Semantics I, UT Spring 2016. From Montague Grammar to DRT

Hans Kamp

University of Stuttgart
University of Texas, Austin

April 11, 2016

1 Making Sense of Donkey Pronouns

When translating English sentences into Predicate Logic we ran into the problem presented by sentences like (1).

(1)

a. If a bear is cornered, it attacks.

b. If a bear is shot, there is a party.

c. If a farmer owns a donkey, he beats it.

d. Every farmer who owns a donkey beats it.

e. Every farmer who owns a donkey is in trouble.

f. If Pedro owns a donkey he beats it.

Each of these sentences has one or more indefinites. And these indefinites cannot be translated by existential quantifiers. For instance, neither (2.a) nor (2.b) is an adequate translation of (1.a).

(2)

a. $(\exists x)(bear(x)\ \&\ cornered(x)) \rightarrow attacks(x)$

b. $(\exists x)((bear(x)\ \&\ cornered(x)) \rightarrow attacks(x))$

c. $(\forall x)((bear(x)\ \&\ cornered(x)) \rightarrow attacks(x))$

d. $(\exists x)(bear(x)\ \&\ shot(x)) \rightarrow (\exists y)party(y)$

In (2.a) the last occurrence of x isn't bound because the scope of the existential quantifier is restricted to the antecedent of the conditional. In (2.b) there is no binding problem, but the truth conditions are all wrong. (This formula is almost devoid of substantive content; it is true so long as there is something that isn't a cornered bear. I am not a cornered bear, and that is enough to make the formula true. But obviously that has nothing to do with what (1.a) says.)

There is a way of avoiding these two problems, we have seen, by translating the indefinite as a universal quantifier with wide scope, as in (2.c). But it is at the very least surprising that such a subterfuge should be needed. Note that if it wasn't for the anaphoric connection between *a bear* and the pronoun *it*, translating the indefinite by an existential quantifier would have done well enough. For instance, (2.d) seems an adequate translation of (1.b).

(1.b) and (1.c) are other sentences that illustrate the same problem – of indefinites in the antecedents of conditionals with anaphoric pronouns in the consequents of those conditionals. These are the examples presented in Geach (1962 (Third revised edition: 1980), which led to the nickname *donkey problem* for the problem such configurations present, that of *donkey sentences* for those in which it manifests itself ((1.a) no less than (1.b) or (1.c)) and *donkey pronoun* for the pronouns that are part of the donkey problem. (Interestingly, the term *donkey indefinite* never made it into semantic parlance, an indication of the view, held by Geach and others, that the problem and its cure had to do with the pronouns and not so much with their indefinite antecedents.) (1.d) differs from (1.c) in the same way that (1.b) differs from (1.a): the absence of an anaphoric pronoun makes it possible to translate the indefinite in the manner suggested by the rule of thumb that indefinites should be translated by existential quantifiers whose scope reflects the syntactic position of the indefinite.

Geach and others focused on donkey sentences as the locus of the donkey problem, presumably because of a general preoccupation with the syntax and semantics of single sentences, which was prevalent at the time of Geach (1962 (Third revised edition: 1980) and which also is an implicit presupposition of Montague Grammar. But the problem also arises, and perhaps more prominently and frequently, within discourse, as for instance in (3).

(3) a. Pedro owns a donkey. He beats it.

b. Unintentionally we cornered a bear. It attacked.

c. John found a cell phone on his desk. Someone had left it there by mistake.

d. A man and a woman entered the pub. She was quite elegantly dressed, but he was in jeans, heavy boots and a lumberjack shirt.

e. I had an idea. It is about how to make a good pet out of a raccoon. You give it ...

f. Someone must have been here. I can smell her perfume.

Such 'donkey discourses' suggest that (i) the indefinites in their first sentences have the function of introducing an individual into the discourse, and (ii) that the pronouns in the second sentences then get their interpretation in the 'discourse contexts' that those first sentences establish. But what can we make of this suggestion?

The problem is that the trans-sentential connection between indefinite antecedent and pronoun affects the meaning of the (multi-sentence) discourse as a whole. So something along the following lines would seem needed: Given a discourse consisting of two sentences S_1 and S_2:

(i) compute the logical form L_1 of S_1 (or the semantic value of S_1 in some model M determined by L_1),
(ii) identify the context C_1 established by S_1,
(iii) compute the logical form L_2 of S_2 (or semantic value in M), while using C_1 for the interpretation of any pronoun whose anaphoric antecedent is contained in S_1 (where the logical forms might be computed in the manner discussed in the preceding part of the course)

It isn't clear from this description, however, that this will give us what we want. For it isn't clear how the semantic *connection* between S_1 and S_2 is captured by L_1 and L_2. To take just one of the examples in (3): the second sentence of (3.a) speaks about the same donkey as the first sentence. This is an essential part of what is conveyed by the two-sentence discourse (3) as a whole, and any proper way of identifying the semantics of the discourse should capture the contribution this makes. In other words: we need a logical form (or semantic value) for the two sentences together which accounts for this sameness.

How can we make sure that the role played by C1 in the resolution of the anaphoric pronoun *it* in S_2 is not only reflected in L_2 but makes its impact on the way in which L_1 and L_2 can be combined to give the semantics of all of (3)? Here is a proposal:

Structure logical forms in such way that

(a) they themselves can play the part of discourse contexts; and
(b) set up the construction of L_2 in such a way that it gets to be *incorporated* into L_1, and in such a way that the resulting structure – call it $L_{1,2}$ – correctly identifies the semantics of the discourse consisting of S_1 and S_2 (and thus qualifies as its logical form).

Note well, it should not be taken for granted that logical forms can be given a structure that enables them to play the part of discourse contexts (in addition to identifying the semantic content of what the part of the discourse to which they have been assigned as logical forms). Nor is it obvious that logical form construction can be specified in a way that leads to 'updating' the given logical form (the one that serves as discourse context for the current sentence) to a new logical form that also incorporates the semantic content of the current sentence.

But also note that from a certain perspective – that of the hearer of a succession of sentences or the reader of a text – it is quite plausible that all this ought to be possible. It seems a reasonable hypothesis that after having processed the first sentence of a two-sentence discourse the hearer/reader should have extracted from the sentence a representation that both identifies its content and can serve discourse context for what he will have to interpret next. A single structure that can do both these jobs – identify the content of what has been processed and serve as context for what is going to be processed next – would seem an economic solution to a task – that of language interpretation – where efficiency is of the utmost importance.[1]

We will refer to the principle that a single type of structure can do both jobs – that of identifying content and that of serving as discourse context – as the *Principle of the Unity of Content and Context*. And we will proceed from this point onwards on the assumption that an account of the semantics of natural languages can be given in a form that verifies this principle.

Evidently, setting up semantic theory in such a way presupposes that we adopt a logic form approach as opposed to a semantic value approach. But we have seen that within the setting of model-theoretic semantics that we have been talking about so far there isn't much to choose between these two approaches: the logical forms that we have been talking about determine semantic values in models, and even someone who prefers the semantic value approach will find it hard to see how to implement his program without de facto assigning logical forms to the expressions of the natural language (fragment) under consideration.

It is also important, however, that we do not throw the baby out with the bathwater. The logical forms we have been considering up to now (formulas of PC, terms of LC) have the capacity of defining semantic values for the expressions with which the theory associates them as logical forms because they belong to formalisms that come with their own model theories. This is a requirement that the new logical forms, which are suited as discourse contexts as well as content identifiers, should satisfy as well: they too should belong to a formalism with its own syntax *and* model theory.

A semantic theory set up along such lines can still be seen as one which treats natural languages as abstract systems and provides a compositional semantics for them, but does so while fully abstracting from issues of lan-

[1]Admittedly, such efficiency considerations cannot be more than suggestive pointers. Language interpretation is a very complex process, and without a clear understanding of all the factors involved and all the ways they interact, it is not really possible to be sure that what looks like a simpler and more efficient architecture of one of the modules involved will result in greater efficiency over-all.

guage use. And in fact, this is how some have interpreted semantic theories with this general architecture. But the original motivation behind the version of this approach that we will adopt from now on was the persuasion that a semantic theory capable of accounting for discourse effects like those in (3) must be a theory of language *interpretation* and not just a theory of meaning for languages as abstract systems. According to this view meaning and interpretation cannot really be separated.

When a semantic theory comes with such a cognitive commitment – the commitment to having to say something of substance about the way language users capture the meaning of the utterances and texts that they hear or read – it is sticking out its neck a good deal more than a theory that 'merely' aims to make correct predictions about entailment relations between sentences. And there is a serious risk that it will be sticking out its neck too far, either because its cognitive implications turn out to be wrong (even when the predications it makes about entailments are right) or, more damaging yet, because it is on closer inspection unclear what exactly the cognitive claims are to which it claims to be committed, so that the question of whether these commitments are right or wrong cannot even be properly raised. This is a general problem for approaches to semantics that carry cognitive implications. It is a problem about which we will have little to say here. But it is important not to lose site of the fact that it is there.

2 Discourse Representation Theory

The approach just outlined is known as *Discourse Representation Theory* or, more briefly, *DRT*. As is true also for other 'dynamic' approaches to natural language meaning that have now been around for some decades – the first implementations of DRT go back to the early eighties – the name 'Discourse Representation Theory/DRT' has acquired a certain ambivalence between a general approach to natural language semantics and particular implementations of that approach. Here we will use the term in the first sense and identify particular implementations with additional terminology. One respect in which DRT implementations vary is in the logical form formalisms they employ. All the formalisms used have certain basic properties in common with each other, which justifies having a general term that covers them all.

So we will refer to all these as *DRS languages*. Here 'DRS' is short for *Discourse Representation Structure*. Discourse Representation Structures are the logical forms that DRT-based accounts of meaning assign to well-formed expressions of the natural language fragments they deal with. DRT implementations may differ (among other things) in the DRSs they make use of as logical forms.

DRS languages are like formalisms such as PC or LC in that they come with their own syntax and model-theoretic semantics. As we will soon see, the DRS languages that are used in different DRT implementations can differ substantially, reflecting importantly different views about semantic analysis. Among other things, DRS languages vary greatly in their expressive power. Using different DRS languages is one way in which DRT implementations may differ, but it is by no means the only way. Implementations can also differ (of course) in that they apply to different natural language fragments – these may be different fragments from the same language (e.g. English) but also fragments of different languages. And, crucially, implementations may differ in how they define the 'syntax-semantics interface'. Even when two implementations describe the same natural language fragment, assume the same syntax for the that fragment and use the same DRS language and end up assigning the same DRSs to the same sentences and texts from the fragment, they may still differ in how they define the procedure that leads from the syntactic structures of sentences to the DRSs that they assign to them as logical forms. In particular, DRT implementations differ in that some of them compute DRSs form syntactic sentence structures 'top down' while others do these computations 'bottom up. The distinction between top down and bottom up DRS construction algorithms will be centrally important in what we are going to do.

One reason why the difference between op down and bottom up algorithms is of such crucial importance is that most of the work that is involved in the development of a DRS implementation goes into the formulation of this component. The DRS construction algorithm of a DRS implementation is its formal and conceptual core, where the ultimate decisions are made which DRSs are assigned to which sentences and texts. Moreover, for those that associate cognitive significance with DRT, a construction algorithm should correctly reflect certain aspects of the actual interpretation processes that take place in the mind of a human interpreter.

The original motivation for DRT came from an investigation of problems of tense and aspect. More specifically it started as an attempt to capture the semantic differences between the French Simple Past (its *Passé Simple*) and its *Imparfait*. (Having two such tense forms is something that French shares with other Romance languages.) We will presently turn to the use of DRT for dealing with this distinction and other problems in the realm of Tense and Aspect. But first a few preliminary words about the DRT account of donkey discourses and donkey sentences.

Consider once more the discourse (3.a):

(3.a) Pedro owns a donkey. He beats it.

As a first introduction to DRT we show how this donkey discourse is treated in DRT's original formulation (or ts first explicit implementation, in the terminology we have been using above) (Kamp (1981)Kamp & Reyle (1993)).

(4) is the DRS for the first sentence of (3.a), in the 'box'-notation for DRSs that has been widely used since the beginnings of DRT and that will be mostly used here too..

(4)

$x \quad y$
$Pedro(x) \quad donkey(y)$ $owns(x, y)$

Like all DRSs, (4) consists of two components, its *Universe* and its *Condition set*. A DRS Universe is a set whose members are *discourse referents* (also *drefs* for short). Drefs are representations of entities. The Condition Set of a DRS consists of *DRS conditions*. In (4) these are all *atomic* conditions, consisting of a predicate of the given DRS language L_{DRT} and drefs to occupy its argument positions. More specifically, *donkey* is a 1-place predicate and *owns* a 2-place predicate. The condition $donkey(y)$ expresses that the individual represented by y has the property of being a donkey and $owns(x, y)$ that the individual represented by x stands in the ownership relation to the individual represented by y. (In the condition $Pedro(x)$ the proper name *Pedro* is formally treated as a 1-place predicate. '$Pedro(x)$' can be read as 'x stands for the referent of the name *Pedro*', or more explicitly, 'x stands for the individual that the name *Pedro* is used to refer to in the utterance of (3.a) of which (4) is a partial representation.' There is a good deal more to be said about the interpretation of proper names, and much of it will have to wait until much later. The present ad-hoc-ish treatment will have to do for now.)

As noted, the DRS language L_{DRT}, to which the DRS (4) belongs as one of its well-formed expressions comes (like any other DRS language) with its own syntax and model theory. We already observed that DRSs consist of a Universe and a Condition Set. We will leave matters of DRS syntax at that for now. But a few things need to be said right now about the model theory of DRS languages. For the comparatively simple L_{DRT} we can make do with correspondingly simple models: models for a language L of Predicate Logic that contains predicate constants for each of the predicate words and proper names of the chosen natural language fragment (one that we will not explicitly define, but that includes all the sample sentences and discourses discussed in this section). Recall that such models are pairs $<U, I>$, where U is a non-empty set and I a function that maps each of the non-logical constants of L to a suitable extension. (For instance, $I(donkey)$ will be a subset of U and $I(owns)$ a set of ordered pairs of elements of U. (We assume that $I(Pedro)$ is a singleton subset of U consisting of the one individual in the model that is the referent of the given use of the name *Pedro*.)

Let M be such a model. What does it mean for a DRS such as (4) to be true in M? For simple DRSs like (4) the informal idea is a simple and intuitive one: (4) is true in M if it can be seen as representing some part of M, consisting of individuals that can be regarded as the 'real referents' that are represented by the 'discourse referents' x and y and that verifies, via this correspondence between drefs in the Universe of the DRS and elements of M, the conditions in the Condition Set of the DRS. In other words, (4) counts as true in M iff U contains individuals a and b (that can be made to correspond to x and y respectively) such that (i) a is the unique member of the singleton set $I(Pedro)$, (ii) b is a member of $I(donkey)$ and (iii) the pair $<a, b>$ is a member of $I(owns)$.

More generally, for simple DRSs like (4), *truth in* a model for PC can be defined as follows:

(5) (Truth definition for Simple DRSs)

A DRS $K = <U_K, CS_K>$ *is true in* a model M iff there is a function f from U_K into U_M such that for each condition $P(x_1, .., x_n)$ in CS_K $< f(x_1), .., f(x_n)> \in I_M(P)$.

The functions f of this definition, which map DRS Universes into the Universes of models, are called *embedding functions*. If f is an embedding function from the Universe of a DRS K into the Universe of a model M and $P(x_1, .., x_n)$ a condition in CS_K such that $f(x_1), .., f(x_n)> \in I_M(P)$, then we say that f *verifies* the condition $P(x_1, .., x_n)$ (in M). If f verifies all conditions from CS_K in M then f is said to *verify K in M*.

In the form in which the truth definition (5) is stated, it will only work for *simple DRSs*, in which all conditions are atomic. Below we will also have to deal with non-atomic DRS conditions. At that point the intuitive idea that the truth of a DRS in a model amounts to it representing some part of that model – as a 'picture' of some of its individuals and their properties and relations – will have to be modified and diluted. But the basic intuition, according to which DRS record, in a sort of pictorial way, part of the information in the model or models described by the sentences and discourses of which they are the logical forms, will still hold in spirit.

The truth definition for DRSs explicates the sense in which a DRS can capture the content of the sentence or discourse to which it is assigned as logical form. But DRSs can at the same time serve as *contexts* for the interpretation of the sentences or discourse segments that follow upon the sentences or discourse segments to which they have been assigned as logical forms. Thus (4) can serve as discourse context in the interpretation of the second sentence of (3.a), and it can do this in the following way. The second sentence of (3.a), 'He beats it.', contains the pronouns *he* and *it* that on the natural interpretation of (3.a) will be anaphoric to the phrases *Pedro* and *a donkey* occurring in the first sentence. According to DRT such anaphoric interpretations of pronouns can be explained as a link between a discourse referent for the anaphoric pronoun and the discourse referent that has already been introduced (into the Universe of the DRS that serves as discourse context in the interpretation of the sentence to which the pronoun belongs) as representative for the phrase that the interpretation identifies as the pronoun's anaphoric antecedent. Thus, in the case of (3.a), the interpretation of the second sentence will involve introducing drefs for the pronouns *he* and *it* – let they be u and v – and linking them with the drefs x and y that represent the anaphoric antecedents —em Pedro and *a donkey* of *he* and *it* in the DRS (4) that at this point is playing the part of discourse context. We represent the links between u and x and between v and y in the form of equations. So the DRS for the second sentence becomes:

(6)
$$\begin{array}{|c|}\hline u \quad v \\ \hline u = x \quad v = y \\ beats(u, v) \\ \hline \end{array}$$

One difference between (4) and (6) is that (6) contains *unbound* drefs – drefs that occur in some condition of the Condition Set of the DRS, but not in its Universe. (In (6) this is true of both x ands y.) To DRSs with unbound drefs the truth definition in (5) is not applicable because an embedding function f, whose domain is the Universe of the DRS, will not be defined for its unbound drefs. So the question whether f verifies a condition containing an unbound dref won't be defined. In the case at hand: if truth is defined as in (5), then (6) fails to have well-defined truth conditions. But note that this is not so for the DRS in (7) that we obtain by *merging* the DRSs (4) and (4). This is once more a DRS in which all drefs are bound:

(7)
$$\begin{array}{|c|}\hline x \quad y \quad u \quad v \\ \hline Pedro(x) \quad donkey(y) \\ owns(x, y) \\ u = x \quad v = y \\ beats(u, v) \\ \hline \end{array}$$

(In general the *merge* of two DRSs K and K' is the DRS whose Universe is the union of the Universes of K and K' and whose Condition set is the union of their Condition Sets.)
Note that the truth definition in (5) predicts the intuitively correct truth conditions for (6): (6) is true in M iff there is a function f that maps the drefs x, y, u and v to individuals a, b, c and d from U_M such that a is the unique Pedro in M, b is a donkey in M, a and b stand in the ownership relation of M, $c = a$, $d = b$ and c and d (that is, in view of these identities, a and b) also stand in the 'beats' relation. This is just the content that (3.a) expresses.

The DRS (7) is the logical form for the discourse (3.a) as a whole. The way we have obtained it is paradigmatic for the construction of multi-sentence discourses: construct a DRS for the first sentence, then construct a DRS for the second sentence using the DRS for the first sentence as discourse context and then merge the new DRS with the old one. And then, for discourses of more than two sentences, repeat: construct a DRS for the third sentence using the DRS for the first two sentences as discourse context, and merge the new DRS with this discourse context in order to obtain the DRS for the first three sentences. And so on.

This is only a rough sketch of how a DRS for the two sentences of (3.a) can be constructed. All the details of how syntactic trees for the first and the second sentence are converted into DRSs are still missing. As a matter of fact, one major change that has taken place within the DRT-based approach since its beginnings in 1980 is the way DRSs are constructed from syntactic sentence trees. Below we present instances of the original construction method, which was tailor-made for the problem of donkey pronouns that is illustrated by the examples in (3) and (1). But first a sketch of the DRS construction of a donkey *sentence*. We focus on (1.f), the 'conditional version' of the donkey discourse in (3.a).

Common between (3.a) and (1.f) are the anaphoric connections between the pronouns *he* and *it* and their antecedents *Pedro* and *a donkey*. The difference is that in (1.f) the anaphoric antecedents belong to the same sentence as the pronouns that are anaphoric to them. However – and this makes the case of (1.f) look more similar to that of (3.a) – the pronouns in (1.f) belong to a different *part* of the sentence than their antecedents: the pronouns are in the consequent of the conditional expressed by (1.f) (its main clause) whereas their antecedents belong to the antecedent of the conditional, realized as its *if*-clause). In virtue of this the anaphoric relations in (1.f) can be explained along lines that closely resemble those we followed in accounting for the anaphoric links in (3.a): the part of (1.f) containing the anaphoric antecedents gets assigned a DRS very much like that assigned to the first sentence of (3.a), the construction of the DRS for the part that contains the pronouns can make use of this DRS for the first part as discourse context, and because of this the links between the pronouns and their antecedents can be established and justified in the same way as before. The only difference is that this time the DRS for the second part and the DRS for the first part must be made to stand in a certain relation that reflects the logical relation between the *if*-clause and the main clause of (1.f), that of being the antecedent and consequent of a conditional.

(8) shows the final result of this construction, in which the double barreled arrow $\Rightarrow$ is used to symbolize the conditional relationship between the *if*-clause DRS and the main clause DRS.

(8)

$$\begin{array}{|c|}\hline \\ \hline \begin{array}{|c|}\hline x \quad y \\ \hline Pedro(x) \quad donkey(y) \\ owns(x,y) \\ \hline \end{array} \Rightarrow \begin{array}{|c|}\hline u \quad v \\ \hline u = x \quad v = y \\ beats(u,v) \\ \hline \end{array} \\ \hline \end{array}$$

3 DRS Construction

The construction methods that have been used in DRT fall into two main categories: *top down* and *bottom up*. Top down algorithms 'break' sentences 'down' along the hierarchical structure imposed by their syntax, constructing the DRS bit by bit as the algorithm moves from the top of the syntactic tree (the highest S node) down to its leaves. Bottom-up algorithms start from the leaves and work their way up to the top. On the face of it bottom-up algorithms are more in the spirit of compositional semantics, as we find it in Montague Grammar. DRT-based analyses that make use of bottom up algorithms are somewhat easier to compare with analyses cast in other compositional frameworks. That has been seen as one reason for preferring them. A more important and substantive reason, and the one why we will adopt the bottom up method here, has to do with the need to be able to deal with *presuppositional phenomena* within the DRT-based approach; but this is a matter that we won't be in a position to broach until much later; until then you will have to take my word for it that this is a good reason for adopting the bottom up approach.[2]

[2] Claims that the top down approach fails to account for the compositionality of natural language semantics often seem to be based on an insufficient understanding on how DRT treatments that make use of this method work in detail. But since we have independent

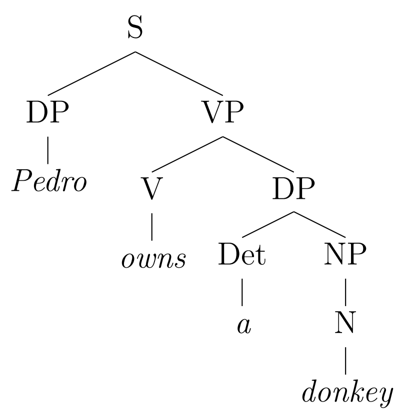
Barring presuppositions, however, the top down method is capable of accounting for many aspects of sentence-internal pronominal anaphora that cannot be accounted for by the bottom up method. Because the top down method was designed to deal with some of these aspects, and more particularly with cases of donkey anaphora, we will illustrate how the method works for (3.a) and (1.f), although we will abandon the method after that in favor of the bottom up approach that treats pronouns as triggers of 'identification presuppositions', which come with a set of constraints on their 'resolution'. We will return to the problems of donkey anaphora, including those presented by the examples in (**??**), when a general account of presupposition triggering and presupposition resolution has been put in place.

To give an impression how the original version of DRT deals with the problems presented by donkey pronouns we go through its treatment of the donkey discourse in (3.a) and the donkey sentence in (1.f). We start with the donkey discourse (3.a).

(3.a) Pedro owns a donkey. He beats it.

We begin by constructing a DRS for the first sentence. The syntactic tree from which we will compute the semantics for the sentence does not involve QR. (However, as indicated at the end of Part I, we will in due time go back to LFs in which DPs have undergone QR.) Our input tree is shown in (9).

(9)

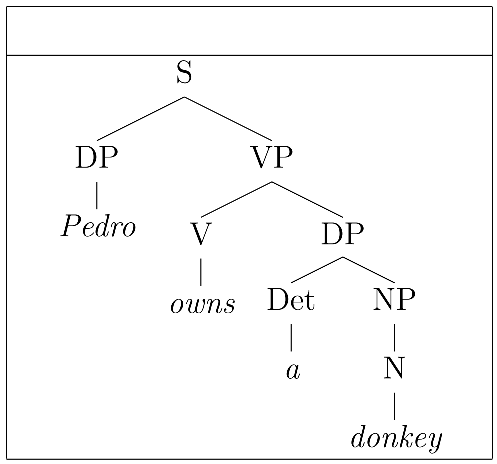


The initial step to the construction of the DRS for the first sentence of

---

reasons for adopting a bottom up approach, these ill=considered methodological disputes need not concern us.

(9) consists in placing its syntactic structure, as a 'reducible DRS condition', in the Condition Set of a DRS. In this case, where we are dealing with the first sentence of a discourse, this DRS is a new one, which is empty but for this one condition.

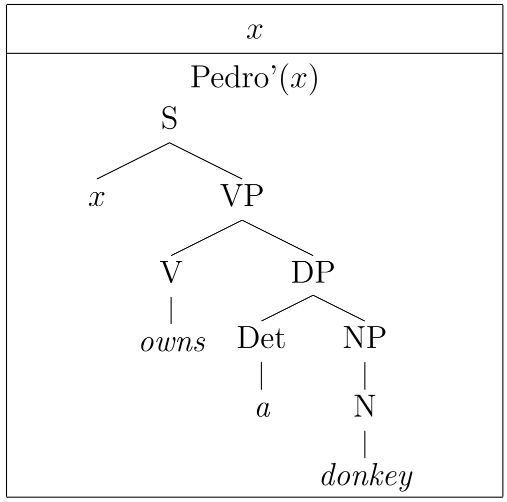
(10) 

The first construction step to be performed on the unprocessed DRS condition in (10) decomposes the S-node 'semantically' into the contribution that is made to the meaning of the sentence by the subject DP and the contribution made by the VP. The syntactic configuration consisting of the S-node and its two daughters is one of several in which one of the daughters plays the part of argument to the predicate that is contributed by the other. In all such cases the argument constituent contributes a dref to the representation of its sister – something that is implemented by inserting the dref into the position of the sister representation that was occupied by the argument phrase. The dref represents the entity or entities that the argument phrase refers to or quantifiers over. What this referent is, or what kind of quantification is involved, depends on the form of the argument phrase.
In addition to putting the chosen dref into the position previously occupied by the argument DP we also place it in the Universe of the DRS.

The dref we have chosen in order to carry these operations is x. But the choice is arbitrary. What matters only is that each time a dref is introduced into a DRS, it is a 'free' one, symbol that hash' yet been used in the DRS construction for the given sentence or discourse.

The result of the described operations is shown in (11)

(11)



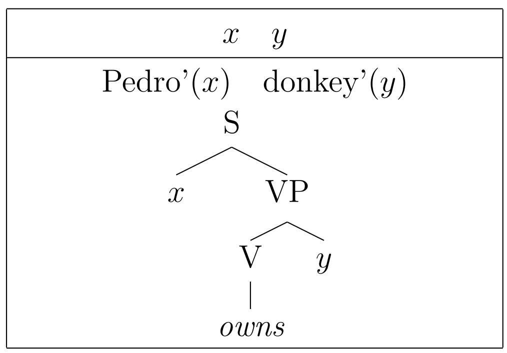
The next step (and only other step in the construction of the DRS for the first sentence of (3.a)) deals with the configuration of the VP node and its two daughters V and DP. Here we have once again a combination of an argument (the DP) and a predicate of which it is an argument (the V); and the general treatment – choose a new dref to represent what the DP refers to or quantifies over, insert that dref into the argument position occupied by the argument phrase in the syntactic structure, represent the descriptive information of the argument phrase (by a DRS condition that is also inserted into the Condition Set), and place the dref in a DRS Universe – is as it was for the first construction step. (This time the dref we have chosen is y, but again the choice is arbitrary, except that the new dref must be different from x.)

However, the DP we are dealing with now is of a different type than the subject DP we dealt with in our previous construction step: it is an *indefinite description*, not a proper name (or, for that matter, a definite DP of any kind). The treatment of indefinite DPs in DRT (and in the independently developed *File Change Semantics* of Irene Heim (see Heim (1982,1988))) was a novelty at the time when these accounts were proposed and it still sets these approaches apart from other ways of doing formal semantics, in which indefinites are treated as existential quantifier phrases (and thus in particular from the treatment we adopted in Part I).

The treatment of indefinites exemplified by the operations just described assimilates indefinite DPs to proper names and other definite DPs in that for all these DP types the newly chosen dref is introduced into a DRS universe that is either at or above the level at which the construction rule is being applied. (For quantifying DPs like *every farmer* this is not so. For details consider early literature on DRT, such as Kamp & Reyle (1993).) The implications of this cannot be appreciated at this point. But one aspect will become clear when we construct the semantics for the second sentence of (3.a).

The result of carrying out the second step is as in (12).

(12)



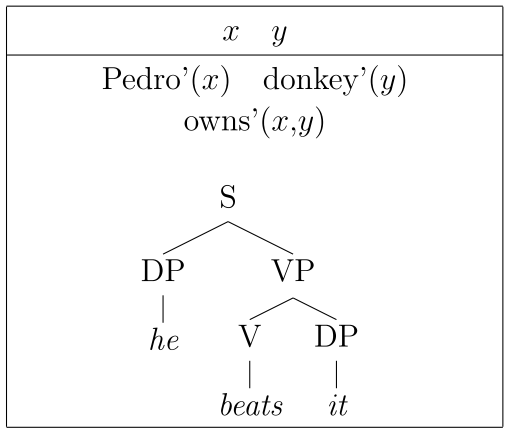
This is the complete DRS for the first sentence of (3.a). We can simplify this DRS by inserting the lexical semantic representation of the verb *owns*, which we take to be the 2-place predicate *owns′*, and then throwing away the remaining syntactic skeleton, which at this point is no longer needed. This notational simplification gets us to (13).

(13)

x y
Pedro'(x) donkey'(y)
owns'(x,y)

To process the second sentence of (3.a) we start by inserting its syntactic structure as a new reducible condition into the Condition Set of the DRS in (13). (This is slightly different from what we showed earlier, when the DRS for the second sentence was first constructed separately and then merged with the context DRS; but it will be easy to see that the final result comes out the same.) Inserting the syntactic structure for the second sentence as a reducible condition into (13) gives us (14).

(14)



Once again the first construction rule to be applied deals with the node combination S-DP-VP. This time the subject DP is a pronoun. We assume it as given that the pronoun – *he* – is being used anaphorically in this example (rather than 'deictically'; deictic uses of pronouns are those where the pronoun is used to defer to an individual that is accessible to the audience via some non-verbal channel, as when A whispers to B: 'He shouldn't be in here.' stealthily pointing at the person A has in mind; when pronouns occur in texts, the default assumption is that they are not used deictically, but anaphorically). We already noted that in DRT anaphoric relations between pronouns and their antecedents are analyzed as identity relations between the drefs representing them. More precisely, the possible antecedents for a given pronoun are those drefs in the DRS that is being constructed that are *accessible* from the position of the pronoun. In the present case the accessible drefs are those in the Universe of the DRS (13) that has been constructed for the first sentence of (3) and that is now being extended with the information contributed by the second sentence. (the DRS that at this point plays the role of discourse context). This Universe contains two drefs, viz. x and y. So treating the pronoun as anaphoric means that it must be either linked to x or to y. Once more, processing the DP involves the introduction of a new dref to represent its 'referent'; we choose u for this purpose. Interpreting the pronoun as anaphoric to x or y can now be expressed via the identities '$u = x$' or '$u = y$ and that is how we will represent these respective options.

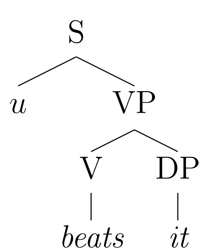
Which of the two options is it going to be? That is a question which DRT in its original form does not address, although the theory was set up in such a way that it could be extended with an 'anaphora resolution module', which would deal explicitly with the identification of anaphoric antecedents.

Intuitively, though, it is clear that the antecedent of *he* could only be x: The individual Pedro represented by x is presumably a person, since in our society human beings are the only animals that can own other things (including donkeys) and the male pronoun *he* is used in English almost exclusively to refer to (male) persons. (Individuals that are introduced as 'donkeys' are not the kind that can be referred back to by means of *he*.) These considerations uniquely select x as anaphoric antecedent for u.

In all other respects the processing of the DP-VP combination is like what we saw in connection with the DP-VP reduction of the first sentence. So the result we get is that in (37)

(15)

$x \quad y \quad u$
Pedro'(x) donkey'(y) $u = x$ owns'(x,y)

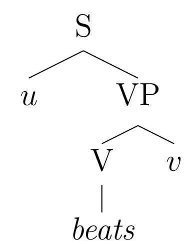


The Pronoun *it* that is direct object to *beats* in (37) is again treated as anaphoric pronoun and again the question comes up which of the discourse referents in the Universe of (34) should be selected as its antecedent. Considerations similar to those just brought to bear on the case of the subject *he* point to y. This time x is ruled out also for grammatical reasons.Once we have chosen x as the antecedent for the subject we cannot use it again as antecedent for the direct object, since that would have led to the interpretation that ‘x *beats* x’ and in English such a semantics can only be expressed with the help of a reflexive, as in ‘Pedro beats himself’, but not with a pronoun, as in ‘Pedro beats him’, let alone as ‘Pedro beats it’.) u, moreover, is excluded for the same reasons as x; the stipulated identity ‘$u = x$’ transfers the reasons that speak against x as anaphoric antecedent from x to u.

All other operations are at this plaint familiar to us. Choosing v as the new dref, we get the result in (16.a), or, after inserting the semantic representation ‘beats’ for *beats* and throwing away the remaining syntax, in (16.b). (16.b) is the DRS for the two sentences of (3.a) together – that is, for the two sentence text as a whole.

(16) a.

$x \quad y \quad u \quad v$
Pedro’(x) donkey’(y) $u = x$ $v = y$ owns’(x,y)



b.

$x \quad y \quad u \quad v$
Pedro’(x) donkey’(y) $u = x$ $v = y$ owns’(x,y) beats’(u,v)

The top down construction algorithm we have been using here has been criticized for being ‘non-compositional’, unlike the bottom up procedures for determining semantic values/logical forms specified in Montague Grammar. One reason that has occasionally been given for this criticism, viz that the construction algorithm works top down, is simply confused:

The bottom up procedures of Montague Grammar can also be used to analyze the semantic values or logical forms of syntactically complex expressions by an analysis that starts at the top. For instance we can start our determination of the truth value of subject predicate sentence S (= DP VP) in a model M like this:

$$[[S]]_{M,\mathbf{a}} = 1 \text{ iff } ([[DP]]_{M,\mathbf{a}})([[VP]]_{M,\mathbf{a}}) = 1 \text{ iff } ...$$

and then slowly work our way down till we reach a statement of the truth value of S in M under **a** which only mentions the semantic values in M of the lexical items from which S is built. (This is precisely the way in which we made use of the unequivocally compositional truth definition for PC in the very first part of the course. No one to my knowledge has ever objected to this 'going from the top down' use of the truth definition for PC as in any way 'non-compositional'; and if any one had done so, the objection would have been dismissed as revealing a total misunderstanding of what compositionality means.)

There is of course a difference between the two cases. The top down use of a truth definition like that for PC is the application of a procedure that assigns semantic values for each syntactically well-formed constituent of the formula to which the truth definition is being applied (going top down). This is not the case for the top down DRS construction algorithm we have just been using; the algorithm doesn't, in any obvious way, provide parts of the DRS that qualify as the logical forms of the syntactic constituents of the sentence that is being 'converted', or has been 'converted', into this DRS. For instance, in the construction we have been going through, it isn't obvious which part of the DRS for the first sentence qualifies as the logical form of the DP *a donkey*, let alone in what precise sense. This is an objection that will disappear when we turn to the bottom up construction method.
. In addition there is also another aspect of the procedure we have been using to derive the DRS for (3.a) that is in conflict with compositionality in its purest form. This aspect has to do with the rules that deal with argument DPs and their predicates: The particular operations that have to be performed to deal with a DP-Predicate configuration depend on the form of the DP. (For instance, the operations for pronouns are different from those for indefinites etc.) So in order that it perform the correct set of operations the algorithm must look down into the structure of the DP in order to determine what kind of DP it is and to perform the operations that fit the type of DP that can be determined this way. According to some conceptions of

compositionality such looking down into constituent structure should not be permitted. At the level at which argument DP and predicate are combined into the predication that results when the DP is made to occupy the relevant slot of the predicate, the internal structure of the DP ought to be invisible to the algorithm

Since the algorithm we have been using to construct the DRS for (3.a) doesn't have to look into DP very deeply, however, but only needs to verify what it looks like one level down, these looking down violations of compositionality are mild ones. But they are violations nevertheless.

These violations of compositionality that are inherent in the original top down algorithms for DRT will also disappear when we switch from top down to bottom up algorithms. However, as hinted at above, going from the top down to a bottom up approach also has its price.
The top down construction of a DRS for the donkey sentence (1.f) closely resembles the one we have used to compute the DRS for (3.a).

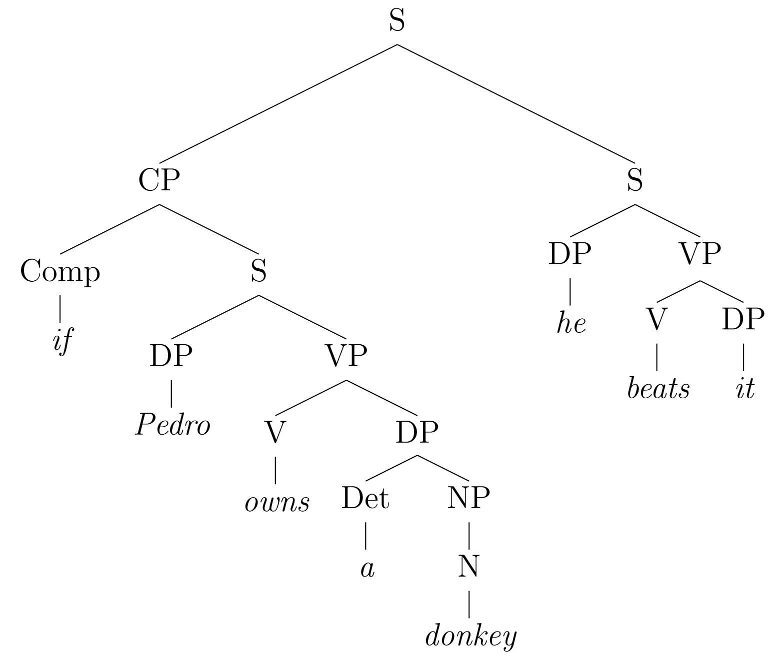
(1.f) If Pedro owns a donkey he beats it.

The first question we have to settle concerns the syntactic structure of (1.f) . The sentence raises two new questions: i what is the syntactic role of *if*; (ii) how do *if*-clause and main clause syntactically fit together?

The answer we adopt to (i) is that *if*-clauses have, like relative clauses, a Comp projection level above S and that *if* fills the Comp position. The answer to (ii): We assume that the *if*-clause is adjoined to the S node of the main clause.

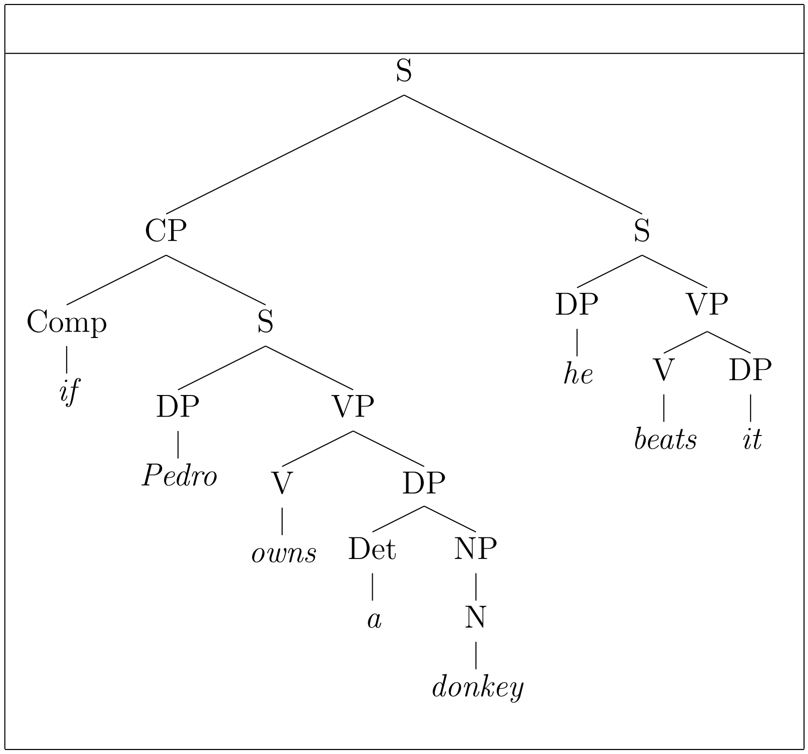
So we assume for (1.f) the following syntax:

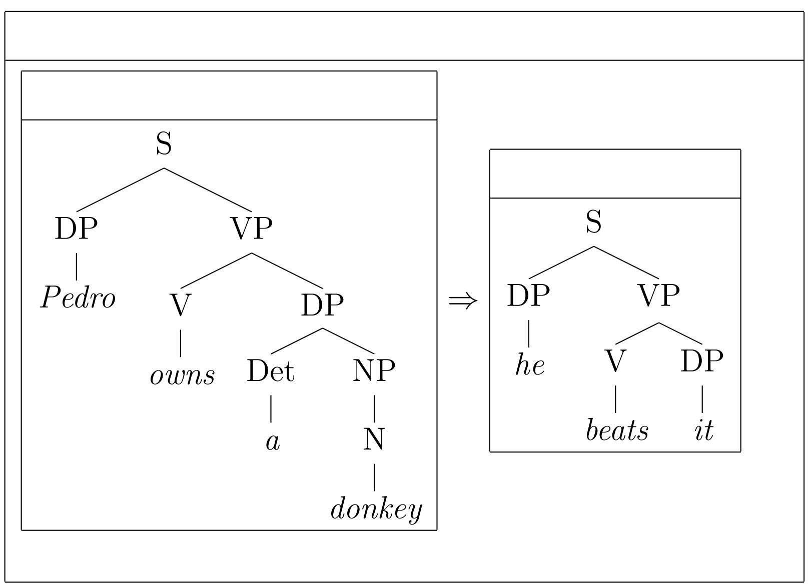
(17)



As before, the construction is initiated by placing this syntactic structure in the Condition Set of an otherwise empty DRS. But the first real construction step now is the one which deals with the combination of main clause and *if*-clause. The precise form in which the separation of these two parts of the sentence is carried out is the semantic contribution made by *if* (as distinct from other fillers of Comp). So, here too we have a mild violation of compositionality in that the algorithm has to look into the subordinate clause in order to determine its Complementizer, just as it has to look into DPs to identify their Determiner (or what else there is in the DP if a Determiner is missing altogether).

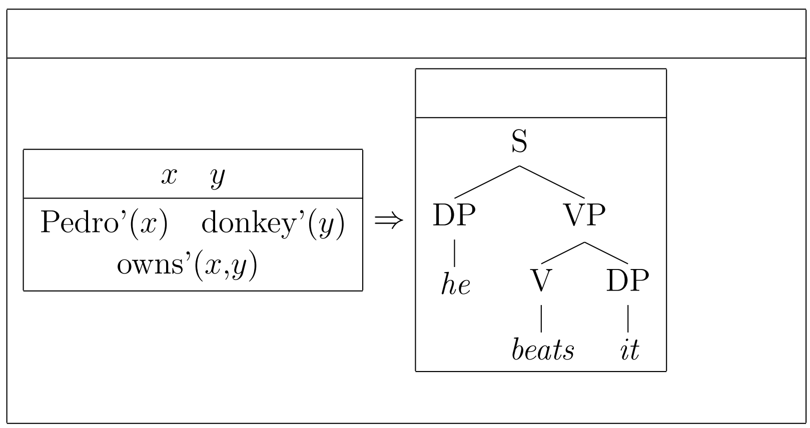
The separation of subordinate clause from main clause dictated by *if* takes the form of introducing a *complex DRS condition*, which consists of two DRSs (the 'antecedent DRS' and the 'consequent DRS') connected by the conditional connective ⇒. So the next two steps are as in (18) and (19).

(18) 
S
CP
S
Comp
S
DP
VP
if
DP
VP
he
V
DP
Pedro
V
DP
beats
it
owns
Det
NP
a
N
donkey

(19) 
S
DP
VP
Pedro
V
DP
owns
Det
NP
a
N
donkey
⇒
S
DP
VP
he
V
DP
beats
it

At this point the algorithm may be applied to the two trees in either order and also by interleaving the operations, on the one tree and the other. In the present case the most straightforward way to arrive at the DRS we want is to first deal completely with the tree in the antecedent DRS (the box to the left of the ⇒) and then turn to the tree in the consequent DRS (the box to the right of ⇒). We have already seen what the algorithm does with the tree in they first box. The very same operations will convert this tree into the discourse referents and DRS conditions shown in (20).

(20)



The DRS construction for the tree in the right hand side box also proceeds in essentially the same way as before. The only difference – inasmuch as it is a difference – is that the anaphoric antecedents for the pronouns are now to be found not in the discourse context established by an earlier sentence, but 'sentence-internally'. However, even this aspect of the construction is much like the one for (3), since what was the discourse context in the last construction is now the DRS for the antecedent of the conditional, whose consequent contains the pronouns that need to be interpreted. So in order that the construction can be completed in the right way we need one further assumption:

(21) the semantic representation for the antecedent of a conditional can play the part of a discourse context in the interpretation of the conditional's consequent.

A consequence of this principle for the interpretation of pronouns can be stated as follows:

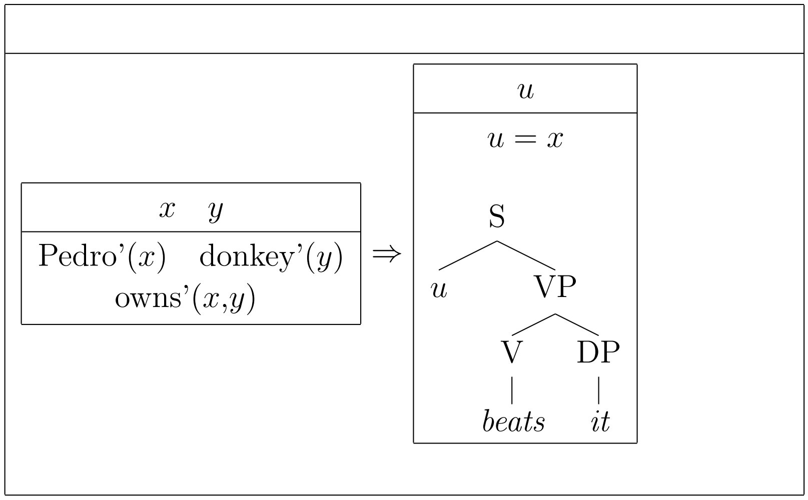
(22) the discourse referents occurring in the Universe of the DRS representing the antecedent of a conditional may be used as anaphoric antecedents for pronouns occurring in its consequent.

The technical term that is used in DRT for this relation between anaphoric pronouns (and other anaphoric noun phrases such as definite descriptions) and drefs – that of the dref being available as a possible antecedent for the anaphoric expression – is that the dref is *accessible from* the expression (in the position the expression occupies in the structure at the point when it is being interpreted). Thus the principle above says about the case before us that the pronouns in the Universe of the DRS for the antecedent of the conditional are accessible to pronouns occurring in the consequent.

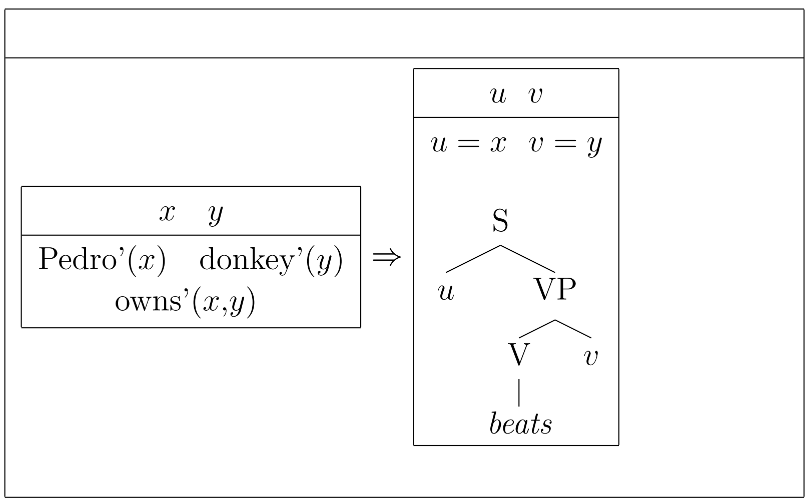
Note that principles (21) and (22) are natural principles, which flow directly from a certain understanding of what conditionals do: the antecedent of a conditional describes a certain type of situation or state of affairs, about which the consequent then makes a further claim. Often there is more than one situation that fits the type described by the antecedent of a conditional. In that case, we will see, conditionals tend to convey a universally quantified meaning, in which there is quantification over all the situations instantiating the type described.

With this additional principle about sentence-internal accessibility, the DRS for (1.f) can now be completed in essentially the same way as we did when we incorporated the contribution of the second sentence of (3.a) into the DRS for the first sentence. We show both steps.

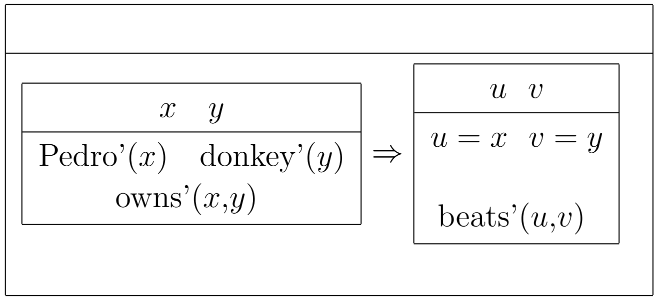
(23)



(24)



Simplifying the last DRS condition from the right hand side box in the by now familiar way turns (24) into (25).

(25) 

(25) is a DRS with an empty Universe and a single, complex DRS condition in its Condition Set. So the truth conditions it determines are entirely fixed by this one condition. What is it for this condition to be fulfilled in a model M?

This is what DRT has to say about this. The verification conditions for DRS conditions of the form '$K_1 \Rightarrow K_2$' are as follows:

(26) An embedding function f into the model M *verifies* $K_1 \Rightarrow K_2$ *in* M iff every function g which extends f with values in M for the drefs in the Universe of K_1 and verifies in M the DRS conditions in the Condition Set of K_1 can be extended to a function h that also assigns values in M to the drefs in the Universe of K_2 and verifies in M the DRS conditions in the Condition Set of K_2.

This sounds like quite a mouthful, but the idea is simple: any way of verifying K_1 in M can be extended to a verification of K_2 in M. But the general statement of this needs to be couched in the more complex terms of (26) because in general the verification of DRS conditions (including conditions of the form ' $K_1 \Rightarrow K_2$ ') arises in a situation where drefs higher up in the DRS have already been assigned values (by the function f of (26)) and these value assignments have to be retained when verifying the DRS condition in question.

However, in the case presented by (25) this complication doesn't arise, since the Universe of the main DRS is empty (and because this DRS isn't itself embedded within some even larger DRS). So the function f spoken of in (26) is simply the 'empty function' in this case – the function that has an empty domain and thus contains no ordered pairs at all. So for this case (26) reduces to the simpler statement:

$K_1 \Rightarrow K_2$ is verified in M (by the empty function) if every function g that assigns values in M to the drefs in the Universe of K_1 and verifies in M the conditions in the Condition Set of K_1 can be extended to a function h that also assigns values to the drefs in the Universe of K_2 and verifies the DRS conditions in the Condition Set of K_2.

Less abstractly, the DRS condition of (25) is verified in M iff:

for every way of assigning an entity $\mathbf{d}_x$ from the Universe U_M of M to x and an entity $\mathbf{d}_y$ from U_M to y such that $\mathbf{d}_x$ is the bearer of the name *Pedro* in M, $\mathbf{d}_y$ is a donkey in M and $\mathbf{d}_x$ owns $\mathbf{d}_y$ in M, it is possible to extend this assignment so that the drefs u and v in the Universe of K_2 are assigned values as well, and in such a way that the conditions in the Condition Set of K_2 are verified too. Since '$u = x$' and '$v = y$' are among these last conditions the only possible assignments to u and v must be the same as those to x and y and we end up with the even simpler statement:

(27) The DRS condition of (25) is verified in M iff every way of assigning an entity $\mathbf{d}_x$ from the Universe U_M of M to x and an entity $\mathbf{d}_y$ from U_M to y such that $\mathbf{d}_x$ is the bearer of the name *Pedro* in M, $\mathbf{d}_y$ is a donkey in M and $\mathbf{d}_x$ owns $\mathbf{d}_y$ in M is such that $\mathbf{d}_x$ beats $\mathbf{d}_y$ in M.

This then also gives the truth conditions for the entire DRS (25):

(25) is verified in M iff the empty function $\emptyset$ verifies the conditions in the Condition Set of (25) in M iff (27) is the case.

Note that this confers upon (25) – and thus also upon sentence (1.f) to which (25) has been assigned as logical form – the universally quantified truth conditions referred to above. This universal aspect enters into the verification condition in (26) through the universal quantification over different possible embedding functions g that verify the antecedent DRS K_1 of the condition '$K_1 \Rightarrow K_2$ '. To specify the verification conditions for conditionals in this form, which makes universal quantification over different possible ways of verifying the antecedent an intrinsic part of their truth conditions, is thus one of the *decisions* that are made as part of formulating DRT. The motivation for building this aspect into the truth conditions of conditional DRS conditions is in part a reflection of an intuition about the situations in which a sentence like (1.f) (repeated once more below) is to be considered true. The crucial cases here are those where our subject Pedro owns more than one donkey.

(1.f) If Pedro owns a donkey he beats it.

Suppose that Pedro owns two donkeys, or three or twenty. Does (1.f) entail that he beats every one of the donkeys he owns? The verification condition for conditionals in (26) reflects the intuition that (1.f) does entail this. But this intuition is not beyond controversy. Some speakers claim that (1.f) can be true for owners of multiple donkeys even if they beat only some – even just one – of the donkeys they own. Others dismiss the sentence as infelicitous under such conditions, as if the use of the phrase *a donkey* in the antecedent and the reference back to it by the pronoun *it* in the consequent, combine to carry a presupposition that one is talking about a person who owns one donkey but no more.

Debates over the exact truth conditions of sentences like (1.f) have been going on for decades and appear to be interminable, in a very literal sense of the word. (At least the debates at which I have been present in person, of which there have been quite a few, never seemed to reach any firm conclusions.) Judgments about the acceptability of various types of donkey sentences in varying contexts seem to depend on many factors, having to do with the tense of the the verb (present or past tense), the kind of verb (state describing verbs like 'owns' or, 'beats' in the dispositional sense in which it must be understood in (1.f) or event verbs like *buy* or *discover*, as well as knowledge about typical situations in which the antecedent of the conditional could be true and how such a situation might then lead to a true instantiation of the consequent. One salient factor is the aspectual status of the verb. Donkey sentences with event verbs are much more easily understood as involving universal quantification than the examples with stative verbs that we have so far looked at, though there are also examples with stative verbs for which speakers seem to get a universal reading more easily than for the example thus far shown. For an example of the first kind consider (28.a) and for an example of the second kind consider (28.b).

(28) a. If Pedro finds a donkey that he likes he buys it.

b. If Pedro likes a donkey he takes good care of it.

These conditionals seem perfectly compatible with multiple instantiations of the situations described by their antecedents; and if there are multiple instantiations – multiple occasions when Pedro finds a donkey he likes, or multiple donkeys that Pedro likes – then the sentences seem to be saying that their consequents hold for everyone of those instantiations. For instance, if Pedro buys some of the donkeys he finds and likes but doesn't buy by any means all of them, then (28.a) would seem false. Likewise, if Pedro likes several donkeys but takes good care of only one of them, then it seems (28.b) is false.

But there are also examples that are perfectly compatible with multiple instantiations of the situations they specify, but seem to require that only one of those instantiations satisfy the claim they make about it. A classical example is the following sentence (due to Robin Cooper).

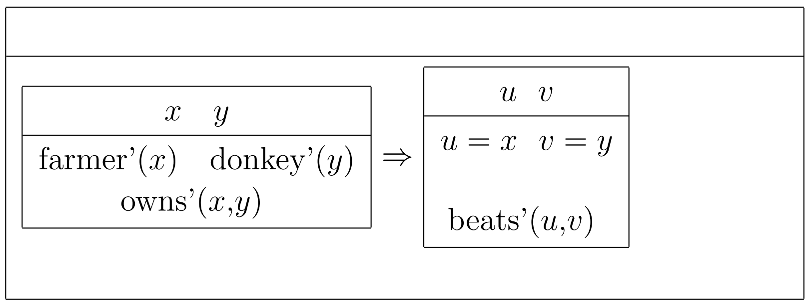
(29) Everyone who had a quarter in his pocket put it in the meter (for the parking pace in which she had parked her car).

To appreciate this example, first a preliminary remark on donkey sentences with universal quantifiers. Among the original donkey sentences (from Geach's 'Reference and Generality') there are not only sentences like (1.f) but also sentences like (1.c) and (1.d), repeated below.

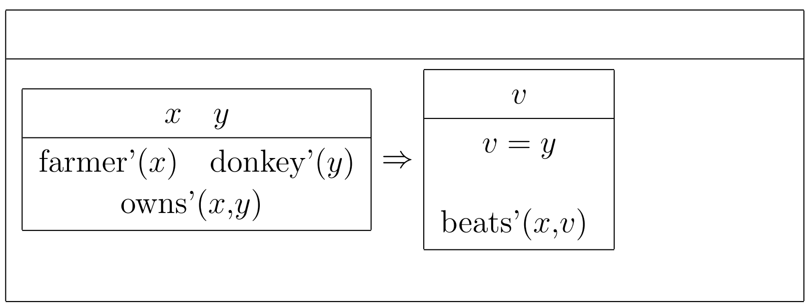
(1.c) If a farmer owns a donkey, he beats it.
(1.d) Every farmer who owns a donkey beats it.

Of these (1.c) can be dealt with just as we have dealt with (1.f). (The only difference is now that the DRS in the left hand side box has a dref x with the constraining condition 'farmer'(x)'.) But dealing with (1.d) requires a new assumption, which has to do with the semantics of universal quantifiers like *every*. The DRT assumption here is that the restrictor of a universal quantifier stands to its nuclear scope in the same semantic relation as the antecedent of a conditional stands to its consequent: in either case the first (antecedent or restrictor) describes a situation in which the second (consequent or nuclear scope) is claimed to hold. In the original version of DRT this similarity was exploited by using as representations for universally quantified sentences the same conditional DRS conditions that were used above to deal with (1.f). Thus the DRS for (1.d) will look much the same as that for (1.c). The only difference is that the representation for (1.c) has a discourse referent for the pronoun *he*, which os missing from the representation for (1.d). (30.a) shows the representation for (1.c), (30.b) that for (1.d).[3]

(30) a.



b.



Note that when (30) is a representation for (1.e), then the left hand side DRS of the conditional DRS condition is the representation of the sister node to the quantifier *every*, viz. the NP *farmer who owns a donkey*, rather than of the *if*-clause *if a farmer owns a donkey*, which it represents when (30) is the DRS for (1.c).

Given that the DRSs for (1.c) and (1.d) come out the same, it follows that the account assigns the same truth conditions to these two sentences. That

[3]I am omitting discussion of the DRS construction rule that introduces the conditional DRS condition in the case of (1.d). for details see Kamp & Reyle (1993).

seems intuitively quite plausible. But note that the problem we noted earlier – what do we take such sentences to say about multiple donkey owners – remains, for both (1.c) and (1.d). Indeed, when the scenario includes multiple donkey owners among the farmers in question, speakers appear to be just as divided about the felicity and truth conditions of (1.c) and (1.d) as they are in relation to (1.f).

We will have more to say about how (30) is constructed as a logical form for (1.d) and also about an alternative format for representing DRS conditions for universally quantified clauses. But we leave these matters for now and return to (29), which gave rise to this preliminary discussion of universally quantifying sentences.

(29) Everyone who had a quarter in his pocket put it in the meter for the parking space where she had put her car.

The point of this example is that according to most speakers it is perfectly felicitous and true if each of the persons in the quantification domain of *everyone* who had one or more quarters in his pocket put just one of those quarters into the meter for the place where she parked. (This example dates from the good old days when parking meters were satisfied with just one quarter. In fact, the original version of the example spoke of dimes – even better days. So in order to copy the judgment, try to put yourself into the mind of person living in such blessed circumstances.)

Cooper and others have taken examples like (29) as indications that the universal readings of indefinites in the antecedents of conditionals and the restrictors of universally quantifying DPs are available only when additional, special conditions obtain. (See in particular Chierchia (1995)).
These various examples are just some of a wide variety that illustrates that judgments about when donkey sentences are felicitous, and under what conditions they are true in case they are felicitous, depends on many factors. We won't engage in any more prodding into this wasps' nest here. All that this exploration was meant to show is that the verification definition in (26) is an option that covers the intuitions of only some speakers about only some of the sentences with donkey pronouns.

One last remark on the question of truth conditions for donkey sentences:

There is much uncertainty among speakers and much controversy among linguists and philosophers about the truth conditions of conditionals in general, not just about those that contain donkey pronouns. One non-trivial question that therefore arises in connection with donkey conditionals is to what extent uncertainties over their truth conditions has to do with the fact that they are donkey sentences or with the uncertainties that attach to conditionals more generally.

4 Syntactic Variants of Donkey Sentences

The truth conditions of donkey conditionals are one set of problems that such sentences present us with. There is another set of problems that have to do with the *configurational* relations which can or must obtain between the donkey pronouns of these sentences and their indefinite antecedents. Here are some examples that illustrate the complications.

(31)

a. If he owns a donkey, Pedro beats it.
b. If he owns a donkey, a farmer beats it.
c. If he owns it, Pedro beats a donkey.
d. If he owns it, a farmer beats a donkey.

e. Pedro beats a donkey, if he owns it.
f. Pedro beats it, if he owns a donkey.
g. He beats it, if Pedro owns a donkey.
h. He beats a donkey, if Pedro owns it.

i. A farmer beats a donkey, if he owns it.
j. A farmer beats it, if he owns a donkey.
k. He beats it, if a farmer owns a donkey.
l. He beats a donkey, if a farmer owns it.

m. If Pedro owns a donkey, he beats one.
n. If Pedro owns one, he beats a donkey.
o. Unless Pedro likes a donkey, he beats it.
p. Pedro beats a donkey, unless he likes it.

The first four of these sentences are like the ones already looked at in that the *if*-clause precedes the main clause. They differ from the earlier examples only in their distribution of pronouns and their presumed antecedents over the two clauses. The remaining sentences all have their *if*-clause in sentence-final position, the first four with *Pedro* as (putative) antecedent for *he* and the second four with *a farmer* instead. The last four sentences (31.m-p) depart from the general pattern on which all of the first twelve sentences are variations, (31.m,n) because they contain the pronoun *one* as an alternative to *it* and (31.o,p), because these contain *unless* instead of *if*.

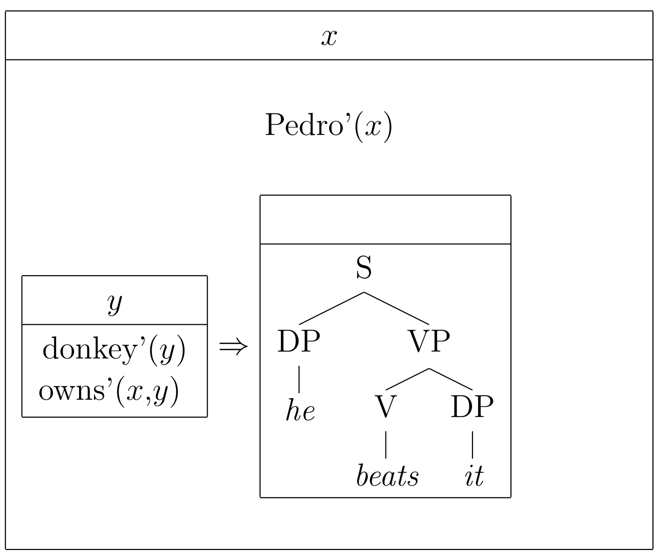
At first site each of (31.a-d) seems problematic for the DRS construction algorithm of which our DRS constructions for (3.a) and for (1.f) were illustrations. But to see what the problems could be we first have to mention an aspect of DRSs structure that we haven't mentioned yet, but that was imposed from the beginning. This is a constraint on accessibility. In constructing htr DRSs for the donkey sentence (1.f), we made use of the fact that the drefs in the Universe of the left hand side DRS of a conditional DRS condition are accessible from positions on the right hand side. We did not say anything about accessibility in the opposite direction. But we must do that now. The thing to say, and the thing that DRT has been saying all along, is that the drefs in the Universe of the right hand side DRS are *not* accessible to positions in the left hand side DRS. This statement is strongly suggested by what we have been saying about the function and meaning of conditionals: the antecedent provides a description of a certain type of situations and the consequent makes statements about situations of that type. Intuitively it seems reasonable that he interpretation of a statement about a type of situation that has just been described should be able to rely on that description (for instance in the resolution of pronouns), but not vice versa. And it was on this assumption that the original formulation of DRT proceeded.

But sentences (31.a-d) do not seem to conform to this constraint on accessibility. Consider, to begin with, (31.a). According to pretty much all speakers consulted, this sentence seems fine – just as good as (1.f) and semantically equivalent to it. But the pronoun *he* is in the *if*-clause and its antecedent *Pedro* in the main clause. So according to the constraint on accessibility, *Pedro* should not be available as anaphoric the antecedent to *he*.

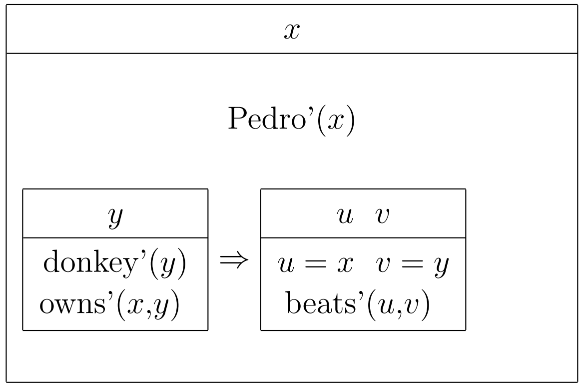
As a matter of fact, the original formulation of DRT has a way of accounting for the acceptability and the truth conditions of (31.a). And that is because its assumptions about the treatment of proper names aren't quite what is implied by our presentation of the DRS construction for (1.f). The assumption that the original DRT account of proper names makes about them is that their drefs are always inserted into the Universe of the main DRS (and the corresponding identifying conditions, like ‘Pedro’(x)’, into the main Condition Set). The motivation for this stipulation is that proper names are never anaphoric, but get their reference from outside the discourse, via the intentions of the speaker, who has some particular individual in mind, knows that it goes by the name of ‘N’, and uses that name to refer to the individual in what she wants to say about it. Furthermore, it is also part of the definition of accessibility that a dref occurring in the Universe of a DRS K is accessible from any *sub-DRS* of K. Thus in the DRS (32) below the dref x in the main Universe is accessible both from positions inside the left hand side DRS of the conditional DRS condition and from its right hand side DRS.

With these additional provisions (31.a) no longer poses any problems. The construction of a DRS for this sentence can now proceed as follows. Processing up to and including the completion of the left DRS of the complex condition, leads to (32).

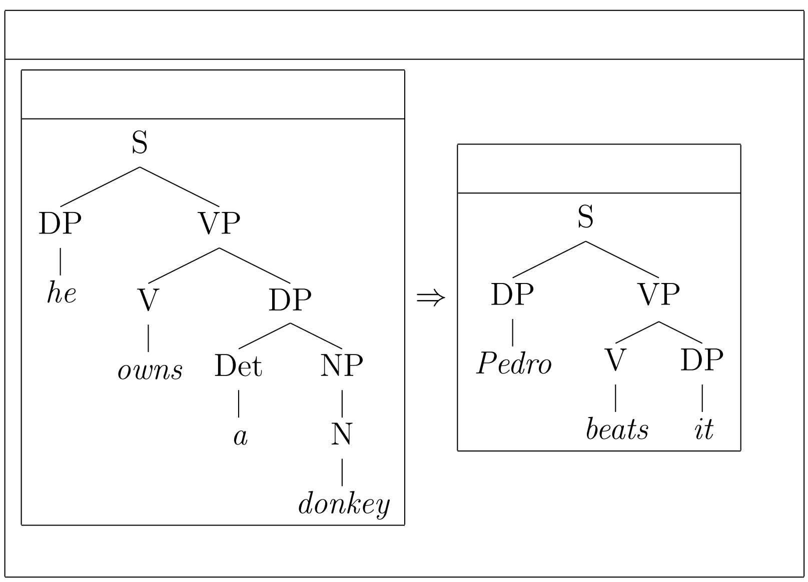
(32)



Given that the dref x, now as a member of the main Universe, is as accessible to the pronoun *he* as it was before, the construction can be completed in the same way as before and we end up with the DRS in (33).

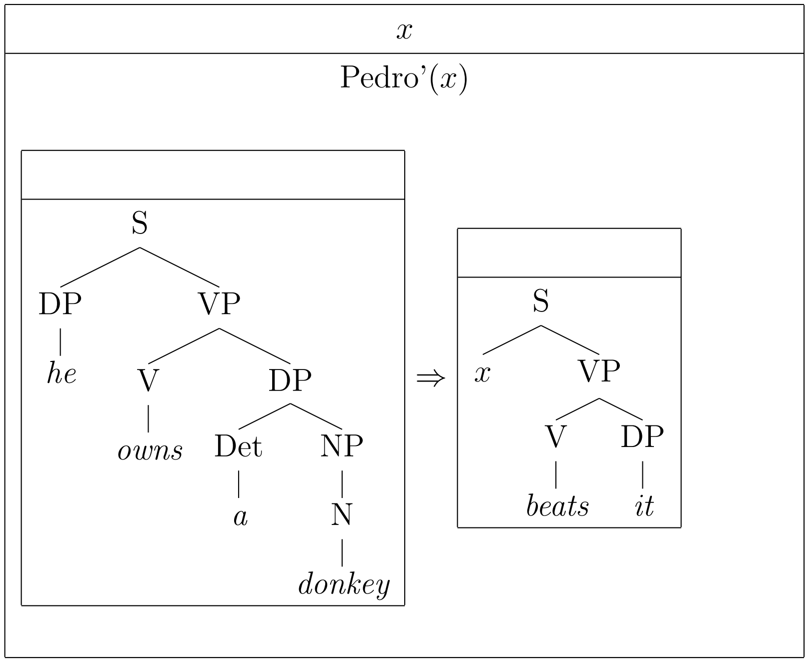
(33) 

We can now also obtain a DRS with the same semantics as (33) for (31.a). That this is possible has to do with a further feature of the top down algorithm we are using, viz. that once *if*-clause and main clause have been separated through the introduction of the conditional DRS condition, construction steps involving the different DRSs can be interleaved in any order. Thus, a DRS for (31.a) can be obtained by proceeding as shown by the following selection of construction stages. (34) is the structure resulting from the introduction of the conditional DRS condition.

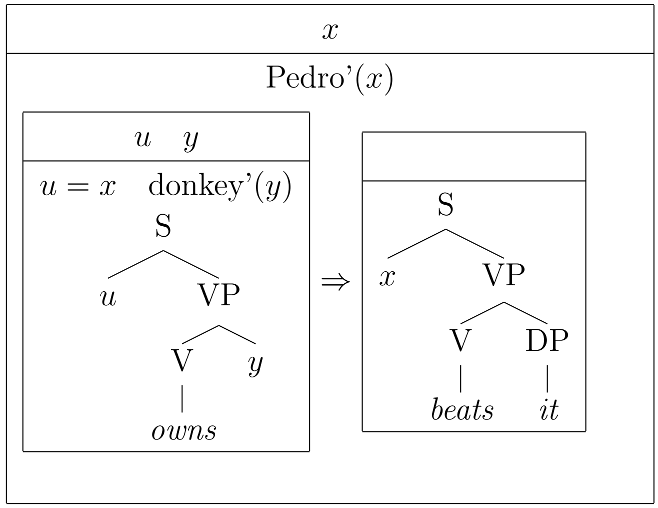
(34) 

Our next step is to deal with the proper name *Pedro*, the subject DP of the syntactic structure in the right hand side DRS. This leads to the insertion of a dref representing Pedro into the main DRS, as in (36).

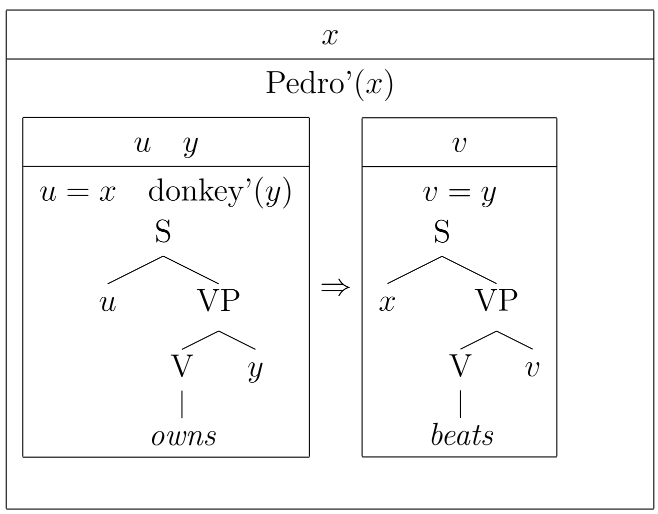
(35)



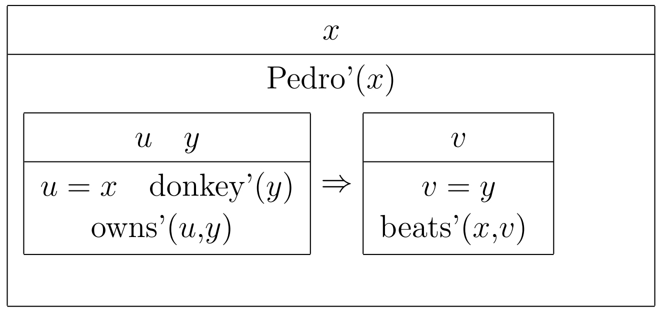
We can now continue the construction by turning to the left hand DRS of the complex DRS condition. We first deal with the pronoun *he*, which we at this point can be interpreted as anaphoric to the dref x that has already been inserted into the main Universe. We then, in the next step, deal with the object DP *a donkey* on the left. This leads to the introduction of a dref y into the Universe of the left hand DRS, as before. (36) shows the result of these two steps

(36) 

The last step then deals with the pronoun *it* on the right. The final result is shown in (37.a), and (37.b) after throwing away the remnants of syntax.

(37) a. 

b.



It is easy to verify that (25), (30) and (37.b) all determine the same truth conditions.

The additional assumptions about accessibility and possible application orders of construction steps give us what we want for the case of (31.a). But without further ado these additional assumption still don't allow us to construct the right DRSs for (31.b,c,d).[4] (31.b,c,d) as things is connected with a feature of the construction algorithm that is illustrated by the DRS constructions we have shown, but that at this point deserves an explicit reminder. In our first pass at DRS construction for (1.f) we inserted all discourse referents for DPs into the Universes of the very DRSs that contained the structures of which those DPs were constituents in their Condition Sets. We then corrected this policy for proper names, by stipulating that their drefs always go to the Universe of the main DRS. But that correction *only* applies to proper names; it does not apply to indefinites. And it shouldn't.

That we shouldn't allow for this in general is made plain by another look at (1.f). Suppose it was possible to place the dref for the indefinite *a donkey* into the Universe of the main DRS, The, as we have just seen in our second DRS construction for this sentence in connection with the anaphoric relation between *he* and *Pedro*, it would be possible to complete the DRS to one in which *it* is construed as anaphoric to *a donkey*. But this DRS would not give the right truth conditions. It would say that there is a donkey such that if Pedro owns it he beats it. For (1.f) such a reading is marginal at best, and

[4]There may be some disagreement about how acceptable these sentences. But my impression is that speakers accept them and take them to have the same meanings as the donkey sentences we discussed in the previous sections. I proceed on the assumption that this is so.

quite a few speakers say that they do not get it at all.[5]

So, how do (31.b,c,d) get the meanings that speakers find they can and do have? Apparently something more radically different from what we have been saying so far is needed to account for this. Presumably the answer has to do with an aspect of donkey sentences that we also find in certain sentences without donkey pronouns. Many sentences in the simple present tense are understood as expressing *generic* or *dispositional* statements, independently of whether they involve donkey pronouns. For instance, (37.a) and (37.b) are generic statements about Mary and her dog.

(38) a. Mary beats her dog.

b. When Mary is in a bad mood, she beats her dog.

These sentences are not about some particular event of Mary beating her dog, but about something that has a tendency to happen regularly: the relationship between Mary and her dog is fraught with repeated episodes of the kind described. In particular, (38.b) says that occasions when Mary is in a bad mood are occasions when she beats her dog. Evidently the element of generic (or 'quasi-universal') quantification that is part of the truth conditions of the sentences in (38) is not conveyed by an indefinite-donkey pronoun relation, as there aren't any donkey pronouns around. Rather, it is the use of the simple present tense in application to an event verb such as *beat*. (38.a) shows that this combination of event verb and simple present tense is enough by itself to trigger such a generic reading. Perhaps the effect is amplified by the presence of a *when*-clause that is also in the simple present, as in (38.b). But as shown by (38.a), even the presence of such a subordinate clause isn't required.

[5]Some indefinite noun phrases, we have seen, allow for specific interpretations. Placing the drefs for such indefinites in the Universe of the main DRS would do justice to the intended reading. But *a*-DPs whose NPs are short and have an large, open-ended set of satisfiers (like the noun *donkey*) are poor candidates for specific interpretation. Moreover, even if a specific reading was possible for *a donkey* in (1.f), the reading one would obtain by putting its dref into the main DRS Universe would not give us the specific reading but only the problematic wide scope existential reading. The correct account of the semantic contributions of specific indefinites is a complex issue that we cannot go into here. But is should be clear that just placing their dref in the main DRS Universe (and the condition(s) contributed by their NP into the main Condition Set) won't do justice to specific interpretations.

Let us assume, in the spirit of these observations, that occurrences of an event verb like *beat* in the simple present tense can trigger conditional DRS conditions like the one we have been using in our representation of (1.f). (So an *if*-clause-main clause combination isn't the only syntactic configuration that gives rise to conditional DRS conditions.) Then it is possible to assume that indefinites that are part of the material that ends up in the left and and right hand DRSs of a conditional DRS condition triggered in this way to deliver their drefs to the Universes of these sub-DRSs. In particular indefinites that end up in the left hand DRS may get represented by drefs in the Universe of that DRS.

In rough outline this gives us an account of why the sentences in (31.b,c,d) are felicitous and have their respective interpretations. A lot is still missing from the account. For one thing, if it is the tense of the verb that has to do with the possibility of dispositional or generic readings of the sentences we are looking at, then we will need more refined syntactic structures for these sentences as inputs to the construction algorithm, in which information about tense (and other syntactic indicators of tense and aspect) is made explicit. To go into this here is out of the question. But in the next part of the course, in which tense and aspect will be our central concern, we will adopt syntactic structures in which such information is made fully explicit. At that point it will be possible to show in greater detail how the right DRS for (31.b,c,d) can be constructed.

In all the remaining sentences in (31) the *if*-clause follows the main clause. The peculiarity of these sentences is that what from a logical point of view comes first, viz. the *if*-clause, comes only at the end. This appears to cause tensions of a sort that is not found in donkey sentences with sentence-initial *if*-clauses. (31) lists 2 times 4 variants of this same pattern,which differ from each other in whether the pronouns occur in main clause or *if*-clause and whether the antecedent for *he* is the proper name *Pedro* (31.e-h) or the indefinite *a farmer* (31i-l). In each group of four I have listed the examples in order of decreasing acceptability. But as I am not a native speaker, my comparative judgments count for naught and readers should try to make up their own minds as to how acceptable they find each of these sentences. (Some of the sentences may seem better without a comma. I have added commas across the board, in an effort not to let the presence or absence of a comma interfere with people's judgments. Take out the commas wherever you feel that this improves the sentence.)

There appears to be substantive agreement that the first sentences of each of the two groups ((31.e) and (31.i)) are acceptable and that for each group the last two sentences,((**??**.g,h) and (**??**.k,l)), are not at all good. The unacceptability of these last four sentences, moreover, is usually blamed on the uninterpretability of the sentence-initial subject pronoun *he*. Our approach explains this if we assume that the sentence-final *if*-clauses in these examples are syntactically attached to the main clause at a lower point than we have been assuming for sentence-initial *if*-clauses. (But in our discussion of generic interpretations above we have already found reasons to assume that the syntax of sentences with sentence-initial *if*-clauses may have to be reconsidered in any case.) On the assumption that the sentence-final *if*-clauses attach below the S level at which the subject DP combines with the VP, it is one of the assumptions of the top down method of DRS construction that the first construction step will have to deal with the subject *he*. At that point no antecedent is available for it, so the construction aborts right there.

Low attachment of sentence-final *if*-clauses does not block DRS construction for the first two sentences of each group ((31.e,f) and (31.i,j)) (at least not if we assume that here too conditional DRS conditions can be triggered by a generic interpretation of the verb). For (31.e) and (31.i), which seem acceptable, this is as now would like things to be. As regards (31.f) and (31.j) however, the situation appears to be less clear. Are these sentences acceptable or are they not? I leave this question to you. Only after it has been answered is there a point in worrying about whether our theory predicts a reading for it or does not come up with any coherent interpretation.

The last four sentences in (31), (31.m-p), are tidbits meant to give a taste of the range of complexities that one has to face up to when the donkey sentence examples we have considered are placed in the wider context defined by all the different kinds of natural language constructs that can be used to express conditional, universally quantified or generically quantified contents. Note that the pronoun *one* in (31.m) also gets an anaphoric interpretation of sorts in this sentence, but that the kind of anaphora involved is of a quite different kind then the *she, he, it* anaphora we have seen examples of so far. The anaphoric antecedents of *one* are nouns, or, more generally, NPs. And when a given noun or NP is identified as the 'antecedent' of *one*, then the contribution that *one* makes as argument phrase is that there is something of the kind described by the noun or NP of which the predicate is true. This in (31.m), where it seems intuitively clear that the antecedent of *one* is the noun/NP *donkey*, the reading we get is that if there is a donkey that Pedro owns, then there is a donkey that he beats. Clearly that is a different meaning from the one that we have been associating with (1.f). It is weaker in that it clearly requires no more than the beating of one donkey (in case the antecedent is true), and moreover that the donkey that Pedro beats need not be one of the donkeys he owns. (Even the sentence 'If Pedro ones a donkey then he beats one of them.', to the extent that that sentence is felicitous, has a weaker reading than the one we have associated with (1.f) in that there is clearly no need for more than one donkey to be beaten and perhaps even an implicature that only one donkey is beaten. This last reading, by the way, can be conveyed somewhat more felicitously by: 'If Pedro owns one or more donkeys then he beats one of them.')

The comparison between donkey pronouns and their replacements by *one* is of particular interest in those cases where the donkey sentence is supposed to have an existential rather than a universal reading. Recall Cooper's example (29), repeated as (39.a) and compare it with the *one*-variant in (39.b).

(39) a. Everyone who had a quarter in his pocket put it in the meter (for the parking pace in which she had parked her car).

b. Everyone who had a quarter in his pocket put one in the meter (for the parking pace in which she had parked her car).

These two sentences appear to be saying more or less the same. But there seems to be some kind of difference between them nevertheless. As regards (39.b) we already noted that this sentence doesn't entail that the quarters put into the meters must have been in the pockets of the ones who insert them. But even if that possibility is excluded, (39.b) seems to carry a different connotation than (39.a): it suggests that finding more than one quarter in one's pocket is to be considered as much of a live option as finding just one. (39.a), although it doesn't explicitly exclude the possibility of finding more than one quarter, doesn't present it as a 'primary option' either.

Nothing in what we have said indicates how this difference might be captured and so we must leave it to be accounted for using what may turn out to be quite different concepts and tools than have been presented up to this point.

That the kind of anaphora involved in the interpretation of *one* is different from that involved in *she, he, it* can also be concluded from a closer look at (31.n), in which *one* is in the *if*-clause and its antecedent in the main clause. This sentence seems perfectly felicitous, whereas this iOS not obviously so for the sentence 'If Pedro owns it, he beats a donkey.'. The difference has to do with the fact that the noun *donkey* to which *one* in (31.n) is arguably just as much available at the highest level (and therefore accessible to an anaphoric element in the antecedent of the conditional) as a proper name like *Pedro*: in either case what we are dealing with is a fixed relationship between an expression and its referent (a person in the one case and a set or property in the other). Thus, if *one*-anaphora is treated in the same way that we have been treating *she, he, it*, with a dref for the noun or NP that is identified as antecedent for *one*, then the dref for *donkey* would go to the main Universe and would be available to *one* in the same way and for the same reason as the dref for *Pedro* is available for *he* in (31.a).

The last two sentences show that sentence-internal donkey anaphora doesn't only arise in conditional sentences but also in sentences with subordinate clauses that relate to their main clauses in semantically and logically different ways than *if*-clauses. This is yet another dimension of variation, indicating that donkey pronoun phenomena interact with a wide variety of other constructs.

4.1 Reflection and Summary

This section has been a bit like fighting a Lernaean Hydra: each time the account of a problem seems in sight three others pop up in its place. You may well have been asking why you should have been exposed to something as incomplete and tentative.

Here are some reasons. First, if it is right that anaphora and its implications for sentence and discourse cohesion are the same in donkey sentences and in donkey discourses, then this should be manifest in the treatment of anaphoric pronouns; the treatment should be essentially the same irrespective of whether their antecedents occur in the same sentence or elsewhere in the discourse. The present proposal involves a number of assumptions – about the representation of linguistic information in the form of DRSs (with their particular repertoire of simple and complex DRS conditions), the incremental building of DRSs for multi-sentence discourses and lastly the construction algorithm that builds sentence DRSs from syntactic structures of individual sentences. As said, most of this we will retain, but the part of the package that has been discussed in this last section – the method of constructing DRSs top down – will be given up. However, in doing that, replacing the top down construction method by a way of constructing DRSs bottom up, we give up something that goes a fair way towards explaining how anaphoric pronouns work (even if, as things stand, it does't go nearly far enough) and that something will have to be made up for in some other way when the switch from top down to bottom up is made. So it is well to pause for a moment and reflect on what it is we will be giving up by making the switch.

The reason that the top down construction algorithm works for anaphoric pronouns to the extent that it does is the requirement that the a pronoun's antecedent has been processed, and its dref thereby made available in a position accessible to that of the pronoun, at the point when it comes to be the pronoun's turn to get interpreted. There are thus two factors that must be satisfied in order that a pronoun can be construed as anaphoric to an antecedent: (i) the antecedent must be accessible to the pronoun (via the dref that represents the antecedent); (ii) the dref for the antecedent must actually be present in the representation that is being constructed at the point when the pronoun gets interpreted. It is important to be aware that these are very different kinds of constraints, and to understand how different they are. (i) is a *configurational* constraint, which has to do with the structural relations between pronoun and antecedent in their respective positions, whereas (ii) is a *processing* constraint, which is related to the order in which the different parts of a sentence are visited by the interpretation device (for us: the construction algorithm).

The configurational constraint (i) is itself the result of interaction between two distinct factors. On the one hand there is the position of the antecedent DP in the syntactic structure of the sentence that is being converted into its logical form (the DRS that is being constructed) and on the other there are the other there is the form of the DP itself. The interpretation of an anaphoric pronoun requires that it stands in a relation of accessibility to the dref that is chosen to serve as its antecedent,and the position of that dref, at the point when the pronoun is interpreted, is a function not only of the syntactic position of the DP that it represents, but also of the DFP' form, for as we have seen different DPs –among them proper names, indefinites and universally quantifying DPs, introduce their drefs into different DRS Universes, some of which may be accessible from the position of a given pronoun while others are not.[6]

But the most important difference between the constraints on anaphora resolution that the top down algorithm brings together (or jumbles up, as a less sympathetic voice might put it) is that between the configurational and the procedural constraints. The first of these two types of constraints the first (i.e. the role of configurational constraints) is generally recognized as essential to the way pronominal anaphora works in a language like English. (For instance, it was a central part of Chomsky's 'Theory of Government and

[6]We have already seen how the difference between proper names and indefinites can have an impact on anaphoric possibilities: In some cases a proper name can serve as antecedent to a given pronoun whereas a non-specific indefinite in the same syntactic position cannot. Here, moreover, is a pair of examples showing that the difference between indefinites and universaly quantifying DPs can make such a difference too. Consider the sentences in (40).

(40) a. Every critic who commented on a movie by Tarantino praised it.

b. Every critic who commented on every movie by Tarantino praised it.

There is wide agreement that the *it* of (40.a) can be unproblematically construed as anaphoric to *a movie by Tarantino* but that an anaphoric relationship between *it* and *every movie by Tarantino* is not possible (or at best very marked). The explanation for this difference is that the dref for *a movie by Tarantino* is inserted into the Universe of the left hand DRS of the conditional DRS condition that is used to capture the semantics of the quantifying DP *every critic who commented on a movie by Tarantino* and is therefore accessible to a pronoun in the right hand side DRS of that condition. In contrast, the dref for *every movie by Tarantino* in (40.b) ends up in the Universe of a conditional DRS condition capturing the contribution of this *every*-DP, which belongs to the condition set of the left hand side box of the condition capturing the contribution of the subject DP, and drefs in this Universe are not accessible from the DRS that contains *it*. (This is a fact about accessibility that we had not yet mentioned, but it is part of how accessibility is defined in DRT. See Kamp & Reyle (1993), Chs. 1,2). The difference is easier to see in the DRSs in (41.a,b).

(41) a.

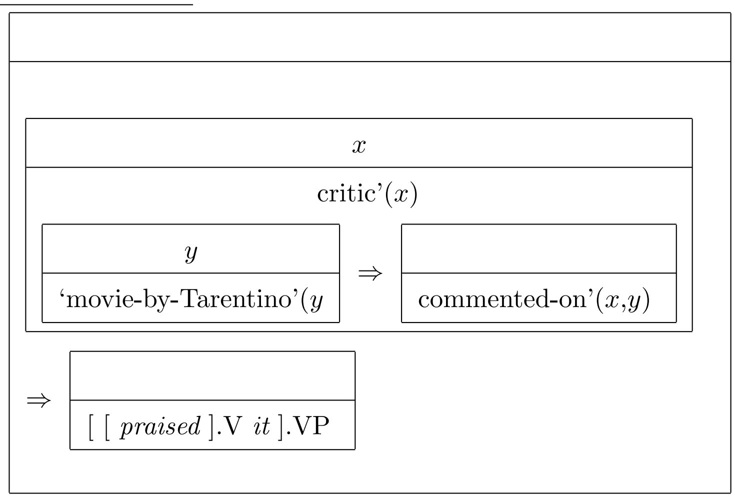
x y / critic'(x) 'movie-by-Tarentino'(y commented-on'(x,y) $\Rightarrow$ [[*praised*].V *it*].VP

Binding' (Chomsky (1982)) which was taken over more or less unscathed in Minimalism, the Chomskyan grammar model today.) The procedural aspect of the top down algorithm are less widely accepted or represented in work on syntax and semantics, for one thing because most theories have no place for the dynamics of logical form construction that is an essential ingredient of DRT. And as we have seen, this aspect is not without its problems. On the one hand it delivers the right results for the paradigmatic donkey sentences given in (1). And it also accounts for why the sentences in (31.k,l) are no good. But on the other hand it was of no direct help in dealing with many of the other sentences in (31).

That of course is no definitive argument against the top down method. But there is a more serious worry: Can/Should procedural constraints be made to play any part at all within the kind of over-all architecture that DRT represents? What should make us pause before we say 'yes' to this question is that DRT's architecture shares with other approaches to the syntax-semantics interface a certain feature that is quite unrealistic from a processing perspective: A complete syntactic structure has to be in place for the entire sentence before semantic processing of its semantics can get off to a start. Given the time pressure that we are under when listening to what others are saying, it

---

b.



Note that the syntactic positions of *a movie by Tarantino* in (40.a) and of *every movie by Tarantino* in (40.b) are identical. So the difference between the anaphoric possibilities in (40.a) and (40.b) can be explained only in terms of the different Universes that *a*-DPs and *every*-DPs send their representing drefs to.

would be amazing if this was the way we actually went about making sense of what they say, for by the time the entire sentence has been uttered – and it is only at that point that we can be sure that we have the syntactic structure for the sentence as a whole – the speaker will have started on the next sentence and so we would constantly be in the predicament of doing two things at the same time: (i) paying attention to the speaker's words and (perhaps) building a syntactic parse for them and (ii) constructing a semantic representation for the preceding sentence from the syntactic structure for it that is now in place. Arguably, discourse processing along these lines cannot be excluded a priori. But on the face of it it doesn't look very plausible. And in fact there has been work in psycholinguistics going back to the sixties which strongly suggests that at least some semantic processing happens on-line (i.e. as the words of the sentence reach the interpreter). And for all we know *a lot* of semantic processing happens while the interpreter receives the successive parts of a sentence, with much or all of the semantic representation in place by the time the last word of the sentence reaches the interpreter.

Presumably most linguists would concur with this assessment of what has some likelihood of going on in human listeners[7] But so far no one has come up with a model of on-line syntactic and semantic processing that comes even close to accounting for the details of syntactic and semantic structure that the generative formal approaches to grammar and meaning have made it their task to chart (and that they have done a creditable job to detect and describe). The model provided by DRT might be seen as a small step towards a more procedure-sensitive account of the syntax-semantics interface. But when it comes to the processing of individual sentences, it isn't clear that the version of DRT discussed in this section constitutes any real progress over static grammar models like MG, which emphatically declare problems of language processing to be outside of their purview, .[8]

[7]And to some extent also in readers, though the situation is different there; notoriously readers shift back to earlier parts of a sentence and, sometimes even to earlier sentences, after they have reached a certain point in the sequence of words on the page.

[8][Here references to work in psycholinguistics that seems too provide some confirmation of DRT as a processing model: Peter Gordon, Ted Gibson, Tessa Warren, ?]

5 Definite Descriptions, Presuppositions and Pronouns

In the remainder of the course we will adopt a bottom up method for constructing DRSs. A central reason for this has to do with a view of the role and nature of pronouns that differs importantly from what is implicit in the treatment they receive in the kind of top down DRT discussed in the previous sections. Switching from top down to bottom up is in part a reflection of, and necessitated by, a very different view of the nature of pronouns than is built into the top down construction algorithm we have seen in action. This is a matter with a long history, which is of some independent interest; and it is a history that is bound up – in a curiously sinuous manner – with the views that have been held, over a successive decades, concerning definite descriptions. (One of the curiosities of the history of formal semantics is the wide gap between the views that people have entertained about the role and functioning of pronouns on the one hand and what they thought about definite descriptions on the other.) Because of this, saying something about the various views that have been held about pronouns is hardly possible without saying something about definite descriptions as well. And since a discussion of definite descriptions ought to be part of any introduction to formal semantics – some time, somewhere – in any case, we may as well say something about them here.

We follow what has become a kind of tradition in starting our (mini-)discussion of definite descriptions with Frege. Frege saw definite descriptions as a threat to formal logic, for the following reason. Some descriptions, such as *the smallest prime number*, properly refer to something, viz. to the unique entity that satisfies the predicate expressed by their NP ('smallest prime number' in this case). Others, such as *the largest prime*, do not refer to anything, in this case because there is no largest prime – no number satisfies the predicate 'largest prime'. Frege saw the difference between definite descriptions that properly denote and descriptions that do not as the difference between *fulfilled* and *failed presuppositions.* And he thought, rightly, that presupposition failure was a threat to the logic he had formulated in his Begriffsschrift (which we now know and use as the classical Predicate Calculus and that he needed as an unquestionably sound foundation for his Grand Project of showing that arithmetic was a branch of pure logic).[9]

One way of protecting predicate logic from the dangers of failing definite descriptions is to simply exclude them from the repertoire of constructs that the syntax of one's predicate calculus admits. This is what is usually done today: The calculus lacks the syntactic mens for forming terms out of formulas, and where the term denotes ti uniques satisfier of the formula, provided the formula has a unique satisfier. (The formulation of PC given in the first part of the course is an example of such a formulation.) Depriving the syntax of predicate logic of definite descriptions may make the translation of natural language sentences with definite descriptions into formulas of the Predicate Calculus a little harder, but one soon comes to realize that this problem is dwarfed by the countless other problems, some of them much harder, that we encounter when we start trying to 'translate natural language into logic' on a larger scale.

In fact, a recipe for translating definite descriptions as part of translating sentences containing them into a predicate logic that has no directly corresponding syntactic construct was offered by Russell as early as 1905 (see Russell (1905)). Russell argued that Frege's worry about presupposition failure of definite descriptions was unnecessary, since there was a perfectly good way of capturing their contributions within a predicate logic that doesn't have (the formal counterparts of) definite descriptions (e.g. Predicate Logic as we have defined it in Part 1 of this course). Russell's proposal is easiest to explain by giving some examples. Here are two. (43.a) is an obviously true sentence involving the properly denoting description *the smallest prime number*, (43.b), with the improper description *the king of France*, is perhaps Russell's most famous example sentence; and it is no doubt the most famous sentence in the by now voluminous presupposition literature. (43.c) and (43.d) are Russell's translations of these two sentences.

(42) a. The smallest prime number is even.

[9] Frege held that when a constituent of a sentence did not function in the way it was supposed to, and thus failed to make the semantic contribution to the sentence expected of constituents of its kind, then the sentence as a whole must fail to do its semantic and logical job, that of denoting a determinate truth value. According to Frege non-denoting definite descriptions were an instance of this: A sentence containing a definite description that doesn't properly denote is thereby prevented from denoting a truth value (and thus from doing what it is for).

b. The King of France is bald.

c. $(\exists x)$(‘smallest-prime-number’(x)[10] & $(\forall y)$((‘smallest-prime-number’(y) $\rightarrow y = x$) & even(x))

d. $(\exists x)$(King-of-France(x) & $(\forall y)$(King-of-France(y) $\rightarrow y = x$) & bald(x))

Each of these translations makes use of the same device: spelling out the contribution of the definite description as saying (i) that there is something satisfying the content of its NP; (ii) that there is only one such thing and thus that the satisfier is unique; and (iii) that this unique satisfier satisfies the predicate to which the definite description is an argument (here the VP, *(is) even* in (43.a) and *(is) bald* in (43.b)). Given this way of analyzing the contributions of definite descriptions sentences containing descriptions will determine a definite truth value irrespective of whether their descriptions properly denote. When the description is proper (in that its content has a unique satisfier), then the truth of the sentence will depend on whether this unique satisfier satisfies its predicate. If the description is improper, then that will often be the decisive factor for the truth or falsity of the sentence and the predication involving the predicate that has the description as a syntactic argument will be irrelevant to it. In particular, when, as in (43.a,b), the analysis of the description is given wide scope (in the sense that its initial existential quantifiers has scope over all other logical operators in the translation of the sentence), then the sentence as a whole is false.

This is a consequence of Russell's analysis that may strike you as counterintuitive: when a non-denoting description such as *the King of France* causes falsity of the sentence to which it belongs, then the sentence is to be regarded as false, but the falsity of the sentence has nothing to do with the predication involving the description and its syntactic predicate. Thus in (43.b) it doesn't make any difference whether the syntactic predicate of *the King of France* is ‘is bald’ or ‘isn't bald’. That is, both sentences in (**??**) are false.

(43) a. The King of France is bald.

[10] A proper formalization of *smallest prime number* would express the superlative *smallest* in terms of the corresponding comparative. So instead of “smallest-prime-number’(x)’ one would use the formula ‘number’(x) & prime’(x) & $(\forall z)$(number’(z) & prime’(z) $\rightarrow \neg$ smaller-than’(z,x)’

b. The King of France is not bald.

So while Russell is right that his theory preserves the valid rules of classical predicate logic, his translation of sentences with descriptions into predicate-logical formulas may seem rather procrustean. [11]

In the course of time Frege's view of how descriptions work in languages like English has won the day, at least within linguistics. Two largely separate developments led to this widely shared perception. The first was a development within philosophy. In 1950 Strawson attacked Russell's 'Theory of Descriptions' for its failure to draw a proper distinction between (i) the question whether a sentence has a well-defined meaning and (ii) the question whether it has a well-defined truth value. For instance, Strawson argued, Russell's sentence 'The King of France is bald' expressed (at the time when Russell used it, in his 1905 paper) a perfectly well-defined *meaning*, just as it did or would have done if used two centuries earlier, at a time when France did have a king). What the sentence didn't do in Russell's own time was to determine a truth value (or, as Strawson put it, express a well-defined proposition, which would have been either true or false). In Russell's account no distinction between being meaningful and determining a proposition is made. And that, Strawson stressed, led Russell to a choice between only two options: (i)

[11] Another bone of contention: Russell saw it as another feature in favor of his account of descriptions that it allows for the possibility that the translation of a definite description can end up within the scope of some other operator. Consider for instance, the analysis in (44.b) that he proposed for the sentence in (44.a).

(44) a. The golden mountain doesn't exist.

b. $\neg(\exists x)(\text{mountain}(x)\ \&\ \text{golden}(x)\ \&\ (\forall y)((\text{mountain}(x)\ \&\ \text{golden}(y)) \rightarrow y = x)$ $[\&\ \text{`exists'}(x)])$

In Russell's formalization of this sentence the sentence comes out as true because the analysis of the description is within the scope of the negation: in (44.b) the formula as a whole is true because it denies the unique satisfaction of 'golden mountain' and since this predicate doesn't have a (unique) satisfier, the denial is correct. (In (44.b) I have put the 'predicate' 'exists(x)' in brackets, since for Russell existence isn't a bona fide predicate; for him it is at best a trivially satisfied predicate like 'being equal to something' or 'being self-identical'; if we follow Russell on this point, we could replace 'exists(x)' in (44.b) by something like '$(\exists z)z = x$'.)
This might initially have seemed a nice additional bonus of Russell's theory. However, it is now widely recognized (for linguistics reasons) that Russell's analysis of such embedded descriptions is in general not really tenable.

declare sentences with non account-denoting sentences as meaningless (and thus a fortiori as failing to determine a truth value) and (ii) account for the meaningfulness of such sentences by assigning them a truth value after all. Against this, Strawson asserted that while there is something wrong with the use of sentences with non denoting- descriptions – they do not determine a truth value – that doesn't entail the intuitively absurd claim that one cannot 'make sense' of such sentences.

Eventually, Strawson incorporated the notions of presupposition and presupposition failure into his views about the meaningfulness of sentences and their determining a truth value: presupposition failure of a definite description affects the latter – when a sentence contains a definite description whose presupposition fails it fails to determine a truth value – but it does not affect the former: sentences with failed presuppositions have a well-defined meaning no less than sentences for which there is no presupposition failure. If one accepts these different parts of Strawson's account, then presuppositions become an important ingredient in an account of the semantic and logical properties of sentences. We will return to the role of presuppositions in the theory of meaning extensively. In the context of the present discussion all that matters is that presupposition failure accounts for the possibility of a sentence being meaningful and yet failing to be either true or false.

Perhaps the Russell-Strawson controversy would have remained a controversy[12] if it hadn't been for a development within linguistics that started in the late sixties. At that point linguists became aware that important aspects of the phenomena which Frege and Strawson had detected in connection with definite descriptions can be observed with other expressions and grammatical constructions as well; and it soon became clear that there are a great many of such 'presupposition carrying' expressions and constructions. Here are just two examples of presupposition carriers other than definite descriptions. (More examples will follow when we turn to presupposition in earnest later on in the course.)

(45) a. Mary stopped smoking.

b. Mary didn't stop smoking.

c. Fred regrets that he bought 5000 shares in BP.

[12]in fact, I believe it still is in some philosophical circles

d. Fred doesn't regret that he bought 5000 shares in BP.

For each of the examples, the one involving the aspectual verb *stop* and the one with the attitudinal verb *regret*, we have given both a sentence containing the verb and the negation of that sentence, as an indication that what we are seeing in either case is something like presupposition and its potential failure. Consider for instance the first two sentences. Suppose that until some given past time t_0 Mary smoked and then gave up. In that case (45.a) would be true and (45.b) false. And if Mary smoked until t_0 and then went on smoking and continued to do so until today, then (45.b) would be true and (45.a) false. But if Mary never smoked at all, then both (45.a) and (45.b) seem funny things to say. They both try to describe cases, one might be inclined to say, that do not apply: *stop* comes with a 'presupposition' that what is described by its complement (here *smoking*) was true up to the time of the stopping, of which (45.a) is trying to say occurred and that (45.b) is denying. (One way the peculiarity of talking about somebody stopping smoking when she never did smoke comes out is when somebody says to you 'Mary stopped smoking' and you know she never did. Simply answering 'No' would give quite the wrong impression in this situation. It would suggest that she is still smoking, not that she never smoked. A more appropriate faction would be: 'What do you mean? She never did smoke.')

The second example pair illustrates these same points. If Fred never bought any BP shares, then there is nothing for him to regret, and saying that he doesn't regret this is just as inappropriate as saint that he does. Neither affirming nor denying such a regret makes sense.

The discovery of a long and growing list of 'presupposition triggers' and the discovery that improper descriptions produce the same effects as failure of the presuppositions that come with other presupposition triggers (including *stop* and *regret*) made resistance to a presuppositional treatment of definite descriptions – be it for reasons of theoretical simplicity or economy, logical hygiene or whatever – seem futile. For even if there were a plausible non-presupposiitonal way of dealing with definite descriptions, the problems that their presuppositional treatment might raise, it had at this point become clear, have to be faced in any case in connection with all the other presupposition triggers; and the prospects of giving non-presuppositional accounts of all those other (apparently) presuppositional phenomena came to be seen as increasingly remote.

Put more succinctly, once it has become as abundantly clear that presupposition is an integral and ubiquitous feature of natural language, reasons to deny definite descriptions the status of presupposition triggers evaporate. Thus it has become common practice at least within linguistics to treat definite descriptions as triggers of presuppositions (if not necessarily as triggers of unique satisfaction presuppositions, a point that will become clear later).

So much for views on definite descriptions.The history of personal pronouns has been a very different one. When people started translating natural language into predicate logic, it was (as far as I can tell) an unwritten practice that anaphoric pronouns were translated by tokens of the variables introduced by the noun phrases to which they were anaphoric. (And that was also what we did when in the beginning of the course we used translation into predicate logic as our first method for exploring truth-conditional content of logically complex sentences. To the extent that philosophers and logicians ever went on record on this matter before the first systematic and detailed treatments of sentence-internally bound pronouns in the accounts of Chomsky and Montague, they seem to have endorsed such a view. For instance, Quine is often quoted as having said that 'pronouns are the variables of natural language' [reference]. Both Chomsky and Montague made such a view of the status and function of pronouns explicit, albeit in very different settings and in quite different ways. Montague generates English sentences with pronouns from underlying 'quasi-sentences' in which both pronouns and the noun phrases that are interpreted as their anaphoric antecedents are represented by tokens of the same variable. In Chomskyan syntax the relation between a pronoun and its antecedent is made explicit by coindexation, with the pronoun being treated in essence as trace. (The way we dealt in the first part of the course with the few pronouns considered there can be seen as an instance of this coindexation method.)

It should be clear from these observations that both at a pre-theoretical level of understanding and at the level of technical implementations of pre-theoretical intuitions pronouns and definite descriptions have fared very differently. The accounts of definite descriptions were always focused on their descriptive content, and for long at the exclusion of all else. Accounts of pronouns, on the other hand, have abstracted away from what descriptive

content they have (which admittedly isn't much, and it certainly isn't very varied) and how they succeed in selecting their antecedents has been largely left a mystery. The different approaches we have mentioned, including the one we used in Part I of this course, have tried to confine the mystery in various ways by imposing constraints on the possible antecedent of previous occurrences. But none of these attempts tell a story that was meant to extend to definite descriptions.
Explicit resistance to this perception of pronouns and descriptions as radically different started in the late seventies and donkey pronouns played a central role in it. It took two distinct forms. The first, arguably already implicit in what Geach had to say about donkey pronouns, was that donkey pronouns – 'pronouns of laziness' as Geach referred to them – were shorthands for definite descriptions: an account of donkey pronouns could be obtained by (i) turning them into definite descriptions of the appropriate kind and then (ii) analyzing the sentence in which the donkey pronouns had been thus replaced by using an account of definite descriptions that was supposed to be already in place (Evans (1980), Evans (1977)). There are two problems with this approach: (i) it isn't always clear where to get a suitable definite description that is to replace a donkey pronoun – a description that will yield intuitively correct truth conditions for the original sentence after replacement – and as Evans himself noted, one cannot hope for an algorithm that reconstructs the right pronoun on the basis of the form of the sentence in which the pronoun occurs; (ii) it isn't clear that the theory of descriptions that Evans was relying on adequately covers definite descriptions and in particular one may have doubts that the account is right for precisely those situations where a definite description occurs in a p;osition where it competes with a pronoun (in the sense that a pronoun might have been put there too, but is perhaps dispreferred because to would lead to ambiguities that the description avoids; in this sense definite descriptions often compete with donkey pronouns).[13]

The second form that the resistance against widely differing accounts of descriptions and pronouns took is found in the dissertation of Heim Heim

[13]Evans' account of donkey pronouns is known as the *E-type pronoun* account. (In this account the name used for what we have been calling'donkey pronouns' is 'E-type pronouns', a name that points not only at the phenomenon as such but also at the particular explanation of it that Evans advocated.) Another version of the E-type account can be found in the work of Cooper Cooper (1979).

(1982,1988). The *File Change Semantics* developed in this book has much in common with the account of donkey pronouns offered by DRT as we have presented it. But FCS does more in that it not only provides an account of donkey pronouns and indefinite descriptions (the anaphoric antecedents of the typical donkey pronouns) but also of definite descriptions, and in a manner that underlines the similarities between definite descriptions and pronouns, instead of treating them as birds of utterly disparate plumage. Heim also proposed, in work done at roughly the same time as her dissertation and in a closely related sprit, that definite descriptions should be treated as presupposition triggers, and that a proper account of presuppositions should involve dynamics of a sort closely akin to (and exemplified by) both FCS and DRT.

It is this last lead we will follow. We are going to treat both definite descriptions and pronouns – in fact all definite noun phrases, including also proper names and demonstratives – as presupposition triggers, which come with presuppositions to the effect that a 'referent' can be identified for them. What kind of identification is involved and by what means it can be achieved, varies from one type of definite to the next, and a proper account of definite noun phrases along these lines will have to be very careful and precise about what the different identification options are. For a pronoun the resolution of its presupposition will amount to what we have thus far been alluding to as 'pronoun resolution', and until we say exactly what that amounts to we won't have said anything of interest. For a definite description one constraint on the identification of its referent is that it satisfy its descriptive content. But again, nothing of much interest will have been said until this is spelled out. And the same goes for the remaining definite noun phrase types.

Once pronouns are treated as presupposition triggers the point of constructing DRSs top down becomes moot. This is because our treatment of presuppositions will follow earlier proposals (in particular that of Van Der Sandt, see Van Der Sandt (1992), Van Der Sandt & Geurts (1991)) according to which presuppositions are first represented as part of the DRS that is being computed from a syntactic sentence structure and will be resolved only after this 'preliminary' sentence DRS has been completed.[14] Given that the

[14]The computation of such preliminary DRSs requires a further component to the construction algorithm, which specifies which lexical items and syntactic constructions trigger

presuppositions of pronouns need to be resolved only after a representation in which all DPs have been assigned, their representing drefs and these drefs have been added to the relevant DRS Universes, the point of the top down method – the constraint it imposes on pronoun resolution by rendering certain DPs unavailable because the algorithm hasn't reached them by the time when the pronoun has to be dealt with – no longer has any applicability.

Not only is this point in favor of the top down algorithm lost when pronouns are treated as presupposition triggers. Dealing with certain other presupposition triggers actually favors the bottom up approach. Why that is so, is hard to explain at this point and an explanation has to wait until later, when a treatment of presupposition will be incorporated into CDRS construction.

From a conceptual perspective switching from a top down to a bottom up construction algorithm might be seen as taking a step back. The top down algorithm used above represents, we saw, some move in the direction of a process-sensitive approach to interpretation and meaning – not perhaps an altogether plausible one insofar as it computes the semantics from a complete syntactic sentence parse, but nevertheless paying tribute to some constraints on the order in which constituents of the represented input string can be processed. The possibility of relying on such ordering constraints is simply abandoned when the anaphoric expressions are treated as presupposition triggers, whose resolution can wait until all constituents occurring anywhere in the sentence string have been visited for their first pass interpretation. Thus, switching from top down to bottom up DRS construction amounts to going back to a more static treatment of the syntax and semantics of single sentences. (The dynamics of discourse processing, with sentences being processed in order and the earlier sentences providing discourse contexts for the later ones is of course not affected by the switch.) This also means that with regard to sentence-internal anaphora we won't be able to rely any longer on constraints imposed by processing order. *All* constraints will now have to be justified and implemented as *configurational* constraints. (So for instance we will need a different account for why (31.g) and (31.h) are incoherent.) While such a move is in the spirit of static accounts of the syntax-semantics interface where processing-based constraints were never on the cards to begin

presuppositions and how representations of these presuppositions can be computed. This component too will be put in place in due time.

with, we should not simply expunge the thought whether we might not, by proceeding in this way, give up on an advantage that eventually will prove to be indispensable after all. (Nothing in these notes will settle this question.)

6 Complex DRS Conditions

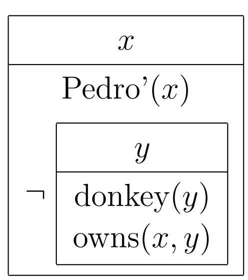
So far we have encountered just one kind of complex condition, of the form '$K_1 \Rightarrow K_2$'. But DRT needs, and has adopted, a number other complex DRS condition types. This section is just a brief overview of those that are needed to make our DRS language suitable as a logical form language for the same sentences belonging to the (informally circumscribed) English fragment covered by the MG of which we presented examples in Part I. The section will take the form of a short compendium, in which the different conditions forms are presented. For details see the somewhat longwinded but mercilessly explicit Kamp & Reyle (1993).

The first complex condition type presented here is DRT's way of representing negation. Observe that the discourse in (46) is incoherent: there is no way to give a coherent interpretation to the pronoun *it* in the second sentence.

(46) Pedro doesn't own a donkey. He beats it.

But before we get to the second sentence of (46) there is the first sentence to contend with. Whatever comes next, this sentence needs a logical form that correctly captures its truth conditions when considered on its own. To this end we introduce a new type of complex DRS condition, which can be represented schematically as '$\neg K$'. In particular the DRS for the first sentence of (46) will be as in (47).

(47)



(Exactly how (47) is obtained from (46) depends for one thing on what syntactic structure we assume for sentences containing negations. As this is a matter that should be kept for later, the question of deriving (47) as logical form of (46) must be kept until later as well.)

As for conditional DRS conditions the minimal specifications for any complex DRS condition are the following:[15]

1. its verification conditions;

2. the accessibility relations between its own constituent DRS(s) and the Universes of other DRSs within a DRS K_0 in which the given DRS condition occurs somewhere

3. conversely, the accessibility relations between its own Universe and other DRSs occurring within K_0 in the complex DRSs of which the given DRS condition could be part.

1. Verification conditions. Intuitively it ought to be clear what we should say. Suppose that '$\neg K$' belongs to the Condition Set of some DRS K', that M is a model, and that we are asking whether the function f (whose domain includes the drefs in the Universe of K' but not those in the Universe of K) verifies '$\neg K$' in M. Then

> f *verifies* '$\neg K$' in M iff there is no embedding function g such that $f \subseteq_{U_K} g$ and g verifies in M all the conditions in the Condition Set of K.

(Put more informally: f verifies '$\neg K$' if there is no way of extending f to a function which also assigns values to the drefs in U_K and verifies the conditions of K.)

[15] Another important aspect of complex DRS conditions is of course what expressions or syntactic constructions may trigger construction rules that lead to the introduction of DRS conditions of the given form. However, that is a matter of how the DRS languages to which these conditions belong are *used* in the analysis of particulate natural language fragments. Since we keep the syntax and semantics of DRS languages separate from their use in linguistic analysis, this is not the place to discuss construction rules involving particular DRS language constructs.

2. As is suggested by the incoherence of (46), the drefs in the Universe U_K of the DRS K of a condition '$\neg K$' should not be accessible at the level of the DRS K' whose Condition Set contains '$\neg K$' as a member. (For if these drefs were accessible at the level of K', then in particular the dref y in (47) would be accessible to the pronoun *it* in the second sentence, which, as we have seen, it should not be.
[16]

3. The accessibility relation between the DRS K of a condition '$\neg K$' and drefs elsewhere in the larger DRS of which the condition is part. Here the assumption made in DRT is the same that we also made in connection with conditional DRS conditions: the drefs of the DRS K' that contains '$\neg K$' as one of the members of its condition set are accessible form K.

In fact, this last specification is a special instance of a more general principle about accessibility that also applies to the complex conditions that will be described below (as well as conditional DRS conditions, for which a complete and completely explicit definition of accessibility has not so far been given) and that will now be stated once and for all, so as to avoid of repeating it for each new condition type.

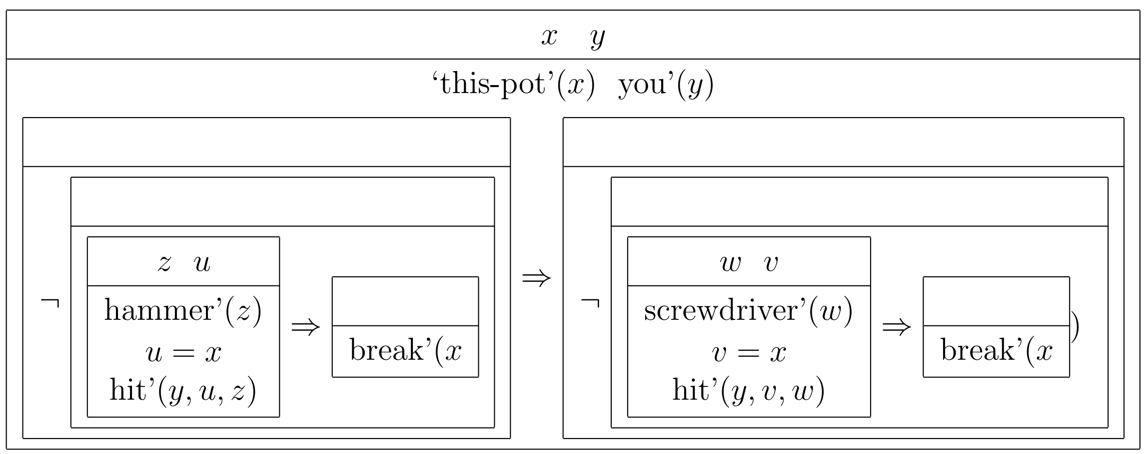
As a preparatory step towards a statement of this general principle, observe that the addition of complex conditions to a DRS language its DRSs can become structures of arbitrary complexity. This is because complex conditions contain DRSs as parts – we have seen that for conditional conditions

[16] Note that there is no prohibition against interpreting *he* in the second sentence as anaphoric to *Pedro*. (When the first sentence of (46) is followed by the sentence 'He is very lucky', the resulting discourse is unproblematic.) There could be two explanations for this: (i) in the syntactic structure for the first sentence of (46) the subject *Pedro* is outside the scope of the negation; (ii) the dref for the proper name *Pedro* goes to the main Universe no matter where the name occurs. Since the following discourse is also acceptable: 'It is not true that Pedro owns a donkey. He is very lucky.' the correct explanation is more likely to be the one that proper names always send their drefs to the main Universe; for in this last example it very much looks like the subject of the sentence is just as much in the scope of the negation as its direct object. Arguably then, the anaphoric relation between *he* and *Pedro* in 'Pedro doesn't own a donkey. He is very lucky.' is unproblematic for two separate reasons (the behavior of proper names and the scope of negation) each of which would have been sufficient on its own. (Compare also 'A farmer didn't own a donkey. He was very lucky.', which seems fine too.)

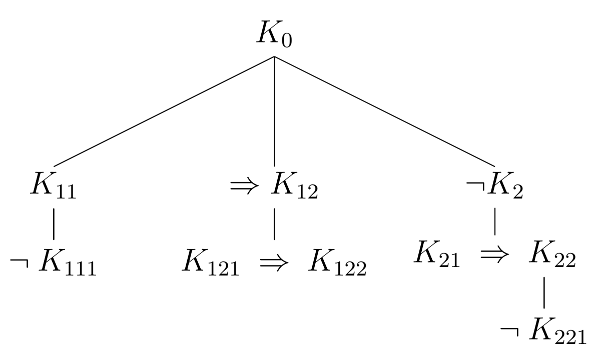
and negation conditions, but it will be equally true for the ones to follow – and these DRSs can contain complex DRSs in their Condition Sets, and so on. For instance, (48.b) gives the DRS for the sentence in (48.a), with conditional conditions embedded under negations that are in their turn embedded within a conditional DRS condition. (49) gives a schematic representation of a DRS in the form of a tree, in which the nodes are DRSs and where the mother-daughter relations mean that the daughter is a condition that belongs to the Condition Set of the mother.

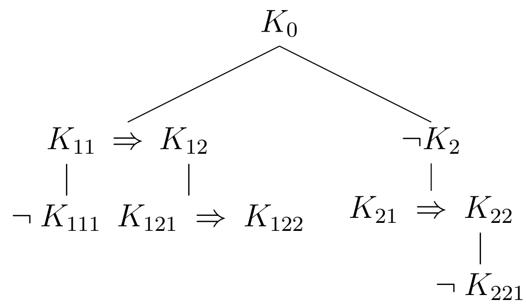
(48) a. If it is not the case that this pot breaks if you hit it with a hammer, then it is not the case that it will break if you hit it with a screwdriver.

b.



(49)





A schematic tree-like representation of a complex DRS reveals a partial order between the DRS as a whole, the DRS conditions it contains and the sub-DRSs of which those condition are made up. This partial order determines a large part of the accessibility relations between parts of a complex DRS:

(50) The drefs accessible to the conditions of a sub-DRS K' of a DRS K are those in the Universe of K' as well as all those that occur in the Universes of sub-DRSs that occur above K' in the partial order revealed by the schematic representation of K.

(N.B. the term 'sub-DRS' is used in the weak sense according to which a DRS K is itself among its sub-DRSs and, likewise, for any sub-DRS K'' of K occurring anywhere in the partial order, K'' is a sub-DRS of K''.)

Applying (50) to (49) we conclude that the drefs un U_{K_0} are accessible to conditions in its own Condition Set, which includes the conditions in K_{11}, those in K_{111}, those in K_{12}, those in K_{121}, etc. Likewise the drefs in $U_{K_{12}}$ are accessible to the conditions in K_{12}, those in K_{121} and those in K_{122}; and so on.

This is not yet a complete definition of accessibility, but it comes fairly close. What is missing from it are specifications of accessibility between DRSs that are part of complex conditions that are made up of more than one DRS. Conditional DRS conditions are of this sort.There the stipulation was that for a condition $K_1 \Rightarrow K_1$ the drefs in U_{K_1} are accessible to the conditions in K_2, but not conversely.

This specification of the accessibility relations between the antecedent DRS K_1 and the consequent DRS K_2 of a conditional DRS condition $K_1 \Rightarrow K_2$

fully fix the accessibility relations for a DRS language in which conditional DRS conditions are the only compile DRS conditions, provided that we take the accessibilities licensed by this stipulation and the general principle in (50) to state between them all positive instances of accessibility. These clauses constitute a complete specification of the accessibility relation for this restricted DRS language provided we understand them in an exhaustive sense: a dref in a DRS Universe U_K is accessible from another DRS K' (where K and K' are both part of the same DRS K_0) if and only if the relation holds by virtue of these clauses.[17]

When a new type of complex DRS condition is added to a given DRS language and this condition involves just one constituent DRS (as in the negation-conditions just introduced), then nothing needs to be added to the defining clauses for the accessibility relation. Clause (50) will take care of all new accessibility questions that adding the new condition may give rise to. But when a newvtype of condition is added that has two or more constituent DRSs, then a new specification is needed which fixes between these condition constituents. (Along the lines of the special specification for conditional DRS conditions: the drefs in the Universe of the antecedent DRS are accessible to the consequent DRS but those in the Universe of the consequent DRS are not accessible to the antecedent DRS.

6.1 Disjunctive DRS Conditions

Our next complex DRS condition type is used to represent disjunctions. Not surprisingly such DRS conditions are represented in the form '$K_1 \vee K_2$'. By way of an example, the DRS in (51.b) is the logical form for the sentence in (51.a).

(51) a. Pedro owns a donkey or he owns a mule.

[17] This is a general method for defining relations: One gives one or more clauses specifying conditions under which the relation holds and then says that this covers all cases of the relation.

b.

<table>
<tr><td colspan="3">x</td></tr>
<tr><td colspan="3">Pedro'(x)</td></tr>
<tr>
<td><table><tr><td>y</td></tr><tr><td>donkey'(y
owns(x, y)</td></tr></table></td>
<td>$\vee$</td>
<td><table><tr><td>z</td></tr><tr><td>mule'(z)
owns(x, z)</td></tr></table></td>
</tr>
</table>

The verification conditions for DRS conditions replicate the truth table for inclusive disjunction (according to which $A \vee B$ is true iff either one of A and B is true or both are true). That is:

(52) an embedding function f versifies $K_1 \vee K_2$ in a model M iff
there is an embedding function g such that $f \subseteq_{U_{K_1}} g$ and g verifies the conditions in the Condition Set of K_1 or
there is an embedding function h such that $f \subseteq_{U_{K_2}} h$ and h verifies the conditions in the Condition Set of K_2 (not excluding the possibility that there both is such a function g and such a function h).

To my knowledge there is no perfect agreement on the topic of accessibility between the left hand side and right hand side DRSs of a $\vee$ condition. Here we state the assumption made in the most widely known and available version of DRT (the one presented in Kamp & Reyle (1993)). According to this assumption, neither the drefs in the left hand DRS are available to the condition of the right hand DRS nor vice versa.

A curious (if perhaps not altogether surprising) fact about accessibility within disjunctive DRS conditions is that drefs from a *negated* first disjunct (the DRS on the left of $\vee$) are accessible to the second disjunct. The point is illustrated by 'Partee's bathroom example' given in (53).

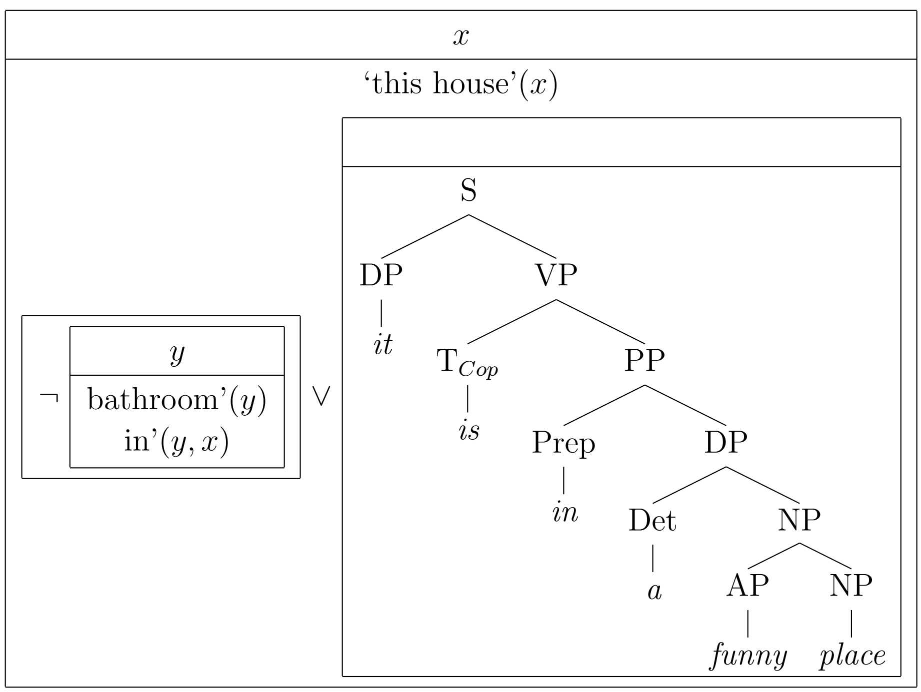
(53) Either there is no bathroom in this house or it is in a funny place.

In (53) the pronoun in the second disjunct can be construed as anaphoric to the DP *no bathroom* in the first disjunct. If this construal is to be explicated along the lines we have been following – i.e. via an anaphoric relation between the drefs representing pronoun and antecedent – then the dref representing *no bathroom* must be accessible from the second disjunct.

If, as we have just assumed, the drefs from the Universe of the first disjunct are not accessible from the second disjunct, then it might seem strange that a dref that is inside a negated DRS condition inside the first disjunct would be accessible. But there is a plausible explanation for why this should be possible. Note that a disjunction 'A or B' is logically equivalent to 'either A or (not A and B)'. For the purposes of explaining why 'bathroom anaphora' is possible it is enough that such a reanalysis is possible when it is needed, and that wanting to interpret a pronoun in the second disjunct of a disjunction as anaphoric to a DP in a negated first disjunct is the kind of need that triggers it.

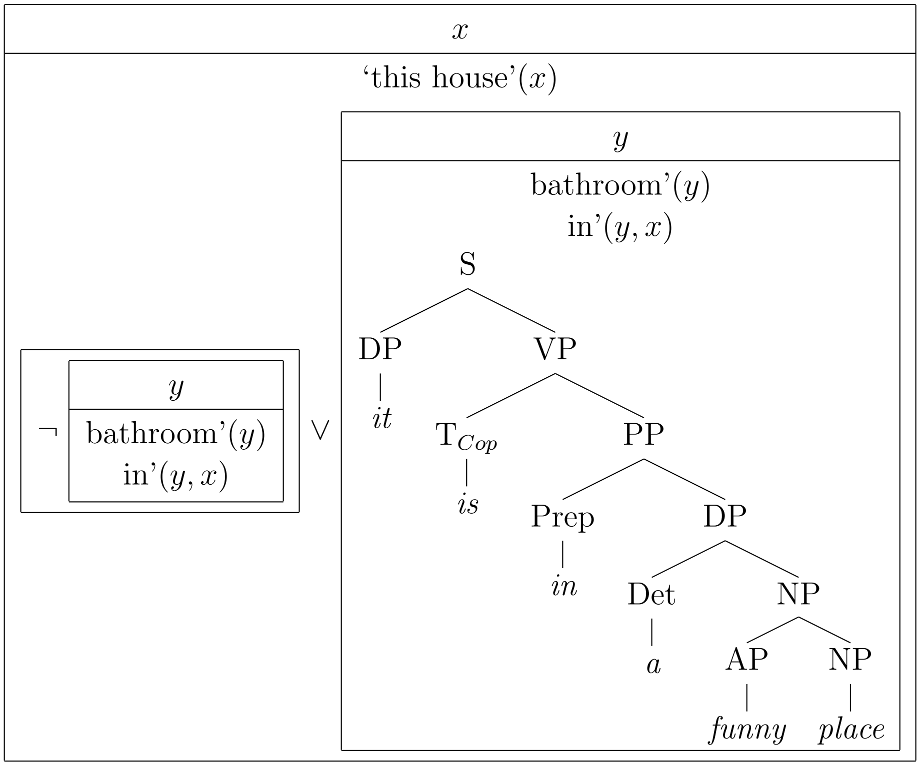
In DRT terms such a reanalysis comes to this: After the disjunctive DRS condition $K_1 \vee K_2$ has been introduced and a negation condition $\neg K_1'$ has been established as member of the Condition Set of K_1, then a copy of K_1' may be merged with K_2[18]. For the case of (53) this operation takes the form displayed in (54), with (54.a) showing the construction stage of the DRS just before the operation and (54.b) its result. The final result is (54.c).

(54) a.

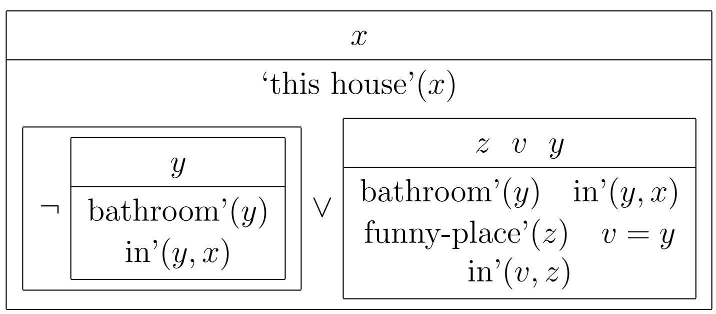


[18]provided none of the conditions in the Condition Set of K_1' contains a dref belonging to U_{K_1}

b.



c.



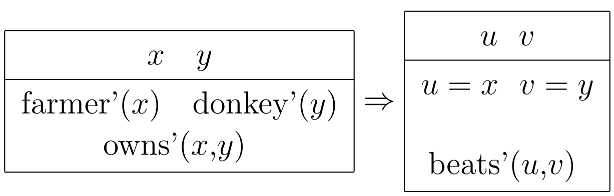
Whether this is the right account for a sentence like (53) may be a matter for further debate. But assuming that it is, it points at yet a further dimension to DRS construction: the possibility of meaning preserving operations on DRSs in the course of their construction. There are various other examples that suggest the need for such operations, that serve to render drefs accessible to pronouns that wouldn't be accessible without those operations. (For details see Krahmer (1995).) A further point of debate is how many operations of this kind will be required to make drefs available in configurations where in the absence of further provisions our definition of accessibility excludes them.

6.2 Duplex Conditions

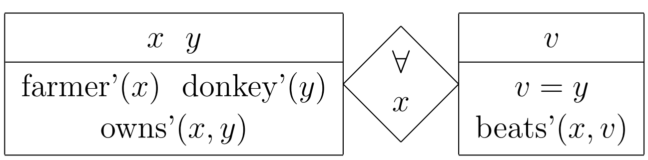
When discussing Cooper's donkey sentence (29), repeated below, we noted that the donkey sentences (1.c) and (1.d), also repeated below, can be represented by the same logical form: a conditional DRS condition with discourse referents for farmer(s) and donkey(s) in the Universe of the Left hand side DRS. But it was also noted at that point that subsequent to the earliest formulations of DRT a different representation format was adopted and that since then this format has been used more commonly. (55) gives the conditional DRS condition we used to represent (1.d). (cf. (30), which has (55) as the only member of its Condition Set.)

(29) Everyone who had a quarter in his pocket put it in the meter for the parking space where she had put her car.

(1.c) If a farmer owns a donkey, he beats it.
(1.d) Every farmer who owns a donkey beats it.

(55) 

(56) shows the new format for the representation of (1.d).

(56) 

The verification conditions of (56) are the same as those for (30.b) (as of course they should be):

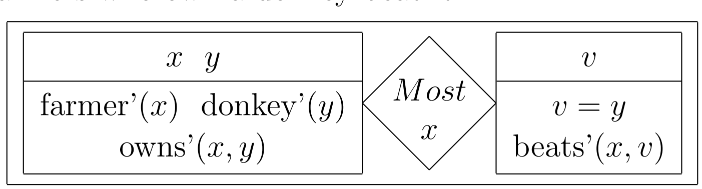
(57) f verifies the DRS condition in (56) in M if every extension g such that $f \subseteq_{\{x,y\}} g$ which verifies the conditions in the left hand DRS can be extended to a function h such that $g \subseteq_{\{v\}} h$ which verifies the conditions in the right hand side DRS.

Why introduce a separate format for the representation of universal quantification if conditional DRS conditions, which are needed in any case, do just as well? The reason is that universal quantifiers pattern with other quantifiers which cannot be represented by conditional conditions. This is true in particular for quantifiers that take the form of quantifying DPs, in which the *quantificational force* is expressed by the determiner (in the way in which the determiner *every* expresses universal quantificational force). Here are some examples of quantifying DRSs in English: *many a farmer*, *no farmer*, *at most one farmer*, *at least one farmer*, *exactly one farmer*, *several farmers*, *most farmers*, *many farmers*, *few farmers*,*two farmers*, *three farmers*, .., *at most/at least/exactly two farmers*, .., *between five and ten farmers*, *half of the farmers*, *more than/less then/at least/at most half of the farmers*. But as you are likely to expect at this point there are many, many more. In fact the set of complex quantificational determiners is open-ended and any generative syntax will treat it as infinite. Many of these DPs are plurals, which is one reason why we have stayed away from them so far (and will for the remainder of the course. For a detailed discussion of issues raised by plural noun phrases in English, as part of a survey of plurals in general, within the context of DRT see Kamp & Reyle (1993), Ch. 4.)

If we set the differences between plural and singular DPs aside, their syntactic and semantic behavior is much the same, and it seems natural therefore to adopt a uniform format for the contributions they make to the semantic representations of the sentences containing them. Conditional DRS conditions are unsuitable for this purpose because their semantics makes them adequate representations of quantifiers with universal force, the force expressed by the determiners *every*, *each*, *all*. But consider, as an example of a determiner with a non-universal quantifying force, the word *most*. The sentence in (58.a) cannot be represented with the help of a conditional DRS condition, for that would assign it the truth conditions of (1.d) and these are obviously not right for (58.a). (58.b) is the representation for (58.a) using the new format.

(58) a. Most farmers who own a donkey beat it.

b.



DRS conditions of the form displayed in (56) and (58.b) are known as *duplex conditions*. The quantificational force of a duplex conditions is indicated by the operator symbol inside the central diamond. The other symbol inside the central diamond is the dref that is *bound* by the central diamond's operator, in essentially the same in which the variable x of PC is bound by the universal quantifier in a PC formula like $(\forall x)(\text{human}(x) \rightarrow \text{mortal}(x))$.

It is the quantificational force operator in the central diamond of a duplex condition which determines its verification conditions. These verification conditions are a combination of what we have seen in connection with (56) (and conditional DRS conditions like (30)) on the one hand and on the other the quantificational force denoted by the operator. When the force is universal, the verification conditions are those exemplified in (56.1). When the quantificational force is different, as in (58.b), then the verification conditions will have to be different too. Roughly speaking, (58.b) is true in a model M iff the set of individuals **d** in M that satisfy both the left hand side DRS and the right hand side DRS when they assigned to the dref x constitutes more than half of the set of **d**'s that satisfy the DRS on the left. We can make this formally precise as in (59).

(59) f verifies the DRS condition in (58) in M if the following sets X and Y stand to each other in the relation $|Y| > 1/2.|X|$[19], where X and Y are defined as follows:

X = the set of individuals **d** from U_M such that there are one or more extensions g of f such that $f \subseteq_{\{x,y\}} g$, g verifies the conditions in the left hand DRS and $g(x) = \mathbf{d}$.

Y = the set of individuals **d** from U_M such that there are one or more extensions g of f such that $f \subseteq_{\{x,y\}} g$, g verifies the conditions in the left hand DRS, $g(x) = \mathbf{d}$ and g can be extended to a function h such that $g \subseteq_{\{v\}} h$ which verifies the conditions in the right hand side DRS.

The investigation of the range of logically possible quantificational forces – of their mathematical properties and of the linguistically motivated ques-

[19]by $|Y|$ we mean the *cardinality* of the set Y, that is the number of elements in the set Y. (What this comes to in cases where, where Y is infinite can be found in any proper introduction to set Theory.)

tions which of those are expressible in which natural languages, or in which particular forms that certain natural languages make available for such purposes – has become a topic in its own right, a subfield on the borderline between linguistics and logic. This subfield is known as *Generalized Quantifier Theory*. Some kind of introduction to Generalized Quantifier Theory has become a fixture of introductions to formal semantics, for one thing because of their importance for the truth-conditional semantics of sentences with quantificational elements, and for another because of the crispness and, in some instances, the mathematical sophistication of Generalized Quantifier Theory, which makes it an attractive battlefield to the logically trained and minded.

So there ought to be something about Generalized Quantifier Theory in the present introduction to semantics too. But there isn't time for everything, and as the present course developed, it became increasingly difficult to see how to fit a discussion of Generalized Quantifiers in. As a kind of minimal substitute, the next and final section of this document gives a maximally bare bones survey of GQT's central concepts and ideas.

6.3 Generalized Quantifiers

One of the things that the logically inspired noted early on when they started to apply formal logic to the analysis of natural language is that there is a tension between the quantifiers $\forall$ and $\exists$ of PC and the quantifiers found in natural languages. The quantifiers of PC are 1-place operators which apply to one formula at a time, transforming that formula into another formula. But the quantifiers we find in natural languages typically do not work that way. This is so in particular for the quantifying expressions we have been looking at – DPs that begin with a quantifying determiner. All these quantifiers can be regarded as operating on two arguments rather than one –they are *2-place*, not *1-place* operators. This can be clearly seen for the two examples of quantified sentences so far considered, the *every*-sentence (1.d) represented in (56) and the *most*-sentence (58.a) represented in (58.b). In each of these representations the DRS condition that represents the quantification consists of three parts – the central diamond and the two DRSs to its left and right. The two DRSs can be regarded as the two 'arguments' to the quantifier occupying the central diamond.

One question that arises for the representation of quantification by duplex conditions (and likewise for the representations of universal quantifications via conditional DRS conditions that we used initially) is what material goes into which of the two boxes. When a quantification-representing DRS condition is triggered by a quantifying DP, it is always clear which part of the clause containing the DP goes into the left hand side DRS (the so-called *restrictor DRS* of the duplex condition)) and which part goes into the right hand side DRS (the *nuclear scope* DRS): the NP that is the sister of the quantifying determiner goes into the restrictor DRS and the 'remainder' is inserted into the nuclear scope DRS, where the 'remainder' is the structure containing the DP as syntactic argument at the point where the construction deals with it and in which the DP has been replaced in its argument position by the dref that is chosen to represent the DP (in the two examples we have been looking at: the dref x). With other quantifying expressions of natural languages – for instance, *quantifying* adverbs like *always*, *sometimes* or *often* – the question what goes into the *restrictor DRS* and what into the *nuclear scope DRS* tends to be less straightforward and in some cases the factors guiding this decision can be quite complex. But we won't go into these complications here.

The second general question that a analysis of quantifiers as operators must answer is: 'What do these operators operate on?' The answer to this question is implicit in the treatment we adopted of quantifying DPs in the version of Montague Grammar offered in Part I. Recall that DPs were treated as second order predicates, whose denotations (in any model M) are (characteristic functions of) sets of (characteristic functions of) individuals. To obtain the semantics for the quantification, this denotation of the quantifying DP is applied to the denotation of the lambda term associated with the DP's sister node (the lower S node of the DP adjunction via QR); this lambda term is obtained by lambda abstraction over the logical form of the lower S node with respect to the variable 'bound' by the DP. The denotation of this lambda abstract is (the characteristic functions of) a set of individuals and applying the denotation of the DP to it results in a truth value. The denotation of the DP, on the other hand, is obtained by applying the denotation of the determiner to that of its sister NP. The latter denotation is also a (characteristic function of a) set and the denotation of the DP is obtained by applying the denotation of the determiner to this set. So the denotation of the determiner is a function which maps sets to functions from sets to truth values. As we have seen, such 'curried' functions can also be regarded as functions from pairs of sets $<X, Y>$ to truth values: the truth value for the pair is obtained by first mapping X to a possible DP denotation and then applying that to denotation to Y. But a function from pairs of sets to ruth values is just the characteristic function of a sets of pairs, and a set of pairs of entities of a certain sort is just a relation between entities of that sort. So in essence the denotations of determiners are relations between sets.

It is this conception of quantifiers – as relations between sets – that is the basic intuition underlying Generalized Quantifier Theory: Generalized Quantifier Theory is the study of set relations, on the one hand of their abstract mathematical properties and on the other of language-related questions which set relations can get expressed in certain natural languages and in what forms and which might be universally expressible. Generalized Quantifier Theory has developed into a field of investigation with its own identity, situated on the border between pure logic and linguistics. It is has proved to be a rich and fruitful field, which derived its popularity, for one thing because of its importance for a proper understanding of the truth conditional implications for quantifier constructions in natural language, but also because of its appeal to the logically minded – crisp definitions and theorems that can be rigorously derived from those definitions and that reach from the comparatively simple to the mathematically sophisticated.

The observations made above about the denotations that are assigned to quantifying determiners in the version of Montague Grammar presented in Part I stressed the fact that on such an account the denotations of determiners are in essence relations between sets. So it should be possible to identify the semantics of particular determiners as particular relations between sets, and indeed that is one of the things that the GQT has been used for. For instance, the set relation that defines the semantics of the determiner *every* is the relation between sets that holds between X and Y iff $X \subseteq Y$. Likewise, the determiners *a* and *some* – assuming that it is right to treat them as quantificational determiners that express existential quantification – are defined by the set relation that holds between X and Y iff $X \cap Y \neq \emptyset$. And if we accept what we said above about semantics of *most*, then the denotation of *most* is defined by the relation that holds between X and Y iff $|Y \cap X| > 1/2.|X|$.
After what has been said at this point about generalized quantifiers and about

the verification conditions of duplex conditions, the relationship between the two should be fairly clear. The place where this relationship became visible most directly was in statement (59) of the verification conditions for the duplex condition in (58). In (59) the sets X and Y are explicitly defined and then the relation is stated that must hold if the duplex condition (58) is to count as verified by f in M. The verification condition for universal duplex conditions, like the one in (56), could also be stated in this format, in which the sets X are defined as in (59) and then the relation between those sets is stated that must hold iff the DRS condition in (56) is to count as verified by f in M. When the verification conditions for (56) are given in this format, then the relevant relation between X and Y must of course be that $X \subseteq Y$.

There is however one point of tension between the analysis of quantifiers via duplex conditions and the treatment they get in the standard versions of Generalized Quantifier Theory. It can be seen when we compare the statement of the relation between X and Y in (59) – $|Y| > 1/2.|X|$ – and the GQT definition of the relation we gave above, viz. $|Y \cap X| > 1/2.|X|$. The reason for this discrepancy has to do with the fact that the duplex condition approach imposes on all quantifiers that are represented by duplex conditions the constraint that the second set Y is a subset of the first set X. This constraint follows from the way in which the sets X and Y are defined for duplex conditions: the constraints on Y always include the constraints on X, so anything that passes the conditions for membership in Y automatically also passes the constants imposed on membership in X.

The condition '$Y \subseteq X$' can be regarded as expressing a property of generalized quantiers: a generalized quantifier R has the property if the condition holds for all its members: if for any X and Y XRY then $Y \subseteq X$. In GQT this property is known as *conservativity*. Note that when conservativity holds, the two statements of the denotation of *most* – $|Y \cap X| > 1/2.|X|$ and $|Y| > 1/2.|X|$ – are equivalent. Likewise, the definition given for the existential quantifier – the condition that $X \cap Y \neq \emptyset$ – can be simplified in the presence of conservativity, viz. to $Y \neq \emptyset$; and likewise for other definitions.

In Generalized Quantifier Theory conservativity is not taken for granted. In principle two sets X and Y can stand in a quantifier relation R without it being the case that $Y \subseteq X$. The question can then be asked which natural language quantifiers are conservative, and whether perhaps all of them are, or if not all perhaps all that are expressed in a certain way, e.g. by quantifying determiners. In this respect the duplex condition treatment of DRT might seem to be at a disadvantage vis-á-vis GQT inasmuch as it is forced to make an assumption that in GQT isn't decided by the logic as such and can be left to be decided at the level of linguistic application.

But we should not be too hasty to condemn DRT on this account; for the reason why its architecture imposes conservativity on the quantifiers represented by duplex conditions is that it appears to be a linguistic fact that the information established by the restrictor material of a quantifier can be used in the interpretation of its nuclear scope (and in particular that drefs established through the interpretation of material in the restrictor can be used as antecedents for pronouns occurring in the nuclear scope. Thus it rather looks as if conservativity is built into natural language quantification; or at least that it is built into the quantification that can be expressed by determiners. The circumstance that it has so far proved very hard to find counterexamples to the claim that all natural language quantifiers are conservative, and that none of the counterexamples thus far proposed seem conclusive, suggests that conservativity is a part of the grammar and not a property that most or all quantifying expressions found in natural languages happen to have.

So much for Generalized Quantifier Theory. It hasn't been much. If you really want to find out more, consult the copious literature.

References

Chierchia, G. (1995), *Dynamics of Meaning: Anaphora, Presupposition, and the Theory of Grammar*, University of Chicago Press, Chicago.

Chomsky, N. (1982), *Some Concepts and Consequences of the Theory of Government and Binding*, MIT, Cambridge.

Cooper, R. (1979), The interpretation of pronouns, *in* 'Syntax & Semantics, vol 10', Academic Press, pp. 61–92.

Evans, G. (1977), 'Pronouns, quantifiers and relative clauses', *Canadian Journal of Philosophy* **7**, 467–536.

Evans, G. (1980), 'Pronouns', *Linguistic Inquiry* **11**, 337–362.

Geach, P. (1962 (Third revised edition: 1980)), *Reference and Generality : An Examination of Some Medieval and Modern Theories*, Cornell University Press, Ithaca.

Heim, I. (1982,1988), *The Semantics of Definite and Indefinite Noun Phrases*, Garland Press.

Kamp, H. (1981), A theory of truth and semantic representation, *in* J. Groenendijk et al., eds, 'Formal Methods in the Study of Language', Mathematisch Centrum, Amsterdam.

Kamp, H. & Reyle, U. (1993), *From Discourse to Logic*, Kluwer.

Krahmer, E. (1995), Discourse and presupposition, PhD thesis, Tilburg University.

Russell, B. (1905), 'On denoting', *Mind* **14**, 479–493.

Van Der Sandt, R. (1992), 'Presupposition projection as anaphora resolution', *Journal of Semantics* **9**, 333–377. Special Issue: Presupposition, Part 2.

Van Der Sandt, R. & Geurts, B. (1991), Presupposition, anaphora, and lexical content, *in* O. Herzog & C.-R. Rollinger, eds, 'Text Understanding in LILOG', Springer.