Dynamic Event Structure and Habitat Theory

James Pustejovsky
Computer Science Departament
Brandeis University
Waltham, Massachusetts, USA
jamesp@cs.brandeis.edu

Abstract

In this brief note, I explore the cognitive mechanisms involved in interpreting the meanings of events, as conveyed through language. Specifically, I examine the notion of *event simulation* in the construction of linguistic meaning. Simulations are a special class of minimal models, generated from linguistic input, under a number of agent-oriented cognitive constraints. An integral part of this model is a dynamic representation of processes and events, such as the *Dynamic Event Structure* presented here. I show how simulations are composed of entity and event *habitats*, which are contextualization functions, acting to embed a proposition into a minimal model.

1 Introduction

This paper presents a new interpretation of the frame-based event structures introduced in Pustejovsky and Moszkowicz (2011), in the context of Dynamic Interval Temporal Logic. The resulting model, Dynamic Event Structure (DES), has several desirable features, including its simplicity as well as its interpretation as a labeled transition system. I show how Aktionsarten distinctions are captured within this system, and point to how these can be deployed in a dynamic analysis of change predicates in language. I then explore the role that these event structures play in the construction of habitats and "event simulations" from linguistic utterances. Simulations are a special class of minimal models, generated from linguistic input, under a number of agent-oriented cognitive constraints.

2 The Semantics of Change

The topic of measuring change in linguistic theory has focused mainly on the few issues of count-mass distinctions, gradability in adjectives, and partitivity (Cresswell 1977, Klein 1991, Kennedy 2001 Link 1983, Gillon 1992, Schwarzschild 2002, Ladusaw 1982, de Hoop 1997). For our discussion, the most relevant discussion concerns telicity in predicates and gradability measures. Linguistic approaches to the analysis of gradable predicates have recently invoked a distinction between different types of scales (cf. Kennedy, 1999, 2003). For example, to explain the ability of verbs such as *eat* to shift between process and completive events, scales are invoked referencing the object extent of the theme:

(1) Incremental theme verbs:
  a. Sam ate ice cream. (atelic)
  b. Sam ate an ice cream cone. (telic)

Similarly, degree achievement behavior is available with predicates measuring some change, either existentially (2a) or quantifiably (2b).

(2) Change of state verbs:
  a. The icicle lengthened (over the course of a week). (atelic)
  b. The icicle lengthened two inches. (telic)

Most directed motion predicates exhibit this same behavior:

(3) Directed motion verbs:
  a. The plane ascended (for 20 minutes). (atelic)
  b. The plane ascended to cruising altitude. (telic)

Hence, as Levin (2009) points out, there are generalizations over scale behavior that can be noted, as summarized below.

(4) a. Property Scales: often found with change of state verbs;
b. Path Scales: most often found with directed motion verbs;
c. Extent Scales: most often found with incremental theme verbs.

While agreeing with the generalizations resulting from much of this work, we take a slightly different approach to how scales play a role in modeling the semantics of linguistic expressions. In the discussion that follows, we propose that all predication involves measuring an attribute against a scale. Further, we measure change according to this scale domain. Hence, scale theory is not peripherally involved in the semantics of selected properties, extent, and motion, but rather touches all aspects of predication in the language.

Any predication invokes reference to an attribute in our model. Often, but not always, this attribute is associated with a family of other attributes, structured according to some set of constraints. The least constrained association is a conventional sortal classification, and its associated attribute family is the set of pairwise disjoint and non-overlapping sortal descriptions (non-super types). Following Stevens (1946), we will call this classification a *nominal* scale, and it is the least restrictive scale domain over which we can predicate an individual. Binary classifications are a two-state subset of this domain.

When we impose more constraints on the values of an attribute, we arrive at more structured domains. For example, by introducing a partial ordering over values, we can have transitive closure, assuming all orderings are defined. This is called an *ordinal scale*. When fixed units of distance are imposed between the elements on the ordering, we arrive at an *interval scale*. Finally, when a zero value is introduced, we have a scalar structure called a *ratio scale*. Stevens' original classification is summarized below.

- Nominal scales: composed of sets of categories in which objects are classified;
- Ordinal scales: indicate the order of the data according to some criterion (a partial ordering over a defined domain). They tell nothing about the distance between units of the scale.
- Interval scales: have equal distances between scale units and permit statements to be made about those units as compared to other units; there is no zero. Interval scales permit a statement of "more than" or "less than" but not of "how many times more."
- Ratio scales: have equal distances between scale units as well as a zero value. Most measures encountered in daily discourse are based on a ratio scale.

Recent work has criticized approaches to the statistical analysis of data that apply Stevens' classification blindly, without acknowledging the subtlety of interpretation of the data (cf. Suppes et al., 1990, Velleman and Wilkinson, 1993, Luce, 1996). In reality, of course, there are many more categories than those given above. But our goal here is to use these types as the basis for an underlying cognitive classification for creating measurements from different attribute types. In other words, these scale types are models of cognitive strategies for structuring values for conceptual attributes associated with natural language expressions involving scalar values. We will show how adjectives and their associated verbs of change can be grouped into these scalar domains of measurement.

In the following discussion, we demonstrate how many aspects of measurement in language can be modeled dynamically. An interesting consequence of this analysis is a straightforward explanation of the distinction between non-incremental and incremental change predicates. In Pustejovsky and Jezek (2011, forthcoming), we explain how blended readings between the two arise, and how such expressions are actually to be expected, given the model.

Before we discuss how change can be structured, let us briefly discuss the domain of attributes to which individuals may be assigned values. In principle, this would refer to any attribute which may be constructed as a predicate over individuals.

Following Suppes et al (1990), we will treat measurement as a function of two variables: the *attribute* being modeled; and the *scale theory* with which it is being interpreted. One rich area of attribute classifications come from work in semantic field analysis (cf. Dixon, 1991, Lyons, 1977). In this work, attributes are categorized according to a thematic organization, centered around a human frame-of-reference, as lexically encoded in the language.

(5) a. DIMENSION: big, little, large, small, long, short

b. PHYSICAL PROPERTY: hard, soft, heavy, light

c. COLOR: red, green, blue

d. EMOTIONS: jealous, happy, kind, proud, cruel, gay

e. TEMPORAL: new, old, young

f. SPATIAL: above, up, below, near

g. VALUE: good, bad, excellent, fine, delicious

h. MANNER: sloppy, careful, fast, quick, slow

We can further distinguish between *intrinsic* (color, volume) and *extrinsic* attributes (distance, orientation) of an object. In principle, any of these attribute domains can be interpreted by means of one of the scale theories: Nominal; Ordinal; Interval; or Ratio.

But, just what is a measurement and what constitutes a scale? Below we introduce the theory as developed within measurement theory as reviewed by Krantz et al (1971) and Suppes et al (1990). Measurement, as stated above, is an assignment of a value, relative to an attribute $\mathcal{A}$ in our domain. The nature of the theory interpreting the attributes depends on what constraints we impose on how the values are assigned. Consider first Stevens' *nominal scale*. This theory has the properties that the objects in the domain $\mathcal{A}$ are distinct from one another relative to a particular attribute: that is, an object has P or does not have P; elements are not ordered relative to one another. A binary classification scheme is the simplest structure possible, as illustrated below for the attribute *animate*.

(6)

+ANIMATE	-ANIMATE
boy	plastic
tree	rock
worm	house
elephant	cup
grass	glass

Hence, no member in the scale -ANIMATE is any more or less an exemplar of that attribute. The elements of this set, $\{plastic, rock, house, cup, glass\}$, can be distinguished only if additional attributive constants are introduced, thereby creating new "scales". Obviously, this simple notion of scale reduces to the general notion of equivalence class and characteristic function.

A simple *ordinal scale* consists of a set of elements, $\mathcal{A}$, exhibiting the attribute to be measured, along with an ordering of A over this attribute, $\preccurlyeq$, where, if $a, b \in \mathcal{A}$, $a \succcurlyeq b$, then element a has at least as much of the attribute as does b: $\langle \mathcal{A}, \preccurlyeq \rangle$. An order-preserving transformation is monotonic, and hence transitivity holds; e.g., if $a \preccurlyeq b$ and $b \preccurlyeq c$, then $a \preccurlyeq c$. For lexically defined scalar positions over homogeneous sortal domains, for example, this can be used to compute transitive closure graphs, but not much else; e.g., the domain model below.

(7) a. John is short.
b. Mary is medium.
c. Bill is tall.
d. $\mathcal{M} \models j \preccurlyeq m \preccurlyeq b$

Of course, there is no clear *metric* to the ordering between two elements of the domain. An *interval scale* is a order-preserving structure that also has a composition operator, $\circ$, that maintains transitive closure within a scale of the composition of two values from that scale. This is lacking in a simple ordinal scale structure: $\langle \mathcal{A}, \preccurlyeq, \circ \rangle$. Comparisons between values on a scale are now possible because standard interval metrics are assumed to underlie the attribute values. Hence, interval scale theories are concatenation structures with commutativity and associativity properties.

3 Dynamic Event Structure

Given the above observations, the focus here is to provide a dynamic interpretation of how change is encoded within event structures. Although many event types can be adequately expressed as tree structures, Pustejovsky and Moszkowicz (2011) introduce a linear box notation, which they call an *event frame structure*, where single frames may extend linearly into frame sequences, but may also compose vertically, in parallel *tracks*. This was seen as a conceptual analogue to the structures used in Barselou's Frame Theory (Barselou, 2003).

Recall first the classic event structure distinctions of Generative Lexicon Theory (cf. Pustejovsky, 1995), shown below:

(8) a. EVENT $\rightarrow$ STATE | PROCESS | TRANSITION
b. STATE: $\rightarrow \; e$
c. PROCESS: $\rightarrow \; e_1 \dots e_n$
d. TRANSITION$_{ach}$: $\rightarrow$ STATE STATE
e. TRANSITION$_{acc}$: $\rightarrow$ PROCESS STATE

Let us assume a GL feature structure for the meaning of a linguistic expression:

$$\begin{bmatrix} P \\ \text{ARGSTR} = \begin{bmatrix} \text{ARG1} = x \\ \dots \end{bmatrix} \\ \text{EVENTSTR} = \begin{bmatrix} \text{EVENT1} = e1 \\ \text{EVENT2} = e2 \end{bmatrix} \\ \text{QUALIA} = \begin{bmatrix} \text{FORMAL} = P_2 \\ \text{AGENTIVE} = P_1 \end{bmatrix} \end{bmatrix}$$

Following general interpretations of qualia structure (cf. Bouillon, 1997), the qualia are naturally ordered over the temporal domain. That is, the predicates associated with each quale are interpreted as a sequence of "frames" of interpretation. This is illustrated below, where the matrix predicate, P, is decomposed into different subpredicates within these frames:

(9) $\mathrm{V}(\mathrm{A}_1, \mathrm{A}_2) \Rightarrow \lambda y \lambda x \boxed{P_1(x,y)}_A \boxed{P_2(y)}_F$

In the discussion that follows, we will adopt this interpretation for qualia structure specifically, and for predicative content more generally, in order to reinterpret our model of events for language. We will assume the model of predication presented in Pustejovsky and Moszkowicz (2011). In order to adequately model change as expressed in language, the representational framework should accommodate change in the assignment of values to the relevant attributes being tracked over time.

A dynamic approach to modeling updates makes a distinction between formulae, ϕ, and programs, π. A formula is interpreted as a classical propositional expression, with assignment of a truth value in a specific state in the model. For our purposes, a state is a set of propositions with assignments to variables at a specific time index. We can think of atomic programs as input/output relations, i.e., relations from states to states, and hence interpreted over an input/output state-state pairing (cf. Naumann, 2001).
Let us now reinterpret the Vendler event classes in terms of dynamic event structures. In order to access the various states in the temporal expressions in language, we adopt the modal operators from Linear Temporal Logic (LTL), $\circ$, $\Box$, $\Diamond$, and $\mathcal{U}$ (cf. Fernando, 2004, Kröger and Merz, 2008). Consider first the definition of a state.

(10) a. Mary was sick today.
b. My phone was expensive.
c. Sam lives in Boston.

We assume that a *state* is defined as a single frame structure (event), containing a proposition, where the frame is temporally indexed, i.e., $e^i \rightarrow \phi$ is interpreted as ϕ holding as true at time i. The frame-based representation from Pustejovsky and Moszkowicz (2011) can be given as follows:

(11) $\boxed{\phi}_e^i$

Propositions can be evaluated over subsequent states, of course, so we need an operation of concatenation, $+$, which applies to two or more event frames, as illustrated below.

(12) $\boxed{\phi}_e^i + \boxed{\phi}_e^j = \boxed{\phi}_e^{[i,j]}$

Semantic interpretations for these are:

(13) a. $[\![\boxed{\phi}]\!]_{\mathbf{M},i} = 1$ iff $V_{\mathbf{M},i}(\phi) = 1$.

b. $[\![\boxed{\phi}\boxed{\phi}]\!]_{\mathbf{M},\langle i,j \rangle} = 1$ iff $V_{\mathbf{M},}(\phi) = 1$ and $V_{\mathbf{M},j}(\phi) = 1$, where $i < j$.

While it may seem to make little difference at this point, we can interpret these two expressions in terms of trivial tree structures, as shown below.

(14)

$$
\begin{tikzcd}[arrows=dash]
e^i \arrow[d] \\
\phi
\end{tikzcd}
$$

$$
\begin{tikzcd}[arrows=dash]
e^i \arrow[d] & + & e^j \arrow[d] & = & e^{[i,j]} \arrow[d] \\
\phi & & \phi & & \phi
\end{tikzcd}
$$

Now let's see how adjacent states can house propositions that change values. This is done with the application of a program, π, which is defined as a mapping from states to states, i.e., $[\![\pi]\!] \subseteq S \times S$ (Harel et al, 2000). Programs, like propositions, can be atomic or complex. They have the following behavior:

(15) a. They can be ordered, $\alpha;\beta$ (α is followed by β);
b. They can be iterated, a^* (apply a zero or more times);
c. They can be disjoined, $\alpha \cup \beta$ (apply either α or β);
d. They can be turned into formulas: $[\alpha]\phi$ (after every execution of α, ϕ is true);
$\langle\alpha\rangle\phi$ (there is an execution of α, such that ϕ is true);
e. Formulas can become programs: $\phi?$ (test to see if ϕ is true, and proceed if so).

Given these operations, Pustejovsky and Moszkowicz (2011) then proceed to model basic event configurations in terms of frame structures. For example, a *simple transition* can be defined in terms of two component elements: (a) a sequence of frames containing a propositional opposition over adjacent states; and (b), a representation of the program, α, which brings about the change from the first frame to the adjacent one. The state transition is shown below.

(16) $\boxed{\phi}^{i}_{e_1}\boxed{\neg\phi}^{j}_{e_2}$

A simple transition includes an atomic program, α, that changes the content of a state in the next adjacent state.

(17) $\boxed{\phi}^{i}_{e_1} \xrightarrow{\alpha} \boxed{\neg\phi}^{i+1}_{e_2}$

Because the frame representation becomes somewhat cumbersome with more complex events, we will modify the classic event structure with state-to-state labels, indicating the program being applied. We call this a dynamic event structure (DES). This is shown below.

(18)

$$
\begin{tikzcd}
& e^{[i,i+1]} \arrow[dl, dash] \arrow[dr, dash] & \\
e_1^i \arrow[rr, "\alpha"] \arrow[d, dash] & & e_2^{i+1} \arrow[d, dash] \\
\phi & & \neg\phi
\end{tikzcd}
$$

Concatenation can, of course, apply independently of the introduction of a program. Consider the sentence in (19).

(19) Mary awoke from a long sleep.

The state of being asleep has a duration, $[i,j]$, who's valuation is gated by the waking event at the "next state", $j+1$.

(20)

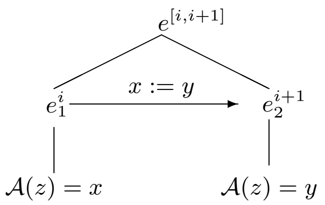
$$
\begin{tikzcd}
& e^{[i,j+1]} \arrow[dl, dash] \arrow[dr, dash] & \\
e_1^{[i,j]} \arrow[rr, "\alpha"] \arrow[d, dash] & & e_2^{j+1} \arrow[d, dash] \\
\phi & & \neg\phi
\end{tikzcd}
$$

Now consider what is needed to model change to an object; that is, not just propositional change, but predicative change. Pustejovsky and Moszkowicz (2011) capture the change in an object attribute that an object with the addition of assignment functions associated with each state at a given time, in order to keep track of the values bound to variables in the expressions being interpreted. Assume an atomic program, *variable assignment*, which associates a specific value to a variable. This requires that we extend the model to pairs of assignment functions (or valuations) (u, v), in addition to temporal index pairs, (i, j). That is, every program, a, in our language, $a \in \pi$, is evaluated with respect to a pair of states, and with each state there is an assignment function. Hence, in order to evaluate a program, a pair of assignment functions is required.

(21) $x := y$ (ν-transition)
"x assumes the value given to y in the next state."
$\langle \mathcal{M}, (i, i+1), (u, u[x/u(y)]) \rangle \models x := y$ iff $\langle \mathcal{M}, i, u \rangle \models s_1 \wedge \langle \mathcal{M}, i+1, u[x/u(y)] \rangle \models x = y$

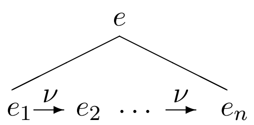
We define the dynamic event structure for this transition in (22), where the attribute, $\mathcal{A}$, of an object, z, changes its value from x to y, i.e., $x \mapsto y$.

(22)



With a ν-transition defined, a *process* can be viewed as simply an iteration of basic variable assignments and re-assignments,

(23)



However, motion verbs (and most processes denoting change) are not simple unguarded ν-transitions, but involve a kind of directionality (directedness). Within a dynamic framework, this is accomplished with a pre-test to ensure distinctness; e.g., the object really did change to a new location.

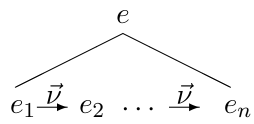
(24) $\overset{x \neq y?}{\boxed{loc(z) = x}_{e_1}} \xrightarrow{\nu} \boxed{loc(z) = y}_{e_2}$

When this test references the ordinal values on a scale, $\mathcal{C}$, this becomes a *directed ν-transition ($\vec{\nu}$)*, e.g., $x \preccurlyeq y$, $x \succcurlyeq y$.

(25) $\vec{\nu} =_{df} \overset{\mathcal{C}?}{e_i} \xrightarrow{\nu} e_{i+1}$

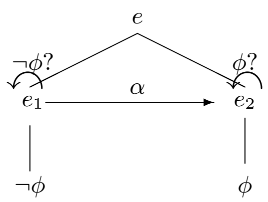
This is what allows us to now dynamically model "directed manner of motion verbs", such as *swim*, *crawl*, and *walk*. That is, they denote processes consisting of multiple iterations of $\vec{\nu}$-transitions, as illustrated in (26).

(26)



It should be clear from the present discussion that achievements are also a species of transition. They require, however, an additional test to ensure that the changed state is not altered after it is achieved. This is accomplished in terms of a pair of tests, as illustrated in (27).

(27)

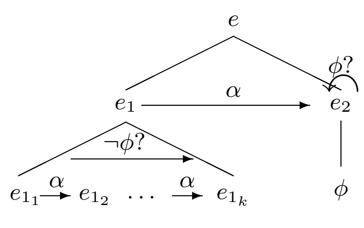


The final event class to model dynamically is that of accomplishment, such as the verbs *build*, *destroy*, and *walk to*.

(28) a. John built a table.
b. Mary walked to the store.

As discussed in Pustejovsky and Moszkowicz (2011), we can think of two parallel changes taking place in such events: there is an internal change (the Agentive activity of a building event, or the movement of the object); but there is also an external change, indicating that a predicate opposition has been satisfied (there is a table built, Mary is at the store). The DITL frame structure for such an event is given below in Figure 1.
This has an elegant treatment in first-order dynamic logic, as shown in the dynamic event structure in (29).

(29)



These and other change predicates receive a fuller treatment in Pustejovsky and Jezek (forthcoming), where a dynamic model of selection is developed.

$build(x,z,y)$	$build(x,z,y)^+$	$build(x,z,y), y = v$	
$\neg table(v)$		$table(v)$	$\langle i,j \rangle$

Table 1: Accomplishment: parallel tracks of changes

4 Habitat Theory

We have focused on the development of a dynamic treatment of predication, framed within discrete event structures, as part of a larger program of research aimed at creating cognitively plausible interpretations of linguistic utterances. There is a growing community of researchers interested in “simulation semantics” (Langacker, 1987, Lakoff, 2009, Evans, 2009, Bergen, 2012), yet the philosophical foundations for this view originate in the 1980s, with Goldman (1989) and Gordon (1986), as an alternative to the “Theory-theory of mind”. The intellectual connections to these various themes are explored elsewhere (Pustejovsky, forthcoming), and we concentrate here on a brief summary of how simulations are constructed from dynamic event structures.

We define an *event simulation* to be a minimal model generated in the context of a temporal trace, from linguistic input, under a number of agent-oriented cognitive constraints. These include an epistemic condition on the individual agent, imposing an evidential *point of view* (POV). The event is situated in the context through an *event localization* procedure, which is facilitated by the construction of *habitats* for the event and its participants.

We start with some general assumptions regarding entity semantics from GL, namely concerning the general structure of objects:

(30) a. Atomic Structure: Formal Quale (objects expressed as basic nominal types)
b. Subatomic Structure: Constitutive Quale (mereotopological structure of objects)
c. Event Structure: Telic and Agentive Qualia structure (origin and functions associated with an object)
d. Macro Object Structure: how objects fit together in space and activity

Now, consider how we contextualize objects through the qualia structure associated with linguistic expressions. For example, a food item has Telic value of *eat*, and an instrument for writing, of *write*, and so forth. Similarly, the artifactual object denoted by the noun *chair* carries a Telic value of "sit in", represented as: $chair : phys \otimes_T sit\_in$. As mentioned previously, this type can be seen as a shorthand for the feature structure representation below:

(31) $$\lambda x \exists y \begin{bmatrix} \textbf{chair} \\ \text{AS} = \begin{bmatrix} \text{ARG1} = x : e \end{bmatrix} \\ \text{QS} = \begin{bmatrix} \text{F} = phys(x) \\ \text{T} = \lambda z, e[sit\_in(e,z,x)] \end{bmatrix} \end{bmatrix}$$

While convention has allowed us to interpret the entire Telic expression as modal, this is inadequate for capturing the deeper meaning of functionality, and this brings in the role of the local modality.

An artifact is designed for a specific purpose, its Telic role; that much is clear. But this purpose can only be achieved under specific circumstances. Let us say that, for an artifact, x, given the appropriate context $\mathcal{C}$, performing the action π will result in the intended or desired resulting state, $\mathcal{R}$. This can be stated dynamically as follows, using the dynamic event structure from above (cf. Pustejovsky, 2012).

(32) $\mathcal{C} \rightarrow [\pi]\mathcal{R}$

This says that, if a context $\mathcal{C}$ (a set of contextual factors) is satisfied, then every time the activity of π is performed, the resulting state $\mathcal{R}$ will occur. The precondition context $\mathcal{C}$ is necessary to specify, since this enables the local modality to be satisfied.

Consider how this works with a classic example in lexical semantics, that of the domain "food". For a noun such as *sandwich*, we have a set of contexts, $\mathcal{C}$, under which, for the object denoted by x, when an individual y eats x, there is a resulting state of nourishment, which we will notate as $\mathcal{R}_{eat}$. Hence, we have the following qualia structure representation, using the dynamic event structures.

(33) $\lambda x[\text{FORMAL}(x) = phys(x) \wedge$

$$\text{TELIC}(x) = \lambda y \lambda e[\mathcal{C} \rightarrow [eat(e, y, x)]\mathcal{R}_{eat}(x)]]$$

This says that if the context is satisfied, then every eating of that substance will result in a "nourishing." In other words, more correctly stated, sandwiches are not "for eating", but rather "for nourishing by eating."

Now let us extend this intuition to introduce the notion of a habitat. Informally, a habitat is representation of an object situated within a partial minimal model; it is a directed enhancement of the qualia structure. Multi-dimensional affordances determine how habitats are deployed and how they modify or augment the context, and compositional operations include procedural (simulation) and operational (selection, specification, refinement) knowledge. As an example, consider the dynamic qualia structure for an artifact such as *table* or *chair* (shown below).

$$\lambda x \begin{bmatrix} \textbf{chair} & \\ \text{AS} = \begin{bmatrix} \text{ARG1} = x : e \end{bmatrix} & \\ \text{QS} = \begin{bmatrix} \text{F} = phys(x) \\ \text{T} = \lambda z \lambda e[\mathcal{C} \rightarrow [sit(e, z, x)]\mathcal{R}_{sit}(x)] \\ \text{A} = \exists w \exists e'[make(e', w, x)] \end{bmatrix} & \end{bmatrix}$$

We construct the habitat for an object by contextualizing it. For example, in order to use a table, the top has to be oriented upward, the surface must be accessible, and so on. A chair must also be oriented up, the seat must be free and accessible, it must be able to support the user, etc. The habitat also includes an *embedding space* and supporting objects. An illustration of what the resulting knowledge structure for the habitat of a chair is shown below.

$$\lambda x \begin{bmatrix} \textbf{chair}_{hab} \\ \text{F} = [phys(x), on(x, y_1), in(x, y_2), orient(x, up)] \\ \text{C} = [seat(x_1), back(x_2), legs(x_3), clear(x_1)] \\ \text{T} = \lambda z \lambda e[\mathcal{C} \rightarrow [sit(e, z, x)]\mathcal{R}_{sit}(x)] \\ \text{A} = [made(e', w, x)] \end{bmatrix}$$

Event simulations are constructed from the composition of object habitats, along with particular constraints imposed by the dynamic event structure inherent in the verb itself. To best illustrate this, consider the following short discourse.

(34) a. A car entered the driveway.
b. A woman stepped out.

First minimal models are constructed from the dynamic event structure for each predicate. This proceeds informally as follows:

(35) Given an event, E: a. Compute the affordance space for each argument, a_i, to E;
b. Compute the object habitat for each a_i;
c. Compute the Event Localization on E. This is the minimal embedding for E;
d. Compute the event habitat for E.

The habitat composition resulting from these two events introduces a number of additional states, processes, and conditions, including a bridging event, statable as a precondition on the second event; namely, that the car was not moving when the woman stepped out of it. The composition creates this presupposition (defeasible as it is), and it is introduced into the event simulation as part of the model.

5 Conclusion

In this brief note, I have illustrated only some of the mechanisms involved in habitat and event simulation construction. A greater understanding of how event participants contribute towards the construction of affordance spaces for events is necessary to better articulate this process. It is clear, however, that a dynamic interpretation of the event structure and qualia structure from GL is an important aspect of modeling linguistic expressions as cognitive simulations.

Acknowledgments

I would like to thank David McDonald, Elisabetta Jezek, and Olga Batuikova, for comments and discussion of this work.

References

Bach, Emmon. 1986. The algebra of events. Linguistics and Philosophy 9:5–16.

Barker, Chris. 1998. Partitives, double genetives, and anti-uniqueness. Natural Language and Linguistic Theory 16:679–717.

Barsalou, Lawrence W. "Grounded cognition." Annu. Rev. Psychol. 59 (2008): 617-645.

Barsalou, Lawrence W. "Grounded cognition." Annu. Rev. Psychol. 59 (2008): 617-645.

Bergen, Benjamin. (2012). Louder than words: The new science of how the mind makes meaning. New York: Basic Books.

Bouillon, P. (1997). Polymorphie et semantique lexicale : le cas des adjectifs, Lille: Presses Universitaires du Spetentrion.

Carlson, Gregory. 1977. Amount relatives. Language 53:520–542.

Cresswell, M.J. 1977. The semantics of degree. In Montague grammar, ed. Barbara Partee, 261–292. New York: Academic Press.

Doetjes, Jenny. 1997. Quantifiers and selection. Doctoral Dissertation, Rijksuniversiteit Leiden.

Evans, Vyvyan. How words mean. Oxford University Press, 2009.

Fernando, Tim 2004. "A Finite-state Approach to Events in NL Semantics". *JLC* 14(1): 79-92.

Gillon, Brendan. 1992. Towards a common semantics for english count and mass nouns. Linguistics and Philosophy 15:597–639.

Goldman, A. I. (1989). Interpretation psychologized. Mind and Language 4: 161-185.

Goldman, A. I. (2006). Simulating Minds: The Philosophy, Psychology, and Neuroscience of Mindreading. New York: Oxford University Press.

Gordon, R. M. (1986). Folk psychology as simulation. Mind and Language 1: 158-171.

Gordon, R. M. (1996). Radical simulationism. In P. Carruthers and P. Smith, eds., Theories of Theories of Mind. Cambridge: Cambridge University Press.

Grosu, Alexander, and Fred Landman. 1998. Strange relatives of the third kind. Natural Language Semantics 6:125–170.

Harel, David, Jerzy Tiuryn, and Dexter Kozen. Dynamic logic. MIT press, 2000.

Hay, Jen, Christopher Kennedy, and Beth Levin. 1999. Scale structure underlies telicity in 'degree achievements'. In Proceedings of SALT IX, ed. Tanya Matthews and Devon Strolovitch, 127–144. Ithaca, NY: CLC Publications.

Heim, Irene. 1987. Where does the definiteness restriction apply? Evidence from the definiteness of variables. In The representation of (in)definiteness, ed. Eric Reuland and Alice ter Meulen, chapter2, 21–42. Cambridge, Mass.: MIT Press.

deHoop, Helen. 1997. A semantic reanalysis of the partitive constraint. Lingua 103:151–174.

Kennedy, Christopher. 2001. Polar opposition and the ontology of 'degrees'. Linguistics & Philosophy 24:33–70.

Klein, Ewan. 1991. Comparatives. chapter32, 673–691. Berlin: de Gruyter.

Krifka, Manfred. 1989. Nominal reference, temporal constitution and quantification in event semantics. In Semantics and contextual expression, ed. Renate Bartsch, Johann van Benthem, and Peter van Emde-Boas, 75–115. Stanford, CA: CSLI Publications.

Krifka, Manfred. 1998. Scope-inversion under the rise-fall contour in german. Linguistic Inquiry 29:75–112.

Kröger, Fred and Stephan Merz 2008. Temporal Logic and State Systems, Springer Verlag.

Ladusaw, WilliamA. 1982. Semantic constraints on the English partitive construction. In Proceedings of WCCFL 1.

Lakoff, George, and Mark Johnson. Philosophy in the flesh: The embodied mind and its challenge to western thought. Basic books, 1999.

Langacker, Ronald W. Foundations of Cognitive Grammar: theoretical prerequisites. Volume 1. Vol. 1. Stanford university press, 1987.

Lehrer, Adrienne. 1986. English classifier constructions. Lingua 68:109–148.

Link, Godehard. 1983. The logical analysis of plurals and mass terms: A lattice theoretical approach. In Meaning, use, and the interpretation of language, ed. R.Bäuerle, C.Schwarze, and A.von Stechow, 302–323. Berlin: de Gruyter.

Naumann, Ralf, 2001. Aspects of changes: a dynamic event semantics, Journal of Semantics 18:27-81.

Parsons, Terrence. 1970. An analysis of mass terms and amount terms. Foundations of Language 6:362–388.

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

Pustejovsky, J. 2001. Type Construction and the Logic of Concepts. In P. Bouillon and F. Busa (eds.), The Syntax of Word Meaning, Cambridge University Press, Cambridge.

Pustejovsky, J. 2006. Type theory and lexical decomposition. Journal of Cognitive Science 6:39-76.

Pustejovsky, J. 2011. Coercion in a general theory of argument selection, Linguistics, vol. 49, no 6. de Gruyter.

Pustejovsky, James. "The Semantics of Functional Spaces." Practical Theories and Empirical Practice: Facets of a Complex Interaction. Ed. Andrea Schalley. John Benjamins Publishing Company, 2012.

Pustejovsky, J. forthcoming. "Event Simulations as Semantic Models".

Pustejovsky, James and Elisabetta Jezek 2011. Scale-shifting and Compo- sitionality, Abstract presented at Workshop on Scalarity in Verb-Based Constructions, Heinrich-Heine-Universit?at Du?sseldorf, April 7-8, 2011.

Pustejovsky, James and Elisabetta Jezek. forthcoming. "Verbal Patterns of Change".

Pustejovsky, James, and Jessica L. Moszkowicz. "The qualitative spatial dynamics of motion in language." Spatial Cognition and Computation 11.1 (2011): 15-44.

Schwarzschild, Roger. 2002. The grammar of measurement. In Proceedings of SALT XII, ed. Brendan Jackson. Ithaca, NY: CLC Publications.

Schwarzschild, Roger, and Karina Wilkinson. 2002. Quantifiers in comparatives: A semantics of degree based on intervals. Natural Language Semantics 10:1–41.

Winter, Yoad. 2001. Measure phrase modification in a vector space semantics. In Proceedings of WCCFL 20, ed. K.Megerdoomina and L.A. Bar-el, 607–620. Somerville, MA: Cascadilla Press.

Zwarts, Joost. 2000. Vectors across Spatial domains. Paper presented at the Worksho on Axes and Vectors in Language and Space, Lincoln University, July 7, 2000. To appear in Emile van der Zee and John Slack, eds., Axes and Vectors in Language and Space. Oxford: Oxford University Press.

Zwarts, Joost. 1997. Vectors as relative positions: A compositional semantics of modified pps. Journal of Semantics 14:57–86.

Zwarts, Joost, and Yoad Winter. 1997. A semantic characterization of locative PPs. In Semantics and Linguistic Theory 7, ed. Aaron Lawson, 294–311. Ithaca, NY: CLC Publications.