

Einführung in eine formale Grammatiktheorie: HPSG

Die Signatur

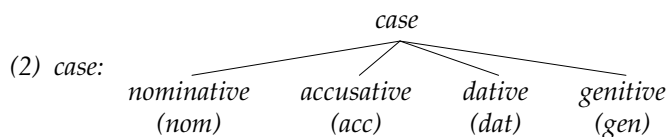
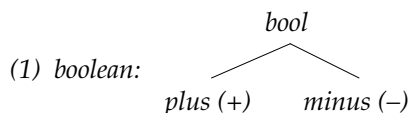
Wir fassen die Signatur zusammen, die wir unseren Überlegungen im Seminar zu Grunde gelegt haben. Sie ist nahezu – aber nicht immer – identisch mit der Signatur des Appendix im Buch von Pollard/Sag. Sinnvolle und für das Deutsche relevante Vorschläge von Sag/Wasow/Bender haben wir mit integriert.

1 Die Sortenhierarchie

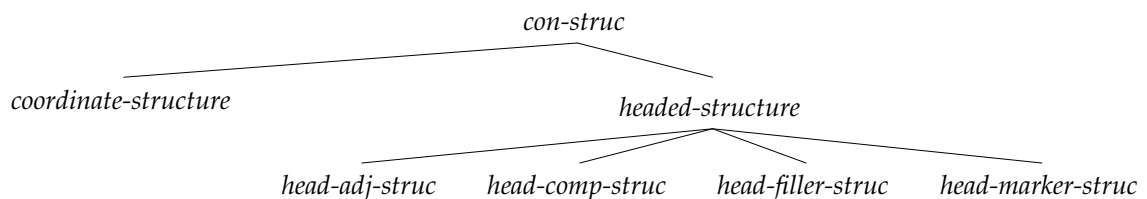
Die Sortenhierarchie als ganzes wird am übersichtlichsten, wenn man sie als einen einzigen Graphen darstellt. Allerdings ist das auf normalgroßem Papier kaum möglich. Daher werden hier, wie auch bei der ersten Signatur über Fahrzeuge, die Zweige des Gesamtgraphen gesondert angegeben: Zunächst listen wir die unmittelbaren Untersorten der generellen Obersorte *object* alphabetisch auf und geben dann nacheinander die Teilgraphen wieder, die diese Untersorten als Wurzeln haben. Wo dies mit dem gegebenen Platz vereinbar ist, wird sowohl der vollständige Sortenname als auch seine gebräuchlichste Abkürzung genannt.

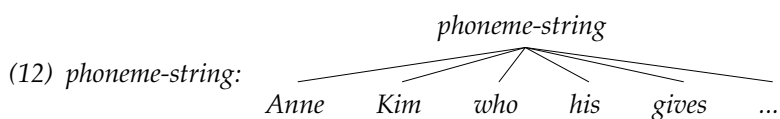
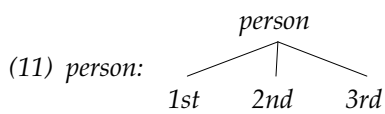
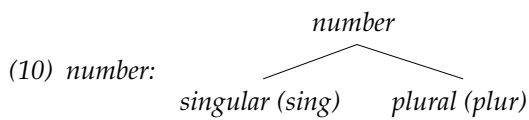
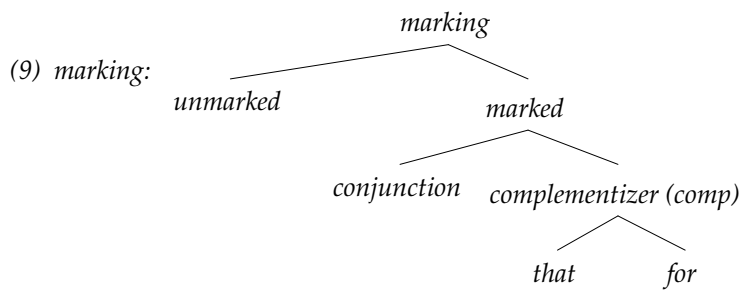
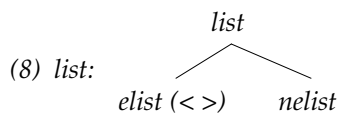
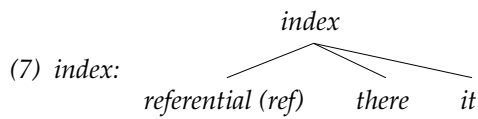
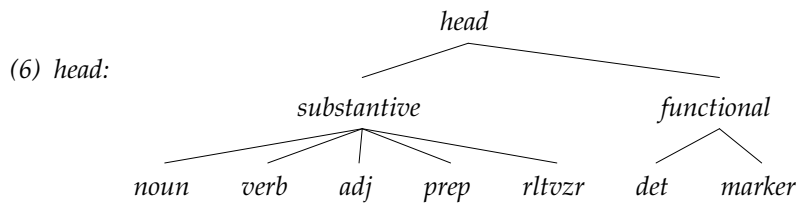
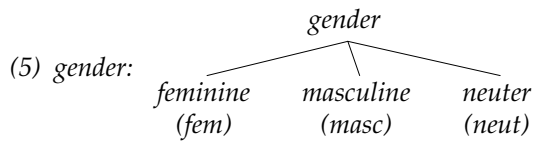
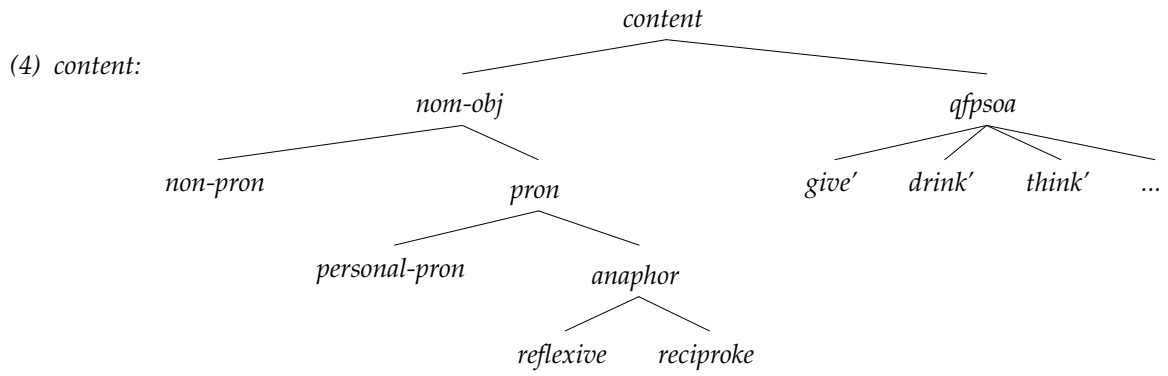
Unmittelbare Untersorten von *object*:

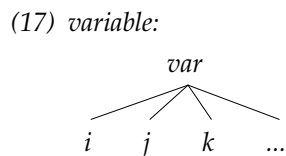
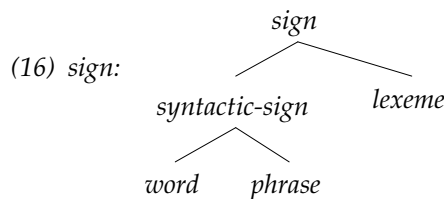
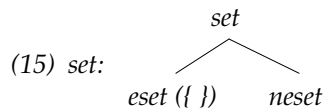
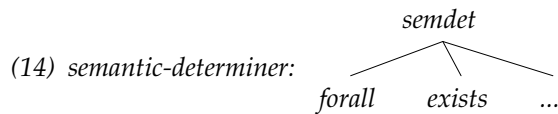
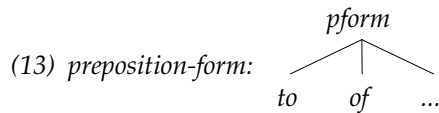
agr, *boolean* (*bool*), *case*, *category* (*cat*), *constituent-structure* (*con-struc*), *content* (*cont*), *context* (*ctxt*), *contextual-indices* (*c-inds*), *gender* (*gend*), *head*, *index* (*ind*), *list*, *local* (*loc*), *marking*, *nonlocal* (*nloc*), *nonlocal1* (*nloc1*), *number* (*num*), *person* (*per*), *phoneme-string* (*phonstring*), *preposition-form* (*pform*), *quantifier-free-parameterized-state-of-affairs* (*qfpsoa*), *semantic-determiner*, *set*, *sign*, *valence*, *variable*, *verbform* (*vform*), *xsel-synsem*.



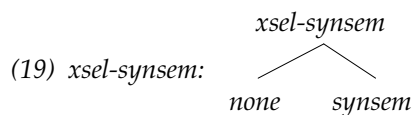
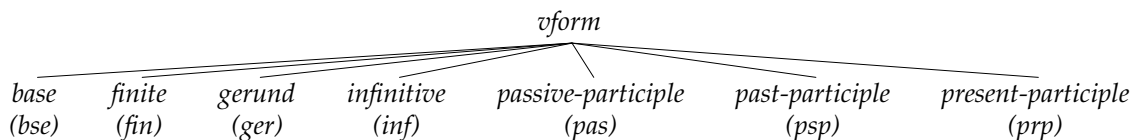
(3) *constituent-structure*:







(18) *verbform*:



1.1 Anmerkungen zur Sortenhierarchie

Im Unterschied zum HPSG-Buch von Pollard und Sag haben wir Listen und Mengen nicht hinsichtlich der Sorte der Elemente parametrisiert, die in ihnen vorkommen dürfen. In den Feature Declarations im nächsten Abschnitt werden wir einfach festlegen, dass Listen und Mengen prinzipiell Objekte beliebiger Sorte enthalten können. In Abschnitt 2.1 kommen wir darauf erläuternd zurück.

In (4) wird *qfpsoa* (für die Semantik von Verben) als unmittelbare Untersorte von *content* behandelt im Unterschied zum Buch, wo an der gleichen Stelle die Sorte *psoa* steht. *psoa* können wir für unsere Belange ignorieren, da wir Quantoren nicht näher betrachten. Außerdem erlauben wir es uns, die umfangreiche (und auch im Buch nur fragmentarisch angedeutete) Hierarchie unter *qfpsoa* wegzulassen. So gäbe es etwa für *geben* eine maximal spezifische Untersorte *give'*, für *trinken* eine Sorte *drink'* etc. Diese haben dann entsprechende Attribute für ihre thematischen Rollen, beispielsweise GIVER, GIVEN und GIFT oder DRINKER und DRUNKEN. Wenn wir diese (und dazu passende Feature Declarations) für unsere linguistischen Überlegungen benötigen, setzen wir deren Existenz einfach voraus. Entsprechendes gilt für die äußerst vereinfacht gehandhabte Phonologie mit Untersorten von *phoneme-string*, sowie für die Untersorten von *pform*. Die Sorte *lexeme* von Sag/Wasow/Bender haben wir in unsere Signatur unter *sign* eingeordnet (16), wobei wir sie von den vorhandenen Sorten für syntaktische Zeichen unterschieden haben. Auf die Darstellung der umfangreichen Untersorten von *lexeme* wird verzichtet (siehe dazu die Seminar-Folien).

2 Feature Declarations

Die Feature Declarations sind im Folgenden alphabetisch angeordnet. Es wird jeweils angegeben, welches die höchste Sorte in der Hierarchie ist, für die ein Attribut angemessen ist, und welchen Wert die Angemessenheitsfunktion für diese Sorte und das jeweilige Attribut annimmt. Zum Beispiel ist *head* die höchste Sorte der *head*-Hierarchie (6), so dass das Attribut XSEL für sie angemessen ist. Der HPSG-Formalismus stellt sicher, dass die Angemessenheit von XSEL auf die Untersorten von *head* vererbt wird. Bei der Beschreibung von linguistischen Zeichen ist *sign* der Ausgangspunkt, deshalb ist diese Sorte besonders gekennzeichnet.

<i>agr</i>	PERSON	<i>person</i>
	NUMBER	<i>number</i>
	GENDER	<i>gender</i>
<i>category</i>	HEAD	<i>head</i>
	SUBCAT	<i>list</i>
	MARKING	<i>marking</i>
Alternative:		
<i>category</i>	HEAD	<i>head</i>
	VALENCE	<i>valence</i>
	MARKING	<i>marking</i>
<i>c-inds</i>	SPEAKER	<i>ref</i>
	ADRESSEE	<i>ref</i>
	UTTERANCE-LOCATION	<i>ref</i>
	...	
<i>context</i>	BACKGROUND	<i>set</i>
	CONTEXTUAL-INDICES	<i>c-inds</i>
<i>coord-struct</i>	CONJ-DTRS	<i>set</i>
	CONJUNCTION-DTR	<i>word</i>
<i>head</i>	XSEL	<i>xsel-synsem</i>
<i>head-adjunct-struct</i>	HEAD-DTR	<i>phrase</i>
	ADJUNCT-DTR	<i>sign</i>
	COMP-DTRS	<i>elist</i>
<i>head-filler-struct</i>	HEAD-DTR	<i>phrase</i>
	FILLER-DTR	<i>sign</i>
	COMP-DTRS	<i>elist</i>
<i>head-marker-struct</i>	HEAD-DTR	<i>phrase</i>
	MARKER-DTR	<i>word</i>
	COMP-DTRS	<i>elist</i>
<i>head-struct</i>	HEAD-DTR	<i>sign</i>
	COMP-DTRS	<i>list</i>
<i>index</i>	VARIABLE	<i>var</i>
<i>local</i>	CATEGORY	<i>category</i>
	CONTENT	<i>content</i>
	CONTEXT	<i>context</i>
<i>nelist</i>	FIRST	<i>object</i>
	REST	<i>list</i>

<i>nom-obj</i>	INDEX RESTR	<i>index</i> <i>set</i>
<i>nonlocal</i>	TO-BIND INHERITED	<i>nonlocal1</i> <i>nonlocal1</i>
<i>nonlocal1</i>	SLASH <i>set</i> REL <i>set</i> QUE <i>set</i>	
<i>noun</i>	CASE AGREEMENT	<i>case</i> <i>agr</i>
<i>phrase</i>	DAUGHTERS	<i>con-struct</i>
<i>preposition</i>	PFORM	<i>pform</i>
<i>quantifier</i>	DET RESTIND	<i>sem-det</i> <i>npro</i>
<i>set</i>	FIRST REST	<i>object</i> <i>set</i>
→ <i>sign</i>	PHONOLOGY SYNSEM	<i>list</i> <i>synsem</i>
<i>syntactic-sign</i>	QSTORE RETRIEVED	<i>set</i> <i>set</i>
<i>substantive</i>	PREDICATIVE	<i>boolean</i>
<i>synsem</i>	LOCAL NONLOCAL	<i>local</i> <i>nonlocal</i>
<i>valence</i>	SUBJ COMPS SPR	<i>list</i> <i>list</i> <i>list</i>
<i>verb</i>	VFORM AGREEMENT AUX INV	<i>vform</i> <i>agr</i> <i>boolean</i> <i>boolean</i>

2.1 Bemerkungen zu den Feature Declarations

2.1.1 Alternative zur SUBCAT-Liste

In weiten Teilen der HPSG-Literatur werden Varianten der Signatur des HPSG-Buches von Pollard und Sag verwendet. Eine prominente Änderung besteht darin, die SUBCAT-Liste in mehrere Valenzlisten aufzuspalten, was bereits im neunten Kapitel des HPSG-Buchs diskutiert wird. Zum einen werden die Angemessenheitsbedingungen für *category* so geändert, dass anstelle von SUBCAT das Attribut VALENCE mit dem neuen Wert *valence* definiert wird. *valence* ist eine maximal spezifische unmittelbare Untersorte von *object*. Zum anderen werden neue Feature Declarations für *valence* eingeführt für Subjekt, Komplemente und Spezifizierer. Es ist klar, dass diese Modifikation unter anderem zu relativ naheliegenden Anpassungen beim Subkategorisierungs-Prinzip und, je nach Ausführung der Idee, bei den ID-Schemata und den Untersorten von *headed-structure* führt.

2.1.2 Externe Selektion

Ein weiterer Unterschied zur Pollard/Sag-Signatur liegt in der Zusammenfassung der beiden Attribute SPEC und MOD zu dem Attribut XSEL, kurz für 'external selection'. Durch dieses Attribut, zusammen mit der Sorte *xsel-synsem* und einem geeigneten Prinzip, welches analog zum SPEC-PRINCIPLE funktioniert, können Nichtköpfe wie beispielsweise Adjektive oder Determinierer (wenn man keine DPs annimmt) ihre Kopftöchter selektieren. So selektiert in der Phrase *blauer Himmel* das Adjektiv *blau* den Kopf *Himmel* über seinen XSEL-Wert.

2.1.3 Parametrisierte Listen und Mengen

Pollard und Sag verwenden nicht einfach Listen und Mengen, wie wir das hier tun, sondern sogenannte parametrisierte Listen und Mengen. So werden zum Beispiel SUBCAT-Listen in ihrer Signatur nicht einfach als von der Sorte *list* angegeben, sondern als *list(synsem)*, das heißt, sämtliche Elemente dieser Listen müssen qua Signatur *synsem*-Objekte sein. Zum einen sind jedoch parametrisierte Listen ein formal gesehen recht kompliziertes Konzept. Zum anderen stellt sich bei näherem Hinsehen außerdem heraus, dass die Verwendung parametrisierter Listen in der Grammatik des Buches weitgehend eine Redundanz darstellt: So wie die Theorie der Grammatik formuliert ist, ergibt sich bereits unabhängig, dass beispielsweise Elemente von SUBCAT-Listen nur *synsem*-Objekte sein können. Aufgrund dieser Gegebenheiten haben wir darauf verzichtet, parametrisierte Listen und Mengen zu verwenden.

2.1.4 Mengen

Die Autoren des HPSG-Buches äußern sich nicht präzise dazu, wie sie sich die Behandlung von Mengen in einer Merkmalslogik vorstellen. So führen sie zwar die Mengensorten *set*, *eset* und *neset* im Appendix ein, sagen aber an anderer Stelle, dass sie diese nicht als Sortensymbole betrachten wollen. Um mit mengenwertigen Merkmalen einfach und dennoch präzise zu arbeiten, haben wir uns dafür entschieden, *set*, *eset* und *neset* als normale Sorten zu behandeln, und haben darüber hinaus der *set*-Hierarchie mit den für die Sorte *neset* deklarierten Attributen FIRST (mit Wert *object*) und REST (mit Wert *set*) geeignete Feature Declarations gegeben. Daneben haben wir für die dadurch eingeführten Mengenobjekte verabredet, dass uns im Unterschied zu den Gegebenheiten bei Listen nicht die Reihenfolge interessiert, in der die Elemente von Mengen in der durch die Interpretation der REST- und FIRST-Attribute gegebenen Anordnung stehen. Außerdem darf ein Element niemals zweimal in einer Menge auftreten und es gibt keine zyklischen Mengen.

2.1.5 Zeichen anstelle von Phrasen als Töchter

Mit der Forderung von Pollard und Sag, dass Adjunkttöchter, Füllertöchter und Komplementtöchter immer Phrasen sein sollen, gibt es Probleme, da aufgrund der gegebenen ID-Schemata viele naheliegende Kandidaten für solche Töchter ausgeschlossen werden: Sie müssten zu Phrasen projizieren, können das aber aufgrund fehlender ID-Schemata nicht. Wir haben eine naheliegende Lösung dieses Problems gewählt, indem wir die Feature Declarations so verändert haben, dass die genannten Töchter sowohl Wörter als auch Phrasen sein dürfen.

2.1.6 Relationen

Der Vollständigkeit halber sei erwähnt, dass eine Signatur nicht nur die Sortenhierarchie, die Attributnamen und die Feature Declarations spezifizieren muss, sondern sie muss auch sämtliche Relationsnamen und die Stelligkeit der einzelnen Relationen angeben (z.B. *append/3*). Relationen werden benötigt, um beispielsweise die geforderten Operationen auf Listen (Elemente hinzufügen oder löschen, Listen miteinander verknüpfen) durchzuführen. Wir werden uns jedoch nicht weiter damit auseinandersetzen und bei Bedarf – wie z.B. auch die Autoren des HPSG-Buchs – immer das Vorhandensein der entsprechenden Relationen voraussetzen.