

## A Processing Account of *That*-Trace Effects

### 1.1. Introduction

A complementizer does not make a semantic contribution to any sentence because it has no real meaning. It serves a grammatical function only, because it indicates where the border of an embedded sentence is. However, the complementizer does not always need to be overt. It has a null-allomorph which can be used in all languages. In matrix clauses, the overt complementizer is never used. Instead, the null allomorph heads all matrix clauses. The complementizer need not be in the same position in all languages either. Languages like Japanese and Chinese have complementizers which occur at the end of the sentence instead of at the beginning of the sentence, while the complementizer is at the beginning of a clause in most other languages. A Japanese example of sentence final complementizers is given in example (1).

1. *anata-wa [John-ga dare-o aisiteiru to] omotteiru no?*  
you-Top John-Nom who-Acc love COMP think Q  
'Who do you think that John loves?'

In Japanese, *to* is an overt complementizer and it does not indicate where the embedded sentence starts, but where the embedded sentence ends. However, some languages really do not need an overt complementizer to indicate the beginning of the sentence. Consider the following example from English, which does not have an overt complementizer.

2. *Mary thinks [John kissed Sue at the party].*

In languages like English, both the overt and silent allomorph of the complementizer are allowed. Clearly in order to process a correct sentence in English, the overt complementizer is not needed to distinguish between embedded and matrix clauses. In languages like Dutch and Italian on the other hand, embedded clauses always begins with an overt complementizer. In the following example, the Dutch obligatory complementizer is illustrated.

3. *Jan denkt dat John Sue gekust heeft.*

Jan thinks that John Sue kissed has.

It seems that Dutch needs a complementizer to distinguish between the matrix clause and the embedded clause. There is thus obviously a difference in processing different languages, because in some languages, the processor needs a clear border for an embedded sentence because it cannot parse a sentence without it.

Although English is one of few languages that both allow overt and null complementizers at the beginning of embedded clauses, there is also one English structure in which the occurrence of an overt complementizer is not allowed. In these sentences, long-distant wh-movement also occurs. This structure is called, the *That*-trace effect. *That*-trace effects occur in more languages than English but in most language the complementizer is either obligatory overt or optional and there does not seem to be a connection to wh-movement.

In the past decades, theoretical linguists have tried to capture the usage of overt or null complementizers in many syntactic theories. During the eighties these theories were based on the Government and Binding framework. Linguists thought they had come up with a adequate theory to explain these effects, but their theory depended on the notion of government. The

idea of government, was abandoned when Chomsky introduced his Minimalist program, in which government did not occur but the notion of feature checking was introduced. This caused linguists to drop the subject because all previous work was now useless. In the late nineties, the subject was picked up again by the authors Pesetsky and Torrego. They tried to capture the structure in a syntactic theory as well. Their theory was based on a new syntactic framework named Minimalism, which was constructed in the early nineties by Noam Chomsky.

In this thesis, I will discuss that *That*-trace effects cannot be explained by syntactic theory because they stem from a processing problem, much like Garden path sentence structures. In the first part of this thesis, I will outline the phenomenon of *That*-trace effects in English. In the second part I will give a historical literature review of the theories that have been put forward in history. I will outline Minimalism and I will give a detailed explanation and critical review of the newest theory as proposed by Pesetsky and Torrego. In the last part of my thesis, I will work out my own theory of *That*-trace effects, which is based on processing. I will first outline the processing theory as proposed by Pritchett and I will then explain the English *That*-trace effects from a processing perspective. Lastly, I will look at cross-linguistic differences and I will show how the processing theory deals with all the varieties of the effect which occur in Dutch, German and Italian.

## 1.2. The phenomenon

*That*-trace effects occur in languages like English which generally do not require an overt complementizer to be present in an embedded sentence. According to Government and Binding theory, English does not allow a trace of movement to immediately follow an overt complementizer. A trace of movement immediately follows an overt complementizer when the CP contains a TP complement with a trace in its specifier position. Since matrix clauses in English never start with a overt complementizer, the phenomenon only occurs when an element is moved from an embedded clause to a higher clause.<sup>1</sup>

In English embedded sentences, overt complementizers are usually optional in an embedded sentence. The complementizer is optional regardless of whether a wh-phrase is moved from the embedded clause or not. The optionality of *that* is illustrated in the following examples.

4.

- a. *Paul thought [(that) Mary invited John to her Party].*
- b. *Who<sub>3</sub> did Mary think [ ( that )John invited t<sub>3</sub> to his party]]?*
- c. *What<sub>3</sub> did Mary say [(that) John saw t<sub>3</sub> ]*
- d. *What<sub>3</sub> did John believe [(that) Mary asked t<sub>3</sub> Paul].*
- e. *Who<sub>3</sub> did Mary think [(that) John kissed t<sub>3</sub>]*

---

<sup>1</sup> There are however, languages like Spanish, which appear to have an overt complementizer in matrix clauses under certain conditions. However, Spanish *que* is more like to be an abbreviation of *porque* which means because.

According to standard assumption, the syntactic structure of (1c) will be as shown in figure 1.

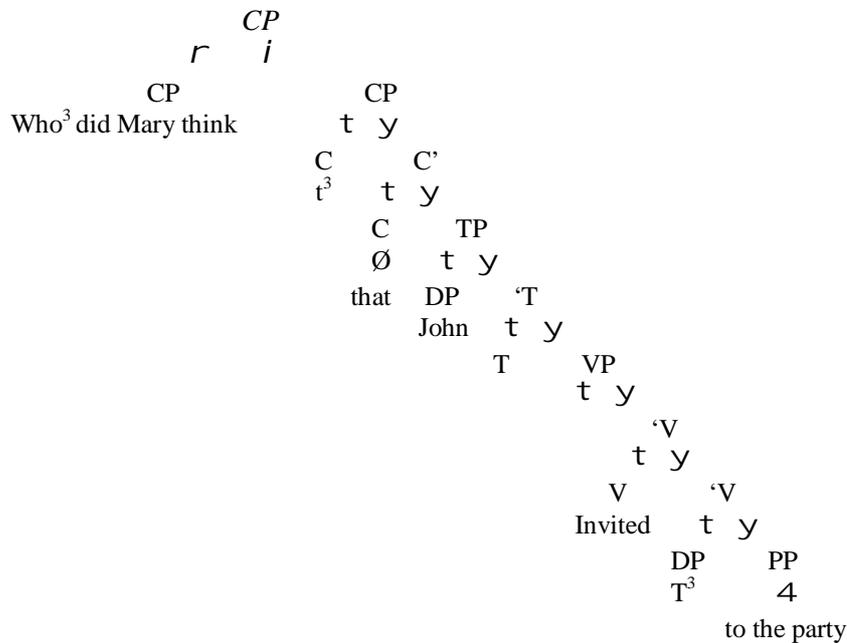


figure 1. Syntactic derivation of 1c

When a subject is moved from an embedded clause into a higher clause, overt complementizers are not allowed in English. This is due to the fact that embedded CP containing an overt complementizer contains a TP complement with a trace in its specifier position.<sup>2</sup> This is called the *That*-trace effect and is illustrated in the examples below.

<sup>2</sup> When a complementizer is not overt, this does not mean that there is no complementizer. The CP is always headed by a complementizer. Alternatively there is a null-complementizer and an overt allomorph of the complementizer.

5.

a. *Who<sub>7</sub> did Mary think [\_ t<sub>7</sub> invited John to the party]]?*

b. \**Who<sub>7</sub> did Mary think [that t<sub>7</sub> invited John to the party]]*

c. *Who<sub>3</sub> did Mary say [\_ t<sub>3</sub> kissed John]*

d. \**Who<sub>3</sub> did Mary say [ that t<sub>3</sub> kissed John]*

e. *Who<sub>3</sub> did Mary believe [\_t<sub>3</sub> left the party early].*

The syntactic trees of sentence (5a) and (5b) are illustrated in figures 3 and 4 repeatedly.

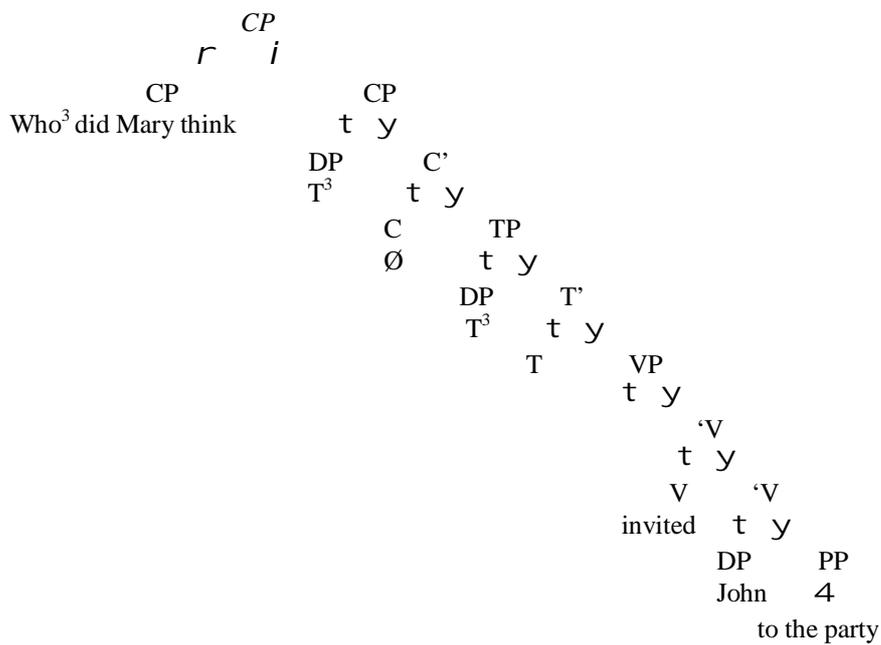


figure 2. Syntactic derivation of 2a

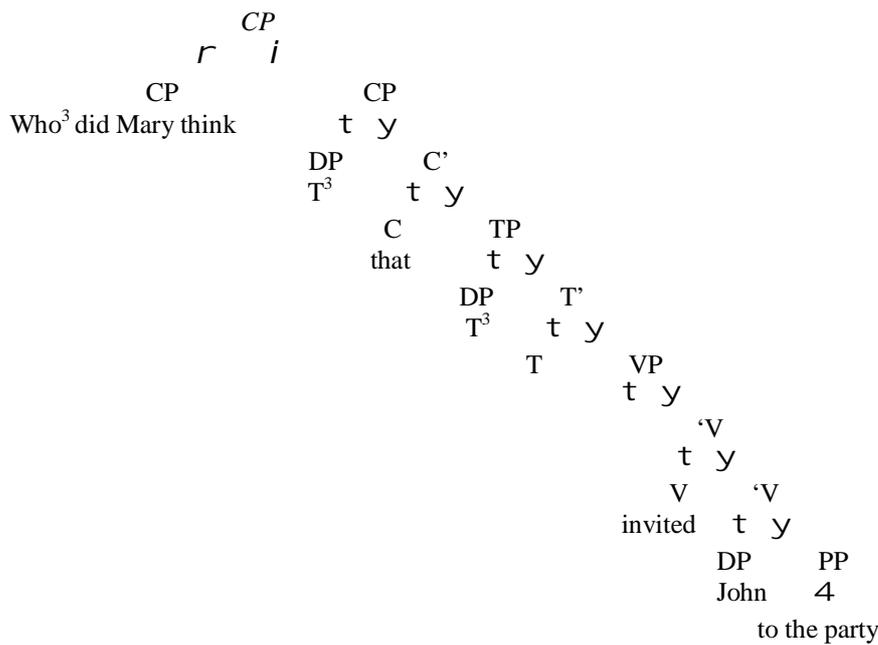


figure 3. Syntactic derivation of the ungrammatical example 2b

In contrast, a *That*-trace effect does not occur in embedded clauses in which a subject is moved from the embedded clause to a higher CP via the embedded [SPEC, CP] of an embedded clause that starts with an Adverbial phrase or a prepositional phrase. In this situation, the complementizer is also optional, even though there is a subject trace in the specifier position of T. Taraldsen (1986) described a similar structure in Norwegian and called it an "anti *that*-trace effect" because it nullifies the usual effect of ungrammaticality. The anti *That*-trace effect is illustrated in the following English examples. The syntactic derivation of example 3a is illustrated in figure 4.

6.

a. *Who<sub>7</sub> did Mary think [ that for all intends and purposes t<sub>7</sub> invited John to the party]]?*

b. *Who<sub>7</sub> did Mary think [\_for all intends and purposes t<sub>7</sub> invited John to the party]]?*

c. *?Who<sub>3</sub> did John say [(that) secretly t<sub>3</sub> left the party early].*

d. *Who<sub>3</sub> did John say[ that of all people t<sub>3</sub> left the party early].*

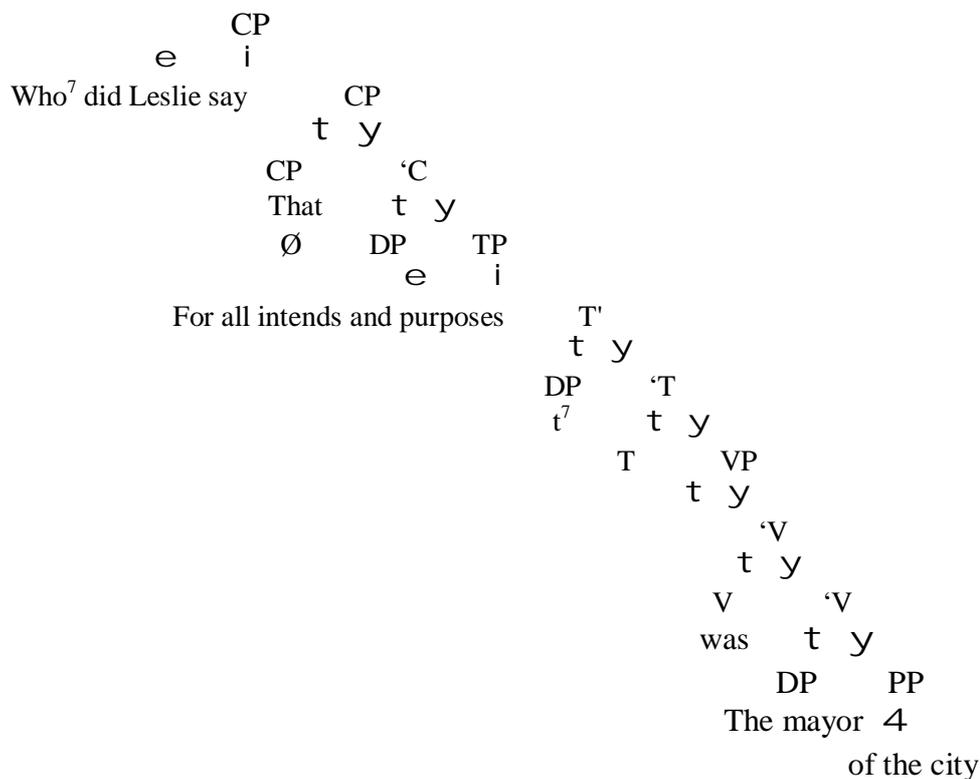


figure 4. Syntactic derivation of an anti *That*-trace effect.

It is argued that *that* is optional, because the overt complementizer does not immediately precede the trace of movement. Instead, there is another phrase in between the CP and the TP, therefore the derivation adheres to the normal rules of optional overt complementizers in embedded clauses.

Apart from optional and prohibited overt complementizers, English sometimes has obligatory complementizers as well. One example of a context in which complementizers have to be overt is the sentential subject. When a sentence functions as a subject, it always has to start with an overt complementizer, as can be seen from the minimal pair in 7 and the syntactic derivation of example 4a which is shown in figure 5.

7.

- a. [That Sue will buy the book] was expected by everyone.
- b. \*[\_ Sue will buy the book] was expected by everone.

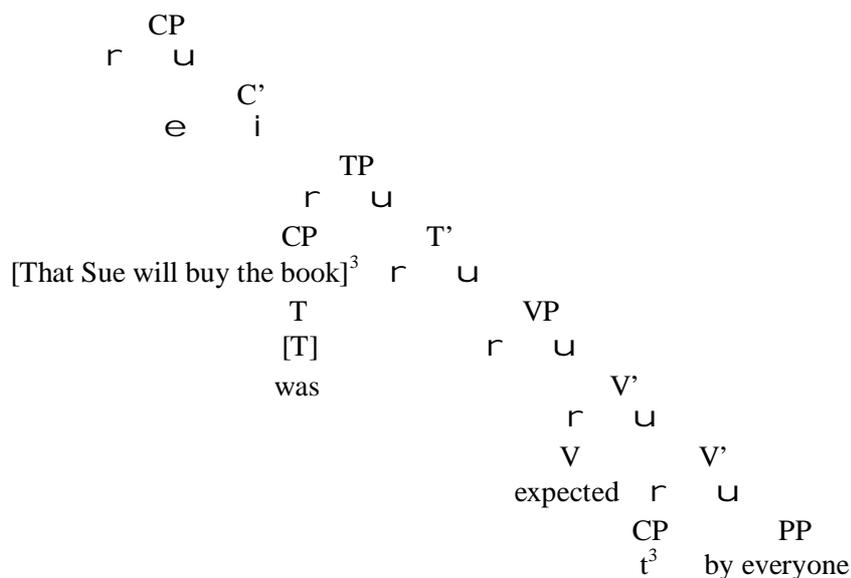


figure 5. Syntactic derivation of sentential subject.

Complementizers are also obligatory in relative sentences, which do not begin with *who*. The English language thus has context in which complementizers can be optional, obligatory or prohibited. In embedded sentences, overt complementizers are optional except when the CP containing *that* contains a complement TP with a trace of movement in its specifier position, which is called the *That*-trace effect.

### 1.3. Previous Literature

In the Government and Binding framework, the English *That*-trace effect was first dealt with by positing a language specific filter which did not allow an overt complementizer to be followed by a trace (Chomsky and Lasnik, 1977). This was called the *That*-trace Filter. In this proposal, language variation was explained by adding parametric variation to this filter. Languages either allowed for traces to immediately follow a complementizer or they did not. In the following years, this proposal was replaced by the Nominative Island constraint (Chomsky 1981), according to which a nominative trace must always be bound in a clause (i.e. in a CP). Following this proposal, Pesetsky(1982) argued that since wh-phrases behave like nominative anaphors, wh-subject traces which bear nominative case should be bound within the CP. Because all functional categories are barriers for movement since a CP is a functional category, it is a barrier for movement. A nominative trace in [SPEC,TP] must furthermore be properly antecedent governed. Antecedent government of a trace occurs when a trace is c-commanded and governed by a non-functional element. Since a subject trace is c-commanded and governed by C, which is a non-functional category, the trace is thus not properly governed because the only element that c-commands it is CP.

Following Chomsky, Pesetsky(1982) proposed the Doubly Filled Comp Filter (DFCF). This prohibited the co-occurrence of an overt complementizer and a trace in the CP. Either the overt complementizer or the trace had to be deleted in order for the sentence to be grammatical. Once again, this proposal assumed parametric variation to deal with language variety.

Subsequently, Rizzi(1991) revised the Empty Category Principle (ECP), which is based on Chomsky's(1981) original version of the ECP. This theory followed Chomsky's

ideas that traces need to be bound and governed by distinct categories and furthermore followed Chomsky's definition of Empty Categories which states that empty categories should either be properly head-governed or antecedent governed. According to Rizzi, traces are subject to the ECP. He argues that traces should be bound by non-functional categories because these are barriers for government. According to Rizzi, CPs are barriers for movement. Rizzi's view on government causes *That*-trace effects to occur only in SVO languages because in VSO and SOV languages T governs the subject and therefore also the subject trace.

This argument is backed up by the fact that English is an SVO language and that the *That*-trace effect do not occur in languages like Dutch and German, which are SOV languages. However, all of these proposals deal with the classic *That*-trace effect that occurs in English when a subject trace is moved via [SPEC, CP] into a higher clause, but they cannot deal with the other effects described in the previous chapter because they are only concerned with subject traces that follow a complementizer. In particular, they do not explain the optionality of the complementizer *that* in embedded clauses from which an object phrase is extracted.

In addition, these earlier theories could not explain the Anti *That*-trace effects which occur when an AP occurs as a complement of the CP. This is a big problem because a theory that deals with issues like the *That*-trace effect should be able to deal with all of the related phenomena. With this in mind, a new theory was constructed by Pesetsky and Torrego (2001). This theory was also based on a new framework, namely The Minimalist Framework of Chomsky (1994). However, in order to fully understand the account of Pesetsky and Torrego, one must first have basic knowledge of Minimalism. Therefore, in the next chapter I will digress from the previous literature on that *That*-trace effects and explain the basic assumptions of Minimalism.

#### 1.4. Minimalism

Just like previous linguistic frameworks, Chomsky's *Minimalist Programme* was developed in order to create one theory that could deal with all language variety that exists, or could exist. In Minimalism, it is assumed that all languages have the same original sentence structure and that language variety exists because of different language features. Derivations are built from the bottom up rather than top down. Derivations thus always start with a VP. In this framework it is assumed that movement is not optional but triggered. The most important trigger to move elements in syntactic derivations is feature checking. Whether or not features exist on a given phrase can differ from language to language, and even in different phrases and words within a language and this is probably one of the main explanations for cross linguistic variety.

Syntactic derivations are made using two distinct operations, merge and move. When a new phrase is added to the derivation and is attached to another phrase, this is called merge. When an element is copied into another part of the derivation, this is called move. Movement always leaves a copy in the original position of the moved element, which is not pronounced in spell-out. This copy is called a trace.

In minimalism, the features phrases can have are of two basic types. There are grammatical features that make no semantic contribution to the sentence, like tense on the DP or gender on T. These are called uninterpretable features. On the other hand there are also features that do make a semantic contribution to the phrase. The tense feature on T is one of these. These features are called interpretable features. A feature that is uninterpretable on T might well be interpretable on some other phrases. For example, gender features make no semantic contribution on T but do make a significant semantic difference on a DP headed by a pronoun.

According to Chomsky, all uninterpretable features must be deleted at the end of the derivation. An uninterpretable feature can only be deleted when it is checked by an interpretable feature of the same kind. When an uninterpretable feature is checked by an interpretable feature, this is called feature checking. Feature checking uses another operation which is called Agree. Agree is the syntactic connection between two items that both carry the same feature F. After Agree takes place, the uninterpretable features are deleted. In some instances, Agree requires an element to be copied from Y into X via movement, since agreement can only take place if both the features are in a specifier-head relation with each other.

However, movement can only take place if one of the features has the EPP<sup>3</sup>-property. The EPP property is not a feature but a property of certain features which enables the feature to attract other features, thus allowing for movement. If a phrase bearing a given uninterpretable feature, which has the EPP property, is in an Agree relation with another phrase which bears an interpretable feature of the same kind, the phrase which bears the interpretable feature is copied into the specifier position of the uninterpretable feature. This operation of movement leaves a trace of movement which is an exact copy of the moved element. This type of movement is also called move  $\alpha$ .

Elements only enter an Agree relation with another element if absolutely necessary. This is due to the Economy Principle (EP) which states that a head only triggers the minimum number of movement operations necessary to satisfy the properties of its uninterpretable features.

In this thesis I will illustrate feature checking in trees with brackets. I will use square brackets for interpretable features and parentheses for uninterpretable features that need to be checked by interpretable features, as is illustrated in the following example.

---

<sup>3</sup> EPP properties allow the head to attract another feature. Without the EPP-property they need to be in a SPEC-head agreement relationship with another feature in order to allow for feature-checking.

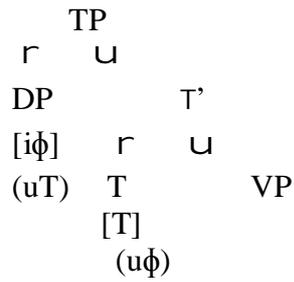


figure 5. Feature checking in trees

In this example, both  $\phi$ -features and tense features occur. The head T has interpretable tense features and uninterpretable  $\phi$ -features while the head of the DP has interpretable  $\phi$ -features and uninterpretable tense features. They can check each-other because they are in a [SPEC, head] relationship. After checking, the uninterpretable features are deleted.

According to Pesetsky and Torrego, the work that has been done on the *That*-trace effect in the 1970s and 1980s met obstacles that could not be successfully overcome at the time because of the theoretical framework in which they were constructed and they furthermore state that their theory can overcome these obstacles because it is set in the Minimalist Framework.

## 1.5. Pesetsky and Torrego: T-to-C movement

Pesetsky and Torrego(2001) argue that nominative case on D is actually uninterpretable tense. According to the authors, there is also an uninterpretable tense feature on C. Somewhat in contrast to mainstream minimalism, they propose that this uninterpretable tense feature on C can either be checked by the interpretable Tense feature on T or by the uninterpretable tense feature on D. The uninterpretable tense feature can be checked by an auxiliary because the auxiliary represents the T feature in T. This is what usually happens in an English interrogative sentence and is illustrated in figure 6. (Here and henceforth broken arrows will indicate movement and solid arrows feature checking.)

8. *Who<sup>7</sup> will Mary kiss t<sup>7?</sup>*

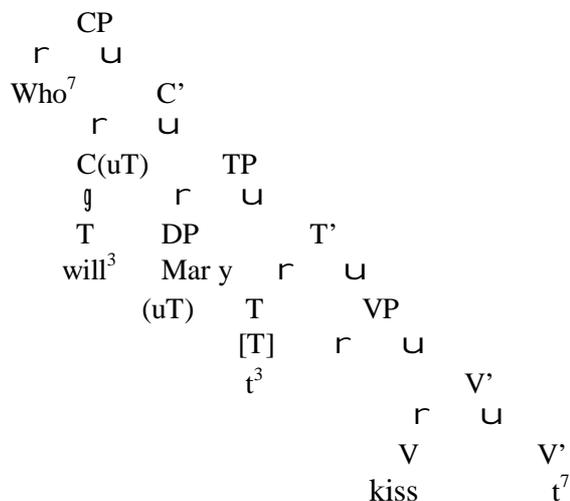


figure 6. Syntactic derivation of 8.

They furthermore argue that *that* in C is no complementizer but merely the phonetic realisation of T moved to C when there is no complementizer in T. When the uninterpretable tense feature on C is checked by the interpretable tense feature on empty T, *that* is overt in C, as is illustrated in the following example.

9. *Who<sup>7</sup> did Bill think [CP that Mary kissed t<sup>7</sup>]?*

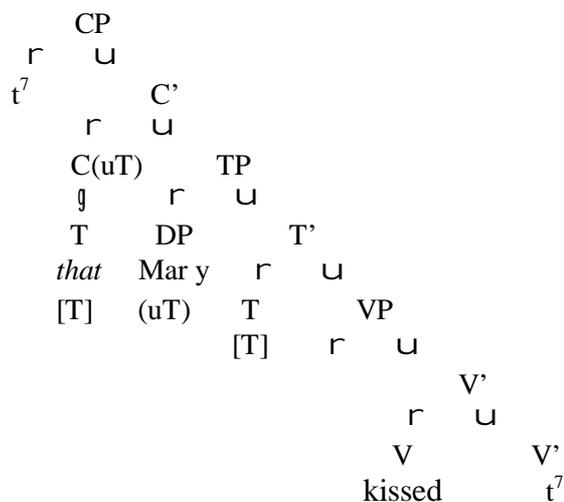


Figure 7. Syntactic derivation of 9.

On the other hand, the uninterpretable tense feature on C can also be checked by the uninterpretable tense feature on the nominative subject when the nominative subject is moved to [SPEC, CP]. First, The uninterpretable tense feature on CP is checked by the interpretable tense feature on T via agreement because the subject and T are in a specifier-head relation. In mainstream Minimalism, the uninterpretable tense feature would be deleted when it is checked. However, Pesetsky and Torrego argue that features are not immediately deleted but merely marked for eventual deletion. Therefore, the uninterpretable tense feature on D can "live on" after being checked by D until the CP is completed. The uninterpretable tense feature then moves to [SPEC, CP] in order to check the uninterpretable tense feature on C. This type of movement is illustrated in example (10).

10. *Who<sup>3</sup> kissed Mary?*

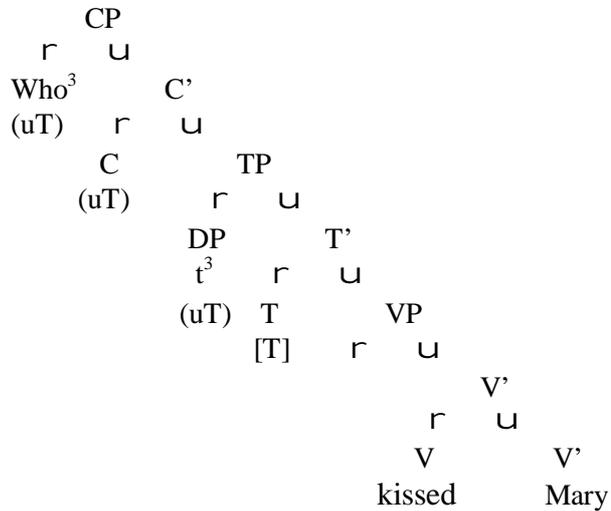


Figure 8. Syntactic derivation of 10.

Although English matrix interrogative clauses never begin with *that*, English embedded declarative clauses sometimes do. When they do not start with *that*, the uninterpretable tense feature on C is checked by the uninterpretable tense feature on C. This is illustrated by the following example of an embedded declarative clause.

11. *Bill thought [<sub>CP</sub> John<sup>3</sup> t<sup>3</sup> kissed Mary].*





in two ways even though *that* is not in front of the sentence. The difference is illustrated in the following two sentences.

13.

a. *What<sup>2</sup> did mary give Paul t<sup>2</sup>?*

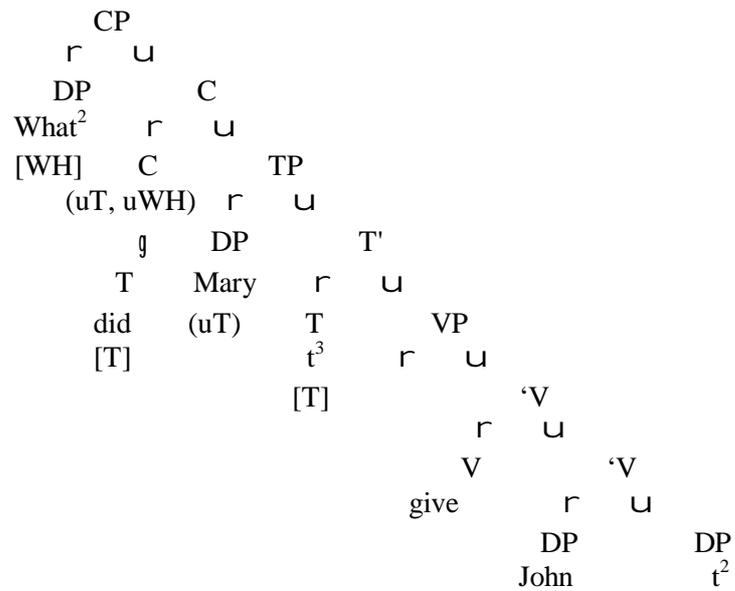


figure 11. Syntactic derivation of interrogative clause with object movement.

b. *Who<sup>2</sup> t<sup>2</sup> gave Mary a present?*

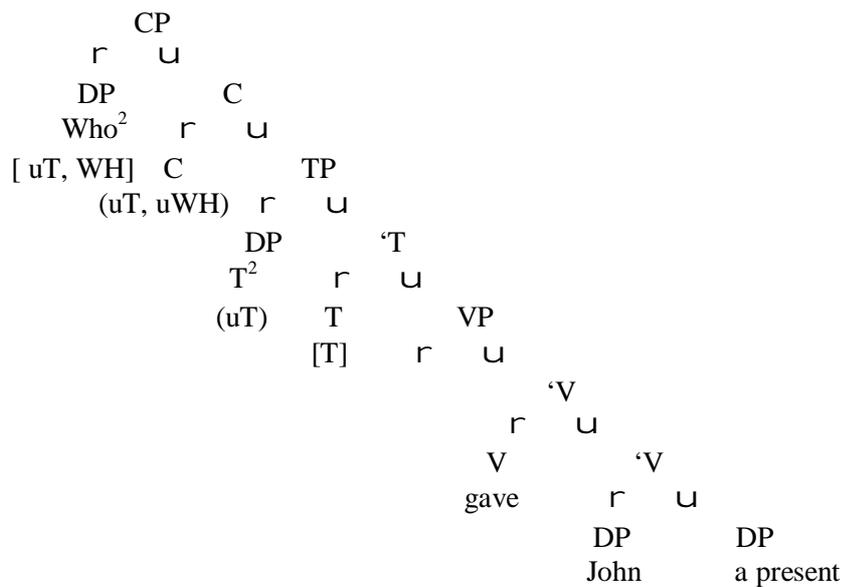


figure 11. Syntactic derivation of interrogative clause with object movement.

In these examples, C is either checked by the nominative wh-element or by movement of T to C. Although Pesetsky and Torrego argue that T to C movement is realised with *that*, matrix questions never begin with *that*. Their explanation for this is that matrix questions, in which the object wh-phrase is moved to C in order to delete the uninterpretable WH-features on C do not start with *that* because the auxiliary is moved to C. As a consequence, matrix questions with an object wh-phrase always have an auxiliary. Thus, when T is moved to C of a clause in which there is only one verb, *that* is the realisation of T moved to C, while in sentences in which there is an auxiliary there is no *that* because C is already filled with the auxiliary, since the auxiliary has already moved to C.

In English embedded sentences, *that* is usually optional. Sentences with long-distant wh-movement of the object is one context that allows this optionality of *that*. The optional occurrence of *that* is illustrated in the following examples.

14.

a. *What<sup>2</sup> did Mary think John gave t<sup>2</sup> to Mary*

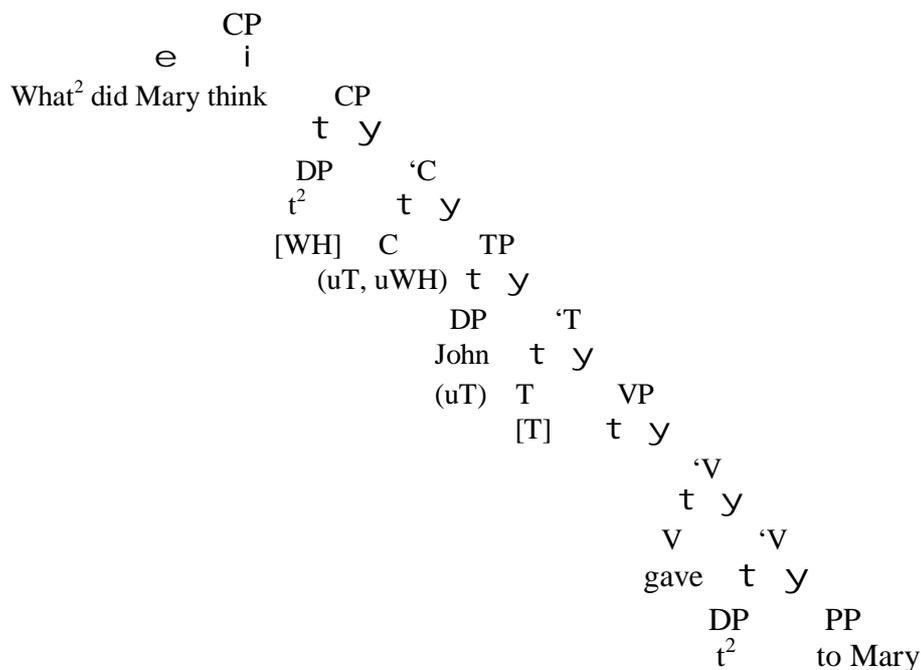


figure 12. syntactic derivation of long-distant object wh-movement.

- b. *What<sup>3</sup> did Mary think that John gave t<sup>2</sup> to Bill*

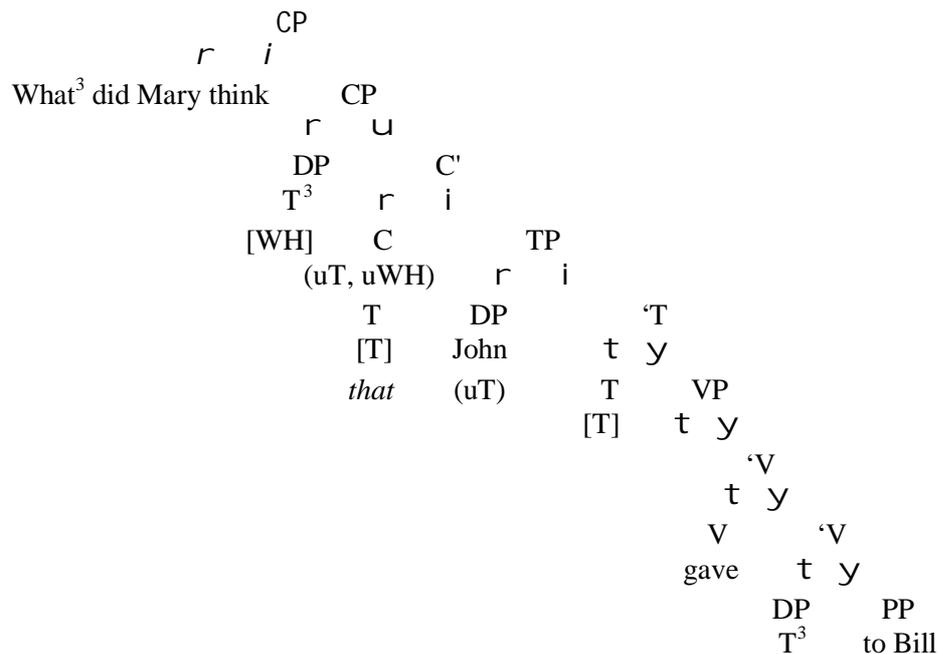


figure 13. syntactic derivation of long-distance object wh-movement.

As illustrated in the examples above, there are two different ways to check the uninterpretable tense feature on embedded C. This is caused by the fact that the uninterpretable wh-feature on C cannot be checked in the same operation as that in which the uninterpretable Tense feature on C is checked. This because the wh-element is an object and therefore not equidistant to the TP, which can check the uninterpretable tense feature on C. In both sentences, the uninterpretable wh-feature on C is checked by moving the wh-object to the [SPEC, CP] position. The difference between the two sentences is in the way in which the uninterpretable tense feature is checked. In the second example, the interpretable tense feature on T checks the uninterpretable tense feature on C. Consequently, *that* is overt in C because it is the phonetic realisation of moving T to C. In the first example, the nominative subject is moved to [SPEC, CP] in order to check the uninterpretable tense feature on C. Both options are possible because the nominative subject and T are equidistant to C. Therefore, the

derivation can choose between the two options because both are equally economical, causing the optionality of *that*.

The second construction in which *that* is optional is a kind of relative clauses. Relative clauses can either begin with either *who* or *that*. Relative clauses are embedded declarative sentences of which C has both an uninterpretable tense and an uninterpretable wh-feature. The following two examples illustrate the analysis.

15.

a. *This is [the man that won the lottery].*

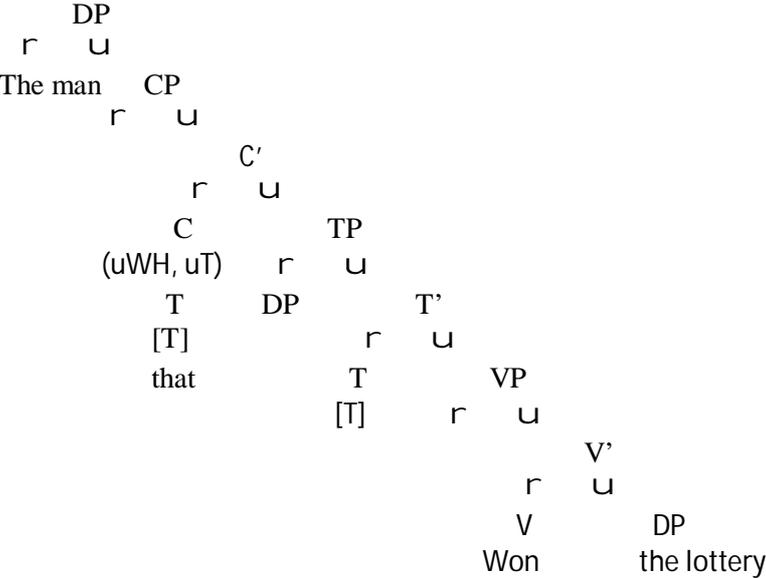


figure 14. Relative clause with overt complementizer

b. *This is [ the man who<sup>2</sup> t<sup>2</sup> won the lottery].*

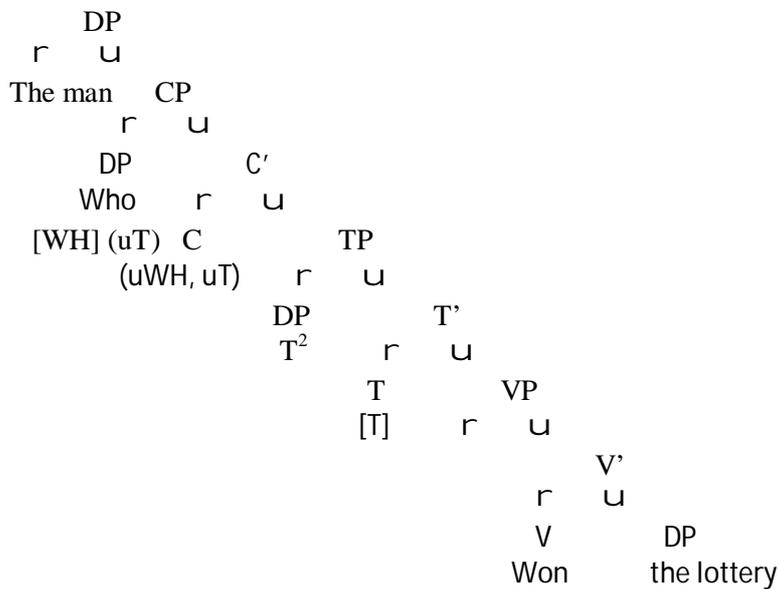


figure 15. relative clause with null complementizer

In this example, the uninterpretable tense feature on C is checked by the interpretable feature on T, causing *that* to appear as a realisation of T moved to C. There is no wh-element in this sentence, therefore the uninterpretable wh-feature on C cannot be checked. In example (12b), on the other hand, the embedded sentence does not begin with *that*, because both the uninterpretable tense feature and the uninterpretable wh-feature can be checked in one operation, by movement of the nominative subject.

The last construction in which *that* is optional is in embedded sentences in which a phrase is situated at the front of the embedded clause. This situation, which is illustrated in the example below, is problematic for the theory proposed by Pesetsky and Torrego.

16. *Who<sup>7</sup> did Leslie say t<sup>7</sup> that, secretly, t<sup>7</sup> kissed Mary.*

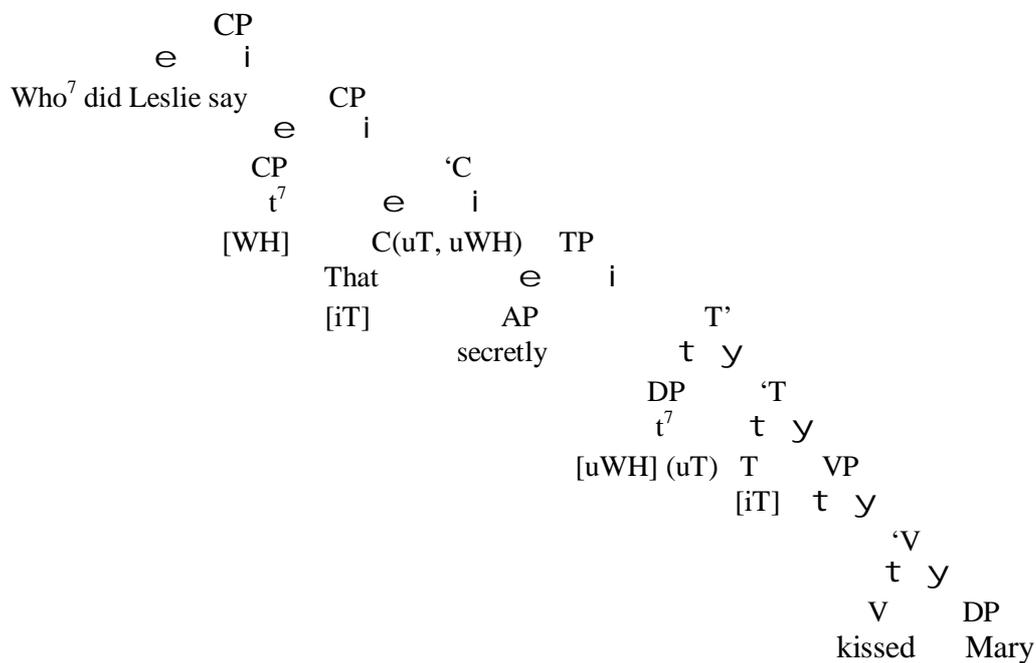


figure 16. topicalised embedded sentence

This type of sentences is problematic because the nominative subject wh-phrase has to move to [SPEC, CP] in order to check the uninterpretable WH-features on C and the existence of *that* shows that the uninterpretable Tense feature can both be checked by the interpretable Tense feature on T and by moving the nominative subject to [SPEC, CP]. This is an issue because the Economy Condition dictates that all features should be checked via the least possible number of operations. Pesetsky and Torrego solve this problem by stipulating that these kind of adverbial phrases are topicalised and thus situated in [SPEC, TP]. Because of this, the nominative subject and TP are no longer equidistant to C. Therefore, it is preferable for the interpretable Tense feature on T to delete the uninterpretable Tense feature on C. After this operation, the uninterpretable wh-feature on C is checked by the interpretable wh-feature on the nominative subject. The problem is that this predicts that *that* should always be overt in

these kinds of sentences, yet native speakers do not agree this is the case because it is often acceptable to delete *that* in these kinds of sentences. Pesetsky and Torrego do not propose a solution for this and assume that, in spite of closeness conditions, the derivation can also choose the nominative subject in order to check both the uninterpretable *wh*-features and uninterpretable tense features. However, if one assumes that a topicalised item does not occur in [SPEC, TP] but in a separate phrase, there does not seem to be a problem because the nominative subject and T are still equidistant then.

In Pesetsky and Torrego's theory, the traditional *That*-trace effect occurs in an embedded declarative sentence whose C-head contains both an uninterpretable tense feature with the EPP property and a WH-feature with the EPP property. Apart from this, it contains a nominative *wh*-phrase. Consider the sentence in (17) and its proposed analysis in figure 17 below.

17.

- a. *Who<sup>3</sup> did Mary think [<sub>CP</sub> t<sup>7</sup> invited John to the party]]?*

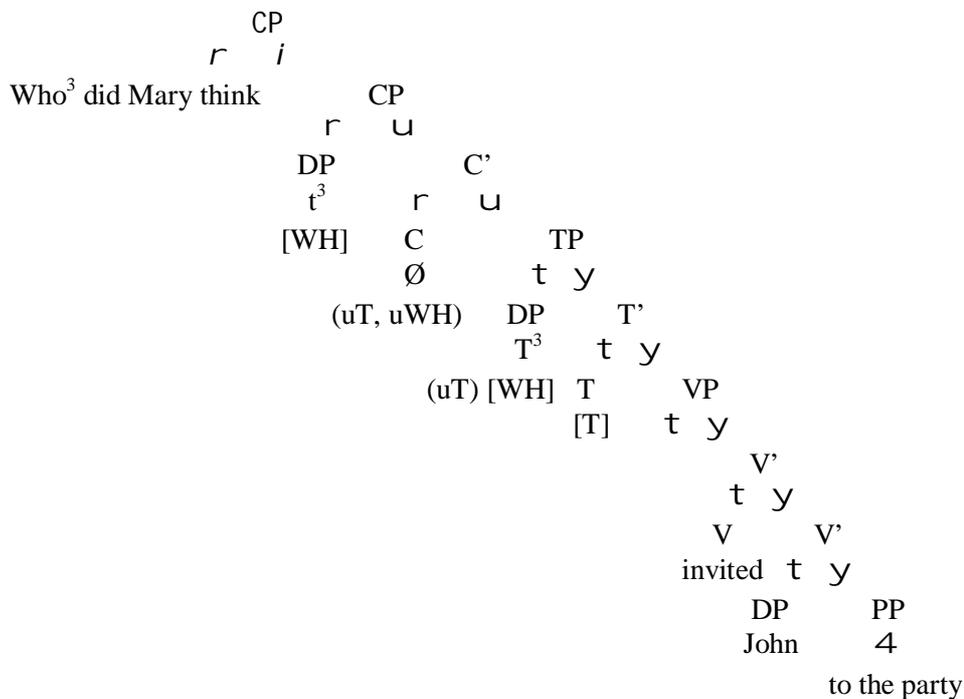


figure 17. The classic *That*-trace effect

In the example above, both the uninterpretable Tense and the uninterpretable WH features are deleted by the nominative subject. According to Pesetsky and Torrego, T cannot move to C in order to delete the uninterpretable tense feature. They argue that although the subject DP and T are equidistant from C, only nominative case on the DP can check both the uninterpretable tense feature and the wh-feature on C because of the EP. The nominative subject can check both the uninterpretable WH-feature and the uninterpretable tense feature on C, while movement of the interpretable tense feature on T only checks the uninterpretable tense feature on C, leaving the uninterpretable WH-feature unchecked. The latter thus has to be checked via another movement operation, movement of the nominative subject. Since movement of the nominative subject can check both uninterpretable features on C, movement of T to C is the less economical one. For this reason, only movement of the nominative subject is allowed here and *that* never occurs at the beginning of such an embedded clause.

The last situation the theory needs to be able to account for is the obligatory use of *that*. According to Pesetsky, when a sentence is used as a subject and obligatorily starts with *that*, *that* marks a clause as nominative and this is precisely what allows clauses to be used as a subject. This is illustrated in the following example.

18. [That Sue will buy the book] was expected by everyone.

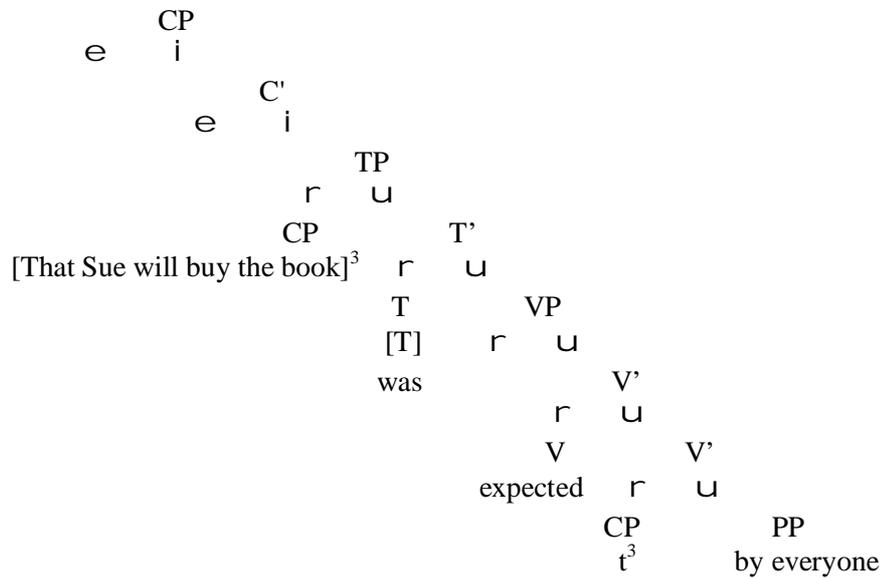


figure 18. Syntactic derivation of a sentential subject.

In short, Pesetsky and Torrego's theory adequately deals with all different situations in which complementizers occur. For English their theory seems to work well.

## 1.5. Critical assessment

Although Pesetsky and Torrego's theory seems to work well for English, they remain quite speculative in their work, their claims are hardly ever backed up by any evidence and most of their ideas do not hold up for other languages.

A first problem concerns one of their core assumptions that *that* in English is not a complementizer but merely a phonetic realisation of T moved to C. This entails that complementizers are never overt in English C. In previous literature *that* is always viewed as an overt complementizer, because it has no semantic function. It merely functions as a marker for subordination. Complementizers mark the border of an embedded sentence. If *that* is not a complementizer, this means that overt C in English does not exist. This goes against standard assumptions in X-bar theory, in which it is assumed that a category can be phonetically null in certain contexts but always has an overt allomorph in some contexts too.

Another problem concerns relative clauses that either start with a *wh*-phrase or with *that*, regardless whether the *wh*-phrase is moved from the object or the subject position. To capture this optionality here Pesetsky and Torrego must allow the EP to be violated. Consider the derivation of relative clauses illustrated below.

19. *The man<sup>7</sup> that/who we saw t<sup>7</sup> yesterday*

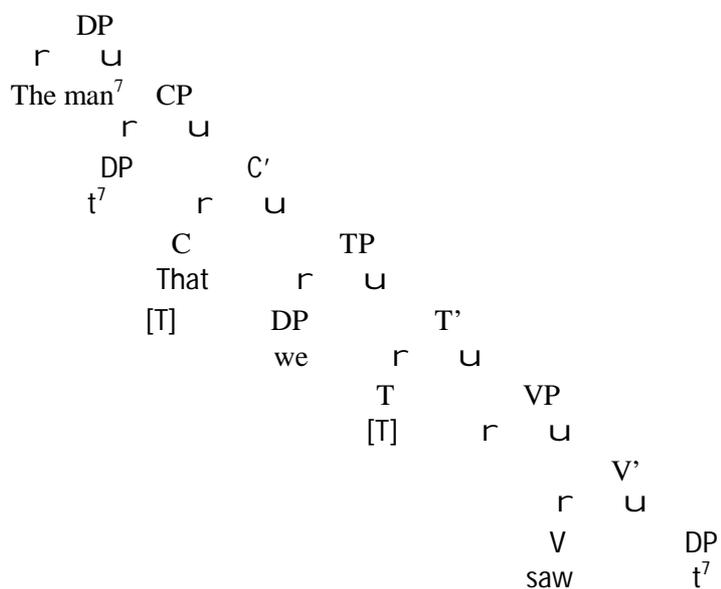
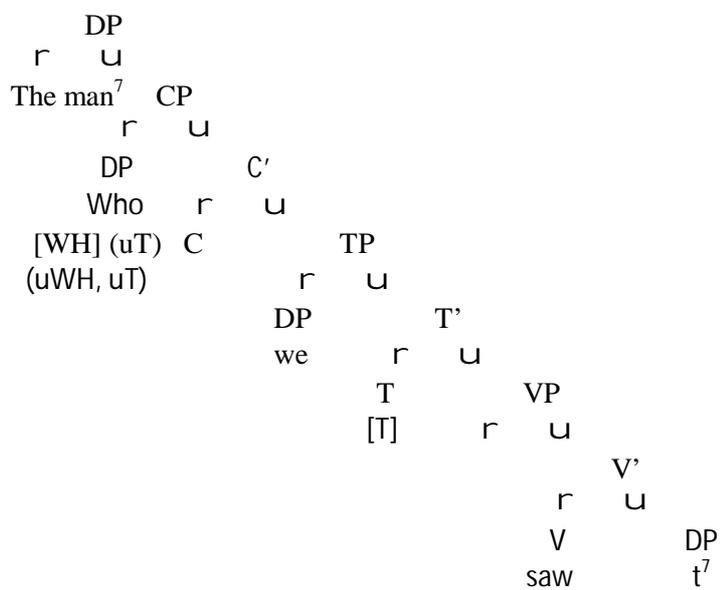
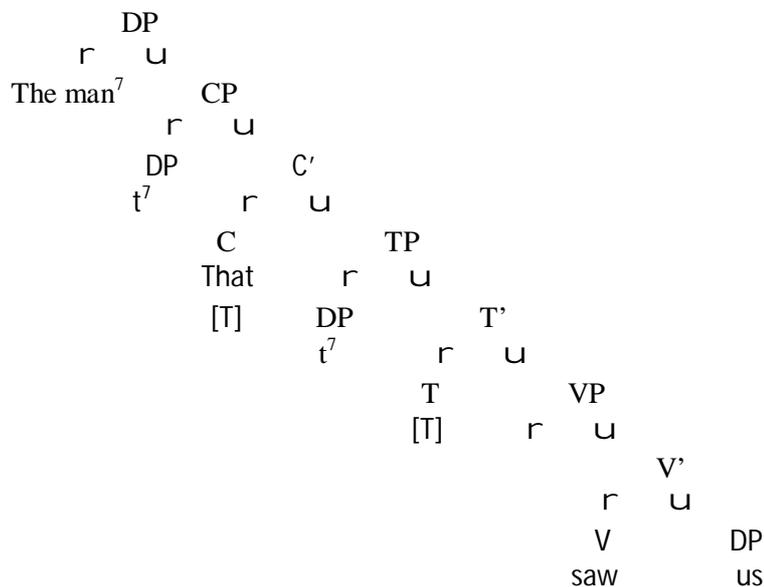


figure 19. syntactic derivations of two relative clauses with object extraction

As can be seen in the two examples above, *that* can be overt in C, when T has moved to C. first the object moves to [SPEC, CP] and leaves a trace there. Then T moves to C in order to check the uninterpretable tense features on C. *That* is overt in C because of this movement. A relative clause with *who*, on the other hand, occurs as illustrated in the first example. *Who* moves to [SPEC, CP] and checks the uninterpretable WH-features on C. However, since it is an object phrase, *who* cannot check the uninterpretable Tense features. According to Pesetsky and Torrego T should move to C in order check the uninterpretable tense features, but apparently this does not happen since *that* does not appear in C. This difference cannot be accounted for in the theory as presented by Pesetsky and Torrego. In relative sentences with a subject movement the construction is illustrated below.

20. *The man<sup>7</sup> that/who t<sup>7</sup> saw us yesterday.*



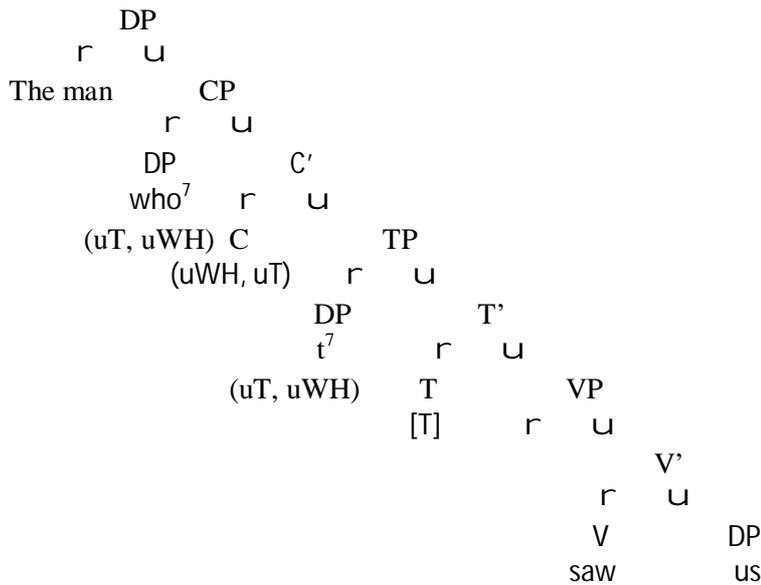


figure 20. syntactic derivations with subject movement.

As illustrated in the sentences above, *who* is moved from the subject position to the beginning of the embedded clause and can delete both the uninterpretable wh-features and the uninterpretable Tense features on C. However, *that* is the realisation of T moved to C, which occurs even though the nominative subject in the first example is moved to [SPEC, CP] and can also delete the uninterpretable T features. When this happens there is no need for T to move to C and economy principles are violated. This has serious consequences for the theory as proposed by Pesetsky and Torrego.

A more plausible account of relative clauses is given by Breznan (1970), who argues that wh-phrases can function as complementizers in clauses which have a C with a Q-morpheme. When *that* is viewed as a complementizer as well, this can also account for the fact that in otherwise exactly the same relative clauses, *who* and *that* can both be in C to indicate the left border of an embedded clause.

Pesetsky and Torrego furthermore argue that uninterpretable features are not deleted after checking but live on to check other features. This does away with the most basic principle of Minimalism, which is that movement is not optional but triggered. If

uninterpretable features can check all other uninterpretable features, phrases can move to positions where they should definitely not be and it allows for a tremendous amount of different movement operations causing movement to become less and less restricted. This could have severe consequences for sentences, as is illustrated in the following example.

21. \**Given<sup>4</sup> was t<sup>4</sup> a flower to my mother*

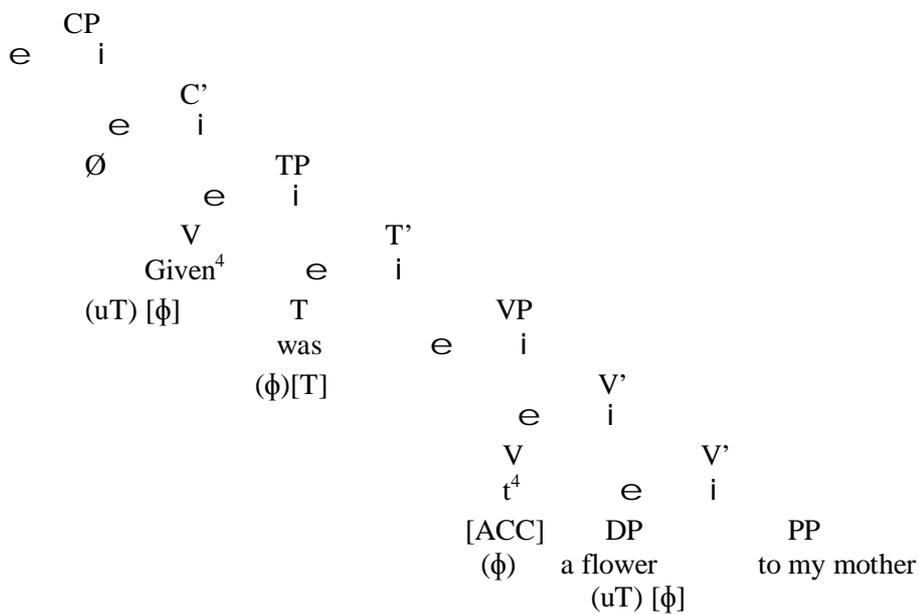


figure 21. Syntactic derivation of a passive sentence.

As is shown in this example, V also has case features and uninterpretable  $\phi$ -features. These uninterpretable  $\phi$ -features are checked by the direct object. If Pesetsky and Torrego are right that uninterpretable features can "live on" and check other uninterpretable features after being checked, both the verb and the direct object can move to [SPEC, TP] to check the uninterpretable  $\phi$ -features on T. This is problematic for the theory as proposed by the authors because the theory falsely predicts that example 19 will be grammatical.

Furthermore, Pesetsky and Torrego argue that their theory covers cross-linguistic differences. If their theory is applied to Dutch, it falsely predicts that the Dutch

complementizers *dat* and *die* will be optional in embedded sentences as well. However, in Dutch complementizers are always obligatory in all embedded sentences. The theory thus captures the grammatical sentences, but fails to explain the ungrammaticality of Dutch embedded sentences that do not start with a complementizer. The difference is illustrated in the examples on the next page.

22.

a. *Wat<sup>3</sup> dacht Mary dat John Andy t<sup>3</sup> gegeven had?*

What thought Mary that John Andy given had?

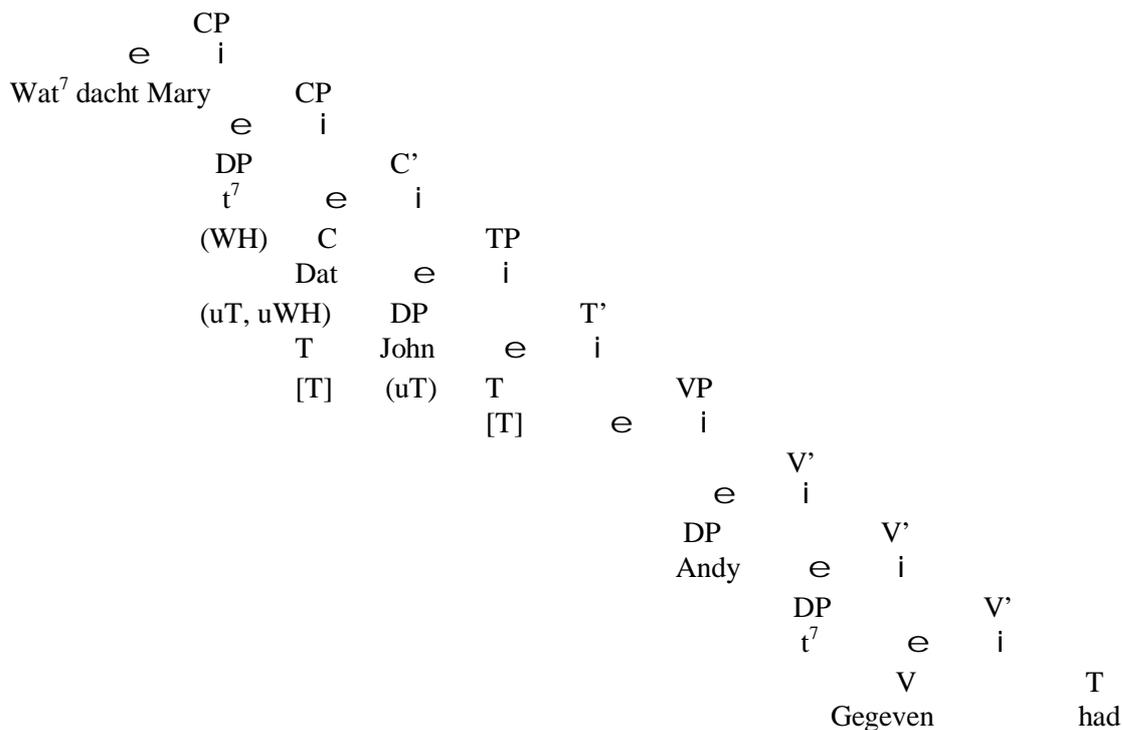


figure 22. object extraction in Dutch.

In the example above, *dat* is the realisation of T moved to C. The object wh-phrase *wat* is moved via [SPEC, CP] to the higher clause. This checks the wh-feature on C. The uninterpretable tense feature on C is then deleted via T to C movement. When T is moved to

C, *dat* is the realisation of T moved to C. This is also completely grammatical in Dutch and therefore the theory seems to be able to deal with the Dutch examples well. However, when an object wh-phrase is extracted from the embedded sentence, *dat* should be optional because the uninterpretable tense feature can also be deleted by moving the nominative subject. This is illustrated in example 23 which is ungrammatical in Dutch.

23. \**Wat<sup>3</sup> dacht Mary John Andy t<sup>3</sup> gegeven had?*

What<sup>3</sup> did Mary think John Andy t<sup>3</sup> given had?

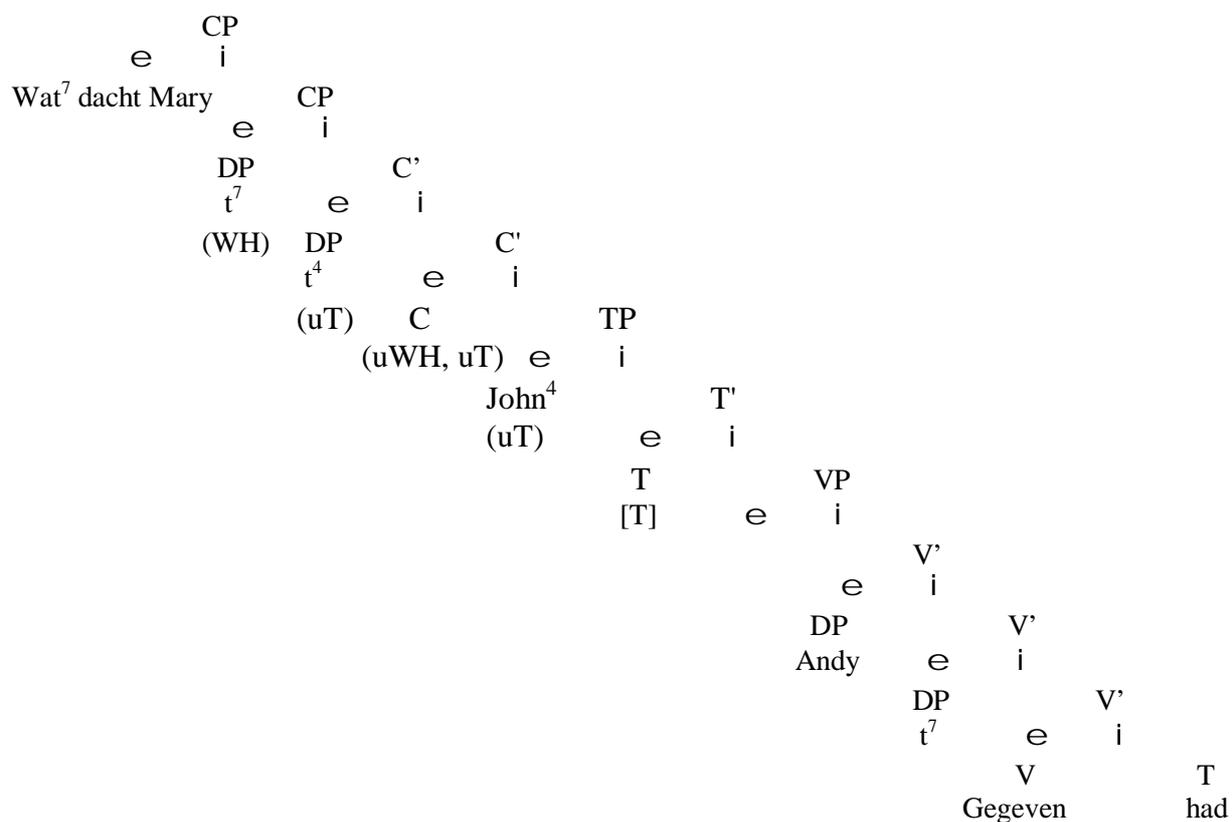


figure 23. ungrammatical sentence with object extraction in Dutch

The sentence above is ungrammatical in Dutch but the theory predicts it to be grammatical. The object wh-phrase is moved to the higher clause via [SPEC, CP] consequently checking the uninterpretable WH-feature on C. It cannot, however, check the

uninterpretable tense features. Since the uninterpretable tense feature can be checked by moving the nominative subject to [SPEC, CP], the theory predicts that this sentence will be completely grammatical, contrary to fact.

When a subject wh-phrase is moved from an embedded clause into a higher clause the situation is even worse because the theory predicts that *dat* will not be allowed in embedded C because the wh-phrase, which moves to the higher clause checks both the uninterpretable wh-feature and the uninterpretable uT feature on C. The problem is illustrated in the following examples.

24. *Wie<sup>3</sup> dacht Mary dat t<sup>3</sup> John gekust had.*

Who<sup>3</sup> thought Mary that t<sup>3</sup> John kissed had.

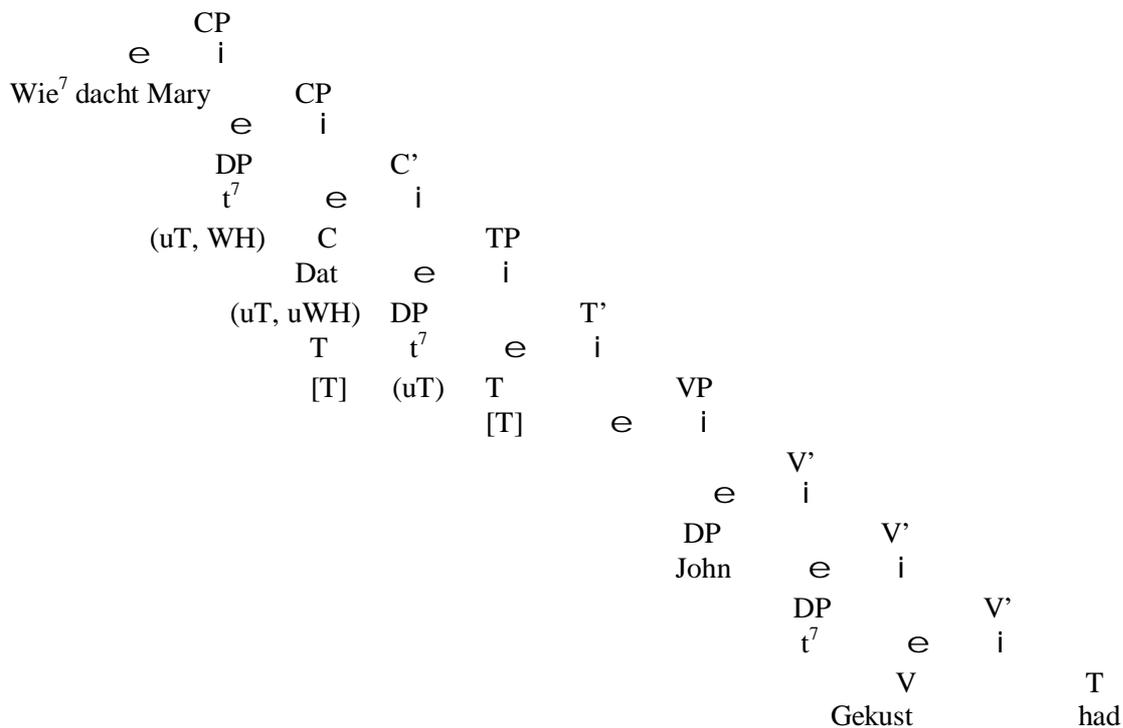


figure 24. Syntactic derivation of subject extraction in Dutch.

Since *dat* is the realisation of T moved to C and the *wh*-phrase can check both the uninterpretable Tense feature and the *wh*-feature on C, this sentence should be ungrammatical when it is in fact the only grammatical option in Dutch. Embedded sentences are always ungrammatical if they do not begin with *dat*. The following example illustrates the structure which is predicted by Pesetsky and Torrego. However, this structure is completely ungrammatical in Dutch.

25. \**Wie<sup>7</sup> dacht Mary t<sup>3</sup> John gekust had.*

Who<sup>7</sup> thought Mary t<sup>7</sup> John kissed had.

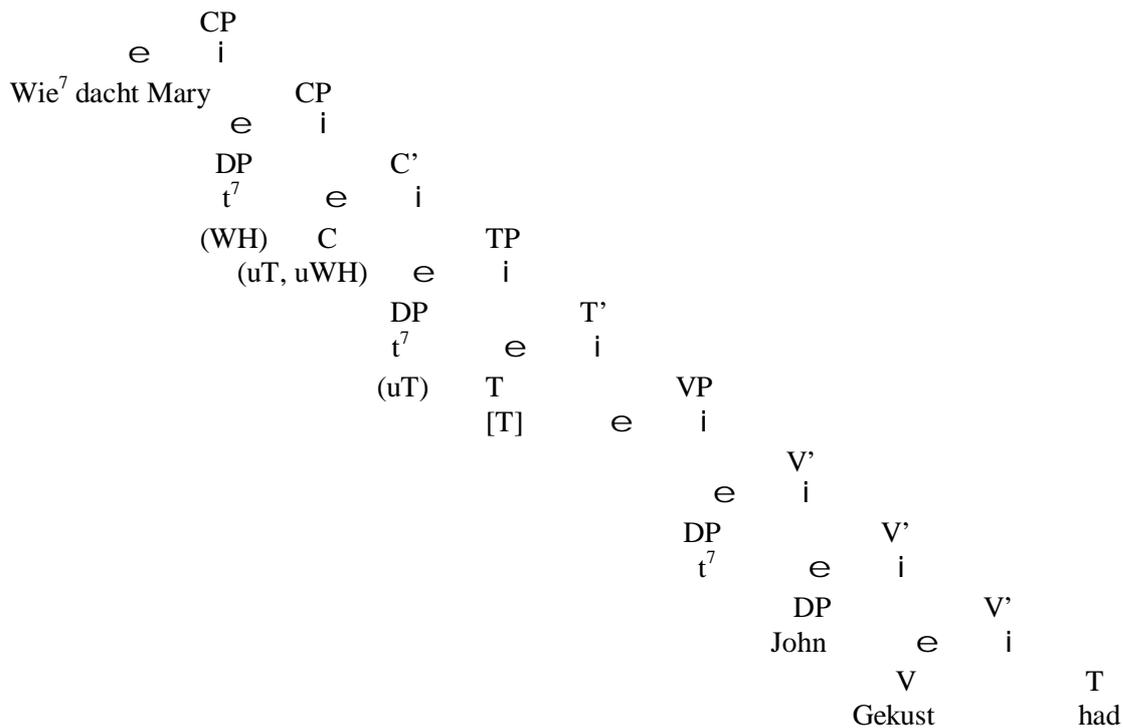


figure 25. ungrammatical sentence with subject extraction in Dutch.

As can be seen in figure (25), the uninterpretable tense feature is checked via movement of the nominative subject, which checks both features on C at once.

The last of Pesetsky and Torrego's proposals that I would like to discuss in this chapter is their proposal that *that* can mark a sentence as nominative, in order for it to become a nominative sentential subject for a matrix clause. If *that* were to mark a sentence as nominative, the consequence would be that an embedded clause which begins with *that* can never be an object in a sentence. In English, counterevidence can be found in sentences that do allow embedded object CPs to begin with *that*. These sentences are perfectly grammatical as is illustrated in the following example.

26.

- a. *She suggested [that I should go to bed early].*
- b. *He denied [that I stole my neighbour's wallet]*

If *that* does indeed mark a sentences as nominative in certain contexts then it should do so in all given contexts, for otherwise we must add some other stipulation to limit this function. However, *That* can never be the only reason subject CPs are marked as nominative. Apart from this, the proposal that C has an uninterpretable nominative tense feature which needs to be checked by interpretable tense does away with the need to mark such a sentence as nominative.

To sum up, although Pesetsky and Torrego's analysis requires that features live on after being deleted, the ramifications of such a claim are extremely negative. Moreover, allowing *That* to mark a sentence as nominative brings additional empirical problems. Furthermore, their theory does not deal with cross-linguistic differences adequately.

## 2.1. Processing

Syntactic theory does not seem able to sufficiently deal with the *that*-trace effect. Therefore I would like to argue that the *that*-trace effect has to do with a parsing mechanism rather than with syntactic theory. Therefore, it is important to first outline the parsing mechanism which will be assumed in this paper. As Pritchett (1988) notes, an important issue in the field of psycholinguistics is the relationship between grammar and the parser. It is not yet clear how closely the processor is related to the grammatical knowledge. According to most psycholinguists, including Pritchett, the grammar and the parser are not the same. However, they are related in a direct way because the parser should have direct access to the grammar in order for the grammar to monitor the processing of grammatical sentences. According to Pritchett the parser is universal while the competence grammar is language specific. The universal parser follows the rules which are specified in the competence grammar.

In most of the processing literature, the difficulties which are encountered during processing are hypothesised to be a result of ambiguities which occur during processing. Most theories argue that the problems occur when a string of words can be attached at two different places and the parser prefers the wrong attachment. This is called a garden path sentence. An example is illustrated in figures (26) and (27), which are based on example (27).

27. *The horse raced past the barn fell.*

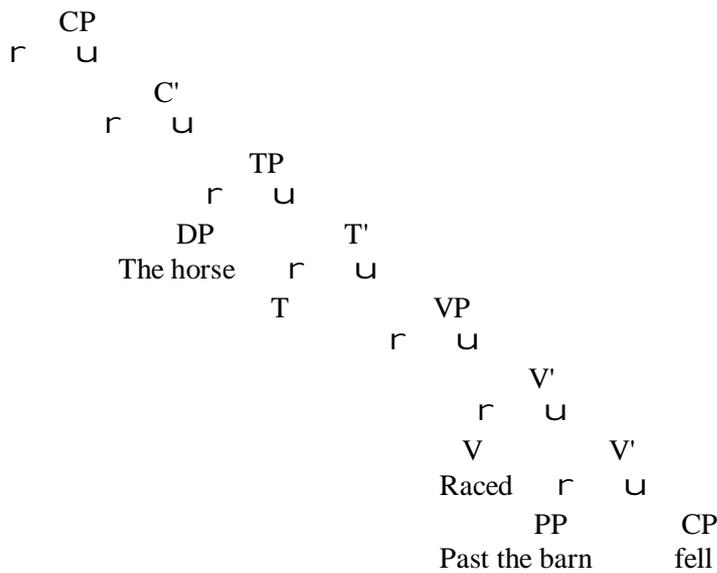


figure 26. First step in processing a garden path sentence.

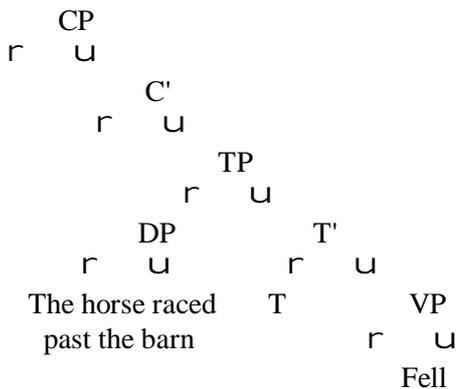


figure 27. Correct processing of the garden path.

The first example shows the way the parser first parses this structure. This sentence is perceived to be ungrammatical because the parser tries to make a matrix clause of *The horse raced past the barn*. Until the VP *fell* is reached there seems to be nothing wrong with the parse in figure 27. However, after, *fell* is encountered, the parser has to restructure the entire sentence in order to construct the correct phrase illustrated in figure 28. There are many structurally ambiguous sentences but not all of them are garden path sentences. Sentences as illustrated below are also ambiguous and people do have a general preference for one of the

two possible readings. However, they are not misparsed and certainly not garden path sentences.

28.

a. *Feeding sharks can be dangerous.*

It can be dangerous to feed a shark.

Sharks that are feeding can be dangerous.

b. *The woman shot the man with a gun.*

The woman used a gun to kill the man

The woman killed the man that was holding a gun.

Although there is more than one reading to these two examples, there is nothing structurally wrong with these sentences as opposed to the sentence in example 27. Only garden path sentences have structural processing problems.

The processor and the competence grammar need to be connected in order to distinguish one ambiguous sentence from the other. Pritchett argues that processing difficulties only occur in sentences that involve theta-theoretic difficulties of a particular sort and that these problems stem directly from the grammar. He first argues that attachment of words is subject to the theta-principles. Referring expressions (and quantified expressions) are linked to theta-roles when a basic expression containing a theta-grid is encountered. Theta-role assigners are usually verbs but can also be prepositions. When a theta-assigner is reached, its maximal theta-grid is recovered from the lexicon and an attempt is made by the processor

to match words that enter the derivation to the theta-roles which are assigned by the verb in order complete the structure. This is called theta- theta-role attachment. In the following examples, boxes are used to illustrate words are stored in working memory until they can be linked to a theta-role. These are lexical items that will violate the theta-criterion if they are not eventually linked to some theta-role.

29. *John kissed Mary.*

*John kissed Mary.*  
 " t U "  
 <<Agent >theme>

<i>John</i>
-------------

*figure 28. Processing procedure of a matrix clause.*

The parse of the sentence in example 22 is started when the word *John* is encountered. This is stored in working memory because at this point there are no theta-roles to assign. The verb *kiss* is processed next and its maximal theta-grid is recovered from the lexicon. *Kiss* has an agent and a theme to assign. In English, the agent theta-role is always assigned to the first referring element to the left of the verb. Therefore, taking John to be an agent, the processor removes it from working memory and attaches it directly to the left of the verb. After this, the processor encounters *Mary*, which is assigned the theme theta-role of the verb. The sentence is then complete and correctly processed. This example is not very complex, but Pritchett's theory also allows for more complicated sentences to be parsed. Consider the sentence in example (30).

30. *Without her contributions we failed.*

Without her contributions we failed.  
 U " " t  
 <Proposition> <Agent>

We

figure 29. Processing procedure with reanalysis.

The preposition *without* is encountered first. A preposition has one propositional theta-role to assign. This theta-role is first assigned to the next word to the right, which is *her*. After this, the word *contributions* is encountered and since there is no other theta-role available for assignment, the word *her* is reanalysed and *contributions* is attached to *her* to form one DP. This entire DP is then assigned a propositional theta-role of *without*. Next, the word *we* is stored in working memory because it cannot be attached to the previous DP and at this point there are no other theta-roles to assign. Finally, since *fail* has an agent theta-role to assign, *we* is taken out of working memory and incorporated into the parse.

In the first example given in this chapter, no reanalysis of assigned theta-roles takes place. Even though the sentence in the previous example does need a reanalysis of assigned theta-roles, it is still not a real garden path sentence.. In Garden path sentence and ambiguous sentences reanalysis is always needed. Such theta- theta-role reanalysis does not need to be problematic but it apparently is in some cases. According to Pritchett, a universal parsing constraint accounts for this. This constraint is given in example 28 below.

31.

**-Theta Reanalysis Constraint:** *Reanalysis which interprets a theta-marked constituent as outside of its initially hypothesised theta-domain and as within another existing theta-domain is costly.*

**-Theta-domain definition** =  $\alpha$  is in the theta-domain of  $\beta$  iff  $\alpha$  receives the  $\gamma$ -role from  $\beta$  or  $\alpha$  is dominated by a constituent that receives the  $\gamma$ -role from  $\beta$ .

Reanalysis of a theta-role is thus only costly if it a word is reanalysed and its theta-role is not the same to which it previously belonged. The effect is illustrated in the following examples in which reanalysis of the theta-roles is needed.

32. *We know her contributions failed to come in.*

Step 1:

*We know [her contributions]*  
 " t u "

<<agent>theme>>

We
----

Step 2:

*We know [her contributions failed to come in].*  
 " t u " " t "

<<agent>theme> <agent>

figure 30. Processing procedure of a theta-role reanalysis.

The processor first encounters the pronoun *we* which is stored in working memory until the verb *know* is encountered. *Know* has two theta-roles to assign and it assigns the agent theta-role to *we* and the theme theta-role to *her*. After *her*, the word *contributions* enters the derivation and *her* is reanalysed *Contributions* is added to *her* to form one DP since there is no other theta-role to assign. The verb *fail* has an agent theta-role to assign which poses a problem for the derivation. *Her contributions* was originally assigned the theme theta-role of *know* and is now assigned the agent theta-role of *fail*. However, reanalysis of the DP *her contributions* is not costly because it stays in its previously hypothesised theta-domain, which is the theme theta-role of *know*, which is now assigned to the entire thematic clause to the

right. For this reason, reanalysis is not costly and this sentence is not a real garden path sentence.

The difference between the previous sentence and a genuine garden path sentence is illustrated in the following example, which has approximately the same structure as the previous example.

33. *I warned her mother hated me.*

Step 1:

*I warned her mother*

" t y "

<<agent>theme>

I
---

Step 2:

<agent<theme>>  
? y t ?

*I warned her [mother hated me].*

" t y " o "

<<agent>theme, prop>

figure 31. costly reanalysis of a garden path sentence

In this example, *I* is first stored in working memory until the verb *warned* is encountered. The verb *warned* has three theta-roles to assign. In English, The agent is always the referring expression which is immediately to the right of the verb. The parser can remove *I* from working memory and attach it in this position. After this *her* is assigned the theme theta-role and *mother* is attached to the theme because it would otherwise have to be stored in working memory until another assigner is encountered. The parser immediately connects it to the pronoun to form one DP, because this is less costly than storing the NP in working memory. When the parser encounters the verb *hate* it knows that it has made a mistake and

that *mother* should not be part of the theme but agent of the second verb. Reanalysis needs to occur and *mother* is taken out of its original theta-role and put into the grid of *hate*. *Mother* is assigned the agent theta-role of *hate*. After this *hate* assigns its theme theta-role to *me* and the entire clause *mother hated me* is assigned the propositional theta-role of *warn*, which completes the sentence. However, the procedure was costly due to the reanalysis of the existing theta theta-role of *mother* from the theta-domain of *without* into the theta-domain of *hate*. According to Pritchett, the costly reanalysis of theta-roles is the garden path structure. The previous example was a combination of an DP with a CP which was mistaken to be one DP. The same structure combined with a PP can make matters even more complex as is illustrated in (34).

34. *Without her [donations to the charity] failed to appear.*

Step 1:

*Without [her donations to the charity]*

U    "  
 <theme>

Step 2:

*Without her [donations to the charity] failed to appear.*

U    "                    U    "    "    t  
 <theme>                    <theme><agent>

figure 32. reanalysis of a DP combined with a PP.

*Without* is an assigner of theta-roles. It assigns its thematic theta-role to the DP right adjacent to it, namely *her*. After this, *her* is reanalysed to be the possessive pronoun of *donations* because there are no other theta-roles to assign yet. This entire DP is assigned the theme theta-role of *without*. The processor does this because it prefers assignment of roles to storing elements in working memory. The latter option is more costly. The preposition *to*

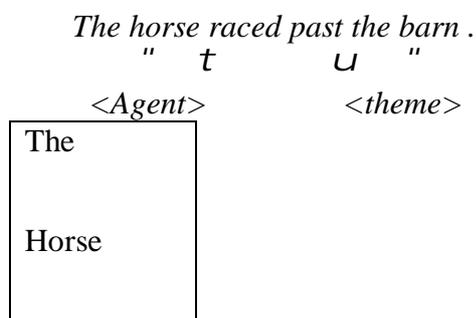
assigns its theme theta-role to the DP, *the charity*. The processor only realises that something went wrong when it reaches the second verb *fail*. This verb has also got an agent theta-role to assign and it can only assign this to a referring expression directly to the left of the verb, which in this case can only be *donations to the charity*. *Her* cannot be part of this NP because the theme theta-role should be assigned to *her*. *Donations to the charity* needs to be reanalysed. This makes the sentence complete, but reanalysis is costly because the NP *donations* is reanalysed to be outside its theta-domain.

There are also other types of structures with which the processor encounters problems like the sentence in example (35). This sentence is extremely difficult to process for the parser, which encounters some grave problems due to reanalysis of the theta-roles..

35. *The horse raced past the barn fell*

Figure 32.

Step 1:



Step 2:

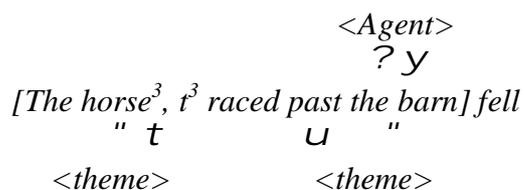


figure 33. Processing procedure of a garden-path.

*The* and *horse* are both first placed in working memory because there are not yet theta-roles which can be assigned to the DP. After the theta-grid from *raced* is retrieved from the lexicon, *the horse* receives an agent role. Note that *race* can also assign a theme theta-role but does not necessarily have to. If the verb assigned a theme theta-role to *the horse*, a reduced relative clause would be constructed. However, after presenting the parser with these two choices, the main clause analysis is chosen. After the agent theta-role of *raced* is assigned the processor encounters the word *past*, which is a preposition that assigns a theme theta-role to the DP *the barn*. Up to this point, the processor has not encountered any problems, but then it encounters the verb *fell*, which has an agent theta-role to assign. This is a problem because the main clause analysis which was built according to the theta-grid of *raced*, does not allow for another agent theta-role to be assigned. The parser does not have a referring expression available to assign anything to. The first part of the sentence thus needs to be totally reanalysed in order to form a relative clause. Then *the horse* is put into working memory and stays there, while the empty category which contains the null wh-element which marks the beginning of the relative clause, is assigned the theme theta-role of *raced*. The propositional theta-role is assigned by *past*. Consequently, *the horse*, which remains in working memory until the processor discovers the verb *fell*, is assigned the agent theta-role of *fell* because it is no longer the entity that races but the one that is being raced. A Garden path sentence occurs in this example, because reanalysis of the relative clause is expensive.

The previous example also illustrated that theta-role assignment is always consistent with a main clause analysis if there is no counterevidence. There are also numerous examples of sentences that appear to have the same structure as the one in example (35), but are not garden path sentences, which is illustrated in the following example.

36. *The bird found in the store died*

Step 1:  
*The bird found*  
 " t  
 <<theme>>

<i>The</i> <i>Bird</i>
---------------------------

Step 2:

<agent>  
 ?u

[*The bird found in the store*] *died*  
 " t u "  
 <theme> <theme>

figure 34. Processing procedure of example 36.

In this sentence, the first DP *the bird* is stored in working memory for a short while until *found* is encountered by the parser. The difference between this verb and *raced*, is that *found* is transitive and therefore only selects a patient. Consequently, the patient theta-role is assigned to the DP *the bird*, which is directly on the left of the verb. Transitivity is a linguistic constraint that describes a syntactic property that many verbs have; in this case, it causes main clauses like \**John found in the store*, to be ruled out. The moment the PP *in the store* is selected, the processor knows that the sentence needs to be reanalysed. The sentence is reanalysed as a reduced relative, and the agent theta-role is assigned to the empty category, the null wh-element which introduces the relative clause. The DP *the bird* is put back in working memory, because it cannot get another theta-role until the verb *died* is encountered and its theta-grid recovered from the lexicon. Since *the bird* is not immediately put into another existing theta-grid, reanalysis is not costly and therefore a garden path sentence does not occur.

The parser can also misinterpret a complement clause which is in fact a relative clause. This also causes garden path sentences to occur and is also caused by wrongfully analysing theta-grids. This misinterpretation is illustrated in (37).

37. *The patient persuaded the doctor that he was having trouble with to leave.*

Step 1:

*The patient persuaded the doctor [that he was having trouble]*  
 " t U " "  
 <<agent > patient ,prop>

<i>The</i>
<i>Patient</i>
<i>He</i>

Step 2:

*The patient persuaded [the doctor<sup>3</sup> that he was having trouble with t<sup>3</sup>] to leave.*  
 " r i " " r U "  
 <Agent < theme >> <<exp> <<theme>>  
 t ?  
 theme>

figure 35. costly reanalysis of several theta-roles.

In this sentence *the patient* is first stored in working memory until the theta-grid of the verb *persuaded* is retrieved from the lexicon. *The patient* is assigned the agent theta-role of this verb and then *the doctor* is assigned the theme theta-role. After that, the parser encounters the word *that* which it takes as an indication that an embedded clause has entered the derivation. It assigns the embedded clause the propositional theta-role of *persuade*. The pronoun *he* is stored in working memory until the theta-grid of the lexeme *have trouble with* is recollected from the lexicon. *Have trouble with* assigns an experiencer theta-role to *he* and has to leave its thematic theta-role open because it has nothing to assign it to. *To* is connected

to the verb *leave* which has an agent theta-role to assign. This is when the parser knows there is something wrong with the structure it created and it needs to dismantle it back to the first verb *persuaded*. The parser has by then realised the embedded clause starting with *that* is not a theme, but a relative clause. The entire clause thus needs to be reanalysed and this is costly because it needs to be moved from the theme theta-role to the patient theta-role of *persuaded*. In the relative clause, the experiencer theta-role of *having trouble with* is still assigned to *he* and its theme theta-role is now assigned to the trace of *the doctor* which is situated at the end of the relative. This leaves all the theta-roles covered until this point of the derivation. *To leave* assigns its agent theta-role to *the doctor that he was having trouble with* and the derivation is complete. However, reanalysis of the relative clause out of the propositional theme-role into the theme-role of the theta grid is very expensive because the relative clause is taken out of its theta-domain and put into another theta-domain. Therefore the sentence produces severe processing difficulties.

Garden path sentences are typically sentences that can be grammatical but in which the parser gives up too soon. Therefore the sentences are rendered ungrammatical too early in the derivation, but can be perfectly grammatical after reanalysis. The processor can thus overcome the problems of the garden path sentences. There are however, also sentences of which the problems cannot be overcome. An example is given in (35)

38. \* *John persuaded the man Sue is giving the money to leave.*

In this sentence, the parser analyses the sentence and the sentence works well until the word *to* is encountered, the parser first assumes that *to* is a preposition which belongs to the verb *give* and is connected with an object trace referring to *the man*. the verb *persuaded* on the contrary, can select an infinitive clause in which *to* is the representation of T. In this sentence,

the latter option is the correct one. However, in this sentence, the parser has to reanalyse *to* and in doing so, it needs to review the lexicon for the other form of *to*. It is the review of the lexical entry that does not allow the parser to go back to fix the sentence. Apparently, lexical access is extremely costly. This is stipulated in the following constraint.

39. ***Lexical reanalysis constraint:*** *The parser cannot fix a garden path sentence by replacing a word that has already been retrieved from the lexicon with a new homophonous lexeme with a grammatical function.*

As has been seen in the analysis of all of the sentences above, processing difficulties can be explained by a theta-driven parser. The processing theory put forward by Pritchett is able to deal with all structures, even the complicated Garden path sentences. Pritchett argues that the parser functions in direct agreement with the competence grammar and follows the theta-attachment principle. He furthermore proposes that it is costly for the parser to reanalyse structures which need to be placed outside their original theta-domain and into a new theta-domain. Pritchett's theory is therefore adopted for the remainder of this thesis in which the that-trace effects will be outlined from a processing perspective.

## 2.2. A processing account of English *That*-trace effects.

Before presenting the account of that-trace effects, which will be outlined in this chapter, it is important to explain the processes the parser goes through when it has to deal with interrogative sentences and embedded sentences. English interrogative clauses can start with an auxiliary or with a wh-phrase. In order to parse a sentence with an initial wh-element, the parser needs to store that wh-element in working memory, not only until it first encounters the verb and its theta-grid is retrieved, but until all other theta-roles are assigned. This is due to the fact that a wh-phrase is always an element that has been moved. For this reason, the parser needs to examine all grids in order to determine where the wh-phrase came from. In more complex sentences, the parser often needs to wait until the end of the sentence as a whole before it can assign a theta-role to the wh-element. It needs to know whether the wh-phrase belongs to a theta-grid of the main clause or perhaps one of an embedded clause. The derivation of interrogative clauses is illustrated below.

40. *Who*<sup>7</sup> *did Mary kiss t*<sup>7</sup>?

*Who*<sup>7</sup> *did Mary kiss t*<sup>7</sup> ?  
 " t U "  
 << Agent>theme>

<i>Who</i>
<i>Mary</i>

figure 37. Processing procedure of an interrogative sentence.

In this question, the first phrase that is encountered by the parser is the wh-expression, *who*, which is stored in working memory. After retrieving the wh-phrase the parser knows that the sentence is an interrogative sentence and that it cannot immediately insert the wh-



agent-role after encountering the verb *see*. Considering that this is the end of the sentence, the processor knows that it has to assign another theta-role to *what*, which is still in working memory. This completes the derivation because every theta-role is assigned and the sentence is completed.

Next, subject gap sentences are considered. An example of a subject gap sentence is given in the following example.

42. *Who<sup>3</sup> t<sup>3</sup> kissed John?*

*Who<sup>3</sup> t<sup>3</sup> kissed John?*  
 " t U "  
 << Agent> theme >

Who

*figure 38. processing procedure of an interrogative sentence with subject movement.*

In this case when *who* is encountered, it is stored in working memory because it is a wh-word. Following this, the word *kissed* is encountered by the processor and its theta-grid is recovered from the lexicon. The parser recognises that it has not yet got a phrase to assign the agent theta-role of *kissed* to and because of this, it proceeds with the next word which is *John*. *John* is not on the left side of the verb, which means that it cannot be assigned the agent role. The thematic theta-role is thus assigned to *John*, leaving the agent theta-role open without a phrase to fill it. The parser thus assigns the agent theta-role to the wh-word. Since the wh-phrase is now the agent of the sentence, the parser knows that an auxiliary T need not be present and the derivation is completed.

In question sentences that contain sentence initial wh-expressions, the parser cannot assign theta-roles to wh-expressions before it has determined where the wh-expression has moved from. Until then, the wh-phrases are stored in working memory.

The second structure that is important to examine, before dealing with that-trace effects is the way the parser handles an embedded clause. In English, embedded clauses can start either with an overt or with a phonetically null complementizer. However, the parser does not need the presence of overt *that* to know that an embedded clause is about to make an appearance because it can also use the theta-structure of the matrix verb which selects a complement clause. Specifically, it is assumed that the theta-grid of a verb like *think* induces a "propositional theta-role, which tells the parser that it must construct an embedded CP. Consider the sentence in the following example and the representation of its parsing in figure 39.

43. *My mother thinks that John kissed Mary.*

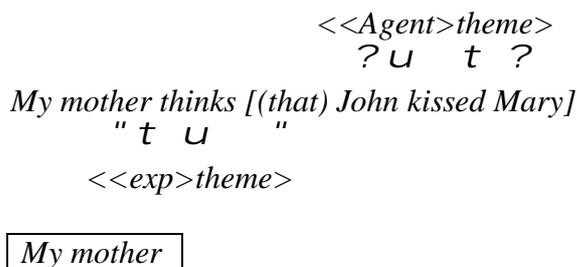


figure 43. processing procedure of example 43.

The parser first encounters the DP *my mother* and stores it in working memory because there are not yet theta-roles available for assignment. When the parser encounters the verb *think*, its theta-grid is retrieved from the lexicon and the experiencer theta-role is assigned to *my mother*. After *that*, *John* is encountered which is put in the propositional theta-

role of think. After the processor encounters the second verb *kissed*, *John* is reanalysed and it is assigned the agent of *kissed*. However, because John is still part of the theme complement clause of *think*, this is not costly. Next, the parser analyses the rest of the embedded clause which consists of the DP *Mary*. This part of the embedded clause which is the theme of *think*, is also assigned the theme theta-role of the verb *kissed* and this completes the sentence. The sentence can also be parsed with *that* as the head of the CP. These sentences can thus be correctly analysed by the parser without costly reanalyses.

In sentences of the type in which a *That*-trace effect can occur (henceforth, *That*-trace sentences), the processor encounter wh-movement from an embedded clauses. As is outlined above, the parser cannot assign a wh-phrase a theta theta-role until it has examined all available theta-grids in the sentence as a whole. In particular, in the case of a *That*-trace sentence, the parser stores a wh-expression in working memory until it has access to the theta-grid of the embedded clause. This is also the case of sentences in which a wh-word is extracted from an object position as is illustrated in the following example. The parser proceeds as is illustrated in figure 40.

44. *What<sup>t</sup> did Mary think (that) John gave t<sup>t</sup> to Mary*

<<agent> theme, propt>  
 ? y f ? ?  
*What<sup>t</sup> did Mary think [that John gave t<sup>t</sup> to Mary]*  
 " t U "  
 <<agent>theme>

<i>What</i> <i>Mary</i>
----------------------------

figure 40. processing procedure of example 41.

The parser first encounters the wh-word *what* and stores it in working memory. *Mary* is also put in working memory. Next, the theta-grid of *think* is recovered from the lexicon and

the experiencer theta-role is assigned to the DP *Mary*, which is the referring element directly to the left of the verb. Next, the derivation can choose whether to insert the overt complementizer *that* or to insert the null complementizer. The processor encounters *that* it will immediately assign it the *theme theta-role* of *think*. If it does not, *John* is the next element to enter the derivation and it will assign its theme theta-role to *John*. However, in both these structures *John will be assigned* the theme theta-role and will have to be reanalysed after the verb *give* is encountered. *John* is assigned the agent theta-role of *give* and *to Mary* is assigned the propositional theta-role of *give*. The parser has then examined all available theta-grids in the sentence and can assign the theme theta-role of *give* to *what*.

When an embedded clause has a topicalised phrase, it also has an optional complementizer In English. The processor still recognises the embedded sentence and knows that it can start with an overt or a null complementizer. The sentence is then parsed in the same manner as a normal embedded sentence as is illustrated in the example below.

45. *Who<sup>7</sup> did John say (that) secretly Mary kissed t<sup>7</sup>?*

*Who<sup>7</sup> did John say [secretly Mary kissed t<sup>7</sup>]?  
 " t u " " t u "  
 <<agent>prop> <<agent>theme>*

<i>Who</i>
<i>John</i>

*figure 41. processing procedure of example 45.*

*Who* is stored in working memory and so is *John*. Next, the verb *say* is encountered and the parser assigns the agent theta-role to the DP *John*. *Say* also selects a propositional theta-role, which it assigns to *secretly*, creating an embedding clause. The parser always prefers assigning theta-roles to phrases over storing phrases in working memory if possible.

Next, the parser encounters the word *Mary*, which is part of the embedded clause and the verb *kissed* is encountered. The parser realises that it should reanalyse *Mary* because it needs to be the agent of *kiss*. However, the whole phrase remains in the theta-domain of *say* and reanalysing the content of the embedded theme clause is not costly. The parser has now examined all the available theta-grids in the sentence and knows that it must assign the theme theta-role to *who*. The sentence is then complete.

Now let us consider a *That*-trace sentence, in which a complementizer cannot be overt because there is a trace of movement in subject position of the embedded clause as is illustrated in the following examples.

46.

- a. Who<sup>7</sup> does John think t<sup>7</sup> kissed Mary
- b. \*Who<sup>7</sup> does John think that t<sup>7</sup> kissed Mary

<<agent>theme>  
 ?u t ?  
 Who<sup>7</sup> does John think [t<sup>7</sup> kissed Mary]  
 " t u "  
 <agent<theme>>

Who John
-------------

Step 1:

\*Who<sup>7</sup> does John think [ that...  
 " t u "  
 <<agent>prop>

Step 2:

<<agent>theme>  
 ?u t ?  
 \*Who<sup>7</sup> does John think [ that kissed Mary]  
 " t u "  
 <<agent> prop >

Step 3:

<<agent>theme>  
 ?u t ?  
 \*Who<sup>7</sup> does John think [ that t<sup>7</sup> kissed Mary]  
 " t u "  
 <<agent< prop >

Who John
-------------

figure 43. processing procedure of the That-trace effect.

First, the sentence in example (46b) will be explained to show where the parser fails and finds this example ungrammatical. First, the parser needs to store the wh-word *who* in working memory. It then encounters the DP *John* and stores it in working memory as well. When the verb *think* is encountered the parser retrieves its theta-grid from the lexicon and takes *John* from working memory to assign the agent theta-role of *think* to *John*. The parser then encounters *that* which it assigns the propositional theta-role of *think*. Next, the verb *kiss* comes in the derivation and the parser retrieves the theta-grid from the lexicon, which

contains of an agent and a theme. Since the agent theta-role is assigned to a DP or a clause immediately to the left of the verb, the parser has to reanalyse *that* and it assigns the agent theta-role to *that*, this time analysing it as a demonstrative pronoun. This operation is not costly, because *that* remains in the propositional theta-role domain of *think*. Next, it assigns the patient theta-role to the DP *Mary*. The sentence then appears to be complete and this is where the derivation encounters its biggest problem since there is still a wh-word in working memory, waiting to be put into the derivation while there is no empty position in any theta-grid to fill. The parser then realises that the demonstrative pronoun *that* is in fact an overt complementizer. The entire embedded sentence needs to be reanalysed. *That* is taken out of the theta-domain of *kiss* since it is only a part of the theta-domain of *think*. This operation is thus costly and therefore the sentence is ungrammatical. Apart from this, the parser needs to retrieve a homophonous lexeme for *that* from the lexeme and the parser cannot recover from this causing the sentence to be ungrammatical.

In example (46a), the embedded sentence starts with a null complementizer and the situation is more straightforward. *Who* is also put in working memory and so is *John*. After the theta-grid of *think* is recovered from the lexicon the agent theta-role is assigned to the element directly to the left of the verb, which is *John*. Immediately after the verb and its grid are recovered, the parser encounters a new verb, *kiss*. It assigns the propositional theta role to the verb because it knows the verb is part of an embedded clause. The new verb *kiss* cannot assign its agent theta-role immediately but it assigns its theme theta-role to the DP *Mary*. This completes the derivation and the parser has now examined all theta-grids and knows what theta-role is must assign to *who*, which is still stored in working memory.

The ungrammaticality of the first sentence is thus due to the fact that the overt complementizer is wrongly analysed as a demonstrative pronoun which is the agent in the embedded sentence while it is the complementizer of the embedded clause.

On the other hand, the structure is not problematic when a subject wh-phrase is extracted from an embedded clause which starts with a topicalised phrase. This structure is illustrated in example 47.

47.

a. *Who<sup>7</sup> does Mary think that secretly t<sup>7</sup> kissed John*

b. *Who<sup>7</sup> does Mary think secretly t<sup>7</sup> kissed John*

<<agent>theme>  
 ?u t ?  
*Who<sup>7</sup> does Mary think [that secretly t<sup>7</sup> kissed John].*  
 " t u "  
 <<agent>prop >

<i>Who</i> <i>Mary</i>
---------------------------

figure 43. Processing procedure of example 47a..

<<agent>theme>  
 ?u t ?  
*Who<sup>7</sup> does Mary think [secretly t<sup>7</sup> kissed John].*  
 " t u "  
 <<agent>prop>

<i>Who</i> <i>Mary</i>
---------------------------

figure 44. Processing procedure of example 47b.

In both sentences above, *who* and *Mary* are first stored in the lexicon because there are not yet theta-roles to assign. The parser assigns the agent theta-role of *think* to *Mary*. In both sentence the propositional theta-role is assigned to the next element. This is either the overt complementizer in sentence (a), or the adverbial phrase in sentence (b). In sentence (a), the

adverbial phrase is later added to the theme theta-role. Immediately after this a new theta-grid is retrieved from the lexicon because the verb *kiss* enters the derivation. *Kiss* has an agent and a theme theta-role to assign. English verbs usually assign their agent roles to DP's or clauses which are situated directly to the left of the verb. Since *secretly* cannot be the agent of the verb the parser proceeds with the next DP *Mary*, which it assigns the theme theta-role of *kiss*. This completes the sentence and all available theta-grids are now examined. The parser assigns the last available theta-role to the wh-word *who* which is assigned the agent theta-role of *kiss*.

This far, all the structures which are depicted in this chapter can be explained by parsing theory. Since relative clauses are already discussed in the previous chapter, there is only one more structure which Pesetsky and Torrego dealt with in their proposal that needs to be discussed, which are sentential subjects. These appear to be a problem for the theory, because the parser of an English native speaker knows that matrix sentences cannot start with an overt complementizer. The structure is illustrated in the following example.

48. *[That Sue will buy the book] was expected by everyone.*

*[That Sue will buy the book]<sup>t</sup> was expected t<sup>t</sup> by everyone.*  
 " t u " u " u "  
 <<agent>theme> <theme> <theme>

<i>That</i> <i>Sue</i> <i>Will</i>
--

figure 45. Processing procedure of sentential subject.

The parser needs to store *that*, *Sue* and *will* in working memory first. *That* can be a demonstrative pronoun or a complementizer. For this reason, the parser cannot place it into the derivation yet, because there are no theta-roles to assign yet, so the parser cannot decide which homophonous lexeme of *that* is the correct one either. Next, the verb *buy* is encountered,

and the agent theta-role is assigned to the DP *Sue*. The theme theta-role is assigned to the next phrase in line which is *the book*. After this, the verb phrase *was expected* is put into the derivation which has its own theta-grid. It has a theme and an agent to assign. Usually, the agent would be assigned to the direct left of the verb, but this is a passive sentence which does not necessarily need an agent. As a consequence the parser does not assign the agent theta-role yet but proceeds with the derivation. The parser reaches the word *by* which has its own theta-role to assign but also indicates that an agent of the verb is mentioned. *By* assigns a theme theta-role to *everyone* and the parser realises that the theta-grid of *expect* has not been completed. It realises that the entire clause *Sue will buy the book* is the theme of *expected* and that the clause has in fact moved from inside the VP to the [SPEC, TP] position. It assigns the theme theta-role to the trace of this clause because the clause is an embedded clause which is moved. *That* is taken out of working memory and inserted at the front as an overt complementizer. This completes the sentence.

To sum up, this processing theory is able to deal with all the types of sentences which are put forward in the section about Pesetsky and Torrego's theory. *That*-trace effects are much like garden path sentences, because they are caused by misinterpretation of the overt complementizer. However, the classical *that*-trace effect reanalysis of *that* is ungrammatical because in order to reanalyse *that*, the parser needs to access the lexicon twice since it originally evaluates *that* as a demonstrative pronoun, while it is in fact a grammatical lexeme with a grammatical function.

### 2.3. Cross-Linguistic differences

According to Pritchett, the parser and the competence grammar are closely related. Grammars are language specific and the parser is universal. The parser thus always processes sentences in the same manner, while applying to the rules of the competence grammar. Language specific grammar rules, dictate the choices of word order and structure the parser needs to make. This is what causes cross-linguistic differences and in this specific case, allows for variations in *That*-trace effects and even Anti *That*-trace effects as they do occur in Dutch. In previous chapters, the Dutch variation of the English phenomenon was illustrated. Dutch embedded clauses always begin with an overt complementizer, even if a subject wh-trace is extracted from the embedded clause and moved into the higher clause. Apart from Dutch, there are other languages in which an overt complementizer needs to introduce the embedded sentence. French and German are among these languages. Dutch structures in which a subject trace is extracted from an embedded clause is illustrated in the following example.

49.

a. *Wie<sup>7</sup> dacht Jan dat t<sup>7</sup> Mary gekust heeft.*

Who<sup>7</sup> thought John t<sup>7</sup> Mary kissed had

$$\begin{array}{ccc}
 & < \textit{theme} > & < \textit{prop} > \\
 e & ? & ? u \\
 \textit{Wie}^7 \textit{dacht Jan} & [\textit{dat t}^7 \textit{Mary gekust heeft}. \\
 u & " & " e \\
 << \textit{agent} >> & << \textit{agent} >>
 \end{array}$$

Wie

*figure 47. Processing procedure of Dutch sentence with subject wh-movement.*

*Wie* is first put in working memory. When the verb is reached the parser knows it has to assign an agent and a propositional role. This is when the main difference between the Dutch grammar and the English grammar becomes important to the parser, because Dutch allows for scrambling and in Dutch the subject can be a complement of the verb in interrogative clauses. In Dutch, the parser cannot always assign the agent theta-role to the referring element directly to the left of the verb, thus the parser needs to judge each DP or clause more precisely and even put more phrases in working memory if necessary. In this sentence, the DP *Jan* is reached after the verb is encountered and therefore the parser can assign the agent theta-role of *dacht* to *Jan* immediately.

Next, the parser encounters the overt complementizer *dat* to which it assigns the propositional theta-role of *dacht*. *Dat* is the complementizer of the embedded clause. *Mary* is also added to the clause. The parser then encounters the verb *gekust*. *Gekust* has an agent and a theme theta-role to assign and the parser can now see where the wh-word needs to be put in the derivation because it has examined all the available theta-grids in the sentence as a whole. It reanalyses the DP *Mary*, which is not costly because *Mary* stays inside the propositional theta-role domain of *dacht*. The subject trace of *who* is inserted in the derivation and the parser assigns the agent theta-role of *gekust* to the wh-word which was still in working memory. The auxiliary *heeft* is then added which completes the derivation.

In the following example, an object trace is extracted from the embedded sentence. The representation of the parse of this structure is given in figure 48.

50. *Wie*<sup>3</sup> *dacht* *Jan* *dat* *Mary* *t*<sup>3</sup> *gekust* *heeft*.

Who<sup>3</sup> did John think that Mary kissed t<sup>3</sup>

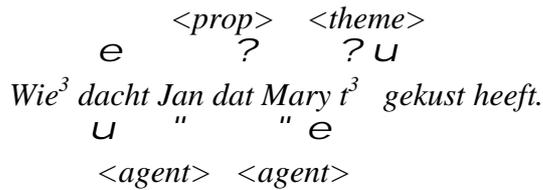


figure 47. Processing procedure of Dutch sentence with object wh-movement.

*Who* is stored in working memory and the theta-grid of *dacht* is retrieved from the lexicon. *Dacht* assigns its agent theta-role to *Jan* and its propositional theta-role to the complementizer *dat*. *Mary* is added to the complementizer clause and so is the verb *gekust* which has a theta-grid containing an agent and a theme. The parser assigns the agent theta-role to *Mary* which leaves one theta-position open. The parser then assigns the theme theta-role to the trace of the wh-phrase *who*, which is then recollected from working memory because all available theta-grids in the sentence as a whole are examined.

If Dutch were to allow for embedded sentences to begin with a phonetically null complementizer, this would be extremely complicated for the parser because Dutch allows for scrambling of objects in embedded sentences. In matrix clauses, the Dutch word order is Subject-Verb-Object while it often is Subject-Object-Verb in embedded clauses. This makes it more difficult for the parser to distinguish to which words the theta-roles need to be assigned. In order to distinguish an embedded sentence clearly, the border of the embedded sentence is clearly indicated by the overt complementizer. The problems the parser can encounter while parsing an embedded clause which does not have an overt complementizer are illustrated in figure 48.

51.  $Wie^3$   $zei$   $Mary$   $t^3$   $John$   $gekust$   $had$ ?  
*Who<sup>3</sup> did Mary say  $t^3$  kissed John?*

<<patient>>  
 ?u

Wie<sup>3</sup>    zei    [    Mary John gekust had]?  
 " r    u    "    "    t

<<agent>theme> <Agent>

Wie

figure 48. Processing procedure of ungrammatical Dutch embedded clause.

First, *Wie* is put in working memory, giving notice to the parser that it has to do with an interrogative clause. *Zei* has an agent to assign, which it can assign to the element to its right, *Mary*. John is then stored in working memory because there is only a theme theta-role to assign, which needs to begin with an overt complementizer. After the verb *gekust* is encountered by the parser, the sentence becomes more problematic because the parser will now try to reanalyse the sentence in order to get all the theta-roles filled. It has to reanalyze John, which is costly because John is taken out of its theta-domain. After John is taken out of its theta-domain, the parser can either reanalyse *Mary* and put *wie* in subject position of the main clause, forcing a complete main-clause reading on the sentence, or it can put *wie* in the subject position of the embedded clause, leaving the parser with an undefined embedded clause which does not begin with an overt complementizer and thus a theme-role which cannot be assigned. In native speaker's judgement, this sentence feels as a complete main clause, because it lacks a complete embedded clause. Thus, the sentence is ungrammatical in accordance with the Dutch competence grammar because the parser cannot assign all possible theta-roles to the existing words. The difference with the English sentence is thus in the fact that for English competence grammar, the position of the agent and the patient is already clear to the parser, which allows for a correct reading in normal embedded sentences due to the fact that the parser can assign theta-roles more readily after encountering the phrases but is also due to the fact that English competence grammar allows for complement clauses to start

without an overt complementizer, which allows for theta-role assignment to complement clauses that start with a null complementizer.

More evidence for this, comes from German which also allows scrambling of objects in embedded clauses and always needs a complementizer in embedded clauses. German does not seem to have a *That*-trace effect either as is illustrated in the following examples.

52.

a. *Wen<sup>3</sup> meint John dass Mary t<sup>3</sup> liebt.*

*\*Wen<sup>3</sup> meint John Mary t<sup>3</sup> liebt.*

*Who<sup>3</sup> does John think Mary loves t<sup>3</sup>*

b. *Wer<sup>3</sup> meint John dass t<sup>3</sup> Mary liebt.*

*\*Wer<sup>3</sup> meint John t<sup>3</sup> Mary liebt.*

*Who<sup>3</sup> does John think t<sup>3</sup> loves Mary.*

German however, does allow null-complementizers in embedded sentences that have verb second and thus the same word order as English embedded sentences have. This is in fact, evidence to the claim that overt complementizers are needed when the parser has not got a clear structure from which to deduct theta-roles which it needs to process a grammatically correct sentence. In German, the parser obviously needs to have an overt complementizer to introduce an embedded sentence when it cannot rely on its basic word order, but it does not when it can rely on the original SVO word order of sentences which is illustrated in the following sentences.

53.

a. *John liebt Mary*

John loves Mary

b. *Wer<sup>3</sup> meint John t<sup>3</sup> liebt Mary.*

Who<sup>3</sup> thinks John t<sup>3</sup> loves Mary

c. *Wen<sup>3</sup> meint John Mary liebt t<sup>3</sup>.*

Who<sup>3</sup> thinks John Mary loves t<sup>3</sup>.

Dutch matrix sentences, also follow the SVO word order and Chomsky's minimalism is based on the idea that all sentences originally have the same word order and that varieties are derived from this SVO word order. The parser thus assigns its theta-roles relying on this structure and needs help when the structures are different than this. Dutch and some German embedded sentence allow for scrambling, which makes it difficult for the parser to assign the theta-roles. Therefore, overt complementizers are obligatory in Dutch.

The final argument in favor of a processing account is the fact that pro-drop languages have an anti That-trace effect as well. Complementizers are obligatory, because subjects can be omitted. This is also confusing for the parser, which then lacks a subject to assign the agent theta-role to, and therefore needs a complementizer to indicate the left border of the embedded clause as is illustrated in the following example.

54. *Chi<sup>3</sup> credi che t<sup>3</sup> verra a visitarci*

*Who think that will visit.*

In these languages, the parser needs to be able to distinguish between the embedded and the higher clause with some help, because it cannot assign an agent theta-role to the immediate left of the verb. Since it has to defer from the original SVO structure an overt complementizer must be present to indicate the embedded sentence border.

### 3. Conclusion

In the past, *That*-trace effects were always described to be a syntactic issue. Although previous theories were able to deal with the English classic *That*-trace effects, they could not explain the varieties of effect which occur in different languages. When the Minimalist programme was first introduced, the notion of government no longer existed. Previous theories could no longer be used in the new syntactic framework.

Pesetsky and Torrego developed a new theory which was based on minimalism. They argued that *that* is not an overt complementizer, but is T moved to C in embedded clauses. They further argued that *that* is optional in English because uninterpretable tense on C can both be checked by T to C movement and via movement of the nominative subject to [SPEC, CP]. According to these authors, *that* is not allowed in the classical *That*-trace effect in English since both the uninterpretable wh-features and the uninterpretable tense feature on C can be erased by one operation of movement in the case of long-distant subject wh-movement and T-to-C movement was not allowed because a feature cannot be checked twice.

The theory as presented by Pesetsky and Torrego can deal with all the English structures, but is not able to deal with the Dutch examples which were given in this thesis. Apart from this, the theory is hardly backed up with theoretical evidence and their assumption that uninterpretable feature can live on to check other features is problematic for Minimalism, the structural theory on which the proposal was based.

In this thesis, a processing account of the *That*-trace effect was put forward. It is based on the theta-driven processing theory which is proposed by Bradley Pritchett. He proposed that the parser creates sentences through theta-role assignment and that reanalysing phrases is costly if they are moved out of their original theta-domain if they need to be reanalysed. In

this thesis, a lexical reanalysis constraint is proposed, which states that words cannot be reanalyzed to be a different homophonous lexeme which has a grammatical function. This is the central issue in my analysis of the English *That*-trace effect. In this thesis, it is put forward that the parser wrongly analyses the overt complementizer as a demonstrative pronoun, when there is a trace of movement in subject position. The parser does this, because it has no other subject available directly at the left to the verb. Reanalysing this word by re-accessing the lexicon is very costly and the parser cannot overcome this. Therefore overt complementizers are not allowed in English when there is long-distant subject wh-movement.

Since scrambling is allowed in Dutch and many other languages, the parser does not have a clear reference as to where the boundaries of theta-attachment are. Therefore, Dutch needs an overt complementizer at the beginning of an embedded clause. This overt complementizer defines where the boundaries for theta-attachment are, and thus eases the work of the parser. Evidence from German provides evidence for this account and so does evidence from Italian, which is a pro-drop language.

The parser assigns theta-roles in a universal manner. Chomsky has already argued in his Minimalist Program, that languages are all built up from the same basic structure. Most matrix clauses adopt this basic structure. This basic structure is adopted by the universal parser to be a subject-verb-object word order. The languages that do not adopt this basic word order in their embedded clauses, need an overt complementizer in order for the parser to understand from which point in the sentence it has to assign new theta-roles. Therefore, the parser can choose whether to use an overt complementizer in English but needs to insert a complementizer in languages like Dutch and Italian. *That*-trace effects thus stem from processing problems instead of syntactic difficulties.

More research is needed in order to provide evidence for the cross-linguistic variance this theory can cope with. There are many more languages which can be taken into account because there seem to be many *That*-trace related effects throughout the world. Especially languages like Japanese and Chinese might be interesting in this perspective, because they have a complementizer which occurs at the end of the embedded clause.

#### 4. Works Cited List

- Bresnan, Joan. 1970. On Complementizers: Toward a Syntactic Theory of Complement Types. *Foundations of Language*. 6 297-221
- Chomsky, Noam . 1980. On Binding. *Linguistic Inquiry*, 11. 1-46.
- Chomsky, Noam. 1981. *Lectures on Government and Binding*. Floris.
- Chomsky, Noam . 1996. *The Minimalist Programme*. MIT Press.
- Chomsky, Noam. 2004. Beyond explanatory adequacy. In *The cartography of syntactic structures*. Vol. 3, *Structures and beyond*, ed. by Adriana Belletti. Oxford: Oxford University Press.
- Haegeman, Liliane.1991. *Introduction to Government and Binding*.
- Pesetsky, David. 1982. Complementizer trace phenomena and the Nominative Island Condition. *The Linguistic Review*. 1,3: 297-343
- Pesetsky, David & Torrego, Esther. T-C Movement: Causes and consequences. *Ken Hale, a Life in Language*. Michael Kenstowicz (ed). Cambridge, Mass. MIT Press.
- Pritchett, Bradley L. 1988. Garden Path Phenomena and the Grammatical Basis of Language Processing. *Language*. 64,3: 539-576
- Rizzi, Luigi. 1990. *Relativised Minimality*. Cambridge, Mass. MIT Press.
- Szczegielniak, Adam. 1999. That-trace effects Cross linguistically and Successive Cyclic movement. *Papers on Morphology and Syntax, Cycle One MIT working papers in linguistics*. Vol 33. 369-393
- Sobin, Nick.1987. The Variable Status of Comp-Trace Phenomena. *NLLT*, 33-60.