Beginning novels and finishing hamburgers - remarks on the semantics of *to begin*[*]

Markus Egg

Abstract

Verbs like *begin* may take either a VP or an NP complement, but their meaning is pretty similar in both cases, e.g., for *begin*, the start of an eventuality is at stake. Pustejovsky's approach captures this similarity in terms of an invariant meaning of the verb, which entails a process of reinterpretation for the transitive variant of the verb. I will show that while the intuitions of this proposal are on the right track, its actual implementation suffers from a number of shortcomings. I will offer an analysis that preserves Pustejovsky's intuition but avoids these shortcomings. My analysis is based on an appropriate underspecification formalism.

1 Introduction

A number of verbs, most notably *begin*, may occur in two syntactic variants, either as a subject control verb with an NP and a VP argument (1)/(3), or as a transitive verb with two NP arguments (2)/(4). These verbs comprise *begin*, *finish*, *enjoy*, *fail*, and others.[1]

(1) Max began to read a novel

(2) # Max began a novel

(3) Nigel finished eating a hamburger

(4) # Nigel finished a hamburger

The verbs subcategorise for different kinds of VPs: the VP must be headed by an infinitive for *fail*, for *enjoy* and *finish*, its head must be a gerund, and for *begin*, it may be either.

[*]I thank the members of the project CHORUS of the SFB 378 in Saarbrücken and two anonymous reviewers for valuable comments and discussion.

[1]Sentences that are acceptable only after reinterpretation (but not in a literal interpretation) are marked by a 'hash' (#) sign.

However, the interpretation of the verbs is pretty much the same for both syntactic variants. E.g., for both (1) and (2) *begin* introduces a relation between an individual and a property, which yields the interpretation that either sentence refers to the beginning of an activity in which Max affects a novel. In the same way, (3) and (4) are similar in that they both express the end of an activity whose participants are Nigel and a hamburger.

In the following, I will concentrate on the case of *begin*. Formally, the relation $\mathbf{begin}'$ as introduced by this verb can be defined as follows:

(5) $\mathbf{begin}'(x, P)(e)$ iff e is the initial part of an eventuality[2] in the extension of $\lambda e'.P(x)(e')$

This definition of the begin-relation neglects the so-called *imperfective paradox*, which is familiar from semantic accounts of the progressive (see e.g. Dowty 1979): what qualifies an eventuality as a part of another eventuality in the extension of P in case there is no complete P-eventuality?

Pustejovsky (1991, 1995) proposes capturing the semantic similarity between both syntactic variants of the verb in terms of an invariant semantics for them. However, this begs the question of how the meaning of sentences with a transitive *begin* is composed: the verb semantically selects for a *property* (a relation between individuals and eventualities; its type is $\langle e, \langle s, t\rangle\rangle$, where s is the type for eventualities) whereas the object NP provides a functor from such properties to *propositions* (sets of eventualities; their type is $\langle s, t\rangle$) in its semantics.

Thus, there is an impending type conflict between these two constituents, which must be avoided in order to keep the semantic representation well-formed. To this end, Pustejovsky assumes a process of *type coercion* for every instance of a transitive *begin* which maps the semantic contribution of the object NP onto a property. Formally, a coercion operator is inserted between a mismatching functor-argument pair, it applies to the argument first, and then the functor is applied to the result of this first application. The schema for type coercion is (6), where F, A, and Op stand for the functor, the argument, and the operator, respectively:

(6) $F(Op(A))$

While I do believe that the intuitions behind Pustejovsky's approach are on the right track, a closer look at his analyses reveals that there are a number of shortcomings in his actual implementation of his intuitions. I will offer an analysis that preserves Pustejovsky's intuitions but avoids these shortcomings. In particular, I will also assume that both syntactic variants of *begin* (control verb and transitive verb) have the same meaning. However, due to their syntactic difference, this meaning may enter the process of semantic construction in different ways: The two syntactic variants are handled by different rules of the syntax-semantics interface.

[2]This term refers to states of affairs of all kinds. It is thus a hyperonym to terms like 'state', 'process', 'event', and the like.

My analysis is based on an appropriate underspecification formalism. Just like any type coercion analysis of sentences like (2), the proposed analysis must be able to give a satisfactory account of the following questions: what does semantic construction for cases of type coercion look like, where does the operator *Op* come from, and how is it integrated with the functor-argument structure.

The term *type coercion* is somewhat unfortunate, because it suggests that such processes can merely adjust the type of an expression, similar to the processes proposed in Partee (1986). However, type coercion processes characteristically involve a *sortal* shift. Such sortal shifts are a much deeper change of an expression than a mere change of its type: They change not only the combinatory potential of an expression but also the way in which it relates to the common-sense ontology that underlies the semantic sorts. (See section 3.3 for details.) However, to keep the terminological confusion to a minimum, I will stick to the established term *type coercion*, but ask the reader to beware of its unwanted implications.

The paper is structured as follows. I present Pustejovsky's analysis and outline the problems for this analysis in section in section 2. On the basis of the results of my discussion of Pustejovsky's approach, I offer an account of the semantics of *begin* in section 4 that preserves his intuitions but avoids the problems noted in section 2. This account employs an underspecification formalism, which will be briefly introduced before in section 3.

2 Pustejovsky's analysis

Pustejovsky's analysis is situated in the global theory of the *Generative Lexicon* (GL). This theory regards the lexicon not as a stative part of linguistic competence, but, rather, as a component of linguistic competence that allows one to generate meanings in a flexible yet compositional way. The basis of this theory are richly structured semantic lexical entries on the one side and rules of semantic composition on the other side that go beyond the standard repertoire of semantic construction in that they can access and combine this rich structure of the semantic lexical entries in various ways. First I will outline the details of this analysis, then I will show that it brings along with it several problems.

2.1 Overview

In the GL theory, semantic lexical entries – especially those for nouns and verbs – are described in the form of 'structured meanings'. That is, the semantic representations provide additional information on a number of essential properties of their main arguments in the form of the so-called *Qualia Structure* (QS). E.g., the semantics of *novel* comprises in its QS information about the coming about of novels (by writing them, the *agentive* quale) and their purpose (reading them, the *telic* quale). Formally, these two qualia are modelled as the property of reading or writing x, where x is the main argument in the semantics of *novel*. Thus, (part of) the semantics of *novel* could be encoded in an attribute-value matrix like (7). This lexical entry illustrates the part of the QS that is relevant for the interpretation of *begin*-sentences, viz., the *telic* and the *agentive* qualia:[3]

(7)
$$\begin{bmatrix} \text{SEM} & \begin{bmatrix} \textit{sem} & \\ \text{LTERM} & \lambda x.\mathbf{novel'}(x) \end{bmatrix} \\ \text{QUALIA} & \begin{bmatrix} \textit{qualia} & \\ \text{TELIC} & \lambda y \lambda e.\mathbf{read}_*(y,x)(e) \\ \text{AGENTIVE} & \lambda y \lambda e.\mathbf{write}_*(y,x)(e) \end{bmatrix} \end{bmatrix}$$

This rich structure of semantic lexical entries is complemented by an augmented repertoire of methods of semantic construction. These methods may refer to parts of structured meanings in case there is no direct match between the semantic contributions of constituents. In particular, impending type-theoretic or sortal conflicts between constituents of an expression can be avoided by rules of semantic construction that involve type coercion of a mismatching argument in terms of the QS. In this way, derivations of meanings of expressions that are usually explained in terms of processes outside the confines of semantics proper are explicitly integrated and licensed in the semantic construction.

The lexicon comprises functions that can access QS information and make it available for semantic construction. These functions can be used as type coercion operators that map the semantics of an NP onto an expression related to certain qualia of its head noun, here, to the telic or the agentive quale. In this way, it is possible to obtain information on novel-related activities that is relevant for the meaning of sentences like (2). One of these functions pertains to the telic quale and is called Q_T. E.g., for *a novel* Q_T is $\lambda \mathscr{P}$ $\lambda x \lambda e.\mathbf{read'}(x, \mathscr{P})(e)$, which relates entities x and relations $\mathscr{P}$ between properties and propositions in terms of the relation $\mathbf{read'}$.

By the use of these operators one can avoid the impending type conflict between the verb and its object NP in a sentence like (2). Their application to an NP semantics turns the NP semantics into a property, which qualifies as possible second argument of a functor like the begin-relation. For (2), this yields two well-typed semantic representations that correspond to two options for the interpretation of (2), viz., that Max began to read or to write a novel.

The derivation of the 'telic' reading of this sentence (minus tense and sentence mood) in Pustejovsky's (1991, 430) framework is given (slightly adapted) in (8). Here 'Tr' is a recursive function that maps syntactic constituents on their semantic contribution:

(8) Tr(*Max began a novel*) =

Tr(*Max*) [Tr(*began a novel*)] =

[3]There are two more qualia in the QS, viz., the *constitutive* and the *formal* quale. E.g., for *novel*, the constitutive quale specifies the relation between the novel and its constitutive parts and the *formal* quale models the distinctive characteristics of novels within a larger domain. But these qualia are not involved in Pustejovsky's account of type coercion for *begin*-sentences, hence, are mostly neglected in the following.

$$\lambda P \lambda e''.P(\mathbf{max'})(e'')\ [\mathrm{Tr}(\mathit{began})[Q_T(\mathrm{Tr}(\mathit{a\ novel}))]] =$$

$$\lambda P \lambda e''.P(\mathbf{max'})(e'')\ [\mathrm{Tr}(\mathit{began})[\lambda x \lambda e.\mathbf{read'}(x, \lambda P \lambda e' \exists y.\mathbf{novel'}(y) \wedge P(y)(e'))(e)]] =$$

$$\lambda P \lambda e''.P(\mathbf{max'})(e'')\ [\mathrm{Tr}(\mathit{began})[\lambda x \lambda e \exists y.\mathbf{novel'}(y) \wedge \mathbf{read}_*(x,y)(e)]] =$$

$$\lambda P \lambda e''.P(\mathbf{max'})(e'')[\lambda P \lambda x \lambda e'.\mathbf{begin'}(x,P)(e')[\lambda x \lambda e' \exists y.\mathbf{novel'}(y) \wedge \mathbf{read}_*(x,y)(e')]] =$$

$$\lambda P \lambda e''.P(\mathbf{max'})(e'')[\lambda x \lambda e'.\mathbf{begin'}(x, \lambda x \lambda e' \exists y.\mathbf{novel'}(y) \wedge \mathbf{read}_*(x,y)(e'))(e')] =$$

$$\lambda e''.\mathbf{begin'}(\mathbf{max'}, \lambda x \lambda e' \exists y.\mathbf{novel'}(y) \wedge \mathbf{read}_*(x,y)(e'))(e'')$$

In prose: the meaning of *Max began a novel* in its 'telic' reading emerges as the set of eventualities e'' that are the beginning of another eventuality e' such that e' is the reading of a novel by Max. Here NPs have meanings of type $\langle\langle e, \langle s,t \rangle\rangle, \langle s,t \rangle\rangle$. In this derivation, I use an extensionalization meaning postulate:

(9) $\forall x \forall \mathscr{P} \forall e.\mathbf{read'}(x, \mathscr{P})(e) \leftrightarrow \mathscr{P}(\lambda y \lambda e'.\mathbf{read}_*(x,y)(e'))(e)$

However, there are a number of problems for this analysis, which I am going to discuss in the following section.

2.2 Problems for Pustejovsky's analysis

Pustejovsky's analysis begs the question of where to draw the boundary between linguistic knowledge and conceptual knowledge (specific as well as generic). I disagree with this analysis here in that I would not include into the lexicon background knowledge on entities like information about their coming about or their purpose. Such information seems to me a piece of generic conceptual knowledge that should hence not be modelled as part of the linguistic knowledge.

What is more, accounting for reinterpretation candidates (this term refers to expressions that have to undergo reinterpretation) like (2) in terms of a small set of lexicalised pieces of additional information about specific entities is problematic in that this set is not large enough to cover the rich possibilities of reinterpretation phenomena, in particular, in cases where reinterpretation is guided by context-specific conceptual knowledge.

This problem shows up e.g. in a context in which Max is a professor of literary studies. Here the reading 'Max began to lecture on a novel' is much more preferable for (2) than its 'author' or its 'reader' reading. But Pustejovsky's approach as it stands would ignore this extralinguistic information and thus fail to derive this reading. This follows from the fact that his approach privileges the 'author' and 'reader' readings of sentences like (2) over other potential interpretations in that only these readings can be derived compositionally. Even though these readings are among the preferred ones in an empty context, appropriate context-specific conceptual knowledge may well induce a preference for other readings.

Lascarides and Copestake (1998) show that this undergeneration problem can only be overcome by encoding the information in the Qualia Structure as *default knowledge*. Like

Pustejovsky they assume that verbs like *begin* (in their transitive variant) determine the property that is their innermost semantic argument with reference to information in the QS of their object NPs. Due to the default nature of this information, however, this does not rashly delimit the range of possible readings to the ones provided by the QS, but captures the observation that other readings may arise in specific contexts, e.g., in the case of (10):

(10) # The professor of literary studies began the novel

The concrete spellout of Pustejovsky's analysis exhibits a number of problems as well. The first open question about (8) concerns the motivation of the insertion of Q_T in the third step of the derivation. At this early stage, we are still busy breaking down the syntactic structure and the contributions of the constituents of (2) are not available yet; i.e., we cannot yet know of the impending type mismatch between *begin* and its second argument. Thus, there seems to be no motivation for this insertion. However, this problem could be overcome by an appropriate syntax-semantics interface that introduces a metavariable between the semantic contributions of *begin* and of its second syntactic argument regardless of their types. This variable could then be instantiated by an operator like Q_T - or be projected away (instantiated as the identity mapping) in case there is no type mismatch.

Another, more serious problem is the spellout of the operator Q_T. The operator Q_T we had to assume in (8) had the meaning $\lambda\mathscr{P}\lambda x\lambda e.\mathbf{read}'(x,\mathscr{P})(e)$, which is rather different from Pustejovsky's (1991, 429) original definition of Q_T as an operator that maps the semantics of a noun onto its telic quale. This time, the operator must coerce the semantics of a whole NP in terms of the telic quale of its head noun. It is not clear to me how this could be effected in a compositional fashion, perhaps in terms of some percolation mechanism that makes available the head noun qualia at the NP level.

At this point the key problem for Pustejovskys analysis surfaces, a problem that underlies the appropriate formulation of coercion operators like Q_T. Since the mismatch in (2) is one between the verb and its object NP, we need an operator that maps the NP semantics onto a property. If one performs this mapping in terms of an operator like Q_T (plus a meaning postulate like (9)), this means that one specifies the *scope* of the quantifier Q that is introduced in the NP's determiner (I neglect proper nouns and pronouns here). However, the exact formulation of the coercion operator depends on the semantics of the head noun of the NP, which is part of the *restriction* of the quantifier Q. I.e., the problem is that the qualia information that is necessary for the coercion arises in the restriction of a quantifier, but must be shifted to its scope.

As an example, consider the instantiation of this shift in sentence (2). Here, Q is an existential quantifier that is introduced by the determiner of *a novel*. Its restriction, the semantics of *novel*, introduces the relation $\mathbf{read}'$ in its telic quale Q_T. But then, Q_T applies to the meaning of *a novel* as a whole, which eventually specifies the scope of the existential quantifier in terms of the relation $\mathbf{read}_*$.

But this strategy proves fatal in cases where type coercion and issues of quantifier scope interact. I expound this with Pustejovsky's analysis (8) of a sentence where type coercion takes place since *begin* appears as a transitive verb. My claim is that Pustejovsky's analysis of type coercion runs into problems for issues of quantifier scope.

His analysis enforces that *begin* outscopes the quantifier that is introduced in its object NP, which is contradicted by the data. This shows up when we substutite a universally quantifying NP for the indefinite object NP in (1) and (2). Compare e.g. (11) and (12): Even if we assume the 'reader' (telic) interpretation of (12), it lacks one of the two readings of (11).

(11) Max began to read every novel

(12) # Max began every novel

Both sentences share the reading 'for every novel, Max began to read it'. From this reading one may deduce that Max read at least a couple of pages in every novel. It remains open, however, whether Max stopped at this point or whether this eventuality became the first part of a larger eventuality in which the novels were read in parallel.[4] The wide scope of the quantifier reveals that a VP argument of *begin* is no scope island, i.e., any quantifier (not just indefinite ones) within this VP may take scope over the matrix verb *begin*.

In contrast, (11), but not (12), can refer to an initial part of an eventuality of Max's reading every novel. This reading entails only that at least an initial part of *one* single novel was read. In sum, the universal quantifier from *every novel* has always wide scope over the relation **begin′** in (12), but is scopally underspecified with respect to this relation in (11).

Now one might argue that this contrast, in particular the missing narrow-scope reading for (12), is to be expected, as it should follow from a more general regularity that lexical items like *begin* in (12) are scope islands.[5] However, this assumption seems to be in conflict with the observation that some lexical items, e.g. causative verbs like *open*, are not:

(13) (a) At first, only half of the windows was open.
(b) But then Max opened every window.

If we interpret (13b) in the context of (13a), this context bars understanding (13b) as 'for every window, Max caused its becoming open'. Thus, its less favoured second reading is forced: 'Max caused a change of state whose result was that every window was open'. The two readings can be described in a decomposition approach to the semantics of causative verbs. Following Dowty (1979), there is an operator CAUSE for causation. Its arguments introduce the causer and the caused eventuality (in the form of an individual and a proposition). The eventuality is a change of state, hence, it is decomposed into

[4] I.e., every part of every novel was read, but in an 'interleaved' fashion where the reading of one novel could always be interrupted by reading part of another one.

[5] This was suggested by one of the two anonymous reviewers.

an operator BECOME and its argument, a proposition that characterises the result or aftermath of the change of state.

The difference between the two readings of (13b) is the scope of the universal quantifier. In its reading in the context of (13a), it has narrowest scope, as shown in (14), whereas it outscopes both CAUSE and BECOME in the other reading, which is represented by (15):

(14) $\text{CAUSE}(\mathbf{max'}, \text{BECOME}(\forall x.\mathbf{window'}(x) \rightarrow \mathbf{open'}(x)))$

(15) $\forall x.\mathbf{window'}(x) \rightarrow \text{CAUSE}(\mathbf{max'}, \text{BECOME}(\mathbf{open'}(x)))$

Thus, the scope contrast between (12) and (11) cannot be put down to more general lexical properties of the respective matrix verbs.

Now the problem for Pustejovsky's analysis of (12) is that the technique of inserting a type coercion operator between a functor and its argument fixes the scope relations between the begin-relation (the functor) and the universal quantifier (which is part of the argument) *exactly the other way round*: since the operator scopally intervenes between the two, the functor must have scope over the argument. I.e., for the semantic representation of cases like (2) and (12) there seem to be two mutually exclusive requirements: scoping the NP over the verb is required to model the interpretations of the examples adequately, while at the same time the type coercion in these sentences calls for narrow scope of the NP. Thus, the issue of quantifier scope cannot be dismissed from analyses of type coercion, short of failing to capture core instances of type coercion like (12).

To overcome this conundrum, one might consider complementing Pustejovsky's analysis by a mechanism that handles quantifier scope, something like Logical Form in Generative Grammar (May 1985) or Cooper Storage (Cooper 1983). In my approach, such a mechanism is available as well, which is crucial for my analyses of reinterpreted *begin*-sentences in section 4. All these approaches distinguish two *parts* of the semantic contribution of the NP, first, a bound variable and second, a part that comprises the binder of this variable and its restriction (the semantics of the noun). Only the first part must always remain in situ.

This distinction is important for the analysis of sentences like (12), since it allows parts of the semantic contribution of the NP to have different scopes. This suggests a resolution of the conundrum in the following way: the condition that the coercion operator applies to the NP semantics is fulfilled in that it applies to the first part of the NP semantics only, which remains *in situ* in the scope of the operator. This, however, does not determine the scope of that part of the NP semantics that comprises the variable binder, therefore this part may have scope over the coercion operator (and also over expressions that have the coercion operator in their scope). My treatment of reinterpreted *begin*-sentences in section 4 adopts this strategy and thus manages to capture the scope relations for these sentences.

However, this strategy does not help for a QS account of data like (12), even though it would allow one to get the scope relations right. If one would let the coercion operator apply to only that part of the NP semantics that is constituted by the bound variable, the other part of the NP semantics, which comprises the Qualia Structure of the head noun, would be outside the scope of the coercion operator. But then there is nothing left in the scope of the coercion operator that provides the information that is crucial for the reinterpretation, viz., information on the purpose or the coming about of novels. It is not clear how the coercion operator could derive this information.

In sum, Pustejovsky's approach integrates part of the type coercion data into compositional semantics – but at a cost: the complexity of semantic representations and the power of semantic composition is enhanced considerably, which raises questions about the exact boundary between semantic knowledge and extralinguistic conceptual knowledge. What is more, this approach does not cover the interaction of type coercion and quantification, since it runs into trouble with the scope of NPs that are the object of *begin*.

I will now turn to the formal background of the proposed analysis, but first I will briefly reflect on the question of whether the decision to assign the same semantics to both syntactic variants of *begin* is sound. In particular, one might consider an alternative option in which the variants of *begin* have a different meaning.

2.3 An alternative analysis?

At a first glance, Pustejovsky's approach to *begin* seems to be not the only available option. One might also consider an analysis which assumes different though related meanings for each of the syntactic variants of *begin*. In this fashion, one could model cases like (2) without reinterpretation. The meaning of the transitive *begin* would then be something like a relation between two individuals x and y, and an eventuality e such that x begins doing something to y at e.

However, such a 1:1-relation between readings and syntactic variants of *begin* would be contradicted by sentences like (16) and (17), where the object of the transitive *begin* introduces an eventuality:

(16) Max began the lecture

(17) The committee began the discussion of the candidates

Intuitively, these sentences do not mean that Max or the committee did something to the lecture or the discussion, respectively. Rather, the matrix clause subject emerges as a participant of the eventuality described in the object NP. E.g., (16) and (17) refer to the beginning of a lecture *delivered by Max* and the beginning of a discussion *among the committee members*. Thus, these sentences can be interpreted literally and do not call for reinterpretation, even though they comprise the transitive variant of *begin*. In this respect, they pattern semantically with the *begin*-sentences where *begin* is a control verb. In fact, they have paraphrases that use this variant of *begin*:

(18) Max began lecturing

(19) The committee began discussing the candidates

This shows that it is not possible to put down differences in the interpretation of these sentences to differences in the syntactic arguments of *begin*.

3 Formal background

Before I present my own analysis to *begin*-sentences, I will first introduce its formal background: the semantic representation formalism in which my analyses are cast (as well as its general property of being an underspecification formalism) and the treatment of scope ambiguities and of reinterpretation that it offers.[6]

3.1 Underspecification formalisms

Underspecification formalisms were designed for the representation of semantically underspecified expressions, i.e., expressions whose meaning can only partially be determined by semantic construction. Consider e.g. the simple quantifier scope ambiguity (20), where either quantifier may outscope the other one:

(20) Every woman loves a man

(20) has two readings, which I gloss as '$\forall\exists$' (for every woman, a [possibly different] man) and '$\exists\forall$' (one single man who is loved by every woman). An assertion of (20) is understood as $\forall\exists \vee \exists\forall$, a denial of (20), as $\neg\forall\exists \vee \neg\exists\forall$. These two interpretations could not be captured if one simply determined the meaning of the ambiguous (20) as the disjunction of its possible disambiguations: in that case, one would falsely obtain $\neg(\forall\exists \vee \exists\forall)$ as the interpretation of the denial of (20).[7] However, if we regard the meaning of the ambiguous expression as the set of its fully specified readings, and assume that understanding such an expression proceeds by forming the disjunction of this set, these interpretations follow directly. For (20), the meaning is thus $\{\forall\exists, \exists\forall\}$. The assertion of (20) is understood as the disjunction of its readings $\{\forall\exists, \exists\forall\}$; its denial, as the disjunction of its readings $\{\neg\forall\exists, \neg\exists\forall\}$, which yields the desired interpretations (van Eijck and Pinkal 1996).

But, for more complex ambiguous expressions, generating and enumerating all their readings may become a very involved task, which is neither efficient nor cognitively adequate. Thus, to characterise the set of possible readings for ambiguous expressions without having to enumerate the readings, a number of formalisms were proposed that provide a compact description of the range of these readings in an *underspecified* representation. Examples for such formalisms are Quasi-Logical Form (QLF, Alshawi 1992), Underspecified Discourse Representation Theory (UDRT, Reyle 1993), Ambiguous Predicate Logic (Jaspars and v. Eijck 1996), Glue Language Semantics (Dalrymple et al. 1997), MRS (Minimal

[6]In the following, I will freely use α-equality (representations are semantically equivalent if they are only distinguished by the names of bound variables) and η-equality (equivalence of $\lambda x.P(x)$ and P, if x does not occur free in the function P) for expository purposes. I will rename bound variables and switch between '$\lambda x.P(x)$' and 'P' whereever necessary.

[7]This shows that the meaning of an ambiguous expression cannot be identified with its weakest reading, either, if there is one (for (20), it would be $\forall\exists$). Negating the weakest reading would give us the *strongest* reading for the denial of (20), viz., $\neg\forall\exists$.

Recursion Semantics, Copestake et al. 1997), Underspecified Semantic Description Language (USDL, Pinkal 1996), or Constraint Language for Lambda Structures (CLLS, Egg et al. 1998; Egg et al. 2001).

These formalisms augment semantic representations with metavariables and subordination relations between fragments of these representations. E.g., the semantics of (20) can be described in such a formalism. The description introduces the two quantifiers and the fact that they both subordinate (have scope over) the verb semantics, and that the arguments of the verb are variables that are bound by the quantifiers. The scope relation between the two quantifiers is left open. Ambiguity resolution can then be modelled as augmenting such descriptions, for (20), by information on the quantifier scope. This augmentation chooses one of the two feasible scope relations between the quantifiers. See (21) below for the formal account of the semantics of (20).

Underspecification formalisms have mainly been used to describe quantifier scope ambiguities. Recently, they were extended to other phenomena as well, e.g., ambiguities that arise for elliptical expressions. Type coercion (as well as other kinds of reinterpretation) is another field that suggests itself for an account in terms of underspecification formalisms. This is motivated by the observation that reinterpretation candidates are semantically underspecified (recall e.g. the leeway in interpreting (2)) and that reinterpretation involves the integration of additional information into partial semantic representations.

Using an underspecification formalism for an account of reinterpretation has the additional advantage that one gets for free an analysis of scope ambiguity (after all, these phenomena still are the most prominent instances of ambiguity that are covered in such formalisms). Such an analysis is necessary since an account of reinterpretation must be able to model the interdependence between reinterpretation and issues of quantifier scope.

After these remarks on underspecification formalisms, the next subsection presents the representation language that is used for the underspecified account of *begin*-sentences.

3.2 The Constraint Language for Lambda Structures (CLLS)

In this subsection I introduce the language CLLS, but only in a simplified form which suffices for the purpose of this paper. For the full information on CLLS, the reader is referred to Egg, Niehren, Ruhrberg, and Xu (1998), and Egg, Koller, and Niehren (2001). The relation between the simplified and the full CLLS notation is discussed in Egg (2000).

The simplified notation is an underspecified description of sets of λ-terms and has three main ingredients:

- *fragments* of λ-terms
- parts of such fragments that are not yet specified, called '*holes*' (written as $\square$)
- *dominance* (part-of) relations between fragments and holes (depicted by dotted lines)

As an example for the shorthand notation, consider (21), the representation for (20). This constraint captures the common ground between the two readings of the sentence:

$$\Box$$

(21) $\forall x.\ \mathbf{woman}'(x) \to \Box \qquad\qquad \exists y.\ \mathbf{man}'(y) \wedge \Box$

$$\mathbf{love}'(x, y)$$

(21) introduces the fragments for the two quantifiers. Their respective scopes are not yet completely fixed, but either one comprises the bottom fragment $\mathbf{love}'(x, y)$ as a proper or improper part. The fact that the semantics of the sentence is still open but comprises the two quantifier fragments is modelled in that the top fragment is merely a hole that dominates the two quantifier fragments. In addition, (21) states that the verb semantics (which is given in the bottom fragment) is part of the scope of either quantifier.

If a binder has scope over a hole it also has scope over any fragment dominated by that hole. In this way, one can represent e.g. the fact that the variable x in the bottom fragment is bound by the universal quantifier of the left fragment in any semantic representation compatible with (21).

All λ-terms that comply to (21) are *solutions* of the constraint. Solutions that comprise only material that is mentioned explicitly in the constraint are called *constructive*. For (21), there are two constructive solutions, which stand for the two readings of (20). This shows that (21) comprises the readings of the ambiguous (20) in an underspecified way.

(22) $\forall x.\mathbf{woman}'(x) \to \exists y.\mathbf{man}'(y) \ \wedge \mathbf{love}'(x, y)$

(23) $\exists y.\mathbf{man}'(y) \ \wedge \forall x.\mathbf{woman}'(x) \to \mathbf{love}'(x, y)$

(21) can also be satisfied by λ-structures that contain material not yet mentioned in the description. One could rule out these solutions by a general *constructiveness* restriction on these descriptions. While such a restriction would be appropriate if one was only concerned with scope ambiguities, it would be too strict for my underspecified account of reinterpretation, which crucially involves the integration of additional material into semantic representations.[8]

3.3 Reinterpretation in CLLS

Reinterpretation can be handled straightforwardly by underspecification formalisms like CLLS. This subsection briefly sketches the treatment of type coercion proposed in Egg

[8]This restriction would also bar an underspecified analysis of ellipses, see Egg, Koller, and Niehren (2001). However, as soon as constructiveness is abandoned in an underspecification formalism, the *capturing problem* arises: How can we ensure proper and invariant binding of variables in an underspecified representation? In particular, how can one preserve binding of variables across dominance relations for solutions where there is additional material between the fragments of binder and of variable: This additional material might comprise new binders, which are in danger of binding (and hence 'capturing') the variable. See Egg, Koller, and Niehren (2001) for the CLLS solution to this problem in terms of an explicit function that relates bound variables and their binders.

(2000) for cases of type coercion in the nominal domain that involve sortal incompatibilities, in particular, between verbs and their NP arguments. These cases are studied extensively in Dölling's work (see in particular Dölling 1992, 1995, 1998). For ease of exposition, eventuality variables and their binders are ignored in this section (except where indicated).

Consider e.g. sentence (24). A λ-term for its literal reading would be ill-formed, since it would introduce conflicting sortal restrictions on a universally bound variable: the subject NP restricts this variable to one sort (here, THING), while the verb restricts it to another, incompatible sort (here, SUBST[ance]).

(24) # Every bottle froze

The impending sortal conflict in (24) can be mended in terms of a reinterpretation operator that mediates between the conflicting sorts by mapping the property of freezing onto the property of *containing* a substance that is freezing (a property of things). Consequently, the interpretation of (24) is that the content of every bottle froze.

I anticipate potential sortal conflicts between a verb and its NP arguments in my semantic representations in that I do not directly combine the verb and the NP semantics during semantic construction. This buffers any such conflicts between them and leaves room for the integration of a reinterpretation operator to mediate these conflicts. Formally, I assume a dominance relation in between the verb semantics and the semantics of its argument(s). This dominance relation is introduced in the syntax-semantics interface (SSI).

The rules that are relevant for the construction of the semantics of (24) are given in (25)-(27) together with the corresponding syntax rule (in a simplified form that neglects the question of how to derive the 'top' holes in the dominance diamonds). In these rules, '$\llbracket \texttt{XP} \rrbracket$' means 'semantic representation of constituent XP'. In addition, the prefix '$\llbracket \texttt{XP} \rrbracket$:' before a fragment in the semantic representation means that this fragment is the main fragment in a complex constraint, which is addressed in semantic composition processes at subsequent steps in the derivation:

(25) $[_{\texttt{VP}}\ \texttt{V}]$ $\quad \overset{\text{(SSI)}}{\Rightarrow} \quad \begin{array}{l} \llbracket \texttt{VP} \rrbracket : \Box \\ \qquad\ \ \vdots \\ \qquad\ \ \llbracket \texttt{V} \rrbracket \end{array}$

(26) $[_{\texttt{NP}}\ \texttt{Det N}]$ $\quad \overset{\text{(SSI)}}{\Rightarrow} \quad \begin{array}{l} \llbracket \texttt{Det} \rrbracket(\llbracket \texttt{N} \rrbracket)(\lambda x.\ \Box\ ) \\ \qquad\qquad\qquad\quad\ \vdots \\ \qquad\qquad\qquad\quad\ \llbracket \texttt{NP} \rrbracket\text{: } x \end{array}$

(27) $[_{\texttt{S}}\ \texttt{NP VP}]$ $\quad \overset{\text{(SSI)}}{\Rightarrow} \quad \llbracket \texttt{S} \rrbracket\text{: } \llbracket \texttt{VP} \rrbracket(\llbracket \texttt{NP} \rrbracket)$

The crucial effect of these rules is that in the semantic representation of (24), the semantic contribution of the verb is set off in a fragment of its own due to (25). Thus, any sortal restrictions that a verb might impose on its arguments are buffered; the constraint is not ill-formed if arguments fail to comply to these restrictions.

Rule (26) introduces a dominance relation into the semantic contribution of an NP: it sets off a (dominating) fragment that comprises the respective quantifier (such fragments are called *N-fragments*) from the NP fragment proper, which consists merely of a bound variable. This dominance relation is crucial for the modelling of scope ambiguities as in (21): it allows other N-fragments to be dominated by the N-fragment of the subject NP.

The grammar fragment returns the semantic representation (28) for sentence (24) (ignoring matters of tense and sentence mood) on the basis of standard semantic representations for *every* ($\lambda Q \lambda P \forall x. Q(x) \to P(x)$), *bottle* ($\lambda x_{\text{THING}}.\mathbf{bottle}'(x)$), and *freeze* ($\lambda y._{\text{SUBST}}\mathbf{freeze}'(y)$).[9] The incompatibility of the sorts THING and SUBST(ance) models the sortal mismatch between NP and VP:

(28)
$$\begin{array}{rl} \forall x_{\text{THING}}.\mathbf{bottle}'(x) \to & \Box \\ & \vdots \\ [\![\mathsf{S}]\!] : & \Box\ (x) \\ & \vdots \\ \lambda y_{\text{SUBST}}. & \mathbf{freeze}'(y) \end{array}$$

Since (28) is the representation of an expression with only one quantifier, its upper dominance relation is of no avail. Hence, to simplify matters, I ignore it for the rest of this section (i.e., the relation of (im-)proper part between the hole of the top fragment and the fragment in the middle is specified as identity), which yields the constraint (29):

(29)
$$\begin{array}{rl} \forall x_{\text{THING}}.\mathbf{bottle}'(x) \to & \Box\ (x) \\ & \vdots \\ \lambda y_{\text{SUBST}}. & \mathbf{freeze}'(y) \end{array}$$

Constraint (29) indicates that reinterpretation must take place, since it has no constructive solution, due to the incompatible sorts that are involved in the two fragments. But the incompatibility of the two sorts does not render the constraint ill-formed (in that it has no solutions at all), since the dominance relation 'buffers' them. The dominance relation in (29) has two more functions: it indicates the position where reinterpretation must take place and models the semantic underdetermination of expressions that are to undergo reinterpretation: (29) has in principle an unlimited number of solutions, since the constraint leaves open that part of its solutions that lies in between the NP and the VP semantics. In other words, reinterpretation candidates are semantically ambiguous.

Next I will show how the extralinguistic contribution to the meaning of (24) in its preferred reading (which is contributed by the conceptual knowledge that bottles are containers for liquids) is integrated with the result of semantic construction.

This extralinguistic contribution is also expressed in the form of a constraint, here, (30). At the heart of this constraint is the container-content relation **CONT**. (30) maps properties P of substances onto properties of things, viz., the property of being a container for a substance for which P holds:

[9]See appendix A.1 for the details of the semantic derivation.

(30) $\lambda z_{\text{THING}} \exists y_{\text{SUBST}}.\mathbf{CONT}(z,y) \wedge \square(y)$

Now the interaction of the result of semantic construction and the extralinguistic reinterpretation operator can be modelled as follows. First, we identify them as crucial ingredients for the determination of the full interpretation of the reinterpretation candidate (24). In particular, the operator is used to resolve the underspecification in the result of semantic construction in that it fills in the 'gap' that is opened up by the dominance relation:

(31)
$$\begin{array}{lcl} \forall x_{\text{THING}}.\mathbf{bottle'}(x) \rightarrow \square\ (x) & & \\ \qquad\qquad\qquad\qquad\quad\ \vdots & \Leftarrow & \lambda z_{\text{THING}} \exists y_{\text{SUBST}}.\mathbf{CONT}(z,y) \wedge\ \square\ (y) \\ \qquad \lambda y_{\text{SUBST}}.\mathbf{freeze'}\ (y) & & \end{array}$$

Next, we integrate the two constraints:

(32)
$$\begin{array}{l} \forall x_{\text{THING}}.\mathbf{bottle'}(x) \rightarrow \square\ (x) \\ \qquad\qquad\qquad\qquad\quad\ \vdots \\ \qquad \lambda z_{\text{THING}} \exists y_{\text{SUBST}}.\ \mathbf{CONT}(z,y) \wedge \square\,(y) \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\ \vdots \\ \qquad\qquad\qquad\qquad \lambda y_{\text{SUBST}}.\ \mathbf{freeze'}(y) \end{array}$$

(32) has the constructive solution (33), since the potential sortal conflict is avoided due to the operator. This λ-term represents the meaning of (24); every bottle has some content which froze:

(33) $\forall x_{\text{THING}}.\mathbf{bottle'}(x) \rightarrow \exists y_{\text{SUBST}}.\mathbf{CONT}(x,y) \wedge \mathbf{freeze'}(y)$

This formalisation of reinterpretation concentrates on demarcating the exact boundary between linguistic and conceptual knowledge in the joint determination of a fully-fledged utterance meaning. However, it does not give a full answer to the question of where the reinterpretation operator comes from and how it is selected.

(24) illustrates that the result of semantic construction cannot restrict the range of potential reinterpretation operators considerably. It can merely specify (by the sortal incompatibility it incorporates) what kind of sortal shift the operator must effect (here, from a property of substances to a property of things).

Dölling (1995) suggests that human processing of reinterpretation eventually falls back on - and is thus constrained by - a suitable common-sense ontology. But then the availability of several ontologically equally plausible reinterpretation options should result in more than one equally preferred reading of a reinterpretation candidate. However, this prediction is not borne out, reinterpretation candidates characteristically have one preferred reading even if several reinterpretation possibilities were ontologically feasible. E.g., in (24) the move from a thing to its content is not the only option, one might also shift a thing to the substance that constitutes it. But this possibility is (at least) extremely dispreferred, which shows that there must be more than just an ontology to guide reinterpretation.[10]

This suggests that human processing of reinterpretation must also employ *preferences* of some sort. But then the meaning of reinterpretation candidates emerges as yet another instance of the phenomenon that there are preferences for the choice between different readings of an ambiguous expression. Further research is needed to work out the mechanisms that guide the selection of the preferred reading, however, I envisage that optimality-theoretic semantics as in Blutner (2000) will provide the tools for such an analysis.

To close off my remarks on the issue of interpretation preferences, I will discuss the issue of preferences for Pustejovsky's approach. It turns out that it would face the same problem here: Either of the two discussed options for interpreting (24) (in terms of the content or the constituting substance of the bottles) is at least echoed in the QS for *bottle*: in the telic quale [bottles have the purpose of storing liquids] and the formal quale [bottles consist of some material, preferably glass]. But the preference for the former option is not explained. What is more, some metonymies go beyond the range of QS, e.g., the metonymy from content to container: In (34) nothing in a QS for *wine* would introduce the container(s) of the wine, which have to undergo the labelling:

(34) Amélie labelled the wine

In sum, my approach to type coercion provides a well-formed semantic representation for examples like (24) and describes how linguistic and extralinguistic information are interleaved in the derivation of their full meaning. This approach (together with the CLLS account of scope ambiguities) is applicable to *begin*-sentences, too. The integrated treatment of reinterpretation and scope ambiguity in CLLS will prove crucial for an analysis of *begin*-sentences that overcomes the problems noted in 2.2 for Pustejovsky's analysis.

4 My analysis

The presentation of my own analysis of *begin*-sentences is given in two subsections, first, I will present sentences where *begin* has a VP complement, then I will move on to the case of *begin* as a transitive verb. Each syntactic alternative will trigger a different interface rule.

4.1 The case of *begin* with a VP complement

In this subsection I will turn to sentence (35) [= (11)] in greater detail.

[10]One of the few counterexamples that come to mind here is (i), adapted from Bierwisch (1983). Here the subject NP must introduce some abstract structure, but this could be provided by a metonymic shift from Bill to his manner or to his content of speech:

(i) Bill is hard to understand

(35) Max began to read every novel

For this case, we yet have to spell out the SSI rule for the combination of a transitive verb with its object NP and for the combination of a control verb like *begin* with its first argument (the other one is then the subject NP). Control verbs are assigned a special verbal category CV.

(36) $[_{\mathtt{VP}}\ \mathtt{V}\ \mathtt{NP}]$ $$\overset{\text{(SSI)}}{\Rightarrow}\ \begin{array}{l} [\![\mathtt{VP}]\!] : \square\ ([\![\mathtt{NP}]\!]) \\ \qquad\quad [\![\mathtt{V}]\!] \end{array}$$

(37) $[_{\mathtt{VP}_1}\ \mathtt{CV}\ \mathtt{VP}_2]$ $$\overset{\text{(SSI)}}{\Rightarrow}\ \begin{array}{l} [\![\mathtt{VP}_1]\!] : [\![\mathtt{CV}]\!](\lambda x.\ \square\ ) \\ \qquad\qquad\qquad\qquad [\![\mathtt{VP}_2]\!](x) \end{array}$$

The rule (36) for the V-NP combination introduces a dominance relation, which sets off the semantic contribution of the verb. In this way, one can model the fact that incompatibility between a transitive verb and its NP arguments may call for reinterpretation. The second rule introduces another dominance relation: It is important since quantifiers within the complement VP may take narrow scope with respect to the control verb. With these SSI rules we obtain the following semantic representation for (35):[11]

(38) $$\begin{array}{ccc} & \square & \\ [\![\mathtt{S}]\!] : \lambda e.\ \mathbf{begin}'(\mathbf{max}', \lambda z \square\ )(e) & & \lambda e''\forall y.\ \mathbf{novel}'(y) \rightarrow \square\ (e'') \\ & \square\ (y)(z) & \\ & \lambda y \lambda x \lambda e.\ \mathbf{read}_*\ (x, y)(e) & \end{array}$$

This semantic representation involves several dominance relations; excepting the one between the fragments '$\square(y)(z)$' and '$\lambda y\lambda x\lambda e.\mathbf{read}_*(x,y)(e)$', their purpose is to model the scope ambiguity between *begin* and *every novel* that is responsible for the two readings of (35).

In contrast, the dominance relation directly below the fragment '$\square(y)(z)$' serves to buffer potential (sortal or type-theoretic) conflicts between the verb and its two arguments. In case of such a conflict, a mediating reinterpretation operator may intervene between the semantic contributions of the verb and the arguments. No such conflict emerges for (35), therefore the dominance relation below '$\square(y)(z)$' is not needed there. This means that it is possible to find a constructive solution for the constraint (38).

There are two possibilities to arrange the fragments in (38): the N-fragment on the right may take scope over or below the *begin*-fragment on the left. The dominance relation

[11] For a detailed account of this derivation, the reader is referred to appendix A.2.

below '$\Box(y)(z)$' does not add any other (unwanted) scoping possibilities. This models the fact that (35) has the two readings (39) and (40) - which are the solutions of (38): (39) represents the reading in which Max read from every novel at least an initial part, (40), the reading in which he read at least an initial part of one of the novels:

(39) $\lambda e''\forall y.\mathbf{novel'}(y) \ \rightarrow \mathbf{begin'}(\mathbf{max'}, \lambda z\lambda e.\mathbf{read}_*(z,y)(e))(e'')$

(40) $\lambda e.\mathbf{begin'}(\mathbf{max'}, \lambda z\lambda e''\forall y.\mathbf{novel'}(y) \ \rightarrow \mathbf{read}_*(z,y)(e''))(e)$

Thus, the integrated treatment of scope ambiguity and reinterpretation does not lead to an overgeneration of readings for these examples, even though both phenomena are treated in a similar fashion (in that constraints are relaxed by the introduction of dominance relations). See Koller et al. (2000) for further discussion of this issue.

I conclude this section with a brief remark on the dominance relation below '$\Box(y)(z)$'. In (38), it was not needed, but there are cases of *begin*-sentences where the argument VP of *begin* exhibits a verb-argument mismatch just like the one expounded for sentence (24) above. In (41), the representation of the verb-object mismatch in the the VP complement of *begin* calls for such a dominance relation (assuming that phoning is a relation between human beings):

(41) # Max began to phone every office

In the next subsection, I move on to the more involved task of deriving an appropriate semantic representation for sentences in which *begin* appears as a transitive verb.

4.2 The case of *begin* as a transitive verb

The example on which I am going to explore the transitive variant of *begin* is (42) [= (12)]:

(42) # Max began every novel

Since *begin* appears as a transitive verb in (42), the crucial interface rule for the derivation of its meaning is (36). The resulting representation is (43):[12]

(43)
$$\begin{array}{rl} & \lambda e\forall z.\mathbf{novel'}(z) \rightarrow \Box\ (e) \\ & \vdots \\ [\![\mathrm{S}]\!] : & \Box\ (z)(\mathbf{max'}) \\ & \vdots \\ & \mathbf{begin'} \end{array}$$

Without any further assumptions, there is no scope ambiguity between the begin-relation and the universal quantifier. This means that my approach does not fall prey to the scope paradox as expounded for Pustejovsky's analysis: it does not falsely enforce narrow scope

[12]For details of the derivation of this structure, see appendix A.2.

of the quantifier. This relieves the type coercion analysis of *begin* in its transitive version of one of its core problems.

The next step in the derivation of one of the readings of (42) consists in deriving a solution from its semantic representation (43). This constraint has no constructive solution, because the dominance relation below the main S fragment relates a hole that would have to be identified with a fragment of type $\langle e, \langle e, \langle s, t\rangle\rangle\rangle$ (i.e., the type of a transitive verb) with a fragment that maps expressions of type $\langle e, \langle s, t\rangle\rangle$ onto expressions of the same type. Thus, the dominance relation between these fragments is crucial in that it opens a gap for the insertion of an appropriate reinterpretation operator.

The other dominance relation is unproblematic in this respect: The hole of the N-fragment requires identification with something of type $\langle s, t\rangle$, hence could be identified with the main `S` fragment.

In order to obtain a solution for (43), further constraints must be integrated with this constraint. In the discussion of Pustejovsky's approach in section 2.2, I have argued that the source of this information is conceptual knowledge: The range of information that may be integrated here by far exceeds those pieces of information that could possibly be encoded in the semantic lexicon (e.g., in a Qualia Structure).

Formally, one can obtain a well-typed constructive solution by augmenting (43) with a further constraint that fills in the gap below the main `S` fragment. The smallest augmentation that leads to a λ-structure with a well-typed constructive solution must loosely speaking turn the first argument z in the main `S` fragment into an expression of type $\langle e, \langle s, t\rangle\rangle$. To this end, we need a reinterpretation operator '$\lambda z.\Box(\varphi(z))$', where φ is a relation of type $\langle e, \langle e, \langle s, t\rangle\rangle\rangle$ (e.g., $\mathbf{read}_*$) and the hole can be identified with a fragment of the type of the bottom fragment $\mathbf{begin}'$.

For the identification of an appropriate φ, one must rely on conceptual knowledge; this is the way in which a piece of knowledge like the fact that novels come into existence through writing is brought to bear on the interpretation of expressions like (42).

This subsumes, but is not restricted to, Pustejovsky's intuitions: the observation that the 'author' (agentive) and the 'reader' (telic) readings of sentence (42) are more prominent than others can be put down to the fact that generic conceptual knowledge (which must be accessed during the derivation of these readings) is privileged over context-specific conceptual knowledge since it is available regardless of context factors. However, this does not preclude the possibility that specific conceptual knowledge (e.g., the fact that Max happens to be a professor of literary studies, see section 2.2) is more salient than pieces of generic conceptual knowledge and hence triggers different interpretations of *begin*-sentences.

The result of inserting such a reinterpretation operator in (43) and deriving a constructive solution from the resulting constraint is shown in (44), for the instantiation of φ by $\lambda y \lambda x \lambda e.\mathbf{read}_*(x, y)(e)$):

(44) $\lambda e'' \forall y.\mathbf{novel}'(y) \rightarrow \mathbf{begin}'(\mathbf{max}', \lambda x \lambda e'.\mathbf{read}_*(x, y)(e'))(e'')$

The resulting λ-term (44) looks familiar, in fact, it is an alphabetic variant of (39), which models one of the readings to (35). Thus, the proposed analysis of *begin* handles both syntactic variants of *begin* and, what is more, brings out the semantic similarity between them: Both (35) and (42) refer to the start of one or more novel-related activities by Max.

The proposed analysis leaves unchanged the semantics of the NP object of *begin*; instead, the meaning of the verb is augmented. Additional evidence for this strategy comes from sentences like (45):[13]

(45) Mary read, and actually finished, a novel by James Joyce

Due to the coordination, reinterpretation cannot affect the NP *a novel by James Joyce*: An attempt to coerce it into an eventuality-denoting expression would lead to a mismatch between the reinterpreted NP and the verb *read*.

In this section, I have shown how the semantic representations of *begin*-sentences are derived and processed. The semantic contribution of the verb is the same in its two syntactic variants, however, this contribution is integrated differently in each case because for each variant of *begin*, there is a different interface rule that applies.

5 Conclusion

To sum up the results of this paper, let us briefly reflect on the question of how the proposed approach handles the problems noted for Pustejovsky's treatment of sentences like (42), which comprise the transitive variant of *begin*. First, his treatment ran into trouble with the scope of the object NP: contrary to intuitions on such sentences, it had to assume narrow scope of this NP with respect to the verb. Otherwise, it was not possible to express the fact that for the transitive *begin*, the reinterpretation operator mediates between the conflicting types of verb and object NP.

In my account, this problem does not arise, since here the NP semantics distinguishes the N-fragment (for quantifier and its restriction) from another fragment for the NP semantics proper. This second fragment comprises only the variable that is bound by the respective quantifier of the NP. Thus, one can insert a reinterpretation operator between the semantic contributions of the verb and of its object NP without determining the scope relation between the two constituents: the operator intervenes between the constraint part for the verb and the fragment comprising the NP semantics proper, but ignores the NP's N-fragment. What is more, the proposed analysis derived the wide scope of the object NP over *begin* without further assumptions.

The second class of problems centred upon the question of how to delimit linguistic knowledge against conceptual knowledge. Deriving the semantics of sentences that comprise the transitive variant of *begin* with the help of a Qualia Structure and composition processes that access suitable qualia predicts a quite restricted range of readings for these sentences. E.g., for (42), it predicts a 'reader' and an 'author' reading. But this does not

[13] This example is due to one of the anonymous reviewers.

mesh with the fact that in appropriate contexts there are many more readings available for such sentences.

Therefore, I describe the meaning of sentences like (42) in such a way that they emerge as instances of reinterpretation. This analysis uses an appropriate underspecification formalism, which can also handle the scope interaction of *begin* and scope-bearing NPs in its second argument. Regarding these sentences as cases of reinterpretation means that they are semantically ambiguous; the derivation of full meanings for them must fall back on conceptual knowledge.E.g., for (42), a constructive solution is obtained only by augmenting its semantic representation (43) by a constraint for a reinterpretation operator that introduces some (contextually appropriate) novel-related activity. The choice of a particular operator was shown to be influenced by preferences. In this respect, the interpretation of (42) and similar cases is like the interpretation of other ambiguous expressions.

The interpretation potential that Pustejovsky encodes in terms of the Qualia Structure is subsumed under this approach: (43) can be augmented with reinterpretation operators that refer to the origin or the purpose of a novel to obtain the readings he predicts. But the semantic representation of (42) does not exclude other readings. In sum, the proposed approach can handle the issues in the semantics of *begin* that were problematic for Pustejovsky's approach.

Author's address:
Universität des Saarlandes, FR 4.7
Postfach 151150
D-66041 Saarbrücken

A The syntax-semantics interface

In this appendix, the interface rules are expounded by showing in detail how the meanings of relevant example sentences of the paper are to be derived. I start with the derivation of the simple sentence (24) (where eventuality arguments are neglected), and then go on to the derivation of the *begin*-sentences.

A.1 A simple example

As a first example for the interface, consider the derivation of sentence (24) - without tense and sentence mood -, repeated here for convenience as (46):

(46) # Every bottle froze

We start with the following lexical entries:

- *every*: $[\![\texttt{Det}]\!]$: $\lambda P \lambda Q \forall y.P(y) \to Q(y)$

- *bottle*: $[\![\texttt{N}]\!]$: $\mathbf{bottle}'$
- *freeze*: $[\![\texttt{V}]\!]$: $\mathbf{freeze}'$

The first rule (25), repeated here as (47), returns (48) as the semantics of the VP *froze*:

(47) $[_{\texttt{VP}}\ \texttt{V}] \quad \stackrel{\text{(SSI)}}{\Rightarrow} \quad \begin{array}{rl} [\![\texttt{VP}]\!] : & \Box \\ & \vdots \\ & [\![\texttt{V}]\!] \end{array}$

(48) $\begin{array}{rl} [\![\texttt{VP}]\!] : & \Box \\ & \vdots \\ & \mathbf{freeze}' \end{array}$

The rule (26), repeated here as (49), constructs the semantic representation for the NP *every bottle* (50):

(49) $[_{\texttt{NP}}\ \texttt{Det N}] \quad \stackrel{\text{(SSI)}}{\Rightarrow} \quad \begin{array}{rl} [\![\texttt{Det}]\!]([\![\texttt{N}]\!])(\lambda x. & \Box\ ) \\ & \vdots \\ & [\![\texttt{NP}]\!]\text{: } x \end{array}$

(50) $\begin{array}{rl} \forall y.\mathbf{bottle}'(y) \rightarrow & \Box \\ & \vdots \\ & [\![\texttt{NP}]\!]\text{: } y \end{array}$

Finally, the rule (27), repeated as (51), is used to derive the semantics of the whole of (46). It 'glues together' (48) and (50) at the main fragments, which returns (52), which is identical to (28), except for the sorts of the arguments of the two properties:

(51) $[_{\texttt{S}}\ \texttt{NP VP}] \quad \stackrel{\text{(SSI)}}{\Rightarrow} \quad [\![\texttt{S}]\!]\text{: } [\![\texttt{VP}]\!]([\![\texttt{NP}]\!])$

(52) $\begin{array}{rl} \forall y.\mathbf{bottle}'(y) \rightarrow & \Box \\ & \vdots \\ [\![\texttt{S}]\!] : & \Box\ (y) \\ & \vdots \\ & \mathbf{freeze}' \end{array}$

A.2 The *begin*-sentences

In this subsection, I will expound the derivation of the semantic representations for the sentences (11) and (12), repeated here as (53) and (54):

(53) Max began to read every novel

(54) # Max began every novel

The semantic representations of the involved lexical entries - this time with explicit eventuality arguments - are the following:

- *every*: $[\![\texttt{Det}]\!]$: $\lambda P \lambda Q \lambda e \forall y. P(y) \to Q(y)(e)$
- *novel*: $[\![\texttt{N}]\!]$: $\mathbf{novel'}$
- *read*: $[\![\texttt{V}]\!]$: $\mathbf{read}_*$ (of type $\langle e, \langle e, \langle s, t \rangle\rangle\rangle$)
- *begin*: $[\![\texttt{CV}]\!]$: $\lambda P \lambda y \lambda e.\mathbf{begin'}(y, P)(e)$
- *Max*: $[\![\texttt{NP}]\!]$: $\mathbf{max'}$ (of type e)

The first rule that applies in (53) is the rule (36), repeated here as (55), which derives the semantic representation of a VP from the representations of a transitive verb and its object NP. Setting out from the semantic representation of *read* and a representation of *every novel* that is analogous to the one of *every bottle* in (50) above, we obtain (56) as the semantic representation of *read every novel.* Here the main fragment is in the middle, which will turn out crucial for the modelling of scope ambiguity:

(55) $[_{\texttt{VP}}\ \texttt{V NP}]$ $\overset{\text{(SSI)}}{\Rightarrow}$
$$\begin{array}{l} [\![\texttt{VP}]\!] : \square\ ([\![\texttt{NP}]\!]) \\ \qquad\quad\ [\![\texttt{V}]\!] \end{array}$$

(56)
$$\begin{array}{r} \lambda e \forall z.\mathbf{novel'}(z) \to \square\ (e) \\ [\![\texttt{VP}]\!] : \square\ (z) \\ \mathbf{read}_* \end{array}$$

The next rule combines the semantics of control verbs and of their VP arguments. In the resulting semantic representation, the contribution of the VP argument ($\texttt{VP}_2$) is set off in a fragment of its own. The semantics of *begin to write a novel* is thus represented by (58), where the scope between the begin- and the novel-fragment is not yet determined. (Recall that the derivation is simplified in that the upper half of the dominance constraint is not derived.)

(57) $[_{\texttt{VP}_1}\ \texttt{CV}\ \texttt{VP}_2]$ $\overset{\text{(SSI)}}{\Rightarrow}$
$$\begin{array}{l} [\![\texttt{VP}_1]\!] : [\![\texttt{CV}]\!](\lambda x.\ \square\ ) \\ \qquad\qquad\qquad\quad [\![\texttt{VP}_2]\!](x) \end{array}$$

(58)
$$\begin{array}{lcl} [\![\texttt{VP}]\!] : \lambda y \lambda e.\mathbf{begin'}(y, \lambda x.\ \square\ )(e) & & \lambda e \forall z.\mathbf{novel'}(z) \to \square\ (e) \\ & \square\ (z)(x) & \\ & \mathbf{read}_* & \end{array}$$

Finally, the rule (51) for the derivation of sentence meanings out of subject and predicate meanings applies to derive the semantics of (53). (51) operates locally on the main fragment of the VP semantic representation, all other fragments remain unchanged. (59) is equivalent to (38) without the top hole:

(59)
$$[\![\mathtt{S}]\!] : \lambda e.\mathbf{begin'}(\mathbf{max'}, \lambda x.\ \Box\ )(e) \qquad\qquad \lambda e \forall z.\mathbf{novel'}(z) \rightarrow \Box\ (e)$$
$$\Box\ (z)(x)$$
$$\mathbf{read}_*$$

The last example to be covered is (54). Here *begin* appears as a transitive verb, which means that rule (55) is used to derive the meaning of *begin a novel*. Wide scope of the universal quantifier over *begin* is derived directly through this rule:

(60)
$$\lambda e \forall z.\mathbf{novel'}(z) \rightarrow \Box\ (e)$$
$$[\![\mathtt{VP}]\!] : \Box\ (z)$$
$$\lambda P \lambda x \lambda e.\ \mathbf{begin'}\ (x, P)(e)$$

Another application of rule (51) then returns (61) as the semantic representation for (54). This is equivalent to (43):

(61)
$$\lambda e \forall z.\mathbf{novel'}(z) \rightarrow \Box\ (e)$$
$$[\![\mathtt{S}]\!] : \Box\ (z)(\mathbf{max'})$$
$$\lambda P \lambda x \lambda e.\ \mathbf{begin'}\ (x, P)(e)$$

References

Alshawi, H. (ed.) (1992). *The Core Language Engine*. Cambridge: MIT Press.

Bierwisch, M. (1983). Semantische und konzeptionelle Repräsentation lexikalischer Einheiten. In R. Růžička and W. Motsch (eds), *Untersuchungen zur Semantik*, 61–99. Berlin: Akademie-Verlag.

Blutner, R. (2000). Some aspects of optimality in natural language interpretation. *Journal of Semantics 17*, 189–216.

Cooper, R. (1983). *Quantification and syntactic theory*. Dordrecht: Reidel.

Copestake, A., D. Flickinger, and I. Sag (1997). Minimal Recursion Semantics. An introduction. Available from `ftp://ftp-csli.stanford.edu/linguistics/sag/mrs.ps`.

Dalrymple, M., J. Lamping, F. Pereira, and V. Saraswat (1997). Quantifiers, anaphora, and intensionality. *Journal of Logic, Language, and Information 6*, 219–273.

Dölling, J. (1992). Flexible Interpretation durch Sortenverschiebung. In I. Zimmermann and A. Strigin (eds), *Fügungspotenzen*, 23–62. Berlin: Akademie-Verlag.

Dölling, J. (1995). Ontological domains, semantic sorts and systematic ambiguity. *International Journal of Human-Computer Studies 43*, 785–807.

Dölling, J. (1998). Modifikation von Resultatszuständen und lexikalisch-semantischen Repräsentationen. In P. Ludewig and B. Geurts (eds), *Lexikalische Semantik aus kognitiver Sicht*, 173–206. Tübingen: Narr.

Dowty, D. (1979). *Word meaning and Montague grammar*. Dordrecht: Reidel.

Egg, M. (2000). *Flexible semantic construction: the case of reinterpretation*. Habilitation thesis, Universität des Saarlandes.

Egg, M., A. Koller, and J. Niehren (2001). The constraint language for lambda-structures. *Journal of Logic, Language, and Information 10*, 457–485.

Egg, M., J. Niehren, P. Ruhrberg, and F. Xu (1998). Constraints over lambda-structures in semantic underspecification. In *Proceedings of ACL/COLING '98*, Montreal, Canada, 353–359.

Jaspars, J. and J. v. Eijck (1996). Underspecification and reasoning. In R. Cooper et al. (eds), *Building the framework*, FraCaS Deliverable D 15, Chapter 10.2, 266–287.

Koller, A., J. Niehren, and K. Striegnitz (2000). Relaxing underspecified semantic representations for reinterpretation. *Grammars 3*, 217–241.

Lascarides, A. and A. Copestake (1998). Pragmatics and word meaning. *Journal of Linguistics 34*, 387–414.

May, R. (1985). *Logical Form. Its structure and derivation.* Cambridge/Mass.: MIT Press.

Partee, B. (1986). Noun phrase interpretation and type-shifting principles. In J. Groenendijk, D. de Jong, and M. Stokhof (eds), *Studies in Discourse Representation Theory and the theory of Generalized Quantifiers*, 115–143. Dordrecht: Foris.

Pinkal, M. (1996). Radical underspecification. In P. Dekker and M. Stokhof (eds), *Proceedings of the 10th Amsterdam Colloquium*, Amsterdam, 587–606. ILLC.

Pustejovsky, J. (1991). The Generative Lexicon. *Computational Linguistics 17*, 409–441.

Pustejovsky, J. (1995). *The Generative Lexicon*. Cambridge: MIT Press.

Reyle, U. (1993). Dealing with ambiguities by underspecification: construction, representation, and deduction. *Journal of Semantics 10*, 123–179.

van Eijck, J. and M. Pinkal (1996). What do underspecified expressions mean? In R. Cooper et al. (eds), *Building the framework*, FraCaS Deliverable D 15, Chapter 10.1, 264–266.