Article:

Trueman, Rob orcid.org/0000-0002-2908-7985 (2025) Determinate compositionality. Inquiry-An interdisciplinary journal of philosophy. ISSN 0020-174X

0.1080/0020174X.2025.2517392

Determinate Compositionality

Robert Trueman

First-order logic is *obviously* compositional. So it is embarrassing that, given standard syntactic and semantic assumptions, first-order logic does not count as compositional by the standard definition (§1). The standard syntax was handed down by Tarski (1935), but a number of philosophers have suggested that Frege's (1879, 1893) earlier approach had already provided a way out of this problem with compositionality.[1] Unfortunately, Pickel and Rabern (2022) have shown that, by itself, this Fregean syntax does not help (§2). However, in this paper, I will argue that the Fregean syntax motivates a redefinition of compositionality, and that this combined Fregean package successfully delivers the compositionality of first-order logic (§§3–4). Tarskians can also secure compositionality in a similar way, but only if they adopt a key Fregean idea: bound variables are not semantically significant constituents of the formulas in which they appear (§5).

1 The problem for Tarskians

Informally, the principle of compositionality states that the semantic value of a complex expression is determined by the semantic values of its constituent expressions. Here is one standard way of formalising this principle (where η is a syntactic operation for forming complex expressions, and $[\![\ldots]\!]$ is a function that maps an expression to its semantic value):[2]

Comp: If $\mathbf{E} = \eta(\mathbf{e}_1, \ldots, \mathbf{e}_n)$ and $\mathbf{F} = \eta(\mathbf{f}_1, \ldots, \mathbf{f}_n)$, and $[\![\mathbf{e}_i]\!] = [\![\mathbf{f}_i]\!]$ for each $i \leq n$, then $[\![\mathbf{E}]\!] = [\![\mathbf{F}]\!]$.

Intuitively, first-order logic should be compositional. However, arranging the syntax and semantics so that first-order logic satisfies *Comp* is surprisingly difficult. In this section, I will present the problem as it confronts the Tarskian approach to first-order logic.

A *Tarskian syntax* starts with disjoint countable infinities of names, of variables, and of predicates of each adicity.[3] Names and variables are grouped together into the

[1] For example: Dummett 1981a: ch. 2; Evans 1977: §2; Partee 2013; Button and Walsh 2018: 13–15; Wehmeier 2018, 2021; Potter 2020: ch. 5.

[2] This is essentially the principle that Pagin and Westerståhl (2010: 254) call Subst($\equiv_\mu$). However, unlike Pagin and Westertåhl, I have assumed that every well-formed expression is meaningful. This is just a simplifying assumption, and is not essential to any of the discussion below.

[3] We also require that none of $\wedge$, $\neg$ and $\forall$ is used as a name, variable or predicate.

syntactic category of *terms*. The formulas of a Tarskian syntax are then constructed using the following syntactic operations (and no others):[4]

- *T-Pred*$_n$ concatenates an n-adic predicate, $\mathbf{P}$, with n terms, $\mathbf{t}_1, \ldots, \mathbf{t}_n$:
  $\textit{T-Pred}_n(\mathbf{P}, \mathbf{t}_1, \ldots, \mathbf{t}_n) = \mathbf{Pt}_1 \ldots \mathbf{t}_n$.
- *T-Conj* flanks $\wedge$ with two formulas, $\mathbf{A}$ and $\mathbf{B}$, and surrounds the result with parentheses:
  $\textit{T-Conj}(\mathbf{A}, \mathbf{B}) = (\mathbf{A} \wedge \mathbf{B})$.
- *T-Neg* concatenates $\neg$ with a formula, $\mathbf{A}$:
  $\textit{T-Neg}(\mathbf{A}) = \neg\mathbf{A}$.
- *T-All* concatenates $\forall$ with a variable, $\mathbf{x}$, and a formula, $\mathbf{A}$:
  $\textit{T-All}(\mathbf{x}, \mathbf{A}) = \forall\mathbf{xA}$.

First-order logic does not have an intended interpretation, and so we cannot single out just one correct semantic theory. However, we can offer a *generalized semantic theory*, i.e. a theory of all the possible interpretations that we might give first-order logic.[5] Here is one standard way of developing a generalized semantics for Tarski's syntax. A model, $\mathcal{M}$, consists of a non-empty set, $D_{\mathcal{M}}$,[6] and a function, $[\![\ldots]\!]_{\mathcal{M}}$. If $\mathbf{a}$ is a name, then $[\![\mathbf{a}]\!]_{\mathcal{M}} \in D_{\mathcal{M}}$; and if $\mathbf{P}$ is an n-adic predicate, then $[\![\mathbf{P}]\!]_{\mathcal{M}} \subseteq D^n_{\mathcal{M}}$. A variable assignment over $\mathcal{M}$, σ, is a function from variables to members of $D_{\mathcal{M}}$. We then introduce a new function, $[\![\ldots]\!]_{\mathcal{M},\sigma}$, which maps each expression of the Tarskian syntax to its semantic value relative to $\mathcal{M}$ and σ (where $\mathbf{a}$ is any name, $\mathbf{x}$ is any variable, $\mathbf{t}_1, \ldots, \mathbf{t}_n$ are any n terms, $\mathbf{P}$ is any n-adic predicate, and $\mathbf{A}$ and $\mathbf{B}$ are any formulas):

(T:i) $[\![\mathbf{a}]\!]_{\mathcal{M},\sigma} = [\![\mathbf{a}]\!]_{\mathcal{M}}$.
(T:ii) $[\![\mathbf{x}]\!]_{\mathcal{M},\sigma} = \sigma(\mathbf{x})$.
(T:iii) $[\![\textit{T-Pred}_n(\mathbf{P}, \mathbf{t}_1, \ldots, \mathbf{t}_n)]\!]_{\mathcal{M},\sigma} = 1$ iff $\langle [\![\mathbf{t}_1]\!]_{\mathcal{M},\sigma}, \ldots, [\![\mathbf{t}_n]\!]_{\mathcal{M},\sigma} \rangle \in [\![\mathbf{P}]\!]_{\mathcal{M},\sigma}$.
(T:iv) $[\![\textit{T-Conj}(\mathbf{A}, \mathbf{B})]\!]_{\mathcal{M},\sigma} = 1$ iff $[\![\mathbf{A}]\!]_{\mathcal{M},\sigma} = [\![\mathbf{B}]\!]_{\mathcal{M},\sigma} = 1$.
(T:v) $[\![\textit{T-Neg}(\mathbf{A})]\!]_{\mathcal{M},\sigma} = 1$ iff $[\![\mathbf{A}]\!]_{\mathcal{M},\sigma} \neq 1$.
(T:vi) $[\![\textit{T-All}(\mathbf{x}, \mathbf{A})]\!]_{\mathcal{M},\sigma} = 1$ iff for all $o \in D_{\mathcal{M}}$, $[\![\mathbf{A}]\!]_{\mathcal{M},\sigma[\mathbf{x}:o]} = 1$, where $\sigma[\mathbf{x} : o]$ is a variable assignment that maps $\mathbf{x}$ to o, and agrees with σ for all other variables.

To make sure that $[\![\ldots]\!]_{\mathcal{M},\sigma}$ is defined for every Tarskian formula, $\mathbf{A}$, we should also add: if $[\![\mathbf{A}]\!]_{\mathcal{M},\sigma} \neq 1$, then $[\![\mathbf{A}]\!]_{\mathcal{M},\sigma} = 0$.

We can now lay out the problem that Tarskians have with compositionality. A generalized semantic theory should count as compositional iff it is a theory exclusively of compositional interpretations. More carefully, a generalized semantic theory is compositional relative to a syntax iff every semantic-value-function described

[4] Throughout this paper, I will use letters in boldface as metalinguistic variables ranging over expressions. Strictly speaking, we should use Quine-quotes in connection with these variables — e.g. $\ulcorner\mathbf{Pt}\urcorner$ — but I will omit them for readability.

[5] This terminology is due to Linnebo and Rayo (2012: 275). I will be careful about the distinction between a semantics and a generalized semantics only when the distinction matters.

[6] I am identifying domains with sets just for the sake of simplicity. We could take domains to be pluralities or properties instead, without making any substantial change to the argument of this paper.

by that theory is compositional relative to that syntax. It is easy to construct a model, $\mathcal{M}$, such that, for any σ, $[\![\forall x Rax]\!]_{\mathcal{M},\sigma} = 1 \neq [\![\forall x Rxa]\!]_{\mathcal{M},\sigma}$. Since σ can be any variable assignment, let $\sigma(x) = [\![a]\!]_{\mathcal{M}}$. We now have a violation of *Comp*: $[\![x]\!]_{\mathcal{M},\sigma} = [\![a]\!]_{\mathcal{M},\sigma}$, and so $[\![\textit{T-Pred}_2(R, a, x)]\!]_{\mathcal{M},\sigma} = [\![\textit{T-Pred}_2(R, x, a)]\!]_{\mathcal{M},\sigma}$ (by clause T:iii); but $[\![\textit{T-All}(x, \textit{T-Pred}_2(R, a, x))]\!]_{\mathcal{M},\sigma} \neq [\![\textit{T-All}(x, \textit{T-Pred}_2(R, x, a))]\!]_{\mathcal{M},\sigma}$.[7]

This is a well known problem for Tarskians. (It is a *problem*, rather than just a surprising result, because, intuitively, first-order logic is a *paradigm* of compositionality.[8]) There is also a well known solution. We can restore the compositionality of first-order logic simply by rethinking what we take the semantic value of a Tarskian expression to be: rather than taking each expression, **e**, to have its semantic value *relative to* a variable assignment, we could identify the semantic value of **e** on model $\mathcal{M}$ with a *function from* each variable assignment, σ, to $[\![\mathbf{e}]\!]_{\mathcal{M},\sigma}$. Wehmeier (2018) provides a clear presentation of the details,[9] but it is already easy to see how making this change solves the problem as it has been presented here. If $[\![\forall x Rax]\!]_{\mathcal{M},\sigma} \neq [\![\forall x Rxa]\!]_{\mathcal{M},\sigma}$, then $D_{\mathcal{M}}$ must have at least two members. In that case, there will be some variable assignment, σ', such that $[\![x]\!]_{\mathcal{M},\sigma'} \neq [\![a]\!]_{\mathcal{M},\sigma'}$. So, x and a will now have different semantic values on $\mathcal{M}$.[10]

However, there is a downside to reconceiving of semantic values as functions from variable assignments:

> Once denotations are compositionally defined in terms of assignment functions, these functions become part of the ontology, with the undesirable consequence that there is more in our ontology than the simple denotations found in the standard semantics. In particular, the semantics of a language has to refer to the variables of the language and thereby becomes language dependent. (Klein and Sternefeld 2017: 66)

To make this objection vivid, imagine that you were devising a formal theory of celestial mechanics. You would expect your ontology to include stars and planets and other astronomical odds and ends. But you would not expect it to include *functions from variable assignments to* stars and planets and so on. After all, you may be theorising *in* a language, but you are not theorising *about* a language. So, insofar as you

---

[7] It is important to emphasise that this problem is not an artefact of choosing to identify the semantic value of a formula with (a number representing) its truth-value. In fact, any choice of semantic values will do, so long as it is possible for a variable, **x**, to have the same semantic value as a name, **a**: one application of *Comp* will imply $[\![\textit{T-Pred}_2(R, \mathbf{a}, \mathbf{x})]\!] = [\![\textit{T-Pred}_2(R, \mathbf{x}, \mathbf{a})]\!]$, and then another will imply $[\![\textit{T-All}(\mathbf{x}, \textit{T-Pred}_2(R, \mathbf{a}, \mathbf{x}))]\!] = [\![\textit{T-All}(\mathbf{x}, \textit{T-Pred}_2(R, \mathbf{x}, \mathbf{a}))]\!]$. (A variant of the same problem will also arise if two variables can have the same semantic value.) The reason that I have chosen to work with a coarse-grained semantics is (yet again) to keep things simple.

[8] As Klein and Sternefeld (2017: 65) put it: 'Compositionality is at the heart of model theoretical semantics and its application to the semantics of natural language. As has become standard practice, linguists translate a fragment of English into an intensional extension of [first-order logic]. Yet, somewhat ironically and strangely, [first-order logic] itself is not compositional'.

[9] See also: Janssen 1997: §2.4; Rabern 2013.

[10] This approach also has the potentially unwanted implication that alphabetic variants have different semantic values (whenever the domain includes at least two objects). For a sophisticated modification of this approach that avoids this result, see Pickel and Rabern 2016: §5.

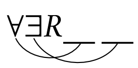
identify the ontology of a theory with the semantic values of its terms, you should not in general take those values to be functions from variable assignments.[11]

2 The problem for Fregeans

Tarskians get into trouble with compositionality because they assign semantic values to bound variables. You might hope, then, to avoid that trouble just by refusing to assign values to bound variables. One strategy for doing exactly that is suggested by Frege's (1879: §§9–11, 1893: §§8 & 21) account of quantifiers as second-level predicates.

Frege thought of predicates as sentences with holes poked in them. A *first-level* predicate is the result of deleting one or more names from a sentence. For example, from the sentence 'Joe loves Sharon', you can form the predicate '__ loves Sharon' by deleting 'Joe', or 'Joe loves __' by deleting 'Sharon'. A *second-level* predicate is then the result of deleting one or more first-level predicates from a sentence. For example, from the sentence 'Joe sleeps and Sharon reads', you can form 'Joe __ and Sharon reads' by deleting '__ sleeps', or 'Joe sleeps and Sharon __' by deleting '__ reads'.

According to Frege, first-order quantifiers are second-level predicates. For example, in 'Everybody sleeps', the first-level predicate '__ sleeps' is plugged into the second-level predicate 'Everybody __'. Or, in a more formal setting, $\forall x Fx$ is the result of plugging the first-level predicate $F\_$ into the second-level predicate $\forall x\_\_x$. Importantly, the variable in this quantifier should not be thought of as an independent syntactic unit, and it does not receive a semantic value. It is just there to indicate how the quantifier reaches into the first-level predicate that it takes as its argument. We need to indicate this to handle multiple-generality: we need to distinguish between $\forall x \exists y Rxy$, where $\forall x\_\_x$ takes $\exists y R\_y$ as its argument, and $\forall x \exists y Ryx$, where $\forall x\_\_x$ takes $\exists y Ry\_$ as its argument. Variables are one solution to this problem, but there are others. For example, we could use the Quine-Bourbaki notation, which draws track-lines between a quantifier and the gaps it reaches into. Here, for example, is how $\forall x \exists y Rxy$ would be written in this notation:[12]



The Quine-Bourbaki notation is especially perspicuous, but it is also a nightmare to typeset. So we will stick with variables, while bearing in mind that, for a Fregean, they are doing nothing more than the Quine-Bourbaki track-lines.

[11] This is what Wehmeier (2018: 213) calls the *ontological purity* worry. He helpfully distinguishes this from a related *language transcendence* worry. Although I have not articulated the latter worry here, it is worth noting that my preferred way of restoring compositionality (see §§3–4) avoids it as well.

[12] See Quine 1981: §12; Bourbaki 1954: ch. 1; Button and Walsh 2018: §1.4; Wehmeier 2018: §4.

Let's formalise this Fregean approach.[13] A *Fregean syntax* starts with disjoint countable infinities of names, of variables, and of simple predicates of each adicity.[14] (We will also assume that the variables have been ordered in some way.[15]) The complex expressions of a Fregean syntax are then constructed using the following syntactic operations (and no others):

- *F-Pred*$_n$ forms a sentence by concatenating a simple n-adic predicate, **P**, with n names, $\mathbf{a}_1, \ldots, \mathbf{a}_n$:
  $\textit{F-Pred}_n(\mathbf{P}, \mathbf{a}_1, \ldots, \mathbf{a}_n) = \mathbf{P}\mathbf{a}_1 \ldots \mathbf{a}_n$.
- *F-Conj* forms a sentence by flanking $\wedge$ with two sentences, **A** and **B**, and surrounding the result with parentheses:
  $\textit{F-Conj}(\mathbf{A}, \mathbf{B}) = (\mathbf{A} \wedge \mathbf{B})$.
- *F-Neg* forms a sentence by concatenating $\neg$ with a sentence, **A**:
  $\textit{F-Neg}(\mathbf{A}) = \neg\mathbf{A}$.
- *F-Ab* forms a complex monadic predicate by deleting every occurrence of a name, **a**, from a sentence in which **a** occurs, **A**:[16]
  $\textit{F-Ab}(\mathbf{a}, \mathbf{A}) = \mathbf{A}[\_\_/\mathbf{a}]$.[17]
- *F-All* forms a sentence by filling the gaps in a complex monadic predicate, **Q**, with the first variable, **x**, that does not occur in **Q**, and prefixing the result with $\forall\mathbf{x}$:[18]
  $\textit{F-All}(\mathbf{Q}) = \forall\mathbf{x}\mathbf{Q}[\mathbf{x}/\_\_]$.

It might be helpful to make three quick comments about this syntax. First, it is only a minor revision of the Tarskian syntax. If they both start with the same stock of simple expressions, then every sentence of the Fregean syntax will be a sentence (i.e. closed formula) of the Tarskian syntax. The converse will not be true, however, since *F-All* does not allow you to pick which variable to use. But this reflects the fact that, from a Fregean point of view, variables are not independent syntactic units; they are there merely to coordinate a quantifier with the gaps it is reaching into. For the same

[13] The following syntax is closely modelled on Wehmeier 2018, 2021.

[14] We obviously still require that none of $\wedge$, $\neg$ and $\forall$ is used as a name, variable or simple predicate.

[15] In the examples below, I assume in particular that x is the first variable and y is the second.

[16] For the purposes of *F-Ab*, we can say that a name occurs in a sentence iff it is a *part* of that sentence. Alternatively, we can define which names occur in a sentence in terms of how the sentence was constructed: let **c** be any name; **c** occurs in $\textit{F-Pred}_n(\mathbf{P}, \mathbf{a}_1, \ldots, \mathbf{a}_n)$ iff $\mathbf{c} = \mathbf{a}_1$ or … or $\mathbf{c} = \mathbf{a}_n$; **c** occurs in $\textit{F-Conj}(\mathbf{A}, \mathbf{B})$ iff **c** occurs in **A** or **B**; **c** occurs in $\textit{F-Neg}(\mathbf{A})$ iff **c** occurs in **A**; **c** occurs in $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$ iff $\mathbf{c} \neq \mathbf{a}$ and **c** occurs in **A**; **c** occurs in $\textit{F-All}(\mathbf{Q})$ iff **c** occurs in **Q**.

[17] In general, $\mathbf{E}[\mathbf{f}/\mathbf{e}]$ is the result of uniformly substituting **f** for **e** in **E**. This is the standard notion of substitution, defined on strings of symbols. (It is the kind of substitution that you can perform with the *Find & Replace* tool on a word processor.) I will introduce a more complex kind of substitution, *determinate substitution*, in §3.

[18] Again, for the purposes of *F-All*, we can say that a variable occurs in a complex predicate iff it is a part of that complex predicate. Alternatively, we can define which variables occur in a complex predicate in terms of how the complex predicate was constructed: let **x** be any variable; **x** does not occur in $\textit{F-Pred}_n(\mathbf{P}, \mathbf{a}_1, \ldots, \mathbf{a}_n)$; **x** occurs in $\textit{F-Conj}(\mathbf{A}, \mathbf{B})$ iff **x** occurs in **A** or **B**; **x** occurs in $\textit{F-Neg}(\mathbf{A})$ iff **x** occurs in **A**; **x** occurs in $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$ iff **x** occurs in **A**; **x** occurs in $\textit{F-All}(\mathbf{Q})$ iff **x** occurs in **Q** or **x** is the first variable not to occur in **Q**.

reason, the Fregean syntax does not permit the appearance of free variables: the open formula Fx, for example, is not well-formed in the Fregean syntax.

Second, *F-Pred* applies *simple* predicates to names, not complex ones. Complex predicates only ever appear as the arguments to quantifiers. This was not a forced choice: we could describe a Fregean syntax that allows complex predicates to be applied to names, if we wanted to (Trueman manuscript). However, that would be unhelpful in the current setting, where we are concerned with compositionality, which is meant to trace how the semantic values of *more complex* expressions depend on the semantic values of *simpler* expressions. (We can think of our restricted syntax as providing *canonical* Fregean construction histories for first-order sentences.)[19]

Third, *F-Ab* abstracts *monadic* complex predicates only.[20] This is just a convenient simplification: the one syntactic rule that acts on complex predicates is *F-All*, and it only takes monadic predicates. We could consider augmented versions of first-order logic, which require complex predicates of higher adicities,[21] but such innovations would change nothing in my discussion here.

We can now give a generalised semantics for this Fregean language. As before, a model, $\mathcal{M}$, consists of a non-empty set, $D_{\mathcal{M}}$, and a function, $[\![\ldots]\!]_{\mathcal{M}}$: if $\mathbf{a}$ is a name, then $[\![\mathbf{a}]\!]_{\mathcal{M}} \in D_{\mathcal{M}}$; and if $\mathbf{P}$ is a simple n-adic predicate, then $[\![\mathbf{P}]\!]_{\mathcal{M}} \subseteq D^n_{\mathcal{M}}$. However, we will no longer make any use of variable assignments. We will instead use $[\![\ldots]\!]_{\mathcal{M}}$ itself to map an expression of the Fregean language to its semantic value relative to $\mathcal{M}$ (where $\mathbf{a}$ is any name, $\mathbf{a}_1, \ldots, \mathbf{a}_n$ are any n names, $\mathbf{P}$ is any n-adic simple predicate, $\mathbf{Q}$ is any monadic complex predicate, and $\mathbf{A}$ and $\mathbf{B}$ are any sentences):

(F:i) $[\![\textit{F-Pred}_n(\mathbf{P}, \mathbf{a}_1, \ldots, \mathbf{a}_n)]\!]_{\mathcal{M}} = 1$ iff $\langle [\![\mathbf{a}_1]\!]_{\mathcal{M}}, \ldots, [\![\mathbf{a}_n]\!]_{\mathcal{M}} \rangle \in [\![\mathbf{P}]\!]_{\mathcal{M}}$.

(F:ii) $[\![\textit{F-Conj}(\mathbf{A}, \mathbf{B})]\!]_{\mathcal{M}} = 1$ iff $[\![\mathbf{A}]\!]_{\mathcal{M}} = [\![\mathbf{B}]\!]_{\mathcal{M}} = 1$.

(F:iii) $[\![\textit{F-Neg}(\mathbf{A})]\!]_{\mathcal{M}} = 1$ iff $[\![\mathbf{A}]\!]_{\mathcal{M}} \neq 1$.

(F:iv) If $\mathbf{a}$ occurs in $\mathbf{A}$: $[\![\textit{F-Ab}(\mathbf{a}, \mathbf{A})]\!]_{\mathcal{M}} = \{o \in D_{\mathcal{M}} : [\![\mathbf{A}]\!]_{\mathcal{M}[\mathbf{a}:o]} = 1\}$, where $\mathcal{M}[\mathbf{a} : o]$ is a model that maps $\mathbf{a}$ to o, but otherwise agrees with $\mathcal{M}$.[22,23]

(F:v) $[\![\textit{F-All}(\mathbf{Q})]\!]_{\mathcal{M}} = 1$ iff $[\![\mathbf{Q}]\!]_{\mathcal{M}} = D_{\mathcal{M}}$.

To make sure that $[\![\ldots]\!]_{\mathcal{M}}$ is defined for every Fregean sentence, $\mathbf{A}$, we should also add: if $[\![\mathbf{A}]\!]_{\mathcal{M}} \neq 1$, then $[\![\mathbf{A}]\!]_{\mathcal{M}} = 0$.

---

[19] In Dummett's (1981b: 271–2) terminology, we are here concerned with the *analysis* of a sentence into its *constituents*, rather than its *decomposition* into mere *components*.

[20] It should also be emphasised that $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$ deletes *every* occurrence of $\mathbf{a}$ from $\mathbf{A}$. So we cannot form *Ra_* from *Raa* by selectively deleting the second occurrence of a. However, this is no limitation in practice, since we have an infinite stock of names. For example, we can form *Ra_* by deleting b from *Rab*.

[21] For example, we could introduce an operator which takes a dyadic predicate as input, and outputs the ancestral of that predicate. See also Wehmeier 2018: 236–8.

[22] More precisely: $D_{\mathcal{M}[\mathbf{a}:o]} = D_{\mathcal{M}}$; $[\![\mathbf{a}]\!]_{\mathcal{M}[\mathbf{a}:o]} = o$; for each name $\mathbf{b}$ other than $\mathbf{a}$, $[\![\mathbf{b}]\!]_{\mathcal{M}[\mathbf{a}:o]} = [\![\mathbf{b}]\!]_{\mathcal{M}}$; and for each simple predicate $\mathbf{P}$, $[\![\mathbf{P}]\!]_{\mathcal{M}[\mathbf{a}:o]} = [\![\mathbf{P}]\!]_{\mathcal{M}}$.

[23] Complex predicates can be formed in more than one way, but (F:iv) assigns each complex predicate just one semantic value, no matter how it was formed. Let $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$ be well-formed (so $\mathbf{a}$ occurs in $\mathbf{A}$). $\textit{F-Ab}(\mathbf{a}, \mathbf{A}) = \textit{F-Ab}(\mathbf{b}, \mathbf{B})$ iff $\mathbf{B} = \mathbf{A}[\mathbf{b}/\mathbf{a}]$ and $\mathbf{A} = \mathbf{B}[\mathbf{a}/\mathbf{b}]$. It follows that, if $\textit{F-Ab}(\mathbf{a}, \mathbf{A}) = \textit{F-Ab}(\mathbf{b}, \mathbf{B})$, then $[\![\mathbf{A}]\!]_{\mathcal{M}[\mathbf{a}:o]} = [\![\mathbf{B}]\!]_{\mathcal{M}[\mathbf{b}:o]}$, for all o. See also Wehmeier 2018: 222–3.

Importantly, this Fregean semantics does not assign any semantic values to variables. As a result, Fregeans do not run into a problem with compositionality at the same point as Tarskians. Let $\mathcal{M}$ be such that $[\![\forall x Rax]\!]_{\mathcal{M}} = 1 \neq [\![\forall x Rxa]\!]_{\mathcal{M}}$. This does not immediately violate *Comp*: if $[\![\textit{F-All}(Ra\_)]\!]_{\mathcal{M}} = 1 \neq [\![\textit{F-All}(R\_a)]\!]_{\mathcal{M}}$, then $[\![Ra\_]\!]_{\mathcal{M}} = D_{\mathcal{M}} \neq [\![R\_a]\!]_{\mathcal{M}}$ (by clause F:v).

Unfortunately, however, as Pickel and Rabern (2022: §3.2) point out, Fregeans face their own version of the problem, one step later.[24] We still get a violation of *Comp*, but it is now generated by *F-Ab* rather than *F-All*. Let $\mathcal{M}$ be as before, but add that $[\![a]\!]_{\mathcal{M}} = [\![b]\!]_{\mathcal{M}}$. *Comp* implies that $[\![\textit{F-Ab}(b, \textit{F-Pred}_2(R, a, b))]\!]_{\mathcal{M}} = [\![\textit{F-Ab}(b, \textit{F-Pred}_2(R, b, a))]\!]_{\mathcal{M}}$. But, by hypothesis, $[\![\textit{F-Ab}(b, \textit{F-Pred}_2(R, a, b))]\!]_{\mathcal{M}} = [\![Ra\_]\!]_{\mathcal{M}} \neq [\![R\_a]\!]_{\mathcal{M}} = [\![\textit{F-Ab}(b, \textit{F-Pred}_2(R, b, a))]\!]_{\mathcal{M}}$.[25]

Like the Tarskians, Fregeans could restore compositionality by rethinking what they take the semantic value of a Fregean expression to be. In particular, rather than taking each expression, **e**, to have its semantic value *relative* to a model, $\mathcal{M}$, a Fregean could restore compositionality by indentifying the semantic value of **e** with a *function* from $\mathcal{M}$ to $[\![\mathbf{e}]\!]_{\mathcal{M}}$. (For details, see Wehmeier 2018: §3.) However, this solution is exactly as problematic for Fregeans as it was for Tarskians: reconceiving of semantic values in this way imports extraneous linguistic elements into ontologies where they do not belong.[26]

3 Introducing determinate compositionality

At this stage, it would be natural for a Fregean to complain that *Comp* is just the wrong way to formulate compositionality for their language. According to *Comp*, the semantic value of *F-Ab*(b, Rab) is a function of the semantic values of b and Rab. But neither b nor Rab actually appears in *F-Ab*(b, Rab): *F-Ab*(b, Rab) is what you get precisely by *deleting* b from Rab, i.e. $Ra\_$. Compositionality should require only that the semantic

[24] For a related problem, see Humberstone 2000: 3–5.

[25] Moreover, as Pickel and Rabern (2022: 985–6) also emphasise, even the appearance that the Fregeans managed to delay the problem a little longer than Tarskians is superficial. It would be easy to augment the Tarskian language with an abstraction operator, λ, which combined with a variable and a formula to make a complex predicate, $\lambda\mathbf{x}.\mathbf{A}$. If we did, we could then have $\forall$ attach to complex predicates, very much as it does in the Fregean syntax. (In fact, this is standard practice in higher-order settings.)

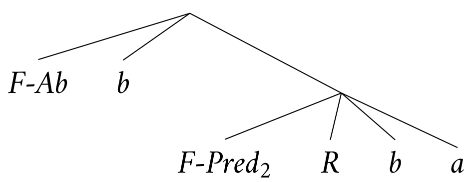
[26] Wehmeier (2018: 242) suggests that 'friends of Fregean abstraction' need not think of functions from models (or, as he would officially prefer, *schmodels*) as themselves constructed out of 'bits of syntax', and so as being linguistic entities. He takes this to show that conceiving of semantic values as functions from models does not violate any requirement of ontological purity. However, even if functions from models are not linguistic entities in the narrow sense of being syntactic constructions, it still seems reasonable to count them as linguistic in a broader sense; this would be analogous to the sense in which a function from natural numbers can itself be described as 'arithmetic'. And, more importantly, whether we classify functions from models as 'linguistic' or not, the point remains that in many cases they will still be extraneous to the intended ontology of a theory: for example, the ontology of a theory of celestial mechanics should include stars and planets, but not functions from models.

value of $Ra\_$ be a function of the values of the expressions that actually appear in that predicate, i.e. R and a.[27]

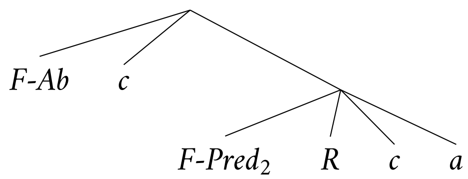
I think that this Fregean response is fundamentally correct. But the challenge is to spell out exactly how *Comp* ought to be revised. The obvious starting point would be something like: the semantic value of a complex expression is a function of the semantic values of its *parts*. However, Pickel and Rabern (2022: 993–4) rightly object to this suggestion, on the grounds that the parts of a Fregean sentence do not generally provide sufficient structure to calculate the value of that sentence. Consider the sentence $\forall x Rxx$. The Fregean semantics determines the value of this sentence from the value of $R\_\_$, but $R\_\_$ is not a part of $\forall x Rxx$ in the straightforward mereological sense: $R\_\_$ has two parts — namely, the two gap markers — that are not parts of $\forall x Rxx$, and mereological parthood is transitive. This problem cannot be dodged merely by erasing these gap markers, since that would just leave R, which is a simple dyadic predicate, not a complex monadic one. (Moreover, there are other universal generalisations that have R as their only semantically significant *part*, such as $\forall x \forall y Rxy$.) Of course, you might still say that $R\_\_$ is a part of $\forall x Rxx$ in the non-mereological sense that $\forall x Rxx$ is syntactically derived from $R\_\_$; but that would lead us straight back to *Comp*.

Importantly, though, there is another way for Fregeans to revise *Comp*. One of the noteworthy features of complex predicates is that, despite being entirely unambiguous (see fn. 23), they can be constructed in more than one way. For example, $\textit{F-Ab}(b, \textit{F-Pred}_2(R, b, a))$ is R_a, but so is $\textit{F-Ab}(c, \textit{F-Pred}_2(R, c, a))$. These construction histories disagree with each other: b appears in the first history but not in the second; and c appears in the second history but not in the first. However, it would obviously be absurd for a Fregean to prefer one of these histories over the other. The better option is for a Fregean to take a broadly *supervaluationist* attitude: the *determinate* syntactic facts about R_a are the facts that all of its histories agree on; everything else about the syntax of R_a is *indeterminate*.

It will be useful to think of construction histories as trees. Each construction rule listed in §2 is represented by a finite labelled ordered tree: the first daughter of the tree is labelled with the rule, and the remaining daughters represent the inputs to the rule; any leaf that is not labelled by a construction rule is labelled by a simple expression. (Leaves are treated as the limiting cases of trees, and so simple expressions represent themselves.) Here, for example, are two trees that represent R_a:



[27] For example, see Dummett 1981b: 286.

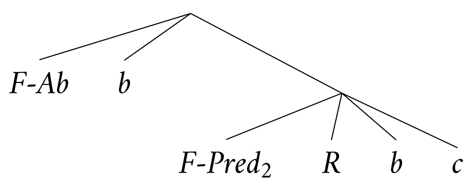


Both trees feature *a* at the same node. (A node on a finite ordered tree can be represented with a finite series of directions telling you how to get from the root to that node. For example, the node of *a* in either of the trees above can be given as $\langle 0, 3, 4 \rangle$: the 0 represents the root, and the subsequent numbers tell you which branches to take.) Moreover, *every* tree that represents *R_a* features *a* at that node: two trees that represent the same expression can differ only over which names were abstracted in the construction of a given predicate.[28] So it is determinate that *a* appears at that node. More generally, expression **e** *determinately appears* at node *n* in the history of expression **E** iff a tree that represents **e** appears at *n* in every tree that represents **E**. Clearly, neither *b* nor *c* determinately appears at any node in the history of *R_a*.

Fregeans can now offer a revised formulation of compositionality: compositionality should require only that the semantic value of a complex expression be a function of the expressions that *determinately* appear in its construction history. To make this more precise, let's first define *determinate substitution* (where **E**, **F**, **e** and **f** are any Fregean expressions):[29]

> **F** is the result of determinately substituting **f** for **e** at node *n* in the history of **E** iff:
>
> (i) **e** determinately appears at *n* in the history of **E**
> (ii) substituting some tree that represents **f** for a tree that represents **e** at *n* in some tree that represents **E** yields a tree that represents **F**
> (iii) **f** determinately appears at *n* in the history of **F**

Take the following tree as an example:



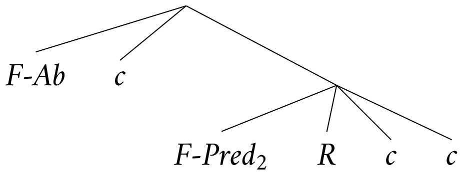
This tree represents *R_c*. It is a result of substituting *c* for *a* at node $\langle 0, 3, 4 \rangle$ in a tree that represents *R_a*. Moreover, *a* determinately appears at that node in the history

[28] Here it is crucial that we work with the restricted syntactic rules given in §2, which do not permit us to form a sentence by applying a *complex* predicate to some names.

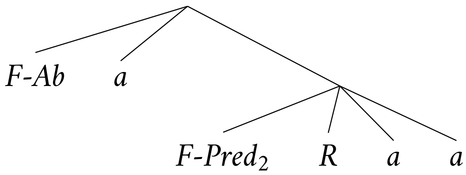
[29] In clause (ii) below, I take for granted the notion of substituting one sub-tree for another at a given node in a larger tree. This is well-defined on finite ordered trees.

of *R*_*a*, and *c* determinately appears at that node in the history of *R*_*c*. So *R*_*c* is the result of determinately substituting *c* for *a* at ⟨0, 3, 4⟩ in the history of *R*_*a*.

It might also be useful to give an example of a *failed* determinate substitution. Take this tree:



This tree represents *R*_ _. It is a result of substituting *c* for *a* at node ⟨0, 3, 4⟩ in a tree that represents *R*_*a*. However, *c* does *not* determinately appear at that node in the history of *R*_ _, as this alternative history for *R*_ _ demonstrates:



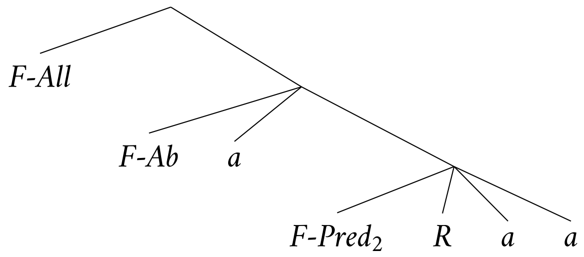
So *R*_ _ is *not* the result of determinately substituting *c* for *a* at ⟨0, 3, 4⟩ in the history of *R*_*a*.[30]

So far we have focussed on determinately substituting one name for another, but we can determinately substitute any kind of expression. Let's consider a case of determiantely substituting one complex predicate for another. Consider the following tree:

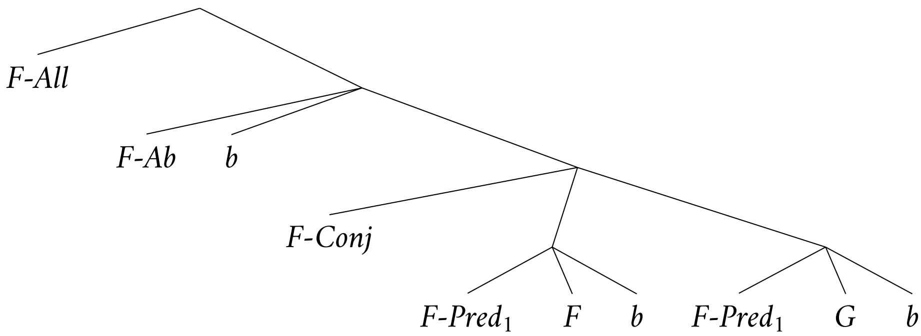
---

[30] More generally, at most one expression can be the result of determinately substituting **f** for **e** at *n* in **E**, which justifies the use of the definite article. *Proof sketch:* Assume that **e** determinately appears at node *n* in the history of **E**. Let $\mathbb{e}_1$ and $\mathbb{e}_2$ be trees that both represent **e**; and let $\mathbb{E}_1$ be a tree that represents **E** and has $\mathbb{e}_1$ at *n*, and $\mathbb{E}_2$ be a tree that represents **E** and has $\mathbb{e}_2$ at *n*. Similarly, let $\mathbb{f}_1$ and $\mathbb{f}_2$ both be trees that represent **f**; and let $\mathbb{F}_1$ be the result of substituting $\mathbb{f}_1$ for $\mathbb{e}_1$ at *n* in $\mathbb{E}_1$, and $\mathbb{F}_2$ be the result of substituting $\mathbb{f}_2$ for $\mathbb{e}_2$ at *n* in $\mathbb{F}_2$. If $\mathbb{F}_1$ and $\mathbb{F}_2$ represent different expressions, then either (i) node *n* of $\mathbb{E}_1$ is under an application of *F-Ab* that abstracts on a name that determinately appears somewhere in the history of **f**, or (ii) node *n* of $\mathbb{E}_2$ is under an application of *F-Ab* that abstracts on a name that determinately appears somewhere in the history of **f**. In case (i), it follows that **f** does not determinately appear in the history of the expression represented by $\mathbb{F}_1$, and so that expression is not the result of determinately substituting **f** for **e** at *n* in **E**. In case (ii), the expression represented by $\mathbb{F}_2$ is not the result of determinately substituting **f** for **e** at *n* in **E**, for exactly the same reason.

It is also easy to see that, if **e** determinately appears at *n* in the history of **E**, and if **f** can be grammatically substituted for **e**, then the result of determinately substituting **f** for **e** at *n* in **E** exists. Since we have an infinite stock of names, we can pick a tree to represent **E** that does not abstract on any names that determinately appear in **f**.



This tree represents $\forall x Rxx$. It features a sub-tree that represents $R\_\_$ at node $\langle 0, 2\rangle$, as will any tree that represents $\forall x Rxx$. We can substitute some other sub-tree at $\langle 0, 2\rangle$, for example:



Now there is a sub-tree which represents $F\_ \wedge G\_$ at node $\langle 0, 2\rangle$. The whole tree represents $\forall x(Fx \wedge Gx)$, and any tree which represents that sentence will have a sub-tree which represents $F\_ \wedge G\_$ at node $\langle 0, 2\rangle$. So $\forall x(Fx \wedge Gx)$ is the result of determinately substituting $F\_ \wedge G\_$ for $R\_\_$ at node $\langle 0, 2\rangle$ in the history of $\forall x Rxx$.

We can now formalise a new compositionality principle, based on determinate substitution:

D-Comp: If **F** is the result of determinately substituting **f** for **e** at some node in the history of **E**, and $[\![\mathbf{e}]\!] = [\![\mathbf{f}]\!]$, then $[\![\mathbf{E}]\!] = [\![\mathbf{F}]\!]$.

And, happily, a simple induction on complexity shows that the Fregean syntax and semantics given in §2 satisfy *D-Comp*.[31]

[31] *Proof sketch:* Every simple expression trivially satisfies *D-Comp* on every model, and it is obvious that every rule other than *F-Ab* preserves *D-Comp*: if the inputs to the rule satisfy *D-Comp* on $\mathcal{M}$, then so does the output of the rule. So let $\mathbf{E} = \textit{F-Ab}(\mathbf{g}, \mathbf{G})$, and assume for induction that **G** satisfies *D-Comp* on every model. Assume also that $[\![\mathbf{e}]\!]_{\mathcal{M}} = [\![\mathbf{f}]\!]_{\mathcal{M}}$, and let **F** be the result of determinately substituting **f** for **e** somewhere in the history of $\textit{F-Ab}(\mathbf{g}, \mathbf{G})$. We can set aside the limiting case, where $\mathbf{e} = \textit{F-Ab}(\mathbf{g}, \mathbf{G})$, as trivial. We can also discard the case where $\mathbf{e} = \mathbf{g}$ as impossible: **g** does not determinately appear anywhere in the history of $\textit{F-Ab}(\mathbf{g}, \mathbf{G})$. So **F** must be $\textit{F-Ab}(\mathbf{g}, \mathbf{G}^*)$, where $\mathbf{G}^*$ is the result of determinately substituting **f** for **e** somewhere in the history of **G**. It follows that **g** does not determinately appear anywhere in the histories of **e** or **f**, and thus $[\![\mathbf{e}]\!]_{\mathcal{M}[\mathbf{g}:o]} = [\![\mathbf{e}]\!]_{\mathcal{M}} = [\![\mathbf{f}]\!]_{\mathcal{M}} = [\![\mathbf{f}]\!]_{\mathcal{M}[\mathbf{g}:o]}$ for all o. So, since we have assumed that **G** satisfies *D-Comp* on every model, $[\![\mathbf{G}]\!]_{\mathcal{M}[\mathbf{g}:o]} = [\![\mathbf{G}^*]\!]_{\mathcal{M}[\mathbf{g}:o]}$ for all o, and thus $[\![\textit{F-Ab}(\mathbf{g}, \mathbf{G})]\!]_{\mathcal{M}} = [\![\textit{F-Ab}(\mathbf{g}, \mathbf{G}^*)]\!]_{\mathcal{M}}$.

4 Determinate compositionality for Fregeans

I want to suggest that *D-Comp* is the right way for a Fregean to formalise compositionality. Informally, the principle of compositionality states that the semantic value of a complex expression is determined by the semantic values of its constituents. But what are the 'constituents' of a complex expression? The orthodox answer is that they are the expressions from which the complex expression is syntactically derived. But that answer looks wrongheaded, from a Fregean point of view. $F\text{-}Ab(b, Rab)$ is syntactically derived from b and Rab, but it would be bizarre for a Fregean to maintain that the semantic value of $F\text{-}Ab(b, Rab)$ depends on the particular value of b. We move from the value of Rab to the value of $F\text{-}Ab(b, Rab)$ precisely by *discarding* the particular value of b! For a Fregean, then, the right answer is that the constituents of a complex expression are the expressions from which it is *determinately* derived.

Of course, to take the measure of my suggestion, we need to consider what a Fregean might stand to lose by swapping *Comp* for *D-Comp*. In general, *D-Comp* is strictly weaker than *Comp*: *Comp* implies *D-Comp*, but not *vice versa*.[32] However, when we focus on a particular language, the difference between *Comp* and *D-Comp* depends on how much the syntax leaves indeterminate. Relative to a syntax that is entirely determinate, i.e. a syntax in which every complex expression has just one construction history, *D-Comp* is equivalent to *Comp*.[33] And, at the opposite extreme, in a syntax that is so indeterminate that no expression determinately appears anywhere in the history of any other expression, *D-Comp* is vacuous.

The Fregean syntax is an intermediate case that displays a limited degree of syntactic indeterminacy. In this Fregean case, *D-Comp* is weaker than *Comp*. *Comp* is a form of *direct* compositionality, in the sense that it requires that the semantic value of a complex expression be a function of its *immediate* constituents (Pagin and Westerståhl 2010: §3.3). Relative to the Fregean syntax, *D-Comp* is not direct. Indeed, it is unclear what a Fregean should say the immediate constituents of $Ra\_$, for example, even are. But what is clear is that evaluating a complex Fregean predicate will always require working all the way down to the leaves on at least one branch of the construction history.

Pagin and Westerståhl issue the following warning about indirect forms of compositionality that determine the semantic values of complex expressions only from the values of their simple constituents:

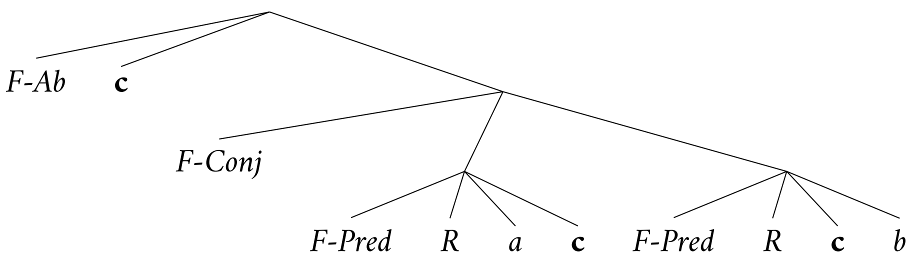
[indirect compositionality does] not serve the language users very well: the meaning operation r_α that corresponds to a complex syntactic operation α cannot be predicted from its build-up

[32] We have already seen that *D-Comp* does not imply *Comp*, because the Fregean syntax and semantics satisfy the former but not the latter. To show that *Comp* implies *D-Comp*, we need to assume that, if $\mathbf{E} = \eta(\mathbf{e}_1, \ldots, \mathbf{e}_n)$, then $[\![\mathbf{e}_i]\!] = [\![\mathbf{e}_i]\!]$ for each $i \leq n$. Given this assumption, *Comp* requires that, if $\mathbf{F} = \eta(\mathbf{e}_1, \ldots, \mathbf{e}_{m-1}, \mathbf{f}_m, \mathbf{e}_{m+1}, \ldots, \mathbf{e}_n)$, and $[\![\mathbf{e}_m]\!] = [\![\mathbf{f}_m]\!]$, then $[\![\mathbf{E}]\!] = [\![\mathbf{F}]\!]$, *regardless* of whether $\mathbf{e}_m$ determinately appears in the history of $\mathbf{E}$ or $\mathbf{f}_m$ determinately appears in the history of $\mathbf{F}$.

[33] I am here assuming that, if $\eta(\mathbf{e}_1, \ldots, \mathbf{e}_n)$ and $\eta(\mathbf{f}_1, \ldots, \mathbf{f}_n)$ are both well-formed, then so is $\eta(\mathbf{e}_1, \ldots, \mathbf{e}_{m-1}, \mathbf{f}_m, \mathbf{e}_{m+1}, \ldots \mathbf{e}_n)$. Given this assumption, you can get from *D-Comp* to *Comp* in a fully determinate syntax just by substituting one expression at a time.

out of simpler syntactic operations and their corresponding meaning operations. (Pagin and Westerståhl 2010: 255)

Fortunately, however, Pagin and Westerståhl's worry does not apply to this case. The Fregean semantics determines the semantic value of each complex Fregean expression from the semantic values of the expressions that determinately appear in its history. (This is the important difference between *D-Comp* and a compositionality principle defined in terms of the *parts* of a complex expression.) It is obvious that clauses (F:i)–(F:iii) are unproblematic. (F:v) is also straightforward (which is noteworthy, since this is where the compositionality principle defined in terms of parthood foundered): *D-Comp* requires that the semantic value of a complex expression be determined by the semantic values of *all* the expressions that determinately appear in its history, and that includes complex predicates. That leaves (F:iv). *F-Ab* is the only operation that introduces indeterminacy into the Fregean syntax. It is indeterminate which name was deleted in the construction of a given complex predicate. But it is determinate where the deleted name determinately appears in the history of the sentence being abstracted on, and it is determinate that the deleted name is not any of the names that determinately appear somewhere in the history of the resulting predicate. That is enough for (F:iv) to determine the semantic value of a complex predicate from the semantic values of its determinate constituents. For example, every construction history for *Ra*_ ∧ *R*_*b* must have this form, where **c** is any name other than *a* or *b*:



If we hold fixed the semantic values of *R*, *a* and *b* — i.e. the expressions that determinately appear in the history of the predicate — then (F:iv) will assign every tree of this form the same semantic value (see fn. 23). The same goes for all other complex Fregean predicates.

All of this has been good news for Fregeans. However, I should end this section by sounding a note of caution.[34] A number of linguists and philosophers have argued that compositionality is trivial, because *any* semantics for *any* syntax can be transformed into a compositional semantics for that syntax.[35] The consensus view appears to be that these arguments are correct, but *only* when we place no additional constraints on the kind of compositional semantics we are after.[36] So, although compositionality

[34] Thanks to an anonymous reviewer for encouraging me to discuss this issue.

[35] One of the best known arguments to that effect was offered by Zadrozny (1994: §2), but Westerståhl (1998) provides a helpful overview of other such arguments.

[36] For example, see: Westerståhl 1998; Wehmeier 2024.

may be trivial *by itself*, it can still be an important part of a non-trivial package of requirements on a semantics. Importantly, though, all of the existing literature on this topic takes *Comp* (or one of its near-equivalents) to be the proper formalisation of compositionality. It is, therefore, not *immediately* obvious that the consensus view should be extended to *D-Comp*. Nonetheless, there is reason for optimism, on two counts. First, the Fregean semantics is so similar to the standard Tarskian semantics that it seems a safe bet that it will satisfy any additional constraints that we might plausibly impose. Second, there are examples of semantic theories that cannot be made to satisfy *Comp* without violating plausible extra constraints, and which do not involve any syntactic indeterminacy; it is to be expected, then, that they also cannot be made to satisfy *D-Comp* without violating those same constraints.[37] However, a proper formal investigation of this matter lies well beyond the scope of this short note.

5 Determinate compositionality for Tarskians

The Fregean syntax and semantics satisfy *D-Comp*, which, I have just argued, is the right way for a Fregean to formalise compositionality. I will now end the paper by asking whether Tarskians could solve their problem with compositionality in the same way.

The first thing to note is that the Tarskian syntax given in §1 is entirely determinate: every formula has a unique construction history. So, relative to *that* syntax, there is no difference between *D-Comp* and *Comp*. If a Tarskian wants to pull these two principles apart, then they will need to introduce some indeterminacy into their syntax. That is certainly possible. For example, they could offer a revised version of *T-All*, call it *T-All**, on the stipulation that $\textit{T-All}^*(\mathbf{x}, \mathbf{A})$ is the result of translating $\textit{T-All}(\mathbf{x}, \mathbf{A})$ into the Quine-Bourbaki notation described in §2.[38] In this revised syntax, $\textit{T-All}^*(\mathbf{x}, \mathbf{A}) = \textit{T-All}^*(\mathbf{y}, \mathbf{A}[\mathbf{y}/\mathbf{x}])$ whenever $\mathbf{y}$ does not appear in $\mathbf{A}$, and so no bound variable ever determinately appears anywhere in the history of any formula.

Revising the Tarskian syntax in this way does not necessitate any substantial changes to the standard Tarskain semantics from §1. All we need to do is substitute *T-All** for *T-All* in (T:vi). The standard semantics will then satisfy *D-Comp*, relative to the revised Tarskian syntax.[39] Moreover, the standard semantics determines the truth-value of a formula in the revised syntax from the semantic values of the expressions that determinately appear in its construction history.[40] So a *reformed Tarskian*,

[37] Indirect contexts provide a helpful example: it seems that 'Joe believes that **A**' can have a different semantic value from 'Joe believes that **B**', even when **A** has the same semantic value as **B**; but 'Joe believes that…' does not introduce any syntatic indeterminacy. Another example, given by Wehmeier (2024: §3.5), is Humberstone's (2022: §2) syntactic idempotent exclusive disjunction.

[38] Pickel and Rabern (2022: 993 fn. 28) make exactly this suggestion.

[39] This can be demonstrated by a straightforward reworking of the proof given in fn. 31.

[40] The key to demonstrating this is to rework fn. 23 to show that each universal generalisation has just one semantic value, no matter how it was formed.

who adopted the revised syntax and settled for *D-Comp* instead of *Comp*, would be in exactly the same boat as a Fregean.

To some extent, I am here agreeing with Pickel and Rabern (2022). The main point of their paper is that Tarskians can match any Fregean solution to the problem of compositionality. But they also go further, and claim that Fregeanism is best understood as *variablisim*, an extreme form of Tarskianism which states that natural language names should be formalised as Tarskian variables (Pickel and Rabern 2022: 988–90). They claim this because they take $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$ to be a way of effectively *binding* **a**, and a variable is precisely a term that can be bound. (It obviously does not matter that Fregeans tend to use letters from the beginning of the alphabet for their 'names', and letters from the end for their 'variables'.)

However, it is at very least deeply misleading to describe Fregeanism as a kind of variabilism. Fregeans would bristle at the claim that $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$ *binds* **a**. They would prefer to say that $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$ *deletes* **a**. Pickel and Rabern (2022: 991 fn. 25 & 993 fn. 29) anticipate this reply, but deny that we can make anything substantial of the Fregean rhetoric around 'deletion'. But we are now in a position to spell out the important syntactic consequence of deleting a name: **a** does not determinately appear anywhere in the history of $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$. Moreover, we have a Fregean story about why this matters for semantics: their preferred version of compositionality is *D-Comp*, which implies only that $\textit{F-Ab}(\mathbf{a}, \mathbf{A})$ is determined by the values of the expressions that do determinately appear in its history.

This brings us to a point worth emphasising. Compositionality requires that the semantic value of a complex expression be determined by the semantic values of its 'constituents'. When we formalise compositionality as *Comp*, we take a stand on what the semantically significant 'constituents' of a complex expression are: they are all the expressions from which the complex expression is syntactically derived. When we formalise it as *D-Comp*, we take a different stand: the semantically significant constituents of a complex expression are all the expressions from which the complex expression is *determinately* derived.

So, reformed Tarskians do not *just* tweak the standard Tarskian syntax. By swapping *Comp* for *D-Comp*, they also change their view on what counts as a semantically significant constituent of a complex expression. In particular, they come to deny that *bound variables* ever count: no bound variable determinately appears in the history of any formula in the revised Tarskian syntax; so *D-Comp* does not ever require the semantic value of a formula in that syntax to be partly determined by the semantic value of a bound variable.

Dummett long ago claimed that there was no real difference between Tarskianism and Fregeanism:

> in the standard [Tarskian] form of explanation, a free variable is treated exactly as if it were a proper name at every stage in the step-by-step construction of a given sentence up to that at which a quantifier is to be prefixed which will bind that variable: at that stage, however, it is treated exactly as if it were one of the [gap markers] Frege uses to indicate the argument-place

in a predicate. Hence we have no real contrast with Frege's explanation of the matter at all, but essentially the very same explanation. (Dummett 1981a: 17)

Now, Dummett was not *quite* right. *Standard* Tarskianism *is* different from Fregeanism. If a Tarskian sticks by the standard syntax presented in §1, then they will treat variables as significant constituents of formulas, no matter whether the variables are free or bound. (That is why they get into trouble with compositionality.) But Dummett was right about *reformed* Tarskianism. For a reformed Tarskian, *free* variables are semantically significant, because they are determinate constituents of the formulas in which they appear. But the moment they are bound, they cease to be determinate constituents, and so cease to make a compositional contribution to the value of a formula. In this sense, for a reformed Tarskian, to *bind* a variable is really just to *delete* it.

Acknowledgments. I would like to thank Tim Button, Salvatore Florio, Chris Jay, Nick Jones, Bryan Pickel, Kai Wehmeier, and the anonymous reviewers for this journal.

References

Bourbaki, Nicolas (1954). *Éléments de Mathématique: Théorie des Ensembles*. Paris: Hermann.

Button, Tim and Sean Walsh (2018). *Philosophy and Model Theory*. Oxford: Oxford University Press.

Dummett, Michael (1981a). *Frege: Philosophy of Language*. 2nd ed. London: Duckworth.

— (1981b). *The Interpretation of Frege's Philosophy*. London: Duckworth.

Evans, Gareth (1977). 'Pronouns, quantifiers, and relative clauses (I)'. *Canadian Journal of Philosophy* 7, pp.467–536.

Frege, Gottlob (1879). *Concept-Script, a formula language modeled upon that of arithemtic, for pure thought*. Breslau: Koebner.

— (1893). *The Basic Laws of Arithmetic*. Vol. I. Jena: Pohle.

Humberstone, Lloyd (2000). 'What *Fa* says about *a*'. *Dialectica* 54, pp.3–28.

— (2022). 'Idempotent variations on the theme of exclusive disjunction'. *Studia Logica* 110, pp.7–28.

Janssen, Theo M. V. (1997). 'Compositionality'. In: *Handbook of Logic and Language*. Cambridge, MA: MIT Press, pp.417–473.

Klein, Udo and Wolfgang Sternefeld (2017). 'Same but different: An alphabetically innocent compositional predicate logic'. *Journal of Philosophical Logic* 46, pp.65–95.

Linnebo, Øystein and Agustín Rayo (2012). 'Hierarchies ontological and ideological'. *Mind* 121, pp.269–308.

Pagin, Peter and Dag Westerståhl (2010). 'Compositionality I: Definitions and variants'. *Philosophy Compass* 5, pp.250–264.

Partee, Barbara (2013). 'The starring role of quantifiers in the history of formal semantics'. In: *The Logica Yearbook 2012*. Ed. by Vít Punčochář and Petr Švarný. London: College Publications, pp.113–136.

Pickel, Bryan and Brian Rabern (2016). 'The Antinomy of the Variable: A Tarskian resolution'. *The Journal of Philosophy* 113, pp.137–170.

Pickel, Bryan and Brian Rabern (2022). 'Against Fregean quantification'. *Ergo* 9, pp.971–1007.

Potter, Michael (2020). *The Rise of Analytic Philosophy, 1879–1930*. New York, NY: Routledge.

Quine, Willard Van Orman (1981). *Mathematical Logic*. Revised edition. Cambridge, MA: Harvard University Press.

Rabern, Brian (2013). 'Monsters in Kaplan's logic of demonstratives'. *Philosophical Studies* 164, pp.393–404.

Tarski, Alfred (1935). 'The concept of truth in formalized languages'. In: Tarski 1956, pp.152–278.

— (1956). *Logic, Semantics, Metamathematics: Papers from 1923 to 1938*. Oxford: Clarendon Press.

Trueman, Robert (manuscript). 'Fregean predication'.

Wehmeier, Kai (2018). 'The proper treatment of variables in predicate logic'. *Linguistics and Philosophy* 41, pp.209–249.

— (2021). 'Are quantifiers intensional operators?' *Inquiry* 64, pp.511–532.

— (2024). 'On the logical substantiveness of compositionality'. *Linguistics and Philosophy* 47, pp.1071–1087.

Westerståhl, Dag (1998). 'On mathematical proofs of the vacuity of compositionality'. *Linguistics and Philosophy* 21, pp.635–643.

Zadrozny, Wlodek (1994). 'From compositional to systematic semantics'. *Linguistics and Philosophy* 17, pp.329–342.