Axiomathes (2022) 32 (Suppl 3):S993–S1039
https://doi.org/10.1007/s10516-022-09649-9

ORIGINAL PAPER

Natural-Language Predicates as Relations of the Relational Model of Data

Olga Poller[1]

Received: 13 October 2022 / Accepted: 20 October 2022 / Published online: 3 December 2022

Abstract
In this paper I review the Neo-Davidsonian semantics of prepositional phrases and secondary predication. I argue that certain types of examples pose challenge to this semantics. I present an alternative to the Neo-Davidsonian analysis which successfully deals with the problematic examples. The core idea lies in representing theta-roles not as functions from events to their participants, but rather as argument-labels encoding the role of each argument in a given verb. As a result, natural-language predicates can now be treated in the manner in which relations are treated in the relational model of data, that is, as naming sets of tuples in which every object is given together with its role named by a corresponding attribute (a theta-role). Such a representation allows the employment of relational algebra operators to calculate the extensions of complex predicates (predicates built out of atomic predicates and/or atomic predicates and prepositions). I lay out the foundations of a relational FOL appropriate for the representation of natural-language predicates and present solutions to the problematic examples. From a practical perspective, the expressions of a relational FOL can be translated to a relational algebra or SQL, which makes it possible to operate with these three languages on the same relational model of data. From a philosophical perspective, a relational FOL permits a return to 'property-based' semantics, one in which properties named by predicates are those of individuals, and not of events.

Keywords Prepositional phrases · Secondary predication · Adjuncts · Theta-roles · The relational model of data · Relational algebra · Modifiers · Neo-Davidsonian semantics · Event semantics · Relational databases · Natural-language predicates

✉ Olga Poller
olga.poller@up.krakow.pl

[1] Pedagogical University of Cracow, Podchorążych 2, 30-084 Kraków, Poland

1 Introduction

Following Frege, a grammatical theory of natural language is considered to have a function-argument structure. It is widely assumed (cf. Chierchia (1984)) that meaning in natural language is organized in a function-argument way, and that the functional dependency between basic and complex semantic entities grounds the notion of compositionality. Therefore, the main task of a formal theory for natural language is to capture and reconstruct the structure of the syntactic and semantic dependencies in natural language. The first step is to establish what should be considered as basic syntactic and semantic entities. There is a widely shared agreement between philosophers and linguists that thematic roles are among such entities (because they '[...] are somehow involved in associating a predicate's meaning with its arguments on semantic interpretation', Dowty (1989): 71) and should probably be considered universal. That is why the notion of how we should construct a model-theoretic account of thematic roles is fundamental.

It is assumed that the meaning of all natural-language predicates semantically entails a thematic role of each of its arguments, one distinct enough that no two arguments fall under the same role definition (cf. Dowty (1989): 79). Natural-language predicates fall under three main categories based on their arity - *intransitive* (with one mandatory argument), *monotransitive* (with two mandatory arguments, I will refer to them simply as *transitive*) and *ditransitive* (with three mandatory arguments). Predicates of higher arity (e.g. tritransitive) are extremely rare in natural language,[1] although we can easily make compound predicates of any arity by adding prepositional phrases to single verbs or by compounding several single verbs into one compound predicate (such a combination is called 'adjunct predication', cf. Rothstein (2004a): 72). These two ways of predicate construction are inductive—you can take *any* predicate, add a preposition to it and receive another predicate; similarly you can take *any* predicate and join it with another, receiving a complex predicate. In this way, a predicate is understood as a recursive structure—it has its basic cases (reflecting three kinds of natural-language predicates) and two inductive steps. But how is this structural induction reflected semantically? Is it recursive as well?

Both ways of constructing compound predicates (by adding a preposition or by compounding predicates together) are handled by Neo-Davidsonian semantics. However, as I will point out, the Neo-Davidsonian analysis faces crucial problems which may drive us to seek alternatives. The paper is structured in the following manner. In Sect. 2 I briefly recall the core tenets of the Neo-Davidsonian analysis of complex predicates, highlighting the semantic properties of theta-roles and explaining how it analyses inferential patterns of sentences with complex predicates. In Sect. 4 I examine occurrences of prepositional phrases and adjunct predication in natural language and highlight groups of examples that resist (simple) Neo-Davidsonian analysis. In Sect. 5 I mention several attempts within Neo-Davidsonian

[1] *Trade* is sometimes given as an example of such verbs, 'John traded Bill an apple for a peach'.

semantics to provide a satisfactory analysis of problematic cases and show why they failed. In the second part of the paper (Sect. 6), I propose an analysis alternative to the Neo-Davidsonian. My analysis is inspired by a way how relations are represented in the relational model of data (RMD). I extend this representation to natural-language predicates and provide syntactic and semantic rules of a relational FOL appropriate for representing compound predicates. In effect, a relational FOL (as well as other languages, a relational algebra or SQL) can be used to operate on *the same* relational model of data in which compound natural-language predicates are represented. I argue that a semantic theory based on RMD should be considered a good semantic theory since, besides the correct truth-conditions, it provides an explanation of the relevant semantic phenomena. In the final sections, I analyze the problematic examples and draw conclusions.

2 Theta-roles in Neo-Davidsonian Semantics

In Neo-Davidsonian semantics (cf. Dowty (1989), Link (1998), Parsons (1990), Landman (2000), Carlson (1984), Krifka (1992)) verbs denote one-place predicates of events, while thematic roles (theta-roles) denote partial functions from the set of events into the set of individuals (Link (1998): 257). Each theta-role (e.g. *Agent*, *Goal*, *Source*, *Location*, *Theme*, *Instrument*) names one function (called a theta-function or thematic function) which takes an event as an argument and returns the event's particular participant (an individual) as a value (cf. Link (1998): 248). For example, the verb *read* denotes a set of reading events. There is a set of theta-roles connected with *read* (the verb's theta-grid) {*Agent*, *Theme*}, which is a representation of the part of the verb's meaning which expresses constraints on participants of reading events. It provides truth-conditionally relevant information about their ways of participation in the events (cf. Rothstein (2004a): 137, 139; Carlson (1984): 266). Because of the truth-conditional relevance, the content of thematic roles is semantic, that is, it allows us to distinguish one participant from another '[...] by virtue of the distinctive properties they have as they participate in an event named by a verb, properties that can be identified ('in the real world') independently of a language or its 'semantic representations'' (Dowty (1989): 73). Thus the logical form of the following sentence 1:

1. John reads 'War and Peace'.

is expressed as a conjunction 1′:

$1'.\exists e(Reading(e) \wedge Agent(e){=}John \wedge Theme(e){=}WP)$.

As is clear upon observing 1′, the verb's arity has no syntactic representation but only a semantic one — every verb, irrespectively of its arity, is represented as a one-place predicate of events (cf. Krifka (1992): 36). In this way, the essence of 'event-based' semantics (as opposed to 'property-based' semantics) is in treating properties named by a predicate as properties of events, and not of individuals,

3 The essential claim of 'event-based' semantics

> '[...] individuals are not primarily thought of having properties or entering relations, but *playing roles in certain events.*'
> (Link (1998): 253).

Note that, despite the fact that theta-roles constitute obligatory parts of the verbs' meaning, not all events with a property expressed by the verb have *all* participants fulfilling these roles (for example, not all events have participants fulfilling agentive roles, similarly for the other theta-roles), therefore theta-functions are necessarily partial functions (cf. Link (1998): 248, 258).

Thematic functions obey several conditions (Dowty (1989): 83–84; Krifka (1992): 41–43). Because they are functions, the situation in which different objects fulfill the same thematic role in the same event is excluded (*uniqueness of objects*, Krifka (1992): 39, def. (P27), *the unique role requirement*, Landman (2000): 38, Parsons (1990): 74). As they may constitute parts of theta-grids of different verbs, their meaning is independent from the meaning of a verb and semantically basic, expressing natural relations between events and their participants (*independence*, Dowty (1989): 84). Because thematic roles are the only way of relating events to their participants (there is no other way of expressing that an individual participates in an event), it follows that every (surface) argument of a verb fulfills a thematic role (*completeness*, Dowty (1989): 83). Because of the uniqueness condition, it is impossible that two (surface) arguments of a verb bear the same thematic role, and so it is possible to distinguish (semantically) one argument from another based on their thematic role (*distinctness*, Dowty (1989): 84).

All of the further conditions that thematic functions have to obey are the consequences of an assumed homomorphism from events to objects which preserves the lattice structure (Krifka (1992): 39). Thus, if a theta-function gives one and the same object as its value, then it has the same event as its argument (*uniqueness of events*, Krifka (1992): 39, def. (P28)).[2] If a theta-function gives values for two events, then for their sum it will give the sum of the values (*summativity*, a connection between theta-functions and join operations, Krifka (1992): 39, def. (P26)).[3] If

[2] This requirement may be viewed as applicable to some but *not all* thematic relations, as it leads to an oversimplified criterion of events identification as listing its participants (two events are one and the same iff they have the same values for their theta-functions), see (Krifka (1992): 44), (Carlson (1984): 274). A problematic consequence of this requirement is clearly seen if we consider one place predicates (e.g. *yawn*, *pray*, *pause* etc.), which have none except one theta-function as a part of their meaning, namely, 'the event's agent'. By *uniqueness of events* all such events with the same agent are one and the same event, which is a highly undesirable result.

[3] One may not want to retain homomorphism between events and objects and in case a theta-function gives different values for different events leave it *undefined* for sum of such events, as Link (Link (1998): 258) did (providing another reason why theta-functions should be partial): 'A sum event or chunk which

a theta-function gives a value (an object) for an event and this event has a subpart, then the mentioned object has a subpart as well, which is the value of the theta-function for the subevent (*mapping to objects*, Krifka (1992): 39, def. (P29)). Finally, if a theta-function gives a value (an object) for an event and this value has a subpart (another object), then the latter will be a value for the theta-function for some subevent (*mapping to events*, Krifka (1992): 39, def. (P30)).[4]

In Neo-Davidsonian semantics, the verb's arity has no syntactic representation but only a semantic one, as conjunctive constraints on participants of events. Now, if we want to characterize events in more detail and bring to attention participants *other* than that *obligatory* involved in the activity or state described by the predicate, one does this with the help of prepositional phrases (PPs). Prepositional phrases which convey such auxiliary information (adjuncts) are treated by the theory in a similar way as thematic functions—they relate events and objects (cf. Krifka (1992): 36-37; Landman (2000): 54). Other adjuncts—adverbs—are treated as predicates of events (cf. Dowty (1989): 92; Landman (2000): 53). In that way, a well-known Davidsonian example:

2. Jones deliberately buttered the toast with a knife, at midnight, in the bathroom.

is analysed as:

$2'$. $\exists e(Buttering(e) \wedge Agent(e)=John \wedge Theme(e)=t \wedge With(e)=k \wedge Deliberate(e) \wedge At(e)=12am \wedge In(e)=b)$.

$2'$ has the form of a conjunction of predicates, predicated of events. Due to such form, it is easy to explain why it is possible to iterate and freely reorder prepositional phrases and adverbs in a sentence without a change in truth-conditions (Carnie (2006): 168). This reordering condition is known as *Permutation* (Davidson (1967); Katz (2008); Landman (2000)). Also, due to the conjunction analysis, it is easy to explain inferential properties of the sentences with prepositional phrases and adverbs. We can infer from 2 that 'Jones buttered the toast deliberately in the bathroom' and that 'Jones buttered the toast deliberately', and that 'Jones buttered the toast'. This type of inference is due to a property called *Drop* (Landman (2000): 3). Furthermore, from the conjunction of 'Jones buttered the toast at midnight' and

Footnote 3 (continued)
is composed of heterogeneous aspects with lots of patients also has no defined patient role since uniqueness fails. If a sum event, however, consists of atomic events which have all of a certain role defined, and, in addition, this role is filled by the same object, then the role is also defined for the sum event, with the same value'.

[4] Because a sum of individuals may become the value of a theta-function, by mapping to events we end up with subevents corresponding to the sum constituents. The consequences may be undesirable in case of counting the events. For example, assume that there is a coin on the table and I am touching it together with a piece of the table. It is clear that there was one act of touching, with two objects being touched. By this token, the sentence 'I performed exactly two acts of touching' should emerge as false. But by *mapping to events* there were two events of touching, as they correspond to two constituents of the sum of touched things, that is why the sentence comes out as true, which leads to a fallacy.

'Jones buttered the toast in the bathroom' we cannot infer that 'John buttered the toast in the bathroom at midnight', as we may be talking about two different events. This type of adverbial entailment failure is known as *Non-Entailment* (Davidson (1967); Katz (2008): 227).

Beside Drop and Non-Entailment, sentences with modifiers have one specific type of entailment. I will use one of Szabó's examples (Szabó (2003): 406) for illustration. Consider: 'John is rational as a chess-player'. Applying *Drop*, we are able to infer from this that John is rational. But we cannot infer that John is rational *simpliciter*—he is rational in quite a specific way, that is, as a chess-player (compare a similar case with another prepositional phrase: 'I am happy about the news' (Szabó (2003): 400). Intuitively, we cannot infer that I am happy *simpliciter*). From the conclusion you get after applying *Drop*, 'X is φ', you cannot infer that 'X is φ *simpliciter*'.[5]

Permutation, together with *Drop*, *Non-Entailment* and *Simpliciter* inferential patterns, constitute semantic requirements that should be incorporated by any semantics of adverbs and prepositional phrases —but are they and how exactly? Let us take a closer look at syntax.

4 In Search of Lost Syntax

Let us have a closer look at the occurrences of prepositional phrases and adjunct predicates. As (Gawron (1986)) I will distinguish the occurrences of PPs as obligatory arguments of verbs (verbal complements) from their occurrences as verbal internal and external modifiers (both adjuncts).[6] As we will see, the PPs' ability of drop, reordering and iteration depends on their syntactic position. When PPs appear as arguments, as in 3-6,

3. I treated my working boots *with* oil.
4. The fog extended *from* London *to* Paris. (Gawron (1986): 350)
5. The song is meant *for* children. (Gawron (1986): 327)
6. John placed the flute *on* the table. (Carnie (2006): 221)

[5] Szabó (Szabó (2003): 403-4) proposed a way to deal with *Simpliciter* entailment. In a simplified manner, if a property P of a state s is persistent, that is, is preserved throughout all extensions of a state s (through all such states that contains s as a part), you can infer that $P(s)$ *simpliciter*.

[6] Despite the fact that I use this distinction after Gawron (1986), I will not use his terminology due to unwanted terminology clash ('co-predication' is already reserved for a different semantic process). Thus I call 'internal modifiers' occurrences of PPs called by Gawron 'co-predicators' and call 'external modifiers' occurrences of PPs called by Gawron (proper) adjuncts. By 'modification' I understand here a syntactical relation defined as follows (Carnie (2006): 85): 'If an XP (that is, a phrase with some category X) modifies some head Y, then XP must be a sister to Y (i.e., a daughter to YP).' Strictly speaking, modifying position for adjuncts on a tree is not to be a sister to N, V, A or P but to N′, V′, A′ or P′ (Carnie (2006): 162), which constitutes the main syntactical difference between adjuncts and complements. However I will leave aside this difference in tree position between complements and adjuncts.

they have obligatory occurrences (thus, they cannot be dropped without a significant change in the meaning of a verb) and even if they are contextually omitted, their presence is still implied (Quirk et al. (1985): 66). Because the theta-role of a prepositional complement (marked by a preposition or a case) constitutes a part of a verb's theta-grid, the prepositional complement cannot be iterated (otherwise the whole will be ungrammatical, which is formally reflected in Government-Binding theory as a violation of the Theta Criterion).

When PPs appear as internal modifiers, they indicate two-place relations between new participants introduced by prepositions and one of the main verb's arguments, but not a whole action described by the main verb ('co-predicators', Gawron (1986): 341-343),

7. I saw Jones through the window *in* the door.
8. Jon saw White Walkers *around* Winterfell.
9. Mary drank John *under* the table. (Rothstein (2004b): 82)
10. I met a fellow journalist *from* Beijing.

In examples 7-10 it is not the actions that were made *in*, *around*, *under* or *from*, but actions' participants (the window, White Walkers, John and the journalist respectively) which was placed *in*, *around*, *under* or have had its origin *from* a new participant introduced by the preposition. Such occurrences of PPs are called 'co-predicative' because the two-place relation indicated by a preposition shares an argument with the main verb (Gawron (1986): 343).

Finally, when PPs appear as external modifiers, they add a new participant to the relation described by the main verb and express a role which the added participant plays with respect to the relation as a whole. Compare occurrences of *for* in the following sentences:

11. Isolde baked cookies *for* Tristan.
12. Isolde left Mark *for* Tristan.

While we can paraphrase 11 as 'Isolde baked cookies and Tristan was the intended *beneficiary of the cookies* she baked' (internal modification, expressing a relation between the cookies and Tristan), we cannot paraphrase sentence 12 in the same way, *'Isolde left Mark and Tristan was intended *beneficiary of Mark* she left'. Rather we would say 'Isolde left Mark and Tristan was the intended *beneficiary of her leaving* Mark' or 'Isolde left Mark and Tristan was *the reason* why she left Mark'. In the following sentences PPs with *on* and *out* appear as external modifiers as well:

13. It's often said that if you're playing [poker] *on* a table full of tight players then you should loosen up.
    (http://www.pokerology.com/lessons/poker-playing-styles/)
14. Mikhail yelled his name *out* the window. (Svenonius (1994): 223)

In 13, it is not you or your partner, but the play itself which is located on the table. In the same vein in 14, it is not Mikhail who is out of the window, but the action of his shouting. My hypothesis is that PPs' ability of being iterated and permuted depends on their syntactic position: PPs can be permuted only if they modify the same node (the same syntactical constituent, represented as a point in a tree diagram[7]) and can be iterated only if they do not modify the same node or express different relationship between objects. For example, PPs in 15 and a PP and a secondary predicate in 16 cannot be permuted because they modify different syntactical constituents (different nodes) and permutation leads to a difference in truth-conditions,

15. Bill made a sweater *for* Mary *for* Miles. (Gawron (1986): 371)
    *Bill made a sweater *for* Miles *for* Mary.
16. Bill fried chicken *for* Mary *naked*.
    *Bill fried chicken *naked for* Mary.

In 18 and 19 PPs can be iterated (forming a sequence of the same PP, a 'path') because they do not modify the same syntactical constituent and in 20 and 21 several locative PPs can appear in one sentence because they express a different space relationship between the object in question and other objects,

17. Mary wrote a letter *in* every town *in* every county *in* every English-speaking country.[8]
18. Look *in* the wardrobe *in* the basement. (cf. Baronian (2006): 34)
19. The duck swam *from* the shore *from* the tree. (Gruber (1962): 112)
20. Joan hit the ball *through* the valley *between* the buildings *into* Mrs Magillacuddy's window. (Gawron (1986): 347)
21. The horse galloped *from in front* of the tree *to under* the tent. (Gruber (1962): 105)

Sentences with the main verb of arity higher than one and a PP used as an internal modifier are often syntactically ambiguous because it is not clear which argument of the main verb is related to a new participant introduced by the PP (the ambiguity between internal modifiers), for example 22,

22. Sherlock saw a man *with* binoculars [Sherlock/a man was with binoculars].[9]

[7] As arguments, PPs are syntactical sisters of zero level projections (heads). Assuming a binary branching structure, it is impossible to represent Gawron's distinction between PPs' occurrences as internal and external modifiers (co-predicators and adjuncts in his terminology)—they both come out as 'adjuncts', that is, syntactic sisters of X' (Carnie (2006): 164).

[8] A similar example can be found in (Partee (2000): 489).

[9] The example is taken from Kai von Fintel. *24.903 Language and its Structure III: Semantics and Pragmatics* Spring 2005. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu. License: Creative Commons BY-NC-SA.

The syntactic ambiguity is also present when it is not clear if a PP is used as an internal or as an external modifier (the ambiguity between internal and external modification), take for example sentences 23, 24 and 37,

23. The spy shot James Bond *from* the roof. (cf. Gawron (1986): 349)
24. The boy threw stones *from* the roof. (Himmelmann and Schultze-Berndt (2005): 38)

23 and 24 are ambiguous between the reading in which James Bond/the stones were propelled from the roof (the PP is used as an internal modifier) and the reading in which the shooting/throwing itself took place on the roof (the PP is used as an external modifier, Gawron (1986): 349-350).

If we look at the modifying uses of predicates, we can see that they are used in a similar manner to PPs. When a predicate is used as a modifier, it can modify a particular argument of the main verb (secondary predication, similar to an internal modifying use of a PP) or it can modify the whole situation described by the main verb (adverbial use, similar to uses of PPs as external modifiers, see Table 1).

Thus in 25 and 26 secondary predicates modify a particular argument of the main verb,

25. Kim ate the steak *raw*. (Burkhart et al. (2017): 21)
26. Kim ate the steak *hungry*. (Burkhart et al. (2017): 21)

All arguments of the main verb may be modified by secondary predicates 27-29,

27. Bill drove the car *broken drunk*. (Rothstein (2003): 557)
28. Ray put the door *shut tired*. (Winkler (1997): 131)
29. Kim ate the steak *raw hungry* (Burkhart et al. (2017): 24)

Similarly to PPs, predicates in their modifying uses can only be iterated if they modify different syntactic constituents (which gives rise to the intuition that modifiers have a 'scope'),[10][11]

30. John hammered the metal *flat hot*. (Winkler (1997): 7)
31. They eat the meat *raw tender*. (Winkler (1997): 7)
32. John *painstakingly* wrote *illegibly*. (Parsons (1970): 324)

or they are modifiers of a different type (e.g. place, mode, time etc.),

[10] In the case of modification on different nodes it is possible to use predicates expressing contradictory properties without getting a contradiction, 'John drove the car drunk soberly' (drunk John was driving in a sober manner, Rothstein (2004b): 67).

[11] Not all accept examples as 30-32, therefore I take the argument from iteration and scope as supportive.

33. The loggers cut a tree *in the front yard into pieces green*. (Winkler (1997): 79).

Similarly to sentences with PPs, sentences with predicates as modifiers may be ambiguous in two ways, when it is unclear which argument is modified by a secondary predicate (examples 34 and 35) and when it is unclear whether a predicate modifies the main verb as a whole or one of its arguments (example 36),

34. I saw Carroll *standing at the window*.
35. Tom talked to Meg *drunk*. (Burkhart et al. (2017): 22)
36. Barbara saw *an occasional* sailor. (Larson (1998): 19)
    ('Barbara saw a person who occasionally sailed / Occasionally, Barbara saw a sailor')

To account for the examples of internal modification and secondary predication (examples 7, 10, 25, 26), 'paths' examples (17, 18), scope examples 15, 16 and provide different logical forms for cases of the ambiguity of two types (examples 34-36) a Neo-Davidsonian has to assume that nouns, similar to verbs, denote one-place predicates of states, while thematic roles denote functions from the set of events and states to individuals (cf. Link (1998): 253 footnote 1). To see how this assumption helps, let us consider example 37:

37. I purchased a property *in* Belize *in* Washington.

Intuitively 37 has different truth-conditions than 'I purchased a property in Belize *and* I purchased a property in Washington'. To obtain the right interpretation, we have to assume that the noun 'property' names a state and thus analyse 37 as 37′:

37′. $\exists e \exists x(Purchase(e) \wedge Agent(e){=}I \wedge Theme(e){=}x \wedge In(e){=}W \wedge \exists s(Property(s) \wedge Experiencer(s){=}x \wedge In(s){=}B))$.

While it is possible to hold (sensibly) that there are states such as 'a property-in-Belize' or 'a wardrobe-in-the-basement', this assumption does not work in all cases (e.g. there are no such states as 'Bond/he-from-the-roof' or 'Sherlock/he-with-binoculars') because proper names, as well as pronouns, *are not predicates*,,[12] so they cannot name states nor assign theta-roles. In the same vein, all of the following examples remain problematic for the Neo-Davidsonian account taken in a simple form:

22′. Sherlock saw Moriarty/him *with* binoculars. (There is no such state as 'Sherlock'/'Moriarty'/'he' which is 'with-binoculars'.)

[12] This argument can be seen as a weaker claim, that proper names and pronouns have *referential* not *predicative* uses in these examples.

Table 1 Modifying uses of PPs and predicates

Role	PPs	Predicates
Modify a particular argument of the main verb	Internal modifiers	Secondary predicates
Modify the main verb with its arguments as a whole	External modifiers	Adverbs

15. Bill made a sweater *for* Mary *for* Miles. (There is no possibility to provide different scope for two external modifiers[13] and thus block their reordering.)

38. I called Tom *from* Beijing *from* London. (*From* in its two occurrences has the same meaning. In its last occurrence *from* is analyzed as a function which returns a participant of the calling event. But what is the correct analysis of its former occurrence?)

Summing up, if we share an intuition that in examples of co-predication such as 8 and 9, it is not the *states* but *objects* which were placed around or under something (that is why the analysis of 9 as ‘there is a drinking under the table with Mary and John as participants’ is intuitively incorrect), then we agree that such examples are especially challenging for Neo-Davidsonian semantics, because they constitute counterexamples to its essential claim, that is, that only events (not individuals) may have properties or enter relations. Taken in its original form, Neo-Davidsonian analysis was put forward to provide a logical form for sentences in which modifiers are used to express properties of events *as a whole* or express relations between events *as a whole* and new individuals introduced by PPs. Because of that, taken in its simple version, it cannot provide a unified account of syntactic ambiguities, explain syntactic dependencies (the dependency of reordering and iteration on syntactic positions), analyse internal modification or explain scope examples.

Perhaps the best potential strategy for Neo-Davidsonians to deal with mentioned problems would be the following:[14]

– hold the claim that only events (not individuals) may have properties or enter relations;
– accept that there are (basic) predicates with ‘built-in’ prepositions (e.g. ‘being-located-in’);
– and try to analyze problematic examples as those in which these (basic) predicates are used as secondary predicates.

[13] Such a reading is clearly possible. We can sensibly ask: What did Bill do for Mary? - Made a sweater. And what did Bill do for Miles? - Made a sweater for Mary.

[14] I owe this idea to :Wojciech Suchoń personal communication.

Even assuming that this strategy will work (so it will be possible to separate internal from external modification, thus analyze, for example, two *froms* in 38 differently), example 15 remains problematic because two *for* in it are external modifiers (that is, they characterize an event as a whole) and are analyzed as conjunctive constraints on the event's participants. As conjunctive constraints, they can be freely permuted without an impact on truth-conditions but, intuitively, permutation leads to change in truth-conditions, therefore it is blocked. It is blocked because two *for* in 15 modify different syntactic constituents—intuitively, the latter *for* in 15 modifies *everything* already modified by the former *for*, together with the complement the former *for* adds (otherwise we will get a reading that Bill made a sweater for both Mary and Miles, which is clearly ruled out). Moreover, if two instances of *for* represent the same function that takes the same event as argument, then something must be given up (because of different values)—either it is not a function, or it does not operate on the same event. The other possibility is to assume that there are *two different* theta-functions, say $Beneficiary_1$ and $Beneficiary_2$, but again in this case it follows from the logical form $15'$ that $15''$,

$15'$. $\exists e(Making(e) \wedge Agent(e){=}B \wedge Theme(e){=}s \wedge Ben_1(e){=}Mary \wedge Ben_2(e){=}Miles)$

$15''$. $\exists e(Making(e) \wedge Agent(e){=}B \wedge Theme(e){=}s \wedge Ben_2(e){=}Miles)$

which is a logical form for 'Bill made a sweater for Miles', the ruled out inference.

In the next section I will explain why secondary predicates' analysis within Neo-Davidsonian semantics is problematic and why some other known attempts to analyze examples of internal modification fail.

5 Attempts to Fix Syntax Within Neo-Davidsonian Theory

5.1 Sum Relation

Rothstein (Rothstein (2004a), Rothstein (2004b)) and Maienborn (Maienborn (2003)) attempted to deal with examples from Sect. 4 and in this section I will demonstrate that all these attempts have undesirable consequences. As it was clear from the previous section, there is a need to provide a difference between internal and external modification. This can be done, for example, by representing internal modification as secondary predication, but as I will argue in this section, the existing semantic analysis of secondary predication proposed by (Rothstein (2004a), Rothstein (2004b)) is problematic in itself. So how does Rothstein propose to analyse secondary predication? Assume for simplicity what $\mathcal{E}$ is a set of atomic events such that $\mathcal{E}$ equal $\{e_1, e_2, e_3\}$. From the completeness axiom (Krifka (1992): 32, def. (P2)), for every pair of events from $\mathcal{E}$ there exists their sum which is also an event (def. (P1)). Alongside this standard join operation on events Rothstein (Rothstein (2004b): 67) proposes to consider another operation $\cup^S$ ($\cup^S : \mathcal{E} \times \mathcal{E} \rightarrow \mathcal{E}$) which maps events e_1 and e_2 to a new singular event e_3 which is equal to their sum. Now, if two events are happening simultaneously and share a participant (e.g. a thematic participant, Rothstein (2004b): 72), they can be summed up using this operation. Let e_1 be the event of John's hammering the metal and e_2 the event[15] of the metal being flat. Sentence 39 is then analyzed as 39′:

39. John hammered the metal flat.

39′. $\exists e_3 \exists e_2 \exists e_1 (e_3 = (e_1 \cup^S e_2) \wedge \mathit{Hammering}(e_1) \wedge \mathit{Agent}(e_1){=}J \wedge \mathit{Theme}(e_1){=}m \wedge \mathit{Flat}(e_2) \wedge \mathit{Arg}(e_2){=}m)$.

In 39′ *Arg* stands for a thematic role which adjectives assign to participants of events (Rothstein (2004a): 137). In this way, sentences with secondary predication are analyzed as having a logical form containing a representation of a complex event which is a result of the summing operation on events (fulfilling time and participants sharing conditions).[16]

Let me point out that the condition that two events should be simultaneous and share a participant (cf. Rothstein (2004b): 72) is insufficient by itself for a sentence with a secondary predication to be true (because of *Non-Entailment*). Imagine, for example, that Jones has marijuana and during the period when he has it, he is caught by the police several times, but nevertheless he was never caught possessing marijuana. Despite Jones being caught is simultaneous with Jones' state of possessing marijuana, and have a common participant (Jones), these conditions are insufficient for 'Jones was caught by the police possessing marijuana' to be true.

My first objection to this analysis stems from the definition of a part relation defined as $x \subseteq y$ iff $x \cup y = y$ (Rothstein (2004b): 67). Essentially if x and y are the same event, then their sum results in this very event. Because of that, the analysis is problematic in cases when both the main and the secondary predicate name the same event. Imagine that there are two mirrors hanging opposite each other and I utter:

40. I am seeing myself seeing myself.

40 should be analysed as:

40′. $\exists e_3 \exists e_2 \exists e_1 (e_3 = (e_1 \cup^S e_2) \wedge \mathit{Seeing}(e_1) \wedge \mathit{Agent}(e_1){=}I \wedge \mathit{Theme}(e_1){=}I \wedge \mathit{Seeing}(e_2) \wedge \mathit{Agent}(e_2){=}I \wedge \mathit{Theme}(e_2){=}I)$.

It would be questionable to claim that there are two events of seeing and that these events are different. If the events are one and the same, then adding the event to itself results in this very event, thus in effect the sentence is true just in case I am seeing myself in a mirror, which is clearly a wrong satisfaction criterion.

15 'Being flat' is a state rather than event, but I follow Rothstein's (2004b: 80) analysis of 'flat' as an event.

16 A similar solution for adverbs was proposed by Schäfer (2008): 366.

My second objection to this analysis stems from the commutativity of sum operation U^S. Take two predicates, P and Q, naming different events and let us mark with an index which predicate is a primary/secondary. By this analysis $P^1Q^2(a)$ is true iff $Q^1P^2(a)$ is true as well, which is a wrong entailment dependency. Assume that sentence 41 is true.

41. Being a suspect, Jones refuses to make a statement.

It immediately follows that sentence 41′ is true as well,

41′. Refusing to make a statement, Jones becomes a suspect.

This is so because both sentences have exactly the same logical form, $\exists e_3 \exists e_2 \exists e_1 (e_3 = (e_1 \cup^S e_2) \wedge Suspect(e_1) \wedge Agent(e_1){=}J \wedge Refusing(e_2) \wedge Agent(e_2){=}J)$. On Rothstein's account, sentences as such 41 and 41′ either both true or both false but intuitively 41 and 41′ may differ in truth-value.

Thus, in addition to the examples of the blocked reordering of adjuncts (such as 15) examples of 'self-joins' (such as 40), as well as examples of blocked interchangeability of roles of being a primary/secondary predicates in adjunctive joins (such as 41 and 41′) are where the Neo-Davidsonian analysis fails to deliver satisfactory interpretative results.

5.2 Part-of Relation

There are several reasons to think that a part-of relation between events may be a clue to providing the correct analysis of examples from Sect. 4. It seems that we may need a part-of relation in order to be able to distinguish between PPs as event-internal and PPs as event-external modifiers. Maienborn (2003) pointed to the contrast between the following sentences:

42. John eats herring in the office.
43. Jones eats herring in cream sauce.

Although it is possible to understand 43 in such a way that it states that Jones ate herring while wading through a sea of cream sauce, this reading is rather bizarre. So in order to understand the sentence properly it has to be assumed that not the whole event of eating but one of its parts, that is the herring, was in cream sauce. In this way, according to Maienborn, event-internal modifiers express relations between event integral constituents. I will skip theta-roles assignment for simplicity. In a simplified manner 43 should be analyzed as:

43′. $\exists e \exists x, y(Eating(e) \wedge Herring(x) \wedge Sauce(y) \wedge (x \subseteq y) \wedge (y \subseteq e) \wedge (J \subseteq e) \wedge \text{IN}(x, y))$.

According to this logical form, Jones, herring and sauce are involved in the eating event and in this event, the herring was located in the cream sauce.

So far, so good, but we also have sentences like the following:

44. Jones hops on one leg.

'On one leg' is an internal modifier, so it should express a relation between constituents of the hopping event, that one of its constituents is situated on another. But in 44 we have a relation which does not obtain between different individuals but between parts of Jones, who in turn is also a part of the event. To express the relation between parts of an individual, Maienborn added the mereological notions of the proper part and the mereological difference ~. Using these notions, she proposes to analyze 44 as stating that in the hopping event, one proper part of Jones (namely 'John-minus-his-leg', $J{\sim}l$) was situated on his other proper part, that is, on his leg (Maienborn (2003): 497), I skip theta-assignments for simplicity:

44′. $\exists e \exists ! l(Hoping(e) \wedge Leg(l) \wedge (J \subseteq e) \wedge (l \subseteq e) \wedge (J{\sim}l \subseteq e) \wedge (l \subseteq J) \wedge \textsc{on}(J{\sim}l, l))$.

Here, the event-internal modifier 'on one leg' provides a location of Jones' remaining body relative to his leg.

There are several problems with this analysis. For one, it requires us to abandon the main thesis of event semantics, that only events (not individuals) may have properties or enter relations. Secondly, the 'part-of' relation is transitive but this transitivity generates intuitively false consequences. Let me exemplify the presence of unwanted transitivity in Maienborn's account. Consider 45:

45. Jones balances on fingers.

This sentence has the following logical form according to Maienborn:

45′. $\exists e \exists f(Balance(e) \wedge Fing(f) \wedge (J \subseteq e) \wedge (f \subseteq e) \wedge (J{\sim}f \subseteq e) \wedge (f \subseteq J) \wedge \textsc{on}(J{\sim}f, f))$.

Jones' palms are proper parts of Jones, $p \subset J$. Take 'John-minus-palms'. This part of Jones is contained in 'Jones-minus-fingers' part. If 'Jones-minus-fingers' is a part of the balancing event and 'Jones-minus-palms' is a part of 'Jones-minus-fingers', then 'Jones-minus-palms' is a part of the balancing event as well. Jones himself is a part of balancing event and his palms are a part of him, and because of that his palms are also a part of the balancing event. We know that the 'Jones-minus-fingers' part is located on the fingers. This is possible only if 'Jones-minus-palms' part is located on Jones' palms. In this way if 'Jones balances on fingers' truly describes the balancing event, so does 'Jones is balancing on his palms', but intuitively this is not so (we can express this by saying 'Jones is balancing on his fingers, not on his palms').

Not only iteration of prepositional phrases but iteration of other modifiers (e.g. subsective adjectives (Francez 2017), adverbs (Schäfer 2008)) is not transitive. For example an application of a subsective adjectives can be paraphrased with *as*-phrases (the possibility of such a paraphrase is considered a diagnosis of their subsective nature Morzycki (2015): 22). Consider:

46. Teryl Austin has been confirmed as being hired as the Lions' defensive coordinator.[17]

Intuitively, Teryl has been confirmed *as being hired as the Lions' defensive coordinator* (modification of a predicate *confirmed* by the already modified predicate *hired as a defensive coordinator*). We can not interpret 46 in a classical Davidsonian way, as a conjunction of predicates predicated of states, because reordering of conjunctive constraints is blocked. Again, we will encounter the mentioned problem using a part-of relation in analysis (e.g. such as proposed in Szabó (2003) for *as*-phrases), due to the unwanted transitivity between the states: the state of *confirmation* is a part of the state of *being hired*, and in turn, the state of *hiring* is a part of *being a defensive coordinator* state. This means that Teryl has been confirmed as the Lions' defensive coordinator, which is not an intuitive analysis of 46. For example, imagine that Teryl has been the Lions' defensive coordinator for years but was not hired as such. Once he was hired and confirmed as being hired as a defensive coordinator, it is not a part of a confirmation that he is the Lions defensive coordinator (this information is already known and requires no confirmation).

What conclusions follow from Sect. 4 and 5 First of all, Neo-Davidsonian semantics cannot stand as it is - something should be added to it which would allow us to properly analyze iteration of modifiers (it is not transitive, as follows from 46), analyze self-joins (example 40), examples of blocked reordering for adjuncts, and to distinguish between event-internal and event-external modification.[18] Moreover, we can't use either the sum operation or the part-of relation for these purposes (as examples 40, 41 and 46 show).

Confronted by the above problems, why not to try developing an event-free semantics which avoids them?

6 Representing Natural-Language Predicates in the Relational Model of Data

What if we represent theta-roles not as functions from events to their participants, but rather as labels on the arguments of the verb encoding the role which each argument plays? The idea itself is not novel (cf. Landman (2000), Dowty (1989), Beaver and Condoravdi (2007), Eckardt (2010)) and the main differences between the existing notions and my own proposal need to be highlighted. Landman (2000), Beaver and Condoravdi (2007) and Eckardt (2010) develop a theory of the kind which Dowty called (1989: 71-72) 'an ordered-argument system', one in which the labels for the roles that various arguments fulfill in a relation are arbitrary, and their use

[17] The example is from 'World News', accessed 11 July, 2014; http://article.wn.com/view/2014/01/18/Lions_hire_Teryl_Austin_as_defensive_coordinator_retain_8_fr/.

[18] I want to highlight that I am *not* making a strong claim that Neo-Davidsonian event semantics is incapable of analyzing particular data, and therefore should be rejected. I take the presented challenges not as evidence against the account but as an inspiration to seek alternatives.

implies no semantic commitment (they are merely syntactic labels for distinguishing one argument from another in the interpreted syntax, not sets of properties, Dowty (1989): 76). I do not use this solution - in the theory developed here, labels for roles *do have* semantic interpretation (in the sense that a truth function maps them to elements of a model domain). I develop a theory which may be called (Landman (2000): 93) 'neo-McConnell-Ginetian', since thematic roles play an essential role in the theory (a verb accesses arguments through the roles only), and because of preserving and expanding the intuition that prepositional phrases increase the verb's valence.[19] What I propose is to use an advantage of such a representation of theta-roles: if we represent them as (meaningful) marks, then we can use an already existing theory and treat predicates in the same way relations are treated in RMD, as naming sets of tuples in which every object is given together with its role named by a corresponding attribute (a theta-role), and use operators of a relational algebra to calculate extensions of compound predicates. In order to develop this idea, I will first briefly present some of the basic notions of RMD.

6.1 The Relational Model of Data. A FOL and a Relational FOL

The relational model of data was invented by Codd (1970) for modelling data using a representation of relations based on tables. Data are represented as a collection of tables (called a relational database). Each table represents a relation; it has its own (unique) name, as well as the names of columns. A row in a table represents a relationship among sets of values and corresponds to a tuple. There is a set of possible values for each column called *a domain*. In a set theory relation of a degree n is a subset of the Cartesian product of n (not necessary distinct) domains, so in order to understand the meaning of a relation, one has to refer to the position of a particular domain in the sequence. But once we have labeled the domains (by attributes, standing for roles domains have in a relation), we may refer to them using these labels, therefore the ordering of domains in the relation is insignificant (cf. Atzeni and De Antonellis (1993): 4).[20][21]

[19] The theory developed here is 'neo-McConnell-Ginetian' only in spirit - ultimately I do not follow Landman's or McConnell-Ginet's semantic analysis.

[20] By this definition, all permutations of a relation (achieved by interchanging the relation's columns) are considered to be one and the same relation (cf. Garcia-Molina et al. (2013): 63). Thus, for example, active and passive voice ('somebody hits something' / 'something is hit by somebody') are permutations of one and the same relation (cf. Dowty (1989): 74).

[21] Since relations are represented as sets, one can apply the usual set operations to them, as well as specific operations allowing us to derive relations from relations. There are several operations on relations specific to a relational algebra (a query language for relational databases), but I confine myself to 'natural join' and 'rename' only (see Definition 8 and Definition 10 in Appendix). Quine (Quine (1960): 344) introduced the operator 'Der' (abbreviation of 'derelativization') which allows to produce a n-1-place predicate out of initial n-place predicate (e.g. 1-place predicate 'bites something' from the 2-place predicate 'bites'). This is an example of projection operation (heavily used in the relational database theory), that is, deleting the 'object' column from 'bites' relation.

I will briefly explain the transition from a FOL to a relational FOL.[22] For the sake of simplicity, I will use two 'toy' languages, a FOL $\mathcal{L}_1$ and a relational FOL $\mathcal{L}_2$. In a FOL an extension of a n-place predicate $[\![P]\!]$ is a relation, a set of tuples $\langle d_1, d_2, \dots, d_n \rangle$. Each of such ordered (assigned a natural number) n-tuple can be rewritten as paired with an index (thus each ordered n-tuple $\langle d_1, d_2, \dots, d_n \rangle$ can be rewritten as an unordered set of n pairs, that is, $\{(1, d_1), (2, d_2), \dots, (n, d_n)\}$). It may be clearly seen that the ordering of elements of the set is without importance because reordering does not change pairing between index i the domain has in the sequence, and d_i (an element of the domain). We can represent all pairings as a table in which each row corresponds to a single mapping f and each column represents a collection of values of a given index paired with a predicate symbol. In that way, for each predicate symbol P each row (a mapping) in a table restricted to $(P, 1), \dots, (P, n)$ columns corresponds to a single tuple included in the relation which is the extension of the symbol P.

Consider an example. Let us take an unary symbol P, a binary symbol Q and a ternary symbol R and define their extensions in a standard way, say as $[\![P]\!] = \{\circ, \bullet, \square, \blacksquare\}$, $[\![Q]\!] = \{\langle \circ, \bullet \rangle, \langle \bullet, \circ \rangle, \langle \blacksquare, \blacksquare \rangle\}$, $[\![R]\!] = \{\langle \circ, \square, \blacksquare \rangle, \langle \square, \blacksquare, \circ \rangle, \langle \blacksquare, \circ, \square \rangle\}$. We can represent same extensions of these symbols in a FOL $\mathcal{L}_1$ as in the following Table 2.

For example, we want to know what the extension of symbol Q is. It is the set of mappings f (rows in the table) restricted to $Q1$, $Q2$ columns. Note again, that every row restricted to these columns correspond to a single tuple in $[\![Q]\!]$ defined in a standard way.

I am now prepared to present a relational language $\mathcal{L}_2$ developed on the basis of $\mathcal{L}_1$. I see RMD as a choice to operate on unordered *named* tuples instead of unordered *indexed* tuples which means that there is a need to choose a set of attributes *Atr* (instead of set of indices) and in the case of every relation define 'a conversion rule', that is a bijection between the set of indices *Ind* and *Atr*. Such a conversion constitutes a core difference between FOL $\mathcal{L}_1$ and a relational FOL $\mathcal{L}_2$.[23] For example, we may define a bijection in such a way that it maps 1 to $\mathfrak{a}$ ('agent'), 2 to $\mathfrak{o}$ ('object') and 3 to $\mathfrak{g}$ ('goal'). Now Table 2 (the set F of $\mathcal{L}_1$) may be rewritten as Table 3 (the set F of a relational $\mathcal{L}_2$). The extension of a n-place predicate P is a set of all rows restricted to $(P, \mathfrak{d}_1), \dots, (P, \mathfrak{d}_n)$ columns.

Now, after explaining the crucial step in understanding the difference between a FOL and a relational FOL, I am ready to present a relational FOL $\mathcal{L}$ appropriate for the representation of natural-language predicates.

6.2 Representation of Natural-Language Predicates: Syntax

The main idea is simple: we use prepositions (as well as other modifiers) to make compound predicates from atomic predicates. The other way of obtaining compound

[22] A relational FOL appropriate for representing natural-language predicates is a modified version of a standard relational FOL proposed by Reiter (1989).

[23] I owe this explicit comparison of tuples in logic and tuples in RMD to Aleksandra Samonek.

Table 2 The set F of $\mathcal{L}_1$

	$P1$	$Q1$	$Q2$	$R1$	$R2$	$R3$
f_1	∘	∘	•	∘	□	■
f_2	•	•	∘	□	■	∘
f_3	□	■	■	■	∘	□
f_4	■	−−	−−	−−	−−	−−

predicates is to join several predicates in one complex adjunctive predication (I will refer to such a join as an 'adjunctive join'). As I have mentioned earlier, I will treat predicates in the way that relations are treated in RMD, that is, as naming sets of tuples in which every object is given together with its role named by the corresponding attribute. Let $\mathbb{T}$ be a set of attribute-types (domain labels) to be used in different relations. From this set we will obtain attributes for every predicate P by taking an attribute-type $\mathfrak{d}$ (a domain label) and qualifying it with a predicate symbol in dot notation, $P.\mathfrak{d}$ (see Definition 2, in Appendix).[24] What are these attribute-types? Inspired by Fillmore's (1968) notion of 'labeled relations', they are: $\mathfrak{a}$ ('agent'), $\mathfrak{o}$ ('object'), $\mathfrak{g}$ ('goal'), $\mathfrak{b}$ ('beneficiary'), $\mathfrak{i}$ ('instrument') and $\mathfrak{l}$ ('location').[25]

There are many other roles which objects play in the relations named by natural-language predicates (e.g. 'reason'). The roles can also be more finegrained ('inner location' for *in*, 'surface location' for *on*) but, for the sake of simplicity, I will restrict myself only to the one from $\mathbb{T} = \{\mathfrak{a}, \mathfrak{o}, \mathfrak{g}, \mathfrak{l}, \mathfrak{i}, \mathfrak{b}\}$. There is a striking resemblance between Fillmore's, Codd's and Chomsky's requirements (for, respectively, 'a particular case relationship' (Fillmore (1968): 42), 'a distinctive role name, which serves to identify the role played by that domain in the given relation' (Codd (1970): 380) and 'a theta-role' (theta-criterion) Chomsky (1981): 36) which can be expressed as a requirement for relational algebra that all attributes in a particular relation must be distinct (Elmasri and Navathe (2001): 67).[26] Accordingly, I will define the basic (atomic) relations together with their relation schemes. Consider a set of prepositions and three sets of relation symbols together with distinguished relation symbols (see Definition 1 in Appendix):

[24] Note that an attribute of a predicate is nothing else but an 'indexed role' (Landman (2000): 32).

[25] I agree with Williams (1981): 81 that actual labels themselves are not important. Besides the attribute-types mentioned I will use a distinguished attribute-type *id* which represents nothing and is used rather as a technical tool needed to be able to join extensions of complex predicates in a valid way (I will return to this notion later in Sect. 6.3.1).

[26] 'Each argument bears one and only one Θ-role and each Θ-role is assigned one and only one argument.' Note that 'the argument' in the criterion is understood as a grammatical argument, as a participant obligatory involved in the activity or state expressed by the predicate (cf. Haegeman (1994): 44). By this token, adjuncts (optional phrasal constituents) are never arguments and they never appear in theta-grids (Carnie (2006): 222). However, if we take a revised version of the theta-criterion (revised in order to accommodate secondary predication) after Rothstein (2004a): 186), the criterion amounts to the general requirement that an n-place theta-grid is lexically realized by a head that has n syntactic arguments.

Table 3 The set F of a relational $\mathcal{L}_2$

	$P\mathfrak{a}$	$Q\mathfrak{a}$	$Q\mathfrak{o}$	$R\mathfrak{a}$	$R\mathfrak{o}$	$R\mathfrak{g}$
f_1	$\circ$	$\circ$	$\bullet$	$\circ$	$\square$	$\blacksquare$
f_2	$\bullet$	$\bullet$	$\circ$	$\square$	$\blacksquare$	$\circ$
f_3	$\square$	$\blacksquare$	$\blacksquare$	$\blacksquare$	$\circ$	$\square$
f_4	$\blacksquare$	$--$	$--$	$--$	$--$	$--$

- Set of symbols $\mathbb{P} = \{\text{IN}, \text{WITH}, \text{FOR}\}$ which correspond to prepositions;
- Infinite sets of symbols IV, TV, DV, Γ, where IV, TV, DV are infinite sets of symbols for intransitive, transitive verbs and ditransitive verbs, respectively; including a set of symbols $\Gamma = \{L, U, D\}$, where L, U, D are interpreted as, respectively, *to be located in*, *to be used as a tool* and *to be destined for*.

Basic relations have a fixed arity (degree, it is a value of function *ar*, Definition 1). Let X be the set of all finite sequences of members of $\mathcal{S} \times \mathbb{T}$ ($\mathcal{S}$ stands for a set of relation symbols and $\mathbb{T}$ for a set of attribute-types). Function ar^* prescribes an element from X for each element of the ordered set of symbols (*arity*) as follows (Definition 1):

- $(P, \mathfrak{a})$ for $P \in IV$;
- $\langle (P, \mathfrak{a}), (P, \mathfrak{o}) \rangle$ for $\langle P, P \rangle$ if $P \in TV$;
- $\langle (P, \mathfrak{a}), (P, \mathfrak{o}), (P, \mathfrak{g}) \rangle$ for $\langle P, P, P \rangle$ if $P \in DV$;
- $\langle (P, \mathfrak{o}), (P, \mathfrak{l}) \rangle$ for $\langle P, P \rangle$ if $P = L$;
- $\langle (P, \mathfrak{a}), (P, \mathfrak{i}) \rangle$ for $\langle P, P \rangle$ if $P = U$;
- $\langle (P, \mathfrak{o}), (P, \mathfrak{b}) \rangle$ for $\langle P, P \rangle$ if $P = D$.

A value of $ar^*(ar(P))$ is called *a relation scheme* and each $(P, \mathfrak{d})$ (or simply $P.\mathfrak{d}$) in the relation scheme is called *a relational attribute* (see Definition 2).[27]

None of the atomic predicates (other than distinguished) have attributes of 'instrument', 'location' or 'beneficiary'. But how do predicates get other attributes, for example $Q.\mathfrak{i}$ ('instrument of Q') or $Q.\mathfrak{l}$ ('location of Q')? With the intuition that *prepositions add a new attribute to a relation scheme*,[28] we need to explain how prepositions gain an attribute, and how to separate (a) prepositions which are added internally, that is, to one of the arguments of the main relation, expressing that the argument is related to one represented by prepositional complement (expressing a relation between two entities (occurrences of PPs as internal modifiers), cf. Quirk

[27] Note that relation schemes of atomic predicates are nothing other than theta-grids (cf. Carnie (2006): 221).

[28] In some case languages (e.g. in Russian) a new attribute can be added to a relation scheme not by means of a preposition but morphologically, with a special case. Case is a morphological marking of a distinctive role-name that an argument has in a relation scheme (cf. Chomsky (1981), Stowell (1981), Garson (1981)).

et al. (1985): 657) from (b) prepositions which are added externally, to a relation as a whole, to modify it with an argument with a new role (occurrences of PPs as external modifiers).

I propose treating prepositions as syncategorematic expressions[29] - despite having their own (partial) meaning, they only become fully meaningful in a combination with a relation. Take the individual cars in a train as an example. There is a head car and a tail car in a train. 'A head car' / 'a tail car' are names of roles that particular cars fulfill in a particular train, and without speaking of a particular train it would be pointless to ask if a car is a head / a tail car or not. Similarly, there is no point in asking if something is a tool or a beneficiary without speaking of a particular relation. With a preposition, we can add a particular attribute to a relation scheme (e.g. 'beneficiary' or 'instrument') and this attribute becomes fully meaningful in this scheme. The treatment of prepositions as syncategorematic expressions uses the same background intuition as the Neo-Davidsonian approach where an object fulfills a particular role not *per se* but with respect to a particular event.

In general, prepositions denote spatial relations between objects (*to*, *at*, *away from*, *on*, *off*, *in*, *out of*, *over*, *under*, *above*, *behind*, etc.), as well as relations of using something as an instrument (*by*, *with*), receiving something (*for*), being (un) accompanied with something (*with*, *together with*, *without*) or being in possession of something (*of*). All these relations can be explicitly revealed in different syntactic patterns of the same argument structure realization ('He is *the recipient* of many awards' - 'Many awards were given to *him*'), as well as function as standalone relations ('I *received* a phone call'). As I noted in Sect. 4, prepositions (as well as predicates) may have internal and external occurrences. Predicates may also incorporate a prepositional meaning (their role in a relation). By 'meaning incorporation' I understand the notion that the relation schemes of relations include an attribute of a type connected with a preposition. In that sense ditransitive verbs incorporate 'goal' attribute-type in their relation schemes (they do not get it with a preposition *to* as other predicates) and predicates such as 'to be located in', 'to be meant for', 'to use something as a tool' incorporate the meaning of prepositions in their relation schemes (*place*, *beneficiary* and *instrument* respectively). Ultimately, I find no reason to maintain that prepositions are ambiguous between their internal and external occurrences - any account treating them in the same way in both their occurrences is semantically preferable than any other stating an ambiguity. Beside semantic uniformity, prepositions in their external and internal occurrences seem to behave in much the same way, for example, new arguments added by modifiers could be modified further.

When a predicate is modified internally I propose to analyze this modification as an adjunct predication, that is, the modification of the main predicate by a relation

[29] By 'syncategorematic expressions' I mean that the expressions become meaningful only in a combination with expressions which have a denotation of their own. Such an account was offered by Henry of Ghent, cf. (Kretzmann 1982, 213): 'And they are called syncategorematic as if to say "consignificant" - i.e., significant together with others […] not because they signify *nothing* on their own, but because they have a signification […] whose definiteness they derive from those [words] that are adjoined to them.'

with a prepositional meaning built-in as a part of its scheme - we combine this relation with the main predicate in order to express a relationship between *one* argument of the main predicate and a prepositional complement. In order to create a new relation of higher arity with a correspondence among *all* arguments of the main predicate and a prepositional complement (external modification), we combine the main predicate with a preposition. Thus prepositions can be seen as operators on relation schemes—added to a predicate, they extend its relation scheme by adding a new (specific) attribute.

Here the formal representation of one of the two inductive ways of building predicates in natural language ('you can take *any* predicate, add a preposition to it and receive another predicate'), that is, a rule of making a predicate symbol for a new relation together with a rule of making the relation scheme for this new relation (see Definition 1 in Appendix):

- If $R \in \mathcal{S}$ and $P \in \mathbb{P}$ then $R{\cdot}P \in \mathcal{S}$;
- The relation scheme of $R{\cdot}P$ is $\langle ar^{*}(ar(R)), (R, \mathfrak{c})\rangle$, where
  - $(R, \mathfrak{c}) = (R, \mathfrak{l})$, if $P = \textsc{in}$;
  - $(R, \mathfrak{c}) = (R, \mathfrak{i})$, if $P = \textsc{with}$;
  - $(R, \mathfrak{c}) = (R, \mathfrak{b})$, if $P = \textsc{for}$.

  In case $(R, \mathfrak{d})$ is already in $ar^{*}(ar(R))$, we denote it $(R, \mathfrak{d}^2)$.[30]

How it works? Let Q be a 3-place relation with a relation scheme $Q.\mathfrak{a}, Q.\mathfrak{o}, Q.\mathfrak{g}$ (or simply $Q(\mathfrak{a}, \mathfrak{o}, \mathfrak{g})$). We make new relations out of Q by adding prepositions as follows:

- $Q \cdot \textsc{with}$ with a relation scheme $Q.\mathfrak{a}, Q.\mathfrak{o}, Q.\mathfrak{g}, Q.\mathfrak{i}$ (or simply $Q(\mathfrak{a}, \mathfrak{o}, \mathfrak{g}, \mathfrak{i})$, 4-place relation with *instrument* attribute added);
- $Q \cdot \textsc{for}$ with a relation scheme $Q.\mathfrak{a}, Q.\mathfrak{o}, Q.\mathfrak{g}, Q.\mathfrak{b}$ (or simply $Q(\mathfrak{a}, \mathfrak{o}, \mathfrak{g}, \mathfrak{b})$, 4-place relation with *beneficiary* attribute added);
- $Q \cdot \textsc{in}$ with a relation scheme $Q.\mathfrak{a}, Q.\mathfrak{o}, Q.\mathfrak{g}, Q.\mathfrak{l}$ (or simply $Q(\mathfrak{a}, \mathfrak{o}, \mathfrak{g}, \mathfrak{l})$, 4-place relation with *location* attribute added);
- $Q \cdot \textsc{with} \cdot \textsc{for}$ with a relation scheme $Q.\mathfrak{a}, Q.\mathfrak{o}, Q.\mathfrak{g}, Q.\mathfrak{i}, Q.\mathfrak{b}$ (or simply $Q(\mathfrak{a}, \mathfrak{o}, \mathfrak{g}, \mathfrak{i}, \mathfrak{b})$, 5-place relation with *beneficiary* attribute added), and so on.

When added to a predicate, a preposition expands the predicate's relation scheme with an attribute of a type connected with it and increases the predicate's arity. We can add a preposition to any predicate, either as a standalone or as a constituent of a joined predicate, e.g.:

[30] To be able to handle examples such as 15 and keep the requirement that all attributes in relation scheme should be different, in case a relation scheme already contains an attribute associated with a preposition, we will distinguish repeating attributes with indexes (see Definition 1 in Appendix). Thus a relation scheme for *Made a sweater for for* in 15 is as follows: $M.\mathfrak{a}, M.\mathfrak{o}, M.\mathfrak{b}, M.\mathfrak{b}^2$.

47. Playing rugby, Oscar was admired by thousands of girls.

47′. Playing rugby *for* his country, Oscar was admired by thousands of girls.
47″. Playing rugby *for* his country *with* artificial legs, Oscar was admired by thousands of girls.
47‴. Playing rugby *for* his country *with* artificial legs, Oscar was admired *in* Beijing by thousands of girls.

I will now show how compound predicates are built from basic predicates using the operation of adjunctive join (our second inductive step of predicate construction). Since predicates name relations, we can join them with other relations (ultimately as we will see, using the operation of natural join). Imagine having a metal toy construction set with metal beams (predicates) with named (by attribute names), regularly-spaced holes (argument places). We may hold two beams together with a 'bolt' (a common argument) in the following way:

– Join beam 1 with beam 2 by connecting any hole of beam 1 with the first hole of beam 2;
– Join two beams in the overlapping manner, renaming the first hole of the joined beam 2.

In an adjunctive join, we join the main predicate on one of its arguments with another predicate on its *agent* argument (cf. Rothstein (2004a): 72). Informally (and simplified): let R, S be relations with relation schemes $R(X)$, $S(\mathfrak{a}, Y)$. Adjunctive join is a concatenation of relations $R\frac{\mathfrak{d}}{S}$ with attributes XY. In adjunctive join $R\frac{\mathfrak{d}}{S}$ R is called 'primary relation', S is called 'secondary relation'. Notation '$R\frac{\mathfrak{d}}{S}$' is read as 'a predicate R is joined with a predicate S on attribute $\mathfrak{d}$ of R'.

Formally we need a rule of making a predicate symbol for a new relation together with a rule of making the relation scheme for this new relation (Definition 1):

– If $R, S \in \mathcal{S}$ and $\mathfrak{d} \in \mathbb{T}$, then $R\frac{\mathfrak{d}}{S} \in \mathcal{S}$[31]
– The relational scheme of $R\frac{\mathfrak{d}}{S}$ is $\langle ar^{*\prime}(ar(R)), ar^{*}({}_{2,\dots,n-1}ar(S))\rangle$, where $ar^{*\prime}(ar(R))$ denotes a sequence identical to $ar^{*}(ar(R))$ except $(RS.\mathfrak{d})$ belongs to this new

[31] One may wonder why we can't use a standard notation from relational algebra and instead of $R\frac{\mathfrak{d}}{S}$ write something like $R \bowtie \rho_{R.\mathfrak{d};/S.\mathfrak{a}}(S)$. I see the choice to use a relational FOL instead of relational algebra or SQL as arbitrary; ultimately I have chosen it because of its intuitiveness and closeness to FOL. Because of the closeness, language $\mathcal{L}$ contains neither $\bowtie$ ('bowtie') nor ρ symbols. And so, contrary to the standard relational algebra, $\alpha \bowtie \beta$, $\rho_{A/B}(R)$ are not expressions of $\mathcal{L}$. Symbol $\bowtie$ appears in a semantic metalanguage but the operation of a natural join is not a primitive operation of relational algebra and is defined in terms of Cartesian product and projection (see Definition 8). The same holds for the operation of attributes renaming (denoted by ρ symbol, see Atzeni and De Antonellis (1993): 36).

sequence instead of $(R, \mathfrak{d})$. In case $R = S$ we denote all $(S, \mathfrak{d}) \in ar^*({}_{2,\ldots,n-1}ar(S))$ as $(S', \mathfrak{d})$.[32]

How it works? Let us take 'give-yawning' predicate ('Yawning, she gave Tom a book') and 'buy-located-in' predicate ('I bought stewed chicken in creole sauce'). The former predicate is a result of adjunctive join in which a primary predicate G ('give') is joined on its 'agent' attribute with a secondary predicate Y ('yawn'); we write this join as $G\frac{\mathfrak{a}}{Y}$ (Fig. 1). The latter predicate is a result of adjunctive join in which a primary predicate B ('buy') is joined on its 'object' attribute with a secondary predicate L ('located in'); we write this join as $B\frac{\mathfrak{o}}{L}$ (Fig. 2).

The rule of making relation schemes for the adjunctive joins reflects the structure of the new relations - one of the attributes becomes 'an overlapping' attribute, the rest of attributes left unchanged. In that way we have predicates

$G\frac{\mathfrak{a}}{Y}$ with $GY.\mathfrak{a}, G.\mathfrak{o}, G.\mathfrak{b}$ relation scheme;
$B\frac{\mathfrak{o}}{L}$ with $B.\mathfrak{a}, BL.\mathfrak{o}, L.\mathfrak{l}$ relation scheme.

The syntactic rule of an adjunct join allows us to expand a predicate further by joining it with another predicate on *any* argument place of the initial predicate. So for example, we can expand a joined predicate 'buy-located-in' ($B\frac{\mathfrak{o}}{L}$) to 'buy-located-in-yawning' ($B\frac{\mathfrak{a}}{Y}\cdot\frac{\mathfrak{o}}{L}$), then to 'buy-located-in-yawning-for' ($B\frac{\mathfrak{a}}{Y}\cdot\frac{\mathfrak{o}}{L}\cdot\frac{\mathfrak{o}}{D}$) and so on, e.g. to 'Yawning, I bought chicken in creole sauce with my credit card, for my nephew who was giving a pack of battery to a fellow journalist in Beijing' (see Fig. 3).

Finally, we combine rules of joining predicates and prepositions. For example, the complex predicate from 'path' example 37 will be written as $P\cdot\text{IN}\cdot\frac{\mathfrak{l}}{L}$.

The notion of a formula is defined in a standard way.

Having explained the syntactic element of the theory, let us inspect the semantics in more details.

6.3 Semantics of Complex Predicates: Conceptual Design

6.3.1 Spurious Tuples and Relationship Relations

The association between attribute-types and domains is established by means of a function *dom* from the set of attribute-types $\mathbb{T}$ $ar^*(ar)$ onto non-empty subsets of D (see Definition 4). With this definition, it is possible for several attribute-types

[32] This requirement is needed to handle cases when we join a relation with itself, for example 'I am chasing a dog chasing a cat', 'I am seeing myself seeing myself'. At first glance such examples seem problematic because they violate the requirement of a relational algebra that all attributes in a joined relation must be distinct. To avoid this problem and be able to join a relation with itself (making so called 'a self-join operation') we need to (make a standard move and) rename one of the copies of a relation (make it relation R') and then join them.

to have overlapping (or the same) domains. So, for example, it is possible that there are overlapping domains of objects for attribute-types *object* and *instrument* - the attributes indicate different roles (same) objects have in a relation (cf. Elmasri and Navathe (2001): 63). The domain of a model is a non-empty set equal exactly to the range of variables of $\mathcal{A}$ (see Definition 4).

Basic relations are considered to be atomic predicates (see Definition 3). Prepositions are treated as syncategorematic expressions (they are not allowed be standalone predicates) which are either built-in to relations' schemes (PPs occurrences as arguments) or added to relations in order to express a relationship between a relation's participants and a set of objects from a domain assigned to prepositions' attribute (PPs occurrences as external verbal modifiers). Verbal internal modification is analyzed as adjunct predication - an adjunct predicate with incorporated prepositional meaning is joined on its *agent* argument with one of the arguments of a main predicate (cf. Rothstein (2004a): 72). Semantically, the extension of such a compound predicate (the result of adjunctive join) is *a subset* of a natural join of the primary and the secondary relations. In a natural join we combine tuples from two relations which have equal values in attributes with the same name and skip duplicate attributes (for example we can combine two 3-place relations with attributes $\mathfrak{a}, \mathfrak{o}, \mathfrak{g}$ and $\mathfrak{a}, \mathfrak{o}, \mathfrak{b}$ and because of skipping two columns-duplicates, receive a new 4-place relation with $\mathfrak{a}, \mathfrak{o}, \mathfrak{g}, \mathfrak{b}$ attributes, see Definition 8). The natural join sign $\bowtie$ ('bowtie') looks like a cross $\times$ with two vertical lines added, suggesting that a natural join is a Cartesian product plus two other operations (selection of rows with equal values for the common columns, followed by projection to delete redundant columns, Garcia-Molina et al. (2013): 198-199; Elmasri and Navathe (2001): 161).

Let me explain why adjunctive joins $R\frac{\mathfrak{b}}{S}$ cannot be analyzed (directly) as a natural join of R and S relations (the same holds for relations with several prepositional phrases $R{\cdot}P_1{\cdot}P_2$ - they are not equal to a natural join of $R{\cdot}P_1$ and $R{\cdot}P_2$ relations). Suppose that Jones is frying meat and frying potatoes. Simultaneously he is using a spoon (to fry potatoes) and a fork (to fry meat), see Tables 4 and 5.

To have *Fry-using* ($F\frac{\mathfrak{a}}{U}$) relation we may rename the attribute $U.\mathfrak{a}$ ('Use agent') in *Use* on $F.\mathfrak{a}$ 'Fry agent') and obtain a 3-place relation with the following relation scheme *Fry-using*(*Agent*, *Object*, *Instrument*). If we define the relation *Fry-using* as equal to the natural join of *Fry* and *Use* relations on 'agent' attribute ($Fry \bowtie \rho_{F.\mathfrak{a}/U.\mathfrak{a}} Use$)[33] then we will get the set of triples illustrated by Table 6.

It is clear that this set of triples cannot serve as an extension of *Fry-using* relation because this joined relation contains 'spurious' tuples (additional tuples that represent erroneous information, cf. Elmasri and Navathe (2001): 554)—it immediately follows that Jones fries meat not only using a spoon, but also using a fork (the same for frying potatoes). To get rid of spurious tuples we need to define a relationship between two relations, e.g. as in Table 7. Relation *Fry-using* is a relationship relation because each tuple in it represents a relationship instance that relates one tuple from *Fry* with one tuple from *Use* (cf. Elmasri and Navathe (2001): 289). The extension of the relationship relation *Fry-using* is a subset of joined relations it relates

[33] $\rho_{A/B}(R)$ denotes operation of renaming of attribute B of R on A, see Definition 10 in Appendix.

($[\![F\frac{\mathfrak{a}}{U}]\!] \subseteq [\![F]\!] \bowtie \rho_{F.\mathfrak{a}/U.\mathfrak{a}}[\![U]\!]$; the same holds for extensions of relations with several prepositional phrases, $[\![R{\cdot}P_1{\cdot}P_2]\!] \subseteq [\![R{\cdot}P_1]\!] \bowtie [\![R{\cdot}P_2]\!]$).

It is clear from this example that defining the extensions of compound predicates as subsets of natural joins of predicates-constituents, allows one to have the *Non-Entailment* semantic requirement (the proof is straightforward and stems from the definition of a natural join, see Statements in Appendix). While it is true that Jones fries potatoes *and* it is true that he fries using a spoon, it is false that Jones fries potatoes using a spoon - no such triple belongs to the relation *Fry-using*. It is clear as well that the *Drop* condition is also satisfied (it is nothing but a restriction of a relation to the selected columns) and the *Permutation* condition follows from the definition of a relations' tuple (a set of attribute:value pairs, an order of which is without importance),[34] In order to explain inferential connections among sentences involving prepositional phrases and adjunct predication, we need not refer to events or states. Consequently, '[t]he chief reason this extra complexity is forced upon us [...]' (Szabó (2003): 398) is repealed.

6.3.2 Explanatory Power: Non-entailment

The main motivation behind event semantics lies in its explanatory power. It is considered a good theory because, together with the truth-conditions, it provides their explanation. For example, why is it that the truth of (a)'Brutus stabbed Caesar with a knife' and (b) 'Brutus stabbed Caesar in the chest' does not guarantee the truth of (c) 'Brutus stabbed Caesar in the chest with a knife'? This is because (a) and (b) may describe two different events: a stabbing with a knife (which might not have been done in the chest) and a stabbing in the chest (which might not have been done with a knife). Similarly, a theory based on RMD should be considered a good theory, as it provides an explanation of why two pieces of information can/cannot be joined together.

Consider an example. Suppose that two astronauts, Dave and Frank, have been flying on a spaceship equipped with two supercomputers, Hal and Sal. Mission Control suspects that the computers have gone out of control. Unfortunately, the connection with the spaceship is lost but Mission Control still has access to the computers' logs and is able to check what data have been added to Hal and Sal and which astronaut has interacted with the computers. Based on this knowledge, Mission Control tries to figure out what happened on the spaceship. So far, Mission Control has the following information (reflected in Tables 8 and 9):

[34] Statements A-H (see Statements in Appendix), expressing *Reordering Drop* and *Non-Entailment*, are not only logical truths, but analytic as well because ultimately concern predicates' *relation schemes*. The relation schemes of compound predicates $R\frac{\mathfrak{b}}{S}$, $R{\cdot}P$ are sets of theta-roles (formally representing the part of the predicates' meaning which expresses the roles domains have in a relation named by the predicate). We may say that the meaning of compound predicates *incorporates* the meaning of the constituents: the relation schemes *XY* and *XZ* of predicate-constituents *are parts* of the relation scheme *XYZ* of a compound predicate (therefore *Drop* and *Non-Entailment* expresses analytic truth); because *XYZ* is an unordered set, it can be freely reordered (resulting in the same set, therefore *Reordering* expresses analytic truth as well). I am grateful to Justyna Grudzińska-Zawadowska for pointing attention to the issue.

Fig. 1 'Give-yawning'

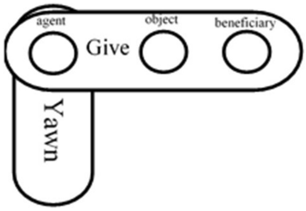
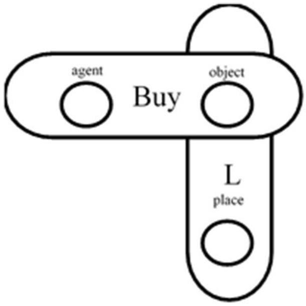


Fig. 2 'Buy-located-in'



1. Dave has uploaded data to Sal;
2. Dave has uploaded data to Hal;
3. Frank has uploaded data to Sal;
4. Sal processed data to calculate Jupiter's orbit;
5. Hal processed data to calculate Jupiter's orbit.

Mission Control concludes that this is what happened on the spaceship:

I. Dave has uploaded data to Sal to calculate Jupiter's orbit
(Justification: Dave has uploaded data to Sal and the only data has been uploaded to Sal concerns Jupiter's orbit);
II. Dave has uploaded data to Hal to calculate Jupiter's orbit
(Justification: Dave was the only person who has interacted with Hal and the only data that has been uploaded to Hal concerns Jupiter's orbit);
III. Frank has uploaded data to Sal to calculate Jupiter's orbit
(Justification: Frank has had an interaction with Sal and the only data that has been uploaded to Sal concerns Jupiter's orbit).

Despite the fact that (4) is a part of the description of two different events, Mission Control has the right to join pieces of information (1) and (4), as well as (3) and (4) together and draw conclusions I-III, because of the functional dependencies present in (1)–(5). If we represent pieces (1)–(5) schematically as a table with columns *Uploader*, *Computer*, *Planet* (Table 10), then it becomes clear that the values in the *Uploader* column functionally determine values in *Planet* column, as well as values in *Computer* functionally determine values in *Planet* column. The presence of

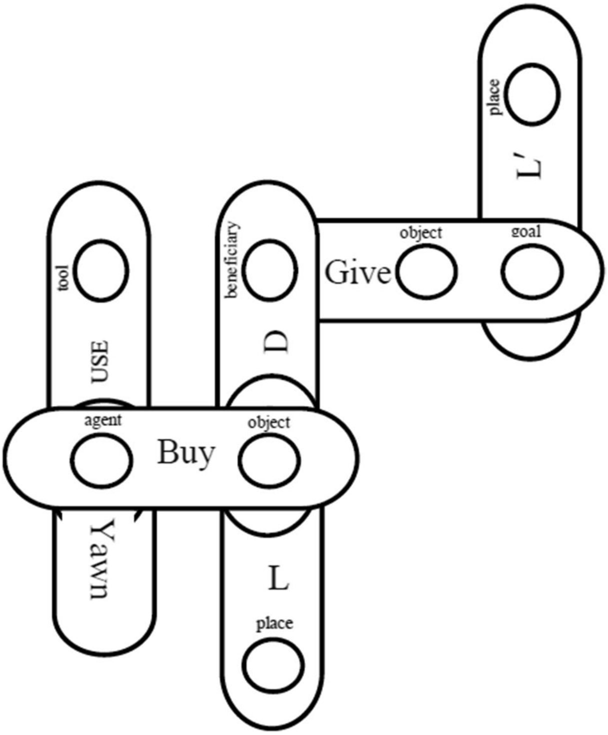


Fig. 3 Complex predicate with several adjuncts

a (specific) functional dependency between pieces of information is `a sufficient condition for the information being joined in a valid way` (cf. Song and Jones (1995); Atzeni and De Antonellis (1993)).[35]

Contrast this scenario with another in which, in addition to information (1)-(5), Mission Control receives the following information reflected in Tables 11 and 12.

Note that Tables 8, 11, and 12 constitute *all three* binary projections of a ternary relation *Uploader*, *Computer*, *Planet*. Mission Control concludes:

I. Dave has uploaded data to Sal to calculate Saturn's orbit.
   (Justification: Sal was the only computer used to calculate Saturn's orbit, Dave has had an interaction with Sal and has uploaded the data needed for calculating Saturn's orbit);
II. [*remains the same*];

[35] Using the tableu method (Theorem 4.4, Atzeni and De Antonellis (1993): 144–145) it is possible to prove that relation R(*Uploader*, *Computer*, *Planet*) in Table 10 can be losslessly decomposed on R_1(*Uploader*, *Computer*) and R_2(*Computer*, *Planet*). Lossless decomposition means that R=$R_1 \bowtie R_2$, that is, it is possible to decompose a complex relation on several projections, and these projections joined together will generate this very relation with no spurious tuples (cf. Chan and Atzeni (1992); Elmasri and Navathe (2001): 535).

Table 4 'Fry' relation

Agent	Object
Jones	Potatoes
Jones	Meat

Table 5 'Use' relation

Agent	Instrument
Jones	Spoon
Jones	Fork

Table 6 $Fry \bowtie \rho_{F.a/U.a} Use$

Agent	Object	Instrument
Jones	Potatoes	Fork
Jones	Meat	Fork
Jones	Potatoes	Spoon
Jones	Meat	Spoon

Table 7 'Fry-using' relation

Agent	Object	Instrument
Jones	Potatoes	Fork
Jones	Meat	Spoon

III. [*remains the same*].

But what Mission Control can neither confirm nor exclude is the possibility that Dave has uploaded data to Sal to calculate Jupiter's orbit - Dave has used both computers and has uploaded data to calculate orbits of both planets and Sal contained data for calculation orbits of both planets. The moral of the story is that even if *all* 2-argument projections are known, it is possible that they are *the same for two different* 3-argument relations describing two different scenarios (Tables 13 and 14).

That is why in situation in which there are no functional dependencies between pieces of known information we refrain from joining them because it is possible that such joining will result in erroneous information (being a part of a description of a wrong scenario). This potential to receive spurious information explains the *Non-Entailment* constraint.[36]

[36] Larson (2010: 701) objected to Pylkkänen's (2002) account of low applicative constructions (constructions in which the possession relation holds not between an event and an individual, but between two individuals) that such account results in an unwanted entailment. For example, assume that John wrote a letter and Bill gave it to Mary. If we analyze the conjunction 'John wrote that letter and Bill gave it to Mary' as a conjunction in which the relation of possession connects Mary and the letter, not Mary and the writing event, we must draw an undesirable conclusion that John wrote the letter to Mary. Intui-

6.4 Semantics of Complex Predicates: Logical Design

6.4.1 Compositionality

So far, there are entities of two primitive types in our semantics, atomic predicates and prepositions. Prepositions are treated as operators on relation schemes, which take a relation scheme as an argument and return another with a new attribute added (e.g. 'IN' adds a *location* attribute, 'WITH' adds an *instrument* attribute). Semantically, a compound predicate (a composition of atomic predicates, atomic predicates and prepositions, or both) is interpreted as a relationship relation(-s) between its components. This relationship relation is a subset of a natural join between the relations it relates. In that way all relationship relations (extensions of compound predicates) appear as *semantically primitive* - despite the fact they depend on extensions of relations they relate, they cannot be obtained from these extensions in a straightforward way (i.e. as their joins).[37] We return to the main question mentioned in the Introduction: if predicates have a recursive syntax, does they have a recursive semantics as well?

We may consider the function version of compositional semantics (Pagin and Westerståhl (2010): 254) and define extension of predicates as a restriction of F to predicates' relation schemes (Definition 4), that is, for each predicate symbol P, $[\![P]\!] = F|_{P.\overline{n}}$, where $P.\overline{n}$=$ar^*(ar(P))$. But is there any way to have a recursive semantics? The answer is positive – all we need to do is to *force* functional dependencies in order to avoid spurious tuples in natural joins.

Indeed, a recursive semantics can be achieved by forcing a functional dependency for relation F. If we add an additional column with attribute *id* to the F relation and place in it a unique index for every row (say, index i of f_i, see Definition 9 in Appendix), then the new relation H obtained by this operation has a lossless join property

Footnote 36 (continued)

tively, we cannot infer from the facts that John wrote a letter and it came in Mary's possession that John wrote the letter to Mary. I agree with Larson - in general ternary relations have no lossless join property (which means that they cannot be obtained as a result of joining their two binary projections and even all three - you may have additional spurious tuples, cf. Song et al. (1993). For example, a ternary relation from Scenario 1 (Table 13) cannot be obtained as a result of joining *any* of its binary projections). In Larson's example the resulting joined relation (from joining 'write' and 'to-the-possession' relations) suggests a relationship between objects, but in reality such relationship does not exist. This type of connection trap is called a *chasm* (Sumathi and Esakkirajan (2007): 59-60). Larson's objection, however, does not apply to the account presented here because I join projections of a universal relation H which has a lossless join property (for a proof see Lemma 1.1 in Appendix).

[37] A relationship relation may be represented as well as a value of a choice function (Pörn 1982). It was noted van Fraassen (1973), Fulton (1979) that there is a problem with considering an extension of a predicate with an adjunct (a modified predicate) to be a value of a choice function defined on the family of subsets of the predicate extension. Because it is a function, in the case of two co-extensive predicates, a choice function would necessarily pick up the same subsets (see Author (0002) for details). This objection does not apply here because relations are considered as sets of 'attribute: value' pairs and in case of two co-extensive predicates P and Q, these sets trivially differ.

(for a proof see Lemma 1.1 in Appendix). Because of this property, it is possible to define extensions of compound predicates as natural joins of respective columns from H in a fully recursive way (ρ stands for rename operation, see Definition 10 in Statements in Appendix, and $R.\mathfrak{c}$ stands for an attribute added by a preposition to the relation scheme of a predicate R):

$$[\![Q]\!] = \begin{cases} H|_{id,Q.\bar{n}}, \text{ where } Q.\bar{n} = ar^*(ar(Q)) & \text{if } Q \text{ is atomic;} \\ [\![R]\!] \bowtie H|_{id,R.\mathfrak{c}}, & \text{if } Q = R{\cdot}P; \\ \rho_{RS.\mathfrak{d}/R.\mathfrak{d}}[\![R]\!] \bowtie H|_{id,RS.\mathfrak{d}} \bowtie \rho_{RS.\mathfrak{d}/S.\mathfrak{a}}[\![S]\!] & \text{if } Q = R\frac{\mathfrak{d}}{S}. \end{cases}$$

For a better understanding of how semantic definitions work, consider 2-place predicates *Sing*, *Use*, 3-place compound predicates *Sing*·IN, *Sing*·*Using*, *Sing*·FOR, 4-place compound predicates *Sing*·IN·*Using*, *Sing*·IN·FOR, *Sing*·*Using*·FOR, 5-place compound predicate *Sing*·IN·*Using*·FOR and consider the following story about singing. Suppose there were three singing fellows, Jones, Smith and Brown. Jones and Brown sing 'Hallelujah' and Smith sings 'Yesterday'. Jones sings simultaneously in a bar and on YouTube, Brown sings on YouTube and we do not know where Smith sings. In a bar, Jones sings using a mike and he sings for the charity organization YMCA. Beside a mike, Jones uses a pen (but not for singing). Brown sings for Jones, and Smith sings for himself. Let us reflect this story by creating the following model which represent extensions of our relations (see Table 15).

By Definition 4 (relation F) we list all attributes which appear in our relations, that is, $S.\mathfrak{a}, S.\mathfrak{o}, U.\mathfrak{a}, U.\mathfrak{i}, S.\mathfrak{l}, S.\mathfrak{b}$, $SU.\mathfrak{a}$ ('singing agent', 'singing object', 'using agent', 'using instrument', 'singing location', 'singing beneficiary', 'singing using agent'). We start to populate our relation F with values from the first row. For any new information (not already contained in the previous rows) we start a new row, and in case information is missing or unknown for some of the attributes, we assign − (*null*) to them.Value − (*null*) represents a missing value (mappings f_i are partial) and is interpreted as 'a value is not applicable' (Elmasri and Navathe (2001): 116).

By Definition 1 atomic and compound predicates have relation schemes as in Table 16. Extension of the predicates is defined as a restriction of F to predicates' relation schemes. However, the extension may be obtained in a recursive way, by joining (by a natural join operation) columns of H for the respective attributes.

What will be the truth conditions for 'John is singing 'Hallelujah' using a mike in a bar for YMCA'? The sentence is true iff (Definition 7) there exists a 5-tuple of attribute-value pairs in $[\![S\frac{\mathfrak{a}}{U}{\cdot}\text{IN}{\cdot}\text{FOR}]\!]$, and the value for *agent* attribute is John, and the value for *object* attribute is 'Hallelujah' etc.

In the next section I will resolve problematic examples from Sects. 4 and 5.

6.4.2 Solving Problematic Examples

From a syntactic point of view, there are two ways of constructing compound predicates. Either we 'go along' with a predicate R and add a preposition P to it (having $R{\cdot}P$, modification of a predicate as a whole), or we 'go orthogonally' and concatenate a predicate R with a predicate S on an attribute $\mathfrak{d}$ of R (having an adjunctive join

Table 8 Mission control data (part 1)

Uploader	Computer
Dave	Sal
Dave	Hal
Frank	Sal

Table 9 Mission Control Data (part 2)

Computer	Planet
Sal	Jupiter
Hal	Jupiter

Table 10 Data received by Mission Control

Uploader	Computer	Planet
Dave	Sal	Jupiter
Dave	Hal	Jupiter
Frank	Sal	Jupiter

Table 11 MC data (part 3)

Uploader	Planet
Dave	Jupiter
Dave	Saturn
Frank	Jupiter

Table 12 MC Data (updated part 2)

Computer	Planet
Sal	Jupiter
Hal	Jupiter
Sal	Saturn

$R\frac{\mathfrak{d}}{S}$). If we modify the predicate as a whole, we increase its arity by adding a new attribute to its relation scheme; the extension of such a complex predicate is a natural join of the extension of the predicate and two columns for two attributes from a universal relation, that is, of *id* attribute (which secures a lossless join) together with the column for the new attribute added by a preposition, $[\![R{\cdot}P]\!] = [\![R]\!] \bowtie H|_{id,R.\mathfrak{d}}$. Note that in case of examples of blocked reordering, such as (15) repeated here,

15. Bill made a sweater *for* Mary *for* Miles. (Gawron (1986): 371)

Table 13 Scenario 1

Uploader	Computer	Planet
Dave	Sal	Saturn
Dave	Hal	Jupiter
Frank	Sal	Jupiter

Table 14 Scenario 2

Uploader	Computer	Planet
Dave	Sal	Saturn
Dave	Hal	Jupiter
Frank	Sal	Jupiter
Dave	Sal	Jupiter

we obtain a modified predicate $Make{\cdot}\text{FOR}{\cdot}\text{FOR}$ with a relation scheme in which there are two 'beneficiary' attributes with different indexes (by Definition 1), $Make{\cdot}\text{FOR}{\cdot}\text{FOR}(M.\mathfrak{a}, M.\mathfrak{o}, M.\mathfrak{b}, M.\mathfrak{b}^2)$. It is possible that attributes $M.\mathfrak{b}, M.\mathfrak{b}^2$ have different values in a universal relation, for example as in Table 17.

While we can reorder the attributes *Beneficiary* and *Beneficiary*^{2} in the relation scheme for *Make*, as well as reorder attributes *with* their values in the tuples (reorder *Beneficiary*:*Mary* and *Beneficiary*^{2}:*Miles*), we cannot keep an order between attributes (with respect to an index) and permute their values (having *Beneficiary*:*Miles* and *Beneficiary*^{2}:*Mary*), because we may obtain a spurious tuple (a tuple which does not belong to a relation) as a result. This explains why the reordering between iterated prepositions is blocked in natural language.

Now let us turn to a semantic definition of an adjunctive join. As I have noted in Sect. 6.3.1, semantically an adjunctive join is a relationship relation between the relations-constituents. This relationship relation encodes the following information: (i) what tuples from two relations are related and (ii) what attribute from a primary relation is considered a 'juncture' attribute, and that (iii) the relations share a 'juncture' point (the same value for junction attribute). The idea behind the definition of the extension of an adjunctive join between relations 1 and 2 can be schematically represented as

$$\boxed{1}\frac{\bigcirc}{\boxed{2}} = \boxed{1} \bowtie \bigcirc \bowtie \boxed{2}$$

$$[\![R\tfrac{\mathfrak{b}}{S}]\!] = [\![R]\!]^* \bowtie H|_{id,RS.\mathfrak{a}} \bowtie [\![S]\!]^*$$

that is, as a join of extensions of relations 1 and 2 with a 'node' of concatenation. A complex relation $\boxed{1}\frac{\bigcirc}{\boxed{2}}$ may itself be an argument of a more complex relation, e.g. $\boxed{\boxed{1}\frac{\circ}{\boxed{2}}} \bowtie \boxed{\boxed{2}\frac{\circ}{\boxed{3}}} \bowtie \ldots$ or $\boxed{\boxed{1}\frac{\circ}{\boxed{2}}} \bowtie \boxed{\boxed{1}\frac{\circ}{\boxed{3}}} \bowtie \ldots$. Such a semantics of an adjunctive join is compositional: denotations of basic constituents (values assigned to attributes) are context-free, the extensions of complex relations are computed by function application and function composition in a way determined (only) by the syntactic structure of the complex relation and finally syntactic components of a complex relation can be computed independently of the relation.

Let us now see how the semantic definition of an adjunctive join works in practice, starting from the example of self-join 40 repeated here,

40. I am seeing myself seeing myself.

Assume that we have a model with a universal relation F={f_1} such that a mapping f_1 assigns values to attributes in a relation scheme for *See* as in Table 18.

When we make an adjunctive join of a relation with itself, we mark the joined relation with the prime symbol $'$, $S\frac{\mathfrak{o}}{S'}$. The relation scheme of $S\frac{\mathfrak{o}}{S'}$ is $S.\mathfrak{a}, SS.\mathfrak{o}, S'.\mathfrak{o}$ (Definition 1). Let us see if 40, written as $S\frac{\mathfrak{o}}{S'}(I,I,I)$, is true (I will assume for simplicity that indexical 'I' denotes John).

40$'$. $\vDash_{I,\rho} S\frac{\mathfrak{o}}{S'}(I,I,I)$ iff (Definition 7) $(S.\mathfrak{a}: \|I\|^{\rho}_{I}, SS.\mathfrak{o}: \|I\|^{\rho}_{I}, S'.\mathfrak{o}: \|I\|^{\rho}_{I}) \in [\![S\frac{\mathfrak{o}}{S'}]\!]$ iff (Definition 4) $(S.\mathfrak{a}: \|I\|^{\rho}_{I}, SS.\mathfrak{o}: \|I\|^{\rho}_{I}, S'.\mathfrak{o}: \|I\|^{\rho}_{I}) \in F|_{S.\mathfrak{a},SS.\mathfrak{o},S'.\mathfrak{o}}$, iff $(S.\mathfrak{a}: \|I\|^{\rho}_{I}, SS.\mathfrak{o}: \|I\|^{\rho}_{I}, S'.\mathfrak{o}: \|I\|^{\rho}_{I}) \in \{S.\mathfrak{a}: \text{John}, SS.\mathfrak{o}: \text{John}, S'.\mathfrak{o}: \text{John}\}$. TRUE

Now I will show that adjunctive joins are non-transitive. Let us take example (46) of non-transitive iteration of modifiers and consider a mapping f_1 which assigns values for attributes of 'Confirmed', 'Hired' and 'Defensive coordinator' relations as in Table 19.

Let 't' stand for 'Teryl Austin' and let compute the values of the following sentences 1-3:

1. $C\frac{\mathfrak{a}}{H\frac{\mathfrak{a}}{D}}(t)$ 'Teryl Austin was confirmed as hired as defensive coordinator'.
2. $C\frac{\mathfrak{a}}{H}(t)$ 'Teryl Austin was confirmed as hired'.
3. $C\frac{\mathfrak{a}}{D}(t)$ 'Teryl Austin was confirmed as defensive coordinator'.

Let us start from 2:

2. $\vDash_{I,\rho} C\frac{\mathfrak{a}}{H}(t)$ iff (Definition 7) $CH.\mathfrak{a}: \|I\|^{\rho}_{t} \in [\![C\frac{\mathfrak{a}}{H}]\!]$ iff (Definition 4) $CH.\mathfrak{a}: \|I\|^{\rho}_{t} \in F|_{CH.\mathfrak{a}}$ iff $CH.\mathfrak{a}: \|I\|^{\rho}_{t} \in \{CH.\mathfrak{a}: \text{Teryl}\}$. TRUE

Now let us compute 1:

1. $\vDash_{I,\rho} C\frac{\mathfrak{a}}{H\frac{\mathfrak{a}}{D}}(t)$ iff (Definition 7) $CHD.\mathfrak{a}: \|I\|^{\rho}_{t} \in [\![C\frac{\mathfrak{a}}{H\frac{\mathfrak{a}}{D}}]\!]$ iff (Definition 4) $CHD.\mathfrak{a}: \|I\|^{\rho}_{t} \in F|_{CHD.\mathfrak{a}}$ iff $CHD.\mathfrak{a}: \|I\|^{\rho}_{t} \in \{CHD.\mathfrak{a}: \text{Teryl}\}$. TRUE

Finally, let us compute 3:

3. $\vDash_{I,\rho} C\frac{\mathfrak{a}}{D}(t)$ iff (Definition 7) $CD.\mathfrak{a}: \|I\|^{\rho}_{t} \in [\![C\frac{\mathfrak{a}}{D}]\!]$ iff (Definition 4) $CD.\mathfrak{a}: \|I\|^{\rho}_{t} \in F|_{CD.\mathfrak{a}}$ iff $CD.\mathfrak{a}: \|I\|^{\rho}_{t} \in \emptyset$. FALSE

The falsity of 3 exemplifies that the iteration of modifiers in the semantics presented is non-transitive: from the truth of 1, it does not follow that 3 is true as well. Note that 'confirmed as hired' is not the same predicate as 'hired as confirmed', that is, $[\![C\frac{\mathfrak{a}}{H}]\!] \neq [\![H\frac{\mathfrak{a}}{C}]\!]$ (so the presented account deals with examples such as 41 and 41′).

Finally, I want to comment on the semantics of *simpliciter* mentioned in Sect. 2. Intuitively, from the sentence you get after applying *Drop*, you cannot conclude that something is done *simpliciter*. For example, from 'She likes him as a quarterback' it follows that 'She likes him [in some way]' but not that 'She likes him *simpliciter*' (or 'She *simply*' likes him'). As I said earlier, a relation in RMD is understood as a set of mappings from a relation scheme to the union of attribute domains. It is possible that in such a set we will find a subset of mappings such that for all attributes added by modifiers, the mappings return value *null*. My hypothesis is that *simpliciter* is an operator on the extension of a predicate which returns such subset. I propose defining it as follows:

Simpliciter, $\mathfrak{S}$ Let R be a relation with attributes X. $[\![\mathfrak{S}R]\!] = \{f_i|_X \in [\![R]\!] : f_i(R.\mathfrak{d})$ is undefined for all $R.\mathfrak{d} \notin X\}$.

Under this interpretation, 'She simply likes him' is true iff she likes him without any qualification.

7 Conclusions

I propose understanding a predicate as a recursive structure, with its basic cases (reflecting three kinds of natural-language predicates) and two inductive steps (reflecting two ways of constructing compound predicates, either 'go along' with a predicate and expand its arity with a preposition or 'go orthogonally' and concatenate predicates in adjunctive join). Semantically, the extension of a predicate is a realization of it's relation scheme, where a relation scheme is understood as a set of theta-roles connected with a predicate which provide a truth-conditionally relevant information about objects' ways of participation in the relation named by the predicate. Compound predicates are no exception; their extension is a relation, that is, a set of mappings from the attributes of the relation's scheme to relative domains. Due to the forced functional dependency (thanks to which relation H has a lossless join property) *every* such a relation can be obtained in a recursive compositional way by joining columns for respective attributes. However, despite the possibility of semantic recursion, compound predicates are semantically primitive in a deep sense as they denote a relationship between relation-components (so such a recursion is rather technical and without philosophical relevance).

If we represent theta-roles as attribute names of a relation, we preserve all of the conditions which theta-roles obey (except that which follow from the assumed homomorphism from events to objects). First of all, attributes have a semantic interpretation and constitute a part of a relation scheme (in the same way as theta-roles constitute a part of a theta-grid). They are truth-conditionally relevant and their meaning is semantically basic, expressing the natural roles individuals play in a relation (independence condition). Every argument of a relation has an attribute

Table 15 Relation F

	$S.\mathfrak{a}$	$S.\mathfrak{o}$	$U.\mathfrak{a}$	$U.\mathfrak{i}$	$S.\mathfrak{l}$	$S.\mathfrak{b}$	$SU.\mathfrak{a}$
f_1	Jones	Hallelujah	Jones	mike	bar	YMCA	Jones
f_2	Brown	Hallelujah	Jones	pen	YT	Jones	–
f_3	Smith	Yesterday	Brown	mike	YT	Smith	–
f_4	Jones	Hallelujah	Brown	pen	–	–	–

Table 16 Extensions of predicates

Natural-language relations	Predicates of $\mathcal{L}$ with relation schemes	Extensions of predicates
Sing	$S(S.\mathfrak{a}, S.\mathfrak{o})$	$H\vert_{id,S.\mathfrak{a},S.\mathfrak{o}}$
Use	$U(U.\mathfrak{a}, U.\mathfrak{i})$	$H\vert_{id,U.\mathfrak{a},U.\mathfrak{i}}$
Sing using	$S\frac{\mathfrak{a}}{U}(SU.\mathfrak{a}, S.\mathfrak{o}, U.\mathfrak{i})$	$[\![S]\!]^* \bowtie H\vert_{id,SU.\mathfrak{a}} \bowtie [\![U]\!]^{*a}$
Sing in	$S{\cdot}\mathrm{IN}(S.\mathfrak{a}, S.\mathfrak{o}, S.\mathfrak{l})$	$[\![S]\!] \bowtie H\vert_{id,S.\mathfrak{l}}$
Sing for	$S{\cdot}\mathrm{FOR}(S.\mathfrak{a}, S.\mathfrak{o}, S.\mathfrak{b})$	$[\![S]\!] \bowtie H\vert_{id,S.\mathfrak{b}}$
Sing using for	$S\frac{\mathfrak{a}}{U}{\cdot}\mathrm{FOR}(SU.\mathfrak{a}, S.\mathfrak{o}, U.\mathfrak{i}, S.\mathfrak{b})$	$[\![S\frac{\mathfrak{a}}{U}]\!] \bowtie H\vert_{id,S.\mathfrak{b}}$
Sing using in	$S\frac{\mathfrak{a}}{U}{\cdot}\mathrm{IN}(SU.\mathfrak{a}, S.\mathfrak{o}, U.\mathfrak{i}, S.\mathfrak{l})$	$[\![S\frac{\mathfrak{a}}{U}]\!] \bowtie H\vert_{id,S.\mathfrak{l}}$
Sing in for	$S{\cdot}\mathrm{IN}{\cdot}\mathrm{FOR}(S.\mathfrak{a}, S.\mathfrak{o}, S.\mathfrak{l}, S.\mathfrak{b})$	$[\![S{\cdot}\mathrm{IN}]\!] \bowtie H\vert_{id,S.\mathfrak{b}}$
Sing using in for	$S\frac{\mathfrak{a}}{U}{\cdot}\mathrm{IN}{\cdot}\mathrm{FOR}(SU.\mathfrak{a}, S.\mathfrak{o}, U.\mathfrak{i}, S.\mathfrak{l}, S.\mathfrak{b})$	$[\![S\frac{\mathfrak{a}}{U}{\cdot}\mathrm{IN}]\!] \bowtie H\vert_{id,S.\mathfrak{b}}$

[a] $[\![S]\!]^*, [\![U]\!]^*$ stand for relations identical to $[\![S]\!], [\![U]\!]$ except attribute $S.\mathfrak{a}$ in all tuples $t \in [\![S]\!]$ is renamed on $SU.\mathfrak{a}$ and attribute $U.\mathfrak{a}$ in all tuples $t \in [\![U]\!]$ is renamed on $SU.\mathfrak{a}$. See Statements in Appendix. For a proof that relations listed in the middle and rightmost columns of Table 16 converge (if we restrict relations in the rightmost column to all attributes of their relation schemes except *id*), see Lemmas 1.2, 1.3 in Appendix

(completeness condition). The uniqueness condition is implied by a standard requirement of relational algebra that all attribute names in a particular relation must be distinct. Distinctness is achieved by using '′' convention (which became necessary in the case of self-joined relations). However, as follows from examples 15 and 19 with prepositional iteration, uniqueness condition is not preserved in natural language which makes this requirement rather technical. Note that because an extension of a relation is a set of mappings from attributes to relative domains, it is possible that the extension contains several tuples which differ from each other only in a value for a particular attribute. Consequently, in cases where several objects fulfill the same thematic role (for example in the situation mentioned in note 4 in which two objects have been simultaneously touched) there be several tuples in a relation with different objects as values for the same attribute. Therefore, if we want to represent formally a situation of exactly one act of touching with two objects touched (that is, to count over acts), we cannot count elements of relation (tuples), because they are 'thinner' than needed and do not represent such acts (so probably 'nested' relations instead of 'flat' should be considered, cf. Atzeni and De Antonellis (1993): 21).

What do we obtain as a reward for such a treatment of theta-roles? We receive a semantics which accommodates internal and external modifiers, non-transitive

Table 17 Universal relation H for *Make for for* example

Id	Agent	Object	*Beneficiary*	*Beneficiary*^{2}	any other attribute
1	Bill	sweater	Mary	Miles	–

Table 18 Values of a mapping f_1 for attributes of 'See'

Id	$S.\mathfrak{a}$	$S.\mathfrak{o}$	$S'.\mathfrak{a}$	$S'.\mathfrak{o}$	$SS.\mathfrak{o}$	any other attribute
1	John	John	John	John	John	–

iteration of modifiers, self-joins, blocked reordering in case PPs modify different nodes ('sweater' example 15) and provides a compositional analysis of complex predication. The representation of theta-roles as attribute names of a relation allows us to use achievements of relational database theory for explanation semantic phenomena of natural language (e.g. I have used the theory of functional dependencies to explain why two pieces of information can/cannot be joined together (*Non-Entailment*)). As I have argued, a relational FOL is intuitive, close to a FOL and can be seen as 'a bridge language' between a FOL and query languages. The expressions of a relational FOL can (potentially) be translated to relational algebra expressions or SQL, which would allow us to operate with these three languages on the same relational model.[38]Further elaboration of ways natural-language expressions are be represented in RMD may allow going back to 'property-based' semantics in which properties named by predicates are properties of individuals, and not of events.

Finally, I would like to highlight something which is philosophically thought provoking. I treat variables in this paper as denoting individual domain elements. However, one may use other of the two equivalent versions of relational calculus called *tuple relational calculus*, whose variables denote tuples (Atzeni and De Antonellis (1993): 74). If we let variables denote mappings $f \in F$ that in turn assign individuals as its value to each relational attribute, then the theory proposed here becomes one of BOUND VARIABLES THEORIES. As it was proven by Dekker (Dekker 2004), under a few assumptions situations are isomorphic to assignment functions. The resemblance between assignment functions in tuple relational calculus and situations is much closer than isomorphism. Here is an observation. The column *id* in H is a set

[38] Codd (1972) has shown that relational calculus (a predicate calculus adapted to the relational model, cf. Abiteboul et al. (1996): 64) is essentially equivalent to relational algebra.
Let me provide a practical example and explain how to calculate extensions of compound predicates written as expressions of a relational algebra. Using notation $\rho_{R.\mathfrak{d} \leftarrow S.\mathfrak{a}}(S)$ instead of $\rho_{R.\mathfrak{d}/S.\mathfrak{a}}(S)$, one can use an online relational algebra calculator (created by J. Kessler and available here: https://dbis-uibk.github.io/relax/calc/local/uibk/local/0) and calculate extensions of compound predicates. The translation rules to relational algebra queries are as follows (assuming relation H as in Definition 9 in Appendix, $R\overline{n}$ is a relation scheme of R, $S\overline{m}$ is a relation scheme of S and $R.\mathfrak{c}$ is a relational attribute added by a preposition P):
If R is atomic: $[\![R]\!] \rightsquigarrow \Pi id, R\overline{n}\ (H)$;
If $R = R{\cdot}P$: $[\![R{\cdot}P]\!] \rightsquigarrow \Pi id, R\overline{n}\ (H) \bowtie \Pi id, R.\mathfrak{c}\ (H)$;
If $R = R\frac{\mathfrak{d}}{S}$: $[\![R\frac{\mathfrak{d}}{S}]\!] \rightsquigarrow \rho_{RS.\mathfrak{d} \leftarrow R.\mathfrak{d}}(\Pi id, R\overline{n}\ (H)) \bowtie \rho_{RS.\mathfrak{d} \leftarrow S.\mathfrak{a}}\ (\Pi id, S\overline{m}\ (H)) \bowtie \Pi_{id,RS.\mathfrak{d}}\ (H)$.

Table 19 Values of f_1 for attributes of 'hired as defensive coordinator' example

id	$C.\mathfrak{a}$	$H.\mathfrak{a}$	$D.\mathfrak{a}$	$CH.\mathfrak{a}$	$HC.\mathfrak{a}$	$HD.\mathfrak{a}$	$CHD.\mathfrak{a}$	$CD.\mathfrak{a}$	other
1	Teryl	Teryl	Teryl	Teryl	–	Teryl	Teryl	–	–

of numbers. Because for each row an assigned number is unique, we can rewrite this dependency as a function $\lambda j.f_j$ (a function which for each index j returns a tuple of F). Let us assume for a moment, that each attribute is a partial function from the set of numbers into a domain assigned to the attribute (e.g. $\lambda j.agent(j)$, $\lambda j.instr(j)$, etc.). Understood in such a way, attributes satisfy the following requirements for events and theta-functions: attributes-as-functions are partial, they are functions (a situation that different objects fulfill the same role in a tuple with the same number is excluded, it is possible to distinguish one object from another in a tuple by their role, two tuples are the same iff attributes-as-functions return the same values). What will be the truth-conditions for a sentence such as 'John is singing 'Hallelujah' in a bar with a mike for YMCA'? - Well, the sentence is true iff there exists an index j in *id* column of H and the value for *agent* function for j is John and the value for *object* function is 'Hallelujah' and the value for *location* function is a bar etc. The only divergence from the truth-conditions provided by the Neo-Davidsonian semantics (in this particular case) is that we do not need to use events and may employ indexes instead.

Appendix

A relational FOL $\mathcal{L}$

Definition 1 (*Alphabet of* $\mathcal{L}$) A first-order relational language $\mathcal{L} = (\mathcal{A}, \mathcal{W})$ is appropriate for representing compound predicates iff $\mathcal{A}$ satisfies the following conditions:

1. the set of constants in $\mathcal{A}$ is nonempty;
2. $\mathcal{A}$ contains the following symbols:

- a symbol = (called *equality*);
- a set of unary predicates (called *types*) of the form $\mathbb{T} = \{\mathfrak{a}, \mathfrak{o}, \mathfrak{g}, \mathfrak{l}, \mathfrak{i}, \mathfrak{b}\}$, where $\mathfrak{a}, \mathfrak{o}, \mathfrak{g}, \mathfrak{l}, \mathfrak{i}, \mathfrak{b}$ are interpreted as, respectively, *agent*, *object*, *goal*, *location*, *instrument*, *beneficiary*;
- a set of symbols $\mathbb{P} = \{\textsc{in}, \textsc{with}, \textsc{for}\}$ which correspond to prepositions;
- a set of symbols $\mathcal{S}$ defined as follows:

- $(\mathbb{P} \cup IV \cup TV \cup DV \cup \Gamma) \subseteq \mathcal{S}$, where IV, TV, DV are infinite sets of symbols for intransitive, transitive and ditransitive verbs, respectively; $\Gamma = \{L, U, D\}$, where L, U, D are interpreted as, respectively, *to be located in*, *to be used as a tool* and *to be destined for*;
- if $R \in \mathcal{S}$ and $P \in \mathbb{P}$ then $R^P \in \mathcal{S}$;
- if $R, S \in \mathcal{S}$ and $\mathfrak{d} \in \mathbb{T}$, then $R\frac{\mathfrak{d}}{S} \in \mathcal{S}$.

  Symbols of $\mathcal{S}$ are called *relation symbols*;
- ar: $\mathcal{S} \to \mathcal{S}^*$ is a function to the free monoid on $\mathcal{S}$ which prescribes an *arity* for each relation symbol as follows:
  - P for $P \in IV \cup \mathbb{P}$;
  - $\langle P, P\rangle$ for $P \in TV \cup \Gamma$;
  - $\langle P, P, P\rangle$ for $P \in DV$;
  - $\langle ar(R), P\rangle$ for $R^P \in \mathcal{S}$;
  - $\langle ar(R),\ {}_{n-1}ar(S)\rangle$ for $R\frac{\mathfrak{d}}{S} \in \mathcal{S}$, where ${}_{n-1}ar(S)$ is a n-1 sequence resulted from n-sequence $ar(S)$;
- X is the set of all finite sequences of members of $\mathcal{S} \times \mathbb{T}$ set. ar^*: $ar(\mathcal{S}) \to X$ is a function which for each element of $ar(\mathcal{S})$ returns an element from X as follows:
  - $(P, \mathfrak{a})$ for $P \in IV$;
  - $\langle (P, \mathfrak{a}), (P, \mathfrak{o})\rangle$ for $\langle P, P\rangle$ if $P \in TV$;
  - $\langle (P, \mathfrak{a}), (P, \mathfrak{o}), (P, \mathfrak{g})\rangle$ for $\langle P, P, P\rangle$ if $P \in DV$;
  - $\langle (P, \mathfrak{o}), (P, \mathfrak{l})\rangle$ for $\langle P, P\rangle$ if $P = L$;
  - $\langle (P, \mathfrak{a}), (P, \mathfrak{i})\rangle$ for $\langle P, P\rangle$ if $P = U$;
  - $\langle (P, \mathfrak{o}), (P, \mathfrak{b})\rangle$ for $\langle P, P\rangle$ if $P = D$;
  - $\langle ar^*(ar(R)), (R, \mathfrak{c})\rangle$ for $\langle ar(R), P\rangle$, where
    - $(R, \mathfrak{c}) = (R, \mathfrak{l})$, if $P = \text{IN}$;
    - $(R, \mathfrak{c}) = (R, \mathfrak{i})$, if $P = \text{WITH}$;
    - $(R, \mathfrak{c}) = (R, \mathfrak{b})$, if $P = \text{FOR}$.

    In case $(R, \mathfrak{d})$ is already in $ar^*(ar(R))$, we denote it $(R, \mathfrak{d}^*)$;
  - $\langle ar^{*'}(ar(R)), ar^*({}_{2,\dots,n-1}ar(S))\rangle$ for $\langle ar(R),\ {}_{n-1}ar(S)\rangle$, where $ar^{*'}(ar(R))$ denotes a sequence identical to $ar^*(ar(R))$ except $(R\frac{\mathfrak{d}}{S}, \mathfrak{d})$ belongs to this new sequence instead of $(R, \mathfrak{d})$. In case $R = S$ we denote all $(S, \mathfrak{d}) \in ar^*({}_{2,\dots,n-1}ar(S))$ as $(S', \mathfrak{d})$.

Definition 2 A value of $ar^*(ar(P))$, written as $P(\mathfrak{d}_1, \dots, \mathfrak{d}_n)$ or $(P.\mathfrak{d}_1), \dots, (P.\mathfrak{d}_n)$, where $\mathfrak{d}_i \in \mathbb{T}$, is called *a relation scheme* and each $(P.\mathfrak{d}_i)$ in the relation scheme is called *a relational attribute*.

Definition 3 Predicates $R^P, R\frac{\mathfrak{d}}{S}$ are called *compound predicates*, predicates other than $R^P, R\frac{\mathfrak{d}}{S}$ are called *atomic predicates*.

Notational convention: we write $R{\cdot}P$ instead of R^P. We write $R[\frac{\mathfrak{c}}{S}][\frac{\mathfrak{d}}{Q}]$ instead of $R\frac{\mathfrak{c}}{S}\frac{\mathfrak{d}}{Q}$. We write indexes instead of * in attributes as follows: for $R.\mathfrak{d}^*$ we write $R.\mathfrak{d}^2$, for $R.\mathfrak{d}^{**}$ we write $R.\mathfrak{d}^3$, etc. We write $(RS.\mathfrak{d})$ instead of $(R\frac{\mathfrak{d}}{S}.\mathfrak{d})$.

The terms of $\mathcal{A}$ are variables and constants of $\mathcal{A}$. Because we allow neither prepositions nor types be standalone predicates in $\mathcal{L}$ the definition of a formula is modified accordingly. A set $\mathcal{W}$ of well-formed formulae (WFFs) is the smallest set which satisfies the conditions:

1. if P is an n-ary predicate of $\mathcal{A}$ other than that of $\mathbb{P}$ and that of $\mathbb{T}$ and $t_1, \dots, t_n$ are terms of $\mathcal{A}$, then $\mathcal{W}$ includes $P(t_1, \dots, t_n)$. ($P(t_1, \dots, t_n)$ is called an atomic formula, or, if $t_1, \dots, t_n$ are constants, a *ground atomic formula*);
2. if α and β are in $\mathcal{W}$ and x is a variable, then the following are also in $\mathcal{W}$: $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\alpha \rightarrow \beta)$, $(\alpha \leftrightarrow \beta)$, $(\neg\alpha)$, $\exists x(\alpha)$, $\forall x(\alpha)$.

Axioms In $\mathcal{L}$, we have:

I. all the standard axioms for propositional calculus,
II. all the standard axioms for first-order logic.

Semantics for $\mathcal{L}$

Definition 4 (*Interpretation for $\mathcal{L}$*) An interpretation I for a relational first-order language $\mathcal{L} = (\mathcal{A}, \mathcal{W})$ is a quintuple $I = (D, dom, K, F, [\![\quad]\!])$, where:

1. $D \neq \emptyset$ is a set called the *domain* of I and equal exactly to the range of variables of $\mathcal{A}$;
2. dom is a mapping from $\mathbb{T}$ onto non-empty subsets of D (not necessary distinct, we will write $D_{\mathfrak{d}}$ for a value of dom for $\mathfrak{d} \in \mathbb{T}$);
3. K is a mapping from constants of $\mathcal{A}$ onto D, such that for each constant c, $K(c) \in D$;
4. F is a set of partial mappings which for each element $(R, \mathfrak{d}) \in \mathcal{S} \times \mathbb{T}$ return a value from $D_{\mathfrak{d}}$; $f(R, \mathfrak{d}) = f(R', \mathfrak{d})$ for any f and $(R, \mathfrak{d})$; $f(R\frac{\mathfrak{d}}{S}, \mathfrak{d}) = f(R, \mathfrak{d})$ if $f(R, \mathfrak{d}) = f(S, \mathfrak{a})$ and f is defined for $f(R\frac{\mathfrak{d}}{S}, \mathfrak{d})$;
5. $[\![\quad]\!]$ is a function from predicate symbols (predicates) of $\mathcal{A}$ onto restrictions of F (that is onto "actual relations"). Let $P.\bar{n}$ abbreviate $(P.\mathfrak{d}_1), \dots, (P.\mathfrak{d}_n)$. For each predicate symbol P, $[\![P]\!] = F|_{P.\bar{n}}$, where $P.\bar{n} = ar^*(ar(P))$.

$[\![P]\!]$ is called the extension of a symbol P in the interpretation I.

Definition 5 (Environment (valuation function) for variables) Given an interpretation $I = (D, dom, K, F, [\![\quad]\!])$ of $\mathcal{L} = (\mathcal{A}, \mathcal{W})$, let ρ be a mapping from the variables of $\mathcal{A}$ into D. For each variable $x \in Var(\mathcal{A})$, $\rho(x) \in D$.

ρ is called an *environment* for the variables of $\mathcal{A}$.

Definition 6 (Valuation of terms) Given an interpretation $I = (D, dom, K, F, [\![\quad]\!])$ of $\mathcal{L} = (\mathcal{A}, \mathcal{W})$ and an environment ρ, define the valuation of terms $\|\cdot\|_I^\rho$ a mapping from the set of terms of $\mathcal{L}$ onto D, such that:

$\|c\|_I^\rho = K(c)$, for every constant $c \in \mathcal{A}$, and
$\|x\|_I^\rho = \rho(x)$, for every variable $x \in Var(\mathcal{A})$.

Definition 7 (Satisfaction)

1. $\vDash_{I,\rho} P(t_1, \dots, t_n)$ iff $((P.\mathfrak{d}_1) : \|t_1\|_I^\rho, \dots, (P.\mathfrak{d}_n) : \|t_n\|_I^\rho) \in [\![P]\!]$, where $P.\overline{n}{=}ar^*(ar(P))$.
2. $\vDash_{I,\rho} \alpha \wedge \beta$ iff $\vDash_{I,\rho} \alpha$ and $\vDash_{I,\rho} \beta$.
3. $\vDash_{I,\rho} \alpha \vee \beta$ iff $\vDash_{I,\rho} \alpha$ or $\vDash_{I,\rho} \beta$.
4. $\vDash_{I,\rho} \neg\alpha$ iff $\nvDash_{I,\rho} \alpha$.
5. $\vDash_{I,\rho} \alpha \rightarrow \beta$ iff $\vDash_{I,\rho} \neg\alpha \vee \beta$.
6. $\vDash_{I,\rho} \alpha \leftrightarrow \beta$ iff $\vDash_{I,\rho} (\alpha \rightarrow \beta)$ and $\vDash_{I,\rho} (\beta \rightarrow \alpha)$.
7. $\vDash_{I,\rho} \forall x(\alpha)$ iff for all $d \in D$ $\vDash_{I,\rho[x\rightarrow d]} \alpha$ where $\rho[x \rightarrow d]$ denotes an environment identical to ρ except that this new environment maps the variable x to the domain element d.
8. $\vDash_{I,\rho} \exists x(\alpha)$ iff $\vDash_{I,\rho} \neg\forall x \neg(\alpha)$.
9. $\vDash_I \alpha$ iff $\vDash_{I,\rho} \alpha$ for all environments ρ ("α is true in the interpretation I").
   Analogously, we say that α is false in the interpretation I iff for no environment ρ it is the case that $\vDash_{I,\rho} \alpha$.

We say that I is a *model* for a set S of formulae iff for all $\alpha \in S$, α is true in I.
The following formulae A-F are tautologies of $\mathcal{L}$ (see Statements for proofs):

Reordering

A. $R{\cdot}P_1{\cdot}P_2(t_1, \dots t_n, t_{n+1}, t_{n+2}) \leftrightarrow R{\cdot}P_2{\cdot}P_1(t_1, \dots t_n, t_{n+2}, t_{n+1})$
B. $R[\frac{\mathfrak{d}_i}{S}][\frac{\mathfrak{d}_j}{Q}](t_1, \dots, t_i, \dots, t_j, \dots, t_n, s_2, \dots, s_m, u_2, \dots, u_l)$
$\leftrightarrow R[\frac{\mathfrak{d}_j}{Q}][\frac{\mathfrak{d}_i}{S}](t_1, \dots, t_j, \dots, t_i, \dots, t_n, u_2, \dots, u_l, s_2, \dots, u_m)$

Drop

III. $R{\cdot}P(t_1, \dots t_n, t_{n+1}) \rightarrow R(t_1, \dots t_n)$
IV. $R\frac{\mathfrak{d}_i}{S}(t_1, \dots, t_i, \dots, t_n, s_2, \dots, s_m) \rightarrow R(t_1, \dots, t_i, \dots, t_n)$
V. $R{\cdot}P_1{\cdot}P_2(t_1, \dots t_n, t_{n+1}, t_{n+2}) \rightarrow (R{\cdot}P_1(t_1, \dots, t_n, t_{n+1}) \wedge R{\cdot}P_2(t_1, \dots, t_n, t_{n+2}))$
VI. $R\frac{\mathfrak{d}}{S}(t_1, \dots, t_n, s_2, \dots, s_m) \rightarrow (R(t_1, \dots, t_n) \wedge S(t_i, s_2, \dots, s_m))$

Non-entailment

G. $[\![R{\cdot}P_1{\cdot}P_2]\!] \subseteq [\![R{\cdot}P_1]\!] \bowtie [\![R{\cdot}P_2]\!]$
H. $[\![R\frac{\mathfrak{d}_i}{S}]\!] \subseteq [\![R]\!] \bowtie \rho_{R.\mathfrak{d}_i/S.\mathfrak{a}}[\![S]\!]$

Statements

Definition 8 (Natural join) Let X, Y, Z be sets of attributes and let $R_1(YX)$ and $R_2(XZ)$ be two relations such that $YX \cap XZ = X$. Natural join is a binary operator on relations R_1, R_2 (written as $R_1 \bowtie R_2$) which produces a relation on XYZ attributes consisting of all the tuples (on XYZ) resulting from concatenation of tuples in R_1 with tuples in R_2 that have identical values for the attributes X:
$R_1 \bowtie R_2 = \{t \textit{ over } XYZ \mid \textit{there exists } t_1 \in R_1, t_2 \in R_2 \textit{ such that } t[XY] = t_1[XY] \textit{and } t[XZ] = t_2[XZ]\}$. (Atzeni and De Antonellis (1993): 15)

Definition 9 (Relation H) Let $G : F \to \mathbb{N}$ be an injective function which prescribes a natural number to each element of F. Let X be a set of all relational attributes of F. Let id be a distinguished attribute. $H(id, X)$ is a set of partial mappings such that $H|_X = F$ and for every $i \in \Pi_{id}(H) : i = G(f_i)$.

Take $H(id, X)$ and $R.\mathfrak{d} \in X$. Let $Y = idX - R.\mathfrak{d}$.
Statement 1 $H|_{id,Y} \bowtie H|_{id,R.\mathfrak{d}} = H(id, X)$ (i.e. $H(id, X)$ has a lossless decomposition with respect to id, Y and id, $R.\mathfrak{d}$).

Proof By definition of H all values in column id of H are unique. Due to that there are the following functional dependencies over attributes idX of H: $id \to id\,R.\mathfrak{d}$ and $id \to id\,Y$. $id\,Y id\,R.\mathfrak{d} = idX$ and $idY \cap id\,R.\mathfrak{d} = id$. By Theorem 4.3 ((Atzeni and De Antonellis 1993): 143) $H(id, X)$ has a lossless decomposition with respect to $id\,Y, id\,R.\mathfrak{d}$, that is, $H(id, X) = H|_{id,Y} \bowtie H|_{id,R.\mathfrak{d}}$. □

Lemma 1.1 follows from Statement 1 by induction:

Lemma 1.1 $H|_{id,R.\mathfrak{d}_1} \bowtie H|_{id,R.\mathfrak{d}_2} \bowtie \cdots \bowtie H|_{id,R.\mathfrak{d}_n}(H) = H(id, X)$.

Let $U = ar^*(ar(R))$ and let $U, R.\mathfrak{c} = ar^*(ar(R{\cdot}P))$. Lemma 1.2 follows from Lemma 1.1:

Lemma 1.2 $[\![R{\cdot}P]\!] = (H|_{id,U} \bowtie H|_{id,R.\mathfrak{c}})|_{U,R.\mathfrak{c}}$.

Let R be any relation and A, B be attributes such that B is an attribute of R.

Definition 10 (Rename) Rename is an unary operator on a relation R (written as $\rho_{A/B}(R)$) which produces from R a new relation R^* identical to R except that B attribute in all tuples $t \in R$ is renamed to A: $\rho_{A/B}(R) = \{t[A/B] : t \in R\}$.

Let $U = ar^*(ar(R)), V = ar^*(ar(S))$. By Definition 4 $[\![R]\!] = H|_U, [\![S]\!] = H|_V$. Let U' be a set identical to U except $(RS.\mathfrak{d})$ belongs to U' instead of $(R, \mathfrak{d})$. Let V' be a set identical to V except $(RS.\mathfrak{d})$ belongs to V' instead of $(S, \mathfrak{a})$. $U' \cup V' = ar^*(ar(R\frac{\mathfrak{d}}{S}))$. Lemma 1.3 follows from Lemma 1.1:

Lemma 1.3 $[\![R\frac{\mathfrak{d}}{S}]\!] = (\rho_{RS.\mathfrak{d}/R.\mathfrak{d}}(H|_{id,U}) \bowtie H|_{id,RS.\mathfrak{d}} \bowtie \rho_{RS.\mathfrak{d}/S.\mathfrak{a}}(H|_{id,V}))|_{U'V'}$.

Observation Let $R.\mathfrak{c}$ stand for the attribute added by a preposition P to a relation scheme of a predicate R. Letting *id* attribute be included to a relation scheme of a predicate, the extension of a predicate symbol Q can be defined recursively as follows:

$$[\![Q]\!] = \begin{cases} H|_{id,Q.\overline{n}}, \text{ where } Q.\overline{n} = ar^*(ar(Q)) & \text{if } Q \text{ is atomic;} \\ [\![R]\!] \bowtie H|_{id,R.\mathfrak{c}}, & \text{if } Q = R{\cdot}P; \\ \rho_{RS.\mathfrak{d}/R.\mathfrak{d}}[\![R]\!] \bowtie H|_{id,RS.\mathfrak{d}} \bowtie \rho_{RS.\mathfrak{d}/S.\mathfrak{a}}[\![S]\!] & \text{if } Q = R\frac{\mathfrak{d}}{S}. \end{cases}$$

Statement [A] $\vDash_{I,\rho} R{\cdot}P_1{\cdot}P_2(t_1, \dots t_n, t_{n+1}, t_{n+2}) \leftrightarrow R{\cdot}P_2{\cdot}P_1(t_1, \dots t_n, t_{n+2}, t_{n+1})$.

Proof $\vDash_{I,\rho} R{\cdot}P_1{\cdot}P_2(t_{\overline{n}}, t_{n+1}, t_{n+2})$ iff (Definition 7) $((R.\overline{n}) : \|t_{\overline{n}}\|_I^\rho, (R.\mathfrak{d}_{n+1}) : \|t_{n+1}\|_I^\rho, (R.\mathfrak{d}_{n+2}) : \|t_{n+2}\|_I^\rho) \in [\![R{\cdot}P_1{\cdot}P_2]\!]$ iff (Definition 4) $((R.\overline{n}) : \|t_{\overline{n}}\|_I^\rho, (R.\mathfrak{d}_{n+1}) : \|t_{n+1}\|_I^\rho, (R.\mathfrak{d}_{n+2}) : \|t_{n+2}\|_I^\rho) \in F|_{R.\overline{n}, R.\mathfrak{d}_{n+1}, R.\mathfrak{d}_{n+2}}$ iff $((R.\overline{n}) : \|t_{\overline{n}}\|_I^\rho, (R.\mathfrak{d}_{n+2}) : \|t_{n+2}\|_I^\rho, (R.\mathfrak{d}_{n+1}) : \|t_{n+1}\|_I^\rho) \in F|_{R.\overline{n}, R.\mathfrak{d}_{n+2}, R.\mathfrak{d}_{n+1}}$ iff (Definition 4) $((R.\overline{n}) : \|t_{\overline{n}}\|_I^\rho, (R.\mathfrak{d}_{n+2}) : \|t_{n+2}\|_I^\rho, (R.\mathfrak{d}_{n+1}) : \|t_{n+1}\|_I^\rho) \in [\![R{\cdot}P_2{\cdot}P_1]\!]$ iff (Definition 7) $\vDash_{I,\rho} R{\cdot}P_2{\cdot}P_1(t_1, \dots t_n, t_{n+2}, t_{n+1})$. □

Statement [B] $\vDash_{I,\rho} R[\frac{\mathfrak{d}_i}{S}][\frac{\mathfrak{d}_j}{Q}](t_1, \dots, t_i, \dots, t_j, \dots, t_n, s_2, \dots, s_m, u_2, \dots, u_l) \leftrightarrow R[\frac{\mathfrak{d}_j}{Q}][\frac{\mathfrak{d}_i}{S}](t_1, \dots, t_j, \dots, t_i, \dots, t_n, u_2, \dots, u_l, s_2, \dots, u_m)$.

Proof $\vDash_{I,\rho} R[\frac{\mathfrak{d}_i}{S}][\frac{\mathfrak{d}_j}{Q}](t_1, \dots, t_i, \dots, t_j, \dots, t_n, s_2, \dots, s_m, u_2, \dots, u_l)$ iff (Definition 7) $((R.\mathfrak{d}_1) : \|t_1\|_I^\rho, \dots, (RS.\mathfrak{d}_i) : \|t_i\|_I^\rho \dots, (RQ.\mathfrak{d}_j) : \|t_j\|_I^\rho \dots, (R.\mathfrak{d}_n) : \|t_n\|_I^\rho, (S.\overline{m}) : \|s_{\overline{m}}\|_I^\rho, (Q.\bar{l}) : \|u_{\bar{l}}\|_I^\rho) \in [\![R[\frac{\mathfrak{d}_i}{S}][\frac{\mathfrak{d}_j}{Q}]]\!]$ iff (Definition 4)$((R.\mathfrak{d}_1) : \|t_1\|_I^\rho, \dots, (RS.\mathfrak{d}_i) : \|t_i\|_I^\rho \dots, (RQ.\mathfrak{d}_j) : \|t_j\|_I^\rho \dots, (R.\mathfrak{d}_n) : \|t_n\|_I^\rho, (S.\overline{m}) : \|s_{\overline{m}}\|_I^\rho, (Q.\bar{l}) : \|u_{\bar{l}}\|_I^\rho) \in F|_{R.\mathfrak{d}_1, \dots, RS.\mathfrak{d}_i, \dots, RQ.\mathfrak{d}_j, \dots, R.\mathfrak{d}_n, S.\overline{m}, Q.\bar{l}}$ iff $((R.\mathfrak{d}_1) : \|t_1\|_I^\rho, \dots, (RS.\mathfrak{d}_j) : \|t_j\|_I^\rho \dots, (RQ.\mathfrak{d}_i) : \|t_i\|_I^\rho \dots, (R.\mathfrak{d}_n) : \|t_n\|_I^\rho, (Q.\bar{l}) : \|u_{\bar{l}}\|_I^\rho, (S.\overline{m}) : \|s_{\overline{m}}\|_I^\rho) \in F|_{R.\mathfrak{d}_1, \dots, RQ.\mathfrak{d}_j, \dots, RS.\mathfrak{d}_i, \dots, R.\mathfrak{d}_n, Q.\bar{l}, S.\overline{m}}$ iff (Definition 4) $((R.\mathfrak{d}_1) : \|t_1\|_I^\rho, \dots, (RS.\mathfrak{d}_j) : \|t_j\|_I^\rho \dots, (RQ.\mathfrak{d}_i) : \|t_i\|_I^\rho \dots, (R.\mathfrak{d}_n) : \|t_n\|_I^\rho, (Q.\bar{l}) : \|u_{\bar{l}}\|_I^\rho, (S.\overline{m}) : \|s_{\overline{m}}\|_I^\rho) \in [\![R[\frac{\mathfrak{d}_j}{Q}][\frac{\mathfrak{d}_i}{S}]]\!]$ iff (Definition 7) $\vDash_{I,\rho} R[\frac{\mathfrak{d}_j}{Q}][\frac{\mathfrak{d}_i}{S}](t_1, \dots, t_j, \dots, t_i, \dots, t_n, u_2, \dots, u_l, s_2, \dots, u_m)$.

□

Statement [C] $\vDash_{I,\rho} R{\cdot}P(t_1, \dots t_n, t_{n+1}) \rightarrow R(t_1, \dots t_n)$.

Proof $\vDash_{I,\rho} R{\cdot}P(t_1,\dots t_n,t_{n+1}) \to R(t_1,\dots t_n)$ iff (Definition 7) $((R.\overline{n}){:}\,\|t_{\overline{n}}\|^{\rho}_{I},(R.\mathfrak{d}_{n+1}){:}\,\|t_{n+1}\|^{\rho}_{I}) \in [\![R{\cdot}P]\!]$ iff (Definition 4) $((R.\overline{n}){:}\,\|t_{\overline{n}}\|^{\rho}_{I},(R.\mathfrak{d}_{n+1}){:}\,\|t_{n+1}\|^{\rho}_{I}) \in F|_{R.\overline{n},R.\mathfrak{d}_{n+1}}$ iff $((R.\overline{n}){:}\,\|t_{\overline{n}}\|^{\rho}_{I}) \in F|_{R.\overline{n}}$ iff (Definition 4) $((R.\overline{n}){:}\,\|t_{\overline{n}}\|^{\rho}_{I}) \in [\![R]\!]$ iff (Definition 7)
$\vDash_{I,\rho} R(t_1,\dots t_n)$. □

Statement [D] $\vDash_{I,\rho} R\frac{\mathfrak{d}_i}{S}(t_1,\dots,t_i,\dots,t_n,s_2,\dots,s_m) \to R(t_1,\dots,t_i,\dots,t_n)$.

Proof $\vDash_{I,\rho} R\frac{\mathfrak{d}_i}{S}(t_1,\dots,t_i,\dots,t_n,s_2,\dots,s_m) \to R(t_1,\dots,t_i,\dots,t_n)$ iff (Definition 7)
$((R.\mathfrak{d}_1){:}\,\|t_1\|^{\rho}_{I},\dots,(RS.\mathfrak{d}_i){:}\,\|t_i\|^{\rho}_{I},\dots,(R.\mathfrak{d}_n){:}\,\|t_n\|^{\rho}_{I},(S.\overline{m}){:}\,\|s_{\overline{m}}\|^{\rho}_{I}) \in [\![R\frac{\mathfrak{d}_i}{S}]\!]$ iff (Definition 4)
$((R.\mathfrak{d}_1){:}\,\|t_1\|^{\rho}_{I},\dots,(RS.\mathfrak{d}_i){:}\,\|t_i\|^{\rho}_{I},\dots,(R.\mathfrak{d}_n){:}\,\|t_n\|^{\rho}_{I},(S.\overline{m}){:}\,\|s_{\overline{m}}\|^{\rho}_{I}) \in F|_{R.\mathfrak{d}_1,\dots,RS.\mathfrak{d}_i,\dots,R.\mathfrak{d}_n,S.\overline{m}}$
iff $((R.\mathfrak{d}_1){:}\,\|t_1\|^{\rho}_{I},\dots,(R.\mathfrak{d}_i){:}\,\|t_i\|^{\rho}_{I},\dots,(R.\mathfrak{d}_n){:}\,\|t_n\|^{\rho}_{I},(S.\overline{m}){:}\,\|s_{\overline{m}}\|^{\rho}_{I}) \in F|_{R.\mathfrak{d}_1,\dots,R.\mathfrak{d}_i,\dots,R.\mathfrak{d}_n,S.\overline{m}}$ iff
$((R.\mathfrak{d}_1){:}\,\|t_1\|^{\rho}_{I},\dots,(R.\mathfrak{d}_i){:}\,\|t_i\|^{\rho}_{I},\dots,(R.\mathfrak{d}_n){:}\,\|t_n\|^{\rho}_{I}) \in F|_{R.\mathfrak{d}_1,\dots,R.\mathfrak{d}_i,\dots,R.\mathfrak{d}_n}$ iff (Definition 4)
$((R.\overline{n}){:}\,\|t_{\overline{n}}\|^{\rho}_{I}) \in [\![R]\!]$ iff (Definition 7) $\vDash_{I,\rho} R(t_1,\dots t_n)$. □

Statement [E] $\vDash_{I,\rho} R{\cdot}P_1{\cdot}P_2(t_1,\dots t_n,t_{n+1},t_{n+2}) \to (R{\cdot}P_1(t_1,\dots,t_n,t_{n+1}) \wedge R{\cdot}P_2(t_1,\dots,t_n,t_{n+2}))$.

Proof $\vDash_{I,\rho} R{\cdot}P_1{\cdot}P_2(t_1,\dots t_n,t_{n+1},t_{n+2})$ iff (Statement [C]) $\vDash_{I,\rho} R{\cdot}P_1(t_1,\dots t_n,t_{n+1})$. $\vDash_{I,\rho} R{\cdot}P_1{\cdot}P_2(t_1,\dots t_n,t_{n+1},t_{n+2})$ iff (Statement [A]) $\vDash_{I,\rho} R{\cdot}P_2{\cdot}P_1(t_1,\dots t_n,t_{n+2},t_{n+1})$ iff (Statement [C]) $\vDash_{I,\rho} R{\cdot}P_2(t_1,\dots t_n,t_{n+2})$. $\vDash_{I,\rho} R{\cdot}P_1(t_1,\dots t_n,t_{n+1})$ and $\vDash_{I,\rho} R{\cdot}P_2(t_1,\dots t_n,t_{n+2})$ iff (Definition 7) $\vDash_{I,\rho} R{\cdot}P_1(t_1,\dots,t_n,t_{n+1}) \wedge R{\cdot}P_2(t_1,\dots,t_n,t_{n+2})$. □

Statement [F] $\vDash_{I,\rho} R\frac{\mathfrak{d}_i}{S}(t_1,\dots,t_i,\dots,t_n,s_2,\dots,s_m) \to (R(t_1,\dots,t_i,\dots,t_n) \wedge S(t_i,s_2,\dots,s_m))$.

Proof $\vDash_{I,\rho} R\frac{\mathfrak{d}_i}{S}(t_1,\dots,t_i,\dots,t_n,s_2,\dots,s_m)$ iff (Statement [D]) $\vDash_{I,\rho} R(t_1,\dots,t_i,\dots,t_n)$. $\vDash_{I,\rho} R\frac{\mathfrak{d}_i}{S}(t_1,\dots,t_i,\dots,t_n,s_2,\dots,s_m)$ iff (Definition 7)
$((R.\mathfrak{d}_1){:}\,\|t_1\|^{\rho}_{I},\dots,(RS.\mathfrak{d}_i){:}\,\|t_i\|^{\rho}_{I},\dots,(R.\mathfrak{d}_n){:}\,\|t_n\|^{\rho}_{I},(S.\mathfrak{c}_2){:}\,\|s_2\|^{\rho}_{I},\dots,(S.\mathfrak{c}_m){:}\,\|s_m\|^{\rho}_{I}) \in [\![R\frac{\mathfrak{d}_i}{S}]\!]$ iff (Definition 4)
$((R.\mathfrak{d}_1){:}\,\|t_1\|^{\rho}_{I},\dots,(RS.\mathfrak{d}_i){:}\,\|t_i\|^{\rho}_{I},\dots,(R.\mathfrak{d}_n){:}\,\|t_n\|^{\rho}_{I},(S.\mathfrak{c}_2){:}\,\|s_2\|^{\rho}_{I},\dots,(S.\mathfrak{c}_m){:}\,\|s_m\|^{\rho}_{I}) \in$
$\in F|_{R.\mathfrak{d}_1,\dots,RS.\mathfrak{d}_i,\dots,R.\mathfrak{d}_n,S.\mathfrak{c}_2,\dots,S.\mathfrak{c}_m}$ iff $((R.\mathfrak{d}_1){:}\,\|t_1\|^{\rho}_{I},\dots,(S.\mathfrak{a}){:}\,\|t_i\|^{\rho}_{I},\dots,(R.\mathfrak{d}_n){:}\,\|t_n\|^{\rho}_{I},$ $(S.\mathfrak{c}_2){:}\,\|s_2\|^{\rho}_{I},\dots,(S.\mathfrak{c}_m){:}\,\|s_m\|^{\rho}_{I}) \in F|_{R.\mathfrak{d}_1,\dots,S.\mathfrak{a},\dots,R.\mathfrak{d}_n,S.\mathfrak{c}_2,\dots,S.\mathfrak{c}_m}$ iff
$((S.\mathfrak{a}){:}\,\|t_i\|^{\rho}_{I},(S.\mathfrak{c}_2){:}\,\|s_2\|^{\rho}_{I},\dots,(S.\mathfrak{c}_m){:}\,\|s_m\|^{\rho}_{I}) \in F|_{S.\mathfrak{a},S.\mathfrak{c}_2,\dots,S.\mathfrak{c}_m}$ iff (Definition 4)
$((S.\mathfrak{a}:\|t_i\|^{\rho}_{I}),(S.\overline{m}){:}\,\|s_{\overline{m}}\|^{\rho}_{I}) \in [\![S]\!]$ iff (Definition 7) $\vDash_{I,\rho} S(t_i,s_2,\dots,s_m)$.
$\vDash_{I,\rho} R(t_1,\dots,t_i,\dots,t_n)$ and $\vDash_{I,\rho} S(t_i,s_2,\dots,s_m)$ iff (Definition 7)
$\vDash_{I,\rho} R(t_1,\dots,t_i,\dots,t_n) \wedge S(t_i,s_2,\dots,s_m))$. □

Statement [G] $[\![R{\cdot}P_1{\cdot}P_2]\!] \subseteq [\![R{\cdot}P_1]\!] \bowtie [\![R{\cdot}P_2]\!]$.

Proof By Definition 4 $[\![R{\cdot}P_1{\cdot}P_2]\!] = F|_{R\overline{n},R.\mathfrak{d}_{n+1},R.\mathfrak{d}_{n+2}}$. $t[R\overline{n}, R.\mathfrak{d}_{n+1}, R.\mathfrak{d}_{n+2}] \in\in F|_{R\overline{n},R.\mathfrak{d}_{n+1},R.\mathfrak{d}_{n+2}}$ iff $t[R\overline{n}, R.\mathfrak{d}_{n+1}] \in F|_{R\overline{n},R.\mathfrak{d}_{n+1}}$ and $t[R\overline{n}, R.\mathfrak{d}_{n+2}] \in F|_{R\overline{n},R.\mathfrak{d}_{n+2}}$ iff (Definition 8) $t[R\overline{n}, R.\mathfrak{d}_{n+1}, R.\mathfrak{d}_{n+2}] \in F|_{R\overline{n},R.\mathfrak{d}_{n+1}} \bowtie F|_{R\overline{n},R.\mathfrak{d}_{n+2}}$ iff (Definition 4)

$t[R\overline{n}, R.\mathfrak{d}_{n+1}, R.\mathfrak{d}_{n+2}] \in [\![R{\cdot}P_1]\!] \bowtie [\![R{\cdot}P_2]\!]$. □

Statement [H] $[\![R\frac{\mathfrak{d}_i}{S}]\!] \subseteq [\![R]\!] \bowtie \rho_{R.\mathfrak{d}_i/S.\mathfrak{a}}[\![S]\!]$.

Proof By Definition 4$[\![R\frac{\mathfrak{d}_i}{S}]\!] = F|_{R.\mathfrak{d}_1,\ldots,RS.\mathfrak{d}_i,\ldots,R.\mathfrak{d}_n,S.\mathfrak{c}_2,\ldots,S.\mathfrak{c}_m}$.

$t[R.\mathfrak{d}_1, \ldots, RS.\mathfrak{d}_i, \ldots, R.\mathfrak{d}_n, S.\mathfrak{c}_2, \ldots, S.\mathfrak{c}_m] \in F|_{R.\mathfrak{d}_1,\ldots,RS.\mathfrak{d}_i,\ldots,R.\mathfrak{d}_n,S.\mathfrak{c}_2,\ldots,S.\mathfrak{c}_m}$ iff

$t[R.\mathfrak{d}_1, \ldots, RS.\mathfrak{d}_i, \ldots, R.\mathfrak{d}_n] \in F|_{R.\mathfrak{d}_1,\ldots,RS.\mathfrak{d}_i,\ldots,R.\mathfrak{d}_n}$ and $t[RS.\mathfrak{d}_i, S.\mathfrak{c}_2, \ldots, S.\mathfrak{c}_m] \in\in F|_{RS.\mathfrak{d}_i,S.\mathfrak{c}_2,\ldots,S.\mathfrak{c}_m}$ iff (Definition 8) $t[R.\mathfrak{d}_1, \ldots, RS.\mathfrak{d}_i, \ldots, R.\mathfrak{d}_n, S.\mathfrak{c}_2, \ldots, S.\mathfrak{c}_m] \in$

$\in F|_{R.\mathfrak{d}_1,\ldots,RS.\mathfrak{d}_i,\ldots,R.\mathfrak{d}_n} \bowtie F|_{RS.\mathfrak{d}_i,S.\mathfrak{c}_2,\ldots,S.\mathfrak{c}_m}$ iff (Definition 4, 10) $t[R.\mathfrak{d}_1, \ldots, R.\mathfrak{d}_i, \ldots, R.\mathfrak{d}_n, S.\mathfrak{c}_2, \ldots, S.\mathfrak{c}_m] \in F|_{R.\mathfrak{d}_1,\ldots,R.\mathfrak{d}_i,\ldots,R.\mathfrak{d}_n} \bowtie \rho_{R.\mathfrak{d}_i/S.\mathfrak{a}}(F|_{S.\mathfrak{a},S.\mathfrak{c}_2,\ldots,S.\mathfrak{c}_m})$ iff

$t[R.\mathfrak{d}_1, \ldots, R.\mathfrak{d}_i, \ldots, R.\mathfrak{d}_n, S.\mathfrak{c}_2, \ldots, S.\mathfrak{c}_m] \in [\![R]\!] \bowtie \rho_{R.\mathfrak{d}_i/S.\mathfrak{a}}[\![S]\!]$. □

Acknowledgements This paper has benefited greatly from the feedback of Katarzyna Kijania-Placek, Zoltán Szabó, Kate Stanton (and fellows from Yale's ELLMM City Workgroup), Abraham Silberschatz, Wojciech Suchoń, Justyna Grudzińska-Zawadowska, Aleksandra Samonek, Marek Jeziorski and my colleagues from the Department of Epistemology, Logic and Cognitive Science of Pedagogical University of Cracow. I dedicate the paper to my beloved teacher Lena, passed away several years ago.

Author Contributions The article was written by Olga Poller in 100%.

Funding This research was supported by the National Science Centre of Poland [grant number 2017/24/C/HS1/00158].

Declarations

Conflict of interest The author has no relevant financial or non-financial interests to disclose. The author has no conflict of interest to declare that are relevant to the content of this article.

References

Abiteboul S, Hull R, Vianu V (1996) Foundations of databases. Addison-Wesley, Reading

Atzeni P, De Antonellis V (1993) Relational database theory. The Benjamin/Cummings Publishing Company, Redwood

Author NN (0002) Suppressed for blind review

Baronian L (2006) Preposition contractions in Quebec French. In: Saint-Dizier P (ed) Syntax and semantics of prepositions. Springer, Dordrecht, pp 27–42
Beaver D, Condoravdi C (2007) On the logic of verbal modification. In: Aloni M, Dekker P, Roelofsen F (eds) Proceedings of the 16th Amsterdam Colloqium, pp 3–9. University of Amsterdam, Amsterdam
Burkhart B, Lichte T, Kallmeyer L (2017) Depictives in English: An LTAG approach. In: Kuhlmann M, Scheffler T editors, Proceedings of the 13th international workshop on tree adjoining grammars and related formalisms (TAG+13), Umeå, Sweden, September, 2017, pp 4–6. Association for Computational Linguistics, 21-30, Stroudsburg, PA
Carlson G (1984) Thematic roles and their role in semantic interpretation. Linguistics 22(3):259–279
Carnie A (2006) Syntax. A generative introduction, 2nd edn. Blackwell, Oxford
Chan E, Atzeni P (1992) Connection-trap-free database schemes. J Comput Syst Sci 44(1):1–22
Chierchia G (1984) Topics in the syntax and semantics of infinitives and gerunds. PhD thesis, ProQuest Dissertations and Theses
Chomsky N (1981) Lectures on government and binding, 5th edn. Foris Publication, Dordrecht
Codd E (1970) A relational model of data for large shared data banks. Commun ACM Assoc Comput Mach 13(6):377–387
Codd E (1972) Relational completeness of data base sublanguages. In: Rustin R (ed) Courant computer science symposium 6: data base systems. Prentice-Hall, Englewood Cliffs, pp 65–98
Davidson D (1967) The logical form of action sentences. In: Davidson D (ed) Essays on actions and events. Claredon Press, Oxford, pp 105–121
Dekker P (2004) Cases, adverbs, situations and events. In: Kamp H, Partee B (eds) Context-dependence in the analysis of linguistic meaning. Elsevier, Amsterdam, pp 383–404
Dowty D (1989) On the semantic content of the notion thematic role. In: Chierchia G, Partee B, Turner R (eds) Studies in linguistics and philosophy, pp 69–129. formerly Synthese Language Library), vol 39. Springer, Dordrecht
Eckardt R (2010) A logic for easy linking semantics. Logic, language and meaning, pp 274–283
Elmasri R, Navathe S (2001) Fundamentals of database systems, 6th edn. Addison-Wesley, Boston
Fillmore C (1968) The case for case. In: Bach E, Harms RT (eds) Universals in Linguistic Theory. Holt Rinehart and Winston, London, pp 1–88
Francez N (2017) A proof-theoretic semantics for adjectival modification. J Logic Lang Inf 26(1):21–43
Fulton JA (1979) An intensional logic of predicates and predicates modifiers without modal operators. Notre Dame J Form Log XX(4):807–834
Garcia-Molina H, Ullman J, Widom J (2013) Database systems: complete book, vol 2. Pearson, Harlow
Garson J (1981) Prepositional logic. Log Anal Louvain 24(93):3–33
Gawron J (1986) Situations and prepositions. Linguist Philos 9(3):327–382
Gruber J (1962) Studies in lexical relations. MIT doctoral dissertation. MIT Press, Massachusetts
Haegeman L (1994) Introduction to government and binding theory, 2nd edn. Blackwell, Oxford
Himmelmann N, Schultze-Berndt C (2005) Issues in the syntax and semantics of participant-oriented adjuncts: an introduction. In: Himmelmann N, Schultze-Berndt E (eds) Secondary predication and adverbial modification. The typology of depictives, pp 1–69. Oxford University Press, New York
Katz G (2008) Manner modification of state verbs. In: McNally L, Kennedy C (eds) Adjectives and adverbs. Oxford University Press, New York, pp 220–248
Kretzmann N (1982) Syncategorema, exponibilia, sophismata. In: Kretzmann N, Kenny A, Pinborg J (eds) The Cambridge history of later medieval philosophy. Cambridge University Press, Cambridge, pp 211–245
Krifka M (1992) Thematic relations as links between nominal reference and temporal constitution. In: Sag I, Szabolsci A (eds) Lexical matters. Center for the Study of Language and Information, Stanford, pp 29–53
Landman F (2000) Events and plurality: the Jerusalem lectures. Kluwer Academic Publishers, Dordrecht
Larson R (1998) Events and modification in nominals. In: Strolovitch D, Lawson A (eds) Proceedings from semantics and linguistic theory (SALT) VIII. Cornell University, Ithaca, pp 1–27
Larson R (2010) On Pylkkänen's semantics for low applicatives. Linguist Inq 41(4):701–704
Link G (1998) Algebraic semantics in language and philosophy. CSLI, Stanford
Maienborn C (2003) Event-internal modifiers: semantic under specification and conceptual interpretation. In: Lang E, Maienborn C, Fabricius-Hansen C (eds) Modifying adjuncts. de Gruyter, Berlin, pp 475–509
Morzycki M (2015) Modification, chapter The lexical semantics of adjectives: more than just scales, pp 13–87. Key topics in semantics and pragmatics. Cambridge University Press, Cambridge
Pagin P, Westerståhl D (2010) Compositionality I: definitions and variants. Philos Compass 3(5):50–264
Parsons T (1970) Some problems concerning the logic of grammatical modifiers. Synthese 21(3):320–334

Parsons T (1990) Events in the semantics of English. A study in subatomic semantics. The MIT Press, Cambridge
Partee B (2000) Some remarks on linguistic uses of the notion of event. In: Tenny C, Pustejovsky J (eds) Events as grammatical objects. CSLI Publications, Stanford, pp 483–495
Pörn N (1982) On the logic of adverbs. Stud Log XLI I(2/3):293–298
Pylkkänen L (2002) Introducing arguments. MIT Press, Massachusetts
Quine WV (1960) Variables explained away. Proc Am Philos Soc 104(3):343–347
Quirk R, Greenbaum S, Leech G, Svartvik J (1985) Comprehensive grammar of the English language. Longman, New York
Reiter R (1989) Towards a logical reconstruction of relational database theory. In: Readings in artificial intelligence and databases, pp 301–327. Elsevier, Amsterdam
Rothstein S (2003) Secondary predication and aspectual structure. In: Lang E, Maienborn C, Fabricius-Hansen C (eds) Modifying adjuncts. de Gruyter, Berlin, pp 553–590
Rothstein S (2004) Predicates and their subjects. Springer, Dordrecht
Rothstein S (2004) Structuring events. A study in the semantics of lexical aspect. Blackwell, Malden
Schäfer M (2008) Resolving scope in manner modification. In: Bonami O and Cabredo Hofherr P (eds) Empirical issues in syntax and semantics 7, pp 351–372. CSSP 7
Song I, Jones T (1995) Ternary relationship decomposition strategies based on binary imposition rules. In: Proceedings of the eleventh international conference on data engineering, pp 6–10, March 1995. Taiwan. IEEE, 485-492
Song I, Jones T, Park E (1993) Binary relationship imposition rules on ternary relationships in er modeling. In: Proceeding CIKM '93 proceedings of the second international conference on information and knowledge management, pp 57–66. ACM, New York
Stowell T (1981) Origins of phrase structure. Doctoral dissertation. MIT, Cambridge
Sumathi S, Esakkirajan S (2007) Fundamentals of relational database management systems. Springer, Berlin
Svenonius P (1994) Dependent nexus: subordinate predication structures in English and the Scandinavian languages. PhD thesis, UCSC
Szabó Z (2003) On qualification. Philos Perspect 17:385–414
van Fraassen B (1973) Extension, intension, and comprehension. In: Munitz M (ed) Logic and ontology. New York University Press, New York, pp 101–131
Williams E (1981) Argument structure and morphology. Linguist Rev 1(1):81–114
Winkler S (1997) Focus and secondary predication. de Gruyter, Berlin