jedoch nicht um eine Frage. Somit läßt sich das Prinzip (5.23) weder auf rein syntaktische noch auf semantische Eigenschaften von ob-Sätzen reduzieren.

Aufgrund des Antezedens der Implikation von Illokutionsprinzip 1 ist es bereits gegeben, daß (5.23) eine Aussage über uneingebettete Zeichen macht. Damit ergibt sich unmittelbar, daß es sich unabhängig von diesem Prinzip bei den von ihm betroffenen Zeichen bereits um Zeichen mit eigener illokutiver Kraft handelt. Illokutionsprinzip 1 hat lediglich noch die

Aufgabe, den ILLOCUTION-Wert uneingebetteter ob-Sätze auf den einen Wert *question* festzulegen. Parallel zu (5.23) ließe sich der Beobachtung Rechnung tragen, daß es sich bei selbständigen daß-VL-Sätzen niemals um Assertionen handeln kann.

Ein interessanterer Fall liegt in (5.24) vor, der zeigt, daß an der Syntax-Pragmatik Schnittstelle durchaus nichttriviale Prinzipien formuliert werden können.

(5.24) *Illokutionsprinzip 2*

w-V2-Sätze sind Fragen.

$$\forall x$$
$$:\left[\text{SS}\left[\begin{array}{l}\text{STATUS}\ \ \textit{complete}\\ \text{LOC CAT HEAD}\ \ \textit{verb2}\end{array}\right]\right] \wedge$$
$$^{x}\left[\text{SS LOC CAT HEAD FR CONST SS LOC CAT HEAD DEF}\ \ \textit{interrogativ}\right] \wedge$$
$$\texttt{contains-fr-word(:,}\ x\texttt{)} \Longrightarrow$$
$$:\left[\text{ILLOCUTION}\ \ \textit{question}\right]$$

Illokutionsprinzip 2 sagt, daß jede syntaktisch vollständige Projektion eine V2-Verbs, also jeder V2-Satz, deren FRONTED-Konstituente eine w-Konstituente ist, pragmatisch eine Frage sein muß. Im Unterschied zu Illokutionsprinzip 1 setzt (5.24) im Antezedens nicht voraus, daß nur über selbständige w-V2-Sätze im Unterschied zu eingebetteten gesprochen wird. Vielmehr macht Illokutionsprinzip 2 eine Aussage über alle w-V2-Sätze. Da das Merkmal ILLOCUTION jedoch nur auf *u-signs*, nicht aber auf *e-signs* definiert ist, hat (5.24) unmittelbar syntaktische Konsequenzen: w-V2-Sätze müssen selbständig und dürfen nicht eingebettet sein. Dies entspricht den Daten in (5.25).

(5.25) a. Wen hat Maria seit ihren Kindertagen bewundert?
b. *Ich frage mich, wen hat Maria seit ihren Kindertagen bewundert.
c. Ich frage mich, wen Maria seit ihren Kindertagen bewundert hat.

Mit anderen Worten folgt aus (5.24) die oftmals thematisierte und für Syntaxtheorien notorisch problematische Beobachtung, daß von einer w-Konstituente eingeleitete eingebettete finite Sätze VL-Sätze sein müssen (vgl. (Reis 1985)). Verbbewegung in eingebetteten w-Sätzen ist nicht zulässig.[23] Über den pragmatischen Status von w-VL-Sätzen hingegen macht Illokutionsprinzip 2 korrekterweise keine Vorhersagen.

[23] Verbbewegung in selbständigen w-Sätzen ist andererseits nicht obligatorisch, wie der selbständige w-VL-Satz (5.17c) belegt.

Die Unterscheidung zwischen eingebetteten und uneingebetteten Zeichen in der *sign*-Hierarchie (5.19) präjudiziert keine Aussagen hinsichtlich der syntaktischen Kategorie von Zeichen, die als selbständige Äußerungen auftreten können. Damit ist insbesondere nicht ausgeschlossen, daß neben finiten Sätzen auch andere Äußerungen als uneingebettete Zeichen auftreten. (5.26) enthält Beispiele für grammatische Äußerungen, die situationsabhängig ohne vorangegangenen Diskurs auftreten können und über illokutive Kraft verfügen.

(5.26) *Einige Beispiele für u-signs*

a. (Dieser) Idiot!	b. Adele.	c. Los!
d. Entschuldigung?	d. Hurra!	e. Sieg heil.
f. Tor!	g. Nein!	h. Ausgezeichnet.

Voraussetzung einer Beschreibung dieser Daten ist eine allgemeine Theorie uneingebetteter Zeichen. In Abwesenheit einer sochen Theorie beschränken wir uns in (5.27) auf denjenigen Ausschnitt davon, der finite Sätze im Sinne des Satzprinzips (5.6) betrifft. Die Beispiele in (5.26) machen deutlich, daß das Konsequens der Implikation (5.27) in einem realistischeren Ansatz zu einer Disjunktion erweitert werden muß.

(5.27) *Illokutionsprinzip 3 (erste Formulierung)*

Nur finite Sätze ohne offene Spuren haben eine eigene illokutive Kraft.

$$:\left[\textit{u-sign}\right] \Longrightarrow$$
$$:\left[\text{SS}\left[\begin{array}{l}\text{STATUS}\ \ \textit{complete}\\ \text{LOC CAT HEAD VFORM}\ \ \textit{finite}\end{array}\right]\right.$$
$$\left.\texttt{no-open-traces(:)}\right]$$

(5.27) bestimmt, daß uneingebettete Zeichen finite Sätze sein müssen. Die Relation no-open-traces stellt die zusätzliche Bedingung, daß in diesen Sätzen jede Spur einen Binder hat. An dieser Stelle soll no-open-traces nicht formalisiert werden. In Abschnitt 5.4 kommen wir auf (5.27) zurück (vgl. (5.73)) und werden zeigen, wie Domänenbegriffe verwendet werden können, um no-open-traces innerhalb einer allgemeinen Konzeption von Inselbeschränkungen für syntaktische Bewegung zu präzisieren.

Durch das Illokutionsprinzip 3 werden die finiten Sätze (5.18b) und (5.18d) ausgeschlossen, wohingegen (5.18a) und (5.18c), in denen sie eingebettet vorkommen, von der Grammatik lizenziert werden. Damit hat

sich gezeigt, daß eine Schnittstelle zu pragmatischen Prinzipien im Zusammenhang mit der Theorie langer Abhängigkeiten eine wesentliche Komponente der Grammatik darstellt, die auch für einfache Implementierungen dieses Phänomenbereichs benötigt wird.

Die in diesem Abschnitt dargelegte Theorie uneingebetteter Zeichen tritt in bewußten Gegensatz zu der von (Kathol 1995) vorgeschlagenen umfangreichen Sortenhierarchie für *clauses*, welche mit der traditionellen *sign*-Hierarchie kreuzklassifiziert wird. In diesem Ansatz muß für jeden Satztyp eine Sorte eingeführt werden, über die Prinzipien operieren, welche die entsprechende Form der jeweiligen Zeichen sicherstellen. Dabei muß diese Sortenhierarchie nicht nur *clauses* (mit den (nichtmaximalen) Untersorten *root*, *subord*, *interrog* und *decl*, siehe (Kathol 1995, S.197)) enthalten, sondern auch Sorten für sogenannte *nonclauses* und *nonconstructions*, wobei bei letzteren an nichtmaximale Projektionen gedacht ist. Hier scheint im wesentlichen eine phrasenstrukturelle Theorie syntaktischer Kategorien und Projektionsstufen als Sortenhierarchie für Phrasen rekonstituiert zu werden, welche Information, die im Grunde bereits in der Struktur der Zeichen enthalten und somit als Eigenschaft der Zeichen gegenwärtig ist, dupliziert. So wissen wir zum Beispiel von einem *u-sign* mit HEAD-Wert *verb2* bereits aufgrund der gegebenen Prinzipien, daß es sich um einen Root-V2-Satz handelt, und müssen dafür nicht eigens eine zusätzliche Sorte unter *sign* einführen. Entsprechend ist eine *e-phrase* mit STATUS-Wert *complete* und MARKING-Wert *daß* ein subordinierter daß-VL-Satz. Insofern unterliegt die Katholsche *sign*-Hierarchie für Satztypen dem gleichen Sparsamkeitsargument für Strukturen, aufgrund dessen wir die Attribute DOM und NONLOCAL mit den mit ihnen jeweils assoziierten Strukturen und eine explizite Kodierung des topologischen Feldermodells in speziell dafür geschaffenen Sorten zurückgewiesen haben.

5.3. Verberstparenthesen statt Extraktion aus V2. Bevor sich Abschnitt 5.4 der Bestimmung von Extraktionsdomänen und der Frage zuwendet, wie sich diese und Inselbeschränkungen in einer mit Relationen arbeitenden Linearisierungsgrammatik formalisieren lassen, gilt es zunächst, näher zu bestimmen, welche Satztypen für lange Bewegung potentiell durchlässig sind. Im vorliegenden Abschnitt wird eine HPSG-Formalisierung der in (Reis 1995) auf der Basis einer ausführlichen empirischen Untersuchung verteidigten Verberstparenthesenanalyse für bestimmte Konstruktionen vorgestellt, die es ermöglichen wird, V2-Sätze als strikte Inseln für Bewegung zu analysieren. Dies wird in 5.4 zu einer unerwarteten Simplizität der Theorie der langen Abhängigkeiten beitragen.

Oben wurde bereits festgestellt (siehe (5.10)), daß V2-Sätze mit overt besetztem Vorfeld strikte Inseln für lange Bewegung sind. Dagegen ist es auf den ersten Blick eine offene Frage, ob Extraktion aus V2-Sätzen dann möglich ist, wenn das Vorfeld nicht overt besetzt ist.

(5.28) a. Wen glaubst du, kann Maria nicht leiden?
b. Den Pförtner meint Jürgen, kann Maria nicht leiden.
c. Wen glaubst du, daß Maria nicht leiden kann?
d. Den Pförtner meint Jürgen, daß Maria nicht leiden kann.
e. Ruben glaubt, Autos machen brummbrumm.
f. Ich meine, Ruben sollte noch etwas mit seinen Legos spielen.

Da sowohl *glauben* als auch *meinen* Brückenverben sind (5.28c–d), die V2-Sätze einbetten (5.28e–f), liegt der Gedanke nicht fern, daß (5.28a–b) Instanzen von Brückenkonstruktionen sind und die Vorfeldkonstituenten *wen* und *den Pförtner* jeweils lang aus dem eingebetteten V2-Satz extrahiert wurden. Diese Überlegungen resultierten etwa in folgender Analyse:

(5.29) a. Wen glaubst du, t' kann Maria t nicht leiden?
b. Den Pförtner meint Jürgen, t' kann Maria t nicht leiden.

In der Tat wird eine Extraktionsanalyse im Sinne von (5.29) in der generativen Literatur allgemein angenommen. Zu ihren Kennzeichen gehört es, daß es neben der ursprünglichen Spur t der bewegten Konstituente eine weitere Zwischenspur t' im Vorfeld des eingebetteten Satzes gibt. Das heißt, es wird gefordert, daß das bewegte Element unabhängig von seiner Ursprungsposition eine Spur im Vorfeld zurückläßt. Wie wir oben gesehen haben, ist daneben das Vorliegen einer Spur im Gegensatz zu einem overten Element Voraussetzung dafür, daß aus V2 überhaupt extrahiert werden kann. Neben dem Analysevorschlag der generativen Linguistik steht die Annahme der traditionellen Grammatiktheorie, die von einer Parenthese ausgeht:

(5.30) a. Wen [glaubst du] kann Maria t nicht leiden?
b. Den Pförtner [meint Jürgen] kann Maria t nicht leiden.

In (5.30) werden *glaubst du* beziehungsweise *meint Jürgen* als Integrierte Verberstparenthesen betrachtet, welche vor dem finiten Verb des V2-Satzes phonologisch eingeschoben werden. In den beiden V2-Sätzen liegt damit einfache Topikalisierung oder w-Bewegung vor. Diese Analyse behandelt (5.30b) parallel zu den unstrittigen Verberstparenthesen in (5.31).

(5.31) Den Pförtner kann Maria (meint Jürgen) nicht leiden (meint Jürgen).

Reis (1995) zeigt in einer detaillierten Studie, daß starke Gründe dafür sprechen, die in Frage stehenden Daten als Integrierte Verberstparenthesen zu behandeln, und viele Gründe dagegen, sich der Extraktionsanalyse anzuschließen. Die Eigenschaften von Integrierten V1-Parenthesen sind V1-Stellung, interpretatorische Abhängigkeit von dem Satz, in den sie integriert sind und der das propositionale Argument ihres Hauptverbs ist, sowie die prosodische Integration in diesen Satz, der sich in einer fehlenden eigenen Fokus-Hintergrundstruktur, einem Fehlen von Betonung und der Abwesenheit von intonatorischen Pausen am Anfang und Ende der V1-Parenthese manifestiert. Diese Eigenschaften liegen beim Stellungstyp (5.28a–b) ebenso wie bei den in (5.31) illustrierten Stellungstypen der Parenthese vor. (Reis 1995) belegt den Parenthesenstatus der Konstruktion mit einer ganzen Reihe von empirischen Tests, auf die hier nicht näher eingegangen werden kann. Zumindest erwähnt sei aber die Tatsache, daß Reis mit der Klasse der Präferenzprädikate eine Gruppe von Brückenverben identifiziert, die als Brückenverben zwar lange Extraktion aus subkategorisierten daß-Komplementen zulassen und auch V2-Sätze einbetten, jedoch nicht in Verberstparenthesekonstruktionen auftreten (Reis 1995, S.45–48).

Auf der anderen Seite hat die Extraktionsanalyse traditionell beträchtliche Beschreibungsprobleme. Während sich die Bedingung, daß das Vorfeld des putativ eingebetteten V2-Satzes nicht overt besetzt sein darf aber eine Spur des extrahierten Elements enthalten muß, noch relativ natürlich anläßt, trägt die sogenannte V2 Route Restriction (Reis 1995, S.34) ausgesprochen häßliche Züge. Die V2 Route Restriction (V2RR) fordert, daß Extraktion aus V2 nur über V2-Sätze und in V2-Sätze erfolgen darf. Eine solche Bedingung widerspricht den allgemein verbreiteten Ansprüchen an einen natürlichen und intuitiven linguistischen Constraint, und die Literatur der Extraktionsanalyse von (5.28a–b) ist vor allem im Lichte der Bemühung zu verstehen, diese Bedingung aus anderen Gesetzmäßigkeiten der langen Bewegung abzuleiten.

(5.32) a. *Ich möchte wissen, wen er meint, t' kann Maria t nicht leiden.
b. Ich möchte wissen, wen er meint, daß Maria nicht leiden kann.
c. *Den Pförtner meint Jürgen, t'' daß Hans glaubt, t' kann Maria t nicht leiden.
d. Den Pförtner meint Jürgen, daß Hans glaubt, daß Maria nicht leiden kann.
e. *Wen Hans nur wieder meint, t' kann Maria t nicht leiden.
f. Wen Hans nur wieder meint, daß Maria nicht leiden kann.

(5.32) illustriert den Effekt der V2RR. Den ungrammatischen Sätzen sind die Spuren beigegeben, welche eine Extraktionsanalyse der Daten üblicherweise postulieren würde. In (5.32a) wurde *wen* aus einem V2-Satz extrahiert und in die linksperiphere Position des einbettenden VL-Satzes w-bewegt. In (5.32c) liegt lange Topikalisierung der Konstituente *den Pförtner* aus dem am tiefsten eingebetteten V2-Satz ins Vorfeld des Matrixsatzes vor; indessen wurde über einen dazwischenliegenden VL-Satz bewegt. (5.32e) schließlich zeigt eine w-Bewegung aus einem eingebetteten V2-Satz in einen uneingebetteten Matrix-VL-Satz. Die beigegebenen Kontrollsätze belegen jeweils, daß parallele Konstruktionen der entsprechenden Brückenverben mit eindeutiger langer Extraktion aus VL-Sätzen stets zulässig ist.

Unter der V1-Parenthesenanalyse und der Annahme, daß lange Bewegung aus V2-Sätzen grundsätzlich verboten ist, ergibt sich die Ungrammatikalität der angeführten Sätze ohne weitere Stipulationen. *er meint* in (5.32a) ist keine wohlgeformte V1-Paranthese. (5.32c) enthält mit *meint Jürgen* zwar eine solche, aber der Satz, in den sie eingeschoben wurde, ist nicht wohlgeformt. In (5.32e) gibt es wiederum keine wohlgeformte V1-Parenthese. Die grammatischen Sätze sind dagegen Instanzen von langer Bewegung und erhalten somit eine unabhängige und gänzlich andere Strukturanalyse.

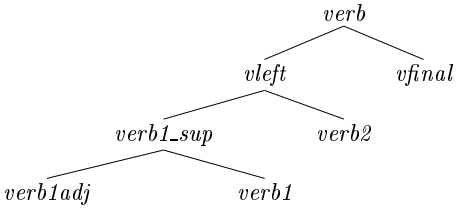
Die im folgenden dargestellte HPSG-Analyse schließt sich den Schlußfolgerungen von Reis (1995) an und behandelt die eingangs in (5.30) und (5.31) zusammengefaßten Daten gleichermaßen als Integrierte V1-Parenthesen. Da Parenthesen bislang in HPSGrammatiken des Deutschen nicht behandelt wurden, stellt sich damit sogleich die Frage nach ihrer strukturellen Position in linguistischen Zeichen und ihrer Interaktion mit dem Rest der Grammatik. Zunächst ist klar, daß sich V1-Verben in V1-Parenthesen von V1-Verben in herkömmlichen V1-Sätzen unterscheiden müssen. Das *verb1*-Verb *sagt* realisiert sein Subjekt und seinen V2-Komplementsatz strukturell als Komplemente, wohingegen das V1-Parenthesenverb *sagt* offenbar nur sein Subjekt dieserart realisiert und seine propositionale Argumentstelle, das heißt den Satz, in den die V1-Parenthese phonologisch integriert wird, auf andere Weise bindet.

Da V1-Parenthesen von Bedeutung und Funktion her ähnliche Eigenschaften wie Satzadverbiale aufweisen — obwohl sie sich von diesen zum Beispiel hinsichtlich ihrer Stellungseigenschaften unterscheiden, vgl. (Reis 1995, S.74–75) —, wird im folgenden eine Zuordnung von Integrierten V1-Parenthesen zu adverbialen Konstruktionen ins Auge gefaßt. Neben den Valenzlisten stellt die HPSG die synsemwertigen Selektionsmerkmale

SPEC und MOD bereit. Ersteres wird in (Pollard und Sag 1994) für Spezifizierer und Marker verwendet, letzteres zur Behandlung der syntaktischen Selektions- und semantischen Modifikationseigenschaften von Adverbialen und Adjektiven. Das heißt, daß V1-Parenthesenverben ihr propositionales Argument mit dem MOD-Merkmal selegieren und das Kopf-Adjunkt-Schema herangezogen wird, um sie in die syntaktische Baumstruktur zu integrieren. Ihr idiosynkratisches Verhalten, auf das im folgenden eingegangen wird, leitet sich aus den Eigenschaften ab, die sie sowohl von Adverbialen als auch von anderen finiten Verbprojektionen unterscheiden.

Für V1-Parenthesenverben wird eine neue Kopfsorte *verb1adj* in die Sortenhierarchie unter *verb* (siehe (3.25)) eingeführt, welche sie von anderen finiten V1-Verben lexikalisch differenziert. Hierzu wird in (5.33) unmittelbar unter *vleft* eine neue, nicht maximal spezifische Sorte *verb1_sup* eingefügt, die sowohl *verb1adj* als auch *verb1* subsumiert.

(5.33) *Die erweiterte verb-Hierarchie*



Eine Besonderheit von *verb1adj*-Verben, die in (5.33) nicht angegeben ist, besteht darin, daß ihre Kopfsorte *verb1adj* nicht nur Untersorte von *verb* sondern zugleich Untersorte von *adjunct*, der Obersorte der Adverbhierarchie, ist und sie damit auch Eigenschaften von Adjunkten teilen. Was das bedeutet, wird in (5.34) deutlich, wo ein Ausschnitt der Angemessenheitsfunktion $\mathfrak{F}$ für die Verbhierarchie wiedergegeben ist, der die Vererbung von Attributen und ihren Werten in der Signatur verdeutlicht.

(5.34) *Die Angemessenheitsfunktion für die erweiterte verb-Hierarchie*

$$
\begin{array}{lll}
\mathit{verb} & \textsc{mod} & \mathit{synsem_or_minus} \\
 & \textsc{vform} & \mathit{vform} \\
 & \mathit{vfinal} & \textsc{mod} \quad \mathit{minus} \\
 & \mathit{vleft} & \textsc{fronted} \quad \mathit{locality} \\
 & & \textsc{vform} \quad \mathit{finite} \\
 & \mathit{verb2} & \textsc{fronted} \quad \mathit{moved} \\
 & & \textsc{mod} \quad \mathit{minus} \\
 & \mathit{verb1_sup} & \textsc{fronted} \quad \mathit{empty} \\
 & & \mathit{verb1} \quad \textsc{mod} \quad \mathit{minus} \\
 & & \mathit{verb1adj}
\end{array}
$$

Wie bereits in Kapitel 3 erläutert, hat *verb* die direkten Untersorten *vfinal* und *vleft*. Für *vleft* ist neben den allgemein auf *verb* definierten Attributen MOD und VFORM auch FRONTED definiert, und Verben, deren Kopfwert von *vleft* beschrieben wird, haben den VFORM-Wert *finite*. Die Signatur bestimmt also bereits, daß Verben in der linken Satzklammer finit sind. Direkte Untersorten von *vleft* sind in der erweiterten *verb*-Hierarchie *verb2* und die neue Sorte *verb1_sup*. Für *verb2* bleibt die Angemessenheitsfunktion gegenüber Kapitel 3 unverändert. Unter *verb1_sup* subsumierte Objekte tragen das FRONTED-Merkmal, und sein Wert ist auf *empty* festgelegt. Während nun aber die Signatur für *verb1* wie für *vfinal* und *verb2* den MOD-Wert *minus* erzwingt, kommt bei *verb1adj* zum Tragen, daß diese Sorte auch Untersorte von *adjunct* ist und damit den MOD-Wert *synsem* erbt. Mit anderen Worten verfügen *verb1adj*-Verben im Unterschied zu allen anderen finiten Verbformen und in Übereinstimmung mit Adverbien über modifizierende Eigenschaften.

Man bemerke auch, daß aufgrund des synsemwertigen MOD-Merkmals V1-Parenthesenverben nicht von dem einheitlichen Lexikoneintrag für alle drei Stellungsvarianten eines finiten Verbs beschrieben werden, der in (3.26) anhand von *geht* exemplifiziert wird. Dies liegt daran, daß dort der MOD-Wert *minus* gefordert wird, der nur mit V2-, VL- und nichtmodifizierenden V1-Verben kompatibel ist. Das heißt aber, daß durch die Erweiterung der *verb*-Hierarchie alleine noch keine weiteren Wörter durch die Grammatik lizenziert werden. Um tatsächlich V1-Parenthesenverben zu erhalten, müssen erst entsprechende Lexikoneinträge (oder lexikalische Regeln) zur Grammatik hinzugefügt werden.[24]

Während weiter unten gezeigt wird, wie V1-Parenthesenverben durch über Verbklassen generalisierende lexikalische Regeln lizenziert werden können, betrachten wir zunächst in (5.35), wie ein konkreter Lexikoneintrag aussehen könnte.

[24] Diese Überlegung geht selbstverständlich davon aus, daß es neben den besprochenen auch keine anderen Lexikoneinträge gibt, die durch die Erweiterung der Sortenhierarchie neue Wörter lizenzieren.

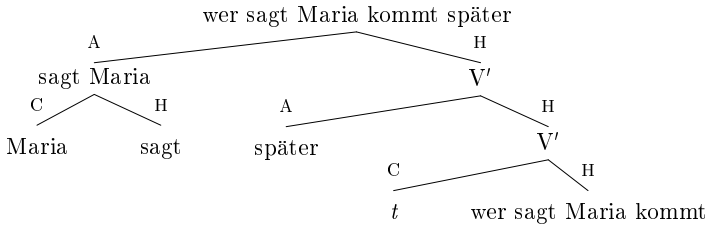
(5.35) *Vorläufiger Lexikoneintrag für ein verb1adj-Verb: sagt*

$$\left[\begin{array}{l} \textit{word} \\ \textsc{phon}\ \langle \textit{sagt} \rangle \\ \textsc{ss loc}\ \left[\begin{array}{l} \textsc{cat}\ \left[\begin{array}{l} \textsc{head}\ \left[\begin{array}{l} \textit{verb1adj} \\ \textsc{mod}\ \left[\begin{array}{l} \textit{synsem} \\ \textsc{loc}\ \left[\begin{array}{ll} \textsc{cat head} & \textit{verb2} \\ \textsc{cont} & \textit{cont} \end{array}\right] \end{array}\right] \end{array}\right] \\ \textsc{marking}\ \textit{unmarked} \end{array}\right] \\ \textsc{val comps}\ \langle \textsc{dp}[\textit{complete}] \rangle \end{array}\right] \end{array}\right]$$

Der FRONTED- und VFORM-Wert (*empty* beziehungsweise *finite*) ergibt sich aus der Signatur, so daß sie nicht im Lexikoneintrag wiederholt werden müssen. *sagt* selegiert nur sein Subjekt auf der COMPS-Liste. Dagegen wird das propositionale Argument über den MOD-Wert selegiert. Die einzige Forderung, die ihm von *sagt* auferlegt wird, besteht in der Forderung nach einer *verb2*-Projektion. Da der CONTENT der modifizierten V2-Projektion zugänglich ist, hat *sagt* darauf genauso Zugriff wie auf den CONTENT-Wert seines Subjektes. Unabhängig von der Wahl der semantischen Strukturen kann folglich davon ausgegangen werden, daß die für eine adäquate semantische Repräsentation erforderlichen Strukturidentitäten mit dem semantischen Wert von *sagt* selbst hergestellt werden können.

Auf der Grundlage des Lexikoneintrages (5.35) ergibt sich folgende syntaktische Struktur eines Satzes mit V1-Parenthese, die freilich sofort einige neue Fragen aufwirft:

(5.36) *V1-Parenthesen in der syntaktischen Struktur*



Das Subkategorisierungsprinzip (3.36) stellt bereits sicher, daß die mit dem Kopf-Adjunkt-Schema (3.32) an die V2-Projektion angefügte V1-Parenthese nur oberhalb der Komplemente des V2-Satzes an den V2-Satz treten kann. Die zusätzliche Bedingung, daß V1-Parenthesen die höchsten Adjunkttöchter im Satz sind, ist durch ein eigenes Prinzip zu erzwingen,

auf dessen (triviale) Formalisierung hier verzichtet wird. Bei seiner Formulierung ist darauf zu achten, daß ein V2-Satz potentiell mehrere V1-Parenthesen enthalten kann.

Auf den ersten Blick weit schwieriger zu beantworten scheint die Frage, wie es möglich ist, daß die Phonologie der V1-Parenthese zwischen die Phonologie der Vorfeldkonstituente und die des finiten Verbs tritt. Auf der Basis des in Kapitel 3 gegebenen Lexikoneintrags für finite Verbformen, der bestimmt, daß die Phonologie eines V2-Verbs aus der Phonologie der Vorfeldkonstituente und dem verbspezifischen phonologischen Eigenbeitrag besteht, ist (5.36) in der Tat ausgeschlossen. In (5.36) besteht die Phonologie des verbalen Kopfes aus der Phonologie des Vorfeldelements, der Phonologie der V1-Parenthese und dem phonologischen Eigenbeitrag des V2-Verbs.

Daneben bestimmt die Definition der Permutationsdomänen von Sätzen in Kapitel 4, daß *sagt Maria* Element der Permutationsdomäne des Gesamtsatzes ist. Das Phonologieprinzip (4.14) fordert, daß die Phonologie einer Phrase eine Permutation der Phonologien der Elemente der Permutationsdomäne dieser Phrase ist. Im Zusammenspiel mit der Openerregel ergeben sich damit bislang lediglich folgende Stellungsmöglichkeiten:

(5.37) a. ??wer kommt [sagt Maria] später
b. wer kommt später [sagt Maria]

Die Ungrammatikalität der Stellung (5.37a) unmittelbar hinter dem finiten V2-Verb ist in der im Linearisierungsmodul zu formulierenden Theorie der Stellung von V1-Parenthesen auszuschließen. Diese Theorie ist einer der wichtigen Punkte, in denen sich Integrierte V1-Parenthesen von Satzadverbialen unterscheiden. Die satzinitiale Stellung wird bereits korrekt durch die Openerregel ausgeschlossen.

Um die bevorzugte Stellung zwischen Phonologie von Vorfeldkonstituente und finitem Verb zuzulassen, muß die Relation construct-phon, die im Lexikoneintrag finiter Verben verwendet wird, modifiziert werden. Dazu wird die ursprüngliche Definition (3.27) durch die nachstehende ersetzt:

(5.38) *Neudefinition von construct-phon*

$$\begin{aligned} &\texttt{construct-phon}(x, y) := \\ &\qquad \neg\, x\textsc{ss loc cat hd fr const} \approx x\textsc{ss loc cat hd fr const} \\ &\qquad \wedge\, x\textsc{phon} \approx y \end{aligned}$$

$\texttt{construct-phon}(x, y) :=$
$\quad x$SS LOC CAT HD FR CONST $\approx x$SS LOC CAT HD FR CONST
$\quad \wedge\ \texttt{initial-segment}(x$SS LOC CAT HD FR CONST PHON$, x$PHON$)$
$\quad \wedge\ \texttt{final-segment}(y, x$PHON$)$

Die erste Klausel bleibt unverändert: Wenn FRONTED CONST auf dem Kopf des Zeichens x nicht definiert ist, dann besteht seine Phonologie aus y. Im Falle der Verwendung von construct-phon im Lexikoneintrag finiter Verben betrifft das aufgrund der Signatur genau die V1- und die VL-Form des Wortes. Die zweite, modifizierte Klausel behandelt den Fall von V2-Verben, das heißt FRONTED CONST ist auf dem Kopf definiert. Die Relation initial-segment fordert, daß die Phonologie von x mit der Phonologie seiner FRONTED-Konstituente beginnt. final-segment bestimmt, daß die Phonologie von x mit y, dem phonologischen Eigenbeitrag, endet. Damit wird die Möglichkeit offengelassen, daß weiteres phonologisches Material zwischen die des Vorfelds und des finiten Verbs tritt.

(5.39) *Klauseln von initial-segment und final-segment*

$\texttt{initial-segment}(x, y) :=$
$\quad x \sim \mathit{elist}$
$\quad \wedge\ y \sim \mathit{list}$
$\texttt{initial-segment}(x, y) :=$
$\quad x$FIRST $\approx y$FIRST
$\quad \wedge\ \texttt{initial-segment}(x$REST$, y$REST$)$

$\texttt{final-segment}(x, y) :=$
$\quad x \approx y$
$\texttt{final-segment}(x, y) :=$
$\quad \texttt{final-segment}(x, y$REST$)$

Die leere Liste x ist das Anfangssegment einer beliebigen Liste y (Klausel 1 von initial-segment). Liste x ist außerdem ein Anfangssegment von Liste y, wenn das erste Element von x mit dem ersten Element von y übereinstimmt und der Rest der beiden Listen in der initial-segment-Relation stehen.

final-segment gilt zwischen x und y, wenn sie identisch sind. Außerdem ist Liste x ein Schlußsegment von Liste y, wenn es in der final-segment-Relation mit dem Rest von y steht.

Es ist offensichtlich, daß initial-segment und final-segment in der Neudefinition von construct-phon das Gewünschte leisten. Die Neudefinition der Relation construct-phon für die Lexikoneinträge finiter Verben hat zur Folge, daß die in (5.36) angegebene Phonologie *wer sagt Maria kommt* des lexikalischen Kopfes der Verbprojektion durch den Lexikoneintrag von *kommt* nun tatsächlich lizenziert wird. Dem ist so, weil die Phonologie der FRONTED-Konstituente *wer* ihr Anfangssegment bildet, und der im Lexikoneintrag von *kommt* geforderte phonologische Eigenbeitrag $\langle \mathit{kommt} \rangle$ das Schlußsegment.

Während die Integration der Phonologie der V1-Parenthese in V2-Opener damit ermöglicht wird, ist noch nicht geklärt, wie die Phonologie der V1-Parenthese in der konkreten Struktur (5.36) in den Opener integriert wird statt daß die V1-Parenthese als Element in die Permutationsdomäne des Gesamtsatzes aufgenommen wird. Außerdem stellt sich die Frage, wie verhindert werden soll, daß plötzlich beliebiges phonologisches Material zwischen Vorfeld und finitem Verb erscheint. Auf der einen Seite ist die Grammatik im Augenblick also nicht restriktiv genug, da nur der Anfang und das Ende des phonologischen Beitrags von V2-Openern kontrolliert wird. Auf der anderen Seite ist sie aber zugleich auch zu restriktiv, weil V1-Parenthesen als Nichtkopftöchter in Sätzen nur in die Permutationsdomäne der Sätze eingehen können und dadurch ihre Phonologie nicht in präfiniter Position erscheinen kann. Dies ist nicht einmal aufgrund der ungenügenden Restriktivität von V2-Verben möglich, weil in Kapitel 4 gefordert wurde, daß jedes Element seinen phonologischen Beitrag nur einmal leistet, da es auf PHON-Listen keine Strukturidentitäten gibt. Das heißt, zwischen Vorfeldphonologie und der Phonologie des finiten Verbs kann zwar beliebiges Material eingeschoben werden, aber keines, das strukturidentisch mit phonologischen Elementen ist, die anderswo auf der betreffenden PHON-Liste stehen.

Wir gehen im folgenden in zwei Schritten vor. Zuerst wird die Behandlung von Permutationsdomänen so erweitert, daß V1-Parenthesen neben den bereits gegebenen Stellungmöglichkeiten auch zwischen Vorfeld und finitem Verb erscheinen können. Im Anschluß daran wird dafür Sorge getragen, das phonologische Integrationspotential von V2-Verben auf V1-Parenthesen zu beschränken.

Die erforderliche Erweiterung des Permutationsdomänenbegriffs ist durch eine zusätzliche Klausel der Relation perm-domain leicht zu erreichen. Ein solcher Eingriff an einem zentralen Punkt der Linearisierungskomponente scheint insofern gerechtfertigt, als hier mit Parenthesen die Behandlung eines genuin neuen empirischen Phänomens in die Grammatik aufgenommen wird. Aus dieser Perspektive ist die Klausel (5.40) als erste Annäherung an eine Formulierung zu verstehen, welche nicht nur speziell auf V1-Parenthesen zugeschnitten ist, sondern auch andere parenthetische Konstruktionen berücksichtigt.

(5.40) *Eine zusätzliche Klausel von perm-domain*

$$\begin{aligned}&\texttt{perm-domain}(x,y) :=\\ &\qquad y\textsc{head-dtr ss loc cat head} \sim \textit{verb2}\\ &\qquad \wedge\ y\textsc{nonhead-dtr ss loc cat head} \sim \textit{verb1adj}\\ &\qquad \wedge\ \texttt{perm-domain}(x, y\textsc{head-dtr})\end{aligned}$$

(5.40) stellt eine alternative Behandlung von Nichtkopftöchtern der Projektion von V2-Verben bereit, die V1-Parenthesen sind. Durch (5.40) müssen V1-Parenthesen nicht in die Permutationsdomäne ihrer Mutter übernommen werden. Man beachte, daß die letzte Zeile impliziert, daß bei der Anwendung der perm-domain Klausel in (5.40) V1-Parenthesen sich nicht direkt mit V2-Wörtern verbinden können, da von der Kopftochter verlangt wird, daß sie eine eigene Permutationsdomäne hat. Permutationsdomänen sind aber nur auf Phrasen, nicht aber auf Wörtern definiert. Bei der Anwendung der neuen Klausel von perm-domain ist in Struktur (5.36) die Permutationsdomäne des höchsten Knotens im Satz einfach identisch mit der Permutationsdomäne seiner Kopftochter.

(5.40) führt einen neuen Typ von Klausel in die Definition von perm-domain ein. Während bislang an jeder Phrase die phonologischen Beiträge aller Töchter bewahrt wurden, indem sie entweder als ganze oder (rekursiv formuliert) in Form ihrer Töchter in der Permutationsdomäne repräsentiert wurden, erlaubt die neue Klausel, V1-Parenthesen bei der Konstruktion der Permutationsdomäne ihrer Mutter zu ignorieren. Aus diesem Grund muß jetzt ein Prinzip explizit formuliert werden, das bislang implizit in der Interaktion der Formalisierung von Permutationsdomänen und dem Phonologieprinzip (4.14) enthalten war, nämlich das Prinzip der Vollständigen Phonologischen Realisierung:

(5.41) *Das Prinzip der Vollständigen Phonologischen Realisierung (PVPR)*

Die Phonologie jedes von einer Phrase P dominierten Wortes ist Teil der Phonologie eines Elementes der Permutationsdomäne von P.

$$\begin{aligned}&\forall x\\ &{}^{:}\left[\textit{phrase}\right] \wedge\\ &{}^{x}\left[\begin{array}{l}\textit{word}\\ \textsc{phon}\ \boxed{1}\end{array}\right] \wedge\\ &\texttt{dominate}(:, x) \Longrightarrow\\ &(\forall a\ \texttt{perm-domain}(a, :) \Longrightarrow (\exists p\ \texttt{signphon}(p, a) \wedge \texttt{sublist}(\boxed{1}, p)))\end{aligned}$$

Das PVPR bewirkt, daß die Nichtaufnahme von V1-Parenthesen in die Permutationsdomäne von V2-Sätzen nicht dazu führen kann, daß auch ihre Phonologie im Gesamtsatz ignoriert werden darf. Aufgrund des PVPR muß der phonologische Beitrag eines jeden Wortes, das Bestandteil einer Baumstruktur ist, in der Phonologie irgendeines Elementes der Permutationsdomäne berücksichtigt werden.

Die Formalisierung des PVPR macht von den mit RSRL bereitgestellten Ausdrucksmöglichkeiten ausgiebig Gebrauch und demonstriert, wie selbst komplexe Prinzipien kompakt und übersichtlich aufgeschrieben werden können. Für jede Phrase P und für jedes Wort x, so daß P das Wort x baumstrukturell dominiert, gilt, daß in jeder für P definierten Permutationsdomäne die Phonologie von x Bestandteil des durch die Permutationsdomäne bestimmten phonologischen Materials von P ist. (5.41) verwendet neben perm-domain und der in Kapitel 4 in (4.18) definierten Relation signphon, die einer Kette von Zeichen die Phonologie ihrer Elemente in der Reihenfolge ihres Auftretens in der Kette zuordnet, die neuen zweistelligen Relationen dominate und sublist. dominate entspricht dem Begriff der Dominanz in einer syntaktischen Baumstruktur, und sublist gibt die Teillisten einer beliebigen Liste an:

(5.42) *Definition der Klauseln für dominate und sublist*

$$\begin{aligned}&\texttt{dominate}(x, y) :=\\ &\qquad x \approx y\\ &\qquad \vee\ \texttt{dominate}(x\textsc{head-dtr}, y)\\ &\qquad \vee\ \texttt{dominate}(x\textsc{nonhead-dtr}, y)\\ &\\ &\texttt{sublist}(x, y) :=\\ &\qquad \texttt{sublist}(x, y\textsc{rest})\\ &\qquad \vee\ \texttt{sublist-aux}(x, y)\\ &\\ &\texttt{sublist-aux}(x, y) :=\\ &\qquad x \sim \textit{elist}\\ &\texttt{sublist-aux}(x, y) :=\\ &\qquad x\textsc{first} \approx y\textsc{first}\\ &\qquad \wedge\ \texttt{sublist-aux}(x\textsc{rest}, y\textsc{rest})\end{aligned}$$

Die Rekursionsverankerung und die beiden Rekursionsfälle von dominate werden in einer disjunktiven Klausel formuliert. x dominiert y genau dann, wenn x und y identisch sind oder die Kopftochter oder Nichtkopftochter von x y dominiert.

sublist verwendet eine Hilfsrelation sublist-aux. x ist eine Teilliste von y genau dann, wenn x in der sublist-Relation zum Rest von y oder in der

sublist-aux-Relation zu y steht. sublist-aux gilt zwischen jeder leeren Liste x und einem beliebigen y (Klausel 1). Außerdem gilt sublist-aux zwischen zwei nichtleeren Listen x und y, wenn ihre ersten Elemente identisch sind und der Rest von x in der sublist-aux-Relation zum Rest von y steht. Die Rekursion terminiert mit der ersten Klausel, wenn die leere Liste x erreicht ist.

Durch Induktion über die Länge von x zeigt man für zwei beliebige endliche Listen x und y, auf denen es jeweils keine Strukturidentitäten gibt, daß sie genau dann in der sublist-Relation stehen, wenn entweder x die leere Liste ist oder es eine (gegebenenfalls nichtechte) Teilliste von y gibt, deren Elemente mit den Elementen von x identisch sind und die auf y in der gleichen Reihenfolge stehen wie die Elemente von x.

Für das PVPR ergibt sich aus den Klauseln in (5.42) folgendes Bild: Betrachten wir eine beliebige Phrase P. Dominiert P eine Spur, dann ist das PVPR für die Spur dadurch erfüllt, daß P's Permutationsdomäne nicht leer ist, es folglich eine durch die Elemente der Permutationsdomäne bestimmte Liste des phonologischen Materials von P geben muß und die leere Liste Teilliste einer beliebigen Liste ist. Die Existenz einer Permutationsdomäne ist aufgrund der definierenden Klauseln von perm-domain für jede Phrase gesichert.

Für jedes von P dominierte Wort, das keine Spur ist, erzwingt das PVPR, daß für jede Permutationsdomäne, die P zugeordnet werden kann, die phonologische Kette, die in der signphon-Relation zur Permuationsdomäne steht, die Phonologie dieses Wortes als Teil enthält. Da mit dem Phonologieprinzip die Phonologie jeder Phrase aus der Phonologie der Elemente ihrer Permutationsdomäne besteht, wissen wir damit, daß die Phonologie jedes von einer Phrase dominierten Wortes in der Phonologie der Phrase enthalten ist. Da es auf der Phonologieliste keine Strukturidentitäten gibt, ist sie überdies genau einmal darin enthalten.

Üblicherweise leisten Wörter und Phrasen ihren phonologischen Beitrag im Satz als (Teil der) Elemente der Permutationsdomäne ihrer Mutter. Für Parenthesen gibt es eine zweite Möglichkeit: Ein anderes Element im Satz kann ihre Phonologie integrieren und ihnen somit zum phonologischen Ausdruck verhelfen. Damit ist eine Theorie dieser phonologischen Gastgeber gefragt. Im vorliegenden Zusammenhang werden ausschließlich V2-Opener als derartige Gastgeber in Betracht gezogen. Die linguistische Intuition hinter dem hier eingeschlagenen Weg ist die von Reis (1995, S.41 und S.73) formulierte Einsicht, daß die Stelle vor dem Finitum ein besonders großes Integrationspotential für Integrierte V1-Parenthesen aufweist. Mit dem Prinzip (5.43) wird das Integrationspotential von V2-Openern

ausschließlich auf V1-Parenthesen beschränkt. Da es Daten gibt, die darauf hinweisen, daß an dieser Stelle auch anderes phonologisches Material integriert werden kann, hat eine umfassendere Theorie die Frage zu klären, welche Elemente genau in der hier ins Auge gefaßten Weise zwischen Vorfeld und finitem V2-Verb auftreten können. Entsprechend ist dann (5.43) auf weitere empirische Phänomenbereiche zu generalisieren.

(5.43) *Das Prinzip des V2-Integrationspotentials*

V2-Opener können V1-Parenthesen zwischen Vorfeld und finitem Verb integrieren.

$$
\begin{array}{l}
\forall x \\
{:}\left[\textsc{ss}\ \left[\begin{array}{l}\textsc{status}\ \ \textit{complete}\\ \textsc{loc cat head}\ \ \textit{verb2}\end{array}\right]\right]\ \wedge \\
{}^{x}\left[\begin{array}{l}\textit{word}\\ \textsc{phon}\ \ \boxed{1}\\ \textsc{ss loc cat head fr const phon}\ \ \boxed{2}\end{array}\right]\ \wedge \\
\texttt{contains-fr-word(:,}\ x\texttt{)}\ \Longrightarrow \\
\\
\exists b\ \exists c \\
\texttt{append3lists(}\boxed{2}\texttt{,}\ b, c, \boxed{1}\texttt{)}\ \wedge \\
\quad ({}^{b}\left[\langle\rangle\right]\ \ \vee\ \ \exists v\ \exists v' \\
\qquad\qquad \texttt{collect-vips(}v\texttt{,:\ )}\ \wedge \\
\qquad\qquad {}^{v}\left[\textit{nelist}\right]\ \wedge \\
\qquad\qquad \texttt{perm(}v', v\texttt{)}\ \wedge \\
\qquad\qquad \texttt{signphon(}b, v'\texttt{)}\ )\ \ \wedge \\
{}^{c}\left[\langle \textit{phon}\rangle\right]
\end{array}
$$

Das Prinzip des V2-Integrationspotentials macht Aussagen über V2-Sätze und beschränkt die zugelassenen Phonologien ihrer Opener. (5.43) fordert, daß in jedem V2-Satz für den Opener x gilt, daß seine Phonologie besteht aus den aneinandergehängten Phonologien der FRONTED-Konstituente, der Phonologie aller V1-Parenthesen des Satzes (in beliebiger Permutation) oder der leeren Liste, und einem einzigen, nicht näher bestimmten phonologischen Element am Ende. Aufgrund der Lexikoneinträge finiter Verben ergibt sich, daß es sich dabei um den jeweiligen phonologischen Eigenbeitrag des finiten Verbs handelt.

In Übereinstimmung mit der Datenlage macht (5.43) die phonologische Reihenfolge von Integrierten V1-Parenthesen nicht von ihrer relativen (semantischen) Einbettung untereinander abhängig (vgl. (Reis 1995,

S.48–50)) sondern erlaubt generell eine beliebige phonologische Permutation. Andererseits fordert das Prinzip, daß entweder alle oder keine V1-Parenthesen integriert werden. Für den Fall der Nichtintegration hat ein entsprechendes Wortstellungsprinzip dafür zu sorgen, daß auch dann alle V1-Parenthesen ohne ein Dazwischentreten anderer Phonologien realisiert werden.

(5.43) verwendet neben den aus der Linearisierungstheorie bekannten Relationen perm (4.16) und signphon (4.18) die zwei neuen Relationen append3lists und collect-vips. append3lists ist eine vierstellige Relation, deren viertes Argument im wesentlichen eine Liste bestehend aus den aneinandergehängten Listen der ersten drei Argumente ist.

(5.44) *Die Klausel von append3lists*

$$\begin{aligned}&\texttt{append3lists}(x, y, z, r) :=\\ &\qquad \exists a[\texttt{append}(x, y, a)\\ &\qquad \wedge\ \texttt{append}(a, z, r)]\end{aligned}$$

In endlichen Objekten stehen x, y, z und r in der append3lists-Relation genau dann, wenn es eine Liste a gibt, so daß a die aneinander appendierten Listen x und y ist und r die aneinander appendierten Listen a und z. Also enthält die Liste r alle Elemente von x, y und z in der Reihenfolge ihres Auftretens auf diesen Listen.

(5.45) *Die Klauseln von collect-vips*

$$\begin{aligned}&\texttt{collect-vips}(x, y) :=\\ &\qquad \neg\ y\textsc{nonhead-dtr} \sim \textit{verb1adj}\\ &\qquad \wedge\ x \sim \textit{elist}\\ &\texttt{collect-vips}(x, y) :=\\ &\qquad y\textsc{nonhead-dtr} \sim \textit{verb1adj}\\ &\qquad \wedge\ x\textsc{first} \approx y\textsc{nonhead-dtr}\\ &\qquad \wedge\ y\textsc{head-dtr} \sim \textit{phrase}\\ &\qquad \wedge\ \texttt{collect-vips}(x\textsc{rest}, y\textsc{head-dtr})\end{aligned}$$

collect-vips operiert auf beliebigen Zeichen y und ordnet ihnen in x die Kette der entlang der Kopflinie unmittelbar auftetenden Nichtkopftöchter zu, die V1-Parenthesen sind.

Mit der ersten Klausel ist x die leere Kette, wenn auf y NONHEAD-DTR nicht definiert ist (also y zum Beispiel ein Wort ist) oder die Nichtkopftochter keine V1-Parenthese ist. Für Phrasen, deren Nichtkopftochter hingegen eine V1-Parenthese ist, verfügt die zweite Klausel, daß die Nichtkopftochter das erste Element der Kette x ist und der Rest von x rekursiv gefunden wird, indem an der Kopftochter weitergesucht wird. Die Rekursion terminiert mit der ersten Klausel, wenn die als nächste betrachtete Kopftochter ein Wort ist, oder bei der ersten Nichtkopftochter, die keine V1-Parenthese ist.

Wie bereits oben erwähnt wird davon ausgegangen, daß V1-Parenthesen die höchsten Nichtkopftöchter an der Projektion des V2-Verbs sind. Unter dieser Voraussetzung betrachten wir auf der Grundlage der beiden neu definierten Relationen das Prinzip (5.43) noch einmal. Für V2-Sätze mit Opener x gilt aufgrund des Relationsaufrufs von append3lists grundsätzlich, daß die Phonologie von x mit derjenigen seiner FRONTED-Konstituente beginnt, einem noch zu bestimmenden b fortgesetzt wird und mit einem einelementigen c endet. Durch die Lexikoneinträge finiter Verben wissen wir, daß c der phonologische Eigenbeitrag des finiten Verbs ist. Für b gibt es zwei Möglichkeiten. Mit dem ersten Disjunkt im zweiten Konjunkt des Konsequens kann b leer sein. Der Opener ist dann kein phonologischer Gastgeber für fremde Phonologie. Mit dem zweiten Disjunkt kann ein nichtleeres b vorliegen. Voraussetzung dafür ist die Existenz einer nichtleeren Kette v, die aus den von collect-vips im ersten Argument bereitgestellten V1-Parenthesen des betrachteten Satzes besteht. Dann ist b die Phonologie dieser V1-Parenthesen, wobei deren Reihenfolge beliebig ist.

Damit läßt sich (5.43) wie folgt zusammenfassen: In V2-Sätzen können Nichtkopftöchter, die V1-Parenthesen sind, phonologisch in den V2-Opener unmittelbar vor dem Finitum aufgenommen werden. Dabei dürfen die Phonologien von mehreren Parenthesen unabhängig von ihrer strukturellen Position in beliebiger Reihenfolge integriert werden. V2-Opener von Sätzen können nur V1-Parenthesen als Gastgeber dienen und nichts sonst. Da im vorliegenden Grammatikfragment des Deutschen V2-Opener die einzigen Elemente sind, welche die Phonologie anderer in sich aufnehmen können, genügt das Prinzip des V2-Integrationspotentials, um das Auftreten von Gastphonologien in Sätzen zu verhindern, deren Gegenwart durch kein vom Gastgeber unabhängiges Zeichen im Satz gerechtfertigt ist.

Damit ist die Diskussion über die technische Integration der Behandlung von V1-Parenthesen in unsere Annahmen über die Satzstruktur abgeschlossen. Die wesentlichen Bestandteile davon sind die Neudefinition der Relation construct-phon für die Lexikoneinträge finiter Verben, eine zusätzliche Klausel der Relation perm-domain (5.40), das Prinzip der Vollständigen Phonologischen Realisierung (5.41) und das Prinzip des V2-Integrationspotentials (5.43). Zur Akkomodation von V1-Parenthesenverben wurde eine neue Sorte *verb1adj* in die Sortenhierarchie unter *verb*

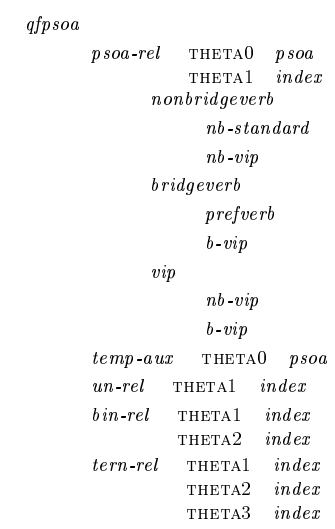
eingefügt, die gleichzeitig eine maximal spezifische Untersorte der *adjunct*-Hierarchie ist und daher über die Eigenschaft eines synsemwertigen MOD-Attributes verfügt. (5.35) hat gezeigt, wie man sich Lexikoneinträge von V1-Parenthesenverben vorstellen kann.

Während dies bereits einen vollständigen Vorschlag zur Behandlung des diskutierten Phänomenbereichs vorstellt, bleibt die unsystematische Behandlung der Lexikoneinträge von V1-Parenthesenverben noch unbefriedigend. Von einer adäquaten Theorie möchte man erwarten, daß sie den zu beobachtenden systematischen Zusammenhang zwischen (den meisten) Brückenverben und V1-Parenthesenverben zum Ausdruck bringt. Ein Beispiel für eine solche generalisierende Zusammenfassung im Lexikon stellt die einheitliche Deskription für die drei Stellungstypen finiter Verbformen dar, welche den Zusammenhang zwischen diesen drei verschiedenen Wortformen formal deutlich macht. Einen vergleichbaren Zusammenhang sollte die Theorie zwischen diesen regulären finiten Verbformen von Brückenverben und ihrer Existenz als V1-Parenthesenverben herstellen. Der Mechanismus, der dafür verwendet werden soll, ist jedoch nicht die deskriptive Unterspezifikation, sondern lexikalische Regeln.

Eine wesentliche Voraussetzung dafür, daß eine Generalisierung der angestrebten Art ausgedrückt werden kann, ist natürlich, daß definierende Eigenschaften von Brückenverben vorhanden sind, aufgrund derer sie von anderen Verbklassen zu unterscheiden sind. In gleicher Weise muß es Kriterien geben, welche innerhalb der Klasse der Brückenverben die Präferenzverben von den übrigen Brückenverben differenzieren, da Präferenzverben sich gerade dadurch auszeichnen, daß sie nicht als V1-Parenthesenverben existieren. In Ermangelung etablierter Unterscheidungskriterien zu diesem Zweck in der HPSG führen wir im folgenden einige erheblich simplifizierende Annahmen ein, welche die erforderlichen Unterscheidungen ermöglichen aber keinesfalls als theoretisch befriedigend anzusehen sind. Es wird angenommen, daß die Unterscheidung zwischen den hier in Frage stehenden Verbklassen vornehmlich semantischer Natur ist. Daher werden ihre Unterschiede in der *content*-Struktur der Verben lokalisiert. Zu diesem Zweck übernehmen wir der Idee nach die einfachen Annahmen zur semantischen Repräsentation von (Pollard und Sag 1994) und modifizieren sie im Hinblick auf die gegenwärtigen Bedürfnisse.

Ansatzpunkt der Unterscheidung zwischen Brückenverben und anderen ist die Hierarchie unter *qfpsoa*, welches der Wert der Angemessenheitsfunktion auf $\langle psoa, \text{NUCLEUS} \rangle$ ist. *psoa* seinerseits ist in (Pollard und Sag 1994) eine Untersorte von *content*.

(5.46) *Die qfpsoa-Hierarchie und ihre Angemessenheitsfunktion*



Im folgenden interessieren wir uns für die Hierarchie unterhalb der Sorte *psoa-rel*, die ihrerseits eine Untersorte von *qfpsoa* ist. *psoa-rels* weisen die zwei Attribute THETA0 mit Wert *psoa* und THETA1 mit Wert *index* auf. Damit sind Verben mit NUCLEUS-Wert *psoa-rel* zweistellige Prädikate, deren eines Argument ein *psoa* und deren anderes Argument ein *index* ist. Bei diesen Verben wird nun zwischen Nichtbrückenverben (*nonbridgeverb*) und Brückenverben (*bridgeverb*) unterschieden. Präferenzverben haben den NUCLEUS-Wert *prefverb*, und die übrigen Brückenverben, zu denen es V1-Parenthesenformen gibt, haben den NUCLEUS-Wert *b-vip*. Neben Brückenverben, die in V1-Parenthesenkonstruktionen auftreten, gibt es Nichtbrückenverben wie *fragen*, die als V1-Parenthesenverben erscheinen können. Sie tragen den NUCLEUS-Wert *nb-vip*. Da der maximal spezifische NUCLEUS-Wert *b-vip* oder *nb-vip* potentieller V1-Parenthesenverben zugleich direkte Untersorte von *vip* ist, kann mittels *vip* über sie generalisiert werden.

Beispiele für Brückenverben, die keine Präferenzverben sind, sind *glauben, sagen, denken* und *behaupten*. *vorziehen* ist ein Präferenzverb und *bezweifeln* ein Nichtbrückenverb, das nicht als V1-Parenthesenverb auftritt und daher in obiger Klassifikation den NUCLEUS-Wert *nb-standard* erhält.

Für temporale Hilfsverben nehmen wir die traditionelle HPSG-Analyse der Argumentanhebung an. Sie tragen den NUCLEUS-Wert *temp-aux*. Ihr semantisches *psoa* Argument unter THETA0 ist der *psoa* des Arguments, das sie syntaktisch neben den angehobenen Argumenten subkategorisieren und dessen Argumente sie syntaktisch erben. Die übrigen in (5.46) angegebenen Untersorten von *qfpsoa* dienen lediglich der Illustration und zeigen, wie ein- bis dreistellige Prädikate mit *index*-wertigen Argumentstellen in die angegebene Skizze der *qfpsoa*-Hierarchie eingearbeitet werden können.

Die Grundidee einer lexikalischen Regel zur Beschreibung von V1-Parenthesenverben ist nun die, zu sagen, daß es für jedes finite Verb mit einem NUCLEUS-Wert, der Untersorte von *vip* ist, Wörter entsprechender Form gibt, die allerdings V1-Parenthesenverben sind. Für diese erste Formulierung stellen jedoch die Daten in (5.47) sofort ein Problem dar:

(5.47) a. Wer sagt Anna, ist zur Party gekommen?
b. Wer hat Anna gesagt, ist zur Party gekommen?
c. Wer hat Anna gesagt gehabt, ist zur Party gekommen?

Offensichtlich ist es so, daß nicht nur Wörter aus der *vip*-Klasse als V1-Parenthesenverben auftreten können, sondern dies ist auch für temporale Hilfsverben wie *haben* der Fall. Allerdings belegt der Test mit dem Nichtbrückenverb *bezweifeln* in (5.48), daß die Fähigkeit von temporalen Hilfsverben, als V1-Parenthesenverben zu fungieren, vom Hauptverb des jeweiligen Satzes abhängt.

(5.48) a. *Wer bezweifelt Anna, ist zur Party gekommen?
b. *Wer hat Anna bezweifelt, ist zur Party gekommen?

Bettet *hat* das Brückenverb *gesagt* ein, zu dem es eine V1-Parenthesenform gibt, kann es selbst V1-Parenthesenverb sein; bettet es hingegen das Nichtbrückenverb *bezweifelt* ein, zu dem es kein V1-Parenthesenverb gibt, kann auch *hat* selbst kein V1-Parenthesenverb sein. Ein Ziel der Formulierung der lexikalischen Regel muß es sein, diese Generalisierung mit zu erfassen.

Unter lexikalischen Regeln wird in der Literatur zumeist ein Mechanismus verstanden, der Beschreibungen von Wörtern zueinander in Beziehung setzt. (Meurers 1995) schlägt alternativ lexikalische Regeln auf Deskriptionsebene vor, die anstelle von Beschreibungen die Wörter im Modell zueinander in Beziehung setzen. Meurers (1995) formalisiert diesen Vorschlag in SRL und argumentiert, daß unter dieser Herangehensweise die tatsächlich in der linguistischen Literatur verwendeten lexikalischen

Regeln umgesetzt und sinnvoll interpretiert werden können. Insbesondere stellt er eine Notation vor, welche der in der Literatur gebräuchlichen weitgehend entspricht und die im folgenden in einer Erweiterung um Relationen verwendet wird. Da eine Formalisierung der Behandlung von lexikalischen Regeln auf der Metaebene noch immer aussteht, wird im folgenden unter lexikalischen Regeln ihre spezifische Konkretisierung in (Meurers 1995) verstanden und angenommen, daß die dort vorgeschlagene technische Behandlung von SRL auf RSRL erweitert werden kann.

Wir nennen den Teil der Beschreibung einer lexikalischen Regel vor dem Pfeil die Inputbeschreibung und den Teil nach dem Pfeil die Outputbeschreibung. Hinsichtlich der jeweils beschriebenen Wörter sprechen wir vom Input beziehungsweise Output der lexikalischen Regel.

(5.49) *Erste lexikalische Regel für V1-Parenthesenverben*

$$\left[\begin{array}{l} word \\ \text{SS LOC} \left[\begin{array}{l} \text{CAT} \left[\begin{array}{ll} \text{HEAD} & verb1 \\ \text{VAL COMPS} & \boxed{1} \end{array}\right] \\ \text{CONT} \boxed{3} \end{array}\right] \end{array}\right] \wedge \texttt{vipverb}(\boxed{3}, \boxed{4}) \wedge$$

$$\exists_x \left[\begin{array}{l} synsem \\ \text{STATUS} \quad complete \\ \text{LOC} \quad \left[\begin{array}{l} \text{CAT MARKING} \quad daß \\ \text{CONT} \ \boxed{4}\ psoa \end{array}\right] \end{array}\right] \wedge \texttt{take-out}(\boxed{1}, x, \boxed{2})$$

$$\longmapsto$$

$$\left[\begin{array}{l} word \\ \text{SS LOC CAT} \left[\begin{array}{ll} \text{HEAD} & \left[\begin{array}{ll} verb1adj & \\ \text{MOD LOC} & \left[\begin{array}{ll} \text{CAT HEAD} & verb2 \\ \text{CONT} & \boxed{4} \end{array}\right] \end{array}\right] \\ \text{VAL COMPS} & \boxed{2} \end{array}\right] \end{array}\right]$$

Die lexikalische Regel (5.49)[25] sagt, daß zu jedem V1-Verb, das entweder ein *vip*-Verb oder ein temporales Hilfsverb ist, das ein solches semantisch einbettet, ein Verberstparenthesenverb einer bestimmten Gestalt existiert. (5.49) verwendet bei der Formalisierung der Regel die aus

[25] Dem aufmerksamen Leser ist sicher aufgefallen, daß in der lexikalischen Regel kein AVM durch : als Beschreibung desjenigen Objekts ausgezeichnet ist, bezüglich dessen die Quantoren der Beschreibung interpretiert werden. Aufgrund der Interpretation der abkürzenden linguistischen Notation für lexikalische Regeln im Ansatz von (Meurers 1995), deren sich (5.49) in intuitiver Erweiterung von SRL auf RSRL bedient, sind Quantoren bezüglich des von der lexikalischen Regel beschriebenen *lex_rule*-Objektes zu interpretieren.

der Linearisierungstheorie bekannte Relation take-out (4.17) und die neue zweistellige Relation vipverb. Letztere ist wie folgt definiert:

(5.50) *Die Klauseln von vipverb*

$$\begin{aligned}&\texttt{vipverb}(x, y) :=\\ &\qquad x\textsc{nucleus} \sim \textit{vip}\\ &\qquad \wedge\ x\textsc{nucleus theta0} \approx y\\ &\texttt{vipverb}(x, y) :=\\ &\qquad x\textsc{nucleus} \sim \textit{temp-aux}\\ &\qquad \wedge\ \texttt{vipverb}(x\textsc{nucleus theta0}, y)\end{aligned}$$

Sowohl das erste als auch das zweite Argument von vipverb ist von der Sorte *psoa*. vipverb gilt, wenn der NUCLEUS-Wert von x *vip* ist und sein THETA0-Wert mit y identisch ist. Im rekursiven Fall ist der NUCLEUS-Wert von x *temp-aux*, und die Relation vipverb gilt zwischen dem NUCLEUS THETA0-Wert von x und y. Die Rekursion terminiert, sobald mit der ersten Klausel ein x mit NUCLEUS *vip* gefunden wird, dessen THETA0 mit y identisch ist.

Unter Rückgriff auf vipverb sagt (5.49), daß es zu jedem V1-Verb, das entweder ein *vip*-Verb ist oder ein temporales Hilfsverb, welches ein *vip*-Verb semantisch einbettet, und das mittels seiner COMPS-Liste einen daß-Komplementsatz subkategorisiert, der semantisches Argument des vorliegenden oder des semantisch eingebetteten *vip*-Verbs ist, ein V1-Parenthesenverb gibt, dessen COMPS-Liste der des in der Inputbeschreibung der Regel beschriebenen Wortes gleicht bis auf die Tatsache, daß es das daß-Komplement nicht subkategorisiert, und der MOD-Wert des V1-Parenthesenverbs selegiert eine V2-Projektion, deren CONTENT im V1-Parenthesenverb an die Stelle tritt, die der CONTENT des daß-Komplements beim V1-Verb des Inputs einnimmt. Das heißt, bei Vorliegen eines *vip*-Verbs ist der CONTENT der V2-Projektion das THETA0-Argument des *vip*-Verbs; bei Vorliegen eines temporalen Hilfsverbs ist der CONTENT der V2-Projektion das THETA0-Argument des semantisch eingebetteten *vip*-Verbs. Bis auf die erwähnten Unterschiede gleichen sich das Input- und Outputwort.

Tabelle (5.51) enthält eine Übersicht über die Subkategorisierungseigenschaften von vier *vip*-Verben, die sich in Bezug auf ihre möglichen satzwertigen Komplemente unterscheiden. Die einzelnen Spalten geben an, ob das jeweilige Verb daß-VL-, V2-, ob-VL- und w-VL-Sätze als Komplemente nimmt oder nicht.

(5.51) *Die Subkategorisierungseigenschaften einiger Verben*

Verb	daß	V2	ob	w-VL
sagen	+	+	+	+
glauben	+	+	-	-
wünschen	+	-	-	-
fragen	-	-	+	+

Zum einen ist zu bemerken, daß unter den angegebenen Verben nur *glauben* und *sagen* Verbzweitsätze als ihr propositionales Argument zulassen. Ausnahmslos alle V1-Parenthesenverben verlangen hingegen, daß ihr propositionales Argument ein V2-Satz ist. Außerdem erlauben *glauben* und *wünschen*, die normalerweise keine ob-Sätze oder w-VL-Sätze subkategorisieren, in ihrer Form als V1-Parenthesenverben w-V2-Sätze:

(5.52) a. Wen glaubst Du, habe ich gestern in der Tübinger U-Bahn getroffen?
b. *Du glaubst, wer das getan hat.
c. Welche Dampfmaschine wünschst Du, soll repariert werden?
d. *Du wünschst, welche Dampfmaschine repariert werden soll.

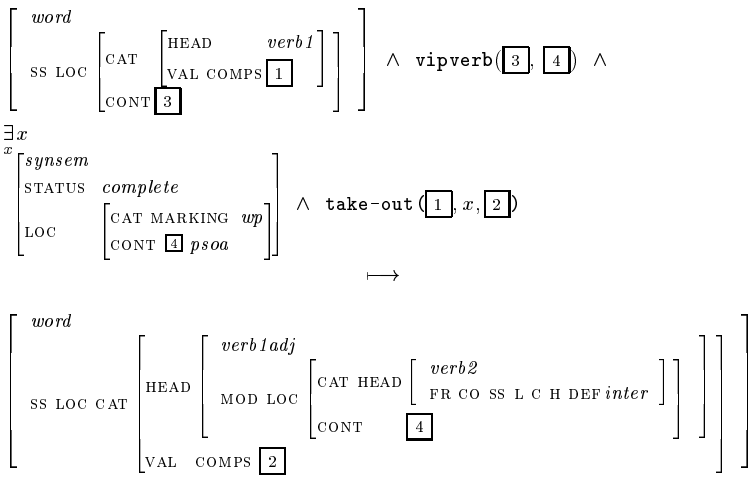
Die lexikalischen Regel in (5.49) beschreibt diese Daten korrekt, da die Outputbeschreibung dem Vorfeldbesetzer des modifizierten V2-Satzes keine Beschränkung auferlegt. Da alle Verben, die nichtinterrogative Komplemente subkategorisieren können, daß-Sätze zulassen, werden alle in Frage kommenden Verben von der lexikalischen Regel tatsächlich erfaßt.

Anders sieht es mit solchen Verben aus, die wie *fragen* ausschließlich Interrogativsätze als satzwertige Komplemente zulassen. Wie Tabelle (5.51) zu entnehmen ist, bettet *fragen* lediglich ob- und w-VL-Sätze ein. Diese Beschränkung auf Interrogativsätze behält es auch als V1-Parenthesenverb bei:

(5.53) a. Wen fragt Detmar, sollte man zur Vorführung einladen.
b. *Alle wichtigen Leute fragt Detmar, sollte man zur Vorführung einladen.

Da *fragen* jedoch keine daß-Komplemente zuläßt, gibt es kein Wort *fragen*, das von der Inputbeschreibung der lexikalischen Regel (5.49) beschrieben wird. Damit ist *fragen* auch nicht Gegenstand dieser lexikalischen Regel. Während somit die Grammatik mit Regel (5.49) zwar nicht übergeneriert, lizenziert sie andererseits keine V1-Parenthesenform von *fragen* wie in (5.53a). Um diese zu erhalten, wird eine zweite lexikalische Regel benötigt, die sich in der Inputbeschreibung von der ersten in einem wichtigen Punkt unterscheidet und hinsichtlich ihres Outputs restriktiver ist:

(5.54) *Zweite lexikalische Regel für V1-Parenthesenverben*



Mit (5.54) existiert zu jedem V1-Verb, das entweder selbst *vip*-Verb oder aber ein temporales Hilfsverb ist, das ein solches semantisch einbettet, und das auf seiner COMPS-Liste ein Interrogativsatzkomplement subkategorisiert, welches semantisches Argument des unmittelbar vorliegenden oder semantisch eingebetteten Verbs ist, ein V1-Parenthesenverb, dessen COMPS-Liste der des Inputs bis auf die Abwesenheit des Interrogativsatzkomplements gleicht, und das mit seinem MOD-Attribut einen w-V2-Satz selegiert, dessen CONTENT im V1-Parenthesenverb an die Stelle des CONTENTS des Interrogativsatzes im V1-Verb tritt. Im übrigen gleichen sich Input und Output strukturell.

Als Konsequenz der zweiten lexikalischen Regel lizenziert die Grammatik nun auch das für Beispiel (5.53a) benötigte V1-Parenthesenverb *fragen*, das nur in einen w-V2-Satz integriert werden darf. Daneben stellt (5.54) einen systematischen Zusammenhang her zwischen den Interrogativsätze einbettenden V1-Formen des Verbs *sagen* aus Tabelle (5.51) und seinen entsprechenden w-V2-Sätze modifizierenden V1-Parenthesenformen. Man beachte, daß diese zusätzlich ausgedrückten Zusammenhänge für Verbformen, die bereits durch die erste lexikalische Regel lizenziert werden, an den durch die Grammatik denotierten Wörtern substantiell nichts ändern. Es ist darüber hinaus bemerkenswert, daß die lexikalischen Regeln in der gegebenen Form unmittelbar die richtigen Vorhersagen für

unter temporale Hilfsverben eingebettete *vip*-Verben machen, die nur Interrogativsatzkomplemente zulassen.

(5.55) a. Wen hast du gefragt, sollten wir t einladen?
b. *Den Guru selbst hast du gefragt, sollten wir t einladen.

Das finite V1-Parenthesenverb *hast*, das in (5.55) verwendet wird, muß einen w-V2-Satz modifizieren, weil das semantisch von ihm eingebettete Verb *gefragt* ausschließlich Interrogativsätze als Komplement zuläßt. Daher kann nur die zweite lexikalische Regel die vorliegende Form von *hast* lizenzieren, und die in (5.55b) illustrierte Übergenerierung ist ausgeschlossen.

In diesem Abschnitt haben wir uns der von Reis (1995) vertretenen Auffassung angeschlossen, daß es sich bei den Konstruktionen, die in der generativen Linguistik als lange Extraktion aus V2-Sätzen analysiert werden, in Wirklichkeit um Fälle von Integrierten Verberstparenthesen handelt. Es wurde gezeigt, wie diese Auffassung mit dem Konzept einer linearisierungsbasierten HPSG-Analyse der deutschen Satzstruktur vereinbart werden kann. Hierzu wurde ein erster Vorschlag für die strukturelle Integration parenthetischer Konstruktionen gemacht, der sich eng an die in (Reis 1995) vorgestellten empirischen Beobachtungen zu den Eigenschaften von Integrierten V1-Parenthesen anlehnt. Wie in den vorangegangenen Kapiteln wurde die in diesem Abschnitt vorgestellte Analyse vollständig in RSRL formalisiert. Indessen geht diese Formalisierung nicht über eine erste Annäherung an die mit Integrierten V1-Parenthesen verbundenen komplexen Fragestellungen hinaus. Klärungsbedürftig sind vor allem die Bedingungen, denen die mögliche Stellungen von Integrierten V1-Parenthesen unterliegen; die Frage einer angemessenen semantischen Repräsentation wurde erst gar nicht angesprochen. Es ist nicht klar, ob der eingeschlagene Weg zur Behandlung dieses speziellen Parenthesentyps auf andere Arten von Parenthesen erfolgreich ausgedehnt werden kann.

Die Parenthesenanalyse sogenannter Extraktion aus V2-Sätzen scheint nicht nur empirisch adäquater, sie bringt auch für die folgende Theorie der Inseldomänen nichtlokaler Abhängigkeiten entscheidende Vorteile. Angesichts der strikten Trennung von Adjunktbewegung und Komplementbewegung wird es möglich sein, V2-Sätze als strikte Inseln für jede Art echter syntaktischer Bewegung zu definieren. Auf Zwischenspuren bei Bewegung kann grundsätzlich verzichtet werden. Obwohl unabhängig von der eigentlichen Bewegungstheorie unserer Grammatik, ist die V1-Parenthesenanalyse von entscheidendem Einfluß auf deren Gestalt.

5.4. Inseldomänen nichtlokaler Abhängigkeiten. Im vorliegenden Abschnitt werden zwei Prinzipien über globale Extraktionsdomänen vorgestellt, welche einen Ausschnitt der Theorie nichtlokaler Abhängigkeiten des Deutschen formal explizieren. Die einschlägigen Extraktionsdomänen werden auf der Grundlage eines Barrierenbegriffs für zwei verschiedene Arten der Bewegung, Topikalisierung und w-Bewegung, formalisiert. Dabei zeigt sich, daß in der gewählten Merkmalsarchitektur die zur Spezifikation der Barrieren erforderlichen Eigenschaften sprachlicher Zeichen bereits vollständig repräsentiert sind und die Bewegungstheorie und ihre Beschränkungen keine zusätzlichen Merkmale oder Sorten notwendig machen.

Der inhaltliche Schwerpunkt liegt auf den Extraktionsdaten bei finiten Satzkomplementen von Brückenverben. Das Hauptziel der Analyse ist es, die Barriereneigenschaften von subkategorisierten V2-, daß-VL- sowie ob- und w-VL-Sätzen unter Brückenverben zu beschreiben. Wie weiter oben bereits motiviert wurde (siehe (5.10)), wird dabei nur Komplementextraktion als syntaktische Bewegung aufgefaßt. Den auch im Deutschen zu beobachtenden Komplement-Adjunkt-Asymmetrien wird dadurch Rechnung getragen, daß für Adjunkte eine spurenlose Extraktionstheorie angenommen wird, die (Sailer 1997) ausführt. Das für das Verhalten subkategorisierter Sätze unter Brückenverben entworfene Bild wird abschließend auf Nichtbrückenverben ausgedehnt.

Hinsichtlich Adjunkten mit adverbialem Status schließen wir uns der Auffassung von (Lutz 1993) an, wo argumentiert wird, daß diese im Deutschen wie finite Adverbialsätze generell Inseln für lange Topikalisierung und w-Bewegung sind. Extraktion aus Nominal- und Präpositionalphrasen wird hingegen vollständig ignoriert. Um ein in sich stimmiges Fragment zu spezifizieren, wird vereinfachend vorausgesetzt, daß es nur einfache Nominal- und Präpositionalphrasen enthält, aus denen nicht extrahiert werden kann und für deren Inselcharakter Sorge getragen wurde.[26]

Traditionell ist der Lexikoneintrag von Spuren in HPSG nahezu völlig unrestringiert. Wie in (5.15) gezeigt, bestehen die Bedingungen im wesentlichen darin, daß der LOC-Wert der Spur in die SLASH-Menge übernommen wird und die Phonologie der Spur leer ist. Daneben werden lediglich noch weitere NONLOCAL-Werte fixiert. Nach der Elimination der *nonlocal*-Struktur aus der Grammatik bleibt als einzige Restriktion für eine Spur

[26] Für einen Überblick über die tatsächliche Datenlage vgl. (Pafel 1993) und (Trissler 1993). Wie die Formalisierung eines simplistischen strikten Inselstatus von Nominalphrasen und Präpositionalphrasen umgesetzt werden kann, ergibt sich unmittelbar aus den weiter unten vorgeschlagenen Barrierenbegriffen.

die Forderung nach einer leeren Phonologie erhalten. (5.56) wiederholt den Lexikoneintrag der Spur.

(5.56) *Der Lexikoneintrag für Spuren*

$$\begin{bmatrix} \textit{word} \\ \text{PHON} \langle\ \rangle \end{bmatrix}$$

Damit kann zunächst prinzipiell alles als Spur auftreten. (Pollard und Sag 1994, S.400) schränkt diese Allgemeinheit für das Englische drastisch durch das Trace Principle ein, das verlangt, daß der SYNSEM-Wert einer jeden Spur auch nichtinitiales Element der SUBCAT-Liste eines Wortes mit einem HEAD-Wert einer Untersorte von *substantive* ist. Diese Option scheidet in dieser Form im vorliegenden Kontext schon deshalb aus, weil die Extraktion von Subjekten einheitlich mit allen anderen Formen der Komplementextraktion als lange Bewegung betrachtet wird, die eine Binder-Spurbeziehung induziert. Die Last der Formulierung korrekter Beschränkungen fällt im gegenwärtigen Ansatz der Theorie der Extraktionsdomänen zu. Darüber hinaus ist die Distribution leerer Elemente in sprachlichen Strukturen Gegenstand dreier weiterer grundlegender Annahmen.

Die erste ist in der Signatur enthalten und betrifft insofern nicht Spuren im Sinne des Lexikoneintrags (5.56), als sie nicht mit Wörtern sondern mit Phrasen zu tun hat. Letztere haben aufgrund der Angemessenheitsfunktion immer den PHON-Wert *nelist* und können somit keine leeren Elemente sein (vgl. (5.20)). Da Spur und Binder in der HPSG traditionell ihren LOCAL-Wert teilen, der keine Information über den Wort- oder Phrasenstatus des Zeichens enthält, ist die Kombination eines phrasalen Binders mit einer Spur, die ein Wort ist, in der HPSG üblich und besteht für phrasale leere Elemente kein unmittelbarer Bedarf. Aus der ontologischen Beschränkung des Auftretens von leeren Elementen auf die Wortebene folgt mit der Konstruktion der Phonologie der Mutter aus den phonologischen Beiträgen ihrer Töchter in durchweg binär verzweigenden Baumstrukturen unmittelbar, daß leere Elemente keine Schwestern voneinander sein können.

Die zweite Annahme über Spuren betrifft Opener und besagt, daß Opener immer phonologisch sichtbar sind.

(5.57) *Das Sichtbarkeitsprinzip für Opener*

Opener müssen phonologisch sichtbar sein.

$$\begin{bmatrix} \textit{word} \\ \text{SYNSEM LOC CAT HEAD FRONTED } \textit{locality} \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{PHON } \textit{nelist} \end{bmatrix}$$

Das linksperiphere Element in deutschen Sätzen spielt eine zentrale Rolle für die Markierung des vorliegenden Satztyps und bei der Bestimmung seines Illokutionstyps. Insofern scheint die Forderung nach seiner phonologischen Sichtbarkeit durchaus den natürlichen Erwartungen an die Grammatik zu entsprechen.

Drittens und letztens wird gefordert, daß zwei Spuren innerhalb eines sprachlichen Zeichens nicht miteinander identisch sein dürfen.

(5.58) *Das Spurprinzip*

Alle Spuren eines linguistischen Zeichens sind voneinander verschieden.

$$\forall t_1\ \forall t_2\ \exists p\ [:\ \sim \mathit{sign} \wedge t_1\textsc{phon} \sim \mathit{elist} \wedge t_2\textsc{phon} \sim \mathit{elist} \wedge p \sim \mathit{phrase} \wedge ((\texttt{dominate}(p,t_1) \wedge \neg\ \texttt{dominate}(p,t_2)) \vee (\neg\ \texttt{dominate}(p,t_1) \wedge \texttt{dominate}(p,t_2)))] \longrightarrow \neg\ t_1 \approx t_2$$

Für alle Zeichen gilt, daß wenn es zu darin enthaltenen Spuren t_1 und t_2 eine Phrase p gibt, die eine der Spuren dominiert aber nicht die andere, Spur t_1 und Spur t_2 voneinander verschiedene Zeichen sein müssen. Da die Signatur nur binär verzweigende Bäume zuläßt, die Phonologie von Phrasen aufgrund der Signatur immer nichtleer ist und die Phonologie der Mutter immer aus den phonologischen Beiträgen ihrer Töchter besteht, also zwei Geschwister im Baum niemals beide Spuren sein können, folgt aus der Beschreibung in (5.58), daß zwei Spuren einer Struktur immer voneinander verschieden sind.

Man beachte, daß (5.58) nicht ausschließt, daß zwei Spuren den gleichen LOCAL-Wert aufweisen. Mit anderen Worten schließt das Spurprinzip nicht a priori die Existenz von Parasitic-Gap-Konstruktionen aus. Was das Prinzip hingegen ausschließt, ist die Existenz von Baumstrukturen, in denen eine Spur gleichzeitig Tochter von zwei oder mehr verschiedenen Müttern ist. Diese Annahme entspricht den üblichen Vorstellungen von der Struktur linguistischer Repräsentationen und ist daher nicht als Besonderheit des hier verfolgten Ansatzes zu werten. Sie ist insonderheit unabhängig von der Gegenwart oder Abwesenheit einer *nonlocal*-Struktur in linguistischen Zeichen.

Es ist bemerkenswert, daß ein Prinzip wie (5.58), das die Verschiedenheit des in der Baumstruktur als Blätter auftretenden lexikalischen Materials erzwingt, nur für Spuren erforderlich ist und für sämtliche Nichtspuren aus unabhängigen Gründen folgt. In der Wortstellungstheorie wird gefordert, daß es auf der Phonologieliste von Zeichen keine Strukturidentitäten zwischen den Listenelementen gibt. Bei den Listenelementen handelt es sich aber um die Phonologie aller Wörter, die von der betreffenden Phrase dominiert werden. Also haben alle Wörter mit nichtleerer Phonologie verschiedene PHON-Werte und können damit nicht identisch sein. In gewisser Weise ist die Notwendigkeit von (5.58) somit eine Folge der Tatsache, daß Spuren nicht in die lineare Ordnung sprachlicher Zeichen eingehen.

Ausgangspunkt für die Unterscheidung von zwei Arten von Extraktionsdomänen im Deutschen sind Beobachtungen, die dahin gehen, daß w-Inseln für die Topikalisierung bestimmter Konstituententypen durchlässig sind, wohingegen sie für w-Bewegung grundsätzlich opak sind (Daten nach (d'Avis 1993)):

(5.59) a. ?Radios weiß ich nicht, wer repariert.
b. ?Schlangen weiß ich nicht, ob Otto züchtet.
c. *Welchen Mann weißt du nicht, wer gesehen hat.
d. *Wen weißt du nicht, wie Brutus erstochen hat.
e. *Wen weißt du nicht, ob Brutus erstochen hat.

Unter der Voraussetzung, daß *wissen* mit w-Komplementen Brückeneigenschaften aufweist,[27] liegt in (5.59a–b) Topikalisierung aus dem w-VL- beziehungsweise ob-VL-Komplement eines Brückenverbs vor. Dagegen zeigen (5.59c–d), daß unabhängig davon, ob im subkategorisierten w-VL-Satz eine w-Konstituente mit Adverbial- oder Komplementstatus in der linken Peripherie steht, w-Bewegung aus w-VL-Sätzen nicht erlaubt ist. Das entsprechende Datum für den ob-VL-Satz (5.59e) zeigt, daß generell subkategorisierte Sätze mit einem MARKING-Wert, der Untersorte von w ist (3.22), w-Inseln darstellen. Diese Daten etablieren eine Asymmetrie zwischen langer Topikalisierung und langer w-Bewegung, welche das Postulat motiviert, daß es im Deutschen zwischen Barrieren für Topikalisierung und Barrieren für w-Bewegung zu unterscheiden gilt.

In Abschnitt 5.4.1 werden auf Topikalisierungsbarrieren beruhende Topikalisierungsdomänen betrachtet und ein Prinzip formuliert, das Topikalisierung auf Topikalisierungsdomänen beschränkt. Abschnitt 5.4.2 wendet sich w-Bewegungsbarrieren und den mit ihnen assoziierten w-Bewegungsdomänen zu und formuliert ein analoges Prinzip für diese Bereiche. Lange Bewegung innerhalb beider Domänen zeichnet sich dadurch aus, daß sie keine Zwischenspuren involviert. Auf der technischen Seite illustriert der hier eingeschlagene Weg, daß eine relationale Herangehensweise an lange Abhängigkeiten in HPSG in direkter Weise von Barrierenbegriffen

[27] Zur Frage des Brückenverbstatus von *wissen* und der Problematik der Annahme einer lexikalisch determinierten Brückeneigenschaft vgl. die Diskussion in (d'Avis 1993, S.75–76).

Gebrauch machen und zwischen verschiedenen Arten von Barrieren differenzieren kann. In der Struktur linguistischer Zeichen affiziert Bewegung ausschließlich die Position des Binders und der Spur und hat keine anderen strukturellen Effekte. Daher kann auf Prinzipien, welche eine lokale Merkmalsperkolation an jeder Phrase steuern, vollständig verzichtet werden.

5.4.1. *Topikalisierungsdomänen.* Die Daten in (5.60) motivieren beispielhaft, welche sprachlichen Zeichen als strikte Barrieren für Topikalisierung in Frage kommen beziehungsweise nicht in Frage kommen.

(5.60) a. Den Hund$_1$ sagt Maria, daß Karl t_1 füttert.
b. *Den Hund$_2$ sagt Maria, Karl$_1$ füttert t_1 t_2.
c. *Den Hund$_1$ sagt Maria, t_1 füttert Karl t'_1.
d. *Den Hund$_1$ macht Maria die Tür zu, bevor Karl t_1 füttert.
e. *Daß Karl jeden Tag t füttert!
f. *Karl$_1$ füttert jeden Tag t_1 t_2.

Aus daß-VL-Sätzen kann topikalisiert werden (5.60a). Damit verhalten sie sich hinsichtlich der Topikalisierung im allgemeinen wie ob- und w-VL-Sätze, für die ebenfalls Fälle langer Topikalisierung belegt sind (5.59a–b). Da im folgenden ein strikter Barrierenbegriff für Topikalisierung definiert wird, müssen ungrammatische Instanzen langer Topikalisierung aus den jeweiligen Sätzen auf andere Eigenschaften der Konstruktionen zurückgeführt werden und können nicht aus einer generellen Unmöglichkeit langer Topikalisierung aus diesen Sätzen abgeleitet werden. Vor allem inhärente Eigenschaften der bewegten Konstituente wie zum Beispiel Definitheit haben Einfluß auf ihre Extrahierbarkeit. Abgesehen von der Frage, ob es sich bei einer Konstituente um ein w-Element oder um kein w-Element handelt, ignoriert die hier vorgestellte Theorie der Bewegungsbarrieren inhärente Eigenschaften der bewegten Elemente, die eine Erweiterung der Grammatik auf größere Datenbereiche zu berücksichtigen hat. Die Idee dieser Vorgehensweise ist die, daß weitere Eigenschaften der bewegten Zeichen nicht Gegenstand der Formulierung von Barrieren sind, sondern die Klasse der in den jeweiligen Domänen tatsächlich bewegbaren Elemente einschränken. Constraints, die dies zum Ausdruck bringen, sind gesondert zu formulieren.

V2-Sätze sind strikte Inseln für Topikalisierung (5.60b). Die V1-Parenthesenanalyse in Abschnitt 5.3 erlaubt es, diese Restriktion uneingeschränkt geltend zu machen. Insbesondere gibt es keine Extraktion aus

V2-Sätzen mit einer Zwischenspur im Vorfeld, wie in (5.60c) angedeutet. Der grammatische Satz mit entsprechender Phonologie enthält die Integrierte V1-Parenthese *sagt Maria*.

(5.60d) illustriert den Inselcharakter von Adjunkten anhand eines finiten temporalen Adverbialsatzes. Wie oben erwähnt wird davon ausgegangen, daß echte adverbiale Adjunkte im Deutschen generell Inseln für Bewegung sind. Uneingebettete Sätze (5.60e–f) sind in dem Sinne Barrieren für Topikalisierung, als daß sie keine unabgebundenen Spuren enthalten dürfen.

Die beobachteten Barrieren für lange Bewegung, die sich als Knoten der syntaktischen Baumstruktur auffassen lassen, unterhalb derer es keine unabgebundenen Topikalisierungsspuren geben darf, werden in (5.61) in einer Relation zusammengefaßt. Die neue Relation top-barrier ermöglicht es, Generalisierungen über Topikalisierungsbarrieren kompakt zu formalisieren.

(5.61) *Die Klausel von top-barrier*

$$\texttt{top-barrier}(x) :=$$
$$^{x}\left[\begin{array}{l}\mathit{phrase}\\ \textsc{synsem}\ \left[\begin{array}{ll}\textsc{status} & \mathit{complete}\\ \textsc{loc cat head} & \mathit{verb2}\end{array}\right]\end{array}\right] \vee$$
$$^{x}\left[\textsc{synsem}\ \left[\begin{array}{ll}\textsc{status} & \mathit{complete}\\ \textsc{loc cat head} & \mathit{adjunct}\end{array}\right]\right] \vee$$
$$^{x}\left[\mathit{u\text{-}sign}\right] \vee$$
$$\exists y\ (y\textsc{const} \approx x)$$

Nach dem ersten Disjunkt der Klausel von top-barrier ist x eine Topikalisierungsbarriere, wenn x ein V2-Satz ist. Satzwertige V2-Wörter könnten nur Spuren sein, die bereits durch das Sichtbarkeitprinzip für Opener (5.57) ausgeschlossen werden. Das zweite Disjunkt in (5.61) nennt syntaktisch vollständige Adjunkte als Barrieren für Topikalisierung. Damit werden automatisch auch V1-Parenthesen zu Barrieren für Topikalisierung. Die Tatsache, daß von Adjunkten als Topikalisierungsbarrieren kein phrasaler Status gefordert wird, wird mit dem Topikalisierungsdomänenprinzip (5.62) unmittelbar dazu führen, daß Adjunktspuren generell nicht auftreten können. Das gleiche gilt auch für die in den letzten beiden Disjunkten von (5.61) genannten Topikalisierungsbarrieren, uneingebettete Äußerungen und Vorfeldkonstituenten, das heißt, sie können selbst keine

Spuren sein. Vorfeldkonstituenten x in einem linguistischen Zeichen werden dadurch identifiziert, daß es ein weiteres Objekt y in dem Zeichen geben muß, dessen CONST-Attribut zu x führt.

Der Umstand, daß nur die FRONTED-Konstituente von Openern als Landeposition für lange Bewegung verfügbar ist, Topikalisierung außerdem auf die FRONTED-Konstituente von V2-Verben beschränkt ist und V2-Sätze Barrieren für Topikalisierung sind, führt zu der Vorhersage, daß über topikalisiert Elemente nicht hinaustopikalisiert werden kann (siehe (5.60b)). Es ist naheliegend, eine Topikalisierungsdomäne als die Menge der zu einer Topikalisierungsbarriere T gehörigen Wörter anzusehen, die T dominiert, aber von keiner Topikalisierungsbarriere dominiert werden, die nicht selbst T dominiert. Das heißt, die Topikalisierungsdomäne einer Barriere T ist die Menge der Wörter, die T dominiert, wobei keine andere Topikalisierungsbarriere zwischen T und den Wörtern interveniert. Dominiert zum Beispiel T keine von T verschiedene Topikalisierungsbarriere, dann bilden alle von T dominierten Wörter eine einzige Topikalisierungsdomäne. Ansonsten ist die Menge derjenigen Wörter von der Menge der insgesamt dominierten Wörter abzuziehen, die von intervenierenden Topikalisierungsbarrieren dominiert werden.

Für das folgende Prinzip (5.62) ist es entscheidend, daß es in der HPSG durch die Strukturgleichheit der LOCAL-Werte von Antezedens und Spur möglich ist, alleine durch Betrachtung der Spur zwischen w-Spuren und nicht-w-Spuren zu unterscheiden, da die w-Eigenschaft explizit im Kopfmerkmal DEF durch den syntaktischen Wert *inter(rogativ)* angegeben wird (vgl. (Winhart 1997)). In der formalen Präzisierung des Prinzips wurde auf eine notwendig unübersichtliche vollständige Klammerung des Satzes verzichtet und stattdessen versucht, die Skopusverhältnisse und Klammerungsstruktur durch Einrückung zu verdeutlichen.

(5.62) *Das Topikalisierungsdomänenprinzip (TDP)*

> In sprachlichen Zeichen hat jede Topikalisierungsspur innerhalb ihrer Topikalisierungsdomäne einen Binder, der zu ihr in der move-α-Relation steht und keine andere Topikalisierungsspur in dieser Domäne bindet, und jedes topikalisierte Element steht innerhalb seiner Topikalisierungsdomäne in der move-α-Relation zu einer Spur.

$$
\begin{array}{l}
\forall x \\
: \sim \mathit{sign} \wedge \texttt{top-barrier}(x) \quad \longrightarrow \\
\quad (\forall t \\
\quad t\textsc{phon} \sim \mathit{elist} \wedge \\
\quad \neg\, t\textsc{ss loc cat head def} \sim \mathit{inter} \wedge \\
\quad \texttt{dominate}(x,t) \wedge \\
\quad [\forall y\ (\texttt{top-barrier}(y) \wedge \texttt{dominate}(y,t)) \longrightarrow \texttt{dominate}(y,x)] \\
\quad \longrightarrow \\
\qquad \exists s\ \exists a \\
\qquad \texttt{contains-fr-word}(s,a) \wedge \\
\qquad s\textsc{ss status} \sim \mathit{complete} \wedge \\
\qquad s\textsc{ss loc cat head vform} \sim \mathit{finite} \wedge \\
\qquad \texttt{dominate}(x,s) \wedge \\
\qquad [\forall y\ (\texttt{top-barrier}(y) \wedge \texttt{dominate}(y,s)) \longrightarrow \texttt{dominate}(y,x)] \wedge \\
\qquad \texttt{move-}\alpha(s, a\textsc{ss loc cat head fr co ss loc}, t) \\
\qquad \wedge \\
\qquad \neg\, \exists t' \\
\qquad t'\textsc{phon} \sim \mathit{elist} \wedge \\
\qquad [\forall y\ (\texttt{top-barrier}(y) \wedge \texttt{dominate}(y,t')) \longrightarrow \texttt{dominate}(y,x)] \wedge \\
\qquad \neg\, t \approx t' \wedge \\
\qquad t\textsc{ss loc} \approx t'\textsc{ss loc}) \\
\quad \wedge \\
\quad (\forall s\ \forall a \\
\quad \texttt{contains-fr-word}(s,a) \wedge \\
\quad s\textsc{ss status} \sim \mathit{complete} \wedge \\
\quad s\textsc{ss loc cat head vform} \sim \mathit{finite} \wedge \\
\quad \neg\, a\textsc{ss loc cat head fr co ss loc cat head def} \sim \mathit{inter} \wedge \\
\quad a\textsc{ss loc cat head fr co ss loc cat head mod} \sim \mathit{minus} \wedge \\
\quad \texttt{dominate}(x,s) \wedge \\
\quad [\forall y\ (\texttt{top-barrier}(y) \wedge \texttt{dominate}(y,s)) \longrightarrow \texttt{dominate}(y,x)] \wedge \\
\quad \longrightarrow \\
\qquad \exists t \\
\qquad t\textsc{phon} \sim \mathit{elist} \wedge \\
\qquad \texttt{dominate}(x,t) \wedge \\
\qquad [\forall y\ (\texttt{top-barrier}(y) \wedge \texttt{dominate}(y,t)) \longrightarrow \texttt{dominate}(y,x)] \wedge \\
\qquad \texttt{move-}\alpha(s, a\textsc{ss loc cat head fr co ss loc}, t))
\end{array}
$$

Zum besseren Verständnis wiederholt (5.63) den RSRL-Satz aus (5.62) in Worten, wobei die Struktur von (5.62) beibehalten wird. Man beachte

dabei, daß die Relation dominate reflexiv definiert ist (5.42), das heißt, jedes Zeichen dominiert sich selbst.

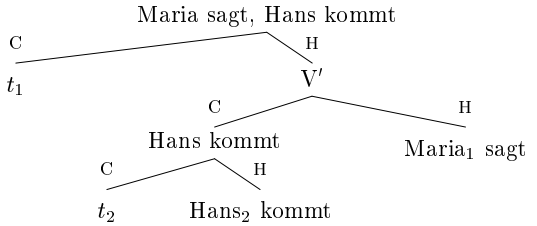
(5.63) *Das TDP in natürlichsprachlicher Formulierung*

Für alle Topikalisierungsbarrieren x in jedem Zeichen gilt:

- für jede Topikalisierungsspur t, so daß x t dominiert und zwischen x und t keine andere Topikalisierungsbarriere interveniert:
  - es gibt einen Satz s mit Opener a, so daß x s dominiert und zwischen x und s keine andere Topikalisierungsbarriere interveniert und die FRONTED-Konstituente von a in s in der move-α-Relation zu t steht, und
  - es gibt keine von t verschiedene Spur t', so daß x t' dominiert und keine andere Topikalisierungsbarriere zwischen x und t' interveniert, die mit t ihren LOCAL-Wert teilt (und daher als zulässige Spur in s in der move-α-Relation zur FRONTED-Konstituente von a steht), und
- für jeden Satz s mit einem Opener a mit topikalisierter FRONTED-Konstituente, so daß x s dominiert und keine andere Topikalisierungsbarriere zwischen x und s interveniert:
  - es gibt eine Spur t, so daß x t dominiert und zwischen x und t keine andere Topikalisierungsbarriere interveniert, und die FRONTED-Konstituente von a steht in s in der move-α-Relation zu t.

Durch das TDP muß jede Spur in ihrer Topikalisierungsdomäne von einer FRONTED-Konstituente gebunden werden, die keine andere Spur in dieser Topikalisierungsdomäne bindet. Dies schließt Parasitic-Gap-Konstruktionen im Deutschen aus. Umgekehrt muß jede topikalisierte Konstituente eine Spur in ihrer Topikalisierungsdomäne binden. Die Baumstruktur in (5.64) illustriert die Wirkung des TDP.

(5.64) *Beispielstruktur für die Wirkung des TDP*



(5.64) enthält vier verschiedene Topikalisierungsbarrieren: den Matrixsatz (T_1), den eingebetteten V2-Satz (T_2), das Vorfeld des Matrixsatzes (T_3) und das Vorfeld des eingebetteten Satzes (T_4). Darüber hinaus ist der Matrixsatz als uneingebettetes Zeichen eine Topikalisierungsbarriere (T_1). Die beiden Vorfeldbarrieren T_3 und T_4 erfüllen das TDP trivial, da sie weder Spuren noch Sätze dominieren. Betrachten wir nun die Wirkung des TDP für die beiden übrigen Topikalisierungsbarrieren, indem wir vom Wurzelknoten der Baumstruktur ausgehen.

Dann gilt für den Matrixsatz als Topikalisierungsbarriere T_1: t_1 ist die einzige Spur, die von T_1 und keiner anderen Topikalisierungsbarriere dominiert wird. Der Matrixsatz ist ein Satz mit einem Opener, den keine andere Topikalisierungsbarriere dominiert, die nicht T_1 dominiert, und die FRONTED-Konstituente *Maria* des Openers *Maria lacht* steht im Matrixsatz in der move-α-Relation zu t_1. Außerdem dominiert T_1 keine zweite Spur, die nicht von einer anderen Topikalisierungsbarriere dominiert wird und den gleichen LOCAL-Wert wie t_1 hat. t_2 kommt bereits deshalb nicht in Betracht, weil sie außer von T_1 auch von T_2 dominiert wird.

Darüber hinaus ist der Matrixsatz selbst der einzige Satz mit topikalisierter FRONTED-Konstituente (*Maria*) in seinem Opener, der außer von T_1 von keiner anderen Topikalisierungsbarriere dominiert wird, die nicht auch T_1 dominiert. Der eingebettete Satz *Hans kommt* kommt nicht in Frage, da er von $T2$ dominiert wird, aber T_2 nicht T_1 dominiert. t_1 ist dann eine Spur, die von T_1 und keiner anderen Topikalisierungsbarriere dominiert wird, und zu der *Maria* im Matrixsatz in der move-α-Relation steht. t_2 kommt wiederum nicht als von *Maria* gebunden in Betracht, da T_2 interveniert.

Damit entsprechen die Binder-Spur-Verhältnisse im Matrixsatz von (5.64) den Anforderungen des TDP. Zu überprüfen bleibt der Fall von T_2: t_2 ist die einzige Spur, die von T_2 aber von keiner anderen Topikalisierungsbarriere dominiert wird, die nicht auch T_2 dominiert. T_1 dominiert zwar ebenfalls t_2, aber T_1 dominiert auch T_2 und interveniert daher nicht

zwischen T_2 und t_2. Der eingebettete Satz *Hans kommt* mit Opener *Hans kommt* ist ein Satz, der von T_2 ohne das Dazwischentreten einer andern Topikalisierungsbarriere dominiert wird, und *Hans* steht in diesem Satz in der move-α-Relation zu t_2.

Auf der anderen Seite ist der eingebettete V2-Satz der einzige Satz mit topikalisierter FRONTED-Konstituente, den T_2 dominiert. Mit t_2 enthält er die erforderliche Spur, zu der die topikalisierte Konstituente *Hans* in move-α-Relation steht.

Das Antezedens des TDP ist nicht nur für den höchsten Knoten im Baum wahr, sondern für alle in sprachlichen Zeichen enthaltenen Objekte einer maximal spezifischen Untersorte der Sorte *sign*. Weil das Wurzelzeichen eines Satzes aber alle anderen Objekte des Satzes als Teile und damit alle darin enthaltenen Topikalisierungsbarrieren enthält, genügt es, das TDP für den höchsten Knoten im Baum zu zeigen. Also haben wir gezeigt, daß das TDP von dem in (5.64) beschriebenen Satz erfüllt wird und keine anderen Antezedens-Spur-Verhältnisse als die angegebenen möglich sind.

Man kann sich nun leicht davon überzeugen, daß das TDP sowohl die lange Topikalisierung (5.59a–b) aus w-Inseln lizenziert als auch alle in (5.60) angeführten Daten korrekt beschreibt, sofern man davon ausgeht, daß es sich bei den ungebundenen Spuren in (5.60e–f) um nicht-w-Spuren handelt. w-Spuren sind Gegenstand des nächsten Abschnitts. Bevor jedoch w-Spuren und w-Barrieren betrachtet werden, soll kurz erläutert werden, warum das TDP Adjunktspuren und Konstruktionen ausschließt, in denen eine topikalisierte Konstituente zwei Spuren gleichzeitig bindet. Hierzu betrachten wir die Sätze in (5.65).

(5.65) a. *Karl$_1$ lacht t_2 t_1.
b. *Dem Kind$_1$ sagt Maria t_1, daß Hans t'_1 das Buch gegeben hat.

Nehmen wir an, t_2 in (5.65a) sei eine Adjunktspur, also ein Wort mit Kopfwert *adjunct* und MOD-Wert *synsem*. Außerdem sei t_2 keine w-Spur. (5.65a) hat drei Topikalisierungsbarrieren: das Vorfeld, den Satz als ganzen und die Adjunktspur t_2. Eine Adjunktspur muß als Nichtkopftochter aufgrund des Kopf-Adjunkt-Schemas (3.32) zwangsläufig syntaktisch *complete* sein, und ist damit notwendig eine Topikalisierungsbarriere (siehe (5.61)). Das TDP fordert folglich unter anderem, daß t_2 einen Satz dominiert, der das Antezedens von t_2 als FRONTED-Konstituente enthält. t_2 dominiert aber nur sich selbst und daher insbesondere keinen finiten Satz mit Opener. Folglich kann t_2 nicht in der geforderten Weise gebunden werden, und das TDP schließt die Adjunktspur aus. Analoge Überlegungen zeigen, daß durch das TDP nicht-w-Spuren im Vorfeld ausgeschlossen werden.[28] In Abschnitt 5.4.2 werden ergänzend entsprechende w-Spuren ausgeschlossen werden.

(5.65b) ist ein Beispiel für unzulässige Bindung zweier Spuren durch eine einzige topikalisierte Konstituente. Der Satz enthält insgesamt zwei Topikalisierungsbarrieren, nämlich den Matrix-V2-Satz und die Vorfeldkonstituente des Matrixsatzes. daß-VL-Sätze sind aufgrund der definierenden Klausel der Relation top-barrier in (5.61) keine Topikalisierungsbarrieren. Infolgedessen kann die topikalisierte Konstituente *dem Kind* die Spur t'_1 durchaus binden, weil keine andere Topikalisierungsbarriere interveniert. Da *dem Kind* der einzige potentielle Binder ist, muß er t'_1 sogar obligatorisch binden, da das TDP für jede Spur ein Antezedens in ihrer Topikalisierungsdomäne fordert. Dann kann aber die Spur t_1, die ebenfalls ein Antezedens braucht, nicht mehr ihren LOCAL-Wert mit dem der Konstituente *dem Kind* teilen, da es sonst eine zweite Spur in der Topikalisierungsdomäne von t'_1 mit identischem *local*-Objekt gibt, was vom TDP verboten wird. Daher ist (5.65b) in der Lesart, in der *dem Kind* gleichzeitig indirektes Objekt von *sagt* und *gegeben* ist, ungrammatisch.

5.4.2. *w-Bewegungsdomänen.* Abgesehen von den oben als Motivation der syntakischen Unterscheidung von Topikalisierung und w-Bewegung genannten Extraktionsdaten aus w-Inseln (5.59a–e) entspricht die Faktenlage bei w-Bewegung der bei Topikalisierung. (5.66) greift die Sätze aus (5.60) wieder auf und ersetzt Topikalisierung durch w-Bewegung. In den Sätzen (5.66e–f) wird eine ungebundene w-Spur t beziehungsweise t_2 angenommen.

(5.66) a. Wen$_1$ sagt Maria, daß Karl t_1 füttert.
b. *Wen$_2$ sagt Maria, Karl$_1$ füttert t_1 t_2.
c. *Wen$_1$ sagt Maria, t_1 füttert Karl t'_1.
d. *Wen$_1$ macht Maria die Tür zu, bevor Karl t_1 füttert.
e. *Daß Karl jeden Tag t füttert!
f. *Karl$_1$ füttert jeden Tag t_1 t_2.

Während aus daß-VL-Sätzen lang w-bewegt werden kann (5.66a), sind adverbiale Adjunkte und V2-Sätze strikte Inseln für w-Bewegung (5.66b–d). Der grammatische Satz, der phonologisch (5.66c) entspricht, enthält

[28] Fälle von Vorfeldellipse werden hier ignoriert. Wir gehen davon aus, daß zur Behandlung solcher Konstruktionen ein leeres Element in die Grammatik eingeführt wird, dessen Verwendung vom Kontext lizenziert werden muß und dessen inhärente Eigenschaften es von Spuren, die von einer bewegten Konstituente gebunden werden müssen, unterscheiden. Damit ist dann eine Differenzierung der einschlägigen Prinzipien hinsichtlich der beiden Arten von leeren Elementen in der Grammatik möglich.

die V1-Parenthese *sagt Maria.* (5.66e–f) sind ausgeschlossen, weil uneingebettete Zeichen keine ungebundenen Spuren enthalten dürfen, unabhängig davon, ob es sich um nicht-w-Spuren oder w-Spuren handelt.

Aus der Datenlage ergibt sich eine zu Topikalisierungsbarrieren weitestgehend parallele Formulierung von w-Barrieren. Der einzige Unterschied besteht darin, daß w-VL- und ob-VL-Sätze zusätzliche w-Barrieren bilden:

(5.67) *Die Klausel von w-barrier*

$$\texttt{w-barrier}(x) :=$$
$$\texttt{top-barrier}(x) \vee$$
$$^{x}\left[\textit{phrase} \quad \text{SYNSEM} \left[\text{STATUS} \quad \textit{complete} \quad \text{LOC CAT} \left[\text{HEAD VFORM} \quad \textit{finite} \quad \text{MARKING} \quad w\right]\right]\right]$$

x ist eine w-Barriere genau dann, wenn x entweder eine Topikalisierungsbarriere oder ein finiter Verbletztsatz mit MARKING-Wert w ist, also ein ob-VL- oder ein w-VL-Satz. Da von der neuen Barriere ein phrasaler Status gefordert wird, sind Spuren von ob-VL- oder w-VL-Sätzen im Unterschied zu Adjunktspuren zulässig, das heißt, diese Sätze können selbst bewegt werden, obwohl aus ihnen nicht w-bewegt werden darf:

(5.68) a. [Ob Peter schon gekommen ist]$_1$ wußte Maria nicht t_1.
b. [Wer das alles machen soll]$_1$ verrät einem natürlich keiner t_1.

Auf der Grundlage der Relation w-barrier kann das w-Bewegungsdomänenprinzip in Entsprechung zum TDP formuliert und formalisiert werden. Es ergibt sich durch eine einfache Ersetzung der Begriffe nicht-w-Spur durch w-Spur, Topikalisierungsbarriere durch w-Barriere und, darauf aufbauend, Topikalisierungsdomäne durch w-Bewegungsdomäne.

(5.69) *Das w-Bewegungsdomänenprinzip (WDP)*

> In sprachlichen Zeichen hat jede w-Spur innerhalb ihrer w-Bewegungsdomäne einen Binder, der zu ihr in der move-α-Relation steht und keine andere w-Spur in dieser Domäne bindet, und jedes w-bewegte Element steht innerhalb seiner w-Bewegungsdomäne in der move-α-Relation zu einer Spur.

$$\begin{array}{l}
\forall x \\
: \sim \textit{sign} \wedge \texttt{w-barrier}(x) \quad \longrightarrow \\
\quad (\forall t \\
\quad t\text{PHON} \sim \textit{elist} \wedge \\
\quad t\text{SS LOC CAT HEAD DEF} \sim \textit{inter} \wedge \\
\quad \texttt{dominate}(x,t) \wedge \\
\quad [\forall y\ (\texttt{w-barrier}(y) \wedge \texttt{dominate}(y,t)) \longrightarrow \texttt{dominate}(y,x)] \\
\quad \longrightarrow \\
\qquad \exists s\ \exists a \\
\qquad \texttt{contains-fr-word}(s,a) \wedge \\
\qquad s\text{SS STATUS} \sim \textit{complete} \wedge \\
\qquad s\text{SS LOC CAT HEAD VFORM} \sim \textit{finite} \wedge \\
\qquad \texttt{dominate}(x,s) \wedge \\
\qquad [\forall y\ (\texttt{w-barrier}(y) \wedge \texttt{dominate}(y,s)) \longrightarrow \texttt{dominate}(y,x)] \wedge \\
\qquad \texttt{move-}\alpha(s, a\text{SS LOC CAT HEAD FR CO SS LOC}, t) \\
\qquad \wedge \\
\qquad \neg\ \exists t' \\
\qquad t'\text{PHON} \sim \textit{elist} \wedge \\
\qquad [\forall y\ (\texttt{w-barrier}(y) \wedge \texttt{dominate}(y,t')) \longrightarrow \texttt{dominate}(y,x)] \wedge \\
\qquad \neg\ t \approx t' \wedge \\
\qquad t\text{SS LOC} \approx t'\text{SS LOC}) \\
\wedge \\
(\forall s\ \forall a \\
\texttt{contains-fr-word}(s,a) \wedge \\
s\text{SS STATUS} \sim \textit{complete} \wedge \\
s\text{SS LOC CAT HEAD VFORM} \sim \textit{finite} \wedge \\
a\text{SS LOC CAT HEAD FR CO SS LOC CAT HEAD DEF} \sim \textit{inter} \wedge \\
a\text{SS LOC CAT HEAD FR CO SS LOC CAT HEAD MOD} \sim \textit{minus} \wedge \\
\texttt{dominate}(x,s) \wedge \\
[\forall y\ (\texttt{w-barrier}(y) \wedge \texttt{dominate}(y,s)) \longrightarrow \texttt{dominate}(y,x)] \wedge \\
\longrightarrow \\
\quad \exists t \\
\quad t\text{PHON} \sim \textit{elist} \wedge \\
\quad \texttt{dominate}(x,t) \wedge \\
\quad [\forall y\ (\texttt{w-barrier}(y) \wedge \texttt{dominate}(y,t)) \longrightarrow \texttt{dominate}(y,x)] \wedge \\
\quad \texttt{move-}\alpha(s, a\text{SS LOC CAT HEAD FR CO SS LOC}, t))
\end{array}$$

(5.70) gibt den RSRL-Satz aus (5.69) in sprachlicher Umschreibung wieder. Dabei wird die Einrückungsstruktur zur Verdeutlichung beibehalten.

(5.70) *Das WDP in natürlichsprachlicher Formulierung*

> Für alle w-Barrieren x in jedem Zeichen gilt:
> - für jede w-Spur t, so daß x t dominiert und zwischen x und t keine andere w-Barriere interveniert:

  - es gibt einen Satz s mit Opener a, so daß x s dominiert und zwischen x und s keine andere w-Barriere interveniert und die FRONTED-Konstituente von a in s in der move-α-Relation zu t steht, und
  - es gibt keine von t verschiedene Spur t', so daß x t' dominiert und keine andere w-Barriere zwischen x und t' interveniert, die mit t ihren LOCAL-Wert teilt (und daher als zulässige Spur in s in der move-α-Relation zur FRONTED-Konstituente von a steht), und
- für jeden Satz s mit einem Opener a mit w-bewegter FRONTED-Konstituente, so daß x s dominiert und keine andere w-Barriere zwischen x und s interveniert:
  - es gibt eine Spur t, so daß x t dominiert und zwischen x und t keine andere w-Barriere interveniert, und die FRONTED-Konstituente von a steht in s in der move-α-Relation zu t.

Der interessante Unterschied zwischen WDP und TDP wird anhand eines Vergleichs des Satzes (5.71a) mit (5.71b) und (5.71c) deutlich.

(5.71) a. ?Radios$_1$ weiß ich nicht, wer$_2$ t_2 t_1 repariert.
b. *[Welchen Mann]$_1$ weißt du nicht, wer$_2$ t_2 t_1 gesehen hat.
c. *Wen$_1$ weißt du nicht, wie Brutus t_1 erstochen hat.

In jedem der drei Sätze bilden die Vorfeldkonstituente des Matrixsatzes und die FRONTED-Konstituente des w-Markers des eingebetteten Satzes Topikalisierungs- und w-Barrieren, von denen wir für die folgenden Überlegungen abstrahieren. TDP und WDP sind für sie jeweils trivial erfüllt, da sie weder Sätze noch Spuren dominieren.

Alle drei Sätze weisen die gleichen interessanten Barrieren und Domänen auf. Der Matrixsatz ist als uneingebettetes Zeichen und V2-Satz sowohl Topikalisierungs- als auch w-Barriere. Der subkategorisierte w-VL-Satz hingegen ist alleine eine w-Barriere. Daraus folgt, daß — abgesehen von den FRONTED-Konstituenten — jeweils eine das gesamte Zeichen umfassende Topikalisierungsdomäne, aber zwei w-Bewegungsdomänen vorliegen. Vereinfachend gesprochen umfaßt die eine w-Domäne den Matrixsatz und die zweite den eingebetteten Satz.

Entscheidend ist nun, daß TDP und WDP hinsichtlich verschiedener bewegter Elemente und Spuren sensitiv sind und deren Bindungsverhältnisse in den jeweiligen Domänen erzwingen. In (5.71a) benötigt aufgrund des TDP die topikalisierte Konstituente *Radios* im Gesamtsatz eine Spur,

und die nicht-w-Spur t_1 sucht im Gesamtsatz ein Antezedens. Das WDP hat zu beiden nichts zu sagen. Seine Aufgabe ist es, innerhalb des eingebetteten w-VL-Satzes die Bindungsverhältnisse zwischen *wer* und seiner w-Spur t_2 zu erzwingen. Abgesehen von den inkompatiblen Kasusanforderungen an das Akkusativobjekt und das Subjekt von *repariert* können die beiden Spuren auch aufgrund des WDP nicht gemeinsam ihren LOCAL-Wert mit *wer* teilen. *wer* darf nur genau eine Spur binden. Darüber hinaus muß *Radios* im eingebetteten Satz eine Spur binden, da im Matrixsatz keine vorhanden ist. Man beachte, daß im eingebetteten Satz zwei Spuren gleichzeitig auftreten.

(5.71b) ist ungrammatisch, obwohl die gleichen Domänenverhältnisse wie in (5.71a) herrschen und die Verteilung von Spuren und Bindern dieselbe ist. Allerdings wurde in (5.71b) eine w-Konstituente aus der w-Insel bewegt, so daß für ihre Antezedens-Spur-Bedingungen nun das WDP anzuwenden ist. Das WDP verlangt aber, daß *welchen Mann* im Matrixsatz eine Spur bindet und t_1 im eingebetteten w-VL-Satz seinen Binder findet.

(5.71c) zeigt, daß es keine Rolle spielt, ob in der w-Insel selbst w-Bewegung vorliegt oder nicht. *wen* findet in seiner w-Domäne keine Spur, und t_1 fehlt in seiner w-Domäne der Binder. Die FRONTED-Konstituente *wie* des w-VL-Satzes ist nicht Gegenstand der Spurforderungen des WDP, da es als Adjunkt den MOD-Wert *synsem* aufweist. Damit liegen in (5.71c) in relevanter Hinsicht genau die gleichen Verhältnisse vor wie in (5.72):

(5.72) *Wen$_1$ weißt du nicht, ob Brutus t_1 erstochen hat.

Aufgabe des Illokutionsprinzips 3 (5.27) in Abschnitt 5.2 ist es zu bestimmen, sprachliche Zeichen welcher Art als selbständige Äußerungen mit eigener illokutiver Kraft und somit als *u-signs* vorkommen dürfen. In grob vereinfachender Annäherung wurden finite Sätze ohne offene Spuren als solche Zeichen festgelegt und eine Relation no-open-traces angedeutet, welche die Funktion hatte, Spuren ohne Antezedens in wohlgeformten *u-signs* auszuschließen. Mit dem TDP und dem WDP liegt jetzt in der Theorie der langen Abhängigkeiten eine technische Präzisierung von no-open-traces vor. Ein *u-sign* ist sowohl eine Topikalisierungs- als auch eine w-Bewegungsbarriere. Jede Spur ist entweder eine w-Spur oder eine nicht-w-Spur. Daraus folgt unmittelbar, daß für jede Spur gleich welchen Typs, die sich in einer für sie einschlägigen, durch ein *u-sign* nach oben begrenzten Domäne befindet, in dieser Domäne auch ein passender Binder gegenwärtig sein muß. Also kann es in uneingebetteten Zeichen keine Spuren ohne Antezedens geben. Damit kann Illokutionsprinzip 3 vereinfacht werden:

(5.73) *Illokutionsprinzip 3 (endgültige Formulierung)*

$$\left[\textit{u-sign}\right] \Longrightarrow$$

$$\left[\text{SS}\left[\begin{array}{l}\text{STATUS}\quad \textit{complete}\\ \text{LOC CAT HEAD VFORM}\quad \textit{finite}\end{array}\right]\right]$$

In der neuen Formulierung beschränkt sich Illokutionsprinzip 3 auf die Forderung, daß selbständige Äußerungen finite Sätze sind, und geht nicht auf in diesen Sätzen herrschende Beziehungen ein. In Abschnitt 5.2 wurde bereits darauf hingewiesen, daß die Annahme, nur finite Sätze könnten selbständige Äußerungen sein, zu restriktiv ist und in einer adäquaten Illokutionstheorie erweitert werden muß.

Die oben definierten Barrierenbegriffe beruhen auf den Daten zu finiten Satzkomplementen von Brückenverben. Nun ist es aber bekannt, daß Nichtbrückenverben restriktivere Anforderungen an ihre finiten Satzkomplemente stellen, indem sie keine lange Extraktion gestatten. (5.74) zeigt dies anhand des Nichtbrückenverbs *bezweifeln*.

(5.74) a. *Radios$_1$ bezweifelt Maria, daß Hans t_1 repariert.
b. *Wen$_1$ hat Maria bezweifelt, daß du t_1 gesehen hast?

Die Ungrammatikalität von (5.74a) kann offensichtlich damit erklärt werden, daß der subkategorisierte daß-VL-Satz eine Topikalisierungsbarriere bildet. Damit fände *Radios* in seiner auf den Matrixsatz beschränkten Topikalisierungsdomäne keine Spur und die nicht-w-Spur t_1 in ihrer Topikalisierungsdomäne keinen Binder. Das gleiche gilt mutatis mutandis für (5.74b). Ist der subkategorisierte daß-VL-Satz eine w-Barriere, findet *wen* keine Spur in seiner w-Domäne und die w-Spur t_1 in ihrer w-Domäne keinen Binder. Man kann daraus schließen, daß finite Komplementsätze, die nach den obigen Barrierenbegriffen keine Barrieren sind, unter Nichtbrückenverben offenbar zu Barrieren sowohl für Topikalisierung als auch für w-Bewegung werden. Mit anderen Worten werden finite Sätze im Kontext von Nichtbrückenverben grundsätzlich zu strikten Inseln für Bewegung.

Insofern, als die bisherigen Barrieren permissiverer Natur sind, als die hier benötigten, stellt es kein Problem dar, entlang beobachteter empirischer Generalisierungen den Barrierenbegriff zu verschärfen. Eine erste Annäherung an die Datenlage bei Nichtbrückenverben bestünde somit in der Generalisierung, daß finite Komplementsätze von Nichtbrückenverben immer w-Barrieren und Topikalisierungsbarrieren sind. Genau dies drückt (5.75) mit der Einführung einer zweiten Klausel für top-barrier (5.61) aus:

(5.75) *Die zweite Klausel von top-barrier*

$$\begin{array}{l}\texttt{top-barrier}(x) :=\\ \quad {}^{x}\left[\begin{array}{l}\textit{phrase}\\ \text{SS}\ \boxed{1}\left[\begin{array}{l}\text{STATUS}\ \textit{complete}\\ \text{LOC CAT HEAD VFORM}\ \textit{finite}\end{array}\right]\end{array}\right] \wedge\\ \quad \exists y\\ \quad {}^{y}\left[\begin{array}{l}\textit{word}\\ \text{SS LOC CONT NUCLEUS}\ \textit{nonbridgeverb}\end{array}\right] \wedge\\ \quad \texttt{member}(\boxed{1}, y\text{SS LOC CAT VAL COMPS})\end{array}$$

(5.75) macht Gebrauch von der in (5.9) definierten Relation member, deren erstes Argument ein Objekt ist, das in der im zweiten Argument stehenden Liste enthalten ist. (5.75) fügt den bisherigen Topikalisierungsbarrieren alle die x hinzu, die phrasale finite Sätze sind, so daß es ein Nichtbrückenverb y gibt, das ihren SYNSEM-Wert subkategorisiert. In der Anwendung im WDP und TDP heißt das konkret, daß x in dem betrachteten Zeichen Z eine Barriere des jeweiligen Typs ist, wenn x ein phrasaler finiter Satz ist und es in Z ein Nichtbrückenverb gibt, dessen Komplement dieser Satz ist.[29] Man beachte dabei, daß die Erweiterung von top-barrier automatische eine Erweiterung von w-barrier (5.67) induziert, da jede Topikalisierungsbarriere auch eine w-Barriere ist.

Mit dem neuen Barrierenbegriff ist der eingebettete daß-VL-Satz in (5.74a) eine Topikalisierungsbarriere, da er vom Nichtbrückenverb *bezweifelt* subkategorisiert wird. Entsprechend ist der daß-VL-Satz in (5.74b) eine w-Bewegungsbarriere, da er von *bezweifelt* subkategorisiert wird. Dabei spielt es keine Rolle, daß *bezweifelt* im ersten Satz das finite Verb ist und im zweiten Argumentvererbung an das temporale Hilfsverb *hat* stattgefunden hat, da der SYNSEM-Wert des Komplementsatzes in jedem Fall auf der COMPS-Liste von *bezweifelt* steht.

Die Formulierung eines Barrierenbegriffs für finite Satzkomplemente von Nichtbrückenverben kann als Beispiel für Barrieren betrachtet werden, die durch bestimmte syntaktische Kontexte induziert werden. Eine weitere Anwendung könnte die damit illustrierte Technik etwa bei der

[29] Diese Formalisierung geht davon aus, daß die *synsem*-Objekte zweier finiter Sätze innerhalb eines linguistischen Zeichens voneinander verschieden sind. Es ist naheliegend anzunehmen, daß dies aufgrund des in *synsem*-Objekten finiter Sätzen enthaltenen Reichtums an Information — wobei insbesondere an die semantische Struktur zu denken ist — immer der Fall ist, selbst wenn es nicht durch ein gesondertes Prinzip erzwungen wird. Sollte dies in der vollständig spezifizierten Grammatik nicht zutreffen, kann diese Eigenschaft durch ein zusätzliches Prinzip erzwungen werden.

Formalisierung solch klassischer Beschränkungen wie des Complex-NP-Constraints finden. Die Tragweite dieses Ansatzes ließe sich so zum Beispiel anhand einer Erweiterung der hier vorgestellten Theorie der langen Abhängigkeiten auf komplexe Nominal- und Präpositionalphrasen überprüfen. Dies muß indessen Gegenstand zukünftiger Forschung bleiben.

5.4.3. *Fazit.* TDP und WDP haben gezeigt, daß die in Kapitel 3 entfaltete Satzstruktur für finite Sätze im Deutschen eine gute Ausgangsbasis für eine kompakte Theorie langer Abhängigkeiten darstellt. Die dabei zur Anwendung gelangten Barrierenbegriffe lehnen sich eng an im Rahmen der GB-Theorie formulierte Konzepte an, obwohl sich ihre Umsetzung in technischen Details aufgrund der unterschiedlichen formalen Grundlagen und des linearisierungsbasierten Ansatzes der hier vorgestellten Grammatik deutlich von der GB-Tradition abhebt.

Eine genauere Betrachtung der Formulierung von WDP und TDP offenbart, daß sie tatsächlich nichts anderes als eine Erweiterung des Frontedbesetzungsprinzips (5.13) aus Abschnitt 5.1 sind. Während das FBP für beliebige Komplemente in der linken Peripherie naiv fordert, daß sie bezüglich ihres Satzes in der move-α-Relation zu einer Spur stehen, betreffen die entsprechenden Forderungen von WDP und TDP w-Konstituenten beziehungsweise nicht-w-Konstituenten in der linken Satzperipherie. WDP und TDP verlangen im Gegensatz zum FBP nicht einfach eine Spur in move-α-Relation an beliebiger Stelle, sondern innerhalb der jeweiligen, durch Barrieren begrenzten Bewegungsdomäne. Im Effekt bedeutet dies, daß die Barrierentheorie das FBP vollständig ersetzt und letzteres aus der Grammatik entfernt werden kann. Indem die Barrierentheorie außerdem sicherstellt, daß alle Spuren eines uneingebetteten Zeichens gebunden sind, und die Verteilung der Binder in der Struktur beschränkt, geht ihre Wirkung über die Leistung des FBP weit hinaus. Es ist indessen interessant festzustellen, daß all diese Bedingungen in zwei Prinzipien realisiert werden konnten, deren intuitiver Gehalt den sprachwissenschaftlich anvisierten Generalisierungen unmittelbar entspricht.

Die relationale Theorie der langen Abhängigkeiten führt keine zusätzliche Struktur in sprachliche Zeichen ein, die — wie die traditionell übliche *nonlocal*-Struktur — linguistisch nur schwer interpretiert oder motiviert werden kann. Die Schwierigkeiten einer sprachwissenschaftlichen Interpretation der *nonlocal*-Struktur rühren daher, daß sie nicht wie beispielsweise *person*, *gender* oder *nominative* linguistischen Beschreibungsanforderungen hinsichtlich von Eigenschaften sprachlicher Objekte nachkommt, sondern alleine einem technisch notwendigen Informationstransfer in sprachlichen Objekten dient. Dabei wird die zu vermittelnde Information selbst

als neuer Teil von Objekten betrachtet, die ursprünglich nicht mit dem untersuchten empirischen Phänomen in Verbindung stehen. Im Gegensatz dazu verwendet die relationale Theorie Relationen, um globale Beschränkungen über Merkmalstrukturen auszudrücken. Das heißt, Wohlgeformtheitsbedingungen über Beziehungen zwischen Teilen sprachlicher Strukturen beschränken deren mögliche Gestalt, ohne daß deren Gestalt an anderer Stelle dadurch verändert oder angereichert wird. Die dadurch erzielte Kompaktheit der Merkmalsgeometrie erleichtert insbesondere in größeren Grammatikfragmenten die Modularisierung einzelner Bestandteile und trägt nicht zuletzt dazu bei, daß die sprachlichen Strukturen für den Linguisten überschaubar bleiben. Die Wichtigkeit dieser Tatsache ist angesichts der Komplexität der in Frage stehenden Zusammenhänge nicht niedrig zu schätzen.

Voraussetzung für die Formulierung der Barrierentheorie ist das Vorliegen eines wohldefinierten Satzbegriffes und eine daran anknüpfende kategoriale Unterscheidung der verschiedenen Satztypen. Nur dadurch ist es überhaupt erst möglich, die gewünschten Generalisierungen direkt in linguistisch naheliegender Weise in der Grammatik auszudrücken, das heißt, Aussagen über V2-Sätze oder VL-Sätze verschiedener Typen direkt anzugeben. Im gleichen Zusammenhang erwies sich die Anbindung einer einfachen Illokutionstheorie als vielversprechender Ansatzpunkt für eine Verzahnung von Syntax, Semantik und Pragmatik, welche für die Formulierung syntaktischer Zusammenhänge genutzt werden kann (vgl. vor allem das Illokutionsprinzip 2 in (5.24)). Eine allgemeine Charakterisierung uneingebetteter Zeichen ist daneben für eine nicht nur in Metaannahmen angesiedelte Barrierentheorie von Bedeutung, indem sie es gestattet, die Existenz unabgebundener Spuren in Root-Clauses theorieintern auszuschließen.

Das Konzept des Openers finiter Sätze bildet in mehrfacher Hinsicht den Brennpunkt der vorgestellten Grammatik des Deutschen. Im Opener treffen sich empirische Generalisierungen mehrerer Module der Grammatik, und Opener sind Gegenstand der wichtigsten Prinzipien. So dienen sie der formalen Präzisierung des Satzbegriffes und bestimmen den Satztyp finiter Sätze; sie übernehmen eine wichtige Interfacefunktion hinsichtlich des Pragmatikmoduls; und sie sind Gegenstand einer zentralen Wortstellungsregel, die über die linke Peripherie der finiten Sätze des Deutschen generalisiert und damit wichtige deskriptive Generalisierungen des klassischen topologischen Feldermodells in die Grammatik inkorporiert. Zuletzt bilden Opener, indem sie die Landeposition für syntaktische Bewegung bereitstellen, die Verankerung der Barrieren- und Bewegungstheorie.

Die wichtigsten empirischen Generalisierungen über Opener betreffen den Verum-Fokus und die Zusammenfassung von finiten Verben und Komplementierern an der linken Satzperipherie zu einer natürlichen Klasse.

Mit der Abschaffung der *nonlocal*-Struktur hat das LOC-Attribut seine unterscheidende Funktion verloren. Es kann und sollte daher aus der Signatur entfernt werden. Damit entfallen *local*-Objekte, und die bislang auf ihnen definierten Attribute sind auf *synsem* zu definieren. Lange Abhängigkeiten betreffen *synsem*-Objekte, wodurch nunmehr auch der syntaktische Vollständigkeitsstatus von Antezedens und Spur identisch sein müssen. Dies ist eine erwünschte Konsequenz der vorgenommenen strukturellen Vereinfachung.

6. LITERATURVERZEICHNIS

Ait-Kaci, H. (1984). *A Lattice Theoretic Approach to Computation Based on a Calculus of Partially Ordered Type Structures*. Ph. D. thesis, University of Pennsylvania.

Brandt, M., M. Reis, I. Rosengren, und I. Zimmermann (1992). Satztyp, Satzmodus und Illokution. In I. Rosengren (Ed.), *Satz und Illokution*, Volume 1, pp. 1 – 90. Tübingen, Niemeyer.

Carpenter, B. (1992). *The Logic of Typed Feature Structures*. Cambridge University Press. Cambridge, Massachusetts, USA.

Courbet, E., K. De Kuthy, D. Meurers, F. Richter, und M. Sailer (1997). Ein HPSG-Fragment des Deutschen, Teil 2: Implementierung. Arbeitspapiere des SFB 340, Universität Tübingen.

d'Avis, F.-J. (1993). Zur Lage auf den w-Inseln. In F.-J. d'Avis, S. Beck, U. Lutz, J. Pafel, und S. Trissler (Eds.), *Extraktion im Deutschen I*, pp. 63–98. Universität Tübingen.

De Kuthy, K. (1996). Der Rattenfängereffekt bei Relativ– und Interrrogativsätzen im Deutschen. Magisterarbeit, Seminar für Sprachwissenschaft, Universität Tübingen.

Ebbinghaus, H.-D., J. Flum, und W. Thomas (1992). *Einführung in die mathematische Logik* (dritte Auflage). B.I.-Wissenschaftsverlag.

Frank, A. (1994). Verb Second by Lexical Rule or by Underspecification. Arbeitsberichte des SFB 340 43, Universität Stuttgart.

Gazdar, G., E. Klein, G. K. Pullum, und I. Sag (1985). *Generalized Phrase Structure Grammar*. Harvard University Press. Cambridge Massachusetts.

Hinrichs, E. und T. Nakazawa (1994). Partial-VP and Split-NP Topicalization in German - An HPSG Analysis. In E. Hinrichs, D. Meurers, und T. Nakazawa (Eds.), *Partial-VP and Split-NP Topicalization in German — A HPSG Analysis and its Implementation*, Arbeitspapiere des SFB 340, Nr. 58, pp. 1–46. Universität Tübingen.

Höhle, T. N. (1982). Explikationen für "normale Betonung" und "normale Wortstellung". In W. Abraham (Ed.), *Satzglieder im Deutschen. Vorschläge zur syntaktischen, semantischen und pragmatischen Fundierung*. Tübingen.

Höhle, T. N. (1986). Der Begriff "Mittelfeld". Anmerkungen über die Theorie der topologischen Felder. In A. Schöne (Ed.), *Kontroversen alte und neue. Akten des 7. Internationalen Germanistenkongresses Göttingen 1985*, Volume 3, pp. 329–340.

Höhle, T. N. (1992). Über Verum-Fokus im Deutschen. In J. Jacobs (Ed.), *Informationsstruktur und Grammatik*, pp. 112–141. Westdeutscher Verlag, Opladen.

Höhle, T. N. (1994). Spurenlose Extraktion. Manuscript. Universität Tübingen. July 7th, 1994.

Johnson, M. (1988). *Attributive-Value Logic and the Theory of Grammar*. CSLI.

Kasper, R. (1994). Adjuncts in the Mittelfeld. In J. Nerbonne, K. Netter, und C. Pollard (Eds.), *German in Head-Driven Phrase Structure Grammar*, Volume 46, pp. 39–69. CSLI.

Kathol, A. (1995). *Linearization-Based German Syntax*. Ph. D. thesis, Ohio State University.

King, P. J. (1989). *A logical Formalism for Head-Driven Phrase Structure Grammar*. Ph. D. thesis, University of Manchester.

King, P. J. (1994). An Expanded Logical Formalism for Head-Driven Phrase Structure Grammar. Arbeitspapiere des SFB 340 59, Universität Tübingen.

King, P. J. (1995). From Unification to Constraint. Unpublished lecture notes, Universität Tübingen.

Lutz, U. (1993). Zur Extraktion aus Adjunktsätzen im Deutschen. In F.-J. d'Avis, S. Beck, U. Lutz, J. Pafel, und S. Trissler (Eds.), *Extraktion im Deutschen I*, Arbeitspapiere des SFB 340, Nr. 34, pp. 133 – 189. Universität Tübingen.

Meurers, D. (1995). Towards a Semantics of Lexical Rules as used in HPSG. Paper presented at ACQUILEX II Workshop on Lexical Rules, Cambridge, UK.

Meurers, W. D. (1997). Statusrektion und Wortstellung in kohärenten Infinitkonstruktionen des Deutschen. In diesem Band.

Müller, G. und W. Sternefeld (1993). Improper Movement and Unambiguous Binding. *Linguistic Inquiry 24*(3), 461 – 507.

Pafel, J. (1993). Eine Überblick über die Extraktion aus Nominalphrasen im Deutschen. In F.-J. d'Avis, S. Beck, U. Lutz, J. Pafel, und S. Trissler (Eds.), *Extraktion im Deutschen I*, Arbeitspapiere des SFB 340, Nr. 34, pp. 191 – 245. Universität Tübingen.

Pollard, C. (1993, October). Lexical Rules as Metadescriptions. Handout. Paper presented at the Conference "Between Syntax, Semantics, and Logic" at the Universität Stuttgart.

Pollard, C., R. Levine, und R. Kasper (1994). Studies in Constituent Ordering. Toward a Theory of Linearization in Head-Driven Phrase Structure Grammar. Manuscript. Research Proposal to the National Science Foundation, Ms. Ohio State University.

Pollard, C. und I. Sag (1987). *Information Based Syntax and Semantics. Vol.1: Fundamentals*. CSLI Lecture Notes 13.

Pollard, C. und I. Sag (1994). *Head-Driven Phrase Structure Grammar*. University of Chicago Press.

Primus, B. (1989). Parameter der Herrschaft: Reflexivpronomina im Deutschen. *Zeitschrift für Sprachwissenschaft 8*, 53–88.

Reape, M. (1989). A Logical Treatment of Semi-free Word Order and Bounded Discontinuous Constituency. In *Proceedings of the 4th Conference of the European Chapter of the ACL*, pp. 103–110.

Reis, M. (1985). Satzeinleitende Strukturen im Deutschen. Über COMP, Haupt- und Nebensätze, w-Bewegung und die Doppelkopfanalyse. In W. Abraham (Ed.), *Erklärende Syntax des Deutschen*, pp. 271–311. TBL Verlag Gunter Narr. Tübingen.

Reis, M. (1995). *Wer glaubst du hat recht?* On the So-called Extractions from Verb-Second Clauses and Verb-First Paranthetical Constructions in German. *Sprache und Pragmatik 36*, 27 – 83.

Richter, F. und M. Sailer (1995). Remarks on Linearization. Reflections on the Treatment of LP-Rules in HPSG in a Typed Feature Logic. Master's thesis, Universität Tübingen.

Rounds, W. T. und R. T. Kasper (1986). A complete logical calculus for record structures representing linguistic information. In *Proceedings of the 25th annual conference of the Association for Computational Linguistics*, pp. 89–96.

Sailer, M. (1997). Adjunkte in einer Linearisierungsgrammatik. Syntax Semantik und Wortstellung. In diesem Band.

Stechow, A. v. und W. Sternefeld (1988). *Bausteine syntaktischen Wissens. Ein Lehrbuch der generativen Grammatik*. Westdeutscher Verlag. Opladen.

Trissler, S. (1993). P-Stranding im Deutschen. In F.-J. d'Avis, S. Beck, U. Lutz, J. Pafel, und S. Trissler (Eds.), *Extraktion im Deutschen I*, Arbeitspapiere des SFB 340, Nr. 34, pp. 247 – 291. Universität Tübingen.

Winhart, H. (1997). Die Nominalphrase in einem HPSG-Fragment des Deutschen. In diesem Band.