1. **Hr∃, Th∃ & ∃v∀rywh∃r∃**

When the clock chimed, Miss Scarlet had already poked Colonel Mustard in the library with a pencil.

(*He screamed, but he was fine. No imaginary characters were harmed in producing this handout.*)

(1) a spy poked a soldier
(1a) $∃x∃y[\text{Spy}(x) & \text{Poked}(x, y) & \text{Soldier}(y)]$

(2) a spy poked a soldier in a library with a pencil
(2a) $∃e[∃x∃y[\text{Spy}(x) & \text{Poked}(e, x, y) & \text{Soldier}(y)] &$

$∃π[\text{In}(e, π) & \text{Library}(π)] & \existsπ[\text{With}(e, π) & \text{Pencil}(π)]}$

$\text{Poked}(e, x, y) \equiv \text{Past}(e) \& \text{PokeByOf}(e, x, y)$

(3) a soldier was poked
(3a) $∃e[\text{Past}(e) & \existsπ[\text{PokeOf}(e, π) & \text{Soldier}(π)]]$

(1) a spy poked a soldier
(1a') $∃e[\text{Past}(e) \& \existsπ[\text{By}(e, π) \& \text{Spy}(π)] \& \existsπ[\text{PokeOf}(e, π) \& \text{Soldier}(π)]]$

*It looks like we need at least this much “thematic decomposition”*

(4) a tailor saw a tinker with a tool
(4a) $∃e[\text{Past}(e) \& ∃x∃y[\text{Tailor}(x) \& \text{SeeOfBy}(e, x, y) \& \text{Tinker}(y) \& \existsπ[\text{With}(π, π) \& \text{Tool}(π)]]}$

#(4b) $∃e[\text{Past}(e) \& ∃x∃y[\text{Tailor}(x) \& \text{SeeOfBy}(e, x, y) \& \text{Tinker}(y) \& \existsπ[\text{With}(x, π) \& \text{Tool}(π)]]}$

(4c) $∃e[\text{Past}(e) \& ∃x∃y[\text{Tailor}(x) \& \text{SeeOfBy}(e, x, y) \& \text{Tinker}(y) \& \existsπ[\text{With}(e, π) \& \text{Tool}(π)]}$

*The tailor and tinker might also be magicians who perform a sawing trick.*

(4d) $\text{Past}(e) \& ∃x∃y[\text{Tailor}(x) \& \text{SawOfBy}(e, x, y) \& \text{Tinker}(y) \& \existsπ[\text{With}(π, π) \& \text{Tool}(π)]]$

#(4e) $\text{Past}(e) \& ∃x∃y[\text{Tailor}(x) \& \text{SawOfBy}(e, x, y) \& \text{Tinker}(y) \& \existsπ[\text{With}(x, π) \& \text{Tool}(π)]]$

(4f) $\text{Past}(e) \& ∃x∃y[\text{Tailor}(x) \& \text{SawOfBy}(e, x, y) \& \text{Tinker}(y) \& \existsπ[\text{With}(e, π) \& \text{Tool}(π)]]$

*If ‘see’ (‘saw’) has a variable position for the perceiver (sawyer), why can’t that position be modified?*

(4a') $∃e[\text{Past}(e) \& \existsπ[\text{By}(e, π) \& \text{Tailor}(π)] \&$

$\existsπ[\text{SeeOf}(e, π) \& \text{Tinker}(π) \& \existsπ'[\text{With}(π, π') \& \text{Tool}(π')]]$

(4c') $∃e[\text{Past}(e) \& \existsπ[\text{By}(e, π) \& \text{Tailor}(π)] \&$

$\existsπ[\text{SeeOf}(e, π) \& \text{Tinker}(π) \& \existsπ'[\text{With}(e, π') \& \text{Tool}(π')]]$

*Tense also turns out to be interestingly complicated…*

$\text{Past}(e) \equiv \existsπ[<(e, π) \& \text{SpeechTime}(π)]$

$\text{PastSimple}(e) \equiv \existsπ[<(e, π) \& \text{ReferenceTime}(π) \& \existsπ'[=(π, π') \& \text{SpeechTime}(π')]]$

$\text{PastPerfect}(e) \equiv \existsπ[<(e, π) \& \text{ReferenceTime}(π) \& \existsπ'[<(π, π') \& \text{SpeechTime}(π')]]$

Reichenbach 47

Hornstein 90
(5) a guest heard a scream in the hall

\[ \exists e \{ \text{PastSimple}(e) \land \exists \pi[\text{By}(e, \pi) \land \text{Guest}(\pi)] \land \]
\[ (a) \exists \pi[\text{HearingOf}(e, \pi) \land \text{Scream}(\pi) \land \text{In-the-hall}(\pi)] \]
\[ (b) \exists \pi[\text{HearingOf}(e, \pi) \land \text{Scream}(\pi) \land \text{In-the-hall}(\pi)] \]

Higginbotham 83

(6) a guest heard a soldier scream in the hall

\[ \exists e \{ \text{PastSimple}(e) \land \exists \pi[\text{By}(e, \pi) \land \text{Guest}(\pi)] \land \]
\[ (a) \exists \pi[\text{HearingOf}(e, \pi) \land \exists \pi'[\text{ScreamBy}(\pi, \pi') \land \text{Soldier}(\pi')] \land \text{In-the-hall}(\pi)] \]
\[ (b) \exists \pi[\text{HearingOf}(e, \pi) \land \exists \pi'[\text{ScreamBy}(\pi, \pi') \land \text{Soldier}(\pi')] \land \text{In-the-hall}(\pi)] \]

Vlach 83

(7) guests heard screams/guests scream/noises/noise after hearing doors slam

(7a) \[ \exists E \{ \text{PastSimple}(E) \land \]
\[ \exists \Pi[\text{By}(E, \Pi) \land \text{Guests}(\Pi)] \land \exists \Pi'[\text{HearingsOf}(E, \Pi) \land \text{Screams}(\Pi)] \land \]
\[ \exists F \{ \text{After}(E, F) \land \exists \Pi'[\text{HearingsOf}(F, \Pi) \land \exists \Pi'[\text{SlamsOf}(\Pi, \Pi') \land \text{Doors}(\Pi')]]} \}

‘\exists’ is not due to ‘a’

(8) water trickled

(8a) \[ \exists E \{ \text{PastSimple}(E) \land \exists \Pi[\text{TricklingOf}(E, ?) \land \text{Water}(\Pi)]} \]

(9) spy pokes soldier in library

(9a) \[ \exists e \{ \exists \pi[\text{By}(e, \pi) \land \text{Spy}(\pi)] \land \exists \pi[\text{PokeOf}(e, \pi) \land \text{Soldier}(\pi)] \land \exists \pi[\text{In}(e, \pi) \land \text{Library}(\pi)]} \]

And don’t forget article-free languages, or Kamp-Heim accounts of English indefinites. It may be that ‘a’ simply marks nouns as singular (+count, –plural).

‘a spy’

(10) \[ \exists e \{ \text{PastSimple}(e) \land \exists \pi[\text{By}(e, \pi) \land \text{Spy}(\pi)] \land \exists \pi[\text{PokeOf}(e, \pi) \land \text{Soldier}(\pi)] \land \]
\[ \exists \pi[\text{In}(e, \pi) \land \text{Library}(\pi)] \land \exists \pi[\text{With}(e, \pi) \land \text{Pencil}(\pi)]} \]

‘a library’

‘a pencil’

(11) brown cows (that are) in fields

(11a) \[ \text{Brown}(X) \land \text{Cows}(X) \land \exists \Pi[\text{In}(X, \Pi) \land \text{Fields}(\Pi)]} \]

at least often, ‘\exists’ is not due to ‘a’

(12) see brown beef on brown plates

(12a) \[ \exists \Pi[\text{SeeOf}(e, ?) \land \text{Brown}(?) \land \text{Beef}(?) \land \exists \Pi[\text{On}(?, \Pi) \land \text{Brown}(\Pi) \land \text{Plates}(\Pi)]} \]

familiar network of one-way implications

(Taylor 1983, citing Evans)

\[ \exists e[\Phi(e) \land \Psi(e) \land \Omega(e)] \rightarrow \exists e[\Phi(e) \land \Psi(e)] \rightarrow \exists e[\Phi(e)] \]
\[ \rightarrow \exists e[\Phi(e) \land \Omega(e)] \]

Two Common Patterns: \[ \Phi(\_) \land \Psi(\_) \] \[ \exists \Lambda(\_, \_) \land \Pi(\_) \]

entity/event participant
2. Who Ordered Those?

**Lexicalist Hypotheses:** many covert quantifiers and conjoiners

‘brown beef on brown plates’ $\Rightarrow$ (sm) [[brown (nd) beef] [on [(sm) [brown (nd) plates]]]]

‘guests heard guests scream’ $\Rightarrow$ (\exists) { [(sm) guests] [heard (\exists) { [(sm) guests] scream} ] }

**Type-Shifting Hypotheses:** there are no covert quantifiers and conjoiners, but it’s as if there are

$$\|\text{cow}\| = \lambda x. \text{T iff Cow}(x) \quad \quad \|\text{brown}\| = \lambda x. \text{Brown}(x)$$

$$\lambda x. \text{Cow}(x) \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \lambda e. \text{Cow}(e) \quad \text{&} \quad \lambda e. \text{Brown}(e)$$

$$\|\text{brown cow}\| = \lambda e. \|\|\text{brown}\|\| (\|\text{cow}\|) = \lambda e. \lambda x. \Phi(x) \& \text{Brown}(x)$$

$$\|\text{poked Mustard in Devon with pencils}\| = \lambda x. \text{PokedMustardInDevonWithPencils}(x)$$

**Combinatorial Hypotheses:** some modes of composition are logically substantive

$$\|\text{brown}_{<\text{et}>} \text{cow}_{<\text{et}>}\| = \lambda e. \|\|\text{brown}_{<\text{et}>}\| (e) \& \|\|\text{cow}_{<\text{et}>}\| (e)$$

$$= \lambda e. \lambda x. \text{Brown}(x)(e) \& \lambda x. \text{Cow}(x)(e) = \lambda e. \text{Brown}(e) \& \text{Cow}(e)$$

$$\|\|\text{poked Mustard}_{<\text{et}>} \text{[in Devon]}_{<\text{et}>}\| = \lambda e. \|\|\|\text{poked Mustard}_{<\text{et}>}\| (e) \& \|\|\|\text{in Devon]}_{<\text{et}>}\| (e)$$

$$= \lambda e. \text{PokeOf}(e, \text{Mustard}) \& \text{In}(e, \text{Devon})$$

*But once we grant that combination need not be logically innocent, it’s no big leap to allow for existential closure as a “default” clausal operation.*

*And if we go this far, we should at least consider a more revisionary hypothesis.*

---

- the common patterns reflect operations that are employed at an *early stage* of computation
- other aspects of meaning reflect other stages of computation (cp. Chomsky 57, Marr 82)
- not even verb-noun combination is logically innocent
- but maybe we can jettison (logically substantive) appeals to truth values and sets/functions

**Simple Typology:**

\[ \langle M \rangle \quad \langle D \rangle \]

\[ \langle M \rangle + \langle M \rangle \rightarrow \langle M \rangle \]

\[ \langle D \rangle + \langle M \rangle \rightarrow \langle M \rangle \]

\[ \Phi(\_)+\Psi(\_) \rightarrow \Phi(\_)^\Psi(\_) \]

\[ \Delta(\_,\_) + \Phi(\_) \rightarrow \exists[\Delta(\_,\_)^\Phi(\_)] \]

**Core Operations:**

joining two monadic concepts yields joining a dyadic concept with a monadic concept

joining a monadic concept with a monadic concept yields a monadic concept that applies to something

if and only if 

if and only if 

both of the joined concepts apply to __ 

participants

**Example:**

POKE-OF(\_,\_) + SOLDIER(\_) \rightarrow \exists[POKE-OF(\_,\_)^SOLDIER(\_)]

IN(\_,\_) + LIBRARY(\_) \rightarrow \exists[IN(\_,\_)^LIBRARY(\_)]

\[ \exists[POKE-OF(\_,\_)^SOLDIER(\_)] + \exists[IN(\_,\_)^LIBRARY(\_)] \rightarrow \exists[POKE-OF(\_,\_)^SOLDIER(\_)]^\exists[IN(\_,\_)^LIBRARY(\_)] \]

BY(\_,\_) + SPRY(\_) \rightarrow \exists[BY(\_,\_)^SPY(\_)]

PAST-SIMPLE(\_) + \exists[BY(\_,\_)^SPY(\_)] \rightarrow PAST-SIMPLE(\_)^\exists[BY(\_,\_)^SPY(\_)]

**Main Idea:** in the simplest case (M-junction), combination indicates restriction; in the next simplest case (D-junction), combination still involves restriction together with a kind of (variable-free) existential closure; this allows for atomic dyadic concepts, but the system only generates monadic concepts.

On this “neo-Medieval” view, not even verb-noun combination is logically innocent. By contrast, Fregean saturation adds no content.

But why expect natural modes of combination to be logically innocent?

It’s a very old idea that negating, disjoining, and conditionalizing are exceptions to a default principle that combining/lengthening is a way of strengthening.

It’s also a very old idea that universal quantification is a logically special case, while existential quantification is the default way of converting predicates into thoughts.

This invites a pair of “minimalist” questions...

—what would have to be added, beyond the Simple Typology and Core Operations, to accommodate cases of quantifier-noun combination as in ‘chased every cow’?

—would the net result be any better than more familiar views that invoke truth values and sets?
4. More Familiar Views and More Angst

(13) every cow is brown
(13a) $\forall x: \text{Cow}(x) [\text{Brown}(x)]$

each way of assigning a cow to the variable ‘$x$’
is a way of assigning a brown thing to ‘$x$’

(13b) $\{x: \text{Cow}(x)\} \subseteq \{x: \text{Brown}(x)\}$

$\exists \alpha [\alpha = \{x: \text{Cow}(x)\}]$

(13c) $\lambda \Psi . \lambda \Phi . T \equiv \{x: \Psi (x) = T\} \subseteq \{x: \Phi (x) = T\} (\lambda x. \text{Cow}(x)) (\lambda x. \text{Brown}(x))$

$\exists F \forall x: \text{Cow}(x) [F(x) = T]$

Function Application is not Saturation. Unsaturated slots are not variables, but they are unsaturated.

$\text{PRECEDES}(_{1}, _) + \exists [\Phi (_) ] \Rightarrow ???$

$<e, ct> + <ct, t> \Rightarrow ???$

$\exists x [\text{PRECEDES}(x', x)]$

$\exists x [\text{PRECEDES}(x, x')]$

$\lambda x . \lambda x'. T \equiv \text{PRECEDES}(x', x) + \text{OSCAR} \Rightarrow \lambda x'. T \equiv \text{PRECEDES}(x', \text{OSCAR})$

$\text{denoter } + \text{denoter} \Rightarrow \text{denoter}$

$\text{cp. PRECEDES}(\_, \text{OSCAR})$

$\text{unsaturated}$

Does (13)—a sentence of English—imply that the cows form a set?

Does the English sentence itself imply that if there are exactly 10 cows, there are at least 14 things:
the 10 cows, the set of those cows, two truth values, and at least one function of type $<e, t>$?

Does (13) imply that if there are exactly 10 cows, there are more than 14 things, including:
the number 10, the ten preceding natural numbers, endlessly many other numbers,
and all the corresponding functions of type $<\alpha, \beta>$? (See Appendix B)

Two ways of hearing “Does sentence S imply that p”?

(i) Do competent speakers comprehend S as a sentence whose truth logically guarantees that p?

Does S have a form that licenses a secure inference to the claim that p?

(ii) Is every world at which S is true a world at which it is the case that p?

Does S have a content that determines some (perhaps improper) subset of the p-worlds?

Does ‘Scarlet poked Mustard in the library with a pencil’ imply that there was a poking of Mustard?

(i) Yes. It has something to do with conjunct reduction. We should try figure out the details.

(ii) Yes. But the sentence also implies that there are infinitely many prime numbers.

Does ‘Sadie is a mare’ imply that Sadie is a horse? (Short form: does ‘mare’ imply ‘horse’?)

(i) Yes. It has something to do with conjunct reduction. We should try figure out the details.

(ii) Yes. But ‘mare’ also implies ‘mammal such that there are infinitely many prime numbers’.

Does ‘Some odd number precedes every prime number’ have two readings, with distinct implications?

(i) Yes. And speakers should reject the “surface” reading if they think that 1 is a prime number.

(ii) No. Even if there are two readings, they have the same implications.
5. Analogy: Proportional Angst

(14) Most of the dots are blue

(14a) \( \forall x: \text{Dot}(x)[\text{Blue}(x)] \)  
(14b) \( \#\{x: \text{Dot}(x) \& \text{Blue}(x)\} > \#\{x: \text{Dot}(x) \& \neg \text{Blue}(x)\} \)  

(14c) \text{OneToOnePlus}[\{x: \text{Dot}(x) \& \text{Blue}(x)\},\{x: \text{Dot}(x) \& \neg \text{Blue}(x)\}]  
(14d) \#\{x: \text{Dot}(x) \& \text{Blue}(x)\} > \#\{x: \text{Dot}(x)\} – \#\{x: \text{Dot}(x) \& \text{Blue}(x)\}  

Let’s agree that (14) is not understood in any “first order” way.

Is (14) understood as implying that the blue dots form a set?
Is (14) understood as implying that the dots that aren’t blue form a set?
Is (14) understood as implying that the dots and the dots that are blue have cardinalities?

According to Lidz et. al., (14) is understood as indicated with (14d), as opposed to (14c) or (14b). On this view, (14) implies that there are some numbers, but not that the non-blue dots form a set. This view may be wrong; the evidence that Lidz et. al. offer is not decisive. The point here is simply that the issue is empirical. Speakers understand expressions as they do. The goal—for cognitive scientists—is to figure out how; cp. Chomsky 1957. (We can talk about other projects in the Q&A.)

In (14d), the relevant numbers are represented as cardinalities of sets. But in the spirit of Lidz et. al., we can compare (14d) with (14e) and ask further empirical questions.

(14d) \#\{x: \text{Dot}(x) \& \text{Blue}(x)\} > \#\{x: \text{Dot}(x)\} – \#\{x: \text{Dot}(x) \& \text{Blue}(x)\} 
(14e) \exists Y \{ \forall x(Yx \equiv \text{Dot}(x)) \& \exists X[\forall x(Xx \equiv Yx \& \text{Blue}(x)) \& \#(X) > \#(Y) – \#(X)] \}

there are some things, the Ys, such that: each thing is one of them \( Y \) iff it is a dot; and there are some things, the Xs, such that each thing is one of them \( X \) iff it is one of the Ys and blue, and their number exceeds the result of subtracting their number from the number of the Ys

Read this way, (14e) implies that there are some dots and some numbers, but not that there are any sets. So we have to justify claims according to which (14) is understood as implying that there are some sets.

6. Bigger Variables, not Bigger Domains

There’s a way of interpreting the formalism in (14e) that makes (14e) different than (14d), at least given the usual way of interpreting the formalism in (14d). We’re not obliged to treat the upper-case variables in (14e) as ranging over sets or other “pluralities;” see Boolos (1998), Schein (1993), Pietroski (2005).

We don’t have to say that each assignment of values to variables assigns exactly one thing to each variable, and that special entities get assigned to upper-case variables. We can say instead that each assignment assigns one or more things to each variable, allowing for special cases: lower case (first-order) variables impose a constraint of singularity—i.e., the one or more values assigned are not more than one; upper case (second-order) variables are neutral; but “essentially plural” expressions, like ‘formed a trio’, require that their one or more values be more than one.
Assignments of values to variables can be depicted many ways, including the two shown below.

<table>
<thead>
<tr>
<th>abcd</th>
<th>0001</th>
<th>0010</th>
<th>0011</th>
</tr>
</thead>
<tbody>
<tr>
<td>abc</td>
<td>abd</td>
<td>acd</td>
<td>bcd</td>
</tr>
<tr>
<td></td>
<td>0100</td>
<td>0101</td>
<td>0110</td>
</tr>
<tr>
<td>ab</td>
<td>ac</td>
<td>ad</td>
<td>bd</td>
</tr>
<tr>
<td>1000</td>
<td>1001</td>
<td>1010</td>
<td>1011</td>
</tr>
<tr>
<td>a</td>
<td>b</td>
<td>c</td>
<td>d</td>
</tr>
<tr>
<td>1100</td>
<td>1101</td>
<td>1110</td>
<td>1111</td>
</tr>
</tbody>
</table>

The diagram on the left invites a “lattice” conception of assignments; see Cartwright 1965, Link 1983. The bottom row is for things in the basic domain. Every other lattice-point indicates an entity in an extended domain that includes sets or “sums” of basic entities—e.g., \{a, b, d\} or \ab\cd. The bottom row also corresponds to the “singletons” of the extended domain. We can say that each assignment assigns an entity in the extended domain to each second-order (capitalized) variable. But invoking more things is not mandatory. We can view ‘abd’ as an assignment of three (basic) entities to an unsingular variable. Recoding in binary makes this vivid: a = 1; b = 10; c = 100; d = 1000. Then ‘1011’ indicates for each entity, whether or not it is one of the one or more assigned values: d, yes; c, no; b, yes; a, yes.

Are the meanings of natural linguistic expressions better described in terms of extended domains or plural assignments? There is a difference, even if for many purposes, we can talk either way.

\[
(15) \, \exists x (\forall y \, ([x \equiv y]) \Rightarrow (x \not\equiv x))
\]

stipulated domain: Gott, Bert (and nothing else)

facts: Gott \not\equiv Gott; Bert \not\equiv Bert; Gott \not\equiv Bert

Given the stipulated domain and the set-theoretic construal, (15) is false.
But given the same domain and the Boolos construal, (15) is true.

While nothing in this domain includes Gott and Bert, there are some things—viz., Gott and Bert—such that each thing (in the domain) is one of them if and only if it isn’t selfelemental.

Now suppose that Gott = \emptyset, Bert = \{\emptyset\}, and the domain is extended to include all of the other pure Zermelo-Frankl sets (but nothing else).

Given the Boolos construal, (15) is still true, while (15) is still false on the set-theoretic construal.

If some domain makes a sentence true on interpretation I and false on interpretation I', then I \not\equiv I'.

Many facts suggest that the Boolos construal is distinctive and attractive for many purposes.

\[
(16) \, \exists x (Fx) \equiv \exists x (\forall y (Fx \equiv Fy))
\]

(trivial on the Boolos construal)

\[
(17) \, \forall x \exists y (x \equiv y)
\]

(fully general on the Boolos construal)

\[
(18) \, \sim \exists x (x \equiv x)
\]

\[
(19) \, \sim \exists y \exists z \exists \theta [\text{OneOf} (\theta, \emptyset) \equiv \sim \text{OneOf} (\theta, \emptyset)]
\]

\[
(20) \, \sim \exists s \exists x ([x \in s] \land \sim (x \in s))
\]

(trivial, but not fully general)

\[
(21) \, \sim \exists y \exists x ([x \in y] \land \sim (x \in y))
\]

\[
(22) \, \text{every set is grounded}
\]

\[
(22a) \, \{x : x \text{ is a set}\} \subseteq \{x : x \text{ is grounded}\}
\]

\[
(22b) \, \exists Y (\forall x (Yx \equiv \text{Set}(x)) \land \exists X [\forall x (Xx \equiv Yx \land \text{Grounded}(x)) \land \text{every } Y \text{ is an } X])
\]
(23) every set is a set
    seems **truistic**, even if you think that nothing includes every set

(24) Some barber shaves all and only the barbers who do not shave themselves.
    seems **false after thinking**: any such barber would be a self-shaver who is not a self-shaver

(25) Vulcan is smaller than Neptune
    **false, or at least a false presupposition**
(26) Vulcan is Vulcan
    **false, or at least a false presupposition**
(27) All sets are sets
    seems **trivial and true, even after thinking**
(28) All sets are grounded sets
    seems **at least plausible, even after thinking**

Positing implications that *can’t* be recognized is a Risky Game. Imagine a Montagovian field linguist on Planet Tarski, where sentences like (29) do not imply the existence of any sets or truth values.

(29) Fa

(29a) \( \lambda x. \top \iff \text{Fish}(x)(\text{Aristotle}) \)

Maybe (30) implies that Aristotle is *one* of the fish, but not that he is *an element of (the set of)* the fish.

(30) Aristotle is a fish

(30a) Aristotle \( \in \{ x: x \text{ is a fish} \} \)

**7. Compensating for Lost Innocence: Limited Quantification without Generalized Quantifiers**

(31) every cow ran
(32) every cow is a cow that ran

(33) \{ x: \text{Cow}(x) \} \subseteq \{ x: \text{Ran}(x) \}  
    **equivalent for ‘\( \subseteq \)’ but not for ‘\( \equiv \)’ or ‘=’**
(34) \{ x: \text{Cow}(x) \} \subseteq \{ x: \text{Cow}(x) \& \text{Ran}(x) \}

(35) \( \exists Y \{ \forall x (Yx \equiv \text{Cow}(x)) \& \exists X[\forall x (Xx \equiv Yx \& \text{Ran}(x)) \& \forall x : Yx(Xx)] \} \)
(14e) \( \exists Y \{ \forall x (Yx \equiv \text{Dot}(x)) \& \exists X[\forall x (Xx \equiv Yx \& \text{Blue}(x)) \& \{ \#(X) > \#(Y) – \#(X) \}] \} \)

(36) every cow which ran
    **OK as a restricted quantifier, but not as a sentence**
(37) *[e [ every cow ]OP [ which ran ]RC]*
(38) *[-> [every cow]<et, t> [which ran]<et>]*  
    **should be OK, or at least comprehensible, as a sentence**

(39) Finn chased every cow
(40) \( \exists Y \{ \forall x (Yx \equiv \text{Cow}(x)) \& \exists X[\forall x (Xx \equiv Yx \& \text{Chased(Finn, x)}) \& \forall x : Yx(Xx)] \} \)

Can (39) be understood *compositionally* in the way suggested by (40)?
And won’t any remotely plausible proposal take us far beyond the common patterns we started with?

\[
\begin{array}{c}
\Phi(\_)&\Psi(\_)
\end{array}
\]

\(\exists[\Delta(\_,\_)&\Pi(\_)]\)
First Step: Treat Sentences as Polarized Predicates

(41) Finn chased Bess
(42) \[ ||[s \text{ Finn chased Bess}] \|^3 = \top \text{ iff CHASED(FINN, BESS)} \]
(43) Val(_, [s Finn chased Bess])^3 iff CHASED(FINN, BESS)

Instead of saying that (41) denotes a truth value, we can say that (41) applies to everything or nothing, depending on whether or not Finn chased Bess. On this Tarskian view, if Finn chased Bess, then (41) applies to you, me, Finn, Bess, the number six, etc. (In general: if P, then we’re all such that P.) Similarly, we can say that relative to any particular assignment, (44) applies to everything or nothing.

(44) Finn chased it
(45) Val(_, [s Finn chased it])^3 iff CHASED(FINN, A[1])

In which case, relative to each assignment A, (44) applies to A[1]—and everything else—if and only if Finn chased A[1]. So we don’t need truth values, together with lambda abstraction, to accommodate relative clauses. Given (46), ‘which Finn chased’ applies to an entity if and only if Finn chased it.

(46) Val(_, [which [s Finn chased t]])^3 iff
  for some/the assignment A* such that =(_, A*[1]) & A* is otherwise just like A,
  Val(A*[1], [s Finn chased t])^3

When we’re not worrying about truth values or sets, we can replace (46) with (47).

(47) ||[which [s Finn chased t]]|^3 = \lambda x. \top \text{ iff CHASED(FINN, x)}

But (47) is no simpler than (46). Relative to any assignment A, ‘\lambda x. \top \text{ iff CHASED(FINN, x)}’ is shorthand for the following mouthful: the smallest function that maps each entity e to \top or \bot depending on whether or not ‘CHASED(FINN, x)’ is satisfied by the ‘x’-variant of A that assigns e to ‘x’

Though before trying to run without sets/functions, let’s be clear that we can walk without truth values, at least if we assume that quantifiers displace as in (48).

(48) [s [\text{ every}_{0} \text{ cow}_{1}]_{01} [s Finn chased t]]

And for these purposes, let’s not worry about how CHASED(FINN, A[1]) gets spelled out eventishly.

(49) \exists e \{S\text{IMPLE-PAST}(E) \& \text{CHASE}(E, FINN, A[1])\}

(49a) \exists e \{S\text{IMPLE-PAST}(E) \& \text{BY}(E, FINN) \& \text{CHASE-OF}(E, A[1])\}

(49b) \exists _{-} \{S\text{IMPLE-PAST}(_{-}) \exists [\text{BY}(_, _) = (\text{FINN}) \& \text{CHASE-OF}(_, _) = (\text{FINN})]\}

(49c) \text{\hat{\Phi}(\_)} is a polarized predicate that applies to everything or nothing, depending on whether or not \Phi(\_) applies to something.
1. $\text{Val}(\alpha, \beta)$, every$_O$ if $\alpha \supseteq \beta$  
   [axiom]

2. $\text{Val}(_, \text{cows$_O$})$ if $\text{COW}(_-)$  
   [axiom]

3. $\text{Val}(\alpha, [...O ...])$ if $\exists \beta[\text{Val}(\alpha, \beta, \ldots) \& \beta = \{x: \text{Val}(x, \ldots)\}]$  
   [axiom]

4. $\text{Val}(\alpha, [\text{every$_O$ cows$_O$}]_{[\ldots]})$ if $\alpha \supseteq \{x: \text{COW}(x)\}$  
   [1, 2, 3]

5. $\text{Val}(_, [s [...t ...]])^{[\ldots]}$ if 

   $\exists \alpha[\text{Val}(\alpha, [...])^{[\ldots]} \& \alpha = \{x: \exists A^*[A^*[1] = x \& A^* \approx A^* \& \text{Val}(A^*[1], [s ...t ...])^{[\ldots]}\}]$  
   [axiom, cp. 46]

6. $\text{Val}(_, [s \text{ Finn chased } t_1])^{[\ldots]}$ if CHASED(FINN, $A^*[1]$)  
   [Appendix A]

7. $\text{Val}(_, [s [\text{every$_O$ cows$_O$}]_{[t]} [s \text{ Finn chased } t_1]])^{[\ldots]}$ if 

   $\exists \alpha[\alpha \supseteq \{x: \text{COW}(x)\} \& \alpha = \{x: \exists A^*[A^*[1] = x \& A^* \approx A^* \& \text{CHASED}(\text{FINN}, A^*[1])\}]$  
   [4, 5, 6]

   cp. Larson & Segal (1995)

Second Step: Treat Quantifiers as Plural Predicates

Rewrite the axiom for ‘every’: $\text{Val}(O, every$_O$)$ if $\exists \forall \exists Y[\text{Externals}(O, X) \& \text{Internals}(O, Y) \& \forall x:Yx(Xx)]$

For any ordered pair $<e, i>$ — a.k.a. $\{e, \{e, i\}\}$ — $e$ is the pair’s external element. 

But we don’t have to say that the Os are pairs of sets that meet a certain set-theoretic condition. 

Let the Os be pairs of entities that meet a plural condition: each of their Internals is one of theirExternals.

$\forall X(O) \text{ iff } \exists \forall \exists Y(\forall x(Xx \equiv \exists Oo[\text{EXTERNAL}(o, x)]) \& \forall y(Yy \equiv \exists oO[\text{INTERNAL}(o, y)]) \& \forall x:Yx(Xx))$

$\exists \forall \exists Y[\text{EXTERNALS}(O, X) \& \text{INTERNALS}(O, Y) \& \forall x:Yx(Xx)]$

Now we can rewrite the derivation above without assuming an extended domain that includes a set of cows.

1. $\text{Val}(O, every$_O$)$ if EVERY(O)  
   [axiom]

2. $\text{Val}(_, \text{cows$_O$})$ if $\text{COW}(_-)$  
   [axiom]

3. $\text{Val}(O, [...O ...])$ if $\forall Y[\text{INTERNALS}(O, Y) \& \forall y(Yy \equiv \text{Val}(y, \ldots))^{[\ldots]}$  
   [axiom]

4. $\text{Val}(O, [\text{every$_O$ cows$_O$}]_{[\ldots]})$ if EVERY(O) & $\forall Y[\text{INTERNALS}(O, Y) \& \forall y(Yy \equiv \text{COW}(y))]$  
   [1, 2, 3]

4a. $\text{Val}(O, [\text{every$_O$ cows$_O$}]_{[\ldots]})$ if EVERY(O) & $\forall Y[\text{Cows}(Y)[\text{INTERNALS}(O, Y)]$  
   [4, abbreviated]

5. $\text{Val}(_, [s [...t ...]])^{[\ldots]}$ if $\exists \forall \{O(O, [...])^{[\ldots]} \& \exists X[\text{EXTERNALS}(O, X) \& \text{INTERNALS}(O, X) \& \forall x(Xx \equiv \exists A^*[A^*[1] = x \& A^* \approx A^* \& \text{Val}(A^*[1], [s ...t ...])^{[\ldots]}\}]$  
   [axiom, cp. 46]

6. $\text{Val}(_, [s \text{ Finn chased } t_1])^{[\ldots]}$ if CHASED(FINN, $A^*[1]$)  
   [Appendix A]

7. $\text{Val}(_, [s [\text{every$_O$ cows$_O$}]_{[t]} [s \text{ Finn chased } t_1]])^{[\ldots]}$ if 

   $\exists \forall \{\text{EVERY}(O) \& \exists Y[\text{Cows}(Y)[\text{INTERNALS}(O, Y)] \& \forall X(XX \equiv \exists A^*[A^*[1] = x \& A^* \approx A^* \& \text{CHASED}(\text{FINN}, A^*[1]))]\}$  
   [4, 5, 6]

   $\equiv \exists A^*[A^* \approx A^* \& \text{CHASED}(\text{FINN}, x)]$

   $\exists A^*[A^* \approx A^* \& \text{CHASED}(\text{FINN}, x)]$

7a. $\text{Val}(_, [s [\text{every$_O$ cows$_O$}]_{[t]} [s \text{ Finn chased } t_1]])^{[\ldots]}$ if 

   $\exists \forall \{\text{EVERY}(O) \& \exists Y[\text{Cows}(Y)[\text{INTERNALS}(O, Y)] \& \forall X[\text{CHASED}(\text{FINN}, X)[\text{EXTERNALS}(O, X)]\}$  
   [7, abb.]
But this still doesn’t capture the restricted/conservative character of quantificational determiners. The axiom for ‘every’ allows for ordered pairs such that some of their external elements

are not among their internal elements. (Finn may have chased many things that are not cows.) And the external/sentential argument of ‘every’ was treated as if it were the relative clause in (50).

(50) every cow which Finn chased

That’s almost as bad as appealing to quantifier raising and the idea that ‘every cow’ is of type <et, t>. But the goal is not to recode this idea, with all its warts, a little more austerely. The “minimalist” hope is that aiming for austerity will help identify which aspects of our notation do the explanatory work.

We want to know why quantificational determiners “live on” their internal arguments;


With regard to (48), we want to explain the semantic asymmetry between cowN and [s Finn chased t1].

(48) [s [everyQ cowN]01 [s Finn chased t1]]

So if the displaced quantifier recombines with the sentence from which it was displaced, maybe we don’t want a semantics that erases this grammatical asymmetry as in (51); cp. Heim & Kratzer (1998).

(51) [<t> [every<et, et, t> cow<et>]<et, t> [<et> 1 [<t> Finn chased t1]]]

Maybe we should return to (40)—a claim about the cows, with no reference to the things Finn chased…

(40) ∃Y {∀x(Yx ≡ Cow(x)) & ∃X[∀x(Xx ≡ Yx & Chased(Finn, x)) & ∀x:Yx(Xx)]}

(40a) tY:Cows(Y){∃X[∀x(Xx ≡ Yx & Chased(Finn, x)) & ∀x:Yx(Xx)]}

… and no reference to any relation exhibited by the (set of) cows and the (set of) things Finn chased. So let me end with two suggestions—perhaps notational variants—about how to get from (48) to (40).

(52) Val(O, […0 …N]0)A iff

Val(O, …0)A & ∃Y[Internals(O, Y) & ∀y(Yy ≡ Val(y, …N)A) & ExternalsAreInternals(O)]

(53) Val(_, [s […0 …N]0]1, [s …l ……]), A) iff

∃O {Val(O, […]0)A & ∃X[Externals(O, X) &

∀x(Xx ≡ ∃A*x:x = A*[i] & Val(A*[i], …N)A & A* ≈ A & {Val(A*[1], […]A*})]})

We can deny that the Os pair their internal entities with independently selected external entities.

We need not (and should not) say that quantificational determiners express second-order relations. The external/sentential argument—a polarized predicate containing a trace of the displaced quantifier—is used to make a secondary selection from values of the internal/nominal argument. On this view, the combinatorics ensures conservativity. So while identity is not a conservative second-order relation, we can still specify the meaning of ‘every’ with an identity condition, as opposed to an inclusion condition.

Val(O, everyQ)A iff ∃Y∃X[Internals(O, Y) & Externals(O, X) & ∀x(Yx ≡ Xx)]

∃Y[Internals(O, Y) & Externals(O, Y)]

tY:Internals(O, Y)[Externals(O, Y)]
Appendix A: Comparing Derivations for ‘Finn chased it’

**Axioms**

a. Val(_, -d_T)^{\alpha} iff PAST-SIMPLE(_)

b. Val(_, \Phi-Finn_N)^{\alpha} iff (=(_, R-FINN)

Proper nouns are probably predicative, and they’re surely not atomic expressions of type <e>.

c. for any index \(i\), Val(_, t)^{\alpha} iff (=(_, \mathcal{A}[i])

d. for any index \(i\), ||t||^{\alpha} = \mathcal{A}[i]

e. Val(_, [chase_{\mathcal{V}} \ldots])^{\alpha} iff \exists[CHASE-OF(_, _)^{\alpha}Val(_, \ldots)^{\alpha}]

f. ||[\ldots e_{\mathcal{V}} \ldots e_{\mathcal{V}}]|^{\alpha} = \lambda.x.\lambda.e.T iff CHASE-OF(e, x)

g. ||[\ldots e_{\mathcal{V}} \ldots e_{\mathcal{V}}]|^{\alpha} = \lambda.x.\lambda.e.T iff

h. Val(_, [\Sigma \ldots])^{\alpha} iff for some e, Val(e, \ldots)^{\alpha}
Note that blaming tense for the matrix $\exists$-closure
would assign two kinds of work to one morpheme: quantification and restriction.

$$\text{[she poke him]} \rightarrow \text{She-poke-him(e)}$$
$$\text{-ed} \rightarrow \lambda \Phi. \exists e \{ \text{PastSimple}(e) \land \Phi(e) \}$$
$$\text{[-ed [she poke him]]} \rightarrow \exists e \{ \text{PastSimple}(e) \land \text{She-poke-him(e)} \}$$

Moreover, the restrictors are not mere conjuncts; see Reichenbach/Hornstein.

$$\text{PastSimple}(e) \equiv \exists \pi [\text{Before}(e, \pi) \land \text{ReferenceTime}(\pi) \land \exists \pi'[=(\pi, \pi') \land \text{SpeechTime}(\pi')]]$$

And perceptual reports (e.g., heard him scream’) suggest that $\exists$-closure doesn’t require embedded tense; while we can apparently get embedded tense without existential closure.

(i) she poked him before he screamed
$$\exists e \{ \text{She-poked-him(e)} \land \forall \pi: \text{PastSimple}(\pi) \land \text{ScreamBy}(\pi, \text{he}) [\text{Before}(e, \pi)] \}$$

(ii) she poked him and then he screamed
$$\exists e \{ \text{She-poked-him(e)} \land \exists \pi: \text{PastSimple}(\pi) \land \text{ScreamBy}(\pi, \text{he}) [\text{Before}(e, \pi)] \}$$

Appendix B: The Frege-Church Type Hierarchy

Level Zero: two basic types $<e>$ $<t>$

**Recursive Principle:** if $<\alpha>$ and $<\beta>$ are types, then $<\alpha, \beta>$ is a type

Level One: four $<0, 0>$ types $<e, e>$ $<e, t>$ $<t, e>$ $<t, t>$

Level Two: eight $<0, 1>$ types, which include $<e, <e, t>>$ and $<t, <t, e>>$; eight $<1, 0>$ types, which include $<<e, e>, e>$ and $<<t, t>, t>$; and sixteen $<1, 1>$ types, which include $<<e, e>, <e, e>>$ and $<<e, t>, <t, t>>$

Level Three: sixty-four $<0, 2>$ types, which include $<e, <e, <e, t>>>$; sixty-four $<2, 0>$ types, which include $<<e, <e, t>>, t>$; one-hundred-and-twenty-eight $<1, 2>$ types, which include $<<e, t>, <<e, t>, t>>$; one-hundred-and-twenty-eight $<2, 1>$ types, which include $<<e, <e, t>>, <e, t>>$; and one-thousand-and-twenty-four $<2, 2>$ types, which include $<<e, <e, t>>, <e, <e, t>>>$.

Level Four: $5632$ $<0, 3>$ or $<3, 0>$ types, including $<e, <e, <e, e, t>>>$ $11,264$ $<1, 3>$ or $<3, 1>$ types; $90,112$ $<2, 3>$ or $<3, 2>$ types, including $<<e, <e, t>>, <<e, <e, t>>, t>>$; and $1,982,464$ $<3, 3>$ types

Level Five: more than $5 \times 10^{12}$ types

*etc.*
Appendix C: Composition as De-Abstraction...Bait and Switch

(1) Finn chased Bess
(1a) [Finn<e> [chased<e, e>] Bess<e>]<e>]

What about tense and adverbial modifiers?

(1b) [... [Finn<e> [chase<e, e>] Bess<e>]<e>]<e> [...]<e>

What about passives and other motivations for “severing” external arguments?

(1c) [... [Finn<e> [<e, e>] [chase<e, e>] Bess<e>]<e>]<e> [...]<e>

Are there any simple cases that motivate the standard typology, in the way that (1) was supposed to?

(2) chase Bess
(2a) [chase<e, e> Bess<e>]<e>

Are names atomic expressions of type <e>? And is (3) as complicated as (3a)? Or is this just a game?

(3) chase cows
(3a) [[(sm)<e> [cow<e> s<e, e>]<e>]<e> [<e, e>] 1 [... [chase<e, e> t1<e>]<e> [...]<e>]<e>]<e>

References (Very Incomplete...for more, see the list in Pietroski 2018)


*Natural Language Semantics* 19:227-56.


