

## Towards meaningful criteria for data structure and grammar design in HPSG-based implementation efforts

**Introduction** For parallel development of grammars for different languages, clearly articulated criteria for determining the nature of the data structures to be used and guidelines for the organization of the grammatical constraints are an essential prerequisite. In this abstract we want to address the issue of developing such criteria for HPSG-based grammar implementation and present some questions which we believe need to be answered to obtain a well-motivated standard practice in multi-lingual grammar development. The questions pertain to where information should be encoded and how it should be organized in order to obtain a compact and transparent grammar. In the context of HPSG-based grammar implementation, this includes the following questions: What motivates introducing a type in a grammar? When should attributes be introduced? How should descriptions used throughout the grammar be organized compactly?

While the design of each grammar implementation platform makes particular assumptions as to how these questions could be answered, there is little explicit discussion of these assumptions. We believe that a thorough discussion of the questions is essential for guiding the further development of implementation platforms towards a well-motivated implementation standard for HPSG-based grammars.

As a contribution to this discussion, this abstract presents an experiment in which we contrast the use of types in the English Resource Grammar (Flickinger et al., 2000, ERG) with a more conservative use of types in a recoding of the ERG that we have undertaken. Our experiment shows that it is possible to eliminate two thirds of the types that are defined in the ERG without changing the meaning of the grammar in terms of which signs are licensed by the grammar and what linguistic properties are associated with them. This is possible since the types which we eliminated encode distinctions which are already encoded by other types in the grammar. These redundant types thus seem to be an unnecessary complication that should be eliminated to obtain a more transparent and compact grammar.

**Types and what they are used for** Typed languages make it possible to verify the well-formedness of the complex data structures used in current linguistic theories by requiring an explicit declaration of the modeled domain. In HPSG (Pollard and Sag, 1994), a signature defines which types of objects exist, and which attributes with which values are appropriate for the different types of objects. A hierarchy of types is defined to make it possible to refer to classes and subclasses of objects. To encode a particular distinction to be modeled in the grammar, one thus has a choice of introducing a new attribute as appropriate for a type, or to define new subtypes of that type. For example, one could encode a distinction between inverted and non-inverted verbs as a type distinction under *verb*, or via a boolean attribute defined for *verb*.

Once a type is introduced, the instances of that type can be constrained by specifying a type constraint associating the type with a description of the required values for its attributes. This mechanism is used to express the grammatical constraints in HPSG linguistics, i.e., such type constraints are used to define which linguistic objects are grammatical and which are not.

In HPSG-based grammar implementation, the option of introducing new types and associating them with descriptions through type constraints has also led to a second use of types, which is not motivated by empirical distinctions that need to be made to correctly model the domain. Instead, types are introduced as mere names for collections of descriptions, i.e., abbreviations (Flickinger, 2000). While such a use of types is a possibility, based on an example from the English Resource Grammar, we will illustrate below that it introduces a significant level of unmotivated complexity into the modeled domain.

**Types in the English Resource Grammar** In the version of the English Resource Grammar (Flickinger et al., 2000, ERG) we worked with, there are 3559 distinct types, 1294 of which are glb types which are introduced by the LKB compiler to obtain unique greatest lower bounds for each pair of types. For our illustration we focus on the 225 subtypes of *word* (ignoring glb types). These types are arranged in the relatively complex type hierarchy under *word* shown in figure 1.

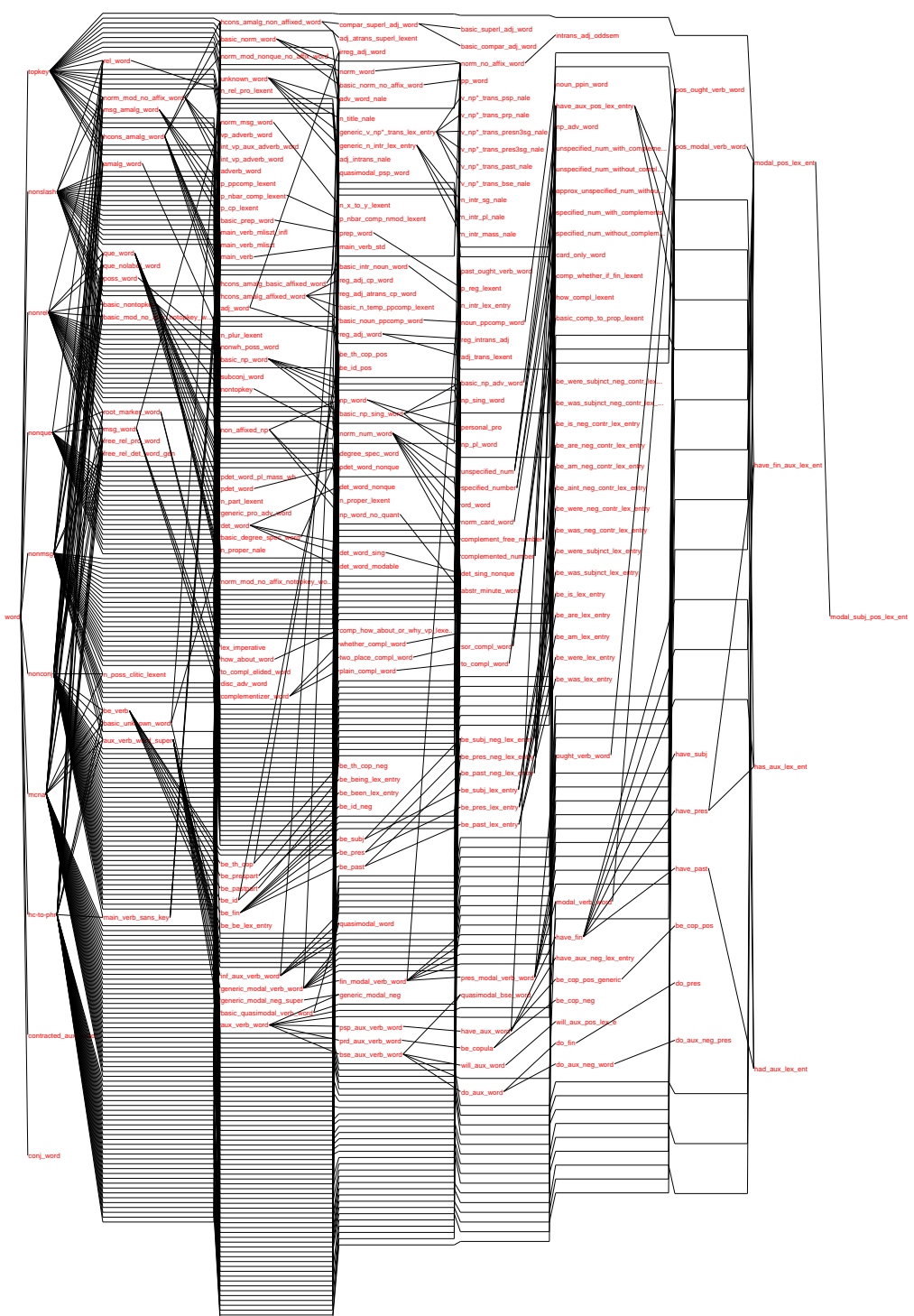


Figure 1: The hierarchy below *word* in the ERG

**An experiment reducing the number of types to what is empirically required** For the experiment, we went through the subtypes of *word* and considered for each of the types whether it introduces a linguistic distinction that is necessary to formulate the grammatical constraints encoded in the ERG. Each type which was found not to introduce such a necessary distinction was eliminated from the hierarchy and instead defined as an abbreviation to keep the grammar specified compactly.<sup>1</sup> This process resulted in the significantly reduced type hierarchy below *word* displayed in figure 2.

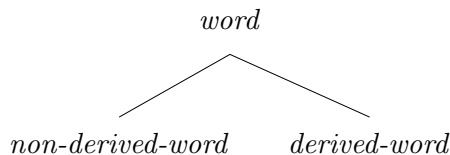


Figure 2: The hierarchy below *word* in MERGE

Since only types duplicating information encoded elsewhere were eliminated, the simplification of the type hierarchy below *word* did not affect the coverage or the analyses provided by the grammar, which was verified using the 1348 item test suite provided with ERG. The complex type hierarchy of figure 1 thus appears to be irrelevant in terms of the structures licensed by the grammar. Furthermore, the simplification did not result in any negative side-effects with respect to computational issues, such as introducing disjunctions into the grammar or requiring complex antecedents for implicational constraints.

**Summary and Outlook** The purpose of this abstract was to contribute to a discussion of the criteria guiding the development of a well-motivated standard practice for HPSG-based grammar development. Focusing on the motivation for type distinctions introduced in a grammar, we applied the principle of Occam’s razor to a part of the ERG type hierarchy and eliminated all type distinctions which are not motivated in terms of the sentences licensed by the grammar and the analyses assigned to them. This eliminated almost the entire hierarchy under *word* and thus resulted in a significant simplification of the grammar. In terms of outlook, we are continuing this experiment with the intention of eliminating all type distinctions which are not motivated in terms of the sentences licensed and the analyses assigned to them. Based on the experience with the *word* subtypes described in this abstract, we are confident that the resulting grammar will be significantly more transparent and compact than the original. We therefore propose such a conservative use of types as a model which HPSG-based grammar writers can follow to achieve a compact and transparent encoding.

---

<sup>1</sup>The experiment was carried out with a recoding of the ERG in the Trale system (Meurers et al., 2002), which includes the possibility of defining macros to abbreviate descriptions.

## References

- Flickinger, Dan (2000). On building a more efficient grammar by exploiting types. *Natural Language Engineering* 6(1), 15–28.
- Flickinger, Dan, Ann Copestake and Ivan A. Sag (2000). HPSG Analysis of English. In Wolfgang Wahlster (ed.), *Verbmobil: Foundations of Speech-to-Speech Translation*, Berlin: Springer, Artificial Intelligence, pp. 254–263.
- Meurers, Detmar, Gerald Penn and Frank Richter (2002). A Web-based Instructional Platform for Constraint-Based Grammar Formalisms and Parsing. In *Proceedings of the ACL-01 workshop on Effective Tools and Methodologies for Teaching NLP and CL*. Philadelphia, PA.
- Pollard, Carl and Ivan A. Sag (1994). *Head-Driven Phrase Structure Grammar*. Chicago, IL: University of Chicago Press.