# Putting Frames in Perspective

**Nancy Chang, Srini Narayanan, and Miriam R.L. Petruck**
International Computer Science Institute
1947 Center St., Suite 600, Berkeley, CA 94704
{nchang,snarayan,miriamp}@icsi.berkeley.edu

## Abstract

This paper attempts to bridge the gap between FrameNet frames and inference. We describe a computational formalism that captures structural relationships among participants in a dynamic scenario. This representation is used to describe the internal structure of FrameNet frames in terms of parameters for event simulations. We apply our formalism to the commerce domain and show how it provides a flexible means of accounting for linguistic perspective and other inferential effects.

## 1 Introduction

FrameNet (Fillmore et al., 2001) is an online lexical resource[1] designed according to the principles of frame semantics (Fillmore, 1985; Petruck, 1996). It thus takes as foundational the assumptions that (1) lexical items draw on rich conceptual structures, or **frames**, for their meaning and function; and (2) conceptually related lexical items may foreground different aspects of the same background frame. Verbs involved with commercial events serve as canonical examples:

(1) a. Chuck bought a car from Jerry for $1000.
b. Jerry sold a car to Chuck for $1000.
c. Chuck paid Jerry $1000 for a car.
d. Jerry charged Chuck $1000 for a car.
e. Chuck spent $1000 on a car.

The sentences in (1) might describe the same interaction – in which one individual (Chuck) transfers money ($1000) to another (Jerry) in exchange for some goods (a car) – but differ in the **perspective** they impose on the scene.

The shared inferential structure of verbs like *buy* and *sell* is captured in FrameNet by the COMMERCE frame, which is associated with a set of situational roles, or **frame elements** (FEs), corresponding to event participants and props. These FEs are used to annotate sentences like those in (1), yielding:

(2) a. [Chuck]$^{\text{Buyer}}$ **bought** [a car]$^{\text{Goods}}$
[from Jerry]$^{\text{Seller}}$ [for $1000]$^{\text{Payment}}$.
b. [Jerry]$^{\text{Seller}}$ **sold** [a car]$^{\text{Goods}}$
[to Chuck]$^{\text{Buyer}}$ [for $1000]$^{\text{Payment}}$.

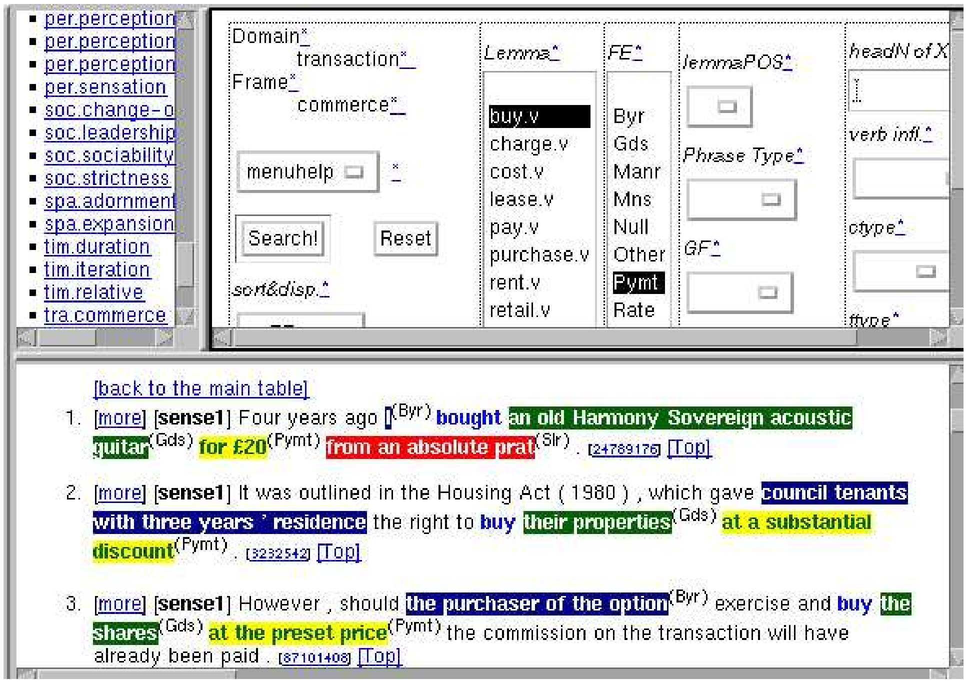
FE tags act as a shorthand that allows diverse verbs to tap into a common subset of encyclopedic knowledge. Moreover, regularities in the set of FEs realized with specific lexical items can be taken as correlated with their favored perspective.

A significant gap remains, however, between the unstructured and intuitively chosen tag sets used in FrameNet and a formal characterization of the interrelated actions and relations holding among them. An explicit representation of such frame-semantic information is needed to fully realize FrameNet's potential use in text understanding and inference (Fillmore and Baker, 2001). In this paper we attempt to bridge the gap by defining a formalism that unpacks the shorthand of frames into structured event representations. These dynamic representations allow annotated FrameNet data to parameterize event simulations (Narayanan, 1999b) that produce fine-grained, context-sensitive inferences. We illustrate our formalism for the COMMERCE frame and show how it can account for some of the wide-ranging consequences of perspective-taking.

## 2 The FrameNet COMMERCE frame

The FrameNet project has thus far produced two databases: a collection of approximately 80 frames with frame descriptions, chosen to cover a broad range of semantic domains; and a hand-annotated dataset of about 50,000 sentences from the British National Corpus (Baker et al., 1998). The databases document both syntactic and semantic behavior of a

[1] http://www.icsi.berkeley.edu/framenet/



[back to the main table]

1. [more] **[sense1]** Four years ago I(Byr) **bought** **an old Harmony Sovereign acoustic guitar**(Gds) **for £20**(Pymt) **from an absolute prat**(Slr) . [24789176] [Top]

2. [more] **[sense1]** It was outlined in the Housing Act ( 1980 ) , which gave **council tenants with three years ' residence** the right to **buy** **their properties**(Gds) **at a substantial discount**(Pymt) . [3232542] [Top]

3. [more] **[sense1]** However , should **the purchaser of the option**(Byr) exercise and **buy** **the shares**(Gds) **at the preset price**(Pymt) the commission on the transaction will have already been paid . [87101408] [Top]

Figure 1: Results of a query on the FrameNet COMMERCE frame, showing annotated data for the verb *buy*.

wide variety of lexical items (or **lemmas**) and thus have the potential to allow corpus-based techniques to be applied to semantically oriented tasks.[2]

The current release of the FrameNet databases[3] defines a COMMERCE frame with frame elements including the familiar Buyer, Seller, Payment and Goods, along with several other FEs needed to cover the data. The frame includes 10 verbs relevant to commercial transactions, for a total of 575 annotated sentences. Figure 1 shows a sampling of data annotated with respect to the COMMERCE frame.

Considerable research has been devoted to explicating the connections among frames, perspective, and argument structure; see Gawron (ms.) and Hudson (2002). But there has been relatively less work that addresses inferential issues related to perspective. The COMMERCE frame, for example, is implicitly associated with a complex, dynamic network of interrelated events, actions and participants. Our proposal is that perspectival effects may be best understood in terms of subtle inferential effects on interpretation licensed by this network.

Our task, then, is to make this inferential structure explicit. We take the original COMMERCE frame as our starting point and define the interrelationships present among its FEs. The additional structure we impose on the COMMERCE frame allows us to distinguish a *perspective-neutral* description of a commercial transaction from the *perspectivized* situations described by particular verbs. The resulting event representation can be integrated with a simulation-based inference engine to account for differences in the interpretation of sentences like those in the annotated FrameNet data.

## 3 Structured event representations

In this section, we present a formal specification used for mapping the flat set of FEs in COMMERCE onto explicitly structured event representations based on the Embodied Construction Grammar (ECG) formalism. ECG is a constraint-based formalism similar in many respects to other unification-based linguistic formalisms, such as HPSG (Pollard and Sag, 1994).[4] It differs from other lingustically motivated proposals in that it is

[2] See (Gildea and Jurafsky, 2000) for some promising initial work in applying statistical techniques to the FrameNet database to automatically label frame elements.

[3] We refer to data from FrameNet I; an interim release of FrameNet II is expected soon.

[4] ECG includes formalisms for both **schemas** (conceptual representations) and **constructions** (conventionalized pairings of form and meaning), described in (Bergen and Chang, 2002). We refer here only to the schema formalism in a simplified form. See (Chang et al., 2002) for a more complete version that has been extended to accommodate additional cognitive linguistic primitives.

designed to support a model of language understanding in which utterances evoke a complex network of conceptual **schemas** that are then mentally simulated in context to produce a rich set of inferences. It is thus ideally suited for our current goal of translating frames to conceptual representations.

Figure 2 presents the ECG schema definition language. The indented block labeled **roles** lists and constrains the schema's local **roles**, which are equivalent to features (or in this case, frame FEs). Roles are declared with a local name (*local-role*) and may be accompanied by type restrictions (indicated with ':'). **Identification** (or binding) constraints (indicated with '⟷') may appear in either the **roles** or the **constraints** block; these cause roles and constraints to be shared between its arguments, similar to unification or coindexation.[5] The **subcase** relation defines a schema inheritance lattice, with the local schema inheriting all roles and constraints.

```
schema name
[subcase of schema
[evokes
    { schema as local-name }
 roles
    ⎧ local-role                          ⎫
    ⎨ local-role : restriction            ⎬
    ⎪ local-role  ⟷ role                  ⎪
    ⎩ local-role  ⟷ role : restriction    ⎭
[constraints
    ⎧ role  ⟷ role        ⎫
    ⎩ phase :: condition  ⎭
```

Figure 2: Schema definition formalism. Keywords are shown in **bold**; a left square bracket ([) marks optional blocks; and curly braces ({}) enclose a set of optional statements. See text for details.

The formalism also has several novel features that we will exploit in representing commercial transactions. The most important of these are: (1) the ability to flexibly evoke and relate multiple schemas, due mainly to the **evokes** relation; and (2) the ability to assert dynamic conditions that apply to specific event stages, through the use of **simulation constraints**. We will describe each of these briefly, deferring details to the example schemas below.

Schemas listed in the **evokes** block are instantiated locally (as *local-name*), but the relationship between the defined schema and the evoked schema is underspecified. This underspecification allows one schema to be defined in terms of another schema without implying either full inheritance of the evoked schema's roles or containment in either direction. In some cases, the evoked schema corresponds to a subpart of the evoking schema; alternatively, the evoked schema may serve as a background schema against which the evoking schema is defined. We will see examples of each below.

Simulation constraints use the '::' notation to assert some *condition* on a particular *phase* of simulation – either a relation that must hold or an event or action that must take place during that phase. Simulation phases correspond to event stages; these constraints serve as the bridging connection to previous work on modeling event structure and linguistic aspect using active representations (Narayanan, 1997; Chang et al., 1998).

We now show how the ECG formalism can be used to define more complex schemas that provide the underlying structure we need to tackle the COMMERCE frame; the key schemas for the current discussion are shown in Figure 3.[6]

The Event schema is of primary importance: it appears directly or indirectly in the rest of the schema definitions, and it serves as the crucial link to simulation. The definition given here is not intended to capture the full complexity of the most generalized event, which may have complex internal structure (start and finish subevents, ongoing period, etc.). At a coarser granularity, however, it may also be viewed as a discrete temporal chunk that takes place between two time slices. The schema as shown reflects this coarser view, which is sufficient for current purposes: its roles include before, after, and transition, all referring to simulation phases. Another role, the nucleus, is constrained only to hold or take place during the transition phase. Together these roles anchor the event to the passage of time.

The other schemas are more complex. The Transfer schema corresponds to an event in which an agent causes a theme to be transferred from the source to the recipient. It is defined as evoking two other schemas: an Action schema (with an actor role) and a Receive schema (in which a receiver comes into possession of the received entity). (These are not shown, nor is the causal relation between them.) Note that both act and rec are conceptually distinct from the

[5] Constraints may refer to locally declared roles, inherited roles, and evoked schemas, as well as any roles available through these structures. Standard slot-chain notation is used to refer to role y of a structure x as x.y.

[6] Some schema definitions have been omitted or simplified to conserve space; relevant details are mentioned in the text.

```
schema Event
  roles
    before : Phase
    transition : Phase
    after : Phase
    nucleus
  constraints
    transition :: nucleus
```

```
schema Transfer
  subcase of Event
  evokes
    Action as act
    Receive as rec
  roles
    agent ⟷ act.actor
    source : Entity
    theme ⟷ rec.received
    recipient ⟷ rec.receiver
  constraints
    transition :: act
    transition :: rec
    after :: has(recipient,theme)
```

```
schema Exchange
  subcase of Event
  roles
    participant1 : Human
    participant2 : Human
    entity1 : Entity
    entity2 : Entity
    transfer1 : Transfer
    transfer2 : Transfer
    agent : Entity
  constraints
    transition :: transfer1
    transition :: transfer2
    transfer1.source ⟷ participant1
    transfer1.theme ⟷ entity1
    transfer1.recipient ⟷ participant2
    transfer2.source ⟷ participant2
    transfer2.theme ⟷ entity2
    transfer2.recipient ⟷ participant1
```

Figure 3: The Event, Transfer and Exchange schemas.

nucleus role inherited from Event, although all are constrained to take place during the event's transition phase. The agent role is constrained to be the same entity as the actor of act. Importantly, the Transfer event schema makes no commitment as to whether its agent – the entity seen as causing the overall event – is the source, recipient or even theme. It is in this respect that the Transfer schema can be considered neutral in perspective.

The Exchange schema is structurally similar to the Transfer schema and provides most of the relevant constraints needed for commercial transactions. It includes two transfer events that occur during the transition phase and are parameterized straightforwardly in the constraints block by two human participants and two entities. An additional agent role is not bound to any particular entity; this schema is thus also perspective-neutral, since either participant (or both) might be viewed as active.

## 4 Commercial transaction schemas

We are now in a position to return to the commerce domain and put our inventory of domain-general schemas to use. We first define the Commercial-Transaction (CT) schema as a subcase of the Exchange schema with appropriate role identifications and an additional type restriction on entity1. The role names in this schema differ slightly from those in FrameNet's COMMERCE, reflecting its perspective-neutral status. But given the obvious mapping to the FrameNet FEs, the CT schema fulfills part of our original objective: based on its inherited and evoked schemas and constraints, it concisely and precisely states the conceptual underpinnings of the basic commercial transaction.

```
schema Commercial-Transaction
  subcase of Exchange
  roles
    customer ⟷ participant1
    vendor ⟷ participant2
    money ⟷ entity1 : Money
    goods ⟷ entity2
    goods-transfer ⟷ transfer1
    money-transfer ⟷ transfer2
```

Figure 4: The Commercial-Transaction schema.

The CT schema provides the underlying infrastructure against which various perspectivized schemas can be defined. As shown in Figure 5, we treat Buy, Sell and Pay as schemas that **evoke** the CT schema and identify their roles with specific participants and event stages of the evoked CT schema. Note the use of the keyword **self** (which we treat as a special kind of role) to refer to the schema being defined: Buy and Sell schemas each identify **self** with the ct.nucleus role (that is, the nucleus of its evoked commercial transaction), and is thus constrained to take place during the evoked CT's transition phase. In contrast, since Pay identifies itself with ct.money-transfer.nucleus, it refers specifically to a subpart of the overall commercial transaction, such that its execution does not necessarily entail the execution of the goods-transfer in the event (i.e., you don't always get what you pay for).

The three schemas also differ in their participant role bindings: all are defined as subcases of

```
schema Buy
  subcase of Transitive-Action
  evokes Commercial-Transaction as ct
  roles
    self ⟷ ct.nucleus
    buyer ⟷ actor ⟷ ct.agent ⟷ ct.customer
    goods ⟷ undergoer ⟷ ct.goods
```

```
schema Sell
  subcase of Transitive-Action
  evokes Commercial-Transaction as ct
  roles
    self ⟷ ct.nucleus
    seller ⟷ actor ⟷ ct.agent ⟷ ct.vendor
    goods ⟷ undergoer ⟷ ct.goods
```

```
schema Pay
  subcase of Transitive-Action
  evokes Commercial-Transaction as ct
  roles
    self ⟷ ct.money-transfer.nucleus
    payer ⟷ actor ⟷ ct.customer
                ⟷ ct.money-transfer.agent
    payment ⟷ ct.money
    payee ⟷ ct.vendor
```

Figure 5: The Buy, Sell and Pay schemas.

Transitive-Action (not shown), which corresponds to a prototypical situation in which an actor entity affects or manipulates an undergoer entity. The Buy and Sell schemas both identify the undergoer with ct.goods, and the actor with ct.agent. But the two schemas impose different views on the same situation by virtue of a single additional constraint on this latter role (which corresponds to the active participant in the overall CT), binding it to either the ct.customer (Buy) or the ct.vendor (Sell). The bindings in the Pay schema assert that its actor is the ct.customer, as well as the agent of the money-transfer.

Other schemas associated with the CT schema lend themselves to similar analyses, though they draw on additional schemas not defined here. For example, the Spend schema evokes a schema for resource consumption (as in (Hudson, 2002)); Charge involves the vendor's communication of the price to the customer as a prerequisite to the overall exchange of goods and money. In general, the CT schema explicitly specifies the internal event structure of a commercial transaction but remains noncommittal about which of its participants is seen as active. This flexibility in representation allows other schemas to effect the bindings that make appropriate commitments on an individual basis.

## 5 Simulation semantics

The structured event formalism we have described allows us to translate FrameNet descriptions into a representation suitable for simulative inference. Central to the representation is an event model called **executing schemas** (or **x-schemas**), motivated by research in both sensorimotor control and cognitive semantics (Narayanan, 1997). X-schemas are active structures that cleanly capture sequentiality, concurrency and event-based asynchronous control. They thus provide a cognitively motivated basis for modeling diverse linguistic phenomena, including aspectual inference (Chang et al., 1998), metaphoric inference (Narayanan, 1999a) and event-based reasoning in narrative understanding (Narayanan, 1999b). In this paper, we focus on the problem of *frame-based inference* and the attendent problem of modeling *perspectival effects*.

The event model is based on the Petri net, which in its basic form is a weighted, bipartite graph consisting of *places* (shown as circles) and *transitions* (shown as rectangles) connected by directed input and output arcs (Murata, 1989; Narayanan, 1997). Places may contain *tokens* (i.e., they may be *marked*), and they typically represent states, resources or conditions that apply. Transitions typically represent actions or events. X-schemas extend the basic Petri net to include typed arcs, hierarchical control, durative transitions, parameterization, typed (individual) tokens and stochasticity.

The most relevant property of the x-schema for this paper is its well-specified execution semantics: a transition is *enabled* when all its input places are marked, such that it can *fire* by moving tokens from input to output places. The active execution semantics serves as the engine of context-sensitive inference in the simulation-based model of language understanding mentioned earlier.

The ECG formalism is designed to allow constraints on x-schema simulation to be expressed. In particular, the Event schema in Figure 3 has roles that refer to event phases; these correspond to x-schema places and transitions. Other schema roles specify x-schema parameters, which allow x-schemas to give rise to different execution traces through the network with different parameters.

The Commercial-Transaction schema has been implemented in the KarmaSIM x-schema simulation environment (Narayanan, 1997); Figure 6 shows part of the network. The phase roles from the schemas in Section 3 have been mapped onto the

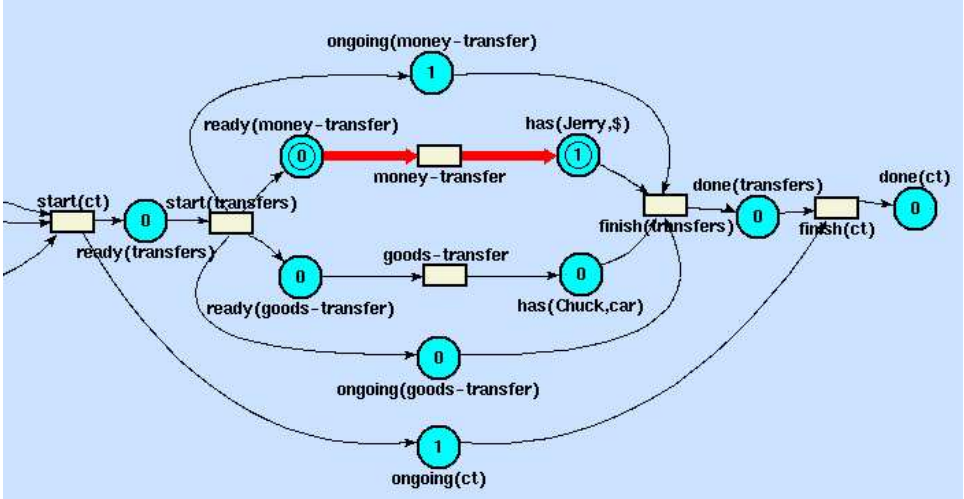


Figure 6: KarmaSIM simulation of the Commercial-Transaction schema. The highlighted execution is associated with the Pay schema, corresponding to the money-transfer event.

fine-grained temporal structure of each event, corresponding to the various control nodes in the network (ready, ongoing, finish, done, etc.); the transition phase referenced in the schemas is expanded as the start, ongoing and finish nodes. As shown, execution of the overall CT schema comprises the execution of two subsidiary events, the goods-transfer and the money-transfer. These need not be synchronized, but both must complete for the overall commercial transaction to complete (enforced by the arcs from ongoing(money-transfer) and ongoing(goods-transfer) to finish(transfers)). All the frame-based inferences of the CT frame (e.g., the seller (buyer) has the goods (money) until the goods-transfer (money-transfer) is completed, and the seller (buyer) has the money (goods) when the money-transfer (goods-transfer) is completed) come from simulating the CT frame.

In the simulation framework, perspectival effects come in at least three flavors. First, the frame element binding patterns may differ among perspectives, as illustrated by Figure 5, in which the lexical item buy identifies the actor of the transitive-action with both the customer of the CT and the agent of the money-transfer. This issue of binding has been the focus of previous work (see Section 2); our approach is similar to construction-based proposals that explicitly represent the binding constraints for different frame element binding patterns.

Second, some perspectives specify the specific subevents (or collection of subevents) to simulate while others require simulating the entire event frame. An example of this is shown in Figure 6, where the highlighted money-transfer portion of the network corresponds to a simulation of the Pay schema. The token in ongoing(ct) shows that there is an ongoing transaction, but the finish(transfers) transition is not enabled. Technically, the done(ct) place is not *reachable* (absent other information), since the simulation of Pay does not provide direct evidence for the occurrence of a goods-transfer.[7] In contrast, both Buy and Sell involve simulating the entire transaction, include both transfers as well as the done(ct) node. (Thus, the entire network in Figure 6 can be considered an expansion of the CT schema's transition phase.)

A third, more subtle aspect of perspective is related to the problem of linguistic focus. The perspectival difference between Buy and Sell, for instance, is only partially captured by their different FE bindings to the CT frame. Another difference stems from the foregrounding of specific relations: buy foregrounds the interaction between the Buyer and the Goods (including the eventual possession of the Goods), while sell foregrounds the interaction between the Seller and the Goods. Work in progress suggests that many foregrounding cases can be handled by simulating different parts of the event at varying degrees of detail. For example, the simulation for Buy could execute x-schemas in which the Buyer interacts with the Goods – such as the goods-transfer and its resulting possession (abbreviated as has(Chuck, car) in Figure 6) – at the default granu-

[7] Contextual or background knowledge could provide evidence for the other transfer or allow it to be inferred by default.

larity, while other x-schemas are collapsed into less detailed simulations. (See (Narayanan, 1997) for a detailed model of simulation at multiple levels of granularity.) While the model is able to handle some of the issues pertaining to foregrounding and focus, a full account remains a topic of ongoing research.

## 6 Discussion and conclusions

FrameNet shows considerable promise for enabling qualitative breakthroughs on NLP applications requiring increased semantic and pragmatic sophistication, including information extraction, word-sense disambiguation, and question answering. FrameNet frames are intended to capture crucial generalizations not available in other lexical resources. WordNet (Fellebaum, 1998), for example, includes only simple taxonomic relations (*buy* and *sell* are listed as hyponyms of *get* and *give*, respectively, and as antonyms of each other). The PropBank project (Kingsbury and Palmer, 2002) is, like FrameNet, geared toward the creation of a semantically annotated corpus (by adding general logical predicates to the Penn Treebank), though without any common background frame structures across lexical items.

While frames and FE tags are meaningful to human interpreters, they are not yet suitable for use in natural language understanding. In this paper we have shown how FrameNet tags can be precisely defined in terms of structured event representations, which can support parameterized simulations that license active inferences. The formalism appears expressive enough for the COMMERCE frame, and uses methods of simulation semantics to handle frame-based inferences and associated perspectival effects.

We are currently automating the process of mapping frame definitions to simulation parameterizations and extending the representation to cover the entire FrameNet II database.

## Acknowledgments

Thanks to Chuck Fillmore, Jerry Feldman, the FrameNet and NTL groups, and the ScaNaLU workshop participants for early feedback.

## References

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet Project. In *Proc. COLING-ACL*, Montreal, Canada.

Benjamin K. Bergen and Nancy C. Chang. 2002. Embodied Construction Grammar in simulation-based language understanding. Technical Report TR-02-004, International Computer Science Institute.

Nancy Chang, Daniel Gildea, and Srini Narayanan. 1998. A dynamic model of aspectual composition. In *Proc. 20th Cognitive Science Society Conference*, Madison, Wisconsin.

Nancy Chang, Jerome Feldman, Robert Porzel, and Keith Sanders. 2002. Scaling cognitive linguistics: Formalisms for language understanding. In *Proc. 1st International Workshop on Scalable Natural Language Understanding*, Heidelberg, Germany.

Christianne Fellebaum, editor. 1998. *WordNet: An Electronic Database*. MIT Press.

Charles J. Fillmore and Collin F. Baker. 2001. Frame semantics for text understanding. In *Proc. WordNet and Other Lexical Resources Workshop*, Pittsburgh. NAACL.

Charles J. Fillmore, Charles Wooters, and Collin Baker. 2001. Building a large lexical databank which provides deep semantics. In *Proc. Pacific Asian Conference on Language, Information and Computation*, Hong Kong.

Charles J. Fillmore. 1985. Frames and the semantics of understanding. *Quaderni di Semantica*, IV(2).

Jean Mark Gawron. 2002. Frames, modifiers, and argument structure. Ms., San Diego State University.

Daniel Gildea and Daniel Jurafsky. 2000. Automatic labeling of semantic roles. In *Proceedings of the 38th Annual Meeting of the ACL*, Hong Kong, October.

Richard Hudson. 2002. Buying and selling in Word Grammar. In press.

Paul Kingsbury and Martha Palmer. 2002. From Treebank to PropBank. In *Proc. 3rd International Conference on Language Resources and Evaluation (LREC-2002)*.

Tadao Murata. 1989. Petri nets: Properties, analysis, and applications. In *Proc. IEEE-89*, volume 77, pages 541–576.

Srini Narayanan. 1997. *Knowledge-based Action Representations for Metaphor and Aspect (KARMA)*. Ph.D. thesis, University of California at Berkeley.

Srini Narayanan. 1999a. Moving right along: A computational model of metaphoric reasoning about events. In *Proc. Sixteenth National Conference of Artificial Intelligence (AAAI-99)*. AAAI Press, Menlo Park.

Srini Narayanan. 1999b. Reasoning about actions in narrative understanding. In *Proc. Sixteenth International Joint Conference on Artificial Intelligence (IJCAI-99)*. Morgan Kaufmann Press.

Miriam R.L. Petruck. 1996. Frame semantics. In J. Verschueren, J. Östman, J. Blommaert, and C. Bulcaen, editors, *Handbook of Pragmatics*. John Benjamins, Philadelphia.

Carl Pollard and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press, Chicago.