DANDELION

Discourse Functions and Discourse Representation:
An Empirically and Linguistically Motivated,
Interdisciplinarily-Oriented Approach to Natural Language Texts

ESPRIT Basic Research Project 6665

Deliverable R2.1.1

Update of Baseline Linguistic Resources in Dandelion

Contributions by John Bateman, Mimo Caenepeel, Claire Grover, Wiebke Ramm, Annely Rothkegel, Claudia Villiger, Erich Steiner

Editor: Jo Calder

October 10, 1994

— CLS Tilburg/Nijmegen —
— University of Edinburgh GMD-IPSI Darmstadt Universität Saarbrücken —
— Universidad Complutense Madrid —

Document ID	:	Deliverable R2.1.1: *Update of Baseline Linguistic Resources in Dandelion*
Issue date	:	October 10, 1994
Responsible	:	University of Edinburgh
Version	:	Final
Status	:	Public
WP number	:	WP2: Grammar integration
Distribution	:	DANDELION Consortium, CEC
Keywords	:	Grammars, systems

Update of Baseline Linguistic Resources in Dandelion

Contributions by John Bateman, Mimo Caenepeel, Claire Grover, Wiebke Ramm, Annely Rothkegel, Claudia Villiger, Erich Steiner

Editor: Jo Calder

University of Edinburgh

Abstract:

In this report, we focus on necessary extensions and refinements of the linguistic resources made available to the project by month 18. In a second step, we identify for the PENMAN and KOMET systems where, relative to the projects needs, extensions and refinements seem desirable. This constitutes the main part of the current report as most of the specification and implementation work undertaken so far has been done in that system. Thereafter, we discuss the grammar developed by Bob Carpenter and Gerald Penn, of CMU, which is likely to form the basis for developments in the grammatical and discourse descriptions of English and Dutch. In an appendix, we shall comment in some detail on the grammar development shell Pleuk made available from Edinburgh—while this is not strictly speaking a *linguistic* resource, it does represent a possible basis for the implementation of the workbench to be pursued in work package 2, and this report seems an appropriate forum in which to discuss advantages and disadvantages of following this route.

Contents

1 Introduction

This report constitutes deliverable R2.1.1 of task 2.1.1. **Baseline linguistic resources: selection, description, and development reporting**. [1] The grammar fragments envisaged in the technical annex were collected and made available by month 18 of the project). Since then, work has been undertaken to extend and refine these resources to suit the specific demands of this projects, thus working towards a better understanding and implementation of a workbench. We shall now concentrate on extensions and refinements necessary for the immediate future, concentrating on the following topics for each of the tools currently available to the project:

- An assessment of the current coverage of the grammar fragment with respect to the research areas to be addressed in the empirical tasks. Where possible, we differentiate between
  - aspects of the empirical tasks already reflected in implementations,
  - what is available as theoretical specification,
  - what kind of problems do we expect to be addressable in principle,
  - for which aspects do we not expect a computational linguistic account to be possible.
- How strongly is each framework tied to a specific directionality (analysis vs. generation)?
- How does the framework relate to the multilingual character of the project, i.e., is the framework specifically oriented at the language it models or do we think cross-linguistic/multilingual generalizations or adaptations to other languages involved in the project are possible?

As will be seen, input here has come from the sites in Edinburgh, Darmstadt, Saarbrücken, and, to some extent, Tilburg, but as foreseen in the programme of work, sites have concentrated to different degrees on specification and implementation so far. This is an intended distribution of labour in the project, so that the different degrees of depth of treatment in this reports, shown for the different sites, are exactly what we expected, given that there are various areas of specialization represented at the different sites.

Our report here is structured in terms of the sites having a share in this workpackage. It should be apparent, though, that there is a particularly strong coherence of work going on in Darmstadt and in Saarbrücken, in the sense that they are both using instantiations of the PENMAN system architecture and implementations of it, made available by ISI/USC and GMD/IPSI respectively. To a lesser extent, a similar convergence can be seen between work at CLS Tilburg/Nijmegen and University of Edinburgh, to the extent that both envisage taking the Ale grammar of Carpenter and Penn as a point of departure. Further, the results reported in R2.2.1 should also be taken into account here: translation of the different linguistic resources exploited by project partners into a common format manipulable by the DANDELION workbench seems at present an achievable goal, in which case the resources described below will ultimately contribute to a homogenous and detailed account of discourse phenomena in various languages.

2 Darmstadt

The resources of principal concern to DANDELION that are being used/considered at IPSI all relate to the development of systemic-functional descriptions of language and the evaluation of these descriptions in text generation. Two base systems are used, viz Penman (USC/ISI) and KOMET (GMD/IPSI).

[1] This deliverable is an update of deliverables R2.1.1a and R2.1.1b, which were project-internal. The editors of previous versions of this deliverable were Wiebke Ramm, Erich Steiner and Mimo Caenepeel.

The first of these has been used as the basis for the systemic-functional account of Dutch reported on in Deliverable R2.1.2; the second forms the starting point of the further resources for German reported on in Deliverable R2.1.3. The central focus of both systems is derived directly from the supporting theory. The aim is to provide accounts of the *functions* of language — both what they are and how they are expressed. Grammatical, semantic, and textual/discourse considerations are all taken into account, but the assignment of phenomena to strata/modules differs slightly from that found in other systems. In particular, much more information is placed at the grammatical stratum — including information that commonly does not appear until pragmatics is dealt with (e.g. textual choices for theme).

In terms of coverage, the grammatical resources tend to be very broad, while sometimes lacking in depth. Figure 1 contains a break down of the grammatical resources according to 'functional regions', i.e., areas of the grammar concerned with a particular kind of meaning. The leftmost columns describe grammatical classes (clauses, phrases, etc.); those to the right describe the areas of meaning which use those classes as domains of realization. Each such area, or functional region, is explicitly linked via the 'inquiry' semantics to particular areas of semantic description. Large, significant subsets of these semantic mappings between semantics and grammar are already defined and documented. For these regions, all the grammatical alternations covered can be directly controlled from the semantic level. For the others, control is possible via the semantic inquiries directly. Virtually no areas of the grammar, therefore, still require *grammatical* control. The regions of particular centrality for DANDELION are those of the *textual* metafunction shown in the rightmost columns.

The figure gives an idea of the range of phenomena covered and their organization. A further indication of the size of is resources is given in the table below, where the resources for German and Dutch are shown relative to the resources for English.

Component	*Languages*		
	English	German	Dutch
grammar	703 systems	560 systems	400 systems
semantics mappings	540 inquiries	350 inquiries	300 inquiries
ontology	250 concepts	250 concepts	250 concepts

Extensive descriptions of the grammatical phenomena concerned are available (cf documentation described below). Also included is a development environment for grammatical resources (the latest version of which is written in CLIM), as well as various software support tools for managing the resources. The general software is written in Common Lisp and has been used on Symbolics and Texas Instruments Lisp Machines, Sparcs, and Macintoshes (for which a separate user interface has been written). It is important to emphasize that these resources and components are all *running* and actively support generation in a number of contexts; installations of the systems have been made in a variety of institutions and a regular process of feedback and improvement exists.

Documentation for each of the languages dealt with is growing and has been made available to the project; the primary references are: for German [18], for Dutch [7] (and DANDELION deliverable R2.1.2), and for English [12]. In addition, there is general information about the organization of the systems and their theoretical motivations. The first lead into the organization and functionalities of the systems and resources available have been placed in the following directory.[2]

[2] The shorter of these has been stored as a straight .tex file (plus a compressed Postscript version), and the longer one as a compressed tar file. Uncompressing the file, and untaring it leaves the paper in a directory containing the source LaTeXfile (with .bbl files already physically included) and, where necessary, a further subdirectory containing EPS figures required for the document. Note that this requires a style file additional to the `dand.sty`, viz. `psfig.sty`. This file is therefore also included in the tared directory. Both papers also use `named.sty` (cf. dandelion Tex-styles) for

rank: primary class:	ideational: logical		ideational: experiential	interpersonal	textual	(cohesion:)
clause	COMPLEXITY		TRANSITIVITY CIRCUMSTANCE	DEPENDENCE MOOD, TAG POLARITY ATTITUDE	THEME CULMINATION VOICE CONJUNCTION	refer-ence, ellipsis and substi-tution
gr-phr / phr / prep.			MINOR TRANSITIVITY			
gr-phr / group / nom-inal like / nom		CLASSIFIC.	NOMINAL TYPE EPITHET QUALIFICATION SELECTION	PERSON ATTITUDE	DETERMINATION	
gr-phr / group / nom-inal like / adj		MODIFIC.	QUALITY TYPE			
gr-phr / group / nom-inal like / quant		MODIFIC.	QUANTITY TYPE			
gr-phr / group / adverb		MODIFIC.	CIRCUMSTANTIAL CIRC. TYPE	COMMENT COMMENT TYPE	CONJUNCTIVE CONJ. TYPE	
gr-phr / group / verbal		TENSE		MODALITY PERSON		
	complexes	*simplexes*				
			information unit:	KEY	INFORMATION	

Figure 1: Functional regions of systemic-functional lexicogrammar

`ipsi.darmstadt.gmd.de:/home/dandi/Systems/SFL/Docs`

The first document, *Computational Systemic Functional Linguistic Systems: generic documentation* introduces the overall organization of the linguistic system and the computational instantiation of that organization. In particular, it distinguishes the following four 'meta-strata' of description: theory, linguistic representation, computational representation, and implementation. These are described briefly, and further pointers to particular papers describing each area are given. The second document *Possible Formalisms for Computational Representation of Generic Systemic-Functional Resources* describes the list of formalisms currently under consideration as candidates for the role of the computational representation and implementation meta-strata. The list is not closed, and further suggestions and comments are welcome. This document will be updated until it is used at the beginning of task 2.2.1, the specification of the Dandelion discourse grammar development environment.

Of more general use for the baseline resources for Dandelion as a whole, Darmstadt has also begun considerations of possible general interfaces that will allow smooth interfacing between the discourse components developed empirically and the existing grammatical resources. There are a number of reasons, both practical and theoretical, for considering the issue of interfacing with lexicogrammatical components at this time.

Theoretically, the 'level' of information to be represented is the next in line in terms of abstraction from the reasonably well understood domain of grammar. It therefore makes good methodological sense to start turning our attention to this level as a target of theoretical interest in its own right. There are many research results in the area, but these often remain fragmented. Considering this level itself can help provide a framework within which these results can be expressed and build on each other. For this reason, the position adopted here is that the question of an interface is not primarily an engineering question; it is instead directly a *linguistic* question. An interface to lexicogrammatical components is to be seen as the stratum of *semantics* in the linguistic system overall.

Practically, there is a real need to determine a language for interfacing with various lexicogrammatical components. There are now a number of quite distinct lexicogrammatical components available and,

the bibliography entries. Hardcopies can be obtained by mailing `bateman@gmd.de`.

even within single projects, there is often lack of agreement over which components are to be adopted. Unfortunately, this lack of agreement can rest on rather pragmatic considerations rather than on issues of theory. For example, a common motivation for a choice is that that some system is more familiar to a developer, or, even worse, that some system is small enough to be quickly understood. Such decisions while understandable in the short term, no doubt reduce the effectiveness of research in the longer term as there is considerable duplication of efforts. As a part of the research situation facts of life, however, this probably has to be accepted. It is then necessary, therefore, that the consequences of adopting divergent components do not trickle upstream to impact negatively on other components. Components whose main concerns are not with grammatical issues should be protected from the consequences of particular choices of grammatical components. For this reason, the definition of a component-independent interface is most important.

These considerations argue for a strong modularity in the specifications and architecture to be adopted. An interface notation should be *purely* (discourse) semantic and should not contain any lexicogrammatical information. No currently available interface notation meets these requirements and so arriving at an agreed upon notation would itself be a very satisfactory result in its own right. Moreover, there is now increasing discussion in the generation community overall concerning the appropriate form that interfaces to generation components should take [9, 13] and general discussion concerning appropriate semantic representations, such as the SEMEVAL electronic discussion group (`semeval@ai.sri.com`). Our main suggestion here should also be taken as a concrete proposal for the interface notation to be used across the DANDELION project and so requires feedback and discussion within the DANDELION context.

A paper describing an initial evaluation of the three kinds of interface notation that have been proposed for Dandelion (Penman Sentence Plan Language, Quasi-Logical Form, and the input notation for the Alvey tools) has been placed in the file:

`ipsi.darmstadt.gmd.de:/home/dandi/Papers/micro.tex`

The main differences between the three notations is the following. Both QLF and the Alvey notation are in theory more closely bound to surface form than the SPL.[3] Many aspects of the surface form still find representation, whereas this is not supported in SPL. This is largely due to the differing origins of the notations. Since SPL is for generation, a user (be it human or a text planning system) does not know in advance what the surface forms will be.

A further distinction is that apart from a few selected representational constructs such as quantifiers and logical operators, both QLF and the Alvey representation rely on introducing predicates directly corresponding to lexical senses of the lexemes found in the surface string. This is not the case in the SPL representation. In SPL the main contents come from the semantic classifications defined in the linguistic resources with respect to which SPL is interpreted. Two very extensive bodies of linguistic resources are currently defined and this information should be built upon in the definition of an actual interface notation of use to the project.

Finally, it is important to recognize that the semantic features that are defined for use in SPL also include an extensive classification of the distinct *textual* variations that may occur. This, again, comes from the generation perspective where it is not sufficient to generate any sentence that happens to be truth-conditionally equivalent with the sentence that would be most textually appropriate. This is in marked contrast with the representation of the Alvey resources. The variations of the textual metafunction are precisely the means by which languages create text, rather than collections of unrelated sentences. It is vital that when text is to be created, these variations are fully under automatic control of the text planner. The specifications found in a logical form that neutralizes these

[3] Note that this criticism applies most strongly to the SRI Core Language Engine variant of QLF; the version developed in the ESPRIT Project PLUS, which has been proposed for use in Dandelion suffers less from this problem, making it a more attractive formalization.

distinctions are not adequate. Also inadequate are attempts to model the necessary differences in terms of presuppositions and the like: far more organization and detail is generally required when the complexities of actual texts are considered. The problems are also increased when we seek an interface notation that is to operate not only across distinct grammar components, but also across distinct languages. For these reasons, some extended version of SPL suggests itself as the best choice of interface notation available to us.

Where the QLF and Alvey notations show some advantages is in their adoption of a logical form to give the basic operations available in the notation. This appears to improve their representations of coordination, for example, with coordinated verbal phrases. Even here it is not clear, however, that the suggested solutions are appropriate. The rather different grammatical analyses that would be offered of such constructions in the grammars for which SPL has been used suggest different specifications. But this is definitely an area that should be investigated further for possible beneficial cross-fertilization. In particular, we suggest that in terms of the *content* of the interface notation, we move towards that specified in the SPL, while in terms of the *formalization*, we move towards the QLF found in the PLUS project as represented in feature structures. More details of this discussion are to be found in the paper referred to above. However, given the value of pursuing this area within Dandelion, it seems likely that the project will also find itself able to make a very significant contribution to the current debate concerning the definition of *standards* for generator input.

3 Edinburgh

This section first describes the Ale version of HPSG and then assesses its applicability within DANDELION. Note that this section also refers to work carried out at CLS Tilburg/Nijmegen.[4]

3.1 The Ale HPSG English grammar

Almost coincident with the publication of [16], Bob Carpenter and Gerald Penn announced the release of a new version of the Ale system [5]. The background to the computational system is described at length in [4] and in the documentation for the system itself. In this section, we will briefly discuss the grammar and its application and reuse, both within the context of Ale, and within other closely related computational systems.

3.2 General description

The HPSG provided with Ale is based very closely on material presented in the first eight chapters of [16]. Ale places some general constraints on the form of grammars it will interpret and these constraints have resulted in a reformulation of the rules and constraints as given by Pollard and Sag. Chief amongst these are:

Grammar Rules Ale deploys a notion of grammar rule much more like that found in standard phrase structure grammars, in that linear order is entirely explicit. Furthermore, as with DCGs [15], constraints associated with the categories that appear in some rule may be expressed by the occurrence of *goals* within rules.

Quantification in principles and rules The metalanguage adopted by [16] is in general a form of English, making use of various statements which are taken to have universal import. For

[4]Due to administrative difficulties, the development of Dutch resources at CLS Tilburg/Nijmegen has not proceeded as quickly as intended.

example, the *Head-Feature Principle* applies to all headed signs and the *Raising Principle* refers to the existence of some element within a list. On the other hand, Ale provides no facilities for the direct expression of quantification of these kinds.

A consequence of these differences is that, in encoding the grammar of [16], principles with universal force are stated as goals associated with all rules to which the universal is taken to imply. For statements with existential force, such as the example above, an encoding is available by means of predicates such as `member`, which allows the non-deterministic choice of an element from a list, perhaps together with further constraints. As an illustration, here is Pollard and Sag's *schema 1*, the *Head-Subject* schema, together with its Ale recoding:

(1) The SYNSEM—LOCAL—CATEGORY—SUBCAT value is <> and the DAUGHTERS value is an object of sort *head-comp-struct* whose HEAD-DAUGHTER value is a phrase and whose COMPLEMENT-DAUGHTERS value is a list of length one.

(2)
```
schema1 rule (Mother,phrase,synsem:loc:(cat):subcat:[]) ===>
cat> (SubjDtr,non_word,synsem:SubjSyn),   % n.b. only one complement permitted
cat> (HeadDtr,phrase), goal> (head_feature_principle(Mother,HeadDtr),
inv_minus_principle(Mother),
subcat_principle(Mother,HeadDtr,[SubjSyn]),
marking_principle(Mother,HeadDtr), spec_principle(SubjDtr,HeadDtr),
      semantics_principle(Mother,HeadDtr,[SubjDtr]),
%     universal_trace_principle: obviated here by parochial
      parochial_trace_principle(SubjDtr),
      nonlocal_feature_principle(Mother,HeadDtr,[SubjDtr]),
      single_rel_constraint(Mother), clausal_rel_prohibition(Mother),
      relative_uniqueness_principle(Mother,[SubjDtr,HeadDtr]),
      conx_consistency_principle(Mother,[SubjDtr,HeadDtr]),
      deictic_cindices_principle(Mother,[SubjDtr,HeadDtr])).
```

Note here the existence of two daughter categories (labelled with `cat` > above), which have been linearized into an order appropriate for English. Further, as stated above, all principles which apply to such constructions have been explicitly added in. Finally, there is a certain amount of optimization that has also gone on. Note that the *Universal Trace Principle* has been commented out. More drastically, the feature geometry used by HPSG for the representation of the daughters participating in some construction has been omitted entirely.

3.3 The Ale grammar in DANDELION

Referring back to the criteria described on p1 above, the Ale HPSG is primarily sentence-oriented, but, as with [16], certain discourse-oriented phenomena are also touched on, for example the constraints placed on the context of discourse by the occurrence of pronouns. Further work by Janet Hitzeman in the context of the Discourse LRE has extended this grammar to provide an integrated account of the use of clue words, tense, aspect and temporal centering.

On the other hand, as the grammar currently stands, it does not currently support the notion of *theme* suggested, for example in R1.2.1, whose definition is, roughly speaking, the first phrasal constituent of some sentence. That it does not do so is to a large extent a consequence of the optimization discussed above, in that information about the substructure of a phrase is not retained at higher levels of analysis. (There are also questions about whether, in an HPSG which did not encode linear order by means of phrase structure rules, the availability of such information would be consistent with standard assumptions.) In other respects, the grammar may provide too much information. For example, the treatment of quantifier scopes assumes that all possible scopings are to be produced. This is a reasonable tactic in the case of grammars designed to demonstrate comprehensiveness with respect to the possible meanings associated with an input. But, if one's main interest is unconnected with quantifier scoping, the proliferation of analyses is something of a practical problem.

Ale comes equipped with a parser, which allows the analyses of inputs to be determined. At the University of Amsterdam, work is also in progress on a generator for Ale (Henk Zeevat, p.c.). Under the Pleuk system (section A below), Ale allows the graphical verification of analyses. One may take these three modes of interaction to demonstrate that the system itself is not in principle limited to a particular directionality.

With respect to multilinguality, Pollard and Sag are careful in their development of HPSG to take account of non-English data, in the expression of their universal principles. The grammar delivered with Ale follows Pollard and Sag's appendix in flagging those constraints which are specific to English. A growing body of work (see for example [14, 17]) examines the application of HPSG within other Germanic languages, and the results of this research can be expected to influence theoretical developments within HPSG, and the content of the grammars exploited in DANDELION.

4 Saarbrücken

Work in Saarbrücken has profitted from being able to import the KOMET grammar and PENMAN system for German from Darmstadt. As shown in detail in our report on German grammar development submitted as deliverable R2.1.3, descriptions and specifications for extensions have been written, and a first pilot implementation of a fragment of these has been completed.

In our report R2.1.3., we also identify a number of areas in which refinements of the KOMET tools in its treatment of the German clause grammar would be helpful, yet we shall not repeat these statements and suggestions here, as our current focus should be more on discourse and semantics in the current report (for those interested in our specifications of extensions and refinements of the clause grammar tool, cf. the section on *A statement of some representational requirements for a German treatment of Theme* in R2.1.3).

In the following, we shall address the questions formulated in our introduction as guidelines for the current report:

- The KOMET grammar has initial treatments of all the grammatical areas necessary for grammatical encoding of types of information addressed in the empirical tasks, down to the level of implementation. During months 6-12, the treatment of Theme for German has been significantly extended and partly restructured. The treatment of Nominal Groups will be extended in the immediate future. These areas are necessary for accounts of tasks under 1.2. *Staging*, 1.1. *Global and local discourse structure*, and 1.4. *Discourse functions and anaphoric expressions.* All this is reported in detail in R.2.1.3., submitted in parallel with this report.

  We are expecting an improved treatment of the interface with the semantics for the future (non-directional implementations of the chooser-and-inquiry framework), together with specifications and implementations of further fragments of the semantic bases *upper model, interaction base, text base*. These are necessary as an instrument for control by some *text planning component* which should be liberated from any direct contact with the grammar to the maximum extent possible. This work should be possible, and is planned, for later phases of the project. We do not, at this stage, foresee a concrete problem on our workplan which should be non-addressable in principle in a computational way, other than in a few areas well inside what would traditionally be called *pragmatics* (cf. below).

- The current PENMAN system is specifically implemented for generation. Parsing components were investigated in earlier work ([11]), but are not currently available to us. Work on parsing with the English NIGEL grammar is currently being undertaken by O'Donnell (ISI, Sydney).
- The current tools are not very strongly geared towards any one specific language, and versions for different languages have, indeed, become available throughout the past few years.
- In case of pragmatic issues, as for example the question of lexical selection under text- or situation-specific criteria, we do not expect that a computational linguistic account will be realistic in any great detail in the lifetime of the current project and inside the project.

Some remarks concerning the *chaining task* in particular (R1.2.2a) are the following:

CHAINING is concerned with the connection of sentences. In this context it is of great importance that within the system, sentences can be combined and that information from one sentence is available to other sentences of a given part of the text in question. Thus, a component must be added to the system that allows an individual representation of the connection of sentences on the one hand, and of the linguistic means of connection on the other. Answering this need, theoretical specifications are presented in R1.2.2a.

The linguistic functional categories underlying the PENMAN and the KOMET system are suitable for the categories needed for the representation of CHAINING at the level of text. Additionally, a second level of interpretation will be placed above the level of sentences and therefore new functional categories will be used within this representation. This new interpretation requires the attachment to different kind of language units (fragments of the text).

A final set of remarks addresses the sentence planning language SPL of PENMAN and further refinements to it in the context of the project. Some of these points are also addressed in Bateman's contribution from Darmstadt's GMD/IPSI in this report:

In the implementation of *inquiries* in the present system, there is a distinction between **-id** and **-q** inquiries: As we understand it, an **-id** inquiry yields as a result the identification of an instance of the type in question, i.e. the member of a set. In our application these would be entities such as indexed referents or lexical items. A **-q** inquiry yields a further subclass of the semantic type in question. In our current Theme inquiries for German (cf. R.2.1.3.), we therefore have predominantly **-q** inquiries, because we are handling the semantic counterpart of a grammatical classification. It is only when we get down to the conflation of some type of Theme with one specific referent, or other lexicogrammatical token, that we get an **-id** inquiry. This understanding should be realized in a declarative implementation of the inquiries.

The second remark relates to the type of information modelled in interface notations as mentioned in Bateman's contribution to this report. Although, as mentioned above, SPL (semantic) input specifications are relatively independent from surface form compared to other notations, nevertheless some kind of isomorphism between the SPL hierarchy and syntactic head ranking/dependency can be observed which has been criticized by Bateman in earlier remarks on the implementation. To illustrate this point we will document one instance of this kind of problem we came across in the context of our first implementations of grammatical Theme into the KOMET grammar component.

After the definition the systems, choosers and inquiries controlling the generation of *identifying Themes* as exemplified by sentences like

(3) Was wir bauen ist ein Haus.
(What we are building is a house.)

we had to construct a respective SPL expression appropriate as input for the grammar to generate the intended type of Theme construction. Ideally, this would have been something like the following SPL expression (features controlling other grammatical areas like determination and lexicalization omitted here):

```
(s1 / dispositive-material-action

    :identifying-theme-q identifying :theme h :actee (h / haus)
    :actor (wir / person)  )
```

The problem with the current definitions of the semantic mappings in the grammar components is that they tend to enforce an isomorphism between the dependency structure of the SPL and the syntactic dependency structure of the sentence to be generated, although, in its true sense, SPL input specifications should not make any commitments to syntactic dependency structure of the structures generated at all. Therefore it is currently not possible to specify the instance of the *dispositive-material-action* as lexicalized with 'bauen' ('building') as the semantic head of the construction and leave it to the new Theme component to generate the respective identifying grammatical construction with the *class-ascription* 'ist' ('is') functioning as the syntactic head. Instead, we had to specify an input expression like the following in order to test the functionality of the identifying Theme implementation in the grammar:

```
(s1 / class-ascription

      :identifying-theme-q identifying :domain (s2 /
      dispositive-material-action :actor (wir / person) :actee (h /
      haus)

      :range (h / haus) )
```

What is not satisfying with this solution is that at the moment one has to express via a logical/ideational configuration (this is what SPL expressions are to represent) what is intended as a reflection of textual organization only.

One approach introducing ranking of heads in order to overcome this problem has been described in (cf. [1]). It could be used at least in our specification language.

Another issue related to the current SPL handling concerns the features controlling textual variations as mentioned in Bateman's contribution above. Although textual variation can be accounted for in the input specifications features specified in SPL expression currently do not explicitly make reference to the different metafunctional resources they control and access in the generation process, i.e., ideational, interpersonal, and textual information gets somewhat mixed in SPL expressions. Therefore the project should develop a strategy to overcome the mixing of heterogeneous types of information in current SPLs by a separation of these distinct types into subtypes within SPLs.

One last question, touching on an issue of basic philosophy underlying the events described in SPLs: As compared to, say DRT or Situation Semantics, relatively little attention is paid to the "part-whole" relationship between events. This is so because questions of (non-) identity, implication, contradiction between events are relatively non-central to a functional approach. That does not mean they are non-existent, though. We also need them for a treatment of time and space, and maybe for decisions between different types of 'fidelity' among translations. Ways should therefore be developed within our current tools to strengthen these notions.

5 Conclusion

We repeat here the conclusion from the second internal version of this deliverable (R2.1.1b):

> Work in the period reported on here has proceeded very much as planned in the technical annex. The first few months were predominantly used to take stock of the available systems, and to install them where they had not been installed previously. The period following that first one allowed us to begin to specify extensions and refinements, in the case of PENMAN/KOMET even in some detail. One of the methodological steps that some parts of the project have taken, that of experimentally working *vertically* on some task, rather than exclusively following the *horizontal* structure of the work packages, has proved helpful in some respects: we have been able to identify questions of implementation even during that first stage of work (cf. R2.1.3.) which in turn is a prerequisite for addressing some issues concerning the further development of our linguistic resources.
>
> We can also see by now certain synergetic effects developing out of the use of one of the systems (PENMAN) for implementations in all three of the project's languages in Darmstadt and Saarbrücken. These are further strengthened by contacts with groups outside DANDELION, using the PENMAN architecture to explore a whole range of different languages.

To this, we would add that these synergies continue to be active in the project, as described in the Introduction. On the basis of the resources identified in this document, we seem well-poised to integrate the theoretical linguistic advances in these resources, in developments of them for which the project has so far been responsible and in future empirical work, with a workbench able to provide genuine support and assistance to workers in the area of discourse.

A The Pleuk Grammar Development System

The following is based primarily on [3].

Pleuk aims to provide a framework within which to embed systems for research in computational linguistics. To this end, while it provides a general organizing system for formalisms for linguistic description and analysis, it attempts not to prejudge any issues to do with the syntax and semantics of such formalisms. For this reason, and also because of experience accumulated with University of Edinburgh in the integration of linguistic formalisms within Pleuk, it has good credentials as far as forming the framework within which the workbench of R2.2.2 is to be developed. As an indication of the generality of Pleuk, the following formalisms are now available:

Cfg A simple context-free grammar system, intended for demonstration purposes.

HPSG-PL A system for developing HPSG-style grammars, produced at Simon Fraser University, Canada, by Fred Popowich, Sandi Kodric and Carl Vogel.

Mike A simple graph-based unification system, enhanced with additional operations for the treatment of free word order proposed by Mike Reape in various publications.

SLE A graph-based formalism enhanced with arbitrary relations in the manner of [10] and [8]. Delayed evaluation is used to compute infinite relations. This system has been used for the development of several HPSG-style grammars.

Term A term-based unification grammar system, originally developed for the support of Unification Categorial Grammar (Zeevat, Klein and Calder).

Ale See above.

vNTag Tree Adjoining Grammars by Gertjan van Noord of Groningen University.

DCG for Definite Clause Grammars, [15].

CUF The *Comprehensive Unification Formalism* produced at Stuttgart, [8].

A.1 Using Pleuk in the Dandelion Project

It will be noted from the list above that two of the potential targets for the workbench, namely Ale and CUF, are now embedded within Pleuk. This suggests that the representational capacity of Pleuk is sufficient to accommodate a typed-feature system of the kind envisaged for the workbench. The crucial question to ask then is whether Pleuk can support the kinds of operations that a discourse analyst might require.

While these requirements are still under discussion, some are becoming clearer. First (and probably the most easily satisfiable) is that, while a directionality which excludes other possibilities is undesirable, it also seems sensible to exploit the facilities that are provided as basic. In the case of Ale, this means that it makes sense to suppose that the parser provided by the system will continue to exist within the workbench. (In the case of CUF, as it operates primarily by attempting to show the validity of some logical expression, analysis (or generation) is performed only as a side effect.)

From preliminary work to do with R2.2.1, it seems clear that we will expect some form of human intervention to be necessary in the course, say, of generating a text using the workbench. Some facilities exist already within Pleuk that allow the construction of derivations, namely the *derivation checker*, [2]. This system relies on the structurally given notion of derivation available within most linguistic theories, including (importantly) HPSG. It allows the graphical construction of derivations, with each step being monitored by the grammatical system in use to determine its legality. This system has been used with essentially no change in conjunction with the formalisms Cfg, HPSG-PL, Ale, DCG and vNTag.

The derivation checker is perhaps to be contrasted with systems in use for interaction with knowledge expressed in systemic-functional terms. In the former, more or less any choice may be made at any point in a derivation. In the latter kind, the choices available at some point are limited by previous choices, in the context of a particular traversal of a network, and are directed by the user only to the extent that the system contains an explicit instruction to query the user. If Pleuk is chosen as a starting point for the development of the workbench, the adaptability of the existing system will have to be assessed. In this connection, it is at least interesting to note that [6] presents an analysis of HPSG-type grammars as systems-of-choice, and also develops a parsing algorithm which exploits this construal of HPSG-like frameworks.

References

[1] John A. Bateman. Upper modelling for machine translation: a level of abstraction for preserving meaning. Technical report, IAI-Eurotra-D, Saarbruecken, FRG, 1989.

[2] Jo Calder. Graphical Interaction with Constraint-Based Grammars. In *Proceedings of the 3rd Pacific Rim Conference on Computational Linguistics*, Vancouver, April, 1993.

[3] Jo Calder, Kevin Humphreys and Mike Reape. Pleuk—a shell for grammar development systems, Unpublished ms., Human Communication Research Centre, University of Edinburgh, 1994.

[4] Bob Carpenter. *The Logic of Typed Feature Structures*, Cambridge Tracts in Theoretical Computer Science, Cambridge University Press, 1992.

[5] Bob Carpenter. *ALE: The Attribute Logic Engine user's guide, version β*, Laboratory for Computational Linguistics, Carnegie Mellon University, Pittsburgh, Pa, August, 1994.

[6] Richard P. Cooper. *Classification-based Phrase Structure Grammar: An Extended Revised Version of HPSG*, Ph.D. Thesis, University of Edinburgh, Centre for Cognitive Science, 1990.

[7] Liesbeth Degand. Towards a systemic functional grammar of dutch for multilingual text generation. Technical report, GMD/Institut für Integrierte Publikations- und Informationssysteme, Darmstadt, Germany, 1993. (Available in abbreviated form in the Proceedings of the Fourth European Workshop on Natural Language Generation, Pisa, Italy, 28-30 April 1993).

[8] Michael Dorna. *The CUF User's Manual* Institut für maschinelle Sprachverarbeitung, Universität Stuttgart, 1994.

[9] Helmut Horacek, Robert Dale, David D. McDonald, Dietmar Rösner, and Eduard Hovy. Special topic: The initial specifications for generation. In *New concepts in natural language generation*, chapter 10, pages 269 – 287. Pinter Publishers, London, 1993. (A collection of position papers).

[10] Rod Johnson and Michael Rosner. A rich environment for experimentation with unification grammars. In *Proceedings of the Fourth European Meeting of the ACL*, pp182–189, Manchester, April 1989.

[11] Robert T. Kasper. An experimental parser for systemic grammars. In *COLING 1988*, Budapest, University of Budapest, 1988.

[12] Christian M.I.M. Matthiessen. Lexicogrammatical cartography: English systems. Technical report, University of Sydney, Linguistics Department, 1992. Ongoing expanding draft.

[13] David McDonald. Issues in the choice of a source for natural language generation. *Computational Linguistics*, March 1993.

[14] John Nerbonne, Klaus Netter and Carl Pollard (eds). *German in Head-Driven Phrase Structure Grammar* CSLI Lecture Note Series, number 46, 1994.

[15] Fernando C. N. Pereira and David H. D. Warren. Definite Clause Grammars for Language Analysis — A Survey of the Formalism and a Comparison with Augmented Transition Grammars *Artificial Intelligence*, 13, pp231–278, 1980.

[16] Carl J. Pollard and Ivan A. Sag. *Head-driven Phrase Structure Grammar*, University of Chicago Press, 1994.

[17] Gerrit Rentier. A Lexicalist Approach to Dutch Cross Serial Dependencies, in *Proceedings of the Chicago Linguistic Society*, vol. 30, 1994.

[18] Elke Teich. Komet: grammar documentation. Technical report, GMD/Institut für Integrierte Publikations- und Informationssysteme, Darmstadt, West Germany, 1992.