Introduction to Glue Semantics: class 3
Event semantics

Jamie Y. Findlay
University of Oslo
jamie.findlay@iln.uio.no

06/08/25

1 Background: why events?

1.1 Referring to events

> Strange goings on! Jones did it slowly, deliberately, in the bathroom, with a knife, at midnight. What he did was butter a piece of toast. We are too familiar with the language of action to notice at first an anomaly: the 'it' of 'Jones did it slowly, deliberately, . . . ' seems to refer to some entity, presumably an action, that is then characterized in a number of ways.
> (Davidson 1967: 81)

- So begins Davidson's influential 1967 paper, which launched the approach to meaning known as EVENT SEMANTICS.

- It does seem that we need to be able to refer to 'happenings', or EVENTS,[1] directly, and even quantify over them:

(1) a. *Willow did it carefully.*
b. *Something scary happened.*

- Based on this fact, and for other reasons we will see below, we introduce a new type of event individual to our semantic ontology, and give it the type v.

- We assume that verbs now take an additional event argument, so that the meaning constructor for e.g. *cast* (*a spell*) is (2):

(2) $\lambda y.\lambda x.\lambda e.\mathbf{cast}(e,x,y) : E(\bullet\ \texttt{obj}) \multimap [E(\bullet\ \texttt{subj}) \multimap [V(\bullet) \multimap T(\bullet)]]$

- We then introduce a special meaning constructor that existentially binds the event variable at the root, once the verb has combined with all its arguments:[2]

[1] Sometimes a split is made between *events*, which are dynamic, and *states*, which are not, with the cover term *eventualities* used to refer to both. Other divisions are also possible.

[2] We will use f and g for variables of type $\langle v, t\rangle$.

(3) $\texttt{root} \rightsquigarrow \lambda f.\exists e.f(e) : [V(\bullet) \multimap T(\bullet)] \multimap T(\bullet)$

- This gives the final meaning in (5) for the sentence in (4); the proof is shown in Figure 1.

(4) *Willow cast a spell.*

(5) $\exists e.\exists x.\mathbf{spell}(x) \wedge \mathbf{cast}(e, \mathbf{willow}, x)$

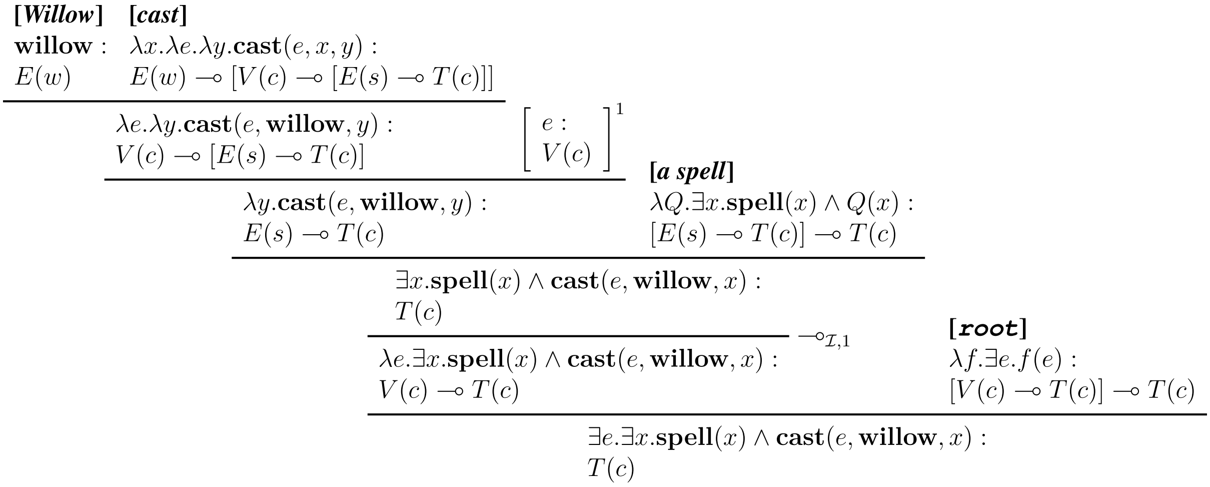

[*Willow*]
$\mathbf{willow} : E(w)$
[*cast*]
$\lambda x.\lambda e.\lambda y.\mathbf{cast}(e,x,y) : E(w) \multimap [V(c) \multimap [E(s) \multimap T(c)]]$
$\lambda e.\lambda y.\mathbf{cast}(e,\mathbf{willow},y) : V(c) \multimap [E(s) \multimap T(c)]$
$\left[e : V(c) \right]^1$
$\lambda y.\mathbf{cast}(e,\mathbf{willow},y) : E(s) \multimap T(c)$
[*a spell*]
$\lambda Q.\exists x.\mathbf{spell}(x) \wedge Q(x) : [E(s) \multimap T(c)] \multimap T(c)$
$\exists x.\mathbf{spell}(x) \wedge \mathbf{cast}(e,\mathbf{willow},x) : T(c)$
$\multimap_{\mathcal{I},1}$
$\lambda e.\exists x.\mathbf{spell}(x) \wedge \mathbf{cast}(e,\mathbf{willow},x) : V(c) \multimap T(c)$
[***root***]
$\lambda f.\exists e.f(e) : [V(c) \multimap T(c)] \multimap T(c)$
$\exists e.\exists x.\mathbf{spell}(x) \wedge \mathbf{cast}(e,\mathbf{willow},x) : T(c)$

Figure 1: Glue proof for an event semantics interpretation of *Willow cast a spell*

1.2 Inference patterns: modifiers

- Aside from the necessity of referring to or quantifying over events directly, another reason to add events to our semantics is that it gives us an immediate explanation for the inference pattern shown in (7):

(6) *Willow cast the spell slowly in the attic at midnight.*

a. $\Rightarrow$ *Willow cast the spell slowly in the attic.*
b. $\Rightarrow$ *Willow cast the spell slowly.*
c. $\Rightarrow$ *Willow cast the spell.*

- That is, modifiers like *slowly*, *in the attic*, and *at midnight* can be dropped *salva veritate*, i.e. without changing the truth value of the sentence.

- If we treat modifiers as introducing additional conditions which hold of the event in question, the inferences go through straightforwardly in our predicate logic:[3]

(7) $\exists e.\mathbf{cast}(e, \mathbf{willow}, \mathbf{the\text{-}spell}) \wedge \mathbf{slow}(e) \wedge \mathbf{location}(e, \mathbf{the\text{-}attic}) \wedge \mathbf{time}(e, \mathbf{midnight})$

a. $\vdash_{\wedge_\mathcal{E}} \exists e.\mathbf{cast}(e, \mathbf{willow}, \mathbf{the\text{-}spell}) \wedge \mathbf{slow}(e) \wedge \mathbf{location}(e, \mathbf{the\text{-}attic})$
b. $\vdash_{\wedge_\mathcal{E}} \exists e.\mathbf{cast}(e, \mathbf{willow}, \mathbf{the\text{-}spell}) \wedge \mathbf{slow}(e)$
c. $\vdash_{\wedge_\mathcal{E}} \exists e.\mathbf{cast}(e, \mathbf{willow}, \mathbf{the\text{-}spell})$

[3] To make things simpler, I translate *the spell* and *the attic* as constants; this doesn't affect the point at hand, it just makes the representations less complex.

- Other approaches (e.g. treating *slowly* as an $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$ modifier of predicates) do not achieve these results so elegantly, if at all.

- How does this work compositionally? The modifiers also operate on the type $\langle v,t\rangle$ meanings that our event closure meaning applies to:

(8)

$$\begin{array}{lll} \textit{slowly} & \rightsquigarrow & \lambda f.\lambda e.f(e) \wedge \mathbf{slow}(e) : \\ & & [V(\hat{\bullet}) \multimap T(\hat{\bullet})] \multimap V(\hat{\bullet}) \multimap T(\hat{\bullet}) \\ \textit{in} & \rightsquigarrow & \lambda x.\lambda f.\lambda e.f(e) \wedge \mathbf{location}(e,x) : \\ & & E(\bullet\ \texttt{obj}) \multimap [V(\hat{\bullet}) \multimap T(\hat{\bullet})] \multimap V(\hat{\bullet}) \multimap T(\hat{\bullet}) \\ \textit{at} & \rightsquigarrow & \lambda x.\lambda f.\lambda e.f(e) \wedge \mathbf{time}(e,x) : \\ & & E(\bullet\ \texttt{obj}) \multimap [V(\hat{\bullet}) \multimap T(\hat{\bullet})] \multimap V(\hat{\bullet}) \multimap T(\hat{\bullet}) \end{array}$$

- Figure 2 shows the relevant final steps of the Glue proof for *Willow cast the spell slowly*.

$$\dfrac{\dfrac{\begin{array}{l}[\textit{Willow cast the spell}]\\ \lambda e.\mathbf{cast}(e,\mathbf{willow},\mathbf{the\text{-}spell}) : \\ V(c) \multimap T(c)\end{array} \quad \begin{array}{l}[\textit{slowly}]\\ \lambda f.\lambda e.f(e) \wedge \mathbf{slow}(e) : \\ [V(c) \multimap T(c)] \multimap V(c) \multimap T(c)\end{array}}{\begin{array}{l}\lambda e.\mathbf{cast}(e,\mathbf{willow},\mathbf{the\text{-}spell}) \wedge \mathbf{slow}(e) : \\ V(c) \multimap T(c)\end{array}} \quad \begin{array}{l}[\texttt{root}]\\ \lambda f.\exists e.f(e) : \\ [V(c) \multimap T(c)] \multimap T(c)\end{array}}{\begin{array}{l}\exists e.\mathbf{cast}(e,\mathbf{willow},\mathbf{the\text{-}spell}) \wedge \mathbf{slow}(e) : \\ T(c)\end{array}}$$

Figure 2: Partial Glue proof for *Willow cast the spell slowly*

2 The neo-Davidsonian turn

- In the above examples, we augmented our existing predicates with a new event argument, so that e.g. a two-place (transitive) predicate becomes a three-place predicate.

- Certain modifiers can then add extra conditions on the event, as we saw with adverbs and locative/temporal PPs.

- An alternative approach is to treat *all* dependents of the verb as denoting separate predicates in the semantics, so that (4) gets the following translation instead of that in (5) (see e.g. Parsons 1990, 1995; Pietroski 2002):

(9) $\exists e.\exists x.\mathbf{cast}(e) \wedge \mathbf{agent}(e,\mathbf{willow}) \wedge \mathbf{theme}(e,x)$

- This approach is known as NEO-DAVIDSONIAN, since it follows the spirit of Davidson's analysis, but diverges from it in on this point.

- Some of the motivation for the neo-Davidsonian approach is more conceptual than empirical, but it does allow us to capture certain inference patterns such as in the transitive–inchoative alternation:

(10) a. *Tara melted the chocolate.*
b. ⇒ *The chocolate melted.*

(11) a. $\exists e.\mathbf{melt}(e) \wedge \mathbf{agent}(e, \mathbf{tara}) \wedge \mathbf{theme}(e, \mathbf{the\text{-}chocolate})$
b. $\vdash_{\wedge_\mathcal{E}} \exists e.\mathbf{melt}(e) \wedge \mathbf{theme}(e, \mathbf{the\text{-}chocolate})$

- The neo-Davidsonian approach also allow us to capture certain generalisations about classes of verbal arguments.

- For instance, if we assume that agents are volitional instigators of events, then we can include a meaning postulate like the below in our semantics to capture this:[4]

(12) $\forall e.\forall x.\mathbf{agent}(e, x) \rightarrow \mathbf{cause}(x, e) \wedge \mathbf{intend}(x, e)$

- But if the agent is just encoded as, say, the first argument of various predicates, we would need as many meaning postulates as there are agentive predicates to achieve the same result:

(13) a. $\forall e.\forall x.\forall y.\mathbf{hit}(e, x, y) \rightarrow \mathbf{cause}(x, e) \wedge \mathbf{intend}(x, e)$
b. $\forall e.\forall x.\forall y.\mathbf{kick}(e, x, y) \rightarrow \mathbf{cause}(x, e) \wedge \mathbf{intend}(x, e)$
c. $\forall e.\forall x.\forall y.\mathbf{choose}(e, x, y) \rightarrow \mathbf{cause}(x, e) \wedge \mathbf{intend}(x, e)$
...

- Clearly this is missing a generalisation!

- The neo-Davidsonian approach also offers the possibility of treating thematic roles as being contributed independently of the verb's main event-describing predicate – for example, different ARGUMENT STRUCTURE frames can come along with whatever particular relations they express.

- This can be useful in capturing what remains constant across different syntactic uses of the same verb. Consider (14), for example (Pietroski 2002):

(14) a. *Nora explained the fact that Fido barked.*
b. *Nora explained that Fido barked.*

- As Pietroski (2002: 98) explains "[(14a)] is roughly synonymous with 'Nora explained why Fido barked"', while "[(14b)] is true (roughly) iff Nora said *that* Fido barked and thereby explained something else".

- Call the complement of *explained* in (14a) a theme (in this case the explanandum, the thing explained), and the complement of *explained* in (14b) the content (the thing said in giving an explanation).

- In a classical Davidsonian account, we could give a single meaning for *explain* only if there was *always* an argument slot for each of the theme and the content. We would then have to say it is existentially closed or otherwise rendered inert when it is not expressed.

[4] I'm not really making a strong claim about whether this particular inference actually goes through, merely that *some* such inferences must do, if thematic roles are to have any bite.

- In the neo-Davidsonian approach, by contrast, we can retain the same predicate of events $\lambda e.\mathbf{explain}(e)$ for both uses, and simply include whichever thematic roles are relevant in a given situation:

(15) a. $\exists e.\mathbf{explain}(e) \wedge \mathbf{agent}(e, \mathbf{nora}) \wedge \mathbf{theme}(e, [\![\textit{the fact that Fido barked}]\!])$
b. $\exists e.\mathbf{explain}(e) \wedge \mathbf{agent}(e, \mathbf{nora}) \wedge \mathbf{content}(e, [\![\textit{that Fido barked}]\!])$

- The question then arises: how do we know which thematic roles to introduce into the semantic composition?

- There is more than one way to approach this, depending on one's prior assumptions and theoretical desiderata.

- For example, we can still include the information in the lexicon, but make the core meaning of the verb and its valency frame into two different meaning constructors:

(16) *cast* $\rightsquigarrow$ $\lambda e.\mathbf{cast}(e)$:
$V(\bullet) \multimap T(\bullet)$

$\lambda f.\lambda y.\lambda x.\lambda e.f(e) \wedge \mathbf{agent}(e, x) \wedge \mathbf{theme}(e, y)$:
$[V(\bullet) \multimap T(\bullet)] \multimap [E(\bullet\ \texttt{obj}) \multimap [E(\bullet\ \texttt{subj}) \multimap [V(\bullet) \multimap T(\bullet)]]]$

- These two meaning constructors compose to give a more standard meaning constructor for the verb as a whole, where it expects a subject and object argument corresponding to its agent and theme arguments respectively:

(17) $\lambda y.\lambda x.\lambda e.\mathbf{cast}(e) \wedge \mathbf{agent}(e, x) \wedge \mathbf{theme}(e, y)$:
$E(\bullet\ \texttt{obj}) \multimap [E(\bullet\ \texttt{subj}) \multimap [V(\bullet) \multimap T(\bullet)]]$

- This might make grammar description easier/more compact, since the valency frame meaning constructor will be shared across many different verbs, while the only idiosyncratic thing about *cast* is kept separate.[5]

- A different solution is to have the thematic role predicates introduced constructionally by certain syntactic relations.

- This offers the possibility of a universal semantic interpretation system that postpones the use of language-specific valency lexica as long as possible in the compositional process – potentially very useful when dealing with low-resource languages for which such information is not readily available.[6]

- But since there is no one-to-one mapping between grammatical relations and thematic roles, this cannot be the whole story; we have to use underspecified relation names, or use syntactic labels in the semantics, and consult a valency lexicon at some later stage.

[5] On this decompositional/modular approach to the lexicon, see e.g. Asudeh et al. (2014), Przepiórkowski (2017), and Findlay (2020).

[6] See e.g. Reddy et al. (2017) and Findlay et al. (2023) for examples of recent work in this vein.

(18) $\texttt{subj} \rightsquigarrow \lambda f.\lambda x.\lambda e.f(e) \wedge \mathbf{arg_1}(e,x)$:
$[V(\hat{\bullet}) \multimap T(\hat{\bullet})] \multimap [E(\bullet) \multimap [V(\hat{\bullet}) \multimap T(\hat{\bullet})]]$

$\texttt{obj} \rightsquigarrow \lambda f.\lambda x.\lambda e.f(e) \wedge \mathbf{arg_2}(e,x)$:
$[V(\hat{\bullet}) \multimap T(\hat{\bullet})] \multimap [E(\bullet) \multimap [V(\hat{\bullet}) \multimap T(\hat{\bullet})]]$

etc.

3 Problems with quantifiers

- Delaying the existential closure of the event variable until we reach the root node causes problems when it comes to quantifiers and other scope-taking elements.

- The current approach predicts the event quantification will interact scopally with other scope-taking elements, but this is not the case – instead, the event quantification always scopes as low as possible:[7]

(19) *Giles read every book.*

a. ✓ $\forall x.\exists e.\mathbf{read}(e) \wedge \mathbf{agent}(e, \mathbf{giles}) \wedge \mathbf{theme}(e, x) \wedge \mathbf{book}(x)$
(For every book, there is a separate reading event.)

b. ✗ $\exists e.\forall x.\mathbf{read}(e) \wedge \mathbf{agent}(e, \mathbf{giles}) \wedge \mathbf{theme}(e, x) \wedge \mathbf{book}(x)$
(There is one single reading event where all books are read.)

(20) *Angel didn't die.*

a. ✓ $\neg[\exists e.\mathbf{die}(e) \wedge \mathbf{theme}(e, \mathbf{angel})]$
(There is no event of Angel dying.)

b. ✗ $\exists e.\neg[\mathbf{die}(e) \wedge \mathbf{theme}(e, \mathbf{angel})]$
(There is an event in which Angel doesn't die.)

- (19b) does not seem to be an available reading of (19), although intuitions get a little murky in these cases. However, it is certainly true that the translation in (19b) is problematic from a technical point of view if thematic roles are understood as being *functions*, since then there should only be one theme per event.

- (20b), though, is clearly not an available interpretation of (20). This reading is *far* too weak: it is true if there is at least one non-Angel-dying event – but almost all events are of this kind!

- Champollion (2015) proposes a solution to this problem: instead of introducing just a predicate on events, the verb introduces the existential quantification as well. But, in order to allow further constraints on the event variable, it also exposes a variable over event predicates.

- The meaning constructor for *cast* is therefore as follows:

(21) $\lambda y.\lambda x.\lambda f.\exists e.\mathbf{cast}(e) \wedge \mathbf{agent}(e, x) \wedge \mathbf{theme}(e, y) \wedge f(e)$:
$E(\bullet\ \texttt{obj}) \multimap [E(\bullet\ \texttt{subj}) \multimap [[V(\bullet) \multimap T(\bullet)] \multimap T(\bullet)]]$

[7] In (5), the translation of (4), the 'incorrect' scope is innocuous, since the relative scope of two of the *same* quantifier does not have any interpretive effect.

- This now has the fairly monstrous type $\langle e, \langle e, \langle\langle v, t\rangle, t\rangle\rangle\rangle$. Once it combines with its nominal arguments, this denotes a set of sets of events.

- Modifiers like *slowly* can plug in for f so that they still apply to the same event variable as the verb:[8]

(22) $\lambda\mathcal{V}.\lambda f.\mathcal{V}(\lambda e.\mathbf{slow}(e) \wedge f(e))$:
$[[V(\hat{\bullet}) \multimap T(\hat{\bullet})] \multimap T(\hat{\bullet})] \multimap [[V(\hat{\bullet}) \multimap T(\hat{\bullet})] \multimap T(\hat{\bullet})]$

- This can be a little counterintuitive, since the adverb consumes the meaning of the predicate only to apply it to its own meaning – that is, the argument in the compositional syntax is the functor in the semantics. Figure 3 shows how this works:

$$
\begin{array}{l}
\begin{array}{ll}
\textbf{[\textit{Willow cast the spell}]} & \\
\lambda f.\exists e.\mathbf{cast}(e) \wedge \mathbf{agent}(e,\mathbf{willow}) \wedge & \textbf{[\textit{slowly}]} \\
\quad \mathbf{theme}(e,\mathbf{the\text{-}spell}) \wedge f(e) : & \lambda\mathcal{V}.\lambda g.\mathcal{V}(\lambda e'.\mathbf{slow}(e') \wedge g(e')) : \\
{[V(c) \multimap T(c)] \multimap T(c)} & [[V(c) \multimap T(c)] \multimap T(c)] \multimap [[V(c) \multimap T(c)] \multimap T(c)]
\end{array} \\
\hline
\lambda g.\Big[\lambda f.\exists e.\mathbf{cast}(e) \wedge \mathbf{agent}(e,\mathbf{willow}) \wedge \mathbf{theme}(e,\mathbf{the\text{-}spell}) \wedge f(e)\Big]\Big(\lambda e'.\mathbf{slow}(e') \wedge g(e')\Big) : \\
{[V(c) \multimap T(c)] \multimap T(c)} \\
\hline
\quad \lambda g.\exists e.\mathbf{cast}(e) \wedge \mathbf{agent}(e,\mathbf{willow}) \wedge \mathbf{theme}(e,\mathbf{the\text{-}spell}) \wedge \Big[\lambda e'.\mathbf{slow}(e') \wedge g(e')\Big](e) : \quad \Rightarrow_\beta \\
\quad [V(c) \multimap T(c)] \multimap T(c) \\
\hline
\qquad \lambda g.\exists e.\mathbf{cast}(e) \wedge \mathbf{agent}(e,\mathbf{willow}) \wedge \mathbf{theme}(e,\mathbf{the\text{-}spell}) \wedge \mathbf{slow}(e) \wedge g(e) : \quad \Rightarrow_\beta \\
\qquad [V(c) \multimap T(c)] \multimap T(c)
\end{array}
$$

Figure 3: Application of the adverb *slowly*

- Since *slowly* is a modifier, we end up with an expression of the same type as the one we started with, so there is still an exposed variable over predicates on events, leaving the expression open to further modification.

- This open variable ultimately needs to be closed off, just as we had to close off the event variable in the earlier theory.

- To see how this works, recall that the meaning of a verb which has composed with its arguments is a set of sets of events (a type $\langle\langle v, t\rangle, t\rangle$ expression):

(23) $\lambda f.\exists e.\mathbf{die}(e) \wedge \mathbf{theme}(e,\mathbf{buffy}) \wedge f(e)$

- (23) is true of any such set that contains an event of Buffy dying.

- What it means for the sentence to be true is for there to be an event of Buffy dying contained in the set of *all* events.

- The set of all events can be expressed by the following meaning, which simply returns True no matter what event it takes as input (Champollion 2015: 40); we once again associate it with the `root` node:

[8] We use $\mathcal{V}$ as a variable over type $\langle\langle v, t\rangle, t\rangle$ meanings, i.e. the type of verbs once they have combined with their arguments.

(24) $\texttt{root} \rightsquigarrow \lambda e.\mathsf{T} : V(\bullet) \multimap T(\bullet)$

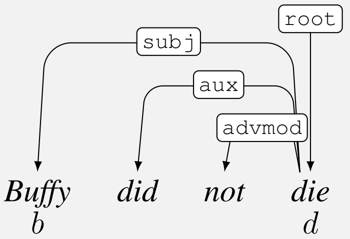
- Figure 4 shows what happens when we pass this as the argument to a fully saturated verbal meaning:

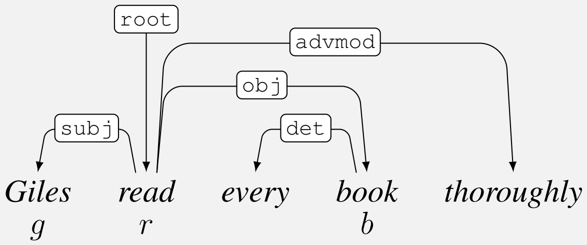
$$\cfrac{\cfrac{\cfrac{\begin{array}{l}[\textbf{\textit{Buffy died}}]\\ \lambda f.\exists e.\mathbf{die}(e) \wedge \mathbf{theme}(e, \mathbf{buffy}) \wedge f(e) :\\ [V(d) \multimap T(d)] \multimap T(d)\end{array} \quad \begin{array}{l}[\texttt{root}]\\ \lambda e.\mathsf{T} :\\ V(d) \multimap T(d)\end{array}}{\exists e.\mathbf{die}(e) \wedge \mathbf{theme}(e, \mathbf{buffy}) \wedge \big[\lambda e.\mathsf{T}\big](e) : T(d)}}{\exists e.\mathbf{die}(e) \wedge \mathbf{theme}(e, \mathbf{buffy}) \wedge \mathsf{T} : T(d)} \Rightarrow_{\beta}}{\exists e.\mathbf{die}(e) \wedge \mathbf{theme}(e, \mathbf{buffy}) : T(d)} \wedge_{\mathsf{Ident}}$$

Figure 4: Closing off a verbal meaning in a Champollion-style event semantics

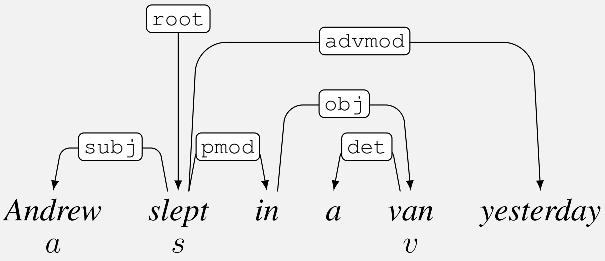
Exercise 1: Event semantics

Give Glue proofs for the following sentences, using a Champollion-style event semantics. Assume the verb's valency frame is part of its meaning constructor, and that *did* makes no semantic contribution.

1.


2.


3. THIS ONE IS TRICKY!



References

Asudeh, Ash, Gianluca Giorgolo & Ida Toivonen. 2014. Meaning and valency. In Miriam Butt & Tracy Holloway King (eds.), *Proceedings of the LFG14 Conference*, 68–88. CSLI Publications. http://web.stanford.edu/group/cslipublications/cslipublications/LFG/19/papers/lfg14asudehetal.pdf.

Champollion, Lucas. 2015. The interaction of compositional semantics and event semantics. *Linguistics and Philosophy* 38(1). 31–66. https://doi.org/10.1007/s10988-014-9162-8.

Davidson, Donald. 1967. The logical form of action sentences. In Nicholas Rescher (ed.), *The logic of decision and action*, 81–120. Pittsburgh, PA: University of Pittsburgh Press.

Findlay, Jamie Y. 2020. Mapping Theory and the anatomy of a lexical entry. In Miriam Butt & Ida Toivonen (eds.), *Proceedings of the LFG20 Conference*, 127–147. Stanford, CA: CSLI Publications. http://web.stanford.edu/group/cslipublications/cslipublications/LFG/LFG-2020/lfg2020-findlay.pdf.

Findlay, Jamie Y., Ahmet Yıldırım, Saeedeh Salimifar & Dag T. T. Haug. 2023. Rule-based semantic interpretation for Universal Dependencies. In *Proceedings of the Sixth Workshop on Universal Dependencies (UDW, GURT/SyntaxFest 2023)*, 47–57. Association for Computational Linguistics. https://aclanthology.org/2023.udw-1.6/.

Parsons, Terence. 1990. *Events in the semantics of English: a study in subatomic semantics*. Cambridge, MA: MIT Press.

Parsons, Terence. 1995. Thematic relations and arguments. *Linguistic Inquiry* 26(4). 635–662. https://www.jstor.org/stable/4178917.

Pietroski, Paul M. 2002. Function and concatenation. In Gerhard Preyer & Georg Peter (eds.), *Logical form and language*, 91–117. Oxford: Clarendon Press.

Przepiórkowski, Adam. 2017. A full-fledged hierarchical lexicon in LFG: the FrameNet approach. In Victoria Rosén & Koenraad De Smedt (eds.), *The very model of a modern linguist: in honor of Helge Dyvik* (Bergen Language and Linguistics Studies 8), 202–219. Bergen: University of Bergen. https://doi.org/10.15845/bells.v8i1.1336.

Reddy, Siva, Oscar Täckström, Slav Petrov, Mark Steedman & Mirella Lapata. 2017. Universal semantic parsing. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, 89–101. Copenhagen, Denmark: Association for Computational Linguistics. https://doi.org/10.18653/v1/D17-1009.