Are Quantifiers Intensional Operators?

Kai F. Wehmeier[*]

Logic and Philosophy of Science and Linguistics

University of California, Irvine

1 Introduction

By an *intensional operator* (simpliciter) I shall mean any expression O that combines with sentences ϕ to form well-formed expressions (usually sentences) $\mathsf{O}\phi$ and whose extension $[\![\mathsf{O}]\!]^{\mathfrak{M},i}$ at an index i in a model $\mathfrak{M}$ takes sentential intensions, i.e. functions from indices to truth values, as arguments.[1] A *strongly intensional operator* is an intensional operator O such that there are models $\mathfrak{M}$, indices i and sentential intensions δ and ε for which $[\![\mathsf{O}]\!]^{\mathfrak{M},i}(\delta) \neq [\![\mathsf{O}]\!]^{\mathfrak{M},i}(\varepsilon)$ even though $\delta(i) = \varepsilon(i)$. A *weakly intensional operator* is an intensional operator that isn't strongly intensional. An *extensional operator* is a

[*]Thanks to the editors of this special issue and an anonymous referee, as well as Greg Scontras, for helpful comments and suggestions. This paper was presented at the Center for Language, Information and Philosophy (CLIP) of the University of Cologne, Germany, and at the Department of Linguistics at the University of Frankfurt, Germany, both in October of 2017. I am grateful to the audiences for valuable feedback, especially Daniel Gutzmann, Stefan Hinterwimmer, and Ede Zimmermann. Funding was provided by the National Endowment for the Humanities (Grant No. FA-232235-16). Any views, findings, conclusions, or recommendations expressed in this article do not necessarily represent those of the National Endowment for the Humanities.

[1]Thus categorial grammar places intensional operators in some category X/S, where X is usually S, and within intensional type theory they are of some type $\langle\langle s,t\rangle,a\rangle$. I here assume a framework for intensional semantics of the sort employed by von Fintel and Heim (2011), where, if O is of category S/S, the type a is t.

sentential operator O whose extension $[\![\mathsf{O}]\!]^{\mathfrak{M},i}$ at any index i in any model $\mathfrak{M}$ takes truth values as arguments.

In the present paper I examine the view that *quantifiers* are intensional operators, with variable assignments playing the role of indices.[2] Certain formulations of extensional type theory indeed support such a view.[3] Others, however, do not, and anyway the view is at least *prima facie* in tension with Frege's (1893) insight that the first-order quantifiers are ordinary second-order functions, an insight central to present-day generalized quantifier theory.

In section 2, we first examine what is perhaps the most familiar version of type theory, here called $\mathcal{L}_{\mathbf{T}}$, where the quantifiers are introduced syncategorematically (i.e. don't have lexical entries). This formulation of the theory doesn't really help settle the question whether quantifiers are intensional operators: On the one hand, it suggests a negative answer, because it doesn't assign the quantifiers any extensions at all, so that our definition of intensional operators is inapplicable. On the other hand, it suggests a positive answer, because its syncategorematic quantifier clauses are structurally identical to the familiar syncategorematic clauses for modal and temporal operators. Since there are independent reasons to be dissatisfied with $\mathcal{L}_{\mathbf{T}}$, we next turn to another popular version of extensional type theory, $\mathcal{L}_{\mathbf{T}}^{\lambda}$, in which syncategorematic lambda abstraction enables lexical entries for the quantifiers. In $\mathcal{L}_{\mathbf{T}}^{\lambda}$, the quantifiers themselves are not intensional operators. Lambda abstraction, however, has an intensional flavor, which the quantifier-lambda combination inherits, and it's not implausible that it is really this

[2] I will be somewhat sloppy in my employment of the terms *quantifier* and *operator*, using them at times for expressions, at times for their extensions. I will also shift back and forth between taking the existential quantifier to be $\exists$ and taking it to be $\exists\mathsf{v}$. More pedantic distinctions are no doubt desirable but would make this paper an even more soporific read.

[3] For a somewhat recent advocate of a form of this view, see Belnap (2005), who emphasizes the "massive likeness of the practices of 'extensional' and 'intensional' semantics" (p. 3) and goes so far as to claim that it is "an illogical and harmful historical aberration that it is seldom made explicit that *quantifiers are non-truth-functional connectives*" (p. 6; emphasis in the original).

combination that corresponds to the syncategorematic quantifier of $\mathcal{L}_{\mathbf{T}}$. So $\mathcal{L}_{\mathbf{T}}^{\lambda}$ doesn't unequivocally answer our question either, and in any case, the lack of a lexical entry for abstraction in $\mathcal{L}_{\mathbf{T}}^{\lambda}$ is still unsatisfactory.

We thus turn, in section 3, to two formulations of type theory in which *every* operator, including the variable-binders, has a lexical entry. One version, $\mathcal{L}_{\mathbf{T}^s}^{\lambda}$, includes the lambda operator; here the quantifiers themselves behave as they do in $\mathcal{L}_{\mathbf{T}}^{\lambda}$ and thus aren't intensional operators; the lambda operator, however, is. The other version, $\mathcal{L}_{\mathbf{T}^s}$, doesn't include lambda abstraction, and here the quantifiers themselves literally *are* intensional operators.

In section 4, I argue that the semantics for $\mathcal{L}_{\mathbf{T}^s}^{\lambda}$ and $\mathcal{L}_{\mathbf{T}^s}$, though compositional and in accord with Frege's Conjecture, are ontologically unsatisfactory due to syntactic contamination of the type hierarchy.

Section 5 introduces a Frege-inspired treatment of extensional type theory that escapes this contamination issue. Crucially, it turns out that under a Fregean construal, neither quantifiers nor abstraction operators are in any sense intensional operators.

In the final section 6, I point out that Fregean type theories appear to be the most satisfactory options. As quantifiers aren't intensional operators in these best versions of type theory, I conclude that the appearance of quantifiers as intensional operators in other, inferior systems cannot be taken as reflecting a fundamental semantic property. I also briefly raise the question whether, with respect to natural language, the concept of an intensional operator is a fruitful one.

2 Syncategorematic Type Theory

The set $\mathbf{T}$ of *extensional types* is inductively generated from the basic types e and t by allowing the formation of a type $\langle a, b\rangle$ from any types a and b. The *extensional type hierarchy* over a non-empty set D is the family $(D_a)_{a\in\mathbf{T}}$ such that

- $D_e = D$ (*individuals*);

- $D_t = \{0, 1\}$ (*truth values*);
- $D_{\langle a,b\rangle} = D_b^{D_a}$ (*functions* from D_a to D_b).

The type-theoretical language $\mathcal{L}_\mathbf{T}$ has as *primitive symbols*

- for each $a \in \mathbf{T}$, a countably infinite set C_a of ("non-logical") *constants* of type a;
- for each $a \in \mathbf{T}$, a countably infinite set V_a of *variables* of type a;
- the sentential *connectives* $\neg$ and $\wedge$;
- the existential *quantifier* $\exists$.

The *well-formed expressions* (wfe's) of $\mathcal{L}_\mathbf{T}$ and their types are defined as follows.

1. Every $c \in C_a$ is an $\mathcal{L}_\mathbf{T}$-wfe of type a.
2. Every $\mathsf{v} \in V_a$ is an $\mathcal{L}_\mathbf{T}$-wfe of type a.
3. The connectives $\neg$ and $\wedge$ are $\mathcal{L}_\mathbf{T}$-wfes of types $\langle t, t\rangle$ and $\langle t, \langle t, t\rangle\rangle$, respectively.
4. If α is an $\mathcal{L}_\mathbf{T}$-wfe of type $\langle a, b\rangle$ and β an $\mathcal{L}_\mathbf{T}$-wfe of type a, then $\alpha\beta$ is an $\mathcal{L}_\mathbf{T}$-wfe of type b.
5. If ϕ is an $\mathcal{L}_\mathbf{T}$-wfe of type t and v a variable of any type, $\exists\mathsf{v}\phi$ is an $\mathcal{L}_\mathbf{T}$-wfe of type t.

A *model* $\mathfrak{M}$ for $\mathcal{L}_\mathbf{T}$ is a pair $\langle D, \mathcal{I}\rangle$ consisting of a non-empty set D (the *domain* of $\mathfrak{M}$) and an *interpretation function* $\mathcal{I}$ that maps, for each $a \in \mathbf{T}$, every element of C_a to an element of D_a.

A *variable assignment over* D is a function that maps, for each $a \in \mathbf{T}$, every member of V_a to an element of D_a. A variable assignment *in a model* $\mathfrak{M}$ is a variable assignment over the model's domain. We let $\aleph_D$ be the set of all variable assignments over D.

We define the *extensions* $[\![\alpha]\!]^{\mathfrak{M},g}$ of $\mathcal{L}_\mathbf{T}$-wfe's α relative to a model $\mathfrak{M} = \langle D, \mathcal{I}\rangle$ and a variable assignment g in such a way that whenever α is of type a, $[\![\alpha]\!]^{\mathfrak{M},g}$ is in D_a. Clauses 1–3 below are lexical entries, while 4 and 5 are composition rules.[4]

[4]Underlined lambdas, as they occur in the clauses for $\neg$ and $\wedge$, belong to the semantic metalanguage. We will shortly encounter object-linguistic lambdas as well, which will not be adorned by underlining.

1. For $a \in \mathbf{T}$ and $\alpha \in C_a$, $[\![\alpha]\!]^{\mathfrak{M},g} = \mathcal{I}(\alpha)$.

2. For $a \in \mathbf{T}$ and $\mathsf{v} \in V_a$, $[\![\mathsf{v}]\!]^{\mathfrak{M},g} = g(\mathsf{v})$.

3. $[\![\neg]\!]^{\mathfrak{M},g} = \underline{\lambda} n \in \{0,1\}\,.\,1 - n$ and $[\![\wedge]\!]^{\mathfrak{M},g} = \underline{\lambda} n \in \{0,1\}\,.\,\underline{\lambda} m \in \{0,1\}\,.\ \min\{n,m\}$.

4. "Functional Application": $[\![\alpha\beta]\!]^{\mathfrak{M},g} = [\![\alpha]\!]^{\mathfrak{M},g}([\![\beta]\!]^{\mathfrak{M},g})$.

5. $[\![\exists\mathsf{v}\phi]\!]^{\mathfrak{M},g} = \max\{[\![\phi]\!]^{\mathfrak{M},h} \mid h \in \aleph_D, h \sim_{\mathsf{v}} g\}$, where $\sim_{\mathsf{v}}$ is the relation of v-variance between assignments.

Strictly speaking, asking whether the quantifier $\exists\mathsf{v}$ as it occurs in $\mathcal{L}_{\mathbf{T}}$ is an intensional operator doesn't make sense, because our definition refers to the operator's extension at an index, and while it is perhaps natural to think of assignments as indices, our semantics for $\mathcal{L}_{\mathbf{T}}$ doesn't assign $\exists\mathsf{v}$ an extension relative to an assignment. In a broader sense, however, it's hard to deny that $\exists\mathsf{v}$ does *look* like an intensional operator. For consider the syncategorematic clause for the possibility operator $\Diamond$ of modal logic that one finds in logic textbooks: $\Diamond\phi$ is true at world w if ϕ is true at some world v that is accessible from w. This is structurally identical to the syncategorematic quantifier clause of $\mathcal{L}_{\mathbf{T}}$: $\exists\mathsf{v}\phi$ is true at assignment g if ϕ is true at some assignment h that is a v-variant of g. Still, it would be nice if we could confirm this impression in a system with a lexical entry for the quantifier—after all, in semantics textbooks the possibility operator *is* given a lexical entry, revealing it to take intensions as arguments.

Moreover, we should trust a formal system only to the extent that it exhibits theoretical virtues. One relevant virtue here is compositionality. It's clear that extensions relative to a fixed $\mathfrak{M}$ and g do *not* abide by the principle of compositionality: By clause 5, it is not just the extension of ϕ relative to $\mathfrak{M}$ and g that figures into the determination of the extension of $\exists\mathsf{v}\phi$ relative to $\mathfrak{M}$ and g; rather, we must take into account the extensions of ϕ relative to $\mathfrak{M}$ and h for a whole range of assignments h. So let the *assignment intension* $[\![\alpha]\!]^{\mathfrak{M}}$ of α in $\mathfrak{M}$ be the function with domain $\aleph_D$ that maps each assignment g over D to the extension $[\![\alpha]\!]^{\mathfrak{M},g}$ of α relative to $\mathfrak{M}$ and g. Assignment intensions have a better chance of satisfying compositionality, since $[\![\exists\mathsf{v}\phi]\!]^{\mathfrak{M}}$ is the function that maps any

g to $\max\{[\![\phi]\!]^{\mathfrak{M}}(h) \mid h \in \aleph_D, h \sim_{\mathsf{v}} g\}$ and is therefore a function of the assignment intension $[\![\phi]\!]^{\mathfrak{M}}$ of ϕ in $\mathfrak{M}$.

Even so, it is debatable whether, strictly speaking, our assignment intensions for $\mathcal{L}_{\mathbf{T}}$ are compositional. The Principle of Compositionality requires that for every syntactic rule ν there is a semantic operation r_ν such that the meaning $\mu(\nu(u_1, \ldots, u_n))$ of $\nu(u_1, \ldots, u_n)$ is equal to $r_\nu(\mu(u_1), \ldots, \mu(u_n))$, where $\mu(u)$ is the meaning of the expression u.[5] Now our syntax suggests that negations and existential quantifications are obtained by the same syntactic composition rule, to wit, concatenation: Negations $\neg\phi$ arise by concatenating $\neg$ and ϕ; existential quantifications $\exists\mathsf{v}\phi$ arise by concatenating $\exists\mathsf{v}$ and ϕ. By compositionality, they should be subject to the same semantic rule; but that is not so—negations are evaluated according to the functional application rule 4, while existential quantifications are evaluated according to the syncategorematic existential quantifier rule 5.[6]

Besides a possible violation of Compositionality, there are a couple of additional reasons to find the quantifier clause unsatisfactory. One such reason is that the syncategorematic introduction of the existential quantifier, that is, the inclusion of a special-purpose rule for wfe's governed by this particular operator, obscures the semantic parallelism between the various kinds of quantifiers: "some," "all," "no," "the," "exactly one," "at most seventeen," "infinitely many" etc. In a language that includes several of these, a tailor-made syncategorematic semantic rule would be required in each case, which fails to bring out the fact, discovered by Frege and central to contemporary generalized quantifier theory, that all of these quantifiers can be construed as higher-order functions. In other words, the semantic composition rule is intuitively the same in all these cases (viz., functional application), the differences arising out of the *lexical* meanings of the quantifier expressions; this isn't reflected in a semantics with no lexical

[5]See e.g. Pagin and Westerstahl (2010).

[6]Technically there's some wiggle room: We could make the syntactic rules for negation and existential quantification artificially distinct, e.g. by introducing parentheses and writing the negation of ϕ as $\neg(\phi)$ but the existential quantification of ϕ as $(\exists\mathsf{v}\phi)$; this difference would allow us to pair them with different semantic rules. This smells of cheating, though.

entries, but a whole smorgasbord of syncategorematic rules, for the quantifiers.

A final reason for dissatisfaction with $\mathcal{L}_{\mathbf{T}}$ is specific to a particular way of pursuing compositional semantics, espoused e.g. by Heim and Kratzer (1998) and guided by "Frege's Conjecture," according to which semantic composition is functional application.[7] Rule 5 for existentially quantified wfe's isn't an instance of functional application and therefore violates Frege's Conjecture; so anyone guided by Frege's Conjecture ought to be wary about syncategorematic clauses.

There is thus considerable pressure to replace the syncategorematic treatment of the existential quantifier with a categorematic one. One way to attempt this would be to count strings $\exists \mathsf{v}$ as wfe's, assign them a type, let rule 5 in the definition of well-formed expressions be subsumed by clause 4, directly assign extensions $[\![\exists \mathsf{v}]\!]^{\mathfrak{M},g}$ to the newly recognized lexical items $\exists \mathsf{v}$, and let the syncategorematic clause 5 in the definition of extensions be subsumed by the functional application clause 4, all the while ensuring that all wfe's of $\mathcal{L}_{\mathbf{T}}$ retain their original extensions. As is well known, this is impossible.[8]

A more promising, and quite popular, strategy is to pass the syncategorematic buck from quantification to abstraction. That is, we can give the existential quantifier a *categorematic* treatment as long as we allow ourselves the *syncategorematic* introduction of an abstraction operator.[9] Concretely, we replace $\mathcal{L}_{\mathbf{T}}$ with a new language $\mathcal{L}_{\mathbf{T}}^{\lambda}$ whose primitive symbols are those of $\mathcal{L}_{\mathbf{T}}$, except that the symbol $\exists$ is replaced with infinitely many symbols $\exists^{a}$, one for each $a \in \mathbf{T}$, and that the abstractor symbol λ as well as

[7]Heim and Kratzer themselves violate Frege's Conjecture with their Predicate Modification and Predicate Abstraction rules. As they explain (1998, §4.3.2), Predicate Modification can be brought into the fold of functional application in a number of ways, and in any case, the rule is of no particular *logical* interest. Predicate Abstraction is another matter, to which we return below.

[8]If $\exists \mathsf{v}\phi$ is a wfe of $\mathcal{L}_{\mathbf{T}}$, ϕ must be of type t, and $\exists \mathsf{v}\phi$ itself is also of type t. Hence $\exists \mathsf{v}$ would have to be of type $\langle t, t\rangle$. So for each $\mathfrak{M}$ and g, $[\![\exists \mathsf{v}]\!]^{\mathfrak{M},g}$ would have to be a unary truth function. That's easily seen to be impossible. Cf. Gamut (1991, 101–102).

[9]Indeed this seems to be the purpose of Heim and Kratzer's Predicate Abstraction rule.

parentheses are added. The $\mathcal{L}^{\lambda}_{\mathbf{T}}$-well-formed expressions and their types are defined as follows.

1. Every $c \in C_a$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type a.

2. Every $\mathsf{v} \in V_a$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type a.

3. The connectives $\neg$ and $\wedge$ are $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe's of types $\langle t, t\rangle$ and $\langle t, \langle t, t\rangle\rangle$, respectively.

4. For any $a \in \mathbf{T}$, $\exists^a$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type $\langle\langle a, t\rangle, t\rangle$.

5. If α is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type $\langle a, b\rangle$ and β an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type a, then $(\alpha\beta)$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type b.

6. If α is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type a and v a variable of type b, then $(\lambda\mathsf{v}\alpha)$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type $\langle b, a\rangle$.

The λ-enriched language $\mathcal{L}^{\lambda}_{\mathbf{T}}$ differs somewhat from the original language $\mathcal{L}_{\mathbf{T}}$. For example, $\mathcal{L}^{\lambda}_{\mathbf{T}}$, but not $\mathcal{L}_{\mathbf{T}}$, contains quantified formulas in which no bound variable is present (such as $\exists^e P$, where P is of type $\langle e, t\rangle$). Moreover, the quantified formulas of $\mathcal{L}_{\mathbf{T}}$ get a new look in $\mathcal{L}^{\lambda}_{\mathbf{T}}$: Instead of the $\mathcal{L}_{\mathbf{T}}$-formula $\exists\mathsf{v}\, P\mathsf{v}$, the language $\mathcal{L}^{\lambda}_{\mathbf{T}}$ has $(\exists^e(\lambda\mathsf{v}(P\mathsf{v})))$. Finally, $\mathcal{L}^{\lambda}_{\mathbf{T}}$, but not $\mathcal{L}_{\mathbf{T}}$, contains lambda-abstracts $(\lambda\mathsf{v}\alpha)$ as wfe's. It's intuitively clear, however, that $\mathcal{L}_{\mathbf{T}}$ is essentially a sublanguage of $\mathcal{L}^{\lambda}_{\mathbf{T}}$. More precisely, let ι be the function from the $\mathcal{L}_{\mathbf{T}}$-wfe's to the $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe's that is defined as follows.

1. For $a \in \mathbf{T}$ and $\alpha \in C_a$, $\iota(\alpha) = \alpha$.

2. For $a \in \mathbf{T}$ and $\mathsf{v} \in V_a$, $\iota(\mathsf{v}) = \mathsf{v}$.

3. $\iota(\neg) = \neg$ and $\iota(\wedge) = \wedge$.

4. $\iota(\alpha\beta) = (\iota(\alpha)\iota(\beta))$.

5. $\iota(\exists\mathsf{v}\phi) = (\exists^a(\lambda\mathsf{v}\iota(\phi)))$, where v is of type a.

The function ι is clearly 1-1 and preserves types. It thus embeds $\mathcal{L}_\mathbf{T}$ in $\mathcal{L}_\mathbf{T}^\lambda$.

We define the extensions $[\![\alpha]\!]_\lambda^{\mathfrak{M},g}$ of $\mathcal{L}_\mathbf{T}^\lambda$-wfe's α relative to $\mathfrak{M}$ and g as follows. Clauses 1–4 are lexical entries, clauses 5 and 6 composition rules.

1. For $a \in \mathbf{T}$ and $\alpha \in C_a$, $[\![\alpha]\!]_\lambda^{\mathfrak{M},g} = \mathcal{I}(\alpha)$.

2. For $a \in \mathbf{T}$ and $\mathsf{v} \in V_a$, $[\![\mathsf{v}]\!]_\lambda^{\mathfrak{M},g} = g(\mathsf{v})$.

3. $[\![\neg]\!]_\lambda^{\mathfrak{M},g} = \underline{\lambda} n \in \{0,1\} \,.\, 1 - n$ and $[\![\wedge]\!]_\lambda^{\mathfrak{M},g} = \underline{\lambda} n \in \{0,1\} \,.\, \underline{\lambda} m \in \{0,1\} \,.\, \min\{n, m\}$.

4. For $a \in \mathbf{T}$, $[\![\exists^a]\!]_\lambda^{\mathfrak{M},g} = \underline{\lambda} f \in D_t^{D_a} . \max\{f(u) \mid u \in D_a\}$.

5. (FA) For α of type $\langle a, b\rangle$ and β of type a, $[\![(\alpha\beta)]\!]_\lambda^{\mathfrak{M},g} = [\![\alpha]\!]_\lambda^{\mathfrak{M},g}([\![\beta]\!]_\lambda^{\mathfrak{M},g})$.

6. If α is an $\mathcal{L}_\mathbf{T}^\lambda$-wfe of type a and $\mathsf{v} \in V_b$, $[\![(\lambda\mathsf{v}\alpha)]\!]_\lambda^{\mathfrak{M},g} = \underline{\lambda} u \in D_b . [\![\alpha]\!]_\lambda^{\mathfrak{M},g[\mathsf{v}:=u]}$, where $g[\mathsf{v} := u]$ is the v-variant of g that maps v to u.

It's easy to see that ι is an *extensionally faithful* embedding of $\mathcal{L}_\mathbf{T}$ in $\mathcal{L}_\mathbf{T}^\lambda$ in the following sense. For every model $\mathfrak{M}$, variable assignment g over $\mathfrak{M}$, and $\mathcal{L}_\mathbf{T}$-wfe α:

$$[\![\alpha]\!]^{\mathfrak{M},g} = [\![\iota(\alpha)]\!]_\lambda^{\mathfrak{M},g}$$

The quantifiers have now been given a categorematic treatment. A quantifier over objects of type a receives type $\langle\langle a, t\rangle, t\rangle$; in particular, the quantifier $\exists^e$ is of type $\langle\langle e, t\rangle, t\rangle$, which accords well not only with Frege's view but also with contemporary generalized quantifier theory. Thus *prima facie* the existential quantifiers $\exists^a$ as they occur in $\mathcal{L}_\mathbf{T}^\lambda$ do not look like intensional operators, because their arguments aren't intensions.

This does not really settle our question, however, for one might argue that the role played by $\exists\mathsf{v}$ in $\mathcal{L}_\mathbf{T}$ is now played by the *combination* of the quantifier $\exists^a$ with the lambda operator $\lambda\mathsf{v}$, so in order to elucidate the logical status of $\exists\mathsf{v}$ in $\mathcal{L}_\mathbf{T}$ we should look not at $\exists^a$ in isolation, but at $\exists^a \circ \lambda\mathsf{v}$, as it were. If we combine the semantic clauses for $\exists^a$ and $\lambda\mathsf{v}$, however, we obtain a syncategorematic clause identical to the one that $\exists\mathsf{v}$ had in $\mathcal{L}_\mathbf{T}$.

Indeed it's not clear that we've made substantive progress at all, for the categorematic treatment of the quantifier has come at the price of a *syncategorematic* introduction of the abstraction operator, and of the three objections to the semantics for $\mathcal{L}_{\mathbf{T}}$, at least two therefore remain: the concern about compositionality *stricto sensu*, arising from the different semantic treatment of concatenation in the cases of negation and lambda abstraction, and the violation of Frege's Conjecture. So it would help if we could give λ a categorematic treatment consistent with Frege's Conjecture. Is that possible?

It depends. Within our current type-theoretical framework, we cannot in general produce an assignment intension for $\lambda \mathsf{v}$ that allows evaluation of λ-terms by way of functional application.[10]

As we will see in the next section, it *is* possible, modulo substantial revisions to the framework, to give λ a categorematic treatment in accord with Frege's Conjecture. Categorematicity, however, comes at a price, which we'll assess in due course.

3 Assignatory Type Theory

As mentioned, making the abstraction operator categorematic requires some changes to the type-theoretical framework.[11]

We begin by expanding the set of types. Let the set $\mathbf{T}^{\mathfrak{s}}$ of *assignatory types* ($\mathfrak{s}$-types for short) be defined as follows: e and t are $\mathfrak{s}$-types, and whenever a and b are $\mathfrak{s}$-types, so are $\langle a, b\rangle$ and $\langle \mathfrak{s}, a\rangle$. The *assignatory type hierarchy* ($\mathfrak{s}$-type hierarchy for short) over a

[10]Let $\mathfrak{M}_0$ be a model with domain the natural numbers $\mathbb{N}$ in which the interpretation of the constant P (of type $\langle e, t\rangle$) maps every number to 0 and in which the interpretation of the constant Q (of type $\langle e, t\rangle$) maps every positive number to 0 but maps 0 to 1. Let v be a variable of type e and let g assign 1 to v. Obviously $[\![(\lambda \mathsf{v}(P\mathsf{v}))]\!]^{\mathfrak{M}_0}_{\lambda}(g)$ is the constant function on $\mathbb{N}$ with value 0, and $[\![(\lambda \mathsf{v}(Q\mathsf{v}))]\!]^{\mathfrak{M}_0}_{\lambda}(g)$ is the function on $\mathbb{N}$ that maps 0 to 1 but every positive number to 0. Since $(\lambda \mathsf{v}(P\mathsf{v}))$ is of type $\langle e, t\rangle$ and $(P\mathsf{v})$ is of type t, $\lambda \mathsf{v}$ would have to be of type $\langle t, \langle e, t\rangle\rangle$. Then $[\![\lambda \mathsf{v}]\!]^{\mathfrak{M}_0}_{\lambda}(g)$ would have to be a function from D_t to $D_t^{D_e}$, so that $[\![\lambda \mathsf{v}]\!]^{\mathfrak{M}_0}_{\lambda}(g)([\![(P\mathsf{v})]\!]^{\mathfrak{M}_0}_{\lambda}(g))$ would have to equal $[\![\lambda \mathsf{v}]\!]^{\mathfrak{M}_0}_{\lambda}(g)([\![(Q\mathsf{v})]\!]^{\mathfrak{M}_0}_{\lambda}(g))$, given that $[\![(P\mathsf{v})]\!]^{\mathfrak{M}_0}_{\lambda}(g) = 0 = [\![(Q\mathsf{v})]\!]^{\mathfrak{M}_0}_{\lambda}(g)$. But then we can't in general have $[\![(\lambda \mathsf{v}\alpha)]\!]^{\mathfrak{M}_0}_{\lambda}(g) = [\![\lambda \mathsf{v}]\!]^{\mathfrak{M}_0}_{\lambda}(g)([\![\alpha]\!]^{\mathfrak{M}_0}_{\lambda}(g))$.

[11]I believe this was first pointed out by Varzi (1993).

non-empty domain D is the family $(D_a)_{a \in \mathbf{T}^{\mathfrak{s}}}$ such that

- $D_e = D$;
- $D_t = \{0, 1\}$;
- $D_{\langle a,b \rangle} = D_b^{D_a}$, provided that $a \neq \mathfrak{s}$;
- $D_{\langle \mathfrak{s},a \rangle} = D_a^{\aleph_D}$.

The primitive symbols of the assignatory language $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$ are those of $\mathcal{L}^{\lambda}_{\mathbf{T}}$, except that the symbol λ is replaced by infinitely many symbols λ^{a}_{b}, one for each pair $\langle a, b \rangle \in \mathbf{T} \times \mathbf{T}$. The $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe's and their types are defined as follows.

1. Every $c \in C_a$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type a.
2. Every $\mathsf{v} \in V_a$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type a.
3. The connectives $\neg$ and $\wedge$ are $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfes of types $\langle t, t \rangle$ and $\langle t, \langle t, t \rangle \rangle$, respectively.
4. For $a \in \mathbf{T}$, $\exists^{a}$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle \langle a, t \rangle, t \rangle$.
5. For $a, b \in \mathbf{T}$ and $\mathsf{v} \in V_b$, $\lambda^{a}_{b}\mathsf{v}$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle \langle \mathfrak{s}, a \rangle, \langle b, a \rangle \rangle$.
6. If α is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle a, b \rangle$ and β an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type a, then $(\alpha\beta)$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type b.
7. If α is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle \langle \mathfrak{s}, a \rangle, b \rangle$ and β an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type a, then $(\alpha\beta)$ is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type b.

We define a function κ from the $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe's to the $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe's as follows.

1. For $a \in \mathbf{T}$ and $\alpha \in C_a$, $\kappa(\alpha) = \alpha$.
2. For $a \in \mathbf{T}$ and $\mathsf{v} \in V_a$, $\kappa(\mathsf{v}) = \mathsf{v}$.
3. $\kappa(\neg) = \neg$ and $\kappa(\wedge) = \wedge$.
4. For $a \in \mathbf{T}$, $\kappa(\exists^{a}) = \exists^{a}$.

5. If α is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type $\langle a, b\rangle$ and β an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type a, $\kappa((\alpha\beta)) = (\kappa(\alpha)\kappa(\beta))$.

6. If α is an $\mathcal{L}^{\lambda}_{\mathbf{T}}$-wfe of type a and v a variable of type b, $\kappa((\lambda\mathsf{v}\alpha)) = (\lambda^{a}_{b}\mathsf{v}\kappa(\alpha))$.

It's easy to see that κ is 1-1 and that $\kappa(\alpha)$ is of the same type as α. Hence κ embeds $\mathcal{L}^{\lambda}_{\mathbf{T}}$ in $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$, and $\kappa \circ \iota$ embeds $\mathcal{L}_{\mathbf{T}}$ in $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$.

A model $\mathfrak{M}$ for $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$ is still a pair $\langle D, \mathcal{I}\rangle$, where $D \neq \emptyset$ and $\mathcal{I}$ is a function assigning to each member of C_a an element of D_a, for every $a \in \mathbf{T}$.[12]

Next we define the extensions $[\![\alpha]\!]^{\mathfrak{M},g}_{\lambda,\mathfrak{s}}$, relative to $\mathfrak{M}$ and g, of the well-formed $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-expressions α. As before, we let $[\![\alpha]\!]^{\mathfrak{M}}_{\lambda,\mathfrak{s}}$ be the function that maps each variable assignment g to $[\![\alpha]\!]^{\mathfrak{M},g}_{\lambda,\mathfrak{s}}$, that is, the *assignment intension* of α (relative to $\mathfrak{M}$). To reduce clutter, we often suppress the model superscript $\mathfrak{M}$; thus e.g. the assignment intension of α will be written $[\![\alpha]\!]_{\lambda,\mathfrak{s}}$. The clauses for extensions can then be given as follows. Clauses 1–5 are lexical entries, clauses 6 and 7 are composition rules.

1. For $a \in \mathbf{T}$ and $\alpha \in C_a$, $[\![\alpha]\!]^{g}_{\lambda,\mathfrak{s}} = \mathcal{I}(\alpha)$.

2. For $a \in \mathbf{T}$ and $\mathsf{v} \in V_a$, $[\![\mathsf{v}]\!]^{g}_{\lambda,\mathfrak{s}} = g(\mathsf{v})$.

3. $[\![\neg]\!]^{g}_{\lambda,\mathfrak{s}} = \underline{\lambda} n \in \{0,1\} \,.\, 1 - n$ and $[\![\wedge]\!]^{g}_{\lambda,\mathfrak{s}} = \underline{\lambda} n \in \{0,1\} \,.\, \underline{\lambda} m \in \{0,1\} \,.\, \min\{n, m\}$.

4. For $a \in \mathbf{T}$, $[\![\exists^{a}]\!]^{g}_{\lambda,\mathfrak{s}} = \underline{\lambda} f \in D_{\langle a,t\rangle} \,.\, \max\{f(u) \mid u \in D_a\}$.

5. For $a, b \in \mathbf{T}$ and $\mathsf{v} \in V_b$, $[\![\lambda^{a}_{b}\mathsf{v}]\!]^{g}_{\lambda,\mathfrak{s}} := \underline{\lambda} f \in D_a^{\aleph_D} \,.\, \underline{\lambda} r \in D_b \,.\, f(g[\mathsf{v} := r])$.[13]

6. Extensional Functional Application: If α is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle a, b\rangle$ and β an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type a, then $[\![(\alpha\beta)]\!]^{g}_{\lambda,\mathfrak{s}}$ is $[\![\alpha]\!]^{g}_{\lambda,\mathfrak{s}}([\![\beta]\!]^{g}_{\lambda,\mathfrak{s}})$.

7. Intensional Functional Application: If α is an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle\langle \mathfrak{s}, a\rangle, b\rangle$ and β an $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type a, then $[\![(\alpha\beta)]\!]^{g}_{\lambda,\mathfrak{s}}$ is $[\![\alpha]\!]^{g}_{\lambda,\mathfrak{s}}([\![\beta]\!]_{\lambda,\mathfrak{s}})$, i.e. $[\![\alpha]\!]^{g}_{\lambda,\mathfrak{s}}\left(\underline{\lambda} h \in \aleph_D \,.\, [\![\beta]\!]^{h}_{\lambda,\mathfrak{s}}\right)$.

[12]Since we have more types in $\mathbf{T}^{\mathfrak{s}}$ than in $\mathbf{T}$, we could allow non-empty sets of non-logical constants of these additional types. For the sake of simplicity, we don't.

[13]Such a clause for $\lambda\mathsf{v}$ was first proposed, I believe, by Rabern (2012, 399).

Note that these extensions live in the *assignatory* type hierarchy over the model's domain. In particular, the extension $[\![\lambda^a_b \mathsf{v}]\!]^g_{\lambda,\mathfrak{s}}$ of $\lambda^a_b\mathsf{v}$ at g belongs to $D_{\langle\langle \mathfrak{s},a\rangle,\langle b,a\rangle\rangle}$ and thus falls outside the extensional type hierarchy $(D_a)_{a\in\mathbf{T}}$ over D.

It can be shown that κ is an extensionally faithful embedding of $\mathcal{L}^\lambda_{\mathbf{T}}$ in $\mathcal{L}^\lambda_{\mathbf{T}^\mathfrak{s}}$ (so that $\kappa \circ \iota$ is an extensionally faithful embedding of $\mathcal{L}_{\mathbf{T}}$ in $\mathcal{L}^\lambda_{\mathbf{T}^\mathfrak{s}}$).[14] Moreover, the extensional faithfulness of ι and κ, and concomitantly of $\kappa \circ \iota$, immediately implies their *intensional faithfulness*, meaning that an expression and its translation always have the same assignment intension. Thus we've finally arrived at a language that faithfully embeds the original type-theoretical language $\mathcal{L}_{\mathbf{T}}$ and has a fully categorematic semantics.

Before we consider the status of quantifiers as intensional operators in $\mathcal{L}^\lambda_{\mathbf{T}^\mathfrak{s}}$, let us note that once we've swallowed the expansion of the type hierarchy to the assignatory type hierarchy and the presence of the intensional functional application rule, the λ-operator is no longer needed for a categorematic treatment of the quantifier: Let the primitive symbols of the *austere* assignatory language $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$ be those of $\mathcal{L}^\lambda_{\mathbf{T}^\mathfrak{s}}$ minus all symbols λ^a_b and the parentheses. The $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$-wfe's and their types are defined as follows.

1. Every $c \in C_a$ is an $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$-wfe of type a.

2. Every $\mathsf{v} \in V_a$ is an $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$-wfe of type a.

3. The connectives $\neg$ and $\wedge$ are $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$-wfe's of types $\langle t,t\rangle$ and $\langle t,\langle t,t\rangle\rangle$, respectivly.

4. For $a \in \mathbf{T}$ and $\mathsf{v} \in V_a$, $\exists^a\mathsf{v}$ is an $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$-wfe of type $\langle\langle \mathfrak{s},t\rangle,t\rangle$.

5. If α is an $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$-wfe of type $\langle a,b\rangle$ and β an $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$-wfe of type a, then $\alpha\beta$ is an $\mathcal{L}_{\mathbf{T}^\mathfrak{s}}$-wfe of type b.

[14]We must show that for every g and α, $[\![\alpha]\!]^g_\lambda = [\![\kappa(\alpha)]\!]^g_{\lambda,\mathfrak{s}}$. The interesting case is that of λ-abstraction. Suppose α is of type a and v of type b. Then $[\![(\lambda\mathsf{v}\alpha)]\!]^g_\lambda = \underline{\lambda} r\in D_b\,.\,[\![\alpha]\!]^{g[\mathsf{v}:=r]}_\lambda$, and $\kappa((\lambda\mathsf{v}\alpha)) = (\lambda^a_b\mathsf{v}\,\kappa(\alpha))$. Thus

$$[\![\kappa((\lambda\mathsf{v}\alpha))]\!]^g_{\lambda,\mathfrak{s}} = [\![(\lambda^a_b\mathsf{v}\,\kappa(\alpha))]\!]^g_{\lambda,\mathfrak{s}} = [\![\lambda^a_b\mathsf{v}]\!]^g_{\lambda,\mathfrak{s}}\,([\![\kappa(\alpha)]\!]_{\lambda,\mathfrak{s}}) = \underline{\lambda} r\in D_b.\,([\![\kappa(\alpha)]\!]_{\lambda,\mathfrak{s}}(g[\mathsf{v} := r])) = \underline{\lambda} r\in D_b.[\![\kappa(\alpha)]\!]^{g[\mathsf{v}:=r]}_{\lambda,\mathfrak{s}}$$

from which the claim follows since the identity of $[\![\kappa(\alpha)]\!]^{g[\mathsf{v}:=r]}_{\lambda,\mathfrak{s}}$ with $[\![\alpha]\!]^{g[\mathsf{v}:=r]}_\lambda$ is guaranteed by the inductive hypothesis.

6. If α is an $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle\langle\mathfrak{s}, t\rangle, t\rangle$ and β an $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type t, then $\alpha\beta$ is an $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type t.

It's obvious that $\mathcal{L}_{\mathbf{T}}$ can be embedded in $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$: Let ρ be the function from $\mathcal{L}_{\mathbf{T}}$-wfe's to $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe's that maps each constant, variable, and connective to itself, that maps $\mathcal{L}_{\mathbf{T}}$-wfe's $\alpha\beta$ to $\rho(\alpha)\rho(\beta)$, and that maps $\mathcal{L}_{\mathbf{T}}$-wfe's $\exists\mathsf{v}\phi$, with $\mathsf{v} \in V_a$, to $\exists^a\mathsf{v}\rho(\phi)$.

Still taking models to be pairs $\mathfrak{M} = \langle D, \mathcal{I}\rangle$, we can define extensions of $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe's relative to models and assignments in the by now familiar way. Clauses 1–4 are lexical entries, and clauses 5 and 6, composition rules.

1. $[\![c]\!]^g_{\mathfrak{s}} = \mathcal{I}(c)$ for $c \in C_a$.

2. $[\![\mathsf{v}]\!]^g_{\mathfrak{s}} = g(\mathsf{v})$ for $\mathsf{v} \in V_a$.

3. $[\![\neg]\!]^g_{\mathfrak{s}} = \underline{\lambda} n \in \{0,1\} \,.\, 1 - n$ and $[\![\wedge]\!]^g_{\mathfrak{s}} = \underline{\lambda} n \in \{0,1\} \,.\, \underline{\lambda} m \in \{0,1\} \,.\, \min\{n, m\}$.

4. For $a \in \mathbf{T}$ and $\mathsf{v} \in V_a$, $[\![\exists^a\mathsf{v}]\!]^g_{\mathfrak{s}} = \underline{\lambda} f \in D_{\langle\mathfrak{s},t\rangle} \,.\, \max\{f(g[\mathsf{v} := r]) \mid r \in D_a\})$.

5. (EFA) If α is an $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle a, b\rangle$ and β an $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type a, then $[\![\alpha\beta]\!]^g_{\mathfrak{s}} = [\![\alpha]\!]^g_{\mathfrak{s}}([\![\beta]\!]^g_{\mathfrak{s}})$.

6. (IFA) If α is an $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type $\langle\langle\mathfrak{s}, t\rangle, t\rangle$ and β an $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$-wfe of type t, then $[\![\alpha\beta]\!]^g_{\mathfrak{s}} = [\![\alpha]\!]^g_{\mathfrak{s}}([\![\beta]\!]_{\mathfrak{s}})$.

Readers will have no trouble establishing that ρ is an extensionally and intensionally faithful embedding of $\mathcal{L}_{\mathbf{T}}$ in $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$. Thus we now have a fully categorematic semantics for a λ-free language that embeds the original type-theoretical language $\mathcal{L}_{\mathbf{T}}$ in a faithful way.[15]

In this setting, $\exists^a\mathsf{v}$ is *literally* an intensional operator, since its extension at any assignment takes assignment intensions as arguments. Indeed it's a strongly intensional operator, for the value of $[\![\exists^a\mathsf{v}]\!]^g_{\mathfrak{s}}$ for some argument intensions does not depend only on

[15]Something like the approach just outlined is suggested by Levin (1988).

those intensions' values at g.[16]

In $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$, on the other hand, as in $\mathcal{L}^{\lambda}_{\mathbf{T}}$, $\exists^{a}$ is not an intensional operator, but rather an ordinary higher-order function, as predicted by Frege and generalized quantifier theory. In particular, for any $a \in \mathbf{T}$, $\exists^{a}$ is of type $\langle\langle a, t\rangle, t\rangle$, which is again in $\mathbf{T}$. Nevertheless, even within $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$ we have intensional operators, to wit, the $\lambda^{t}_{a}\mathsf{v}$, which are of type $\langle\langle \mathfrak{s}, t\rangle, \langle a, t\rangle\rangle$. Indeed, the lambda abstractors λ^{t}_{a} are strongly intensional operators, for there are models $\mathfrak{M}$ and assignments g such that the value of $[\![\lambda^{t}_{a}\mathsf{v}]\!]^{g}_{\lambda,\mathfrak{s}}$ for an intension as argument doesn't just depend on that intension's value at g.[17] But this means that, if we consider the compound operator $\exists^{a} \circ \lambda^{t}_{a}\mathsf{v}$, which is arguably the counterpart of $\exists^{a}\mathsf{v}$ in $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$, we find this "quantifier" to be an intensional operator, indeed a strongly intensional operator, *stricto sensu*.

There thus seems to be considerable support for the view that quantifiers are intensional operators. But before we endorse that view, we should ask whether the formulations of type theory that most strongly underwrite it, $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$ and $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$, are in fact theoretically satisfactory.

4 What Price Categorematicity?

Let's take a step back and assess the price we've paid for eliminating syncategorematicity from the semantics of type theory—and what we've gotten in return.

$\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$ and $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$ each employ two formally distinct functional application rules in their semantics, namely extensional functional application, according to which the extension of $(\alpha\beta)$ at g is the result of applying the *extension* of α at g to the *extension* of β at g, and

[16]Let D be the two-element set $\{x, y\}$, let the type a be e, let P and Q be of type $\langle e, t\rangle$, suppose that $\mathcal{I}(P)$ is the constant function on D with value 0, and that $\mathcal{I}(Q)$ maps x to 1 but y to 0. Let g be an assignment mapping every variable of type e to y. The value of $[\![\exists^{a}\mathsf{v}]\!]^{g}_{\mathfrak{s}}$ for an intension as argument cannot in general depend only upon that intension's value at g, for $[\![\exists^{a}\mathsf{v}]\!]^{g}_{\mathfrak{s}}([\![P\mathsf{v}]\!]_{\mathfrak{s}}) = 0 \neq 1 = [\![\exists^{a}\mathsf{v}]\!]^{g}_{\mathfrak{s}}([\![Q\mathsf{v}]\!]_{\mathfrak{s}})$ even though $[\![P\mathsf{v}]\!]^{g}_{\mathfrak{s}} = 0 = [\![Q\mathsf{v}]\!]^{g}_{\mathfrak{s}}$.

[17]Consider our previous example. Note that $[\![\lambda^{e}\mathsf{v}]\!]^{g}_{\lambda,\mathfrak{s}}([\![Q\mathsf{v}]\!]_{\lambda,\mathfrak{s}}) \neq [\![\lambda^{e}\mathsf{v}]\!]^{g}_{\lambda,\mathfrak{s}}([\![P\mathsf{v}]\!]_{\lambda,\mathfrak{s}})$ but $[\![P\mathsf{v}]\!]^{g}_{\lambda,\mathfrak{s}} = [\![Q\mathsf{v}]\!]^{g}_{\lambda,\mathfrak{s}}$.

intensional functional application, according to which the extension of $(\alpha\beta)$ at g is the result of applying the *extension* of α at g to the *assignment intension* of β. This might strike one as problematic, especially given our earlier worries regarding the compositionality of theories with syncategorematic rules. For we again have *two* semantic rules—(EFA) and (IFA)—corresponding to *one* syntactic rule: concatenation of α and β. Moreover, we now have a language in which an expression stands at times for its extension and at times for its intension[18]—a feature that hasn't always sat well with philosophers of logic.[19]

Technically speaking, this feature of $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathfrak{s}}}$ and $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$ is a design artifact. As mentioned in footnote 1, we've here followed the principle adopted by von Fintel and Heim (2011), namely to minimize intensionality and retain extensional functional application wherever possible. We could, however, intensionalize the system in the manner suggested by von Fintel and Heim (2011, 12) and thereby do away with the *extensional* functional application rule. To illustrate, we could change the extension of, e.g., negation at g to $[\![\neg]\!]^{g}_{\mathfrak{s}} = \underline{\lambda} f \in D_{\langle \mathfrak{s}, t\rangle} \,.\, (1 - f(g))$, which would put it into type $\langle\langle \mathfrak{s}, t\rangle, t\rangle$.[20] Importantly, negation would now also be governed by the intensional functional application rule, since

[18]The wfe $(P\mathsf{v})$, as it occurs in $(\neg(P\mathsf{v}))$, contributes to the extension of $(\neg(P\mathsf{v}))$ at g a truth value, but as a constituent of $(\exists^{e}\mathsf{v}(P\mathsf{v}))$, $(P\mathsf{v})$ contributes to the extension at g of $(\exists^{e}\mathsf{v}(P\mathsf{v}))$ an assignment intension.

[19]Thus Wittgenstein (1922): "Two different symbols can therefore have the sign (the written sign or the sound sign) in common—they then signify in different ways" (TLP 3.321). "Thus there easily arise the most fundamental confusions [...]" (TLP 3.324). "In order to avoid these errors, we must employ a symbolism which excludes them, by not applying the same sign in different symbols [...] A symbolism, that is to say, which obeys the rules of logical grammar—of logical syntax [...]" (TLP 3.325). Frege, on the other hand, did allow for a sign to signify in different ways, e.g. to stand for its reference in ordinary contexts but for its sense in indirect contexts (Frege 1892). His treatment of identity in *Begriffsschrift* (1879, §8) in fact requires such a feature, since expressions are stipulated to refer to themselves when flanking the identity sign but to their ordinary referents when occurring elsewhere. See Pardey and Wehmeier (forthcoming) for details.

[20]Negation would then be a weakly intensional operator: Its extension at g would take intensions as arguments, but only the argument intension's value at g would matter to the function value assigned by the operator's extension.

$[\![\neg\phi]\!]^g = [\![\neg]\!]^g([\![\phi]\!])$, and extensional functional application is no longer needed.

So we can fiddle with the design to remove the awkwardness of having two functional application rules. There is a drawback to the fiddling, however, for it eliminates extensional operators altogether.[21] The distinction between extensional (*de iure* substitutive, as it were) and weakly intensional (*de facto* substitutive) operators that we can draw in $\mathcal{L}_{\mathbf{T}^{\mathsf{s}}}$ and $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathsf{s}}}$ disappears in the revised semantics, which seems like a conceptual disadvantage. But it's not clear how much weight such considerations ultimately carry.

There is a more serious problem, which might be called *syntactic pollution* of the semantic ontology. We set out to describe a simple non-linguistic structure, the extensional type hierarchy over a base set D. As long as we're willing to employ some syncategorematic means of variable-binding, as in $\mathcal{L}_{\mathbf{T}}$ and $\mathcal{L}^{\lambda}_{\mathbf{T}}$, we can accomplish this description by assigning as extensions to our well-formed expressions objects naturally occurring in this type hierarchy: elements of D_e, D_t, and function spaces arising from these. But as soon as we insist on lexical entries for all variable-binders, it turns out that this ontology does not suffice: We must expand the type hierarchy over D by allowing "conditionalization" on the set $\aleph_D$ of assignments. That is, in order to achieve a fully categorematic semantics for a language describing the extensional type hierarchy, we must assign to some expressions extensions that live outside that type hierarchy; indeed, extensions that essentially involve *syntactic* material, *viz.* the variables. In the case of $\mathcal{L}_{\mathbf{T}^{\mathsf{s}}}$, the quantifiers live in $D_{\langle\langle \mathsf{s},t\rangle,t\rangle}$ and hence outside $(D_a)_{a\in\mathbf{T}}$; in the case of $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathsf{s}}}$, the abstractors λ^a_b live in $D_{\langle\langle \mathsf{s},a\rangle,\langle b,a\rangle\rangle}$, and hence likewise outside $(D_a)_{a\in\mathbf{T}}$. Our semantic ontology has been contaminated with syntax.[22]

[21]Negation, for example, which is extensional in $\mathcal{L}_{\mathbf{T}}$, $\mathcal{L}^{\lambda}_{\mathbf{T}}$, $\mathcal{L}_{\mathbf{T}^{\mathsf{s}}}$, and $\mathcal{L}^{\lambda}_{\mathbf{T}^{\mathsf{s}}}$, because it takes truth values as arguments, becomes (weakly) intensional when we tweak its semantics in the way just indicated.

[22]Objections to this kind of contamination are, I take it, part of the motivation for the program of variable-free semantics—as Jacobson (2003, 58) points out, one wants the "meaning of any linguistic expression [to be] simply some normal, healthy model-theoretic object—something constructed only out of the 'stuff' that any theory presumably needs: individuals, worlds, times, perhaps events, etc." Indeed, building extensions out of syntactic material seems to violate the basic idea of referential semantics, by

Rejection of syntactic contamination is not merely a philosophical matter of ontological purity; there are "practical" consequences to constructing extensions out of assignments. To see this, let $\mathcal{L}'_{\mathbf{T}^{\mathfrak{s}}}$ be exactly like $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$ except that a particular variable w of $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$ is replaced by a new variable w'. Fix a model $\mathfrak{M}$. Then $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$ and $\mathcal{L}'_{\mathbf{T}^{\mathfrak{s}}}$ have no compositional meanings (assignment intensions) in common, because these compositional meanings are functions on assignments, and $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$ and $\mathcal{L}'_{\mathbf{T}^{\mathfrak{s}}}$ have no assignments in common. Now suppose we want to analyze a fragment of English by way of indirect interpretation through a type-theoretical language. Then the compositional meanings assigned by this method to English phrases will differ depending on our choice between $\mathcal{L}_{\mathbf{T}^{\mathfrak{s}}}$ and $\mathcal{L}'_{\mathbf{T}^{\mathfrak{s}}}$. But surely this choice is completely arbitrary and shouldn't affect which model-theoretic object is assigned to a natural language expression as its meaning. Something has gone badly wrong: The meanings assigned are not, in fact, purely model-theoretic objects, but rather depend for their identity on the variables present in the underlying language.

It is sometimes suggested that syntactic pollution can be avoided by conceiving of a type a assignment $g_a : V_a \to D_a$ as the composition $\delta_a \circ \varepsilon_a$ of a bijection $\varepsilon : V_a \approx \mathbb{N}$ between the variables in V_a and the natural numbers $\mathbb{N}$ with a sequence $\delta_a : \mathbb{N} \to D_a$. The job originally carried out by the variables in V_a is then effectively taken over by their proxies modulo ε_a in $\mathbb{N}$, and the job of variable assignments is effectively taken over by the sequences $\delta_a : \mathbb{N} \to D_a$.[23] Accordingly we can, as it were, let the type $\mathfrak{s}$ consist not of assignments $g = (g_a)_{a \in \mathbf{T}}$ but rather of families $(\delta_a)_{a \in \mathbf{T}}$ of sequences $\delta_a : \mathbb{N} \to D_a$. Then the denotations that fall outside of the original extensional type hierarchy are constructed not out of syntactic material, but simply out of ordinary set-theoretical constructs, viz. natural numbers—or so the argument goes.

One way to see that this is no solution to the contamination problem is by observing that the proposal essentially requires taking the natural numbers to *be* the variables:

making concrete representations (here, variables) essential to denotations.

[23]Heim and Kratzer (1998, 111; 213) take this route, though to be sure they do not claim that they thereby avoid syntactic pollution.

The entities officially called variables that are bijectively correlated with the natural numbers are completely idle in the semantics, it's the correlated numbers that are doing all the work; hence the natural numbers are the *de facto* variables of such a system. But then the natural numbers *are* syntactic material, and the fact that meanings are constructed out of functions on the natural numbers does nothing to mitigate the syntactic pollution issue.

Another way to reach the same conclusion is to notice that the proposal rules out by mere *fiat* using other countably infinite sets, such as the rational numbers $\mathbb{Q}$ or the integers $\mathbb{Z}$, as bijective correlates (proxies) of variables, and constructing meanings out of functions from *these* proxy sets into the type hierarchy over D. If these other proxy sets are allowed, meanings depend on the choice of proxy set, just as in the original set-up they depend on the choice of variables. But there is no principled reason to disallow $\mathbb{Q}$ or $\mathbb{Z}$, or any other countably infinite set, since there is nothing canonical about the choice of the natural numbers as proxies for variables. That is to say, one can obviously use a type-theoretical language based on rational or integer rather than natural number proxies for variables in the service of indirectly interpreting a fragment of English, and such a choice will materially affect which model-theoretic objects are assigned to English quantifier phrases as their denotations. This, however, clearly should not happen.

5 Fregean Type Theory

None of the versions of type theory we've canvassed appears to be entirely satisfactory: On the one hand, we have versions that bite the syncategorematic bullet and place all extensions within the extensional type hierarchy, but at the sacrifice of leaving variable-binders without lexical entries, thus arguably violating both Compositionality and Frege's Conjecture. On the other hand, we have versions that provide lexical entries for all operators, including the variable-binders, and satisfy both Compositionality and Frege's Conjecture, but at the sacrifice of contaminating the type-theoretic hierarchy with syntactic material, thereby making extensions language-dependent.

In this section, I argue that there is an attractive alternative version of extensional type theory that is superior to the systems considered so far in that it avoids both syncategorematicity *and* syntactic pollution. This alternative is extensional type theory in the spirit of Frege's *Grundgesetze* (1893).

Fregean type theory works with the original types in $\mathbf{T}$ and places its extensions within the extensional type hierarchy $(D_a)_{a\in\mathbf{T}}$ over any given model's domain D. The primitive symbols of the Fregean language $L_{\mathbf{T},F}$ are those of $\mathcal{L}_{\mathbf{T}^s}$ plus, for each $a \in \mathbf{T}$, a marker ξ^a for gaps of type a. The wfe's of $L_{\mathbf{T},F}$ and their types are defined as follows.

1. Every $c \in C_a$ is an $L_{\mathbf{T},F}$-wfe of type a.

2. The connectives $\neg$ and $\wedge$ are $L_{\mathbf{T},F}$-wfe's of types $\langle t,t\rangle$ and $\langle t,\langle t,t\rangle\rangle$, respectively.

3. For each $a \in \mathbf{T}$, the existential quantifier $\exists^a$ of type a is an $L_{\mathbf{T},F}$-wfe of type $\langle\langle a,t\rangle,t\rangle$.

4. If α is an $L_{\mathbf{T},F}$-wfe of type $\langle a,b\rangle$ and β an $L_{\mathbf{T},F}$-wfe of type a, and no gap marker occurs in either α or β, then $\alpha\beta$ is an $L_{\mathbf{T},F}$-wfe of type b.

5. If α is an $L_{\mathbf{T},F}$-wfe of type t, $c \in C_a$ occurs in α, and no gap marker occurs in α, then $\alpha_c[\xi^a]$ is an $L_{\mathbf{T},F}$-wfe of type $\langle a,t\rangle$.[24]

6. If π is an $L_{\mathbf{T},F}$-wfe of type $\langle a,t\rangle$ in which ξ^a occurs, and $\mathsf{v} \in V_a$ doesn't occur in π, then $\exists^a\mathsf{v}\pi_{\xi^a}[\mathsf{v}]$ is an $L_{\mathbf{T},F}$-wfe of type t.

We next define the *Fregean extension* $[\![\alpha]\!]^{\mathfrak{M}}_F$ of an $L_{\mathbf{T},F}$-wfe α relative to a model $\mathfrak{M} = \langle D, \mathcal{I}\rangle$. Clauses 1–3 are lexical entries, clauses 4 and 5 are composition rules, and clause 6 is a *sui generis* rule corresponding to the syntactic operation of deletion.

1. For $c \in C_a$, $[\![c]\!]^{\mathfrak{M}}_F = \mathcal{I}(c)$.

[24]In general, $\alpha_x[\beta]$ is the result of replacing each occurrence of x in α with β. Thus $\alpha_c[\xi^a]$ is the result of replacing each occurrence of c in α with the gap marker ξ^a of type a. More poetically, it is the result of erasing every occurrence of c in α, leaving behind an expression with gaps fit to receive wfe's of type a.

2. $[\![\neg]\!]^{\mathfrak{M}}_F = \underline{\lambda} n \in \{0,1\}. 1 - n$ and $[\![\wedge]\!]^{\mathfrak{M}}_F = \underline{\lambda} n \in \{0,1\}. \underline{\lambda} m \in \{0,1\}. \min\{n,m\}$.

3. For $a \in \mathbf{T}$, $[\![\exists^a]\!]^{\mathfrak{M}}_F = \underline{\lambda} f \in D_{\langle a,t\rangle}. \max\{f(u) \mid u \in D_a\}$.

4. $[\![\alpha\beta]\!]^{\mathfrak{M}}_F = [\![\alpha]\!]^{\mathfrak{M}}_F \left([\![\beta]\!]^{\mathfrak{M}}_F\right)$.

5. $[\![\exists^a \mathsf{v} \pi_{\xi^a}[\mathsf{v}]]\!]^{\mathfrak{M}}_F = [\![\exists^a]\!]^{\mathfrak{M}}_F \left([\![\pi]\!]^{\mathfrak{M}}_F\right)$.

6. $[\![\alpha_c[\xi^a]]\!]^{\mathfrak{M}}_F = \underline{\lambda} u \in D_a. [\![\alpha]\!]^{\mathfrak{M}^u_c}_F$, where $\mathfrak{M}^u_c$ is the model that is just like $\mathfrak{M}$ except that it interprets the constant c as the object u.

Some comments are in order.

First, every *operator* of $L_{\mathbf{T},F}$ has its own lexical entry. This is so notwithstanding the fact that there is no lexical entry for the gap markers; these are obviously not operators in any syntactic sense.[25]

Second, the only semantic operation corresponding to syntactic modes of composition *stricto sensu* in $L_{\mathbf{T},F}$ is functional application; this is so notwithstanding the fact that (i) the syntactic operation of deletion does *not* correspond to functional application (deletion is, after all, not a mode of *composition*) and (ii) the left hand sides of clauses 4 and 5 look somewhat different (their right hand sides clearly show them both to be rules of functional application).[26] The semantics for $L_{\mathbf{T},F}$ is thus arguably in accord with Frege's Conjecture.

Third, all Fregean extensions in a model live in the extensional type hierarchy over the model's domain. Indeed, variable assignments play no role in the semantics at all. In particular, the quantifiers are obviously not intensional operators, since their extensions are ordinary higher-order functions located within the extensional types $\mathbf{T}$.

Fourth, if we define the *Fregean intension* $[\![\alpha]\!]_F$ of an $L_{\mathbf{T},F}$-wfe α as the function on the

[25]In fact, if I may be allowed to wax metaphorical, they are mere *absences* of syntactic material.

[26]The difference in appearance of the left hand sides is due merely to the fact that quantifications require additional punctuation (provided by the variable) so that the predicate argument be uniquely reconstructible from the quantified sentence.

class of all models that maps each model $\mathfrak{M}$ to the Fregean extension $[\![\alpha]\!]^{\mathfrak{M}}_F$ of α in $\mathfrak{M}$, Fregean intensions are compositional.[27]

If desired, lambda-abstraction can easily be integrated into Fregean type theory. Let the primitive symbols of the lambda-enriched Fregean language $L^{\lambda}_{\mathbf{T},F}$ be those of $L_{\mathbf{T},F}$ together with the symbol λ^b_a for each pair $\langle a, b\rangle \in \mathbf{T} \times \mathbf{T}$, as well as parentheses. The $L^{\lambda}_{\mathbf{T},F}$-wfe's are defined as follows:

1. Every $c \in C_a$ is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type a.
2. The connectives $\neg$ and $\wedge$ are $L^{\lambda}_{\mathbf{T},F}$-wfe's of types $\langle t, t\rangle$ and $\langle t, \langle t, t\rangle\rangle$, respectively.
3. For each $a \in \mathbf{T}$, the existential quantifier $\exists^a$ is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type $\langle\langle a, t\rangle, t\rangle$.
4. For each $\langle a, b\rangle \in \mathbf{T} \times \mathbf{T}$, the abstractor λ^b_a is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type $\langle\langle a, b\rangle, \langle a, b\rangle\rangle$.
5. If α is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type $\langle a, b\rangle$ and β an $L^{\lambda}_{\mathbf{T},F}$-wfe of type a, and no gap marker occurs in either α or β, then $(\alpha\beta)$ is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type b.
6. If α is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type b and $c \in C_a$ occurs in α, and no gap marker occurs in α, then $\alpha_c[\xi^a]$ is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type $\langle a, b\rangle$.
7. If π is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type $\langle a, b\rangle$ in which ξ^a occurs, and $\mathsf{v} \in V_a$ doesn't occur in π, then $\lambda^b_a \mathsf{v} \pi_{\xi^a}[\mathsf{v}]$ is an $L^{\lambda}_{\mathbf{T},F}$-wfe of type $\langle a, b\rangle$.

Needless to say, existential quantification can be handled via clause 5, $L_{\mathbf{T},F}$-wfe's $\exists^a \mathsf{v} \phi_{\xi^a}[\mathsf{v}]$ being replaced by $L^{\lambda}_{\mathbf{T},F}$-wfe's $\left(\exists^a \left(\lambda^t_a \mathsf{v} \phi_{\xi^a}[\mathsf{v}]\right)\right)$.

Fregean extensions $[\![\alpha]\!]^{\mathfrak{M}}_{F,\lambda}$ for $L^{\lambda}_{\mathbf{T},F}$-wfe's α relative to a model $\mathfrak{M}$ are defined as follows. Clauses 1–4 are lexical entries, clauses 5 and 6 are composition rules, and clause 7 is a *sui generis* rule corresponding to the syntactic operation of deletion.

1. For $c \in C_a$, $[\![c]\!]^{\mathfrak{M}}_{F,\lambda} = \mathcal{I}(c)$.
2. $[\![\neg]\!]^{\mathfrak{M}}_{F,\lambda} = \underline{\lambda} n \in \{0, 1\}. 1 - n$ and $[\![\wedge]\!]^{\mathfrak{M}}_{F,\lambda} = \underline{\lambda} n \in \{0, 1\}.\underline{\lambda} m \in \{0, 1\}. \min\{n, m\}$.

[27]This follows *mutatis mutandis* from results of (Wehmeier 2018).

3. For $a \in \mathbf{T}$, $[\![\exists^a]\!]^{\mathfrak{M}}_{F,\lambda} = \underline{\lambda} f \in D_{\langle a,t\rangle}.\max\{f(u) \mid u \in D_a\}$.

4. For $a, b \in \mathbf{T}$, $[\![\lambda^b_a]\!]^{\mathfrak{M}}_{F,\lambda} = \mathsf{Id}_{\langle a,b\rangle}$ (the identity function on $D_{\langle a,b\rangle}$).

5. $[\![(\alpha\beta)]\!]^{\mathfrak{M}}_{F,\lambda} = [\![\alpha]\!]^{\mathfrak{M}}_{F,\lambda}\left([\![\beta]\!]^{\mathfrak{M}}_{F,\lambda}\right)$.

6. $[\![\lambda^b_a \mathsf{v}\pi_{\xi^a}[\mathsf{v}]]\!]^{\mathfrak{M}}_{F,\lambda} = [\![\lambda^b_a]\!]^{\mathfrak{M}}_{F,\lambda}\left([\![\pi]\!]^{\mathfrak{M}}_{F,\lambda}\right)$.

7. $[\![\alpha_c[\xi^a]]\!]^{\mathfrak{M}}_{F,\lambda} = \underline{\lambda} u \in D_a.[\![\alpha]\!]^{\mathfrak{M}^u_c}_{F,\lambda}$.

Quantification now falls under the functional application clause 5; but lambda abstraction, too, is handled by functional application, as is evident from the right hand side of clause 6, notwithstanding the fact that the lambda-abstractor is interpreted trivially as the identity function of the relevant type.[28]

Now define the Fregean intension $[\![\alpha]\!]_{F,\lambda}$ of an $L^{\lambda}_{\mathbf{T},F}$-wfe α as the function on the class of all models that maps each model $\mathfrak{M}$ to the Fregean extension $[\![\alpha]\!]^{\mathfrak{M}}_{F,\lambda}$ of α in $\mathfrak{M}$. These Fregean intensions are compositional.[29]

In Fregean type theory, whether in the shape of $L_{\mathbf{T},F}$ or $L^{\lambda}_{\mathbf{T},F}$, we thus have

- a compositional semantics
- that abides by Frege's Conjecture
- in which all operators have lexical entries and
- whose ontology is not contaminated by syntactic material.

We do have a *sui generis* syntactic rule, deletion, that is distinct from concatenation, and corresponding to it a *sui generis* semantic rule that isn't functional application. But

[28]Frege (1893) employs a device similar to lambda-abstraction: the value-range operator. This operator is *not* interpreted trivially, as it is intended to inject $D_{\langle e,e\rangle}$ into D_e, thereby engendering inconsistency; see (Wehmeier 2004). The lambda-operator, despite being *interpreted* trivially, is not itself trivial, as it enables complex predicates to figure as functors or arguments in the syntactic concatenation rule 5.

[29]Cf. again (Wehmeier 2018).

given the drawbacks of the alternatives—syncategorematic semantic clauses and concomitant violations of compositionality on the one hand, syntactic contamination of the semantic ontology on the other—the Fregean theories seem preferable to any of the more familiar type theories we've examined.

Moreover, the Fregean has an intuitive story to tell about the new syntactic and semantic rules: How do we arrive at the complex predicate *respects Laszlo and despises Strasser*? We first construct the sentence *Rick respects Laszlo and Rick despises Strasser*, then delete the name *Rick* to obtain *ξ respects Laszlo and ξ despises Strasser*. Finally we apply λ and get the one-place complex predicate *λx. x respects Laszlo and x despises Strasser*. How do we compute that predicate's extension? It's identical to that of *ξ respects Laszlo and ξ despises Strasser*, which we find by taking any sentence from which it could've been obtained by deletion, say *Rick respects Laszlo and Rick despises Strasser*, and going through all the logically possible extensions of the deleted name. Any object that, taken as the extension of the name, makes the sentence true, goes into the predicate's extension; any object that, taken as the name's extension, makes the sentence false, stays out of the predicate's extension. Thus Richard Blaine is in, for when the name *Rick* refers to Blaine (as it in fact does), *Rick respects Laszlo and Rick despises Strasser* is true; on the other hand, Major Strasser's aide Heinz is out, since *Rick respects Laszlo and Rick despises Strasser* is false when *Rick* refers to Heinz.

Crucially for our purposes, in neither Fregean theory do we have *any* intensional operators. This is trivially true on one reading of the claim: A Fregean model only contains entities living in one of the extensional types, where functions defined on variable assignments *qua* indices simply don't occur; accordingly, no operator extension can be a function defined on such indices.

The claim is still trivially true if we relax the definition of intensional operator to allow syncategorematic semantic clauses structurally identical to those for the modal operators; this is because the Fregean theories don't employ any syncategorematic operator clauses.

But the claim is *still* true on a broader interpretation of "intension" as encompassing functions whose domain is the class of all models (taken as surrogates for indices): The Fregean extension, in any given model, of any operator in $L_{\mathbf{T},F}$ or $L^{\lambda}_{\mathbf{T},F}$, takes as arguments entities living in the extensional type hierarchy over the model's domain; in other words, no operator extension ever takes as arguments functions whose domain is the class of all models. There is thus *no* sense in which $L_{\mathbf{T},F}$ or $L^{\lambda}_{\mathbf{T},F}$ contain intensional operators.[30]

6 Closing

If Fregean versions of type theory are indeed superior to the other formulations we have surveyed, we should presumably look to $L_{\mathbf{T},F}$ or $L^{\lambda}_{\mathbf{T},F}$ when trying to understand the semantic status of quantifiers. And as we observed at the end of the previous section, there is no sense in which the quantifiers appear as intensional operators in $L_{\mathbf{T},F}$ or $L^{\lambda}_{\mathbf{T},F}$; not, in any case, if we define intensional operators the way we did at the beginning of this paper.

That definition derives from the familiar possible-world treatment of modal and temporal operators. While it has long been a commonplace that these operators *can be construed as* quantifiers over possible worlds or times, linguistic semantics seems to take more and more seriously the notion that they literally *are* quantifiers.[31] But if modals and tenses are actually quantifiers, and quantifiers aren't intensional operators, as I've argued here, then intensional operators are uninteresting, merely theoretical constructs to which no feature of natural language answers.

[30]Moreover, neither the quantifiers nor the lambda abstractors satisfy even the syntactic criterion for intensional operators, to wit, that they operate on sentences: The syntactic arguments of $\exists^a$ in $L_{\mathbf{T},F}$ are *never* sentential, but rather explicitly predicative, in that they must contain a gap marker and have references that live in predicative types of the form $\langle a, t\rangle$, and likewise for the syntactic arguments of λ^t_a in $L^{\lambda}_{\mathbf{T},F}$.

[31]See e.g. King (2003), von Fintel and Heim (2011, §8.2), and Schaffer (2012).

References

[1] Belnap, N. (2005): "Under Carnap's Lamp: Flat Pre-Semantics," *Studia Logica* 80, 1–28.

[2] von Fintel, K., and I. Heim (2011): *Intensional Semantics*, unpublished lecture notes, spring 2011 edition, MIT.

[3] Frege, G. (1879): *Begriffsschrift*, Halle: Louis Nebert. Translated as *Concept Script* by S. Bauer–Mengelberg in J. vanHeijenoort (ed.), *From Frege to Gödel: A Source Book in Mathematical Logic 1879–1931*, Cambridge, MA: Harvard University Press, 1967.

[4] Frege, G. (1892): "Über Sinn und Bedeutung," *Zeitschrift für Philosophie und philosophische Kritik* NF 100, 25–50. Translated as "On Sense and Reference" by M. Black in *Translations from the Philosophical Writings of Gottlob Frege*, P. Geach and M. Black (ed. and transl.), Oxford: Blackwell, 3rd edition, 1980.

[5] Frege, G. (1893): *Grundgesetze der Arithmetik*, Volume I, Jena: Hermann Pohle. English translation in P. Ebert and M. Rossberg (editors and translators), *Gottlob Frege: Basic Laws of Arithmetic*, Oxford: Oxford University Press, 2013.

[6] Gamut, L.T.F. (1991): *Logic, Language, and Meaning*, vol. 2, Chicago: University of Chicago Press.

[7] Heim, I., and A. Kratzer (1998): *Semantics in Generative Grammar*, Oxford: Blackwell.

[8] Jacobson, P. (2003): "Binding without pronouns (and pronouns without binding)," in R. Oehrle and G.-J. Kruiff (eds), *Resource-Sensitivity, Binding, and Anaphora*, Dordrecht: Kluwer, 57–96.

[9] King, J. (2003): 'Tense, Modality, and Semantic Values,' *Philosophical Perspectives* 17 (Language and Philosophical Linguistics), 195–245.

[10] Levin, H. (1988): "A Philosophical Introduction to Categorial and Extended Categorial Grammar," in W. Buszkowski, W. Marciszewski, and J. van Benthem (eds), *Categorial Grammar*, Amsterdam/Philadelphia: John Benjamins, 173–195.

[11] Pagin, P., and D. Westerstahl (2010): "Compositionality I: Definitions and Variants," *Philosophy Compass* 5 (3), 250–264.

[12] Pardey, U., and K. Wehmeier (forthcoming): "Frege's *Begriffsschrift* Theory of Identity Vindicated," *Oxford Studies in Philosophy of Language*.

[13] Rabern, B. (2012): "Monsters in Kaplan's Logic of Demonstratives," *Philosophical Studies* 164, 393–404.

[14] Schaffer, J. (2012): 'Necessitarian Propositions,' *Synthese* 189(1), 119–162.

[15] Varzi, A. (1993): "Do We Need Functional Abstraction?", in J. Czermak (ed.), *Philosophy of Mathematics—Proceedings of the 15th International Wittgenstein Symposium*, Part 1, Vienna: Hölder-Pichler-Tempsky, 407–415.

[16] Wehmeier, K. (2004): "Russell's Paradox in Consistent Fragments of Frege's *Grundgesetze*," in G. Link (ed.), *One Hundred Years of Russell's Paradox: Proceedings of the 2001 Munich Conference*, Berlin: de Gruyter, 247–257.

[17] Wehmeier, K. (2018): "The Proper Treatment of Variables in Predicate Logic," *Linguistics and Philosophy* 41 (2), 209–249.

[18] Wittgenstein, L. (1922): *Tractatus Logico-Philosophicus*, London: Kegan Paul.