Dynamic semantics, discourse semantics and continuations

Nicholas Asher
CNRS IRIT, Université Paul Sabatier, Toulouse
&
Depts. of Philosophy and Linguistics, University of Texas at Austin
Joint work with Sylvain Pogodalla

September 4, 2010

- DRT: a door for linguists to the world of formal interpretation beyond the sentence.
- The original "fil conducteur": anaphoric expressions in a variety of contexts (quantificational, modal, attitudinal).
- The turn to discourse structure: each sentence or elementary discourse unit is an anaphoric expression linking to the discourse context with one or more discourse relations with implications for:
  temporal structure, presupposition, pronominal anaphora and ellipsis of various kinds, scalar implicatures, *inter alia*.
- Empirical investigations on corpora (definition of elementary discourse unit, verification of SDRT's right frontier)
- Dialogue and strategic conversation

- The original fragment of DRT is rather classical. A semantic equivalence between DRSs and first order formulas (Fernando).
- All dynamics takes place at the level of representation via DRS merge.
- Applying DRT to modality and attitudes changed the simple relationship between DRS and model theory; sets of world assignment function pairs became standard denotations for DRSs. Problems of well-foundedness (Frank).
- The status of standardly declared discourse referents became unclear logically (at least for me), somewhere between free and bound variables.
- Extending DRT to deal with structured discourse contexts featuring the discourse roles of constituents confirmed these trends. The representational level became heavy (cf. also treatments of presupposition).

- deriving compositional meanings forced us to postulate "strange" types: the type of assignment and the type of functions from assignments to truth values (the type of a sentence denotation or proposition).
- The semantic value of a dynamic formula or DRS is a relation over pairs (w, g). where w is a world and g an assignment. To make this functional we need to "lift" the type to a function from the powerset of such pairs to itself (Fernando 1994). We can model a set of pairs as a function from pairs to T.
- Sowe need a type of assignments σ and a type for worlds s. The type of formulas Ω becomes rather complex:
  $((s \times \sigma) \rightarrow \text{T}) \rightarrow ((s \times \sigma) \rightarrow \text{T})$, cf also Broseavnu (2009)
- Dynamic quantification required the introduction of a type v for discourse referents or stores (Muskens 1996, Asher 1993)

Other problems

- problems with variable clash
- strange consequence relation with the relational interpretation if we want to capture anaphoric dependencies

$\phi \models \psi$ iff for all structures M and for all assignments f and g such that $f\|\phi\|^M g$, there is an h such that $g\|\phi\|^M h$

(1) John walked. So he moved.

(2) $Fx \wedge \exists x \neg Fx \not\models Fx \wedge \exists x \neg Fx$

We were missing something

- Many of us were ignorant of earlier but parallel developments to dynamic semantics in the theory of computation.
- The theory of continuations in an abstract setting by Moggi (1991) abstracts away from the nature of a programming environment and isolates key features— Continuation semantics (CS) carries this over into linguistics (de Groote 2006, Barker & Shan 2006, Bernardi & Mortgat 2010)
- Divisions of labor: Why not complicate lexical entries and incorporate the relational nature of meaning into them but leave the logic and the method of composition simple? That's the view of CS.
- CS abstracts away from DRT but also DPL and Update Semantics.

What do you need for continuation semantics

- a lexicon that uses the simply typed λ calculus (IL or TY2)
- a notion of what a left context is (data structure)
- a binder rule: A text meaning $\times$ sentence meaning $\longrightarrow$ Text meaning.

- DPL: a context is an assignment function, binder rule is relational composition
- DRT: a context is a DRS, binder rule DRS merge
- Stalnakerian semantics: a context is a set of possible worlds, a binder rule is $\cap$

- change the type of propositions again to include not only a "left" or "input" context but also a "right" or output context. This idea embeds dynamic semantics into classical HOL.
- Sentence Terms and Types:
  - $\lambda i \lambda o \phi$
  - Ω: $\gamma \rightarrow (\gamma \rightarrow \textsc{prop}) \rightarrow \textsc{prop}$

  The final outcome of a discourse should be a proposition. So an output context is a defined type $\gamma \rightarrow \textsc{prop}$
  where PROP is the type of propositions functions from worlds or more complex tuples of indices to truth values, structured propositions or simply truth values).

The last bit of the basic idea is to say how a text T which is sensitive to both a left and a right context and so has the form $\lambda i \lambda o \phi$ combines with a sentence, which is of the same type, to its right.

Where $\|T\|$ stands for the λ term or meaning of T:

(3) $\|T.S\| = \lambda i \lambda o \ \|T\| i (\lambda i' \|S\| i' o)$

That is, the text to date T takes the meaning of S as its right context, or rather the meaning of S suitably applied and abstracted so that it can be of o type.

A quick type check on $\lambda i' \|S\| i' o$ confirms that this is indeed the right output: $\|S\|\colon \gamma \rightarrow (\gamma \rightarrow \textsc{prop})\textsc{prop})$; $\lambda i'[\|S\| i' o]\colon \gamma \rightarrow (\gamma \rightarrow (\gamma \rightarrow \textsc{prop}) \rightarrow \textsc{prop})[\gamma][(\gamma \rightarrow \textsc{prop})]$ which is just $\gamma \rightarrow \textsc{prop})$.

The following types permit us to get the right logical forms expressing classical truth conditions for discourses corresponding to those in the simple first order fragment of DRT:

- $\gamma \rightarrow \textsc{prop} := \sigma$
- The type $\Omega : \gamma \rightarrow \sigma \rightarrow \textsc{prop}$ instead of $\textsc{prop}$.
- The type of a noun: In MG we have $\textsc{e} \rightarrow \textsc{prop}$; so here we have $\textsc{e} \rightarrow \gamma \rightarrow \sigma \rightarrow \textsc{prop}$ or $\textsc{e} \rightarrow \Omega$
- $man : \lambda x \lambda i \lambda o \ \mathrm{man}(x) \wedge o(i))$
- The type of a DP $\textsc{dp}$: $(\textsc{e} \rightarrow \Omega) \rightarrow \Omega$. But in de Groote's system, this means: that we have: $(\textsc{e} \rightarrow \gamma \rightarrow \sigma \rightarrow \textsc{prop}) \rightarrow (\gamma \rightarrow \sigma \rightarrow \textsc{prop}$

- the type of determiners is as usual: $\textsc{n} \longrightarrow \textsc{dp}$
- the entry for *every*: $\lambda P \lambda Q \lambda i \lambda o (\forall x (\neg (P x i (\lambda i' \neg Q x (i' + x) \lambda i \top))) \wedge o(i))$
- the entry for *a*: $\lambda P \lambda Q \lambda i \lambda o \exists x P x i (\lambda i' Q x (i' + x) o)$
- the entry for *sleep* $: \lambda \Phi^{\mathrm{DP}}\ \lambda i \lambda o \Phi (\lambda x \lambda i' \lambda o'\ \mathrm{sleep}(x) \wedge o'(i'))io$

Example

(4) A man is sleeping. He is snoring.

(5) $\lambda i \lambda o.\exists x\ (\mathrm{man}(x) \wedge \mathrm{sleeping}(x) \wedge o(i+x))$

(6) $\lambda i \lambda o.(\mathrm{snoring}(\mathrm{sel}(i)) \wedge o(i))$

The derivation:

$$\lambda i \lambda o[\lambda i \lambda o \exists x\ (\mathrm{man}(x) \wedge \mathrm{sleeping}(x) \wedge o(i+x))]i(\lambda i'(\lambda i \lambda o(\mathrm{snoring}(\mathrm{sel}(i)) \wedge o(i))i'o) \longrightarrow_\beta$$

$$\lambda i \lambda o.\exists x\ (\mathrm{man}(x) \wedge \mathrm{sleeping}(x) \wedge (\lambda i'(\lambda o(\mathrm{snoring}(\mathrm{sel}(i')) \wedge o(i'))o)(i+x))] \longrightarrow_\beta$$

$$\lambda i \lambda o.\exists x\ (\mathrm{man}(x) \wedge \mathrm{sleeping}(x) \wedge \mathrm{snoring}(\mathrm{sel}(i+x)) \wedge o(i+x))$$

With an empty input context and the tautologous continuation we get:
$\exists x\ (\mathrm{man}(x) \wedge \mathrm{sleeping}(x) \wedge \mathrm{snoring}(\mathrm{sel}(< x >)) \wedge \top)$

Crucial to DIL's account of intersentential anaphora are the modalities, $\langle x/d \rangle$, where d identifies an assignment.

$\langle x/d \rangle \phi$ evaluates ϕ relative to an assignment in which the value of d = the value of x in the current state.

In DIL, quantification over x in $\langle x/d \rangle \phi$ provides dynamic binding Dekker (1999):

(7) $\lambda p.\exists x.\langle x/d \rangle {}^{\vee} p[{}^{\wedge} U d] \rightarrow_{\beta} \exists x.Ux$

The Binder rule for DIL:

$$[\![T]\!].[\![S]\!] = \lambda p([\![T]\!]({}^{\wedge}[\![S]\!](p))) \tag{8}$$

(9) $\lambda p.\exists x \langle x/d \rangle.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge^{\vee} p)$

(10) $\lambda p.\mathbf{snoring}\ (d) \wedge^{\vee} p)$

(11) $\lambda p.\exists x.(\mathbf{man}\ x) \wedge (\mathbf{sleeping}\ x) \wedge \mathbf{snoring}\ (d) \wedge^{\vee} p)$

The DIL derivation capitalizes on the reduction properties of state switcher $\langle x/d \rangle$ and crucially depends on DIL's unorthodox "worlds", which are assignments.

Compare de Groote's CS, whose model theory and logic are completely classical; and there are no "funny entities" in E or types for assignment functions.

The left context list structure builds in effect assignment functions *internally*, via the interpretation of update operator :: and captures the structure of an assignment function

More generally:

- the greater generality of a left context lets us extend dynamic semantics to new areas relatively straightforwardly.
- a slight extension of de Groote 2006 to handle dynamic generalized quantifiers entails conservativity (due to the combination rules).

- First slogan: Epistemic modal facts are dependent on non modal facts but not vice versa (Veltman)

(12) It might be sunny. But it's not sunny (easy update)

(13) # It's not sunny. But it might (for all I know) be sunny.

Because of the presuppositional nature of the modalities and the test for consistency, we must redefine our left contexts.

$$
\begin{aligned}
i &: \gamma \triangleq \textsc{Theory} \\
k &: \textsc{Theory} \longrightarrow t \\
:: &\ \mathsf{prop} \longrightarrow \textsc{Theory} \longrightarrow \textsc{Theory}
\end{aligned}
$$

- for an atomic static formulas p of type t: $\lambda i\ k.p \wedge (k\ (p :: i))$;
- for P a dynamic proposition (of type $\Omega \triangleq \gamma \rightarrow (\gamma \rightarrow t) \rightarrow t)$), we define (with $\mathbb{T} \triangleq \lambda i.\top$ the trivial continuation):
  - the dynamic negation $\neg_d \triangleq \lambda P.\lambda i\ k.(\neg(P\ i\mathbb{T})) \wedge (k\ ((\neg P\ i\ \mathbb{T}) :: i))$;
  - the dynamic conjunction: $\wedge_d \triangleq \lambda P\ Q.\lambda i\ k.P\ i\ (\lambda i'.Q\ i'\ k)$;
  - the dynamic modality $\Diamond_d \triangleq \lambda P.\lambda i\ k.(\textsc{test}\ P)\ i\ (\lambda i'.(k\ i') \wedge (\Diamond(P\ i'\ \mathbb{T})))$ where $\Diamond$ is the classic static modality.

The Test Operation

$$\text{TEST } P = \lambda i\ k.\texttt{if } (\texttt{EVAL } P\ i\ \mathbb{T})\ \texttt{then } (k\ i)\ \texttt{else } (\texttt{raise Halt}) \tag{14}$$

$\lozenge_d$ yields a continuation iff there is at least one world that verifies all the information in the discourse context together with the content under the scope of the modal.

If not it raises an exception `Halt`, whose effect is to stop the evaluation. This captures Veltman's intuition that there is no possible continuation in the troublesome case.

A reminder:

(15) A wolf might walk in. It would eat you first.

(16) A wolf might walk in. It *will eat you first.

(17) A wolf is outside. He might eat you.

Classic DRT predicts (16) to be bad and (17 to be good, as intuitions warrant. But it also predicts (16) is bad.

- Second slogan: Some modal facts are dependent on other modal facts

The logical form for (15) in DRT accounts (Roberts 1987, Frank 1997) is:

(18) $\Diamond(?, \exists x(\text{wolf}(x)...)) \wedge \Box(\exists x(\text{wolf}(x)...), \text{eat}(x, u))$.

CS for Modal subordination requires left contexts to be records.

The contain:

- a modal base
- a store of factually introduced discourse entities $\mathsf{m\_ref}$
- a store of possible discourse entities $\mathsf{f\_ref}$

Indefinites will be able to add the introduced variables in $\mathsf{m\_ref}$ when in a modal context
$(\{\mathsf{m\_ref} = x :: i.\mathsf{m\_ref}; \mathsf{f\_ref} = i.\mathsf{f\_ref}\})$
and in $\mathsf{f\_ref}$ when in the actual context
$(\{\mathsf{m\_ref} = i.\mathsf{m\_ref}; \mathsf{f\_ref} = x :: i.\mathsf{f\_ref}\})$.

A pronoun also will have a different selection function depending on its environment–
$\mathtt{sel}\ i.\mathsf{f\_ref}$ and $(\mathtt{sel}\ i.\mathsf{m\_ref} \cup i.\mathsf{f\_ref})$

Our CS now will use two continuations: one containing facts about the actual world, one containing facts about live possibilities the discourse describes.

Instead of producing at the end the t type, sentences now produce a pair of claims: one for the epistemic worlds, one for the factual world. We model the pair with the higher-order type function of the type $(t \to t \to t) \to t$.

These changes induce a change in type for Ω:

$$[\![s]\!] = \gamma \to (\gamma \to t) \to (\gamma \to t) \to (t \to t \to t) \to t \triangleq \Omega \qquad (19)$$

$$
\begin{aligned}
[\![c_a^m]\!] &= \lambda P\,Q.\lambda i\,k_1\;k_2\;f.f(\exists x.P\,x\,\{i\;\texttt{with}\;\textsf{m\_ref} = x :: i.\textsf{m\_ref}\}\,(\lambda i'.Q\,x\,i'\,k_1\;k_2\;\Pi_1)\;k_2\Pi_1)\;(k_2\;i)\\
[\![c_a^f]\!] &= \lambda P\,Q.\lambda i\,k_1\;k_2\;f.\exists x.f[k_1\,\{i\;\texttt{with}\;\textsf{f\_ref} = x :: i\}]\\
&\qquad\qquad [P\,x\,\{i\;\texttt{with}\;\textsf{f\_ref} = x :: i\}\,k_1\,(\lambda i'.Q\,x\,i'\,k_1\,k_2\Pi_2)\;\Pi_2]\\
[\![c_{it}^m]\!] &= \lambda P\,i\,k_1\;k_2\;f.P\,(\texttt{sel}_b\,i.\textsf{m\_ref} \cup i.\textsf{f\_ref})\,i\,k_1\;k_2\;f\\
[\![c_{it}^f]\!] &= \lambda P\,i\,k_1\;k_2\;f.P\,(\texttt{sel}_b\,i.\textsf{f\_ref})\,i\,k_1\;k_2 f\\
[\![c_{might}]\!] &= \lambda v\,s.\lambda i\,k_1\;k_2\;f.f(\Diamond(i.\textsf{base} \wedge (v\,s\,i\,k_1\;k_2\Pi_1)))\;(k_2\;i)
\end{aligned}
$$

Table 1: Modal and factual contexts

Together with the lexical entries of Table 1, we get (with $t_2 = c_{will}\;c_{growl}\;c_{it}$):

$[\![t_0]\!] = \lambda i\,k_1\;k_2 f.f[\Diamond(i.\textsf{base} \wedge \exists x.(\mathbf{wolf}\;x) \wedge ((\mathbf{enter}\;x)$
$\wedge\,(k_1\;\{i\;\;\texttt{with}\;\;\textsf{m\_ref} = x :: i.\textsf{m\_ref}\;\;\texttt{and}\;\;\textsf{base} = (\mathbf{wolf}\;x) \wedge (\mathbf{enter}\;x) \wedge i.\textsf{base}\})))]\;[k_2\;i]$

$[\![t_1]\!] = \lambda i\,k_1\;k_2 f.f[\Box(i.\textsf{base} \Rightarrow ((\mathbf{growl}\;(\texttt{sel}\;i.\textsf{m\_ref} \cup i.\textsf{f\_ref}))$
$\wedge\,(k_1\;\{i\;\;\texttt{with}\;\;\textsf{base} = (\mathbf{growl}\;(\texttt{sel}\;i.\textsf{m\_ref} \cup i.\textsf{f\_ref})) \wedge i.\textsf{base}\})))]\;[k_2\;i]$

$[\![t_2]\!] = \lambda i\,k_1\;k_2 f.f[k_1\;i]\;[(\mathbf{growl}\;(\texttt{sel}\;i.\textsf{f\_ref})) \wedge (k_2\;i)]$

To combine our more complex logical forms, we require a new binder rule:

$$[\![S_1 \ . \ S_2]\!] = \lambda i \ k_1 \ k_2 \ f.[\![S_1]\!] \ i \ (\lambda i'.[\![S_2]\!] \ i' \ k_1 \ k_2 \Pi_1) \ (\lambda i'.[\![S_2]\!] \ i' \ k_1 \ k_2 \Pi_2) \ f \quad (20)$$

Interpreting our examples with empty environments ($_{\mathsf{empty}} = \{_{\mathsf{m\_ref}} = \mathtt{nil};_{\mathsf{base}} = \top;_{\mathsf{f\_ref}} = \mathtt{nil}\}$), trivial continuations ($\mathbb{T} = \lambda i.\top$). Conjunction of the two components ($\mathsf{Conj} = \lambda b_1 b_2.b_1 \wedge b_2$) yields the type t. This yields:

(21) $[\![t_0 \ . \ t_1]\!]_{\mathsf{empty}} \ \mathbb{T} \ \mathbb{T} \ \mathsf{Conj} = [\Diamond(\top \wedge (\exists x.(\mathbf{wolf} \ x) \wedge ((\mathbf{enter} \ x) \wedge (\Box(((\mathbf{wolf} \ x) \wedge (\mathbf{enter} \ x)) \Rightarrow (\mathbf{growl} \ (\mathtt{sel} \ ((x :: \mathtt{nil}) \cup \mathtt{nil}))))))))] \wedge \top$

(22) $[\![t_0 \ . \ t_2]\!]_{\mathsf{empty}} \ \mathbb{T} \ \mathbb{T} \ \mathsf{Conj} = [\Diamond(\top \wedge (\exists x.(\mathbf{wolf} \ x) \wedge ((\mathbf{enter} \ x) \wedge \top)))] \wedge [\mathbf{growl} \ (\mathtt{sel} \ \mathtt{nil})]$

For (15), represented by $t_0.t_1$ the `sel` function has access to x, , is predicted to be OK

For (16), represented by $t_0.t_2$, is predicted to be bad, because x is not part of the accessible environment.

Now consider again (17).

(23) 17 A wolf is outside. He might eat you.

(24) $\lambda i\ k_1\ k_2\ f.\exists x.f[k_1\{i\ \texttt{with}\ {}_{\mathsf{f\_ref}} = x :: i.{}_{\mathsf{f\_ref}}\}][(\textbf{wolf}\ x) \wedge ((\textbf{Outside}\ x) \wedge (k_2\ \{i\ \texttt{with}\ {}_{\mathsf{f\_ref}} = x :: i.{}_{\mathsf{f\_ref}}\}))]$

Integrating the modalized second sentence:

(25) $\exists x.[\Diamond(\top \wedge (\textbf{eat you}\ (\texttt{sel}\ \texttt{nil} \cup (x :: \texttt{nil}))))] \wedge [(\textbf{wolf}\ x) \wedge ((\textbf{Outside}\ x))]$

We now add to the context a field $\textsf{theory}$ to the context that will contain the theory under construction which is tested as before and we have

$$\gamma \triangleq \{\textsf{m\_ref} : \gamma'; \textsf{base} : t; \textsf{f\_ref} : \gamma'; \textsf{theory} : \textsc{Theory}\} \tag{26}$$

Our final lexical entry for *might* is this:

(27) $[\![c_{might}]\!] = \lambda v\, s.\lambda i\, k_1\ k_2\ f.(\lambda P.(\textsc{test}\ P)\ i.\textsf{theory}$
$(\lambda i'\ o'_1\ o'_2\ f'.f'(\Diamond(i'.\textsf{base} \wedge (P\ i'\ o'_1\ o'_2\ P i_1))))) (v\ s)\ i\ k_1\ k_2\ f$

First we specify our language. In keeping with earlier work, we assume the language is that of IL together with a set of labels $\pi, \pi_1, \pi_2, \ldots$, representing discourse constituents and a set of relation symbols that represent discourse relations over constituents. So in addition to the normal IL terms (with an extra argument for labels), we will also have:

- $\pi, \pi_1, \ldots : \ell$
- $R(\pi_1, \pi_2, \pi) : t$
  where R is a relation symbol for a discourse relation. This formula says that the discourse relation R holds between π_1 and π_2 in constituent π.

Left contexts are records

Binder rule is standard

Sentence semantics is more complicated (could have complicated the binder rule)

- $?_R(\pi_S, ?, ?) \wedge \pi_S\colon \|S\|$

That is, a sentence requires the resolution of an attachment point in some environment with some discourse relation.

In CS, this means:

$$[\![S]\!] = \lambda io.\exists \pi_s.P_S \wedge \texttt{sel}_\rho(\texttt{sel}_L(i), \pi_s, \texttt{sel}_L(i)) \wedge (o\, \upsilon(i, \pi_2)) \qquad (28)$$

- $\texttt{sel}_L : \gamma \rightarrow \ell$ extracts a label from the left context that is SDRT accessible
- $\texttt{sel}_E : \gamma \rightarrow \ell \rightarrow e$ extracts a discourse referent from the set of accessible discourse referents associated with a label.
- $\texttt{sel}_\rho : \gamma \rightarrow \ell \rightarrow \ell \rightarrow \ell \rightarrow t$. This function is used to pick a discourse relation (*i.e.* a ternary relation) linking a label chosen from i the current context and returns a proposition.
- $\upsilon: \gamma \rightarrow \ell \rightarrow \gamma$. This is the update function that changes the left context record in virtue of the information contained in S and the linking of its label to some label in i via a chosen discourse relation. This update function is defined in terms of SDRT's glue logic which operates on fields of a left context.

The sentential semantics rule presupposes that there are at least two labels in the left context. When this is not met, we have the exception handling clause:

$$[\![S]\!] = \lambda io.\exists\pi.\exists\pi_s.P_S \wedge \mathtt{sel}_\rho(\mathtt{sel}_L(i), \pi_s, \pi) \wedge (o\, \upsilon(i, \pi_S)) \tag{29}$$

We illustrate some simple cases of updating:

(30) (π_1) A man walked in. (π_2) He coughed.

Here we compute a discourse relation between π_1 and π_2, which is *Narration* (meaning $\texttt{sel}_\rho(\texttt{sel}_L(i'), \pi_2, \pi)$ gets resolved to $Narration(\pi_1, \pi_2, \pi)$). π_2 must be interpreted as in (28) because only one label is available:

(31) $\lambda io.\exists\pi_1.\exists x.M(x, \pi_1) \wedge W(x, \pi_1) \wedge o(\upsilon(i, \pi_1)))i$
$[\lambda i'.\exists\pi.\exists\pi_2.C(\texttt{sel}_E(i'), \pi_2) \wedge \texttt{sel}_\rho(\texttt{sel}_L(i'), \pi_2, \pi) \wedge o(\upsilon(i', \pi_2))] \rightarrow_\beta$

(32) $\lambda io.\exists\pi_1.\exists x.M(x, \pi_1) \wedge W(x, \pi_1) \wedge \exists\pi.\exists\pi_2.C(\texttt{sel}_E(\upsilon(\upsilon(i, \pi_1), \pi_2), \pi_2))$
$\wedge Narration(\pi_1, \pi_2, \pi) \wedge o(\upsilon(\upsilon(i, \pi_1), \pi_2))))$

The stages in the computation reveal the evolution of the left context as the discourse is processed. Supposing that we have a record i_0 with empty fields for contents, discourse entities and discourse labels, the first sentence provides us with an update to the left context as follows:

$$\begin{bmatrix} \textsf{Labels} = & \pi_1 \\ \textsf{Available Labels} = & \pi_1 \\ \textsf{Discourse entities} = & (\pi_1, x) \\ \textsf{Content} = & \exists x \exists \pi_1 . M(x, \pi_1) \wedge W(x, \pi_1) \end{bmatrix} \tag{33}$$

After the update with the second sentence of (29), assuming $\texttt{sel}_E(\upsilon(\upsilon(i_0, \pi_1), \pi_2), \pi_2) = x$ we have for our context:

$$\begin{bmatrix} \textsf{Labels} = & \pi_1, \pi_2, \pi \\ \textsf{Available Labels} = & \pi_2, \pi \\ \textsf{Discourse entities} = & (\pi_1, x), (\pi_2, x) \\ \textsf{Content} = & \exists x \exists \pi_1.(M(w, \pi_1) \wedge W(x, \pi_1)) \wedge \exists \pi.\exists \pi_2.C(x, \pi_2) \wedge Narration(\pi_1, \pi_2, \pi) \end{bmatrix} \tag{34}$$

(35) Sam wants to marry an Italian. He hopes she will be rich.

(36) Hob thinks a witch has blighted his mare, and Nob thinks she has killed his cow.

The intuitive truth conditions for (34) and (35) require a modal independence. In (34) Sam doesn't *want* to hope that his Italian is rich—he simply hopes that she will be rich. Similarly, (35) is not intuitively a report about what Hob believes about Nob or vice-versa.

CS is flexible enough to allow lexical entries inducing a wide scope reading for the existential quantifier over unembedded modalities for MS.

(37) $\exists x\, (\mathcal{B}_h(\mathbf{witch}(x) \wedge \mathbf{blighted\ ...}(x)) \wedge \mathcal{B}_n(\mathbf{gave\ warts...}(\mathtt{sel}\ ((x :: \mathtt{nil})))))$

But the truth of (36) problematically requires that there is an object in the world of evaluation that is a witch in all of Hob and Nob's belief worlds.

By exploiting the contents of DRSs as sets of world assignment pairs we can in DRT coordinate the interpretations of two attitude descriptions by constraining the proper embeddings of each to agree on assignments to certain pairs of discourse referents.

On such an approach, *a witch* in (35) was treated with narrow scope;various possible witches could be the value of this variable or discourse referent under the coordinated assignments. The Hob-Nob examples, like the MS examples, showed the peculiarities of the logical framework of DRT in which discourse referents have a kind of hybrid status, somewhere between bound and free variables.

Asher & Pogodalla propose a story that exploits coercion and our TEST operator. The pronoun in the second sentence presupposes the presence of an antecedent of the appropriate type.

Like other presuppositional triggers, the pronoun places a TEST on the antecedent context that there be an antecedent of the appropriate type in the left context. Given our interpretive assumptions, this is not the case. Thus, the TEST fails.

But the semantics of the pronoun also licenses an accommodation mechanism for the exception, whereupon the antecedent changes its type from e to $s \rightarrow e$, the type of an individual concept.

To treat the exception, we then redo the entire computation having lifted the type of the indefinite to a quantifier over individual concepts.

To treat the exception case for the TEST properly, we must precisify the particular kind of individual concept at issue.

If witch A blighted Hob's mare in a Hob belief world, then at least some of the belief worlds of Nob will have witch A giving his cow warts; and anyone who gave Nob's cow warts in one of Nob's belief worlds is a witch who blighted Hob's mare in at least one of Hob's belief worlds.

This is what the coordinated dependencies in DRT capture.

This requires a more complex binder rule making special use of the modal continuation.

- Continuation style semantics puts the dynamicity of dynamic semantics into a more abstract setting embedding it the classical notion of consequence.
- CS focuses attention on left contexts, binder rules and lexical entries.
- CS's use of the λ calculus makes the construction of logical forms for discourse semantics something familiar with pleasing inferential and computational properties.
- Enables a tight connection to syntax via Abstract Categorial Grammars (Kanazawa, Salvati)
- Since left contexts can be basically any data structure, a wide variety of context sensitive phenomena can be treated: not only anaphoric dependencies, but discourse dependencies involving larger structures, and lexical phenomena such as coercions.