Frame Semantics and Phrasal Semantics

Reinhard Muskens

Tilburg Center for Logic and Philosophy of Science (TiLPS)

Düsseldorf, 6-6-2014

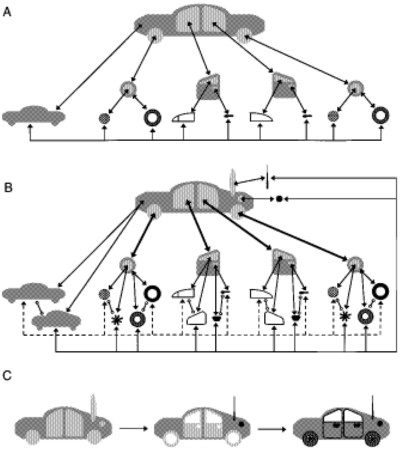
How to Combine Frame Semantics and Phrasal Semantics?

- In this talk I would like to share some ~~unripe~~ somewhat preliminary ideas about how lexical semantics (in the frame tradition of Barsalou, Löbner (year?), and others; the Fillmore tradition seems somewhat different) can be combined with phrasal semantics (in the type logical tradition initiated by Richard Montague).
- The hope is to develop a form of natural language semantics that is 1) cognitively plausible and 2) offers more structure for semantic operations to act upon than is available in standard Montagovian accounts.
- The basic idea is to formalise frames and matching between frames in (partial) type logic.
- There will be similarities with the 'record types' approach of Cooper (2005), Ginsberg, and others, but the frames reside in semantic domains and are denoted by terms, not by types.

Frames and Frame Semantics

- Barsalou (e.g. Barsalou, 1999) proposes a perceptual theory of knowledge based on frames, which, he claims, "can implement a fully functional conceptual system".
- Frames in Barsalou's theory are symbols stored in long-term memory that play a pivotal role in language and thought. They are extracted from perceptual states in sensory-motor systems and subsequently can act as simulators that produce "limitless simulations of the component".
- Barsalou's frames have been taken up in lexical semantics by Löbner, Osswald, Petersen, and others and lead to a version of frame semantics that is somewhat different from Fillmore's.
- [Barsalou's is very much an empiricist approach to cognition. See also Jesse Prinz's *Furnishing the Mind*, MIT Press, 2002. (Prinz, 2002)]

Car (Barsalou 1999)

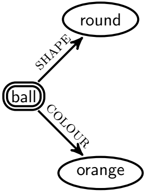


- A Establishing an initial frame for car after processing a first instance.
- B The frame's evolution after processing a second instance.
- C Constructing a simulation of the second instance from the frame in B.

- In A and B arrows represent excitatory connections, lines with circles inhibitatory connections. (Barsalou 1999, p. 590)

Frames as Feature Structures

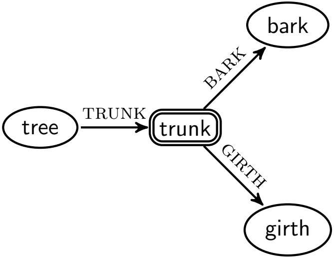
- Barsalou (1992) and Petersen and Osswald (2013) represent frames essentially as feature structures.
- A minor difference being that in P&O (2013) the center of the frame need not be its root.
- There obviously is loss w.r.t. Barsalou's (1999) richer conception. But for the moment, let's accept this loss..
- Below left is a frame for basketball given in Petersen and Osswald (2013). On the right a corresponding AVM.



$$\begin{bmatrix} \text{ball} & \\ \textsc{shape} & \text{round} \\ \textsc{colour} & \text{orange} \end{bmatrix}$$

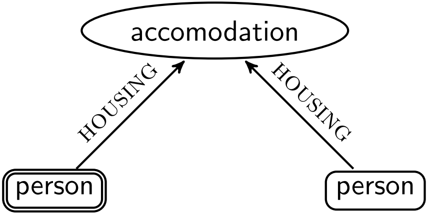
Trunk, Daughter

The following are also from Petersen & Osswald (2013). Note that the center of the trunk frame (marked with two edges) is not the root.



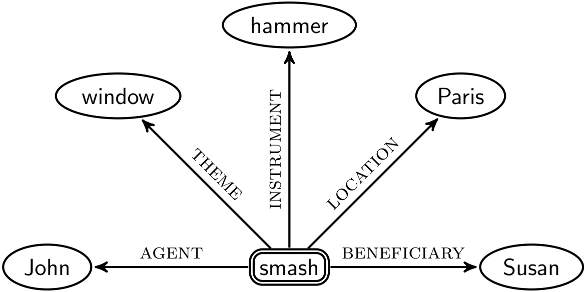
Flatmate

Here is an example from Petersen & Osswald (for flatmate) in which there is no unique root.



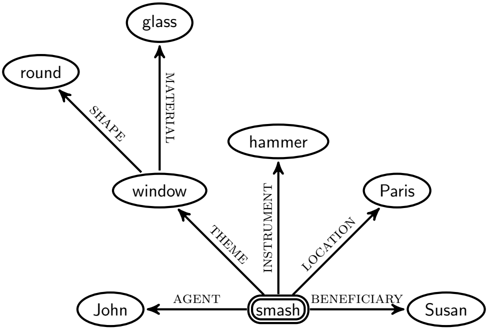
John smashed the window with a hammer in Paris for Susan

Verbal frames seem relevant too. (This one not from P&O.)



'Recursion'

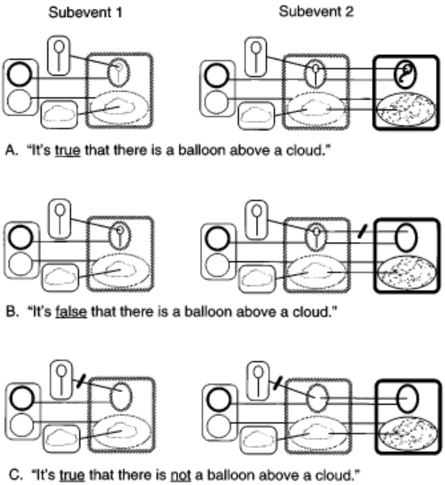
Barsalou (1992) particularly emphasizes the importance of the possibility that values for certain attributes come with further attributes. This distinguishes frames from mere 'feature lists'.



Abstract Concepts

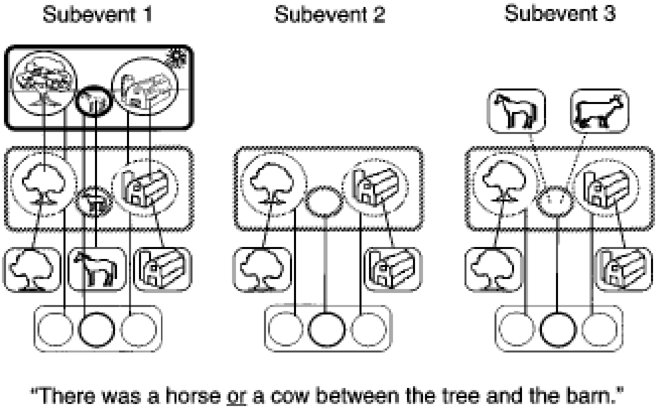
- The main difficulty for any empiricist theory of concept formation is the existence of abstract concepts.
- Barsalou (1999) comes with a nice set of comments (and responses to these) and many of these ask questions about concepts like entropy, democracy, evolution, electricity. What is the perceptual basis of such concepts?
- And what is the perceptual basis of logical concepts, including modal ones?
- There is some discussion of truth, negation, disjunction, implication, and syllogism in Barsalou (1999), but there is no treatment of reasoning that would convince a logician.

Truth and Negation in Barsalou (1999)



- Truth as correspondence between simulation and perceived situation.
- Falsity as lack of such correspondence.
- Negation as mismatch between simulator and simulation.

Disjunction in Barsalou (1999)



Algorithmic Aspects of Matching

- Barsalou's accounts of logical operations seem to be based on certain matching procedures.
- But no formally precise account of these procedures is given.
- Let us try to formulate precise specifications of such procedures.
- And don't let's worry too much about the question to what extent these procedures or specifications are grounded in perception. (To me it seems that Barsalou's data structures are perceptual, but that his algorithms are not.)
- In any case, our move will be one from (plain) empiricism to logical empiricism.

Frames Formalised

- How can frames be represented logically?
- Answer: there are various ways. Hayes (1979) already used predicate logic for this (see also, e.g., Johnson, 1991; Smolka, 1988). Various feature logics are also relevant here.
- My point of departure will be the Data Logic of Veltman (1985). This logic can be provided with a semantics based on certain possible facts, which I will reinterpret as mental representations of possible facts or frames.

Facts

Veltman (1985) considers certain semi-lattices of facts called data lattices. The following axioms are taken from him.

- $\exists f\; f \neq 0$
- $f \circ f = f$
- $f \circ g = g \circ f$
- $(f \circ g) \circ h = f \circ (g \circ h)$
- $0 \circ f = 0$

- Here we have used f, g and h as variables of a type $\mathfrak{f}$ and 0 is a constants of this type. $\circ$ is of type $\mathfrak{f}(\mathfrak{f}\mathfrak{f})$ (infix notation).
- $f \circ g$ is called the combination of f and g. The 0 is used for convenience; $f \circ g = 0$ says that f and g are incompatible (i.e. their combination is undefined).

Abbreviations and an Extra Axiom

Let's use the following abbreviations.

- $f \leq g$ abbreviates $f \circ g = f$ (f incorporates g, a partial ordering).
- $\mathrm{A}f$ abbreviates $f \neq 0 \wedge \forall g((g \leq f \wedge g \neq 0) \rightarrow g = f)$
  (f is an atom or world).

And let's turn our semi-lattices into certain atomic semi-lattices by imposing the following axiom.

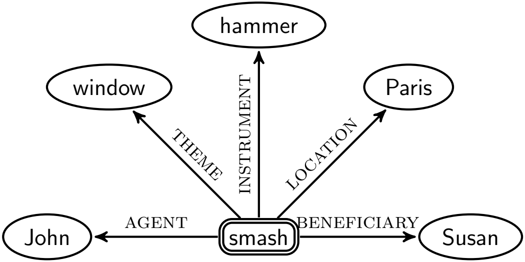
$$f \not\leq g \rightarrow \exists h(\mathrm{A}h \wedge h \leq f \wedge h \not\leq g)$$

The following extensionality principle will hold.

$$f = g \leftrightarrow \forall h(\mathrm{A}h \rightarrow (h \leq f \leftrightarrow h \leq g))$$

(See also Varzi's 2007 paper on mereology.)

Facts and Frames



smash $e \circ \textsc{ag}\, ex \circ$ John $x \circ \textsc{th}\, ey \circ$ window $y \circ \ldots$

Here AG is a constant of type $ee\mathfrak{f}$, for example, so that $\textsc{ag}\, ex$ is of type $\mathfrak{f}$ (a fact). The whole term also denotes a (complex) fact.

Incompatibility and Unifiability

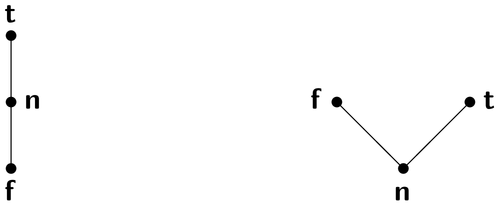
- In order for this to work we need meaning postulates such as
  $\forall exy(x \neq y \rightarrow \textsc{ag}\, ex \circ \textsc{ag}\, ey = 0)$
  $\forall exy(x \neq y \rightarrow \textsc{th}\, ex \circ \textsc{th}\, ey = 0)$
  $\forall x\, \mathsf{John}\, x \circ \mathsf{Mary}\, x = 0$
- The first two postulates here put functionality requirements on AG and TH; the third says that there is a constant-constant clash between John and Mary.
- Note that combination of smash $e \circ \textsc{ag}\, ex \circ \mathsf{John}\, x \circ \ldots$ with smash $e \circ \textsc{ag}\, ey \circ \mathsf{Mary}\, y \circ \ldots$ is now no longer possible (i.e. $= 0$). The two frames are not unifiable.
- There seems to be no need to demand that *all* attributes are functional (as Löbner and Petersen & Osswald do). E.g. the PARENT relation does not seem to be naturally decomposable into functions.

Evaluating Facts/Frames wrt Facts/Frames

- Our aim is to model frames in type logic and it seems a promising idea to use facts for that.
- We also need entities such as possible worlds or situations in order to be able to assign truth and falsity to atomic sentences.
- For these we will use our facts/frames/feature structures as well and the picture that emerges will not be *very* far from the one emerging from Barsalou's matching procedures.
- Atoms in our lattices of facts carry maximal information that is compatible with consistency. They will play the role of possible worlds.
- We diverge from Veltman's (1985) approach here, in which subsets of filters of facts ('data sets') are used to play a role analogous to that of possible situations.

Partial Type Theory I

- Evaluating frames wrt frames does not necessarily lead to an outcome that is well-defined, so it is best to base our formalisation on a partial type theory.
- I will use the logic TY_2^3 (3-valued type theory with 2 basic types $\neq$ t) here. See Muskens (1995).
- The logic is based on the values **t**, **f**, and **n**, ordered in the following ways. (The first diagram gives a logical ordering, the second an approximation ordering.)



Partial Type Theory II

- $\neg$, $\wedge$, and $\vee$ behave in the usual Strong Kleene way, but there are now more truth functions that can be considered, e.g. $\otimes$, which denotes meet in the approximation ordering, and T, which returns **t** if its input was **t** and **f** otherwise. Here are the relevant truth tables:
-

$\otimes$	**t**	**f**	**n**
t	**t**	**n**	**n**
f	**n**	**f**	**n**
n	**n**	**n**	**n**

T	
t	**t**
f	**f**
n	**f**

- Equalities $a = b$ are always bivalent (evaluate as **t** or **f**) in the system of Muskens (1995). Axioms are also required to have this property.
- Note that if φ and ψ are bivalent and φ entails ψ, then $\varphi \otimes \psi$ is true iff φ is true and $\varphi \otimes \psi$ is false iff ψ is false.

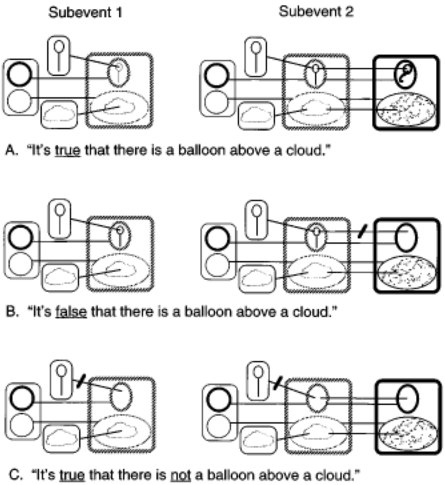
Partial Type Theory III

- Muskens (1995) gives details of the partial type theory considered here and of a four-valued variant. The theory is used to implement a form of Situation Semantics in a broadly Montagovian framework.
- Muskens (1999) and Wintein and Muskens (2012) give analytic Gentzen sequent calculi and tableau systems for the first-order fragments of these logics.
- Many rules (e.g. λ-conversion, rules for $\exists$ and $\forall$) work as usual. Only the propositional part of the logic diverges.

Truth and Falsity Relative to Facts in Type Logic I

- We begin by defining the operator ϑ ('that') of type $\mathfrak{f}\mathfrak{f}t$ as $\lambda g \lambda f.(f \leq g \otimes g \circ f \neq 0)$.
- Instead of $\vartheta\varphi$ (where φ is of type $\mathfrak{f}$) we typically write $[\varphi]$.
- So $[\text{smash}\, e \circ \text{ag}\, ex \circ \text{John}\, x \circ \text{th}\, ey \circ \text{window}\, y]$ is the function that, given any fact $f \neq 0$, returns a true statement if the fact $\text{smash}\, e \circ \text{ag}\, ex \circ \text{John}\, x \circ \text{th}\, ey \circ \text{window}\, y$ is incorporated in f and a false statement if that fact is incompatible with f.
- If g is neither incorporated in f nor incompatible with f then $[g]f$ is undefined, i.e. gets the value **n**.
- Note that our approach with respect to falsity is slightly different from Barsalou's, in which falsity was just lack of truth.

Truth and Negation in Barsalou (1999)



- Truth as correspondence between simulation and perceived situation.
- Falsity as lack of such correspondence.
- Negation as mismatch between simulator and simulation.

Defining Logical Operators

We can now define further operators in a standard way.

$$
\begin{aligned}
\text{not} &= \lambda p \lambda f.\neg pf \\
\text{and} &= \lambda pq \lambda f.(pf \wedge qf) \\
\text{or} &= \lambda pq \lambda f.(pf \vee qf) \\
\text{every} &= \lambda P' P \lambda f \forall x.(P'xf \rightarrow Pxf) \\
\text{a} &= \lambda P' P \lambda f \exists x.(P'xf \wedge Pxf) \\
\text{the} &= \lambda P' P \lambda f \exists x.(\forall y(P'yf \leftrightarrow x = y) \wedge Pxf)
\end{aligned}
$$

Here p and q are of type $\mathfrak{f}t$ (propositions), while P and P' are of type $e\mathfrak{f}t$ (properties).

Adding Some Content Words (to be revised)

Let's set up a mini-fragment.

$$
\begin{array}{rcl}
\text{talk} & = & \lambda x \lambda f \exists e.[\mathit{talk}\, e \circ \textsc{ag}\, ex]f \\
\text{man} & = & \lambda x \lambda f \exists y.[\mathit{person}\, x \circ \mathit{sex}\, xy \circ \mathit{male}\, y]f \\
\text{apple} & = & \lambda x \lambda f.[\mathit{apple}\, x]f \\
\text{eat} & = & \lambda yx \lambda f \exists e.[\mathit{eat}\, e \circ \textsc{ag}\, ex \circ \textsc{th}\, ey]f
\end{array}
$$

- Here *talk* is if type $e\mathfrak{f}$, *sex* is if type $ee\mathfrak{f}$, etc.
- We have made no provision for modifiers and will need to change that.

A man eats an apple

The following are equivalent.

- (a man)λx.(a apple)λy.eat yx
- $\lambda f \exists x(\exists z[\mathit{person}\, x \circ \mathit{sex}\, xz \circ \mathit{male}\, z]f \wedge \exists y[\mathit{apple}\, y]f \wedge \exists e.[\mathit{eat}\, e \circ \textsc{ag}\, ex \circ \textsc{th}\, ey]f)$
- $\lambda f \exists xzye.[\mathit{person}\, x \circ \mathit{sex}\, xz \circ \mathit{male}\, z]f \wedge [\mathit{apple}\, y]f \wedge [\mathit{eat}\, e \circ \textsc{ag}\, ex \circ \textsc{th}\, ey]f$

If the last of these terms is applied to a world w_0 it is possible to simplify things a bit further.

- $\exists xzye.[\mathit{person}\, x \circ \mathit{sex}\, xz \circ \mathit{male}\, z]w_0 \wedge [\mathit{apple}\, y]w_0 \wedge [\mathit{eat}\, e \circ \textsc{ag}\, ex \circ \textsc{th}\, ey]w_0$
- $\exists xzye.[\mathit{person}\, x \circ \mathit{sex}\, xz \circ \mathit{male}\, z \circ \mathit{apple}\, y \circ \mathit{eat}\, e \circ \textsc{ag}\, ex \circ \textsc{th}\, ey]w_0$

Every man eats an apple

The following are equivalent.

- (every man)λx.(a apple)λy.eat yx
- $\lambda f \forall x(\exists z[person\, x \circ sex\, xz \circ male\, z]f \rightarrow$
  $\exists ye([apple\, y]f \wedge [eat\, e \circ \textrm{AG}\, ex \circ \textrm{TH}\, ey]f))$

Applying to w_0 gives

- $\forall x(\exists z[person\, x \circ sex\, xz \circ male\, z]w_0 \rightarrow$
  $\exists ye([apple\, y]w_0 \wedge [eat\, e \circ \textrm{AG}\, ex \circ \textrm{TH}\, ey]w_0))$
- $\forall x(\exists z[person\, x \circ sex\, xz \circ male\, z]w_0 \rightarrow$
  $\exists ye[apple\, y \circ eat\, e \circ \textrm{AG}\, ex \circ \textrm{TH}\, ey]w_0)$

Allowing for Modification

We need to allow for the possibility of further modification of frames along the phrasal projection of words. Our translations will be altered as follows.

$$\begin{array}{rcl} \text{talk} & = & \lambda x \lambda M \lambda f \exists e.[M\, talk\, e \circ \text{AG}\, ex]f \\ \text{man} & = & \lambda x \lambda M \lambda f \exists y.[M\, person\, x \circ sex\, xy \circ male\, y]f \\ \text{apple} & = & \lambda x \lambda M \lambda f.[M\, apple\, x]f \\ \text{eat} & = & \lambda yx \lambda M \lambda f \exists e.[M\, eat\, e \circ \text{AG}\, ex \circ \text{TH}\, ey]f \end{array}$$

Here M is of type $(e\mathfrak{f})e\mathfrak{f}$. The treatment shares important aspects with that of Champollion (2014) (but M modifies $e\mathfrak{f}$ type objects, not just events).

A Word on Composition

- A basic way of combining meanings in type logical semantics is by way of application.
- But application is not enough, we also need some form of type shifting in order to glue meanings together.
- Type shifts, inasfar as they are purely combinatorial, can usually be thought of as the result of applying some linear combinator (a pure lambda term in which each binder binds exactly one variable) to an item (e.g. a quantifier or a verb meaning).
- We are interested in type $\mathfrak{f}t$ terms that are obtained from lexical items and linear combinators with the help of application.
- (This is the set-up of Philippe de Groote's (2001; 2002) Abstract Categorial Grammars and my (2001; 2003) Lambda Grammars. Reduce to a limited set of type shifters if a more conventional set-up is wanted.)

Red Apple

- Let $\mathsf{I} := \lambda Z.Z$, with Z of type $e\mathfrak{f}$ and let red be of type $(e\mathfrak{f})e\mathfrak{f}$.
- No modification: use $\lambda x.\text{apple}\, x\, \mathsf{I}$, which reduces to $\lambda x \lambda f.[\mathit{apple}\, x]f$, the earlier translation.
- $\lambda x.\text{apple}\, x\, \text{red} \rightsquigarrow \lambda x \lambda f.[\text{red}\, \mathit{apple}\, x]f$, but now no further modification is possible.
- In order to allow for further modification 'lift' red to $\lambda N \lambda M.N(\lambda Z.M(\text{red}\, Z))$, with N of type $((e\mathfrak{f})(e\mathfrak{f}))\mathfrak{f}t$, and apply to apple. The result is $\lambda x \lambda M \lambda f.[M(\text{red}\, \mathit{apple})\, x]f$.
- Get rid of the λM by applying to I.
- But what to do with $\lambda x \lambda f.[\text{red}\, \mathit{apple}\, x]f$? It obviously is not what we want. We want a real frame.

Red Apples and Red Grapefruits

- We need postulates such as
  $$\forall f \forall x(f = (\text{red } \mathit{apple}\, x) \leftrightarrow \exists yz(f = (\mathit{apple}\, x \circ \text{SKIN}\, xy \circ \text{COLOUR}\, yz \circ \mathit{red}\, z)))$$
  $$\forall f \forall x(f = (\text{red } \mathit{grapefruit}\, x) \leftrightarrow \exists yz(f = (\mathit{grapefruit}\, x \circ \text{INSIDE}\, xy \circ \text{COLOUR}\, yz \circ \mathit{red}\, z)))$$
- Such postulates embody lexical knowledge that must be provided to the system in some way.
- $\lambda x \lambda f.[\text{red } \mathit{apple}\, x]f$ is now equivalent with $\lambda x \lambda f \exists yz.[\mathit{apple}\, x \circ \text{SKIN}\, xy \circ \text{COLOUR}\, yz \circ \mathit{red}\, z]f$

Conclusion and Further Desiderata

- We have shown how a form of lexical semantics—frame semantics in the tradition of Barsalou and Löbner—can be integrated within the Montagovian framework.
- A formalisation of frames along the lines of Veltman's Data Logic was given. Matching relations between frames led to notions of relative truth and falsity.
- Further logical operators led to a system not far removed from the variant of Situation Semantics in Muskens (1995).
- What is needed in this respect is a treatment of the propositional attitudes within this frame-based semantics.
- And, obviously, far more needs to be done in order for the full potential of lexical semantics to be realised.

References I

L. Barsalou. Frames, Concepts, and Conceptual Fields. In A. Lehrer and E. F. Kittay, editors, *Frames, Fields, and Contrasts*, pages 21–74. Lawrence Erlbaum Associates, 1992.

L. Barsalou. Perceptual Symbol Systems. *Behavioral and Brain Sciences*, 22:577–660, 1999.

L. Champollion. The interaction of compositional semantics and event semantics. `http://semanticsarchive.net/Archive/2JhMWFhN/`, 2014.

R. Cooper. Records and record types in semantic theory. *Journal of Logic and Computation*, 15(2):99–112, 2005.

Ph. de Groote. Towards Abstract Categorial Grammars. In *Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter, Proceedings of the Conference*, pages 148–155, Toulouse, France, 2001. ACL.

References II

Ph. de Groote. Tree-Adjoining Grammars as Abstract Categorial Grammars. In *TAG+6, Proceedings of the Sixth International Workshop on Tree Adjoining Grammars and Related Frameworks*, pages 145–150, 2002.

P. J. Hayes. The Logic of Frames. In D. Metzing, editor, *Frame Conceptions and Text Understanding*. Walter de Gruyter & Co., Berlin, 1979.

M. Johnson. Logic and Feature Structures. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, Sydney, Australia, 1991.

S. Löbner. Functional Concepts and Frames. Available at `http://semanticsarchive.net/Archive/jI1NGEwO/`, year?

R.A. Muskens. *Meaning and Partiality*. CSLI, Stanford, 1995.

References III

R.A. Muskens. On Partial and Paraconsistent Logics. *Notre Dame Journal of Formal Logic*, 40(3):352–374, 1999.

R.A. Muskens. Categorial Grammar and Lexical-Functional Grammar. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG01 Conference, University of Hong Kong*, pages 259–279, Stanford CA, 2001. CSLI Publications. `http://cslipublications.stanford.edu/LFG/6/lfg01.html`.

R.A. Muskens. Language, Lambdas, and Logic. In Geert-Jan Kruijff and Richard Oehrle, editors, *Resource Sensitivity in Binding and Anaphora*, Studies in Linguistics and Philosophy, pages 23–54. Kluwer, 2003.

W. Petersen and T. Osswald. Concept Composition in Frames—Focusing on Genitive Constructions. In Th. Gamerschlag, D. Gerland, R. Osswald, and W. Petersen, editors, *Frames and Concept Types: Applications in Language and Philosophy*, volume 94 of *Studies in Linguistics and Philosophy*. Springer, Dordrecht, 2013.

References IV

J. Prinz. *Furnishing the Mind*. MIT Press, 2002.

G. Smolka. A Feature Logic with Subsorts, 1988. Technical Report. IBM Deutschland.

Achille C. Varzi. Spatial reasoning and ontology: Parts, wholes, and locations. In Marco Aiello, Ian Pratt-Hartmann, and Johan van Benthem, editors, *Handbook of Spatial Logics*, pages 945–1038. Springer, 2007.

F. Veltman. *Logics for Conditionals*. PhD thesis, University of Amsterdam, 1985.

Stefan Wintein and Reinhard Muskens. A Calculus for Belnap's Logic in Which Each Proof Consists of Two Trees. *Logique & Analyse*, (220): 643–656, 2012.