A Binding Rule for Government-binding Parsing

Nelson CORREA
IBM Thomas J. Watson Research Center
P. O. Box 704
Yorktown Heights, NY 10598
USA

Abstract

In this paper I propose a *Binding rule* for the identification of pronoun and anaphor referents in phrase-structure trees, assuming the general framework of the Government-binding theory outlined by Chomsky (1981). The Binding rule, specified by means of an attribute grammar, is a particular instantiation of the Free Indexing rule and binding axioms in Chomsky's Binding theory, with certain empirical and practical advantages. The complexities of the Binding rule proposed, as well as that inherent in Chomsky's Binding theory, are studied, and it is shown that the new rule is more psychologically plausible and computationally efficient than the original theory on which it is based. The fragment of the attribute grammar shown here is part of an English grammar and parser being developed in the Prolog and PLNLP languages.

Introduction

Binding is a component subtheory of Government-binding which applies in the derivation of the logical form of utterances from their surface representation. The area of semantic interpretation dealt with by the binding theory is that of *anaphora*. Binding theory defines only syntactic conditions on anaphora; the reader is referred to /Hobbs, 1978/ for some of the extra-syntactic factors that might be involved. Binding assumes an Indexing rule which applies to an input S-Structure tree and annotates it, assigning to every NP node in the input tree a *referential index*, which represents the coreference relation of the NP with other NPs in the input.

In this paper research is continued on the use of attribute grammars to provide a fully explicit and computationally oriented statement of the Government-binding (GB) theory /Correa, 1987/. The Binding rule presented here improves over the standard statement of the Binding theory in two respects: From an empirical point of view, the new rule accounts for crossover binding phenomena /Kuno, 1987/ without recourse to reconstruction /Chomsky, 1981/; from a practical point of view, the new rule is more computationally sensible than the generate-and-test approach understood in Chomsky's theory, and hence is a plausible candidate for incorporation in natural-language parsers that account for anaphora. Previous literature on GB parsing /Wehrli, 1984; Sharp, 1985; Kashket, 1986; Kuhns, 1986; Abney, 1986/ has not addressed the issue of implementation of the Binding theory.[1] The present paper intends in part to fill this gap.

In the development below I will assume that the reader is familiar with attribute grammars and the basic concepts and terminology of Government-binding, although not necessarily with the Binding theory. The reader is referred to Waite and Goos (1984) for a concise introduction to attribute grammars, and Sells (1985) for the basic assumptions of Government-binding.

Chomsky's Binding Theory

Binding theory defines the syntactic constraints on coreference that exist between the noun phrases in a sentence. In the course of doing this, the theory indirectly determines constraints on the distribution of certain kinds of noun phrases. In this section we review the standard formulation of the Binding theory; the reader already familiar with it may proceed to the next section.

The referential possibilities of a noun phrase depend on the functional *type* of the NP and the Binding conditions for that type. Government-binding distinguishes three types of overt NP, shown in (1).

---

[1] Sharp (1985) checks correctness of binding in traces; we consider lexical NPs here.

(1) a. anaphor (reflexive and reciprocal)

b. pronominal

c. referential

An anaphor is an expression that has no independent reference and *must* take its reference from some other expression in the sentence in which it occurs. English has *reflexive* and *reciprocal* anaphors, such as *'themselves'* and *'each other'* in (2). The NP from which an anaphor or pronominal takes its reference is called its *antecedent*. since an anaphor must have an antecedent within the sentence in which it is used, we obtain the contrast between (2.a) and (2.b). If there is no appropriate antecedent, the string is ill-formed at the Logical Form level. The antecedent of the anaphor must, furthermore, c-command the anaphor and be found within a certain *local domain*, notions to be made precise below. Thus, in (2.c), although there is a potential antecedent for the anaphor, namely *'Greeks'*, it is not within the required local domain. In (2.d), there is a potential antecedent *'donkey'*, but it does not c-command the anaphor. Hence the string is also ill-formed.

(2) a. Greeks like *themselves/ each other*.

b. * *Each other/ Themselves* like Greeks.

c. * Greeks$_i$ think that *each other$_i$/themselves$_i$* are smart.

d. * Every man who owns a donkey$_i$ beats *itself$_i$*.

A pronominal is a pronoun in any of its inflected forms (e.g., as due to agreement and Case-marking), as in (3). Pronominals exhibit a distribution in phrase structure trees nearly complementary to that of anaphors. A pronominal need not pick its reference from some other NP in the sentence, but rather may have independent (deictic) interpretation, as in the first reading of (3.a). The pronominal may also be read anaphorically, having its reference determined by some other NP in the sentence (3.a-b). In this case, though, the antecedent must either be outside the local domain of the pronominal, or not c-command it. Hence, the assigned coreference in (3.a-b) is possible, while that in (3.c) is not. Within a local domain, where an anaphor must have an antecedent, a pronominal cannot.

(3) a. Brigitte$_i$ said that *she$_{j/i}$* is tired.

b. Every man who owns a donkey$_i$ beats *it$_i$*.

c. * Sibylle$_i$ loves *her$_i$*.

Lexical or fully referential expressions are names like *'John'* and *'the man'* in (4); the class includes all nominals headed by a common or proper noun. A referential expression defines its reference independently and must be free in every domain, in the sense that it may not have a c-commanding antecedent. Thus the interpretations in (4.a-b) are unwarranted. Coreference between referential NPs is possible only if the first NP does not c-command the second (4.c-d); the result, though, may be awkward or place emphasis on the anaphoric noun phrase.

(4) a. * John$_i$ likes *John$_i$*.

b. * John$_i$ wants that *John$_i$* leaves.

c. The man who hired John$_i$ likes *John$_i$*.

d. John$_i$ came and *John$_i$* left.

The most difficult area of the Binding theory is the formulation of the notion *local domain* referred to above. This notion is defined such that it is identical for anaphors and pronouns. We note in advance, however, that while the notion is nearly identical for both, it should not be defined the same, as sentences (5.a-b) show (Chomsky, 1986). In this paper we shall not be concerned with the solution of this still open problem.

(5) a. The children$_i$ like *each other's$_i$* pictures.

b. The children$_i$ like *their$_i$* pictures.

Chomsky's axiomatic statement of the Binding theory is as follows. Chomsky (1981) assumes a *Free Indexing rule* which applies at LF and assigns (randomly) a referential index to every NP in the input structure. Two NPs are said to be coreferential if they bear the same referential index. The indexing rule massively overgenerates logical forms, and indiscriminately assigns unwarranted coreference relations. The annotated structures produced by the rule are subject to a number of well-formedness conditions, which are constraints on the assigned coreference relations.

The most elementary condition is the agreement condition (6). The main component of the theory is given by the Binding axioms (7), where the notions of *binding* and *local domain* are as in (8) and (9), respectively. Notice that the definition (9) of local domain does not distinguish between anaphors and pronominals, and thus is problematic, as the examples (5) indicate. We assume this definition, though, for the development below. The notion of c-command used in (8) is given in (10).

(6) **Agreement Condition**

If NP_1 and NP_2 are coindexed, then their agreement features $AGR = <Person, Gender, Number>$ agree.

(7) **Binding Axioms**

A. An *anaphor* must be bound within its local domain.

B. A *pronominal* must be free within its local domain.

C. A *referential expression* must be free in every domain.

(8) For nodes α and β, α *binds* β if (i) α is coindexed with β, and (ii) α c-commands β. A node α is *free* (within a given domain) if it is not bound (within that domain).

(9) The *local domain* of a node α is the subtree dominated by MGC(α), where

MGC(α), the *minimal governing category* of α, denotes the maximal projection μ nearest to α such that

μ dominates α, and

μ has an accessible Subject, and

μ dominates a governor γ of α

(10) For nodes α and β, α *c-commands* β if the first branching node dominating α also dominates β.

It is a straightforward task to verify that the Binding axioms in (7) explain the grammaticality judgements and interpretation possibilities of the examples presented thus far, except those in (5). The theory is explanatorily adequate, in the sense that it applies to a wide range of natural languages.

Procedural Binding

The Binding theory just outlined follows the style of most recent work within the Government-binding framework. Extremely general rules, such as the *Free Indexing rule*, are assumed for the generation and annotation of syntactic structure; the bulk of the grammar then consists of well-formedness conditions or *axioms* that must be satisfied by the generated structures. This approach, due to its extreme inefficiency, is problematic as a model of linguistic performance or natural language parsing. It seems more appropriate to view the general rules and axioms that constrain them as high-level *specifications* of certain grammatical processes, rather than as models of how the processes are actually carried out.

The refinement of the general rules and axioms associated with them into procedural rules which may be used to derive structure that *already* satisfies the axioms is not a straightforward task, and has only recently begun to be addressed /Abney and Cole, 1986; Barton, 1984/. The incorporation of axioms into the rules leads to grammars which are more sensitive to psychological issues/linguistic processing, rather than mere linguistic description. It seems clear that only these new rules may be used in practical natural language parsers. Furthermore, the formulation of procedural mechanisms provides a new way of looking at linguistic phenomena, which may in turn lead to insights for the solution of outstanding problems. I offer the following Binding rule as an illustration.

The *Binding rule* is defined by means of attribution rules associated with productions in the base. It applies at S-Structure and assigns to each NP node in the structure a referential index, in such way that the Binding axioms are satisfied by the assignment. The generate-and-test method implicit in Chomsky's account is avoided. In those S-Structures for which there is no possible correct assignment, the rule blocks, and the structures are marked ill-formed, due to some violation of the Binding theory. The rule applies after the functional type of every NP has been determined, according to lexical features of the head nominal and principles of the Government and Case theories. Functional classification of an NP consists of determining the values of its attributes *anaphoric* and *pronominal* /van Riemsdijk and Williams, 1986/. The first approximation to the rule is limited to cases of backward reference only; assignment of forward coreference, as in (11), will not be covered by the rule. Also, we ignore cases where referential expressions may be used anaphorically, as in (4.c-d).

(11) Men who met *her*$_i$ saw how kind Mary$_i$ was.

The formulation of the rule relies crucially on the following hypothesis: For every NP node in an S-Structure, it is possible to define two sets of nominal expressions *AAS* and *PAS*, which contain, respectively, potential anaphoric and pronominal antecedents. Given a mechanism to compute the two sets noted, an antecedent for the current node may be selected from the appropriate set, according to the current node's functional type, as in (12). Attribution rule (12) is associated with every production for NP and defines the value of the NP's referential index. The function *select-from* takes an ordered set as argument and selects (arbitrarily) the first element that morphologically agrees with the NP.[2]

(12) **Binding Rule:**
NP.*RefIndex* ←
if NP.*anaphoric* then
if NP.*pronominal* then /*Control*/
else *select-from*(*AAS*)
else if NP.*pronominal*
then *select-from*(*PAS*)
else NP.*node*

The main component of the Binding rule consists of the attribution rules that define the values of the *AAS* and *PAS* sets at each node. I now proceed to describe the types of the attributes involved in the computation and the manner in which these values are defined.

Binding attributes and their types

Assume integer-valued attributes *node* and *RefIndex*. The attribute *node* is associated with every node in an S-Structure tree, enumerating them in preorder. Thus the *node* number of an NP may be used to identify the NP. *RefIndex* represents the referential index of the NP with which it is associated. This attribute is synthesized by rule (12) and its value is equal to the referential index of the first NP with which the current NP corefers (assuming a preorder enumeration of tree nodes). When NP.*RefIndex* = NP.*node*, for some NP, we say the NP has independent reference.

The attribute *AAS* contains, for a given NP, the sequence of c-commanding NPs found within the local domain of the current node. Thus, any NP in this set is a potential antecedent for the current node, if that node is anaphoric. Each element in the *AAS* is a pair of the form <NP.*RefIndex*, NP.*AGR*>, for some NP to the left of the current node. NPs are ordered in the *AAS* in such way that the most recently found NP is ranked first (*AAS* is a stack, or ordered set). The attribute *PAS* is similar to the *AAS*, except that each element in it either does not c-command the current node, or is outside its local domain. Thus, each NP in the *PAS* is a potential antecedent for the current node, if that node is pronominal.

An important difference between the *AAS* and *PAS* is that, if the current node is an NP, say NP_i, the pair $<NP_i.node, NP_i.AGR>$ is a member of *PAS*, but not *AAS*. Because of this, a pronominal's referential index may be set to its own *node* number (i.e., may be interpreted deictically), while an anaphor's may not. This difference between the *AAS* and *PAS* need not be stipulated as a special case, but rather follows naturally if we assume the c-command relation is irreflexive.

The distribution of values for the *AAS* and *PAS* attributes in an S-Structure may be illustrated by means of example (13), in which the subscripts are NP *node* numbers; we ignore their actual values.

(13) John_h told [his_i parents]$_j$ about himself_k.

The values that result for the *AAS* and *PAS* are shown in (14); the reader may verify their correctness with the aid of examples (15). For the first NP, *'John'*, there is no potential anaphoric antecedent (15.a), so the *AAS* is empty (14.a). However, at that position it is possible to have a free pronoun, so the *PAS* contains a single entry, the pair $<h, AGR_h>$. For the second NP, *'his'*, the values of *AAS* and *PAS* are as in (14.b). Thus the *AAS* is empty and no anaphor is permissible at the position (15.b), while a pronoun is, in which case it may be interpreted deictically or anaphorically, referring back to *'John'*. The values of the *AAS* and *PAS* attributes associated with NP_j and NP_k are as shown in (14.c-d).

(14) a. $\mathbf{NP}_h.AAS = \{\ \}$

$\mathbf{NP}_h.PAS = \{<h,AGR_h>\}$

b. $\mathbf{NP}_i.AAS = \{\ \}$

$\mathbf{NP}_i.PAS = \{<i,AGR_i>, <h,AGR_h>\}$

c. $\mathbf{NP}_j.AAS = \{<h,AGR_h>\}$

$\mathbf{NP}_j.PAS = \{<j,AGR_j>\}$

d. $\mathbf{NP}_k.AAS = \{<j,AGR_j>, <h,AGR_h>\}$

$\mathbf{NP}_k.PAS = \{<k,AGR_k>, <i,AGR_i>\}$

(15) a. * *Himself*/ *He*$_i$ told his parents about himself.

b. John_i told [**himself's*/ *his*$_{j/i}$ parents] about himself.

c. John_i told *himself*$_i$/ **him*$_i$ to stop smoking.

d. John_i told [Mary_j's parents]$_k$ about *himself*$_i$/ *each other*$_k$/ *her*$_{h/j}$

[2] No theoretical significance is attached to the order of the elements in the *AAS* and *PAS*. Psycholinguistic evidence, however, suggests that gaps "reactivate" their antecedents, which bears on the order of the sets.

The attribution rules that define the values of the AAS and PAS sets are given in the Appendix. Only those rules associated with the categories I, C, N, and V and projections are given. An auxiliary attribute PAS_S is used in the computation of PAS. PAS_S is synthesized and adds to PAS all pronominal expressions contained in the phrase bearing the attribute.

Chain Binding

The Binding rule presented thus far does not account for *crossover* cases such as (16). In this example, NP_j has undergone Wh-movement from its D-Structure position e_j to its surface position in the matrix complementizer, crossing over the subject *'John'*. It is possible to interpret *'himself'* referring to *'John'*. The Binding rule presented associates an empty AAS with *'himself'*, and thus fails to account for this coreference possibility.

(16) [Which picture of *himself*$_i$]$_j$ does John$_i$ like e_j.

Example (16) is also problematic for the axiomatic Binding theory defined by (6)-(10) since, according to definition (8), the anaphor is not bound by its antecedent, and thus axiom A of the theory is violated. Chomsky (1981) proposes a rule of *Reconstruction*, which applies at LF prior to application of the Binding axioms, and has the effect of "reconstructing" the moved phrase at its D-Structure position, so that a structure similar to (17) is obtained. To this structure the axioms may now apply to yield the correct results.

(17) [∃x: x a picture]$_j$ does John$_i$ like [x of *himself*$_i$].

The rule of reconstruction, in addition to having its own set of problems /van Riemsdijk and Williams, 1986/, seems undesirable in the grammar, since it complicates the grammar specifically for the purposes of the Binding theory. The function of the rule, undoing the previous application of a transformation, is not very appealing.

Following in part a proposal by Barss (1983), the procedural Binding rule may be refined to account for (16) without recourse to reconstruction. Barss observes that the antecedent of an anaphor must c-command and be within the local domain of either the anaphor, or one of the traces of the phrase in which the anaphor is embedded. This represents a significant reformulation of the binding axioms, to include reference to A-Bar chains. The attribution rules that define the values of the AAS and PAS sets may be modified to take into account this observation. The change required is to define the values at the root of the moved phrase as the union of the value defined by the current rule, plus the values defined at the traces in the chain headed by the phrase. Thus, in (16) $NP_j.AAS$ and $e_j.AAS$ include the element $<i, AGR_i>$, and the attribution rules described in the previous section make this value accessible to the anaphor *himself*.

The exact reformulation of the attribution rules for AAS and PAS may be done in different ways. One alternative is to compute the extended AAS and PAS sets in two passes through the tree, with the second pass used to compute the unions noted. A second approach delays anaphora resolution for expressions inside an antecedent (chain head) until the last trace of the chain is found. At this time the AAS and PAS sets of the antecedent may be evaluated, having access to the corresponding sets in the traces. This second approach seems more sensible, since it permits application of the Binding rule in one top-down, left-to-right pass through the tree. We do not pursue the details of the revised rule here.

Complexity of Binding

First we consider Chomsky's Binding theory. The combination of the Free Indexing rule and Binding axioms defines a generate-and-test algorithm. Given assumptions of the $\bar{X}$ theory, the number of NPs in an input string is linearly related to the length of the string. Hence for some fixed and small k, for a sentence of length n there will be n/k NP nodes. Assuming a slight modification of the indexing rule (which improves it), according to which it selects integers in the range $1, ..., n/k$ to assign as potential referential indices to the NPs involved, there will be $(n/k)^{n/k}$ candidate LF assignments to be checked against the Binding axioms (7). Assuming that the Binding axioms may be checked in constant time, the running time for the algorithm is exponentially related to the length of the input string.

For the procedural Binding rule formulated here, the time needed to compute the synthesized AAS and PAS attributes at each node *from the attributes at that node on which AAS and PAS directly depend* may be assumed to be constant; the operations involved are assignment, push, and pop only. Assuming further that the number of empty categories inserted between terminal elements is proportional to the length of the input string, the number of nodes in the derivation trees generated is proportional to the input length. Since the AAS and PAS attributes are computed at most once at each node in the tree, the processing time for the new Binding rule is linear -- a significant improvement over the abstract specification (6)-(10).

Conclusions

In this paper an attribute-grammar specification of Binding rule for the identification of pronoun and anaphor referents has been proposed. The rule provides a correct account of backward reference of NPs, and also of forward reference due to movement, without recourse to reconstruction. The rule presents a model of Binding in which sets of potential anaphoric and pronominal antecedents are incrementally defined at each node in a tree. Storage use may be optimized by use of global storage cells, as described by /Sonnenschein, 1985/.

In more general terms, this rule presents a trend complementary to that of recent linguistic theory. The rule formulation indicates how conditions on representations may be incorporated into the rules which generate the representations in the first place. This leads to grammars more geared to linguistic processing, and to which a higher degree of "psychological reality" may be ascribed. The rule is a likely candidate for incorporation in natural language parsers.

Acknowledgements

I would like to thank my dissertation advisors Susumu Kuno and Jaklin Kornfilt for discussion of Binding, George Heidorn and Karen Jensen, who read an earlier version of this paper, and Dr. Edward Stabler at Syracuse University for long time support of my dissertation project. Prof. Kuno provided the initial reference to the manuscript by Barss, and kindly made it available.

References

Abney, Steven, 1986. "Licensing and Parsing." *manuscript*, MIT.[3]

Abney, Steven and Jennifer Cole, 1986. "A Government-binding Parser." *manuscript*, MIT.[3]

Barss, Andrew, 1983. "Chain Binding." *manuscript*, MIT.[3].

Barton, Edward, 1984. "Towards a Principled-based Parser." A.I. Memo No. 788, MIT.

Chomsky, Noam, 1981. *Lectures on Government and Binding*. Foris Publications, Dordrecht.

Chomsky, Noam, 1986. *Knowledge of Language*. Praeger, New York.

Correa, Nelson, 1987. "An Attribute Grammar Implementation of Government-binding Theory." *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, Stanford, California.

Hobbs, Jerry, 1978. "Resolving Pronoun References." *Lingua* 44.

Kashket, Michael, 1986. "Parsing a Free-word Order Language." *Proceedings of the 24th Annual Meeting of the Association for Computational Linguistics*, New York.

Kuhns, Robert, 1986. "A PROLOG Implementation of Government-binding Theory." *Proceedings of the 1986 Annual Conference of the European Chapter of the Association for Computational Linguistics*,

Kuno, Susumu, 1987. *Functional Syntax: Anaphora, Discourse and Empathy*. The University of Chicago Press, Chicago, Illinos.

Sharp, Randall, 1985. *A Model of Grammar Based on Principles of Government and Binding*. M.S. Thesis, Department of Computer Science, University of British Columbia. Vancouver, Canada.

Sells, Peter, 1985. *Lectures on Contemporary Linguistic Theories*. Chicago University Press, Chicago, Illinois.

Sonnenschein, Michael, 1985. "Global Storage Cells for Attributes in an Attribute Grammar." *Acta Informatica* 22.

van Riemsdijk, Henk and Edwin Williams, 1986. *An Introduction to the Theory of Grammar*. The MIT Press, Cambridge, Massachussets.

Waite, William M. and Gerhard Goos, 1984. *Compiler Construction*. Springer-Verlag, New York.

Wehrli, Eric, 1984. "A Government-binding Parser for French." Institut Pour les Etudes Semantiques et Cognitives, Universite de Geneve. Working Paper No. 48.

[3] May be obtained on request from the author.

Appendix: The Binding rule

(Rules for *AAS* and *PAS* computation)

Clause and sentence rules:

a. **Z** → **CP**
attribution:
CP.*AAS* ← [ ]
CP.*PAS* ← [ ]

b. **CP** → (**NP**) **CB**
attribution:
CB.*AAS* ← if **CP**.*tense* = + then [ ] else **CP**.*AAS*
CB.*PAS* ← if **CP**.*tense* = +
then **CP**.*AAS* $\bigcup$ **CP**.*PAS* else **CP**.*PAS*
CP.PAS_S ← set-diff(**CB**.PAS_S, **CP**.*AAS*)

c. **CB** → **C** **IP**
attribution:
IP.*AAS* ← **CB**.*AAS*
IP.*PAS* ← **CB**.*PAS*
CB.PAS_S ← **IP**.PAS_S

d. **IP** → **NP** **IB**
attribution:
NP.*AAS* ← **IP**.*AAS*
NP.*PAS* ← [< **NP**.*node*, **NP**.*AGR* > | **IP**.*PAS*]
IB.*AAS* ← [< **NP**.*RefIndex*, **NP**.*AGR* > | **IP**.*AAS*]
IB.*PAS* ← **NP**.PAS_S
IP.PAS_S ←
[< **NP**.*RefIndex*, **NP**.*AGR* > | **IB**.PAS_S]

e. **IB** → **I** **VP**
attribution:
VP.*AAS* ← **IB**.*AAS*
VP.*PAS* ← **IB**.*PAS*
IB.PAS_S ← **VP**.PAS_S

Verb-phrase rules:

a. **VP** → ... **VB** ...
attribution:
VB.*AAS* ← **VP**.*AAS*
VB.*PAS* ← **VP**.*PAS*
VP.PAS_S ← **VB**.PAS_S

b. **VB** → **V**
attribution:
VB.PAS_S ← **VB**.*PAS*

c. **VB** → **V** **XP**, for **XP** = **NP**, **CP**
attribution:
XP.*AAS* ← **VB**.*AAS*
XP.*PAS* ← **VB**.*PAS*
VB.PAS_S ← if **XP** = **NP**
then [< **NP**.*RefIndex*, **NP**.*AGR* >]
else **XP**.PAS_S

d. **VB** → **V** **NP** **XP**, for **XP** = **PP**, **CP**
attribution:
NP.*AAS* ← **VB**.*AAS*
NP.*PAS* ← **VB**.*PAS*
XP.*AAS* ←
[< **NP**.*RefIndex*, **NP**.*AGR* > | **VB**.*AAS*]
XP.*PAS* ← **NP**.PAS_S
VB.PAS_S ← **XP**.PAS_S

Noun-phrase rules:

a. **NP** → (**Det**) **NB**
attribution:
NB.*AAS* ← **NP**.*AAS*
NB.*PAS* ← tail(**NP**.*PAS*)
NP.PAS_S ← **NB**.PAS_S

b. $\mathbf{NP}_1$ → $\mathbf{NP}_2$ **NB**
attribution:
$\mathbf{NP}_2$.*AAS* ← $\mathbf{NP}_1$.*AAS*
$\mathbf{NP}_2$.*PAS* ←
[< $\mathbf{NP}_2$.*node*, $\mathbf{NP}_2$.*AGR* > | tail($\mathbf{NP}_1$.*PAS*)]
NB.*AAS* ← [ ]
NB.*PAS* ← $\mathbf{NP}_2$.PAS_S
$\mathbf{NP}_1$.PAS_S ←
[< $\mathbf{NP}_2$.*RefIndex*, $\mathbf{NP}_2$.*AGR* > | $\mathbf{NP}_2$.PAS_S]

c. **NB** → **N**
attribution:
NB.PAS_S ← **NB**.*PAS*

d. **NB** → **N** **XP**, for **XP** = **PP** or **CP**
attribution:
XP.*AAS* ← **NB**.*AAS*
XP.*PAS* ← **NB**.*PAS*
NB.PAS_S ← **XP**.PAS_S