6.2. Referring

6.2.0. Overview

Much of the work in analyzing predications lies in recognizing the individual terms that fill their blanks and in analyzing these individual terms further.

Glen Helman 25 Aug 2005

6.2.1. Individual terms

The chief examples of individual terms are proper names, for the central function of a proper name is to refer to the bearer of the name. But a proper name is not the only sort of expression that refers to an individual; the phrase *the first U. S. president* serves as well as the name *George Washington*. In general, descriptive phrases coupled with the definite article *the* at least purport to refer of individuals. These phrases are the ***definite descriptions*** discussed briefly in 1.3.4, and we have been counting them as individual terms. Still other examples of individual terms can be found in nouns and noun phrases modified by possessives—for example, *Mt. Vernon's most famous owner*. Indeed, expressions of this sort can generally be paraphrased by definite descriptions (such as *the most famous owner of Mt. Vernon*). Still examples of expressions that refer to individuals are demonstrative pronouns *this* and *that* and other pronouns whose references are determined by the context of use—such as *I*, *you*, and certain uses of third person pronouns. On the other hand, ***anaphoric pronouns***, pronouns that have other noun phrases as their antecedents, do not refer independently even though they play much the same grammatical role as expressions we do count as individual terms.

There is no traditional grammatical category or part of speech that includes individual terms but no other expressions. In particular, the class of nouns and noun phrases is too broad because it includes simple common nouns, such as *president*, as well as ***quantifier phrases***—such *no president*, *every president*, or *a president*. And neither common nouns nor quantifier phrases make the kind of reference that is required for an individual term.

The following table collects the examples we have just seen on both sides of the line between individual terms and other noun phrases:

Individual terms	*Noun phrases that are not individual terms*
proper names (e.g., *George Washington*)	common nouns (e.g., *president*)
definite descriptions (e.g., *the first U. S. president*)	quantifier phrases (e.g., *no president*, *every president*, *a president*)
noun phrases with possessive modifiers (e.g., *Mt. Vernon's most famous owner*)	
non-anaphoric pronouns (e.g., *this*, *you*)	

In a moment, we will look further at the reasons for drawing the line in this way; but one way of seeing the difference between individual terms and other nouns and noun phrases is to note that, while a proper name or a definite description provides a direct answer to the question *Which person, place, thing, or idea are you referring to?*, a common noun or quantifier phrase either provides no answer at all or, as in the case of *a president*, constitutes only an incomplete or evasive one.

Perhaps the most that can be done in general by way of defining the idea of an ***individual term*** is to give the following rough semantic description: an individual term is

an expression that refers, or purports to refer,
to a single object in a definite way

At any rate, this formula can be elaborated to explain the reasons for rejecting the noun phrases at the right of the table above.

The formula above is intended as a somewhat more precise statement of the idea that an individual term "names a person, place, thing or idea." It uses *object* in place of the list *person, place, thing, or idea* partly for compactness and partly because that list is incomplete. Indeed it would be hard to ever list all the kinds of things that might be referred to by individual terms. If the term ***object*** and other terms like ***entity***, ***individual***, and ***thing*** are used in a broad abstract sense, they can apply to anything that an individual term might refer to. In particular, in this sort of usage, these terms apply to people. The main force of the formula above then lies in the ideas of *referring to a single thing* and *referring in a definite way*.

The requirement that reference be to a single thing rules out most of noun phrases on the right of the table above. First of all, if a common noun can be said to refer at all, it refers not to a single thing but to a class, such as the class of all presidents. Now this class can be thought of as a single thing and can be referred to by the definite description just used—i.e., *the class of all presidents*—but the common noun *president* "refers" to this class in a different way. Common nouns are sometimes labeled ***general terms*** and distinguished from ***singular terms***, an alternative label for individual terms. The function of a general term is to indicate a general kind (e.g., dogs) from which individual things may be picked out rather than to pick out a single thing of that kind (e.g., Spot), as an individual term does. Thus the individual term *the first U. S. president* picks out an individual within the class indicated by the common noun *president*; and *the class of all presidents* picks out an individual within the class indicated by the common noun *class*. That is, a general term indicates a range of objects from which a particular object might be chosen while an individual term picks out a particular object. Although there is much that might be said about the role of general terms in deductive reasoning, we will never identify them as separate components in our analyses of logical form, and the word *term* without qualification will be used as an abbreviated alternative to *individual term*.

The remaining noun phrases at the right of the table are like individual terms in making use of a common noun's indication of a class of objects. However, they do not do this to pick out a single member of the class but instead to contribute to claims made about the class as a whole. The claims to which they contribute can all be described roughly as saying something about the number of members of a class that have or lack a certain property, and that is the reason for describing them as "quantifier" phrases. It's probably clear that the phrases *every president* and *no president*, even though they are grammatically singular, do not serve the function of picking out a single object. But that may be less clear in the case of *a president*.

Sentences containing quantifier phrases like *a president* and *some president* share with those containing definite descriptions, such as *the president*, the feature that they can be true because of a fact about a single object. For example, *The first U. S. president wore false teeth* and *A president wore false teeth* can be said to both be true because of a fact about Washington. The difference between the two sorts of expression can be seen by considering what might make such sentences false. If Washington had not worn false teeth, *The first U. S. president wore false teeth* would be false but *A president wore false teeth* might still be true. That's because the second could be true because of facts about many different presidents (in many different countries), so its truth is not tied to facts about any one of them. It would be made false only by a fact about all presidents. So the expression *a president* does not function to single out a particular president facts about whom will determine the truth of a sentence which has this expression as its subject. It merely marks the claim that there is at least one example of a president of whom the predicate is true. If the expression *a president* is thought of as referring at all, its reference is an indefinite one. This is one reason for adding the qualification *definite* to the formula for individual terms given above, but this qualification also serves as a reminder that the presence of a definite article marks an individual term while an indefinite article indicates a quantifier phrase.

The hedge *or purports to refer* used in the formula acknowledges the fact that not all individual terms actually succeed in picking out an individual as their reference. In spite of notorious exceptions like the name *Santa Claus*, proper names can usually be relied on to refer to something. But definite descriptions succeed in referring only when there is something that fits the description they offer and that does so without real competition. Mathematicians sometimes speak of these two requirements for a definite description to make a definite reference as ***existence*** and ***uniqueness***. Both must be met before a mathematician can speak of, say, "*the* solution" of a certain equation; there must be a solution (the solution must exist) and there must be no more than one (the solution must be unique). When a description does not have these two properties, a definite description employing it does not succeed in referring to an individual; if there is no solution or more than one, the phrase *the solution* does not refer.

At least this is so for the strictest and most explicit use of language. In most cases where a description is fulfilled by several entities, something in the context will distinguish one among them, and this one will be taken as the reference of the definite description. In such cases, the definite description functions as if the description it contains was more specific than its explicit statement suggests and the requirement of uniqueness really was satisfied. That is, we will understand, for example, *the college* as perhaps *the college (we all know and love)* and *the task* as perhaps *the task (at hand)*. The philosopher David Lewis (1941-2001) suggested that definite descriptions drew on a general contextual feature of ***salience***. One way of using this idea is to think of *the X* as *the (most salient) X*.

Of course, existence, too, may fail; there may be no entity at all that fulfills the description. As was noted in 1.3.4, we assume that every individual term has a ***reference value***. When the term succeeds in referring, this reference value is the individual the term refers to. When a term does not refer to anything, we say that the term is ***undefined*** and its reference value is an ***empty*** or ***nil*** value. It is natural to assume that there is just one nil value that is shared by all undefined terms. That is because reference values are *extensions* in the sense of 2.1.7 and are intended to capture only the object of reference, not all aspects of meaning. When an individual term has a defined reference, its reference value is the same no matter how the reference was made. This is why, in the example used in 2.1.7, the definite descriptions *the author of Poor Richard's Almanack* and *the inventor of the lightning rod* both have the same reference value. And, while undefined terms may refer in different ways, all are alike in what they refer to because none of them refers to anything.

6.2.2. Functors

Truth-functional connectives express truth-valued functions of truth values, and predicates express truth-valued functions of reference values. A third sort of function not only takes reference values as input but also issues them as output. We will refer to this sort of function as a ***reference function*** or, in contexts where we do not need a more general concept, simply as a ***function***. We will refer to expressions that are signs for these functions as ***functors*** and refer to the operation of applying a functor as ***function application***. We can speak of the result of a function application as a ***compound term***.

Functors are incomplete expressions that stand to individual terms as connectives stand to sentences, so we can extend the table of operations in 6.1.1 as follows:

operation	*input*	*output*
connective	sentence(s)	sentence
predicate	individual term(s)	sentence
functor	individual term(s)	individual term

Signs for mathematical functions provide examples of functors. The expression *7* + *5* can be analyzed as

Individual terms: *7* *5*
Functor: _ + _

But functors are not limited to mathematical vocabulary. Any individual term that contains one or more individual terms can be seen as the result of applying a functor to those component terms. Thus *the oldest child of Ann and Bill* can be analyzed as

Individual terms: *Ann* *Bill*
Functor: *the oldest child of* ___ *and* ___

And the more complex individual term *the location of the home of Ann's father's best friend* has the following analysis:

Individual term	*Ann*
Functors	___*'s father*
	______*'s best friend*
	the home of ____________
	the location of __________________

The notation of lambda abstraction was introduced in 6.1.4 with an example of a mathematical reference function, and that notation can be applied to any reference functions. Using it, the first two examples above could be given the analyses:

[λxy (x + y)] *7* *5*
[λxy (*the oldest child of* x *and* y)] *Ann* *Bill*

In the case of the third example, we need to use parentheses to show grouping

[λx (*the location of* x)] ([λx (*the home of* x)] ([λx (x*'s best friend*)] ([λx (x*'s father*)] *Ann*)))

And, in general, compound terms should be enclosed in parentheses when they fill a place of a functor or predicate.

In full symbolic notation, unanalyzed functors will be represented by lower case letters and will be written before the individual terms filling their places. Our English notation for a compound term

$$\zeta\tau_1 \ldots \tau_n$$

will be

ζ `of` τ_1, …, `'n` τ_n

which is in keeping with the usual way of reading a function application. When we need a general variable for functors we will use ζ, as has been done here, or sometimes ξ.

Using this symbolic and English notation, we can express the final analyses of the examples above as follows:

symbolic notation	*English notation*	*key*
psf	p `of` s `'n` f	[p: λxy (x + y); f: *5*; s: *7*]
oab	o `of` a `'n` b	[o: λxy (*oldest child of* x *and* y); a: *Ann*; b: *Bill*]
l(h(d(fa)))	l `of` h `of` d `of` f `of` a	[d: λx (x*'s best friend*); f: λx (x*'s father*); d: λx (*the home of* x); l: λx (*the location of* x); a: *Ann*]

The symbolic notation for functors that is used here is different from the most common notation for function applications. Here are some examples for comparison

common mathematical notation	*symbolic notation used here*	*English notation*
f(a)	fa	f of a
f(a, b)	fab	f of a ’n b
f(g(a))	f(ga)	f of g of a
f(a, g(b))	fa(gb)	f of a ’n g of b
f(g(a), b)	f(ga)b	f of (g of a) ’n b
f(g(a, b))	f(gab)	f of (g of a ’n b)

The notation used here is a common one in logic and is designed to minimize parentheses and commas. The general rule for interpreting it is this: (i) after a predicate—i.e., after a capital letter—each unparenthesized letter and each parenthetical unit occupies one place of the predicate and (ii) within a parenthetical unit the first letter is a functor and each following unparenthesized letter and each parenthetical unit occupies one place of this functor.

While the English notation for compound terms provides a way of reading logical forms, the last two examples above show that it does not enable us to completely avoid parentheses, for the English notation for these two different forms would be the same without the parentheses. Because the letters used to represent functors and non-logical predicates do not have a fixed number of places associated with them, parentheses can be needed to show where a compound term ends. Although there are verbal ways of dealing with this problem, we will simply use parentheses when they are necessary to avoid ambiguity. Of course, parentheses, like other punctuation, can be reflected in speech and it is natural to mark the difference between f of (g of a) ’n b and f of (g of a ’n b), respectively, by varying the speed with which they are spoken in ways that might be indicated by “f of g-of-a ’n b” and “f of g of a-’n-b”.

When analyzing sentences, functors are uncovered by analyzing terms as compound. Here is an example:

The cat on the mat was asleep and the dog that had chased it was, too

the cat on the mat was asleep ∧ *the dog that had chased the cat on the mat was asleep*

[λx (x *was asleep*)] *the cat on the mat* ∧ [λx (x *was asleep*)] *the dog that had chased the cat on the mat*

S (*the cat on the mat*) ∧ S (*the dog that had chased the cat on the mat*)

S ([λx (*the cat on* x)] *the mat*) ∧ S ([λx (*the dog that had chased* x)] *the cat on the mat*)

S(cm) ∧ S (d (*the cat on the mat*))

Scm ∧ S (d ([λx (*the cat on* x)] *the mat*))

S(cm) ∧ S(d(cm))

both S fits c of m and S fits d of c of m

[S: λx (x *was asleep*); c: λx (*the cat on* x); d: λx (*the dog that had chased* x); m: *the mat*]

It will be easier to make a full analysis of a sentence if you choose the largest individual terms possible when analyzing an atomic sentence or compound term. Otherwise, part of the logical form will end up being obscured by an abbreviation unless you go on to analyze the body of an abstract. That sort of problem would have arisen in this example if we had analyzed the first conjunct as

the cat on the mat was asleep

[λx (*the cat on* x *was asleep*)] *the mat*

While it is possible to recover from such choices by analyzing the bodies of abstracts, some care is needed in the choice of variables so each variable ends up having the correct antecedent, and we will not go on to consider how this may be done. (It would not merely create difficulties but actually be wrong to choose *the cat* as a component individual term of *the cat on the mat was asleep*; we will discuss this issue in 6.2.3.)

In the presence of functors, the potential for undefined terms increases considerably. Even if *the cat on the mat* has a non-nil reference value, *the cat on the refrigerator* may not—to say nothing of *the cat on the house of Ann’s father’s best friend* or *the cat on 6*. That is, functors accept a large variety of inputs and can be expected to issue output with undefined reference for some of them. This problem can be reduced somewhat by limiting functors to input of certain sorts. That is usually done by assigning individual terms to various ***types*** and allowing only individual terms of certain types to serve as inputs to a given functor. For example, the functor λxy (x + y) might be restricted to numerical input. However, it is not easy to eliminate all undefined terms by use of types, and we will not introduce into our analyses the complications needed to use types.

Glen Helman 25 Aug 2005

6.2.3. Examples and problems

We will begin with a couple of extended but straightforward examples.

If Dan is the winner and Portugal is the place he would most like to visit, he will visit there before long

Dan is the winner and Portugal is the place he would most like to visit → *Dan will visit Portugal before long*

(*Dan is the winner* ∧ *Portugal is the place Dan would most like to visit*) → *Dan will visit Portugal before long*

(*Dan* is *the winner* ∧ *Portugal* is *the place Dan would most like to visit*) → *Dan* will visit *Portugal* before long

(*Dan* = *the winner* ∧ *Portugal* = *the place* *Dan* *would most like to visit*) → [λxy (x *will visit* y *before long*)] *Dan* *Portugal*

(d = n ∧ p = [λx (*the place* x *would most like to visit*)] *Dan*) → Vdp

(d = n ∧ p = ld) → Vdp

`if both` d `is` n `and` p `is` l `of` d `then` V `fits` d `'n` p

[V: λxy (x *will visit* y *before long*); l: λx (*the place* x *would most like to visit*); d: *Dan*; n: *the winner*; p: *Portugal*]

Al won't sign the contract Barb's lawyer made out without speaking to his lawyer

¬ *Al will sign the contract Barb's lawyer made out without speaking to his lawyer*

¬ (*Al will sign the contract Barb's lawyer made out* ∧ ¬ *Al will speak to his lawyer*)

¬ (*Al* *will sign* *the contract Barb's lawyer made out* ∧ ¬ *Al* *will speak to* *Al's lawyer*)

¬ ([λxy (x *will sign* y)] *Al* *the contract Barb's lawyer made out* ∧ ¬ [λxy (x *will speak to* y)] *Al* *Al's lawyer*)

¬ (S a (*the contract* *Barb's lawyer* *made out*) ∧ ¬ P a (*Al's lawyer*))

¬ (S a ([λx (*the contract* x *made out*)] *Barb's lawyer*) ∧ ¬ P a ([λx (x*'s lawyer*)] *Al*))

¬ (S a (c ([λx (x*'s lawyer*)] *Barb*)) ∧ ¬ Pa(la))

¬ (Sa(c(lb)) ∧ ¬ Pa(la))

`not both` S `fits` a `'n` c `of` l `of` b `and not` P `fits` a `'n` l `of` a

[P: λxy (x *will speak to* y); S: λxy (x *will sign* y); c: λx (*the contract* x *made out*); l: λx (x*'s lawyer*); a: *Al*; b: *Barb*]

Apart from telling the difference between individual terms and quantifier phrases, the chief problem that is likely to arise in analyzing predications is making sure that you have identified the whole of a definite description you locate in an atomic sentence. What you are most likely to miss are modifiers, usually prepositional phrases or relative clauses, that follow the main common noun of the definite description. For example, although *the place* might be an individual term in its own right in other cases, here is it only part of the term *the place Dan would most like to visit*. Similarly, *the contract* is only the beginning of the individual term *the contract Barb's lawyer made out*. In both of the these cases, the rest of the definite description is a relative clause with a suppressed relative pronoun; that is, they might have been stated more fully as *the place **that** Dan would most like to visit* and *the contract **that** Barb's lawyer made out*, respectively. Notice that relative clauses are not individual terms that are components of larger individual terms (in the way *Barb* is a component of *Barb's lawyer*), for they do not appear as components of the final analysis at all.

Let's look at these issues in more detail by returning to the example at the end of 6.2.2.

The cat on the mat was asleep and the dog that had chased it was, too

The cat on the mat was asleep
∧ *the dog that had chased the cat on the mat was asleep*

It would be wrong to remove only *the cat* from the first conjunct because the sentence-with-a-blank ____ *on the mat was asleep* does not express a predicate any more than does the sentence-with-a-blank ____ *tle were in the meadow* that would be obtained by removing *the cat* from *The cattle were in the meadow*. Of course *cat* is a separate word in the first case, but the two words *the* and *cat* do not form a separable unit because *the* properly applies to the common-noun-plus-modifier *cat on the mat*. This means that the phrase *the cat*, although often an individual term, is not an individual term in this sentence. Similarly, the sentence-with-a-blank ____ *that had chased the cat on the mat* does not express a predicate and it would be wrong to remove only *the dog* from the second conjunct.

There are some cases where a prepositional phrase or relative clause following a common noun should not be counted as part of a definite description. Some prepositional phrases can modify both nouns and verbs, and a prepositional phrase following a noun within a grammatical predicate might be understood to modify either it or the main verb. The sentence *The dog chased the cat on the mat* is ambiguous in this way since the mat might be understood to be either the location of the chase or the location of the cat, who might have been chased elsewhere. This sort of ambiguity can be clarified by converting the prepositional phrase into a relative clause, which can only modify a noun; if this transformation—e.g.,

The dog chased the cat that was on the mat

—preserves meaning, then the prepositional phrase is part of the definite description. On the other hand, since anaphoric pronouns cannot accept modifiers, replacing a possible noun phrase by a pronoun will show the result of taking a prepositional phrase to modify the verb. This can be done by moving the noun phrase to the front of the sentence, joining it to the remaining sentence-with-a-blank by the phrase *is such that*, and filling the blank with an appropriate pronoun (*he*, *she*, or *it*). In this example, that would give us

The cat is such that the dog chased it on the mat

So, if the *on the mat* should be taken to modify *cat* or *chased* depending on whether the first or second of the displayed sentences best captures the meaning of the original. Of course, when a potentially ambiguous sentence is taken out of context, it may not be clear which of two alternatives does this; in such a case, either analysis is legitimate.

Not all relative clauses contribute to determining reference. Those that do are ***restrictive*** clauses while others are ***non-restrictive***. Non-restrictive clauses cannot use the word *that* and, when punctuated, are marked off by commas. Restrictive clauses are not marked off by commas in standard English punctuation and may use *that* (but are not limited to this relative pronoun), and they can in some cases be expressed without a relative pronoun. The sorts of relative clause are most sharply distinguished when more than one of these differences is exhibited. For example, the relative clause *The cat that the dog had chased was asleep* or *The cat the dog had chased was asleep* is clearly restrictive while the one in *The cat, who the dog had* *chased, was asleep* is clearly non-restrictive. This means that the relative clause in the first is part of the definite description *the cat that the dog had chased*. The relative clause in the second would instead be analyzed as a separate conjunct:

The cat, who the dog had chased, was asleep

the dog had chased the cat ∧ *the cat was asleep*

[λxy (x *had chased* y)] *the dog* *the cat* ∧ [λx (x *was asleep*)] *the cat*

Adc ∧ Sc

both A fits d ʼn c and S fits c

[A: λxy (x *had chased* y); S: λx (x *was asleep*); c: *the cat*; d: *the dog*]

When the relative pronouns *who* or *which* are used without commas, both interpretations are possible; and *The cat who the dog had chased was asleep* might be given either analysis depending on what was meant. The differences in meaning between these two interpretations are subtle. When the relative clause is restrictive, the property of being chased by the dog is used to narrow down the class of cats to the point where *the* is appropriate. When the sentence is true on the second analysis, this narrowing is possible; but it might be appropriate to use it if a particular cat is sufficiently salient in any case that adding this qualification to the definite description would lead someone to wonder what other cats were around.

Another indication of the difference between the two sorts or relative clause is that the non-restrictive clause can modify a proper name—as in *Puff, who the dog had chased, was asleep*. And, since neither prepositional phrases nor restrictive relative clauses can modify a proper name, putting a proper name in the blank or blanks of a sentence-with-blanks can show whether it really expresses a predicate. For example, *Puff on the mat was asleep* and *Puff that the dog had chased was asleep* are both ungrammatical. (You may need to shift between *which* and *who* in order for a sentence that applies a non-restrictive relative clause to a proper name to sound grammatical since these relative pronouns are appropriate for different names and for uses of the same name to refer to different things: when *Old Betsy* names a cow, *who* can be used, but *which* is used when *Old Betsy* names a car.)

6.2.s. Summary

In addition to proper names, the individual terms include definite descriptions and various non-anaphoric pronouns. They do not include certain other noun phrases, quantifier phrases in particular. We will speak of the “person, place, thing, or idea” referred to by an individual term by using such words as *object*, *entity*, *individual*, and *thing*, understanding these to apply to anything that might be named. Common nouns are also not individual terms. Indeed, they may be labeled general terms to distinguish their function of indicating a class of objects from the function of individual terms, also called singular terms, which is to refer to a single individual in a definite way. The word *term* will often be used as shorthand for *individual term*. Definite descriptions refer by way of a description only when there is one and only one object satisfying the description; that is, an object satisfying the description must exist and be unique. This is often so for ordinary definite descriptions only when the description is understood to contain an implicit qualification that the object be the most salient one satisfying other parts of the description. An individual term is understood to always have a reference value; when the term is undefined, its reference value is the empty or nil value.

A functor is an operation that takes one or more individual terms as input and yields an individual term as output. Just like other operations, it expresses a reference function, which yields reference values when applied to reference values. Although a reference function is a particular sort of function we will use that term primarily for reference functions. The operation of combining a functor with input is application, and the individual term that is the output is a compound term. For any functor, there will almost always be some terms for which the application of the functor yields an undefined term. Although this problem can be reduced by limiting the input of functors to objects of certain types, we will not include this complication in our account of logical forms.

It can be difficult to recognize the individual terms that fill the places of a predicate or a functor. It is important in include in a definite description all the modifiers that are part of it. Some of these may be prepositional phrases or relative clauses which follow the common noun. In some cases, a prepositional phrase in this position might either be part of a definite or modify a verb; but such an ambiguity cannot arise with relative clauses so a prepositional phrase can be made into a relative clause in order to test what it modifies. Relative clauses must therefore be part of the definite description when they are restrictive; on the other hand, non-restrictive clauses (the sort set off by commas) are analyzed using conjunction.

Glen Helman 25 Aug 2005

6.2.x. Exercise questions

1. Analyze each of the following sentences in as much detail as possible.

- **a.** *Reagan's vice president was the 41st president*.
- **b.** *Tom found a fly in his soup and he called the waiter*.
- **c.** *Tom found the book everyone had talked to him about and he bought a copy of it*.
- **d.** *Wabash College is located in Crawfordsville, which is the seat of Montgomery County*.
- **e.** *Sue and Tom set the date of their wedding but didn't decide on its location*.

2. Synthesize idiomatic English sentences that express the propositions associated with the logical forms below by the intensional interpretations that follow them.

- **a.** (Sab ∧ ¬ Sa(fc)) → ¬ b = fc
  [S: λxy (x *has spoken to* y); f: λx (x*'s father*); a: *Ann*; b: *Bill*; c: *Carol*]
- **b.** (B(fa)(mb) ∨ S(ma)(fb)) → Cab
  [B: λxy (x *is a brother of* y); C: λxy (x *and* y *are cross-cousins*); S: λxy (x *is a sister of* y); f: λx (x*'s father*); m: λx (x*'s mother*); a: *Ann*; b: *Bill*]
- **c.** Pab(m(sb)(sc)) ∧ Pac(m(sb)(sc))
  [P: λxyz (x *persuaded* y *to accept* z); m: λxy (*the best compromise between* x *and* y); s: λx (x *'s proposal*); a: *Ann*; b: *Bill*; c: *Carol*]

Glen Helman 25 Aug 2005

6.2.xa. Exercise answers

1. **a.** *Reagan's vice president was the 41st president.*
Reagan's vice president = the 41st president
[λx (x*'s vice president*)] *Reagan* = [λx (*the* x*th president*)] *41*

vr = pf
v of r is p of f

[p: λx (*the* x*th president*); v: λx (x *'s vice president*); f: *41*; r: *Reagan*]

b. *Tom found a fly in his soup and he called the waiter*
Tom found a fly in his soup ∧ *Tom called the waiter*
Tom found a fly in Tom's soup ∧ *Tom called the waiter*
[λxy (x *found a fly in* y)] *Tom Tom's soup* ∧ [λxy (x *called* y)] *Tom the waiter*
Ft(*Tom's soup*) ∧ Ctr
Ft([λx (x*'s soup*)] *Tom*) ∧ Ctr

Ft(st) ∧ Ctr
both F fits t 'n s of t and C fits t 'n r

[C: λxy (x *called* y); F: λxy (x *found a fly in* y)]; s: λx (x*'s soup*); r: *the waiter*; t: *Tom*]

c. *Tom found the book everyone had talked to him about and he bought a copy of it*
Tom found the book everyone had talked to him about ∧ Tom bought a copy of the book everyone had talked to him about
Tom found the book everyone had talked to Tom about ∧ *Tom bought a copy of the book everyone had talked to Tom about*
[λxy (x *found* y)] *Tom the book everyone had talked to Tom about* ∧ [λxy (x *bought a copy of* y)] *Tom the book everyone had talked to Tom about*
Ft(the book everyone had talked to *Tom* about) ∧ Bt(the book everyone had talked to *Tom* about)
Ft([λx (*the book everyone had talked to* x *about*)] *Tom*) ∧ Bt([λx (*the book everyone had talked to* x *about*)] *Tom*)

Ft(bt) ∧ Bt(bt)
both F fits t 'n b of t and B fits t 'n b of t

[B: λxy (x *bought a copy of* y); F: λxy (x *found* y); b: λx (*the book everyone had talked to* x *about*); t: *Tom*]

d. *Wabash College is located in Crawfordsville, which is the seat of Montgomery County*
Wabash College is located in Crawfordsville ∧ *Crawfordsville is the seat of Montgomery County*
[λxy (x *is located in* y)] *Wabash College Crawfordsville* ∧ *Crawfordsville* = *the seat of Montgomery County*
Lbc ∧ c = [λx (*the seat of* x)] *Montgomery County*

Lbc ∧ c = sm
both L fits b 'n c and c is s of m

[L: λxy (x *is located in* y); s: λx (*the seat of* x); b: *Wabash*; c: *Crawfordsville*; m: *Montgomery County*]

e. *Sue and Tom set the date of their wedding but didn't decide on its location*
Sue and Tom set the date of their wedding ∧ *Sue and Tom didn't decide on the location of their wedding*
Sue and Tom set the date of Sue and Tom's wedding ∧ ¬ *Sue and Tom decided on the location of Sue and Tom's wedding*
[λxyz (x *and* y *set* z)] *Sue Tom the date of Sue and Tom's wedding* ∧ ¬ [λxyz (x *and* y *decided on* z)] *Sue Tom the location of Sue and Tom's wedding*
Sst(*the date of Sue and Tom's wedding*) ∧ ¬ Dst(*the location of Sue and Tom's wedding*)
Sst([λx (*the date of* x)] *Sue and Tom's wedding*) ∧ ¬ Dst([λx (*the location of* x)] *Sue and Tom's wedding*)
Sst(d(*Sue and Tom's wedding*)) ∧ ¬ Dst(l(*Sue and Tom's wedding*))
Sst(d([λxy (x *and* y*'s wedding*)] *Sue Tom*)) ∧ ¬ Dst(l([λxy (x *and* y*'s wedding*)] *Sue Tom*))

Sst(d(wst)) ∧ ¬ Dst(l(wst))
both S fits s, t, 'n d of (w of s 'n t) and not D fits s, t, 'n l of (w of s 'n t)

[D: λxyz (x *and* y *decided on* z); S: λxyz (x *and* y *set* z); d: λx (*the date of* x); l: λx (*the location of* x); w: λxy (x *and* y

’s wedding); s: Sue; t: Tom]

2. a. ([λxy (x has spoken to y)] Ann Bill ∧ ¬ [λxy (x has spoken to y)] Ann ([λx (x’s father)] Carol)) → ¬ Bill = [λx (x’s father)] Carol

(Ann has spoken to Bill ∧ ¬ [λxy (x has spoken to y)] Ann Carol’s father) → ¬ Bill = Carol’s father

(Ann has spoken to Bill ∧ ¬ Ann has spoken to Carol’s father) → ¬ Bill is Carol’s father

(Ann has spoken to Bill ∧ Ann hasn’t spoken to Carol’s father) → Bill isn’t Carol’s father

Ann has spoken to Bill but not to Carol’s father → Bill isn’t Carol’s father

If Ann has spoken to Bill but not to Carol’s father, then Bill isn’t Carol’s father

b. (B([λx (x’s father)] Ann)([λx (x’s mother)] Bill) ∨ S([λx (x’s mother)] Ann)([λx (x’s father)] Bill)) → [λxy (x and y are cross-cousins)] Ann Bill

([λxy (x is a brother of y)] Ann’s father Bill’s mother ∨ [λxy (x is a sister of y)] Ann’s mother Bill’s father) → Ann and Bill are cross-cousins

(Ann’s father is a brother of Bill’s mother ∨ Ann’s mother is a sister of Bill’s father) → Ann and Bill are cross-cousins

Ann’s father is a brother of Bill’s mother or Ann’s mother is a sister of Bill’s father → Ann and Bill are cross-cousins

If Ann’s father is a brother of Bill’s mother or Ann’s mother is a sister of Bill’s father, then Ann and Bill are cross-cousins

c. Pab(m([λx (x’s proposal)] Bill)([λx (x’s proposal)] Carol))
∧ Pac(m([λx (x’s proposal)] Bill)([λx (x’s proposal)] Carol))

Pab([λxy (the best compromise between x and y)] Bill’s proposal Carol’s proposal)
∧ Pac([λxy (the best compromise between x and y)] Bill’s proposal Carol’s proposal)

[λxyz (x persuaded y to accept z)] Ann Bill the best compromise between Bill’s proposal and Carol’s proposal
∧ [λxyz (x persuaded y to accept z)] Ann Carol the best compromise between Bill’s proposal and Carol’s proposal

Ann persuaded Bill to accept the best compromise between his and Carol’s proposals
∧ Ann persuaded Carol to accept the best compromise between Bill’s proposal and hers

Ann persuaded each of Bill and Carol to accept the best compromise between their proposals