Notre Dame Journal of Formal Logic
Volume 51, Number 2, 2010

Syntax in *Basic Laws* §§29–32

Bryan Pickel

Abstract In order to accommodate his view that quantifiers are predicates of predicates within a type theory, Frege introduces a rule which allows a function name to be formed by removing a saturated name from another saturated name which contains it. This rule requires that each name has a rather rich syntactic structure, since one must be able to recognize the occurrences of a name in a larger name. However, I argue that Frege is unable to account for this syntactic structure. I argue that this problem undermines the inductive portion of Frege's proof that all of the names of his system denote in §§29–32 of *The Basic Laws*.

1 Introduction

In §28 of *The Basic Laws of Arithmetic*, Frege announces as a "leading principle" that all of the names[1] of his formal language, the *Begriffsschrift* (BG), are to have a semantic property; they are to *denote something*.[2] In §§29–32, Frege argues that every name in BG has this property.[3] At first glance, the argument seems to anticipate later developments in formal logic. On closer inspection, however, Frege's argument looks very different from anything one finds in a contemporary logic book.

One frequently mentioned difference lies in the semantic property—denoting something—which Frege is trying to prove every name possesses.[4] Frege does not precisely define this property. Rather, he gives criteria for detecting when a name denotes something. These criteria test for whether a name, X, denotes something in terms of whether more complex names containing X do. By way of contrast, in contemporary semantics one often shows that an expression has a semantic property by showing that the simpler expressions from which it is derived do.

I will discuss another difference between Frege's argument and contemporary arguments in metalogic. In contemporary systems, when one wants to prove that every expression of a language, $\mathcal{L}$, has a semantic property, one typically argues by induction on the syntactic complexity of expressions. Arguments of this sort require

Received March 6, 2008; accepted September 12, 2009; printed May 20, 2010
2010 Mathematics Subject Classification: Primary, 03A05, 00A30
Keywords: philosophy of logic, Frege's proof of referentiality
 10.1215/00294527-2010-016

a syntactic theory which inductively specifies the set of well-formed expressions of $\mathcal{L}$. In modern logics, one develops a theory T such that for every expression X of language $\mathcal{L}$, T yields as a theorem that X is an expression. To inductively specify the set of well-formed expressions, the theory must specify a list of primitive expressions and a list of formation rules for getting complex expressions out of basic expressions. Finally, the theory must contain a "that's all" clause circumscribing the expressions of the language to include only the basic expressions and those derivable by applying the formation rules to the basic expressions.

Frege's syntactic theory contains an enumeration of the primitives, a list of inductive rules for generating new names out of old names and a "that's all" clause. In §29, Frege argues that applying the formation rules to denoting name of BG always yields denoting names. The argument in §§29–32 therefore appears to be an early argument by induction on syntactic complexity. In contrast to modern syntactic theories for a language, however, the syntactic theory explicitly offered in *Basic Laws* does not entail of each expression X of BG that it is an expression. The reason is that the formation rules are framed in terms of other syntactic facts about a name, facts about its constituents and facts about what the result of replacing its constituents are. I will say that these facts determine or constitute a name's *syntactic structure*. The syntactic structure of the names of BG is not characterized by the syntactic theory which Frege explicitly offers.

I will argue that this difference from the modern symbolism causes two problems for Frege's argument. The first problem is that unless he supplements the syntactic theory on offer in *Basic Laws*, Frege does not provide sufficient instruction for applying the formation rules. I will argue there is no straightforward way to produce instructions for applying the rules which are consistent with Frege's principles. The reason is that Frege's specification of his syntax requires that the names of BG have a syntactic structure which is richer than he is in a position to give them. This argument will be bolstered by a discussion of Frege's proof theory, which makes essentially the same demands on the structure of the names of BG as his formation rules.

The other problem I will raise follows from the fact that Frege's argument in §§29–31 makes use of the assumption that there is a tight correspondence between the syntactic structure of a name and its syntactic derivation. In particular, Frege's argument requires that if a name occurs in another, then it also occurs in the latter's syntactic derivation. The problem I will raise is that for a significant portion of BG, this correspondence cannot hold. Moreover, there is no reason to believe that it holds for the rest of the language. Therefore, unlike those who have purported to find a problem with the base case of the argument, where Frege argues that his primitive names denote,[5] I will argue that there are significant problems with the inductive portion of the argument as well.

2 Frege's Recursive Syntax

Frege lays out the primitive names of BG in *Basic Laws* §§5–25. These names include the following:

(1) the horizontal '$-\xi$',
(2) the negation sign '$_\top \xi$' ['$\neg\xi$'],
(3) the description operator '$\backslash\xi$',

(4) the conditional '$\begin{array}{l} \top\, \xi \\ \llcorner\, \zeta \end{array}$' ['$\zeta \Rightarrow \xi$'],

(5) the identity sign '$\xi = \zeta$',
(6) the first-order universal quantifier '$\smile\!\!\!\!^{\mathfrak{a}}\ \Phi(\mathfrak{a})$' ['$\forall x \Phi x$'],
(7) the smooth breathing which makes an object name from a monadic function name '$\grave{\epsilon}\Phi(\epsilon)$',
(8) the second-order quantifiers ranging over one place functions '$\smile\!\!\!\!^{\mathfrak{F}}\ \Omega_{\beta}(\mathfrak{F}(\beta))$' ['$\forall F(\Omega_{\beta}(F(\beta)))$'],
(9) the second-order quantifiers ranging over two place functions '$\smile\!\!\!\!^{\mathfrak{F}}\ \Omega_{\beta,\gamma}(\mathfrak{F}(\beta,\gamma))$' ['$\forall F(\Omega_{\beta,\gamma}(F(\beta,\gamma)))$'].

All of the primitive names of BG are function names.[6] The expressions ξ and ζ mark argument places in an expression apt to be filled by saturated names. The expression $\Phi(\xi)$ which occurs in names such as $\forall x\Phi(x)$ marks argument places which are apt to be filled by monadic first-level functions.

In §30, Frege introduces two rules for forming names out of other names. One rule is that the result of saturating a function name with the name of an argument of the right kind is itself a name of BG. As he puts it, one gets, "a [saturated] proper name from a [saturated] proper name and a name of a first-level function of one argument" or "from a name of a first-level function and a name of a second-level function of one argument" and so on.[7] One gets a new name of BG by replacing a mark of incompleteness in a function name of BG with an argument of the appropriate type. Thus, one can get a name by saturating a function name which takes first-level functions of one argument such as the quantifier $\forall x\Phi(x)$ with a first-level function of one argument such as the horizontal $-\xi$, yielding $\forall x - x$.

Iteratively applying this rule to the nine primitives fails to generate all of the names which Frege would want his system to include. For instance, consider a multiply quantified sentence such as $\forall x\forall y(x = y)$. According to the Fregean picture, $\forall x\forall y(x = y)$ is the result of the monadic first-level function name $\forall y(\xi = y)$ saturating the quantifier $\forall x\Phi(x)$. The quantifier takes monadic first-level function names; so, this is a legitimate application of the first rule. The function name $\forall y(\xi = y)$ is not a basic name of BG. Therefore, it is derived from other names. Yet one cannot arrive at $\forall y(\xi = y)$ by successive applications of the first rule to the primitive names of BG. It looks as though this function name is composed by the quantifier $\forall y\Phi(y)$ taking the identity sign $\xi = \zeta$ in its argument position. But this cannot be right, since Frege's second-level function name $\forall y\Phi(y)$ takes only monadic first-level function names in its argument position. I shall call this the typing problem for Fregean analyses of quantification.

In systems developed since Tarski, the sentence $\forall x\forall y(x = y)$ is constructed using variables. One begins with the identity sign, =, and concatenates it with two variables, x and y, to form an *open sentence*, $x = y$. One then attaches the interior quantifier $\forall y$ which takes a sentence and yields a new open sentence, $\forall y(x = y)$. One finally applies the quantifier $\forall x$ to yield the closed sentence $\forall x\forall y(x = y)$. Thus, the Tarskian solution to the typing problem is that the quantifiers are not second-level function names but rather operators on open sentences which may contain any number of variables. The challenge posed by the typing problem for Frege may be stated more pointedly as follows. In order to construct multiply quantified names, Frege must be able to construct a monadic first-level function name everywhere that the Tarskian can construct an open sentence.

Frege's second formation rule is meant to be a principled method for generating the function names which Frege needs. This formation rule differs greatly from those used in more contemporary systems. Frege characterizes this second rule as follows.

> We begin by forming a name in the first way, and we then exclude from it at all or some places, a proper name that is a part of it (or coincides with it entirely)—but in such a way that these places remain recognizable as argument-places of type 1 [which take saturated names]. The function-name resulting from this likewise always has a denotation if the simple names from which it was formed denote something; and it may be used further to form names in the first way or in the second.[8]

This rule works as follows. If Φ is a saturated name of BG "containing" another saturated name, Δ, then the result of removing the occurrence of Δ from Φ so that the locations where Δ occurred remain recognizable as being apt to take a saturated name is also a function name of BG.[9] In Frege's system this is done by inserting a mark of incompleteness, ξ, at the locations where Δ was removed.[10] The second rule seems to allow Frege to construct first-level function names of one argument which can then be taken as arguments of the second-level quantifier. Consider the following potential derivation of $\forall x \forall y(x = y)$.

1. $\xi = \zeta$ is a name of BG containing a mark of incompleteness, ξ.
2. Δ is a saturated name of BG.
3. By the first rule, if X is a name of BG containing a mark of incompleteness, ξ, and Y is a saturated name of BG, then the result of replacing every occurrence of ξ with Y in X is also a name of BG.
4. $\Delta = \zeta$ is the result of replacing ξ in $\xi = \zeta$ with Δ. Therefore, by (1), (2), and (3), it is a name of BG. It is a first-level function name of BG, containing one argument place, ζ.
5. $\forall y \Phi(y)$ is a name of BG containing a mark of incompleteness, $\Phi(\xi)$, which is typed for first-level function names of one argument.
6. By the first rule, if X is a name of BG containing a mark of incompleteness, $\Phi(\xi)$, typed for first-level function names of one argument, and Y is a first-level function name of one argument BG, then the result of replacing every occurrence of $\Phi(\xi)$ with Y in X is also a name of BG.
7. $\forall y(\Delta = y)$ is the result of replacing Φ with $\Delta = \zeta$ in $\forall y \Phi(y)$. Therefore, by (5) and (6), it is a saturated name of BG.
8. Δ occurs in $\forall y(\Delta = y)$.
9. By the second rule, if X is a saturated name of BG and Y is a saturated name of BG occurring in X, then the result of replacing every occurrence Y in X with a mark of incompleteness, ξ, is also a name of BG.
10. $\forall y(\xi = y)$ is the result of replacing Δ in $\forall y(\Delta = y)$ with a mark of incompleteness, ξ. It is therefore a monadic first-level function name of BG by (7), (8), and (9).
11. $\forall x \Phi(x)$ is a name of BG containing a mark of incompleteness, $\Phi(\xi)$ which is typed for first-level function names of one argument.
12. $\forall x \forall y(x = y)$ is the result of replacing the mark of incompleteness, $\Phi(\xi)$, with the function name $\forall y(\xi = y)$ in $\forall x \Phi(x)$. Therefore, by (6), (10), and (11), it is a function name of BG.

The function name $\forall y(\xi = y)$ does not occur in the derivation as the result of one name saturating another according to the first rule. Rather, it is introduced in step (10) by removing the saturated name Δ from the name $\forall y(\Delta = y)$ and replacing it with a mark of incompleteness.

Frege concludes §30 by remarking, "All correctly formed names are formed in this manner."[11] I assume that Frege means that all of the names of BG are either primitive or are derived from the primitives by iterated applications of the first and second formation rules. This is exactly what one would expect if Frege were trying to inductively specify the set of well-formed names of BG. If his specification is successful, then he is in a position to offer recursive definitions for the names of BG—as one does for the expressions of a standard logic—and to run arguments by induction on syntactic complexity.

At this point, I will introduce a convenient notational device. I will need to speak of the result of replacing one expression X with another Y in a larger expression, Z. I shall denote the result of this operation using the standard convention $Z[Y/X]$. This is not to say that Frege really can define such an operation. I shall in fact argue that he cannot.

3 More on the Second Rule

As Frege presents his second rule in §30, it allows one to remove a saturated name from a name of BG and replace it with a mark of incompleteness. One thereby generates more functions which can serve as arguments of the first-level quantifier. However, with the rule as stated, Frege is still not able to generate all of the names he wants for BG. The reason is that Frege also needs to construct names containing the second-level quantifiers of BG. For instance, Frege needs to define the expression $\cap$ which corresponds to the "element-of" relation in standard set theory. His definition is framed in terms of a universal quantification over functions.[12] The problem is that the second rule, as stated, can produce only first-level function names; it cannot produce second-level function names of one or two arguments to serve as arguments for Frege's two second-level quantifiers. It is natural to suppose that Frege intends for his second rule for forming a name to allow one to replace not only a saturated name, but also the name of a first-level function of one or two arguments with a mark of incompleteness of the right type. Frege's need for such a clause has been pointed out by a number of commentators with whom I am in agreement.[13] I will therefore assume that the second rule allows one to form a name by replacing both saturated names and monadic and dyadic first-level function names with appropriately typed marks of incompleteness.

4 Why Frege Needs More Syntactic Theory

The axioms of Frege's syntactic theory given so far do not entail of every name of BG that it is a well-formed name. Consider steps (8) and (10) in the derivation provided above. Step (8) is the claim that Δ occurs in $\forall y(\Delta = y)$ and step (10) is the claim that $\forall y(\xi = y)$ is the result of replacing Δ in $\forall y(\Delta = y)$ with a mark of incompleteness, ξ. Notice that neither of these assumptions is generated by the syntactic theory given so far. The problem is that the syntactic theory is framed in terms of the notions constituency and replacement which are not yet defined and do not yet have substantive enough characterizations in the theory to entail claims (8) and (10). As a consequence, Frege's syntactic theory and his ability to make use of it in his argument are held hostage to whether these notions can be sufficiently characterized.

Another way to put this point is that the rules for producing new names out of old names are not themselves well specified, unless there are rules determining whether a name X is a constituent of another name Y and whether a name W is the result of replacing expression X in name Y with expression Z. That is, the success of the specification of the rules depends on whether Frege can specify the syntactic structures of the names of BG. To do so, Frege would need to be able to specify rules of the following two sorts.

A. There must be rules determining whether a name X occurs in a name Y.
B. There must be rules determining the result of replacing X in Y with Z.

I will argue that as a result of the second rule itself there is no straightforward way of extending what Frege has said to produce principles determining whether one name is a constituent of another or whether one name is the result of "replacing" occurrences of a name of BG with another expression in a name of BG. I will therefore argue that Frege has provided insufficient instructions for applying the second rule to names of BG.

4.1 Syntax in contemporary systems One might ask whether the requirement I am suggesting is too stringent. Is Frege required to characterize the basic notions of his syntactic theory so that for every well-formed name it yields a theorem of the form X is a well-formed name? He has, after all, provided inductive rules for forming new names out of old ones. Do contemporary syntactic theories really do more? The answers are that the requirement is not too stringent and that contemporary syntactic theories for formal languages do in fact yield as a theorem that X is a well-formed expression for every well-formed expression X.

Very few of these systems specify the formation rules in terms of the relation of constituency or a rule of replacement. Instead, the syntactic formation rules tend to be framed in terms of only one syntactic relation, concatenation. Each expression is the result of concatenating one dominant functional expression with a number of arguments of the right type. For example, the rule for negation says that if X is a well-formed sentence, then $\neg X$ is also a well-formed sentence. Whether a given expression X is a sentence is to be evaluated in terms of its syntactic structure. In order for a contemporary syntactic theory for a language $\mathcal{L}$ to entail, for every expression X of $\mathcal{L}$, the claim that X is a well-formed expression of $\mathcal{L}$, the theory must contain a characterization of the syntactic structure of the expressions of $\mathcal{L}$. Thus, contemporary theorists must frame their syntactic derivation rules in terms of the structure of the expressions to which they apply just as Frege does.

In contrast to Frege, the characterization of the structure of an expression of a contemporary logical language is often defined by its syntactic derivation. In most contemporary systems, the derivational history of an expression defines a tree structure which uniquely identifies it. Each node results from an application of a formation rule to a dominant n-ary connective and n arguments. The terminal nodes of the tree are the primitive expressions. The nonterminal nodes are the complex constituents of the larger expression. The tree structure reflects all of the syntactic features which are relevant to semantic assessment and proof theory. That is, the syntactic properties of the expressions of a language which are available to the semantics or proof theory can be recursively specified in terms of their syntactic derivational properties.

These recursive specifications can be offered in two steps. First, a contemporary theorist enumerates the syntactic types of the primitive expressions. Then, the theorist recursively defines the syntactic structure of an expression in terms of the properties of the expressions from which it is uniquely derived. As a result, one can assess whether an expression, X, has a syntactic property in a contemporary language by appeal to X's derivation.

One way to bring this out is to consider how contemporary logicians explain the notions of constituency and replacement. Contemporary logicians often make use of these notions in order to specify the proof rules for the system. In particular, the rules for existential generalization and universal instantiation are specified in terms of a replacement operation. In most systems the rule of existential introduction licenses the inference from a sentence A which may or may not contain a constant a to the sentence $\exists x A[x/a]$. Similarly, the rule of universal instantiation licenses the inference from $\forall x A$ to $A[a/x]$ where a is a constant. The operation of replacement is defined so that one replaces expression A by expression B in expression C by replacing A with B at every point in C's derivation. A constituency relation can be similarly defined.[14]

4.2 Syntactic structure in BG As a result of the fact that the syntactic derivations and the syntactic structures of expressions coincide in modern systems, it is natural to think of an expression as being "built up out of" the names which occur in its derivational history. In part, this is because conditions (C) and (D) hold trivially for expressions of contemporary logic.

C. An expression X occurs in an expression Y only if X occurs in the derivation of Y.
D. The expression which results from replacing X with Y in the derivation of the expression Z is the expression $Z[Y/X]$.

As we shall see, conditions (C) and (D) definitely do not hold for many of the expressions of BG even though they are required by Frege's argument that all of the names of BG denote.[15] On the contemporary view, the facts concerning the syntactic structure of a name and those concerning its derivation are defined at the same time. The syntactic derivation is thought of as a construction or building up of the names from the primitives.

Frege cannot think of the syntactic derivation of a name as building it in this way, if only because the second rule allows Frege to remove names in the course of a derivation and replace them with marks of incompleteness. Dummett somewhat misstates this point.

> It is of great importance that the predicate itself is not thought of as having been built up out of its component parts: we do not need to invoke the conception of the conjunction of two predicates 'ξ is charming' and 'ξ is sincere', to explain the formation of the predicate 'ξ is charming and sincere'; nor do we need to invoke the idea of the application of a quantifier to a two-place predicate, with respect to a specific argument-place, to explain the formation of the predicate '$\exists y\ \xi$ killed y'.[16]

This needs to be more explicit. It's not merely that Frege need not syntactically derive these names in the way described. The fact is that he *cannot* do so without violating his typing principles. To change Dummett's example, if Frege thought that $\forall y(\xi = y)$ was derived by applying a quantifier to a dyadic predicate, then the quantifier would be able to take predicates of multiple types as arguments.

Nonetheless, Dummett seems to acknowledge that the quantifier and the dyadic relation name are in some sense components or constituents of $\forall y(\xi = y)$. One reason Frege needs these names to be constituents of the larger name is that these names are apt candidates for removal by the second rule. For instance, one should be able to remove the name $\xi = \zeta$ from $\forall x \forall y(x = y)$ and replace it with mark of incompleteness $\Phi(\xi, \zeta)$, yielding $\forall x \forall y \Phi(x, y)$. One must proceed in this way to syntactically derive a name such as $\forall F \forall x \forall y F(x, y)$.

Another reason stems from Frege's proof theory which, like contemporary proof theories, is specified in terms of the syntactic constituents of a name. Rather than introducing a rule of universal instantiation as one might find in a more contemporary sequent calculus, Frege has an axiom of universal instantiation. As a result, Frege's appeal to the syntactic structure of various names in specifying his proof rules is slightly disguised. Frege's proof system makes use of expressions called Roman letters. These are expressions which stand in for genuine names in Frege's system. Roman marks containing Roman letters are constructed using the two methods of forming names. The axioms of BG are asserted using the Roman marks. If one is allowed to assert the Roman mark $f(a)$ which contains Roman letters f and a, then one is allowed to assert any of its instances, such as $\Delta = \Delta$.

Frege's axiom of universal instantiation consists in his assertion of the Roman mark $\forall x f(x) \Rightarrow f(a)$ where f and a are Roman letters.[17] Analogous axioms hold for Frege's other universal quantifiers. Each use of the axiom of universal instantiation one makes will be validated by facts about the constituents of a name. In order to infer a particular claim such as $\Delta = \Delta$ from a universal generalization such as $\forall x(x = x)$, one must first recognize $\forall x(x = x)$ as being an instance of the Roman mark $\forall x f(x)$. One must then recognize $\Delta = \Delta$ as being an instance of the Roman mark f(a). That is, one must be able to *replace* the Roman letter f with the identity sign, and the single occurrence of the Roman letter a with two occurrences of the name Δ. One may then infer the following particular instance of the axiom of universal instantiation: $\forall x(x = x) \Rightarrow \Delta = \Delta$. More generally, any position in a name of BG apt for removal by the second rule is also an apt position to quantify into. Insofar as Frege's proof system makes use of these replacement operations, it issues requirements on his syntax which are fundamentally the same as the requirements issued by his formation rules.

4.3 Is an account necessary? Even if contemporary logicians do offer explicit characterizations of the notions they make use of in their syntactic theories, will Frege run into any problems if he leaves his characterizations of constituency and replacement intuitive? I do not believe that this option is available to Frege. One problem with leaving the account intuitive is that the notions of replacement and constituency which Frege is working with are not the intuitive notions. To bring this out, I will consider an exaggerated case. Frege explicitly allows that one may form a name using the second rule by removing a saturated name Φ from itself and replacing it with a mark of incompleteness, ξ.[18] This entails that ξ itself is a function name. Assume further, as seems inevitable, that Frege needs the second rule to allow one to replace monadic first-level function names in saturated names by marks of incompleteness. Now try to replace all occurrences of the function name ξ with a mark of incompleteness typed for monadic first-level functions, $\Phi(\xi)$, in a larger name, say $\Delta = \Gamma$. It is difficult to determine whether and where ξ occurs in $\Delta = \Gamma$. It is even more difficult to see how one would go about replacing all of its occurrences with occurrences of another expression. Clearly, it is not sufficient for Frege to have an intuitive notion in mind; he must offer a further discussion of what "syntactic constituency" and "replacement" mean in this context.

The strangeness is not, I think, an artifact of the peculiar function name I have chosen to remove. The name $\Delta = \Delta$ needs to be viewed as containing the dyadic function name $\xi = \zeta$ which is removable by the second rule or by existential generalization. It also needs to be viewed as containing the monadic function name $\xi = \xi$. The need for a further characterization of the constituency relation is apparent, because neither function name is explicitly presented in $\Delta = \Delta$, nor is there any intuitive sense in which these functions names are contained in $\Delta = \Delta$. I also doubt that this is merely a problem for function names. Whether a name Δ occurs in a larger name Φ needs to be determined on principled grounds. There need to be rules distinguishing this case from the case in which a mere assemblage of marks occur in Φ which look like Δ.

Another problem is that some of Frege's own goals require that he deliver an account of the syntactic structures—the constituency and replacement facts—of his expressions. One goal is proof-theoretic; the other, semantic. The problem with the proof-theory arises because Frege demands that his proof rules be "specified in advance."[19] Frege will not, I think, have achieved this goal if the proof rules are framed in terms of facts about syntactic structures of which Frege has offered no account. Whether a proof rule applies in a given case will still be undecided by Frege's proof theory. The problem with the semantics arises because Frege's argument requires that assumptions similar to (C) and (D) hold. He is therefore in no position to merely rely on the intuitive notions of constituency and replacement which may fail to validate these assumptions.

5 Accounting for Syntactic Structure

I will assume that Frege needs to have available a more substantive characterization of the syntactic structures of the names of BG. In contemporary systems, the constituency relation and the replacement operation are defined in one of two ways. Some modern systems simply define the relation of constituency and the replacement operation in terms of the syntactic derivation of a name. For these modern systems, the syntactic structure of a name is *given by* the derivation which produced it. Other modern systems define the syntactic structure of a name independently of its derivational history. For instance, each expression may be treated as a structured array of symbols. The constituency relation and the replacement operation are defined in terms of the properties of the arrays. But in these systems the formation rules are specified in terms of the properties of the arrays, so that conditions (C) and (D) hold. In both cases the syntactic properties of expressions can be recursively specified in terms of their syntactic derivations. Thus, on either approach, the correspondence between syntactic structure and derivational history comes out to be a trivial matter. I will argue that neither approach is readily available to Frege.

5.1 Structure as derivation The problem with characterizing the syntactic structure of a name of BG in terms of its derivational history is that the names of BG have multiple syntactic derivations. Consider the above derivation of $\forall x \forall y(x = y)$. In order to construct this name, I first constructed the name $\forall y(\Delta = y)$. I then removed the saturated name Δ by the second rule to produce the function name $\forall y(\xi = y)$. I finally fed this function name into the second-level function name $\forall x \Phi(x)$. In this derivation, the choice of Δ in this construction was arbitrary. I could have just as easily constructed the name using some other saturated name, Γ. Thus, there are at least two syntactic derivations of $\forall x \forall y(x = y)$. One derivation proceeds by constructing $\forall y(\Delta = y)$ and then removing Δ, the other by constructing $\forall y(\Gamma = y)$ and then removing Γ.

The two syntactic histories of $\forall x \forall y(x = y)$ differ rather trivially. There is a sense in which they have the same structure. For instance, the derivations contain the same number of steps. However, the names of BG have multiple syntactic histories which differ in their structures as well. To see this, consider the two following syntactic histories of $-\Gamma$. It could result from saturating $-\xi$ with Γ. But it could also result from the following construction procedure. Saturate $-\xi$ with some random name Δ. Then, immediately strip Δ according to the second rule yielding $-\xi$. Finally, construct $-\Gamma$ by saturating $-\xi$ with Γ by the first rule. In fact, every name of BG occurs in at least one derivation of every other name of BG. One can always undo the consequence of introducing a name X into a syntactic derivation by later stripping all occurrences of X from the names into which it has been introduced by the second rule, or by stripping occurrences of names containing X.

Names have derivations which diverge even more radically. Consider the name $\Delta \Rightarrow (\Gamma = \Sigma)$. One derivation of this name straightforwardly constructs $\Gamma = \Sigma$ by the first rule. It then saturates $\xi \Rightarrow \zeta$ with Δ and $\Gamma = \Sigma$, yielding $\Delta \Rightarrow (\Gamma = \Sigma)$. A different syntactic derivation of $\Delta \Rightarrow (\Gamma = \Sigma)$ proceeds as follows. The first step is to construct $\Delta \Rightarrow (\Omega = \Sigma)$. The next step is to remove Ω by the second rule, yielding $\Delta \Rightarrow (\xi = \Sigma)$. $\Delta \Rightarrow (\Gamma = \Sigma)$ can finally be constructed by saturating $\Delta \Rightarrow (\xi = \Sigma)$ with Γ. In this second derivation, the name $\Gamma = \Sigma$ which clearly should be counted as syntactically occurring in $\Delta \Rightarrow (\Gamma = \Sigma)$ never shows up.

As a consequence of the fact that $\Delta \Rightarrow (\Gamma = \Sigma)$ has a derivation which does not contain an application of a formation rule to its apparent constituent $\Gamma = \Sigma$, the result of replacing $\Gamma = \Sigma$ in $\Delta \Rightarrow (\Gamma = \Sigma)$ with another name, Θ, will not correspond with the result of replacing $\Delta = \Sigma$ with Θ in the derivation of $\Delta \Rightarrow (\Gamma = \Sigma)$. The result of replacing $\Gamma = \Sigma$ in $\Delta \Rightarrow (\Gamma = \Sigma)$ with Θ should be $\Delta \Rightarrow \Theta$. However, the result of replacing all applications of a rule to $\Gamma = \Sigma$ with Θ in the second derivation of $\Delta \Rightarrow (\Gamma = \Sigma)$ considered above would yield back $\Delta \Rightarrow (\Gamma = \Sigma)$, since this derivation contains no applications of a rule to $\Gamma = \Sigma$.

The fact is more striking for function names. The name $\Delta = \Delta$ considered above has a two step derivation. One first saturates $\xi = \zeta$ with Δ, yielding $\xi = \Delta$. One then saturates it again, yielding $\Delta = \Delta$. I should note that this is the shortest derivation for $\Delta = \Delta$ available. Intuitively, Frege should be able to remove the monadic function names $\xi = \xi$ and $\Delta = \zeta$ from this name by the second rule, or in the process of existentially generalizing.

An immediate consequence of the fact that names of BG have multiple derivations is that assumptions (C) and (D) which hold for modern systems fail. The constituents of a name needn't occur in its derivation, and one doesn't automatically yield $X[Y/Z]$ by replacing Y with Z in the derivation of X, since there is no unique thing called "the derivation" of X. But it is clear that (C) and (D) fail in a deeper way. Each name has derivations in which the names which Frege would assume are its constituents do not show up, and has derivations where names show up which he would not assume to be constituents. Similarly, $X[Y/Z]$ is often not the result of replacing Z with Y in a given derivation of X.[20]

These worries can be put more sharply as follows. In contemporary logical languages, the relation "derives from" is well-founded. This fact allows contemporary logicians to offer recursive definitions of notions such as replacement and constituency. By way of contrast, "derives from" is not a well-founded relation for BG. The fact that X results from an application of a formation rule to Y and Z does not guarantee that Y and Z have shorter minimal derivations than X. Attempting to step-by-step reverse the derivation of a name X may result in a never ending loop or an infinite descending chain. For instance, the name $\forall x(x = \Delta)$ may be formed by inserting Δ into $\forall x(x = \xi)$. This name may in turn be derived by removing Δ from $\forall x(x = \xi)$ and so on. Thus, standard recursive definitions cannot be framed in terms of this relation. This is not to say that Frege cannot offer any recursive definitions. Every name of his system has a minimal, or *privileged*, set of derivations. These derivations contain the shortest number of applications of the first and second rules. Frege can run recursive definitions and arguments on the length of a name's minimal derivations. However, these privileged derivations don't seem to get the right results. The minimal derivations of $\Delta = \Delta$ do not contain $\xi = \xi$ which nevertheless must be a constituent of $\Delta = \Delta$ for the proof theory to do what it is supposed to. Similarly, any minimal length derivation of a name which requires the second rule to be derived, such as $\forall x \forall y(x = y)$, will inevitably contain names which do not plausibly count as its constituents.

6 Extra-Derivational Accounts of Structure

It is possible to define the syntactic structure of an expression without reference to its syntactic derivation. Instead, the syntactic structure of an expression can be defined prior to and independently of its syntactic derivation. The formation rules may then be framed in terms of these syntactic properties. Usually in modern systems, care is taken so that the constituents of an expression correspond with the expressions in its syntactic history and assumptions (C) and (D) are, therefore, true.[21] I want to investigate whether Frege can adopt such a strategy, perhaps jettisoning assumptions (C) and (D). This would require explicating the structure of the names of BG without appeal to the derivation rules. I will argue that combining this sort of account with Frege's system threatens to produce a position which is at best unstable, and at worst in conflict with his core typing principles. I will first sketch what I take to be the most natural strategy of this sort. I will expose the problem it raises. I will then present a more abstract characterization of the source of the problem.

In some systems, each expression is identified with a structured array of marks. The structural properties of an expression are then defined in terms of properties of its array. I will attempt to develop such an account to serve Frege's purposes. Frege's system would first need a basic set of symbols such as ξ, ζ, $=$, $\forall$ and so on.[22] The names of BG would be structured n-tuples of these basic symbols. For example, $\forall x \forall y(x = y)$, $\xi = \zeta$, and $\forall y \Phi(y)$ might be identified with the arrays $\langle \forall, x, \forall, y, (, x, =, y,) \rangle$; $\langle \xi, =, \zeta \rangle$; and $\langle \forall, y, \Phi, (, y,) \rangle$, respectively.

Using this strategy, one could then define the relation of constituency and the operation of replacement for BG in terms of the structural properties of the arrays. A saturated name X, or more generally any contiguous, consecutive expression, occurs in another name Y if and only if the elements of the array associated with X occur contiguously in the array associated with Y. Thus, on such a strategy, the name $\Gamma = \Sigma$ would occur in $\Delta \Rightarrow (\Gamma = \Sigma)$, since the elements of $\langle \Gamma, =, \Sigma \rangle$ occur contiguously in $\langle \Delta, \Rightarrow, (, \Gamma, =, \Sigma,) \rangle$. The result of replacing a saturated expression X with another Y in an expression Z could be evaluated by replacing the array associated with X with the array associated with Y in the expression Z. Thus, one arrives at the formula $\Delta \Rightarrow (\Gamma = \Sigma)[\Theta/(\Gamma = \Sigma)]$ by replacing the contiguous, consecutive occurrence of the members of $\langle (, \Gamma, =, \Sigma,) \rangle$ in $\langle \Delta, \Rightarrow, (, \Gamma, =, \Sigma,) \rangle$ with a contiguous, consecutive occurrence of the members of $\langle \Theta \rangle$, yielding $\langle \Delta, \Rightarrow, \Theta \rangle$.[23]

Constituency and replacement for expressions which have the same type as function names are more difficult to define, because the constituents of a function name do not appear as a contiguous sequence of marks in larger names which contain the function name. Rather a name such as $\Delta = \Delta$ contains a name such as $\xi = \xi$ by containing a contiguous sequence of expressions which are the result of replacing all occurrences of each argument marker in $\xi = \xi$ with the same sequence of expressions. More generally, an expression X contains a function name Y with argument ξ if and only if X contains a contiguous sequence of expressions Z which differs from the sequence Y only by replacing all occurrences of the mark of incompleteness in Y with the same expression.[24] Replacement for function names would be defined similarly. Replacing an expression X which contains a mark of incompleteness ξ with another expression Y which contains ξ in Z would proceed by identifying in Z all consecutive, contiguous occurrences of the elements of X such that all occurrences of ξ are uniformly replaced by a consecutive, contiguous sequence of expressions, $\langle a_1, \ldots, a_n \rangle$, and then replacing these occurrences of X with the result of replacing all occurrences of ξ in Y with $\langle a_1, \ldots, a_n \rangle$. Thus, $\Delta = \Delta[\xi = \xi/F(\xi)]$ can be evaluated as follows. $\Delta = \Delta$ is identified with the array $\langle \Delta, =, \Delta \rangle$. This sequence contains a contiguous, consecutive sequence of elements, namely, $\Delta, =, \Delta$, which results from replacing all occurrences of ξ in the sequence $\xi, =, \xi$ with Δ. Now replacing ξ with Δ in the sequence $\langle F, (, \xi,) \rangle$ yields $\langle F, (, \Delta,) \rangle$. Replacing the sequence $\Delta, =, \Delta$ with the elements of $\langle F, (, \Delta,) \rangle$, yields $\langle F, (, \Delta,) \rangle$, or $F(\Delta)$, which is exactly what one would expect.

This account has some notable advantages. One advantage is that it seems to account in a better way for what I shall call cross-categorical replacement. The account of syntactic structure needs to be able to explain more than just how to replace a saturated name with another saturated name, a monadic function name with another monadic function name, and so on. It also needs to do work in explaining what the result of replacing an expression of BG with a mark of incompleteness or a Roman mark is. Since the marks of incompleteness and Roman marks are not actually names of BG, we should never really have expected to be able to recover the replacement facts from the facts about the derivational history of an expression. By way of contrast, because the account on offer is sensitive only to the graphic array associated with an expression, and not to its type or even whether it is a well-formed name of BG, cross-categorical replacements run smoothly. Another advantage is that the account seems robust enough to meet Frege's syntactic needs. Consider the derivation of $\forall x \forall y(x = y)$ given above. This derivation required claims about syntactic structure. For instance, it required (8) Δ occurs in $\forall y(\Delta = y)$ and (10) $\forall y(\xi = y)$ is the result of replacing Δ in $\forall y(\Delta = y)$ with a mark of incompleteness, ξ. These facts are easily recoverable on the current account of syntactic structure.

Unfortunately, these two strengths of the account, its indifference to typing and its robustness, should make concerns about the typing problem re-emerge. Consider again the function name $\forall y(\xi = y)$. Put baldly, the worry is that on the present account $\forall y(\xi = y)$ results from replacing the mark of incompleteness $\Phi(\zeta)$ in $\forall y \Phi(y)$ with the function name $\xi = \zeta$. The array corresponding to $\forall y(\xi = y)$ contains a sequence which is identical to the result of replacing ζ with y in $\xi = \zeta$. In other words, the worry is that $\forall y(\xi = y)$ is the name $\forall y \Phi(y)[\xi = \zeta/\Phi(\zeta)]$. Intuitively, this result makes sense. There is an initial pull to regard $\forall y(\xi = y)$ as composed when a quantifier takes an identity sign as an argument. This may be because we are pretheoretically inclined to accept an account of syntactic structure such as the one on offer.

Another reason that this identification is intuitively plausible is that names containing $\forall y(\xi = y)$ clearly contain the identity sign. For instance, $\xi = \zeta$ is available in $\forall x \forall y(x = y)$ for removal by the second rule or by existential generalization, but is not available in $\forall x f(x)$ which is the result of replacing $\forall y(\xi = y)$ with the Roman mark $f(\xi)$ in $\forall x \forall y(x = y)$. This strongly suggests that $\xi = \zeta$ occurs in $\forall x \forall y(x = y)$ only by occurring in its component name $\forall y(\xi = y)$. If $\xi = \zeta$ does occur in $\forall y(\xi = y)$, then it should be available for the replacement operation.

The names of BG are supposed to have types corresponding to the types of their denotations. Frege says, "The expression for a *function* is *in need of completion, unsaturated*."[25] Function names of BG are supposed to admit in their argument positions only names whose types correspond to their argument markers. The last result is that, on the present proposal, $\forall y(\xi = y)$ is the result of replacing the argument marker $\Phi(\zeta)$ typed for monadic function names in $\forall y \Phi(y)$ with a dyadic function name $\xi = \zeta$. This result is worrying because it seems to violate the spirit, if not the letter, of Frege's typing requirements. If the result were correct, then Frege would have to put some distance between the claim that the quantifier $\forall y \Phi(y)$ takes $\xi = \zeta$ as an argument in the name $\forall y(\xi = y)$ and the claim that $\forall y(\xi = y)$ is the result of replacing the argument marker in the quantifier $\forall y \Phi(y)$ with the function name $\xi = \zeta$. But these two notions are so close together that it is difficult to see how Frege can adopt the present proposal without violating his dictum that "*Functions of two arguments* [and their names] are just as fundamentally different from *functions of one argument* [and their names] as the latter are from *objects* [and their names]. For whereas objects are wholly *saturated*, functions of two arguments are saturated to a lesser degree than functions of one argument, which are already *unsaturated*."[26]

The problematic result is not an artifact of the toy characterization of syntactic structure. It arises for any characterization of syntactic structure which validates some natural assumptions about the replacement operation. Frege is committed by his syntactic derivation to identifying $\forall y(\xi = y)$ with

$$\forall y \Phi(y)[\xi = \zeta[\Delta/\xi]/\Phi(\zeta)][\xi/\Delta]$$

for suitable choice of Δ. In most systems systems, this latter name would be identical to the name $\forall y \Phi(y)[\xi = \zeta[\Delta/\xi][\xi/\Delta]/\Phi(\zeta)]$. In other words, we are trying to replace saturated name Δ with ξ in $\forall y(\Delta = y)$. This latter name results from replacing $\Phi(\zeta)$ in $\forall y \Phi(y)$ with $(\Delta = \zeta)$. Δ occurs only in the last of these names.

So, for suitable choice of Δ, the result of replacing saturated name Δ with ξ in $\forall y(\Delta = y)$ should be the same as the result of replacing $\Phi(\zeta)$ in $\forall y\Phi(y)$ with $(\xi = \zeta)$. Identifications of this sort would not hold generally—even in contemporary systems but would hold for the suitable choices of Δ which Frege would use in his derivation. Given a suitable choice of Δ it is difficult to make sense of an operation as a replacement operation if it fails to validate this identification. For example, the toy account discussed above does validate the identification.

It is natural to think that the successive replacement of ξ with Δ and then Δ with ξ in $\xi = \zeta[\Delta/\xi][\xi/\Delta]$ cancels itself out. I shall argue later that Frege is committed to this principle in his semantic argument. Therefore, it seems that $\forall y\Phi(y)[\xi = \zeta[\Delta/\xi][\xi/\Delta]/\Phi(\zeta)]$ will be identical to $\forall y\Phi(y)[\xi = \zeta/\Phi(\zeta)]$ no matter how syntactic structure is cashed out for BG. But, this is the problematic result we were facing before.

Frege therefore faces a choice. He can characterize the syntactic structure of expression of BG in a way which does not validate the replacement principle of the last paragraph. It is far from obvious that such an account is readily available. Alternatively, Frege could accept the result and try to argue that there is more to the claim that the quantifier $\forall y\Phi(y)$ takes $\xi = \zeta$ as an argument in the name $\forall y(\xi = y)$ than the claim that $\forall y(\xi = y)$ is the result of replacing the argument marker in the quantifier $\forall y\Phi(y)$ with the function name $\xi = \zeta$.

This strategy amounts to a bifurcation of the syntax of each expression. On the one hand, an underlying syntax is stipulated in order to generate the syntactic structure for each name. This syntactic structure is made use of to specify the formation rules and the proof theory of BG. It does not respect Frege's typing principles. On the other hand, another syntax which respects the typing principles is postulated. This other syntax accords a special significance to the formation rules. The strategy would be plausible if the two syntaxes were completely independent. But, as I have been emphasizing, the rules of syntactic derivation cannot even be specified without the underlying syntactic structure. Even worse news for this proposal is that it is likely that if Frege can get his underlying syntax off the ground, then there would be no need to use Frege's formation rules. The set of well-formed names could be specified in terms of his underlying syntax. As a result, adopting either option seems to be a difficult path and would take the discussion considerably beyond anything explicitly in Frege.

It is worth repeating at this point that post-Tarskian logicians have no similar difficulties. In contemporary systems, quantifiers operate on expressions which are of the same type as sentences. Thus, the Tarskian logicians will say that the quantifier $\forall x$ takes $\forall y(x = y)$ in its argument position in $\forall x\forall y(x = y)$. They will readily add that $\forall y$ takes $(x = y)$ in its argument position in $\forall y(x = y)$. The key difference here is, of course, that quantifiers operate on open sentences, which can contain an arbitrary number of free-variables. In Frege's system, however, quantifiers are constrained to operate only on monadic function names, thus creating the problem.

7 Syntactic Assumptions in Frege's Argument

I have argued that Frege's view faces difficulties arising from needs internal to his syntactic theory even when it is supplemented with an account of syntactic structure. I will now consider whether Frege's syntactic theory fulfills the additional requirements generated by his semantic theory. In particular, elements of Frege's recursive specification of the denotations of the names of BG—his argument in §§29–31 that all names in fact have denotations—requires that assumptions similar to (C) and (D) hold. A name's structure must be reflected in its syntactic derivation. In order to explain this, I will first need to discuss how Frege thinks about the semantic property, denoting something, which he is trying to show every name possesses.

7.1 On denoting something Unfortunately, Frege never precisely explains what having a denotation involves. Instead, §29 contains criteria for detecting when a name denotes something. According to these criteria, one may test for whether a name denotes something by determining whether complex names in which it occurs denote something.[27] Frege's criteria test for whether a monadic (or dyadic) first-level function name, $\Phi(\xi)$, denotes something by testing for whether replacing the mark of incompleteness, ξ, with any denoting, saturated name, Δ, always yields a saturated name which denotes something (or in the dyadic case a monadic function name which denotes a monadic function). Similarly, one tests for whether a function name of a higher level denotes by determining whether replacing its marks of incompleteness with arguments of the right type always denotes. Finally, one tests for whether a denoting saturated name, Δ, denotes something by determining whether the name that results from putting Δ in the argument position of any first-level function name denotes something.

The criteria test for whether a name has a denotation in terms of whether other names have a denotation. This entails that Frege would not be able to prove that any names denote something unless he has some independent means of recognizing that some names do.[28] Fortunately, Frege does have other means for recognizing whether a name denotes something. Prior to §29, Frege has issued semantic stipulations concerning his nine basic names. He has made two sorts of stipulation.[29]

For some of his primitive names, Frege seems to have stipulated a denotation outright. For instance, concerning the horizontal, $-\xi$, Frege says,

> I regard [the horizontal] as a function name, as follows:
> $-\Delta$
> is the True if Δ is the True; on the other hand it is the False if Δ is not the True.[30]

I believe that the best way of making sense of Frege's stipulation is to interpret him as stipulating that the horizontal denotes a function, f, such that for any object, a, $f(a)$ is the true if and only if a is the true.[31]

For other names, such as the identity sign $\xi = \zeta$, Frege does not stipulate a denotation outright. Rather, he stipulates when complex names containing the identity sign have denotations. He says,

> "$\Gamma = \Delta$" shall denote the True if Γ is the same as Δ; in all other cases it shall denote the False.[32]

I take Frege to mean here that the complex name $\Gamma = \Delta$ denotes the True when its component names Γ and Δ denote objects a and b such that a is the same as b. Frege, therefore, has two methods for proving that a name denotes something. He may prove that a name denotes using the criteria for denoting something, or he may prove that it does by appealing to his semantic stipulations.

Many commentators have objected that the criteria are not good tests for whether a name denotes something. Frege believes that all functions must be total, or have a value for every object. The criteria test for whether a function name denotes by testing whether the result of putting a saturated name in its argument position always denotes something. This would be a good test for whether the name denotes a total function provided that every possible argument of the function denoted by the function name had a name in BG, which is an assumption that we cannot endorse.

But even setting these problems aside, many commentators agree that Frege's argument fails, even when considered on Frege's own terms.[33] I agree with this assessment and will highlight what I take to be a particularly salient failure of the proof at its outset. I will therefore have nothing else to say about whether the tests are good tests for whether a name denotes something. Frege's argument will be successful if he can show of every name in BG either that it satisfies the criteria or that it denotes according to his stipulations.

7.2 The inductive step of the argument I will describe what I take to be Frege's argument that applying his formation rules to denoting names always produces a denoting name. It is rather trivial to show that applying the first rule to denoting names of BG always produces a denoting name. Suppose Φ is a monadic denoting function name of BG with mark of incompleteness, ξ, and that Δ is a denoting saturated name of BG. By the criteria, one may assume that the result of replacing the mark of incompleteness ξ with any saturated name in Φ is always denoting. Δ is a saturated name. Therefore, $\Phi[\Delta/\xi]$, the result of replacing ξ with Δ in Φ, is a denoting saturated name. This argument can easily be extended to show that the result holds for function names of higher levels and of different adicities.[34]

Frege also needs to prove that applying the second rule to denoting names always produces a denoting name. By the criteria, it suffices to show that for any denoting saturated name Φ containing denoting saturated name Δ, the name $\Phi[\xi/\Delta]$ is a denoting function name. $\Phi[\xi/\Delta]$ is a denoting function name if the result of replacing the mark of incompleteness ξ with any denoting name Γ denotes. Frege may therefore show that $\Phi[\xi/\Delta]$ denotes something by showing that $\Phi[\xi/\Delta][\Gamma/\xi]$—the result of replacing Δ with ξ in Φ and then replacing ξ with Γ—denotes something for any denoting Γ.

The argument for the claim that $\Phi[\xi/\Delta]$ denotes something seems to appear just before the rule is introduced in §29. Immediately after considering names formed in the first way, Frege says,

> A proper name can be employed in the present process of formation only by its filling the argument places of one of the simple or composite [names of][35] first-level functions. Composite names of first-level functions arise in the way provided above only from simple names of first-level functions of two arguments by a proper name's filling the ξ- or ζ-argument-places. Thus the argument-places that remain open in a composite function name are always also the argument-places of a simple name of a function of two arguments. From this it follows that a proper name that is part of a name formed in this way, whenever it occurs, always stands at an argument-place of one of the simple names of first-level functions.[36] If now we replace this proper name [my Δ] at some or all places by another [my Γ], then the function-name so arising is likewise formed in the way stated above, and thus it also has a denotation, if all the simple names employed as well succeed in denoting.[37]

Frege is restricting his attention to denoting names formed in the first way. He wants to argue that applying the second rule to such denoting names always produces denoting names. Thus, Frege's argument will show at most that removing denoting saturated names from denoting names formed entirely in the first way denote. Frege would need to do additional work to show that any application of the second rule preserves denotation. In particular, he would need to run another induction showing that a name whose derivation includes an arbitrary number of applications of the second rule to denoting names denotes. I believe following Heck [8] that Frege is supposing that such an inductive argument is easily available.[38]

The general structure of the argument proceeds as follows. Suppose that Δ is a denoting name occurring denoting saturated name Φ which has a derivation invoking only the first rule. Frege needs to show that $\Phi[\Gamma/\Delta]$ denotes for any denoting Γ. If he can show this, Frege thinks he can infer that the function name $\Phi[\xi/\Delta]$ denotes. This inference is a little fast. In order to show that $\Phi[\xi/\Delta]$ denotes, Frege really needs to show that $\Phi[\xi/\Delta][\Gamma/\xi]$ denotes for any denoting Γ. This suggests that Frege makes the natural assumption that $\Phi[\xi/\Delta][\Gamma/\xi]$ just is $\Phi[\Gamma/\Delta]$. I argued above that this assumption, though natural, is nontrivial, and even has the potential to lead Frege into trouble. Nonetheless, I will suppose that it is legitimate, and see what other syntactic assumptions are required by Frege's argument.

In order to show that $\Phi[\Gamma/\Delta]$ denotes, for any denoting Γ, Frege considers the derivation of Φ which contains applications of the first rule only. Frege supposes that Δ must have made its way into Φ via a step in this derivation. Since Φ is derived wholly in the first way, the only way for Δ to have made it into Φ's derivation is by saturating the argument place of a denoting one place function name $\Psi(\xi)$ or a denoting two place function name $\Omega(\xi,\zeta)$ at various steps in the derivation.[39]

In order to argue that $\Phi[\Gamma/\Delta]$ denotes something, Frege thinks that he needs to show only that replacing Δ with Γ at these steps in the derivation yields denoting names. The reason is that, by assumption, all other steps in the derivation of Φ yield denoting names. Thus, by mimicking the derivation of Φ using Γ rather than Δ, Frege thinks that he will have derived $\Phi[\Gamma/\Delta]$. This is what I take Frege to mean when he says that $\Phi[\Gamma/\Delta]$ is formed in the same way as Φ. Γ occurs in the derivation only by saturating the argument positions of first-level function names.

The last two paragraphs impose a requirement on the names of BG similar to assumptions (C) and (D) for expressions of contemporary logic.

C1. A name X occurs in a name Y which has a derivation D using only the first rule only if X occurs in D.
D1. The name which results from replacing X in the derivation of name Z, derived using only the first rule, with Y is the name $Z[Y/X]$.

Frege is explicitly assuming here that (C1) and (D1) hold for names of BG formed entirely in accordance with the first rule. Call such a name Φ and its derivation D. Frege is assuming that if a saturated name Δ occurs in the name Φ, then there is a step in D in which Δ saturates the argument place of a first-level function name. Another way to put this point is that there are no "emergent" names which occur in a name Φ, but fail to occur in its derivation. These names would, as it were, develop out of a derivation without ever being explicitly introduced as a step in the derivation. (C1) and (D1) are substantive assumptions about the notion of an occurrence. I shall argue that Frege is in no position to make these assumptions.

If assumptions (C1) and (D1) hold up, Frege's argument requires only that he be able to establish the following claim. For any denoting, first-level function name $\Psi(\xi)$ or $\Omega(\xi,\zeta)$, if saturating an argument place of this function name with denoting

Δ in the construction of Φ yielded a denoting name, then so does saturating that argument place with denoting Γ. But this claim follows immediately from the fact that $\Psi(\xi)$, $\Omega(\xi, \zeta)$, and Δ denote something.[40]

Before I move on, I want to comment about how Frege could have mimicked the argument in §30 to argue that a name formed by removing a denoting second-level function name from a denoting saturated name denotes.[41] For names formed in the first way, the argument would run as follows. Suppose that denoting Φ was constructed entirely in the first way and contains monadic first-level function name Ψ. By the reasoning above, the result of removing Ψ from Φ denotes if $\Phi[\Sigma/\Psi]$ denotes for any denoting, monadic first-level function name Σ.

If assumptions (C1) and (D1) hold, then the function name Ψ must have made its way into the construction of Φ by occurring at some point in its syntactic derivation. This could happen in one of two ways, since Φ was constructed only by applications of the first rule. Either Ψ takes a denoting saturated name in its argument place, or Ψ is inserted into the argument position of a denoting second-level function name. Now replacing Ψ with denoting Σ in either sort of construction will produce a denoting name. Thus, by the reasoning above, the name which results from replacing Ψ with Σ in the syntactic derivation of Φ denotes. Therefore, $\Phi[\Sigma/\Psi]$ denotes, if we agree with the assumptions above. The argument, presumably, can be extended by induction on the number of applications of the second rule.[42]

7.3 The correspondence between derivation and structure The problem which now arises is that assumptions (C1) and (D1) are demonstrably false in the case of function names. Recall the derivation of $\Delta = \Delta$ in which the argument places of $\xi = \zeta$ are successively replaced by the name Δ. This derivation makes use only of the first rule. $\xi = \xi$ is a monadic first-level function name of BG and consequently should be removable by the second rule. Nevertheless, $\xi = \xi$ definitely does not occur in the derivation of $\Delta = \Delta$. Thus, the name $\xi = \xi$ "emerges" in the name $\Delta = \Delta$ in the sense described above.

This should raise doubts about whether assumptions (C1) and (D1) hold for saturated names with derivations which contain only the first rule. I have shown that some names contain saturated names which do not occur in some of their derivations. I have not produced a case in which a name contains another saturated name which does not occur in its derivation by the first rule alone. Nevertheless, there is no positive reason to believe that this is impossible. Assumptions (C1) and (D1) are therefore illegitimate without some sort of argument in Frege's proof.

Moreover, Frege will need something stronger than the claim that (C1) and (D1) hold. Frege's argument has established, if anything, that replacing denoting Δ in name Φ with a mark of incompleteness ξ denotes if Φ is formed entirely in the first way. Frege is clear, however, that he wants to be able to remove denoting Δ from an arbitrary denoting saturated name Φ. He says that the second rule can then be applied to further names formed in the first or second ways. This suggests that Frege thinks the argument can be expanded to these cases by induction to cover all of the names of BG.[43] I will shortly sketch such an inductive argument. It seems that in offering any such argument, Frege will have to make use of an assumption which connects the fact that a name contains another name to the fact that the latter name occurs in syntactic derivation of the former. The assumptions which I appeal to are (C2) and (D2).

C2. If a name Y can be constructed using n applications of the second rule, then there is a derivation D of Y which contains n or fewer applications of the second rule such that every name X which occurs in Y also occurs in D and neither X, nor any of its constituents, nor any name in which X occurs is ever removed from any expression by the second rule in any of the steps in D.

D2. If X contains Y and can be constructed in n or fewer applications of the second rule, then there is a derivation, D, of X such that (i) D contains n or fewer applications of the second rule and (ii) replacing all occurrences of name Y with Z in D, results in a derivation of the name $X[Z/Y]$.

(C2) and (D2) are weaker assumptions than (C) and (D). However, in light of the fact that names of BG, such as $A \Rightarrow (B = C)$, have constituents apt for removal by the second rule that do not occur in their derivations it would be difficult to prove that even (C2) and (D2) are true. Frege would, of course, have to say more about how he thinks of the relation of constituency and the replacement operation before one could assess whether these claims were true in his system. Even if this preliminary project could be satisfactorily accomplished, it is far from obvious it would deliver (C2) and (D2) given the highly idiosyncratic nature of Frege's formation rules.

I have presented what I take to be Frege's argument that if denoting, saturated name Δ occurs in denoting, saturated name Φ formed entirely in the first way, then the result of replacing Δ with a mark of incompleteness is a denoting function name. This argument may be extended by induction to any denoting name, if we are willing to assume (C2) and (D2). Before I formulate the inductive hypothesis, I need to introduce some terminology. Each name in BG can be the result of derivations which differ in the number of times that the second rule is applied. But, for each name, there will be a set of derivations that appeal to the second rule fewer times than any other derivation. Call this the set of minimal derivations of a name. For an inductive hypothesis assume that for any denoting X whose minimal derivations contain fewer than j applications of the second rule if X contains Z, then result Y of applying second rule removing denoting Z from X, also has a denotation.

Let Φ be a denoting name the construction of which requires j applications of the second rule and suppose it contains Δ. I need to argue that $\Phi[\Gamma/\Delta]$ denotes for any denoting Γ. By assumption (C2), there is a derivation of Φ containing $k \leq j$ applications of the second rule such that replacing Δ with Γ in the derivation of Φ yields $\Phi[\Gamma/\Delta]$. Then, by (C2) Δ makes its way into Φ only by saturating denoting monadic or dyadic first-level function names. By the reasoning above, replacing Δ at these steps by Γ still yields a denoting name. By the inductive hypothesis, all of the other steps in the derivation are denotation preserving, since they involve fewer than j applications of the second rule. Therefore, the result of replacing Δ with Γ in the derivation of Φ denotes. By assumption (D2), this name will be $\Phi[\Gamma/\Delta]$.

8 Conclusion

I have argued that Frege's syntax in *The Basic Laws* faces two problems. One problem is that the syntactic structure of a name isn't reflected in its syntactic derivations. An independent account of syntactic structure does not seem forthcoming for Frege. As a result, Frege has provided insufficient instruction for applying his formation rules. The other problem is that Frege's argument requires that the derivations of his names coincide with their structure. I have argued that for a large class of names, function names, this assumption is demonstrably false. This casts doubt on whether Frege is licensed to make this assumption for the rest of his names. It is certainly not something which he can assume without argument in his proof.

It falls out of this that Frege has not solved the typing problem. But if this is correct, then Frege's account of quantification is incomplete. He has not adequately explained which positions quantifiers may bind. If this is right, then what has been heralded as Frege's greatest innovation—his theory of iterated quantification—was, in fact, unsatisfactory.[44]

This should not be as surprising as it sounds. The Tarskian method for constructing sentences and its corresponding semantics were genuine innovations. The development required a complete rethinking of the semantic properties of sentences. In order to think about how one sentence contributes to the semantic value of a sentence that contains it, Tarski was forced out of thinking that the semantic contribution a sentence makes to a sentence that contains it (even in an extensional context) is its truth-value. The fundamental semantic property of a sentence, Tarski realized, is not that it has a certain truth-value, but that it is satisfied by certain sequences. I will not here defend the success of Tarski's project. I want to note that it is different from and a significant improvement over Frege's treatment of the semantics of quantification.

None of this is meant to detract from Frege as a great innovator. Frege's analysis of sentences containing multiple quantifiers did in fact lead to this insight. Frege, in fact, developed something like the notation we now use. But like most mathematical innovators, he had an insufficiently precise grasp of the system which he was developing.

Notes

1. At this point in Frege's development, all of the expressions of his system are counted as names. The names are divided into saturated and unsaturated names. The unsaturated names are further divided according to the number and level of the marks of incompleteness they contain.

2. Frege [5], p. 83.

3. According to an interpretation defended in Ricketts [13] and Weiner [15], either Frege offers no semantics for BG or his use of semantic terminology is somehow eliminable. The frequent use of 'denotes' throughout *The Basic Laws* and especially in §29–32 poses a challenge to an advocate of this view. Weiner [14] defends the interpretation by arguing that Frege is offering *elucidations* in §29–32. These would not require Frege to be involved in the sort of semantic projects which the advocate of this interpretation finds objectionable. Throughout this paper, I will suppose that Frege is using the word 'denotes' seriously, though I shall take no stance about his position on its eliminability.

4. This objection is pushed in Resnik [12] and Dummett [3]. The issue is also discussed by [8], [14], and Linnebo [9].

5. For example, Parsons [11] is a prime example of someone who purports to find a problem with the base case of the proof. Martin [10] somewhat mysteriously argues that Frege's proof is flawed because it relies on a false premise, Basic Law V. [12] convincingly argues that Martin conflates Frege's stipulations concerning the semantics of the language with Basic Law V, which is a proof rule within Frege's system.

6. From this point on, I will omit quotation marks from formulas of BG except in excerpts.

7. [5], p. 85.

8. [5], p. 86.

9. This second rule might be seen as a technical implementation of a few informal remarks in *Begriffsschrift*. There, Frege said that a "content" may be carved into several function and argument structures. Frege says "The same conceptual content may be regarded as a function of this or that argument, so long as function and argument are completely determinate" (Frege [6], p. 14). Frege in *Basic Laws* holds a saturated name may be carved into function name and argument name in many different ways.

10. I am slightly diverging from Frege here, as he allows that any number of occurrences of Δ may be replaced with a mark of incompleteness. I am ignoring the added flexibility present in Frege's formulation of the second rule, because it makes no difference to the issues I am discussing.

11. [5], p. 86.

12. See [5], p. 92.

13. See [3], pp. 217–22, and [8], p. 440 and p. 450.

14. To take just one instance, Ebbinghaus et al. [4], p. 23, recursively define a function, SF, from formulas to their set of subformulas as follows: "$SF(t_1 = t_2) = \{t_1 = t_2\}$[;] $SF(Rt_1 \ldots t_n) = \{Rt_1 \ldots t_n\}$[;] $SF(\neg\varphi) = \{\neg\varphi\} \cup SF(\varphi)$[;] $SF(\varphi * \psi) = \{\varphi * \psi\} \cup SF(\varphi) \cup SF(\psi)$ for $* = \wedge, \vee, \rightarrow, \leftrightarrow$[;] $SF(\forall x\Phi) = \{\forall x\varphi\} \cup SF(\varphi)$[;] $SF(\exists x\Phi) = \{\exists x\varphi\} \cup SF(\varphi)$." A is a subformula of B if and only if it is a member of $SF(B)$. The replacement operation is recursively defined ([4], p. 53).

15. Frege actually needs claims slightly weaker than (C) and (D) which do not hold either.

16. Dummett [1], pp. 15–16.

17. See the discussion in [8], pp. 443–46.

18. "We begin by forming a name in the first way, and we then exclude from it at all or some places, a proper name that is a part of it (*or coincides with it entirely*)—but in such a way that these places remain recognizable as argument-places of type 1 [which take saturated names]" ([5], p. 86). Emphasis added.

19. [5], p. 2.

20. In some places, considerations like these push Dummett toward the view that function names which result from applications of the second rule lack constituents at all. He says in Dummett [2], p. 296, "For a predicate obtained by decomposition of a sentence [by the second rule], we need no notion of *its* constituents. Dispensing with this notion, however, lands us in difficulties how to explain what are the *ultimate* constituents

of quantified sentences.” Dummett goes on to argue that Frege does not need the notion of a constituent of a complex predicate or quantified sentence in order to account for the fact that one may derive a claim such as $\forall y((a = y) = b)$ from the claim $\forall x \forall y((x = y) = b)$, and that the syntactic derivation is sufficient. This argument is strikingly myopic. Frege may not need the notion of a constituent of $\forall y((\xi = y) = b)$ or $\forall x \forall y((x = y) = b)$ in order to account for one particular inference involving them. However, there are other inferences such as the inference from $\forall x \forall y((x = y) = b)$ to $\exists z \forall x \forall y((x = y) = z)$ for which Frege will need to invoke the fact b is a constituent of $\forall x \forall y((x = y) = b)$. Similar considerations apply, if one tries to existentially generalize on, say, the identity sign itself.

21. This system discussed in [4] is merely one example.

22. The primitives of Frege’s systems will not do, since Frege needs to be able to identify and replace constituents of these expressions.

23. I am ignoring some complications due to parentheses which can be easily repaired.

24. It is important that the mark of incompleteness in Y need not actually be replaced by a name of BG in Z. For instance, the quantifier $\forall x \Phi(x)$ needs to contain the mark of incompleteness $\Phi(\xi)$ which has the same type as a monadic, first-level function name. This is required in order to apply the first rule to the quantifier. The toy characterization validates this principle, because the sequence $\langle \forall, x, \Phi, (, x,) \rangle$ contains the sequence $\langle \Phi, (, x,) \rangle$ and $\langle \Phi, (, x,) \rangle$ differs from $\Phi(\xi)$ only by x replacing ξ. This is so, despite the fact that x is not a name of BG.

25. [5], p. 34.

26. [5], p. 34.

27. It is easy to see these criteria as an implementation of the context principle from *The Foundations of Arithmetic* (Frege [7]).

28. Frege even says in §30 that his criteria are not meant to define having a denotation “…because their [the criteria’s] application always presupposes that we can already recognize some names as denoting” ([5], p. 85).

29. Thanks to an anonymous referee for the Notre Dame Journal of Formal Logic for pointing out that some of Frege’s stipulations are of the first sort.

30. [5], p. 38.

31. This reading might be contentious since, rather than explicitly introducing an object, a, and saying it is the referent of ‘Δ’, Frege reuses Δ to give the truth conditions for ‘$-\Delta$’. If one thought that Frege was not offering a substantive semantic theory, then one might reject my assumption that Frege is implicitly supposing that Δ is the denotation of ‘Δ’.

32. [5], p. 40.

33. Among these commentators are [11] and [12]. [3] proposes that Frege's argument fails for second-level quantification.

34. Frege does not explicitly argue for the claim that applying the first rule to denoting names of BG produces denoting names. I believe that the most natural account of this is that it follows so immediately from the criteria.

35. This is Furth's addition.

36. I have deleted Furth's addition of "of two arguments" from the end of this sentence. In doing so, I am following [8].

37. [5], pp. 85–86.

38. Heck makes this suggestion in his [8], p. 467, footnote 17.

39. Δ may also make its way into Φ's derivation by being Φ itself in accordance with Frege's suggestion in §30. I will ignore this case for the purposes of this argument, since a repair is relatively trivial.

40. Let $\Omega(\xi, \zeta)$ be any denoting first-level function name. Assume without loss of generality that saturating the ξ argument place of $\Omega(\xi, \zeta)$ with Δ yields a denoting function name, $\Omega(\Delta, \zeta)$. Frege needs to show that $\Omega(\Gamma, \zeta)$ denotes as well. By the criteria for denoting something, this can be done by showing that $\Omega(\Gamma, \Sigma)$ denotes, for any denoting Σ. Because Γ denotes, Frege may assume that $\Omega(\Gamma, \Sigma)$ denotes, for any denoting name Σ and any two-place, denoting, first-level function name $\Omega(\xi, \zeta)$. Let Σ be a denoting name. Then, $\Omega(\Gamma, \Sigma)$ denotes. Therefore, $\Omega(\Gamma, \zeta)$ denotes as well. Since replacing Δ with Γ at every occurrence in the derivation of Φ denotes, it falls out that $\Phi[\Gamma/\Delta]$ denotes something, if Φ is constructed in the first way.

41. The fact that Frege fails to actually offer this proof has led Dummett to complain about Frege's "insouciance" concerning the second-level quantifiers ([3] p. 218).

42. In [8], Heck interprets Frege as making the syntactic claim that if Φ is formed entirely in the first way, so too is $\Phi[\Gamma/\Delta]$. Heck puts a lot of weight on the fact that Frege says that $\Phi[\Gamma/\Delta]$ is formed in the same way as Φ, or as Frege says "in the way stated above." Heck takes this to mean that $\Phi[\Gamma/\Delta]$ is formed entirely in the first way. If this claim is true, then Frege can conclude that $\Phi(\Gamma)$ denotes for any denoting Γ. The reason is that its construction will completely mimic the construction of the original. Unfortunately, this claim is false as Heck points out, since Γ itself need not be formed in the first way, thus $\Phi[\Gamma/\Delta]$ may contain applications of the second rule. Heck thinks that his observation is devastating to Frege as I—and the majority of commentators—have interpreted him. Heck then proposes an innovative, but in my view radical, reinterpretation of Frege. I think that Heck is misreading Frege's argument. All that Frege needs is that Γ makes its way into the syntactic derivation of $\Phi[\Gamma/\Delta]$ in the same way that Δ makes its way into the derivation of Φ. That is, all that Frege needs is that denoting Γ makes its way into $\Phi[\Gamma/\Delta]$ by saturating the argument positions of denoting first-level function names. Frege does not need the stronger claim that $\Phi[\Gamma/\Delta]$ is constructed entirely in the first way for his proof to go through. Thus, at least as far as securing the inductive argument goes, one does not need to reinterpret Frege. See [8], p. 448. Heck has other arguments

for his interpretation, but they concern the interpretation of the criteria for denotation and are therefore outside of the scope of this paper.

43. “The function-name resulting from this [the second rule] likewise always has a denotation if the simple names from which it was formed denote something; and it may be used further to form names in the first way or in the second” ([5], p. 86).

44. Frege may not have been alone in this respect. In the introduction to the second edition of *Principia Mathematica*, Russell and Whitehead suggest that multiply quantified sentences are formed by removing individual constants from sentences which contain them. Despite this similarity, it is not clear to me the extent to which Russell and Whitehead’s formalism follows Frege’s. See Whitehead and Russell [16], Section II.3, pp. xx–xxii.

References

[1] Dummett, M., *Frege: Philosophy of Language*, Harvard University Press, Cambridge, 1981. 273

[2] Dummett, M., *The Interpretation of Frege’s Philosophy*, Harvard University Press, Cambridge, 1981. 273

[3] Dummett, M., *Frege: Philosophy of Mathematics*, Harvard University Press, Cambridge, 1991. MR 1154309. 272, 273, 275

[4] Ebbinghaus, H.-D., J. Flum, and W. Thomas, *Mathematical Logic*, Undergraduate Texts in Mathematics. Springer-Verlag, New York, 1996. MR 1278260. 273, 274

[5] Frege, G., *The Basic Laws of Arithmetic. Exposition of the System*, edited by M. Furth, University of California Press, Berkeley, 1967. Zbl 0155.33601. MR 0176905. 272, 273, 274, 275, 276

[6] Frege, G., *Begriffsschrift*, pp. 1–20 in *Philosophical Writings of Gottlob Frege*, edited by P. Geach and M. Black, Basil Blackwell, Oxford, 1970. 273

[7] Frege, G., *The Foundations of Arithmetic: A Logico-Mathematical Enquiry into the Concept of Number*, edited by J. L. Austin, Northwestern University Press, Evanston, 1996. 274

[8] Heck, R. G., Jnr., “*Grundgesetze der Arithmetik*, I, §§29–32,” *Notre Dame Journal of Formal Logic*, vol. 38 (1997), pp. 437–74. Zbl 0915.03005. MR 1624970. 269, 272, 273, 275

[9] Linnebo, Ø., “Frege’s proof of referentiality,” *Notre Dame Journal of Formal Logic*, vol. 45 (2004), pp. 73–98. Zbl 1104.03005. MR 2130801. 272

[10] Martin, E., Jr., “Referentiality in Frege’s *Grundgesetze*,” *History and Philosophy of Logic*, vol. 3 (1982), pp. 151–64. MR 691225. 272

[11] Parsons, C., “Frege’s theory of number,” pp. 150–75 in *Mathematics in Philosophy. Selected Essays*, Cornell University, Ithaca, 1983. MR 728948. 272, 275

[12] Resnik, M., “Frege’s proof of referentiality,” pp. 177–95 in *Frege Synthesized*, edited by L. Haaparanta and J. Hintikka, D. Reidel Publishing Company, Boston, 1986. 272, 275

[13] Ricketts, T., "Objectivity and objecthood," pp. 65–95 in *Frege Synthesized*, edited by L. Haaparanta and J. Hintikka, D. Reidel Publishing Company, Boston, 1986. 272

[14] Weiner, J., "Section 31 revisited: Frege's elucidations," pp. 149–82 in *From Frege to Wittgenstein: Perspectives on Early Analytic Philosophy*, edited by E. H. Reck, Oxford University Press, New York, 2002. 272

[15] Weiner, J., "Semantic descent," *Mind*, vol. 114 (2005), pp. 321–54. MR 2159934. 272

[16] Whitehead, A. N., and B. Russell, *Principia Mathematica to* *56, Cambridge University Press, New York, 1962. Zbl 0101.24902. MR 0130163. 276

Acknowledgments

Thanks to Derek Ball, Daniel Bonevac, Cory Juhl, Robert Koons, and two anonymous referees at the *Notre Dame Journal of Formal Logic*. Special thanks are due to Josh Dever and Mark Sainsbury.

The University of Texas at Austin
WAG 316
C3500
Austin TX 78712
USA
bpickel@mail.utexas.edu