Semantic Transition Rules
(of the Semantic Module of the Meaning-Text Linguistic Model)

Igor Mel’čuk

OLST – Université de Montréal
C.P. 6128, Centre-ville, Montréal H3C3J7, Canada
igor.melcuk@umontreal.ca

Abstract

The paper sketches the structure of the linguistic part within the main component of the Semantic Module of the Meaning-Text linguistic model, namely—the structure of the set of Semantic Transition Rules (the semantics-syntax interface). These rules fall into two major types: Lexicalization rules, which specify the labels on the nodes of the DSyntS, vs. Arborization rules, which characterize the DSynt-tree itself. Lexicalization rules include lexical rules (for the lexical part of the node labels) and inflectional rules (for the grammemic part of such labels); lexical rules describe either genuine lexical units (actual ones, stored in the lexicon, and potential ones, constructed by productive composition and derivation rules) or fictitious lexemes, which represent meaningful syntactic constructions. Arborization rules establish the top node of the DSyntS and describe its branches (specifying the branch labels).

Keywords


1 General Overview of Semantic Transition Rules

The goal of this paper is to bring some logical order into our ideas about the organization of the Sem-Module of the Meaning-Text linguistic model [= MTM]. Two important provisos are in order:

• Only static, i.e., linguistic rules, which constitute the module’s purely descriptive part, are considered; the dynamic, i.e., procedural rules, which put linguistic rules to work and actually run the module, are fully ignored in what follows.

• The paper does not present new facts or new methodologies; it aims exclusively at providing a clear-cut view of an important component of the MTM.

The Sem-Module of the MTM (roughly, Semantics-Syntax interface), if we leave aside its dynamic core, consists of three components, or sets of rules: 1) Sem-Paraphrasing, or Sem-equivalence, rules, which produce for a given Sem(antic) R(epresentation) equivalent SemRs; 2) Sem-Transition, or Sem-expression, rules, which produce for a given SemR the correspon-
ling D(EEP-)SYNT(actic)Rs; 3) DSynt-Paraphrasing, or DSynt-equivalence, rules, which produce for a given DSyntR equivalent DSyntRs.¹ (Under implementation, the three sets of rules are pressed into service by three corresponding formal grammars.) Here only the Sem-Transition rules are considered.

**NB. 1.** To express correspondences, two symbols are used:

- \( A \Leftrightarrow B \) represents a correspondence between elements of two adjacent levels of linguistic representation, of which \( A \) is deeper (= closer to the meaning); this is transition.
- \( A \equiv B \) stands for a correspondence between elements of the same level; this is paraphrasing.

² Given the limitations of space, sufficient familiarity with the Meaning-Text approach is taken for granted. Its basic notions, terms and abbreviations are used without explanation.

The Sem-Transition rules of the Sem-Module establish correspondences between

\[
\text{SemR} = \langle \text{SemS} ; \text{Sem-CommS} ; \text{Sem-RhetS} ; \text{RefS} \rangle
\]

and

\[
\text{DSyntR} = \langle \text{DSyntS} ; \text{DSynt-CommS} ; \text{DSynt-AnaphS} ; \text{DSynt-ProS} \rangle
\]

Since the Referential Structure of a SemR is not explicitly expressed in the resulting sentence,² the Sem-Module ensures the correspondences between three structures of the Sem-level and all four structures of the DSynt-level. This gives 12 logically possible types of Sem-Transition rules (if we consider only the pairs of structures such that some elements of one structure correspond to some elements of the other). Not all of these 12 types of rules occur; here are four major blocs of Sem-Transition rules (for short, Sem-rules):

1. SemS \( \Leftrightarrow \) \( \{\text{DSyntS}, \text{DSynt} - \text{AnaphS}\} \) rules: lexical-syntactic Sem-rules.

Applied to a SemS, they specify the corresponding DSyntSs with DSynt-AnaphSs superimposed onto them; this is done observing the constraints that come from Sem-CommS, Sem-RhetS, and RefS. The lexical-syntactic Sem-rules constitute quantitatively and qualitatively the most important group of Sem-rules.


Applied to a SemS, they specify semantic prosodemes (for the chunks of meaning such as ‘assertion’ vs. ‘interrogation’ vs. ‘exclamation’; or ‘amazement’ vs. ‘irony’ vs. ‘threat’; etc.) and relate them to corresponding Deep-Syntactic subtrees.


Based on the Sem-CommS, they specify the DSynt-CommS of the sentence: Sem-CommS \( \Leftrightarrow \) DSynt-CommS rules. However, some communicative values can be expressed lexically (i.e., by communicative particles, like the Somali Rheme markers WAA, BAA, etc., the Focalization marker IT IS X THAT ..., etc.) or inflectionally (i.e., by affixes, like the Japanese ¹

The inclusion of the DSynt-paraphrasing rules in the Sem-Module as its upper component is justified by the fact that this module is supposed to associate a given SemR with all possible DSyntRs, which makes considering DSynt-Paraph rules as part of the Sem-Module unavoidable. However, these rules also belong to the DSynt-Module (as its lower component). In point of fact, their inclusion in one or the other module of the MTM is rather a matter of convention and has no relevant consequences.

² But it is vital for the SemR \( \Leftrightarrow \) DSyntR transition: information drawn from the RefS is used to condition SemS \( \Leftrightarrow \) DSyntS rules.
Theme marker -wa); moreover, the Sem-CommS can affect the organization of the DSyntS. This is why the communicative Sem-rules include also Sem-CommS ⇔ DSyntS rules.

4. Sem-RhetS ⇔ \text{DSynt} \setminus \text{ProsS} \text{ rules: rhetorical Sem-rules.}

The ‘artistic’ intentions of the Speaker can be realized lexically/inflectionally (i.e., by the choice of syntactic constructions) or prosodically: hence two subtypes of rhetorical Sem-rules.

This paper concentrates on purely linguistic lexical-syntactic Sem-rules, that is, on the SemS ⇔ DSyntS rules. These rules are subdivided into two groups:

- Lexicalization rules (Section 2) provide the labels on the nodes of the DSyntS. They specify a) the lexical (part of the) labels and b) the inflectional (= grammemic) subscripts to these labels; since they deal with nodes, they are called nodal rules. Lexicalization Sem-rules for a given language \( \mathbf{L} \) are very numerous (about one million), because, among other things, they describe all lexical units [= LUs] of \( \mathbf{L} \).

- Arborization rules (Section 3) provide the formal ‘skeleton’ of the DSyntS. They a) determine the top node of the DSyntS, b) characterize its branches, and c) specify the DSynt-Relation labels on these branches. Since the rules performing tasks b) and c) deal with the arrows of syntactic dependencies, they are called sagittal rules (from Lat. \textit{sagitta} (arrow)); rules responsible for the task a) are of a special type.

See in Fig. 1, next page, the (partial) typology of Sem-rules.

Sem-Rules—that is, the rules that specify the Sem ⇔ DSynt-transition—were studied for the first time in Iordanskaja & Polguère 1988 and Polguère 1990; this presentation uses many of their ideas and results, especially in what concerns the form of these rules. Another source of mine is the outline of the Sem-Module in Kahane & Mel’čuk 1999, in particular as far as the general organization of the module and the format of Sem-rules are concerned (p. 49ff); several ideas of this paper bearing on the formalization of the semantics-syntax interface in the Meaning-Text approach have been developed by S. Kahane (for instance, Kahane 2005 and Kahane & Lareau 2005).

2 Lexicalization Semantic Rules

Lexicalization Sem-rules, or Lex-Sem-rules, are naturally subdivided into lexical Lex-Sem-rules, which produce labels on the nodes of DSynt-trees—i.e., LUs of all kinds (2.1), and inflectional, or grammemic, Lex-Sem-rules, responsible for inflectional subscripts to the node labels, that is, for grammemes (2.2).

2.1 Lexical Lex-Sem-Rules

A lexical Lex-Sem-rule of language \( \mathbf{L} \) establishes the correspondence between a subnetwork ‘\( \sigma \)’ of the Sem(antic) S(tructure) and a Deep LU \( \mathbf{L} \) of \( \mathbf{L} \) that carries the meaning ‘\( \sigma \)’ and labels a node of the Deep-SyntS of the corresponding sentence.

A typical lexical Lex-Sem-rule has the following form: ‘\( \sigma \)’ ⇔ \( \mathbf{L}(\sigma) \mid \mathbf{C} \), where \( \mathbf{L}(\sigma) \) stands for a Deep LU with the meaning ‘\( \sigma \)’, and \( \mathbf{C} \) is the set of contextual conditions under which the
Figure 1: The Linguistic Sem-Transition Rules of the Sem-Module (of the MTM)
[The typology in Fig. 1 covers only Lexical-Syntactic Sem-rules; it does not show other ramifications.]

given correspondence holds. Such a rule is in fact a lexical entry, or a dictionary article, for the LU L. As a result, the set of lexical Lex-Sem-rules of this type is the lexicon of L. (Besides individual lexical Lex-Sem-rules, the Sem-Module uses some very general lexical Lex-Sem-rules based on Sem-Comm-Dominance, which ensure the coherence of Lexicalization, see Mel’čuk 2001: 35, 120, 230; these rules are not considered here.)
A Deep LU L can be either 1) a genuine LU, which ends up in the sentence as a wordform, a phrase of an analytical form or a phraseme (idiom or collocation, the latter described in terms of lexical functions), or 2) a fictitious lexeme, which encodes a meaning-bearing syntactic construction. Consequently, there are two groups of Lexical Lex-Sem-rules: Genuine-Lexical Lex-Sem-rules (2.1.1) and Constructional Lex-Sem-rules (2.1.2).

2.1.1 Genuine-Lexical Lex-Sem-Rules

The huge majority of genuine-lexical Lex-Sem-rules of L are rules for L’s actual LUs. Under synthesis, they specify Deep LUs, i.e., labels on the nodes of the DSynt-tree, which are addresses in L’s lexicon; the number of rules for actual LUs is close to one million.

Along with these rules, languages have (to varying degrees) word-formation rules, i.e., rules for producing potential LUs: compound and derived lexemes that are completely compositional and productive and therefore should not be stored in the lexicon of a linguistic model. (I have no claim about what happens with the lexical storage in the brain of speakers.) Rules of this type are much less numerous: at most a couple of hundred per language.

2.1.1.1 Genuine-Lexical Lex-Sem-Rules for Actual LUs

2.1.1.1.1 General Remarks

A genuine-lexical Lex-Sem-rule for an actual LU establishes the correspondence between the minimal decomposition of a lexical meaning, i.e., a semanteme, σ, and this meaning σ itself. Since a semanteme is, by definition, the signified of an LU of L, a lexical Lex-Sem-rule for an actual LU maps the decomposition of the signified of L—of σ onto L. (This is done in two steps; cf. Lexemic Lex-Sem-rule 1 below.)

Lexical Lex-Sem-rules for actual LUs are of three types, corresponding to three types of LUs that may appear in a DSyntS: lexemic (they produce lexemes), phrasemic (they produce idioms and quasi-idioms), and lexical-functional (they produce standard LF; paradigmatic LFs result in LUs (actual and/or potential), and syntagmatic LFs—in collocations).

Conventions used in writing lexical Lex-Sem-rules

1. As shown in Polguère 1990: 119 ff, it is crucial to indicate in the left-hand side of any Lex-Sem-rule the communicatively dominant node; this is done by underscoring the semantemic label of such a node.
2. In a Lex-Sem-rule, the shaded area indicates the context, that is, the elements not affected by the rule.
3. A Lex-Sem-rule is supplied with the indication of the (near-)obligatory Unitariness of the semanteme configuration σ in its left-hand side (on the Sem-Comm-opposition of Unitariness, see Mel’čuk 2001: 228 ff). This is the degree of obligatoriness with which the configuration σ must be reduced to one semanteme, but not be lexicalized as is, semanteme by semanteme. The Sem-Comm-value ‘Unitary’ in the left-hand side of a Lex-Sem-rule means that this rule—or another rule with the same left-hand side—must apply from left to right if it is applicable to the given SemS, so that σ is maximally reduced; ‘Strongly preferably Unitary’ says that, if the SemS contains the semanteme configuration σ, the latter is most often reduced, although it can remain unreduced under special conditions; etc. The absence of such an indication means that the lexicalization of σ is optional: it can be reduced or remain unreduced. (My thanks to I. Boguslavskij, A. Polguère and E. Savvina for their suggestions concerning the conditions of Lex-Sem-rules and, in particular, the Unitariness of σ.)
2.1.1.2 Lexemic Lex-Sem-Rules

**Lexical Lex-Sem-Rule 1 (lexemic)**

The verb \[\text{to} \ \text{ASSASSINATE}\] (King Alexander was assassinated by Croat nationalists in 1934)

![Diagram of Lex-Sem-Rule 1 (lexemic)]

The proposed form of lexemic Lex-Sem-rules is an abbreviation; Lex Sem-rule 1 as written above represents a sequence of two simpler rules (Polguère 1990: 86-89):

- First, a semantic equivalence rule describes a particular case of the purely semantic operation of reduction/expansion (defined on Sem-networks):

![Diagram of semantic equivalence rule]

\[\text{[X murders Y for a political reason]} \equiv \text{[X assassinates Y]}\]

Such a rule, when applied from left to right, says that a particular subnetwork of the SemS can be reduced to one semanteme (of which this subnetwork is the minimal decomposition); when applied from right to left, it says that this semanteme can be expanded into its shallowest (= minimal) decomposition. This is in fact a rule of Sem-paraphrasing; such rules constitute perhaps the most interesting part of the Sem-Module (cf. Miličević 2007: 181ff on Sem-equivalence, i.e., Sem-paraphrasing, rules).

- Second, a maximally reduced Sem-subnetwork—i.e., one semanteme—is mapped onto the corresponding lexeme:

![Diagram of mapping to lexeme]

For economy’s sake, the dictionary of the Meaning-Text model uses abbreviated lexemic Lex-Sem-rules. Here are two more examples of those.
Semantic Transition Rules

**Lexical Lex-Sem-R 2 (lexemic)**

The verb \([to] \text{BLAST}\) (*The congressman blasted the administration for their mistakes*)

\[
\text{‘intensely’} \rightarrow \text{‘criticize’} \quad \leftrightarrow \quad \text{BLAST(V)}
\]

This rule is optional in the following sense: its left-hand side can be readily expressed as is, i.e., without reduction: *harshly criticized the administration...* In other words, the semantemic configuration in its left-hand side is not necessarily Unitary.

**Lexical Lex-Sem-R 3 (lexemic)**

The verb \([to] \text{AWAKE}\) (*I awoke at night / In a terrible fright*)

\[
\text{‘cease’} \rightarrow \text{‘sleep’} \quad \leftrightarrow \quad \text{AWAKE}
\]

The expression \(*\text{cease to sleep}\) cannot be used to refer to an individual event (‘awake’); it can, however, denote a prolonged state: *Soon I ceased to sleep altogether, insomnia set in.* Thus, under the appropriate conditions, the application of this rule (or of a different rule with the same left-hand side—for instance, a rule that specifies the verb \(\text{WAKE UP}\) as the lexical expression in its right-hand side) is obligatory.

Each lexemic Sem-rule of this type is a particular way of representing a lexical entry.

Lexemic rules include also a different type of rule, namely—rules that state generalizations about LUs of \(L\), for instance:

**Lexical Lex-Sem-R 4 (lexemic)**

Adjectives of color (*red liquid, green roof, yellow dress*)

\[
\text{‘color’} \quad \leftrightarrow \quad \text{L(‘X’)ATTR L(‘P.b’)‘color’}
\]

Any adjective \(A.b\) that refers to a color expresses the meaning ‘[being of] A.a color’: \(red_b\) liquid \(\leftrightarrow\) ‘liquid of red\(_a\) color\(_a\)’, \(green_b\) roof \(\leftrightarrow\) ‘roof of green\(_a\) color\(_a\)’, etc.
2.1.1.3 Phrasemic Lex-Sem-Rules

A phrasemic Lex-Sem-rule describes the meaning of an idiom (or a quasi-idiom) in the same way as a lexemic Lex-Sem-rule describes the meaning of a lexeme: by the decomposition of the corresponding semanteme. In other words, a phrasemic Lex-Sem-rule is also a dictionary entry—for an idiom. The number of such rules is several tens of thousands.

**Lexical Lex-Sem-R 5 (phrasemic)**

The idiom "PULL [Y’s] LEG" (I seriously thought they *were pulling my leg* )

\[
\text{‘Purpose'} \quad \overset{1}{\longrightarrow} \quad \text{‘lie'} \quad \overset{1}{\longrightarrow} \quad \text{‘fun'}
\]

\[
\overset{1}{\text{‘Purpose'}} \quad \overset{2}{\longrightarrow} \quad \overset{1}{\text{‘fun'}}
\]

[‘X is lying to Y with the purpose of having fun at the expense of Y’ ≡ ‘X is pulling Y’s leg’]

Within the Sem-Module, lexemic and phrasemic Sem-rules are similar. The important distinction between them consists in the different treatment of their outputs by DSynt-rules: a lexemic node remains in the SSyntS as is, while a phrasemic node is expanded into the SSynt-subtree of the corresponding idiom.

2.1.1.4 Lexical-Functional Lex-Sem-Rules

A lexical-functional Lex-Sem-rule describes the meaning of a standard lexical function \( f \) and thus specifies an LF-node \( f(L) \) of the DSynt-tree. For a paradigmatic LF, the node \( f(L) \) corresponds to an actual or potential LU that is an element of this LF’s value and appears in the SSyntS. For a syntagmatic LF, the node \( f(L) \) specifies a collocate for the LU \( L \) (= the base of a collocation; on LFs in the DSyntS, see Wanner & Alonso 2005). For instance:

**Lexical Lex-Sem-R 6 (lexical-functional)**

Lexical function \( \text{Magn} \) (an intensifier)

\[
\text{‘intense animo} \quad \overset{1}{\longrightarrow} \quad \text{‘intensely apologize'} \quad \overset{1}{\longrightarrow} \quad \text{‘intensely upset'}
\]

\[
\overset{1}{\text{‘intense animo} \quad \overset{2}{\longrightarrow} \quad \text{‘intensely apologize'} \quad \overset{2}{\longrightarrow} \quad \text{‘intensely upset'}}
\]

\[
\text{SemS} \quad \overset{1}{\text{‘intense animo} \quad \overset{2}{\longrightarrow} \quad \text{‘intensely apologize'} \quad \overset{2}{\longrightarrow} \quad \text{‘intensely upset'}}
\]

\[
\text{DSyntS} \quad \overset{1}{\text{‘intense animo} \quad \overset{3}{\longrightarrow} \quad \text{‘intensely apologize'} \quad \overset{3}{\longrightarrow} \quad \text{‘intensely upset'}}
\]

\[
\text{SSyntS} \quad \overset{1}{\text{‘intense animo} \quad \overset{4}{\longrightarrow} \quad \text{‘intensely apologize'} \quad \overset{4}{\longrightarrow} \quad \text{‘intensely upset'}}
\]
Unlike lexemes and idioms, an LF, because of its special nature, can be ‘polysemous’: it can correspond alternatively to different meanings. Thus, for the LF \( \text{Magn} \) the Sem-Module needs several rules; here is another lexical-functional Lex-Sem-rule for this LF:

**Lexical Lex-Sem-R 7 (lexical-functional)**

\[ \text{L}(\text{X}) \text{ ATTR} \]

\[ \text{Magn} \]

The semanteme ‘\( X \)’ has ‘magnitude’ as the Comm-dominant node of its decomposition, i.e., ‘\( X \)’ is a magnitude: ‘length’, ‘temperature’, ‘price’, etc.; ‘\( \alpha \)’ is the value of the magnitude ‘\( X \)’.

The collocations produced by this rule include *considerable length, high temperature, heavy price* [in the figurative sense], etc.

Lexical-functional Lex-Sem-rules include also more complex (and more interesting) types of rules, where the meaning of the LF in question is linked to the meaning of its keyword via a third meaning (see Iordanskaja & Polguère 2005); it is, however, impossible to consider them here.

2.1.1.2 Lexical Lex-Sem-Rules for Potential LUs

2.1.1.2.1 General Remarks

The lexemic Lex-Sem-rules that describe potential LUs of \( L \) cover all cases of regular word-formation. Potential LUs are not (and most often cannot be) stored in \( L \)’s lexicon; they are constructed by word-formation rules of \( L \) out of its actual LUs. Word-formation functions in two modes:

- Either by using two or more LUs to create a new LU; this is compounding.
- Or by using a special linguistic means—an operator that applies to an LU and derives from it another LU; this is derivation.

As a result, lexemic Lex-Sem-rules for potential LUs form two groups: compounding rules and derivational rules.

Here only 100%-compositional and productive word-formation is considered—compound lexemes (= compounds\(_1\)) and derived lexemes (= derivatives\(_1\)) that are absolutely regular and unrestricted semantically, formally and with respect to their cooccurrence.

2.1.1.2.2 Compounding Lexemic Lex-Sem-Rules

Unlike lexemic Lex-Sem-rules proper, a compounding Lex-Sem-rule is, in point of fact, a rule schema: an abstract formula covering two open-ended sets of particular lexemes that can be united to form a compound\(_1\) lexeme, which occupies one node in the DSyntS. The name of a compound\(_1\) lexeme is included in square brackets (to show its lexemic unity); within the name of a compound\(_1\) lexeme an arrow shows the dependency relation between compounding radicals: in a compound lexeme \( L = L_1 \rightarrow L_2 \) the radical (of) \( L_2 \) modifies the
radical (of) \( L_1 \). This indication is used by morphological rules to position \( L_2 \) with respect to \( L_1 \) and, if need be, to inflect \( L_2 \)—for instance, adding the necessary interfix. (Compounding is recursive, so that the radicals entering in a compound may be compound themselves.)

**Lexical Lex-Sem-R 8 (compounding)**

*German compound nouns of the TEXTVERARBEITUNG ‘text processing’ type*

\[
\begin{array}{c}
\circ \\
1 \downarrow \circ \\
2 \\
\circ \\
\end{array}
\Leftrightarrow
\left[ L(\langle Y \rangle) \lhd L(\langle X \rangle) \right]
\]

1) \( L(\langle X \rangle) \) and \( L(\langle Y \rangle) \) are nouns;  
2) \( GP(L(\langle X \rangle)) \circ \llbracket II \Leftrightarrow N \rrbracket \)

If a German noun \( L(\langle X \rangle) \) has Semantic Actant 2, \( L(\langle Y \rangle) \), which can also be realized as a noun (Condition 1), then instead of the corresponding Deep-Syntactic Actant \( II \) the model can use compounding, provided this possibility is specified in the Government Pattern of \( L(\langle X \rangle) \) (Condition 2).

These and similar compounds are fully productive and current in German. Among other things, they are recursive:

- \( \text{Donau} + \text{schifffahrt} \) lit. ‘Danube navigation’ = ‘navigation on the Danube’
- \( \text{Donau} + \text{schifffahrt} + s + \text{gesellschaft} \) ‘Society for navigation on the Danube [= SND]’
- \( \text{Donau} + \text{schifffahrt} + s + \text{kapitän} \) ‘captain of the SND’
- \( \text{Donau} + \text{schifffahrt} + s + \text{gesellschaft} + s + \text{kapitän} + s + \text{kajüte} \) ‘captain of the SND’s cabin’

**2.1.1.2.3 Derivational Lexemic Lex-Sem-Rules**

A derivational lexemic Lex-Sem-rule establishes the correspondence between a Sem-sub-network and a derivateme, which is to be attached to the appropriate lexeme. A derivateme is associated with a family of derivational means that express it on the surface; this family can be referred to by a conventional name of our choice. (The nature of the derivational means selected—i.e., whether the derivateme in question is expressed by an affix, a replication or a conversion, and then by exactly which affix, replication or conversion—is irrelevant at this stage.)

**Affixal derivational rules**

**Lexical Lex-Sem-R 9 (suffixal derivational)**

*Spanish IMPACT derivation*

\[
\begin{array}{c}
\circ \langle \text{impact} \rangle \\
1 \\
\circ \\
\end{array}
\Leftrightarrow
\left[ L(\langle Z \rangle) \oplus \text{IMPACT} \right]
\]

The derivateme ‘impact of Z [on Y]’ can be expressed in Spanish by the suffix -azo: \( \text{botell} + \text{azo} \) ‘impact of a bottle’, \( \text{bayonet} + \text{azo} \) ‘impact of a bayonet’, \( \text{sill} + \text{azo} \) ‘impact of a chair’, etc.
Lexical Lex-Sem-R 10 (suffixal derivational)

Spanish $\textit{DIMIN}$(utive) derivation

The derivateme $\textit{DIMIN}$ ‘imagined by people as something little [and] nice’ is expressed by the suffixes -it(-o/a), -cit(-o/a), -ill(-o/a) or -ecill(-o/a): mes+it(-a) ‘nice little table’, poem+it(-a) ‘nice little poem’, noche+cit(-a) ‘nice little night’, mujer+cit(-a) ‘nice little woman’, arbol+it(-o)/arbol+ill(-o) ‘nice little tree’, flor+ecill(-a) ‘nice little flower’, etc. The choice of a specific suffix is left to the DMorph-Module.

Lex-Sem-rule 10 is almost obligatory: if the SemS ‘$S$’ contains the semanteme configuration of its left-hand side, it has to apply—except for some very special conditions; under normal conditions the said configuration cannot be expressed lexically. But the presence itself of the diminutivity configuration with a semanteme is not obligatory: it is not the case that each noun in Spanish has to feature a marker of diminutivity: if ‘$S$’ does not contain the triggering semanteme configuration, the Sem-Module need not bother to look for necessary chunks of meanings—as is the case with the inflectional meanings.

Conversional derivational rules

Lexical Lex-Sem-R 11 (conversional derivational)

Spanish $\textit{TREE}$ derivation

In Spanish, the name of a tree that bears fruits Ys is formed in most cases by the conversion ‘fem(inine) ⇒ masc(uline);’ this conversion changes the gender mark in the syntactics of the radical denoting the fruit and replaces accordingly its gender-marking suffix:

- naranja $\text{[fem]}$ ‘orange’ $\sim$ naranjo $\text{[masc]}$ ‘orange tree’
- manzana $\text{[fem]}$ ‘apple’ $\sim$ manzano $\text{[masc]}$ ‘apple tree’
- ciruela $\text{[fem]}$ ‘plum’ $\sim$ ciruelo $\text{[masc]}$ ‘plum tree’

This conversion is rather productive; it applies even to the names of exotic fruits, like in papaya $\text{[fruit]}$ $\sim$ papayo $\text{[tree]}$ or guayaba $\text{[fruit]}$ $\sim$ guayabo $\text{[tree]}$. 
Lexical Lex-Sem-R 12 (conversional derivational)

English CAUSE_MOVE derivation: derived lexemes of the type SNEEZE the napkin off the table, DANCE her out of the room, SCARE John into hiding, etc.

The meaning ‘X’s doing P which affects Y causes1 that Y moves from Z to W over T’ can be expressed in English by a ‘new’ lexeme L(\(P\))\(^n^o\) of the vocable \(L(\(P\))\), productively derived by conversion.

Comments
1. The variables Z, W and T stand for the starting point, endpoint and trajectory of Y.
2. The symbol \(n^o\) in the name of the derived (potential) lexeme L(\(P\))\(^n^o\) represents its lexicographic number, which is different from the numbers that have all actual lexemes of the vocable L(\(P\)), to which this L(\(P\))\(^n^o\) belongs. (SCARE in SCARE John into hiding and SCARE in SCARE John are different lexemes.)

2.1.2 Constructional Lex-Sem-Rules

A constructional Lex-Sem-rule establishes the correspondence between a Sem-subnetwork and the corresponding fictitious Deep lexeme—an artificial symbol introduced by the researcher in order to represent in the DSyntS a syntactic construction that carries a meaning of lexical type. A stock example is the Russian approximate-quantitative construction ‘N + NUM,’ in which the anteposition of the noun with respect to the numeral expresses the meaning \(\approx\) (the uncertainty of the Speaker as for the exact quantity):

dvadcat’ kilo  ‘20 kilos’ \(\sim\) kilo dvadcat’  ‘maybe 20 kilos’
dlja sta čelovek  ‘for 100 people’ \(\sim\) čelovek dlja sta  ‘for maybe 100 people’

In the SSyntS this construction is represented by a special Russian SSyntRel, which sets it apart from the ‘normal’ quantitative construction:

\[\text{DVADCAT} \leftarrow \text{quantitative–KILO} 20\text{ kilos} \sim \text{dvadcat’ kilo} \text{ } \sim \text{dkvadcat’ kilo} 20\text{ kilos} \sim \text{dvadcat’ kilo}\]

In the DSyntS this construction is represented by a fictitious lexeme «APPROXIMATELY».

An example of constructional Lex-Sem-rule can be given based on the ‘N by N’ construction of English:

1. Page by page, you feel that something is taking shape in your mind.
2. The police searched the neighborhood house by house.
3. Step by step he built his argument.
The boldfaced expressions mean, roughly, ‘addressing one N, then the other N, then the third N, etc.’. ‘Addressing’ is to be interpreted here in a vague sense of ‘doing with N what the context indicates is to be done with N; and ‘one N, then the other N, then the third N, etc.’ can be represented by an operator semanteme ‘consecutively’ \(\approx\) ‘one after the other’. Let us designate this meaning with a fictitious lexeme «BYnº» (this is a special sense of the preposition BY that marks the construction in question); then the following constructional rule can be written:

**Lexical Lex-Sem-R 13 (constructional)**

\[
\begin{array}{c}
\circ \, \text{(P)} \\
\downarrow \\
1 \\
\end{array}
\quad \text{‘consecutively’} \\
\quad \text{‘addressing’} \\
\begin{array}{c}
\circ \, \text{(Y)} \\
\downarrow \\
2 \\
\end{array}
\begin{array}{c}
\downarrow \\
\circ \, \text{set_and} \\
\end{array}
\quad \iff
\begin{array}{c}
\circ \, \text{(P)} \\
\downarrow \\
1 \\
\end{array}
\begin{array}{c}
\downarrow \\
\text{L(Y) pl} \\
\end{array}
\]

**Comment**

SemAs of the semanteme ‘set_and’ are not numbered, since they are, so to speak, equal and their number is unlimited; see also Inflectional Lex-Sem-Rule 2 below.

### 2.2 Inflectional Lex-Sem-Rules

An inflectional, or grammemic, Lex-Sem-rule describes a correspondence between a subnetwork \(\sigma\) and a Deep grammmeme \(g\) that carries the meaning \(\sigma\) and is subscripted to the LU that labels the node of the DSyntS of the corresponding sentence and whose signified is characterized by \(\sigma\). An Infl-Sem-rule has the following form:

\[
\begin{array}{c}
\circ \, \text{(X)} \\
\downarrow \\
1 \\
\end{array}
\quad \iff
\begin{array}{c}
\circ \, \text{L(X) pl} \\
\end{array}
\]

The inflectional Lex-Sem-rules were first formalized—under the name of Grammatical-Semantic Rules—in Polguère 1990: 93ff.

Inflection is obligatory: a grammeme of the corresponding inflectional category must accompany the LU that is its target. Therefore, all Infl-Lex-Sem-rules are applied obligatorily. Note that, like an LF, a grammeme can correspond to several meanings, i.e., have several semantic sources. Here are examples: three Infl-Lex-Sem-rules for the Spanish nominal plural.

**Inflectional Lex-Sem-rule 1**

\[
\begin{array}{c}
\circ \, \text{more.than.one} \\
\downarrow \\
1 \\
\end{array}
\quad \iff
\begin{array}{c}
\circ \, \text{L(X) pl} \\
\end{array}
\begin{array}{c}
\text{L(X) is a noun} \\
\end{array}
\]

This rule describes ‘normal’ nominal plural: *ciudad‘city’ ~ ciudad\(+\)es ‘cities’, *hombre‘man’ ~ hombre\(+\)s ‘men’, *casa‘house’ ~ casa\(+\)s ‘houses’.
Inflectional Lex-Sem-rule 2

This rule produces a special nominal plural that refers to a married couple: rey ‘king’ \(\sim\) rey+es ‘the king and the queen’, padre ‘father’ \(\sim\) padre+s ‘father and mother’ \(\sim\) ‘parents’, tío ‘uncle’ \(\sim\) tio+s ‘the uncle and his wife’, abuelo ‘grandfather’ \(\sim\) abuelo+s ‘grandfather and his wife’.

Inflectional Lex-Sem-rule 3

This is another special nominal plural: it gives rise to names of enterprises (pluralia tantum) such as hormigón ‘concrete’ \(\sim\) Hormigon+es Chávez ‘Chávez’ Concrete Production and Supply, harina ‘flour’ \(\sim\) Harina+s Hidalgo ‘Hidalgo’s Flour Production and Supply’, aceite ‘olive oil’ \(\sim\) Aceite+s Alonso ‘Alonso’s Olive Oil Production and Supply’.

3 Arborization Semantic Rules

Arborization Sem-rules perform two major tasks: they determine the top node of the DSyntS of the sentence being produced and establish its branches, specifying their orientation and labeling. As a consequence, two types of such rules are distinguished: Top-node Arbor-Sem-rules (3.1) and Branch Arbor-Sem-rules (3.2).

3.1 Top-Node Arbor-Sem-Rules

Syntacticization of a SemS is formally a transition from a network to a dependency tree, i.e., arborization. For the construction of a tree the crucial decision is the choice of its top node; therefore, the arborization of a SemS presupposes the determination of the entry node in the starting SemS—a semantic node that will produce the top node of the DSyntS being synthesized. The rules for the determination of the entry nodes are presented in Mel’čuk 2001: 38-48; therefore, I will not dwell on them here.

3.2 Branch Arbor-Sem-Rules

A Branch Arbor-Sem-rule associates an arc of the SemS with an arc (= a branch) of the DSyntS; it thus ensures the ‘translation’ of a Sem-dependency relation, in a specific semantic and lexical context, by a DSynt-dependency relation (see Polguère 1990: 123ff and Kahane & Mel’čuk 1999: 53ff). Here are five examples of Branch Arbor-Sem-rules (they are no more than an illustration, so that their conditions are not properly stated; more complex—discontinuous—Arbor-Sem-rules, needed, among other things, for extractions, are not given: see Kahane & Mel’čuk 1999: 58ff).
Branch Arbor-Sem-rule 1

Expression of SemA 1 of a semanteme implemented by a non-passive finite verb

\[ (X') \xrightarrow{I} (Y) \]

\[ L(X')(Y) \]

1) \( L(X') \) is a finite verbal form;
2) if \( L(X') \) is a verb that has voices, then it is in the active voice

SemA 1 of a Comm-dominant semanteme ‘\( X \)’ can be realized by DSyntA I of the LU L that expresses ‘\( X \)’, if the verb \( L(X') \) is in a finite form and does not distinguish voices; if, however, \( L(X') \) is a verb having different voice forms, it must be in the active voice. (In other words, it must be in the basic—lexicographic—diathesis.)

Examples

'John←I–sleep' ⇔ JOHN←I–SLEEP (John sleeps/is sleeping/…)

'John←I–write' ⇔ JOHN←I–WRITE (John writes/is writing/…)

This rule and the next one are controlled by the Sem-CommS; for simplicity’s sake, the corresponding conditions are not mentioned.

Branch Arbor-Sem-rule 2

Expression of SemA 1 of a semanteme implemented by a passive verb

\[ (X') \xrightarrow{I} (Y) \]

\[ L(X')(Y)_{\text{passive}} \]

SemA 1 of a semanteme implemented by a passive verb is expressed by this verb’s DSyntA II. This rule must of course be supplied with complex semantic and communicative conditions under which a semanteme can or has to be realized as a verb in the passive. (Here, the verb \( L(X') \) need not be in a finite form.)

Example

'John←I–write' ⇔ JOHN←II–WRITE_{\text{passive}} (is written by John)

Branch Arbor-Sem-rule 3

Expression of SemAs 1 and 2 of a semanteme implemented by a reflexive verb (in French)

\[ (X') \xrightarrow{I} (Y) \]

\[ L(X')(Y)_{\text{reflexive}} \]

The referentially identical SemAs 1 and 2 of a semanteme are expressed by DSyntA I of the verb that implements this semanteme; the verb receives the grammeme of the reflexive.
Example
Fr. ‘John←1→shave’ ⇔ JOHN←1→RASERreflexive (John se rase ‘John is shaving’)

Branch Arbor-Sem-rule 4
Introduction of an adjectival modifier

If SemA 1 ‘Y’ of a semanteme ‘X’ is Comm-dominant and lexicalized as a noun, then ‘X’ can be lexicalized as an adjective, L(‘Y’) being its DSynt-governor via DSyntRel ATTR. The discussion of this type of Branch Arbor-Sem-rules—with the inversion of the DSynt-dependence with respect to the Sem-dependence—is found in Polguère 1990: 139-146.

Examples
‘dress←1→green’ ⇔ DRESS←ATTR→GREEN (green dress)
Fr. ‘robe←1→green’ ⇔ ROBE←ATTR→VERT (robe verte)

Branch Arbor-Sem-rule 5
Introduction of a parenthetical

If SemA 1 ‘Y’ of a semanteme ‘X’ is Comm-dominant and lexicalized as a verb, while ‘X’ is lexicalized as a parenthetical adverb, then L(‘Y’) becomes its DSynt-governor via DSyntRel APPEND(itive).

Example
‘unfortunate←1→leave←1→Mary’ ⇔ UNFORTUNATELY←APPEND←LEAVE←1→MARY
(Unfortunately, Mary left)

Conclusion
On the one hand, this paper presents a (hopefully) exhaustive typology of the Sem-Transition rules of a Sem-Module, Meaning-Text style. Each type of rule described may contain subtler divisions, but—barring major errors—any new rule will fall into one of the divisions foreseen by the typology. On the other hand, a strict distinction is drawn between purely linguistic static rules and general procedural dynamic rules; this might facilitate the elaboration of Sem-rules by linguists, making these rules linguistically more transparent.
As for future research, two specific points to be developed are already evident:

- The notion of a fictitious lexeme, the marker of a meaning-carrying syntactic construction, has to be made more precise.
- The types of Unitariness of the semanteme configuration in the left-hand side of a lexical Sem-rule (obligatory, strongly preferable, preferable, etc.) as well as the corresponding conditions need deeper theorization, including a more rigorous interpretation of the way a Sem-rule has to or can be applied following the specification of the Unitariness of its left-hand side.

Acknowledgments

The paper has been read and commented upon by M. Alonso Ramos, D. Beck, L. Iomdin, L. Iordanskaja, S. Kahane, J. Milićević, A. Polguère, and L. Wanner. I most cordially thank them all for their remarks, criticisms and suggestions, which allowed me to substantially improve the presentation.

Bibliography


