Exclusively indexical deduction

Dekker, P.

DOI
10.1017/S1755020316000125

Publication date
2016

Document Version
Final published version

Published in
Review of Symbolic Logic

THE REVIEW OF SYMBOLIC LOGIC
Volume 9, Number 3, September 2016

EXCLUSIVELY INDEXICAL DEDUCTION

PAUL DEKKER
ILLC/Department of Philosophy, University of Amsterdam

Abstract. This paper presents a proof system for discourse representation theoretic reasoning and dynamic predicate logical inference. It gives a sound and complete characterization of the dynamic declaration of discourse referents and the essentially indexical means to refer back to them. The indexical outlook upon discourse reference is argued to further our understanding of some issues deemed relevant both theoretically (philo-logically) and practically (computationally).

The framework of discourse representation theory (*DRT*, Kamp 1981; Kamp, van Genabith & Reyle 2011; Heim 1982) has been developed to model the dynamic construction of representations of the contents of an unfolding discourse. *DRT* has initially been proposed to marry logical and cognitive approaches to meaning, it has been used to tackle logical and linguistic puzzles in the interpretation of discourse, and it has been applied widely and successfully in the computational semantics of natural language.

Besides displaying some well-motivated idiosyncracies, the (discourse) representations that *DRT* produces are essentially classical and so is their interpretation and so is their proof-theory. These representations are composed, in *DRT*, at an intermediate level that especially, and essentially, serves the establishment of discourse reference. Such composition, too, has been given several, nonclassical, interpretations—in terms of a dynamic semantics or compositional *DRT*. For these interpretations, however, no tailor-made proof theory exists. This paper provides proof-theoretic tools to cope with precisely that: establishing discourse referential inference.

The tools enabling discourse reference, with associated inference rules, are introduced on the basis of a most minimal deduction calculus for a propositional logic, so as to highlight and focus upon the minimally required machinery. The minimal propositional logic is presented in §1. Tools for discourse reference, and associated deduction rules, are presented and motivated in §2 and §3, respectively. The resulting system is referred to as *PLI*, Predicate Logic with Indices. §4 settles the relation between *PLI* and classical first order predicate logic, and establishes completeness of the calculus on its intuitive, dynamic, interpretation. §5 discusses some practical and theoretical benefits of the indexical take on discourse reference that this paper can be seen to endorse.

§1. Propositional string calculus.

In order to focus on the tools that are minimally required to enable discourse reference, I will start off from a most minimal propositional logical deduction system operating with only one connective. It may belong to common wisdom that we can do propositional calculus with one connective only (Peirce, 1880; Wittgenstein, 1922), and the most suitable connective to start with is the so-called Sheffer Stroke '|' (Sheffer, 1913). A proposition written as '$(p \mid q)$' has to be read as 'not both

Received: September 4, 2015.

doi:10.1017/S1755020316000125

p and q'. Although 1 (one) is of course the smallest number of connectives that one can meaningfully employ, the usual calculus can still be simplified and generalized somewhat more. Once the connective ('|') is the only one assumed, it can be left out, and we may choose to write '(pq)' instead of '$(p \mid q)$'. Also, since the negation of p is rendered as '$(p \mid p)$', now '(pp)', we may choose to write this less redundantly as '(p)'. And once we allow the brackets to embrace a number of one proposition, and of two propositions, it is just tempting to allow them to accommodate any finite number of propositions. And that is what we are going to do.

Let us assume some alphabet P, of proposition letters, which may be infinite. Our language, the language of *Propositional String Logic* or *PSL* as we will call it, will include any *string* s of propositions $s_1, \ldots, s_i$, where propositions are either atomic, that is, proposition letters from the alphabet, or *exclusions* of strings of propositions. For any sequence, or string, s of propositions, we count (s) as a proposition, which is the *exclusion* of s. The exclusion (s) states that not all of s are true, or, just simply, that s, as a whole, is excluded.

DEFINITION 1.1 (*PSL*-Syntax). *Given an alphabet of proposition letters P,*

- *a* PSL*-string is a finite sequence of* PSL*-propositions;*
- *a* PSL*-proposition is an atomic proposition $p \in P$ or the exclusion (s) of a* PSL*-string s.*

By the above definition we allow for an empty string of propositions, and this, as we will see, figures as a $\top$-element.[1] I take a string of propositions to characterize the world as being a certain way, and the empty string then figures as a way of saying the world is, in colloquial terms, 'some way or other'. By bracketing, excluding, the empty string, writing '( )', we of course obtain a bottom element $\bot =_{def} (\top) = (\ )$. This (nonempty) string excludes the world to being 'some way or other', that is, it excludes the world, and, hence, it is always false.

In what follows I will use r, s, and t, as (meta-)variables for strings—always finite, and possibly empty—and simply indicate the concatenation of r, s, and t by their concatenation rst. When a string rst is excluded, as in the proposition (rst), we have various options of phrasing this. The proposition (rst) most directly states that rst is excluded, but this means, by the same token, that r excludes st, and, equivalently, e.g., that rs excludes t.[2]

Understanding concatenation as a form of conjunction, and exclusion as a form of negation, it is obvious how to relate *PSL*-strings to standard propositional logical formulas. That is, $p \wedge q$ can be equated with pq, $\neg p$ can be equated with (p), and, thus, $p \rightarrow q$ and $p \vee q$ can be equated with $(p(q))$ and $((p)(q))$, respectively. All of this of course easily generalizes to strings and formulas of any complexity.[3]

[1] The empty string is usually rendered in official notation as $\langle\ \rangle$, but we actually take it to be: ␣. In the previous sentence '␣' does *not* belong to the empty string, but is employed as the LaTeX-sign that makes (empty) space text-visible, and it really underscores: .

[2] It can be revealing technically, but perhaps not intuitively, that (rst) by the same token expresses that rst excludes the world to being 'some way or other', and, equivalently, that the world 'whatever' excludes rst. Both actually indicate that rst is excluded itself, and this is precisely what all these formulations mean.

[3] It may be noted that it is not for being conservative that I define the expressions of the language as strings. For the purposes of classical propositional logic the order of the strings is practical, as usual, but redundant and immaterial. However, for the purposes of dynamic first order logics, like the one developed in the next sections, order is quintessential. For these aim to capture,

Truth of *PSL*-strings is, as usual, defined relative to a situation—formally a valuation—in which some (or all, or none) of the proposition letters are true. The corresponding valuation $V(p) \in \{0, 1\}$ indicates that p is true (iff $V(p) = 1$) or false (iff $V(p) = 0$), for any $p \in P$. We say that a valuation (situation) V satisfies a string s iff $V \models s$, according to the following definition.

DEFINITION 1.2 (*PSL*-Semantics).

- $V \models s_1 \ldots s_j$ *iff* $V \models s_i$, *for all* $1 \leq i \leq j$;
  $V \models p$ *iff* $V(p) = 1$, *for* $p \in P$;
  $V \models (s)$ *iff* $V \not\models s$.
- *String* r *entails string* s, $r \models s$, *iff* $V \models s$ *whenever* $V \models r$.

A string of propositions s is satisfied iff all propositions p in the string s are satisfied. The exclusion (s) of s is satisfied iff s is not satisfied. The concept of (semantic) entailment is defined in the usual way.

We can establish that, by definition, the empty string is valid (entailed unconditionally), and that anything is entailed by its exclusion. Hence the empty string indeed figures as a $\top$-element ("Verum"), and ( ) as a bottom element $\perp$ ("Falsum").

The notion of semantic entailment can be characterized syntactically, proof-theoretically, by means of a Fitch-style calculus of natural deduction (Fitch, 1952). This calculus serves to define the derivability of a conclusion s from a sequence of premises $r_1, \ldots, r_n$ in terms of the existence of a proof of r from assumptions $r_1, \ldots, r_n$.

A proof is a deduction which consists of a numbered and annotated list of, in the present case, *PSL*-strings, all of whom are either assumptions, or derived from *PSL*-strings earlier in the list by means of the deduction rules specified below. (The annotations unambiguously indicate according to which of these rules a line is added to the list.) If a rule is applied the annotation may be referring to items in the list, provided that these are not an assumption or dependent on assumptions that are withdrawn when the rule is employed. Such 'inaccessible' items—withdrawn assumptions or lines dependent on them—are, as usual, indicated by means of vertical lines. If, by an application of one of the rules stated below, an assumption gets withdrawn, it is always the very last assumption in the list that has not (yet) been withdrawn.

We say that a conclusion s is derivable from a sequence of assumptions $r_1, \ldots, r_i$, $r_1, \ldots, r_i \vdash s$, iff there is a *derivation*

1. r_1 [ass.]	
$\vdots$	
i. r_i [ass.]	without any pending assumptions (or declarations) between lines i and n.
$\vdots$	
n. s	

In a proper derivation every line specifies either an assumption, or a conclusion that follows from previous lines by logical rules of deduction. As said, there may be no further active assumptions, not withdrawn, besides the ones from which the conclusion is said to follow. (The ban on further declarations can only be explained below.)

not the dynamics in the world the reasoning is about, but the linear flow of information in reasoning.

The *PSL*-deduction rules are quite simple and intuitive, which is one of the reasons why I have chosen for the present set up. Assuming, as indicated above, a classical, two-valued understanding of the system, the calculus only needs rules for the use and introduction of exclusions, and a contradiction rule. (Effectively, a rule of excluded middle.) The rule of use is labeled a rule of ‘Exclusion’ (‘X’), and the introduction rule one of ‘Retraction’ (‘R’).

Exclusion (X)

$$\begin{array}{lll} & \vdots & \\ \text{i.} & r & \\ & \vdots & \\ \text{j.} & (rs) & \\ & \vdots & \\ \text{n.} & (s) & [\text{X, i, j}] \end{array}$$

Retraction (R)

$$\begin{array}{lll} & \vdots & \\ \text{i.} & r & [\text{ass.}] \\ & \vdots & \\ \text{n-1.} & (s) & \\ \hline \text{n.} & (rs) & [\text{R}] \end{array}$$

Contradiction (C)

$$\begin{array}{lll} & \vdots & \\ \text{n-1.} & ((s)) & \\ \text{n.} & s & [\text{C}] \end{array}$$

The exclusion rule demonstratively figures as the principal deduction pump. If we have r (line i), and r excludes s (line j), then s is excluded, as is concluded on line n. The retraction rule allows us to conclude to an exclusion by hypothetical reasoning. If, in a certain context, the assumption that r entails the exclusion of s, then we can conclude that r excludes s, in that context. The contradiction rule classically states that once it is excluded that s is excluded, then s can be concluded. And that’s it. The rules are quite simple indeed, but sufficient for propositional logical reasoning, as I will now proceed to show.

Although the exclusion rule figures as a kind of deduction pump, the empty string, semantically valid, does not need to be pumped up from anything. It can be proven valid, i.e., concluded from no premises.

THEOREM 1.3 (Sequitur Verum).

$$\vdash$$

Proof. The following derivation serves as a proof:

$$\begin{array}{lll} \text{1.} & (\,) & [\text{ass.}] \\ \hline \text{2.} & ((\,)) & [\text{R}] \\ \text{3.} & & [\text{C}] \end{array}$$

□

The assumption ( ) on line 1 leads to, actually *is*, the exclusion of the empty string. The conclusion of this on line 2 is that the hypothetical assumption ( ) itself is excluded: (( )). Line 3 concludes, from the fact that the exclusion of the empty string is excluded, that the empty string holds.

The empty string being shown generally valid, its exclusion, the falsum ( ), can be used to derive anything. For any arbitrary string s:

THEOREM 1.4 (Ex Falso Sequitur Quodlibet).

$$(\,) \vdash s$$

Proof. The following derivation proves the point:

1. ( )	[ass.]	
2. (s)	[ass.]	
3. (( ))	[ass.]	[Lines 3–5 here only serve to reproduce, on line 6, the contradiction we already started from at line 1. Repetition of a line is always possible, of course, as we will see below.]
4. ( )	[X, 1, 3]	
5. ((( )))	[R]	
6. ( )	[C]	
7. $((s))$	[R]	
8. s	[C]	

□

The next theorem may serve to show that the calculus also covers more familiar and substantial ground.

THEOREM 1.5 (Subtraction and Addition). *The following two deduction rules are licensed by the three deduction rules above:*

Subtraction (−)

$$\begin{array}{ll} & \vdots \\ i. & rst \\ & \vdots \\ n. & s \quad [-, i] \end{array}$$

Addition (+)

$$\begin{array}{ll} & \vdots \\ i. & rt \\ & \vdots \\ j. & s \\ & \vdots \\ n. & rst\ [+, i, j] \end{array}$$

Proof. In the appendix we show that $rst \vdash s$, and that $rt, s \vdash rst$. The derivations shown there easily generalize to the schematic rules formulated in this theorem. □

It may have to be emphasized that the derived rules are valid in full generality. They hold for arbitrary r, s, and t, so also if one of these is the empty string. Thus, where r is the empty string, subtraction tell us that $st \vdash s$, which, properly understood, corresponds to the familiar rule of conjunction elimination, extracting the left conjunct. If t is taken to be the empty string, the rule corresponds to the extraction of the right conjunct: $rs \vdash s$. And, as already alluded to above, if both r and t are empty, the rule of subtraction collapses into a rule of repetition: $s \vdash s$. The addition rule, analogously, covers three types of concatenation. It may figure as the insertion rule that it appears to be. However, if r is set to be the empty string, it corresponds to a converse conjunction introduction rule ($t, s \vdash st$), while a nonconverse conjunction rule ($r, s \vdash rs$) shows up if we take t to be the empty string.

The present findings already suffice to draw the following conclusion.

THEOREM 1.6 (Completeness for *PSL*).

$$r \models s \textit{ iff } r \vdash s$$

Proof (Sketch). Soundness is easily checked. As for completeness, notice that there is an obvious translation to and from a classical propositional logic with ¬ (negation) and ∧ (conjunction) only. Since such a fragment is functionally complete, and since *PSL* accommodates mirror images of the classical rules for negation and conjunction, the system itself is complete. □

The rules for conjunction are mirrored by the subtraction and addition rules. The rule for the use of a negation is taken over by the exclusion rule, restricted to cases in which the exclusion leads to the falsum ( ). Similarly, the introduction of a negation is taken care of by the retraction rule, viz., by those of its instances in which an hypothetical assumption is brought to a contradiction. We have already seen that we have an Ex Falso rule, and a double negation rule is directly mirrored by our contradiction rule.

It may have occurred to the reader that the three deduction rules, which I think are as elementary and transparent as can be, are rather tedious and laborious to actually and strictly work with. The system, however, becomes agreeable once we allow ourselves the (logically justified) use of derived inferences and inference schemes, like those of subtraction and addition. We also have, e.g., modus ponens: $(r(s)), r \vdash s$, of course, and contraposition. It is fairly easily seen that $rs \vdash t$ iff $rs(t) \vdash ()$ iff $r \vdash (s(t))$, with any of r, s, or t possibly the empty string.

§2. Introducing discourse referents. The system of *PSL* has been presented here as a minimal ground on which to grow devices for discourse reference. If we want to go in the air, we have to grow first order, in syntax and intended semantics, and allow talk of and reasoning about the ascription of properties to 'individuals', and about the relations in which they stand.

Now it seems to me that logic, by itself, does not give us such individuals. Logic should enable one to state the assumption that individuals exist (and the existence of individuals with certain properties), but logic should be possible also without such an assumption.[4] So, to begin with, we need a means to actually *state*—or assume, or derive—the existence of an individual. For this purpose I introduce a device to issue such a declaration, an elementary proposition '1' that serves as an arbitrary sign that an object or individual exists—without any properties associated (yet).[5] We can now think, e.g., of a string '$p\ 1\ q\ 1\ r$', that consists of five atomic propositions, three of which are proposition letters, 'p', 'q' and 'r', and the string hosts two declarations. Notice that this string does not state the existence of two individuals. While it states the existence of an individual, and while it does so twice, it does not guarantee they are distinct.

A declaration '1' can be taken to introduce a discourse referent, an arbitrary individual, subject to further, possibly hypothetical, specification, description, and reasoning.[6] Subtly, but significantly, I do not consider it essential that a discourse referent comes under a name, or gets labeled by a variable. It is without any doubt very convenient to actually employ names or labels for discourse referents, for the purpose of exchanging information

[4] In the formulations I have employed above, logic does come with the assumption that "there is a world". However, these locutions only served to express the essentially logical assumption that there is truth.

[5] Of course one may want to distinguish types or sorts of individuals or objects or stuff, and use 1_t, 1_e, 1_i, ..., or t_1, e_1, i_1, ..., to indicate (the assumption of) the existence of a time, event, instance, ...

[6] Such a declaration should indeed be seen to mark the introduction of a 'discourse referent' in the sense of Lauri Karttunen and Bonnie Lynn Webber, also pictured as the 'creation of a file card' by Irene Heim. (See Karttunen 69/71; Webber 1978; Heim 1982.)

about particular individuals, no matter fictional or real. However, as we will see, for proof-theoretic purposes they are not needed, and as a matter of fact names or variables rather tend to complicate matters, logically speaking. For this reason I will keep to a pure device for introducing discourse referents, as said, the atomic proposition of declaration '1', and render the naming of them entirely optional.

Of course, logic, like natural language, cannot do without a means to distinguish and identify discourse referents, once several of them have been introduced. Discourse referents may be declared anywhere in a string or proof. Their first (and last, and only) identifying properties then consist in the place in the string or proof where they are introduced. This locus of introduction is uniquely identifiable from within the same string or proof, in an entirely indexical manner, e.g., as "the discourse referent declared three lines above the present line". Since declarations are the only relevant distinctive acts in a string or proof, we do not need to invoke lines or other idiosyncratic features of a string or proof, and we can identify a discourse referent as the one introduced by the i-th declaration before the locus of an identifying term. And this identifying term then can be abbreviated to simply $\imath$. In short, we can, and will, use indices as referring devices, as terms. An index $\imath$, on any locus of its use as a term, refers to the discourse referent introduced i declarations back in the discourse or proof before its occurrence.

Thus we can interpret a term ('1') as the first (nonsuspended) declaration ('1') that you find when you go back in a proof, that is, when you move, from its locus of occurrence, leftwards in a string, and upwards in a proof. Surely, any other discourse referent in a string or proof can, similarly, and uniquely, be identified and retrieved by an indexical term. For, for some number i, it will have been introduced by precisely i declarations before the occurrence of that term.[7]

Armed with these two new devices, declarations 1 and indexical terms $1, 2, \ldots$, we can develop our first order string language. Like in standard first order predicate logic I assume a sorted vocabulary of j-ary relational predicates, the nullary ones just being proposition letters, and the unary ones ordinary predicates. Indices $1, \ldots$ figure as terms.[8] If R is a j-ary relation expression, and υ is a sequence of j terms, then $R\upsilon$ is an atomic proposition. As said, I use '1' for declarations. This gives us three sorts of propositions: atomic propositions, declarations, and exclusions.[9] As before, strings are (finite) strings of propositions, and exclusions are exclusions of (finite) strings of propositions. This gives us a language of *Predicate Logic with Indices*, or *PLI*, as I will call it, formally defined as follows.

---

[7] This use of indices replicates Nicolaas de Bruijn's use of indices in type theory. There, the occurrence of an index i at a position indicates that that position is bound by the i-th λ found when going up in the construction tree, from its locus of occurrence (de Bruijn, 1972). Jan van Eijck 2001 uses a similar, 'reverse', method in his 'incremental dynamic logic'. Discourse referents are identified there by counting, calendar-style, from the beginning of a discourse. The beginning is explicitly, but also indexically, indicated in the syntax of evaluable formulas.

[8] On occasion I will also employ 0 as a dummy term, a distinguished and fixed variable, which is replaced eventually by proper terms. The term itself is—for the time being—not considered part of the language. In the final section, however, we will expand somewhat on making sense, and even use, of this zero-index.

[9] Since I want to focus, primarily, on deduction in this paper, I sidestep the ontological and epistemological issues related to naming and identity, not even mentioning their social embedding. Surely, when it comes to philosophically and linguistically interesting applications, we will have to introduce names, or individual constants, and identity, '='.

DEFINITION 2.1 (*PLI*-Syntax). *Given a sorted vocabulary* P,

- *a* PLI-*string is a finite string of* PLI-*propositions;*
- *a* PLI-*proposition is an atomic declaration* 1*, an atomic proposition* $R\upsilon$ *or the exclusion* (s) *of a* PLI-*string* s*;*
- *an atomic proposition* $R\upsilon$ *consists of a* j*-ary predicate expression* $R \in P$*, and a sequence* $\upsilon = \upsilon_1 \ldots \upsilon_j$ *of indices.*

Let me briefly run over some examples of this somewhat unorthodox language. A string '1 D1' reads that there is something and that *it* is D, or, for short, that there is a D. A string '1 D1 R21' reads that there is a D and that the thing introduced *before it* Rs it. A string '1 F1 1 D1 R21' reads that there is an F and a D and that the F Rs the D. (In more classical terms this is rendered as $\exists x(Fx \wedge \exists y(Dy \wedge Rxy))$, translating, for instance, "Some Farmer Raises a Dog.") A string '1 F1 (1 D1 R21)' reads that there is an F and that it is excluded that there is D such that the F Rs the D. (More familiarly $\exists x(Fx \wedge \neg\exists y(Dy \wedge Rxy))$, translating "Some Farmer does not Raise a Dog.") A string '(1 F1 (1 D1 R21))' finally reads that it is excluded that there is an F such that it is excluded that there is D such that the F Rs the D. (This translates the rather classical "No Farmer does not Raise a Dog", or, better, "Every Farmer Raises a Dog" $\forall x(Fx \rightarrow \exists y(Dy \wedge Rxy))$.)[10]

An index $\imath$, or, better, any occurrence of this index in a string, unambiguously refers to the i-th declaration before its locus of occurrence. Of course this is no guarantee that such a declaration exists. If such a declaration exists, that is, if the occurrence of $\imath$ in a string s is preceded by i nonexcluded declarations in s, I will say that (the occurrence of) the term is *resolved*, in s—unresolved otherwise. In what follows we need to be able to specify when an index, or its occurrence, counts as resolved, and, more generally, when a whole string does. For this we need to establish some, relatively straightforward, technical vocabulary. For that purpose I specify the number of discourse referents declared in a string, and that of its reach, or range, over previous discourse. The number n_s of discourse referents declared in a string s is the number of nonexcluded declarations in s; the range r_s is the required number of declarations to be supplied for s to become resolved; a string s then can be said to be resolved if its range is zero.

DEFINITION 2.2 (Resolution).

- *The number* n_s *of discourse referents declared in a string* s *satisfies:*

  $n_s = 0$ *if* s *is the empty string, atomic, or an exclusion;*
  $n_s = 1$ *if* s *is a declaration;*
  $n_s = n_r + n_t$ *if* s *is a string* rt.

- *The number* r_s *of discourse referents required by a string* s *satisfies:*

  $r_s = 0$ *if* s *is the empty string, or a declaration;*
  $r_{R\upsilon} = \mathrm{MAX}(\upsilon)$*;* $r_{(s)} = r_s$*;* $r_{st} = \mathrm{MAX}(r_s, r_t - n_s)$.

- *A string* s *is* resolved *iff* $r_s = 0$.

[10] Let me emphasize that I have no intention to advocate the present language as providing us with readable, linguistically revealing, translations of natural language sentences. Even so, in the final section of this paper I do want to demonstrate that, in its terms, we can supply logically transparent and computationally convenient representations for them.

It is easily checked that this definition adequately renders the intuition behind it. Among the propositions only declarations contribute a (single) discourse referent. An atomic proposition or an exclusion contributes none. (Obviously an excluded string s may contribute a discourse referent, but in the exclusion (s) the discourse referent is only part of something excluded, not contributed.) The number of discourse referents declared in a nonatomic string of propositions is just the sum of the numbers of discourse referents declared in its parts.

Of course the empty string does not impose on previous discourse, whence its range is zero, and similarly for (atomic) declarations. The range, the number of discourse referents required by an atomic proposition $Ri_1 \ldots i_j$ equals the highest index among the terms $i_1, \ldots, i_j$. If at least that number of discourse referents is supplied by previous discourse, then all indices in that atomic string are resolved. An exclusion does not increase, nor decrease, the range of the excluded string.

The requirement issuing from a string st equals the strongest (highest) among the requirements issuing from s and t, observing that those of the latter have possibly been reduced by declarations in the string s preceding t. The requirements issuing from t in the sequence st are not given by r_t, but by $r_t - n_s$.[11]

I will be brief about the semantics of *PLI*. It is that of a notational variant of a fragment of the so-called *PLA*-language detailed in (Dekker, 2012), predated by my own (Dekker, 1994). It is stated as a relation characterizing the satisfaction of *PLI*-strings s in models M by sequences of individuals e, written as $M, e \models s$. A model $M = \langle D, I \rangle$ consists of a nonempty[12] set of individuals D and an interpretation $I(R) \subseteq D^j$ of j-ary relational predicates R, as a set of j-tuples of individuals. This is as usual. The sequences e witness the possible values of (sequences of) discourse referents, both those declared by the strings at issue, as well as those required or assumed in previous discourse. (It is assumed, throughout, that the witness sequences are of sufficient length. It will only be said that $M, e \models s$ (or, for that matter, $M, e \not\models s$) if e is a sequence of at least $r_s + n_s$ individuals.)

DEFINITION 2.3 (*PLI*-Semantics).

$$
\begin{array}{ll}
M, de \models 1 & \textit{iff } d \in D; \\
M, e \models Ri_1 \ldots i_j & \textit{iff } \langle e_{i_1}, \ldots, e_{i_j} \rangle \in I(R); \\
M, e \models (s) & \textit{iff } M, ce \models s \textit{ for no } c \in D^{n_s}; \\
M, e_j \ldots e_1 \ldots \models s_1 \ldots s_j & \textit{iff } M, e_i \ldots e_1 \ldots \models s_i \textit{ for all } 1 \leq i \leq j, \\
\end{array}
$$

$$\textit{where } e_i \in D^{n_{s_i}} \textit{ for all } 1 \leq i \leq j.$$

The semantics is fairly simple. Any individual satisfies a declaration—so a declaration only requires there to be any. A sequence of individuals satisfies an atomic formula if the targeted individuals stand in the said relation, in the order given.[13] An exclusion is satisfied iff there are no such (tuples of) individuals that it declares excluded. This is to say that no

[11] Notice that, semantically speaking, nothing stands in the way of alternative methods of resolution, like, for instance, intended co-instantiation with future discourse referents. When constructing a proof or deduction, however, such does not seem to make much sense.

[12] But, see footnotes to follow.

[13] In case R is a nullary predicate, i.e., a proposition letter, its truth (satisfaction) consists in its interpretation $I(R)$ having the empty sequence as an element, i.e., in being 1 according to the usual conventions.

sequence of witnesses satisfies the referents declared in it. A string $s = s_1 \ldots s_j$ is satisfied by a sequence of witnesses $e_j \ldots e_1 \ldots$ iff each proposition s_i, for $i \leq j$, is satisfied by $e_i \ldots e_1 \ldots$. The latter sequence is the original sequence with the witnesses stripped of for declarations made *after* s_i in s. The idea is that each substring s_i may introduce discourse referents, witnessed by satisfying sequences e_i, which can subsequently be constrained further by continuations of s_i.

If we conceive of a string s as composed of two substrings r and t, so if $s = rt$, then we can spell out the satisfaction conditions as those of a dynamic conjunction, like in (Dekker, 1994, 2012). For, we find that:

$$M, ace \models rt \text{ iff } M, ce \models r \text{ and } M, ace \models t,$$
$$\text{where } c \in D^{n_r} \text{ and } a \in D^{n_t}.$$

It may help to conceive of this interpretation as a dynamic or incremental one. One starts the interpretation of a *PLI*-string rt with a sequence e of witnesses, which serve as potential referents (witnesses) of indices unresolved in rt. The substring r subsequently introduces an additional number of discourse referents, witnessed by a sequence of individuals c, and to the effect that the combined sequence ce provides the potential referents of indices which are themselves unresolved in the next substring t. This substring t may in addition contribute further discourse referents, witnessed by a satisfying sequences a, which ends up in front of the whole satisfying sequence, as the most nearby candidates for, again, subsequent coreference.

§3. Indexical deduction.

When one wants to say today, what one has expressed yesterday using the word 'today', one should replace this word by 'yesterday'. (Frege, 1918, p. 64)

Gottlob Frege here makes a point that we must take to heart in all that follows. If we say Fi, that a discourse referent introduced i declarations ago is an F, a farmer, for instance, and if we want to say exactly that again, after m more declarations have been made, we have to say $F i{+}m$, that the discourse referent introduced $i{+}m$ declarations ago is an F. Such is trivial, annoying, indeed, but unavoidable in the present setting, and surely not insurmountable.

Before we turn to the rules of deduction in *PLI*, I will therefore have to introduce one or two more notation devices. As I hope one will see, their definition is involved, but the concept is simple. Consider the following snapshot of a derivation, with one term ${}^{i\leq}s$ to be explained and defined.

$\vdots$
i. s
$\vdots$
n. ${}^{i\leq}s$

The string ${}^{i\leq}s$ is supposed to say at line n what s said at line i. This means that indices i unresolved in s have to be adjusted, updated, with the fact that from line i and up till line n a specific number, say m, of discourse referents have been declared. These have to be replaced by $i{+}m$. I write ${}^{+m}s$ for the result, which is defined below. In general, if ${}^{i\leq}s$ occurs on some line j, it is short for ${}^{+m}s$, where m is the collective sum $\bigoplus_{i \leq x < j} n_x$ of discourse referents introduced from line i up to, and not including, j. Thus, what ${}^{i\leq}s$ says at any line j after or under i, is exactly what s said, *or would have said* at line i.[14]

Notice, not unimportantly, that ${}^{+m}s$ should not affect or change indices which are resolved in s itself. For this reason ${}^{+m}s$ cannot be defined by a blind substitution of all indices i by $i{+}m$, but only of the unresolved ones. In the definition of ${}^{+m}s$ we therefore have to keep track of an auxiliary counter l which counts the number of terms introduced in the string s itself, a counter that has to be updated itself when moving through the string. At the start this counter is set to 0.

DEFINITION 3.1 (String Update).
The update ${}^{+m}s$ *of* s *is* $[^{+m}_{0}]s$, *where* $[^{+m}_{l}]s$ *is defined by:*

- $[^{+m}_{l}]i = i \quad$ *if* $i \leq l$;
  $[^{+m}_{l}]i = i{+}m \quad$ *if* $l < i$.
- $[^{+m}_{l}]Ri_1 \ldots i_j = R[^{+m}_{l}]i_1 \ldots [^{+m}_{l}]i_j$; $\quad [^{+m}_{l}]1 = 1$;
  $[^{+m}_{l}](s) = ([^{+m}_{l}]s)$; $\quad [^{+m}_{l}]rt = [^{+m}_{l}]r[^{+m}_{l+n_r}]t$.

Like I said, while the concept is simple, its formulation is involved. By way of example, consider $[^{+3}]D1\ O21\ 1\ C1\ O31\ H12$. This equals $[^{+3}_{0}]D1\ O21\ 1\ [^{+3}_{1}]C1\ O31\ H12$, which is $[^{+3}_{0}]D1\ [^{+3}_{0}]O21\ [^{+3}_{0}]1\ [^{+3}_{1}]C1\ [^{+3}_{1}]O31\ [^{+3}_{1}]H12$, i.e., $D4\ O54\ 1\ C1\ O61\ H15$.

Before we move to the *PLI* rules of deduction, I have to come back at some subtle detail in the specification of the notion of a derivable consequence above. A conclusion was said to follow from certain premises if it could be derived from them without further pending assumptions *and declarations*. The reason is, very simply, that the conclusion might otherwise refer back to discourse referents which are introduced in a derivation after the premises have been stated, and which are no longer present when we summarize it as a conclusion following from these premises. Here is a classical, stilted, example, taken from the literature, that serves to illustrate the point. The reasoning in (A) combines two valid inferences, (A′) and (A″), calling on rules already presented above.

(A) It is not the case that there is no Welsh doctor in London. So, there is a Welsh doctor in London. So, he is Welsh.

$((1\ W1\ DL1)) \vdash 1\ W1\ DL1$ *(A′)*
1. $((1\ W1\ DL1))$ [ass.]
2. $1\ W1\ DL1$ [C]

$1\ W1\ DL1 \vdash W1$ *(A″)*
1. $1\ W1\ DL1$ [ass.]
2. $W1$ [−, 1]

However, we cannot cut out the middle string, because it introduces the discourse referent that the conclusion expands upon. The reasoning in (B) is dubious, not because the associated derivation is wrong, but because the derivation should not be summarized in the way it is formulated.

(B) It is not the case that there is no Welsh doctor in London. So,? he is Welsh.

$((1\ W1\ DL1)) \nvdash W1$ *(B′)*
1. $((1\ W1\ DL1))$ [ass.]
2. $1\ W1\ DL1$ [C]
3. $W1$ [−, 2]

[14] The notation ${}^{i<}s = {}^{i+1\leq}s$ can be inferred from this. It is the update of s that takes into account the number of discourse referents that have been introduced *after* line i. It states on its line of occurrence what s would have stated just after line i.

While the derivation under (B′) is correct, we cannot properly summarize it. The above is not to say, of course, that we should not, and need not, make declarations in derivations. Quite to the contrary. The moral is that in a successful derivation discourse referents may get introduced, if that happens under assumptions that eventually are withdrawn, so that upon the withdrawal of these assumptions we witness the withdrawal of the declarations made under them.

We have now set the stage for the presentation of the deduction rules. The three propositional logical rules above are preserved, but two of them need to be adjusted to the current, dynamic, format. Also, another rule needs to be added for the declaration of discourse referents. I first give a (re-)statement of the first three rules, with comments following it, in reverse order.

Exclusion (X)

$$\begin{array}{ll} & \vdots \\ \text{i.} & r \\ & \vdots \\ \text{j.} & ({}^{i\leq}rs) \\ & \vdots \\ \text{n.} & {}^{j<}({}^{r_{i\leftarrow j}}s)\ \text{[X, i, j]} \end{array}$$

Retraction (R)

$$\begin{array}{lll} & \vdots & \\ \text{i.} & r & \text{[ass.]} \\ & \vdots & \\ \text{n-1.} & {}^{i<}(s) & \\ \hline \text{n.} & (rs) & \text{[R]} \end{array}$$

Contradiction (C)

$$\begin{array}{lll} & \vdots & \\ \text{n-1.} & ((s)) & \\ \text{n.} & s & \text{[C]} \end{array}$$

The contradiction rule is left unaffected. The retraction rule essentially states what it stated before. If the assumption that r leads to the exclusion of s, then we conclude that r excludes s. The inference that leads from the assumption that r to the exclusion of s may involve a number of declarations, which we have to pass over when we summarize in conclusion that r excludes s. So we can reach this conclusion (rs) if an update ${}^{i<}s$ of s is excluded which, by the end of the hypothetical derivation, states what s would have stated directly following r, on line i+1, that is.

The exclusion rule has been modified in three respects. The first two adjustments involve mere updates that are required in the present format. The proposition $({}^{i\leq}rs)$ on line j hosts an update ${}^{i\leq}r$ of the occurrence of r on line i. This part of the excluded proposition thus repeats what r stated at line i. The update ${}^{j<}(\ldots)$ likewise states on line n what we could have concluded, directly, after line j.

The truly dynamic potential of the exclusion rule resides in the *instantiation* ${}^{r_{i\leftarrow j}}s$ of s in the conclusion. The occurrence of r on line i figures as a *particular* instantiation of a *generic* occurrence of r that on line j is stated to exclude s. Given this instance of r on line i, we conclude that a corresponding instance of s is excluded. For instance, r may be $1\ M1$, for "There is something which is a man." And (rs) may be $(1\ M1\ I1)$, for "It is excluded that there is something which is a man and which is immortal," or "No man is immortal," for short. We can conclude that the declared man is excluded to be immortal: $(I1)$.

So, crucially, the occurrence of s on line j may host indices that relate to declarations in the occurrence of r on line j. The instantiation of ${}^{r_{i\leftarrow j}}s$ of s on line n has these indices redirected to the corresponding declarations in the occurrence of r on line i. Such is achieved, formally, by (i) *increasing* indices $1, \ldots, n_r$ with a number m (m being the number of declarations between lines i and j to be bridged), and (ii) *decreasing* subsequent indices by the number n_r of declarations no longer present. In the definition of ${}^{r_{i\leftarrow j}}s$, we need, again, a threshold l to keep track of the number of discourse referents introduced *within* s, to the effect that indices resolved in s, those below the threshold, are left unaffected.[15]

[15] In the definition I only specify the crucial and effective clauses; the clauses left implicit are schematic copies of those, also trivial, in the definition of string updates above.

DEFINITION 3.2 (Instantiation). *The instantiation* ${}^{r_i \leftarrow j}s$ *of* s *is* $[^{+m}_{0,n_r}]s$*, where* $m = \oplus_{i<x<j} n_x$ *and* $[^{+m}_{l,n}]s$ *is defined by (effective clauses only):*

- $[^{+m}_{l,n}]i = i,$ if $i \leq l$; $\quad [^{+m}_{l,n}]i = i{+}m,$ if $l < i \leq l{+}n$; $\quad [^{+m}_{l,n}]i = i{-}n,$ if $l{+}n < i$.
- $[^{+m}_{l,n}]rt = [^{+m}_{l,n}]r[^{+m}_{l+n_r,n}]t.$

The first clause in this definition relates to the cases where an index i is below the threshold ($i \leq l$). They are resolved in s and remain unaffected. The second clause establishes the reinstantiation. This clause handles indices in s which, in (rs), target declarations in that generic occurrence of r. These references are redirected to the corresponding declarations in the particular occurrence of r on line i, by increasing them with the number m of declarations to be bridged, that is the number of declarations of those between lines i and j. The third clause handles indices i which are unresolved in rs. These indices are reduced to account for the fact that the directly preceding string r fails now, so that the number of declaration to be bridged is decreased by $n = n_r$. The final clause secures the updates of the threshold.

The following example illustrates essential features of the new rule of exclusion.

(C) There is a farmer. He owns a dog. (He also owns a cat.) No farmer beats a dog that he owns. So, he [the farmer] doesn't beat it [the dog].

$1\ F1,\ 1\ D1\ O21,\ 1\ C1\ O31,\ (1\ F1\ 1\ D1\ O21\ B21) \vdash (B32)$

1.	$1\ F1$	[ass.]
2.	$1\ D1\ O21$	[ass.]
3.	$1\ C1\ O31$	[ass.]
4.	$(1\ F1\ 1\ D1\ O21\ B21)$	[ass.]
5.	$(1\ D1\ O41\ B41)$	[X, 1, 4]
6.	$(B32)$	[X, 2, 5]

In the application of the exclusion rule at line 5, the excluded string s is the substring $1\ D1\ O21\ B21$ from line 4 and the excluding string r is the string $1\ F1$ from line 1. The occurrence of s on line 4 contains two occurrences of an index relating to the occurrence of r on line 4, and these are redirected at line 5 to the occurrence of r on line 1. The instantiation ${}^{r_i \leftarrow j}s$ of s therefore reads ${}^{r_1 \leftarrow 4}s = [^{+2}_{0,1}]s$, which is $1\ D1\ O41\ B41$. In the application of the exclusion rule at line 6 the excluding string r is the string $1\ D1\ O21$, and ${}^{i\leq}r$ is ${}^{+2}r$ is the corresponding generic substring $1\ D1\ O41$ at line 5. The excluded substring s from line 5 is $B41$. The instantiation ${}^{r_i \leftarrow j}s$ of s is ${}^{r_2 \leftarrow 5}s = [^{+1}_{0,1}]s$, which redirects index 1 by an update with $m = 1$, and reduces index 4 by $n_r = 1$. The result reads $B32$ which is eventually excluded. The occurrence of the index 3 on this line relates to the farmer declared on line 1, and the occurrence of the index 2 relates to the dog introduced on line 2.

Interestingly, it may happen that the instantiation of an excluded string redirects an index in that string to a declaration that is also directly addressed by an index in the very same string. Upon reflection, this is as it should be as the following example may serve to show.

(D) There is a farmer. Every farmer adores him. So, he adores himself.

$1\ F1,\ (1\ F1\ (A12)) \vdash A11$

1.	$1\ F1$	[ass.]
2.	$(1\ F1\ (A12))$	[ass.]
3.	$((A11))$	[X, 1, 2]
4.	$A11$	[C, 3]

On line 2 we find an instance of s which is (A_{12}). The first index, $_1$ here, relates to the generic declaration in $1\ F_1$ on line 2, and the exclusion rule dictates that it is redirected to the instantiating declaration on line 1. This declaration on line 1, however, is also one that the second index $_2$ on line 2 relates to. So in the end they turn out coreferential. Formally, it works out as follows. The redirection of the first index $_1$ does not actually involve a *change* of the index, because no declarations have been made between lines 1 and 2.[16] The (unresolved) second index is reduced by 1, in order to secure that it targets the declaration that it did on line 2. As a result the conclusion reads $((A_{11}))$, from which we (classically) conclude A_{11}.[17]

The set of deduction rules for *PLI* is completed by a rule for the introduction of discourse referents. If we have established a certain body of information about something introduced any number—say j—of declarations earlier, then this rule allows us to infer that there is something such as established. This effectively is the familiar rule of existential generalization appearing in the clothing of *PLI*.

In order to mark the positions in a string s that relate to the j-th referent before s we now use the dummy numeral 0 in a dummy string s', and the correlation with the j-th discourse referent preceding s then is given by the substitution $_{[j/0]}s'$. The substitution is defined as follows.

DEFINITION 3.3 (Substitution). $_{[j/0]}s$ *is defined by (effective clauses only):*

- $_{[j/0]}i = j$ *if* $i = 0$ *and* $_{[j/0]}i = i$ *otherwise;*
  $_{[j/0]}rt = {}_{[j/0]}r\ {}_{[j+n_r/0]}t$.

The result is a string in which all occurrences of 0 (if any) are replaced by an index that *there*, where the substitution is actually realized, relate to the j-th declaration preceding s. The introduction rule likewise substitutes these for indices that relate to a discourse referent declared new.

Introduction (I)

$$\begin{array}{ll} & \vdots \\ \text{i.} & _{[j/0]}s \\ & \vdots \\ \text{n.} & ^{i\leq}1\ _{[1/0]}{}^{+1}s\ [\text{I}, \text{i}(j)] \end{array}$$

What $_{[j/0]}s$ states, in the above scheme on line i, about the j-th referent before s, is what $_{[1/0]}{}^{+1}s$ says on line n about the discourse referent declared on the very same line just before that occurrence of s. The rule accommodates two updates. First, ^{+1}s says what s

[16] We can still speak of "redirection" because the *location* of the index has changed.

[17] Without any doubt examples like these can be discussed more easily if we have associated labels, or variables, with the various declarations. Such a practice is, of course, actually familiar from the work done in *DRT* and dynamic semantics, and I will also eventually advocate it. It should be noted, though, that it turns out to be tricky business to try and formulate the exclusion rule in terms of such labels, instead of indices. As has already been indicated, the mechanism of indexical discourse reference is fully effective and unambiguous. Labels, or variables, by contrast, only work efficiently if they are *employed* equally unambiguously. Now it is surely easy to conform to common practice and always use "new variables" when new declarations are made. The logical rules themselves, however, ought not to constrain the freedom required in their practical application. Logically speaking, labels or variables or names turn out to be a stand in the way.

would have said if 1 had not occurred before it, as actually happens now on line n. (Notice, not unimportantly, that this update does not, by definition, affect the index 0.) Second, the update ${}^{i\leq}\_$ ensures that what is said at line n is what would have been said at line i.[18]

The following example makes crucial use of the introduction rule.

(E) He owns a dog. He is a farmer. No farmer who owns a dog beats it. So, he does not beat it.

$1\ D1\ O21,\ F2,\ (1\ F1\ 1\ D1\ O21\ B21) \vdash (B21)$

1.	$1\ D1\ O21$	[ass.]
2.	$F2$	[ass.]
3.	$(1\ F1\ 1\ D1\ O21\ B21)$	[ass.]
4.	$B21$	[ass.]
5.	$1\ D1\ O31\ B31$	[+, 1, 4]
6.	$F3\ 1\ D1\ O41\ B41$	[+, 2, 5]
7.	$1\ F1\ 1\ D1\ O21\ B21$	[I, 6(3)]
8.	$()$	[X, 7, 3]
9.	$(B21)$	[R]

The proof proceeds by showing that the assumption $B21$ leads to a contradiction. The reason is that this assumption licenses the string on line 6, which relates to the referent declared on line 1, and the declared existence of such a referent is precisely what is excluded on line 3.[19] The introduction rule here employs a dummy string $s = F0\ 1\ D1\ O01\ B01$, with 0 instantiated by 3 on line 6, and declared new on line 7.[20]

By now we have four deduction rules at our disposal, which are conceptually simple, I think, and justified, and bothersome only because of the persistently required updates. They are jointly strong enough, too, to derive all intuitively valid inferences, as we will show in §4. As a leg up, let us consider how universally quantified inferences, the hallmark of classical predicate logics, come about in the system of *PLI*.

Universally quantified statements in ordinary predicate logic show up as a particular sort of excluded declarations in *PLI*. A string $(1\ (s))$ excludes there to be something that is excluded to be s. Thus, on pain of contradiction, *any thing* is s, any arbitrary discourse referent, j, or any other. Indeed this is something we can infer, as the following, derived,

[18] According to the above rule of introduction, we have unconditional existential generalization, and this licenses and assumes the common requirement of a nonempty domain. One may, however, wonder why this should be a *logical* assumption. In a chapter on *Mathematics and Logic*, Bertrand Russell affirmed that "There does not even seem any logical necessity why there should be even one individual—why, in fact, there should be any world at all." and he added in a footnote that "The primitive propositions in *Principia Mathematica* are such as to allow the inference that at least one individual exists. But I now view this as a defect in logical purity." (Russell, 1919, pp. 203) It turns out that we can save our logic from such an, unmotivated, post-Aristotelean assumption of existential import, by simply demanding the introduction rule to apply to occurrences of strings $[j/0]s$ only if j is resolved there and then. This would, semantically speaking, allow for the possibility of an empty domain, as required, and this would be unproblematic because we don't need (an interpretation of) individual constants or variables.

[19] Notice that the application of the exclusion rule on line 8 uses its arguments in a converse order. This is fine, as is established in theorem 4.3 below. Theorem 4.1 details the required updates of the addition rule.

[20] Line 6 hosts the string $[3/0]F0\ 1\ D1\ O01\ B01$ so line 7 reads ${}^{6\leq}1\ [1/0]^{+1}F0\ 1\ D1\ O01\ B01$, which is ${}^{+1}1\ [1/0]F0\ 1\ D1\ O01\ B01$, i.e., ${}^{+1}1\ F1\ 1\ D1\ O21\ B21$, which is $1\ F1\ 1\ D1\ O21\ B21$.

rule of universal instantiation shows. In converse, if we can infer, about a totally arbitrary, hypothetical, discourse referent, that it is s, then this must imply that everything is s. The generalization summarizes what can be inferred about an arbitrary individual, upon the assumed declaration of one. (In the following theorem, $[^{j/1}_{-1}]s$ is short for $[^{+j-1}_{0,1}]s$, which is s with all unresolved indices reduced by 1, except the first, which is related to a discourse referent which is introduced j declarations before, for an arbitrary index j.)

THEOREM 3.4 (Universal Exclusion). *The following two deduction rules can be derived from the four deduction rules above:*

Universal Instantiation (UI)

$$\begin{array}{ll} & \vdots \\ i. & (1\ (s)) \\ & \vdots \\ n. & {}^{i<}[^{j/1}_{-1}]s\ [UI,\ i] \end{array}$$

Universal Generalization (UG)

$$\begin{array}{l|ll} & \vdots & \\ i. & 1 & [ass.] \\ & \vdots & \\ n\text{-}1. & {}^{i<}s & \\ \hline n. & (1\ (s)) & [UG] \end{array}$$

Proof (Sketch). The instantiated string s' on line n follows because its exclusion (s') licenses a declared instantiation 1 (s'') which is excluded on line 1.[21] A universal generalization is legitimate because it can be obtained by retracting 1 from the double exclusion ${}^{i<}((s))$ that we get from line n-1. □

The instantiation rule is essentially standard.[22] As for the rule of universal generalization, Kees Vermeulen has already observed before that a universal claim can be validated, in a dynamic framework, in the smooth and intuitive way that we find above (Vermeulen, 1993). A statement holds of all individuals if it can be shown to hold of an arbitrary individual, atomically declared. The (derived) rule of universal generalization here is as neat as the familiar explanations in the textbooks would have wanted it, if it were not for the fact that these have to explain the required use of 'new' or 'fresh' variables. Obviously no such worries trouble us here.

§4. Soundness and completeness. The derivation schemes and schematic rules valid in *PSL* remain valid in *PLI*, of course, although their formulation may require an update. In the following observation the (sub-)strings ${}^{+m}s$, ${}^{+m'}r$, and ${}^{+m''}t$ state in the conclusion exactly what s, r, and t stated in the premises.

THEOREM 4.1 (*PLI*-Subtraction and Addition).

$$\begin{array}{rl} rst \vdash {}^{+n_{st}}s & (-) \\ rt, s \vdash {}^{+n_{rts}}r\,{}^{+n_{sr}}s\,{}^{+n_{tsrs}}t & (+) \end{array}$$

[21] More precisely, let r be the string ${}^{-1}[0/1]s$, in which the first index is mapped to the rigid dummy zero, and all other unresolved indices are reduced by 1. The conclusion, neglecting the update by ${}^{i<}$, then equals $[j/0]r$. Instantiation of its exclusion $[j/0](r)$ yields 1 $[1/0]^{+1}(r)$, which is 1 $[1/0]^{+1}({}^{-1}[0/1]s)$ which is 1 (s). The (update of the) latter is excluded on line i.

[22] To add up on previous footnotes, if the introduction rule is restricted to declared discourse referents, then of course the instantiation rule is likewise restricted as well. This is to say, that if we give up the nonempty domain requirement, then generic strings can only be instantiated by indices which relate to explicitly, possibly hypothetically, declared discourse referents.

Proof (Sketch). The derivations are the same as those for propositional subtraction and addition given in the appendix, but for the fact that we employ suitable updates of r, s, and t, respectively. For instance, in the proof of $rst \vdash s$ we use $(^{+n_{st}}s)$ on line 2, $^{n_{rst}}r$ on line 3, etc. □

The derived rule of addition, however, only makes up for a *conservative* form of conjunction. As can be easily checked, the rule yields $1\ F1,\ G1 \vdash 1\ F1\ G2$, which is desirable and correct, but it does not yield $1\ F1,\ G1 \vdash 1\ F1\ G1$, which would be an instance of an, equally desirable, form of *dynamic* conjunction. A dynamic form of conjunction actually is derivable as well. (In the statement of the following theorem the update $^{+n_{rs}}$ applies to the whole string rs, contrary to the practice assumed above.)

THEOREM 4.2 (Dynamic Conjunction).

$$r, s \vdash {}^{+n_{rs}}rs \text{ (DC)}$$

Proof. In the appendix a more general rule of dynamic conjunction is derived, of which the present theorem constitutes an instance. □

Dynamic conjunction is intimately related to dynamic exclusion. The following theorem can be conceived of as displaying a form of dynamic implication, familiar from systems of dynamic semantics.

THEOREM 4.3 (Converse Exclusion).

$$(rs), r \vdash (s) \text{ (CX)}$$

Proof. In the appendix a more general rule of converse exclusion is derived, of which the present theorem constitutes a schematic instance. □

Converse exclusion may be labeled 'dynamic', because indices in the conclusion (s) may relate back to declarations in the premise r. (Notice that dynamic exclusion does not strictly pattern with the exclusion rule as defined above, because the order of premises is reversed.)

Dynamic conjunction and converse exclusion, while on the face of it entirely classical, establish the dynamics of interpretation and inference, but 'below the surface', that is. The same can be said of the, virtually classical, laws of contraposition and conditionalization. These are also unconditionally valid, and preserve internal correlations, if kept in the right sequential order.

LEMMA 4.4 (Contraposition).

$$rs \vdash t \textit{ iff } rs(t) \vdash () \textit{ iff } r \vdash (s(t))$$

Proof. See the appendix. □

The present observations may suffice to establish that the four *PLI* deduction rules provide for an intuitively correct characterization of the notion of a valid dynamic inference. In order to show that they also do so formally, we need to present a formal definition of dynamic semantic entailment first. I use the one presented, for one premise entailments only, in (Dekker, 2012). The definition, as defined for *PLA*-formulas there, is adapted here to the format of *PLI*-strings.

DEFINITION 4.5 (*PLI*-Entailment). *String r entails string s, $r \models s$, iff, for all models M, for all $c \in D^{n_r}$, if $M, ce \models r$ then there is a $a \in D^{n_s}$: $M, ace \models s$.*

Employing a picturesque metaphor, this says that r entails s iff every case or situation c that instantiates r will also, always, support s. Soundness of the deduction rules can be

easily checked now. An illuminating way to establish completeness builds on a sound and complete deduction system for standard first order logic. Completeness of the latter can be seen to imply completeness of the present rules for the classical section of *PLI*. Validity in the nonclassical extension of *PLI* can subsequently be seen to be derivable from its classical core. So let us first expand upon the classical fragment of *PLI*.
The formulas of ordinary predicate logic can be translated into what I will call nude *PLI* normal forms (*NNF*). These are the *PLI*-strings s with no nonexcluded declarations (so $n_s = 0$), and such that any excluded substring is in 'Kamp canonical form' (cf. below). Such an excluded substring consists of an initial series of n, possibly 0, declarations and a substring t in nude normal form itself. The excluded proposition will be written as $({}_n t)$. The translation is stated for a simple version of first order predicate logic, without names, identity, and function symbols, and with a relatively minimal set of connectives.[23]

The translation of a *PL*-formula ϕ into a *PLI*-string s starts with a substitution of all free variables x_i in ϕ by indices i, and a reverse translation starts with a substitution of all unresolved indices i in s by x_i. We may write $[i/x_i]_{i\in N}\phi$ and $[^{x_i}_{l}/i]_{i\in N}s$ for these substitutions.[24] For the resulting *hybrids*, *PL*-formulas ϕ' containing (unresolved) indices, and *PLI*-strings s' containing (free) variables, I define the following recursive translations $\lceil\phi'\rceil$ into *PLI*, and $\lfloor s'\rfloor$ back into *PL*. (In the following definition, υ is short for any string of terms, variables or indices.)

DEFINITION 4.6 (Translation between *PL* and PLI^{nnf}).

$$\begin{array}{llll}
\lceil\top\rceil & = (()) & \lfloor\ \rfloor & = \top \\
\lceil R\upsilon\rceil & = R\upsilon & \lfloor R\upsilon\rfloor & = R\upsilon \\
\lceil\neg\phi\rceil & = (\lceil\phi\rceil) & \lfloor({}_0 s)\rfloor & = \neg\lfloor s\rfloor \\
\lceil(\phi\wedge\psi)\rceil & = \lceil\phi\rceil\,\lceil\psi\rceil & \lfloor rs\rfloor & = (\lfloor r\rfloor\wedge\lfloor s\rfloor)^a \\
\lceil\forall x\phi\rceil & = (1\,[1/x]^{+1}(\lceil\phi\rceil)) & \lfloor({}_n\, s)\rfloor & = \forall x\lfloor[^{x/1}_{-1}]({}_{n-1}\, s)\rfloor^b
\end{array}$$

[a] This clause makes the translation indeterminate because a string can be cut up in various ways; the indeterminacy is model- and proof-theoretically as spurious, though, as conjunction is associative.

[b] Of course x must be 'new'.

The translations are meaning preserving. For any assignment g, let us use e_g for a sequence of individuals $g(x_1)g(x_2)\ldots$ of sufficient length, and for any sequence of individuals e, let g_e be any variable assignment such that $g(x_i) = e_i$.

THEOREM 4.7 (PLI^{nnf} has a Classical Semantics).

$$M, e \models s \text{ iff } M, g_e \models \lfloor s'\rfloor, \text{ if } s \text{ is in nude normal form}$$
$$M, g \models \phi \text{ iff } M, e_g \models \lceil\phi'\rceil$$

Proof (Sketch). By induction on the construction of s and ϕ, respectively. □

[23] To be precise, I will assume a version of predicate logic (*PL*) in what follows with a top element ($\top$), atomic formulas that consist of relational predicates R in combination with the right number of variables $(x, \ldots)$, negation $(\neg)$, conjunction $(\wedge)$, and universal quantification $(\forall x)$. I also assume we have an enumeration of the variables.

[24] The definitions themselves are omitted here; it suffices to observe that the x_i in the substitutions are fixed, and that l, originally set to 0, again functions as a threshold, for the indices $l+i$ to be replaced by x_i.

The *NNF PLI*-fragment, thus, has a classical semantics, no more nor less. It is also easily seen that $s \vdash \lceil\lfloor s\rfloor\rceil \vdash s$, and that $\phi \vdash \lfloor\lceil\phi\rceil\rfloor \vdash \phi$. This fact next enables us to establish that proofs can be translated salva validate.

THEOREM 4.8 (PLI^{nnf} has a Classical Logic).

$$\phi \vdash_{pl} \psi \textit{ iff } \lceil\phi\rceil \vdash_{pli} \lceil\psi\rceil$$

Proof (Outline). For the *if*-part, it suffices to observe that *PL*-formulas translate into *NNF*s, that $\vdash_{pli}$ is sound, that the semantics of the *NNF PLI*-fragment is classical, and that $\vdash_{pl}$ is complete. Hence, in brief, $\vdash_{pli}$ implies $\models_{pli}$, which implies $\models_{pl}$, which implies $\vdash_{pl}$. For the *only if*-part, we can show that we can translate every derivation in *PL* into a derivation in *PLI*, applying *PLI*-correlates of the rules employed in *PL* to the *PLI*-translations of the *PL*-formulas involved. □

The upshot of these observations is that the *PLI*-deduction system is complete for its *NNF* fragment. For if an *NNF*-string is semantically valid, its classical translation is semantically valid; by completeness for classical logic the translation is classically derivable, and by the latter observation it, hence, is *PLI*-derivable.
Completeness of the set of *PLI* deduction rules is established for the full system, including the non-*NNF* strings, by showing that the general notion of validity can be reduced to that of the validity of so-called Kamp canonical forms, and that the validity of the latter can be derived from classical validity, i.e., from the validity of nude normal forms.

The Kamp canonical form of a string consists of a series of atomic declarations, and a subsequent string of conditions which are either atomic, or exclusions of Kamp canonical forms. These canonical forms replicate the *discourse representation structures* from *DRT*, which consist of a set or string of discourse markers, paired with a set or string of conditions. The following definition specifies a rewriting procedure that turns a *PLI*-string in its Kamp canonical form. (And, as indicated above, the nude normal form is just a Kamp canonical form with the initial series of declarations stripped of.)

DEFINITION 4.9 (Canonical Forms).

- *The* Kamp canonical form s^* *of* s *is* z *iff* ${}^i s \rightsquigarrow z^i$, *where:*

$n\ r\ {}^i\ 1\ t \rightsquigarrow n\ 1\ {}^{+1}r\ {}^i\ t;$	*[On the first line* n *abbreviates an*
$r\ {}^i\ Rv\ t \rightsquigarrow r\ Rv\ {}^i\ t;$	*initial series of* n *declarations and*
$r\ {}^i\ (s)\ t \rightsquigarrow r\ (s^*)\ {}^i\ t.$	r *is declaration free.]*

- *The* nude normal form s^- *of* s *is* z *iff* $s^* = 1^{n_s}\ z$.

Rewriting s to its Kamp canonical form proceeds by prefixing it with an initialization device i which moves through the string and puts all occurring, nonexcluded, declarations at the front of it. It does, recursively, the same for excluded substrings.[25] A Kamp canonical form preserves the meaning of its original.

THEOREM 4.10 (Normalization).

$$M, e \models s \textit{ iff } M, e \models s^*$$
$$s \vdash t \textit{ iff } s^* \vdash t$$

[25] For instance, rewriting the sequence in (C) above, we find: ${}^i 1\ F1\ 1\ D1\ O21\ 1\ C1\ O31 \rightsquigarrow 1\ F1\ {}^i 1\ D1\ O21\ 1\ C1\ O31 \rightsquigarrow 11\ F2\ {}^i\ D1\ O21\ 1\ C1\ O31 \rightsquigarrow 11\ F2\ D1\ O21\ {}^i 1\ C1\ O31 \rightsquigarrow 111\ F3\ D2\ O32\ {}^i\ C1\ O31 \rightsquigarrow 111\ F3\ D2\ O32\ C1\ O31^i$.

Proof (Outline). Semantic (satisfaction) equivalence is proved straightforwardly by induction on the normalization of s, from left to right and bottom up. Logical (derivability) equivalence is proved in the appendix. □

We can establish a further, final, reduction. The logical role of a Kamp canonical form as a premise is just like that of a nude normal form, that is, it behaves classically.

THEOREM 4.11 (Initialization).

$$s^* \models t \textit{ iff } s^- \models t$$
$$s^* \vdash t \textit{ iff } s^- \models t$$

Proof (Outline). These two propositions generalize two elementary observations, (i) that $1s \models t$ iff $s \models t$, and (ii) that $1s \vdash t$ iff $s \vdash t$. The first one of these is obvious; the second is proved in the appendix. □

With the tools and theorems developed, we now have enough of the machinery in place to present the main result of this paper and to outline the proof.

THEOREM 4.12 (Completeness for *PLI*). *With* $\vdash$ *drawing from the current rules of exclusion, retraction, contradiction, and introduction we find that:*

$$s \models t \textit{ iff } s \vdash t$$

Proof (Outline).

$s \models t$ iff (contraposition)
$s(t) \models ()$ iff (normalization)
$s(t)^* \models ()$ iff (initialization)
$s(t)^- \models ()$ iff (classical)
$s(t)^- \vdash ()$ iff (initialization)
$s(t)^* \vdash ()$ iff (normalization)
$s(t) \vdash ()$ iff (contraposition)
$s \vdash t$ □

Let me sum up and stage the results so far. In the preceding sections, I have presented a first order deduction system building upon an indexical notion of discourse reference. The system has been minimal, in order to facilitate the exposition and proofs, and in order to highlight what, I believe, are the essential ingredients, *logically* speaking. Just four deduction rules, intuitive ones, I think, allow us to do first order logic in a natural, classical as well as dynamic, fashion.

It may have become clear that the *use* of such a minimal language, so *practically* speaking, is somewhat awkward. Notably, precisely due to its indexical architecture, it is not quite easy to simply *see*, in a string, which terms target one and the same discourse referent. It is for this reason indeed very useful to mark declared discourse referents with (unique) labels, or variables. So has been done in the past in classical predicate logic and in, e.g., *DRT* and *DPL*, and so I will also advocate and do below. The convenient possibility of (optionally) labelling discourse referents enables one to frame *DRT* analyses in terms of *PLI* strings directly and perspicuously.

We can then stage the formal result as follows. In standard *DRT* (Kamp, 1981), as well as sophisticated extensions of it[26], a practically neat distinction is made between the

[26] Like that of, e.g., Presuppositional DRT (van der Sandt, 1992), Underspecified DRT (Reyle, 1993), and Segmented DRT (Asher & Lascarides, 2003).

representations that eventually result from processing a stretch of discourse and preliminary structures that have to be resolved, preferably by discourse referential means in the first place. The first are standardly interpreted in the usual fashion, in terms of satisfaction conditions, but the latter often figure at a level of construction that precedes semantic interpretation. Compositional interpretations of the *DRT* method of establishing discourse reference have been given as well.[27] Such interpretations usually assume discourse referents to be labeled, and adopt some dynamic semantic notion of composing meanings. In particular for this dynamic composition of meanings we have not so far, in turn, seen a tailor-made, sound and complete, proof system.[28] Now I claim that the *PLI* inference system does after all present the logic underlying the dynamic semantics of a, suitably representational, *DRT*-style theory of interpretation. One can think of *PLI* as formulating the logic of systems of dynamic semantics that has remained hidden under the labels in terms of which their semantics has usually been stated.

So what do we get out of this? Somewhat belittling *PLI* one can say it justifies one to keep on doing *DRT* (or, e.g., *DPL*, for that matter) and be happy there is a proof theory for it. I think, however, it adds something substantial. As I will argue in the final section, this does not come from the logic provided above, nor from the absence of a labeling practice (which we after all deem useful), but from the indexical take on discourse reference.

§5. The indexical turn. In the *PLI*-system discourse referents show up by means of declarations, occurrences of the atomic proposition '1', and its terms are interpreted exclusively indexically. The interpretation of a *PLI*-term entirely resides in the way in which an occurrence of the term relates to material in the context of *its occurrence*. In this section I will briefly reflect upon some of the linguistic benefits of this indexical outlook, a logico-philosophical point that can be made out of it, and a perspective it may provide upon language in action.[29] The aims of this section being rather programmatic and speculative I may have to warn the reader that its findings will be somewhat sketchy and informal too.

The *PLI*-notion of inference is contextual in the obvious sense that premises and conclusions are taken to hold in the contexts of developing proofs and of pending assumptions. Semantically, a stretch of discourse is taken to be evaluated, and made sense of more in general, in such local contexts of interpretation, and this outlook on meaning is one that it shares with classical *DRT* and dynamic semantics. However, precisely by its indexical

[27] Almost immediately, in the elementary system of File Change Semantics (FCS, Heim 1982), and subsequently in, e.g., dynamic predicate logic (*DPL*, Staudacher 1987; Groenendijk & Stokhof 1991), or in typed extensions like that of dynamic Montague Grammar (*DMG*, Groenendijk & Stokhof 1990), compositional *DRT* (Muskens 1996) or, more recently, continuation semantics (Barker 2002; de Groote 2006).

[28] Van Eijck's system of incremental dynamic interpretation 2001 does have a proof-theory for a kind of in-text-entailment, one that computes inferences on strings with their (con-)textual beginnings indicated. However, it has no independent notion, model-theoretically nor proof-theoretically, of dynamic composition.

[29] A, likewise inverted, indexical outlook has been advanced, in different, but affiliated, areas of application, by various authors such as Arthur Prior, Hans Kamp, John Perry, David Kaplan and David Lewis (Prior, 1967; Kamp, 1971; Perry, 1979; Kaplan, 1979; Lewis, 1979), possibly including the early Wittgenstein (Wittgenstein, 1922, Satz 5.6v). For a proper conception of our metaphysics of time, modality, knowledge, and even logic, it can be argued, one should assume (adopt, inhabit) a point or situation *within* some temporal, modal, epistemic or logical space. The notion of a (discourse) referent may be properly understood once one, likewise, adopts a standpoint *within* a discourse- or proof-theoretical space.

orientation, *PLI* allows one to explicitly and efficiently state precisely *where* such material has to be made sense of and relative to *which* context: the locus is that of its occurrence and the context is that of this occurrence, indexically identified. An index i associates with the i-th discourse referent before its occurrence, so that—somewhat artificially, but significantly—its contextual nature is encoded in its meaning.

The *PLI*-architecture does not hereby force us to understand such terms or other expressions only by means of their identification with a previously declared discourse referent. Once we recognize that expressions have to be made sense of in their context of use, we witness, of course, a whole repertoire of ways in which such can be done. An anaphoric term, for sure, can be *identified* with a previously introduced discourse referent. But, as is well-known in theories of discourse, a reported event can be related to a previously reported event as, for instance, *succeeding it*, *explaining it*, *elaborating on it*, or what not. The time of a described event, a topic under discussion, they can be related to a current discourse time, or a current discourse topic, in various ways. Now, no matter what relation should be established in order to make sense of a discourse, the point is that the requirement that such a relation obtains is one that should be satisfied locally. An event, for instance, that is reported upon can be required to be related to an event previously reported upon, and the relation must hold then between the event presented by that very specific report and the event presented by the previous one. The reference points for the relata thus are the occurrences in the discourse mentioning them, and the most direct and effective way of identifying these occurrences is in an indexical manner.

The extensive and formally rigorous work of Maria Bittner may serve to substantiate this point. (See, e.g., Bittner 2001, 2014b.) Bittner uses tools of *PLA* (essentially *PLI*) to characterize discourse structural properties of a variety of typologically different languages. Her findings and typological characterizations are formulated in an update system with a centered notion of (nominal, temporal, modal and eventive) discourse reference. The formal language that she employs, which can be considered a refinement and adjustment of the *PLI*-language, is indexical at the heart. Bittner's results, of course, do not by themselves provide us with a robust, linguistic, or methodological, argument that an indexical account of the data is the only one possible. They do, however, strongly suggest that it is not just the context dependent nature of interpretation, nor a dynamic conception of meaning, but the indexical anchoring of meaning ('centering') that allows for the compelling formulation of her findings.

For the same reason indexical representation also appear to be very suitable for practical applications of *DRT*. Let me emphasize, to begin with, that *PLI*-deduction is by no means meant to model the type of processes involved in the interpretation of actual discourse. The language of *PLI* may, like the language of discourse representation structures, provide for a medium in which to frame the kind of representations that the users of a language can be taken to construct in response to occurring discourse—representations on the basis of which, and in terms of which, they can be taken to arrive at their actual interpretations. The interpretive processes that are actually employed may be abductive and nonmonotonic, they may draw from lots of contextual clues and associative assumptions, and the details of these do not concern us here. Nevertheless, the language of *PLI*, like that of *DRT*, can be taken to render the (idealized) input and output of such processes, and for that purpose indexical representations like that of *PLI* may be more appropriate than has appeared from the logical investigations we have been concerned with so far.

This point may be appreciated best by means of an illustration. The following exposition purports to frame, using the language of *PLI*, the treatment of a minimal pair of examples extensively detailed in (Kamp et al., 2011, pp. 198, 201–6). It serves the expository purposes if I display the *PLI*-strings below in the kind of user-friendly 'boxes', familiar from *DRT*. In addition, in order to improve working conditions, it will also turn out to be useful to start 'labeling' discourse referents with 'discourse markers'. It may also become clear, thus, why this labelling practice can be troublesome too, logically speaking.

The first example is Kamp et al.'s (85b).

(85b) Josef turned around. The man pulled his gun from its holster.

This snapshot of a text apparently presupposes a context in which Josef is familiar, and also some man is, who is presumably different from Josef. I assume these are staged here by means of two declarations, in an already given, 'surrounding', context, and which are conveniently labeled j and m in the representation below.[30] The preliminary representations for the two sentences are presented by two canonical forms, displayed as half-open discourse representation structures, which are appended to the schematic context that comprises j and m.

$\ldots j \ldots m \ldots$

$\vdots$

$1\ e_1$

$REL[1,$ *"josef"*$]$
$e_1 : TA[1]$
$e_1 < t_0$
$REL[e_1]$

$1\ e_1$

$REL[1,$ *"the man"*$]$
$e_1 : PG[1]$
$e_1 < t_0$
$REL[e_1]$

Both canonical forms (actually: strings) (re-)introduce a discourse referent and declare an eventuality.[31] The condition $REL[1,$ *"josef"*$]$ can be conceived of as an 'instruction' to relate this discourse referent, the referent of this use of the name "Josef", to the current context. Likewise for "the man". (I here follow the practice from Kamp et al.) In a subsequent stage these conditions are assumed to be resolved by the equation of the two terms with, we assume, j and m, respectively.[32] The eventualities are described as events in which the declared subject turns around $TA[1]$ and in which the subject pulls a gun $PG[1]$, respectively. Both sentences are in past form, whence the conditions $e_1 < t_0$, stating that the declared events, or rather their temporal locations, lie before the present time. (More on this 'present' below.)

Kamp et al. focus on what is rendered here, twice, as the condition $REL[e_1]$. This issues the arguably pragmatic condition that the currently declared eventuality be suitably related to the current discourse.[33] Due to lack of further details, the relation of the event of turning around is left unspecified here, as it is in Kamp et al. But a proper way of relating the event of pulling a gun, Kamp et al. argue, is by assuming it to follow the earlier mentioned event, that of Josef turning around. The result of such interpretation may therefore yield the

30 Of course, actually, the occurrences of "j" and "m" are just two occurrences of "1".

31 What is written as e_1 here is actually a declaration "1" of something of eventuality type.

32 More precisely, with the declarations which were labeled with 'j' and 'm', respectively.

33 Semantically, we can understand *REL* as a relational predicate, actually a polymorphic one, relating the current eventuality discourse referent to possibly any previously introduced one. Practically, the predicate can be understood as an instruction to specify or resolve it.

representation ('reading') below. Next to it, I have displayed the same result in a structure in which the declared events are labeled.

$$\begin{array}{|l} \hline \ldots j \ldots m \ldots \\ \hline \vdots \quad \begin{array}{|l} \hline 1\ e_1 \\ \hline j = 1 \\ e_1 : TA[j] \\ e_1 < t_0 \\ REL[e_1] \end{array} \begin{array}{|l} \hline 1\ e_1 \\ \hline m = 1 \\ e_1 : PG[m] \\ e_1 < t_0 \\ e_2 < e_1 \end{array} \end{array} \qquad \begin{array}{|l} \hline \ldots j \ldots m \ldots \\ \hline \vdots \quad \begin{array}{|l} \hline 1\ e^x \\ \hline j = 1 \\ e^x : TA[j] \\ e^x < t_0 \\ REL[e^x] \end{array} \begin{array}{|l} \hline 1\ e^y \\ \hline m = 1 \\ e^y : PG[m] \\ e^y < t_0 \\ e^x < e^y \end{array} \end{array}$$

One may observe here that it is not anything about the two events themselves, that makes one infer that the one (directly) precedes the other. For, to begin with, the snapshot of a discourse does not provide any "de re" access to any described event. So, in a very specific sense we simply don't know, and don't need to know, which events we are talking about. Moreover, nothing in the qualitative description of the individual events would invite to the conclusion that the events are so related. Rather, it is the fact that the two are consecutively mentioned, described, or referred to, in the current discourse, that makes one suppose that they are so related, whichever the two actually happen to be. It is thus a relation which is, defeasibly, concluded to hold between the previously mentioned and the currently introduced discourse referent, and this relation is issued essentially indexically.

Example (85b) is contrasted with the slightly different (85a).

(85a) Josef turned around. The man was pulling his gun from its holster.

The second sentence in example (85a) is in the past progressive, whereas it was in the simple past in example (85b). Arguably, a verb in the progressive describes a 'stative' eventuality. The discussion of various similar examples in the literature suggests that the integration of state descriptions into previous discourse tend to be different from that of event descriptions. Thus, and pending further considerations, and without any further background, and without any clues about the context of a use of example (85a), one can resolve the condition that the described eventuality (a state) is related to previous discourse by rendering the state as one including the previously described event. This, then may serve to 'resolve' the condition $REL[e_1]$ here. This interpretation ('reading') of (85a) is given by the following structure. To enhance readability I have again supplied a representation with labels associated.

$$\begin{array}{|l} \hline \ldots j \ldots m \ldots \\ \hline \vdots \quad \begin{array}{|l} \hline 1\ e_1 \\ \hline j = 1 \\ e_1 : TA[j] \\ e_1 < t_0 \\ REL[e_1] \end{array} \begin{array}{|l} \hline 1\ e_1 \\ \hline m = 1 \\ e_1 : PPG[m] \\ e_1 < t_0 \\ e_2 \subseteq e_1 \end{array} \end{array} \qquad \begin{array}{|l} \hline \ldots j \ldots m \ldots \\ \hline \vdots \quad \begin{array}{|l} \hline 1\ e^x \\ \hline j = 1 \\ e^x : TA[j] \\ e^x < t_0 \\ REL[e^x] \end{array} \begin{array}{|l} \hline 1\ e^y \\ \hline m = 1 \\ e^y : PPG[m] \\ e^y < t_0 \\ e^x \subseteq e^y \end{array} \end{array}$$

The two examples serve to illustrate two points. The first, actually relatively minor, point is that they show that a practice of labeling declarations, and subsequent cross-references, not only enhances readability, but actually produces structures more familiar from the classical theory of discourse representation and dynamic semantics. The practice is not only convenient, or even recommendable, but as one can see also completely harmless. A subsidiary subtle point, but not a trivial point, is that a term labeled j, or x or e^x, can be taken to be suitably related to a discourse referent declared under the same label, but only provided that its underlying index gets us there. The relation between a newly declared discourse referent and a previous one is not issued by their being labeled by the same variable, but by their actual co-occurrence in a discourse under construction. Observe that if several discourse referents happen to go by one and the same label, a mechanism of discourse reference mediated by labels would get distorted.[34]

The discussion of the two examples may, more importantly, serve to show the benefits of employing indexical expressions in a *DRT*-style model of interpretation. Like I said, any real interpretation of any actually occurring discourse will be established by processes which are not *PLI*-deductions; they can be and actually are obtained otherwise. However, the idealized inputs and outputs of the interpretative processes are most efficiently given an indexical formulation, for as far as they can be cast in any representational format at all. As one can see we can do with just a single indexical template for the initial, unresolved, representation of the contribution of a verbal segment. Also, we can work out any specification or resolution of it precisely in the context in which such a template is put to use. Crucially, the resolution is not induced by a property of the topic independent of the discourse about them. Rather, it is the property of, e.g., the two events of being successively referred to, or, better, the property of the latter event of being referred to just after the former event has been mentioned in the current discourse.[35] In case (85b), for instance, it is not the fact that the currently described eventuality is stative, and that the earlier mentioned eventuality is eventive, that leads to the conclusion that the one mentioned earlier is temporally included in the currently described one. Like I said it is the fact that the two eventualities are consecutively mentioned and described thus, in the current discourse, at the current location in the discourse.[36]

Of course we can make speculations and generalizations about the ways in which two events can be supposed to be related when they are consecutively referred to in a discourse. A branch of the literature indeed addresses this theoretical issue. Here I want to emphasize a more practical point. In the actual interpretation of a discourse, one is supposed to settle on a specific interpretation of the relation involving things mentioned on specific occasions in the discourse. (Such actual, ‘one time’, interpretations are properly called ‘readings’, they are literally a way of making sense of actual discourse.) Now, no matter how one achieves that, and no matter which generalizations or constraints are called upon, it seems that this is most efficiently characterized in the indexical manner advocated here. Surely it

34 Of course, it can be a happy contingent fact that such unfortunate labeling does simply not occur. For such labelings are as a matter of decent practice, not by logical rule, excluded by those who practically work within the framework of *DRT*. And indeed, *provided that* one can make the practical, not logical, assumption that variables are not used to introduce a discourse referent twice, a transparent system of natural deduction can be easily given for a language that uses labels, or variables, as terms. However, see, e.g., (Veltman, 2000) for some of the complications that arise once such a, truly nonlogical, assumption can no longer be taken for granted.

35 The qualification “in the current discourse” is absolutely essential, for, obviously, the *very same discourse* may, on different occasions, yield different interpretations, or ‘readings’. This fact is easily accounted for upon the indexical understanding, but hard to make sense of under some generic, context independent, account of discourse relations.

36 More strikingly, perhaps, it is certainly not the property of events of being labeled e^x and e^y, that induces the conclusion, unless it be the, totally irrelevant, property of being labeled so at the *current* locations in the *current* discourse.

makes sense to say, in general, that the successive mentioning of an eventive and a stative eventuality yields, by default, a reading on which one is included in the other. But in order to settle at actual interpretations (again: readings), we have to state specifically *when* or *where* such 'conclusions' are actually drawn, and *when* or *where* they are not. We have to be able to indicate the specific occasions in which an event and a state are mentioned and in which a constraint does or did apply, and also those in which the constraint does or did not. In order to do so we can of course *point* at representations, and say that *this* occurrence (pointing in the representation) has been related to *that* occurrence (again pointing in the representation) in such-and-such way. This *outsider* pointing, however, would be truly indexical itself then, and it would be more efficient, it seems, to directly encode this *within* the representations themselves, preferably in the indexical manner advocated here.[37]

Kamp et al. refer to, e.g., "pragmatic" factors involved in "the specification of the relations in which [reference time and perspective time, PD] stand to location time and utterance time", and note that "[i]t appears that if we want to make substantial further progress on these problems we need a framework in which these other factors can be treated in a systematic way. As it stands DRT does not provide this framework." (Kamp et al., 2011, p. 221) I hope that the discussion of the two examples here suffices to at least indicate that the kind of indexical representation that we find in *PLI* establishes the kind of frames or framework called for.

The indexical analysis of terms proposed in this paper has further, more conceptual, potential. Proof-theoretically, discourse referents are identified counting backwards in a proof or discourse. Theoretically, one might also count forwards. Of course, it does not make much sense to refer to discourse referents that have not yet been introduced in a proof or discourse, but it does make obvious sense to relate to material that is, logically speaking, absent, in a figurative sense of *what belongs there*. Negative indices, $-1, -2, \ldots,$ can be conceived of as indicating the arguments of *functions*, arguments that are expected or required to be supplied, in the first place, in the second place, etc. As a matter of fact this is how de Bruijn uses indices to indicate the positions that are abstracted over in a variable-free λ-calculus (de Bruijn, 1972). I will not, here, compare and combine the two calculi and leave this exercise for another occasion. I do take it, however, as a leg up for a more minimal extension of *PLI* that it brings to mind, and with which I want to conclude this paper.

As explained, an index i refers to the discourse referent introduced i declarations before the index's occurrence, so obviously i's own occurrence, whenever it occurs, figures as its first and only secure point of reference. We can therefore think of a zero index 0 as referring to just that occurrence, that is, as a self-referential device. Its occurrence, or more liberally that of the string in which it occurs, thus may count as its denotation. Let me use *PL0* for the system of *PLI* that includes such a self-referential device.[38] Once we have such a

[37] We would arrive at the same conclusion if we were to use, e.g., variables as labels. The reason is essentially that it does not really make sense to say that two variables are coreferential or that a quantifier $\exists x$ binds a variable x. For it is *occurrences* of quantifiers that bind or do not bind *occurrences* of variables.

[38] Semantically, this device requires us to conceive of occurring strings as entities in the domain. They must be able to figure as a parameter of interpretation, like, for instance, designated worlds in pointed Kripke models do. I will largely ignore the model-theoretic implications here, and focus on the proof-theoretical implications of using zero-indices. However, one model-theoretic implication I do have to mention. The use of self-referential indices implies a nonempty domain. This means, proof-theoretically, that we should be able to prove the existence of something.

device of self-reference we can introduce derivative devices to also refer to the "now" or the "here" (or the "thus") whenever or wherever (or however) they can be seen to occur. So we can as well have, e.g., temporal and possibly eventual indexical terms, t_0 and e_0, which then serve to indicate the time of their occurrence, or the utterance event in which they occur, again, whenever they occur.

Such derivative zero-indexical terms may fill a serious, and seriously neglected, gap that we find in existing dynamic frameworks of interpretation that set out to deal with properly contextual indexicals. Within the framework of *DRT*, quite commonly a distinguished reference marker n is assigned the job of indicating the temporal location of the discourse, one which remains fixed throughout, just like the indexical parameters that are assigned a fixed value in, e.g., Kaplan's theory of demonstratives (Kaplan, 1979; Kamp et al., 2011). Observe that, while apparently everybody seems to take it for granted that contexts may change, the indexical parameters simply do not, not in static nor in current dynamic semantic systems of interpretation.[39] Such a, counterintuitive, mishap is cured, instantaneously, by the use of zero-indices. The temporal zero-index t_0 does the required job, because it refers to the time of its own occurrence by definition. Moreover, since multiple occurrence of t_0 each refer to the temporal location of their own occurrence, it is possible (not necessary) that subsequent occurrences of t_0 relate to subsequent locations.[40] Self-referential terms can be expedient, too, in the kind of analysis of performatives initially proposed by John R. Searle and recently revived in the formal semantics literature. Conceiving of indices as referring expressions, zero-indices provide the referent that they refer to, and hence their use is self-satisfying. Performative utterances, or the (performative) use of (performative) sentences, are likewise characterized by an element of self-reference and self-satisfaction. Searle conceives of such utterances as "performances of the act named by the main verb (...) in the sentence", and he assumes them to be

---

And we actually can, with the rules supplied above. Suppose, in the spirit of Russell, that we do not stipulate the existence of anything. Formally, this requires the introduction rule [I] to only apply to instances $[j/0]s$ where j is resolved. Thus, we have existential generalization over resolved indices only, that is over discourse referents already declared. But now the use of the self-declaring index 0 enable us to proves the existence of something. Here is a formal, and minimal, (self-validating) proof that $\vdash 1$.

1.	()	[ass.]
2.	(())	[R]
3.		[C, 2]
4.	1	[I, 3(0)]

[Line 3 hosts the string '[0/0]␣', the empty string in which the self-declaring index 0 replaces an (absent) dummy 0. The conclusion on line 4 is formally spelled out then as the string "$^{+0}1\ [1/0]^{+1}$␣" which is "1 ".]

We have a proof (demonstration) that *something* exists, namely the demonstration (proof) itself. Somewhat in the spirit of Descartes the proof says, figuratively: "Demonstro ergo sum," and somewhat more acutely we can observe: "Demonstrat quod est demonstrandum."

39 Essentially the same observation has recently been made by (Bittner, 2014a), who proposes essentially the same solution.

40 It should not be a logical law that different occurrences of t_0 refer to distinct temporal locations. For this would depend on the *extension* of the temporal location of the occurrence of an expression and that appears to be a genuinely semantic issue, and, hence, a matter of interpretation, not logic. One may, even so, consider it necessary that reference to previously declared referents and times presupposes that the declarations have occurred before the references are made, so that the assumed order of time should at least be consistent with the order of occurring discourse. If so, then such should be made to follow from additional axioms or postulates or principles which, among quite a few other intuitive ones, are needed for a sensible, indexical, Priorian, logic of time.

"self-guaranteeing" (Searle, 1989, p. 539). Regine Eckardt has furthered an analysis of performative verbs along these lines in an event-based semantics. Eckardt conceives of performative verbs as verbs that, like other verbs, describe communicative events, and takes their performative use to consist in their being used self-referentially. Thus, an event can be characterized, descriptively, as, e.g., a promise, and if the event of characterizing itself is the event characterized, then it is truthful if and only the characterizing event is a promise. By, "hereby", referring to itself, it is used to say, on occasion, what it is used to do, on that occasion.[41]

Maša Močnik 2015 has recently pointed out that such distinct uses of performative verbs can be adequately accounted for in the framework of *DRT*, provided, that is, that we have a device of self-reference at our disposal. Using *PL0*, we have one. Two possible readings of a performative verb like "promise" can be seen to emerge from two *resolutions* ('readings') of its eventive argument, viz., two ways of relating a described event to events in a current discourse. In order to actually conclude this paper let me briefly sketch how such an account may come about.

Let us assume or stipulate in the lexical entry for the verb "promise" that it is a predicate of communicative events, with suitable constraints on its satisfaction, the details of which can be left to one's favourite idea or theory of what a 'true' promise actually is.[42] Let me also adopt the style of representation from earlier in this section. Then we can render the following structure as an initial representation of the contribution of an utterance of "I promise to finish" by me, the present author, or of "Paul promises to finish," with reference to me, again.

$$\begin{array}{|l}
\hline
1\ e_1 \\
\hline
REL[1, \text{“}I\text{”}] \\
e_1 : PRM[1, END] \\
e_1 \bigcirc t_0 \\
REL[e_1]
\end{array}$$

This representation invites a resolution of the speaker, and declares a promise event, the content of the promise being to finish, or *END*.[43] The utterance can be understood descriptively, as a description of an event that I/Paul is presently, i.e., *simultaneously*, performing.[44] So I/Paul may report, e.g., by telephone, that he is at the very moment sending an email in which he promises someone else to finish. In the following representation, the first schematic substructure (string) establishes a context with a declaration labeled p, for me,

[41] More requirements may have to be met for a self-characterizing event to count as a performative in the linguistic sense. Speaker's intentions, and speaker's verdicts, and institutional facts and social preference orders may all come into play. (See, e.g., Searle 1989; Condoravdi & Lauer 2011; Eckardt 2012; Kaufmann 2012.)

[42] It is certainly not inconvenient that we do not even need to decide the issue what a promise 'really' is, in order to make the present point. I cannot but agree with Eckardt who claims that "This is relegated to the observable and consensual intuitions of speakers of English who decide whether an actual act constitutes a promise (...) or not." (Eckardt, 2012, p. 31)

[43] Most obviously, the speaker must be, essentially indexically, resolved as the agent responsible for the occurrent utterance of the first person pronoun "I".

[44] The condition $e_1 \bigcirc t_0$ serves to indicate that the time of the declared event coincides with the time of the utterance under consideration. The precise interpretation of 'coincidence' I would like to leave for independent investigation.

and a previous declaration of an event probably in a sequence of activities that I am reporting on and that I am involved in doing, simultaneously, by email.

$$\begin{array}{|l} \hline \ldots p \ldots e_1 \\ \hline \vdots \quad \begin{array}{|l} \hline 1\ e_1 \\ \hline p = 1 \\ e_1 : PRM[p, END] \\ e_1 \bigcirc t_0 \\ e_2 < e_1 \end{array} \end{array} \qquad \begin{array}{|l} \hline \ldots p \ldots e^x \\ \hline \vdots \quad \begin{array}{|l} \hline 1\ e^y \\ \hline p = 1 \\ e^y : PRM[p, END] \\ e^y \bigcirc t_0 \\ e^x < e^y \end{array} \end{array}$$

The embedded substructure on the left constitutes a representation for the utterance of "I/Paul promise to finish", with the "I" already resolved to me, i.e., p. The declared event of promising thus includes me, p, as the agent. The condition $REL[e_1]$ has been specified as directly following some previously mentioned eventuality: $e_2 < e_1$. For instance, while pushing the SEND-button, I may report to Jane (on the phone) that "I (hereby!) promise Mary to finish (by email)". The representation on the right displays the result with labels attached.

Alternatively, my utterance can be taken to describe the very utterance event itself. Within the *DRT*-style of representation, Močnik observes, this reading results from another *resolution* of the described event, or, rather, of the way in which the reported event e_1 is supposed to be integrated in the representation of the current discourse. The specification of $REL[e_1]$ then consists in the identification of the declared event with the utterance event itself.

$$\begin{array}{|l} \hline \ldots p \ldots e_1 \\ \hline \vdots \quad \begin{array}{|l} \hline 1\ e_1 \\ \hline p = 1 \\ e_1 : PRM[p, END] \\ e_1 \bigcirc t_0 \\ e_0 = e_1 \end{array} \end{array} \qquad \begin{array}{|l} \hline \ldots p \ldots e^x \\ \hline \vdots \quad \begin{array}{|l} \hline 1\ e^y \\ \hline p = 1 \\ e^y : PRM[p, END] \\ e^y \bigcirc t_0 \\ e_0 = e^y \end{array} \end{array}$$

The (sub-)structure that results now conveys the information that it involves an event that states of itself that it is performed by me, that it is a promise, and that the content of the promise is to finish. Rendering the described event, which is the utterance event, as a promise actually means that I engage in an utterance event that is a promise. And that is just to say that I actually make a promise.

If we focus on the *representation*, we can actually *point* at the place where this promise is made, which is where (in what is actually a string) it is described (by the string). And we can also truthfully assert that it happened *there*, or *then*. For we understand the discourse (via its representation) as being felicitously embedded in an actual context. The string itself then represents the event that characterizes itself as a promise, whether it does so successfully, or not.

I leave further discussion of these and related matters for further work. (For more details about the treatment of performatives I refer to Močnik 2015.) I hope, however, that the above discussion demonstrates the potential of the indexical treatment of discourse reference developed in the first parts of this paper. In these main parts of the paper I have provided proof-theoretic motivation for an indexical account of discourse reference. With this final section I have aimed to show that such an account contributes to more computational applications of *DRT*, especially by its capacity to present indexical formulations of the resolvanda and the resolvans of underspecified structures. The implicitly self-referential character of *PLI*-indices can also be exploited further using explicitly self-referential indices so as to account for properly indexical expressions in natural language. Such may also contribute to, for instance, furthering recently proposed reflexive accounts of performative utterances.[45]

Appendix.

Subtraction (–) and Addition (+)

$rst \vdash s$

1.	rst	[ass.]
2.	(s)	[ass.]
3.	r	[ass.]
4.	s	[ass.]
5.	t	[ass.]
6.	$((s))$	[ass.]
7.	$()$	[X, 2, 6]
8.	$(((s)))$	[R]
9.	(s)	[C, 8]
10.	$()$	[X, 4, 9]
11.	(t)	[R]
12.	(st)	[R]
13.	(rst)	[R]
14.	$()$	[X, 1, 13]
15.	$((s))$	[R]
16.	s	[C, 15]

$rt, s \vdash rst$

1.	rt	[ass.]
2.	s	[ass.]
3.	(rst)	[ass.]
4.	r	[ass.]
5.	t	[ass.]
6.	$((rst))$	[ass.]
7.	$()$	[X, 3, 6]
8.	$(((rst)))$	[R]
9.	(rst)	[C, 8]
10.	(st)	[X, 4, 9]
11.	(t)	[X, 2, 10]
12.	$()$	[X, 5, 11]
13.	(t)	[R]
14.	(rt)	[R]
15.	$()$	[X, 1, 14]
16.	$((rst))$	[R]
17.	rst	[C, 16]

[45] In section one above, I have said that the annotations in deductions "unambiguously indicate" on what grounds a formula or string is added to a proof, on that very line. Many people on many different occasions say or assume things essentially alike without this raising any questions, and rightly so. Upon reflection such statements may after all have to be qualified, essentially indexically. For consider a line in a proof which says "n. 1 *s* [I, i]". The annotation "[I, i]" indicates that the string 1 *s* is obtained by an application of the introduction rule to the string on line i. What is unambiguous about this? Very obviously, and without any apparent notification required, it is supposed to be line i in the proof that hosts this line n, surely not in any other proof in the paper. Strictly speaking the reference to line i should better be understood as a reference to the line n-i lines back from the line where the annotation occurs. The inversion does not halt here. The introduction rule referred to will without any doubt be the (last version of an) introduction rule presented in the current paper. It surely goes against the law of common academic reason to disqualify some line n in a proof under consideration as wrong, by calling on a different introduction rule, formulated in another document, in a neighboring discipline perhaps. So, "unambiguous" it is indeed, as long as we assume that the line, the proof in which it occurs, the paper in which it appears, the discipline it belongs to, and the language in which it is written, is fixed. The expanding scope of such a series of assumptions should not mask their essentially indexical root, which is the very occurrence of the annotation presently under discussion. No 'name', 'label' or 'eternal description' can by itself, independently, secure a reference to the language, discipline, paper, or proof in which *this note* occurs, because—why not?—otherwise it could be something else, in a parallel universe perhaps, a universe which is alike, but which is not *this one*'s.

[Notice that in both derivations the lines 6–9 might be skipped, because they only serve to repeat, on line 9, what we already had on line 2 and 3, respectively.]

Dynamic Conjunction (DC) and Converse Exclusion (CX)

$$
\begin{array}{lll}
 & \vdots & \\
\text{i.} & r & \\
 & \vdots & \\
\text{j.} & {}^{i<}s & \\
 & \vdots & \\
\text{n-4.} & {}^{i\leq}(rs) & \text{[ass.]} \\
\text{n-3.} & {}^{i<}(s) & \text{[X, i, n-4]} \\
\text{n-2.} & () & \text{[X, j, n-3]} \\
\text{n-1.} & {}^{i\leq}((rs)) & \text{[R]} \\
\text{n.} & {}^{i\leq}rs & \text{[C]}
\end{array}
\qquad
\begin{array}{lll}
 & \vdots & \\
\text{i.} & (rs) & \\
 & \vdots & \\
\text{j.} & {}^{i\leq}r & \\
 & \vdots & \\
\text{n-1.} & {}^{i\leq}(rs) & [-, \text{i}] \\
\text{n.} & {}^{r_{j\leftarrow(n-1)}}([^{i\leq}_{n_r}]s) & \text{[X, j, n-1]}
\end{array}
$$

[Converse exclusion is obtained by a repetition, on line n-1, of the exclusive string on line i, in order to get the premises in the right order of appearance. The general formulation of the result looks awkward, but is not difficult to grasp. The indices in s that relate to the occurrence of r at line n-1, are redirected, at line n, to the occurrence of r at line j. (The indices below the threshold n_r receive the update ${}^{j<}\_$ and those above the threshold are decreased by n_r.) The indices in s *unresolved* in rs are updated by the two occurrences of ${}^{i\leq}$. (The subscript n_r figures as the threshold on line n again.)]

The Normalization Lemma The proof of the normalization lemma draws from three other lemmata.

LEMMA 5.1 (Uniform Update and Monotonicity).

$$\textit{If } s \vdash t \textit{ then } {}^{+m}s \vdash [^{+m}_{n_s}]t \textit{ (UU)}$$
$$\textit{If } s \vdash t, \textit{ then } rs \vdash t \textit{ and } sr \vdash {}^{+n_r}t \textit{ (M)}$$

Proof (Sketch). Follows from the (schematic) way in which the deduction rules are formulated. □

LEMMA 5.2 (Contraposition).

$$rs \vdash t \textit{ iff } rs(t) \vdash () \textit{ iff } r \vdash (s(t))$$

Proof. I show that [C]: $r \vdash (s(t))$ if [B]: $rs(t) \vdash ()$
[B]: $rs(t) \vdash ()$ if [A]: $rs \vdash t$
[A]: $rs \vdash t$ if [C]: $r \vdash (s(t))$

$$
\begin{array}{lll}
\multicolumn{3}{l}{r \vdash (s(t))} \\
1. & r & [\text{ass.}] \\
2. & s(t) & [\text{ass.}] \\
3. & {}^{1\leq}rs(t) & [\text{DC}, 1, 2] \\
4. & () & [\text{B}, 3] \\
\hline
5. & (s(t)) & [\text{R}]
\end{array}
\qquad
\begin{array}{lll}
\multicolumn{3}{l}{rs(t) \vdash ()} \\
1. & rs(t) & [\text{ass.}] \\
2. & {}^{+n_{rs}}rs & [\text{ass.}] \\
3. & [^{+n_{rs}}_{n_{rs}}]t & [\text{A}, 2] \\
4. & {}^{2<}[^{+n_{rs}}_{n_{rs}}]((t)) & [\text{NC}, 3] \\
\hline
5. & {}^{+n_{rs}}(rs(t)) & [\text{R}] \\
6. & () & [\text{X}, 1, 5]
\end{array}
\qquad
\begin{array}{lll}
\multicolumn{3}{l}{rs \vdash t} \\
1. & rs & [\text{ass.}] \\
2. & (t) & [\text{ass.}] \\
3. & {}^{1\leq}rs(t) & [\text{DC}, 1, 2] \\
4. & {}^{1\leq}r & [\text{ass.}] \\
5. & [^{1\leq}_{n_r}](s(t)) & [\text{C}, 4] \\
\hline
6. & {}^{1\leq}(rs(t)) & [\text{R}] \\
7. & () & [\text{X}, 3, 6] \\
\hline
8. & ((t)) & [\text{R}] \\
9. & t & [\text{C}, 8]
\end{array}
$$

[Note that hypotheses [B], [A], and [C] are employed in conjunction with (UU).] □

LEMMA 5.3 (Initialization). *$r \vdash s$ iff $1r \vdash s$*

Proof. The *only if*-part is monotonic. For the *if*-part we can first show that $r \vdash ()$ if $1r \vdash ()$ by using the introduction rule and the fact that if $1r \vdash ()$ then ${}^{+m}1r \vdash ()$, by (UU). Now suppose $1r \vdash s$. By contraposition we have $1r(s) \vdash ()$ and, hence, $r(s) \vdash ()$. By contraposition again we find $r \vdash s$. □

LEMMA 5.4 (Normalization). *$s \vdash t$ iff $s^* \vdash t$*

Actually this says that if ${}^{i}s \rightsquigarrow z^{i}$ then $s \vdash c$ iff $z \vdash c$. (I use 'c' now to avoid confusion with 't' below.)

Proof. The proof proceeds by induction on the initialization of s, from left to right and depth first. There are three cases, the second trivial.

1. $n\ r\ 1\ t \vdash c$ iff $n\ 1\ {}^{+1}r\ t \vdash c$,
2. $r\ Rv\ t \vdash c$ iff $r\ Rv\ t \vdash c$, and
3. $r\ (s)\ t \vdash c$ iff $r\ (z)\ t \vdash c$, if ${}^{i}s \rightsquigarrow z^{i}$.

In the proofs we may safely assume that $n_r = 0$. For, by the definition of $\rightsquigarrow$, the initial string r is already in Kamp canonical form, i.e., a string nr', where n is a string of n declarations, and $n_{r'} = 0$. The initialization lemma implies that the initial string of declarations can be safely ignored.

CASE 1. I first show that $r1s \vdash ()$ [A] iff ${}^{+1}rs \vdash ()$ [B]. (Note that $n_r = 0$.)

$$
\begin{array}{lll}
\multicolumn{3}{l}{\textit{[A] if [B]}} \\
1. & r1s & [\text{ass.}] \\
2. & {}^{+n_s+1}r & [-, 1] \\
3. & {}^{+n_s}s & [-, 1] \\
4. & {}^{+2n_s+1}r^{+2n_s}s & [+, 2, 3] \\
5. & () & [\text{B}]
\end{array}
\qquad
\begin{array}{lll}
\multicolumn{3}{l}{\textit{[B] if [A]}} \\
1. & {}^{+1}rs & [\text{ass.}] \\
2. & {}^{+n_s+1}r & [-, 1] \\
3. & {}^{+n_s}s & [-, 2] \\
4. & {}^{+2n_s+1}1s & [\text{I}, 3(1)] \\
5. & {}^{+3n_s+2}r1s & [+, 2, 4] \\
6. & () & [\text{A}]
\end{array}
$$

[Line 4 on the left has the string ${}^{+1}rs$ updated with the fact that s has occurred twice; Line 4 and 5 on the right have two updates of the whole strings $1s$ and $r1s$, respectively.] Now we find that $r1t \vdash c$ iff [contraposition] $r1t(c) \vdash ()$ iff [above] ${}^{+1}rt(c) \vdash ()$ iff [initialization] $1^{+1}rt(c) \vdash ()$ iff [contraposition] $1^{+1}rt \vdash c$.

CASE 3. We can use the induction hypothesis [IH] that $s \vdash c$ iff $z \vdash c$. I first show that if [IH] then $r(s) \vdash r(z) \vdash r(s)$.

$r(s) \vdash r(z)$

1.	$r(s)$	[ass.]
2.	r	[$-$, 1]
3.	(s)	[$-$, 1]
4.	z	[ass.]
5.	${}^{3\leq}s$	[IH, 4]
6.	$()$	[CX, 5, 3]
7.	(z)	[R]
8.	$r(z)$	[+, 2, 7]

That $r(z) \vdash r(s)$ is shown analogously, with s and z exchanged.

[Since $s \vdash {}^{+n_s}s$ we have $z \vdash {}^{+n_s}s$ by [IH]. Since $n_s = n_z$, we conclude ${}^{+n_z}s$ at line 5, which can be written as ${}^{3\leq}s$. As for lines 3 and 8, recall that $n_r = 0$.]

Now we find that $r(s)t \vdash c$ iff [contraposition] $r(s) \vdash (t(c))$ iff [above] $r(z) \vdash (t(c))$ iff [contraposition] $r(z)t \vdash c$. □

§6. Acknowledgments Previous versions of the work reported upon here have been presented at a workshop held in the honor of Hans Kamp at the Université Lorraine, October 2014, and at the workshop "Going Heim" held at the University of Connecticut, May 2015. I want to thank Hans Kamp, Irene Heim, and the organizers of and audiences on both occasions, as well as Maša Močnik, Martin Stokhof, Henk Zeevat and an anonymous reviewer, for insightful remarks and comments. All errors, formal and conceptual, are surely mine.

BIBLIOGRAPHY

Asher, N. & Lascarides, A. (2003). *Logics of Conversation*. Cambridge: Cambridge University Press.

Barker, C. (2002). Continuations and the nature of quantification. *Natural Language Semantics*, **10**(3), 211–42.

Bittner, M. (2001). Surface composition as bridging. *Journal of Semantics*, **18**, 127–177.

Bittner, M. (2014a). Perspectival discourse referents for indexicals. In Greene, H., editor. *Proceedings of the Seventh Conference on the Semantics of Under-Represented Languages in the Americas (SULA 7, 2012)*. Massachusetts, GLSA, Amherst, pp. 1–22.

Bittner, M. (2014b). *Temporality: Universals and Variation*. Oxford: Wiley-Blackwell.

Condoravdi, C. & Lauer, S. (2011). Performative verbs and performative acts. In Reich, I., Horch, E., and Pauly, D., editors. *Proceedings of Sinn und Bedeutung 15*. Saarbrücken: Saarland University Press, pp. 1–15.

de Bruijn, N. G. (1972). Lambda calculus notation with nameless dummies: A tool for automatic formula manipulation, with application to the Church-Rosser theorem. *Indagationes Mathematicae*, **34**, 381–392.

de Groote, P. (2006). Towards a Montagovian account of dynamics. In Gibson, M. and Howell, J., editors. *Proceedings of Semantics and Linguistic Theory XVI*. CLC Publications, Ithaca, New York, pp. 1–16.

Dekker, P. (1994). Predicate logic with anaphora (seven inch version). In Santelmann, L. & Harvey, M., editors. *Proceedings of SALT IV*. DMLL Publications, Cornell University, pp. 79–95.

Dekker, P. (2012). *Dynamic Semantics*. Berlin: Springer.

Eckardt, R. (2012). Hereby explained: An event-based account of performative utterances. *Linguistics and Philosophy*, **35**, 21–55.

Fitch, F. B. (1952). *Symbolic Logic*. New York: Roland Press.

Frege, G. (1918). Der Gedanke. Eine logische Untersuchung. *Beiträge zur Philosophie des deutschen Idealismus 2*, **1**, 58–77.

Groenendijk, J. & Stokhof, M. (1990). Dynamic Montague grammar. In Kálmán, L. and Pólos, L., editors. *Papers from The Second Symposium on Logic and Language*. Budapest: Akadémiai Kiadó, pp. 3–48.

Groenendijk, J. & Stokhof, M. (1991). Dynamic predicate logic. *Linguistics and Philosophy*, **14**(1), 39–100.

Heim, I. (1989 (1982)). *The Semantics of Definite and Indefinite Noun Phrases*. Ithaca: Garland. Originally appeared, 1982, Ph.D. Thesis, University of Massachusetts, Amherst.

Kamp, H. (1971). Formal properties of 'now'. *Theoria*, **37**, 227–273.

Kamp, H. (1984 (1981)). A theory of truth and semantic representation. In Groenendijk, J., Janssen, T. and Stokhof, M., editors. *Truth, Interpretation and Information*. Dordrecht: Foris, pp. 1–41. Originally appeared in Groenendijk, J. et al. (1981). *Formal Methods in the Study of Language*. Amsterdam: Mathematical Centre, pp. 277–322.

Kamp, H., van Genabith, J. & Reyle, U. (2011). Discourse Representation Theory. In Gabbay, D. and Guenthner, F., editors. *Handbook of Philosophical Logic*, Vol. 15. Dordrecht: Reidel, pp. 125–394.

Kaplan, D. (1979). On the logic of demonstratives. In French, P. A., Uehling, T. E. and Wettstein, H. K., editors. *Contemporary Perspectives on the Philosophy of Language*. Minneapolis: University of Minnesota Press, pp. 401–412.

Karttunen, L. (1976 (1969/71)). Discourse referents. In McCawley, J., editor. *Syntax and Semantics*, Vol. 7. Notes from the Linguistic Underground. New York: Academic Press, pp. 363–85. Originally appeared in the Proceedings of Coling 1969, reproduced by the RAND Corporation in 1971, Indiana University Linguistics Club.

Kaufmann, M. (2012). *Interpreting Imperatives*. Dordrecht: Springer.

Lewis, D. (1979). Attitudes de dicto and de se. *The Philosophical Review*, **88**(4), 513–45.

Močnik, M. (2015). Slovenian perfective and imperfective explicit performative utterances. Master's thesis, ILLC, Universiteit van Amsterdam.

Muskens, R. (1996). Combining Montague semantics and discourse representation. *Linguistics and Philosophy*, **19**(2), 143–186.

Peirce, C. S. (1933 (1880)). A boolian algebra with one constant. In Charles Hartshorne, P. W. and Burks, A., editors. *Collected Papers of Charles Sanders Peirce IV*. Cambridge: Cambridge University Press, pp. §12–20.

Perry, J. (1979). The problem of the essential indexical. *Noûs*, **13**, 3–21.

Prior, A. N. (1967). *Past, Present and Future*. Oxford: Clarendon.

Reyle, U. (1993). Dealing with ambiguities by underspecification: Construction, representation and deduction. *Journal of Semantics*, **10**, 123–179.

Russell, B. (1919). *Introduction to Mathematical Philosophy*. London: George Allen and Unwin.

Searle, J. R. (1989). How performatives work. *Linguistics and Philosophy*, **12**(5), 535–558.

Sheffer, H. M. (1913). A set of five independent postulates for boolean algebras, with application to logical constants. *Transactions of the American Mathematical Society*, **14**(4), 481–488.

Staudacher, P. (1987). Zur Semantik indefiniter Nominalphrasen. In Asbach-Schnitker, B. and Roggenhofer, J., editors. *Neuere Forschungen zur Wortbildung und Historiographie der Linguistik. Festgabe für Herbert E. Brekle*. Tübingen: Gunter Narr Verlag, pp. 239–258.

van der Sandt, R. A. (1992). Presupposition projection as anaphora resolution. *Journal of Semantics*, **9**(4), 333–377.

van Eijck, J. (2001). Incremental dynamics. *Journal of Logic, Language and Information*, **11**(3), 319–351.

Veltman, F. (2000). Proof systems for Dynamic Predicate Logic. Handout, available from http://philpapers.org/rec/VELPSF.

Vermeulen, C. F. (1993). Sequence semantics for dynamic predicate logic. *Journal of Logic, Language and Information*, **2**, 217–54.

Webber, B. L. (1978). A formal approach to discourse anaphora. BBN Report No. 3761, Bolt Beranek and Newman Inc.

Wittgenstein, L. (1922). *Tractatus Logico-Philosophicus*. Oxford: Routledge and Kegan. Originally appeared in 1921 as "Logisch-Philosophische Abhandlung" in the *Annalen der Naturphilosophie*, 14.

ILLC/DEPARTMENT OF PHILOSOPHY
UNIVERSITY OF AMSTERDAM
THE NETHERLANDS
E-mail: P.J.E.Dekker@uva.nl