7.4.1. Multiple generality

Frege suggested we understand the interaction of several quantifier phrases in a single sentence by thinking of them as operations that are applied to the sentence one at a time so that a sentence might already contain one quantifier phrase when another is applied to it. In such a case, the second phrase applied to the sentence would have the first in its scope, and the ambiguities of quantifiers in relation to one another could be understood as ambiguities regarding relative scope.

However, not all differences in scope make for differences in meaning and we will look first at some that do not. Consider, for example, the sentence *Everyone read each application*. We can analyze this as we have analyzed earlier examples—except that we will analyze quantifier phrases twice. If we take them in the order in which they appear in the sentence, the analysis will go as follows:

Everyone read each application
Everyone is such that (he or she read each application)
(∀x: x *is a person*) (x *read each application*)
(∀x: x *is a person*) (*each application is such that (*x *read it)*)
(∀x: x *is a person*) (∀y: y *is an application*) (x *read* y)

(∀x: Px) (∀y: Ay) Rxy
∀x (Px → ∀y (Ay → Rxy))

[A: λx (x *is an application*); P: λx (x *is a person*); R: λxy (x *read* y)]

Before discussing the significance of this analysis, there is a technical point to be made. Notice that we chose a new variable when analyzing the second quantifier phrase. At that stage in the analysis, we were analyzing the formula x *read each application*. When we put this in expanded form, we had *each application is such that (*x *read it)*. In order to express this symbolically, we replaced the pronoun *it* with a variable. Using the variable x again would have gotten us the wrong antecedent because, while the abstract λy (x *read* y) expresses the property of being read by x, the abstract λx (x *read* x) expresses the property something has when it read itself. In more technical terms, the formula x *read each application* has a free occurrence of the variable x, so our symbolic version of this formula should also. And, while that it true of the formula (∀y: y *is an application*) x *read* y, the expression (∀x: x *is an application*) x *read* x has no free variables. Instead of being a formula that says something about an unspecified thing x, it is a complete sentence that says *every application read itself*. In short, when analyzing a formula that already contains a variable, you should choose a new variable for any quantifier phrase you analyze. In the example above, the variable y was chosen to analyze the quantifier phrase in the formula x *read each application*, but any variable other than x could have been used.

If we apply subject-predicate expansion to the above sentence while leaving it in English, we get something like *Every person is such that each application is such that he or she read it*. We could state this also as *Every person is such that each application is such that the former read the latter*, and the phrases *the former* and *the latter*, in this use of them, play much the same role here as the distinct variables x and y play in our symbolic analysis. When more than two independent references are needed, we can resort to *the first*, *the second*, etc. Like *the former* and *the latter*, these are definite descriptions in form but they describe what they refer to by way of earlier expressions in the sentence (as shorter forms of expressions like *the first thing referred to*). Consequently, they function like anaphoric pronouns in picking up their references from earlier material in the sentence. Other definite descriptions can be used in this way, too, and the sentence in expanded form might have been rendered as *Every person is such that each application is such that the person read the application*, where, for example, *the person* amounts to *the aforementioned person*.

Now suppose we had instead analyzed this sentence first as a generalization concerning applications. That would have led us to the following analysis:

Everyone read each application
Each application is such that (everyone it)
(∀y: y *is an application*) (*everyone read* y)
(∀y: y *is an application*) (*everyone is such that (he or she read* y*)*)
(∀y: y *is an application*) (∀x: x *is a person*) (x *read* y)

$$(\forall y: Ay)\ (\forall x: Px)\ Rxy$$
$$\forall y\ (Ay \rightarrow \forall x\ (Px \rightarrow Rxy))$$

The variable y is chosen before x here only in order to facilitate comparison with the first analysis. The form we end up with is equivalent to the one we derived earlier, as can be seen by comparing subject-predicate expansions that correspond to the two analyses:

Every person is such that he or she read each application
Each application is such that every person read it

Either way, we state a double generalization, one that generalizes on the two dimensions of people and applications.

These equivalent forms are an example of a general principle we can state as follows (adapting the notation introduced in 7.3.2 to speak of either restricted or unrestricted quantifiers):

$$(\forall x\ldots)\ (\forall y{-}{-}{-})\ \phi \Leftrightarrow (\forall y{-}{-}{-})\ (\forall x\ldots)\ \phi.$$

Here ϕ can be any formula though it will normally contained free occurrences of both x and y. Dashes as well dots have been used in the notation for quantifiers to allow for the possibility that the quantifiers for the two variables have different restrictions (which must be brought along when their order is reversed) and to allow also for the possibility that the quantifier for one variable is restricted and the other unrestricted. To insure that no variables become unbound in the interchange, we must require that any restriction on a quantifier not contain free occurrences of the variable bound by the other quantifier. (An example where that restriction would not be met will be discussed below.)

Any generalization of the form displayed above can be described as a generalization over pairs. We can express it this way in English by using a subject-predicate expansion with a paired subject.

Every person and application are such that the former read the latter

It would not be difficult to extend our symbolic notation to get the same effect by using quantifiers that apply to many-place predicates. That is, the generalization at hand can be understood to say that the extension of the predicate λxy (x *is a person and* y *is an application*) is included in the extension of λxy (x *read* y), and we could capture this interpretation symbolically by an operation comparable to ∀ that applied to 2-place predicates. Other examples might lead us to consider quantifiers applying to predicates of 3 or more places. However, there are costs that attend the use of further notation, and we will not pay them here. We will continue to analyze double, triple, and other multiple generalizations by analyzing quantifier phrases in sequence. Still it will help to remember that when we find a sequence of universal quantifiers (with or without attached restrictions) the effect is the same as having a single quantifier over pairs, triples, or longer sequences.

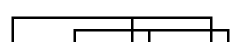
There is one type of case where our approach to such sentences will make analyses a little awkward. Consider the sentence *Not every employer and employee get along*. This is the denial of a generalization over pairs, so we can expect it to be analyzed as the negation of a sentence that begins with a pair of universal quantifiers. However, this case is unlike the one we considered above in that the two universal quantifications are not restricted in independent ways. The generalization is not over all pairs consisting of someone who is a employer and someone who is an employee but rather over pairs consisting of someone who is an employer and someone who is *his or her* employee. That is, the universal quantification is restricted to pairs whose members stand in the employer-employee relation. So we must ask how to represent such a restriction when we use two separate quantifiers. The answer is that we need not restrict the first, outer, quantifier at all, but we must restrict the second, inner, quantifier with reference to the outer one. This is illustrated in the following analysis:

$\neg$ *every employer and employee get along*
$\neg \forall x$ x *and every employee of* x *get along*
$\neg \forall x$ *every employee of* x *is such that (*x *and he or she get along)*
$\neg \forall x$ ($\forall y$: y *is an employee of* x) x *and* y *get along*
$\neg \forall x$ ($\forall y$: x *employs* y) Gxy

$$\neg \forall x\ (\forall y\text{: } Exy)\ Gxy$$

[E: λxy (x *employs* y); G: λxy (x *and* y *get along*)]

(The formula y *is an employee of* x has been restated as x *employs* y to make it easier to compare this example with the next one.) Notice the pattern of binding in this form.


$$\neg \forall x\ (\forall y\text{: } Exy)\ Gxy$$

We cannot simply reverse the expressions $\forall x$ and ($\forall y$: Exy) (as we did with the quantifiers in the earlier example) because the variable x in the restricting predicate of the second would be moved outside the scope of $\forall x$ and would no longer be bound.


$$\neg\ (\forall y\text{: } Exy) \forall x\ Gxy$$

On the other hand, if we were to analyze the two quantifiers in the other order we would get the following:

$\neg$ *every employer and employee get along*
$\neg \forall y$ *every employer of* y *and* y *get along*
$\neg \forall y$ *every employer of* y *is such that (he or she and* y *get along)*
$\neg \forall y$ ($\forall x$: x *is an employer of* y) x *and* y *get along*
$\neg \forall y$ ($\forall x$: x *employs* y) x *and* y *get along*

$$\neg \forall y\ (\forall x\text{: } Exy)\ Gxy$$

Again the first quantifier in the analysis is unrestricted and the second is restricted in a way that refers back to it. This asymmetry is the compensation we must pay for using an asymmetric notation to represent an essentially symmetric claim. The asymmetry is mitigated if we use unrestricted quantification, for then we have the following two symbolic forms:

$$\neg\, \forall x\, \forall y\, (Exy \to Gxy)$$
$$\neg\, \forall y\, \forall x\, (Exy \to Gxy)$$

Here the only difference is in the order of the expressions ∀x and ∀y, and the predicate E can be seen to restrict both of them together.

Glen Helman 06 Nov 2004