A Generative Perspective on Verb Alternations

Manfred Stede*
Technische Universität Berlin

Verb alternations have been researched extensively in linguistics, but they have not yet received a systematic treatment in natural language generation systems; consequently, generators cannot make informed choices among alternatives. As a step towards overcoming this discrepancy, we review some linguistic work on several prominent alternations, revise and extend it, and suggest a set of rules that allow the series of alternated forms to be produced from a single base form of the verb, the lexical entry. The framework has been implemented in the MOOSE *sentence generator, which can thus choose a particular verb alternation in order to accomplish generation goals such as placing emphasis on the most important element of the sentence.*

1. Introduction

In this paper, we approach the problem of verb alternations from the perspective of knowledge-based natural language generation (NLG), which aims at producing a set of verbalizations from a common underlying representation. This viewpoint places some specific requirements on the nature of lexical representations, which will be explained in Section 2. Thereafter, we will investigate the problem of systematically generating a number of verb alternations—a problem that so far has received little attention in the NLG community.

Why should a language generator have knowledge about producing alternations? Because a sophisticated discourse production may call for using a verb in one or another configuration, depending on the current situation of utterance. Alternations can place the emphasis on different elements of the sentence, and distribution of emphasis is influenced by the development of the discourse—and thus related to a whole range of other generation decisions. To illustrate, consider the following alternative beginnings of a little story:

(1) (a) Tom was in a hurry, but he had to change the oil before hitting the road. He crawled under the car and unscrewed the drain bolt. The engine *drained* in 20 seconds....

(b) Time was short, but the oil had to be changed before Tom could hit the road. Within 20 seconds, he *drained* the engine. Then ...

(c) Tom was in a hurry, but he had to change the oil before hitting the road. Quickly, he crawled under the car and unscrewed the drain bolt. For 20 long seconds, the oil *drained* from the engine....

* TU Berlin, FB Informatik, Sekr. FR 6–10, Franklinstr. 28/29, 10587 Berlin, Germany

(d) Tom was in a hurry, but he had to change the oil before hitting the road. Crawling under the car, he *drained* the old oil from the engine, and then ...

Depending on how the story develops, which might hinge on stylistic parameters, a different configuration of *to drain*, with different subjects, objects, and prepositional phrases, should be produced. Therefore, if an NLG system is expected to be able to cope with such differences, it needs knowledge of what alternations are possible for a given verb, and how the different syntactic configurations relate to differences in meaning.

In linguistics, the central goal of research on alternations is to uncover the relationships between syntax and semantics (linking rules), and to form classifications of verbs according to their alternation behavior (Levin 1993). To accomplish these goals, the need for fine-grained lexical-semantic representations is pointed out, although there is no strong consensus yet on exactly what such representations should look like (see the discussion in Levin and Rappaport Hovav [1995, chapter 1]). NLG, in any case, needs representations to work with; and in order to account for verb alternations, we need to devise rather fine-grained ones. In particular, a generator has to relate the (possible) changes in meaning to the changes in form, so that—from a given representation—the correct set of "alternated verb forms" can be produced. In other words, the generator needs to know the conditions under which some input representation licenses the use of a specific alternation.

As a step in this direction, we consider a number of alternations that affect the aspectual category (or *Aktionsart*) of the verb—a group that Levin (1993, 12) chose not to focus on. We look at the differences in meaning that coincide with such alternations, and propose suitable representations for input specifications (the underlying ontology), for verb meaning, and for the alternation rules. These rules are able to sequentially derive the various alternated forms from a single base form, which is stated in the lexical entry. The approach has been implemented in a bilingual generation system, which can produce the "alternated paraphrases" in English and German. To demonstrate its capabilities, we will show how a salience parameter associated with the input can give rise to selecting one or another of the alternatives; specifically, the production of the alternative *drain* sentences (1a–d), from a common underlying representation, will be demonstrated.

The paper is organized as follows: Section 2 discusses the specific requirements of lexical information in NLG, focusing on verbs, and suggests a format for dictionary entries. Section 3 develops our approach to verb alternations and proposes rules that account for several such alternations. Section 4 describes the implementation of the alternation framework in the MOOSE generator, and Section 5 concludes and compares our approach to related research. While Sections 2 and 3 are concerned with linguistic representations and make only little reference to NLG, Section 4 presupposes some knowledge of the generation concepts and systems that MOOSE is built upon.

2. Representing Verb Meaning for Generation

2.1 Lexical Information

The central characteristic of knowledge-based NLG is the existence of a **domain model**, which anchors the representations serving as input to the generator. The domain model provides the basis for drawing inferences on the representations. One class of such inferences is the **subsumption check**, which our approach exploits for the task of language generation (see Section 4). The NLG system is thus in charge of mapping a network of instantiated domain concepts to linguistic utterances, and the specific role of lexemes within this process is to "carry over" the meaning from the underlying domain model to utterances in natural language. To accomplish this step, the lexicon of the generator requires two basic components: it has to explain what words mean with respect to the domain model, and it has to explain how words can be combined into well-formed sentences. At this point, the study of lexical semantics becomes relevant—it should systematically relate these two tasks.

In NLG, however, lexical semantics has for a long time been relatively neglected; words and concepts were often conveniently put into a simple one-to-one correspondence. The key to incorporating lexical semantics into NLG is breaking up this tight correspondence and arranging for more flexible mappings. As soon as entities in the knowledge representation scheme do not correspond to words in a direct manner, the relationship between word meaning and entities in the knowledge base (KB) needs to be specified in some more elaborate way. Now, lexical semantics has to supply the interface between knowledge and words: It has to specify what words can be used to express what parts of what KB entities, and, possibly, under what circumstances. To this end, the relevant work in linguistics needs to be identified, extended, and adapted for generation purposes. This adaptation is not straightforward, however, because the starting position of linguistics is different from that of NLG: For a linguist, the syntax-semantics interface is of central concern, whereas in NLG, there is the additional level of instantiated knowledge, which needs to be systematically related to linguistic levels. Our approach is to employ a level of sentence-semantic representation that mediates between the knowledge-level input and syntactic realization. A well-motivated semantic level allows us to encapsulate all syntactic decisions in a front-end generation module; we use the Penman system (Penman Group 1989) for this purpose. For the other mapping, from KB to sentence-semantics, we use the lexicon as the primary source of information.

In this approach, lexical entries therefore have two main components: a **denotation** that defines the applicability conditions of the lexeme with respect to the domain model (i.e., it can be matched against the generator's input), and a **partial semantic specification** (PSemSpec), which specifies the contribution that the lexeme makes to sentence meaning.[1] The task of lexicalization in our generator thus consists of first finding lexemes that can convey parts of the input, and then determining the preferred combination of candidate lexemes, yielding a sentence-semantic specification (SemSpec), which is then processed by the surface generator. The generation architecture is presented in Section 4.

The next two sections explain the denotation and the partial semantic specification associated with verb lexicon entries, and thereby also the two levels of representation used in the generation system.

2.2 Event Ontology and Aktionsart

The development of the domain model and the underlying ontology for our system focused on the treatment of events so that they can be appropriately verbalized in different languages. The hierarchy of **situations**, shown in Figure 1, is organized along a variant of the ontological categories proposed by Vendler (1967) and developed further by Bach (1986), inter alia. We briefly discuss the three types of situation in turn.

1 The lexical entries in our system have several other components, which are listed in Section 2.4.

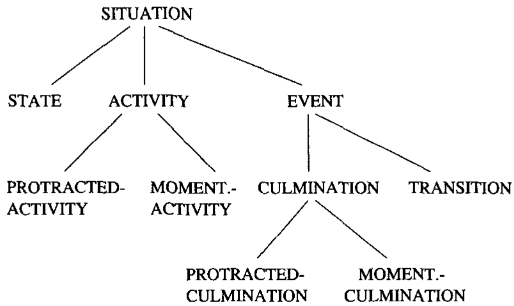

Figure 1
Situation types in the ontology of MOOSE.

States are seen much in the same way as Bach sees them: Something is attributed to an object for some period of time, and the object is not perceived as "doing" anything. *The bottle is empty* is true for the bottle without it doing anything about it. We do not make further distinctions among states here.

Activities were called "processes" by Bach, but we need this term on a different level of description (see below). They are quite similar to states, but there is always something "going on," as in *The water was flowing toward the sea*. We distinguish two subtypes here: **protracted activities** take place over an extended period of time, whereas **momentaneous activities** occur in an instant; a "point adverbial" such as *at noon* serves as a linguistic test.

Events are occurrences that have a structure to them; in particular, their result, or their coming to an end is included in them: *to destroy a building, to write a book*. As their central feature we take them to always involve some change of state: the building loses its integrity, the book comes into existence, or gets finished. While Bach (1986) did not investigate the internal structure of events, others suggested that this needs to be done (e.g., Moens and Steedman 1988; Parsons 1990). Pustejovsky (1991) treated Vendlerian accomplishments and achievements as transitions from a state Q(y) to NOT-Q(y), and suggested that accomplishments in addition have an intrinsic agent performing an activity that brings about the change of state.

We follow this line, but modify it in some ways. Basically, we see any event as involving a change of state; an activity responsible for the change can *optionally* be present. A plain **transition** is necessarily momentaneous (*The room lit up*), whereas a transition-with-activity inherits its protracted/momentaneous feature from the embedded activity. We call these tripartite events **culminations**.[2] They are composed of a pre-state (holding before the event commences), a post-state (holding when the event is over), and an optional activity that brings the transition about. Generalizing from Pustejovsky's proposal, we take state transitions to be more than merely oppositions of Q(y) and NOT-Q(y); they can also amount to a gradual change on some scale, or involve other values. Also in contrast to Pustejovsky, we do not regard the presence of a volitional agent as responsible for any of the category distinctions; rather, the agentivity feature cuts across the categories discussed. Other aspects of our ontology are designed following proposals by Jackendoff (1990), in particular his analysis of movement events.

2 Moens and Steedman (1988) also use this term, but they restrict it to momentaneous events. Unfortunately, the terminology used in the literature for these kinds of categories varies so much that a standardization seems out of reach.

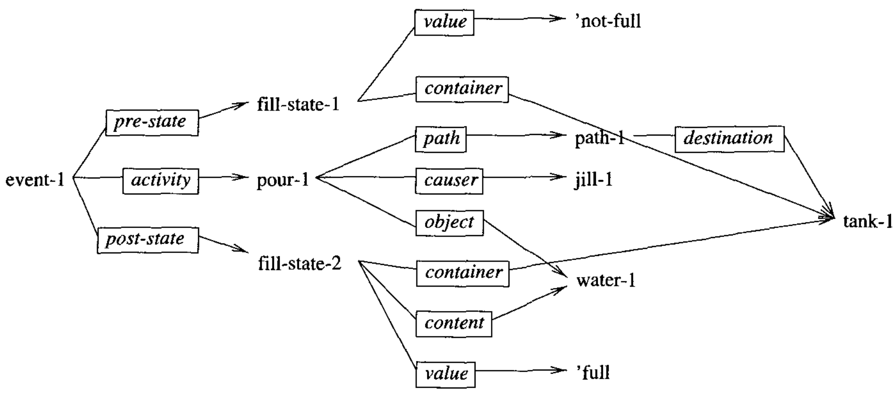

Figure 2
SitSpec representing a fill-event.

Subsumed by the general ontological system, a domain model is defined that holds the concepts relevant for representing situations and that specifies the exact conditions for their well-formedness. We use the term **SitSpec** for a network of instances of domain model concepts, which will be the input to our generator. The root node of any SitSpec is of the type situation. As an example, the event of a person named Jill filling a tank with water is shown in Figure 2 in a graphical KL-ONE notation (Brachman and Schmolze 1985), with relation names appearing in boxes. The event combines the activity of Jill pouring water into the tank with the fill-state of the tank changing to full. A verbalization of this event can emphasize either of these aspects.

Since we decompose event structure in such a way, it follows that the denotations of verbs for verbalizing events need to be fairly complex. The type of event denoted relates to the **Aktionsart** of the verb: the inherent features characterizing (primarily) the temporal distribution of the event denoted.[3] A generator needs to know these features when verbalizing different kinds of events, so that it can produce (for example) the correct temporal modifier to express the duration of either an activity or a culmination. The variety of phenomena in Aktionsart are far from clear-cut, and there is no generally accepted and well-defined set of features. In the following, we use the terms given by Bussmann (1983) and discuss only those Aktionsart features that are directly relevant for us because they relate types of situations to denotations of verbs. Thus, within the context of our system, we define Aktionsart features in terms of patterns of verb denotations. Table 1 lists the correspondences.

Simple cases are **stative** verbs like *to own* or *to know*. **Durative** verbs characterize continuous occurrences that do not have internal structure, like *to sleep, to sit*. In the class of nondurative verbs we find the **semelfactive** ones, which denote a single occurrence, thus in our system a momentaneous activity, as, for example, *to knock*. Interestingly, an iterative reading can be enforced on a semelfactive verb by a durative adverbial: *She poked me for an hour*. **Transformative** verbs involve a change of some state, without a clearly recognizable event that would be responsible for it: *The room lit up*. The denotation of such verbs thus involves a pre-state and a post-state. In our ontology, these are transitions. **Resultative** verbs, on the other hand, characterize situations in

3 This is often treated on a par with *aspect*, but we prefer to make a terminological distinction between the grammaticalized categories such as progressive versus nonprogressive in English (aspect), and the static verb-inherent features.

Table 1
Correspondences between Aktionsart and denotations.

Aktionsart	Denotation pattern
stative	(state X)
durative	(protracted-activity X)
semelfactive	(momentaneous-activity X)
transformative	(event (PRE-STATE X) (POST-STATE not-X))
resultative	(event (ACTIVITY X) (POST-STATE Y)
causative	(activity (CAUSER X))

which something is going on and then comes to an end, thereby resulting in some new state (culminations in our ontology). Their denotation includes an activity and a post-state. In the literature, such verbs are often also called **inchoative**.[4] The final verb-inherent feature we use is the well-known **causative**, which reflects the presence of a causer in the denotation (as in Figure 2).

Verb alternations, as we will discuss them shortly, can involve a shift in Aktionsart and thus a systematic change of the denotation. But first we have to introduce the second major component of verb semantics—sentence meaning.

2.3 Sentence Meaning

A SitSpec representing a possibly complex event structure can be verbalized by a variety of sentences, which can differ in terms of their argument structure, aspectual composition, etc. From the viewpoint of NLG, we wish to select the most appropriate sentence on the grounds of target parameters, such as the salience assignment mentioned in the beginning of the paper. In order to produce a sentence that accomplishes semantic goals of this kind, it is impractical to map the very abstract SitSpec directly to a syntactic structure. Instead, we use a sentence-semantic level of description that allows us, on the one hand, to control those generation decisions that affect the meaning of the sentence, and, on the other hand, to encapsulate the syntactic realization decisions in the front-end generation grammar.

2.3.1 Halliday's Ideational Structure. To describe sentence meaning, we use the "ideational structure" introduced by Halliday (1985). It resembles other approaches based on semantic case roles, but an important feature of Halliday's work is his thorough classification of process types and of the semantic relationships holding between the verb and the other elements in a clause.[5] This extensive analysis renders the approach particularly useful for sentence generation. Halliday's process classification has been further developed for NLG purposes by C. Matthiessen, J. Bateman and others (see, for instance, Matthiessen and Bateman [1991]). The resulting "upper model" (UM) is part of the Penman generator and used in our system as well. The UM is a taxonomy of

4 The term inchoative is used to cover a rather broad range of phenomena, including the beginning of an event (e.g., *to inflame*) or its coming to an end. We think the term is overloaded and prefer to use resultative for the latter group.

5 Halliday proposes two additional levels of sentence description ("metafunctions"), which operate in parallel to ideational structure: the interpersonal and the textual. For our present purposes, we can neglect them; for a broader scope of sentence generation, they are very important.

Jill poured water into the tank until it was filled.
```
(x1 / anterior-extremal
  :domain (x2 / directed-action :lex pour
           :actor (x3 / person :name jill)
           :actee (x4 / substance :lex water)
           :destination (x5 / three-d-location :lex tank))
  :range (x6 / nondirected-action :lex fill
           :actee x5))
```

Figure 3
SemSpec and a corresponding sentence.

linguistic categories that directs the grammar in verbalizing objects (in the generator's input) in terms of these categories. Hence, the UM can be characterized as mirroring the distinctions made in surface linguistic realizations: Typically, any two distinct UM types correspond to some difference in English sentences.

The largest part of the UM is devoted to processes, which are characterized by their verbalization patterns. For our purposes here, we need only a small fragment of the process hierarchy, namely the subtree of **material processes**. They can be characterized by the fact that English verbalizations of them in present tense typically use the progressive form, as in *the house is collapsing* (unmarked) as opposed to *the house collapses* (marked). They typically involve the participant roles "actor" and "actee"[6] but differ in terms of constraints on the types of the role fillers, and with respect to their realization in language. Material processes have two subgroups, one of which are **nondirected-actions**. They do not involve external agency and are mostly intransitive. With such processes, the actee is not a genuine participant, but rather an elaboration of the process. Verbs falling into this category are those of movement, of expressing skills, as well as support verbs like *to take* as in *take a shower*. The other subgroup, **directed-actions**, are always transitive, and they involve an external agent of the process.

The upper model thus reflects the semantic distinctions made by the language, and the systemic-functional grammar takes care of the syntactic realization of these distinctions. Accordingly, the lexicalization component we are proposing here is in charge of producing a sentence-semantic specification along the lines of ideational structure (using the upper model categories), such that the relevant decisions affecting sentence meaning can be controlled during lexical choice. As an example, Figure 3 shows one of the SemSpecs and an English sentence that can be derived (as explained in Section 4) from the SitSpec given in Figure 2. Besides actor and actee, the role "destination" is used in the SemSpec; later we will also encounter "source."

The UM is a good starting point, but in some respects the process classification is not quite fine-grained enough. A deficiency that is directly relevant for our treatment of alternations concerns the **valency** patterns of verbs, where some additional distinctions are needed.

2.3.2 Valency. As introduced by Tesnière (1959), valency refers to the distinction between **actants** and **circumstantials** (central participants associated with the verb versus temporal, locational, and other circumstances). This separation is in principle widely accepted, but views differ on where to draw the line and how to motivate it. The notion of valency was further developed predominantly in German linguistics, with a culmination point being the valency dictionary of German verbs by Helbig and

6 Actee is the upper model role that conflates what more often is called patient, theme, and goal.

Schenkel (1973). They made an additional distinction between obligatory and optional actants; Somers (1987, chapter 1) proceeded to propose six different levels of valency binding. He also pointed out that there are different opinions on the *type* of entities that are subject to a verb's valency requirements: some authors describe them by syntactic class, some by semantic deep cases, and some by their function (subject, object, etc.).

Halliday (1985), in his classification, essentially adopts the basic Tesnièrian distinction and suggests some semantic and syntactic criteria for deciding between actants, which he calls participants, and circumstances. Spatio-temporal information, for instance, is generally treated as a circumstance. As a syntactic indicator, for Halliday, participants are typically realized as nominal groups (with some obvious exceptions, as in *say that x*), and circumstances as prepositional phrases or as adverbs. But neither this syntactic division corresponding to participants and circumstances (direct or indirect object versus adverbs or prepositional phrases), nor the semantic postulate that spatio-temporal aspects are circumstances, holds in general. Regarding spatial relationships, we find verbs that specifically require path expressions, which cannot be treated on a par with circumstances: Consider, for example, *to put*, which requires a direct object and a destination. Causative *to pour* requires a direct object as well as a path with either a source, or a destination, or both: *pour the water from the can into the bucket.*[7] Some verbs, as is well-known, can occur with either a path (*Tom walked into the garden*) or a place (*Tom walked in the garden*), and only *in the garden* can here be treated as a circumstance. And *to disconnect* requires a direct object (the entity that is disconnected) and a source (the entity that something is disconnected from), which can be omitted if it is obvious from the context: *Disconnect the wire!*

As a step toward a more fine-grained distinction between participants and circumstances, we adopt the three categories proposed by Helbig and Schenkel (1973) and thus distinguish between obligatory and optional participants on the one hand, and circumstances on the other. Moreover, we differentiate between requirements of process types (as encoded in the process taxonomy) and requirements of individual verbs, which are to be encoded in the lexical entries. In a nutshell, valency (as a lexical property) supplements the participant/circumstance requirements that can be stated for types of processes.

To encode the valency information, we introduce the partial semantic specification (PSemSpec) as one central component of lexical entries. The participant roles stated in the PSemSpec are either obligatory or optional; in the latter case they are marked with angle brackets:

```
to disconnect
PSS: (x / directed-action
       :actor A  :actee B  < :source C >)
```

With obligatory participants, the verb is only applicable if the elements denoted by these participants are present in the input structure to be verbalized (the SitSpec). Optional participants need not be included in the verbalization: If they are present in the SitSpec, they may be omitted if there is some good reason (e.g., a stylistic preference); if they are not present in the SitSpec, the verb can be used anyway. The *disconnect* example illustrates that, in contrast to Halliday, we allow for verbs selecting path expressions, here as an optional complement. We can thus distinguish between

7 Given a suitable context, though, the sentence *She poured the wine* is perfectly acceptable. But this usage seems to be restricted to a small class of digestible liquids.

cases like the following:

- *Tom disconnected the wire {from the plug}. To disconnect* requires a source, but it can be omitted in a suitable specific context.
- *Sally ate.* While *to eat* usually requires a direct object, it can also be used intransitively due to the strong semantic expectation it creates on the nature of the object—*independent* of the context.
- *Tom put the book on the table. To put* requires a destination, and it cannot be omitted, no matter how specific the context.
- *The water drained from the tank {in the garage}.* Locative circumstances like *in the garage* are not restricted to particular verbs and can occur in addition to paths required by the verb.

The criterion of optionality, as indicated above, singles out the obligatory complements from the other two categories. But how, exactly, can we motivate the distinction between optional participants and circumstances in our framework? By relating the PSemSpec to the SitSpec, via the denotation. In the *disconnect* case, for instance, the two items connector and connectee are both integral elements of the situation. The situation would not be well-formed with either of them absent, and the domain model encodes this restriction. Therefore, both elements also occur in the denotation of *to disconnect,* and a coindexed variable provides the link to the PSemSpec. Only when building the sentence SemSpec is it relevant to know that the connectee can be omitted. The connectee in the denotation therefore must have its counterpart in the PSemSpec—that is the source, but there it is marked as optional (see Figure 4 below).

With circumstances, the situation is different: A SitSpec is complete and well-formed without the information on, for instance, the location of an event. Hence, a verb's denotation cannot contain that information, and it follows that it is not present in the PSemSpec, either.

2.4 Lexical Entries

We have introduced the two central components of lexical entries and now give a complete list of the components used in our system. The connotations are not directly relevant for our mechanism of handling verb alternations, therefore they will not be dealt with here. Salience assignment will be discussed in Section 4.

Denotation: A partial SitSpec that defines the **applicability condition** of the lexeme: If its denotation subsumes some part of the input SitSpec, then (and only then) it is a candidate lexical option for the verbalization.

Covering: The subset of the denotation nodes that are actually expressed by the lexeme. One of the constraints for sentence production is that every node be covered by some lexeme.

Partial SemSpec (PSemSpec): The contribution that the lexeme can make to a sentence SemSpec. By means of shared variables, the partial SemSpec is linked to the denotation.

Connotations: Stylistic features pertaining to formality, floridity, etc.

Salience assignment (for verbs only): A specification of the different degrees of prominence that the verb assigns to the participants.

DISCONNECT CAUSER → :actor CONNECTOR → :actee CONNECTEE → <:source>	**OPEN** OBJECT → :actor *CAUSER ——— resultative-causative
POUR PATH-SOURCE → :actor OBJECT → <:actee> *PATH-DESTINATION *CAUSER ——— substance-source durative-causative	**SPRAY** CAUSER → :actor OBJECT → :actee PATH-DESTINATION → :destination ——— spray-load
DRAIN OBJECT → :actor PATH-SOURCE → <:source> *PATH-DESTINATION *CAUSER ——— durative-causative locative/clear-intransitive resultative-causative	**FILL** CONTENT → :actor CONTAINER → :actee VALUE → <:destination (default)> *CAUSER ——— stative-resultative resultative-causative
MOVE/WALK OBJECT → :actor *PATH *CAUSER ——— durative-causative	**LEAK** PATH-SOURCE → :actor OBJECT → <:actee> *PATH-DESTINATION ——— substance-source

Figure 4
Excerpts from sample lexical entries for verbs.

Alternation rules: (for verbs only): Pointers to lexical rules that represent alternations the verb can undergo (see Section 3).

Morphosyntactic features: Standard features needed by the surface generator to produce correct utterances.

Figure 4 gives excerpts from sample lexical entries, which demonstrate the linking between entities from the denotation and the PSemSpec. Notice that the linking is shown for the base form of the verb, which can be quite simple, as in *open* or *move*. Items appearing with an asterisk in front of them are *optional in the SitSpec*: for example, a SitSpec underlying an open-event is well-formed without a causer being present. These items get verbalized with the help of rules such as the alternation rules to which we turn in the next section. In the lexical entries, the names of applicable alternation rules are listed below the line.

3. Alternations

Having explained denotations and PSemSpecs, specifically for verbs, we can now turn to the task of accounting for the different alternations a verb can undergo. Under this heading, we will look both at the so-called transitivity alternations, which are characterized by a change in the number of participants (e.g., the causative), and at diatheses, which only affect the mapping between the participants and syntactic realization (e.g., the passive). Thus, a variant such as topicalization does not qualify as an alternation, since the syntactic realization of the participants remains unchanged; they are merely reordered. The most comprehensive source of information on alternations is the compilation by Levin (1993); we will now look at some of the more prominent alternations listed there and characterize them in terms of changes in denotation and valency of the verbs.

3.1 Alternations as Meaning Extensions

A simple way of treating alternations is to use a separate lexical entry for every configuration, but that would clearly miss the linguistic generalizations. Instead, we wish to represent the common "kernel" of the different configurations only once, and use a set of lexical rules to derive the alternation possibilities. Jackendoff (1990) is concerned with this problem for a number of alternations; specifically, in his framework of lexical-conceptual structure (LCS) he seeks to explain the relationships between stative, inchoative, and causative readings of a verb. In Jackendoff's analysis, the forms are derived sequentially by embedding in the primitives INCH and CAUSE, respectively:

- stative: $BE([_{Thing}\ ]_{\langle A\rangle}, [IN_d\ [_{Thing}\ ]_A\ ])$
- inchoative: $INCH\ [BE([_{Thing}\ ]_{\langle A\rangle}, [IN_d\ [_{Thing}\ ]_A\ ])]$
- causative: $CAUSE([_{Thing}\ ]_A, INCH\ [BE([_{Thing}\ ]_{\langle A\rangle}, [IN_d\ [_{Thing}\ ]_A\ ])]$

For our NLG purposes, the idea of deriving complex verb configurations from more basic ones is attractive, but it is necessary that we relate verb meaning to our explicit treatment of event structure, instead of masking that structure with a primitive such as INCH. When verbalizing a SitSpec, we first have to determine candidate lexemes, i.e., match the SitSpec against lexicon entries; having only one lexical entry for a verb reduces the search space considerably. Moreover, since the verb entry will be the most basic form, its denotation is relatively simple and therefore the matching is inexpensive. Finding more complex verb configurations will then require some further matching, but *only* locally and to those verbs that have already been determined as verbalization options.

In general, the idea is to see verb alternations not just as relations between different verb forms, but to add directionality to the concept of alternation and treat them as functions that map one into another. From this viewpoint, there are two groups of alternations: (1) Alternations that do not affect the denotation of the verb. Examples are the passive or the substance-source alternation (*The tank leaked oil; Oil leaked from the tank*): The truth conditions do not change. (2) Alternations that do change the denotation of the verb.

The second group is the critical one, because if we derive verb configurations from others and rewrite the denotation in this process, it has to be ensured that the process is monotonic.[8] Therefore we define directionality for group (2) to the effect that an alternation always *adds* meaning: The newly derived form communicates more than the old form—the denotation gets extended. This notion is different from the standard, nondirectional way in which alternations are seen in linguistics; to label the difference, we call alternations of group (2) **extensions**. In this section, we will introduce a number

8 Monotonicity might be too strong a requirement for alternations in general, though. See Section 5.3.

of extension rules for which we can give a clear definition in terms of Aktionsart features, as they were introduced in Section 2.2. These rules extend the denotation of a verb and rewrite its PSemSpec in parallel to reflect the change in valency; the result is a new verbalization option, which can differ from the previous one in terms of coverage or attribution of salience (see Section 4). The rules will be conveniently simple to state, thanks to the upper model, which provides the right level of abstraction from syntax.

To illustrate the goal, we return to the example of Tom removing the oil from an engine. If a SitSpec encodes this situation, then *to drain* is a candidate lexeme. While it can appear in a number of different configurations, we wish to match only one of its forms against the SitSpec, though. This is the most basic one, denoting an activity: *The oil drained from the engine.* Here, the case frame of the verb has to encode that *from the engine* is an optional constituent. Now, an extension rule has to systematically derive the causative form: *Tom drained the oil from the engine.* And also from the first configuration, another rule derives the resultative reading, which adds the information that the engine ended up empty: *The engine drained of the oil.* Here, *of the oil* is an optional constituent. To this last form, a causative extension can apply and yield *Tom drained the engine of the oil.*

To compute these configurations automatically, such that valency and meaning are changed in parallel, we define an alternation or extension rule as a 5-tuple with the following components:

`NAM`: a unique name;

`DXT`: extension of denotation;

`COV`: additions to the covering-list;

`ROC`: role changes in PSemSpec;

`NRO`: additional PSemSpec roles and fillers.

The `DXT` contains the denotation subgraph that the new verbalization has in addition to the old one. The syntax is, of course, the same as that of the denotation of a lexical entry. Specifically, it can contain variables; these can co-occur in the `COV` list: the items that the new verbalization covers appear in addition to those of the old one. `ROC` is a list of pairs that exchange participant role names or the UM type in the PSemSpec; this replacement can also change optionality. For example, `(< :actee > :actor)` means "replace the term `:actee` in the PSemSpec of the old verbalization, where it was optional, with `:actor`, which is not optional." Finally, `NRO` contains new roles and fillers that are to be added to the new PSemSpec; these will also contain variables from the denotation extension.

Applying such a rule to a verbalization option *vo* works as follows: Add the contents of `DXT` to the denotation of *vo*, and match the new part against the SitSpec. If it matches, make a copy *vo'* of *vo* and assign it a new name as well as the denotation just formed. Add the `COV` list, which has been instantiated by the matching, to the covering-list of *vo'*. Exchange the role names in the PSemSpec of *vo'* as prescribed by `ROC`, and, importantly, in the order they appear there. Finally, add `NRO` to the PSemSpec.

In the following, we first give an example for a rule that changes only the PSemSpec without affecting the denotation. Afterwards, we describe those alternations that change the Aktionsart of the verb and thus the form of the SitSpec expressed (as discussed in Section 2.2): the stative-resultative, causative, and locative alternations.

Before introducing these rules, it should be emphasized that we do not provide applicability conditions for the alternation and extension rules, which would inspect some verb denotation and on that basis decide whether an alternation can apply; instead, the rules are triggered directly from the lexical entry of a verb. Whether general applicability conditions can be specified, so that the rules need not be attached to each individual verb, is a central open research question that linguistic alternation research is concerned with.

3.2 Encoding Alternation Rules

The best-known alternation that affects only the valency of the verb is the passive, which we do not investigate here. Instead, we show one alternation that is particularly relevant for verbs in the domain of substances and containers.

Substance-Source Alternation. Example: *The tank leaked water / Water leaked from the tank.* This is an alternation discussed by Levin (1993); to make use of it here, we have to add directionality and declare one of the two configurations as more basic. Levin lists verbs of "substance emission" as undergoing it, for example *drip, radiate, sweat,* and *leak.*[9] To decide on the more basic form, we use the fact that in *The tank leaked water* the *water* is an optional constituent, and hence the minimal configuration of the verb is *The tank leaked.* With the *from* configuration, no deletion is possible.

As a representative of the verb class, we show the denotation and PSemSpec of *to leak*:

```
DEN: (leak (OBJECT A)
           (PATH (SOURCE B)))

PSS: (x / nondirected-action
       :lex leak :actor B  < :actee A >)
```

The following alternation rule applies to all these substance emission verbs and derives the *from* configuration:

```
NAM: substance-source
DXT: ()
COV: ()
ROC: ((:actor :source)
      (< :actee > :actor))
NRO: ()
```

Let us now consider several alternations that change denotation, and hence are extensions.

Stative-Resultative. Example: *Water filled the tank / The tank filled with water.* In discussing verbs that denote a state, Jackendoff (1990) points out that *fill, cover, surround,* and *saturate* can describe either a state or an inchoative event, and encodes the difference with the primitive INCH we have shown in the introduction to this section. Our goal is to do without the primitive, and to define the change in terms of the Aktionsart of the verb; to this end, we use resultative in place of inchoative (see Section 2.2).

9 Unnoticed by Levin, *to leak* can also be a verb of substance "intrusion," as in *The camera leaked light.* This reading, which we do not handle here, reverses the directionality of the path involved.

On a similar matter, Levin (1993) describes the "locatum subject" alternation, which for instance holds between *I filled the pail with water* and *Water filled the pail.* It thus relates a causative and a noncausative form. Levin states that the alternation applies to a class of "fill verbs," of which there are many more than the four given by Jackendoff. Her alternation is not exactly the one we need here, since it also involves a causative form; deriving the causative is a separate step in our framework.

What we need here is a mixture of Jackendoff's and Levin's insights: Several of Levin's fill verbs can be both transitive and intransitive, and some of the intransitive readings denote 'to become Xed'. Among these verbs are *fill, flood, soak, encrust,* or *saturate*: *The kitchen flooded with water* means the same as *The kitchen became flooded with water.* For this subgroup of the fill verbs, we define an extension rule that derives a resultative reading from a state reading.[10] Notice that this is different from Levin's locatum subject alternation, since it does not involve a causer.

```
NAM: stative-resultative
DXT: (event (Y (ACTIVITY X)))
COV: (X Y)
ROC: ((:actor :inclusive)
      (:actee :actor)
      (directed-action nondirected-action)
NRO: ()
```

To illustrate the rule with an example, consider the denotation and PSemSpec of the state reading of *fill*:

```
DEN: (fill-state (CONTAINER A)
                 (CONTENT B)
                 (VALUE C))

PSS: (x / directed-action :lex fill
        :actor B  :actee A  < :destination C >)
```

When matching it against a SitSpec with a tank and water, this yields the verbalization *The water filled the tank,* covering only the post-state of the SitSpec. Now, the alternation rule extends the denotation to also cover the event and the activity that brings the filling about. Applying the changes to the PSemSpec results in

```
(x / nondirected-action :lex fill
  :inclusive B :actor A < :destination C >)
```

which corresponds to the sentence *The tank filled with the water.*

A few stative verbs cannot be resultative without being also causative. Consider *to cover* in these examples from Jackendoff:

Snow covered the ground.

*The ground covered with snow.

Bill covered the ground with snow.

10 The two roles "inclusive" and "of-matter" (used later) are the roles used by Penman to realize the desired structure, but they are not very good descriptions of these semantic relationships. For a more systematic treatment, for instance along the lines of Somers (1987), the upper model needs to be extended. See Section 4.1.

For these, a stative-culmination extension derives the resultative + causative form directly from the stative one. The rule is similar to the one given above, so we do not show it here.

Causative Extensions. Example: *The napkin soaked / Tom soaked the napkin.* Levin discusses a causative/inchoative alternation that applies to a large number of verbs. The class formed by them is somewhat heterogeneous with respect to Aktionsart, though; it contains, for example, *to turn* as well as *to open.* The former is in its basic form durative (*The wheels turned*), and the latter transformative (*The door opened*). Accordingly, we split the alternation in two, which only differ in the DXT component, reflecting the difference in Aktionsart. The durative-causative extension adds a causer to the denotation and makes the former `:actor` the new `:actee`. It equally applies to semelfactive verbs denoting a momentaneous activity: *The bell rang / The visitor rang the bell.* The resultative-causative extension also covers the activity, because *Tom opened the door* expresses that Tom did something to achieve the change of state. The causer itself is not covered though, because it still has to be verbalized separately.

```
NAM: durative-causative
DXT: (activity (CAUSER X))
COV: ()
ROC: ((:actor :actee))
NRO: (:actor X)

NAM: resultative-causative
DXT: (event (ACTIVITY (X (CAUSER Y))))
COV: ()
ROC: ((:actor :actee))
NRO: (:actor Y)
```

The first rule derives, for example, *Tom walked the dog* from *The dog walked,* and the second *Tom closed the door* from *The door closed.*

Locative Extensions. Example: (a) *Sally sprayed the wall with paint.* / (b) *Sally sprayed paint onto the wall.* The locative alternation has been studied by lexical-semanticists extensively. Its characteristic is that configuration (a) of the verb conveys that something is performed in a "complete" or "holistic" manner, whereas configuration (b) lacks this facet of meaning. Levin points out that this alternation has received much attention in linguistics research and notes that, in spite of the efforts, a satisfactory definition of the holistic facet has not been found. Jackendoff, in his treatment of the alternation, suggests encoding the holistic feature in a primitive: The function ON_d is a derivative of ON and means that something "distributively" covers a surface, e.g., the paint covers all of the wall. Introducing a primitive, though, amounts to conceding that no explanation in terms that are already known can be given. We cannot solve the question of "holisticness," either, but we want to point to the fact that the two verb configurations correlate with a change in Aktionsart: *Sally sprayed paint onto the wall* is durative (she can do it *for two hours*), whereas *Sally sprayed the wall with paint* is transformative (she can do it *in two hours*). That observation leads us to propose that the example is best analyzed as involving a mere activity in the *with* configuration, and an additional transition in the *onto* configuration. Support for this analysis comes from Pinker (1989), who postulates a change in meaning when moving from one configuration to the other: In (b) above, Sally causes the paint to move onto the wall, whereas in (a), Sally causes the wall to change its state by means of moving the paint onto it.

Sally sprayed paint onto the wall.

```
(spray-1 (CAUSER sally-1)
         (OBJECT paint-1)
         (PATH (path-1 (DESTINATION wall-1))))
```

Sally sprayed the wall with paint.

```
(event-1 (PRE-STATE (covered-state-1 (OBJECT wall-1)
                                     (VALUE (not 'covered))))
         (ACTIVITY (spray-1 (CAUSER sally-1)
                            (OBJECT paint-1)
                            (PATH (path-1 (DESTINATION wall-1)))))
         (POST-STATE (covered-state-1 (OBJECT wall-1)
                                      (VALUE 'covered))))
```

Figure 5
SitSpecs for configurations of *to spray*.

Pinker sees (a) as derived from (b) and suggests, as a constraint on the applicability of the alternation, that the motion (here: *spray*) *causes an effect* on the surface/container. While we decided not to discuss applicability conditions here, we support the idea that the difference between (a) and (b) can be expressed with an additional change of state. In our framework, we thus assign two different SitSpecs to the sentences, one activity and one event, as shown in Figure 5.

The crucial point now is that the first SitSpec is fully embedded in the second; this is in correspondence with the truth conditions: If Sally has sprayed the wall with paint, then she also has sprayed paint onto the wall. To generalize the correspondence to an extension rule, we need to assume in the domain model a concept like **completion-state**, which is to subsume all those states in the domain model that have "extreme" values: an empty bucket, a fully loaded truck, a completely covered surface, and so forth. The exact interpretation of completion-state is the open question that Levin (1993) referred to, and that Jackendoff treated with his *d* subscript. We do think, though, that an abstract state in the domain model, which subsumes a range of the concrete states, is preferable to introducing a primitive on the linguistic level (unless the primitive is relevant for other linguistic phenomena as well).

The following alternation rule applies to durative verb readings that denote activities of something being moved to somewhere, and extends them to also cover the post-state, which must be subsumed by completion-state. In this way, it derives reading (a) from (b) in the *spray* example, and analogously for the other verbs undergoing the alternation, e.g.: *Tom loaded hay onto the wagon / Tom loaded the wagon with hay; Jill stuffed the feathers into the cushion / Jill stuffed the cushion with the feathers.* The PSemSpec is modified as follows: The former `:destination` (*wall*) becomes the new `:actee`, whereas the former `:actee` (*paint*) now fills the role `< :inclusive >`, and is optional there, because *Jill stuffed the cushion* is also well formed.

```
NAM: locative-transitive
DXT: (event
       (MOVE (OBJECT X)
             (PATH (DESTINATION Y)))
       (POST-STATE (Z completion-state (OBJECT Y))))
COV: (Z)
ROC: ((:actee < :inclusive >)
      (:destination :actee))
NRO: ()
```

Levin distinguishes two kinds of locative alternation: the spray/load alternation just discussed and the clear (transitive) alternation. The latter applies only to the verbs *clear, clean, drain, empty* and can be seen as the "semantic inverse" of the spray/load alternation, because one group of verbs denotes activities of placing something somewhere, and the other describes activities of removing something from somewhere; but both have the same holistic effect in one of the verb configurations. Thus, the rule for the clear-alternation is very similar to the one just shown. It derives, for example, *Tom drained the container of the water* from *Tom drained the water from the container.*[11]

```
NAM: clear-transitive
DXT: (event
       (MOVE (OBJECT X)
             (PATH (SOURCE Y)))
       (POST-STATE (Z completion-state (OBJECT Y))))
COV: (Z)
ROC: ((:actee < :of-matter >)
      (:source :actee))
NRO: ()
```

The *clear* verbs, except for *to clean,* can in addition be intransitive, and Levin states a separate alternation for them. For *to drain,* the first configuration is *The water drained from the tank,* and the second is either *The tank drained* or *?The tank drained of the water.* According to Levin, "the intransitive form may be best in the absence of the *of*-phrase" (Levin 1993, 55). The SitSpec denoted by the first configuration is:

The water drained from the tank.

```
(move-1 (OBJECT water-1)
        (PATH (path-1 (SOURCE tank-1))))
```

Note that our durative-causative extension rule given above applies in this case and extends the coverage of the SitSpec to one corresponding to *Tom drained the water from the tank.* A rule that is parallel to that for the transitive case is given below; it derives *?The tank drained of the water*; since the < `:of-matter` > is optional, we can also produce the preferred *The tank drained.*

```
NAM: locative/clear-intransitive
DXT: (event
       (MOVE (OBJECT X)
             (PATH (SOURCE Y)))
       (POST-STATE (Z completion-state (OBJECT Y))))
COV: (Z)
ROC: ((:actor < :of-matter >)
      (:source :actor))
NRO: ()
```

3.3 Deriving Alternations Successively

The extension rules, as we have introduced them above, constitute a framework for systematically deriving more complex verb configurations from simpler ones; the output produced by one rule serves as input to another. Figure 6 provides a synopsis: The

11 We ignore the role of the definite determiner here, which in fact has critical influence on the holistic interpretation of mass nouns. See, for example, White (1994).

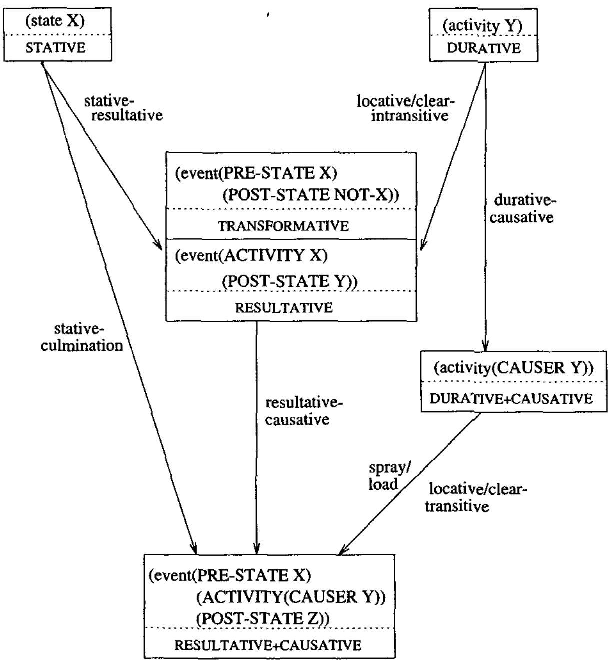

Figure 6
Dependency of extension rules.

boxes contain the denotation patterns that correspond to the Aktionsart feature, and the rules transform a configuration with one Aktionsart into another. In this graph, every verb base form has an entry point corresponding to the Aktionsart of its most basic configuration. Examples: *to fill* is stative, *to drain* is durative, *to open* is transformative, *to remove* is resultative + causative. The "double box" in the middle is the entry point for both transformative and resultative verbs, but the incoming arrows produce resultative forms. From the entry point of a verb, arcs can be followed and rules applied if the respective alternation is specified in the lexical entry. The six categories account for the Aktionsart features listed in Section 2.2, and the rules take care of possible shifts between them. Thus, the full range of SitSpecs that our ontology allows is being covered.

To illustrate the functionality, we return to the example of *to drain*. Figure 7 shows how the extension rules successively derive the various configurations. Apart from the passive, this is the complete "alternation space" of *to drain* according to Levin's (1993) catalogue. Notice that the examples given also cover the four different *drain* clauses needed to produce the alternative sentences given in (1) in the introduction.

4. Implementation: Two-Step Sentence Generation with MOOSE

The MOOSE sentence generator grew out of experiences with building the TECHDOC system (Rösner and Stede 1994), which produces instructional text in multiple languages from a common representation. Specifically, MOOSE accounts for the fact that events can receive different verbalizations even in closely related languages such as English and German. It is designed as a sentence generation module that pays attention to language-specific lexical idiosyncrasies, and that can be incorporated into

```
Denotation: (activity (OBJECT A)
                      (PATH (SOURCE B)))
PSemSpec: (x1 / nondirected-action  :lex drain
           :actor A :source B)
```

(0) *The oil drained from the engine.*

Locative / clear-intransitive of (0):

```
Denotation: (event (ACTIVITY (OBJECT A)
                             (PATH (SOURCE B)))
                   (POST-STATE (C (OBJECT B))))
PSemSpec: (x1 / nondirected-action  :lex drain
           :of-matter A :actor B)
```

(1) *The engine drained of the oil.*

Durative-causative of (0):

```
Denotation: (activity (OBJECT A)
                      (PATH (SOURCE B))
                      (CAUSER C))
PSemSpec: (x1 / directed-action  :lex drain
           :actee A :source B :actor C)
```

(2) *Tom drained the oil from the engine.*

Resultative-causative of (1):

```
Denotation: (event (ACTIVITY (OBJECT A)
                             (PATH (SOURCE B))
                             (CAUSER C))
                   (POST-STATE (C (OBJECT B))))
PSemSpec: (x1 / directed-action  :lex drain
           :of-matter A :actee B :actor C)
```

(3) *Tom drained the engine of the oil.*

Figure 7
Derivation of *drain* configurations.

a larger-scale text generator.[12] In the following, we first describe the overall system architecture, then discuss the process of lexicalization in some detail, and finally turn to the selection of a verb alternation on the basis of salience parameters.

4.1 System Architecture

Figure 8 provides an overview of the architecture of MOOSE. The generator assumes a language-neutral level of event representation, the situation specification SitSpec (see the example in Figure 2). The SitSpec instantiates concepts from a domain model, which is implemented in the KL-ONE language LOOM (MacGregor and Bates 1987). Using the denotations of the lexicon entries of the target language, the lexical options for verbalizing the SitSpec are determined. In the next step, for verbs, the applicable alternations and extensions are computed and added to the set of options. Then a language-specific semantic specification SemSpec (see the example in Figure 3) is constructed in accordance with generation parameters pertaining to brevity, salience, and stylistic features. The SemSpec is then handed over to a surface generator: Penman (Penman group 1989) for English, and a variant developed at FAW Ulm for German.

The SemSpec language is a subset of the input representation language that was developed for Penman, the sentence plan language (SPL) (Kasper 1989). An SPL expression consists of variables, types, and case roles; an example was given in Figure 3. Penman and SPL are based on the upper model (UM) (Bateman et al. 1990) introduced

12 Fröhlich and van de Riet (1997) describe how MOOSE is employed in the generation component of an information system.

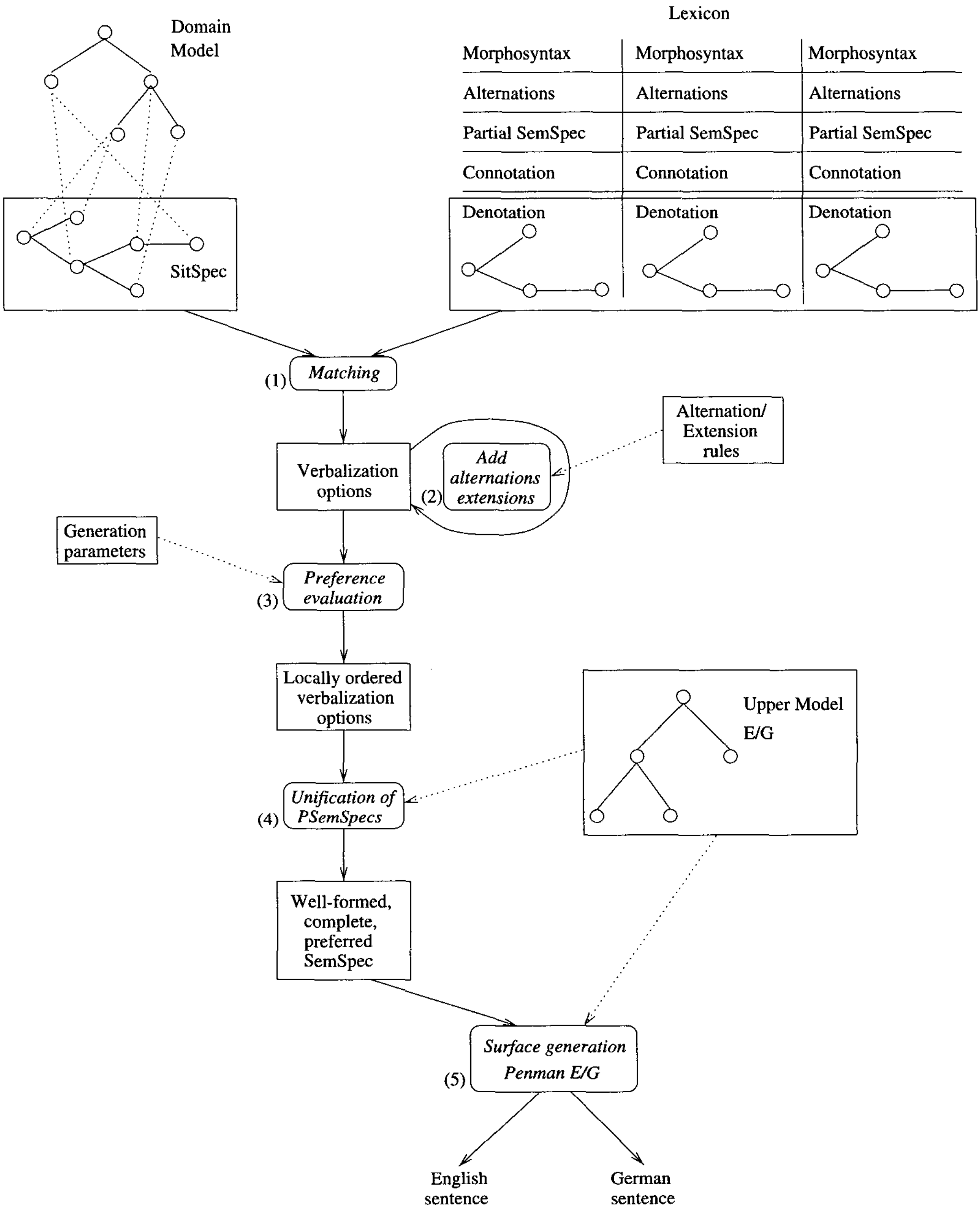

Figure 8
MOOSE system architecture.

in Section 2.3.1. For any type appearing in an SPL, Penman needs to know by which UM type it is subsumed, so that appropriate generation decisions can be made.

The way we use Penman and the UM in the MOOSE architecture is somewhat different from the original Penman conception. In Penman, the domain model was supposed to be subsumed by the UM, which indeed simplifies generation from input that uses domain model concepts. However, the range of alternative verbalizations that can be produced from the same input is seriously limited under this approach (see Stede and Grote [1995]), and therefore MOOSE opts for a complete separation of DM and UM; they are distinct taxonomies. Consequently, as opposed to a general SPL term used in Penman, a SemSpec used in MOOSE must contain only upper model concepts and no domain model concepts.

Furthermore, since our system takes lexicalization as the decisive task in mapping a SitSpec to a SemSpec, the UM concepts referred to in a SemSpec must be annotated with `:lex` expressions. Thus, a SemSpec is a lexicalized structure, and accordingly, MOOSE interprets the upper model as a taxonomy of lexical classes. This contradicts the Penman philosophy of viewing the UM as abstract semantics and clearly distinct from the generation grammar, which in accordance with systemic-functional linguistics is an integrated lexicogrammar, with "lexis as most delicate grammar" (Hasan 1987). This idea, however, has been a theoretical rather than a practical one, and lexical matters thus have not been a strong point of Penman. For instance, the distinction between obligatory and optional participants of a verb was quite blurred. Also, Penman allowed only for very simple lexical choice mechanisms, as it assumed a straightforward one-to-one mapping between concepts and words. MOOSE overcomes these problems by assigning a central role to the lexicon, placing a lot of information in it, and taking it as the crucial device for the SitSpec-SemSpec mapping. SemSpec, then, is an intermediate level of representation that reflects sentence semantics and that mediates between the language-neutral conceptual representation and linguistic realization. The simple form of our alternation rules shown in the last section, which abstract over syntactic realization, demonstrates the utility of SemSpec as a level of description.

In practice, our aim to upgrade SPL from a convenient input notation of a front-end NLG module to a systematic and well-motivated level of description involves not only building MOOSE "around" Penman, but also making some changes to the upper model and the generation grammar. But for the purposes of this paper, which focuses on the semantics of verb alternations and their role in NLG, we avoid dealing with Penman's internals and rather treat it as a "black box."

4.2 Lexicalization

In order for serious lexical choices to be possible, the first step of lexicalization in MOOSE consists of determining the set of verbalization options: all the lexemes whose denotations can potentially cover some part of the input SitSpec. Since we represent the internal structure of events, the denotation of a lexeme need not be a single concept; instead, it can be a complete configuration of concepts and roles. The consequences are a higher computational cost in finding lexical options, but also a greater flexibility in finding different verbalizations of the same event. As an example, consider the denotation of the causative reading of *to fill*:

```
(event (PRE-STATE (fill-state (VALUE (not 'full))
                               (CONTAINER A)))
       (ACTIVITY (CAUSER B))
       (POST-STATE (fill-state (VALUE < D 'full >)
                                (CONTAINER A)
                                (CONTENT C))))
```

Given some input SitSpec involving filling, the variables of the denotation are bound to instances or atomic values of the SitSpec when it is matched against the denotation. The filler of the **value** role in the post-state appears in angle brackets because it is a default value. The accompanying partial SemSpec of *to fill* contains the same variables:

```
(x / directed-action  :lex fill
  :actor B  :actee A  :inclusive C  <:destination D>)
```

When the denotation is matched against a SitSpec, the variable bindings are propagated to the partial SemSpec; and when it is later unified with the partial SemSpecs corresponding to the other elements, a complete SemSpec results, from which Penman produces a sentence like *Jill filled the tank with oil.* (If the value is different from `'full`, it also gets verbalized, such as in *Jill filled the tank to the second mark.*)

Importantly, the matching between denotations and SitSpec does not test for identity, but for subsumption—it exploits the functionality provided by LOOM. In this way, the selectional restrictions of verbs are checked when the lexical options are determined. Moreover, the matcher finds not only the most specific word, but also the applicable more general candidates, which is helpful, for instance, in achieving stylistic effects, and in avoiding undue repetitions of the same specific term.

Since we are using relatively fine-grained representations for SitSpecs and denotations, the generation of variants in incorporation is enabled by the covering mechanism in conjunction with the subsumption check. In the example *go by plane/fly,* the general verb *to go* covers only the move concept, and the role instrument-plane is left to be expressed by a prepositional phrase; whereas the specific verb *to fly* covers the whole configuration. In this fashion, quite different coverings of the input SitSpec are possible; for instance, MOOSE produces *Tom poured water into the tank until it was full* and *Tom filled the tank with water* (amongst others) as paraphrases of the same event.

After the initial matching between denotations and SitSpec, the various alternations are computed for those verbs whose base form has been found as a candidate (step 2 in Figure 8). Their lexical entry specifies which alternation/extension rules apply, and they are executed sequentially, as outlined in the previous section, and demonstrated for *to drain* in Figure 7. For any extension rule that adds new items to the denotation, the new material is matched against the SitSpec to ensure that the alternation is applicable, and to compute the additional covering. In this way, all the applicable alternated forms of a verb are added to the pool of verbalization options.

The set of all lexemes that successfully matched some part of the SitSpec, together with the alternated verb forms, constitute the search space for constructing an appropriate SemSpec. The options are first brought into an order of preference (step 3 in Figure 8) according to various parameters such as the desired salience assignment, which is explained in the next section. Considering the options in this order, a complete and well-formed SemSpec is built from the partial SemSpecs that are associated with some of the lexical options—those that collectively cover the entire SitSpec and thus will take part in the sentence. This is done by a unification process driven by the candidate verb options; recall that their PSemSpec consists of an upper model process and the mappings from situation elements to process participants, which is achieved by co-indexing with positions in the denotation. By means of sharing this information between denotation and PSemSpec, the lexicon entries serve as a "bridge" between the SitSpec to be verbalized and the intermediate representation SemSpec.

For more details on the kinds of mono- and multilingual variation produced by MOOSE, and on the lexicalization algorithm, see Stede (1996b).

4.3 Producing Salience Variation with Alternations

Having explained the basic machinery of MOOSE, we now demonstrate how the generator can make an informed choice among the set of possible verb alternations on the basis of a salience parameter. Since a full-fledged treatment of the role of salience is well beyond the scope of this paper,[13] we will merely sketch a possible division of labor between text planning and sentence planning, and then describe the role of verb alternations as one means of realizing generation goals related to salience. Specifically, we show under what circumstances the various *drain* sentences given in the examples in (1) in Section 1 (and in Figure 7) are produced by MOOSE.

When text planning has been completed and linearization as well as the "chunking" of the material into sentence-size pieces is accomplished, MOOSE takes over to perform the necessary sentence planning, which includes lexicalization. Thus, text planning has produced a sequence of SitSpecs, which may be enriched with information pertaining to the relative salience of the elements. This information can result from constraints on theme development, from rhetorical strategies, or from other considerations at the discourse level. For instance, in the sample texts given in (1) at the beginning of Section 1, different theme developments are responsible for the different usages of *to drain*.

For sentence planning, we assume that individual nodes of a SitSpec can have a foreground, background, or optional label attached to them (but they need not). Then, a realization is to be found that signals the differences in prominence on the linguistic surface. In general, there is no one-to-one correspondence between the configuration of salience labels and linguistic realization, though. Instead, we view salience goals as goals that the generator tries to fulfill if possible, similar to certain stylistic goals (see Stede [1996a]). Thus, generation becomes a matter of constraints (say the right thing) and preferences (try to say it in a particular way), similar to Hovy's (1988) distinction between "prescriptive" and "restrictive" planning.

What, then, is the role of verb alternations in assigning different degrees of salience? Talmy (1988) listed a number of morphological and syntactic means to distribute salience across the elements of a clause. For instance, he suggested the hierarchy subject > direct object > indirect object > oblique, ranging from the most salient to the least salient. From a slightly different perspective, Kunze (1991) was concerned with differences in salience between similar verbs. He advanced the view that they share a common underlying base form and differ, inter alia, in distributing salience via their case roles. For our purposes here, we can adapt these insights (with some simplification) and state that an element is placed in the foreground if it is mapped to the role actor (best) or to actee (second best). Correspondingly, it is placed in the background if it corresponds to a circumstance, i.e., a role that is not part of the verb's case frame.

Now, consider again the sentences in Figure 7. On the one hand, (0) and (1) omit the fact of Tom's causing the event, and hence are preferred only if the respective SitSpec node is labeled as optional. On the other hand, (0, 2) and (1, 3) differ in that the former render the oil prominent, while the latter emphasize the engine. Figure 9 shows the common SitSpec underlying the four sentences, and a set of salience labels attached to three nodes, where the numbers correspond to the target sentences. For example, when sentence (1) is the preferred output, the SitSpec would have an opt label at node tom-1 and an fg label at node engine-1.

For any verbalization option, base forms and alternations alike, the number of fulfilled salience goals can be computed straightforwardly: Since variables in denotations and PSemSpecs are co-indexed, we can determine for every salience label in the SitSpec how the corresponding element participates in the SemSpec. Using the criteria given above, preference values result for the various options, and they are factored into the overall preference ranking of the verbalization options. All other things being

13 Pattabhiraman (1992) devoted a dissertation to the topic of salience in NLG.

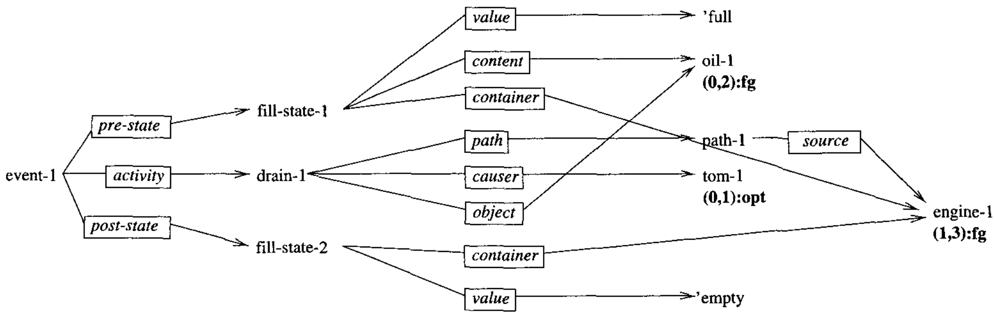

Figure 9
SitSpec representing a *drain* event.

equal, the verb alternation that accomplishes the best realization of the salience labels outranks the other options and thus gets selected for building the SemSpec. Again, notice that other syntactic and morphological means (e.g., expressing an element with a separate word versus incorporating it into another word) for assigning salience can be integrated into this scheme.

5. Summary and Related Work

We have proposed an NLG framework, together with suitable representation schemes, that can systematically produce a range of verb alternations from a common underlying input representation and select the most appropriate form on the basis of salience parameters. Productive rules derive the more complex forms from a basic one, which is the only one that needs to be stated in the lexical entry of the verb. We have focused on those alternations that affect the Aktionsart of the verb: They imply a type change in aspectual classifications such as those of Bach (1986).

For generation, our approach uses two distinct ontologies: a language-neutral domain model for event categorization, and a language-specific taxonomy, the upper model developed by Bateman et al. (1991) on the basis of Halliday's (1985) work. The lexicon acts as the mediator between these two realms and serves to map a conceptual input representation to a semantic sentence specification, which can be further processed by a front-end realization component. Within this framework, multilingual generation is possible once language-specific upper models and front-ends are used (but multilinguality was not addressed in this paper). The approach has been implemented in the MOOSE system, which uses the Penman generator (Penman Group 1989); MOOSE can serve as a plug-in sentence production module to a larger text generator.

The examples discussed in this paper (including the alternated forms) were generated from a domain model that encodes knowledge about automobile engines, tanks, and related liquids. It consists of 150 concepts and relations; the associated English lexicon has 200 words, including 50 verbs. Given the nature of the work, which depends on quite fine-grained representations in both domain model and lexicon, it is difficult to make statements on how well the approach "scales up." Large-scale (automatic) acquisition of dictionary entries typically does not result in representations of the kind needed here, and furthermore, the domain model needs to be developed in tandem with the lexicon. The precise shape of such models, on the other hand, also depends on the specific application the generator is used for; even though some steps towards standardization in ontologies are being taken, this is still a bottleneck in knowledge-based NLG.

In the following, we compare our approach to some related work on verb alternations and on lexicalization in NLG. Finally, we draw some conclusions as to the overall scope of the work and its utility for NLG.

5.1 Alternations

Starting from the aspectual categories proposed by Bach (1986), the verb classifications of Levin (1993), and the lexical representations given by Jackendoff (1990), we have developed a new synthesized approach for dealing with verb alternations that affect the Aktionsart of verbs. Our ontology for input representations and the specifications for lexical meaning have benefited from the earlier work just mentioned but essentially constitute a new framework in which the specific alternation/extension rules could be formulated.

The utility of these rules demonstrates the importance of defining a place for fine-grained lexical-semantic representations in language generation. To our knowledge, no other generator can systematically derive the various forms for the alternations discussed in this paper. Recently, Dorr and Olsen (1996) suggested using verb representations based on Jackendoff's (1990) LCSs for NLG; specific kinds of LCSs are proposed to represent different classes of verbs on the basis of telicity. Rules are proposed that relate telic and atelic versions of the same verb. The central difference of our approach is our distinction between SitSpec and SemSpec, and thus between denotation and PSemSpec in the lexical entries. Dorr and Olsen map directly between LCSs and syntax, so there is no systematic link to background knowledge yet (which, as we have pointed out, would be useful for generation). Besides, as we mentioned in Section 3, LCS representations use primitives (BECOME, INCH, $_d$), where we opt for a more fine-grained decomposition of the underlying event.

For the alternations investigated, we have chosen the approach of defining a single base form from which alternated forms are derived. For other alternations, this might not be feasible or practical—in such cases, different lexical entries are to be used. There is, on the other hand, a line of research that questions the utility of distinguishing a base form from a more complex one in an alternation. For example, Saint-Dizier (1996) states that his approach to alternations deliberately *avoids* three difficulties: the need to define a basic form from which alternations are produced; the need to explain the relation between the basic form and the alternated one; and the need to account for changes in meaning produced by the alternation. It seems that the work presented in this paper aims precisely at those questions that Saint-Dizier's approach proposes to better leave aside. For generation, however, we believe that a system must know about the fine-grained changes in meaning that a verb alternation implies—a generator has to relate some semantic input representation to verb meaning, after all, and that includes alternations. And if the semantic change induced by an alternation can be described by a general rule that covers a whole class of verbs, a useful abstraction is gained.

The final point to consider is the question of admitting lexical rules into one's framework. For example, Sanfilippo (1994) argues against this instrument on the grounds that there is no general control regime on lexical rules that would deterministically restrict any polysemic expansion. Instead, he advocates coding the alternative lexical forms in a hierarchy of typed feature structures, where the underspecified forms subsume the specific ones. His criticism applies to the notion of rules that are triggered automatically and proceed to derive new forms without principled limitations. Our "defensive" approach of listing applicable rules in the lexical entries avoids this problem but at the same time raises the question of why rules should be preferable to a simple enumeration of forms. We return to this point in section 5.3.

5.2 Lexicalization in NLG

In MOOSE, the lexicon is the central device for mapping between input representations and intermediate sentence-semantic representations. The idea of using the lexicon early in the generation process is not new; it has been realized in several other generators, for example in the frame-oriented system DIOGENES (Nirenburg and Nirenburg 1988). In contrast to earlier systems, however, MOOSE strengthens the role of lexical semantics in the generation process by distinguishing between the SitSpec and SemSpec levels and clearly specifying the relationships between the two (as done with the alternation rules). Furthermore, we have emphasized that lexical choice should be seen as a constraint satisfaction process, similar to Reiter (1991), who focused his attention on nouns, while we have concentrated on verbs.

There are several other generators using Penman as a front-end. For example, the DRAFTER system (Paris et al. 1995) builds SPLs and hands them over to Penman; contrary to MOOSE, however, the domain model in DRAFTER is subsumed by the upper model, which significantly limits the range of lexical variation, as pointed out above.

Working in the framework of systemic-functional grammar (SFG), both Wanner (1992) and Teich and Bateman (1994) employ SPL as an intermediate description, but they emphasize the integration of the SPL construction process into SFG. Wanner uses system networks to make fine-grained lexical choices in line with the three systemic metafunctions. Teich and Bateman develop system networks describing genre and register variation to drive the generation process, and they query an external domain model when building the SPL. In related work, Teich, Firzlaff, and Bateman (1994) present an implementation of Kunze's theory of semantic emphasis (cf. Section 4.3). From a "basic semantic scheme" annotated with emphasis labels, an SPL with appropriate roles and upper model concepts is constructed. The SPL can also contain an emphatic/nonemphatic feature, which might lead, for instance, to a dative shift. Hence, this work shares our interest in salience and indeed goes a step further than our present account in that the generation grammar can employ additional means for salience variation. However, in these three approaches, all lexical matters are taken to be part of the (huge) grammar processing the SPL. Thus, the central difference to the MOOSE approach is our step of promoting the lexicon to the crucial device for mapping between conceptual and sentence-semantic representations. We have argued that this step of keeping the lexicon separate and accessing it early has a number of advantages.

Essentially the same difference holds between MOOSE and GOSSiP (Iordanskaja, Kittredge, and Polguère 1991), which also emphasizes the importance of lexical choice and paraphrasing abilities. Here, a powerful lexicalization mechanism is embedded in a meaning–text generation model following the theory of Mel'cuk, where lexical functions play a central role in mapping between different levels of representation. These are semantic and syntactic levels, though, whereas MOOSE focuses on the interface between conceptual and semantic representations, and employs the lexicon at that point.

Representations more similar to ours have been used by Dorr and Voss (1996), who employ Jackendoff's (1990) LCSs as an interlingua in machine translation, and by Di Eugenio (1993), who also represents LCS in a KL–ONE language but for purposes of analysis rather than generation. More specifically for NLG, structure mappings between fine-grained representations have been suggested for instance by Horacek (1990), Nogier and Zock (1992), and Nicolov, Mellish, and Ritchie (1996). In all these approaches, the input structure is directly mapped to a syntactic structure, though, while we have argued that an intermediate sentence-semantic level is advantageous in order to explore generalizations (such as the alternation rules) as well as for multilingual purposes.

5.3 Conclusions

In a computational approach to the lexicon, word sense enumeration should not be the rule but be reserved for the exceptions (Pustejovsky 1995). In line with this view, our approach seeks to exploit generalizations by accounting for different forms of a verb with explicit alternation and extension rules that relate the changes in meaning to the changes in form. Ultimately, such an account establishes correspondences not only between different forms of the same verb but also between different verbs; for example, applying the causative extension to *to rise* yields (one form of) *to raise*. Interconnections of this kind have not yet been integrated into the system presented here, though.

Three assumptions have guided the development of our account of verb alternations: (1) There is a single base form from which other forms can be derived. (2) Alternation rules leave the denotation unchanged, and extension rules always add facets of meaning to the simpler denotation. (3) The changes in denotation correspond to changes in form, which can be characterized on a case-role level of description. In dealing with the telicity-related alternations discussed in Section 3, these assumptions have proven useful. For generalizing the approach to other alternations, assumption (2) could turn out to be too strong; in fact, even the causative extension might not always be monotonic when temporal adverbials are part of the sentence. In our framework, monotonicity is not a problem as long as the order of rule application is fixed anyway (cf. Figure 6). As soon as nonmonotonic rules are allowed, and the applicability of rules is no longer defined in the lexicon entries but triggered directly by the input, circularity is to be avoided: It needs to be ensured that rules *reducing* meaning reduce only parts that are not added by a different rule.

Our selection of alternations was guided by their relationship to Aktionsart, in particular to causation and telicity. Since the notion of Aktionsart is not a well-demarcated one in linguistics, and since the most comprehensive catalogue of alternations, the one by Levin (1993), has largely excluded Aktionsart-related problems, it is rather difficult to evaluate our approach in terms of "how many alternations" it covers. (Besides, we have argued in Section 3 that some of Levin's categorizations need refinement.) Clearly, there are other alternations involving telicity that we have not discussed here. Dorr and Olsen (1996) state that 27 of Levin's alternations add the telicity feature to a verb's meaning; many of these are rather specific and apply only to very few verbs. Among the more prominent ones are the unspecified object alternation (*Tom ate/Tom ate a pizza*) and the conative alternation (*John cut at the bread/John cut the bread*). Both lend themselves to extension rules as in our framework, because one form entails the other and adds information: When it holds that Tom ate a pizza, then it holds that Tom ate. Other alternations involve specific prepositions, such as Levin's *through/with* alternation: *Alison pierced the needle through the cloth/Alison pierced the cloth with a needle.* This does not pose problems for representing the changes in denotation, but renders a reliance on case roles—assumption (3) above—questionable; if suitable generalizations to similar prepositions cannot be found, the change in form ought to be stated directly on the syntactic level.

Finally, we look at the question of evaluating our approach from the perspective of natural language generation. From a descriptive viewpoint, as argued above, general lexical rules are to be preferred over enumerating word senses. Whether this preference also carries over to the design of practical NLG systems, however, merits some additional discussion. For the lexicalization step, we can either successively apply alternation rules to a successfully matched base form, or compile out the various alternated forms, which must then all be considered in matching against the input representation. While the first option obviously yields a much smaller lexicon, it is not self-evident whether it is faster or slower in a running system.

As long as all alternated forms individually enter the matching phase, the compile-out option is hardly useful. Rather, compilation can be advantageous if only the most preferred form of the verb is considered first, and the other ones only upon request if the first did not work out. In this case, we are spared the effort of applying the rules to reach the desired form at run-time. Overall, the compilation decision hinges on the kind of criteria that the generator employs for its lexical choices. If the desired salience distribution is the central factor, then storing precompiled options and their salience information will be most effective. If considerations of lexical style lead to preferring one verb over a set of others irrespective of the specific alternation, then applying alternation rules only to the preferred verb will be more effective (in turn depending on how many similar verbs are ruled out and thus spared from the matching process). Thus, there appears to be no general answer; the size of the lexicon, including the ranges of nearly synonymous verbs, and the choice criteria used by the generator have to be taken into account.

Acknowledgments
The research reported in this paper originated at the University of Toronto (Canada) and at the Research Center for Applied Knowledge Processing (FAW) in Ulm (Germany). In the respective places, thanks to Graeme Hirst and Dietmar Rösner for discussions and advice, and to the Natural Sciences and Engineering Research Council of Canada (NSERC) and to FAW for financial support. I am grateful to the anonymous reviewers for their valuable suggestions for improving an earlier version of this paper.

References

Bach, Emmon. 1986. The algebra of events. *Linguistics and Philosophy* 9.

Bateman, John, Robert Kasper, Johanna Moore, and Richard Whitney. 1990. A general organization of knowledge for natural language processing: The penman upper model. Technical Report, USC/ISI, Marina del Rey, CA.

Brachman, Ronald and James G. Schmolze. 1985. An overview of the KL-ONE knowledge representation system. *Cognitive Science* 9(2).

Bussmann, Hadumod. 1983. Lexikon der Sprachwissenschaft. Kröner, Stuttgart.

Di Eugenio, Barbara. 1993. *Understanding Natural Language Instructions: A Computational Approach to Purpose Clauses.* Ph.D. thesis, Department of Computer and Information Science, University of Pennsylvania. Technical Report MS-CIS-93-91.

Dorr, Bonnie and Mari Broman Olsen. 1996. Multilingual generation: The role of telicity in lexical choice and syntactic realization. *Machine Translation* (11):37–47.

Dorr, Bonnie and Claire Voss. 1996. A multi-level approach to interlingual MT: Defining the interface between representational languages. *International Journal of Expert Systems* 9(1).

Fröhlich, Marcel and R. van de Riet. 1997. Conceptual models as knowledge resources to text generation. In *Proceedings of the Third International Workshop on the Application of Natural Language to Information Systems*, Vancouver.

Halliday, Michael. 1985. *Introduction to Functional Grammar*. Edward Arnold, London.

Hasan, Ruqaiya. 1987. The grammarian's dream: Lexis as most delicate grammar. In Michael Halliday and Robin Fawcett, editors, *New Developments in Systemic Linguistics*. Volume 1. Pinter, London.

Helbig, Gerhard and Wolfgang Schenkel. 1973. Wörterbuch zur Valenz und Distribution deutscher Verben. VEB Verlag Enzyklopädie, Leipzig.

Horacek, Helmut. 1990. The architecture of a generation component in a complete natural language dialogue system. In Robert Dale, Chris Mellish, and Michael Zock, editors, *Current Research in Natural Language Generation*. Academic Press, London.

Hovy, Eduard H. 1988. *Generating Natural Language Under Pragmatic Constraints*. Lawrence Erlbaum, Hillsdale, New Jersey.

Iordanskaja, Lidia, Richard Kittredge, and Alain Polguère. 1991. Lexical selection and paraphrase in a meaning-text generation model. In Cécile Paris, William Swartout, and William Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*. Kluwer, Dordrecht.

Jackendoff, Ray. 1990. *Semantic Structures*. MIT Press, Cambridge, MA.

Kasper, Robert. 1989. A flexible interface for linking applications to Penman's sentence generator. In *Proceedings of the DARPA Workshop on Speech and Natural Language Processing*, University of Pennsylvania.

Kunze, Jürgen. 1991. Kasusrelationen und semantische Emphase (studia grammatica XXXI). Akademie Verlag, Berlin.

Levin, Beth. 1993. *English Verb Classes and Alternations*. University of Chicago Press, Chicago.

Levin, Beth and Malka Rappaport Hovav. 1995. *Unaccusativity*. MIT Press, Cambridge, MA.

MacGregor, Robert and Robert Bates. 1987. The Loom Knowledge Representation Language. Technical Report ISI/RS-87-188, USC/ISI, Marina del Rey, CA.

Matthiessen, Christian and John Bateman. 1991. *Text Generation and Systemic Functional Linguistics: Experiences from English and Japanese*. Pinter, London.

Moens, Marc and Mark Steedman. 1988. Temporal ontology and temporal reference. *Computational Linguistics* 14(2).

Nicolov, Nicolas, Chris Mellish, and Graeme Ritchie. 1996. Approximate generation from non-hierarchical representations. In *Proceedings of the Eighth International Workshop on Natural Language Generation*, Herstmonceux Castle, England.

Nirenburg, Sergei and Irene Nirenburg. 1988. A framework for lexical selection in natural language generation. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING-88)*, Budapest.

Nogier, Jean-Francois and Michael Zock. 1992. Lexical choice by pattern matching. *Knowledge Based Systems* 5(3).

Paris, Cécile, Keith Vander Linden, Markus Fischer, Anthony Hartley, Lyn Pemberton, Richard Power, and Donia Scott. 1995. A support tool for writing multilingual instructions. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-95)*, Montreal.

Parsons, Terence. 1990. *Events in the Semantics of English: A Study in Subatomic Semantics*. MIT Press, Cambridge, MA.

Pattabhiraman, T. 1992. *Aspects of Salience in Natural Language Generation*. Ph.D. thesis, School of Computing Science, Simon Fraser University.

The Penman group. 1989. Unpublished Documentation of the Penman Sentence Generation System. USC/ISI, Marina del Rey, CA.

Pinker, Steve. 1989. *Learnability and Cognition: The Acquisition of Argument Structure*. MIT Press, Cambridge, MA.

Pustejovsky, James. 1991. The syntax of event structure. *Cognition* 41:47–81.

Pustejovsky, James. 1995. *The Generative Lexicon*. MIT Press, Cambridge, MA.

Reiter, Ehud. 1991. A new model of lexical choice for nouns. *Computational Intelligence* 7:240–251.

Rösner, Dietmar and Manfred Stede. 1994. Generating multilingual documents from a knowledge base: The TECHDOC project. In *Proceedings of the International Conference on Computational Linguistics (COLING-94)*, Kyoto.

Saint-Dizier, Patrick. 1996. Verb semantic classes based on 'alternations' and on WordNet-like semantic criteria: A powerful convergence. In *Proceedings of the Workshop on Predicative Forms in Lexical Semantics and Lexical Knowledge Bases*, Toulouse.

Sanfilippo, Antonio. 1994. Word knowledge acquisition, lexicon construction, and dictionary compilation. In *Proceedings of the International Conference on Computational Linguistics (COLING-94)*, Kyoto.

Somers, Harold. 1987. *Valency and Case in Computational Linguistics*. Edinburgh University Press, Edinburgh.

Stede, Manfred. 1996a. *Lexical Semantics and Knowledge Representation in Multilingual Sentence Generation*. Ph.D. thesis, Dept. of Computer Science, University of Toronto. Technical Report CSRI-347.

Stede, Manfred. 1996b. Lexical paraphrases in multilingual sentence generation. *Machine Translation* 11:75–107.

Stede, Manfred and Brigitte Grote. 1995. The lexicon: Bridge between language-neutral and language-specific representations. In *Working notes of the IJCAI Workshop on Multilingual Text Generation*, Montreal.

Talmy, Leonard. 1988. The relation of grammar to cognition. In B. Rudzka-Ostyn, editor, *Topics in Cognitive Linguistics*. John Benjamins, Amsterdam.

Teich, Elke and John Bateman. 1994. Towards the application of text generation in an integrated publication system. In *Proceedings of the Seventh International Workshop on Natural Language Generation*, Kennebunkport, ME.

Teich, Elke, Beate Firzlaff, and John Bateman. 1994. Emphatic generation:

Employing the theory of semantic emphasis for text generation. Paper presented at the International Conference on Computational Linguistics (COLING-94), Kyoto.

Tesnière, L. 1959. *Éléments de syntaxe structurale*. Klincksieck, Paris.

Vendler, Zeno. 1967. *Linguistics and Philosophy*. Cornell University Press, Ithaca, New York.

Wanner, Leo. 1992. Lexical choice and the organization of lexical resources in text generation. In *Proceedings of the European Conference on Artificial Intelligence (ECAI92)*, Vienna.

White, Michael. 1994. *A Computational Approach to Aspectual Composition*. Ph.D. thesis, Dept. of Computer and Information Science, University of Pennsylvania.