1 Remarks on type-shifting in general

- Some people are suspicious of type-shifters. Sometimes, it is quite sensible to feel this way; people have proposed type-shifters that do rather remarkable things to their arguments.

- But many type-shifters make excellent logical sense in certain settings — they can even be entailments of one’s basic assumptions (van Benthem 1991; Winter 2002; Asudeh 2004). So it is unwise to accept or reject type-shifting across the board. We should decide case-by-case.

- Type-shifters often do the work of silent lexical items or LF movement operations. It can be illuminating to find these correspondences. It also helps to focus attention on the underlying logical or empirical issues.

2 Total functions

2.1 Type raising

(1) \text{ident} \sim \lambda x \lambda y (x = y)

(2) lift \sim \lambda x \lambda f (f \ x)

(3) \text{THE} \sim \lambda f \lambda g \exists x (\forall y ((f \ y) \leftrightarrow (x = y)) \land (g \ x))

(4) \text{a} \sim \lambda f \lambda g \exists x ((f \ x) \land (g \ x))

2.2 Type lowering

(5) \text{BE} \sim \lambda P \lambda x (P \ (\lambda y \ (y = x)))

3 Partial functions (both lowering operations)

(6) \|\text{iota}\|^M = \text{the function } F \in D_{\langle e,t \rangle,e} \text{ such that for all } f \in D_{\langle e,t \rangle} \text{, } (e, t) \text{ to } e

- \text{F(f) is defined iff there is exactly one } d \in D_e \text{ such that } f(d) = T
- \text{where defined } F(f) \text{ is the unique } d \in D_e \text{ such that } f(d) = T

(7) \|\text{lower}\|^M = \text{the function } L \text{ such that for all } P \in D_{\langle e,t \rangle,t} \text{, } \langle (e, t), t \rangle \text{ to } e

- \text{L(P) is defined iff there is exactly one } d \in D_e \text{ such that for all } f \in D_{\langle e,t \rangle}, P(f) \text{ iff } f(d)
- \text{where defined, } L(P) \text{ is the unique } d \in D_e \text{ such that } P(f) \text{ iff } f(d)
4 Summary diagram

*outputs the empty set of sets if the input is not a singleton

† outputs the empty set if the input has no singletons in its domain
5 Notes

• Some things can move back and forth between the \(e, t\) and \(\langle e, t \rangle, t\) domains without ever being able to get to the entity domain.
• Entities can move everywhere freely. (The diagram commutes for them.)
• We have a variety of equivalences. For instance:
  i. \(\text{lower}(\text{lift}(\text{bart})) = \text{bart}\)
  ii. \(\text{iota}(\text{ident}(\text{bart})) = \text{bart}\)
  iii. \(\text{BE}(\text{A(dog)}) = \text{dog}\)
  iv. \(\text{BE}(\text{no(man)}) = \text{not(man)}\)
• \(\text{THE}\) delivers \textit{falsity} for any second argument if its first argument is not a singleton (because \(\exists x (\forall y ((f \ y) \leftrightarrow (x = y)))\) is false in such situations). It contrasts on this point with \(\text{iota}\), which is \textit{undefined} in such situations.
• \(\text{BE}\) delivers the empty set wherever its first argument has no singletons in its domain. (It essentially gives the input quantifier a singleton, of which only singletons and the emptyset have a chance of being a subset — think \(X \subseteq Y\), where \(Y\) is the nuclear scope argument.)
• \(\text{BE}\) is not a good meaning for copular verbs. A likely denotation (at least upon simple inspection) is \(\lambda f f\) (which just allows the predicate to apply to the subject). If we express this (equivalently) with \(\lambda f \lambda x (f \ x)\), then we can see the possibility of switching the two arguments around, which might be useful for specificational copular sentences like \(\text{The winner is Susan}\).
• Partee discusses nominalizing and predicating functions. The first maps properties to their kind-level correlates. The second maps kinds to their predicate-level correlates. We’ll discuss these if we explore the entity domain later in the semester.

6 Hypotheses to explore

i. The NP argument in \textit{there be NP} can be filled only by nominals that can get into the \(e, t\) domain with sensible denotations (McNally 1998).

ii. The second constituent in small clauses might be fillable by all and only the nominals that have sensible \(e, t\) denotations (Partee 1987).

iii. In specificational copular clauses (\textit{The winner is Susan}), the subject definite description has shifted by \textit{ident}. Its property-level denotation explains why we use \textit{it} in tag-questions (\textit{The winner is Susan, isn’t it/*she}), among other things (Mikkelsen 2004).

iv. Natural language anaphora might be restricted to things that can make it into the \(e\) domain (Landman 2005).
Ling 230b, Stanford (Potts)

References


