Hypertextual Appendix
to
The Signs of Determination

http://www.geocities.com/v_allegranza/book2007.html

Valerio Allegranza

v.allegranza@tiscali.it

October 2007

Contents

Preamble

This document is conceived as an external Appendix to my book *The Signs of Determination. Constraint-Based Modelling Across Languages* (published by Peter Lang, Frankfurt am Main, 2007) [1]. Since its inclusion would have enlarged that volume beyond the size agreed upon with the publisher, it is offered here in electronic form for readers interested in a systematic summary of grammatical technicalities coming up at different points in the book proper. Moreover, I have adopted an hypertextual organization for the new document (exploiting PDF), so as to allow easy and fast navigation across its sectional units. In the Adobe Reader, you can click them as Bookmarks (corresponding to the Table of Contents) within the navigation pane on the left side of the window, but throughout the running text you will find many other hypertextual links (distinguished by blue colour and underlining).

Considering that the constraint-based grammatical approach developed in *The Signs of Determination* leads to a revised version of the Head-driven Phrase Structure Grammar by Pollard and Sag (P&S-94)[2], it should be hardly surprising that the primary source of inspiration for what I have done here – hypertextuality aside – was a similar Appendix annexed to their book. Thus there are two main parts, the former presenting the 'Sort Hierarchy' of such a grammar and the latter a number of grammatical statements which I call 'Path Constraints'. Related sorts of the hierarchy are grouped into sections (e.g., **4. Parts of speech**), whose subsections are devoted to a single sort each (e.g., **4.2 *major***); you can view and click Bookmarks for the various sorts in the navigation pane of the Adobe Reader by expanding top-level Bookmarks with ⊞.

The following conventions will be applied when presenting a given sort:

✎ ***subsort***$_1$, ... ***subsort***$_n$ displays the sort's direct subsorts (if any),
☞ [$\mathcal{A}_1\mathcal{V}_1$, ... $\mathcal{A}_m\mathcal{V}_m$]   displays its feature declarations (if any).

As in P&S-94, declarations consist of 'appropriateness conditions' on features qua attribute-value pairs, showing only the ***sort*** of values $\mathcal{V}_i$ appropriate for

---

[1]Vol. 16 of *SABEST – Saarbrücker Beiträge zur Sprach- und Translationswissenschaft*, series edited by A. Gil, J. Haller and E. Steiner from the Department of Applied Linguistics, Translation and Interpreting of the Saarland University (Saarbrücken, Germany).

[2]C. Pollard and I. Sag (1994) *Head-driven Phrase Structure Grammar*. Stanford: CSLI Publications / Chicago: Chicago University Press.

a feature attribute $\mathcal{A}_i$.[3] Hence the difference from so-called 'path constraints' allowing arbitrarily long paths of feature structure as well as token-identities of values, in order to encode grammatical principles or phrasal schemata as the second part of the document will exemplify. Even though the constraints are written 'in prose' for notational convenience (again as done by `P&S-94`), I refer to my book for strictly corresponding encoding in terms of the original HPSG formalism of sorted feature structures with logical connectives[4].

To make immediately evident how the version of HPSG I developed in *The Signs of Determination* relates to and departs from the 'standard' `P&S-94` version, summarized in the Appendix to their book, here statements of direct subsorts, feature declarations and path constraints will be written against differently coloured types of background. (For neatly readable text in case of coloured background, set the Adobe Reader's Zoom precisely to 100%.)

Legend:

XYZ = green background for substantial novelties of the system proposed in my book (and, to some extent, in earlier related publications[5]);

XYZ = white background with green frame for notions that are the same as in the `P&S-94` Appendix but differently located within my system;

XYZ = yellow background for other differences from the standard version (like substantial changes already proposed in Ch. 9 of `P&S-94` or other well-known literature, or purely terminological changes, etc.);

XYZ = just white background when there are no differences.

Many statements will be briefly commented on, but clearly the comments do not aim at being self-contained, given the function of this document as a mere supplement. (References to pages of my book where the differences/novelties are found will be introduced by the 📖 symbol.)

V. A. – October 2007

---

[3]For this and other formal aspects of the relevant sort hierarchies, see B. Carpenter (1992) *The Logic of Typed Feature Structures*. Cambridge: Cambridge University Press.

[4]From Ch. 2 of C. Pollard and I. Sag (1987) *Information-based Syntax and Semantics*. Stanford: CSLI Publications.

[5]Above all, V. Allegranza (1998) "Determiners as Functors: NP Structure in Italian", in S. Balari and L. Dini (eds.), *Romance in HPSG*. Stanford: CSLI Publications. Cf. also F. Van Eynde (1998) "The Immediate Dominance Schemata of HPSG", in P.-A. Coppen, H. Van Halteren and L. Teunissen (eds.), *Computational Linguistics in the Netherlands, Selected Papers from the 8th CLIN Meeting*. Amsterdam: Rodopi.

► **Part A)** ◄

Sort Hierarchy:

---

1.   Generalities

1.1   *object*

✎ *list**, *set**, *boolean*, *sign*, <mark>**phonology (phon)**</mark> , <mark>**morphology (morph)**</mark> , <mark>**daughters (dtrs)**</mark> , <mark style="background:lime">**head-slot**</mark> , *local*, *nonlocal*, *category* (***cat***), <mark>**part-of-speech (p-o-s)**</mark> , *marking*, <mark>**valence**</mark> , *case*, *verbform* (***vform***), *preposition-form* (***pform***), <mark style="background:lime">**semantic-head (semhead)**</mark> , *content* (***cont***), *context*, *contextual-indices* (***c-inds***), *quantifier-free-parametrized-state-of-affairs* (***qfpsoa***), <mark style="background:lime">**range-of-reference-typology (rrt)**</mark> , *index*, *person* (***pers***), *number* (***numb***), *gender* (***gend***), ...

Comment: The hierarchy's unique most general sort is labelled *object* according to P&S-94. The direct subsorts of *object* enumerated here are presented in the rest of the current section "Generalities" when they do not just fall under the heading of one of the thematically motivated sections (from "Signs" to "Content-related sorts"). Moreover, nothing will be said on *phonology* and *morphology* apart from their relevance for the declaration of homonymous features (in section 2).

1.2   *list/set**

✎ *nonempty-list/set** (***nelist/neset****), *empty-list/set* (***elist/eset***)

1.3   *nelist**

☞ [FIRST *, REST ***list****]

Comment: For notational convenience, in this document the parametrization of (non-empty) lists and sets is indicated by a superscript and the asterisk is used as a variable to be replaced by the relevant sort name in specific cases – e.g. *list^phrase*, *set^quant*.

1.4   *boolean*

✎ *plus* (+), *minus* (−)

1.5   *head-slot*

✎ | *syntax-semantics* (*synsem*), *none* |

📖 pp. 262-3

Comment: The sort *head-slot*, which replaces *mod-synsem* of P&S-94 and keeps its subsorts, will label the value of the homonymous feature <u>HEAD–SLOT</u>. (This is used for the cases where 'standard HPSG' would apply one of the features MOD or SPEC, abolished in the new system.)

1.6   *synsem*

☞ [LOCAL *local*, NONLOCAL *nonlocal*]

Comment: The *nonlocal* values with their internal features will not be presented here; we refer to P&S-94 for the standard HPSG treatment.

1.7   *local*

☞ [CATEGORY *cat*, SEMHEAD *semhead*, CONTEXT *context*]

📖 p. 287

Comment: The novel feature SEMHEAD is declared for *local* instead of <u>CONTENT</u>, which gets embedded one level deeper into the feature structure as shown by the next subsection.

1.8   *semhead*

☞ [CONTENT *cont*, QUANTIFIER-STORE (QSTORE) set^{quant}]

📖 pp. 286-7

Comment: Both CONTENT and QSTORE come from P&S-94, but in the new system they are grouped together as *semhead* features to account for their parallel behaviour (a result of purely lexical retrieval of quantifiers, cf. § <u>8.5</u>).

1.9 *context*

☞ [BACKGROUND *set^{psoa}*, CONTEXTUAL-INDICES *c-inds*]

1.10 *contextual-indices*

☞ [SPEAKER *ref*, ADDRESSEE *ref*, NAMING *ref*, ...]

📖 pp. 306-7

Comment: The NAMING feature takes as value a contextual index accounting for the acquisition of the intended name reference, in a sense reminiscent of Kripke's causal theory of proper names[6].

2. Signs

📖 pp. 251-2 (with the exceptions indicated below)

2.1 *sign*

✎ *lexical-sign* (*lex-sign*) , *phrase*

☞ [PHONOLOGY *phon* , SYNSEM *synsem*, ...]

2.2 *lex-sign*

✎ *pure-word* (*word*), *lexical-phrase* (*lex-phrase*)

☞ [MORPHOLOGY *morph* , RETRIEVED *list^{quant}* , POOL *set^{quant}*, ...]

📖 pp. 286, 288 (for RETRIEVED and POOL)

Comment: The sort label *lexical-sign* comes from the early version of HPSG by Pollard & Sag (1987). It is revived in this system to identify lexical items (listed as primitive in the lexicon or derived by lexical rules), irrespective of their bar level under a flexible

---

[6]Cf. S. Kripke (1972) "Naming and Necessity", in G. Harman and D. Davidson (eds.), *Semantics of Natural Language*. Dordrecht: Reidel.

version of X-bar Theory. That is, not every lexical sign occurs as a bar-less, 'level zero' syntactic unit, conventionally labelled *pure-word* (or just *word* for convenience). Note also the declaration of RETRIEVED, a *sign* feature in P&S-94, which gets shifted to *lex-sign* here because quantifier retrieval is meant to be lexical only; on that issue and the ancillary feature POOL, see §8.5.

2.3   *phrase*

✎ ***lex-phrase***, ***construction*** (***constr***)

☞ [DAUGHTERS ***dtrs***]

Comment: In this system, a syntactic unit at a 'non-null bar level' is called *phrase* as in P&S-94, but can be either lexical or a *construction* combining syntactic units. The purpose of *lexical-phrase* – common subsort of *lexical-sign* and *phrase* in a hierarchy with 'multiple inheritance' *à la* Carpenter (1992) – is to account for those phrases that consist of a single lexical item, without building any one-branch constituent structure.[7]

2.4   *lex-phrase*

☞ [DAUGHTERS ***no-structure*** (***no-struc***)]

Comment: The lack of constituent structure in a lexical phrase is explicitly indicated by the atomic value *no-struc* of DTRS (instead of just omitting the feature), so as to allow an easier statement of generalizations.

2.5   *construction*

☞ [DAUGHTERS   ***constituent-structure*** (***con-struc***)]

Comment: Since constituent structure pertains to constructions as phrases that are not lexical, its sort is introduced by the declaration of DTRS relative to *construction*, not *phrase* as in P&S-94.

---

[7]For an alternative approach assuming one-branch constructions, but compatible with the 'functor'-based analysis of §8.3 below (or previous publications already known to the authors), see A. Branco and F. Costa (2006) "Noun Ellipsis Without Empty Categories", paper presented at the 13th International Conference on Head-Driven Phrase Structure Grammar (HPSG 2006), Varna, Bulgaria.

3.   Daughter structures

3.1   *dtrs*

✎ *con-struc*, **complement-saturation-structure (compsat-struc)**

3.2   *compsat-struc*

✎ **head-complement-structure (head-comp-struc)** , **no-struc**

📖 pp. 252 (for *dtrs* too), 268

Comment: The sort *dtrs* has been introduced as a supersort of *constituent-structure* to allow also the option *complement-saturation-structure*, which is relevant to phrases with *head-complement-structure* or *no-structure* as regards constituency (see § 9.1 for explanations).

3.3   *con-struc*

✎ **headed-structure (head-struc)**, **coordinate-structure (coord-struc)**, ...

Comment: As in P&S-94 and much other HPSG literature, only 'headed' structures are treated, while at least coordinate structures are assumed to be head-less and left aside for future extensions of the system.

3.4   *head-struc*

✎ **head-argument-structure (head-arg-struc)**, **head-functor-structure (head-functor-struc)** , ...

☞ [HEAD-DTR **sign**, COMP-DTRS **list**^{phrase}, SUBJ-DTR **list**^{phrase}]

📖 pp. 252-3, 266

Comment: Note the introduction of two novel subsorts interposing between *headed-structure* and more specific sorts in the hierarchy. (Further subsorts may well be called for, like *head-filler-structure* in the usual HPSG treatment of nonlocal dependencies, neglected here.) The feature SUBJ-DTR was already adopted in Chapter 9 of P&S-94, but not in the 'standard' HPSG system as summarized by the Appendix to their book.

3.5   *head-arg-struc*

✎   [*head-comp-struc*] , **head-subject-structure** (**head-subj-struc**)

📖   pp. 266

Comment: Exploiting Carpenter's (1992) approach to sort hierarchies, *head-comp-struc* turns out to be common subsort of *head-arg-struc* and *compsat-struc* (cf. § 3.2).

3.6   *head-comp-struc*

☞   [HEAD-DTR **word**, SUBJ-DTR **elist** , COMP-DTRS **nelist**]

📖   pp. 254, 268

Comment: Due to the ban on one-branch structures (v. § 2.3), any head-complement structure has some complement daughter(s); hence a *nonempty-list* value of COMP-DTRS. The other declarations for *head-comp-struc* follow Chapter 9 of P&S-94.

3.7   *head-subj-struc*

☞   [HEAD-DTR **phrase**, COMP-DTRS **elist**]

📖   pp. 254, 268

Comment: In addition, the feature SUBJ-DTR is inherited from the supersort *head-struc* (see § 3.4). To complete the original declaration of *head-subject-structure* as in Chapter 9 of P&S-94, SUBJ-DTR should be restricted to lists with a single element, but § 9.2 will show how the restriction obtains independently.

3.8   *head-functor-struc*

✎   [**head-adjunct-structure** (**head-adj-struc**)] , **head-specifier-structure** (**head-spr-struc**)

☞   [HEAD-DTR **phrase**, SUBJ-DTR **elist**, COMP-DTRS **elist**, FUNCTOR-DTR **sign**]

📖   pp. 260, 266, 268-9

Comment: A crucial point. Readers are referred to § 8.3 for detailed explanations.

4.   Category-related sorts

📖 pp. 263, 265 (with the exceptions indicated [below](#))

4.1   *category*

✎ *saturated-category* (*satcat*), ...

☞ [HEAD **p-o-s** , VALENCE **valence** , MARKED **marking**]

Comment: Apart from *satcat*, the subsorts of *category* remain to be detailed (e.g., depending on whether a 'lexical vs. nonlexical' distinction should be encoded at the level of categories too). The feature VALENCE with the homonymous sort of values was not mentioned by P&S-94 but seems useful for a Valence Principle formulation as in Chapter 9 of their book (cf. §[8.2](#)) and is often found in subsequent HPSG literature. Concerning MARKED, the terminological change compared with the original attribute name MARKING of P&S-94 is motivated by the fact that the present system includes also a distinct feature with values of sort *marking*, viz. [MARKER](#), since a constituent marking a phrasal node may well be a phrase that in turn gets differently marked by some internal constituent.

4.2   *satcat*

☞ [VALENCE **saturated**]

Comment: Given the feature declarations for *saturated* (§[4.4](#)), *satcat* will be cited in my principle of §[8.3](#) to impose saturation on the FUNCTOR-DTR, whether phrasal or [*minor*](#). (One could do without such a detailed sortal set-up by showing VALENCE and its feature structure in a representationally enlarged version of the principle.)

4.3   *valence*

✎ *saturated*, ...

☞ [SUBJ **list**^{synsem}, COMPS **list**^{synsem}]

Comment: As these two valence features from Chapter 9 of P&S-94 give rise to four combinations depending on whether the lists are empty or not, corresponding subsorts of *valence* would adequately complete the partition, although only *saturated* has been explicitly mentioned.

4.4   *saturated*

☞ [SUBJ *elist*, COMPS *elist*]

4.5   *marking*

✎ *marked*, *unmarked*

4.6   *marked*

✎ *determination-marking (detmark)*, ...

☞ [SPECIFIER (SPR) *boolean*]

📖 also pp. 249, 272

Comment: The feature SPR, indicating 'maximal bar level' on phrases ([SPR+]) or else the need for a specifier ([SPR–]), has undergone considerable changes compared with the original from Chapter 9 of P&S-94, even if the attribute name remains the same for familiarity's sake. Instead of a valence feature, *list*-valued, a simpler feature with *boolean* values (reminiscent of Netter's FCOMPL[8]) is used as part of the marking mechanism in this system. Among the various areas that look potentially relevant for fine-grained marking, the present work covers determination (as shown in section [6]).

5.   Parts of speech

📖 pp. 30-1, 68-70 (with the exceptions indicated below)

5.1   *p-o-s*

✎ *major*, *minor*

☞ [MARKER *marking* , HEAD-SLOT *head-slot*]

📖 p. 263 (for MARKER and HEAD-SLOT)

---

[8]From K. Netter (1994) "Towards a Theory of Functional Heads: German Nominal Phrases", in J. Nerbonne, K. Netter and C. Pollard (eds.), *German in HPSG*. Stanford: CSLI Publications.

Comment: The distinction between *major* and *minor* 'parts of speech' (i.e. morpho-syntactic classes of lexical items) is frequently found in the literature, but not always with exactly the same meaning. Following the GPSG book by Gazdar et al. [9], here the intended sense is that minor words do not project phrase structure (bar levels), unlike major ones. In the present grammar the result can be achieved by lexically encoding any *minor* item – e.g. article, complementizer, etc. – as a *pure-word* (from §2.2) fully saturated via *satcat*; these two conditions prevent the item from occurring as *lexical-phrase* and/or combining with dependents. Features MARKER and HEAD–SLOT are applied in the way shown by §8.3.

5.2   *major*

✎ *noun*, *verb*, *preposition*, *adjective*

☞ [N *boolean*, V *boolean*, ...]

Comment: There are precisely four *major* parts of speech, owing to their traditional analysis in terms of *boolean*-valued features N and V borrowed from the X-bar theoretic literature (and GPSG). Such additional features as CASE, VFORM and PFORM for the respective subsorts of *major* can be introduced as shown hereafter (without getting into language-specific subtyping of values).

5.3   *noun*

☞ [N *plus*, V *minus*, CASE *case*, ...]

5.4   *verb*

☞ [N *minus*, V *plus*, VFORM *vform*, ...]

5.5   *preposition*

☞ [N *minus*, V *minus*, PFORM *pform*, ...]

5.6   *adjective*

☞ [N *plus*, V *plus*, ...]

---

[9]G. Gazdar, E. Klein, G. Pullum and I. Sag (1985) *Generalized Phrase Structure Grammar*. Oxford: Basil Blackwell.

6. Determination marking

📖 pp. 272-80

6.1 *detmark*

✎ *switch-determination* (*switch-det*), *specifierless-determination* (*specless-det*)

☞ [QUARK *quark*, OMAR *boolean*, POLIS *list^{synsem}*]

Comment: In the typical cases of determination marking, a determiner attached to a nominal mother node marks this node for some feature value while demanding that the adjacent nominal head be marked for an incompatible value of the same feature. The features concerned are SPR (which is inherited from the supersort *marked* of *detmark*) and those allowing fine-grained determination marking: QUARK in quantitative (and cardinal) marking, OMAR in ordinal marking and the *list*-valued POLIS in a marking-based treatment of possessives. Readers are referred to the book for detailed discussion and exemplification of the use of such features in pertinent lexical entries of English and Italian.

6.2 *switch-det*

✎ *leftmarking-determination* (*leftmark-det*), *inner-determination* (*inner-det*)

Comment: Hierarchically, *switch-det* interposes between *detmark* and the 'terminal' sorts *leftmark-det* and *inner-det*, so as to exclude only *baretype-det*. Thus it is useful for those determiners that are lexically ambiguous w.r.t. their MARKER|SPR value.

6.3 *specless-det*

✎ *inner-det*, *baretype-determination* (*baretype-det*)

Comment: The *specifierless-determination* marking on a phrase with nominal category implies the absence of any internal specifier, whether the phrase calls for an external one (in the case of *inner-det*, the subsort in common with *switch-det* within a Carpenter-style hierarchy) or rejects specifiers altogether (see *baretype-det*).

6.4   *leftmark-det*

☞ [SPR *plus*]

Comment: The label *leftmarking-determination* is meant to distinguish those [SPR+] nominal signs that cannot be modified by a qualifying adjective on their left.

6.5   *inner-det*

☞ [SPR *minus*]

Comment: The label *inner-determination* for [SPR–] nominal signs means that these can only occur inside some broader nominal construction, at least because they call for an external specifier.

6.6   *baretype-det*

☞ [SPR *plus*]

Comment: The label *baretype-determination* applies to 'bare plurals' and the other self-determining [SPR+] nominal signs which do not fall under *leftmark-det*.

7.   Content-related sorts

7.1   *content*

✎ *parametrized-state-of-affairs* (*psoa*), *quantifier* (*quant*), *nominal-object* (*nom-obj*)

7.2   *psoa*

☞ [QUANTIFIERS (QUANTS) $list^{quant}$, NUCLEUS *qfpsoa*]

7.3   *qfpsoa*

📖 pp. 159-60

Comment: The detailed hierarchy of subsorts of *qfpsoa* ('quantifier-free parametrized state of affairs' in P&S-94 jargon) and their feature declarations, being an open question for HPSG, are accordingly left aside here. Suffice it to give only few possible examples:

book   👉 [INSTANCE *ref*] $\simeq$ *book(x)*

eat   👉 [AGENT *ref*, THEME *ref*] $\simeq$ *eat(x,y)*

give   👉 [AGENT *ref*, THEME *ref*, GOAL *ref*] $\simeq$ *give(x,y,z)*

7.4   *quant*

👉 [DETERMINATION-RANGE (DET) *rrt*,

RESTRICTED-INDEX (RESTIND) *nom-obj*]

📖 pp. 173, 308

Comment: Although a *quantifier* is structured essentially in the same way as proposed by P&S-94 [10], there are terminological and conceptual differences. Here DET does not abbreviate DETERMINER, because the semantic contribution to be encoded need not correspond to a determiner (e.g., self-determining nominal words have a DET value as well). The concept of 'range of reference typology' – in the sense made clear below – motivates the *rrt* sort of values appropriate for DET and also the attribute name DETERMINATION-RANGE which DET abbreviates. As to the value of RESTIND, it is simply typed by the sort *nominal-object*, instead of the more specific *nonpronoun* (cf. § 7.7), because pronouns too allow a treatment as quantifiers.

7.5   *rrt*

✎ *universal* (*univ*), *free-domain*

👉 [DEFINITE (DEF) *boolean*]

7.6   *free-domain*

👉 [DEF *minus*, EXISTENTIAL (EXIST) *boolean*]

For both §§ 7.5 – 7.6 **:**

📖 pp. 123-5, 165-7, 173-4

Comment: The semantic notion of 'determination' developed in my book concerns the type of reference conveyed to the Noun Phrase through relevant dependents of the

---

[10]Cf. J. Barwise and R. Cooper (1981) "Generalized quantifiers and natural language", *Linguistics and Philosophy*, 4, pp. 159-219.

nominal head, the determiners, or self-determined by proper names, pronouns, bare plurals, etc. The values of sort *range-of-reference-typology* (*rrt*) for the DET feature of a *quantifier* (cf. § 7.4) are interpreted so as to yield *universal*, [EXIST+] (existential) or [EXIST–] (negated existential) reference, according to how it ranges over the domain of quantification. In the *universal* case a presupposition of definiteness can be imposed on the domain through the feature [DEF+], while the domain w.r.t. EXIST cases remains presupposition-free in that respect (hence [DEF–] for *free-domain*). See in *The Signs of Determination* a comprehensive discussion on such notions and how to connect them with mainstream logical semantics and a compatible view of pragmatics.

7.7   *nom-obj*

✎ *nonpronoun* (*npro*), *pronoun* (*pron*)

☞ [INDEX *index*, RESTRICTION (RESTR) *set^{psoa}*]

Comment: There is no difference from P&S-94 in the subsorts and feature declarations for *nominal-object*, but according to the semantics adopted in my book, various kinds of determiners can also contribute a *psoa* to the RESTR set, as P&S-94 already do for possessives by assuming something like ***poss*(*x*,*y*)**. A cardinal numeral $\mathcal{N}$, e.g., will contribute the cardinality condition ***card*$\mathcal{N}$(*x*)** on the index *x* for a 'plural entity'.[11]

7.8   *index*

✎ *referential* (*ref*), ...

☞ [PERSON *person*, NUMBER *number*, GENDER *gender*]

7.9   *person*

✎ *1st*, *2nd*, *3rd*

7.10   *number*

✎ *singular* (*sing*), *plural* (*plur*)

7.11   *gender*

✎ *masculine* (*masc*), *feminine* (*fem*), *neuter* (*neut*)

---

[11]See also D. Kolliakou (1996) "Definiteness and the Make-up of Nominal Categories", in C. Grover and E. Vallduví (eds.), *Studies in HPSG*, vol. 12 of *Edinburgh Working Papers in Cognitive Science*. University of Edinburgh.

▶ **Part B)** ◀

Path Constraints:

---

8.   Some Universal Principles

8.1   Head Feature Principle (HFP)

A **phrase** with DTRS value of sort **head-struc** has SYNSEM | LOCAL | CAT | HEAD value token-identical with the DTRS | HEAD-DTR | SYNSEM | LOCAL | CAT | HEAD value.

Comment: The HFP remains the same as in P&S-94.

8.2   Valence Principle

In a **phrase** with DTRS value of sort **head-struc**, for each **valence** feature $\mathcal{F}$: the value of DTRS | HEAD-DTR | SYNSEM | LOCAL | CAT | VALENCE | $\mathcal{F}$ is the concatenation of the phrase's SYNSEM | LOCAL | CAT | VALENCE | $\mathcal{F}$ value with the list consisting of the SYNSEM values (and preserving the order) of the elements on the list value of DTRS | $\mathcal{F}$-DTR(S), where $\mathcal{F}$-DTR(S) stands for the **head-struc** feature corresponding to $\mathcal{F}$.

📖 pp. 255-6

Comment: This metagrammatical statement of the Principle looks essentially the same as in Ch. 9 of P&S-94; but once the present version of the object grammar is considered, the metavariable $\mathcal{F}$ gets restricted to the features COMPS and SUBJ by the declaration of _valence_. (The status of SPR is quite different, cf. §4.6.) Hence the double grammatical constraint that follows by instantiation of $\mathcal{F}$.

8.3   Functor Application Principle (FAP)

In a **phrase** with DTRS value of sort **head-functor-struc**:

(a) the DTRS | FUNCTOR-DTR | SYNSEM | LOCAL | CAT value is of sort **satcat**

with MARKED|SPR value ***plus*** and HEAD|HEAD-SLOT value token-identical with the phrase's DTRS|HEAD-DTR|SYNSEM value, and
(b) the phrase has SYNSEM|LOCAL|CAT|MARKED value token-identical with the DTRS|FUNCTOR-DTR|SYNSEM|LOCAL|CAT|HEAD|MARKER value and SYNSEM|LOCAL|SEMHEAD value token-identical with the DTRS|FUNCTOR-DTR|SYNSEM|LOCAL|SEMHEAD value.

📖 pp. 260-6, 287-8

Comment: The FAP combines a FUNCTOR-DTR with a HEAD-DTR whose SYNSEM value is selected by the former constituent, which also acts as marker and semantic head of the resulting phrase. Indeed the notion of 'functor' is motivated by the fact that this constituent has some selection and projection properties – stated in (a) and (b) respectively – even though it is not the syntactic head. (The HEAD value of the resulting phrase comes from the HEAD-DTR via HFP.) Note that the declaration of FUNCTOR-DTR in §3.8 just calls for a *sign*, but the position is reserved for 'minor words' or maximal phrase projections, its category being constrained by the FAP in terms of saturation (*satcat*) and [SPR+] markedness. Moreover, the partition of *head-functor-struc* into two subsorts, *head-spr-struc* and *head-adj-struc* (see again §3.8), implies that functors can be specifiers or adjuncts, depending on whether they fit the Head-Specifier Schema or the Head-Adjunct Schema.

8.4   Unselected Head Principle (UHP)

A ***phrase*** with DTRS value of sort ***head-arg-struc*** has SYNSEM|LOCAL|CAT|MARKED value token-identical with the DTRS|HEAD-DTR|SYNSEM|LOCAL|CAT|MARKED value and SYNSEM|LOCAL|SEMHEAD value token-identical with the DTRS|HEAD-DTR|SYNSEM|LOCAL|SEMHEAD value.

📖 pp. 266-7, 287-8

Comment: This constraint resembles part (b) of the FAP, but there is no FUNCTOR-DTR according to the declaration of *head-arg-struc* in §3.5 and the resulting phrase shares the MARKED value of the unselected syntactic head, which acts as semantic head too. The selection of arguments (SUBJ-DTR or COMP-DTRS) by HEAD-DTR already follows from the Valence Principle in case of *head-arg-struc*, so that the UHP does not include any selection part comparable to (a) of the FAP.

8.5 Quantifier Retrieval Constraint (QRC)

In a **_lexical-sign_**, the RETRIEVED value is a list whose set of elements is $\mathcal{R}$, a subset of the set value of POOL, and the SYNSEM | LOCAL | SEMHEAD | QSTORE value is the set of all those elements of the POOL set that do not belong to $\mathcal{R}$.

📖 pp. 288-9

Comment: Contrary to the original version due to Pollard and Yoo [12], here the QRC is restricted to lexical signs. It is up to the encoding of specific lexical entries to show how input quantifiers are collected from values of *valence* features or HEAD-SLOT, so as to form the POOL value, and how quantifiers on the RETRIEVED list are introduced into the CONTENT value. (Though retrieved quantifiers will always get to a QUANTS list, this may be embedded differently depending on whether the CONTENT value is of sort *psoa* or not.) While referring to the book for detailed discussion and exemplification of such a purely lexical retrieval of quantifiers, there is an important consequence to be observed: phrase structure simply preserves the results of quantifier storage and/or retrieval by the semantic head (see the token-identities of SEMHEAD values in FAP and UHP).[13]

9.   Main ID Schemata

📖 pp. 267-9

In this system, phrasal schemata of Immediate Dominance (ID) comparable to X-bar Theory and elaborating on the original versions proposed in Chapter 9 of P&S-94 are formally analyzed into two components: **a.** the feature declarations for a suitable sort of DTRS value; **b.** a 'path constraint' on any phrase with DTRS value of that sort. Only **b.** will be explicitly stated in the current section, but readers are reminded that an integral part of each schema is information which accrues to it from declarations stated in section 3 of the sort hierarchy and not repeated below.

---

[12]See C. Pollard and E. J. Yoo (1998) "A Unified Theory of Scope for Quantifiers and *Wh*-Phrases", *Journal of Linguistics*, 34, pp. 415–445.

[13]On the other hand, the behaviour of *wh*-phrases becomes quite different from that of quantifiers and calls for some nonquantificational interpretation, a conclusion in line with recent research in the field: cf. J. Ginzburg and I. Sag (2001) *Interrogative Investigations: The Form, Meaning and Use of English Interrogatives.* Stanford: CSLI Publications.

9.1   (Generalized) Head-Complement Schema

A **phrase** with DTRS value of sort **head-comp-struc** (more generally: **compsat-struc**) has an empty list as SYNSEM | LOCAL | CAT | VALENCE | COMPS value.

Comment: By joint application of the declarations for *head-comp-struc* in § 3.6 and the Valence Principle, the HEAD-DTR must be a *pure-word* with a nonempty list as SYNSEM | LOCAL | CAT | VALENCE | COMPS value. Then neither a *minor* lexical item (whose VALENCE is *saturated* by definition) nor a *phrase* can head this construction. But the constructional version proper just instantiates a more general constraint, also valid for 'lexical phrases' (§ 2.4), thanks to the supersort of *head-comp-struc* labelled *compsat-struc* (§ 3.2).

9.2   Head-Subject Schema

A **phrase** with DTRS value of sort **head-subj-struc** has exactly one element on the list occurring as DTRS | SUBJ-DTR value.

Comment: Here the fact that the SYNSEM | LOCAL | CAT | VALENCE | SUBJ value of the resulting phrase must be an empty list follows from the Valence Principle and the lexical coding of both verbs and predicative nonverbs, because their entries describe exactly one element on the SUBJ list for purposes of syntactic/semantic subcategorization. Note that the present version of the Schema makes a restriction of SUBJ-DTR to *nelist* superfluous in the declaration of *head-subj-struc*, whereas this restriction would be required to avoid one-branch 'looping' with an alternative version of the Schema, specifying an empty list as SYNSEM | LOCAL | CAT | VALENCE | SUBJ value but not the element on the DTRS | SUBJ-DTR list.

9.3   Head-Specifier Schema

A **phrase** with DTRS value of sort **head-spr-struc** has SYNSEM | LOCAL | CAT | MARKED | SPR value **plus** and DTRS | HEAD-DTR | SYNSEM | LOCAL | CAT | MARKED | SPR value **minus**.

Comment: Given the Functor Application Principle and the fact that *head-spr-struc* inherits the feature declaration of *head-functor-struc* in § 3.8, this Schema represents

the construction in which a FUNCTOR-DTR acts as specifier, i.e. the kind of dependent that marks phrases as [SPR+] while selecting heads marked as [SPR–]. According to the interpretation of SPR in terms of X-bar Theory, the specifier combines with a one-bar head so as to yield a maximal, two-bar projection.

9.4   Head-Adjunct Schema

A ***phrase*** with DTRS value of sort ***head-adj-struc*** has SYNSEM | LOCAL | CAT | MARKED | SPR value token-identical with the DTRS | HEAD-DTR | SYNSEM | LOCAL | CAT | MARKED | SPR value.

Comment: Given the Functor Application Principle and the fact that *head-adj-struc* inherits the feature declaration of *head-functor-struc* in §3.8, this Schema represents the construction in which a FUNCTOR-DTR acts as adjunct, i.e. the kind of dependent that marks phrases with the same SPR value as the heads it selects. In X-bar Theory terms, adjunction is bar-preserving.

References

These references were specified in full by the respective footnotes:

Allegranza (1998)   fn. 5
Barwise and Cooper (1981) fn. 10
Branco and Costa (2006)   fn. 7
Carpenter (1992)   fn. 3
Gazdar et al. (1985)   fn. 9
Ginzburg and Sag (2001)   fn. 13
Kolliakou (1996)   fn. 11
Kripke (1972)   fn. 6
Netter (1994)   fn. 8
Pollard and Sag (1987)   fn. 4
Pollard and Sag (1994)   fn. 2
Pollard and Yoo (1998)   fn. 12
Van Eynde (1998)   fn. 5

But you can find many more references in *The Signs of Determination*, whose Bibliography is 11 pp. long.