

A Dynamic Temporal Logic of Events, Intervals and States for Nominalization in Natural Language

Ralf Naumann

Seminar für Allgemeine Sprachwissenschaft

University of Düsseldorf

D-40225 Düsseldorf, Germany

Abstract

The interpretation of nominalized expressions in English poses several problems. First, it must be explained how their meanings are derived from the meanings of the underlying verbs. Second, different forms of nominalizations differ in their semantic behaviour. Finally, aspectual restrictions which exist for *ing_{of}*-nominals must be explained. The solution to be proposed is based on the assumption that non-stative verbs denote changes. Changes can be conceived of in two different ways, either as objects which bring about a particular result or as relations between states. A dynamic structure of events, intervals and states is defined in which both perspectives can be expressed by means of sorting the universe D. The basic idea is to augment a transition system for Dynamic Logic (DL) by a further domain of events such that programs from DL can be described either as objects or relations between states. The interpretation of verbs is based on the second perspective: they denote (generalized) relations between states. The interpretation of nominalized expressions uses the first perspective: they denote changes as objects. Different forms of nominalizations denote different sorts of objects which are systematically related to the denotation of the underlying verb.

1 Data and Evidence

In English (at least) three forms of nominalizations exist.

- (1) a. the performance of the song (by the soprano)
- b. the performing of the song (by the soprano)

- c. the soprano's performing the song

The NP in (1a) is usually called a *derived* nominal, the NP in (1b) an *ing_{of}* nominal and the NP in (1c) a gerundive nominal. It is commonly assumed that these forms are morphologically and semantically derived from the underlying verb, i.e. *perform* in (1). Evidence for this assumption comes from the following data. First, sentence (2a) is true just in case its corresponding 'statement of occurrence' (Zucchi (1993)) (2b) is true.

- (2) a. The soprano performed the song at 9:00.
- b. A performance/performing of the song by the soprano occurred at 9:00.

For gerundive nominals a similar equivalence exists. (3a) is a paraphrase of (3b) which is true just in case (3b) is true.

- (3) a. The soprano's singing the song surprised us.
- b. That the soprano performed the song surprised us.

From this equivalence it cannot be inferred that a gerundive nominal and a *that*-clause have the same denotation. Only the latter can be arguments of verbs like *believe* and *know* or of predicates like *is true* or *is false*. This is the so-called *distribution*-problem. This first set of data gives rise to the following constraint which a semantic analysis of nominalizations must satisfy (Zucchi (1993)): The semantic properties of *ing_{of}*, derived and gerundive nominals should be accounted for without assuming that they are listed in the lexicon, but rather by assuming that they are generated by a rule from the meanings of the corresponding verbs.

Second, similar to 'ordinary' nouns like 'man' derived/*ing_{of}*-nominals (but not gerundive

nominals) allow determiners (4a), yet all forms of nominalization can occur in subject-position (4b,c).

- (4) a. the/a/one performing/performance of the song
b. Performing the song is strange.
c. The performing (performance) of the song is strange.

These data show that nominalized NPs should get an interpretation which makes their type compatible with that assigned to 'ordinary' NPs like 'a man' or 'John'. Otherwise, predicates like 'is strange' will be type ambiguous, making it necessary to explain in a systematic way the (semantic) relationship that exists between different occurrences of a predicate. This yields a second constraint: Derived and *ing_{of}*-nominals must be assigned meanings in such a way that it is possible to give a uniform interpretation of determiners like *some* or *the*.

A third set of data concerns semantic differences between derived and *ing_{of}* nominals (which behave semantically (almost) alike) on the one hand and gerundive nominals on the other. First, whereas a sentence containing a derived/*ing_{of}*-nominal has two readings, the corresponding sentence with a gerundive nominal admits of only one reading.

- (5) a. The soprano's performing of the song surprised us.
b. The soprano's performing the song surprised us.

(5a) has two readings. According to the first, the *existential-reading*, it is the mere performance which surprised us and not some specific quality the performance had. On the second reading, the *property-reading*, it is some property the performance had which was surprising and not the fact that it occurred at all. By contrast, sentence (5b) with a gerundive nominal has only the existential-reading according to which the soprano was not expected to perform the song. The property-reading is excluded. A second difference concerns selectional distinctions: gerundive and *ing_{of}*-nominals do not occur with the same range of predicates.

- (6) a. The performing of the song by the sopra-

no was slow/sudden/took a long time.

- b. *The soprano's performing the song was slow/sudden/took a long time.

Whereas gerundive nominals are not possible with predicates like *is slow* or *take a long time*, they are acceptable with *surprise*. For *ing_{of}*-nominals there are no such restrictions. A final difference has to do with restrictions on which predicates do at all admit a particular type of nominalization. Whereas all verbs admit gerundive nominals, *ing_{of}*-nominalization is excluded with stative verbs like *know* or *love*.

- (7) a. *John's knowing of the answer (*the knowing of the answer by John)
b. *Bill's loving of Mary (*the loving of Mary by Bill)

A third constraint therefore is that an analysis must admit an explanation of the semantic distinctions between the different forms of nominalization.

2 A Dynamic-Temporal Event-Structure

In Naumann (1996,1997a,b) a theory of aspectual phenomena has been developed which is based on the intuition that non-stative verbs like *eat* or *run* express changes. The intuitive notion of a change comprises at least the following two aspects which are complementary to each other.

- (i) something (an object) which brings about the change
(ii) something (a result) which is brought about by the change which did not hold before the change occurred

The notion of a change that is used in (i) and (ii), respectively, refers in each case to the other aspect. In (i) 'change' is understood as the result that is brought about, i.e., in the sense that is captured by (ii), whereas in (ii) 'change' is meant as the object that brings about the result.

The second aspect can be described as a transformation of state. Before the change occurred, the world was in a particular state, say *s*, in which some condition (property) ϕ did not hold,

whereas after the change has occurred, the world is in a state s' in which ϕ does hold. This perspective is captured in Dynamic Logic (DL). Some input-state s in which $\neg\phi$ holds is transformed into an output-state s' in which ϕ holds. Program-letters π are interpreted as binary relations on the domain S of states: $[\pi] \subseteq S \times S$. The elements of $[\pi]$ (pairs of states) can be interpreted as particular executions of the program. Each program letter defines an accessibility relation such that DL is a polymodal logic. The fact that $(s, s') \in [\pi]$ means that state s' can be reached (or is accessible) from state s by executing the program π in s . It is therefore possible to view the program letter π as a label by which transitions between states can be decorated: $s \rightarrow(\pi) s'$ holds just in case $(s, s') \in [\pi]$ and is to be read as 'by executing π in state s state s' is reached'.

The disadvantage of the perspective which underlies DL is that changes are only derived objects, i.e. pairs of states or relations between states, that are defined in terms of the domain S of states. There is no separate domain E the elements of which are interpreted as bringing about the transformations of a state s to a state s' denoted by π . It is therefore not possible to interpret π , when it is used to label a transition, as the object that brings about this transition because it is not interpreted as an independent object. What is not captured in DL, therefore, is the first perspective mentioned above in (i) according to which a change can be understood as an object which brings about a particular result. This perspective is captured, among others, in event-semantics (ES), Krifka (1992), where events are interpreted as elements of a separate domain E . What is missing in ES is the second perspective of a change as a state-transformer according to which events effect changes by bringing about some result ϕ .

The two perspectives can be combined in the following way. Each event e is of a particular type P_v where the index v corresponds to some (non-stative) verb from English, say *eat* or *run*. The elements of the event-type P_v are characterized by bringing about (or effecting) a change with respect to some property. For instance, events of type *eat* bring about changes with respect to the mass of an object. This object is denoted by the internal argument of *eat*, say *an apple* in *eat an apple*. The result ϕ that holds with respect to this property upon termination of the event is that the mass is

zero, i.e., the object has vanished. Events of type *run*, on the other hand, bring about a change with respect to the location of the object that is denoted by the external argument, e.g. John in *John ran*. The location of the object at the end of the event is different from its location at the beginning of the event (to be precise, the change need not always consist in a change of location. Events of type *run around the park* are possible counterexamples. Rather, the object traverses a non-empty path. Whereas the path is empty when the event is started, it is non-empty at its end).

Events of type *eat* differ from those of type *run* in the way the result ϕ is brought about. This can be illustrated by means of the notion of a nucleus-structure (Moens/Steedman (1988)). A nucleus-structure consists of three parts: the inception-point (IP), the development-portion (DP) and the culmination-point (CP).

Nucleus-Structure

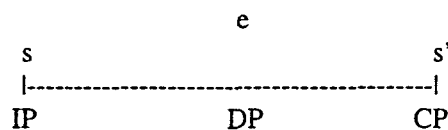


Figure 1

The result ϕ is evaluated at each part of the nucleus-structure. (Sequential) events of type *eat* and *run* agree on how ϕ is evaluated at the IP and at the CP: it is false at the IP and true at the CP. They differ with respect to the DP. Whereas ϕ is constantly false on the DP for events of type *eat*, it continuously holds for events of type *run*. From this it follows that for events of type *eating* ϕ only holds at the CP and is therefore true at exactly one point of the nucleus-structure, whereas for events of type *running* ϕ is false at only one point of the nucleus-structure, namely at the IP. The distinction between events of type *eat* and those of type *run* is therefore made in terms of how the result ϕ is evaluated at intermediate states of an occurrence. Aspectually it is therefore not the mere input-output behaviour that is important but also what happens in between the beginning and the end of an occurrence.

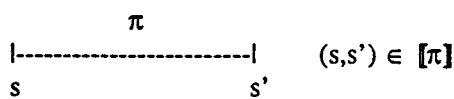
From a procedural perspective the nucleus-structure can be interpreted as the execution-sequence σ of some program π . The IP corresponds

to the input-state s , the CP to the output-state s' and the DP to the set of intermediate states s'' in between s and s' : $s < s'' < s'$ (where $<$ is the ordering on σ or some global ordering induced on the domain S). Thus, to each event e corresponds its execution-sequence $\sigma (= (s, s'))$ such that each event-type P_v induces a binary relation R_v on S such that each element of R_v is the execution sequence of a (completed) event $e \in P_v$. The binary relation R_v is the result of applying a so-called dynamic mode to a property of states Q . The dynamic modes corresponding to an accomplishment-verb like *eat* and an activity-verb like *run* are given in (8a) and (8b), respectively.

- (8) a. Con-BEC mode:
 $\lambda Q \lambda s s' [-Q(s) \wedge Q(s') \wedge \forall s'' [s < s'' < s' \rightarrow Q(s'')]]$
 b. Min-BEC mode:
 $\lambda Q \lambda s s' [-Q(s) \wedge Q(s') \wedge \forall s'' [s < s'' < s' \rightarrow \neg Q(s'')]]$

In (8) Q is a property of states corresponding to the result that an event of type P_v is supposed to bring about. Not all events of type P_v will satisfy $OP_v(Q)$ (for OP_v , the dynamic mode assigned to P_v) on their execution-sequence but only a certain subset, namely the completed sequential ones (for details see Naumann (1998) and section (2.1) below). The MIN-BEC-mode (partly) characterizes the class of accomplishments. This means that a verb-type v which belongs to this class is assigned this mode. Activity-verbs like *push* are assigned Con-BEC, the mode characterizing the class of activities. Other aspectual classes are characterized by different modes, Naumann (1997a,b), Naumann/Mori (1998). The dynamic mode assigned to an event-type P_v determines therefore the way in which a change effected by an event of this type is basically brought about. On this view events are a sort of 'reification' of particular executions of the program π . In terms of π as a label of transitions this means that instead of $s \rightarrow(\pi) s'$ one now has $s \rightarrow(e) s'$ such that $e \in P_v$ and $(s, s') \in R_\pi (= [\pi])$ where R_π is the binary relation induced by P_v . In contrast to the DL-perspective the label e can be interpreted as the object that brings about the transition from s to s' .

DL-perspective



new perspective

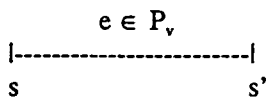


Figure 2

The above considerations are made precise in the following definition.

Definition: a dynamic-temporal structure of events, intervals and states of signature $\langle \text{VERB}_{ns}, \text{VERB}_s, \text{VAR}, \text{TR}, \text{DM} \rangle$ is a tuple $\langle D, E, S, O, T, I, \alpha, \beta, \mu, \gamma, \delta, \kappa, \kappa^*, \{R_v\}_{v' \in \text{VERB}_s}, \{\theta_i\}_{i \in \text{TR}}, \omega_{\text{ing(of)}}, \omega_{\text{gn}} \rangle$ such that

- (i) D is the domain of entities (or individuals)
- (ii) $E = \langle E, \sqsubseteq, \{P_v\}_{v \in \text{VERB}_{ns}} \rangle$ is an eventuality structure with
 - E is a set of events
 - \sqsubseteq is a binary relation on E (the part-of relation on E) which is a partial order
 - the P_v are unary relations on E
- (iii) $S = \langle S, \{R_{OP(Q)}\}_{OP \in \text{DM}}, \{Q_p\}_{p \in \text{VAR}} \rangle$ is a transition structure with
 - S is a set of states
 - $\{R_{OP(Q)}\}_{OP \in \text{DM}}$ is a set of parametrized relations on S . For each $Q \in \{Q_p\}_{p \in \text{VAR}}$ $R_{OP(Q)}$ is a set of (finite) sequences, i.e. a subset of S^* (S^* is the set of all finite sequences of elements from S) on which Q is evaluated in the same way.
 - the Q_p are unary relations on S ; they represent properties of states which can be changed by events of an appropriate type.
- (iv) O is the domain of urelements ('ordinary' objects)
- (v) $T = \langle T, < \rangle$ is a (time-)point-structure with
 - T the domain of time-points (moments of time)
 - $<$ is a strict partial, linear and discrete ordering on T
- (vi) $I = \langle I, <, \sqsubseteq \rangle$ is the interval-structure induced by T such that

- I is the set of all non-empty finite convex sets (intervals) over $\langle T, \diamond \rangle$
- $[t, t'] <_I [t_1, t_1']$ iff $t' < t_1$
- $[t, t'] \sqsubseteq [t_1, t_1']$ iff $t_1 \leq t$ and $t' \leq t_1'$

Instead of construing the interval-structure I from the point-structure, one can take intervals as primitive and define a relation *In* between elements of T and elements of I which holds just in case t is an element of i in the structure I induced by T.

- (vii) $\alpha : E \rightarrow T$ and $\beta : E \rightarrow T$ assign to an event e its beginning-time $\alpha(e)$ and end-time $\beta(e)$, respectively (it is required that $\alpha(e) < \beta(e)$). The functions α and β induce a function $\tau : E \rightarrow I$ which assigns to each $e \in E$ its run-time interval $\tau(e)$ such that $\tau(e) = i = \{t \in T \mid \alpha(e) \leq t \leq \beta(e)\}$. Thus, τ can also be defined as the product-mapping $\langle \alpha, \beta \rangle : E \rightarrow T \times T$.
- (viii) $\mu : T \rightarrow S$ is a function which assigns to each time point $t \in T$ the state $\mu(t) \in S$ which holds at t
- (ix) $\gamma : \{P_v\} \rightarrow \{Q_p\}$ is a function which assigns to each P_v a dynamic mode, i.e., a relation between properties of states and (finite) sequences of states (see (8) above for details).
- (x) $\delta : \{P_v\} \rightarrow \{Q_p\}$ assigns to each event-type P_v a property of states Q_p . This definition is too simple because the property depends on some argument x (e.g., it is the mass of an apple that gets changed). Therefore, δ should be defined as a function which assigns to each P_v for some v a function from D to $\{Q_p\}$. As the simplified definition does have no negative consequences for the analysis, it will be chosen for the sake of simplicity.
- (xi) The $\{R_v\}$ $v' \in \text{VERB}_s$ are n+2-ary relations on $D^n \times S \times S$ (for variable n) which are used to interpret stative verbs. $\text{VERB}_s = \{\text{know, love, believe, be ill, ...}\}$ is the set of stative verbs in English. Thus, $\text{VERB} = \text{VERB}_{ns} \cup \text{VERB}_s$ is the set of (basic) verbs in English.
- (xii) The θ_i $i \in \text{TR}$ are thematic relations (in effect, partial functions) like Agent or Pa-

tient, i.e. relations between events and individuals (not necessarily urelements, that is individuals of sort o).

- (xiii) $\kappa : \{P_v\} \rightarrow \theta^*$ assigns to each P_v an n-tuple of θ -relations, namely all those relations for which events of type P_v are defined (θ^* is the set of finite sequences (tuples) of thematic relations). The role of κ is similar to that of axioms like (i) which are used in ES:

$$(i) \forall e [P_{give}(e) \rightarrow \exists x_1, x_2, x_3 [(e, x_1) \in \theta_{Agent} \wedge (e, x_2) \in \theta_{Source} \wedge (e, x_3) \in \theta_{Goal}]]$$

κ^* is a (partial) function which assigns to an event-type P and an event e an n-tuple of objects:

$$\kappa^*(P)(e) = \langle x_1, \dots, x_n \rangle \text{ iff (i) } e \in P, \text{ (ii) } \kappa(P) = \langle \theta_1, \dots, \theta_n \rangle \text{ and (iii) } \kappa^i[\kappa(P)](e) = x_i$$

where κ^i is the n-th projection function which when applied to the value of κ for some P yields the n-th element of $\kappa(P)$: if $\kappa(P) = \langle \theta_1, \dots, \theta_n \rangle$, then $\kappa^i[\kappa(P)] = \theta_i$ if $1 \leq i \leq n$, undefined otherwise.

- (xiv) $E \subseteq D, S \subseteq D, T \subseteq D, O \subseteq D, I \subseteq D$
- (xv) The functions $\omega_{ing(of)}$ and ω_{gn} are defined below in sections (2.1) and (2.2) respectively.

The composition of α and μ as well as that of β and μ assign to each event e its source-state $\mu \cdot \alpha(e)$ and target-state $\mu \cdot \beta(e)$, respectively.

- (xvi) $\alpha^*(e) = s$ iff $\alpha(e) = t$ and $\mu(t) = s$
 $(\alpha^* = \mu \cdot \alpha)$
 $\beta^*(e) = s'$ iff $\beta(e) = t'$ and $\mu(t') = s'$
 $(\beta^* = \mu \cdot \beta)$

The product-mapping $\langle \alpha^*, \beta^* \rangle (= \tau^*) : E \rightarrow S^*$ assigns to each event e its execution sequence $\tau^*(e)$. The function μ which assigns to each t its corresponding state $\mu(t)$ can be extended to intervals $i \in I$ such that $\mu^* : I \rightarrow S^*$ assigns to each interval i a finite sequence of states in accordance with μ , i.e., $\mu^*(i) = \sigma$ such that $\forall i : 1 \leq i \leq n, s_i = \mu(t_i)$. Thus, $\sigma = \mu(t_1) \dots \mu(t_n)$ with $t_1 = t, t_n = t'$ and $i = [t, t']$. τ^* is the composition of τ and μ^* : $\tau^* = \mu^* \cdot \tau$.

According to (xiv), the domain D of entities is structured such that there are the subdomains E , S , T , I and O corresponding to the sorts ε (event), s (state), tp (time-point), i (interval) and o (urelement, 'ordinary' object). D itself corresponds to the universal sort e (note that 'e' is used ambiguously: it is either the universal sort or an element of E). Objects of basic sort α are α, α' and α'' for $\alpha = i, s, tp$ or o . Thus, α will be used ambiguously for sort symbols and objects of this sort. x_i are objects of the universal sort, i.e. elements from D .

2.1 The Interpretation of ing_{of} -Nominals

In the previous section it was shown that the intuitive notion of a change covers two different, yet complementary perspectives. A change can be viewed either as an object or as a transformation of state. The first perspective is related to the eventuality structure E whereas the second perspective is linked to the dynamic transition structure S . These two complementary perspectives will be used for the interpretation of verbs and derived/ ing_{of} -nominals. In principle, each of the two perspectives can be used for the interpretation of both verbs and derived/ ing_{of} -nominals. In ES the first perspective is chosen for the interpretation of verbs. An n -place verb v is interpreted as the $n+1$ -place relation (9).

$$(9) \quad \lambda x_1 \dots \lambda x_n \lambda e [P_v(e) \wedge \theta_1(e, x_1) \wedge \dots \wedge \theta_n(e, x_n)]$$

In contrast to ES , the translation of verbs in the present approach will be based on the second perspective of a change. This decision is based on two reasons. First, 'ordinary', i.e. non-derived nouns, like *man* for instance, are usually analyzed as denoting sets of objects of the basic domain D . Using type-theory, this means that they are translated as expressions of type $\langle e, t \rangle$ (where $D_e = D$). From this it follows that derived and ing_{of} -nominals should be interpreted as denoting sets of basic entities too. In the present framework there are five basic domains: O, E, I, T and S . It is therefore possible to distinguish between five different (non-logical) sorts of basic expressions because each domain can be taken to be the domain for expressions of a corresponding sort β where β is either o, ε, i, tp or s , i.e. the sort of objects, events, intervals, time-points or states. Above D was defined as

the union of the basic domains. The sorting of the universe D can be used at the level of types of expressions by letting each expression of type $\langle \beta, t \rangle$ for $\beta \in \{o, \varepsilon, i, s, tp\}$ be also of type $\langle e, t \rangle$. It is then possible to translate nouns uniformly as expressions of type $\langle e, t \rangle$, i.e. as denoting sets of elements from D , which can be further subclassified according to the sorts o, ε, i, tp and s . For instance, the translation of *man* will be an expression of type $\langle o, t \rangle$, denoting sets of 'ordinary' objects (that is, elements from O). Derived nouns like *performance of the song by the soprano* and ing_{of} -nominals like *performing of the song by the soprano* are translated as expressions of type $\langle \varepsilon, t \rangle$ which denote sets of events. The second reason has to do with the distinction between the translations of verbal and nominal expressions. On the present account, an n -place verb will be translated as an expression of type $\langle \alpha_1, \dots, \langle \alpha_n, \langle s, \langle s, t \rangle \rangle \rangle \dots \rangle$, where the $\alpha_i, 1 \leq i \leq n$, are the types of the (translations of the) subcategorized ('ordinary') arguments. From this it follows that the translations of sentence radicals like *John run* or *Mary eat an apple* are of type $\langle s, \langle s, t \rangle \rangle$. They denote binary relations on S and, consequently, no sets of basic objects. This is in contrast to nominal expressions the translations of which are of type $\langle \alpha^n, \langle \beta, t \rangle \rangle$ for α^n a (possibly empty) sequence of types and β a basic type. After the first n arguments have been discharged, nominal expression like *man* or *performing of the song by the soprano* therefore denote sets of elements of some basic domain D_β . Thus, verbal expressions are semantically distinguished from nominal ones. Whereas (the dynamic components of) the latter denote sets of entities, the (dynamic components of) former denote binary relations on S . The main task consists in showing how the non-relational perspective can be derived from the relational perspective that is used for the interpretation of verbs in the lexicon.

Each event e of type P_v for some v is related to an $n+2$ tuple $\langle x_1, \dots, x_n, s, s' \rangle$ of objects (individuals): $\forall e \in P_v \exists x_1 \dots \exists x_n \exists s s' [(e, x_1) \in \theta_1 \wedge \dots \wedge (e, x_n) \in \theta_n \wedge \tau^*(e) = (s, s')] by means of the functions $\alpha^*, \beta^*, \kappa$ and κ^* . The pair (s, s') is the execution-sequence of e . The x_i are the values for e of the thematic relations for which the event-type P_v is defined according to κ . From this it follows that each P_v induces an $n+2$ -ary relation R_v defined in (10).$

- (10) a. $R_v(x_1) \dots (x_n)(s)(s') = 1$ iff there is an $e \in P_v$ such that
- (i) $\kappa^*(P_v)(e) = \langle x_1, \dots, x_n \rangle$
 - (ii) $\alpha^*(e) = s$
 - (iii) $\beta^*(e) = s'$
- b. $R_v = \{ \langle x_1, \dots, x_n, s, s' \rangle \in D^n \times S \times S \mid \exists e [e \in P_v \wedge (e, x_i) \in \theta_i \wedge \tau^*(e) = (s, s')] \}$

The condition (i) requires that the value for e of each θ -role for which P_v is defined is an argument of the relation and that the i -th argument of R is the value of e for the i -th θ -role assigned to P_v . The second requirement implies that $(e, x_i) \in \theta_i$. In (10b) and the definitions that follow condition (i) will therefore be reduced to $(e, x_i) \in \theta_i$.

Let $\{R_v\} v \in \text{VERB}_{ns}$ be the set of (induced) binary relations corresponding to (10). R_v will be used in the interpretation of the verb v in the lexicon although this relation itself cannot be taken as the denotation of the verb. Rather, v denotes a subset of R_v , namely that subset such that $(s, s') \in \text{OP}(Q)$ with $\text{OP} = \gamma(P_v)$ and $Q = \delta(P_v)$. Recall that on the present account each transformation of states is supposed to be brought about by some event e of a particular type P_v . For a particular v , the change is brought about in some specific way determined by the dynamic mode $\gamma(P_v)$ with respect to the property $Q (= \delta(P_v))$. For instance, events of type eating bring about a change with respect to the mass of an object that is denoted by the internal argument and the property Q is that the mass be zero. An attempt to bring about Q by eating can fail. For instance, if only half of the object is eaten, Q does not hold in the final state of the execution. At the program-level the execution is a non-completed one which is a proper prefix of a completed one, where a completed execution is an execution (s, s') such that $(s, s') \in \gamma(P_v)(\delta(P_v))$. If $\gamma(P) = \text{Min-BEC}$, no proper prefix of an execution-sequence that is an element of the program is itself an element of the program. Yet these non-completed execution-sequences too are transformations of states that are brought about by an event e' of type eating but they are not denoted by *eat* because $\delta(P_v)$ does not hold in their final states.

Events e of type P_v the execution-sequences of which satisfy $\gamma(P_v)(\delta(P_v))$ are denoted by expressions where the arguments are non-mass and singular like *John eat the apple* or *The student push the cart to the station*. From a programming perspective, these transitions correspond to (exe-

cutions of) a sequential program. If the arguments are plural as in *eat three apples* or *eat apples*, the program must be executed for each apple separately such that one gets a *parallel* program that consists of three (or n) sequential subprograms each of which is of type $\gamma(P_v)(\delta(P_v))$. The execution sequence of the parallel program need not satisfy $\gamma(P_v)(\delta(P_v))$ which is exactly what is required witness the well-known aspectual contrast between *eat three apples* and *eat apples*. As shown in Naumann (1997a, 1998), given δ and γ , it is possible to calculate the dynamic mode for the result Q' (which is a function of the results for the sequential subprograms). The relation that is denoted by verbs in the lexicon is given in (11) ($\text{OP} = \gamma(P_v)$, $Q = \delta(P_v)$).

$$(11) R_{v(\text{OP}(Q))} = \{ \langle x_1, \dots, x_n, s, s' \rangle \in D^n \times S \times S \mid \exists e [e \in P_v \wedge (e, x_i) \in \theta_i \wedge \tau^*(e) = (s, s') \wedge (s, s') \in \text{OP}(Q)] \} \quad (1 \leq i \leq n)$$

For a given v , $R_{v(\text{OP}(Q))}$ is a subset of R_v . Similarly to R_v , it is determined by κ , κ^* , τ^* and, additionally, by δ and γ . According to (11), a non-stative verb in the lexicon denotes a (generalized) relation between states, i.e., it is an expression of sort $\langle \alpha, \langle s, \langle s, t \rangle \rangle \rangle$ where α is a (non-empty) sequence of basic sorts, as required at the beginning of this section. The relation R_v (or $R_{v(\text{OP}(Q))}$) represents the perspective of a change as a relation between states.

The relationship between P_v and $R_{v(\text{OP}(Q))}$ can also be expressed by means of a function $\rho : \{P_v\} \rightarrow \{R_{v(\text{OP}(Q))}\} v \in \text{VERB}_{ns}$ which maps an event-type P_v to the corresponding relation $R_{v(\text{OP}(Q))}$ (P is a variable which ranges over the P_v).

$$(12) \rho = \lambda P \lambda x_1 \dots \lambda x_n \lambda s s' \exists e [e \in P \wedge (e, x_i) \in \theta_i \wedge \tau^*(e) = (s, s') \wedge (s, s') \in \gamma(P_v)(\delta(P_v))]$$

The perspective on which a change is a basic object can be defined in a similar way by means of the functions κ , κ^* , τ^* , δ and γ .

- (13) a. $R_{e(v)(\text{OP}(Q))}(x_1) \dots (x_n)(e) = 1$ iff
- (i) $e \in P_v$
 - (ii) $\kappa^*(P_v)(e) = \langle x_1, \dots, x_n \rangle$
 - (iii) $\tau^*(e) \in \gamma(P_v)(\delta(P_v))$
- b. $R_{e(v)(\text{OP}(Q))} = \{ \langle x_1, \dots, x_n, e \rangle \in D^n \times E \mid [e \in P_v \wedge (e, x_i) \in \theta_i \wedge \tau^*(e) \in \gamma(P_v)(\delta(P_v))] \}$

Similarly to the relational perspective, a function $\rho^* : \{P_v\} \rightarrow \{R_{e(v)(OP(Q))} \mid v \in \text{VERB}_{ns}\}$ can be defined that expresses the relationship between P_v and $R_{e(v)(OP(Q))}$.

$$(14) \rho^* = \lambda R \lambda x_1 \dots \lambda x_n \lambda e [e \in P \wedge (e, x_i) \in \theta_i \wedge \tau^*(e) \in \gamma(P_v)(\delta(P_v))] (1 \leq i \leq n)$$

There is a 1-1 correspondence between the P_v and the $R_{e(v)}$ (or $R_{e(v)(OP(Q))}$). On the assumption that for $v \neq v'$, $v, v' \in \text{VERB}_{ns}$, $R_{v(OP(Q))} \neq R_{v'(OP(Q))}$, there is 1-1 correspondence η between $\{R_{e(v)(OP(Q))}\}$ and $\{R_{v(OP(Q))}\}$. If $\tau^*(e) = (s, s')$ then $\langle x_1, \dots, x_n, s, s' \rangle \in R_{v(OP(Q))} \Leftrightarrow \langle x_1, \dots, x_n, e \rangle \in R_{e(v)(OP(Q))}$. For a given v , the relation (13) is the denotation of the corresponding *ing_{of}*-nominal. The *ing_{of}*-morphem is then interpreted as mapping $R_{v(OP(Q))}$ to the corresponding relation $R_{e(OP(Q))}$, i.e. as the function $\omega_{ing(of)} : \{R_{v(OP(Q))}\} \rightarrow \{R_{e(v)(OP(Q))}\}$ which assigns to each element in its domain the corresponding relation in its range: $\omega_{ing(of)}(R_{v(OP(Q))}) = R_{e(v)(OP(Q))}$. Using λ -notation (and the equivalence between sets and their characteristic functions), $\omega_{ing(of)}$ is defined in (15). (R ranges over the relations in the domain of $\omega_{ing(of)}$).

$$(15) \omega_{ing(of)} = \lambda R \lambda x_1 \dots \lambda x_n \lambda e [(\eta^{-1}(R))(x_1) \dots (x_n)(e)]$$

According to (15), $\omega_{ing(of)}$ is the inverse of η . The *ing_{of}*-morphem does not change the argument-grid: each argument of the verb is also an argument of the nominalized expression (see Zucchi (1993) for details). Furthermore, *ing_{of}* is polymorphic in the sense that it applies to expressions of different arity.

The definition of $\omega_{ing(of)}$ in (15) is based on a relation between $R_{e(v)(OP(Q))}$ and $R_{v(OP(Q))}$. As was shown above, the relations $R_{v(OP(Q))}$ and $R_{e(v)(OP(Q))}$ can be determined as the ranges of two functions ρ and ρ^* , respectively, which map P_v to these relations. From these two relationships it follows that $R_{e(v)(OP(Q))}$ can be defined in terms of $R_{v(OP(Q))}$ if it is possible to 'recover' P_v from $R_{v(OP(Q))}$. This is the case if ρ is injective (i.e., for $v \neq v'$, $\rho(P_v) \neq \rho(P_{v'})$). $\omega_{ing(of)}$ can then be defined as the composition of ρ^{-1} and ρ^* , (16a), which yields (16b) when spelled out.

$$(16) \text{ a. } \omega_{ing(of)} = \rho^* \cdot \rho^{-1} \\ \text{ b. } \omega_{ing(of)} =$$

$$\lambda R \lambda x_1 \dots \lambda x_n \lambda e [e \in \rho^{-1}(R) \wedge (e, x_i) \in \theta_i \wedge \tau^*(e) \in \gamma(P_v)(\delta(P_v))] (1 \leq i \leq n)$$

According to (16), first ρ^{-1} is applied to $R_{v(OP(Q))}$ (i.e., the result of ρ for P_v), yielding P_v , and next ρ^* is applied to P_v yielding $R_{e(v)(OP(Q))}$: $R_{v(OP(Q))} \rightarrow (\rho^{-1}) P_v \rightarrow (\rho^*) R_{e(v)(OP(Q))}$.

The aspectual restriction on *ing_{of}*-nominals is explained as follows. Stative verbs like *know* or *love* do not express changes. This is captured by interpreting them as generalized binary relations on S to which no events correspond (the relations R_v , $v' \in \text{VERB}_i$ in (xi) above). The relations are therefore *not* derived from event-types P_v but are rather taken to be primitive. Consequently, there are no $R_{e(v)}$ (and $R_{e(v)(OP(Q))}$) such that the function ρ (or ρ^*) and therefore $\omega_{ing(of)}$ is not defined for them. Stative verbs denote sequences of states on which some condition Q continuously holds. The dynamic mode that corresponds to stative verbs is $\text{HOLD} = \lambda Q \lambda s s' \forall s'' [s \leq s'' \leq s' \rightarrow Q(s'')]]$ which can be interpreted as a kind of (iterated) test-program from DL.

2.2 The Interpretation of gerundive Nominals

In section (2.1) the derivation of *ing_{of}*-nominals from verbs was semantically analyzed as a mapping that makes use of the double perspective in which a change can be described in the dynamic event-structure defined above. The analysis of gerundive nominals is based on a third perspective. Whenever an event e of some type P occurs, a corresponding sequence σ is (partially) realized on the run-time $\tau(e)$ of e , namely that sequence σ such that $\tau(e) = i$ and $\mu^*(i) = (s, s')$ (i.e. $\tau^*(e) = (s, s')$). The occurrence of e therefore partly determines the world when it occurs by bringing about $\delta(P)$ in the way corresponding to $\gamma(P)$. Thus, each event e can be said to leave some trace behind. This trace concerns the property that is changed by the event, e.g. the location of some object. The trace of an event can be defined as a pair $\langle i, OP(Q) \rangle$ and thus as a property of intervals. Properties of intervals can be taken to be states of affairs (SOAs), i.e., partial constellations the world is in during some period. SOAs as properties of intervals are a third perspective in which a change can be described. It will be argued that gerundive nominals denote SOAs.

First, intervals are objects which are neither slow or sudden nor can they be said to take a long time. Second, on the existential reading it is the occurrence of an event which is surprising. But whenever an event has occurred, a corresponding SOA has been realized. This explains the equivalence between *The singing of the song by the soprano surprised us* and *The soprano's singing the song surprised us* on the existential reading. What is surprising about *e*'s run-time interval is the fact that $OP(Q)$ is realized on it. Third, for an event *e* there is the further possibility that some other property of it is surprising besides its property of having occurred at all. Fourth, as the domain *D* is structured such that *E*, *I*, and *O* are subsets of *D* and gerundive nominals are of sort $\langle i, t \rangle$, each expression of this sort is also of sort $\langle e, t \rangle$. One can therefore explain why NPs with gerundive nominals (and *ing_{of}*-nominals) can occur in subject-position of verbs like *is strange* without having to assume that the latter is ambiguous. Fifth, not only traces of events are states of affairs but any property of intervals for some *OP*, including *HOLD*, such that there are no aspectual restrictions on gerundive nominals. Therefore, the notion of a SOA not only comprises changes as intervals on which a change of some property *Q* occurred but also intervals on which *Q* continuously holds (as a result of some change brought about by an event *e*). The state of affairs that $OP(Q)$ has been realized is defined in (17).

$$(17) SOA_{OP(Q)} = \{i \in I \mid [\mu^*(i) \in OP(Q)]\}$$

$SOA_{OP(Q)}$ is that property which an interval *i* has just in case $OP(Q)$ is realized on it in the sense that the corresponding sequence $\mu^*(i)$ satisfies this property. $SOA_{OP(Q)}$ is the largest set such that $OP(Q)$ is realized on some *i*. The following subsets can be defined. First, the set of those intervals which are the run-time of an event *e* which brings about $OP(Q)$: $SOA_{e,OP(Q)}$, (18a). The events corresponding to $SOA_{e,OP(Q)}$ need not all belong to the same type P_v because $OP(Q)$ can be brought about by events of different types. The restriction of $SOA_{e,OP(Q)}$ to intervals which are the run-times of events of type P_v bringing about $OP(Q)$, $SOA_{v,OP(Q)}$, is defined in (18b).

$$(18) \text{ a. } SOA_{e,OP(Q)} = \{i \in I \mid \exists e \exists P [e \in P \wedge \tau(e) = i \wedge \delta(P) =$$

$$Q \wedge \gamma(P) = OP \wedge \mu^*(i) \in OP(Q)]\}$$

$$\text{ b. } SOA_{v,OP(Q)} = \{i \in I \mid \exists e \in E [e \in P_v \wedge \tau(e) = i \wedge \delta(P_v) = Q \wedge \gamma(P_v) = OP \wedge \mu^*(i) \in OP(Q)]\}$$

Syntactically, *ing_{of}*-nominals differ from gerundive nominals by the fact that the former can co-occur with determiners whereas this is excluded for the latter: *the performing of the song (by the soprano)* vs. **the performing the song*. There are different methods of how this difference can be handled (Zucchi (1993)). I follow Zucchi (1993) and apply the *ing_{gn}*-morphem after the ordinary arguments have been discharged that such *ing_{gn}* applies to an expression of sort $\langle s, \langle s, t \rangle \rangle$. For the sake of simplicity I will assume that the result is basically an expression of sort $\langle i, t \rangle$. Using a type-shifting rule the type can be lift to $\langle \langle i, t \rangle, t \rangle$.

The *ing_{gn}* denotes the function $\omega_{gn} : \{R^*_{v,OP(Q)}\} \vee \in VERB_{ns} \cup \{R^*_{v',HOLD(Q')}\} \vee \in VERB_s \rightarrow \wp(I)$ which maps those binary relations *R* on *S* which result from an *n*+2-nary relation $R_{v,OP(Q)}$ $\vee \in VERB$ after the first *n* arguments have been discharged to the set of intervals *i* on which an element of *R* has been actualized. Thus, for a non-stative verb *v* with non-mass singular arguments, $R^*_{v,OP(Q)}$ is the parametrized relation $\{\langle s, s' \rangle \in S \times S \mid \exists e [e \in P_v \wedge (e, x_i) \in \theta_i \wedge \tau^*(e) = (s, s') \wedge (s, s') \in \gamma(P_v)(\delta(P_v))]\}$ such that $\langle x_1, \dots, x_n, s, s' \rangle \in R_{v,OP(Q)}$. For other types of arguments, i.e. plural ones, the situation is more complicated, as explained in section (2.1). In this case the sequence (s, s') need not be an element of $\gamma(P_v)(\delta(P_v))$ because it is the execution-sequence of a parallel event. What is the case, rather, is that the execution-sequence of each sequential part is an element of $\gamma(P_v)(\delta(P_v))$. $R^*_{v',HOLD(Q')}$ is defined similarly for a stative verb *v'*. ω_{gn} is defined in (19) using λ -notation. (*R* ranges over the domain of ω_{gn})

$$(19) \omega_{gn} = \lambda R \lambda i \exists s s' [R(s)(s') \wedge \mu^*(i) = (s, s')]$$

On the assumption that *R* is denoted by an expression with non-mass singular arguments, the following cases can be distinguished. If $R = R^*_{v'}$ for some $v' \in VERB_s$, each $(s, s') \in R$ satisfies $HOLD(Q_v')$. If, on the other hand, $R = R^*_{v,OP(Q)}$ for some $v \in VERB_{ns}$, *R* corresponds to an event-type P_v , such that each $(s, s') \in R$ satisfies $OP(Q)$ for $Q = \delta(P_v)$ and $OP = \gamma(P_v)$. The intervals *i* are

then the run-times of the events of the corresponding subtype P'_v of P_v . (for instance, if R is the relation denoted by *John eat the apple*, $P'_v = P_{\text{John eat the apple}}$). If R is denoted by an expression with plural arguments, one gets the following. For a non-stative verb, the sequence (s,s') is the join of the execution-sequences σ of a set of sequential events such that each σ is an element of $\gamma(P_v)(\delta(P_v))$. For a stative verb v' , (s,s') is the join of sequences each of which is an element of $\text{HOLD}(Q_v)$. In each case the interval i is the join of intervals i' such that $i' \in \text{SOA}_{v(\text{OP}(Q))}$.

3 Conclusion

The analysis of nominalization in English presented in this paper is based on two assumptions.

- (i) Non-stative verbs denote changes
- (ii) The notion of change comprises different aspects

From (i) it follows that any theory must admit to model the intuitive notion of change. This seems to require some form of Dynamic Logic or some other form of programming logic in which the development of something going on in time can be represented. The second assumption seems to require the use of a (many-) sorted model such that the domains corresponding to the different sorts are systematically related to each other. Each domain must be equipped with some structure. The different aspects that the notion of change comprises are then linked to the different domains and their corresponding structures. Thus, besides sorting the universe, there must be enough structure in order that the different aspects can be modeled in an adequate way. This makes it necessary to use some form of a decompositional analysis of verbs. In the present approach the decomposition is done first with respect to the argument structure: n -place verbs are analyzed as denoting $n+2$ -ary relations, and second with respect to the semantic content of a verb. This content is split into a dynamic and a static component. The dynamic component denotes some type of transition between states which is brought about by an event. This component is therefore closely related to the first decompositional

component. The static component concerns the relationship between the arguments that are introduced by the decompositional analysis and the 'ordinary', subcategorized arguments. This relationship is expressed by thematic relations.

References

- Krifka, M. (1992) Nominal Reference, Temporal Constitution and Thematic Relations, in: Sag, I. et al. (eds.): *Lexical Matters*, CSLI Publications, Stanford, pp.29-53
- Moens, M./Steedman, M. (1988). Temporal Ontology and Temporal Reference, *J. of Computational Linguistics*, 14
- Naumann, R. (1996). Aspectual Composition in Dynamic Logic, Proc.10th Amsterdam Colloquium
- (1997a). A Dynamic Temporal Logic for Aspectual Phenomena in Natural Language, to appear in Proceedings 2. International Conference on Temporal Logic, (D. Gabbay/H. Barringer (eds.)), Kluwer
- (1997b). A Dynamic Decompositional Event-Semantics, talk given at the 2. conference on 'Sinn und Bedeutung', Berlin, submitted to *Linguistics and Philosophy*
- (1998). A Dynamic Logic of Events and States for the Interaction between Plural Quantification and Verb Aspect in Natural Language, to appear in Proc. 5th WoLLIC, Sao Paulo, Brazil
- Naumann, R./Mori, Y. (1998). An Analysis of Definite and Indefinite Japanese Aspectualities in Dynamic Decompositional Event Semantics, to appear in Proc. 6th Symposium on Logic and Language, Budapest
- Zucchi, A. (1993). *The Language of Propositions and Events*, Dordrecht, Kluwer