Negative concord in English and Romance: syntax-morphology interface conditions on the expression of negation
Tubau Muntañá, s.

Citation for published version (APA):
# Theoretical framework

## Introduction

This chapter is expository in nature, since a number of theoretical devices that are crucial to understand the characterisation of NC offered in forthcoming chapters are introduced. The theoretical framework where the syntactic analysis is embedded is the latest development of the Principles and Parameters model (P&P), known as the Minimalist Program (MP) (Chomsky 1993, 1995, 2000, 2001 and 2005). The MP, which is a program for research rather than a theoretical model proper, is enhanced with the assumptions and machinery of Distributed Morphology (DM) (Halle and Marantz 1993; Embick and Noyer 2007).

The chapter is organised as follows: section 2.2 presents a general overview of the main assumptions of the P&P framework, while section 2.3 contains an outline of the MP with special emphasis on those aspects of the model that are relevant for this dissertation. Section 2.4 summarises the main assumptions of DM, a non-lexicalist theory of the syntax-morphology interface which assumes that only one generative system exists to construct words, phrases and clauses. In that sense, DM crucially diverges from the MP in assuming that a Lexicon of the kind described in section 2.3 does not exist. Rather, the functions of the Lexicon in the MP are distributed – hence the name of the model– through the other components of the T-model of grammar that both the MP and DM share.

Another salient difference between the MP and DM is the role of the PF interface, which is more active in the latter than it is in the former. In DM, PF mediates between syntactic and morphological structure. That is, the output of syntax can be affected by a number of PF operations that might result in syntax-morphology mismatches. Section 2.5, finally, contains a summary of the chapter.

## The Principles and Parameters model

The P&P model of the 1980s was the outcome of a series of ideas about language that started materialising fifty years ago with Chomsky’s (1957) seminal work. Since then, generative grammar has focused on describing and characterising the faculty of language (FL) –a component of the mind especially devoted to language and genetically determined for all human beings. FL has an initial state that seems to be uniform across the human species and various attained states that result from the individual’s exposure to linguistic input. The theory of the initial state of FL is known as Universal Grammar (UG). Together with particular grammars, which is
how the attained states of FL are referred to, they are the two objects of study of the P&P approach.

In generative grammar a distinction exists between *I*-language and E(xternalised)-language. While the former is a technical term used to refer to the knowledge that makes it possible for human beings to produce, understand and make intuitional judgements on a language they are familiar with, the latter captures all those aspects of language that do not belong to I-language and plays almost no role in linguistic generative theory. The ‘I’ in *I*-language stands for (i) internal, since linguistic knowledge is mentally represented in our brain; (ii) individual, since knowledge of language is internalised in each individual’s mind and (iii) intensional due to the fact that the computational system that is part of our grammar contains important formal features and interprets everything it is exposed to. Speaking about language in generative grammar, therefore, amounts to talking about I-language.

At the heart of the P&P approach lies the tension between descriptive and explanatory adequacy. A linguistic theory is said to be descriptively adequate if it succeeds in capturing the native speaker’s linguistic intuitions, while explanatory adequacy is met if a linguistic theory can account for how the speaker acquired a particular grammar.

In order to be both descriptively and explanatorily adequate, comparable effort has been devoted in the P&P framework to the research on both the characterisation of particular grammars and the formulation of a theory of UG. This is why the P&P approach has focused on two types of linguistic properties: a universal set of principles which are common to all natural languages, and parameters, which are language-specific and vary across languages according to a number of choices that UG makes available. These two notions have proved very useful to describe the inner workings of particular grammars and to account for cross-linguistic variation. In addition, the P&P model also provided researchers with descriptive and analytic tools to explain the observed uniformity in the process of acquisition of different particular grammars, and to characterise the various stages children go through when attaining their adult grammar, which is understood as a process of parameter-setting.

### 2.3 The Minimalist Program

The MP, which has guided generative syntactic research since the beginning of the 1990s, is not a theory, as Chomsky (2000, 2005) remarks. Rather, it is a set of guidelines that should ideally lead to a maximal simplification of the theoretical apparatus used to describe and account for linguistic data.
In subsection 2.3.1, the basic model of grammar assumed for the derivation of linguistic structures is presented. Subsection 2.3.2 is devoted to the property of displacement and the interpretability of features, which are two issues that go hand in hand. Merge, Move and Agree, which are the three core syntactic operations, are addressed in subsection 2.3.3, while the concept of phase, introduced in Chomsky (2000), is explored in subsection 2.3.4.

2.3.1 Minimalist grammar

In an attempt to establish what are the minimum conditions that make generative linguistic theory adequate, one of the main Minimalist concerns is to address the question of how well-designed human language is (Chomsky 1995, 2000, 2001 and 2005). A new line of research opens in trying to substantiate the idea that language may be designed in an optimal way to satisfy the legibility conditions imposed by the interfaces (Chomsky 2000: 93); in Chomsky’s later work (2001, 2005), the Strong Minimalist Thesis (SMT) is formulated.

(1) Strong Minimalist Thesis
Language is an optimal solution to the interface conditions that FL must satisfy.

In order to fully understand the SMT, more has to be said at this point about the main assumptions that sustain the MP apparatus and the mechanisms that are claimed to make the derivation of linguistic structures possible. In line with the P&P model of the 1980s and 90s, human beings are claimed to be biologically endowed with FL, a component of the mind devoted to language. This approach to the nature of language emerged 50 years ago with the generative grammar tradition and is nowadays referred to as the biolinguistic perspective (Chomsky 2001, 2005). Such a label captures the assumption that FL is a ‘language organ’ (Chomsky 2000: 90) or an ‘organ of the body’ (Chomsky 2005: 1) that interacts with other systems of the organism.

The initial state of FL –UG– is assumed to be some shared genetic endowment which allows us to acquire different states of FL, namely, the various languages that exist in the world. The characterisation of UG and the study of particular grammars are primary concerns of a theory of language according to the generative perspective and should, according to Chomsky (2001: 2), be guided by the Uniformity Principle in (2).

(2) Uniformity Principle
In the absence of compelling evidence to the contrary, assume languages to
be uniform, with variety restricted to easily detectable properties of utterances.

The principle in (2) formalises the assumption that cross-linguistic variation is not random and unpredictable, but constrained by UG, which is made up of principles common to all human languages, and parameters, which are the different options that are available for a given principle. In MP, the locus of parameterisation is to be found in the Lexicon (Chomsky 1995 and subsequent work), with the features that lexical items (LIs)\(^{25}\) contain being the only elements used for computation (Chomsky 2005: 2).

In the MP, it is assumed that language (L) includes a cognitive system that stores information about sound and meaning. L provides this information to the performance systems, external to FL\(^{26}\), in the form of levels of representation or interfaces. Each of the two performance systems, namely the sensorimotor system (S-M)\(^{27}\) and the conceptual-intentional system (C-I), accesses one of the interface levels: the S-M accesses the Phonetic Form (PF)\(^{28}\), where sound is represented, and the C-I accesses the Logical Form (LF), the level of representation that deals with meaning. Within this model, therefore, L is regarded as a generative device that produces expressions \(\text{Exp} = \langle \text{Phon}, \text{Sem} \rangle\) (Chomsky 2000: 91, Chomsky 2001: 10), Phon being instructions for the S-M, and Sem being instructions for the C-I. These instructions are given in the form of features that are interpreted at PF and LF.

Returning to the issue that opened the present section, namely the SMT, it is hypothesised that FL is well-designed to the extent that it is an optimal solution to the legibility conditions imposed by the interaction of FL with S-M and C-I systems of the mind. In other words, for SMT to hold, FL must be the best possible design to

\(^{25}\) In section 2.4, the notion of LI is replaced by the term morpheme, and a distinction between Roots (i.e. lexical categories) and abstract morphemes (i.e. functional categories) in terms of the kind of features they contain is introduced. In DM, not all elements of the Lexicon are sound-meaning pairs. Actually, it is assumed that functional categories, which only contain grammatical features, are assigned phonological features during the course of the derivation. Such a view, however, is not incompatible with the assumption that the locus of variation is in the Lexicon, though the conception of the latter needs to be modified.

\(^{26}\) Performance systems are assumed to be part of FL in Chomsky (1995, 1998), but are claimed to be external to FL in Chomsky (2000, 2001 and 2005).

\(^{27}\) Articulatory-perceptual in Chomsky (1995).

\(^{28}\) DM offers a different view of PF. As discussed in section 2.3, the PF-interface is understood as ‘a sequential derivation that terminates in a phonological representation’ (Embick and Noyer 2007: 293). That is, operations can take place post-syntactically, having an effect on the final phonological representation of a given morpheme. DM’s conception of PF is very much related to the assumption that not all morphemes are pairs of sound and meaning: abstract morphemes (i.e. functional categories) do not have phonological features until the derivation is completed, their phonological features being assigned post-syntactically at PF on the basis of the syntactic and morphological context in which the morpheme is inserted.
satisfy PF and LF requirements. This is why interface-related issues become central to the MP.

For the SMT in (1) to be maintained, the Exp that L generates must contain only information that can be read at the interfaces. Otherwise, the derivation crashes (i.e. it cannot be successfully computed at the interfaces). If a given Exp only contains instructions of the relevant type that can be read at PF and LF, Exp is said to converge.

Some theory-internal terminology must be introduced before proceeding. *Features* of lexical items (LIs), which have been defined as instructions for the interfaces that make computation possible, can be interpretable or uninterpretable (Chomsky 1995 and subsequent work). While the former are legible to S-M and C-I systems at the interfaces, the latter are not. This amounts to saying that only those Exp that contain solely interpretable features will satisfy the legibility conditions of the interfaces. Uninterpretable features, consequently, constitute a point of departure from the SMT, since they make the design of L less optimal with respect to the conditions imposed by PF and LF.

Uninterpretable features are taken to be imperfections which need to be accounted for. Likewise, displacement, which is a property of language that has been the subject of extensive research in generative grammar, is also considered an imperfection of L, since it allows LIs to be pronounced in a different position from the one in which they have been generated. This issue will be taken up again in section 2.3.2, where it will be shown that it is thanks to uninterpretable features that displacement can be implemented.

Let us address now the issue of how Exp are generated and how their featural content is determined according to Chomsky (1995 and subsequent work). The assumption is that UG makes available a universal set of features, $F$, and a computational procedure for human language, $C_{HL}$, which accesses $F$ and performs operations to derive linguistic expressions. Each language L maps these universal features to a given set of Exp by selecting a subset $[F]$ of the universal set $F$ and assembling $[F]$ to a Lexicon, Lex. In other words, L selects a lexical array (LA), or a *numeration* if some LIs are selected more than once (Chomsky 2001: 10-11), which is then mapped to Exp. These are one-time operations in the sense that, once $[F]$ has been selected, L does not access $F$ anymore; likewise, once Lex has been constructed, no new assembly of features $[F]$ into LIs of Lex is possible. While in the narrow syntax (i.e. the computation of LF) only Lex and the features assembled in the LIs it contains, but not $[F]$, are accessed, this seems not to be the case for the computation at PF, where new features can be introduced in the course of the derivation.

As Chomsky (2001: 101) puts it, L uses (3a) and (3b), together with a process of parameter-setting, to determine particular languages, and (3c) and (3d) to generate
Exp in those languages. The operations that the C_{HL} performs with Lex to generate linguistic structures are described and discussed in subsection 2.3.3.

(3)  
   a. Select \([F]\) from the universal feature set \(F\).
   b. Select Lex, assembling features from \([F]\).
   c. Select LA from Lex.
   d. Map LA to Exp, with no recourse to \([F]\) for narrow syntax.

(Chomsky 2000: 101)

At this point, we turn our attention to how Lex, the Lexicon, is characterised in the MP. In Lex, which can be defined as a mental dictionary, LIs are stored as bundles of features which can be phonological, semantic or formal. The phonological and semantic features of LIs encode information for these LIs to be interpreted at PF and LF respectively. Formal features (FFs), by contrast, carry information that is relevant for syntactic computation.

As discussed in section 2.4, not every model within the generative tradition accepts the existence of a Lexicon as described herein. In DM (Embick and Noyer 2007) for instance, no generative Lexicon is assumed to exist. Rather, the relevant information is stored in three different kinds of lists (each containing a different type of information) that are accessed at different stages of the derivation. This issue is taken up again in section 2.4.1.

Returning to the distinction between interpretable and uninterpretable features that has been introduced earlier, it must be clarified that it refers only to FFs. These are considered interpretable if they have a semantic interpretation; if they do not, they are uninterpretable. Phi-features (number, person and gender) are a good example of FFs. In English, phi-features are interpretable on nouns, but uninterpretable on verbs. To put it in another way, the plural morpheme \(-s\) in \(birds\) imposes a difference in meaning with respect to the singular \(bird\), hence the phi-features of this noun must be interpretable. However, phi-features of the verb \(sing\) must be uninterpretable, since \(sing\) can be interpreted either as a singular or a plural form as in \(I\ sing\) and \(they\ sing\), respectively.

To finish with this section, let us point at a number of consequences brought about by the adoption of the SMT as a guide for linguistic research. If language is assumed to be an optimal solution to some minimal design specifications, it is natural to assume that FL provides us with the simplest mechanisms that are needed to comply with legibility conditions. Any extra machinery constitutes a deviation from the SMT and needs, therefore, to be carefully scrutinised. This is the reason why, contrary to what is the case in the P&P model of the 1980s and early 90s, the only linguistically significant levels in the MP are PF and LF, the interface levels. Other
levels of representation that have been proposed in the generative literature such as Deep-structure and Surface-structure are not required anymore, since the linguistic facts they tried to explicate can be accounted for relying only on the PF and LF interfaces. The model of grammar defended in the MP is represented in (4), Spell-Out being the point where LF-uninterpretable features are removed from the derivation and syntactic objects are sent to PF in a very strict cyclic fashion (Chomsky 2001: 5), as will be seen in subsection 2.3.4.1.

(4)  

<table>
<thead>
<tr>
<th>Lexicon</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numeration</td>
</tr>
<tr>
<td>Spell-Out ——— Phonetic Form (PF)</td>
</tr>
<tr>
<td>Logical Form (LF)</td>
</tr>
</tbody>
</table>

The Inclusiveness Condition also plays a major role in the MP in its attempt to stick to the SMT. As Chomsky (1995) puts it

‘A “perfect language” should meet the condition of inclusiveness: any structure formed by the computation (...) is constituted of elements already present in the lexical items selected for N[umeration]; no new objects are added in the course of computation apart from rearrangements of lexical properties (in particular, no indices, bar levels in the sense of X-bar theory, etc; (...)’

(Chomsky 1995: 228)

Inclusiveness only allows rearrangement of LIs and of those syntactic objects that have been constructed during the derivation. It also permits deletion of features, but under no circumstances does it allow the \( C_{HL} \) to introduce new features to the derivation once Lex has been accessed to select an LA. The fact that elements that cannot be traced back to the Lexicon are not part of the derivation greatly simplifies the formal apparatus and forces linguists to find new ways to account for the linguistic evidence without resorting to artefacts.

However, as discussed in section 2.4, syntax-morphology mismatches are indeed attested across languages which might be difficult to explain within a model that is not open to the possibility that further operations take place at the interfaces. DM, by attributing a dynamic role to the PF interface, is particularly successful in accounting for a number of puzzling data where the output of narrow syntax does
not strictly correspond to what is finally uttered. Together with DM’s conception of
the Lexicon as distributed through the various components of the model of grammar,
PF operations of the kind proposed in DM allow us to provide a uniform account of
the expression of negation in various languages.

The DM framework can naturally capture the differences that exist in the featural
composition of the items that participate in NC across languages, which, as
discussed in further chapters, is central to accounting for their distribution in NC and
polarity constructions. In addition, since negation and the expression of NC are
domains where a number of puzzling mismatches between syntax and morphology
occur, a model that makes it possible to seek a principled explanation in a number of
constrained PF operations is especially welcome.

2.3.2 Displacement and features

Chomsky (2000: 112) proposes to look for ‘imperfections’ of language in order to
determine where the SMT fails. He points at the phonological component as being
one of the most obvious ‘design flaws’ (Chomsky 2000: 117, Chomsky 2001: 3) of
L. The reason for such an assumption follows from the fact that syntactic objects
that have been constructed by the computational operations CHL are transformed into
PF-representations in the phonological component, which results in the violation of
the principles of Inclusiveness and Interpretability.

The Inclusiveness Condition has been mentioned at the end of the previous section,
but nothing has been said so far about the Interpretability Condition\(^{29}\), which
establishes that LIs have no features other than those interpreted at the interface
(Chomsky 2000: 113). This condition is straightforwardly false, as Chomsky
acknowledges: for instance, features such as structural case in nouns or phi-features
in verbs are uninterpretable.

Recapitulating, Chomsky (2000: 17) argues that the phonological component not
only violates Inclusiveness, but also the Interpretability Condition, since there is a
mismatch between the phonological properties of LIs and the narrow phonetic
realisations of these when combined in linguistic productions. Inclusiveness is
violated because the phonological component allows operations that introduce new
elements such as prosodic structure and narrow phonetics during the course of the
derivation.

\(^{29}\) In Chomsky (1995), this condition is known as the Full Interpretation Principle.
Chomsky suggests that it may be the case that phonological features of LIs do not appear at the level of PF, with ‘input’ and ‘output’ of the phonological component being separate. In this case, while the Interpretability Condition would still be violated, the Inclusiveness Condition would not apply. This leads Chomsky (2000: 118) to assume that Inclusiveness only holds of narrow syntax. That is, features are either interpreted at LF or associated with phonetic features by the phonological component.

Two other obvious imperfections are observed in narrow syntax: the dislocation property and the uninterpretable features of LIs. The remainder of the section will be dedicated to explaining two crucial ideas for the model of grammar assumed in the MP. First, it will be argued that these two issues are closely related, one crucially relying on the other and, second, that they are not real imperfections, but a device imposed by the design specifications of FL.

Displacement is a property of natural language by which certain elements in a derivation might surface in a position which is different from the one they have been generated in. As illustrated in (5), a wh-phrase such as which books has been generated as the object of the main verb, as shown by the trace. However, it moves to a fronted position in direct questions yielding a chain {which books, t}, where which books c-commands the trace that has been left behind. A relation of identity between the moved element and its trace allows the chain to be correctly interpreted at LF. In other words, both which books and t are taken to be occurrences of the same element at the semantics interface.

(5)  
a. Mary [bought Tom Sharpe’s books]  
b. Mary [bought which books]  
c. Which books did Mary buy t ?

According to Chomsky (2000: 120) neither uninterpretable features nor the dislocation property are part of special-purpose symbolic systems, which leads to the assumption that they are related to externally imposed legibility conditions. To put it in another way, both phenomena can be seen as design-specifications built into the linguistic system so that it can be optimal to the interface conditions that the other systems external to language impose.

When trying to account for the mechanisms that make displacement possible, Chomsky suggests that it should be considered whether this property is related to uninterpretable features, the other observed imperfection of the system. In the event that the two phenomena are related, two imperfections would reduce to one, namely the dislocation property.

Based on the observation that uninterpretable features must be deleted in the course of the computation to LF for a derivation to converge, Chomsky (2000: 121, 2001:
4) assumes that displacement and uninterpretable features are indeed closely related, the latter being what allows the former to be implemented.

‘Displacement is implemented by selecting a target P and a related category K to be moved to a position determined by P –P a probe that seeks K. The target/probe P determines the kind of category that can be moved to this position (a nominal phrase, a wh-phrase, etc.). If uninterpretable inflectional features are the devices that implement displacement, we expect to find uninterpretable features of three kinds:

(2) a. To select a target/probe P and determine what kind of category K it seeks
   b. To determine whether P offers a position for movement
   c. To select the category K that is moved’

(Chomsky 2001: 4)

In a sentence such as (6) below, the three different kinds of uninterpretable features Chomsky mentions can be observed. The example corresponds to a passive construction, where the DP, a rocket, has been generated in an object position, but ultimately surfaces as the subject of the clause thanks to movement driven by uninterpretable features.

(6) [TP A rocket [T was sent t to Mars]]

Let us assume that the category T bears uninterpretable agreement features, also known as phi-features, which identify T as a target for movement. In addition, the phi-features on T seek for matching phi-features in a category K. Since the DP a rocket hosts those matching phi-features, a relation of Agree (see subsection 2.3.3) can be established, which causes the subsequent elimination of uninterpretable features from the derivation. The EPP-feature of T determines that T offers a position for movement; finally, the structural case features on K, a rocket, are responsible for its movement to Spec, TP.

---

30 This assumption will be modified in subsection 2.3.4 after the discussion of phases; however, this has been the standard explanation for subject raising and passivisation until very recently.

31 In former versions of the P&P model, EPP stands for the Extended Projection Principle, which is a requirement for all clauses to have a subject. In the MP, EPP is considered a feature of functional categories which determines whether these categories can project a Spec position or not.
Up to this point, it has been assumed that, in line with Chomsky (1995 and 2000), uninterpretable features are simply erased or eliminated in the course of the computation because, otherwise, they cannot be read at the interfaces and the derivation crashes. In Chomsky (2001), nonetheless, a substantial change is introduced by assuming that ‘uninterpretable features, and only these, enter the derivation without values, and are distinguished from interpretable features by virtue of this property’ (Chomsky 2001: 5). Unvalued features need to be valued by means of Agree in the narrow syntax. Once valued, they are indistinguishable from interpretable features at LF, but can be accessed by the phonological component. Valuation must take place before Spell-Out, which is the operation in charge of transferring the derivation to the phonological component once the material that is uninterpretable at LF has been eliminated. If any unvalued features remain in the derivation at the moment Spell-Out applies, the derivation crashes.

Another difference with respect to earlier work in generative grammar concerns the distinction between overt and covert movement, which was maintained throughout the P&P model of the 1980s and 90s, as well as in Chomsky (1995), but which is dispensed with in more recent Minimalist work. Movement was assumed to take place either before or after Spell-Out: if it took place before, movement was overt and its effects could be observed in the linearisation of the elements of the derivation; if it applied after Spell-Out, movement was covert and had only interpretive effects. In Chomsky (2000, 2001, 2005), by contrast, there is only one narrow-syntactic cycle which proceeds in parallel with the phonological cycle and, therefore, uninterpretable features cannot be valued after Spell-Out, which is understood as strictly cyclic. I will expand on this issue in subsection 2.3.4, as part of the discussion on phases.

To conclude, in this section it was shown that the two main ‘imperfections’ of a system that is assumed to be optimal to the legibility conditions of the interfaces can be reduced to only one, the property of dislocation that is observed in natural language. Uninterpretable features have been claimed to implement displacement in the sense that they encode information on which elements can be moved and to which positions they can be moved.

### 2.3.3 Syntactic operations: Merge, Move and Agree

The three main operations that apply in the narrow syntax to derive linguistic expressions are Merge, Move and Agree. According to Chomsky (2000: 101), Merge takes \((\alpha, \beta)\), two syntactic objects, and forms \(K(\alpha, \beta)\), a new syntactic object, from them. Merge is limited to two objects at a time (Chomsky 2005: 5) and yields the structural relation of Immediately-Contain (Is-a-member-of) for \((K, \alpha)\) and \((K, \beta)\). That is, both \(\alpha\) and \(\beta\) are members of \(K\) by virtue of having been merged into a
single new syntactic object. Iterated Merge (Chomsky 2001: 3), which consists in
the recursive application of Merge, results in the relation Contain (is a Term-of).

Chomsky (2005: 5) claims that the operation Merge does not change any of the two
elements that are merged due to a ‘no-tampering condition’ (NTC), formalised in
(7). According to the NTC, Merge cannot alter the featural composition of the
elements that undergo Merge, or add new features to those elements.

(7) No-tampering condition
Merge of X and Y leaves two syntactic objects unchanged.

If the NTC holds, therefore, LIs must contain some inherent feature that allows them
to undergo Merge with other LIs to form complex structures. Chomsky (2005: 6)
refers to this feature as the edge-feature (EF) of an LI. Only interjections, which are
full expressions that do not merge with other elements, do not have an EF. It is
assumed that all LIs have an EF and, thus, Merge is recursive in the sense that it can
take place in an iterative fashion.

In Chomsky (2000: 133) there is a difference between Set-Merge and Pair-Merge,
which are the new labels for substitution and adjunction. Set-Merge is in principle a
symmetric operation and as such, either \( \alpha \) or \( \beta \) in \( (\alpha, \beta) \) can project as the label of the
new syntactic object \( K(\alpha, \beta) \). However, since Merge takes place with the purpose of
satisfying the (semantical)-selectional properties of only one of the two elements that
are merged. Set-Merge is inherently asymmetric: only the selector, which can be
either \( \alpha \) or \( \beta \), but not both at the same time, can be the label of the new pair. Pair-
Merge, on the contrary, has no selector and is optional.

Chomsky (2005: 6) also distinguishes between two different types of Merge, namely
external Merge (EM) and internal Merge (IM). If \( \beta \) is not part of \( \alpha \), it is a case of
EM; by contrast, if \( \beta \) is part of \( \alpha \), we have a case of IM, which has been traditionally
referred to as Move. The two types of Merge have different effects at the interfaces.
While EM is responsible for argument structure, IM is concerned with discourse-
related properties and scopal effects (Chomsky 2005: 7).

The second fundamental operation, Agree, which has been mentioned in the
previous subsection as mediating between Probes and Goals, is language-specific
and establishes an agreement or case-checking relation between an LI and a feature
F in a restricted search space known as its domain. This operation eliminates /
values uninterpretable features of two elements with matching sets of features: one
with interpretable features and the other with uninterpretable ones, so that the
derivation can satisfy the Interpretability Condition (see section 2.3.2).

In Chomsky’s most recent work (2005), there is also the possibility of having
Multiple Agree, which is a concept that is owed to Hiraiwa. In a Multiple Agree
configuration, a single Probe establishes an Agree relation with various Goals in its domain provided there is no Goal with unvalued features which introduces intervention effects.

The last central operation, Move, is a combination of the two other operations, Merge and Agree. As has been pointed out, it is also referred to as IM in Chomsky (2005). Move can be defined in the following technical terms:

‘Move establishes agreement between α and F and merges P(F) to αP, where P(F) is a phrase determined by F (...) and αP is a projection headed by α. P(F) becomes the specifier (Spec) of α ([Spec, α]).’

(Chomsky 2000: 101)

Take the example in (8), where wh-movement has taken place. The DP which book (or P(F)) can be assumed to bear an interpretable [wh] feature, as seen in its morphology. The category C (or α), on the contrary, bears an uninterpretable [wh] feature, which needs to be deleted in the course of the computation.

(8) \[CP Which book [C did [TP Mary [t buy t]]]

The mechanism by which this uninterpretable [wh] feature can be eliminated is Agree. That is, C probes a matching feature, which is found in the object DP, the Goal, which carries an interpretable [wh] feature with which it agrees. The DP which book becomes the Spec of C, possibly due to the fact that C has an EPP-feature. This type of movement is A-bar movement, and is distinguished from A-movement, where phi-features are involved.

An example of A-movement can also be found in the example in (8), as the presence of a trace in the subject position shows. Under the well-motivated assumption that subjects are generated in a position which is internal to the VP (Koopman and Sportiche 1991), the DP Mary must have moved out from that position to the Spec, TP position, where it surfaces. In this case, the subject DP carries interpretable phi-features, while T bears uninterpretable phi-features that need to be deleted. T probes the subject DP, the Goal, and Agree takes place between these two elements, with subsequent erasure of uninterpretable phi-features. Then, the DP moves to Spec, TP to satisfy the EPP-feature of T. Structural case

---

32 As will be seen in subsection 2.3.4, Chomsky (2005) assumes that it is the EF of the wh-expression that triggers movement to Spec, CP, the edge of a strong phase. This seems to be consistent with the view of the operation Move as IM.

33 A-movement is revisited in subsection 2.3.4, where it is shown that the phi-features that motivate it originally come from C and not from T.
features on the Goal are also deleted in the Agree relation that has been established between the subject and T.

The erasure of structural case features on the Goal is the consequence of a slightly different mechanism in Chomsky (2001 and subsequent work), where valuation of uninterpretable features is assumed to take place. Structural case can be nominative or accusative, which makes it reasonable to assume that the DP in (8) has an unvalued structural case feature. It is depending on the element that values the case feature that the DP is in the nominative or the accusative case. Let us assume that if the valuing Probe is T, the case feature of the Goal will be valued as nominative, while if the Probe is v or a defective T[det] (i.e. a head with an incomplete set of phi-features), the case will be accusative. Once valued, the structural case features delete.

Since Move is a combination of two independent operations, Merge and Agree, it is assumed to be more complex than its components. Hence, it is a ‘last resort’ operation that is used when it is the only possible operation to satisfy a requirement imposed by a formal feature.

To sum up, this section has addressed the properties of the three operations that apply to LIs in the narrow syntax. Merge is used to create new syntactic objects; Agree is the mechanism by which uninterpretable features are eliminated from the computation as required by interface conditions, and Move is a combination of the two which applies when Agree alone cannot get rid of uninterpretable features. An example has been used to illustrate the functioning of Move, the most complex operation, and Agree.

2.3.4 Phases

The Minimalist framework imposes a strongly derivational approach, in the sense that information is handed out to the interfaces in a dynamic fashion, dispensing with the creation of ‘expressions’ as independent entities. In this context, therefore, the notion of phase (Chomsky 2000, 2001 and 2005) greatly simplifies operative complexity, since it endows the computation of structural derivations with strict cyclicity.

The main assumption in Phase Theory is that to generate a derivation, an LA is selected from the Lexicon in a one-time operation. After that, at various stages of the

---

34 Agree can operate independently of Merge; that is, Agree is not always part of Move and can be implemented in isolation (Chomsky 2000: 132). This possibility was not considered in Chomsky (1995).
derivation, different subsets from the original LA –LA_i, LA_j and so forth– are selected, placed in active memory and built into hierarchical structures by the operations in the narrow syntax. Once a subarray is exhausted, another is selected, if needed by the computation, until all subarrays have been used. At the end of each phase, the syntactic object is transferred to LF and PF. Once sent to the interfaces, the information is no longer accessed in later stages of the derivation, as expressed by the Phase Impenetrability Condition (PIC) in (9). Derivations and Spell-Out are strongly cyclic under this view.

\[(9) \text{ Phase Impenetrability Condition}\]
\[
\text{In phase } a \text{ with head } H, \text{ the domain of } H \text{ is not accessible to operations outside } a, \text{ only } H \text{ and its edge are accessible to such operations.}
\]

(Chomsky 2000: 108)

Phases are claimed to be propositional. Such an assumption follows from the observation that subarrays should determine syntactic objects which are relatively independent in terms of interface properties, such as a verb phrase with complete theta-role assignment, or a full clause with tense and force. The closest counterpart of these syntactic objects in terms of meaning would be a proposition (Chomsky 2000: 106). Such an observation has led Chomsky (2005) to endorse the view that a phase is, therefore, CP or vP, but not TP for the reasons that are outlined in next section. Section 2.3.4.2, by contrast, presents Gallego’s (2005, 2007) discussion of some empirical evidence in Iberian Romance languages that challenges Chomsky’s (2005) claim that only v^P and CP are phases. He proposes that TP is turned into a phase as a consequence of v^*-to-T movement, which allows T to inherit phase head properties from v^*. He calls such a process Phase Sliding.

### 2.3.4.1 Strong phases in Chomsky (2005)

Before considering the arguments Chomsky (2005) puts forward not to grant TP strong phase status, it should be mentioned that, in the MP, a crucial distinction is made between substantive and functional LIs. N(ouns), V(erbs), and A(djectives) belong to the former group, since they have descriptive content. Categories such as C, T or D, which encode grammatical information, fall into the latter group. There are other functional categories (FCs) such as Foc(us), Top(ic), Neg(ation), but only C, T and v are considered core FCs (Chomsky 2000: 102).

C expresses force and mood, while T expresses tense and event structure. Little v heads transitive constructions. T and v have phi-features, while these are optional for
As for their s(emantic)-selectional features, $v$ and $T$ need to be selected by some other category, while $C$ can either be unselected or selected by substantive categories. $T$ can be selected by $C$, thus having a full set of phi-features, or by $V$, which causes $T$ to be defective (i.e. lacking a complete set of phi-features). Both $T$ and $v$ can select $V$, but $v$ can also select a DP, which will be its external argument. Each of these three functional categories has EPP-features, which allow them to project a Spec position that can be filled in by a moved $wh$-phrase in the case of $C$, a subject in the case of $T$ and a raised object in the case of $v$.

Returning to the discussion on phases, only $CP$ and $v^*P$ are claimed to be strong phases, where $v^*$ is ‘the functional head associated with full argument structure, transitive and experiencer constructions’ (Chomsky 2005: 9). $TP$ is argued not to be a phase because its phi-features are, according to Chomsky (2005: 9), inherited from $C$. That is, $T$ only has phi-features if it is selected by $C$; if it is selected by $V$, $T$ lacks phi-features and tense. The passive and unaccusative constructions cannot be phases, either, since they lack external arguments. When $T$ is selected by $C$ and inherits phi-features from it, it can serve as a Probe in the CP phase (Chomsky 2005: 15).

$CP$ and $v^*P$ are strong phases (Chomsky 2001: 12) in the sense that they contain available landing sites for moved elements. A further assumption is that cyclic Spell-Out occurs at the strong phase level.

Despite not having semantic content, uninterpretable features may have phonological effects; therefore, valuation of uninterpretable features must take place within a strong phase. That is, valuation must take place before the syntactic object is transferred to the phonological component (Chomsky 2005: 20). Once all uninterpretable features have been erased from the narrow syntax by means of valuation and the syntactic object is transferred to the phonological component, the phase can no longer be accessed by the computational system, as required by the PIC in (9). Only the head of the phase and its edge (i.e. a specifier of $H$ or elements adjoined to $HP$) can be accessed in the next strong phase. This means that an element must be raised to the edge of the phase in order to be extracted and participate in another phase. The PIC guarantees the strong cyclicity of the Move operation together with the requirement in (10).

\[
\text{(10) Properties of the probe/selector $\alpha$ must be satisfied before new elements of the lexical subarray are accessed to drive further operations.}
\]

\[(Chomsky 2000: 132)\]

If the properties of $\alpha$ have not been satisfied within the phase, the derivation necessarily crashes. This is due to the fact that, given the PIC, $\alpha$ can no longer be
accessed once the phase is over and the syntactic object has been handed out to the phonological component.

To conclude this section, let us comment on the distinction between A- and A-bar movement, which is an issue that needs clarification in the light of the assumptions made on phases. It has been claimed in this section that TP is not a phase, since T does not have intrinsic phi-features, which it inherits from the category C. However, in section 2.3.3, it has been claimed that in A-movement, the interpretable phi-features of a DP in the vP region Agree with the uninterpretable phi-features of T, which induces deletion of those features; in addition, T is assumed to have an EPP-feature that triggers movement to Spec, TP. Obviously, this view can no longer be maintained if TP is not a phase, since only strong phases are targets for movement.

Chomsky (2005) offers a solution to this problem by arguing that both A-movement and A-bar movement are triggered by a Probe in C. In the case of A-bar movement (wh-movement, topicalisation, focalisation, and any other movement related to the left periphery) the EF of C acts as a Probe. Different kinds of potentially movable elements such as wh-phrases, topics and focus-phrases may be Goals and eventually undergo movement to the edge of C35.

By contrast, in the case of A-movement, it is thanks to the fact that T inherits the phi-features of C that it becomes a probe that can attract a DP to the Specifier position of TP. A further assumption is that all phase-heads have the property of transferring their phi-features to other non-phase heads: similarly to what happens with C and T, V inherits the phi-features of v*, thus being able to act as a Probe for objects with structural case, which raise to Spec, VP.

2.3.4.2 Phase Sliding (Gallego 2005, 2007)

The term Phase Sliding refers to the claim that in Null Subject Languages (NSLs), T inherits edge features from v* when v*-to-T movement takes place. Such a claim, which is supported by extensive data from a number of Iberian Romance languages, captures an old observation that Spec, TP seems to have both A- and A’-features in NSLs. Within Gallego’s (2005, 2007) account, this puzzling double-sided behaviour of T is attributed to the fact that it has phi-features, which are responsible for its A-properties, but also edge features (inherited from v*) and, consequently, A’-properties, too.

35 C is here understood, in line with Chomsky (2000, 2001), as shorthand for the left periphery, which contains a number of other functional categories (Rizzi 1997). Actually, Chomsky (2000: 108) defines A-bar movement as involving what he calls “P-features of the peripheral system (force, topic, focus, etc.).”
Assuming that $v^*\text{-to}-T$ movement is a syntactic operation, Gallego claims that it extends (or ‘pushes’, as he puts it) the $v^*P$ phase up to TP, which reprojects. The immediate consequence of Phase Sliding, then, is that those operations that were supposed to apply at the $v^*P$ level do so at the $v^*/TP$ level. For instance, as will be discussed more in depth in chapter 5, while subextraction from Spec, $v^*P$ is generally possible, it is no longer available after $v^*\text{-to}-T$ movement has occurred and the $v^*P$ phase has extended. Gallego also discusses data on intervention effects in VOS order, which are avoided after Phase Sliding. He argues that this is due to the fact that both nominative and accusative Case are assigned in parallel within the $v^*/TP$ phase.

Another point of interest is the observation that in French, which is not an NSL, TP does not seem to constitute a phase as is claimed to be the case in other Iberian Romance languages. Gallego (2005) argues that despite displaying $v^*\text{-to}-T$ movement, French is comparable to English in that it has the EPP, which prevents subjects from remaining in a $v^*P$-internal position and forces them to raise higher up in the clause. Judging from these observations as well as the fact that, unlike other Romance languages, it displays very poor or even no peripheral phenomena in the CP layer, he claims that $v^*\text{-to}-T$ movement in French is not truly syntactic, but rather, phonological.

As will be seen in chapter 5, the fact that in French, like in English, Phase Sliding does not take place has important implications for the analysis in the present dissertation. Not only does French pair up with English in this aspect, it also seems to do so with respect to the expression of negation and NC. The connection of such an observation with Phase Theory will be made explicit from chapter 3 onwards.

### 2.3.4.3 Interpretability and valuation (Pesetsky and Torrego 2004)

Pesetsky and Torrego (2004) propose to understand agreement as ‘feature-sharing’. In addition, they defend the view that valuation and interpretability of features are independent concepts.

In Chomsky’s work, (un)interpretability, which is related to whether a given feature makes a semantic contribution, is biconditionally related to valuation: if a feature is unvalued, it is necessarily uninterpretable. Uninterpretable features must be valued by means of Agree, and must subsequently delete after or upon Transfer.

Pesetsky and Torrego agree with Chomsky in that an unvalued feature of a Probe scans its c-command domain to find a Goal with a matching feature with which to Agree. However, as formalised in (11), they claim that the Agree mechanism replaces the Probe with the Goal, which results in an instance of a valued feature
occupying the place of the initially unvalued Probe. It is in this sense that Agree is feature-sharing.

(11) **Agree (feature sharing version)**

(i) An unvalued feature F (of a probe) on a head H at syntactic location α (F_α) scans its c-command domain for another instance of F (a goal) at location β (F_β) with which to agree.

(ii) Replace F_α with F_β, so that the same feature is present in both locations.

(Pesetsky and Torrego 2004: 4)

By eliminating Chomsky’s (2000, 2001) Biconditional in (12), two more possible combinations of features exist. These are (i) uninterpretable but valued and (ii) interpretable but unvalued. Adding those to the ones already discussed in Chomsky’s work yields the picture in (13).

(12) **Valuation / Interpretability Biconditional** (Chomsky 2001: 5)

A feature F is uninterpretable iff F is unvalued.

(Pesetsky and Torrego 2004: 3)

(13) **Types of features**

\[
\begin{align*}
 uF & \quad \text{uninterpretable, valued} & iF & \quad \text{interpretable, valued} \\
 uF [ ] & \quad \text{uninterpretable, unvalued} & iF [ ] & \quad \text{interpretable, unvalued}
\end{align*}
\]

(Pesetsky and Torrego 2004: 5)

Assuming that unvalued features act as Probes, two kinds of features can probe for a Goal in Pesetsky and Torrego’s system: interpretable unvalued and uninterpretable unvalued. They illustrate probing of an interpretable unvalued feature by considering the [Tense] feature of the syntactic category T.

On the finite verb, [Tense] is uninterpretable and enters in an Agree relation with the [Tense] feature of T. As T c-commands the finite verb, it is assumed that it must be the Probe in the Agree relation. This is only possible if interpretable features can be unvalued, and uninterpretable features can be valued. Pesetsky and Torrego illustrate this situation in the following way:
Pesetsky and Torrego also commit to Chomsky’s claim that an uninterpretable feature must enter in an Agree relation with an interpretable matching feature. In Chomsky’s work, Agree is a precondition for deletion of the uninterpretable feature: Agree values the uninterpretable feature, which must subsequently delete. However, as noted by Pesetsky and Torrego, Chomsky’s (2000, 2001) account does not explain why a feature should be valued to be deleted.

Combining Brody’s (1997) Thesis of Radical Interpretability, formalised in (15), with Pesetsky and Torrego’s notion of feature-sharing results in the conclusion that what deletes at LF are not uninterpretable features, but uninterpretable instances of features, with every feature having at least one valued interpretable instance. In the example in (14), therefore, the [Tense] feature on the finite verb must enter in an Agree relation with the unvalued interpretable feature of [Tense] on T because, otherwise, the [Tense] feature would not be semantically interpreted in any syntactic location, and (15) would be violated.

\[\text{(15) Thesis of Radical Interpretability (Brody 1997)}\]
Each feature must receive a semantic interpretation in some syntactic location.

(Pesetsky and Torrego 2004: 8)

### 2.4 Distributed Morphology

#### 2.4.1 Core concepts

Distributed Morphology (DM) (Halle and Marantz 1993; Embick and Noyer 2007) is a non-lexicalist theory of the syntax-morphology interface in that it rests on the

*Values of features are expressed with numbers.*
hypothesis that words are construed by syntactic rules, which precludes the existence of a generative Lexicon. Since all complex objects, regardless of whether they are words or larger syntactic structures, are generated by the same system, the interface between syntax and morphology is direct. That is, morphology, in principle, reflects syntactic structure.

With respect to what was said in previous sections about the Lexicon and the sound-meaning connection of LIs, DM takes a different view. In this model, grammar is understood as a generative system that consists of a set of syntactic rules, whose output is affected by further operations that take place at the PF and LF interfaces. Crucially, PF ‘is understood as a sequential derivation that terminates in a phonological representation’ (Embick and Noyer 2007: 293). This means that post-syntactic readjustments of syntactic terminals can take place at PF, these being responsible for a particular phonological representation. Such a division of labour, i.e. word formation takes place in the syntax but also at the PF interface, motivates the name Distributed Morphology.

In DM, it is assumed that morphemes are the primitive units that are subject to the core syntactic operations Merge and Move. Morphemes are defined as bundles of features of two types: phonological and grammatical / syntactico-semantic. The morphemes themselves can also be divided into two distinct types: Roots and abstract morphemes.

While Roots include elements that are complexes of phonological features, they do not contain grammatical features. By contrast, abstract morphemes, which amount to functional categories, as defined in previous sections, are made up of grammatical features, and lack phonological ones.

The features of abstract morphemes are universal, whereas Roots are pairs of sound and meaning that are language-specific. In that sense, only Roots are comparable to what has been referred to as LI in previous sections. Functional categories (abstract morphemes in DM) do not have phonetic content in the syntactic derivation; rather, once the syntactic computations have taken place, phonological features are assigned to the abstract morphemes involved in the derivation, as these kind of features are instructions for the articulatory-perceptual system.

Recapitulating, not every element that is stored in our mental dictionary is a sound-meaning pair in the model of grammar that is put forward in DM. This is significantly different from what is assumed in the MP model: the concept of morpheme in DM is incompatible with the view of LIs as fixed sets of semantic, syntactic and morphophonological features. However, the fact that functional categories do not bear a pre-established one-to-one correspondence with a phonological realisation makes it easier to explicate a number of puzzling linguistic phenomena, the expression of negation and NC across languages being one of them. Chapters 3, 4 and 5 show how this tenet of DM is very inspiring to account for the
(im)possibility of the sentential negative marker to co-occur with n-words (see chapter 1) in NC constructions.

Having established that abstract morphemes, i.e. functional categories, are supplied phonological features in the course of the derivation, more has to be said about Vocabulary Insertion, the mechanism that makes it possible. The Vocabulary is the list of the different phonological forms (phonological exponents) that abstract morphemes can be assigned. For each form, the conditions on insertion are specified, forming a pair that is known as a Vocabulary Item.

Borrowing the example that Embick and Noyer (2007) use to illustrate Vocabulary Items, let us focus on English plural nouns. The first assumption is that English has the abstract morpheme [pl] as part of a functional head labelled as ‘Number’, which syntactically combines with nouns. The phonological exponent of [pl] is /z/, as formalised by the Vocabulary Item in (16), which results in /z/ being assigned to [pl].

(16) \[ z \leftrightarrow [pl] \]

Vocabulary Insertion is constrained by the Subset Principle in (17). Such a principle determines which phonological exponent of a Vocabulary Item is inserted into a position in cases where more than one is possible.

(17) The Subset Principle

The phonological exponent of a Vocabulary Item is inserted into a position if the item matches all or a subset of the features specified in that position. Insertion does not take place if the Vocabulary Item contains features not present in the morpheme. Where several Vocabulary Items meet the conditions for insertion, the item matching the greatest number of features specified in the terminal morpheme must be chosen. (Halle 1997)

(Embick and Noyer 2007: 298)

English plural morpheme [pl] has other phonological exponents apart from /z/. The contextual condition which determines whether /ən/ (as in oxen) or Ø (as in fish-Ø) will occur for [pl] is the Root. The different realisations of English plural, therefore, are contextual allomorphs.

As can be seen in the outline above, underspecification of the phonological exponents of a Vocabulary Item is central to DM. That is, while the syntactic terminals where material is to be inserted are fully specified in terms of syntactico-
semantic features, the Vocabulary Items that have to be inserted into these terminals are not fully specified.

### 2.4.2 Derivation of linguistic expressions

In DM, the model of grammar that is assumed is the one in (18). It departs from the MP model presented in (4) in various respects, which are addressed here in turn.

\[(18) \quad \text{Syntactic Derivation} \quad \downarrow \quad \text{(Spell-Out)} \]
\[\quad \text{Morphology} \quad \downarrow \quad \text{PF} \quad \text{LF} \]

As can be seen in (18), no generative Lexicon is assumed to be part of the model. Instead, DM assumes that grammar accesses a number of Lists at different stages of the derivation. From this point of view, the information contained in the MP Lexicon that has been described in section 2.3.1 is scattered (or distributed) throughout the components of the computational system.

\[(19) \quad \text{Lists} \]
\[\quad \text{a. The Syntactic Terminals:} \]
\[\quad \text{The list containing the Roots and the Abstract Morphemes.} \]
\[\quad \text{b. The Vocabulary:} \]
\[\quad \text{The list of Vocabulary Items, rules that provide phonological content to abstract morphemes.} \]
\[\quad \text{c. The Encyclopedia:} \]
\[\quad \text{The list of semantic information that must be listed as either a property of a Root, or of a syntactically constructed object (idioms like kick the bucket).} \]

(Embick and Noyer 2007: 301)
The lists in (19) are accessed at different stages of the model in (18). As shown in (20), the items that participate in the syntactic derivation are taken from the list *Syntactic Terminals*, while the *Vocabulary*, which contains the rules that assign a phonological realisation to the abstract morphemes that are part of the selection of items taken from the *Syntactic Terminals* list, is accessed at PF. Finally, the *Encyclopedia* is relevant after PF and LF.

\[(20)\]

- **Access to Syntactic Terminals** → Syntactic Derivation → (Spell-Out)
- **Access to The Vocabulary** → PF → LF
- **Access to The Encyclopedia** → (Interpretation)

(Embick and Noyer 2007: 301)

The core operations of grammar, Merge and Move, which were discussed as being central to the model proposed in the MP, are also assumed to be central to DM, both for word formation and for the assembling of more complex syntactic objects. Likewise, the distinction between interpretable and uninterpretable features depending on whether they have a semantic interpretation or not is still kept.

### 2.4.3 PF operations

In DM, PF is sequential, which means that operations or processes can take place after syntax and have an influence in the final phonological representation that a given morpheme is assigned. In addition, PF operations are held responsible for any mismatches that can be observed between syntax and morphology. Since in DM it is assumed that all morphology is syntax, whenever this relationship is not transparent (i.e. the order of morphemes does not result from the syntactic operations that have been performed during the derivation), an explanation is sought in PF processes that could have potentially rearranged morphemes yielding the attested output. As stated
by Embick and Noyer (2007: 304), ‘operations that apply at PF are minimal readjustments, motivated by language-particular requirements’. For instance, they discuss the possibility of adding morphemes at PF, otherwise known as ornamental morphology.

This process can account for the presence of agreement morphemes in a number of languages in spite of the fact that no AGR syntactic terminal exists in the syntax: if the language in question has a morphological requirement for agreement to occur with finite Tense, which is a syntactic terminal, an agreement morpheme can be adjoined to T at PF. Another PF process is the introduction of features via Vocabulary Insertion, which is the operation by which phonological features are assigned to abstract morphemes.

Other processes that are also assumed to take place in the PF branch are Impoverishment, Fission and Obliteration. While Impoverishment is responsible for the deletion of a feature of a morpheme and the insertion of a default (i.e. less marked) form, Fission allows more than one Vocabulary Item to be inserted in a syntactic terminal. Obliteration results in the elimination of a syntactic node from the morphological representation. The result is that the terminal in question escapes lexical insertion.

Another PF operation is Fusion, which takes two syntactic terminals that are sisters and fuses them into a single terminal, with the consequence that only one Vocabulary Item is inserted. For example, in English, it seems to be the case that v-to-T movement, which has been argued to take place after syntax, is a case of Fusion fed by head-to-head movement.

For the present dissertation, the focus is on the characterisation of Impoverishment and Obliteration, as they are relevant processes for the hypothesis that is put forward in further chapters to account for the expression of negation in English (both Standard and Non-Standard) and a variety of Romance languages. Fission will also be briefly considered, for it is assumed in the next chapter to be involved in the phenomenon of do-support (Flagg 2002).

2.4.3.1 Impoverishment

Impoverishment, a concept which is originally owed to Bonet (1991) and her treatment of clitics in Romance, deletes features of morphemes. The result of this operation for an unmarked value of a feature is that no Vocabulary Item that requires the deleted feature will be inserted for the morpheme in question. However, in the case of deletion of marked values, markedness rules, which are assumed to be part of the grammar, supply the default value for the deleted feature. Let us illustrate that
with an example from adjectival suffixes in Norwegian. Consider the paradigms in (21).

(21)  

\begin{center}
<table>
<thead>
<tr>
<th>Gender</th>
<th>Number</th>
<th>Suffix</th>
</tr>
</thead>
<tbody>
<tr>
<td>STRONG</td>
<td>-neuter</td>
<td>[+neuter]</td>
</tr>
<tr>
<td>[-pl]</td>
<td>Ø</td>
<td>-t</td>
</tr>
<tr>
<td>[+pl]</td>
<td>-e</td>
<td>-e</td>
</tr>
<tr>
<td>WEAK</td>
<td>-neuter</td>
<td>[+neuter]</td>
</tr>
<tr>
<td>[-pl]</td>
<td>-e</td>
<td>-e</td>
</tr>
<tr>
<td>[+pl]</td>
<td>-e</td>
<td>-e</td>
</tr>
</tbody>
</table>
\end{center}

(Harley and Noyer 1999)

As can be observed in (21), Norwegian adjectives can bear three different suffixes when they occur in a ‘strong’ syntactic position; however, when they occur in a ‘weak’ syntactic position, the only possible suffix is -e. Sauerland (1985), as reported in Harley and Noyer (1999), analyses the paradigm above as a result of an Impoverishment rule, formalised in (23), which affects the Vocabulary Items in (22).

(22)  

\begin{center}
<table>
<thead>
<tr>
<th>Gender</th>
<th>Number</th>
<th>Suffix</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.</td>
<td>/t/</td>
<td>[ ____ , -pl, +neut] / Adj + ___</td>
</tr>
<tr>
<td>b.</td>
<td>Ø</td>
<td>[ ____ , -pl, -neut] / Adj + ___</td>
</tr>
<tr>
<td>c.</td>
<td>/e/</td>
<td>Elsewhere / Adj + ___</td>
</tr>
</tbody>
</table>
\end{center}

(23)  

\[ [+\text{neuter}] \text{Impoverishment} \]
\[ [+\text{neuter}] \rightarrow \emptyset \]

The rule in (23) deletes the values for neuter gender in weak syntactic positions, thus blocking the insertion of /t/ or Ø. The default form -e is inserted instead.

As discussed in the next chapter, Impoverishment will be argued to be responsible for the insertion of the any-phonological exponents of the n-indefinites in negative contexts. It will be argued that Impoverishment is just one possibility (the other one being Obliteration of the negative marker) to avoid the violation of the Filter that prevents the accidental repetition of negative features in particular domains.

In chapter 4, it is argued that n-indefinites in Non-Standard varieties of English do not trigger Obliteration of the negative marker nor are affected by Impoverishment when occurring post-verbally. This means that post-verbal n-indefinites with overt negative morphology can co-occur with the sentential negative marker without
resulting in a Double Negation reading. I argue that this fact, which distinguishes Standard English from Non-Standard varieties of English, is a consequence of n-indefinites not behaving, syntactically speaking, in the same way in Standard and Non-Standard English.

For Standard English, it is assumed in chapter 3 that n-indefinites raise to Spec, NegP. Movement out of the position where they are base-generated in either forces Obliteration of the negative marker or Impoverishment of the n-indefinite when these co-occur in the same Spell-Out domain.

Obliteration, which is discussed in the next section, is also observed in some Non-Standard varieties of English and in Romance languages that implement Non-Strict Negative Concord. In these varieties / languages, the negative marker cannot possibly co-occur with a pre-verbal n-indefinite. The ultimate trigger of Obliteration is, as discussed in further chapters, a language-particular condition that prevents too many negative features from co-occurring in the same Spell-Out domain, which is defined according to Chomsky’s theory of phases.

2.4.3.2 Obliteration

Obliteration is a morphological operation that deletes a whole syntactic node, rather than a feature. As discussed in Arregi and Nevins (2007a), Obliteration is formally distinct from Impoverishment. Arregi and Nevins discuss the Basque $g/z$-constraint, which I present here to illustrate how Obliteration works.

The data from various dialects of Basque show that many of these disallow the co-occurrence of 1PL with 2nd person clitics, which result in certain agreement restrictions, known as the $g/z$-constraint. That is, in certain dialects, *you-us and *we-you combinations are not possible and repair operations are triggered when these are syntactically generated. Consider, for instance, the examples in (24) and (25). These illustrate the deletion of the first plural dative morpheme, $sku$, when it occurs in the context of a second person ergative morpheme.

(24) (Hik guri emon) d- o- $sku$ na $\rightarrow$ d- o- na
(You us gave) 3SG.ABS AUX.TRANS 1PL.DAT 2SG.F.ERG $\rightarrow$ 3SG.ABS AUX.TRANS 2SG.ERG
‘You (F.SG) [gave] it to us’

(25) (Suek guri emon) d- o- $sku$ sue $\rightarrow$ d- o- sue
(Y’all us gave) 3SG.ABS AUX.TRANS 1PL.DAT 2PL.ERG $\rightarrow$ 3SG.ABS AUX.TRANS 2PL.ERG
‘You (pl) [gave] it to us’

(Arregi and Nevins 2007a)
The examples in (26) and (27), by contrast, illustrate the deletion of *gu*, which corresponds to the first plural ergative morpheme, when it co-occurs with a second person dative morpheme. Following Halle (1997), who assumes that the feature [+Participant] is common to first and second person, and Calabrese (2004), according to whom Ergative and Dative share the feature [+Motion], Arregi and Nevins (2007a) group the examples in (24)-(27), which show that microvariation exists with respect to how the *g/-z-* constraint is repaired, as the output of an Obliteration rule such as (28).

\[(26)\]  (Guk hiri emon) d- \(\text{u- a- gu} \rightarrow \text{d- xa- k}\)
(We you gave) 3SG.ABS AUX.TRANS 2SG.MSC.DAT 1PL.ERG \(\rightarrow\) 3SG.ABS AUX.APPLIC 2SG.MSC.DAT

\`{\textbf{We [gave] it to you (m.sg)}}`\`

\[(27)\]  (Guk suek ikusi) s- aittu- e- gu \(\rightarrow\) s- ari- e
(We you saw) 2ABS AUX.TRANS ABS.PL 1PL.ERG \(\rightarrow\) 2.ABS AUX.INTRANS ABS.PL

\`{\textbf{We [saw you)}}`\`

\[(28)\]  Obliterate the Node containing [+Motion, +Participant, +Author, +Pl] in the environment [+Participant]

(Arregi and Nevins 2007a)

In Arregi and Nevins (2007b), it is argued that in Ondarru Basque the *g/-z-* constraint is repaired by means of Obliteration of the first person plural absolutive morpheme. That is, given the sequence in (29), Obliteration deletes the morpheme in bold type.

\[(29)\]  (Suk gu-Ø ikusi) g- atxu -su \(\rightarrow\) d- o- su
(You us see-PREF) ABS.IPL PRS ERG.2SG \(\rightarrow\) EP37 PRS ERG.2SG

\`{\textbf{You have seen us}}`\`

(Arregi and Nevins 2007b)

What all these examples above have in common is the fact that the feature [+Participant] is marked. Hence, when more than one [+Participant] feature co-occurs in a given structure, post-syntactic operations are triggered to repair the situation. Obliteration of dative, ergative and absolutive morphemes is just one possible strategy. Impoverishment of second person ergative is also possible in certain dialects of Basque in the context of first person plural absolutive. This is illustrated in (30).

\[\text{EP stands for enthetic proclitic.}\]
In the case of (30), an Impoverishment rule, formalised in (31) applies. It deletes the feature [+participant] of a node (rather than deleting the node as a whole), and a default form is inserted. The default form, which is phonologically null, is identical to the third person singular ergative morpheme.

(30) (Suk gu-Ø ikusi)  
    (You us see-PRF)  
    \( g^- \)aittu -su \( \rightarrow g^- \)aittu -Ø  
    ABS.1PL PRS ERG.2SG \( \rightarrow \) ABS.1PL PRS ERG.3SG  

    ‘You (pl.) [saw] us’  

(Arregi and Nevins 2007a)

In conclusion, Obliteration and Impoverishment are different operations: while the former deletes a Node as a whole, thus allowing it to escape lexical insertion, the latter deletes just a feature of a given Node. Impoverishment is associated with the insertion of a less marked form, i.e. a default form, while Obliteration is not.

2.4.3.3 Fission

Fission consists in having a single morpheme inserted as several phonological units. That is, Fission can ‘split’ the features of a morpheme so they are phonologically represented as various pieces. According to Harley and Noyer (1999), whenever a morpheme has been affected by Fission, Vocabulary Items are no longer in competition for a single position; rather, an additional position is made available every time a Vocabulary Item is inserted.

An example of Fission is found in Huave verbal conjugation, which shows that more than one Vocabulary Item corresponds to a single AGR morpheme. Consider the paradigm in (32): ‘1’, ‘2’ and ‘3’ stand for first, second and third person; ‘1’ is first person exclusive, while ‘12’ is first person inclusive.
Huave verbal conjugation: present (atemporal) tense of -rang ‘make, do’

(32)

<table>
<thead>
<tr>
<th>Person</th>
<th>Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>s-a-rang</td>
</tr>
<tr>
<td>12</td>
<td>a-rang-ar</td>
</tr>
<tr>
<td>2</td>
<td>i-rang</td>
</tr>
<tr>
<td>3</td>
<td>a-rang</td>
</tr>
</tbody>
</table>

(Embick and Noyer 2007: 315)

The list of the Vocabulary Items corresponding to each morpheme occurring in (32) is formalised in (33). Attached to the root there are two theme vowels, i- for the second person, and a- elsewhere. The rest of morphemes express person and number. Notice that, as can be observed in (32), person features are expressed either by means of a prefix or a suffix, but not both at the same time. That is, when specific suffixes such as -aac and -aw occur, which express both number and person features, no person feature is realised as a prefix. When the opposite happens, i.e. when the suffix does not express person features, either the prefix s- or a floating [-back] feature that ablauts the theme vowel to -i do.

(33) Vocabulary Items for Huave verbal paradigm in (32): Exponents of AGR

-acc ↔ [+I, +you, +pl]
-a ↔ [+I, +you]
-s- ↔ [+I]
-[back]- ↔ [+you]
-a ↔ [-I -you +pl]
-an ↔ [+pl]

Embick and Noyer (2007: 315) assume the structure in (34) for the Huave verbal forms. As can be seen, only one AGR node is present in the syntax.

(34)
The thematic vowel is linearised as left-adjacent to the Root, while AGR, obeying a language-particular requirement, must be linearised as adjacent (but not necessarily left-adjacent) to the complex formed by $v$ and the Root. This means that AGR can be realised as a suffix or as a prefix as long as it is adjacent to $(v \ast \sqrt{\text{ROOT}})$, which implies that there is only one affix position at each side of $v$.

When a syntactic node undergoes Fission and its features can be phonologically realised in various positions of the string, blocking effects may be observed. The data in (32) show featural and positional blocking effects, as discussed by Embick and Noyer (2007: 317).

In the case of featural blocking, Spelling-out a feature in a particular position precludes its Spell-Out in some other position. This follows from the fact that once the insertion of a Vocabulary Item has been conditioned by a given feature, this feature counts as discharged and, thus, cannot be realised again. This is, for instance, what happens with the insertion of the 12 plural $-\text{aac}$: once this feature is Spelled-Out as $-\text{aac}$, $-\text{ar}$, $-\text{an}$ and $-\text{aw}$ cannot be inserted, and neither can the prefixes $s$- and $[-\text{back}]$.

When the insertion of one Vocabulary Item prevents the insertion of another Vocabulary Item in the same position, blocking is defined as positional. In the case of Huave, the requirement that AGR be adjacent to the $v\sqrt{\text{ROOT}}$ complex makes it impossible for more than one phonological exponent to occur on the same side of $v\sqrt{\text{ROOT}}$.

2.5 Summary and assumptions

In this chapter, the core assumptions of the MP have been presented, together with DM, a model which capitalises on the claim that no separate Lexicon exists where all LIs are stored as sound-meaning pairs. In DM, abstract morphemes / functional categories only bear grammatical features but no phonological content. The latter kind of features are assigned after syntax and are sensitive to the context where the abstract morpheme / functional category occurs. Other processes that readjust the syntactic output and determine the final phonological realisation of a morpheme are also claimed to take place at PF, which is given a central role in DM. The dynamic nature of PF is a departure from the core assumptions of the MP.

In both models, the MP and DM, the faculty of language is understood as an optimal solution to the legibility conditions imposed by the interfaces with the sensorimotor system and the conceptual-intentional system. Two levels of representation, PF and LF, mediate between language and the other two performance systems, and receive
instructions for computation in the form of features. Interface conditions, are thus seen as central for the study of linguistic phenomena.

Borrowing the idea that functional categories / abstract morphemes are not sound-meaning pairs, as is claimed in DM, I will assume that negation, which is argued to be a functional category in the next chapter, is a syntactic terminal that bears the feature \([\text{Neg}]\), which is interpretable, \([\text{iNeg}]\), whenever it has a semantic interpretation. The interpretable versus uninterpretable distinction on the basis of semantic interpretation is a shared assumption of the MP and DM. However, I align with DM in not relating such a feature to any phonological realisation a priori.

Again in line with DM, I adopt the view that PF is dynamic in that a number of operations can take place post-syntactically. These operations determine the phonological realisation of abstract morphemes / functional categories. As a functional category, negation will be assumed to be assigned a phonological realisation at PF which is context-sensitive and sometimes constrained by some further conditions. Which type of context the expression of negation is sensitive to and what is the nature of the constraints that determine the final Spell-Out of negation in a variety of languages are two issues that are discussed in further chapters.

As is done in the two models, the MP and DM, Merge and Move are claimed to be the core operations that allow syntactic structures to be assembled. I also take Agree, which is a mechanism that permits uninterpretable features to be checked and thus erased from the derivation, to be part of the core operations of the computational system.

Phase Theory is also central to the account that is put forward in the chapters that follow. In line with Chomsky (2000 and subsequent work), linguistic derivations are taken to be strongly cyclic (i.e. they proceed phase by phase) and subject to the PIC in (9), which forces material that is to participate in further phases to be extracted to the edge of the phase where it is base-generated.

As discussed in more detail in chapter 5, Gallego (2005, 2007) considers some linguistic evidence that can receive a principled explanation if the \(v^*P\) phase is somehow expanded into the \(v^*P/TP\) phase. Phase Sliding, as this phenomenon has been called, follows from \(v^*\)-to-\(T\) movement being truly syntactic and has consequences for the positioning of the negative syntactic terminal.

To conclude the present chapter, let us say that the conception of the Lexicon that is assumed herein is the one offered by DM because the idea that some room is left for variation in the phonological realisation of functional categories, as suggested by the proponents of DM, offers new insights into the study of NC and the expression of negation, which is an issue that is far from being settled. In addition, the fact that the variation in the Spell-Out of abstract morphemes is heavily constrained by the
context in which they occur is desirable from a theoretical point of view, as the occurrence of one phonological realisation or another is predictable.
3 The System of Negation in Standard English

3.1 Introduction

The present chapter is devoted to describing and analysing how negation works in Standard English. For this purpose, in sections 3.2 and 3.3, some terminology is introduced, and a number of theoretical considerations that are relevant to the syntactic account that is put forward in section 3.5 are discussed.

As stated in section 3.2, two types of negation have traditionally been distinguished in English: these have been labelled sentential and constituent negation (Klima 1964) and can be, in principle, told apart by means of a number of tests. It is shown, however, that these tests are problematic to a certain extent, which calls for a redefinition of the sentential versus constituent negation distinction in terms of the scope of negation with respect to the matrix predicate of the clause.

In section 3.3, it is established that English negation involves the projection of a Neg(ative) P(hrase). This conclusion is part of a more general discussion on whether negative markers can be granted functional status.

Further assumptions are made on the internal structure of NegP in Standard English which depart from the conventional view of this language as having negative quantifiers and, thus, disallowing the phenomenon of NC (see chapter 1). In addition, not and the contracted form -n’t, which are two possible phonological realisations of the sentential negative marker are assumed to be contextual allomorphs of the same syntactic head.

This view also departs from recent analyses of not and -n’t as Specifier of NegP and Neg* respectively (Haegeman 1995; Zeijlstra 2004), but is in line with Distributed Morphology (DM) accounts of negation (Frampton 2001; Flagg 2002 and Parrott 2007). In line with Laka (1990), it is also assumed that NegP is just one possible value of a broader syntactic category, labelled here as PolP.

Section 3.4 contains an overview of the diachronic evolution of the system of negation from Old English to Present Day (Standard) English. The several steps in the history of English negation are commented on by making reference to Jespersen’s Cycle (1917), a cyclic process in the expression of negation first described by (and hence named after) the linguist Otto Jespersen to account for the cross-linguistic similarities in the development of negation through time.

---

58 The original term is Klima (1964) is sentence negation, but the label sentential has also been used in the literature.