Modeling Unicorns and Dead Cats: Applying Bressan's ML$^\nu$ to the Necessary Properties of Non-existent Objects

Abstract

Should objects count as necessarily having certain properties, despite their not having those properties when they do not exist? For example, should a cat that passes out of existence, and so no longer is a cat, nonetheless count as *necessarily* being a cat? In this essay I examine different ways of adapting Aldo Bressan's ML$^\nu$ so that it can accommodate an affirmative answer to these questions. Anil Gupta, in *The Logic of Common Nouns*, creates a number of languages that have a kinship with Bressan's ML$^\nu$, three of which are also tailored to affirmatively answering these questions. After comparing their languages, I argue that metaphysicians and philosophers of language should prefer ML$^\nu$ to Gupta's languages in most applications because it can accommodate essential properties, like being a cat, while being more uniform and less cumbersome.

1 Introduction

Aldo Bressan, working in contact with Carnap, but unaware of parallel developments in modal logic due to Kripke, presented a uniform, higher-order, quantified modal language, ML$^\nu$, in (Bressan, 1972). Of the major systems treated today, it is most like Montague's *IL*. Later Anil Gupta, working under Nuel Belnap and Rich Thomason, developed a series of languages in the spirit of ML$^\nu$ that distinguished common nouns from predicates, both syntactically and semantically (Gupta, 1980).[1]

Both Bressan and Gupta develop their languages to accommodate the possibility of some objects not existing in a case. In doing so, Gupta addresses an issue for how to develop the semantics of necessity that Bressan overlooks.[2] Should an object that falls under a predicate in every case in which it exists, but does not fall under it in some case where it does not, count as necessarily falling under that predicate? For example, take the cat Madrone. Madrone has passed away. Arguably, Madrone no longer exists, and non-existent or merely possible cats are not cats. Should Madrone, who was a cat in those cases in which he existed, count as necessarily a cat? Similarly, take Pokey the unicorn. In the present world Pokey doesn't exist but in all of the worlds where Pokey does exist, she is a unicorn. Is Pokey necessarily a unicorn, despite the empty extension of 'unicorn' in this world?

In different contexts we may want languages that answer these questions differently, so I will be developing the semantics of ML^{ν} to accommodate both an affirmative and negative answer.[3] Specifically, in section 7, I look at two options for developing ML^{ν} so that objects which pass out of existence or are merely possible in some cases can count as necessarily falling under a predicate, as they can in Gupta's L_3. Attempting to think through how to adjust ML^{ν} so that it preserves the same intuitions as L_3 will clarify Gupta's motivations for how he set up his languages. In section 8, I will urge that preserving those intuitions isn't worth the cost in usability and uniformity, since ML^{ν} can pre-

[1] For an introduction to ML^{ν} that is more accessible than Bressan's book, see (Belnap, 2006). Since submitting this essay for review, Belnap and Müller have published two essays self-consciously developing the first order fragment of ML^{ν} (Belnap and Müller, 2014b) & (Belnap and Müller, 2014a). In the first of these they do an especially nice job of explaining the virtues of ML^{ν}, and their own Case-Intensional First Order Logic (CIFOL), in comparison to other quantified modal logics. I refer the reader to this essay for a more comprehensive discussion of the related languages than I will provide.

[2] Bressan first suggests how his account can be modified to deal with objects that may not exist in all cases in (Bressan, 1972, p. 89). He amends this suggestion in (Bressan, 1993, p. 372). In both of these discussions he passes over our problem for the semantics of necessity in silence. Gupta, however, struggles mightily with the issue in developing his L_3. It is worth noting that Belnap and Müller end up treating the semantics of necessity in CIFOL in a very similar manner to the way that I suggest we should treat it in ML^{ν} (Belnap and Müller, 2014b, esp. 419).

[3] Following Bressan, I will be using the term 'case' in order to stay neutral between interpreting modal indices either as worlds or times. And although, of course, there are quite important differences that arise when interpreting modal indices in different ways, as much as possible I will be attempting to work at a level of abstraction that is above these.

serve a notion of essential properties, which will be properties a thing has in every case that it exists, without modifying the semantics for necessity. In prior sections, I will informally introduce the languages, give some of the semantics for ML^{ν}, discuss a somewhat analogous issue that arises with quantification over non-existent objects, and then introduce the notions of an absolute and quasi-absolute concept.

2 A preliminary comparison of languages

Before I begin my formal introduction of ML^{ν}, in this section I will informally introduce what is distinctive about Bressan's and Gupta's languages. Here we will see why our problem for the semantics of necessity does not tend to be thought of as an interesting problem for standard modal logics. We will also get a better sense of the contrasting motivations behind Bressan's and Gupta's languages, as well as a preliminary sense of their respective virtues.

A main advantage of intensional modal logics over other modal logics is that they can grant explicit logical criteria for when a predicate will provide suitable principles of trans-case identity. Predicates that satisfy these criteria will allow one to trace the objects under them across times or worlds.[4] Standardly, this kind of work is done through the structure of the model—the worlds and their domains. For example, a ship d in a world w, is the same ship as a ship d' in a world w', if and only if $d = d'$. In this way, principles of trans-case identity are standardly taken to be vacuous, because we have the same object in different cases. In Bressan's and Gupta's languages, however, these principles will not be vacuous: a could be Madrone w, while b is him in w'. Those predicates like 'is a cat' that allow one to trace the same individual from case to case will have distinctive logical properties. In ML^{ν} these predicates will primarily be what Bressan calls 'absolute concepts,' while in Gupta's languages they will primarily be 'substance sorts,' and both are notions I will return to below.[5]

[4] Gupta presents an argument for the distinct logical treatment of common nouns that bring with them such criteria of identity in (Gupta, 1980, esp. ch. 1, §5). Bressan offers some assessment of this argument in (Bressan, 1993, §N6). And McCawley has a nice brief discussion in McCawley (1982).

[5] Although it will be predicates that provide principles of trans-case identity in these languages, we need not take this to be an implicit endorsement of 'contingent' or 'relative identity.'

Montague's *IL*, despite being just as mathematically powerful as ML$^\nu$ or Gupta's languages, lets the model and rigid designation do the job of providing principles of trans-case identity. This is why I will not be discussing it at length. Languages that build principles of trans-case identity into the model structure can deal with the puzzle I introduced for the semantics of necessity by excluding individuals that do not exist in a case from that case's domain.[6] But because much of the appeal of ML$^\nu$ and Gupta's languages stems from their power to avoid doing this metaphysical work by relying on the model structure, for both it is useful to have a treatment of necessity and non-existence that does not rely on variable domains.

Gupta claims two main philosophical advantages for his languages over ML$^\nu$. The first is that while in ML$^\nu$ the main class of objects are Carnapian individual concepts—functions that take one from a world to an extension—Gupta's main class of objects are extensional individuals. Since Quine the (perhaps now fading) received dogma has been that in the first instance we should model objects through extensions at a world or case, because these correspond best to this or that physical object. Bressan's own motivations for constructing ML$^\nu$ stem from physics, and the attempt to model dispositions and powers that may not be actualized.[7] However, if we reject Quine's claim that such modal features of objects are not susceptible to logical treatment,[8] the motivation for not taking our primary class of objects to be functions from cases to extensions becomes less clear.

Gupta's second main philosophical motivation in setting up his languages is capturing both semantically and syntactically the differences between common nouns, intensional predicates, and extensional predicates (Gupta, 1980, p. 1-3). All three of these are standardly rendered as predicates with principles of application that determine their extension at each world (and time). Within

Rather, we just need to think common nouns like 'cat,' 'horse,' or 'person,' have distinctive semantic properties that distinguish them from predicates like 'red,' or 'smooth' that do not, and that this difference is worth modeling in our language.

[6] cf. e.g. (Cresswell and Hughes, 2004, ch. 15 & 16)

[7] Specifically, as a practicing physicist, Bressan wanted to capture Mach's definition of "mass" in terms of possible experiments.

[8] (Quine, 1980). Another major motivation for this extensionalism seems to have come from reading intension as on the side of the mental. Although there was plenty of historical warrant for this, it is clear that Bressan's intensions are patterns of extensions-at-a-case, as the case varies. There is nothing mental about them.

these, however, common nouns are special in that they also have principles of identity, which allow us to trace individuals from case to case. While extensional predication (like 'John finds a unicorn') and intensional predication (like 'John seeks a unicorn') differ in that the former predicates are true of mere extensions, while the latter are true of individual concepts.

Both Bressan's ML^{ν} and Gupta's languages mark these differences semantically. Gupta goes farther, however, and distinguishes them syntactically as well.[9] This is closely related to Gupta's extensionalism. Because the objects of Gupta's languages are extensional individuals, when assessing extensional predicates at a world we just look at the objects they are true of at that world. The syntax for assessing intensional predicates, however, relies on variables that are assigned to a pair consisting of a 'sort' and an object in a world's extensional domain, which together fix how to track this object to other worlds, and allow the consideration of these counterparts when assessing intensional predicates. Common nouns, on the other hand, indicate the 'sort' of thing an extensional individual is, and so are not just assessed intensionally, but determine the counterparts of the thing in question.[10]

In order to capture syntactic differences between these three grammatical roles, Gupta sacrifices the uniformity and simplicity found in ML^{ν}, making his languages significantly more cumbersome. Considering that in ML^{ν} there are still significant semantic differences between each of these roles, however, we may wonder with Bressan whether Gupta's comparatively unwieldy adaptation is worth it (Bressan, 1993, p. 371). Without developing a response to our issue for the semantics of necessity when objects may not exist in some cases, however, ML^{ν} remains significantly underdeveloped as compared to Gupta's languages, and so it is to this that we now turn.

[9] Bressan puts the point that common nouns, extensional predicates and intensional predicates will be treated in a syntactically uniform way in ML^{ν} rather strongly by claiming that "no *a priori* distinction is made in ML^{ν} between common nouns and 1-ary predicates" (Bressan, 1993, p. 351). Bressan treats all predication as intensional and there are not different semantic rules for assessing extensional and intensional predicates. Nonetheless, extensional predicates are distinctive, since their truth in a case only depends on the extension in that case of the individual concepts falling under it, not the extensions of these concepts in other cases. This allows Bressan to preserve a uniform treatment of predication while still capturing significant semantic differences for extensional predication. (The relevant technical details for understanding how this works will be presented in the next section.)

[10] I give a formal statement of these differences in the Appendix on Gupta's languages.

3 Basic semantic elements

I will briefly lay out the basic elements of the semantics of ML^{ν} before turning to the question of how the introduction of non-existent objects affects the semantics of quantification, necessity and predication.[11] A *Bressan Model* $M = \langle S, I \rangle$ combines a Bressan structure $S = \langle \nu, D, \Gamma, * \rangle$ with an interpretation-function I; where ν is the number of basic types and $Type_{\nu}$ is the set of types of ML^{ν}; D is the domains-function defined on $Type_{\nu}$ such that D_{τ} is the domain of extensional individuals of type τ; Γ is the set of all cases γ; and $*$ is a function defined on $Type_{\nu}$ such that $*_{\tau} \in D_{\tau}$ for every domain D_{τ}. We will think of $*_{\tau}$ (or just $*$ when I am, as will be usual, suppressing types) as the non-existent entity of type τ.[12]

An interpretation-function I is a function that maps all of the atomic expressions ($A \in Atom$) of the language to objects of the appropriate type.[13] Except at the bottom (τ_n where $n \in \{0\} \cup \bar{\nu}$, 0 is the type of truth values, and $\bar{\nu}$ is the set of basic types) these objects are intensional. Accordingly, the mapping is usually from the atomic expressions of each type to intensional objects of that type.

[11] In doing this I will not use the form of ML^{ν} presented in (Bressan, 1972), but the one Belnap sketches in (Belnap, 2006). For a more detailed explanation of the type hierarchy and the basic elements of the semantics than I give here, see (Belnap, 2006).

[12] Bressan, Gupta, and Belnap, all designate non-existence by having a single non-existent entity of each type. A few other ways of representing non-existence seem available to us. First, we could leave individual concepts undefined in the cases in which they don't exist. Second, we could have lots of non-existent entities—most intuitively one for every possible object. Using the second of these would have the advantage of trivializing the problems that arise for Bressan with the introduction of non-existence. Since that would make for an uninteresting essay, and mean accepting a vast menagerie of non-existing things into our ontology, I will leave it aside. The first option however, will come up again below. (Gupta considers these two options briefly (Gupta, 1980, p. 68).)

Montague treated terms like *John* as denoting not an individual, John, but a set of John's properties, where properties are intensional, mapping indices to sets of extensions. This Russalian treatment made a corresponding treatment of non-existent objects natural. Accordingly, when translating a sentence like 'John seeks a unicorn' into *IL*, 'a unicorn' will be treated as a property of properties. (For a nice explanation see (Dowty et al., 1981, ch. 7, §V).) This kind of approach means that one can avoid countenancing non-existent objects in one's ontology, give a decent rendering of the sentence, and get close to the specificity that one could have by allowing a different non-existent entity for every merely possible object. This kind of strategy is very different than the ones we will be pursuing, in part because we will treat singular terms as designating individuals in a Fregean manner.

[13] This means that what A ranges over will shift according to application—variables and constants will be atomic expressions of one type, whereas truth values will be another, predicates taking constants as arguments another, etc.

Definition 1 (Intensional Domain)**.** Call this set of objects the *intensional domain* of type τ, and symbolize it as: $Int_\tau =_{df} (\Gamma \mapsto D_\tau)$ (or just $Int =_{df} (\Gamma \mapsto D)$ when types are suppressed). For every $A \in Atom$, $I(A) \in Int_{\tau(A)}$.

A shifted interpretation $[d/x]I$, given an interpretation I, an atom x of type τ, and an entity $d \in Int_{\tau(x)}$ is defined by $([d/x]I)(x) = d$ and $([d/x]I)(y) = I(y)$ for $\tau(y) = \tau(x)$ but $x \neq y$. This notion of a shifted interpretation will be the foundation for the semantics of quantification.[14]

The way that we will approach the semantics of the operators is via the function that gives the extension of an expression in a particular case on a given interpretation.

Definition 2 (Quasi-Extension)**.** Let the *quasi-extension* of A in case γ on interpretation I be given by the following:[15]

$$QE_{\gamma,I}(A) = (I(A))(\gamma).$$

Since I assigns a member of Int to A, an interpretation I of such an A will be a function, $I(A)$, that takes cases as arguments and outputs extensions.

The semantics of the truth functions and identity are both extensional. Following (Belnap, 2006, p. 36), let *Neg* be the negation function on D_0=**2**={T,F} and set $QE_{\gamma,I}(\neg A) =_{df} Neg(QE_{\gamma,I}(A))$. In a similar way, the truth values of the other four non-modal connectives, $\wedge, \vee, \rightarrow, \leftrightarrow$ will depend only on the QE at γ, I of A and B. Identity is also extensional. $QE_{\gamma,I}(A_1 = A_2) = T \leftrightarrow_{df} QE_{\gamma,I}(A_1) = QE_{\gamma,I}(A_2)$. Thus, two individual concepts that have the same extension in a case will come out identical, although not necessarily so.[16]

[14] Having these definitions down, for the most part, in the rest of the essay I will suppress types because they are largely irrelevant to the issues under consideration.

[15] Belnap defines this notion as a counterpart to Bressan's quasi-intension function. "$QI_I(A)$" signifies "the quasi-intension on interpretation I of A". This gives a function from the set of cases Γ to expressions of the appropriate type τ, that is, a member of Int_τ.

[16] This makes formal issues over contingent vs. strict identity quite clear. For a nice defense of why treating equality this way is preferable in ML$^\nu$ see (Belnap, 2006, p. 36-37).

4 Adjustment of quantification

The semantics of quantification, modality, and predication, will all potentially need adjustments. Beginning with quantification will get us warmed up by raising some issues with interpretation shifts and domains that will come up again in more complex ways with modality. How exactly to treat predication will depend on how we treat modality, and modality will be difficult.

Dealing with quantification requires a bit of subtlety because we need to decide whether the quantifiers should range over only existing objects or also non-existing ones. Uniformity seems to demand that the quantifiers range over both and giving quantifiers a wider scope will likely be useful in a number of contexts, especially counterfactual and modal ones. It is also likely, however, that there will be many contexts, as when speaking of all of the men in the room, in which it will be useful to restrict quantification to existing objects. Thus, we will want to develop both accounts, although for uniformity's sake let's take quantification that ranges over possible objects as well as actual ones to be primary. Thus, the addition of non-existent objects doesn't affect the semantics for our primary notion of quantification, so that the semantics for it will be relatively straightforward. Let us use the following clauses:

$$QE_{\gamma,I}(\forall x\ A) =_{df} T \text{ iff } QE_{\gamma,[d/x]I}(A) = T \text{ for every } d \in Int$$

$$QE_{\gamma,I}(\exists x\ A) =_{df} T \text{ iff } QE_{\gamma,[d/x]I}(A) = T \text{ for some } d \in Int$$

There will, however, likely also be contexts in which we want to only quantify over the things which exist in a given case. Aside from (1) making all of the individuals that don't exist in γ irrelevant for evaluating the formula by adjusting the above definitions of the quantifiers, we could also backhandedly restrict the range of the quantifiers in two ways. (2) We could restrict our domain, Int, such that all the individual concepts in the intensional domain in a particular case exist in that case; or (3) we could modify how interpretation shifts work and only shift to objects that exist in the relevant case, while still including merely possible objects in our intensional domain. Accordingly, there are three obvious ways to pull off the more restrictive quantifications.

Although option (1) is obviously the most straightforward, a move similar to that of option (2) will seem enticing when dealing with modality, and arguably Gupta should implement a strategy like (3). Working backwards, option (3) has the obvious disadvantage of introducing a second notion of shifting—having one for unrestricted quantification and another for restricted quantification. Still, the third option is arguably the most natural within Gupta's languages because quantification is always relative to a common noun and is over extensions, not individual concepts. For, because quantification is restricted to a kind, it depends on an assignment function that is already a bit baroque, and because quantification is over extensions, the modifications required are relatively straightforward. In ML$^\nu$, however, how one should modify the notion of a shifted interpretation is not obvious. Thus, compared to (3), in ML$^\nu$ it seems better to bring about the restriction in the more straightforward way of option (1).

Option (2), where we only include existing individual concepts in the domain of a case, is a bit more tempting. This proposal contrasts with Bressan's original formulation because on his formulation the intensional domain is constant across cases. This is not, as one might first think, a dodgy metaphysical assumption on Bressan's part. Rather, having differing domains in the various cases of ML$^\nu$ would breed confusion, since it suggests that some kind of relationship between the domains can be gleaned from their differences. As discussed above, however, this is not how these kinds of relationships are fixed in ML$^\nu$, and differences of domain end up being pointless, since the logic of tracing objects from case to case is independent of the names objects receive in each case.[17]

On the other hand, option (1), which leaves the domain function and the notion of shifting untouched, simply and transparently discounts non-existent individuals when evaluating quantified formulas. Using $\forall^e$ to designate the non-standard, more restrictive quantification over only existing individuals, let us use the following clauses (remember, identity is extensional):

$$QE_{\gamma,I}(\forall^e xA) =_{df} T \text{ iff } (QE_{\gamma,[d/x]I}(A) = T \text{ or } d = *) \text{ for all } d \in Int$$

[17] Furthermore, parallel to the doubling of option (3), in order to preserve unrestricted quantification as well on option (2) one would need two distinct domains, one which includes merely possible objects, and one that doesn't. Such a complication seems worth avoiding, if possible.

$$QE_{\gamma,I}(\exists^e xA) =_{df} T \text{ iff } (QE_{\gamma,[d/x]I}(A) = T \text{ and } d \neq *) \text{ for some } d \in Int$$

Since this last account both preserves the uniformity of ML$^\nu$ and places the restriction in the quantification clause where it belongs, it is the best option.

5 Absolute and quasi-absolute concepts

In order to broach the problem that arises for the semantics of necessity, we will first need to understand Bressan's conception of absoluteness and quasi-absoluteness. Absolute concepts, remember, are how Bressan semantically captures what is distinctive of substances and their associated common nouns, over and above intensional predicates like "x is fragile." The latter, although they have criteria of application, will not pick out the same individual in every case, while common nouns supply criteria for how to identify, count, and individuate objects, which allow those individuals to be traced from case to case.

First, let's assume that the relevant objects exist in all cases. The primary notion of an absolute concept, as well as the two relevant auxiliary notions, modal constancy and modal separation, are defined as follows:

Definition 3 (Absolute Concept)**.** An intensional predicate F will be *absolute* if it is both *modally separated* and *modally constant.*

$$Abs =_{df} MConst \cap MSep$$

Definition 4 (Modal Constancy)**.** The intensional predicate F is *modally constant* if every individual concept that falls under F in some case does so in every case:

$$F \in MConst =_{df} \forall x\ (\Diamond Fx \to \Box Fx)$$

Definition 5 (Modal Separation)**.** An intensional predicate F is *modally separated* in a case if no two individual concepts falling under it share the same extension in any case:

$$F \in MSep =_{df} \forall x\ \forall y\ [(Fx \& Fy) \to (\Diamond(x = y) \to \Box(x = y))]$$

The idea behind constancy is roughly that if something falls under a constant predicate in one case, then it falls under it in all cases. The thought behind separation is that if the extensions of two individual concepts falling under a separated predicate in a case overlap at any case, then the concepts are identical.

Modal separation and modal constancy are independent properties—having one neither precludes nor guarantees having the other. Although modal separation is case relative, modal constancy and absoluteness are not. This is because constancy requires that every individual concept falling under the predicate in some case, fall under it in all cases. So if a predicate isn't modally separated in one case but is modally constant, then it won't be separated in any cases and thus won't be absolute.[18]

In order to adapt absoluteness to non-permanent substances, Bressan adjusts it along with modal constancy and separation in the following ways.

Definition 6 (Quasi-Absolute Concept)**.** His adapted notion of *absoluteness*, *quasi-absoluteness* is:

$$QAbs =_{df} QMC \cap QMS$$

where *QMC* and *QMS* are short for his adjusted *quasi-modal constancy* and *quasi-modal separation* respectively.

Modal constancy demands that an individual concept falling under the predicate fall under it in every case. Once we admit the non-existence of substances in some cases, it seems we don't want to have to say that the individual concept has to fall under the concept even in those cases where it doesn't exist. For

[18] Gupta tinkers with Bressan's notions. Although every common noun does provide a principle of identity for tracing the objects it is true of across cases, not every common noun designates a kind of substance. For example, 'man born in Jerusalem' does not, since being born in Jerusalem is not an essential property of the man. Unlike Bressan, who is only concerned with modeling substance kinds through his absolute concepts, Gupta marks this difference. He does this by distinguishing between sorts and substance sorts, which are the intensions assigned to the two kinds of common nouns. Roughly, sorts provide principles of identity that allow one to trace an object from case to case because they are separated intensional predicates, while substance sorts also indicate essential properties, and so are constant. Gupta maintains that for every sort, there is a substance sort that underlies it, which accounts for why it is separated (for discussion, see (McCawley, 1982)). Although Gupta's substance sorts correspond to Bressan's absolute concepts, and Gupta does not need to substantively alter Bressan's modal constancy, because Bressan's modal separation is case relative it will not do for modeling a principle of trans-case identity. For some of the technical details on how Gupta modifies modal separation so that it can effectively model the principles of identity of common nouns see the appendix.

example, take the sort ‘cat.’ This seems like a paradigmatic substance sort. Yet the extension of an individual concept designating an imaginary, yet to be born, or dead cat should arguably be the non-existent object. And it is at least not obvious that we should insist the non-existent cat fall under the sort even in these cases.[19] We will return to this issue, but for now, to allow the possibility of accommodating this thought, let us adjust modal constancy so that we discount cases in which the thing doesn’t exist.

Definition 7 (Quasi-Modal Constancy)**.** All cases in which an object doesn’t exist are irrelevant to evaluating the *quasi-modal constancy* of the predicate under which the object falls:

$$F \in QMC =_{df} \forall x \, [\Diamond Fx \rightarrow \Box(Fx \vee x = *)]$$

Modal separation also needs to be modified. Since we are using a single entity to designate non-existence, when two objects (individual concepts) don’t exist in a case their extensions will be the same. But if the extensions of two F’s overlap in any case, including a case at which they are not currently being evaluated, then F is not modally separated. Thus the predicate ‘x is a cat’ will not be modally separated if there are two cats which, though distinct in every case in which they exist, happen to both not exist in some case. Accordingly, we should discount these.

Definition 8 (Quasi-Modal Separation)**.** Thus, a predicate will be *quasi-modally separated* if it is *modally separated* in all cases except those in which the indi-

[19] Gupta goes so far as to suggest an added condition on sorts that they never apply to non-existent objects (Gupta, 1980, p. 69n).

viduals falling under it happen to not exist in the same case: [20]

$$F \in QMS =_{df} \forall x \, \forall y \, [(Fx \wedge Fy \wedge \Diamond[x = y \neq *]) \rightarrow \Box(x = y)]$$

Parallel to above, neither quasi-modal separation nor quasi-modal constancy preclude or guarantee the other. But unlike above, quasi-absoluteness and quasi-modal separation are case relative notions, while only quasi-modal constancy is not. Quasi-absoluteness is case relative because quasi-modal constancy is not enough to guarantee quasi-modal separation in every case. That is, although a predicate may be quasi-modally constant and quasi-modally separated in some case, that does not mean that it is quasi-modally separated in every case.[21] The upshot of the case relativeness of quasi-absoluteness is that it is inadequate for representing non-permanent substances like living beings. Gupta gives the following illustrative example. Now assuming there are only two cases, w_1, w_2 and only two objects in the structure, b and the non-existing object *, define the intentional property $\mathscr{S}$ as follows:

$$\mathscr{S} : \begin{array}{|cc|} \hline w_1 & w_2 \\ \hline bb & bb \\ & {}^{*}b \\ \hline \end{array}$$

Figure 1

[20] In defining Quasi-Modal Separation at (Bressan, 1993, p. 372), Bressan accidentally omits the diamond. (The lozenge is not forgotten in (Bressan, 1972, p. 94).) The ramifications of this omission illuminate the relative importance of QMC and QMS. Leaving it out weakens the requirement because without the diamond a predicate can still be quasi-modally separated in a case even if two individual concepts falling under it overlap, as long as they don't overlap in the case under consideration. Keeping the diamond means the individual concepts can't overlap in any case if the concept they are falling under is separated in that case. Since quasi-absolute concepts are also quasi-modally constant, whether the diamond is included or not makes no difference to them. Still, including the diamond is preferable because quasi-modal separation is intended to let us trace the same object from case to case, and if two objects overlap in some case, then from that case it is impossible to know which object to trace back through the other cases. These considerations help show that separation is more important than constancy for tracing objects. In Chapter 4 of his book Gupta elaborates an elegant solution to Chisholm's trans-world identity problem for inanimate objects (like bikes or Theseus's ship) that admits such objects aren't even quasi-modally constant but which shows that as long as the corresponding sortal predicate (e.g. 'x is a bike') is quasi-separated in every world, this is enough to trace them across worlds (Gupta, 1980, p. 86-107, esp. p. 104-106).

[21] This is the gist of the footnote on p. 70 of (Gupta, 1980).

Here the strings of characters in each row stand for individual concepts, whose left character indicates its extension in w_1 and whose right character indicates its extension in w_2. The columns indicate different worlds or cases. And the individual concepts listed in each column will be the individual concepts that the intensional property $\mathscr{S}$ is true of for the case in question. Like other predicates in ML$^\nu$, $\mathscr{S}$ will take individual concepts as arguments and yield truth values. What this picture represents is that the individual concept *bb* is $\mathscr{S}$ in both cases but $*b$ is only $\mathscr{S}$ in w_2. Here $\mathscr{S} \in QMS$ in w_1 but not in w_2, while $\mathscr{S} \in QMC$ in both cases, since quasi-modal constancy is not case relative. Thus, $\mathscr{S} \in QAbs$ in w_1 but not w_2. Responding to this point of Gupta's, Bressan modifies his account of what is needed for representing essential properties like being a certain kind of living being, claiming $\Box QAbs$ is what is needed, rather than just $QAbs$ (Bressan, 1993, p. 371-372). This is an adequate response to the objection because $\Box QAbs$ guarantees quasi-modal separation and quasi-modal constancy in every case.[22]

6 The dilemma for necessity

In order to give support to the claim of adequacy for $\Box QAbs$ Bressan goes on to prove the equivalence of it to $QMC \wedge \Box(\Diamond F \in QMS)$, his paraphrase in ML$^\nu$ of Gupta's notion of a *substance sort* from $L_2 - L_4$ (Bressan, 1993, p. 372). Namely, for a property F:

$$F \in \Box QAbs \equiv F \in QMC \wedge \Box(\Diamond F \in QMS)$$

This is slightly misleading, however, because Gupta's $L_2 - L_4$ face the problem discussed in the introduction that arises for the semantics of necessity once non-permanent substances are introduced—whether, e.g., cats which aren't cats in cases where they don't exist should count as necessarily cats or not—while (Bressan, 1993) gives no hint that there is a parallel problem for ML$^\nu$.[23] Accordingly, although Bressan adequately paraphrases Gupta's *substance sort*, he

[22] Again, I give some of the details of Gupta's account of these notions in the appendix.

[23] For Gupta's discussion of the problem see chapter 3, §2 (Gupta, 1980, p. 71-78).

doesn't acknowledge that more work needs to be done to give an account of non-permanent substances in ML$^{\nu}$ on a semantic par with Gupta's. In the remainder of the essay I will present the problem as it arises for ML$^{\nu}$ and propose what I think is the best solution.

The problem can be seen clearly in the definition of the quasi-extension function for necessity in (Belnap, 2006). Belnap defines the semantics of the quasi-extension of modal statements in the standard way. For necessity in particular:

$$QE_{\gamma,I}(\Box A) =_{df} T \text{ iff } QE_{\gamma,I}(A) = T \text{ for all } \gamma \in \Gamma$$

This just says that A will be necessary in case γ if it is true in every case. In this essay Belnap eschews dealing with non-existence. If he hadn't, he would have faced our dilemma. Is an animal that is a cat in every case in which it exists, necessarily a cat? If we treat 'x is a cat' as a quasi-absolute concept and don't modify the above definition of necessity, the answer would be no. Let me illustrate with a picture.

	γ_1	γ_2	γ_3
Cx:	abc	abc	abc
	eh^*	eh^*	
	f^{**}		

Figure 2

Here again the strings of characters stand for individual concepts whose extension in each case is indicated by the character in the corresponding position in the string, while the columns indicate different cases, and the individual concepts in each column will be the individual concepts the predicate is true of in that case. Cx is quasi-absolute, but because $QE_{\gamma_3,I}C(eh^*) = F$ and $QE_{\gamma_3,I}C(f^{**}) = F$, eh^* and f^{**} are not necessarily cats.[24] In order for eh^* and f^{**} to come out as necessarily cats while holding on to the above definition of necessity, they would have to fall under the predicate 'x is a cat' even in cases

[24] There is a pardonable abuse of notation here that I will continue in what follows. $QE_{\gamma_3,I}C(eh^*) = F$ abbreviates: $QE_{\gamma_3,I}C(x) = F$ where $I(x) = eh^*$, and similarly for $QE_{\gamma_3,I}C(f^{**}) = F$.

where they don't exist:

$$C'x\colon \begin{array}{|ccc|}\hline \gamma_1 & \gamma_2 & \gamma_3 \\ \hline abc & abc & abc \\ eh^* & eh^* & eh^* \\ f^{**} & f^{**} & f^{**} \\ \hline\end{array}$$

Figure 3

The dilemma then is this: Should a thing count as having a property necessarily if it only has it in every case in which it exists or should it only count as having the property necessarily if it has it in every case, regardless of whether or not it exists?

7 The former route: Gupta's advantage

In ML^{ν} the less permissive option, which requires an individual concept have a property even in the cases where it doesn't exist for it to have that property necessarily, works better. According to Gupta, the major drawback of this route is that it doesn't really accord with the everyday usage of common nouns. We tend to think dead cats are still necessarily cats, but don't think that merely possible cats are cats. As a result, Gupta maintains we should both think of cats as necessarily of their kind, but not hold that 'is a cat' is true of them when they do not, no longer, or have yet to, exist.[25] Going the less permissive route means giving up one or the other of these. If this is unacceptable in a given application, then Gupta's languages (especially L_3) are preferable, since they

[25] How much we agree with Gupta here will depend on what we are using our logic for and how we are interpreting our cases. At first at least, it will seem we won't want to count merely possible cats as cats, if we interpret cases as worlds, since we don't want to have to consider all of the merely possible men in the room when we talk about men. On the other hand, if cases are interpreted as times or moments in possible histories, and we want to model, "Mama could have had two more kittens than she in fact had" it seems odd to insist that what we are referring to are not cats.

This rule of thumb will certainly not be hard and fast, and I do not take deciding between these options to be a matter for logic. Still, perhaps it is worth mentioning my own view is that merely possible cats, men, or unicorns, should usually count as of their kind, in line with *Figure* 3 and against Gupta. Possible, dead, or imaginary men seem to be no less men than do living ones and their non-existence is marked by their having the non-existent object as their extension. (Kant's remark about the hundred Thalers comes to mind (*CpR*, A599/B627).)

preserve both.[26] We can now see a further reason Gupta has for giving up the uniformity of ML$^\nu$ that I spoke of above. He wants to hold on to both of these thoughts, but as long as predication is primarily intensional, which is what is largely responsible for the uniformity of ML$^\nu$, it is not obvious how to build a semantics for necessity more permissive than the one I gave on p. 15.

We can see this difficulty more clearly if we look at how to go about setting up the semantics of necessity in ML$^\nu$ in accord with the first, more permissive option that Gupta favors. This section will be devoted to doing this by formalizing the following:

Definition 9 (Rough Definition)**.** $QE_{\gamma,I}(\Box A) = T$ iff $QE_{\gamma,I}(A) = T$ in every case γ in which the truth value of A does not depend on an individual that doesn't exist in γ.[27]

The trick is finding a definition that captures the second clause uniformly for each kind of formula A. The options are in a way similar to the ones we faced with quantification because again we are trying to discount individuals that don't exist in the case in which we are evaluating the formula. The major difference here, however, is that there is no easy way to simply and uniformly discount non-existent individuals in the definition itself because what is in question here is not individual concepts but formulas, and there is no straightforward way to identify a formula, A, that involves a non-existent object.

The first option we could try, crudely analogous to quantification option (1) in section 4, is to mark the truth values of formulas that are only invalidated by non-existent objects and adjust the semantics for necessity accordingly. We will see that this option has serious drawbacks. The second, similar to quantification option (2) above, would involve restricting the domains so that only individual concepts that exist belong to the domain of a particular case. As with quantification, however, this would not be in the spirit of ML$^\nu$ because it relies on distinctions in the structure of the model for securing principles of trans-case identity. With necessity in ML$^\nu$ option (3), however, is even less

[26] L_4, however, is an exception. It gives up on the thought that possibly non-existent cats are necessarily cats and is much closer to ML$^\nu$ than Gupta's other languages because its variables range over individual concepts rather than extensions.

[27] This is closely related to Gupta's "initial intuition" (Gupta, 1980, p. 71).

natural than it was with quantification, since there is no notion of shifting that could be restricted. Gupta, however, attempts a modification of the assignment function with some success and I will look briefly at the advantages of this over implementing option (2) in ML^{ν}.

7.1 Option 1: the 'new-truth-value' way

Let us start thinking about how the first option would work by considering the atomic case. Continuing to use the above example, $QE_{\gamma_3,I}C(ef*) = F$ as in *Figure* 2, but we don't want this to falsify $QE_{\gamma_1,I}\Box C(ef*) = T$. Since predication is of individual concepts, either an individual concept falls under a predicate in a case or it does not. If it does not, then the predication is false. But on the guiding thought of our *Rough Definition*, instances where the individual concept does not exist are special for the semantics of necessity. Failure here doesn't falsify the necessity. Accordingly, a straightforward way to proceed is to mark the truth value resulting from these cases (with say T*), and make them irrelevant for evaluating the necessity of a formula. For monadic atomic sentences[28] let's mark such cases as follows:[29]

$$QE_{\gamma,I}\phi(i) = T^{*} \text{ iff } QE_{\gamma,I}\phi(i) \neq T \text{ and } i(\gamma) = *$$

We can extend this to quantified formulas as well. The case where quantification is restricted to existing objects is fairly straightforward and will either come out true or false. The less restrictive, standard quantification requires, however, some subtlety. Since we have the more restrictive form of quantification, it makes sense to set the value of the quantified formula equal to T* in those cases where the formula is only falsified by merely possible objects.

[28] Perhaps the semantics can, somewhat controversially, be extended to two (or more) place relations by the following maneuver:

$$QE_{\gamma,I}\psi(i,j) = T^{*} \text{ iff } QE_{\gamma,I}\psi(i,j) \neq T \text{ and either } i(\gamma) = * \text{ or } j(\gamma) = *$$

Here, on a temporal reading, "I am the great grandchild of my great grandfather" would be T* (just as with "Socrates is a man"), since my great grandfather has passed away, if we treat the predicates as 'is a cat' in *Figure* 2. If we treated them as in *Figure* 3, however, their value would be T, and we need not treat all predicates one way or the other.

[29] N.B. if formulas only depend on objects whose extension is non-existent and are false of those, they will still come out T*. For example, "Socrates is sitting" is T*.

Accordingly,

$$QE_{\gamma,I}(\forall x\ A) = T^* \quad \text{iff} \quad \begin{array}{r} QE_{\gamma,[d/x]I}(A) = T \text{ for all } d \in Int \text{ where } d \neq * \text{ and} \\ QE_{\gamma,[d/x]I}(A) \neq T \text{ for some } d \in Int \text{ where } d = * \end{array}$$

Now from these considerations it might seem we only have to adjust our definition of necessity slightly: $QE_{\gamma,I}(\Box A) = T$ iff $QE_{\gamma,I}(A) \neq F$ for every $\gamma \in \Gamma$.

This is not, however, right. For the most part, the semantics for the truth functional connectives are straightforward—e.g. $T \vee T^* = T$; $T \wedge T^* = T^*$; $T \to T^* = T^*$, etc.—but negation is not. If a formula is T^* then it is true for all objects that exist in a case, but not true of some object that doesn't exist. If that formula is negated then the negation will be false of all existing objects and should in most contexts be counted false. Still, it would be true for some non-existent object,[30] which may matter occasionally, in which case it makes sense to give it the value F^*, where F^* merely means just true of some non-existent object. This then suggests the corresponding adjustment for existential quantification:

$$QE_{\gamma,I}(\exists x\ A) = F^* \quad \text{iff} \quad \begin{array}{r} QE_{\gamma,[d/x]I}(A) = F \text{ for all } d \in Int \text{ where } d \neq * \text{ and} \\ QE_{\gamma,[d/x]I}(A) = T \text{ for some } d \in Int \text{ where } d = * \end{array}$$

With this new truth value, the slightly more complicated semantics of necessity should likely go:

$$QE_{\gamma,I}(\Box A) = T \text{ iff for every } \gamma \in \Gamma, \text{ neither } QE_{\gamma,I}(A) \neq F \text{ nor } QE_{\gamma,I}(A) \neq F^*$$

I think with these elements on the table, we can see that this strategy will be unattractive for many reasons. Although parts of the semantics of this don't seem so bad (e.g. $F \vee F^* = F^*$; $T \wedge F^* = F^*$; $T \to F^* = F^*$; $F^* \to F = T^*$, and let's say $T^* \to F = F^*$, which was problematic before), when the new truth values interact it gets rather byzantine, especially with the conditional.[31] Also, it is not clear how we should think of these new values; they do not seem to have

[30] Arguably, an example might be, 'there is something that is a flying horse,' where the intensional object that makes this true is Pegasus.

[31] The full tables for the two place logical operators are:

a counterpart in our normal considerations of truth. In some ways they seem to indicate a truth value gap since, for example, sentences about non-existent objects like “Socrates is sitting” will count as T^*. With quantified sentences, however, where it is primarily existing objects that are in question, these will be more like their non-stared counterparts. Still, perhaps this strategy could be developed to keep cats necessarily cats.

7.2 Option 2 in ML$^\nu$ and option 3 in Gupta’s languages

The second way to implement the *Rough Definition*—by restricting the intensional domain of each case to individual concepts that exist in it—will also work. Although it leaves the semantics of necessity straightforward, as discussed in section 4, it has the disadvantage of introducing distinctions in domain which are otherwise unmotivated in ML$^\nu$.

Gupta’s languages, however, present a rather ingenious way to accommodate the thought of the second part of the *Rough Definition* through a shift in how the assignments are considered. Again, take the atomic case. Having put to one side marking cases with a special truth value, it seems if we leave predication intensional we are stuck saying those predications where the predi-

$\wedge$	T	T^*	F^*	F
T	T	T^*	F^*	F
T^*	T^*	T^*	F^*	F
F^*	F^*	F^*	F^*	F
F	F	F	F	F

$\vee$	T	T^*	F^*	F
T	T	T	T	T
T^*	T	T^*	$T^*(T)$	T^*
F^*	T	$T^*(T)$	F^*	F^*
F	T	T^*	F^*	F

$\rightarrow$	T	T^*	F^*	F
T	T	T^*	F^*	F
T^*	T	$T^*(T)$	F^*	F^*
F^*	T	$T^*(T)$	$T^*(T)$	T^*
F	T	T	T	T

$\leftrightarrow$	T	T^*	F^*	F
T	T	T^*	F^*	F
T^*	T^*	$T^*(T)$	$F^*(F)$	F^*
F^*	F^*	$F^*(F)$	$T^*(T)$	T^*
F	F	F^*	T^*	T

I have listed two truth values for some of the operations because while they will usually have the first value, if the reasons that the values of the component formulas was T^* or F^* in the first place line up, then it seems they should have the second value. For example, if two formulas involving universal quantification are T^*, then for each there will be individual concepts, $d_1 \ldots d_n$, whose extensions are non-existent and which falsify them. Taking one of these formulas as the antecedent and the other as the consequent, if those individual concepts that falsify the antecedent are a superset of those on which the consequent is false, then it should be: $T^* \rightarrow T^* = T$. Or if the antecedent is F^* and the consequent T^*, then the way the conditional will come out T is if none of the values on which F^* is true are also those on which the consequent is false. So that the operations are truth functional, it makes sense to assign the value that is not in parentheses, despite the fact that in specific cases the assignment of the other value can be justified.

cation comes out false due to non-existent objects are just false. If, however, we abandon the primacy of intensional predication, making extensional predication primary again, then there is a way to get around the problem. Going this route, Gupta introduces an assignment function that assigns extensions and sorts to variables.[32] He defines this so that he preserves the ability to trace objects from case to case via sorts, while at the same time leaving undefined what the counterpart of that object is in cases where it doesn't exist. Then, in the semantics for necessity, he makes irrelevant all cases in which an assignment function for a variable in the formula comes out undefined (Gupta, 1980, p.71-75).

Despite the loss of uniformity, Gupta's way around the problem of formalizing the *Rough Definition* is preferable to restricting the domain to existing individuals in ML^{ν}, while keeping predication primarily intensional. First, Gupta's way doesn't make it difficult to model talk of merely possible objects in the language. One way to see this is that there is still a use for the non-existent object in Gupta's languages, while in the restricted domain version of ML^{ν} there wouldn't be. Using ML^{ν}, suppose we had an object ab^* in γ_1 (so $a \in D_{\gamma_1}$ and $ab^* \in Int_{D_{\gamma_1}}$). If we restrict the intensional domains of the cases just to existing objects then $ab^* \notin Int_{D_{\gamma_3}}$. The first thing we can see is that $*$ will never be the extension in a particular case of an individual concept in the intensional domain of that case. Put slightly differently, if we try to trace an individual concept to a case in which it doesn't exist we won't find that concept. Accordingly, on the domain restriction way it would be more accurate to leave individual concepts undefined in cases where they don't exist than to use an object to mark non-existence. But if an object is always undefined in cases where it doesn't exist, and its individual concept isn't in the intensional domain of such cases, then there is literally nothing picked out in that case by names of merely possible objects. So the language can't effectively model talk of merely possible objects unless we include a second unrestricted domain.

In Gupta's languages, on the other hand, there is still a use for the non-existent object. Individual concepts fall under sorts and variables are assigned to extensions and sorts. The counterpart of an extensional individual assigned to a variable is determined by first finding the individual concept under the

[32] Again, for technical details see the appendix.

assigned sort whose extension is that individual in the starting case. Then one looks at the extension of that individual concept in the case to which one is tracing the assigned object. This extension is the counterpart. Suppose for example that 'Cx' of $Figure$ 2 on page 15 is one of Gupta's sorts, and that the assignment function assigns the object 'e' and the sort 'C' to the variable 'x' in γ_1. The individual concept this assignment corresponds to is eh^* and the counterpart in γ_2 of x in γ_1 is h. If the counterpart would be the non-existent object, then the assignment function comes out undefined for that variable in that case. For example, since we find the non-existent object when trying to trace x in γ_1 to γ_3, the assignment function is undefined for x in γ_3. Thus, merely possible objects are still in Gupta's intensional domains and it makes perfect sense to talk about them. At the same time, however, Gupta also has a relatively easy way to discount formulas that depend on non-existent objects when evaluating a necessary formula in L_3. He stipulates in his definition of necessity that necessary formulas are true if and only if they are true in all cases in which nothing in them is undefined.[33]

Another reason the loss of uniformity that comes with making predication extensional isn't a terrible mark against Gupta's strategy is that being committed to preserving the thought about possibly non-existent cats still necessarily being cats already seems to belie a commitment to thinking that predication should primarily be extensional. This is in part because if one keeps predication intensional there is a very simple, straightforward definition of necessity on which possibly non-existent cats are not necessarily cats. But perhaps more deeply (and more obscurely) this is because holding on to the intuition means modifying an intensional property, necessity, in light of an extensional fact, existence. On the commitments we have been trying to preserve with the formalizations of the *Rough Definition*, a thing can belong necessarily to a property despite failing to belong to it in cases where it has a certain extensional property (non-existence). This means modifying the account of necessity, making it less

[33] There are, however, serious issues with the semantics of quantification and necessity for this strategy. Specifically, as Gupta notes (Gupta, 1980, p.72-75), although intuitively the assignments of variables that are not free in a formula should be ignored when figuring out their semantic value, implementing this is difficult, and for both L_2 and L_3, the schema '$A \rightarrow (\forall K, x)A$' is invalid.

straightforward, in order to make an exception in cases where the thing in question has the extensional property of non-existence. Thus, by holding on to the intuitions behind the *Rough Definition*, Gupta is already privileging extensional properties over intensional ones. Accordingly, it makes sense to have extensional predication be prior to intensional predication, and to make variables range over individuals rather than individual concepts.

We have seen two potential motivations for Gupta's choosing to sacrifice the uniformity of ML^{ν} aside from his main motivation of capturing the special syntactic and semantic character of common nouns. First, he thinks in most contexts predication is primarily extensional and should be modeled that way.[34] Second, giving up the uniformity of intensional predication allows him to preserve the thought that non-existent cats are not cats and that possibly non-existent cats can still necessarily be cats. This feature of Gupta's languages goes unmentioned in (Bressan, 1993). In instances where one would like to preserve both of these thoughts or would like an extensional language, despite its unwieldiness, Gupta's languages will be preferable to ML^{ν}.

8 The latter route: essential properties in ML^{ν}

We might wonder, however, whether it is really so important to say that possibly non-existent cats are necessarily cats. There is, after all, a stricter sense of necessity on which if merely possible cats are not cats, then possibly non-existent cats are not necessarily cats. If, while holding on to this notion of *necessity* (which is just the one on page 15), we have another notion that applies to predicates which hold necessarily of the existing objects falling under them, then we will still be able to give an account of what is special about predicates like 'is a cat'.

What is distinctive of 'x is a cat' (Cx) in *Figure* 2 is that it is a quasi-absolute concept. In particular, it is not modally constant, but merely quasi-modally constant (see the definition on p. 12). Looking back at *Figure* 3 on page 16, which made 'x is a cat' a necessary property, but also counted merely possible cats as cats, we can see that this table's version of 'x is a cat' ($C'x$), on the

[34] Gupta comments on this (Gupta, 1980, p. 77-78).

contrary, will be both modally constant (see p. 10) and quasi-modally constant. This difference can be captured using the following notion:

Definition 10 (Modally Constant, Quasi-Absolute Concept)**.**

$$QAbs^{MC} =_{df} MConst \cap QMS$$

$C'x$, but not Cx, will be a modally constant, quasi-absolute concept. Although all $QAbs^{MC}$ are quasi-absolute, because not all quasi-modally constant concepts are modally constant, $QAbs^{MC}$ is more restrictive than $QAbs$. This further restriction means that in contrast to $QAbs$, $QAbs^{MC}$ is not case relative. That is, if $F \in QAbs^{MC}$ in some case, then $F \in QAbs^{MC}$ in all cases.[35]

The final notion that will be useful to us is:

Definition 11 (Modally Constant Expansion)**.**

$$\begin{gathered} F' \text{ is the } \textit{modally constant expansion} \text{ of } F \text{ iff} \\ \forall x\ [\Diamond F(x) \rightarrow \Box F'(x)] \end{gathered}$$

If *Figure* 2 is for $C(x)$, 'x is a cat', then *Figure* 3 is for the modally constant expansion of 'x is a cat', $C'(x)$. Since all objects falling under a quasi-absolute concept fall under it in all of the cases in which they exist, quasi-absolute concepts will be true of exactly the same existing objects as their modally constant expansions, but may differ from them in that non-existent objects falling under the expansions may not fall under the original quasi-absolute concepts.

Following (Bressan, 1993, p. 372) we can define a *substance* as something that falls under a necessarily quasi-absolute concept (a concept that is quasi-absolute in every case) and a *substance sort* as such a necessarily quasi-absolute concept. This is, remember, his paraphrase of Gupta's *substance sort*, introduced above on page 14. Then, although a substance won't necessarily fall under its substance sort because the sort could not include it in a case in which it didn't exist, it will necessarily fall under the modally constant expansion of that sort.

[35] This means the problem with *QAbs* that Gupta points to in his footnote (which was discussed on page 13 above) is not one for $QAbs^{MC}$ (Gupta, 1980, p. 70).

Now, with all of these notions in place, we might think of substance sorts as indicating *essential* properties of those things that fall under them. For, something that falls under them falls under them in all cases in which it exists (i.e. they are quasi-modally constant).[36]

This conception of an essential property, however, has a few features that may be undesirable. For one, although all necessary properties will be essential properties, essential properties will not generally be necessary. This makes essentiality a weaker notion than necessity, although it is usually thought of as stronger. Further, this conception of an essential property is intensional, having to do with extensions in various cases, not hyperintensional, which would distinguish essential properties from non-essential properties that are had in all cases.[37] As a result, it cannot be used to model the distinction between essential properties and non-essential properties that something has in every case in which it exists.[38] Of course, this is what we should expect from an intensional system such as ML^{ν}.

9 Conclusion

We have attempted to develop Bressan's ML^{ν} so that it can accommodate both an affirmative and a negative answer to the question of whether possibly non-existent things should count as necessarily the sort of thing they are. If one wants to give an affirmative answer and preserve the thought that possibly non-existent substances are still necessarily (rather than merely essentially) substances of the sort they are, as Gupta does, then his languages will be preferable to ML^{ν}. Still, as is clear in Gupta's own discussion (which I alluded to in footnote 33 and look at in the appendix) there are significant (and it seems quite deep) unresolved issues for this implementation. Because of these, we might wonder whether holding onto whatever original motivation we had to

[36] Belnap and Müller hit upon the same conception of an essential property in developing CIFOL (Belnap and Müller, 2014b, § 5.3). They do a nice job of showing how absolute concepts will not be the only essential properties. Properties like the sex of horse, which we commonly take to be essential to it qua horse, will also come out as essential in this sense.

[37] Here I am adapting Gupta's distinction between the extension, intension, and hyperintension of an expression (Gupta, 1980, p. 17), which he borrows from (Cresswell, 1975).

[38] Fine is after this distinction with his example of Socrates and his singleton (Fine, 1995, p. 241).

implement this answer to our question is worth it. This is especially so considering that in our development of ML$^{\nu}$ we have both preserved the uniformity of Bressan's original formulation and counted 'x is a cat' as a substance sort that designates an essential property, although it does not necessarily apply to cats, because they are born and die.

Appendix: Gupta's languages

In this appendix I will give some of the technical details concerning two aspects of Gupta's Languages. First, I will present Gupta's various versions of the notions of a substance sort, constancy, and separation, which correspond to Bressan's absolute concepts, modal constancy, and modal seperation. Second, I will give some of the details of Gupta's treatment of the semantics of quantification and necessity. When he introduces non-existent objects, Gupta struggles mightily to adapt the semantics of necessity so that possibly non-existent cats can still be necessarily cats, but as I mentioned in the conclusion, he does not ultimately arrive at a satisfying formulation. (Gupta recognizes this, to a degree (Gupta, 1980, p. 75).) I will try to give some sense of why, without going through the full story (Gupta, 1980, ch. 3). All page numbers included with the definitions are to the corresponding definitions in (Gupta, 1980).

Before doing either of these, some preliminaries are in order. In addition to the standard logical categories, Gupta's L_1 includes a category for common nouns. Although the syntactic rules of L_1 are fairly straightforward and I will not rehearse all of them, quantification is always restricted to quantification over a certain sort of thing by a common noun, so the syntactic rules governing these are worth presenting:

Definition 12 (Some of Gupta's syntax; cf. p. 7)**.**

(i) If K is a common noun, x is a variable, and A is a formula, then $(\forall K, x)A$ is a formula.

(ii) If K is a common noun, x is a variable, and A is a formula, then $(K, x)A$ is a common noun.

The second clause allows for complex common nouns built from simpler ones, such as 'Man who likes Margret.'[39]

The semantics of Gupta's languages begins standardly enough with

Definition 13 (Model Structure for L_1; p. 18)**.** A *model structure* for L_1 is an ordered triple $\langle W, D, i^* \rangle$, where:

[39] For more details and discussion see (Gupta, 1980, ch. 1, §1-2).

(i) W is a nonempty set,

(ii) D is a function that assigns to each member of W a nonempty set,

(iii) i^* is a function that assigns to each member w of W a member of $D(w)$.

Think of W as the set of possible worlds (or cases), $D(w)$ as the set of objects that exist in w, and i^* as the individual concept whose extension in all worlds is the non-existent object (i.e. $i(w) = *$ in all w).

As with Bressan's absolute concepts, Gupta models substance sorts through an intensional property that is constant and separated:

Definition 14 (Gupta's Substance Sort in L_1; p. 35)**.** A *substance sort* in a model structure $\mathfrak{A}$ is a modally constant and separated intensional property.

Modal constancy in L_1 is not substantively different from Bressan's, although Gupta states it slightly differently. Where $\mathfrak{A}$ is a model structure:

Definition 15 (Gupta's Modal Constancy; p. 27)**.** An intensional property $\mathscr{S}$ in $\mathfrak{A}$ is *modally constant* iff $\mathscr{S}(w) = \mathscr{S}(w')$ at all worlds $w, w' \in W$.

That is, $\mathscr{S}$ will be constant if the individual concepts in the extension of $\mathscr{S}$ for any world are the same.

I mentioned in footnote 18 that Gupta adjusts Bressan's notion of modal separation because it is case relative, and so it will not do for modeling a principle of trans-case identity. Gupta's preferred modification is his own 'separation:'

Definition 16 (Gupta's Seperation; p. 29)**.** An intensional property $\mathscr{S}$ in $\mathfrak{A}$ is *separated* iff all individual concepts i, i' that belong to $\mathscr{S}$ at any worlds w, w' are such that if $i(w_1) = i'(w_1)$ at a world w_1, then $i = i'$.

He uses this to define his notion of a sort:

Definition 17 (Gupta's Sort in L_1; p. 33)**.** A *sort* in a model structure $\mathfrak{A}$ is an intensional property in $\mathfrak{A}$ which is separated.

In general Gupta uses variables '$\mathscr{S}$', '$\mathscr{S}'$', '$\mathscr{S}_1$,' etc. to range over sorts in a fixed model structure $\mathfrak{A}$.

In addition to Gupta's separation, there is also a weaker notion that is like Bressan's world relative separation, except that it holds at every case.

Definition 18 (Gupta's World Relative Seperation; p. 29n)**.** An intensional property $\mathscr{S}$ in $\mathfrak{A}$ is *separated in the world* w iff all individual concepts i, i' that belong to $\mathscr{S}$ at w are such that if $i(w_1) = i'(w_1)$ at a world w_1, then $i = i'$.

Definition 19 (Gupta's Weak Seperation; p. 29n)**.** An intensional property $\mathscr{S}$ in $\mathfrak{A}$ is *weakly separated* iff $\mathscr{S}$ is separated in every world.

Intuitively, the difference is that while separation says that the extension of two individual concepts that are $\mathscr{S}$ in any world (even if these worlds are different) will never overlap at a world, weak separation just says that at each world the individual concepts that are $\mathscr{S}$ in that world will not have the same extension in any world. Weak separation can provide a principle of identity, and Gupta develops a closely related notion that also incorporates a treatment of non-existence in L_5 of chapter 4.

In order to accommodate non-existence, in chapter 3 Gupta modifies L_1 in three different ways, treating the semantics of quantification and necessity slightly differently in each of L_2, L_3, and L_4. Still, the way that he treats modal constancy and separation in each of these is the same:

Definition 20 (Gupta's Near Constancy; p. 69-70)**.** An intensional property $\mathscr{S}$ is *nearly constant* in $\mathfrak{A}$ iff, if an individual concept i belongs to $\mathscr{S}$ at any world w, then i belongs to $\mathscr{S}$ at all worlds w' such that $i(w') \neq i^*(w')$.

Definition 21 (Gupta's Near Seperation; p. 69)**.** An intensional property $\mathscr{S}$ is *nearly separated* in $\mathfrak{A}$ iff all individual concepts i, i' that belong to $\mathscr{S}$ at some worlds (i.e., $i \in \mathscr{S}(w_1)$ and $i' \in \mathscr{S}(w_2)$, for some $w_1, w_2 \in W$) are such that if $i(w) = i'(w) \neq i^*(w)$ at a world w, then $i = i'$.

As I just noted here and in footnote 18, Gupta is concerned with distinguishing two forms of principles of identity associated with common nouns, sorts and substance sorts, where only the latter apply to essential properties.

Definition 22 (Gupta's Sort in L_2-L_4; p. 69)**.** $\mathscr{S}$ is a *sort* in a model structure $\mathfrak{A}$ iff $\mathscr{S}$ is an intensional property and $\mathscr{S}$ is nearly separated in $\mathfrak{A}$.

Definition 23 (Gupta's Substance Sort in L_2-L_4; p. 70)**.** $\mathscr{S}$ is a *substance sort* in $\mathfrak{A}$ iff $\mathscr{S}$ is a sort in $\mathfrak{A}$ and $\mathscr{S}$ is nearly constant in $\mathfrak{A}$.

As with Gupta's separation and Bressan's modal separation, Gupta's near separation differs from Bressan's quasi-modal separation in that it is not case relative and so can serve as a principle of trans-case identity. In chapter 4, Gupta modifies separation and constancy again, this time endorsing for his 'quasi-separation' something like Bressan's 'quasi-modal separation' but in every case. For the details, as well as the corresponding required adjustment of constancy for L_5, see (Gupta, 1980, p. 104, & p. 107).

There are a few background notions that we need to have in place before we can look at Gupta's treatment of the semantics for quantification and necessity. First, he defines two sets of brackets. Given a sort $\mathscr{S}$ in $\mathfrak{A}$, he designates by $\mathscr{S}[w]$ the set of objects that fall under $\mathscr{S}$ in w, and by $\mathscr{S}[\![w]\!]$ the set of objects that are possibly $\mathscr{S}$.

Definition 24 (p. 35)**.**

$$\begin{aligned}\mathscr{S}[w] =_{df} \{d : d \in D(w) &\text{ and there is an individual concept}\\ &i \in \mathscr{S}(w) \text{ such that } i(w) = d.\}\end{aligned}$$

Definition 25 (p. 36)**.**

$$\begin{aligned}\mathscr{S}[\![w]\!] =_{df} \{d : d \in D(w) &\text{ and there is an individual concept}\\ i \in \mathscr{S}(w') &\text{ for some } w' \in W \text{ such that } i(w) = d\}.\end{aligned}$$

Next he defines what it means to be 'the same $\mathscr{S}$' and 'an $\mathscr{S}$ counterpart:'

Definition 26 ('the same $\mathscr{S}$' in L_1; p. 36)**.** d in w is the same $\mathscr{S}$ as d' in w' iff there is an individual concept, i, that belongs to $\mathscr{S}$ at some world, and $i(w) = d$ and $i(w') = d'$.

Definition 27 ('an $\mathscr{S}$ counterpart'; p. 36)**.** The $\mathscr{S}$ counterpart in w' of the individual d in w (abbreviated $\mathscr{S}(w', d, w)$) is the unique individual d' such that d' in w' is the same $\mathscr{S}$ as d in w.

As Gupta points out, "$\mathscr{S}(w', d, w)$ is well defined if $d \in \mathscr{S}[\![w]\!]$. For if $d \in \mathscr{S}[\![w]\!]$, then there is an individual concept i belonging to $\mathscr{S}$ at some world such that $i(w) = d$. The separation of $\mathscr{S}$ implies that i is unique. Hence there is a unique d', namely $i(w')$, which is the same $\mathscr{S}$ as d in w" (Gupta, 1980, p. 36).

Now to define the assignment function, we first need the notion of a model:

Definition 28 (Model for L_1; p. 37)**.** A *model* for L_1 is an ordered quintuple $\langle W, D, i^*, m, \rho\rangle$, where:

(i) $\langle W, D, i^*\rangle$ is a model structure,

(ii) m is a function that assigns (a) to each individual constant of L_1 an individual concept, (b) to each n-ary predicate an n-ary relation, and (c) to each atomic common noun a sort,

(iii) $\rho \in W$.

Through the function m a model in L_1 assigns an intension to each atomic expression and ρ specifies the real world.

With the notion of a model, an assignment is:

Definition 29 (Assignments for L_1; p. 38)**.** An *assignment* for L_1 relative to a model $M = \langle W, D, i^*, m, \rho\rangle$ is a function that assigns to each variable of L_1 an ordered pair $\langle \mathscr{S}, d\rangle$, where $\mathscr{S}$ is a sort relative to the model structure and $d \in U (= \cup_{w \in W} D(w))$.

Here if a is an assignment, $a_o(x)$ is the object assigned to x by a and $a_s(x)$ is the sort assigned to x by a. Using this Gupta defines a few notions that he then deploys in defining the semantic value of formulas involving quantification and necessity:

Definition 30 (Normal assignments for L_1; p. 38)**.** An assignment a (for L_1 relative to a model M) is *normal* in w iff $a_o(x) \in a_s(x)[\![w]\!]$ for all variables x.

Definition 31 ($\mathscr{S}$ variants for L_1; p. 38-39)**.** An assignment a′ *is an $\mathscr{S}$ variant of* a *at x in* w iff

(i) a' is just like a except perhaps at x (abbreviated to $a' \underset{x}{\overset{\frown}{=}} a$),

(ii) $a'_s(x) = \mathscr{S}$,

(iii) $a'_o(x) \in \mathscr{S}[w]$.

Definition 32 (World variants for L$_1$; p. 39)**.** The w$'$ *variant of a relative to* w (abbreviated to $f(w', a, w)$) is the unique assignment a' that meets the following conditions:

(i) $a'_s(x) = a_s(x)$ at all variables x,

(ii) $a'_o(x)$ in w' is the same $a'_s(x)$ as $a_o(x)$ in w, at all variables x.

If these conditions are not met by any assignment then $f(w', a, w)$ is undefined.

Now, having defined M, w, and a, Gupta then defines through induction on the length of expression α the concept: "*the semantic value of* α *at a world* w *in a model* M *relative to the assignment* a *normal in* w" (Gupta, 1980, p. 40). He abbreviates this to $V^w_{M,a}(\alpha)$. Before giving this definition, however, it will help with quantification to have defined one more function that gives the intension of an expression α in a model M, for an assignment a, and a world w:

Definition 33 (Intension function for L$_1$; p. 40)**.** Let M, w, a, and α be as above. Then $I^w_{M,a}(\alpha)$ is a function with domain W that satisfies the following condition:

$$(I^w_{M,a}(\alpha))(w') = V^{w'}_{M,f(w',a,w)}(\alpha)$$

Gupta uses m to define the valuation function V as expected for individual constants, variables, and common nouns, and the value of an equality or truth function are found in the standard ways (Gupta, 1980, cf. p. 40-41). I include the definition of V for n-ary relations to give a better sense of how things run:

Definition 34 (Part of $V^{w'}_{M,a}(\alpha)$; p. 40-41)**.** Let M, w, a, and α be as above. Then V is defined by induction on α:

(i) If α is the atomic formula $F(t_1, \ldots, t_n)$, then $V^w_{M,a}(\alpha) = T$
if $\langle V^w_{M,a}(t_1), \ldots, V^w_{M,a}(t_n)\rangle \in m(F)(w)$. Otherwise $V^w_{M,a}(\alpha) = F$.

(ii) If α is the formula $\Box A$, then $V^w_{M,a}(\alpha) = T$
if $V^{w'}_{M,f(w',a,w)}(A) = T$ at all worlds $w' \in W$. Otherwise $V^w_{M,a}(\alpha) = F$.

(iii) If α is the formula $(\forall K, x)A$, then $V^w_{M,a}(\alpha) = T$
if $V^w_{M,a'}(A) = T$ for all assignments a' that are $I^w_{M,a}(K)$ variants of a at x in w. Otherwise $V^w_{M,a}(\alpha) = F$.

(iv) If α is the common noun $(K, x)A$, then $V^w_{M,a}(\alpha)$ is the set of individual concepts i such that $i \in V^w_{M,a}(K)$ and $V^w_{M,a'}(A) = T$, where $a' \underset{x}{\overset{\frown}{=}} a$ and $a'_s(x) = I^w_{M,a}(K)$ and $a_o(x) = i(w)$.

Working back through the definitions, with this semantics one can see how assignment functions contribute to fixing principles of trans world identity in the way described in section 7.2.

In accommodating non-existent objects Gupta takes over many of these definitions, only modifying them when necessary. The main difficulty comes with the semantics of necessity and how it interacts with quantification. To give the modifications of these he first revises the notions of being 'the same $\mathscr{S}$:'

Definition 35 ('the same $\mathscr{S}$' with non-existents; p. 71)**.** d in w is the same $\mathscr{S}$ as d' in w' iff $d \neq i^*(w)$ and $d' \neq i^*(w')$ and there is an individual concept, i, which belongs to $\mathscr{S}$ at some world, and $i(w) = d$ and $i(w') = d'$.

Deploying this new version of 'the same $\mathscr{S}$' then has the effect of changing the sense of 'an $\mathscr{S}$ counterpart' (abbreviated $\mathscr{S}(w', d, w)$), and a 'world variant' (abbreviated $f(w', a, w)$), although the wording of the definitions of these notions can stay the same.

Now the intuition that Gupta tries to capture with the semantics of necessity is "an object d of the sort $\mathscr{S}$ satisfies $\Box Fx$ in w iff d satisfies Fx in w, and at all worlds w' at which $\mathscr{S}(w', d, w)$ is defined, $\mathscr{S}(w', d, w)$ satisfies Fx in w'" (Gupta, 1980, p. 71). He does this with the following in L_2, which replaces Definition 34.(ii):

Definition 36 (Box rule for L_2; p. 72)**.** Let M, w, a, and α be as above:

(i) If α is the formula $\Box A$, then $V^w_{M,a}(\alpha) = T$
if $V^{w'}_{M,f(w',a,w)}(A) = T$ *at all worlds* w' *at which* $f(w', a, w)$ *is defined.*
Otherwise $V^w_{M,a}(\alpha) = F$.

This definition runs into serious trouble, one aspect of which I eluded to in footnote 33, because $f(w', a, w)$ is undefined at a world w' now whenever there is no $\mathscr{S}$ variant at a w' for one of the variables that gets assigned an object by a. This will be true, even if the variable in question does not figure in the formula under consideration. As a result, the semantic values of the formula at worlds which should be considered end up being discounted. And even though Gupta makes progress on this issue in the semantics of L_3, there are still serious issues (Gupta, 1980, cf. p. 72-75).

To respond to these, in L_4 Gupta gives up on the assignment of sorts and objects to variables and instead just assigns them individual concepts. With necessity, he now takes into account all of the worlds.

Definition 37 (Box rule for L_4; p. 76)**.** Let M, w, a, and α be as above:

(i) If α is the formula $\Box A$, then $V^{w}_{M,a}(\alpha) = T$
if $V^{w'}_{M,a}(A) = T$ at all worlds $w' \in W$. Otherwise $V^{w}_{M,a}(\alpha) = F$.

Now that he is not doing anything to discount those cases where the objects figuring in A in various worlds have * as their extension in that world, only the cats of *Figure* 3 on p. 15 will count as necessarily cats, not the ones in *Figure* 2. That is, since $C(eh*) = F$ in γ_3, $eh*$ will not necessarily be a cat, even though it is one in all of the cases in which it exists. In this way, with L_4 Gupta gives up on accommodating our 'rough definition' of necessity (Definition 9).

References

Belnap, N. (2006). Bressan's type-theoretical combination of quantification and modality. In Lagerlund, H., Lindström, S., and Sliwinski, R., editors, *Modality Matters: Twenty-Five Essays in Honour of Krister Segerberg*, volume 53, pages 31–53. Dept. of Philosophy, Uppsala University, Sweden.

Belnap, N. and Müller, T. (2014a). BH-CIFOL: Case-intensional first order logic. *Journal of Philosophical Logic*, 43(5):835–866.

Belnap, N. and Müller, T. (2014b). CIFOL: Case-intensional first order logic. *Journal of Philosophical Logic*, 43(2):393–437.

Bressan, A. (1972). *A General Interpreted Modal Calculus.* Yale University Press.

Bressan, A. (1993). On Gupta's book *The Logic of Common Nouns. Journal of Philosophical Logic*, 22(4):335–383.

Cresswell, J. and Hughes, G. (2004). *A New Introduction to Modal Logic.* Taylor & Francis.

Cresswell, M. (1975). Hyperintensional logic. *Studia Logica*, 34(1):25–38.

Dowty, D., Wall, R., and Peters, S. (1981). *Introduction to Montague Semantics*, volume 11 of *Synthese language library.* D. Reidel Publishing Company.

Fine, K. (1995). The logic of essence. *Journal of Philosophical Logic*, 24(3):241–273.

Gupta, A. (1980). *The Logic of Common Nouns: An Investigation in Quantified Modal Logic.* Yale University Press.

McCawley, J. (1982). Review of *The Logic of Common Nouns. Journal of Philosophy*, 79:512–517.

Quine, W. V. O. (1953, 1980). Reference and modality. In *From a Logical Point of View: 9 Logico-philosophical Essays*, Logico-philosophical essays. Harvard University Press, 2nd edition.