Compositionality and Context

Two and a half 90-minute lectures

`Tim.Fernando@tcd.ie`

ESSLLI XIII (Helsinki, August 2001)

These notes are based on lectures constituting the second half of a week-long course on **Compositionality and Context**, the first half of which was taught by Dag Westerståhl. The lectures were intended to complement DW's lectures, although I have tried to make them reasonably self-contained. §§1, 2 are primarily about contexts for *compositionality*, putting (as it were) compositionality *in* context. With a scheme for compositional frameworks in place (and some motivation from §2.5), §3 turns to notions of context within such a framework, extending these to include choices among (and leaps between) such frameworks. That is, §3 is about *context* in compositionality (incorporating context change into meaning).

1 Complicating Hodges?

References

Wilfrid Hodges, Formal features of compositionality. *J. Logic, Language and Information*, 10(1):7-28, 2001. [`www.maths.qmw.ac.uk/~wilfrid`]

Marcus Kracht, Strict compositionality and literal movement grammars. In *Logical Aspects of Computational Linguistics '98*, LNAI 2014. Springer, 2001. [`www.math.fu-berlin.de/~kracht`]

Peter Pagin and Dag Westerståhl, Editorial: Compositionality: Current Issues. *J. Logic, Language and Information*, 10(1):1-5, 2001.

1.1 Compositionality

"The standard formulation" (Pagin and Westerståhl 2001, page 1)

(C) *The*[a] *meaning*[b] *of a complex*[c] *expression*[d] *is determined*[a] *by the*[a] *meaning of its parts*[e] *and the*[a] *"mode of composition."*[f,g]

a- Does 'the' presuppose existence and uniqueness? In particular, does (C) take for granted that every expression has a unique meaning, and a unique mode of composition?

b- What sort of thing is a 'meaning'? Model-theoretic? Representational?...

c- What does it mean to be 'complex'? Is (C) an inductive recipe for calculating meanings of complex expressions, based on meanings of non-complex (*atomic*) expressions?

d- Is the notion of an 'expression' subject to theoretical construction/bias, or is it composed of theory-neutral, strictly perceptible ingredients (audible and/or visible)?

e- Are ‘parts’ immediate (as opposed to more general hereditary) constituents? Are parts expressions?

f- What is a ‘mode of composition’? Is that built into the notion of a [well-formed] expression?

g- ‘and by nothing else’?

To give (C) bite, the answer to g must be *yes*, surely?
Not so fast: it is easy to read Hodges 2001 and Kracht 2001 as saying *no* (or better: *not necessarily*) to g (as well as a).

1.2 From expressions to grammatical terms (Hodges)

Hodges collects the expressions in a set E, and is careful to distinguish an expression e from a a *grammatical term* $t \in GT$ that provides a “structural analysis” of e assuming a certain function val (defined simultaneously with GT) maps t to $\mathsf{val}(t) = e$.

> In general one expects a semantics to give different meanings to an expression under different structural analyses of the expression. So I shall assume that meanings attach to elements of the grammatical term algebra GT rather than to elements of E. In fact the set E plays only a parasitic role from this point onwards. [page 9]

Before the passage above, Hodges constructs GT from a *grammar*[1] $(E, A, \underline{\alpha})_{\alpha \in \Sigma}$ where

- A is the subset of E consisting of *atomic expressions*
- Σ is a set of n-ary function symbols α, interpreted as partial n-ary functions $\underline{\alpha}$ on E.

Putting aside for the moment the precise definition of GT,[2] the point is that Hodges takes a *semantics* for E to be a function μ, the domain of which is a subset of GT.[3] Hodges examines the problem of extending μ to a function $\hat{\mu} \supseteq \mu$ satisfying $dom(\hat{\mu}) = GT$ and

[1]We follow DW’s modification here, trading quotes ‘·’ (marking terms) for underlines $\underline{\cdot}$ (marking interpretations).

[2]For the record, GT and $\mathsf{val} : GT \to E$ are defined by simultaneous induction as follows:

(i) for every $a \in A$, $a \in GT$ and $\mathsf{val}(a) = a$

(ii) for every $\alpha \in \Sigma$ with arity n and all $t_1, \ldots, t_n \in GT$ s.t. $(\mathsf{val}(t_1), \ldots, \mathsf{val}(t_n)) \in dom(\underline{\alpha})$,

$$\alpha(t_1, \ldots, t_n) \in GT \quad \text{and} \quad \mathsf{val}(\alpha(t_1, \ldots, t_n)) = \underline{\alpha}(\mathsf{val}(t_1), \ldots, \mathsf{val}(t_n)).$$

Note that E is used implicitly in (ii), $\underline{\alpha}$ being a partial n-ary map on E. (Another natural notation for $\underline{\alpha}$ would be $\mathsf{val}(\alpha)$.)

[3]Writing M for the range/co-domain of μ (i.e. M is the set of “meanings”) and $\rightharpoondown$ for partial functions, we get the picture $\begin{array}{ccc} & & \mu & \\ & GT & \rightharpoondown & M \\ \mathsf{val} & \downarrow & & \\ & E & & \end{array}$

there is a function r such that for every complex term $\alpha(t_1, \ldots, t_n) \in GT$,

$$\hat{\mu}(\alpha(t_1, \ldots, t_n)) \quad = \quad r(\alpha, \hat{\mu}(t_1), \ldots, \hat{\mu}(t_n)) \ .$$

(GT is closed under subterms, but need not include every term built from (A, Σ).) Now, an expression e has, for every structural analysis $t \in GT$ of e (i.e. $\mathsf{val}(t) = e$), the/a meaning $\hat{\mu}(t)$. By requiring that val be surjective (onto E), Hodges ensures that every expression has a $\hat{\mu}$-meaning (and, assuming val is not 1-1, possibly several).

1.3 A question

Returning to compositionality (C), let us apply Hodges' terminology to formulate the following question.

(P) How is a meaning $\hat{\mu}(t)$ of an expression $\mathsf{val}(t)$ determined by meanings of the parts of $\mathsf{val}(t)$? In fact, what are "the parts of $\mathsf{val}(t)$"?

A possible reaction to Question (P) is to suggest that "expression" in (C) be identified with an element *not* of E, but of GT, and use the notion of "subterm" for "part." But, so long as a distinction between GT and E is drawn, what is to stop us from raising (P) for E? And in `www.math.fu-berlin.de/~kracht/` (Stand: Thu Jul 5 14:00:00 2001), we read

> My main intuition about compositionality is that present theories do not contain a notion of part of an expression. I have written on this in 'Strict Compositionality and Literal Movement Grammars'.

That paper (hereafter Kracht 2001) provides "a formal set-up of language as a semiotic system" of "signs," where, to a first approximation, expressions become "exponents."

1.4 From grammatical terms to signs (Kracht, Part 1)

Given

- a set E of *exponents* $e, \ldots$
- a set T of *types* $t, \ldots$ (not to be confused with Hodges' terms/structural analyses, though, in some sense, shadowy projections of them)
- a set M of *meanings* $m, \ldots$

Kracht defines

- a *sign over* (E, T, M) to be a triple $(e, t, m) \in E \times T \times M$ (with exponent e, type t and meaning m)
- a *language over* (E, T, M) to be a set of signs over (E, T, M)
- ε to be the first projection on $E \times T \times M$, τ the second, and μ the third,

whence for a sign σ over (E, T, M),

$$\sigma \quad = \quad (\varepsilon(\sigma), \tau(\sigma), \mu(\sigma)) \ .$$

Examples. Kracht gives the following examples of signs

$$\begin{array}{rcl} \text{`a'} & : & (\mathsf{a},\ np/n,\ \lambda P.\lambda Q.(\exists x)(P(x) \wedge Q(x))) \\ \text{`man'} & : & (\mathtt{man},\ n,\ \lambda x.\mathsf{man}'(x)) \\ \text{`walks'} & : & (\mathtt{walks},\ np\backslash n,\ \lambda x.\mathsf{walk}'(x)) \end{array}$$

the intuition being that exponents are visible (audible) while types provide some form of structural analysis. [end of Examples]

Next, to specify languages given by a grammar, he

- fixes a *signature* (F, Ω) where F is a set of function symbols, with arities specified by the function Ω from F to natural numbers
- defines a *partial* Ω*-algebra* to be a pair $\mathbf{A} = (\mathsf{A}, I)$, where A is a non-empty set (the *carrier set of* $\mathbf{A}$) and I is a function from F $(= dom(\Omega))$, mapping each $f \in F$ to a partial $\Omega(f)$-ary function $I(f)$ on A
- defines an (E, T, M)*-grammar* G to consist of a finite signature (F, Ω) and functions I_E, I_T, and I_M making (E, I_E), (T, I_T) and (M, I_M) partial Ω-algebras.

Given such an (E, T, M)-grammar G and an n-ary $f \in F$, let $I(f)$ be the partial n-ary function on $E \times T \times M$ such that for all $\sigma_1, \ldots, \sigma_n \in E \times T \times M$,

$$\begin{array}{rcl} I(f)(\sigma_1, \ldots, \sigma_n) & \simeq & (I_E(f)(\varepsilon(\sigma_1), \ldots, \varepsilon(\sigma_n)),\ I_T(f)(\tau(\sigma_1), \ldots, \tau(\sigma_n)), \\ & & I_M(f)(\mu(\sigma_1), \ldots, \mu(\sigma_n))) \end{array}$$

by which it is understood that (in particular)

$$\begin{array}{rcl} (\sigma_1, \ldots, \sigma_n) \in dom(I(f)) & \text{iff} & (\varepsilon(\sigma_1), \ldots, \varepsilon(\sigma_n)) \in dom(I_E(f)) \text{ and} \\ & & (\tau(\sigma_1), \ldots, \tau(\sigma_n)) \in dom(I_T(f)) \text{ and} \\ & & (\mu(\sigma_1), \ldots, \mu(\sigma_n)) \in dom(I_M(f)) \ . \end{array}$$

Example. Developing the example above, Kracht considers a binary $\bullet \in F$ (called *merge*), taking $I_E(\bullet)$ to be concatenation with a blank inserted between arguments

$$I_E(\bullet)(\mathtt{a}, \mathtt{man}) \quad = \quad \mathtt{a\ man}$$

$I_T(\bullet)$ to be slash-cancellation

$$I_T(\bullet)(np/n, n) \quad = \quad np$$

and $I_M(\bullet)$ to be function application

$$I_M(\bullet)(\lambda P.\lambda Q.(\exists x)(P(x) \wedge Q(x)),\ \lambda x.\mathsf{man}'(x)) \quad = \quad \lambda Q.(\exists x)(\mathsf{man}'(x) \wedge Q(x))$$

so that

$$I(\bullet)(\text{`a'},\ \text{`man'}) \quad = \quad (\mathtt{a\ man},\ np,\ \lambda Q.(\exists x)(\mathsf{man}'(x) \wedge Q(x))) \ .$$

1.5 Signs versus expressions (Kracht, Part 2)

Kracht defines the *language of* G, $L(G)$, to be the set of signs generated by Ω and I as follows. For every natural number k, let S_k be the set of signs defined from the subsets

$$F_n \quad = \quad \{f \in F : \Omega(f) = n\} \qquad \text{(for every } n\text{)}$$

of F by

$$\begin{aligned} S_0 &= \{(I_E(f), I_T(f), I_M(f)) \;:\; f \in F_0\} \qquad \text{[the \textit{lexicon} of } G\text{]} \\ S_{k+1} &= \bigcup_{n>0} \bigcup_{f \in F_n} \{I(f)(\sigma_1, \ldots, \sigma_n) \;:\; (\sigma_1, \ldots, \sigma_n) \in dom(I(f)) \cap {S_k}^n\} \;. \end{aligned}$$

The language of G is the union of these sets

$$L(G) \quad = \quad \bigcup_{k \geq 0} S_k \;.$$

Referring to $f \in F$ as a mode, Kracht writes

> Notice that our definition of grammar is modular ... we assume that modes operate independently on the exponents, the types and the meanings of the signs. Therefore, in order to define a grammar, one needs only specify the interpretation of the modes in each of the three sets E, T and M independently. This defines the algebras (E, I_E), (T, I_T) and (M, I_M). The rest is completely determined.

Relating Kracht to Hodges, we have, as a first approximation,

	Hodges		*Kracht*
E	expressions		exponents
	A (qua terms)		F_0
modes	$\alpha \in \Sigma$		$f \in F$
	$\underline{\alpha}$		$I_E(f)$
	GT	$\sim$	$L(G)$
	val	$\sim$	ε
	μ	$\sim$	μ

Table 1

The last three rows of Table 1 are not quite compatible with the first row, and can be sharpened as follows. Backing up from the set $L(G)$ of signs, define the subset $tm(G)$ of the free Ω-algebra simultaneously with a map $\theta : tm(G) \to L(G)$ by the clauses

(i) for every $f \in F_0$, $f \in tm(G)$ and $\theta(f) = (I_E(f), I_T(f), I_M(f))$

(ii) for all $n > 0$, $f \in F_n$ and $g_1, \ldots, g_n \in tm(G)$ such that $(\theta(g_1), \ldots, \theta(g_n)) \in dom(I(f))$,

$$f(g_1, \ldots, g_n) \in tm(G) \text{ and } \theta(f(g_1, \ldots, g_n)) = I(f)(\theta(g_1), \ldots, \theta(g_n)).$$

Note the similarity with Hodges' definition of GT and val. We get

Hodges		*Kracht*
GT	$\approx$	$tm(G)$
val	$\approx$	$\theta; \varepsilon$
μ	$\approx$	$\theta; \mu$
?	$\approx$	$L(G)$

Table 2

where ; is function composition, applied sequentially so that

$$\begin{array}{ccc} \textit{Hodges} & & \textit{Kracht} \\ \mathsf{val} : GT \to E & \approx & \theta; \varepsilon : tm(G) \xrightarrow{\theta} L(G) \xrightarrow{\varepsilon} E \\ \mu : GT \to M & \approx & \theta; \mu : tm(G) \xrightarrow{\theta} L(G) \xrightarrow{\mu} M \ . \end{array}$$

We keep the marks $\approx$ of imperfection and add a row '? $\approx L(G)$' as otherwise Table 1 hides the role played by $L(G)$. Notice that the definition of $tm(G)$ uses $I(f)$, and not just Table 1's $\underline{\alpha}$-counterpart, $I_E(f)$. $L(G)$ is crucial to Kracht 2001. This fact should not be lost when replacing $L(G)$ in Table 1 by $tm(G)$ in Table 2. The prominence Table 1 gives to expressions/exponents ought to be balanced against the centrality of $L(G)$, hinted by the last row of Table 2. (And remember, there is more to a sign σ in $L(G)$ than its exponent $\varepsilon(\sigma)$; σ carries with it also its type $\tau(\sigma)$, and meaning $\mu(\sigma)$.)

1.6 Back to Parts (Kracht, Part 3)

Indeed it is between signs in $L(G)$ that Kracht sets out to develop a notion of *part*, although this is based largely on the signs' exponents, which he takes to be strings.[4] "To talk meaningfully about parts of a sign," a principle called *analyticity* is imposed, the main force of which is that for every n-ary mode f and every $(\sigma_1, \ldots, \sigma_n) \in dom(I(f)) \cap L(G)^n$,

> each $\varepsilon(\sigma_i)$ occurs at least once in $\varepsilon(I(f)(\sigma_1, \ldots, \sigma_n))$.

(Or more simply, if $I_E(f)(e_1, \ldots, e_n)$ is defined then it deletes none of $e_1, \ldots, e_n$.) An important (additional) requirement is that in the generation of $L(G)$, "every step" from S_k to S_{k+1} "leaves a visible trace on the exponent," as illustrated by the following two conditions.

> *Nonemptiness*. No sign has an empty exponent.

> *Productivity*. If σ is composed from σ' by applying a unary mode, then $\varepsilon(\sigma')$ is shorter than $\varepsilon(\sigma)$.

Interestingly, Kracht concedes that

> There is plenty of evidence that in language there are empty signs and also non-productive modes. However, their use must obviously

[4]As strings, exponents have a completely straightforward notion of part — substring.

be highly restricted otherwise the determination of meaning from sound can become infeasible. So, when one looks closely at the matter it often appears that the use of empty signs and non-productive modes can be eliminated…

He adds further conditions on G, with the goal of ensuring that there be a polynomial-time decision procedure to check

given an exponent e, is there a sign $\sigma \in L(G)$ the exponent of which, $\varepsilon(\sigma)$, is e?

Now, the meaning of a sign σ is just its third projection $\mu(\sigma)$. So if we collect all the signs with a particular exponent e, we collect what e could mean — that is, arguably e's meaning. Of course, to say that

we can collect what e could mean, given (the parts of) e

is different from saying that

we can collect what e could mean, given what the parts of e could mean.

The latter comes closer to question (P) from §1.3 (above), but fails to mention a mode of composition. A somewhat simplified version of question (P) in Kracht's framework is

Given an n-ary mode f, what is the meaning of an expression built from f and parts with meanings $m_1, \ldots, m_n$?

The answer is simple: $I_M(f)(m_1, \ldots, m_n)$. A trickier version of (P) is

Given a partial n-ary function φ which is $I_E(f)$ for some n-ary mode f, what is the meaning of an expression built from φ and parts with meanings $m_1, \ldots, m_n$?

In this case, there can be many f's such that $\varphi = I_E(f)$ and accordingly many answers $I_M(f)(m_1, \ldots, m_n)$. The problem is to get our hands on them/one. Notice that in the above versions of question (P), what is important is not so much whether we are talking about an "expression" or a "sign." It is whether a "mode" gives us the function $I_M(f)$ or not.

All this suggests that perhaps question (P) isn't very interesting afterall. The surjectivity (assumed by Hodges) of val and the generation (by Kracht) of $L(G)$ from F both point to the conclusion that it is the modes that matter (the terms built from which have [like all terms] a straightforward notion of 'part' — subterm). The conditions of analyticity, non-emptiness and productivity Kracht incorporates into *strict compositionality* are all directed at reconstructing modes from their exponent projections. Moreover, Kracht's assertion that "our definition of grammar is modular" should not obscure the fact that

(†) modes (in Ω) had better be choosen carefully for an interesting language $L(G)$ to emerge from "independent" definitions (after Ω is fixed) of the modules (E, I_E), (T, I_T) and (M, I_M).

The challenge in designing a common skeleton Ω for exponents, types and meanings could well make claims of modularity sound hollow.[5] Indeed, one might, as a result, shy away from global semiotic ambitions, and seek more local and modest applications of compositionality. That is to say, rather than worrying about designing modes for three "independent" algebras (E, I_E), (T, I_T) and (M, I_M), one might focus on getting one right (impossible though that is). And for that, Hodges' simpler framework might do.

1.7 So, what next?

Section 2 is an attempt to get away with an even simpler framework than Hodges'. While perhaps not completely successful, it is, I think, instructive. It brings out a notion of *Context* that we will try to develop in different directions in the remainder of these notes/lectures, keeping in mind questions `a` to `g` from §1.1, and also turning to some concrete examples (of expressions, modes and meanings).

2 Simplifying Hodges?

Reference (in addition to Hodges 2001)

Tim Fernando, Ambiguity under changing contexts. *Linguistics and Philosophy*, 20(6):575-606, 1997.[6]

2.1 Simplification short of trivialization?

In his first lecture for this course, DW characterized triviality claims asserting the emptiness of compositionality as follows

> "Any semantics can be made 'compositional' by suitable syntactic and/or semantic manipulations." More formally:
>
> (1) For any $\mu : E \rightarrow M$ there is *another* semantics $\mu' : E' \rightarrow M'$ which is compositional and related to μ in some natural way.
>
> The **interest** of (1) depends entirely on *how* μ' is related to μ.

What DW underscores here is the tension between the $\forall$-claim in (1), implying triviality, and the relationship between μ and μ', which had better be interesting if the triviality claim is to have any content. By requiring that $\mu \subseteq \mu'$, Hodges stops short of claiming triviality. It is easy to choose a μ that has *no* compositional extension μ'.

[5] None of this is meant to deny that the semiotic framework Kracht proposes *is* interesting.

[6] Pages 585 and 589-594 are especially relevant (with the criticism of Assumption A in p.585 motivating an investigation of refinements, and pages 592-594 giving what is presented below as Theorem 4). We depart somewhat from the notation in Fernando 1997 for an easier comparison with Hodges 2001.

All papers by Fernando referred to in these notes can be obtained from `www.cs.tcd.ie/Tim.Fernando`.

Drawing on Fernando 1997, the present section relaxes the requirement $\mu \subseteq \mu'$, proposing instead that μ' be what might be called (for reasons that will become clear shortly) a *Fregean refinement* of μ. As it turns out, every semantics μ has a compositional Fregean refinement μ', raising the question: aren't we flirting with triviality here? Or rather, just how interesting could this notion of a Fregean refinement (whatever it is) be? We will see that for μ's that have compositional extensions, Fregean refinements of μ coincide exactly with Hodges' Fregean extensions of μ. The present section simplifies Hodges by

(i) sidestepping the question of when compositional extensions exist, and

(ii) passing from functions μ to their synonymies $\equiv$, setting aside the distinction between E and GT.

The shift from functions to synonymies is natural given that it is up to synonymy that Fregean extensions/refinements are unique. As for cases where compositional extensions fail to exist, I claim that Hodges' formal analysis of Frege's *Context Principle* in terms of Fregean extensions does not suffer when generalizing Fregean extensions to Fregean refinements. Indeed, it is sharpened by the construction of Fregean refinements presented below, which applies whether or not compositional extensions exist (outputing one if they do). The thrust of that construction is to view compositionality not so much as an inductive recipe for computing the meaning of an expression by computing the meaning of its parts, but rather as a co-inductive system of constraints on synonymies imposed by a suitable notion of *context*.

2.2 Compositionality "without parts": congruences

Let us review some well-known facts, mainly to fix notation and to relate that to Hodges' framework. Given an equivalence relation $\equiv$ on a set E (of "expressions"), $e \in E$, and an n-ary function $f : E^n \to E$ on E,

(i) let $e^{\equiv}$ denote the $\equiv$-equivalence class

$$e^{\equiv} \quad = \quad \{e' \in E : e \equiv e'\}$$

(ii) define $f^{\equiv} : Pow(E)^k \to Pow(E)$ by

$$f^{\equiv}(\epsilon_1, \ldots, \epsilon_n) \quad = \quad \{f(e_1, \ldots, e_n)^{\equiv} : e_1 \in \epsilon_1, \ldots, e_n \in \epsilon_n\}$$

for all $\epsilon_1, \ldots, \epsilon_n \in Pow(E)$

(iii) call $\equiv$ a *congruence relative to* f if

$$\frac{e_1 \equiv e'_1 \qquad \cdots \qquad e_n \equiv e'_n}{f(e_1, \ldots, e_n) \equiv f(e'_1, \ldots, e'_n)}$$

for all $e_1, \ldots, e_n, e'_1, \ldots, e'_n \in E$.

Fact 1. *An equivalence relation $\equiv$ on E is a congruence relative to $f : E^n \to E$ iff*

$$(*) \qquad f(e_1,\ldots,e_n)^{\equiv} \;=\; f^{\equiv}(e_1^{\equiv},\ldots,e_n^{\equiv}) \;\; \textit{for all } e_1,\ldots,e_n \in E.$$

Construing $e^{\equiv}$ to be the meaning of e, $(*)$ can be read as a restricted form of compositionality (C) for complex expressions $f(e_1,\ldots,e_n)$ with n parts $e_1 \ldots e_n$ and a total mode of composition f — provided we can make sense of the words "complex" and "parts" here.

But that is easy, if E is the set of terms built from function symbols f. Adopting Hodges' terminology (as modified by DW, and writing f instead of α), we assume that E comes with a grammar $(E, A, \underline{f})_{f\in\Sigma}$ where $A = \{f \in \Sigma : \text{arity}(f) = 0\}$ and for every $f \in \Sigma$ with arity n, $\underline{f}$ is the (total) n-ary function on E such that for all $e_1 \ldots e_n \in E$,

$$\underline{f}(e_1,\ldots,e_n) \quad = \quad \text{the term `} f(e_1,\ldots,e_n)\text{'} \; .$$

This trivial instance of Hodges' set-up dispenses with partiality, inviting the question

(‡) how are we to address the problem of extending a partial semantics μ if $E = GT =$ the set of terms built from a set Σ of function symbols?

The short answer is: by looking at equivalence classes on E.
For a more informative answer (developed in the remainder of this section), some notation is handy. Given a function $\mu : E \to M$ (from "expressions" to some set M of "meanings"), let $\equiv_\mu$ be the equivalence relation (of "synonymy") on E given by

$$e \equiv_\mu e' \qquad \text{iff} \qquad \mu(e) = \mu(e')$$

for all $e, e' \in E$.

Fact 2. *Fix a function* $\mu : E \to M$.

(a) Given $f : E^n \to E$, *there is a function* $r_f : M^n \to M$ *such that*

$$\mu(f(e_1,\ldots,e_n)) \;=\; r_f(\mu(e_1),\ldots,\mu(e_n)) \quad \textit{for all } e_1,\ldots,e_n \in E$$

iff $\equiv_\mu$ *is a congruence relative to* f.

(b) Let $\mu^\circ : E \to Pow(E)$ *map* $e \in E$ *to its* $\equiv_\mu$*-equivalence class* $\{e' : e \equiv_\mu e'\}$. *Then* $\equiv_\mu$ *is* $\equiv_{\mu^\circ}$.

2.3 A co-inductive characterization of compositionality

An equivalence relation on E is a *congruence relative to* a set Σ of functions on E (of various arities) if it is a congruence relative to every function in Σ. Given an equivalence relation $\equiv$ on E, let $\mathsf{Cng}[\equiv, \Sigma]$ be the set of congruences relative to Σ that are $\subseteq$-contained in $\equiv$

$$\mathsf{Cng}[\equiv, \Sigma] \quad = \quad \{R \subseteq \equiv : \; R \text{ is a congruence relative to } \Sigma\} \; .$$

What's so interesting about $\mathsf{Cng}[\equiv, \Sigma]$? To say $R \subseteq \equiv$ is to say R refines $\equiv$ — i.e., respects all distinctions made by $\equiv$. (Form contra-positive of implication expressing $R \subseteq \equiv$). But, of course, not all elements of $\mathsf{Cng}[\equiv, \Sigma]$ give interesting semantics. For instance, the $\subseteq$-least element of $\mathsf{Cng}[\equiv, \Sigma]$, $=$, does no work at all (corresponding to syntactic identity[7]). The purpose of this subsection is to show that $\mathsf{Cng}[\equiv, \Sigma]$ has a $\subseteq$-largest element, which we will relate in the next subsection to what Hodges calls *Fregean extensions*.

Towards that end, let us re-state the defining condition for an equivalence relation $\equiv$ on E to be a congruence relative to a *unary* function f on E as

$$\equiv \quad \subseteq \quad \equiv^f$$

where

$$\equiv^f \quad = \quad \{(e, e') \in \equiv : \ f(e) \equiv f(e')\} \ .$$

More generally,

Lemma 3. *Given $f : E \to E$ and $\equiv \subseteq E \times E$, let*

$$\begin{aligned} \equiv_0 \quad &= \quad \equiv \\ \equiv_{k+1} \quad &= \quad {\equiv_k}^f \qquad \text{for every integer } k \geq 0 \ . \end{aligned}$$

Then $\bigcap_{k \geq 0} \equiv_k$ is the $\subseteq$-largest element of $\mathsf{Cng}[\equiv, \{f\}]$.

Proof. Shortening $\bigcap_{k \geq 0} \equiv_k$ to $\equiv_\omega$, it is easy to verify that $\equiv_\omega$ is an equivalence class contained in $\equiv$, and

$$\begin{aligned} e \equiv_\omega e' \quad &\text{iff} \quad (\forall k)\ e \equiv_k e' \\ &\text{iff} \quad e \equiv_0 e' \text{ and } (\forall k)\ e \equiv_{k+1} e' \\ &\text{iff} \quad e \equiv e' \text{ and } (\forall k)\ f(e) \equiv_k f(e') \\ &\text{iff} \quad e \equiv e' \text{ and } f(e) \equiv_\omega f(e') \end{aligned}$$

for all $e, e' \in E$. Finally, to see that every element R of $\mathsf{Cng}[\equiv, \{f\}]$ is $\subseteq$-contained in $\equiv_\omega$, it suffices to show that $R \subseteq \equiv_k$ for every $k \geq 0$. This follows by a routine induction on k. ⊣

Next, what about a *binary* function f? We have two arguments to check, and the obvious modification of $\equiv^f$ (to get a congruence) is

$$\{(e, e') \in \equiv \ : \ (\forall (d, d') \in \equiv)\ f(e, d) \equiv f(e', d') \text{ and } f(d, e) \equiv f(d', e')\} \ .$$

The occurrence of '$(d, d') \in \equiv$' above is not positive, and we lose monotonicity (used to get fixed point from $\equiv_0 \supseteq \equiv_1 \supseteq \equiv_2 \cdots$ and cardinality argument). Fortunately, we can make do with $d' = d$ as follows.

Given an $(n+1)$-ary $f : E^{n+1} \to E$, let us break down the constraint for a congruence $\equiv$ relative to f

$$\frac{e_1 \equiv e_1' \quad e_2 \equiv e_2' \quad \cdots \quad e_{n+1} \equiv e_{n+1}'}{f(e_1, e_2, \ldots, e_{n+1}) \equiv f(e_1', e_2', \ldots, e_{n+1}')}$$

into $n+1$ parts, with the idea of working our way through the rule, one argument at a time.

[7] This is W. Zadrozny's celebrated result.

$$\cfrac{\cfrac{\cfrac{e_1 \equiv e'_1}{f(e_1,\ldots,e_{n+1}) \equiv f(e'_1,e_2,\ldots,e_{n+1})} \quad e_2 \equiv e'_2}{f(e_1,\ldots,e_{n+1}) \equiv f(e'_1,e'_2,e_3,\ldots,e_{n+1})} \quad e_3 \equiv e'_3 \quad \vdots \quad e_{n+1} \equiv e'_{n+1}}{f(e_1,\ldots,e_{n+1}) \equiv f(e'_1,\ldots,e'_{n+1})}$$

Accordingly, we form unary functions from f by freezing all but one of f's arguments as follows. For all $i \in \{1,\ldots,n+1\}$ and $\vec{e} \in E^n$, define the unary function $f_{i,\vec{e}} : E \to E$ on E by

$$f_{i,\vec{e}}(e) \quad = \quad f((e,\vec{e})_i)$$

where $(e,\vec{e})_i$ is $\vec{e}$ with e inserted at the ith position. For convenient reference, let us collect these unary projections of f in the set

$$\mathsf{u}(f) \quad = \quad \bigcup_{1\leq i\leq n+1} \{f_{i,\vec{e}} \ : \ \vec{e} \in E^n\} \ .$$

Now, we get an obvious analog of Hodges' notion of *1-compositional* (simplified a bit):

> an equivalence relation $\equiv$ on E is a 1-*congruence relative to* a set Σ of functions on E (of various arities) if for every $f \in \Sigma$, $\equiv$ is a congruence relative to every function in $\mathsf{u}(f)$.

Theorem 4. *Let $\equiv$ be an equivalence relation on E and Σ be a set of functions on E.*

(a) $\equiv$ is a congruence relative to Σ iff it is a 1-congruence relative to Σ.

(b) $\mathsf{Cng}[\equiv,\Sigma]$ has a $\subseteq$-largest element, namely $\bigcap_{k\geq 0} \equiv^{\Sigma}_{k}$ where

$$\begin{aligned} \equiv^{\Sigma}_{0} \quad &= \quad \equiv \\ \equiv^{\Sigma}_{k+1} \quad &= \quad \bigcap_{g\in\Sigma} \ \bigcap_{f\in\mathsf{u}(g)} (\equiv^{\Sigma}_{k})^{f} \end{aligned}$$

(with $\cdot^{f}$ as defined before Lemma 3).

Remark. It is natural to read $\bigcap_{k\geq 0} \equiv^{\Sigma}_{k}$ as the coarsest refinement of $\equiv$ respecting the contexts given by $\bigcup_{f\in\Sigma} \mathsf{u}(f)$.

2.4 Refining extensions

Given a set E and a function μ the domain of which is a subset of E, let $\equiv_{\mu,E}$ be the equivalence relation on E given by the 1-point extension $\mu_{\perp} \supseteq \mu$ on E defined by

$$\mu_{\perp}(e) \quad = \quad \begin{cases} \mu(e) & \text{if } e \in \mathit{dom}(\mu) \\ \perp & \text{otherwise} \end{cases}$$

where, by assumption, $\perp \notin image(\mu)$. That is, $\equiv_{\mu,E}$ is

$$\equiv_\mu \quad \cup \quad ((E - dom(\mu)) \times (E - dom(\mu))) \ .$$

(Notice that if $dom(\mu) = E$, then $\equiv_{\mu,E}$ is just $\equiv_\mu$.)

There is, of course, no guarantee that $\equiv_{\mu,E}$ is a congruence relative to a set Σ of (multi-ary) functions on E. But we know from the preceding subsection how to fix that: form the $\subseteq$-largest element of $\mathsf{Cng}[\equiv_{\mu,E}, \Sigma]$, call it $\equiv_\mu^\Sigma$, and define the *Fregean refinement of* μ *relative to* Σ to be the function $\mu^\Sigma : E \to Pow(E)$ mapping e to its $\equiv_\mu^\Sigma$-equivalence class $\{e' : e \equiv_\mu^\Sigma e'\}$.

Proposition 5. *Let E be the set of terms built from Σ, and μ be a partial function with domain $\subseteq E$. If μ has an extension to E that is Σ-compositional, then μ^Σ is (equivalent to) one (i.e. for all $e \in dom(\mu)$, $\mu^\Sigma(e) = \{e' : \mu(e) = \mu(e')\}$), and is what Hodges calls a Fregean extension of μ.*

Remark. $\equiv_\mu^\Sigma$ is *not* necessarily the coarsest Σ-congruence refining $\equiv_\mu$. Indeed, there isn't always a coarsest extension. Suppose μ were defined on exactly two expressions a and b with $\mu(a) \neq \mu(b)$. Given $c \notin \{a, b\}$, an extension of μ to $\{a, b, c\}$ that is maximally synonymous would make c synonymous with a or b but cannot do both. How do we choose? We don't, inventing instead new meanings for expressions outside $dom(\mu)$. Whether or not it is desirable that an extension of μ distinguish its domain from $dom(\mu)$ probably depends on the particular choice of μ — which is to say, claims that Fregean extensions are fully abstract ought probably to be evaluated on a case-by-case basis.

2.5 Applications: ambiguity and changing contexts

Recall our assumption in §2.2 that the grammar of E is $(E, \{f \in \Sigma : \text{arity}(f) = 0\}, \underline{f})_{f \in \Sigma}$ where for every $f \in \Sigma$ with arity n, $\underline{f}$ is the (total) n-ary function on E such that for all $e_1 \ldots e_n \in E$,

$$\underline{f}(e_1, \ldots, e_n) \quad = \quad \text{the term '}f(e_1, \ldots, e_n)\text{'} \ .$$

This assumption has the effect of establishing

(Tot) $\quad GT =$ the set of terms built from Σ

as well as $E = GT$. Question (‡) from §2.2 asked how we are to investigate partial semantics, given (Tot) and $E = GT$. Our longer answer to (‡) comes down to: weaken/generalize requirement of extension to refinement.

Now, for applications to ambiguity, it will pay to be a bit more careful about E, requiring of $\underline{f}$ only that it be a total n-ary function on E. This suffices to secure (Tot), without forcing $\mathsf{val} : GT \to E$ to be the identity function on GT (or even 1-1). That is, we stop short of eliminating the distinction between "parasitic" expressions and grammatical terms ($\supseteq dom(\mu)$), with a view to putting Hodges' framework to greater use. Given (Tot), what possible difference could $\mathsf{val} : GT \to E$ make? As far as semantics μ (total or partial) for

GT alone are concerned, none. But, coupling μ with val to produce meanings for $e \in E$ as in §1.2, define $\mu_{\mathsf{val}} : E \to Pow(M)$ by

$$\mu_{\mathsf{val}}(e) \quad = \quad \{\mu(t) \ : \ t \in dom(\mu) \text{ and } \mathsf{val}(t) = e\} \ .$$

(Assuming wlog that $E \cap GT = \emptyset$, we can extend μ_{val} to GT so as to contain μ.) The difference between μ_{val} and μ is genuine only if val is not 1-1 (allowing expressions in E to be ambiguous). The question is

just how illuminating a semantics of ambiguity is μ_{val}?

$\mu_{\mathsf{val}}(e)$ says nothing about disambiguation (and disambiguation is arguably crucial to any conception of ambiguity). Worse, $\mu_{\mathsf{val}}(e)$ quantifies away the structural analyses $t, \ldots$ linking e to μ.

Enter Frege's *Context principle*:

(X) *never ask what a word means in isolation, but only in the context of a proposition.*

It is natural to broaden (X) from "proposition" to *discourse*, interpreting context as co-text. As everyone knows, discourse can disambiguate; compare

```
A pal of every student in her class loves squash
```

(in isolation) with

```
Doris couldn't believe it.
A pal of every student in her class loves squash.
He is the only kid in town who eats it raw.
```

Taking E to be a set of discourse fragments, structured by a merge operation $\bullet$, we get the following ingredients for disambiguating E on the basis of some subset $\Phi \subseteq E$ with well-defined interpretations [i.e. $\Phi \approx GT$]

$$\begin{array}{rcll} \bullet & : & (E \times E) \to E & [\text{“merge”}] \\ \delta & : & E \to Pow(\Phi) & [\text{ compare to } \mathsf{val}^{-1} : E \to Pow(GT) \text{]} \\ [\![\cdot]\!] & : & \Phi \to M & [\text{ compare to } \mu : GT \to M \text{]} \end{array}$$

Question (concerning what sort of subset of E is Φ):

Does discourse E constitute (in Hodges' terminology) an *end extension* of sentences Φ? Or is Φ *cofinal* in E?

Argument for cofinal extension: a 2-sentence discourse can be turned into a single sentence, using "and" or perhaps "but", "because" ...

But is that really the case? Consider the following discourse, analyzed at length by Nicholas Asher and Alex Lascarides (e.g. *Linguistics and Philosophy*, 16(5):437-493, 1993).

```
John fell.  Mary pushed him.
```

The full stop (period '.') between `fell` and `Mary` can be disambiguated as "because" or "and." But without further information (as to which of the two [or more?] to choose), one might argue that '.' is simply underspecified. A further argument that discourse constitutes an end extension (with '.' mapping a sentence to a discourse) is provided by the following example, due to B. Partee.

> `Exactly one of the ten balls is not in the bag. It is under the sofa.`
>
> `Exactly nine of the ten balls are in the bag.` $^{?}$`It is under the sofa.`

A semantics for E (recognizing both arguments): take the largest element of $\mathsf{Cng}[\equiv_\nu, \{\bullet\}]$ where $\nu : E \to Pow(M)$ is given by

$$\nu(e) \quad = \quad \{[\![\varphi]\!] : \varphi \in \delta(e)\} \qquad [\text{ compare to } \mu_{\mathsf{val}}(e) \;] \;.$$

Fernando 1997 explores the largest element $\equiv_\nu^\Sigma$ of $\mathsf{Cng}[\equiv_\nu, \Sigma]$ where Σ might contain not only $\bullet$ but further multi-ary functions on E, invoking non-well-founded sets to characterize the resulting refinements of $\nu(e)$.

Two further twists

1. Ideally, one would derive $\equiv_\nu^\Sigma$ independently of the co-inductive construction in §2.3, appealing instead to some notion of meaning more explanatory than equivalence classes given by synonymy. That is, it is desirable to justify a "fully abstract" semantics "denotationally" — e.g. via semantic entities like first-order models (for which, in fact, the natural equivalence, isomorphism, may fall short of full abstraction, elementary equivalence).

   This is *not* to deny the importance of "operational semantics," concerning which an alternative non-deterministic analysis of the merge $\bullet$ above can be found in

   > Tim Fernando, Ambiguous discourse in a compositional context. *J. Logic, Language and Information*, 10(1):63-86, 2001.

   That paper couples operational semantics with a novel modal logic (constituting a "logical semantics"), and characterizes full abstractness in terms of *bisimulations*.

2. Context may well change during the interpretation of an expression — e.g. anaphora (where discourse can, say, resolve $\texttt{she}_?$ to $\texttt{she}_{\text{Mary}}$)

   Contrast 2 kinds of context:

   (i) context as bridge to $dom(\mu)$ — e.g. inverting val, given e

   (ii) context implicit in choice of [say] a grammar [instance] in Hodges' sense (including E)

   Whereas (i) is internal, (ii) is external.

   Neither is explicit in compositionality (C).

Synonymies leave nature of meaning open, which can be approached abstractly (denotationally/mathematically) or algorithmically (operationally/from a processing view).

Next time: incorporate notion of context into meaning, using more concrete and familiar semantic notions (with the basic theme: meaning as context change) .

3 Updating Montague

References (highly biased towards the last two below)

J. Groenendijk and M. Stokhof, Dynamic predicate logic. *Linguistics and Philosophy*, 14, 1991.

Irene Heim, *The Semantics of Definite and Indefinite Noun Phrases.* Dissertation, University of Massachusetts, Amherst, 1982. (Garland Press, NY, 1988.)

H. Kamp and U. Reyle, *From Discourse to Logic.* Kluwer, Dordrecht, 1993.

Lauri Karttunen, Presupposition and linguistic context. *Theoretical Linguistics*, pages 181–194, 1974.

Aarne Ranta, *Type-Theoretical Grammar.* Oxford Univ Press, Oxford, 1994.

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

Göran Sundholm, Proof theory and meaning. In D. Gabbay and F. Guenthner (eds.), *Handbook of Philosophical Logic*, vol 3. Reidel, Dordrecht, 1986.

Tim Fernando, A type reduction from proof-conditional to dynamic semantics. *J. Philosophical Logic* 30(2):121-153, 2001a.

Tim Fernando, Conservative generalized quantifiers and presupposition. *Semantics and Linguistic Theory XI*, New York, 2001b.

3.1 Context change for disambiguated expressions

This section is about *context* in a system of *dependent types* that can be

- interpreted model-theoretically
- extended with generalized quantifiers
- reduced to *Discourse Representation Theory* (DRT, Kamp and Reyle 1993)
- conceived *compositionally* by equating meaning with *context change potential* (Heim 1982).

Why bother? Linguistic interest in "dynamic semantics" can (to a large measure) be traced to the attention given in Karttunen 1974 to the notion that

$$\text{a context } \Gamma \ \underbrace{\text{satisfies the presuppositions of}}_{\textit{admits}, \ \rhd} \ \text{an expression } A.$$

To analyze '$\Gamma \rhd A$', Karttunen examines the effect of context change within and between sentences. Some pertinent examples are

(a) `Buganda has a king and the king of Buganda is bald.`

(b) `If Buganda has a king, the king of Buganda is bald.`

(c) `If Buganda has a king, he is bald.`

(d) `If a farmer owns a donkey, he beats his donkey.`

(e) `If a farmer owns a donkey, he beats it.`

Intuitively, the underlined descriptions in (a) and (b) can be satisfied by a context that is updated by the preceding clause `Buganda has a king`. Similarly for the underlined material in (c), (d) and (e), assuming we read the pronouns in accordance with the co-indexing below (ruling out other [e.g. deictic] readings).

(c)′ If Buganda has a^x king, he$_x$ is bald.

(d)′ If a^x farmer owns a^y donkey, he$_x$ beats [his$_x$ donkey]$_y$.

(e)′ If a^x farmer owns a^y donkey, he$_x$ beats it$_y$.

How exactly to pass from (c)-(e) to (c)′-(e)′ is the vexing problem of *anaphora resolution*. For (a) and (b), that passage is unambiguous (modulo the choice of the variable x, which is arguably inessential/un-interesting).

(a)′ Buganda has a^x king and the$_x$ king of Buganda is bald.

(b)′ If Buganda has a^x king, the$_x$ king of Buganda is bald.

Assuming primed representations (with determinate co-indexing), there remains the problem of checking that the appropriate descriptive conditions are met — e.g. x is, in (a)′, the king of Buganda, or, in (c)′, masculine. It is this problem on which we will focus, which is admittedly only a tiny part of the challenge of interpreting English.[8] In particular, the fundamental insight from Karttunen

[8]Whether that part is conceived as belonging to or coming after anaphora resolution need not concern us here. There is a good deal of leeway in how to draw the lines between anaphora and presupposition, although certain distinctions seem to be uncontroversially useful (e.g. that between anaphoric binding and accommodation).

The basic point, at any rate, is that the pressure (both theoretical and computational) to decompose natural language interpretation into different (albeit related) processes is irresistible. Integrating these processes leads, I think, to a refinement (as opposed to extension; see §2) of notions of meaning. A further complication is the need to view meaning as inputs for further processing — which complicates the conception of meaning as context change by changing the kind/notion of process/context involved. The notion of context studied in the present section (§3) must clearly be enriched, some steps towards which are reported in

> Tim Fernando, Three processes in natural language interpretation. In *Reflections: A Collection of Essays in Honor of Solomon Feferman*, W. Sieg, R. Sommer and C. Talcott (eds.), Association for Symbolic Logic, to appear.
>
> Tim Fernando, Towards a many-dimensional modal logic for semantic processing. Advances in Modal Logic 2000, CSLI Lecture Notes, to appear.

of interest to us here is that context build-up accounts for the following generalization of (a) and (b)

$$\text{in } A \wedge B \text{ and } A \supset B,\ \ A \textit{ filters } B\text{'s } \textit{presuppositions}. \tag{1}$$

Karttunen's account of (1) can be put in terms of Martin-Löf type theory, following Sundholm 1986 (and developed extensively in Ranta 1994), as

$$\frac{\Gamma \rhd A \qquad\qquad \Gamma, z\!:\!A \rhd B}{\Gamma \rhd (Qz\!:\!A)B} \tag{2}$$

for $Q = \Sigma, \Pi$. Σ and Π are dependent type constructs that, as will be explained shortly, apply uniformly to the propositional connectives $\wedge$ and $\supset$, and the quantifiers $\exists$ and $\forall$. Given this uniformity, (2) accounts for not only presupposition filtering in (1) but also truth equivalences of the following kind

`Some ants bite` *is true exactly if* `some ants are ants that bite.`

`All ants bite` *is true exactly if* `all ants are ants that bite.`

The truth equivalences above are widely assumed to hold for generalized quantifiers (which are said to be *conservative*). To turn (2) into an account of that, we must first understand how the the variable typing $z : A$ in (2) tracks the dependence of B on A, representing context build-up.

Note. *I am afraid the notes get (from here on) increasingly sketchy, and serve mainly to highlight points made at length in Fernando 2001a,b.*

3.2 Dependent types and the contexts on which they depend

Cartesian products and function spaces made dependent

$$(\Sigma x : A)B \;=\; \{\langle a,b\rangle \mid a\!:\!A \text{ and } b\!:\!B[x/a]\}$$

$$(\Sigma\_ : A)B = A \times B \qquad \text{(dummy variable \_)}$$

$$l\langle a,b\rangle = a \qquad r\langle a,b\rangle = b$$

$$(\Pi x : A)B \;=\; \{\text{functions mapping } a \text{ in } A \text{ to } b \text{ in } B[x/a]\}$$

$$(\Pi\_ : A)B = A \to B$$

Predicate logic from Σ, Π

	force Σ	Π
set	$\exists$	$\forall$
proposition	$\wedge$	$\supset$
restrictor A		

$$(\exists/\forall x \in A)B \;=\; (\Sigma/\Pi x\!:\!A)B$$

$$A \wedge/\!\supset B \;=\; (\Sigma/\Pi\_\!:\!A)B$$

Next, we relativize dependence to a notion of context.

A *context* Γ is a finite sequence of variable typings $x : A$.
A type (well-formed formula) may contain terms constructed from proofs in a context on which the type depend.
Moreover, context changes (from Γ to $\Gamma, x : A$) both locally and globally.

3.3 A type reduction: context as variable assignment

From Martin-Löf/Sundholm/Ranta... to Kamp/Heim/DPL ...

'and'/'.'	as	relational composition $\circ$ /;
'not'	as	divergence [complement of halting problem]

Equate '$\Gamma \rhd A$' with a formalization of '$\Gamma \Rightarrow A$ *prop*' relative to a signature.

DRSs as pretty-printed type expressions ... Fernando 2001a

3.4 Dependent quantifiers

$$\begin{array}{rcl} \text{some}(A, B) & \text{iff} & A \cap B \neq \emptyset \\ \text{all}(A, B) & \text{iff} & A \subseteq B \end{array}$$

Determiners D are *conservative* (Barwise & Cooper 1981, Keenan & Stavi 1986)

$$\text{D}(A, B) \quad \text{iff} \quad \text{D}(A, A \cap B)$$

A useful distinction: types A versus unary predicates B_u, from which we can form types $B_u(x)$ [$=B$] in a context that types x appropriately.

Sets: formulate the set of x's in A satisfying B as

$$\{x : A \mid B\} \quad = \quad l[(\Sigma x : A)B]$$

supporting relational characterizations

$$\begin{array}{lcl} (\Pi x : A)B & \text{iff} & A \subseteq \{x : A \mid B\} \\ (\Sigma x : A)B & \text{iff} & \underbrace{A \cap \{x : A \mid B\}}_{\{x : A \mid B\}} \neq \emptyset \end{array}$$

SELECTIVE QUANTIFICATION

```
Every boy with a nice suit will wear it tomorrow.
```

Collect tuples on which to quantify to the left, and apply relation Q_r to those

$$(Qx{:}A)B \quad \text{iff} \quad Q_r(l[A], l[\{x : A \mid B\}])$$

For external anaphoric links,

$$(Qx{:}A)B \quad = \quad \{(\Sigma x{:}A)B\} \times Q_r(l[A], l[\{x : A \mid B\}])$$

Dependent quantifiers are conservative not only in the sense of GQT (based on a set-theoretic conception of GQs), but also with respect to DRT-style syntax/semantics interface (from Kamp and Reyle 1993 on).

Moreover, the underlying proofs provide a type-theoretic approach to presupposition and eventualities. ... Fernando 2001b