Upper Modeling: A general organization of knowledge for natural language processing

John A. Bateman[1]
USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695
e-mail: bateman@isi.edu

Paper prepared for the Workshop on Standards for Knowledge Representation Systems, Santa Barbara, March, 1990. A version of this paper also appears in the Proceedings of the International Language Generation Workshop (Pittsburgh, June 1990).

Abstract

A general, reusable computational resource has been developed within the Penman text generation project for organizing domain knowledge appropriately for linguistic realization. This resource, called the **upper model**, provides a domain- and task-independent classification system that supports sophisticated natural language processing while significantly simplifying the interface between domain-specific knowledge and general linguistic resources. Although levels of abstract semantic organization similar to the upper model are now being sought in many natural language systems, often with a view to exploring appropriate formalization techniques, these mostly suffer from either a lack of theoretical constraint concerning their internal contents and organization and the necessary mappings between them and surface realization, or a lack of abstraction which binds them too closely with linguistic form. This paper presents the results of our experiences in designing and using the upper model in a variety of applications over the past 5 years. In particular, we present our conclusions concerning the appropriate organization of an upper model, its domain-independence, and the types of interrelationships that need to be supported between upper model and grammar and semantics. We consider the organization and interrelationships that we have found necessary both useful for organizing domain knowledge consistently and vital for sophisticated generation capabilities.

[1] Financial support was provided by AFOSR contract F49620-87-C-0005, and in part by DARPA contract MDA903-87-C-641. The opinions in this report are solely those of the author.

1 Introduction: interfacing with a text generation system

Consider the task of interfacing a domain-independent, reusable, general text generation system with a particular application domain, in order to allow that application to express system-internal information in one or more natural languages. Internal information needs to be related to strategies for expressing it. This could be done in a domain-specific way by coding how the application domain requires its information to appear. This is clearly problematic, however: it requires detailed knowledge on the part of the system builder both of how the generator controls its output forms and the kinds of information that the application domain contains. A more general solution to the interfacing problem is thus desirable.

We have found that the definition of a mapping between knowledge and its linguistic expression is facilitated if it is possible to classify any particular instances of facts, states of affairs, situations, etc. that occur in terms of a set of general objects and relations of specified types that behave systematically with respect to their possible linguistic realizations. This approach has been followed within the PENMAN text generation system [26, 38] where, over the past 5 years, we have been developing and using an extensive, *domain- and task-independent* organization of knowledge that supports natural language generation: this level of organization is called the **upper model** [5, 25, 36].

The majority of natural language processing systems currently planned or under development are now recognizing the necessity of some level of abstract 'semantic' organization similar to the upper model that classifies knowledge so that it may be more readily expressed linguistically.[2] However, they mostly suffer from either a lack of theoretical constraint concerning their internal contents and organization and the necessary mappings between them and surface realization, or a lack of abstraction which binds them too closely with linguistic form. It is important both that the contents of such a level of abstraction be motivated on good theoretical grounds and that the mapping between that level and linguistic form is specifiable.

Our extensive experiences with the implementation and use of a level of semantic organization of this kind within the PENMAN system now permit us to state some clear design criteria and a well-developed set of necessary functionalities. In this paper, I sketch the practical application and general linguistic significance for the upper model (Section 2), and then go on to show the sources of constraints we have found for its contents and organization (Section 3 onwards).

[2] Including, for example: the Functional Sentence Structure of XTRA: Allgayer, Harbusch, Kobsa, Reddig, Reithinger and Schmauks (1989); Moens *et al* (1989); Chen and Cha (1988); Dahlgren, McDowell, and Stabler (1989); Emele (1989); POLYGLOSS: Emele *et al.*, (1990); certain of the Domain and Text Structure Objects of SPOKESMAN: Meteer (1989); TRANSLATOR: Nirenburg, Raskin and Tucker (1986); the Semantic Relations of EUROTRA-D: Steiner *et al.* (1987); JANUS: Weischedel (1989); Zajac (1989).

2 The Upper Model's Contribution to the Solution to the Interface Problem: Domain independence and reusability

The upper model decomposes the mapping problem by establishing a level of *linguistically motivated* knowledge organization specifically constructed as a reponse to the task of constraining linguistic realizations.[3] While it may not be reasonable to insist that *application domains* organize their knowledge in terms that respect linguistic realizations — as this may not provide suitable organizations for, e.g., domain-internal reasoning — we have found that it *is* reasonable, indeed essential, that domain knowledge be so organized if it is also to support expression in natural language relying on general natural language processing capabilities.

The general types constructed within the upper model necessarily respect generalizations concerning how distinct semantic types can be realized. We then achieve the necessary link between particular domain knowledge and the upper model by having an application *classify* its knowledge organization in terms of the general semantic categories that the upper model provides. This does not require any expertise in grammar or in the mapping between upper model and grammar. An application needs only to concern itself with the 'meaning' of its own knowledge, and not with fine details of linguistic form. This classification functions solely as an interface between domain knowledge and upper model; it does not interfere with domain-internal organization. The text generation system is then responsible for realizing the semantic types of the level of meaning with appropriate grammatical forms.[4] Further, when this classification has been established for a given application, application concepts can be used freely in input specifications since their possiblities for linguistic realization are then known. This supports two significant functionalities:

- interfacing with a natural language system is radically simplified since much of the information specific to language processing is factored out of the input specifications required and into the relationship between upper model and linguistic resources;
- the need for domain-specific linguistic processing rules is greatly reduced since the upper model provides a domain-independent, general and reusable conceptual organization that may be used to classify all domain-specific knowledge when linguistic processing is to be performed.

[3]Although my discussion here is oriented towards text generation, our current research aims at fully bi-directional linguistic resources [19, 20]; the mapping is therefore to be understood as a *bi-directional* mapping throughout.

[4]This is handled in the PENMAN system by the grammar's *inquiry semantics*, which has been described and illustrated extensively elsewhere (e.g., [2, 24, 27]).

An example of the simplification that use of the upper model offers for a text generation system interface language can be seen by contrasting the input specification required for a generator such as MUMBLE-86 [33] — which employs realization classes considerably less abstract than those provided by the upper model — with the input required for Penman.[5] Figure 1 shows corresponding inputs for the generation of the simple clause: *Fluffy is chasing little mice.* The appropriate classification of domain knowledge concepts such as *chase, cat, mouse*, and *little* in terms of the general semantic types of the upper model (in this case, *directed-action, object, object,* and *size* respectively — cf. [5] automatically provides information about syntactic realization that needs to be explicitly stated in the MUMBLE-86 input (e.g., `S-V-O_two-explicit-args`, `np-common-noun`, `restrictive-modifier`, `adjective`). Thus, for example, the classification of a concept *mouse* as an *object* in the upper model is sufficient for the grammar to consider a realization such as, in MUMBLE-86 terms, a `general-np` with a particular `np-common-noun` and `accessories` of `gender neuter`. Similarly, the classification of *chase* as a *directed-action* opens up linguistic realization possibilities including clauses with a certain class of transitive verbs and characteristic possibilities for participants, corresponding nominalizations, etc. Such low-level syntactic information is redundent for the PENMAN input.[6]

The further domain-independence of the upper model is shown in the following example of text generation control. Consider two rather different domains: a navy database of ships and an expert system for digital circuit diagnosis.[7] The navy data base contains information concerning ships, submarines, ports, geographical regions, etc. and the kinds of activities that ships, submarines, etc. can take part in. The digital circuit diagnosis expert system contains information about subcomponents of digital circuits, the kinds of connections between those subcomponents, their possible functions, etc. A typical sentence from each domain might be:

> *circuit domain:* The faulty system is connected to the input
> *navy domain:* The ship which was inoperative is sailing to Sasebo

The input specifications for both of these sentences are shown in Figure 2. These specifications freely intermix upper model roles and concepts (e.g., *domain, range,*

[5]Note that this is not intended to single out MUMBLE-86: the problem is quite general; cf. unification-based frameworks such as [29], or the Lexical Functional Grammar (LFG)-based approach of [35]. As mentioned above, the current developments within most such approaches are now considering extensions similar to that covered by the upper model.

[6]Moreover, when additional information is required, that information is supplied in *semantic* terms rather than in terms of morphosyntactic labeling such as `:number plural` — in this case this is represented in inquiry semantics by the inquiry response pairs {:multiplicity-q multiple} and {:singularity-q nonsingular}. This is also the case for 'tense': the *:tense present-progressive* specification in the PENMAN input is only a standard abbreviation that PENMAN supplies for applications that do not want to undertake time reasoning. Further semantic information, such as speech act type and thematic (textual) organization, has been defaulted in the example shown, but can also be made explicit when required. For descriptions of all these distinctions in detail, see the PENMAN documentation [38].

[7]These are, in fact, two domains with which we have had experience generating texts using the upper model.

```
(general-clause
  :head (CHASES/S-V-O_two-explicit-args
          (general-np
            :head (np-proper-name "Fluffy")
            :accessories (:number singular
                          :gender masculine
                          :person third
                          :determiner-policy no-determiner))
          (general-np
            :head (np-common-noun "mouse")
            :accessories (:number plural
                          :gender neuter
                          :person third
                          :determiner-policy initially-indefinite)
            :further-specifications
              ((:attachment-function restrictive-modifier
                :specification (predication-to-be *self*
                                  (adjective "little"))) )) )
  :accessories (:tense-modal present :progressive
                :unmarked) )
```

Input to MUMBLE-86 for the clause: *Fluffy is chasing little mice*
from: Meteer, McDonald, Anderson, Forster, Gay, Huettner, and Sibun (1987)

```
(e / chase
   :actor (e / cat :name Fluffy)
   :actee (m / mouse
             :size-ascription (s / little)
             :multiplicity-q multiple :singularity-q nonsingular)
   :tense present-progressive)
```

Corresponding input to PENMAN

Figure 1: Comparison of input requirements for MUMBLE-86 and PENMAN

property-ascription) and the respective domain roles and concepts (e.g., *system, faulty, input, destination, sail, ship, inoperative*). Both forms are rendered interpretable by the subordination of the domain concepts to the single generalized hierarchy of the upper model. This is illustrated graphically in Figure 3. Here we see the single hierarchy of the upper model being used to subordinate concepts from the two domains. The domain concept `system`, for example, is subordinated to the upper model concept *object*, domain concept `inoperative` to upper model concept *quality*, etc. By virtue of these subordinations, the grammar and semantics of the generator can interpret the input specifications in order to produce appropriate linguistic realizations: the upper model concept *object* licenses a particular set of realizations, as do the concepts *quality*, *material-process*, etc.

These realizations are not in a one-to-one correspondence with upper model concepts, however. The relationship needs to be rather more complex and so the question of justification of upper model concepts and organization becomes significant.

```
(v1 / connects
    :domain (v2 / system
                :relations (v3 / property-ascription
                               :domain v2
                               :range (v4 / faulty)))
    :range (v5 / input)
    :tense present)
```

Input for digital circuit example sentence:
The faulty system is connected to the input

```
(v1 / sail
    :actor (v2 / ship
               :relations (v3 / property-ascription
                              :domain v2
                              :range (v4 / inoperative)
                              :tense past)
    :destination (sasebo / port)
    :tense present-progressive)
```

Input for navy example sentence:
The ship which was inoperative is sailing to Sasebo

Figure 2: Input specifications from navy and digital circuit domains

3 Degree of Abstraction vs. Linguistic Responsibility

The general semantic types defined by a level of meaning such as the upper model need to be 'linguistically responsible', in that mappings between them and linguistic form may be constructed. In addition, to be usable by an application, they must also be sufficiently operationalizable so as to support *consistent* coding of application knowledge. Both of these requirements have tended to push the level of organization defined closer towards linguistic form. However, it is also crucial for this organization to be *sufficiently abstract*, i.e., removed from linguistic form, so that it is possible for an application to achieve its classification purely on grounds of meaning. It is thus inadequate to rely on form-oriented criteria for upper model construction because grammatical classifications are often non-isomorphic to semantic classifications: they therefore need to deviate from semantic organization in order to respect the syntactic criteria that define them. This has been noted, for example, in Heid, Rösner and Roth's [14] attempt to interface the explicitly form-oriented process type classification system developed in the context of the EUROTRA-D work on machine translation [41] with the SEMSYN text generator [39], and in our own use of the upper model within PENMAN [3]. Reliance on details of linguistic realization also compromises the design aim that the applications should not be burdened with *grammatical* knowledge.[8]

[8]This is also resonant with the design aim in text generation that higher level processes — e.g., text planners — should not need direct access to low level information such as the grammar [16].

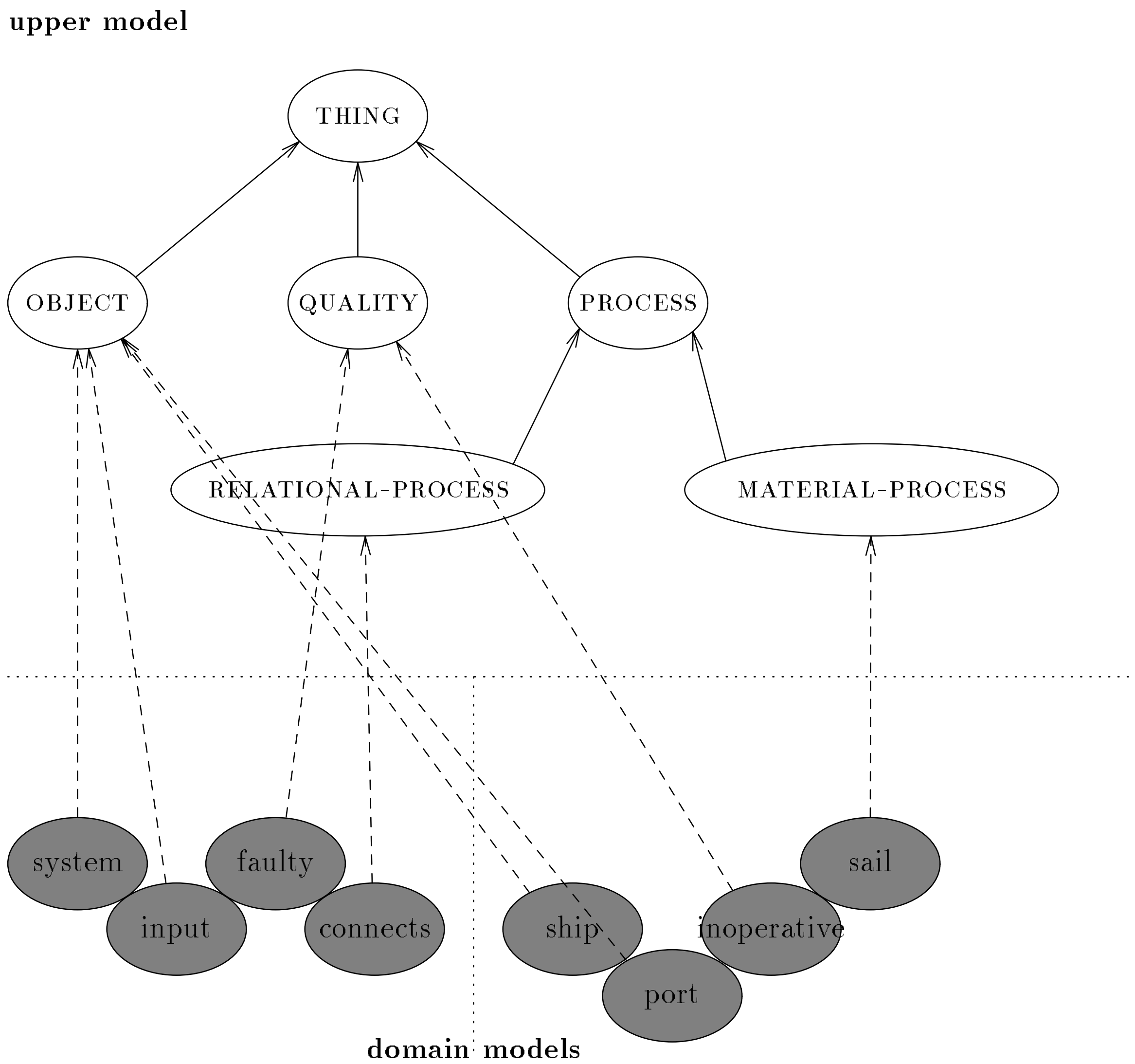

Figure 3: Upper model organization reuse with differing domains

Thus, the level of abstraction of an upper model needs to be sufficiently high that it generalizes across syntactic alternations, without being so high that the mapping between it and surface form is impossible to state. This tension between the requirements of abstractness and linguistic responsibility presents perhaps the major point of general theoretical difficulty and interest for future developments of upper model-like levels of meaning. Without a resolution, substantive progress that goes beyond revisions of what the PENMAN upper model already contains is unlikely to be achieved. It is essential for constraints to be found for what an upper model should contain and how it should be organized so that an appropriate level of abstraction may be constructed.

4 Constraining the Organization of an Upper Model

Several methodologies have been pursued for uncovering the organization and contents of a level of meaning such as an upper model; Figure 4 sets out most of them, with examples of approaches that have adopted them, along the continuum of abstraction introduced in the previous section. While the problem of being too bound to linguistic form has been mentioned, there are also severe problems with attempts to construct an upper model independent of form and motivated by other criteria, e.g., a logical theory of the organization of knowledge *per se*. Without a strong theoretical connection to the linguistic system the criteria for organizing an abstraction hierarchy remain ill-specified; there is very little guarantee that such systems will organize themselves in a way appropriate for interfacing well with the linguistic system.[9]

An alternative route is offered by the approaches in the middle of the continuum, i.e., those which abstract beyond linguistic form but which still maintain a commitment to language as a motivating force. This is further strengthened by the notion, now resurgent within current linguistics, that the organization of language informs us about the organization of 'knowledge' (e.g., [12, 18, 22, 28, 42]): that is, *the relation between grammar and semantics/meaning is not arbitrary*. Detailed theories of grammar can then be expected to provide us with insights concerning the organization that is required for the level of meaning.

We have found that the range of meanings required to support one particular generalized functional region of the grammar developed within the PENMAN system provides a powerful set of organizing constraints concerning what an upper model should contain. It provides for the representation of 'conceptual' meanings at a high level of abstraction while still maintaining a mapping to linguistic form. This functional region corresponds with the Systemic Functional Linguistic notion of the *experiential*

[9] Furthermore, the experience of the JANUS project (e.g., [43]) has been that the cost of using a sufficiently rich logic to permit axiomatization of the complex phenomenon required is very high, motivating augmentation by an abstraction hierarchy very similar to that of the upper model and facing the same problem of definitional criteria.

nonlinguistic	reality	ontological — ‘logical’	Weischedel (1989)
	knowledge	cognitive — ‘psychological’	Langacker (1987)
		situational — ‘socio/psycho-logical’	Steiner (fc)
linguistic	meaning	grammatical semantics inquiry semantics clause-based	Halliday & Matthiessen (fc) PENMAN UPPER MODEL
		lexical semantics word senses word-based	Jackendoff (1983), LFG Mel’čuk & Žholkovskij (1970)
	form	syntactic realization classes syntax	Steiner *et al.* (1987) LFG

Figure 4: Sources of motivations for upper model development

metafunction [28], one of four generalized meaning types which are simultanously and necessarily made whenever language is used. Any sentence must contain contributions to its function from all four ‘metafunctions’ — each metafunction providing a distinct type of constraint. The value of this factorization of distinct meaning types as far as the design of an upper model is concerned can best be seen by examining briefly what it *excludes* from consideration for inclusion within an upper model: i.e., all information that is controlled by the remaining three metafunctions should not be represented.

- The *logical metafunction* is responsible for the construction of composite semantic entities using the resources of interdependency; it is manifested in grammar by dependency relationships such as those that hold between the head of a phrase and its dependents and the association of concepts to be expressed with particular heads in the sentence structure. The removal of this kind of information permits upper model specifications to be independent of grammatical constituents and grammatical dominance relations.

  This relaxes, for example, the mapping between objects and processes at the upper model level and nominals and verbals at the grammatical level, enabling generalizations to be captured concerning the existence of verbal participants in nominalizations, and permits the largely textual variations shown in (1) and (2)[10] to be removed from the upper model coding.

  (1) It will probably rain tomorrow
  It is likely that it will rain tomorrow
  There is a high probability that it will rain tomorrow

  (2) independently
  in a way that is independent

  No change in upper model representation or classification is required to represent these variations.

[10]Example taken from [31].

This can be seen more specifically by considering the following PENMAN input specification that uses only upper model terms:

```
((c0 / cause-effect
     :domain discharge
     :range breakdown)
 (discharge / directed-action
     :actee (electricity / substance))
 (breakdown / nondirected-action
     :actor (system / object)))
```

This states that there are two configurations of processes and participants — one classified as an upper model *directed-action*, the other as a *nondirected-action* — which are related by the upper model relationship *cause-effect*. Now, the assignment of concepts to differently 'ranked' heads in the grammar governs realization variants including the following:

> *Electricity being discharged resulted in the system breaking down.*
> *Because electricity was discharged, the system broke down.*
> *Because of electricity being discharged the system broke down.*
> *...the breakdown of the system due to an electrical discharge...*
> *Electricity was discharged causing the system to break down.*
> *...an electrical discharge causing the breakdown of the system...*
> etc.

Many such 'paraphrase' issues are currently of concern within the text generation community (e.g., [31, 17, 7, 4]).

- The *textual metafunction* is responsible for the creation and presentation of text in context, i.e., for establishing textual cohesion, thematic development, rhetorical organization, information salience, etc. The removal of this kind of information allows upper model specifications to be invariant with respect to their particular occasions of use in texts and the adoption of textually motivated perspectives, such as, e.g., theme/rheme selections, definiteness, anaphora, etc. Thus, with the same input specification as above, the following variations are supported by varying the textual constraints:

> *It was the electricity being discharged that resulted in the system breaking down.*
> *The discharge of electricity resulted in the system breaking down.*
> *The system breaking down — the electricity being discharged did it!*
> etc.

And similarly, the following variation is supported within nominal phrases:

> *...the discharge of electricity...*
> *...a discharge of electricity...*
> *...some particular electrical discharge...*

...it...
etc.

These textual variations are controlled during the construction of text (cf. [28, 10, 15, 32, 6]) and, again, are factored out of the upper model.

- The *interpersonal metafunction* is responsible for the speaker's interaction with the listener, for the speech act type of an utterance, the force with which it is expressed, etc. Thus, again with the same input specification, the following variants are possible:

  Did electricity being discharged result in the system breaking down?
  Electricity being discharged resulted surprisingly in the whole damn thing breaking down.
  I rather suspect that electricity being discharged may have resulted in the system breaking down.
  etc.

This leaves the upper model with the task of representing the speaker's experience in terms of generalized linguistically-motivated 'ontological' categories. More specifically, the following information is required (with example categories drawn from the current PENMAN upper model):

- abstract specifications of process-type/relations and configurations of participants and circumstances (e.g., NONDIRECTED-ACTION, ADDRESSEE-ORIENTED-VERBAL-PROCESS, ACTOR, SENSER, RECIPIENT, SPATIO-TEMPORAL, CAUSAL-RELATION, GENERALIZED-MEANS),
- abstract specifications of object types, for, e.g., *semantic* selection restrictions (e.g., DECOMPOSABLE-OBJECT, ABSTRACTION, PERSON, SPATIAL-TEMPORAL),
- abstract specifications of quality types, and the types of entities which they may relate (e.g., BEHAVIORAL-QUALITY, SENSE-AND-MEASURE-QUALITY, STATUS-QUALITY),
- abstract specifications of combinations of events (e.g., DISJUNCTION, EXEMPLIFICATION, RESTATEMENT).

Our present upper model contains approximately 200 such categories, as motivated by the requirements of the grammar, and is organized as a structured inheritance lattice represented in the LOOM knowledge representation language [23]. It is described in full in [5, 3].

The metafunctional factorization thus permits the upper model to specify experiential meanings that are invariant with respect to the linguistic alternations driven by the other metafunctions. That is, a specification in upper model terms is consistent with a set of linguistic realizations that may be regarded as 'experiential paraphrases': the specification expresses the 'semantic' content that is shared across those paraphrases and often provides just the level of linguistically decommitted representation required for nonlinguistically oriented applications. Generation of any unique surface realization is achieved by additionally respecting the functional constraints that the other metafunctions bring to bear; particular surface forms are only specifiable when a complete set of constraints from each of the four metafunctions are combined. The application of these constraints is directly represented in the PENMAN grammar, which provides for the perspicuous and modular integration of many disparate sources of information. The interdependencies between these constraints and their conditions of applicability are also directly represented in the grammar. This organization of the grammar allows us to construct a rather abstract upper model while still preserving the necessary mapping to linguistic form. The value of achieving the abstract specification of meaning supported by the upper model is then that it permits a genuinely form-independent, but nevertheless *form-constraining*, 'conceptual' representation that can be used both as a statement of the semantic contents of an utterance and as an abstract specification of content for application domains that require linguistic output.

5 Summary and Conclusions

A computational resource has been developed within the PENMAN text generation project that significantly simplifies control of a text generator. This resource, called the *upper model*, is a hierarchy of concepts that captures semantic distinctions necessary for generating natural language. Although similar levels of abstract semantic organization are now being sought in many natural language systems, they are often built anew for each project, are to an unnecessary extent domain or theory specific, are required to fulfill an ill-determined set of functionalities, and lack criteria for their design. This paper has presented the results of our experiences in designing and using the upper model in a variety of applications; in particular, it presented our conclusions concerning the appropriate source of constraints concerning the organization of an upper model. We have found that restricting the information contained in an upper model to *experiential* meaning has significantly improved our understanding of how a semantic hierarchy should be organized and how it needs to relate to the rest of the linguistic system. We strongly feel, therefore, that subsequently constructed semantic organizations should follow the guidelines set out by the metafunctional hypothesis; the factorization that it provides concerning what should, and should not, be represented in an 'abstract semantic knowledge' hierarchy supports functionalities well beyond those envisioned in current text generation/understanding systems.

Furthermore, work on many more recently proposed semantic hierarchies is presently concentrating on the exploration of appropriate formalization techniques [11, 34, 21]. This is clearly crucial, but it is important to know what it is that must be represented. We consider the organization and interrelationships that we have developed for the upper model vital for sophisticated generation capabilities. Properties such as those we describe are likely to become increasingly important for natural language processing systems. Thus, we would suggest that work on formalization should take into consideration the set of functionalities and properties presently constructed and under construction within the PENMAN upper model. The development of more formal representations of the capabilities of the upper model will contribute significantly to our understanding of the nature of language and its computational modeling.

Acknowledgments

The upper model has been under development for several years, and many have and continue to contribute to it. The ideas I have reported on here would not have been possible without that development. Those responsible for the present form of the upper model include: William Mann, Christian Matthiessen, Robert Kasper, Richard Whitney, Johanna Moore, Eduard Hovy, Yigal Arens, and myself. Thanks also to Cécile Paris and Eduard Hovy for improving the paper's organization.

References

[1] Allgayer, Jürgen, Harbusch, Karin, Kobsa, Alfred, Reddig, Carola, Reithinger, Norbert and Schmauks, Dagmar (1989) 'XTRA: a Natural-Language Access System to Expert Systems'. In *International Journal of Man-Machine Communication.*

[2] Bateman, John A. (1988) 'Aspects of clause politeness in Japanese: an inquiry semantics treatment'. *Proceedings of the 26th. Annual Meeting of the Association for Computational Linguistics.*

[3] Bateman, John A. (1989a) 'Upper Modelling: current states of theory and practise'. USC/Information Sciences Institute: Penman Development Note.

[4] Bateman, John A. (1989b) 'Upper Modelling for Machine Translation: a level of abstraction for preserving meaning'. Eurotra-D Working Paper No. 12, Saarbrücken: Institut für Angewandte Informationsforschung and USC/Information Sciences Institute Technical Report.

[5] Bateman, John A., Kasper, Robert T., Moore, Johanna D., and Whitney, Richard A. (1989) 'The PENMAN Upper Model - 1989'. USC/ Information Sciences Institute: Penman Development Note.

[6] Bateman, John A. and Matthiessen, Christian M.I.M. (forthcoming) 'Uncovering the Text Base'. Proceedings of the first international conference on texts and language research, Jiao-tong University, Xi'an, P.R. China, March, 1989.

[7] Bateman, John A. and Paris, Cécile L. (1989a) 'Phrasing a text in terms a user can understand'. In *Proceedings of the International Joint Conference on Artificial Intelligence*, IJCAI-89, Detroit, Michigan.

[8] Chen, Keh-Jiann and Cha, Chuan-Shu (1988) 'The design of a conceptual structure and its relation to the parsing of chinese sentences'. In *Proceedings of the 1988 International Conference on Computer Processing of Chinese and Oriental Languages*, August 29 - September 1, 1988, Toronto, Canada.

[9] Dahlgren, Kathleen, McDowell, Joyce, Stabler, Edward P. (1989) 'Knowledge Representation for Commonsense Reasoning with Text'. *Computational Linguistics*, **15**.3, pp149-170.

[10] Dale, Robert (1989) 'Cooking up referring expressions'. ACL-89.

[11] Emele, Martin C. (1989) 'A typed-feature structure unification-based approach to generation'. *Proceedings of the WGNLC of the IECE*, Oita University, Japan.

[12] Halliday, M.A.K. (1978) *Language as Social Semiotic.* London: Edward Arnold.

[13] Halliday, M.A.K. and Matthiessen, C.M.I.M. (in preparation) *The Bloomington Lattice.* Linguistics Department, University of Sydney.

[14] Heid, Ulrich, Rösner, Dietmar, and Roth, Birgit (1988) 'Generating German from Semantic Relations: Semantic relations as an input to the SEMSYN generator'. In Steiner, Erich H., Schmidt, Paul, and Zelinksy-Wibbelt, Cornelia (eds.) (1988) *From Syntax to Semantics: experiences from Machine Translation.* London: Frances Pinter.

[15] Hovy, Eduard H. and McCoy, Kathleen (1989) 'Focusing your RST'. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 1989.

[16] Hovy, Eduard H., David McDonald and Sheryl Young (ed.)(1988) 'Workshop on text planning and Natural Language Generation'. St Paul, Minnesota: Sponsored by AAAI.

[17] Iordanskaja, Lidija, Kittredge, Richard, and Polguère, Alain (1988) 'Lexical Selection and Paraphrase in a Meaning-Text Generation Model'. Paper Presented at the Fourth International Workshop on Natural Language Generation, Santa Catalina Island, California, July. 1988. To appear in Cécile L. Paris, William R. Swartout and William C. Mann (eds.) *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, Kluwer Academic Publishers.

[18] Jackendoff, Ray (1983) *Semantics and Cognition*; The MIT Press.

[19] Kasper, Robert T. (1988) 'An Experimental Parser for Systemic Grammars'. In *Proceedings of the 12th International Conference on Computational Linguistics*, pp. 309-312, Budapest, Hungary, August 1988.

[20] Kasper, Robert T. (1989) 'Unification and Classification: an experiment in information-based parsing'. In *International Workshop on Parsing Technologies*, 28-31 August, 1989, Carnegie-Mellon University, Pittsburgh, Pennsylvania, USA, pp1-7.

[21] Emele, Martin, Heid, Ulrich, Kehl, Walter, Momma, Stefan and Zajac, Rémi (1990) 'Organizing linguistic knowledge for multilingual generation'. Paper submitted to COLING-90. Project Polygloss, University of Stuttgart, West Germany.

Kehl, Walter (in preparation) 'Die semantische Repräsentation des Projekts POLYGLOSS'. Projekt POLYGLOSS, Institut für maschinelle Sprachverarbeitung, University of Stuttgart.

[22] Langacker, Ronald W. (1987) *Foundations in Cognitive Grammar.* Stanford, California: Stanford University Press.

[23] MacGregor, R. and Bates, R. (1987) 'The LOOM Knowledge Representation Language'. In *Proceedings of the Knowledge-Based Systems Workshop*, held in St. Louis, Missouri, April 21-23, 1987. Also available as ISI reprint series report, RS-87-188, USC/Information Sciences Institute, Marina del Rey, California.

[24] Mann, William C. (1982) 'The Anatomy of a Systemic Choice'. ISI Technical Report, RR-82-104. USC/Information Sciences Institute, Marina del Rey, California.

[25] Mann, William C. (1985) 'Janus Abstraction Structure — Draft 1'. USC/Information Sciences Institute, Internal draft.

[26] Mann, W.C., Matthiessen, C.M.I.M. (1985) 'A demonstration of the Nigel text generation computer program'. In Benson, J.D. and Greaves, W.S. (eds.) *Systemic Perspectives on Discourse: Volume 1; Selected Theoretical Papers from the 9th International Systemic Workshop.* New Jersey, Ablex. pp50-83.

[27] Matthiessen, C.M.I.M. (1987a) 'Semantics for a Systemic Grammar: The chooser and inquiry framework'. ISI Technical Report, RR-87-189. USC/Information Sciences Institute, Marina del Rey, California.

[28] Matthiessen, C.M.I.M. (1987b) 'Notes on the Organization of the environment of a text generation grammar'. In Kempen, G. (ed.) *Natural Language Generation: Recent Advances in Artificial Intelligence, Psychology, and Linguistics.* Martinus Nijhoff Publishers.

[29] McKeown, Kathleen R. and Paris, Cécile L. (1987) 'Functional Unification Grammar Revisited'. *Proceedings of the 25th Annual Meeting of the Association of Computational Linguistics*, pp97-103.

[30] Mel'čuk, Igor, A. and Žholkovskij, A.K. (1970) 'Towards a Functioning "Meaning-Text" Model of Language.' *Linguistics*, **57**, pp10-47.

[31] Meteer, Marie, W. (1988) 'The Implications of Revisions for Natural Language Generation'. Presented at the Fourth International Workshop on Natural Language Generation, July, Santa Catalina Island, California. To appear in Cécile L. Paris, William R. Swartout and William C. Mann (eds.) *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, Kluwer Academic Publishers.

[32] Meteer, Marie, W. (1989) 'The SPOKESMAN Natural Language Generation System'. BBN Report No. 7090. BBN Systems and Technologies Corporation, Cambridge, MA.

[33] Meteer, Marie W., McDonald, D.D., Anderson, S.D., Forster, D., Gay, L.S., Huettner, A.K. and Sibun, P. (1987) 'Mumble-86: Design and Implementation'. COINS, University of Massachusetts.

[34] Moens, Marc, Calder, Jo, Klein, Ewan, Reape, Mark, Zeevat, Henk (1989) 'Expressing Generalizations in Unification-based Formalisms'. *Proceedings of the 4th. Conference of the European Chapter of the Association for Computational Linguistics*, Manchester, England, pp174-181.

[35] Momma, Stefan and Dörre, Jochen (1987) 'Generation from f-structures'. In Klein, Ewan and Van Bentham, J. (eds.) *Categories, Polymorphism and Unification.* Cognitive Science Centre, University of Edinburgh.

[36] Moore, Johanna D. and Arens, Yigal (1985) 'A Hierarchy for Entities'. USC/Information Sciences Institute, Internal Draft.

[37] Nirenberg, Sergei, Raskin, V., and Tucker, A. (1987) 'The structure of Interlingua in TRANSLATOR'. In Nirenberg, Sergei (ed.) *Machine Translation: Theoretical and Methodological Issues.* Cambridge: Cambridge University Press.

[38] The Penman Project (1989) 'The PENMAN documentation: Primer, User Guide, Reference Manual, and Nigel Manual'. USC/ Information Sciences Institute.

[39] Rösner, Dietmar (1988) 'The generation system of the SEMSYN project: towards a task-independent generator for German'. In Zock, M. and Sabah, G. (eds.) *Advances in Natural Language Generation: an interdisciplinary perspective; volume 2.* London: Frances Pinter.

[40] Steiner, Erich H. (forthcoming) *A model of goal-directed-action as a structuring principle for the context of situation in systemic linguistics.* Berlin, etc.: Mouton and de Gruyter.

[41] Steiner, Erich H., Eckert, Ursula, Weck, Birgit and Winter, Jutta (1987) 'The development of the Eurotra-D system of semantic relations'. EUROTRA-D Working Papers No. 2. Saarbrücken: Institut für Angewandte Informationsforschung.

[42] Talmy, Leonard (1987) 'The relation of grammar to cognition'. In B. Rudzka-Ostyn (ed.) *Topics in Cognitive Linguistics.* John Benjamins.

[43] Weischedel, Ralph M. (1989) 'A hybrid approach to representation in the Janus Natural Language Processor'. In *27th Annual Meeting of the Association for Computational Linguistics*, Vancouver, British Columbia, pp193-202.

[44] Zajac, Rémi (1989) 'A transfer model using a typed feature structure rewriting system with inheritance'. *Proceedings of the 27th. Annual Meeting of the Association for Computational Linguistics*, pp1-6; 26-29 June 1989, University of British Columbia, Vancouver, British Columbia, Canada.