Appendix. Comparing Five Theories of Presupposition Projection

In this Appendix, we define one base language (called *L*) and study five accounts of presupposition projection in that language: dynamic semantics, the Transparency theory, our version of local satisfaction, a supervaluationist alternative, and a Strong Kleene alternative.

1. **Syntax of L**
   -Generalized Quantifiers: Q ::= Q_i
   -Predicates: P ::= P_i | P_iP_k
   -Propositions: **p ::= p_i | p_ip_k**
   -Formulas **F ::= p | (not F) | (F and F) | (F or F) | (if F. F)** | (Q_i P . P)
   To state some of our principles, the official object language is enriched with:
   (a) predicate conjunction: if *P* and *P'* are predicates, so is *(P and P')*.
   (b) restrictions of predicative and propositional types:
   if *c'* is a predicative context variable and if *P* is a predicate, ${}^{c'}P$ is a predicative expression*;*
   if *c'* is a propositional context variable and if *F* is a formula, ${}^{c'}F$ is a formula.
   Terminology: The 'propositional fragment' of L is the language defined by the expressions in bold

We start by defining a classical semantics, which we call *I*. On a technical level, we assume that:
-each propositional letter is assigned by I a function of type $<s, t>$
-each predicate is letter is assigned by I a function of type $<s, <e, t>>$
-each generalized quantifier Q_i corresponds to a 'tree of numbers' f_i, which associates a truth value to each pair of the form *(a, b)* with *a* = the number of elements that satisfy the restrictor but not the nuclear scope and *b* = the number of elements that satisfy both the restrictor and the nuclear scope.
Logical constants are given a syncategorematic semantics.
Instead of writing I(F)(w) = 1, we sometimes use the notation w |= F. We will also abbreviate I(F)(w) as $[\![F]\!]^w$ or even as F^w . When we use an extended language to state some of our principles, we will sometimes need assignment functions, and we will write w $|=^s$ F, I_s(F)(w) or $[\![F]\!]^{w,s}$ to indicate relativization of the relevant notions to the assignment function s.
When certain elements are optional, we place angle brackets (< >) around them and around the corresponding part of the semantic rules.

2. **Classical Semantics** (called *I* in what follows)
   w |= p iff $p^w = 1$
   w |= $\underline{p}p'$ iff $p^w = p'^w = 1$
   w |= (not F) iff w |≠ F
   w |= (F and G) iff w |= F and w |= G
   w |= (F or G) iff w |= F or w |= G
   w |= (if F. G) iff w |≠ F or w |= G
   w |= $(Q_i\underline{<P>}P'.\underline{<Q>}Q')$ iff $f_i(a_w, b_w)=1$ with $a_w=\{d\in D$: $<P^w(d) = 1$ and$>$ $P'^w(d) = 1$ and ($<Q^w(d) = 0$ or$>$ $Q'^w(d) = 0)\}$, $b_w=\{d\in D$: $<P^w(d) = 1$ and$>$ $P'^w(d) = 1$ and $<Q^w(d) = 1$ and$>$ $Q'^w(d) = 1\}$
   Additions (a) and (b) in 1 can be interpreted thanks to 5b below.

For reasons of simplicity, we will assume throughout that L is *extremely* expressive: any proposition or property can be expressed by an atomic expression:

3. **Expressivity**
   Every proposition and every property is denoted by some atomic expression of L.

We repeat from the text our definitions of generalized entailment and generalized conjunction; they are intended for much richer type-theoretic languages, but are applicable in the present framework.

4. **Generalized Entailment**
   a. If x and x' are two objects of a type τ that 'ends in t', and can take at most n arguments, x ≤ x' just in case whenever $y_1, ..., y_n$ are objects of the appropriate type, if $x(y_1) ... (y_n) = 1$, then $x'(y_1) ... (y_n) = 1$
   b. If E and E' are two expressions of a type τ that 'ends in t',
   w $|=^s$ (E ≤ E') iff $[\![E]\!]^{w,s} \leq [\![E']\!]^{w,s}$

5. **Generalized Conjunction**
   a. If x and x' are two objects of a type τ that 'ends in t', and can take at most n arguments of types τ_1 , ..., τ_n

respectively, then
$x \wedge x' = \lambda y_{1\tau_1} \quad \lambda y_{n\tau_n} \quad x(y_1) \ldots (y_n) = x'(y_1) \ldots (y_n) = 1$
b. If E and E' are two expressions of a type τ that 'ends in t',
$[\![{}^{E'}E]\!]^{w,s} = [\![(E' \text{ and } E)]\!]^{w,s} = [\![E']\!]^{w,s} \wedge [\![E']\!]^{w,s}$

We define on the basis of 2 a dynamic semantics which is in full agreement with Heim's analysis (Heim 1983), except for disjunction, which she does not discuss; here we follow Beaver 2001.

6. **Dynamic Semantics**
$C[p] = \{w \in C: p^w = 1\}$
$C[\underline{p}p'] = \#$ iff for some $w \in C$, $p^w = 0$; if $\neq \#$, $C[\underline{p}p'] = \{w \in C: p'^w = 1\}$
$C[(\text{not } F)] = \#$ iff $C[F] = \#$; if $\neq \#$, $C[(\text{not } F)] = C - C[F]$
$C[(F \text{ and } G)] = \#$ iff $C[F] = \#$ or ($C[F] \neq \#$ and $C[F][G] = \#$); if $\neq \#$, $C[(F \text{ and } G)] = C[F][G]$
$C[(F \text{ or } G)] = \#$ iff $C[F] = \#$ or ($C[F] \neq \#$ and $C[\text{not } F][G] = \#$); if $\neq \#$, $C[(F \text{ or } G)] = C[F] \cup C[\text{not } F][G]$
$C[(\text{if } F.\ G)] = \#$ iff $C[F] = \#$ or ($C[F] \neq \#$ and $C[F][G] = \#$); if $\neq \#$, $C[(\text{if } F.G)] = C - C[F][\text{not } G]$
$C[(Q_i \langle\underline{P}\rangle P'.\langle\underline{R}\rangle R')] = \#$ iff $\langle$for some $w \in C$, for some $d \in D$, $\underline{P}^w(d) = 0\rangle$ or $\langle$for some $w \in C$, for some $d \in D$, $\langle\underline{P}^w(d) = 1$ and$\rangle$ $P'^w(d) = 1$ and $\underline{R}^w(d) = 0\rangle$. If $\neq \#$, $C[(Q_i\langle\underline{P}\rangle P'.\langle\underline{R}\rangle R')] = \{w \in C: f_i(a^w, b^w) = 1\}$ with $a^w = \{d \in D: P'^w(d) = 1 \text{ and } R'^w(d) = 0\}$, $b^w = \{d \in D: P'^w(d) = 1 \text{ and } R'^w(d) = 1\}$

The Transparency theory is based on the classical semantics in 2; it comes in an incremental version and in a symmetric version.

7. **Transparency: Principles**
a. Be Articulate
In any syntactic environment, express the meaning of an expression *<u>d</u>d'* as *(d and <u>d</u>d')*
b. Be Brief - Incremental Version
Given a context set C, a predicative or propositional occurrence of *d* is infelicitous in a sentence that begins with *α (d and* if for any expression *γ* of the same type as *d* and for any good final *β*, $C \models \alpha\ (d \text{ and } \gamma)\ \beta \Leftrightarrow \alpha\ \gamma\ \beta$.
c. Be Brief - Symmetric Version
Given a context set C, a predicative or propositional occurrence of *d* is (somewhat) infelicitous in a sentence of the form *α (d and d') β* if for any expression *γ* of the same type as *d*, $C \models \alpha\ (d \text{ and } \gamma)\ \beta \Leftrightarrow \alpha\ \gamma\ \beta$.
d. Ordering of Principles
Be Brief [in either version] >> Be Articulate

8. **Transparency: Derived Notions**
a. Incremental Transparency = Be Articulate + Incremental Version of *Be Brief*
Let C be a context set and *F* be a formula. *F satisfies Incremental Transparency relative to C* (abbreviation: *$Transp^i(C, F)$*) just in case for any presuppositional expression *<u>d</u>d'*, for any strings *α* and *β*, if $F = \alpha\ \underline{d}d'\ \beta$, then for any constituent *γ* of the same type as *d* and for any good final *β'*,
$C \models \alpha\ (d \text{ and } \gamma)\ \beta' \Leftrightarrow \alpha\ \gamma\ \beta'$
b. Symmetric Transparency = Be Articulate + Symmetric Version of *Be Brief*
Let C be a context set and *F* be a formula. *F satisfies Symmetric Transparency relative to C* (abbreviation: *$Transp^s(C, F)$*) just in case for any presuppositional expression *<u>d</u>d'*, for any strings *α* and *β*, if $F = \alpha\ \underline{d}d'\ \beta$, then for any constituent *γ* of the same type as *d*,
$C \models \alpha\ (d \text{ and } \gamma)\ \beta \Leftrightarrow \alpha\ \gamma\ \beta$

In a broad range of cases, the incremental version of the Transparency theory is equivalent to standard dynamic semantics.

9. **Incremental Transparency vs. Standard Dynamic Semantics** (from Schlenker 2007a)
a. Non-Triviality
Let $C \subseteq W$ be a context set and let *F* be a formula. $\langle C, F\rangle$ satisfies **Non-Triviality** just in case for any initial string of *F* of the form *α A*, where *A* is a quantificational clause (i.e. a formula of the form *(Q_i G. H)*), there is a good final *β* such that:
$C \not\models \alpha\ A\ \beta \Leftrightarrow \alpha\ T\ \beta$
$C \not\models \alpha\ A\ \beta \Leftrightarrow \alpha\ F\ \beta$
where *T* is a tautology and *F* is a contradiction.
b. Constancy

Let C be a context set and *F* be a formula. <C, F> satisfies **Constancy** just in case (i) the (unique) domain of individuals is of constant finite size over C, and (ii) the extension of each restrictor that appears in *F* is of constant size over C.

c. Theorem 1: Consider the propositional fragment of L. Let $C \subseteq W$ be a context set and let *F* be a formula. Then:
(i) $Transp^i(C, F)$ iff $C[F] \neq \#$.
(ii) If $C[F] \neq \#$, $C[F] = \{w \in C: w \models F\}$.

d. Theorem 2 [here we only state a *consequence* of Theorem 2 from Schlenker 2007a]: Let $C \subseteq W$ be a context set and let *F* be a formula of L. Suppose that <C, F> satisfies Non-Triviality and Constancy. Then:
(i) $Transp^i(C, F)$ iff $C[F] \neq \#$.
(ii) If $C[F] \neq \#$, $C[F] = \{w \in C: w \models F\}$.

We now turn to the definition of transparent restrictions, of local contexts and of local satisfaction.

10. **Transparent Restrictions**

Let $C \subseteq W$ be a context set and let *a d b* be a formula, where *d* has a type that 'ends in t'; let *c'* be a variable of the same type as *d*.

a. $tr^i(C, d, a\_b)$ = {x: x is an object of the type specified by *d* and for every constituent *d'* of the same type as *d*, for every good final *b'*, $C \models^{c' \to x}$ a $^{c'}$d' b' $\Leftrightarrow$ a d' b'}

b. $tr^s(C, d, a\_b)$ = {x: x is an object of the type specified by *d* and for every constituent *d'* of the same type as *d*, $C \models^{c' \to x}$ a $^{c'}$d' b $\Leftrightarrow$ a d' b}

11. **Local Contexts**

a. $lc^i(C, d, a\_b)$ = the bottom element of $tr^i(C, d, a\_b)$, if such an element exists; # otherwise.
b. $lc^s(C, d, a\_b)$ = the bottom element of $tr^s(C, d, a\_b)$, if such an element exists; # otherwise.

12-16 are concerned with the existence of local contexts.

12. **Lemma 1. Propositional Fragment.**

For any $C \subseteq W$, for any formula *E*, if *a E b* is a formula of the propositional fragment, $lc^i(C, E, a\_b) \neq \#$ and $lc^s(C, E, a\_b) \neq \#$.

Proof: We define
LC^i := λw. 1 iff for some formula g, for some good final b', w $\not\models^{f \to F}$ a ^{f}g b' $\Leftrightarrow$ a g b', where F is a contradiction.
Step 1: $LC^i \in tr^i(C, E, a\_b)$
For every $w \in C$, either $LC^i(w) = 1$, in which case the contextual restriction LC^i is innocuous at w and w $\models^{c' \to LC^i}$ a $^{c'}$g b' $\Leftrightarrow$ a g b'; or $LC^i(w) = 0$, which means that for every formula g, for every good final b', w $\models^{f \to F}$ a ^{f}g b' $\Leftrightarrow$ a g b', and thus w $\models^{c' \to LC^i}$ a $^{c'}$g b' $\Leftrightarrow$ a g b'.
Step 2: For every $x \in tr^i(C, F, a\_b)$, LC^i entails x.
Suppose, for contradiction, that $LC^i(w) = 1$ and $x(w) = 0$. Since $x \in tr^i(C, F, a\_b)$, for every formula g, for every good final b', w $\models^{c' \to x}$ a $^{c'}$g b' $\Leftrightarrow$ a g b' and since $x(w) = 0$ and the logic is extensional, w $\models^{f \to F}$ a ^{f}g b' $\Leftrightarrow$ a g b'. But by the definition of LC^i this means that $LC^i(w) = 0$, contrary to hypothesis.
The proof of $lc^s(C, F, a\_b) \neq \#$ is similar.

13. **Lemma 2: Closure under Finite Conjunction**

For any $C \subseteq W$, for any formula *a d b*, $tr^i(C, d, a\_b)$ and $tr^s(C, d, a\_b)$ are closed under finite conjunction.
Proof: Assume that x', x'' $\in tr^i(C, d, a\_b)$. We have in particular that for any admissible *d'* and for any good final *b'*,

1. $C \models^{e \to x' \wedge x'', c' \to x', c'' \to x''}$ a ^{e}d' b' $\Leftrightarrow$ a $^{c'}$(c'' and d') b'	[by the semantics of $^{c'}F$]
2. $C \models^{e \to x' \wedge x'', c' \to x', c'' \to x''}$ a $^{c'}$(c'' and d') b' $\Leftrightarrow$ a (c'' and d') b'	[because $x' \in tr^i(C, d, a\_b)$]
3. $C \models^{e \to x' \wedge x'', c' \to x', c'' \to x''}$ a (c'' and d') b' $\Leftrightarrow$ a $^{c''}$d' b'	[by the semantics of $^{c''}F$]
4. $C \models^{e \to x' \wedge x'', c' \to x', c'' \to x''}$ a $^{c''}$d' b' $\Leftrightarrow$ a' d' b'	[because $x'' \in tr^i(C, d, a\_b)$]
5. $C \models^{e \to x' \wedge x'', c' \to x', c'' \to x''}$ a ^{e}d' b' $\Leftrightarrow$ a' d' b'	[by 1.-4.]
6. $C \models^{e \to x' \wedge x''}$ a ^{e}d' b' $\Leftrightarrow$ a' d' b'	[by 5., since *c'* and *c''* don't appear]
7. $(x' \wedge x'') \in tr^i(C, d, a\_b)$	

The proof is similar for $tr^s(C, d, a\_b)$.

14. **Lemma 3: Finite Sets**
For every $v \in \{i, g\}$, if $tr^v(C, d, a\_b)$ is finite, $lc^v(C, d, a\_b) \neq \#$.
Proof: Immediate from 13.

15. **Lemma 4: Point-wise Construction of Local Contexts**
(This lemma crucially relies on the extensionality of the fragment)
For every $v \in \{i, g\}$, if for every $w \in C$, $lc^v(\{w\}, a\ d, a\_b) \neq \#$, then $lc^v(C, d, a\_b) \neq \#$.

Proof: The idea is to construct the bottom element point-wise, i.e. world by world.
We define LC^v as $\lambda w_s . lc^v(\{w\}, d, a\_b)$ if $w \in C$; z otherwise
where z is the null object of type t if *d* is propositional, and where z is the null object of type <e, t> if *d* is predicative.
-It is immediate that $LC^v \in tr^v(C, d, a\_b)$ (because of the extensionality of the fragment).
-Suppose, for contradiction, that for some $x \in tr^v(C, d, a\_b)$, x is not entailed by LC^v. Then there is some world w such that x(w) is not entailed by $LC^v(w)$. It couldn't be that $w \notin C$, since in that case $LC^v(w) = z$, which entails everything. So $w \in C$. Clearly, since $x \in tr^v(C, d, a\_b)$, $x(w) \in tr^v(\{w\}, d, a\_b)$. But by assumption $LC^v(w)$ is the bottom element of $tr^v(\{w\}, d, a\_b)$, so it entails x(w), *contra* hypothesis.

16. **Existence Theorem: Existence of Local Contexts**
Let $C \subseteq W$ be a context set and let *a E b* be any formula.
a. If *a <u>d</u>d b* belongs to the propositional fragment, then for every $v \in \{i, s\}$, $lc^v(C, E, a\_b) \neq \#$.
b. If for every $w \in C$, the domain of individuals in w is of finite size, then for every $v \in \{i, s\}$, $lc^v(C, E, a\_b) \neq \#$.
Proof:
(a) is just Lemma 1 [12]
(b) follows from Lemma 3 [14] and Lemma 4 [15], together with the following observation:
if the domain of individuals in w, D_w, is finite, then for any intensional type τ there are only finitely many functions of type τ with $D_s = \{w\}$[23]. It follows that $tr^v(\{w\}, E, a\_b)$ is finite. By Lemma 3 [14], we construct $lc^v(\{w\}, E, a\_b)$ for every $w \in C$. By Lemma 4 [15], this makes it possible to construct $lc^v(C, E, a\_b)$.

17. **Definition of Local Satisfaction - Special Case** (when local contexts exist)
Let $C \subseteq W$ be a context set.
a. For every $v \in \{i, s\}$, for all expressions *<u>d</u>d'*, *a*, *b*, if $lc^v(C, \underline{d}d', a\_b) \neq \#$, $Sat^v(C, \underline{d}d', a\_b)$ just in case $lc^v(C, \underline{d}d', a\_b) \leq \mathbf{d}$
b. For every $v \in \{i, s\}$, for every formula *F* , if for all expressions *a, b, <u>e</u>e'* such that F = a <u>e</u>e' b, $lc^v(C, \underline{d}d', a\_b) \neq \#$, $Sat^v(C, F)$ just in case for every expression *<u>e</u>e'*, for all strings *a, b*, if F = a <u>e</u>e' b, then $Sat^v(C, \underline{e}e', a\_b)$.

18. **Definition of Local Satisfaction - General Case** (local contexts need not exist)
Let $C \subseteq W$ be a context set.
a. For every $v \in \{i, s\}$, for all expressions *<u>d</u>d'*, *a*, *b*, $Sat'^v(C, \underline{d}d', a\_b)$ iff for some $X \in tr^v(C, \underline{d}d', a\_b)$, for every X', if [$X' \leq X$ and $X' \in tr^v(C, \underline{d}d', a\_b)$], then $C \models^{c' \to X'} c' \leq d$
b. For every $v \in \{i, s\}$, for every formula *F* , $Sat'^v(C, F)$ just in case for every expression *<u>e</u>e'*, for all strings *a, b,* if F = a <u>e</u>e' b, $Sat'^v(C, \underline{e}e', a\_b)$.

19. **Lemma 5: when local contexts exist, 18 and 17 are equivalent.**
For every $v \in \{i, s\}$, if $lc^v(C, \underline{d}d', a\_b) \neq \#$, $Sat^v(C, \underline{d}d', a\_b)$ iff $Sat'^v(C, \underline{d}d', a\_b)$
Proof: First, if $Sat^v(C, \underline{d}d', a\_b)$, then by taking $X = lc^v(C, \underline{d}d', a\_b)$, we can find an $X \in tr^v(C, \underline{d}d', a\_b)$

[23] The proof is by induction on primitive types relative to w. Clearly, D_w and {0, 1} are finite. Furthermore, if E is finite, so are $D_w \to E$ and $\{0, 1\} \to E$.

such that, for every X’, if [X’ ≤ X and X’ ∈ tr^{v}(C, $\underline{d}$d’, a_b)], then C $\models^{c' \to X'}$ c’ ≤ d. Second, if Sat’^{v}(C, $\underline{d}$d’, a_b), there is some X such that, for every X’, if [X’ ≤ X and X’ ∈ tr^{v}(C, $\underline{d}$d’, a_b)], then C $\models^{c' \to X'}$ c’ ≤ d. Since lc^{v}(C, $\underline{d}$d’, a_b) is the bottom element of tr^{v}(C, $\underline{d}$d’, a_b), lc^{v}(C, $\underline{d}$d’, a_b) ≤ X, and therefore C $\models^{c' \to lc^{v}(C, \underline{d}d', a\_b)}$ c’ ≤ d. By the definition tr^{v}(C, $\underline{d}$d’, a_b) and lc^{v}(C, $\underline{d}$d’, a_b), it must be the case that lc^{v}(C, $\underline{d}$d’, a_b) ≤ C. Therefore lc^{v}(C, $\underline{d}$d’, a_b) ≤ **d**. In other words, Sat^{v}(C, $\underline{d}$d’, a_b).

Before we go further, it is worth pointing out that the incremental version of local satisfaction systematically predicts presuppositions that are at least as strong as those predicted by the symmetric version (the same conclusion holds of the incremental vs. symmetric version of all the theories under study in this Appendix).

20. **Incremental Satisfaction predicts stronger presuppositions than Symmetric Satisfaction**
For any context set C, for all expressions *$\underline{d}$d’* and for all strings *a, b*,
a. tr^{i}(C, $\underline{d}$d’, a_b) ⊆ tr^{s}(C, $\underline{d}$d’, a_b).
Furthermore, if lc^{s}(C, d, a_b) ≠ # and lc^{i}(C, d, a_b) ≠ #,
b. lc^{s}(C, d, a_b) ≤ lc^{i}(C, d, a_b)
c. if Sat^{i}(C, d, a_b), then Sat^{s}(C, d, a_b)
Proof: Immediate.

We now turn to a comparison between Incremental Satisfaction, Incremental Transparency and Dynamic Semantics.

21. **Theorem. Equivalence with Transparency**
Let C ⊆ W be a context set. Then:
(i) For any v ∈ {i, s}, for every formula that has the form *a $\underline{d}$d’ b*, Sat’^{v}(C, $\underline{d}$d’, a _ b) iff Transp^{v}(C, $\underline{d}$d’, a _ b).
(ii) In particular, for any v ∈ {i, s}, for every formula *F*, Sat’^{v}(C, F) iff Transp^{v}(C, F).

Proof of (i):
We start with the incremental version.
=>: Suppose that Sat’^{i}(C, $\underline{d}$d’, a_b). Then for some X ∈ tr^{i}(C, $\underline{d}$d’, a_b),
C $\models^{c' \to X}$ c’ ≤ d
For every expression *d”* of the same type as *d*, for every good final *b’*,
C $\models^{c' \to X}$ a (d and d”) b’ ⇔ a $^{c'}$(d and d”) b’
C $\models^{c' \to X}$a $^{c'}$(d and d”) b’ ⇔ a $^{c'}$d” b’
C $\models^{c' \to X}$a $^{c'}$d” b’ ⇔ a d” b’
Hence
C ⊨ a (d and d”) b’ ⇔ a d” b’

<=: Suppose that Transp^{i}(C, $\underline{d}$d’, a_b) .
Clearly, **d** ∈ tr^{i}(C, $\underline{d}$d’, a_b). Furthermore, for every X’,
[X’ ∈ tr^{i}(C, $\underline{d}$d’, a_b) and X’ ≤ **d**] => C $\models^{c' \to X'}$ c’ ≤ d. So Sat’^{i}(C, $\underline{d}$d’, a_b).

The argument is similar for the symmetric version of *Sat* and *Transp*.

Proof of (ii): Immediate given (i).

22. **Theorem. Equivalence with Standard Dynamic Semantics**
Let C ⊆ W be a context set and *F* be a formula which satisfy Non-Triviality and Constancy. Then:
(i) for all expressions *a, b, $\underline{d}$d’* , if F = a $\underline{d}$d’ b, lc^{i}(C, $\underline{d}$d’, a _ b) ≠ #. Furthermore,
(ii) Sat^{i}(C, F) iff Sat’^{i}(C, F) iff C[F] ≠ #.

Proof of (i): Immediate from 16b and the fact that Constancy implies that each w ∈ C, the set of individuals in w is of finite size.
Proof of (ii): The first equivalence follows from (i) and the Lemma in 19. The second equivalence follows from (i), 9c and 21(ii).

Before going further, we consider an example in which local contexts do not exist, which makes it necessary to resort to the alternative definition of satisfaction, *Sat’*, whose incremental version yields full equivalence with Dynamic Semantics

23. **Infinitely Many**
Consider the formula F = (Infinitely-many P . $\underline{Q}$Q'). We assume that there are infinitely many elements in **P**(w).
a. *$\underline{Q}$Q'* has no local context in the context set {w}.
b. Sat'^{i}(C, F) (or equivalently Transp^{i}(C, F)) need not entail that C |= (Every P . Q)
Proof:
a. First, we note that if **c'** is transparent, **c'**(w) must itself contains infinitely many elements. For if not, *(infinitely-many P . $^{c'}$P)* would be false at w but *(infinitely-many P . P)* would be true - and *c'* wouldn't be transparent. Second, we show that for any transparent value x for *c'*, we can find a 'smaller' value x' which is also transparent. Since x(w) must contain infinitely many elements, we just take one arbitrary element out of x(w), obtaining an x'(w) distinct from x(w) with x'(w) ≤ x(w) (and also x ≤ x'). And it is clear that x', which itself contains infinitely many elements, is transparent (because the truth of the statement *infinitely many Ps are Qs* is insensitive to whatever happens to any given finite set of elements). Since x was arbitrary, we have shown that the set of transparent context denotations simply does not have a bottom element.
b. Assume that in w *Q* holds true of all *P*-individuals except a (non-zero) finite number. We have that for any predicative *D*, w |= (Infinitely-many P . (Q and D)) ⇔ (Infinitely-many P . D), so Transp^{i}(C, F), and therefore (by 21) Sat^{i}(C, F). Still, w |≠ (Every P . Q).

We turn to the trivalent alternatives to existing theories of presupposition; we start from a version of a supervaluationist account (somewhat adapted to the present framework), and then turn to a version of Strong Kleene (derived from the supervaluationist account). The following systems are approximations of theories pioneered by Peters, Beaver and Krahmer, and more recently George, and Fox. This entire discussion relies heavily on George's and Fox's ongoing research (George 2007, Fox 2007).

24. **Supervaluations I: Extensions of an interpretation**
Let I be an interpretation defined by 2. We start from the idea that $\underline{p}$p'(w) or $\underline{P}$P'(w)(d) are indeterminate if p(w) ≠ 1 and P(w)(d) ≠ 1. The set Ext(I) of extensions of the interpretation I is defined by the following procedure.
1. First, we treat expressions of the form *$\underline{p}$p'* or *$\underline{P}$P'* as *atomic* symbols of a classical language [rather than as complex symbols as in 2]; in effect, we obtain a new language L* which is identical to L except that expressions of the form *$\underline{p}$p'* are *$\underline{P}$P'* are part of the lexicon (in L they are syntactically complex). For simplicity, we identify formulas of L* with string identical formulas of L (despite the fact that they may be generated in distinct ways). We call I* the interpretation of L* which agrees with I, in the sense that for any expressions of the form *$\underline{p}$p'* or *$\underline{P}$P'* , I*($\underline{p}$p') = I($\underline{p}$p') and I*($\underline{P}$P') = I($\underline{P}$P'). It is immediate that such an interpretation I* can be found.
2. Second, for all interpretations i and i' of L*, we define i ∠ i' (meaning that *i' is an extension of i*) just in case for every atomic expression *E*:
-if *E* does not contain any underlined element, i(E) = i'(E)
-for every *p, p'*, if *E* = $\underline{p}$p', then for every world w, if i(p)(w)= 1, i'=($\underline{p}$p')(w) = i($\underline{p}$p')(w).
-for every *P, P'*, if *E* = $\underline{P}$P', then for every world w, for every individual d, if i(P)(w)(d) = 1, then i'($\underline{P}$P')(w)(d) = i($\underline{P}$P')(w)(d).
Note that in case i(p)(w) = 0, i'($\underline{p}$p')(w) may freely take the values 0 and 1; similarly, in case i(P)(w)(d) = 0, i'($\underline{P}$P')(w)(d) may freely take the value 0 and 1. This captures the intuition that in such cases the value of *$\underline{p}$p'* or *$\underline{P}$P'* is unknown, and could be 'resolved' in various ways.
3. Third, we define: Ext(I) := {i': i' is a classical interpretation of L* and I* ∠ i'}

25. **Supervaluations II: Truth**
As before, I is the classical interpretation defined by 2. For any formula *F* of L:
F is super-true at w (*notation*: Super(F, w) = 1) if and only if for every i ∈ Ext(I), i(F)(w) = 1
F is super-false at w (*notation*: Super(F, w) = 0) if and only if for every i ∈ Ext(I), i(F)(w) = 0
F is super-indeterminate at w (*notation*: Super(F, w) = #) if and only if *F* is neither true nor false at w.

26. **Supervaluations III: Acceptability**
a. Incremental Acceptability
F is incrementally super-acceptable in C (*notation*: Super-accept^{i}(C, F)) just in case for any presuppositional expression *$\underline{d}$d'*, for any strings α and β, if F = α $\underline{d}$d' β, then for any good final β' **which does not contain any underlined expressions**, for any world w ∈ C, Super(α $\underline{d}$d'β', w) ≠ #
b. Symmetric Acceptability

F is symmetrically super-acceptable relative to C (abbreviation: *Super-accept^s(C, F)*) just in case for any world $w \in C$, Super($\alpha\ \underline{d}d'\beta$, w) $\neq$ #.

Strong Kleene logic is usually defined as a compositional trivalent logic (by contrast, supervaluationist semantics is not compositional). It will be convenient, however, to define Strong Kleene in terms of supervaluations; we will show in 40-41 that in the propositional case our definition derives the same results as the standard definition (the equivalence extends to the full quantificational fragment, but giving a proof would require that we discuss the extension of Strong Kleene to generalized quantifiers, which isn't so common to begin with).

27. **Definitions: $[s]^\#$, L** and I****

a. Let s be any string. We define $[s]^\#$ to be identical to s, except that every occurrence of the form *$\underline{p}p'$* or *$\underline{P}P'$* is replaced with $\underline{p}p'^n$ and $\underline{P}P'^n$ if it is the n^{th} occurrence of its type in s (counting from left to right). We abbreviate *$[F]^\#$* as *$F^\#$* when there is no risk of ambiguity.

Examples: $[((p_1 \text{ and } \underline{p}_2p_3) \text{ or } \underline{p}_1p_2)]^\# = ((p_1 \text{ and } \underline{p}_2p_3{}^1) \text{ or } \underline{p}_1p_2{}^1)$
$[((p_1 \text{ and } \underline{p}_2p_3) \text{ or } \underline{p}_2p_3)]^\# = ((p_1 \text{ and } \underline{p}_2p_3{}^1) \text{ or } \underline{p}_2p_3{}^2)$

b. We define a set for formulas L** = L* $\cup$ {$F^\#$: F is a formula of L*}. L** is a subset of a language defined directly by the syntactic rules in 1, supplemented with: P ::= $\underline{P}_iP_k{}^n$ and p ::= $\underline{p}_ip_k{}^n$.

c. For every atomic expression *e* of L, we define $e^- = e$ if e does not contain any superscript; and if $e = e'^n$ for some atomic expression *e'* and some superscript n, $e^- = e'$.

d. We define an interpretation I** for L** as: for every atomic expressions e of L, I**(e) = I*(e^-)

28. **Strong Kleene I: Truth from Supervaluations**

Let *F* be a formula of L, interpreted by I. I** is defined for L** as specified in 27, and we apply the definition of extensions of an interpretation and supervaluationist truth in 24(2-3) and 25, but replacing L* and I* with L** and I** respectively. Finally, we define:
F is Kleene-true at w (*notation*: Kleene(F, w) = 1) iff *$F^\#$* is super-true at w (i.e. Super($F^\#$, w) = 1).
F is Kleene-false at w (*notation*: Kleene(F, w) = 0) iff *$F^\#$* is super-false at w (i.e. Super($F^\#$, w) = 0).
F is Kleene-indeterminate at w (*notation*: Kleene(F, w) = #) iff *$F^\#$* is super-indeterminate at w (i.e. Super($F^\#$, w) = #).

29. **Strong Kleene II: Acceptability**

a. Incremental Acceptability
F is incrementally Kleene-acceptable relative to C (*notation*: Kleene-accept^i(C, F)) just in case for any presuppositional expression *$\underline{d}d'$*, for any strings α and β, if F = $\alpha\ \underline{d}d'\ \beta$, then for any good final β' **which does not contain any underlined expressions**, for any world $w \in C$, Kleene($\alpha\ \underline{d}d'\beta'$, w) $\neq$ #.
b. Symmetric Acceptability
F is symmetrically Kleene-acceptable relative to C (*notation*: Kleene-accept^s(C, F)) just in case for any world $w \in C$, Kleene(F, w) $\neq$ #.

30. **Lemma 6.** Let *F* be any formula of L, and assume that for every expression *$\underline{d}d'$*, if *$\underline{d}d'$* occurs in *F*, it occurs exactly once. Then for any world w, Super(F, w) = Kleene(F, w)

Proof: The result is trivial, since in that case *$F^\#$* is obtained from *F* by uniform replacement of certain propositional and predicative constants with other constants that have the same value.

31. **Lemma 7.** For any formula *F*, for any world w, if Kleene(F, w) $\neq$ #, then Super(F, w) $\neq$ # and Super(F, w) = Kleene(F, w).

Proof: Suppose that Kleene(F, w) $\neq$ #. This means that for some $v \in \{0, 1\}$, for all i' $\in$ Ext(I**), i'($F^\#$) = v. Let E be the set {i: i is an interpretation of L** and for all integers n, k, for all underlined expressions *$\underline{d}d'$*, i($\underline{d}d'^n$) = i($\underline{d}d'^k$)}. For each formula *F* of L, for each interpretation i' of Ext(I*), there is an interpretation i of

E such that $i(F^{\#}) = i'(F)$. It follows that for all $i' \in Ext(I^{**})$, $i'(F) = v$, and thus Super(F, w) $\neq$ # and Super(F, w) = v = Kleene(F, w).

32. **Lemma 8.** For every $v \in \{i, s\}$, for any formula *F*, for any context set $C \subseteq W$, if Kleene-accept^v(C, F), then:
(i) Super-accept^v(C, F), and
(ii) for every $w \in C$, Kleene(F, w) = Super(F, w).

Proof: This is a consequence of 31.
-Suppose that Kleene-accept^i(C, F). For any presuppositional expression $\underline{dd}'$, for any strings α and β, if $F = \alpha\ \underline{d}d'\ \beta$, then for any good final β' which does not contain any underlined expressions, for any world $w \in C$, Kleene($\alpha\ \underline{d}d'\beta'$, w) $\neq$ #, whence by 31 Super($\alpha\ \underline{d}d'\beta'$, w) $\neq$ #. It follows that Super-accept^i(C, F).
-Suppose that Kleene-accept^s(C, F), then for every $w \in C$, Kleene(F, w) $\neq$ #, and by 31 Super(F, w) $\neq$ #. Therefore Super-accept^s(C, F).
-If Kleene-accept^i(C, F), *a fortiori* Kleene-accept^s(C, F). And if Kleene-accept^s(C, F), Kleene(F, w) $\neq$ # and by 31 Super(F, w) = Kleene(F, w).

33. **Definitions**
For any formula *F*, for any context set $C \subseteq W$, for every integer $n \geq 0$:
a. Super-accept^i(C, F, n) iff for every underlined expression $\underline{dd}'$, for all strings α, β, if $F = \alpha\ \underline{d}d'\ \beta$ and there are at most n underlined tokens in $\alpha\ \underline{dd}'$, then for every good final β' which does not contain any underlined expressions, for every world $w \in C$, Super($\alpha\ \underline{d}d'\beta'$, w) $\neq$ #.
b. Kleene-accept^i(C, F, n) iff for every underlined expression $\underline{dd}'$, for all strings α, β, if $F = \alpha\ \underline{d}d'\ \beta$ and there are at most n underlined tokens in $\alpha\ \underline{dd}'$, then for every good final β' which does not contain any underlined expressions, for every world $w \in C$, Kleene($\alpha\ \underline{d}d'\beta'$, w) $\neq$ #.
c. Transp^i(C, F, n) iff for every underlined expression $\underline{dd}'$, for all strings α, β, if $F = \alpha\ \underline{d}d'\ \beta$ and there are at most n underlined tokens in $\alpha\ \underline{dd}'$, then for every good final β' which does not contain any underlined expressions, Transp^i(C, $\alpha\ \underline{d}d'\beta'$).
d. Ext(F, n) = {i: for all atomic expressions e, if e is not one the first n underlined tokens of *F*, $i(e) = I^{**}(e)$ and if *e* is one of the n first underlined tokens of *F*, $I^{**}(e) \angle i^{*}(e)$}

Note: Super-accept^i(C, F, 0), Kleene-accept^i(C, F, 0) and Transp^i(C, F, 0) are all trivially true, since it is never the case that $F = \alpha\ \underline{d}d'\ \beta$ with at most 0 underlined tokens in $\alpha\ \underline{dd}'$.

34. **Lemma 9**
a. Kleene-accept^i(C, F, n) iff for every underlined expression $\underline{dd}'$, for all strings α, β, if $F = \alpha\ \underline{d}d'\ \beta$ and there are at most n underlined tokens in $\alpha\ \underline{dd}'$, then for every good final β' which does not contain any underlined expressions, for all $i, i' \in Ext(F^{\#}, n)$, for all $w \in C$, $i([\alpha\ \underline{d}d']^{\#}\ \beta') = i'([\alpha\ \underline{d}d']^{\#}\ \beta')$.
b. If Kleene-accept^i(C, F, n), then for all strings α, β, if $F = \alpha\ \beta$ and there are at most n underlined tokens in α, then for any good final β' that does not contain any underlined material, for every world $w \in C$, Kleene($\alpha\ \beta'$, w) = $I^{**}(\alpha\ \beta')(w)$.

Proof:
a. Immediate given the definition of Kleene-accept^i(C, F, n).
b. The case n = 0 is trivial. For $n \geq 1$, if Kleene-accept^i(C, F, n), then for every underlined expression $\underline{dd}'$, for all strings α, β, if $F = \alpha\ \underline{d}d'\ \beta$ and there are at most n underlined tokens in $\alpha\ \underline{dd}'$, then for every good final β' which does not contain any underlined expressions, for all $i, i' \in Ext(F^{\#}, n)$, for all $w \in C$, $i([\alpha\ \underline{d}d']^{\#}\ \beta') = i'([\alpha\ \underline{d}d']^{\#}\ \beta')$. But it is clear that $I^{**} \in Ext(F^{\#}, n)$, so for all $i \in Ext(F^{\#}, n)$, for all $w \in C$, $i([\alpha\ \underline{d}d']^{\#}\ \beta') = I^{**}([\alpha\ \underline{d}d']^{\#}\ \beta') = I^{**}(\alpha\ \underline{d}d'\ \beta')$ (by the definition of I^{**}). The result follows.

35. **Lemma 10**
a. If Super-accept^i(C, F, n), then for every underlined expression $\underline{dd}'$, for all strings α, β, if $F = \alpha\ \underline{d}d'\ \beta$ and there are at most n underlined tokens in $\alpha\ \underline{dd}'$, then for any good final β' that does not contain any underlined material, Super-accept^i(C, $\alpha\ \underline{d}d'\ \beta'$) and therefore Super-accept^s(C, $\alpha\ \underline{d}d'\ \beta'$).
b. If Kleene-accept^i(C, F, n), then for every underlined expression $\underline{dd}'$, for all strings α, β, if $F = \alpha\ \underline{d}d'\ \beta$ and there are at most n-1 underlined tokens in α, then for any good final β' that does not contain any underlined material, Kleene-accept^i(C, $\alpha\ \underline{d}d'\ \beta'$) and therefore Kleene-accept^s(C, $\alpha\ \underline{d}d'\ \beta'$).

Proof: Immediate given the relevant definitions.

36. **Theorem. Incremental Kleene = Incremental Supervaluations**
For any formula *F*, for any context set $C \subseteq W$,
a. Kleene-accept^i(C, F) iff Super-accept^i(C, F), and
b. if Kleene-accept^i(C, F), then for every world $w \in C$, Kleene(F, w) = Super(F, w).

Proof:
-Part (b) is just 32 (ii).
-Part (a): By 32 (i), if Kleene-accept^i(C, F), then Super-accept^i(C, F). To prove the converse, we establish by induction the following result for every integer $n \geq 1$:

P(n): If Super-accept^i(C, F, n), Kleene-accept^i(C, F, n).

By choosing n greater than the number of underlined expressions in *F*, the desired result will follow.

n = 0: Trivial.
n = m+1: Suppose that Super-accept^i(C, F, m+1). Then in particular Super-accept^i(C, F, m), and by the induction hypothesis Kleene-accept^i(C, F, m). Suppose that $F = \alpha\ \underline{d}d'\ \beta$, where α contains m underlined expressions. We consider the set E = {D: D is a non-underlined expression of the same semantic type as $\underline{d}d'$ and for some i' such that $I^{**} \angle i'$, $I^{**}(D) = i'(\underline{d}d')$}. By our assumption of Expressivity (in 3), E fully describes the possible extensions of $\underline{d}d'$: $\{I^{**}(D): D \in E\} = \{i'(\underline{d}d'): I^{**} \angle i'\}$. Since Kleene-accept^i(C, F, m), for all $D \in E$, for every good finals β' that does not contain any underlined expression, for all $w \in W$, for all i' such that $I^{**} \angle i'$, for some $v \in \{0, 1\}$, $i'(\alpha\ D\ \beta')(w) = v$. Since $D\ \beta'$ does not contain any underlined expressions, by the induction hypothesis Kleene-accept^i(C, α D β', m), hence by 34b Kleene-accept^s(C, α D β'), and thus for some $v' \in \{0, 1\}$, for all i' such that $I^{**} \angle i'$, $i'(\alpha^\#\ D\ \beta', w) = v'$. By 31 v = v', and thus for all i' such that $I^{**} \angle i'$, $i'(\alpha^\#\ D\ \beta', w) = i'(\alpha\ D\ \beta', w)$. It follows that for all $D \in E$, for every good final β' that does not contain any underlined material, for all i' such that $I^{**} \angle i'$, for all $w \in W$, for some $v \in \{0, 1\}$, $i'(\alpha^\#\ D\ \beta', w) = v$ - which shows that Kleene-accept^i(C, $\alpha\ \underline{d}d'\ \beta$, m+1); in other words, Kleene-accept^i(C, F, m+1).

37. **Lemma 11**
If Kleene-accept^s(C, F), then Super-accept^s(C, F), but in general the converse is not true.

Proof:
-The *if* part follows from 32 (i).
-To see that the converse is not true, consider the formula *($\underline{p}$p' or (not $\underline{p}$p'))*: it is symmetrically super-acceptable relative to W, the set of all possible worlds, but which is not symmetrically Kleene-acceptable relative to W if *p* is non-tautologous.

38. **Theorem: Incremental Transparency predicts stronger presuppositions than Incremental Kleene and Incremental Supervaluations.**
If Transp^i(C, F), then Kleene-accept^i(C, F) (and thus also Super-accept^i(C, F)) - but in general the converse does not hold.

Proof:

a. We show by induction on $n \geq 0$ that if Transp^i(C, F, n), then Kleene-Accept^i(C, F, n). The result will follow because if Transp^i(C, F), then Transp^i(C, F, n) for n ≥ the number of underlined tokens in F. From Kleene-Accept^i(C, F, n) it then follows that Kleene-Accept^i(C, F).
n = 0: Trivial.
n = m+1: Suppose that α contains m underlined expressions and let $\alpha\ \underline{d}d'$ be an initial string of *F*. We assume that Transp^i(C, F, m+1). Thus for every *g* of the same type as *d*, for every good final β',

(i) $C \models \alpha$ (d and g) $\beta' \Leftrightarrow \alpha\ g\ \beta'$

Let w be any world of C. We define:
G = {g: g is an expression of the same type as *d* and I**(d and g)(w) = I**(d and d')(w)}
G' = {I**(g)(w): g ∈ G}
D = {i'($\underline{d}$d'$^{m+1}$)(w): i' ∈ Ext(F$^{\#}$, m+1)}
Step 1. We establish that D' ⊆ G.

Case 1: *d* is propositional.
-Suppose I**(d)(w) = 1. Then if i' ∈ Ext(F$^{\#}$, m+1), i'($\underline{d}$d'$^{m+1}$)(w) = I**(d')(w), and since d' ∈ G, i'($\underline{d}$d'$^{m+1}$)(w) ∈ G'.
-Suppose I**(d)(w) = 0. Then if i' ∈ Ext(F$^{\#}$, m+1), i'($\underline{d}$d'$^{m+1}$)(w) may have values 0 or 1. By choosing g_0 such that w |≠ g_0 and g_1 such that w |= g_1 (which we can do by the assumption of Expressivity in 3), it is clear that I**(d and g_1)(w) = I**(d and g_2)(w) = 0 = I**(d and d')(w) [no matter what the value of *d'* is at w]. So g_0 and g_1 both belong to G, hence 0 and 1 both belong to G'. In other words, for any i' ∈ Ext(F$^{\#}$, m+1), i'($\underline{d}$d'$^{m+1}$)(w) ∈ G'.

Case 2: *d* is predicative.
If i' ∈ Ext(F$^{\#}$, m+1), i'($\underline{d}$d'$^{m+1}$)(w) guarantees that for any individual x, if I**(d)(w)(x) = 1, then i'($\underline{d}$d'$^{m+1}$)(w)(x) = I**(d')(w)(x). By Expressivity, we find *g* such that for every individual x, I**(g)(w)(x) = I**(d')(w)(x) [= i'($\underline{d}$d'$^{m+1}$)(w)(x)] if I**(d)(w)(x) = 1 and I**(g)(w)(x) = i'($\underline{d}$d'$^{m+1}$)(w)(x) if I**(d')(w)(x) = 0. It is clear that I**(g)(w) = i'(d)(w), and furthermore that for every individual x I**(d and g)(w)(x) = I**(d and d')(w)(x), i.e. that I**(d and g)(w) = I**(d and d')(w). So *g* ∈ G, and thus I**(g)(w), which is identical to i'(d)(w), belongs to G'.

Step 2. Now we show that for every string α, for every underlined expression *$\underline{d}$d'*, for every string β, if α *$\underline{d}$d'* contains at most m+1 underlined expressions and F = α $\underline{d}$d' β, then for any good final *β'* which does not contain any underlined expressions, if i' ∈ Ext([α $\underline{d}$d']$^{\#}$ β', m+1), for any world w ∈ C, i'([α $\underline{d}$d']$^{\#}$ β')(w) = I**(α $\underline{d}$d' β')(w); by 34 this will derive the desired result, namely that Kleene-Accept^{i}(C, F, m+1)
• Since D' ⊆ G, for some g ∈ G, i'([α $\underline{d}$d']$^{\#}$ β')(w) = i'($\alpha^{\#}$ $\underline{g}$ β')(w) with g satisfying I**(g)(w) = i'($\underline{d}$d'$^{m+1}$)(w). Furthermore, by the induction hypothesis Kleene-Accept^{i}(C, F, m), and thus by 34 Kleene($\alpha^{\#}$ $\underline{g}$ β')(w) = I**(α $\underline{g}$ β')(w), and thus also i'($\alpha^{\#}$ $\underline{g}$ β')(w) = I**(α $\underline{g}$ β')(w).
• Since Transp^{i}(C, F, m+1), I**(α g β')(w) = I**(α(d and g) β')(w). By the definition of g, I**(d and g)(w) = I**(d and d')(w), hence i'($\alpha^{\#}$ $\underline{g}$ β')(w) = I**(α $\underline{g}$ β')(w) = I**(α(d and g) β')(w) = I**(α(d and d') β')(w). This completes the proof.

b. To show that Kleene-accept^{i}(C, F) need not entail that Transp^{i}(C, F), we consider the formula and the situation in (ii):

(ii) a. (No P . $\underline{Q}$Q')
b. C = {w}, and there are two P-individuals d_1 and d_2 in w: d_1 does not satisfy *Q*, but d_2 does, and d_2 also satisfies *Q'*.

It was shown in (35) that Incremental Satisfaction predicts that (iia) presupposes that *every P-individual in w satisfies Q*. But since d_2 satisfies both *Q* and *Q'*, Incremental Kleene and Incremental Supervaluations predict that (iia) should be acceptable in C.

39. **Theorem. Symmetric Transparency and Symmetric Kleene are incomparable.**
a. Sometimes Symmetric Kleene and Symmetric Supervaluations predict stronger presuppositions than Symmetric Transparency.
b. Sometimes Symmetric Transparency predicts stronger presuppositions than Symmetric Kleene and Symmetric Supervaluations.

Proof:
a. Let us consider the formula and the situation in (i):

(i) a. ($\underline{p}$p' and $\underline{q}$q')
b. C = {w_1, w_2}, w_1 |≠ p, w_1 |≠ q, w_2 |= p and w_2 |= q

Let w belong to C. If w = w_1 , Symmetric Transparency is satisfied because for all *g*:

(ii) a. w |= ((pp' and (q and g)) ⇔ (pp' and g) [both sides are false because w_1 |≠ p and thus w_1 |≠ pp']
b. w |= ((p and g) and qq') ⇔ (g and qq') [both sides are false because w_1 |≠ q and thus w_1 |≠ qq']

If w = w_2, Symmetric Transparency is trivially satisfied in w.
By contrast, both Symmetric Kleene and Symmetric Supervaluations predict the sentence to be a presupposition failure.

b. The example is the same as in 38 b (in this case there is no difference between incremental and symmetric notions, because the presupposition trigger appears at the end of the formula).

40. **Strong Kleene: Standard definition (propositional case)**
We only give rules for the atomic case, as well as *and* and *not*; the semantics of the other connectives follows from standard rules of inderdefinability (which of course hold in Supervaluationist semantics as well).
We assume once again that a classical interpretation I is defined as in 2, and we use it to define a standard Strong Kleene interpretation - with the usual convention that *pp'* is indeterminate whenever *p* is false. As before, we abbreviate $I_w(F)$ as F^w, and we define the standard Strong Kleene (called here Standard-Kleene) by specifying that for any world w and any formula *F*:
if F = pp', Standard-Kleene(F, w) = 1 iff $p^w = p'^w = 1$; Standard-Kleene(F, w) = 0 iff $p^w = 1$ and $p'^w = 0$; Standard-Kleene(F, w) = # otherwise.
if F = (G and H), Standard-Kleene(F, w) = 1 iff Standard-Kleene(G, w) = Standard-Kleene(H, w) = 1; Standard-Kleene(F, w) = 0 iff Standard-Kleene(G, w) = 0 or Standard-Kleene(H, w) = 0; Standard-Kleene(F, w) = # otherwise.
if F = (not G), Standard-Kleene(F, w) = 1 iff Standard-Kleene(G, w) = 0; Standard-Kleene(F, w) = 0 iff Standard-Kleene(G, w) = 1; Standard-Kleene(F, w) = # otherwise.

41. **Theorem. Equivalence between the two definitions of Strong Kleene in the propositional case.**
If *F* belongs to the propositional fragment of 1, Standard-Kleene(F, w) = Kleene(F, w)

Proof:
As before, we start from a classical interpretation I for the fragment in 1, and we call I** the classical interpretation determined for the set of formulas of the form *F* or $F^\#$, where *F* is a formula of 1. We can then study the interpretation of these formulas according to our version of Strong Kleene and Standard Strong Kleene.
Since Standard Strong Kleene is compositional, it is immediate that for any formula *F* of 1, for any world w, Standard-Kleene(F, w) = Standard-Kleene($F^\#$, w). By the definition of our version of Strong Kleene, it is also clear that Kleene(F, w) = Kleene($F^\#$, w). All that remains to show, then, is that Standard-Kleene($F^\#$, w) = Kleene($F^\#$, w). Since by construction $F^\#$ contains at most one occurrence of any underlined expression, by 30 all we have to show is that Standard-Kleene($F^\#$, w) = Super($F^\#$, w). The proof is by induction on the construction of $F^\#$.

If for some i, F = p_i, the result is immediate.
If some i, k, n, F = $p_i p_k{}^n$, Standard-Kleene(F, w) = # iff $p_i{}^w = 0$, iff Super(F, w) = #. It is immediate that if Standard-Kleene(F, w) ≠ #, Standard-Kleene(F, w) = F^w = Super(F, w).
If F = (not G), the result is immediate.
If F = (G and H), there are three cases to consider: Standard-Kleene(F, w) = 1, Standard-Kleene(F, w) = 0, and Standard-Kleene(F, w) = #.
• Case 1: Standard-Kleene(F, w) = 1. Then Standard-Kleene(G, w) = Standard-Kleene(H, w) = 1, hence by the induction hypothesis, for every i'∈ Ext(I), i'(G) = i'(H) = 1, and hence i'(F) = 1. Therefore Super(F, w) = 1.
• Case 2: Standard-Kleene(F, w) = 0. Then Standard-Kleene(G, w) = 0 or Standard-Kleene(H, w) = 0. Let us assume that Standard-Kleene(G, w) (the other case is similar). By the induction hypothesis, for every i' ∈ Ext(I), i'(G) = 0, and thus i'(F) = 0. Therefore Super(F, w) = 0.
• Case 3: Standard-Kleene(F, w) = #.
(a) If Standard-Kleene(G, w) = 1 and Standard-Kleene(H, w) = # (or conversely - the reasoning is the same), by the induction hypothesis we have that for every every i' ∈ Ext(I), i'(G, w) = 1 and for some j', j" ∈ Ext(I), j'(H) = 1 and j"(H) = 0. It follows that j'(F) = 1 and j"(F) = 0, hence Super(F, w) = #.

(b) If Standard-Kleene(G, w) = Standard-Kleene(H, w) = #, for some bivalent i', i", j', j" $\in$ Ext(I), i'(G) = j'(H) = 1, i"(G) = j"(H) = 0. Let us call At the set of all atomic propositions, and At(F) the set of atomic propositions that occur in *F*. We can define two new interpretations of atomic elements as follows:

$i'+j' = i'_{/(At-At(G))} \cup j'_{/At(H)}$

$i''+j'' = i''_{/(At-At(G))} \cup j'_{/At(H)}$

Since i', i", j', j" $\in$ Ext(I), it is also clear that i'+j' $\in$ Ext(I) and i"+j" $\in$ Ext(I). Since At(G) $\cap$ At(H) = Ø, i'+j'(G) = i'(G) and i'+j'(H) = j'(H) ; and similarly i"+j"(G) = i"(G) and i"+j"(H) = j"(H). It follows in particular that

i'+j'(G) = i'+j'(H) = 1, hence i'+j'(F) = 1

i"+j"(G) = i"+j"(H) = 0, hence i"+j"(F) = 0

and thus Super(F, w) = #.

References

Abusch, D.: 2002, Lexical Alternatives as a Source of Pragmatic Presuppositions. In B. Jackson, ed., *Proceedings of SALT XII,* CLC Publications, Ithaca NY.

Beaver, D.: 2001, *Presupposition and Assertion in Dynamic Semantics*. CSLI, Stanford.

Beaver, D. and Krahmer, E.: 2001, A partial account of presupposition projection. *Journal of Logic, Language and Information* 10.147–182.

Chemla, E.: 2007a, Presuppositions of Quantified Sentences: Experimental Data. Manuscript, LSCP.

Chemla, E.: 2007b, Similarity: Towards a Unified Account of Scalar Implicatures, Free Choice Permission and Presupposition Projection. Manuscript, LSCP.

Chemla, E.: 2007c, *Présuppositions et implicatures scalaires: études formelles et expérimentales*, Doctoral dissertation, EHESS.

Davis, S. (ed): 1991, *Pragmatics: A Reader*, Oxford University Press.

Fox, D.: 2007, Lecture notes on presuppositions. Hand-out, MIT.

Gazdar, G.: 1979, *Pragmatics: Implicature, Presupposition and Logical Form*. Academic Press.

George, B.: 2007, Variable Presupposition Strength in Quantifiers. Poster, XPrag Conference, Berlin.

Geurts, B.: 1999, *Presupposition and Pronouns*. Elsevier.

Heim, I.: 1983, On the Projection Problem for Presuppositions. In D. Flickinger et al. (eds), *Proceedings of the Second West Coast Conference on Formal Linguistics*, 114-125. Reprinted in Davis 1991.

Heim, I.: 1990, Presupposition Projection. In R. van der Sandt (ed.), *Reader for the Nijmegen Workshop on Presupposition, Lexical Meaning, and Discourse Processes*. U. of Nijmegen.

Heim, I. (1992). Presupposition projection and the semantics of attitude verbs. *Journal of Semantics*, 9(3), 183-221.

Hurford, J.: 1974, Exclusive or inclusive disjunction. *Foundations of Language*, 11:409–411.

Karttunen, L.: 1974, Presupposition and Linguistic Context. *Theoretical Linguistics* 1: 181-194. Reprinted in Davis 1991.

Keenan, E. 1996, The Semantics of Determiners, in Lappin, S. (ed.) *The Handbook of Contemporary Semantic Theory*

LaCasse, N.: 2007, Constraining Dynamic Semantics. Manuscript, UCLA.

Lewis, D.: 1973, *Counterfactuals*. Blackwell.

Moltmann, F.: 1997, Contexts and Propositions. Manuscript.

Moltmann, F.: 2003, Contexts, Complex Sentences, and Propositional Content. Manuscript, U. of Stirling.

Peters, S.: 1979, A truth-conditional formulation of Karttunen's account of presupposition. *Synthese*, 40:301-316.

Schlenker, P.: 2007a, "Anti-Dynamics (Presupposition Projection Without Dynamic Semantics)". *Journal of Logic, Language and Information* 16, 3: 325-256

Schlenker, P.: 2007b, Be Articulate: A Pragmatic Theory of Presupposition Projection. To appear as a target article (with commentaries) in *Theoretical Linguistics*.

Schlenker, P.: 2008, Local Contexts: A Précis. Manuscript, IJN and NYU.

Simons, M.: 2001, On the Conversational Basis of some Presuppositions. In Hasting, R., Jackson, B. and Zvolenzky, S. (eds.) *Proceedings of SALT 11*, CLC publications, Cornell University.

Singh, R.: 2007a, On the Interpretation of Disjunction: Asymmetric, Incremental, and Eager for Inconsistency. Manuscript, MIT.

Singh, R.: 2007b, Assertability Constraints and Absurd Assertions. Manuscript, MIT.

Spector, B., Fox, D. and Chierchia, G.: 2008, Hurford's Constraint and the Theory of Scalar Implicatures. Manuscript, MIT and Harvard.

Soames, S.: 1989, Presupposition. In D. Gabbay and F. Guenthner (eds), *Handbook of Philosophical Logic IV*, 553-616.

Stalnaker, R.: 1974, Pragmatic Presuppositions. In Munitz, M. and Unger, P. (eds.) *Semantics and Philosophy*. New York: New York University Press. Reprinted in Davis 1991.

Stalnaker, R.: 1978, Assertion. In Cole, P. (ed.) *Syntax and Semantics,* vol. 9: *Pragmatics*: 315-322. Reprinted in Davis 1991.

Unger, C. and van Eijck, J.: 2007, The Epistemics of Presupposition Projection. Abstract, Amsterdam Colloquium 2007.

van Rooij, R.: 2007, Strengthening Conditional Presuppositions. *Journal of Semantics,* 2007, 24: 289-304

van der Sandt, R.: 1988, *Context and Presupposition*, Croom Helm.

van der Sandt, R.: 1993, Presupposition Projection as Anaphora Resolution. *Journal of Semantics* 9(4): 333-377.