Leśniewski and the Logic of Names: Lecture Seven
Intensive Course STOQ 50547
II Semester April 26–30, 2004

Prof. Nino B. Cocchiarella

April 29, 2004, 11:15–12:45

1 Introduction

Referential concepts in conceptual realism are based a logic of proper and common names as parts of quantifier phrases. This conceptualist logic of names is similar to Leśniewski's logic of names in that the category of names in Leśniewski's system also contains common as well as proper names. Leśniewski's logic is different, however, in that names do not occur as parts of quantifier phrases but are of the same category as objectual variables. Leśniewski described his logic of names as "ontology," apparently because it was to be the initial level of a theory of types, which Leśniewski called semantic categories.[1] Leśniewski's general framework also included mereology, which is a theory of the relation of part to whole, and protothetic, a quantificational logic over propositions and n-ary truth-functions, for all positive integers n.

Leśniewski's logic of names has been used for years as a framework in which to interpret and reconstruct various doctrines of medieval logic.[2] Recently, we have given an alternative interpretation and reconstruction of medieval logic in terms of the framework of conceptual realism.[3] It is relevant therefore to see how, or in what respect, Leśniewski's logic of names is similar to our conceptualist logic of names. In fact, as we will explain, Leśniewski's logic of names can be completely interpreted, and in that sense is reducible, to our conceptualist logic of names.[4]

Leśniewski's based his system of mereology, i.e., his logic of the relationship between parts and wholes, on his logic of names, and though the exact form of this connection is not clear it has something to do with the notion of classes

[1] See Lejewski 1958, p. 152, Slupecki 1955 and Iwanuś 1973 for a description of Leśnieski's general framework as well as his logic of names.

[2] See, e.g., Henry 1972.

[3] See Cocchiarella 2001 for the details of such a reconstruction.

[4] See Cocchiarella 2001 for a detailed proof that Leśniewski's logic of names is reducible to our conceptualist logic of names.

in a collective sense as opposed to a distributive sense. Our conceptualist logic of names, on the other hand, is the basis of a logic of classes as many, i.e., a logic of classes in the distributive sense, which in some respects is similar to Leśniewski's mereology, but in other respects it is also different. Unlike the connection between Leśniewski's logic of names and his mereology, however, the connection between our logic of classes as many and our simple logic of names is both precise and a fundamental part of conceptual realism.

We will first briefly describe Leśniewski's logic of names and then formulate the simple logic of names that is a fragment of our broader, more comprehensive formal ontology for conceptual realism. We will then explain how Leśniewski's system can be interpreted within our logic and how certain oddities of Leśniewski's system can be explained in terms in our logic where those oddities do not occur. We will then explain how the logic of classes as many is developed as an extension of the simple logic of names.

The logic of names of Leśniewski's general framework and of our framework of conceptual realism provide, incidentally, another illustration, or paradigm, of how different parts or aspects of a formal ontology can be developed independently of, or even prior to, the construction of a comprehensive system all at once.

2 Leśniewski's Logic of Names

In Leśniewski's logic of names, as in our conceptualist logic, there is a distinction between

> (1) shared, or common names, such as 'man', 'horse', 'house', etc., and even the ultimate superordinate common name 'thing', or 'object';
>
> (2) unshared names, i.e., names that name just one thing, such as proper names; and
>
> (3) vacuous names, i.e., names that name nothing.[5]

There is a categorial difference between names in Leśniewski's logic and names in our conceptualist logic, however. In Leśniewski's logic names are of the same category as the objectual variables, which means that they are legitimate substituends for those variables in first-order logic. In our conceptualist logic, names belong to a category of expressions to which quantifiers are applied and that result in quantifier phrases such as 'every raven', 'some man', 'every citizen over eighteen, etc.

The one primitive of Leśniewski's logic, aside from logical constants, is the relation symbol 'ε' for singular inclusion, which is read as the copula 'is (a)', as in 'John is a teacher', where both 'John' and 'teacher' are names.[6] Using 'a',

[5] See Lewjeski's 1958 for a detailed description of Leśniewski's logic of names.

[6] Apparently, it was Łukasiewicz who prompted Leśniewski to develop his logic of names when he expressed dissatisfaction with the way G. Peano used '$\in$' for the copula in set theory. Cp. p. 414 of Rickey's 1977.

'b', 'c', etc., as objectual constants and variables for names, the basic formula of the logic is '$a \; \varepsilon \; b$', where either shared, unshared, or vacuous names may occur in place of 'a' and 'b'.

> A statement of the form '$a \; \varepsilon \; b$' is taken as true if, and only if 'a' names exactly one thing and that thing is also named by 'b', though 'b' might name other things as well, as in our example of 'John is a teacher'.

Identity is not a primitive logical concept of Leśniewski's system, as it is in our conceptualist logic, but is defined instead as follows:

$$a = b =_{df} a \; \varepsilon \; b \wedge b \; \varepsilon \; a.$$

That is, '$a = b$' is true in Leśniewski's logic if, and only if, 'a' and 'b' are unshared names that name the same thing. That seems like a plausible thesis, except that then, where 'a' is a shared or vacuous name, '$a = a$' is false. In fact, because there are necessarily vacuous names, such as the complex common name 'thing that is both square and and not square', the following is provable in Leśniewski's logic:

$$(\exists a)(a \neq a),$$

which does not seem at all like a plausible thesis. Of course, this means that

$$(\forall a)(a = a)$$

is not a valid thesis in Leśniewski's system. Leśniewski does include a weak notion of identity, which is defined as follows

$$a \circ b =_{df} (\forall c)(c \; \varepsilon \; a \leftrightarrow c \; \varepsilon \; b),$$

and which does not have these odd features. This notion, of course, means that a and b are co-extensive, not identical. But then, Leśniewski insisted on his logic being extensional, and not intensional, in which case $a \circ b$ does amount to a kind of identity when a is either a shared or unshared name. Of course, in that case all vacuous names, such as 'Pegasus' and 'Bucephalus' are identical in this weak sense. It also means that Leśniewski's ontology is not an appropriate framework for tense and modal logic, or for intensional contexts in general.[7]

Another valid thesis of Leśniewski's logic is,

$$\varphi(c/a) \rightarrow (\exists a)\varphi(a),$$

which seems counter-intuitive when 'c' is a vacuous name. The following, for example, would then be valid

$$\neg(\exists b)(b = Pegasus) \rightarrow (\exists a)\neg(\exists b)(b = a),$$

[7] One can intensionalize Leśniewski's framework, of course, even though he himself was against such a move.

and therefore, because the antecedent is true, then so is the consequent, which says that something is identical with nothing.

Perhaps these oddities can be explained by interpreting the Leśniewski's first-order quantifiers substitutionally rather than as referentially. That, however, is not how Leśniewski understood his logic of names, which, as we have said, he also called ontology.[8] A logic that interprets the quantifiers of its basic level substitutionally, rather than referentially, would be an odd sort of formal ontology. Does that mean that a name would be required for every object in the universe, including, e.g., every grain of sand and every microphysical particle?

We should also note that Leśniewski's epsilon symbol, 'ε', for singular inclusion should not be confused with the epsilon symbol, '$\in$', for membership in a set. In particular, whereas the following:

$$a \ \varepsilon \ b \rightarrow a \ \varepsilon \ a,$$

$$a \ \varepsilon \ b \wedge b \ \varepsilon \ c \rightarrow a \ \varepsilon \ c,$$

are both theorems of Leśniewski's system, both are invalid for membership in a set.

Finally, the only nonlogical axiom of ontology—i.e. the only axiom in addition to the logical axioms and inference rules of first-order predicate logic *without* identity—assumed by Leśniewski was the following:

$$(\forall a)(\forall b)[a\,\varepsilon\,b \leftrightarrow (\exists c)c\,\varepsilon\,a \wedge (\forall c)(c\,\varepsilon\,a \rightarrow c\,\varepsilon\,b) \wedge (\forall c)(\forall d)(c\,\varepsilon\,a \wedge d\,\varepsilon\,a \rightarrow c\,\varepsilon\,d)].$$

This axiom alone does not suffice for the *elementary* logic of names, however, i.e., for Leśniewski's logic of names as formulated independently of the type-theoretic part of Leśniewski's framework. It has been shown, however, that adding the following two axioms to the one above does suffice[9]:

$$(\forall a)(\exists b)(\forall c)[c \ \varepsilon \ b \leftrightarrow c \ \varepsilon \ c \wedge c \not\varepsilon \ a], \qquad (\textbf{Compl})$$

$$(\forall a)(\forall b)(\exists c)(\forall d)[d \ \varepsilon \ c \leftrightarrow d \ \varepsilon \ a \wedge d \ \varepsilon \ b]. \qquad (\textbf{Conj})$$

Expressed in terms of our conceptualist logic, where names are taken to express name (or nominal) concepts, what these axioms stipulate is that there is a complementary name concept corresponding to any given name concept, and, similarly, that a conjunctive name concept corresponds to any two name concepts with singular inclusion taken conjunctively.

3 The Simple Logic of Names

The simple logic of names, which, as we said, is an independent fragment of the full logic of conceptual realism, can be described as a version of an identity logic that is free of existential presuppositions regarding singular terms—i.e., free

[8] See, e.g., Lejewski1958 and Küng and Canty 1979 for a discussion of this issue.

[9] See Iwanuś 1973 for a proof of this claim.

objectual variables and expressions that can be properly substituted for such. It contains both absolute and relative quantifier phrases, i.e., relative quantifier phrases such as $(\forall xA)$ and $(\exists xA)$, as well as absolute quantifier phrases such as $(\forall x)$and $(\exists y)$, which are read as $(\forall xObject)$ and $(\exists yObject)$, respectively. We will continue to use x, y, z, etc., with or without numerical subscripts, as objectual variables, as we did in our second lecture.

We will now also use A, B, C, with or without numerical subscripts, as name, or "nominal", variables. As explained in the previous lecture, complex names are formed by adjoining (so-called "defining") relative clauses to names, and we use '/', as in 'A/φ' to represent the adjunction of a formula φ to the name A (which may itself be complex). Thus, e.g., the quantifier phrase representing reference to a house that is brown would be symbolized as $(\exists xHouse/Brown(x))$.

We continue to take the universal quantifier, $\forall$, the (material) conditional sign, $\rightarrow$, the negation sign, $\neg$, and the identity sign, $=$, as primitive logical constants, and assume the others to be defined in the usual (abbreviatory) way. The absolute quantifier phrases $(\forall x)$ and $(\exists x)$ are read as 'Every *object*' and 'Some *object*', or, equivalently, as 'Every*thing*' and 'Some*thing*', respectively. That is, the absolute quantifiers are understood as implicitly containing the most general or ultimate common name 'object' (which we take to be synonymous with 'thing'). The quantifier phrases $(\forall A)$ and $(\exists A)$ are taken as referring to every, or to some, name concept, respectively. Name constants are introduced in particular applications of the logic.[10]

Because complex names contain formulas as relative clauses, *names* and *formulas* are inductively defined simultaneously as follows[11]:

- (1) every name variable (or constant) is a *name*;
- (2) for all objectual variables x, y, $(x = y)$ is a *formula*; and
- if φ, ψ are formulas, B is a name (complex or simple), and x and C are an objectual and a name variable respectively, then
  (3) $\neg\varphi$,
  (4) $(\varphi \rightarrow \psi)$,
  (5) $(\forall x)\varphi$,
  (6) $(\forall xB)\varphi$, and
  (7) $(\forall C)\varphi$ are *formulas*, and
  (8) B/φ and
  (9) $/\varphi$ are *names*, where $/\varphi$ is read as 'object that is φ'.

We assume the usual definitions of bondage and freedom for objectual variables and of the proper substitution of one objectual variable for another in a formula, and similarly we assume the definitions of bondage and freedom of occurrences of name variables in formulas, and the proper substitution in a

[10] The absolute quantifiers and the quantifiers for name concepts are understood to be relativized to a given universe of discourse in an applied form of the logic.

[11] We adopt the usual informal conventions for dropping parentheses and for sometimes using brackets instead of parentheses.

formula φ of a name variable (or constant) B for free occurrences of a name variable C.

Definition: A complex name B/ξ *is free for* C *in* φ with respect to an objectual variable x (as place holder) if
(1) for each variable y such that $(\forall yC)$ occurs in φ and C is free at that occurrence, then y is free for x in B/ξ, and
(2) no variable, name or objectual, other than x that is free in B/ξ becomes bound when a free occurrence of C in φ is replaced by an occurrence of $B/\xi(y/x)$.[12]

Note: If a name B (complex or simple) is free for C in φ with respect to a variable x, then the proper substitution of B for C in φ with respect to x is represented by $\varphi(B[x]/C)$.

Among the rules, or meaning postulates, of our logic of names are four that were mentioned in our previous lecture. The first two connect relative quantifier phrases with absolute phrases, and the next two amount to export and import rules for quantifier phrases with complex names.

$$(\forall xA)\varphi \leftrightarrow (\forall x)[(\exists yA)(x = y) \rightarrow \varphi], \tag{\textbf{MP1}}$$

$$(\exists xA)\varphi \leftrightarrow (\exists x)[(\exists yA)(x = y) \wedge \varphi], \tag{\textbf{MP2}}$$

$$(\forall xB/\varphi)\psi \leftrightarrow (\forall xB)[\varphi \rightarrow \psi], \tag{\textbf{MP3}}$$

$$(\exists xB/\varphi)\psi \leftrightarrow (\exists xB)[\varphi \wedge \psi]. \tag{\textbf{MP4}}$$

Of course, strictly speaking, **(MP2)** and **(MP4)** are redundant because $(\exists xA)$ is taken as an abbreviation for $\neg(\forall xA)\neg$, whether A is simple or complex. For this reason, we will restate **(MP1)** and **(MP3)** as **axioms 10** and **11**. below.

The axioms of the simple logic of names are those of the free logic of identity plus the axioms for name quantifiers:

Axiom 1: All tautologous formulas;

Axiom 2: $(\forall x)[\varphi \rightarrow \psi] \rightarrow [(\forall x)\varphi \rightarrow (\forall x)\psi]$;

Axiom 3: $(\forall C)[\varphi \rightarrow \psi] \rightarrow [(\forall C)\varphi \rightarrow (\forall C)\psi]$;

Axiom 4: $(\forall C)\varphi \rightarrow \varphi(B[x]/C)$, where B is free for C in φ with respect to x;

Axiom 5: $\chi \rightarrow (\forall C)\chi$, where C is not free in χ;

Axiom 6: $\chi \rightarrow (\forall x)\chi$, where x is not free in χ;

Axiom 7: $(\forall x)(\exists y)(x = y)$, where x, y are different variables;

Axiom 8: $x = x$;

Axiom 9: $x = y \rightarrow (\varphi \rightarrow \psi)$, where φ, ψ are atomic formulas and ψ

[12] The use of '/' in '$\xi(y/x)$' represents the result of properly substituting y for x in ξ, and should not be confused with the use of '/' to generate complex names.

is obtained from φ by replacing an occurrence of b by a[13]

Axiom 10: $(\forall xA)\varphi \leftrightarrow (\forall x)[(\exists yA)(x = y) \to \varphi]$, where x, y are different variables;

Axiom 11: $(\forall xA/\psi)\varphi \leftrightarrow (\forall xA)[\psi \to \varphi]$.

We assume as primitive inference rules *modus ponens* **(MP)** and universal generalization **(UG)** to absolute quantifiers indexed by either an objectual or a name variable. The rule of universal generalization for relative quantifiers,

$$if \ \vdash \varphi, \ then \vdash (\forall xA)\varphi, \tag{$\mathbf{UG}_N$}$$

is derivable by **(UG)** from **Axiom 10**. The usual laws for interchanging provably equivalent formulas and for rewriting bound variables are easily seen to be derivable as well. The universal instantiation law in free logic for objectual variables,

$$(\exists x)(x = y) \to [(\forall x)\varphi \to \varphi(y/x)], \tag{$\exists/\mathbf{UI}$}$$

where x, y are distinct variables and y is free for x in φ, is derivable by Leibniz's law (LL), i.e., **Axiom 9**, **(UG)**, **Axioms 2** and **6**, and tautologous transformations. The theorems that are counterparts to **Axioms 10** and **11** for the existential quantifiers, namely,

T1: $\vdash (\exists xA)\varphi \leftrightarrow (\exists x)[(\exists yA)(x = y) \wedge \varphi]$,

and

T2: $\vdash (\exists xA/\psi)\varphi \leftrightarrow (\exists xA)[\psi \wedge \varphi]$

are also derivable by elementary transformations and the definitions for $\wedge$ and $\exists$. Also, because absolute quantifiers are viewed as implicitly containing the common name 'object', we assume that **Axiom 11** has the following schema as a special instance.

T3: $\vdash (\forall x/\psi)\varphi \leftrightarrow (\forall x)[\psi \to \varphi]$.

The following are some obvious theorems that are easily seen to be provable.

T4: $(\forall x)\varphi \to (\forall xA)\varphi$.

T5: $(\forall xA)\varphi \to [(\exists zA)(y = z) \to \varphi(y/x)]$, where y is free for x in φ.

T6: $(\exists xA)(y = x) \to (\exists x)(y = x)$.

T7: $(\forall x)\varphi \leftrightarrow (\forall A)(\forall xA)\varphi$, where A is not free in φ.[14]

[13] As noted in our second lecture, the full version of Leibniz's law is derivable from this axiom so long as no intensional operators are introduced into the logic.

[14] **Proof:** The left-to-right direction follows by **T4**, **(UG$_N$)**, quantifier laws. The right-to-left direction follows by first universally instantiating A to *thing identical to itself*, i.e., $/(x = x)$, so that by axiom 4 we have $\vdash (\forall A)(\forall xA)\varphi \to (\forall x/x = x)\varphi$, and, by **T3**, $\vdash (\forall x/x = x)\varphi \to (\forall x)[x = x \to \varphi]$, from which, by **axiom 8**, **(UG)**, and **axioms 2** and **1**, $\vdash (\forall x/x = x)\varphi \to (\forall x)\varphi$; and from this the right-to-left direction of **T7** follows.

Finally, the following comprehension principle

$$(\forall A)(\exists B)(\forall x)[(\exists yB)(x = y) \leftrightarrow (\exists yA)(x = y)] \qquad (\mathbf{CP_N})$$

is immediately derivable from **Axiom 1,** or really from the contrapositive of **Axiom 1**, which amounts to a form of existential generalization for name concepts.

4 The Consistency and Decidability of the Simple Logic of Names

The simple logic of names that we have formulated in the previous section is both consistent and decidable.[15] This follows by noting that the logic is actually equiconsistent with monadic predicate logic, which is known to be consistent and decidable. We will not go through all of the details of showing this here, but we give a general outline of the proof.[16]

First, let us note that by the rule **(MP3)** and a simple inductive argument it can be shown that every formula of our conceptualist system in which a complex name occurs is provably equivalent to a formula in which no complex name occurs.

Metatheorem 1: If φ is a formula of the simple logic of names—i.e., of the free first-order logic of identity extended to include name (name) variables, quantification over such, and restricted quantifiers with respect to such—then there is a formula ψ in which no complex name occurs such that φ is provably equivalent to ψ in this logic, i.e., $\vdash \varphi \leftrightarrow \psi$.

Because of the above metatheorem, we can, in what follows, restrict ourselves to formulas in which no complex name occurs.[17] We assume a one-to-one correlation of the name variables A, B, C, D, etc., with one-place predicate variables F_A, F_B, F_C, F_D, etc., and inductively define a translation function trs^* from the formulas of our simple logic of names in which no complex name occurs into formulas of second-order monadic predicate logic (with identity) as follows:

1. $trs^*(x = y) = (x = y)$,
2. $trs^*(\neg\varphi) = \neg trs^*(\varphi)$,
3. $trs^*(\varphi \to \psi) = [trs^*(\varphi) \to trs^*(\psi)]$,
4. $trs^*((\forall x)\varphi) = (\forall x)trs^*(\varphi)$,

[15] It is important to keep in mind here that the only relation symbol of the system is the identity sign.

[16] For all of the details, see Cocchiarella 2001a.

[17] We also ignore name constants and concern ourselves only with formulas in which no applied descriptive constants occur.

5. $trs^*((\forall xA)\varphi) = (\forall x)[F_A(x) \to \varphi]$,
6. $trs^*((\forall A)\varphi) = (\forall F_A)trs^*(\varphi)$.

It is clear that the translation under trs^* of every theorem of our simple logic of names in which no complex name occurs becomes a theorem of second-order monadic predicate logic. The restriction to formulas in which no complex names occur can be dropped by allowing, for each formula φ in which a complex name does occur, the translation function trs^* to assign $trs^*(\psi)$ to φ, where ψ is the first formula (in terms of some alphabetic ordering) in which no complex name occurs and such that $\vdash \varphi \leftrightarrow \psi$. By extending trs^* in this way, it then follows that every theorem of our conceptualist logic of names is translated into a theorem of second-order monadic predicate logic, and hence, given the known consistency of the latter, that our conceptualist system is consistent.[18]

Metatheorem 2: If φ is a theorem of our present conceptualist logic, then $trs^*(\varphi)$ is a theorem of second-order monadic predicate logic. Therefore, our simple conceptualist logic is consistent.

We can also show, but will not do so here, that our simple conceptualist logic of names is decidable. The proof involves a translation function, trs', which translates each formula of second-order monadic predicate logic into a formula of our simple logic of names.[19] We then have the following metatheorem.

Metatheorem 3: If φ is a theorem of second-order monadic predicate logic, then $trs'(\varphi)$ is a theorem of our present conceptualist logic.

Finally, by metatheorem 1, to show that our conceptualist logic of names is decidable, we need only show that the formulas in which no complex names occur are decidable. To show this we first prove the following metatheorem by induction on these formulas.

Metatheorem 4: If φ is a formula of our conceptualist logic and no

[18] For the consistency of second-order monadic predicate logic, see Church 1956, p. 303.

[19] Because monadic predicate variables are in a one-to-one correspondence with nominal variables, we can take each predicate variable to have the form F_A, where A is the nominal variable corresponding to that predicate variable. The translation function, trs', is then defined as follows:

1. $trs'(x = y) = (x = y)$,
2. $trs'(F_A(x)) = (\exists yA)(x = y)$, where y is the first individual variable other than x,
3. $trs'(\neg\varphi) = \neg trs'(\varphi)$,
4. $trs'(\varphi \to \psi) = (trs'(\varphi) \to trs'(\psi))$,
5. $trs'((\forall x)\varphi) = (\forall x)trs'(\varphi)$,
6. $trs'((\forall F_A)\varphi) = (\forall A)trs'(\varphi)$.

complex names occur in φ, then $\vdash \varphi \leftrightarrow trs'(trs^*(\varphi))$.[20]

It follows, accordingly, that if φ is a formula of our conceptualist logic of names in which no complex names occur, then to decide whether or not φ is a theorem of this logic it suffices to decide whether or not $trs^*(\varphi)$ is a theorem of second-order monadic predicate logic. If the latter is not a theorem of second-order monadic predicate logic, then, by metatheorem 2, φ is not a theorem of our conceptualist logic; and if $trs^*(\varphi)$ is a theorem of second-order monadic predicate logic, then, by metatheorem 3, $trs'(trs^*(\varphi))$ is a theorem of our conceptualist logic, and therefore, by metatheorem 4, so is φ. Hence, by metatheorem 1, the decision problem for our conceptualist logic is reducible to that of second-order monadic predicate logic.

Metatheorem 5: Our present conceptualist logic is both consistent and decidable.

5 A Conceptualist Interpretation of Leśniewski's System

We now turn to a translation of Leśniewski's logic of names, as briefly described in section 2, into our conceptualist logic of names. We assume that the name variables a, b, c, d (with or with out numerical subscripts) of Leśniewski's logic are correlated one-to-one with the name variables A, B, C, D (with or without numerical subscripts) of our conceptualist logic, i.e., that

A is correlated with a,
B is correlated with b,
C is correlated with c,
etc.

Because the only atomic formulas of the system are of the form '$a\,\varepsilon\,b$', the following inductive definition of a translation function trs translates each formula of Leśniewski's logic into a formula of our conceptualist logic (with a replaced by A, b by B, etc.):

1. $trs(a\,\varepsilon\,b) = (\forall xA)(\forall yA)(x = y) \wedge (\exists xA)(\exists yB)(x = y)$,
2. $trs(\neg\varphi) = \neg trs(\varphi)$,
3. $trs(\varphi \to \psi) = [trs(\varphi) \to trs(\psi)]$,

[20] **Proof.** As noted, we prove this metatheorem by induction on the formulas of our conceptualist logic in which no complex names occur. The case for atomic formulas, which consist only of identities, is of course immediate; and for negations and conditionals, again the proof is immediate. Suppose the metatheorem holds for φ; then again it follows immediately that it holds for $(\forall A)\varphi$. The only interesting case is for $(\forall xA)\varphi$. But, by definition of trs^*, $trs^*((\forall xA)\varphi) = (\forall x)[F_A(x) \to trs^*(\varphi)]$, and therefore, by definition of trs', $trs'(trs^*((\forall xA)\varphi)) = (\forall x)[(\exists yA)(x = y) \to trs'(trs^*(\varphi))]$; and therefore, by the inductive hypothesis and (MP1), $\vdash (\forall xA)\varphi \leftrightarrow (\forall x)[(\exists yA)(x = y) \to trs'(trs^*(\varphi))]$, which completes our proof by induction. ■

4. $trs((\forall a)\varphi) = (\forall A)trs(\varphi)$.

In regard to the translation of an atomic formula of the form $a\,\varepsilon\,b$, note that the first conjunct,

$$(\forall xA)(\forall yA)(x = y),$$

of the translation is interpreted as saying that at most one thing is A, and therefore, because A is correlated with a, that at most one thing is a. The second conjunct,

$$(\exists xA)(\exists yB)(x = y),$$

on the other hand, says that some A is a B, and therefore, that some a is a b. The two conjuncts together are then equivalent to saying that exactly one thing is A, and hence a, and that thing is a B, i.e., a b, which is how Leśniewski understood '$a\ \varepsilon\ b$' as singular inclusion.

Note also that where φ is a logical axiom of the first-order logic of Leśniewski's system, then $trs(\varphi)$ is a theorem of our conceptualist logic.[21] Modus ponens and **(UG)** also preserve validity under trs. Accordingly, to show that this interpretation amounts to a reduction of Leśniewski's ontology, we need only prove that trs translates the axioms of Leśniewski's logic into a theorem of our present system. For example, both of the axioms, **(Compl)** and **(Conj)**, of Leśniewski's logic—one stipulating that every name has a complementary name, and the other that there is a name corresponding to the conjunction of singular inclusion in any two names—can be derived from the comprehension principle **(CP$_N$)** of our conceptualist logic.[22] The derivation of the translation of Leśniewski's principal axiom, which is the only one remaining, is relatively trivial, but long on details, and we will not go into those detail here.[23] In any case, we have the following metatheorem.

Metatheorem 6: If φ is a theorem of Leśniewski's (first-order) logic of names, then $trs(\varphi)$ is a theorem of our conceptualist simple logic of names.

Finally, let us turn to an explanation of the oddities of Leśniewski's logic of names, i.e., an explanation in terms of our translation of Leśniewski's logic into our simple logic of names. First, in regard to the seemingly implausible thesis,

$$(\exists a)(a \neq a),$$

of Leśniewski's logic, note that by Leśniewski's definition of identity (and hence of nonidentity) $(a \neq a)$ is really short for $\neg(a\ \varepsilon\ a \wedge a\ \varepsilon\ a)$, which is equivalent to $\neg(a\ \varepsilon\ a)$. On our conceptualist interpretation, this formula translates into

$$\neg[(\forall xA)(\forall yA)(x = y) \wedge (\exists xA)(\exists yA)(x = y)],$$

[21] In fact, **Axioms 1-3** are just the translations of the quantifier axioms assumed in the first-order theory for Leśniewski's ontology. By definition of trs, moreover, it is obvious that the translation of a tautologous formula is also a tautologous formula.

[22] See Cocchiarella 2001a for the details.

[23] Again, see Cocchiarella 2001a for those details.

which in effect says that it is not the case that exactly one thing is an A, a thesis that is provable in our conceptualist logic when A is taken as a necessarily vacuous common name, such as 'object that is not self-identical', which is symbolized as '$/(x \neq x)$', or more fully as '$Object/(x \neq x)$'. That is,

$$\begin{array}{c}\text{object that is not self-identical}\\ \downarrow \\ /(x \neq x)\end{array}$$

In other words, the translation of Leśniewski's thesis,

$$(\exists a)(a \neq a),$$

is equivalent in our conceptualist logic of names to

$$(\exists A)\neg[(\forall xA)(\forall yA)(x = y) \wedge (\exists xA)(\exists yA)(x = y)],$$

which is provable in this logic.

Note also that because $(a = b)$ in Leśniewski's system means $(a\ \varepsilon\ b \wedge b\ \varepsilon\ a)$, then the translation of $(a = b)$ into our conceptualist logic becomes

$$(\forall xA)(\forall yA)(x = y) \wedge (\exists xA)(\exists yB)(x = y) \wedge (\forall xB)(\forall yB)(x = y) \wedge (\exists xB)(\exists yA)(x = y),$$

which in effect says that exactly one thing is A and that thing is B, and that exactly one thing is B and that thing is A, a statement that is true when A and B are proper names, or unshared common names, of the same thing, and false otherwise, which is exactly how Leśniewski understood the situation.

Now the form of existential generalization that we found odd in Leśniewski's, namely,

$$\varphi(c/a) \rightarrow (\exists a)\varphi(a),$$

is translated into our conceptualist logic as:

$$\varphi(C/A) \rightarrow (\exists A)\varphi(A),$$

which, if C is free for A in φ, is provable in our simple logic of names, and yet, of course, from this it does not follow, as it does in Leśniewski's logic, that something is identical with nothing. Our earlier example of this oddity was the conditional

$$\neg(\exists b)(b = Pegasus) \rightarrow (\exists a)\neg(\exists b)(b = a).$$

Now what the antecedent $\neg(\exists b)(b = Pegasus)$ says under our conceptualist interpretation is that there is no name concept B such that B names exactly one thing and that thing is Pegasus, which of course is true, given that Pegasus does not exist. The consequent, $(\exists a)\neg(\exists b)(b = a)$, on the other hand, says that for some name concept A, there is no name concept B such that A names exactly one thing and that thing is a B. That statement is in fact is true for any name A that names nothing, e.g., where A is the complex common name 'object that is not self-identical'. In other words, although the above formula about Pegasus seems odd as a formula in first-order logic, what it means under our conceptualist translation is not odd at all, but quite natural.

In regard to the following thesis of Leśniewski's logic,

$$a \, \varepsilon \, b \to a \, \varepsilon \, a,$$

note that its translation into our conceptualist logic is,

$$(\forall xA)(\forall yA)(x = y) \wedge (\exists xA)(\exists yB)(x = y) \to (\forall xA)(\forall yA)(x = y) \wedge (\exists xA)(\exists yA)(x = y),$$

which says that if exactly one thing is an A and that thing is a B, then exactly one thing is an A and that thing is an A, which is unproblematic. Similarly, the translation, which we will avoid writing out in full here, of Leśniewski's seemingly odd transitivity thesis,

$$a \, \varepsilon \, b \wedge b \, \varepsilon \, c \to a \, \varepsilon \, c,$$

says that if exactly one thing is an A and that thing is a B and that if exactly one thing is a B—which therefore is the one thing that is an A—then exactly one thing is A and that thing is C, which again is easily seen to be valid in our conceptualist logic.

Finally, putting aside Leśniewski's definition of identity, it is noteworthy that although '$A = B$', unlike '$x = y$', is not a well-formed formula of our simple logic of names, nevertheless, it will be well-formed in our next section where we will extend the simple logic of names to include a logic of classes as many. This extension involves a transformation of names as parts of quantifier phrases to "singular" terms that can be substituted for object variables. In this extended framework, as we will see, when A and B are proper names, or unshared common names, of the same thing, then $A = B$ will be true independently of Leśniewski's definition of identity.

6 On the Pragmatic Uses of Proper and Common Names

The apparent oddities of Leśniewski's logic of names are the result of treating both proper and common names as if they were "singular terms," i.e., expressions that can be substituends of objectual variables and occur as arguments (subjects) of predicates. That, in any case, is how they are understood in Leśniewski's elementary ontology as an applied first-order logic (without identity). Of course, that is also how proper names, but not common names, are usually analyzed in modern logic, a practice we ourselves initially followed in our lecture on tense logic. But then, before the development of free logic where proper names that denote nothing are allowed, it was sometimes also the practice to transform proper names into monadic predicates. The proper name 'Socrates', for example, became the monadic predicate 'Socratizes', which was true of exactly one thing, and the name ‘Pegasus’ became ‘Pegasizes’, which was true of nothing. In this way, the statement that Pegasus does not exist could be analyzed as saying that nothing Pegasizes.[24]

Common names, on the other hand, have usually been analyzed as, or really transformed into, monadic predicates in modern logic, both before and after the development of free logic. Of course, in our general framework of conceptual realism there are complex monadic predicates that are constructed on the basis of both proper and common names. Thus, where A is a name, proper or common, then

$$[\lambda x(\exists yA)(x = y)]$$

is a monadic predicate, read as ‘x is an A’ when A is a common name, and as ‘x is A’ when A is a proper name.

Leśniewski’s logic of names is viewed as odd in modern logic, as we have said, because it takes common names to be more like proper names than like monadic predicates, and in particular it represents them the way that “singular terms” are represented in modern logic. On this view, if common names were to be put in the same syntactic category as proper names, then it should be by taking both as monadic predicates.

Now in our conceptualist logic, proper names and common names are in the same syntactic category, but it is not the category of monadic predicates, nor is it the category of “singular terms”. Proper and common names belong to a more general category of names, and as such they are taken as parts of quantifier phrases, i.e., phrases that stand for referential concepts. This is not at all like taking them as “singular terms,” the way they are in Leśniewski’s logic, though, as we will explain shortly, they can be transformed into “singular terms”, i.e., terms that can be substituends of objectual variables and occupy the argument positions of predicates.

The important point is that unlike the view of names in Leśniewski’s logic, the occurrence of names as parts of quantifier phrases, i.e., of referential expressions, is an essential component of how the nexus of predication is understood in conceptual realism. In other words, having a single category of names containing both proper and common names is a basic part of our theory of reference. This does not mean that we cannot distinguish a proper name from a common name in our logic. In particular, a proper name, when it introduced into an applied formal language, brings with it a meaning postulate to the effect that the name can be used to refer to at most one thing. If the language also contains tense and modal operators, then not only is it stipulated that a proper name can be used to refer to at most one thing, but that it must be the same thing at any time in each possible world in which that thing exists. Common names, on the other hand, are not introduced into an applied formal language with such a meaning postulate.

Now there are other uses of proper and common names as well. Both, for example, can be used in simple acts of naming, as when a parent teaches a child

[24] See, e,g„ Quine 1960, p. 179.

what a dog or a cat is by pointing to the animal and saying ‘dog’, or ‘cat’.[25] A simple act of naming is not an assertion and does not involve the exercise of either a predicable or a referential concept. Also, names, both proper and common, can be used in greetings, or in exclamations as when someone shouts ‘Wolf!’or ‘Fire!’, which again are not assertions and do not involve the exercise of a referential act. A common name such as ‘poison’ is also used as a label, which again is not a referential act. Nor are referential acts involved in the use of name labels that people wear at conferences. These kinds of uses of names, especially proper names or sortal common names, i.e., names that have identity criteria associated with their use, are conceptually prior to the referential use of names in sentences. But the analysis of these kinds of uses as well as the use of names in referential acts belongs to the discipline of pragmatics, and not that of semantics, which deals exclusively with denotation and truth conditions.[26]

7 Classes as Many as the Extensions of Names

Now in addition to these pragmatic uses of names there are also “denotative” uses of names as well, as when we speak of mankind, or humankind, by which we mean the totality, or entire group, of humans taken collectively—but not in the sense of a set or class as an abstract object.[27] Thus, we say that Socrates is a member of mankind, as well as that Socrates is a man. Also, instead of ‘mankind’, we can use the plural of ‘man’ and say that Socrates is one among men. These in fact are transformations of the name ‘man’ into an “objectual term,” i.e., an expression that can occur as an argument of predicates. But it is not a “singular term” in the sense that it denotes a single entity, e.g., a set or a class as an abstract object.

> Instead of using the phrase ‘singular term’, which suggests that we are dealing with a “single” entity, a better, or less misleading, phrase is ‘objectual term’, which we have used instead. An “object” such as mankind is not a “single” entity, but a plural object, i.e., a plurality taken collectively.

The transformation of ‘man’ into ‘mankind’, or ‘human’ into ‘humankind’, and ‘dog’ into ‘dogkind’, etc. is different from the nominalizing transformation of a predicate adjective, such as ‘human’, into an abstract noun—i.e., an abstract “singular term”—such as ‘humanity’, or into the gerundive phrase ‘being human’, or into the related infinitive and gerundive phrases ‘to be a man’ and ‘being a man’, all of which are represented in our logical syntax by a nominalization of the (complex) predicate phrase $[\lambda x(\exists y Man)(x = y)](\ )$.

[25] See, e.g., Geach 1980, p. 52.

[26] Pragmatics and semantics are two of the three principal parts of semiotics. The third is syntax.

[27] See, e.g., Sellars 1963, p. 253.

> The transformation of a predicate, e.g., 'is human', into an abstract noun, 'humanity', results in a genuine "singular term", i.e., a term that purports to denote a single object, albeit an abstract intensional one.
>
> But the transformation of 'man' into 'mankind', or 'dog' into 'dogkind', does not result in a "nominal" expression that purports to denote a single object; nor does it purport to denote an abstract intensional object.
>
> What such a noun as 'mankind', or 'dogkind', purports to denote is a plural object, namely, men, or dogs, taken collectively as a group—but not as a set or a class as a single object. The expressions 'mankind' and 'dogkind', are indeed "nominal" expressions, i.e., nouns, and therefore, logically, they should be represented as "objectual terms," but not as "singular" terms in the sense of nominal expressions that denote single objects. What they denote are pluralities, i.e., plural objects.

Now a plural object, such as a group of things, is what Bertrand Russell once called a class as many, as opposed to a class as one. Russell allowed that a class as many could consist of just a single object, as when a common name has just one object in its extension, in which case the class as many is the same as that one object. On the other hand, there is no class as many that is empty.[28]

There is more than one notion of a class, in other words, and in fact there is even more than one notion of a class in the sense of the iterative concept of a set, i.e., the concept of a set based on Cantor's power-set theorem. The iterative concept of a set can be developed, for example, either with an axiom of foundation or an axiom of *anti*-foundation.[29] But in neither case can there be a universal set, and yet there are set theories, such as Quine's **NF** and the related set theory **NFU**, in which there is a universal set. The notion of a universal class is part of the traditional notion of a class as the extension of a predicable concept (*Begriffsumfange*), and, as we have noted in lecture four, this was how classes were understood by Frege in his *Grundgetsetze.* Now our point here is that classes in all of these senses are single objects, not plural objects, i.e., they are each a class as one, a single abstract object.

It is not the notion of a class as one, i.e., as a single abstract object, that we are concerned with here, but the notion of a class as many, i.e., of a class as a plurality, or plural object. It is this notion that is implicitly understood as the extension of a common count noun, or what we have been calling a common name. In the development of our analysis of this notion, we will also take a class as many consisting of just one object as the extension of a nonvacuous proper name.

[28] See Russell 1903, §§69–70.

[29] For a development of set theory with an anti-foundation axiom, see Aczel 1988.

Membership in a class as many can be defined once names are allowed to be "nominalized" and occur as objectual terms. The definition is as follows:

$$x \in y =_{df} (\exists A)[y = A \wedge (\exists z A)(x = z)],$$

where A is a name variable or constant. Note that in this definition, A occurs as a "nominalized", objectual term in the conjunct '$y = A$' as well as part of the quantifier phrase in the second conjunct '$(\exists z A)(x = z)$'. An obvious theorem of the logic of classes as many, which we will develop in the next lecture, is the following,

$$x \in A \leftrightarrow (\exists z A)(x = z),$$

where '$x \in A$' can be read as 'x is an A', or 'x is one among the A', or 'x is a member of the class as many of A', or simply as 'x is a member of A'.

Now as understood by Russell, there are three important features of the notion of a class as many as the extension of a common name. These are:

> (1) first, that a vacuous common name, i.e., a common name that names nothing, has no extension, which is not the same as having an empty class as its extension. Thus, according to Russell, "there is no such thing as the null class, though there are null class-concepts," i.e., common-name concepts that have no extension.[30]
>
> (2) Secondly, the extension of a common name that names just one thing is just that one thing. In other words, unlike the singleton sets of set theory, which are not identical with their single member, the class that is the extension of a common name that names just one thing is none other than that one thing.
>
> (3) The second feature is related to our third, namely, that unlike sets, classes as the extensions of names are literally made up of their members so that when they have more than one member they are in some sense pluralities (*Vielheiten*), or "plural objects," and not things that can themselves be members of classes.
>
> Thus, according to Russell, "though terms [i.e., objects] may be said to belong to ... [a] class, the class [as a plurality] must not be treated as itself a single logical subject."[31]

It is a class as many that is the extension, or denotatum, of a common name, and, on our analysis, also of a proper name. On our analysis, the logic of classes as many is a direct and natural extension of the simple logic of names described earlier in this lecture. The idea is that names, both proper and common, can be transformed into, or "nominalized" as, objectual terms that can be

[30] Russell 1903, §69.

[31] Russell 1903, §70.

substituends of objectual variables and occur as arguments of predicates. When so transformed, what a name denotes is its extension, which in the case of a common name with more than one object in its extension is a plural object, which we will also call a group. The extension of a proper name, on the other hand, is the object, if any, that the name denotes as a "singular" term.

The resulting logic of classes as many is not entirely unlike the analysis given in Leśniewski's logic of names, where names occur only as objectual terms. In fact we can even formulate counterparts in this logic to certain of the oddities of Leśniewski's logic. But there is also a difference in that the counterparts of Leśniewski's problematic oddities are refutable, and the counterparts that are not refutable do not appear as odd but as natural consequences of an ontology with both single and plural objects.

References

[1] Aczel, Peter, 1988, *Non-Well-Founded Sets*, CSLI, Stanford.

[2] Church, Alonzo, 1956, *Introduction to Mathematical Logic*, Princeton University Press, Princeton.

[3] Cocchiarella, Nino B., 2001a, "A Conceptualist Interpretation of Leśniewski's Ontology," *History and Philosophy of Logic* 22: 29–43.

[4] Cocchiarella, Nino B., 2001b, "A Logical Reconstruction of Medieval Terminist Logic in Conceptual Realism," *Logical Analysis and History of Philosophy* 4: 35–72.

[5] Geach, Peter, 1980, *Reference and Generality*, 3rd. edition, Cornell University Press, Ithaca.

[6] Henry, D.P., *Medieval Logic and Metaphysics*, Hutchison University Library, London, 1972.

[7] Iwanuś, B. 1973, "On Leśniewski's Elementary Ontology," *Studia Logica* xxxi: 73-119. (Reprinted in Srzednicki *et al* 1984.)

[8] Küng G., and T. Canty, 1979, "Substitutional Quantification and Leśniewskian Quantifiers," *Theoria*, 36: 165–182.

[9] Lewjewski, Czesław, 1958, "On Leśniewski's ontology", *Ratio* 1: 150–176.

[10] Quine, Willard V.O., 1960, *Word and Object*, MIT Press, Cambridge.

[11] Rickey, V.F., 1977, "A Survey of Leśniewski's Logic," *Studia Logica*, 36: 407–426).

[12] Russell, Bertrand, 1903, *The Principles of Mathematics*, second edition, Norton & Co., N.Y., 1938.

[13] Russell, Bertrand, 1919, *Introduction to Mathematical Philosophy*, George Allen & Unwin, LTD., London.

[14] Schein, Barry, 1993, *Plurals and Events*, MIT Press, Cambridge.

[15] Sellars, Wilfrid F., 1963, "Grammar and Existence: A Preface to Ontology," in *Science, Perception and Reality*, London, Routledge & Kegan Paul.

[16] Slupecki, J., 1955, "S. Leśniewski's Calculus of Names," *Studia Logica,* vol III: 7-72. (Reprinted in Srzednicki *et al* 1984.): 150-176.

[17] J. Srzednicki, V.F. Rickey, and J. Czelakowski, eds., 1984, *Leśniewski's Systems, Ontology and Mereology*, Martinus Nijhoff Pub., The Hague.