TBTA Tutorial Lesson 13:
Structural Adjustment Rules

(video: TBTA-13A Structural Adjustment Rules - Multiple Concepts)
(video: TBTA-13B Structural Adjustment Rules - Copying Constituents)
(video: TBTA-13C Structural Adjustment Rules - Combining Propositions)

13.0 Introduction

This lesson discusses the final type of rule in TBTA's transfer grammar: Structural Adjustment rules. The previous tutorials have introduced you to TBTA's ontology, the semantic representations, the target lexicon, the various types of rules in the synthesizing grammar, the grammar tools, and the other types of rules in the transfer grammar. Now this tutorial will describe the structural adjustment rules. The structural adjustment rules use the same dialog as the constituent movement rules, the pronoun identification rules, and the theta grid adjustment rules, so you've already seen the dialog for structural adjustment rules. And during the tutorial for pronoun identification rules, you built several input structures and then made some simple modifications in the output structures. Even though you're familiar with these rules, this tutorial will introduce you to the primary capabilities of these rules.

13.1 Structural Adjustment Rules

The structural adjustment rules are capable of performing the following tasks:
- inserting new constituents into the semantic representations,
- deleting constituents from the semantic representations,
- copying constituents,
- moving constituents,
- setting features,
- copying features, and
- providing new translations of concepts.

Let's begin by writing a structural adjustment rule that sets a feature. In the tutorial that described the process of generating pronouns, we wrote a Feature Copying and Setting rule that set a feature called 'Adverbial Clause Position' to PrePosed for all adverbial clauses that begin with AFTER-A, BEFORE-A, and WHEN-A. You can delete that rule now by opening the node for Feature Copying and Setting Rules, then expanding the Clauses node, and then expanding the group node called Adverbial Clauses. Click on the rule's name and push the Delete key. You may also delete the group name by clicking on it and pushing the Delete key.

To build a new structural adjustment rule, double click on Structural Adjustment Rules in the grammar tree, and enter a new group called **Subordinate Clauses**. Then double click on that group name, and you'll be prompted to select a syntactic category. Every structural adjustment rule will be assigned to one syntactic category, and you should select the category that is most responsible for triggering that rule. For this rule, we'll select Adpositions because adverbial clauses always begin with an adposition.

After you've selected Adpositions from the dropdown, you'll see a new structural adjustment rule. Enter a name for this rule such as **Some Adverbial Clauses are PrePosed, others are PostPosed.** Now we need to insert prepositions such as AFTER-A, BEFORE-A, and WHEN-A into the input structure, so click on the button labeled 'Insert Word.' Then click in the input structure somewhere between the opening and closing clause boundaries. You'll see a dialog that prompts you to select either a source concept or a target word. The default is 'Source Concept' and that's what we want, so you don't need to change that. Select Adpositions in the dropdown at the top, and then select AFTER-A in the large dropdown at the bottom. Then click OK, and you'll see AFTER-A inserted into the input structure. To add other adpositions, click the button labeled 'Add Words,' and then click on the adposition in the input structure. You'll see a dialog that will let you select many adpositions. Click the radio button labeled 'Entire Ontology' at the top of the dialog, and the upper table will be loaded with all of the adpositions that are in the ontology. Scroll down and click on BEFORE-A and WHEN-A, and then click the Select button that's directly below the upper table. You'll then see that AFTER-A, BEFORE-A, and WHEN-A are in the lower table, meaning that they've been selected to participate in this rule. Close that dialog by clicking OK, and then rest the cursor on AFTER-A in the input structure. You'll see a popup showing that all three adpositions will participate in this rule even though only AFTER-A is displayed in the input structure.

   In order to indicate that adverbial clauses which begin with these adpositions should be preposed, click the radio button labeled 'Output Structure.' You'll then see the input structure copied to the output structure. Click the button labeled Features, and then click on the opening clause boundary in the output structure. You'll see a message saying that one of the words in the input structure has multiple concepts, and it asks if you want to set up a table so that you can specify a particular feature value for each concept. We don't want to do that, so click No, and then you'll see a dialog showing all of the features for clauses. Scroll to the right and you'll see the feature called 'Adverbial Clause Position.' Select PrePosed, and then click OK. If you rest the cursor on the clause's features in the output structure, you'll see a popup showing that 'Adverbial Clause Position' will be set to PrePosed by this rule. Assign this rule to the grammar topic called 'Adverbial Clauses.' Your rule should now look like the one shown in Figure 13-1.
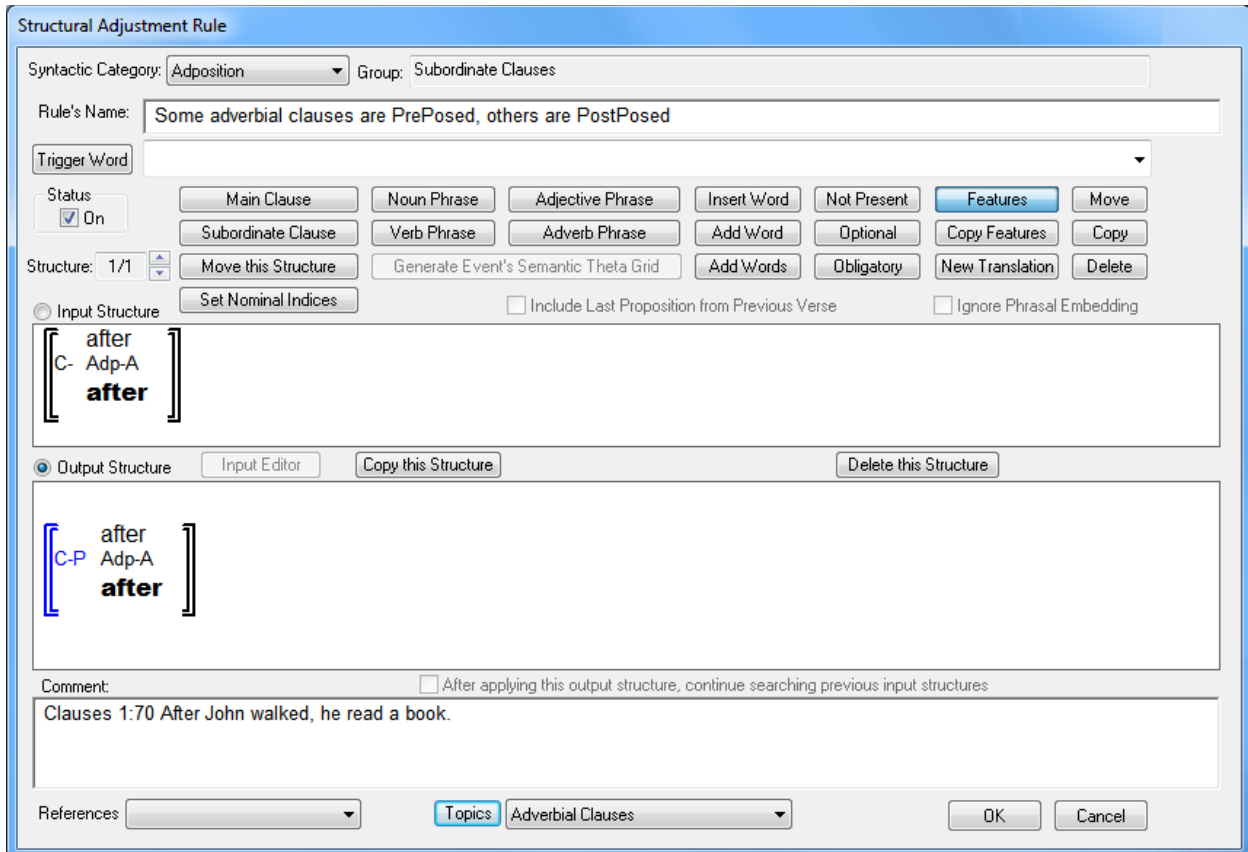
Figure 13-1. Structural Adjustment Rule that sets the Feature 'Adverbial Clause Position'

Save this rule, and you can generate Clauses 1:68, 69, and 70 again with the correct results.

Now let's add another structure to the structural adjustment rule in Figure 13-1. This second structure will be used to initialize the feature called 'Adverbial Clause Position' to PostPosed for all adverbial clauses. Then the first structure will change the value to PrePosed for adverbial clauses that begin with certain adpositions. Open the structural adjustment rule and click the up arrow in the structure spinner located immediately below the Status checkbox. You'll see that the input structure from structure 1 is copied to the input structure of structure 2, and the output structure of structure 2 is empty. Let's use this second structure to initialize our feature called 'Adverbial Clause Position' to the value PostPosed. We want this structure to set 'Adverbial Clause Position' to PostPosed for all adverbial clauses, so delete the adposition in the input structure by clicking the Delete button, and then clicking on the adposition in the input structure. Now let's specify that this clause must be an adverbial clause, so click the button labeled Features, and then click on the opening clause boundary in the input structure. You'll see a dialog with all of the features for clauses; in the Type column select 'Event Modifier' which is the semantic equivalent of 'adverbial clause,' and close the features dialog. Then click the radio button labeled 'Output Structure.' You'll see a warning message saying that there must be at least one adposition in the input structure because this rule is triggered by adpositions. We don't want to insert an adposition into this structure, so let's change the rule's syntactic category from adposition to clause. To

do this, open the dropdown in the upper left corner of the rule labeled 'Syntactic Category,' and select Clause. Now click the 'Output Structure' radio button again, and you'll see the input structure copied to the output structure. Click the Features button, and then click on the opening clause boundary in the output structure. Scroll the features to the right, and you'll see the column labeled 'Adverbial Clause Position.' Select PostPosed, and close the features dialog. This structure will now find every adverbial clause, and set its 'Adverbial Clause Position' to PostPosed. When structural adjustment rules are executed, TBTA begins with the last structure in the rule and checks if it matches the current situation in the semantic representation. If the structure doesn't match, then TBTA will move to the preceding structure and check again. This process will be repeated until either there are no more structures to check, or a structure matches. When a structure matches the semantic representation, the changes specified in the output structure will be made. Then TBTA will examine the checkbox at the bottom of that structure labeled 'After applying this output structure, continue searching previous input structures.' If that checkbox is not checked, then TBTA will go to the next rule in the grammar. However, if that checkbox is checked, then TBTA will resume the process of checking the previous structures to see if any of them match the current situation in the semantic representation. For our current rule, after the second structure has initialized the feature called 'Adverbial Clause Position' to PostPosed, we want the rule's first structure to be examined and executed if it matches the current adverbial clause in the semantic representation. So check the box that tells TBTA to check the previous structures after applying this structure. Your rule with this second structure should now look like the one shown in Figure 13-2.
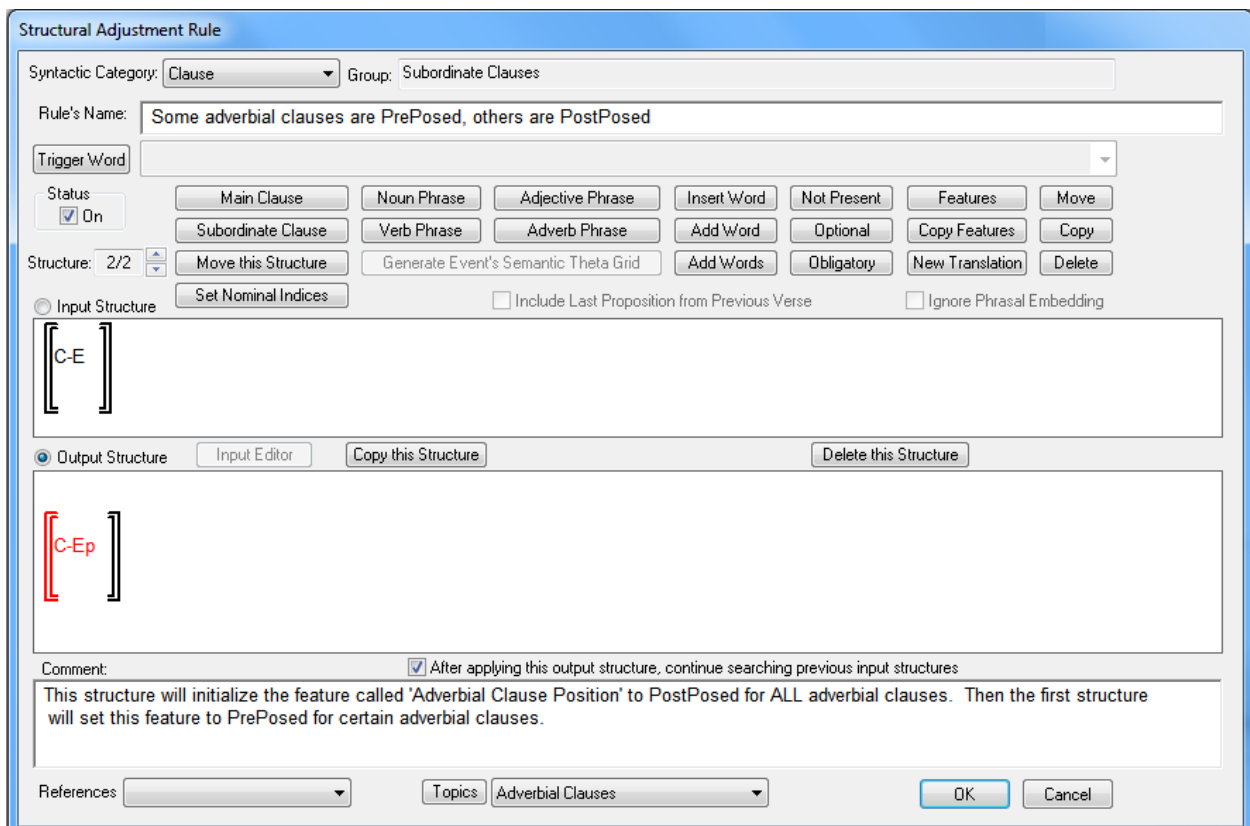
Figure 13-2. Structural Adjustment Rule that Initializes 'Adverbial Clause Position'

Save this rule, and generate Clauses 1:70 again. Then rest the cursor near the bottom of the adverbial clause boundary in the semantic representation window, and you'll see in the popup that the feature called 'Adverbial Clause Position' was set to PostPosed by structure 2, and then set to PrePosed by structure 1.

Now let's write a structural adjustment rule that inserts constituents into the semantic representations. Go to Clauses 1:66 which produces *It pleased Mary that John read a book*. This proposition has a subject complement *John read a book*. English postposes subject complements and inserts the pronoun *it* as the clause's subject. Link PLEASE-A to *please*, and click the Generate button. You'll see *Pleased Mary. John read a book*. The clause PSR doesn't yet know where to position subject complement clauses, so open your clause PSR and click the button labeled 'Insert New Constituent into Rule.' Then select Clauses from the dropdown, and select Agent for the clause's Type ('agent proposition' is the semantic equivalent of 'subject complement'). For the constituent's name, enter **Subject Complement**, and click the OK button. You'll then see 'Clause – Subject Complement' inserted at the top of the rule. Drag that constituent down the list and drop it on 'Clause - Patient' because that's where English positions its postposed subject complements. Close the PSR and generate this verse again. This time you'll see *Pleased Mary John read a book*. Now let's write a structural adjustment rule that will insert the pronominal subject *it* and the complementizer *that* whenever there's a subject complement clause. Open the Structural Adjustment Rules node and double click on the group called 'Subordinate Clauses.' You'll be prompted to select a syntactic category; select Clauses because we want this rule to apply to all subject complement clauses. Give this rule a name such as **Insert *it* as the Subject of Sentences that have a Subject Complement Clause**. Now let's start building the input structure. Insert a subordinate clause into the input structure, and set its Type to Agent. That's all we need to specify in the input structure, so click the radio button labeled 'Output Structure.' We need the output structure to insert a subject NP into the sentence, so click the button labeled 'Noun Phrase,' and then click somewhere between the closing subordinate clause bracket and the closing main clause bracket in the output structure. Note that it doesn't matter where you insert this NP into the output structure (but don't insert it into the adverbial clause); the phrase structure rules always determine the final constituent order. You'll see an opening NP boundary inserted into the output structure. Then click again somewhere after the opening NP boundary, and you'll see the NP's closing bracket inserted into the output structure. Now click the button labeled 'Insert Word,' and click somewhere inside the NP that you just inserted. We want to generate *it*, so let's insert the target noun *thing*, and we'll set its 'Surface Realization' value to 'Unambiguous Pronoun' so that it will eventually be realized with *it*. In the 'Select a Word' dialog, click the radio button labeled 'Target Words,' and then select Nouns in the dropdown at the top. Open the large dropdown, and you'll see that we haven't yet entered *thing* into the lexicon, so click the button labeled 'New Word.' Then enter *thing*, and after you've specified the features, form, and gloss, you'll see *thing* inserted into the NP. We want this NP to be the subject of the sentence, so now we need to specify its features. Click the Features button, and then click on the opening NP boundary in the output structure. Set its 'Semantic Role' to Agent, 'Thing-Thing Relationship' to

Unspecified, Relativized to No, and its 'Grammatical Relation' to Subject. Then click on *thing* in the output structure, and set its 'Participant Tracking' to Generic, and its 'Surface Realization' to 'Unambiguous Pronoun.' Assign this rule to the grammar topic called 'Subject Complement Clauses.' Then save this rule and generate the verse; you'll see *It pleased Mary John read a book*. All we need to do now is insert the complementizer *that*. Let's go back to the structural adjustment rule we just wrote and insert the complementizer *that* into the subject complement. So open that rule and click the button labeled 'Insert Word.' Then click somewhere inside the subject complement clause in the output structure. You'll then see the dialog that lets you enter a word. This time select the radio button labeled 'Target Words in User Defined Categories.' For the Tag, select Complementizer, and enter *that* into the 'New Word' text box. Close this dialog by clicking OK. Your rule should now look like the one in Figure 13-3.
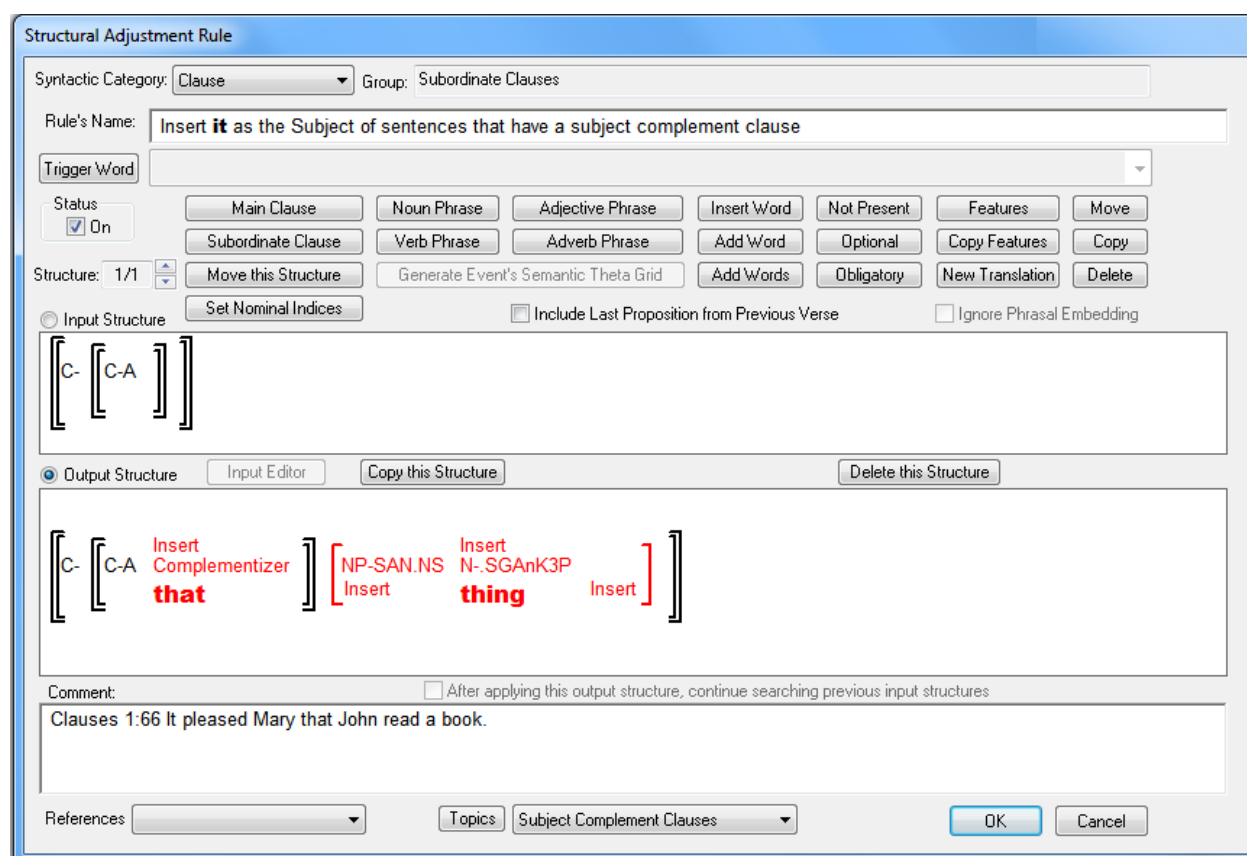

Figure 13-3. Structural Adjustment Rule for Subject Complement Clauses

Save this rule and generate the verse again. This time you'll see *It pleased Mary that John read a book*.

Now let's write a structural adjustment rule that deletes a constituent. Go to Clauses 1:80 which produces *John read this book instead of reading that book*. If you click the Generate button now, you'll see *John read this book instead of John reading that book*. In adverbial clauses that begin with INSTEAD OF, we want to delete the subject NP because it is always coreferential with the matrix subject. Double click on the group called 'Subordinate Clauses' again, and select Adpositions for the rule's syntactic

category because INSTEAD OF is an adposition. Enter a name for this rule such as **Delete the Subject NPs in *instead of* Adverbial Clauses**. Now let's build the input structure, so click on the button labeled 'Insert Word.' Then click in the input structure somewhere between the opening and closing clause boundaries. You'll see the dialog that prompts you to enter a word. Select Adpositions from the dropdown at the top; that will cause all of the adpositions in the ontology to be loaded into the large dropdown. Rather than searching through all the adpositions for INSTEAD OF-A, click the button labeled 'Lexical Subset.' You'll see a dialog, and it will always be initialized with the reference of the current verse, in this case Clauses 1:80. Click OK to close that dialog, and then open the large dropdown in the 'Select a Word' dialog again. This time you'll see only INSTEAD OF-A because it is the only adposition in this verse. Select INSTEAD OF-A, and then close the dialog. Now insert an NP into the input structure, and set its 'Grammatical Relation' to Subject. That's all that's necessary for the input structure, so click the 'Output Structure' radio button. We need this rule to delete the subject NP, so click the button labeled Delete, and then click the opening NP boundary in the output structure. You'll see that both the opening and closing boundaries for the NP will be marked as deleted. This is an indication that the entire phrase will be deleted from the semantic representation.   Assign this rule to the grammar topic called 'Adverbial Clauses.' Your rule should now look like the one shown in Figure 13-4.
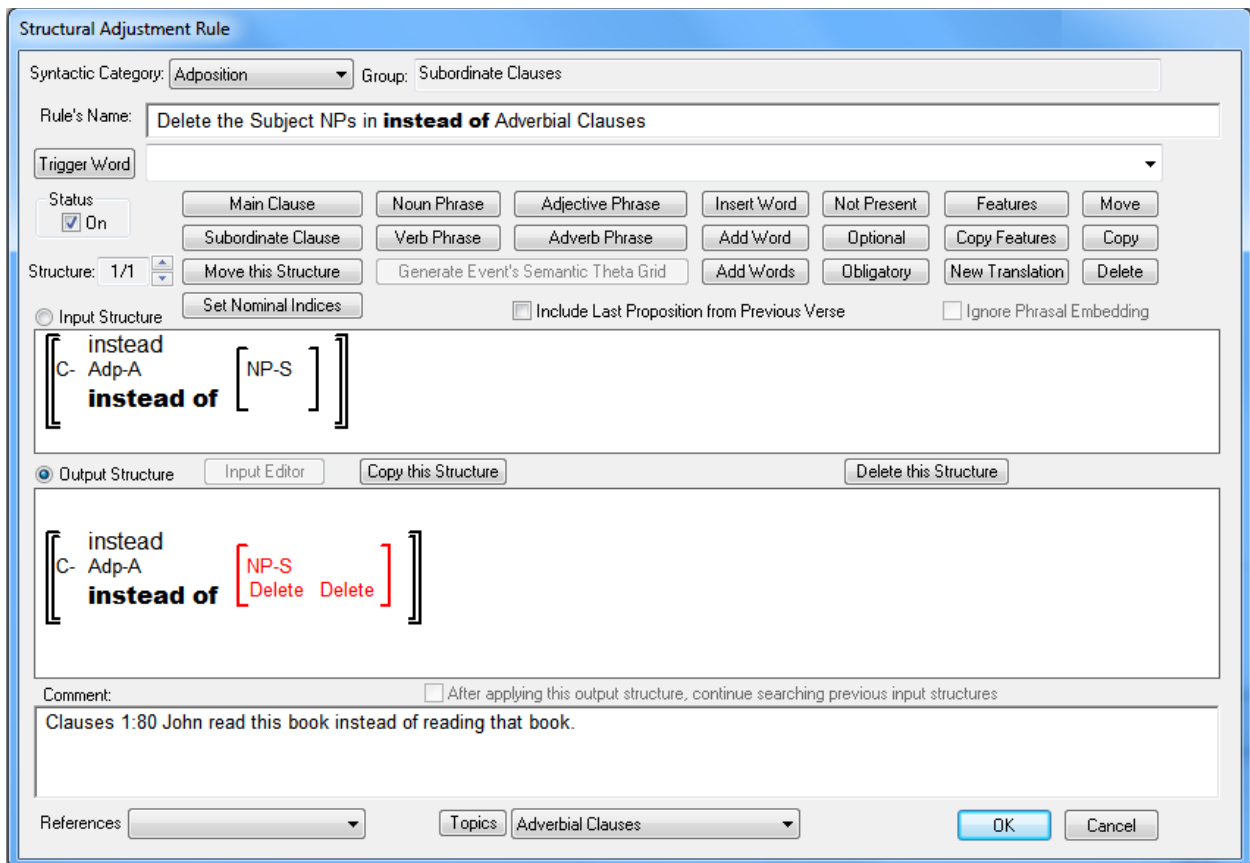

Figure 13-4. Structural Adjustment Rule that Deletes Subject NPs

Click the Generate button now, and you'll see *John read this book instead of reading that book*. If you rest the cursor near the bottom of the opening subordinate clause boundary in the semantic representation window, you'll see a popup indicating that *John* was deleted by the rule we just wrote.

Now let's write a rule that will copy a constituent. Go to Adverbs 1:2 which produces *John walks more quickly than Mary*. Link QUICKLY to *quickly*. Suppose the equivalent in your language is *John walks more quickly than Mary walks*. If that's the case, you need a structural adjustment rule to copy the VP from the main clause into a new type of clause that we'll call 'Adverbial Comparative.' To accomplish this, enter a new group called **Comparative Adverb Constructions** into the structural adjustment rules' node. Then double click on that group name, and select Adverbs as the syntactic category because we want this rule to be triggered by comparative adverbs. Give the rule a name such as **For comparative adverb constructions, insert an Adverbial Comparative Clause and copy the VP from the main clause into the Adverbial Comparative Clause**. Then build the rule's input structure as shown in Figure 13-5. Be sure to specify that the adverb's Degree is Comparative.
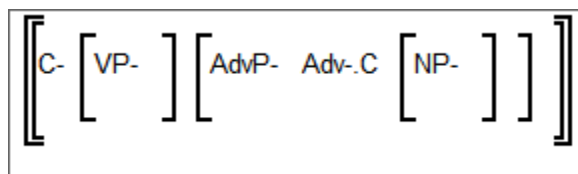

Figure 13-5. Input Structure with a Comparative Adverb

After you've completed the input structure, click the 'Output Structure' radio button. Let's begin by inserting a subordinate clause around the NP that's embedded in the adverbial phrase. Click the 'Subordinate Clause' button, and insert the opening clause boundary before the NP, and the closing clause boundary after the NP. Now let's copy the VP from the main clause into this new subordinate clause. Click the button labeled Copy, and then click on the opening VP boundary. You'll then see the label in the highlighted button change to Paste indicating that you need to specify where to paste the copied constituent. Click somewhere after the embedded NP's closing boundary but before the new embedded clause's closing boundary. You'll then see a new VP inserted into the subordinate clause, and a red arrow will be going from the original VP to this new VP. Now we need to specify the features, so click the Features button. Click on the opening subordinate clause boundary and you'll see a dialog showing all of the features and their values for clauses. We need to add a new value to the Type feature, so click the 'Feature Set' button at the bottom of the dialog. You'll see the same dialog where we always define our features and feature values. The dialog comes up with the Type feature for clauses selected, so enter **Adverbial Comparative** into the first blank row in the large table. Click OK, and then in the features dialog you'll see that value added to the list of values under Type, and it will be highlighted in purple indicating that it is a user defined value and not one of the standard values in the semantic representations. Select that value for Type, and close the features dialog. Then click on the opening phrase boundary for the embedded NP, and set its 'Grammatical Relation' to Subject. Then click on the button labeled 'Insert Word,' and insert the complementizer *than* into the embedded clause. Assign this rule to the

grammar topic called Comparatives. Your structural adjustment rule should now look like the one shown in Figure 13-6.
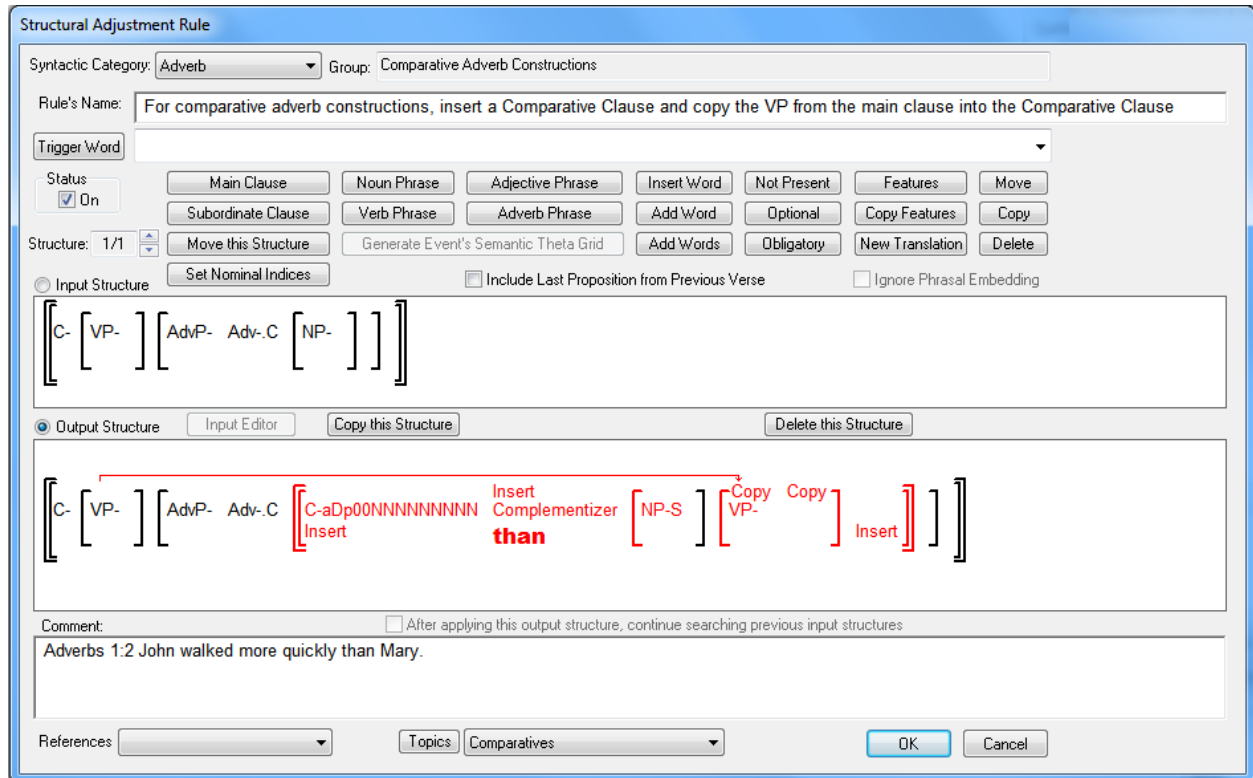

Figure 13-6. Structural Adjustment Rule for Comparative Adverbs

Save this rule and generate the verse. You'll see *John walked. Quickly than Mary walked*. So the structural adjustment rule copied the VP, and inserted both the subordinate clause and the complementizer. However, the clause PSR doesn't yet know where to position adverbial phrases. Modify your clause PSR so that it positions adverb phrases immediately after the VP, and use Grammar Wizard to write a spellout rule that will add the word *more* to adverbs that have a Degree value of Comparative. That spellout rule should look like the one shown in Figure 13-7.
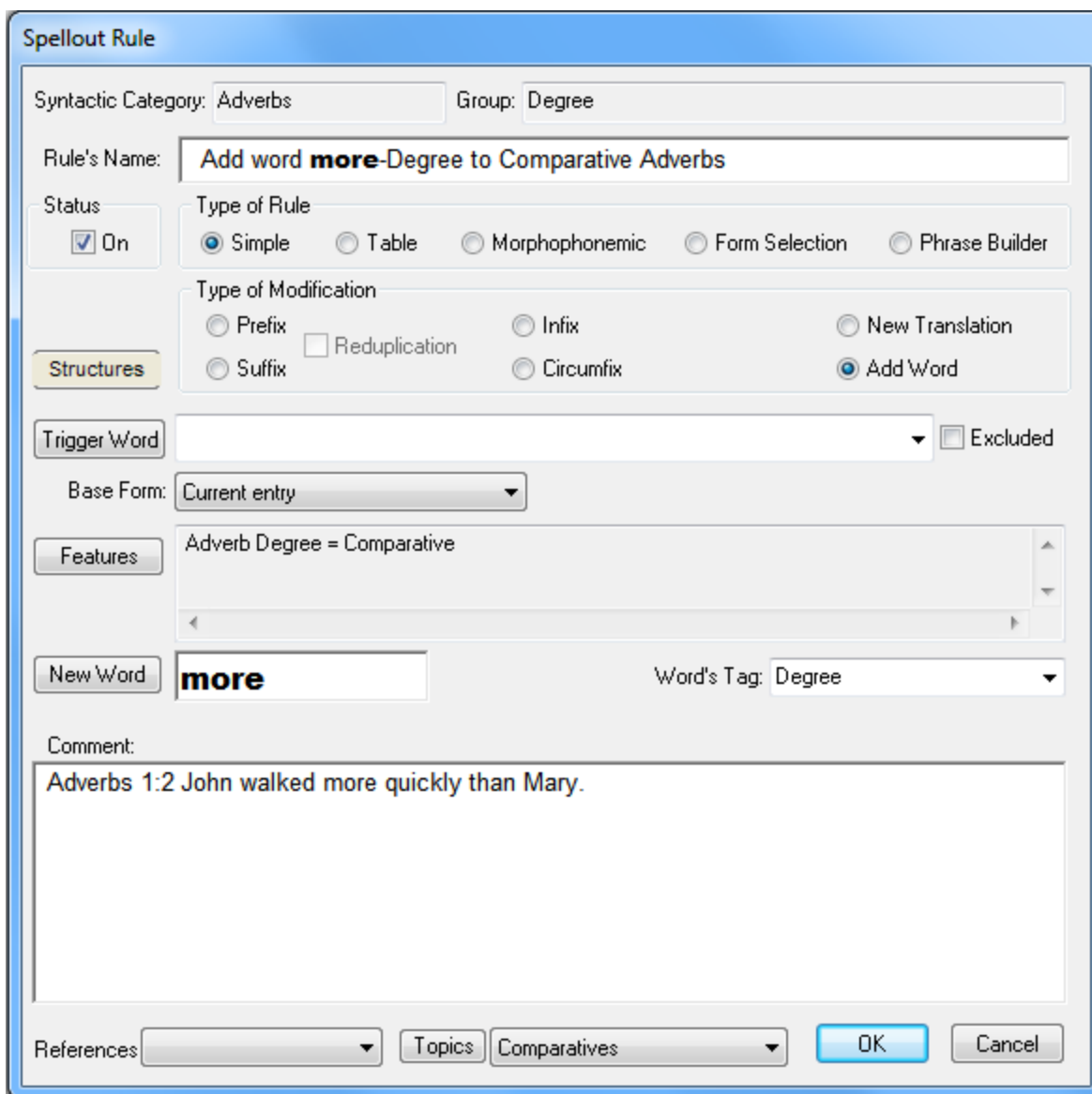
Figure 13-7. Spellout Rule for Comparative Adverbs

Save your spellout rule, and generate this verse again. This time you'll see *John walked more quickly than Mary walked.*

Now let's write a structural adjustment rule that copies feature values from one constituent to another. Let's build a rule that will insert aspectual auxiliaries such as *finish, start, continue*, etc. In English these auxiliaries are marked for tense rather than the semantic main verb, so we'll need to copy the 'Target Tense & Form' value from the main verb to the new auxiliary. Begin by entering a new group called **Verbal Auxiliaries** into the node for Structural Adjustment rules. Then double click on that group and select Verbs as the syntactic category. Let's deal with completive aspect first, so build an input structure that contains a VP and a verb, and specify that the verb's 'Target Aspect'[1] must be Completive. The input structure for this rule is shown below in

---

[1] Be sure to use the feature called 'Target Aspect' rather than the feature called 'Aspect.'

Figure 13-8. After building the input structure, click the 'Output Structure' radio button. Now we need to insert the target verb *finish* into the output structure, so click the button labeled 'Insert Word,' and then click somewhere in the output structure after the opening VP boundary but before the verb. You'll then see the 'Select a Word' dialog. We want to insert the target verb *finish*, so click the radio button labeled 'Target Words,' and then select Verbs in the Syntactic Category dropdown. If you open the large dropdown, you'll see that we haven't yet entered *finish* into the lexicon, so click the button labeled 'New Word.' Enter *finish*, and specify its forms and gloss. It will then be selected in the large dropdown, so click OK and you'll see *finish* inserted into the output structure. Now we want to copy the 'Target Tense & Form' value from the main verb to this auxiliary, so click the button labeled 'Copy Features.' Then click on the main verb in the output structure, and you'll see a dialog listing all the features for verbs. Select 'Target Tense & Form,' and then click OK. You'll see that the label on the highlighted button has been changed from 'Copy Features' to 'Paste Features' indicating that you can now click on the constituent that you want to copy the features to. Click on *finish*, and you'll see that 'Target Tense & Form' will be copied to *finish*.[2] Assign this rule to the grammar topic called Aspect, and click the OK button. Your rule should now look like the one shown in Figure 13-8.
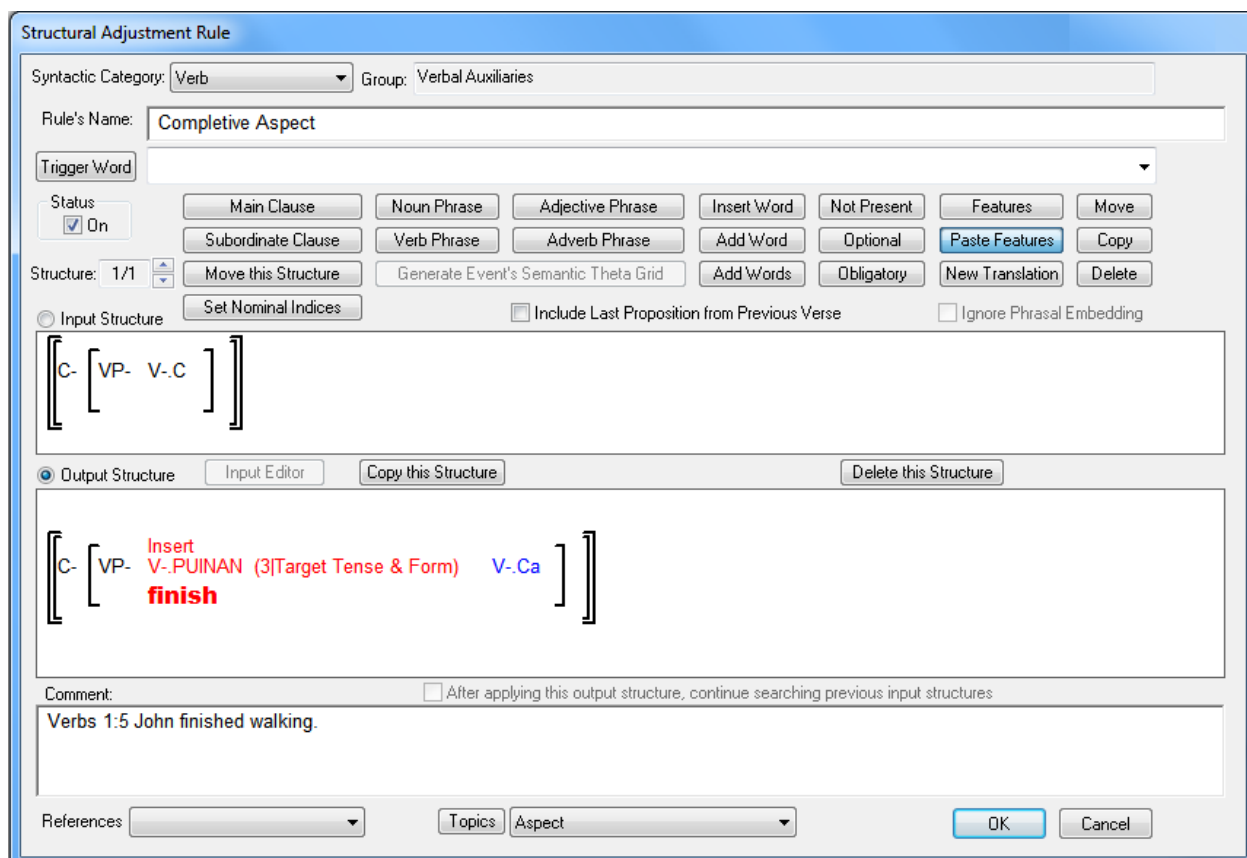

Figure 13-8. Structural Adjustment Rule that Copies a Feature

---

[2] In the features for *finish*, you'll see "(3|Target Tense & Form)." This means that the value of 'Target Tense & Form' from constituent 3 in the input structure will be copied to this word. Constituent '3' in the input structure is the main verb because clause and phrase boundaries are counted as constituents.

Go to Verbs 1:5 which produces *John finished walking*, and click the Generate button. You'll see *John finished walking*.[3] Now go to Verbs 1:6 which produces *John started walking*. Add another structure to the rule we just built, and use that structure to insert *start* into the VP whenever the verb's Target Aspect is Inceptive. After Verbs 1:6 is being generated correctly, go to Verbs 1:7 which produces *John stopped walking*. Add another structure to the structural adjustment rule that will insert the verb *stop* whenever the verb's Target Aspect is Cessative. Repeat this process for Verbs 1:8 which produces *John continued walking*.

Now let's write a structural adjustment rule that will move a constituent. In a previous tutorial you used Grammar Wizard to write a constituent movement rule, but this time we'll write a structural adjustment rule that will move one proposition into another proposition under certain circumstances. This is very helpful because the semantic representations consist of short, simple sentences, and if the generated texts also consist of only short, simple sentences, the texts sound immature and require extensive editing. Let's write a structural adjustment rule that will look for two consecutive propositions that have the same subject nominal, and the two propositions are joined with AND-A. When a situation such as that is found, the rule will move the second proposition into the first, and delete the second proposition's subject nominal. Go to Avian Influenza 1:2 and look at the last two propositions. They produce *This disease is in many countries. And this disease has killed many people*. We want to combine this into a single proposition that reads *This disease is in many countries and has killed many people*. Link all of the unmapped concepts to target words. If you generate the verse now, you'll see "*The disease be in many countries. The disease kill many people. And*" Modify the table spellout rule that inserts articles so that it includes a Proximity value of 'Contextually Near with Focus' and inserts *this*. Also modify the clause PSR so that it includes a Conjunction at the beginning of the sentence. Now when you generate this verse, you'll see "*This disease be in many countries. And this disease kill many people.*" We're not going to deal with the problems of tense in these two sentences; we're only going to combine them into a single proposition. Add a new group called **Combine Simple Propositions**, and double click on that group name. For this rule's syntactic category, we could choose Nouns because two nouns with the same List Index value are required to trigger this rule, or we could choose Conjunctions because we want the two propositions to be joined with AND-A, or we could choose Clauses. Any of these would work, but let's select Nouns. After selecting the rule's syntactic category, enter a name for this rule such as **Combine two simple propositions that have the same subject nominal and are joined with *and* into a single proposition**. In order to add a second proposition to the input structure, click the button labeled 'Main Clause.' You'll then see a second proposition added to the input structure. You may put as many propositions into an input structure as you need. Then insert subject NPs into both propositions, and insert nominals into both NPs. Then specify that the two nominals must have the same List Index value. Also insert AND-A into the second proposition. Your input structure should now look like the one shown in Figure 13-9.

---

[3] If you see *John finishes walking*, check your verb spellout rules. You may need to turn off a rule from an earlier tutorial that uses the feature called Time.
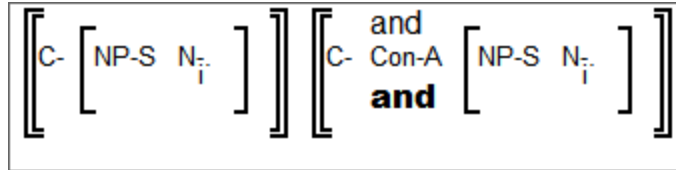
Figure 13-9. Input Structure for Combining Propositions

Now we're ready to build the output structure, so click the 'Output Structure' radio button. Then click the Move button, and click on the opening boundary of the second proposition. You'll then see the label on the highlighted button change to Insert, indicating that you can now specify where the proposition should be inserted. When moving a constituent, we always click on the opening phrase or clause boundary that we want the constituent to be moved into. Remember that it is always the phrase structure rules that determine the final constituent order. In this case we want to move the second proposition into the first, so click on the opening boundary of the first proposition. You'll then see a red arrow indicating that the second proposition will be move into the first. Save this rule and generate the verse now. You'll see *This disease be in many countries. And this disease kill many people.* If you look in the semantic representation window, you'll see that the second proposition has been moved into the first. We need to modify our rule so that it will delete the period at the end of the first sentence, and also delete the subject NP in the second sentence. So open your rule and click the 'Insert Word' button. Then click in the input structure somewhere between the closing NP boundary and the closing proposition boundary of the first proposition. We want to insert a period so that this rule can delete it, so select the radio button labeled 'Target Words in User Defined Categories.' In the Tag text box enter **period** because that is the tag for all periods. You don't need to enter anything into the 'New Word' text box, so click the OK button. You'll then see the period inserted into the first proposition in both the input and output structures. Now click the Delete button, and then click on the period in the output structure, and also click on the subject NP in the second proposition. Your final rule should look like the one shown in Figure 13-10.
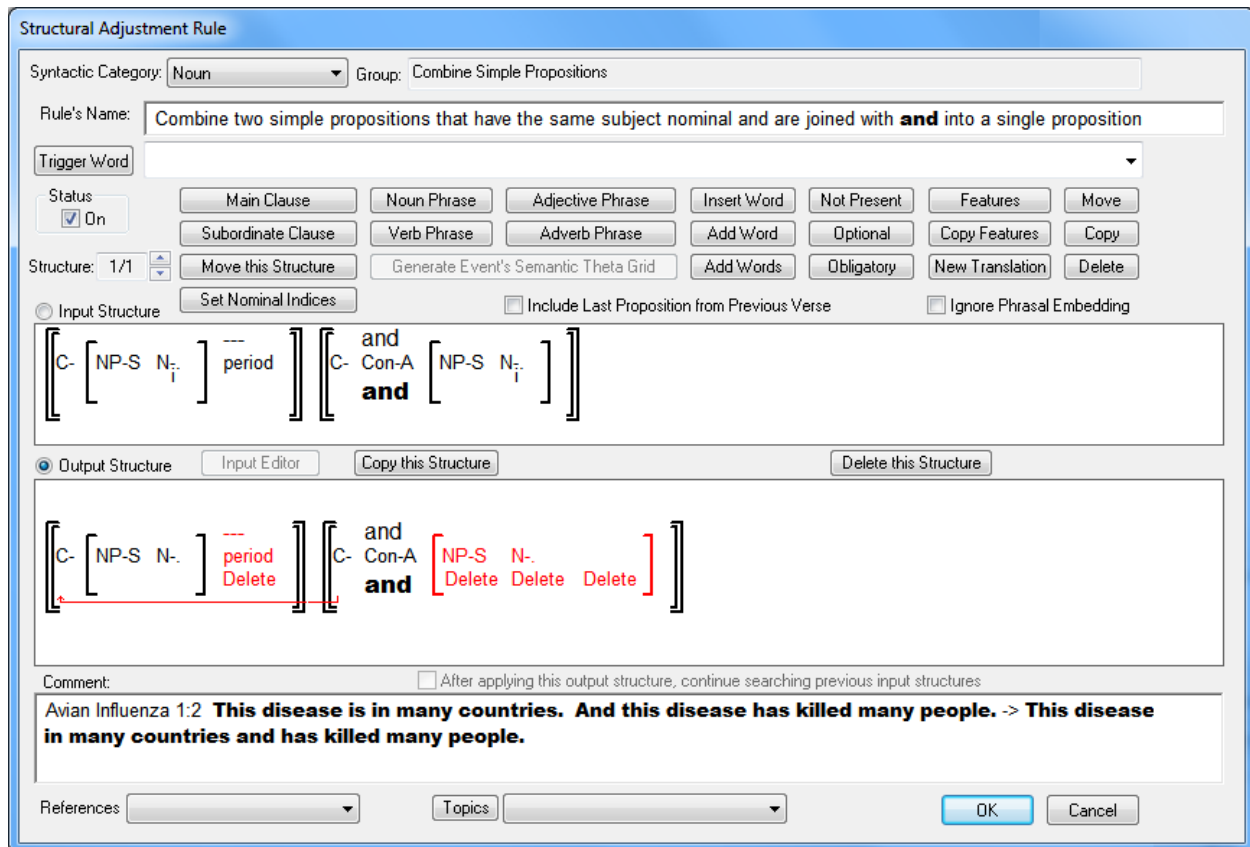
Figure 13-10. Structural Adjustment Rule that Combines Propositions

Save this rule and generate the verse again. This time you'll see *This disease be in many countries and kill many people*. As was mentioned earlier, we won't deal with the tense problems in this tutorial.

Now go to Infected Eye 1:10 which produces "*Then Netty came to Melissa's house. After Netty looked at Melissa's eyes, she said to Melissa, 'Your eyes are very sick. Some germs have entered your eyes. We have to wash your eyes. We have to clean your eyes thoroughly. And we have to clean your eyes each day until they become healthy again.*" There are many concepts in this verse that aren't linked to target words, but let's ignore that issue. Click the Generate button, and then look at the last two propositions in the semantic representation window. You'll see that the last proposition has been moved into the preceding proposition by the rule that we just wrote. However, the last proposition includes an embedded adverbial clause (*until they become healthy again*), and when there's an embedded adverbial clause, we generally don't want the propositions combined. In order to prevent our rule from combining propositions that include an embedded proposition, open the rule and insert an embedded proposition anywhere in the second proposition in the input structure. Then click the button labeled 'Not Present,' and click on the opening boundary of that embedded proposition. You'll see that the embedded proposition is then covered with black cross hatches. Those black cross hatches indicate that the constituent must not be found when searching the semantic representations; if the constituent is found, then the rule won't apply. The input structure for your rule should now look like the one in Figure 13-11.
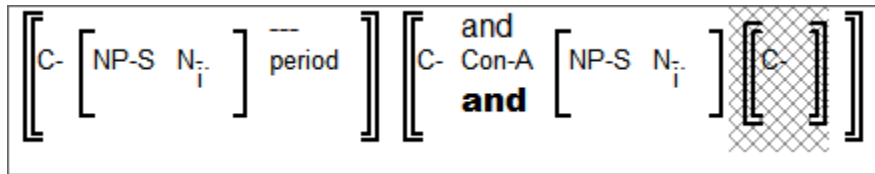
Figure 13-11. Input Structure with a Constituent tagged as 'Not Present'

Close this rule and generate the verse again. This time the rule won't combine the last two propositions.

   Structural adjustment rules are also able to provide new translations for concepts in specific environments. However, this process is very straight forward so it will only be briefly described here. Open your structural adjustment rule that inserts aspectual auxiliaries, and click on the 'New Translation' button. Then click on the main verb in the output structure (not the word *finish* that you inserted), and you'll see a dialog that will let you select a target word. Select the verb *sit* and then click OK. You'll see *sit* displayed below the main verb in the output structure. This rule will now change the main verb to *sit* whenever it is applied. Your output structure should now look like the one shown in Figure 13-12.
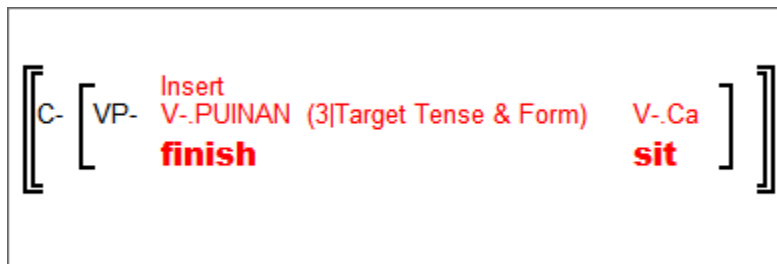

Figure 13-12. Output Structure with 'New Translation'

Close the rule and go back to Verbs 1:5. This time when you generate the verse, you'll see *John finished sitting*.

13.2 Conclusions

   You now know how to use all the types of rules in both the transfer and synthesizing grammars. It is important to remember that the purpose of the transfer grammar is to produce a new underlying representation that is appropriate for your target language. Therefore, after all the rules in the transfer grammar have been executed, the semantic representation should be in a new format that consists of your target language's words, features, and structures. The rules in the synthesizing grammar aren't capable of restructuring the semantic representations; that is the task of the transfer grammar.