Paralleling recent shifts within Grammatical Theory away from rule-based and toward principle-based systems, there has arisen widespread interest in the possibility of similar refocusing with respect to natural language processing (cf. Abney (1988), Berwick & Weinberg (1984), Clark (1987), Fong (1990), Gibson (1987), Johnson (1988), Kashket (1988), Pritchett (1987, 1988, 1990, in press, forthcoming), Stabler (1989), among others).

Fundamental to principle-based as opposed to rule-based models of parsing is the hypothesis that the Parser itself adheres to a version of the Projection Principle which maintains that each level of syntactic representation is a projection of the lexical properties of heads. With respect to parsing, the PP implies that a node cannot be projected before the occurrence of its head since the relevant features which determine its categorical identity and license its own and its arguments' attachment are theretofore undetermined. This paper describes an ongoing project in the implementation of an object-oriented (Smalltalk-80) Government and Binding parser which adheres to the strong competence hypothesis that principles of Universal Grammar are employed directly in parsing. Specifically, the parse operates by projecting phrasal structure as determined by the lexical properties of heads and licensing local attachments which maximally satisfy on-line principles of Universal Grammar at every point during a parse. Though this model was originally motivated with regard to its psychological plausibility, in this paper we focus primarily on issues of implementation (see Pritchett: op. cit. for a more detailed discussion of the psycholinguistic issues).

In the implemented parser, the following new Object subclasses are defined:

- Object
- PrincipleBasedParser
- Lexicon
- LexicalItem
- Node
  - EmptyNode
  - FullNode
  - DoubleBarNode
  - SingleBarNode
  - ZeroBarNode
- Chain
- LicensingRelation
  - ThetaRoleAssignment
  - CaseAssignment
  - SpecHeadAgreement
  - XPSelection

An instance of PrincipleBasedParser (henceforth simply the parser) itself acts as the buffer for tree structures. The parse of a string succeeds if at the end of input, there is exactly one tree in the parser and all grammatical principles are satisfied for every Node in that tree.

The syntactic structures actually created and manipulated by the parser are subinstances of the class Node. Nodes accord with a binary-branching version of X' Theory and each Node exists as an element of a maximal projection: \[ xP \ [yP \ [x' \ [X \ [zP \]]]] \]. Phrase Structure constraints on the linear order of Nodes is specified in the pool variable, HeadParameter; in this note we assume the English configuration. The specifier and complement positions themselves are either fully specified maximal projections or instances of the special class EmptyNode. Nodes respond in the expected fashion to a range of messages concerning configurational structure, such as c-commands, m-commands, governs, mother, sister, etc.

Each Node may be associated (coindexed) with other Nodes via an instance of the class Chain, a subclass of SortedCollection, where Node \( \alpha \) precedes Node \( \beta \) in an instance of Chain iff \( \alpha \) c-commands \( \beta \). Given this definition, two Nodes may cooccur within the same Chain only if they are contained in the same tree structure. Every Node has an associated Chain, though in the default case a Node is the Chain's singleton member. For a Node to be globally licit, all relevant grammatical principles must be satisfied with respect to its Chain.

Subinstances of the abstract class LicensingRelation represent the actual principles of Grammar which license Nodes, such as the \( \theta \)-criterion and Case Theory. Each Node keeps track of all licensing relations in which it participates via the instance variables licensorRelations and licenseeRelations.

As an illustration of the model as discussed so far, consider how a simple sentence, *Vampires were seen*, is processed. This sentence is fed to the processor one PF-word at a time by the procedure:

```
| parser |
parser newWord: 'vampires'.
parser newWord: 'were'.
parser newWord: 'seen'.
parser output
```

First a parser with an English lexicon and English parameter settings (e.g. the
possible attachments of one into positions in the
will have an effect. This method examines the
last two trees in the parser and attempts all
time, the message
a result, a changed message is sent, and the
and its maximal projection, an I_P, added: \[yp \{e\}\]
encountering.
projected as relevant heads have not been
role!ess. No higher structure, including IP, is
parser reaches a steady state with no licenser
available and the NP unavoidably left locally
parser contains only the NP
represented as
this fashion that grammatical principles are
fundamental to the parser's operation as it is in
any of the three messages sent by update:
actively determine attachments. Furthermore, if
attachLastTwoTrees wherein the 0-
buildChainsInLastTree
Next, and crucial to the on-line application of
grammatical principles, the changed message
sent, indicating that the parser's contents have
altered and signaling that the reapplication of
grammatical principles is relevant. Whenever
the parser receives the message changed, it is
automatically sent the message update: by
the Smalltalk-80™ system, which is defined as
follows:
\[\text{update: } \text{dummy}\]
self attachLastTwoTrees.
self expandLastTree.
self buildChainsInLastTree
The most important message in this method is
attachLastTwoTrees wherein the 0-
criterion and Case Theory (among others)
actively determine attachments. Furthermore, if
any of the three messages sent by update:
itself makes changes to the parser's contents, it
too will in turn send changed messages to the
parser, again triggering the sending of
update:. In this way, the parser manipulates
its contents continually until a local steady state
is reached with all grammatical principles
maximally satisfied. Hence, this
\[\text{changed/update: } \text{message sequence is fundamental to the parser's operation as it is in}
this fashion that grammatical principles are
represented as on-line in the system.
Returning to the example, none of the
messages within update has any effect when the
parser contains only the NP vampires, and the
parser reaches a steady state with no licenser
available and the NP unavoidably left locally
roleless. No higher structure, including IP, is
projected as relevant heads have not been encountered.
Next, the word 'were' is sent to the parser,
and its maximal projection, an I_P, added: \[\text{np} \{e\}\]
\[\{n' \{v\} \text{vampires} \{e\}\}\], \[\{p \{e\}\} [r' \{r \} \text{were} \{e\}\}\]. As a
result, a changed message is sent, and the
update: message's method is executed. This
time, the message attachLastTwoTrees
will have an effect. This method examines the
last two trees in the parser and attempts all
possible attachments of one into positions in the
other. The method then chooses the attachment
which is licensed to the highest degree. An
attachment is defined as licensed to degree n if by
making the attachment, n different licensing
relations will be newly discharged. (See
Pritchett cited above for psycholinguistic
justification of this selection procedure as well as
some alternative approaches to the notion
'maximally licensed'.) Given adjacency
requirements, two attachments are considered
in this example: the attachment of the IP into
the complement of NP and the attachment of the NP
into the specifier of IP. Only the second result
in the discharge of a licensing relation, namely
the case assigned by I under government. Hence,
this attachment is chosen, so that the parser now
contains only one element: \[\{ip \{v\} \text{vampires} \{r' \{r \} \text{were} \{e\}\}\]\]. The requirements of Case Theory are
satisfied to the maximum degree possible in the
local string—both with respect to the target NP
which requires these features and the head which
discharge them.
Next the method expandLastTree is
sent. In this case, the method causes the IP to
expand into a CP. As a result, the contents of the
parser becomes: \[\{cp \{e\} \{c' \{c \} \text{tip} \{\text{vampires} \{i' \{i \} \text{were} \{e\}\}\}\}\]. The last message in the
method for update:, buildChainsInLastTree is sent but has
no effect. Since the first two messages sent in
update: caused changes to the contents of the
parser, they both send changed messages, with
the result that update: is executed again.
However, none of the three messages in
update: has any effect this time around as
there is a single tree in the parser, and a local
steady state has been reached, with all structure
recognized to the maximum degree possible with
respect to UG principles.
Finally, the word seen is sent to the parser.
Seen is identified as a passive participle which,
as a lexical property, assigns an internal 0-role
but no Case. In the VP which is projected, the
V acts as the licenser in a licensing relation,
namely an instance of ThetaRoleAssignment
under government. Again, since the parser's
contents have changed, update: is sent, invoking
attachLastTwoTrees forcing the VP attachment as a complement of INTL: \[\{\text{cp} \{e\}\]
\[\{c' \{c \} \text{tip} \{\text{vampires} \{i' \{i \} \text{were} \{e\}\}\}\}\]. (This is carried out by means of
an instance of XPSelct: a subclass of
LicensingRelation relevant to functional heads.)
The message expandLastTree is sent but
has no effect. Next, the message
buildChainsInLastTree is sent. The
method associated with this message attempts to
associate Nodes and EmptyNodes (through
Chain building) in order to more fully satisfy
Case Theory and the 0-criterion. In this example
the empty complement of VP is added to the
Chain associated with the NP vampires and the
V's 0-role assigned to this empty position. As a
result, the Chain possesses both a 0-role and Case position since its head (the NP) is in a Case position and its tail (the empty node) in 0-position. The contents of the parser are now: 
\[ cp[e] [c c] [w [np vampires]], [t [i were] [[vp [e] [v' [v seen] [e]]]]]. \]
Input terminates and the message output is sent to the parser, which checks that all mandatory licensing relations have been fulfilled and returns the final structure.

At this point, we will briefly discuss how the head-driven principle-based model here predicts certain psycholinguistic facts. This discussion will be schematic and the reader is referred to Pritchett (op. cit.). Consider for example, well-known garden-path effects of the sort found in an example like, *After John drank the water evaporated.* Informally, the problem for the human parser in such examples is that the post verbal NP is prematurely construed as the complement of the verb, which causes difficulty when it must be reinterpreted as a subject. In terms of our implementation, once the parser has been sent the words up through *water*, it contains the following tree: 
\[ cp[e] [c c] [w [np vampires]], [t [i were] [[vp [e] [v' [v seen] [e]]]]]. \] Input terminates and the message output is sent to the parser, which checks that all mandatory licensing relations have been fulfilled and returns the final structure.

The architecture of the parser also arguably provides a processing, as opposed to a grammatical, account of effects deriving from Huang’s (1982) Constraint on Extraction Domains which prohibits movement from within positions which are not properly governed. For example, it proscribes examples such as, *Who do pictures of e i bother John.* To give just one example, according to our parsing-theoretic account, extraction from within subjects is impossible since there is simply no local option of forming the requisite Chain at the time the subject constituent is being parsed, given the fact that the parser is strictly head driven. Recall that a sentence (IP) is not projected until either an inflectional element or a verb possessing inflectional features is processed. Before a category is projected, it is impossible to license its specifier, the subject. Consequently, in the previous example, after the word of is processed, the parser contains the following two unIntegrated Nodes: 
\[ cp [np who] [c c do] [e]], and [np [e] [v [pictures] [vp [o of] [e]]]]. \] These two Nodes cannot be locally integrated before the projection of IP and hence the requisite Chain cannot be formed between the wh-word in SPEC-CP into the NP pictures of as the two phrases are not locally constituents of the same parse tree. In other words, the NP is not locally a subject at that point during the parse but is rather unattached. See Pritchett (to appear) for details. Thus our implementation begins to provide an existence proof that a parser driven by the Projection Principle and the on-line application of global grammatical principles is both psychologically and implementationally realistic.

**References:**

Abney, Steven. 1988. *On the notions GB-parser and psychological reality.* in The MIT Parsing Volume, 1987-88.

Berwick, Robert & Amy Weinberg. 1984. *The Grammatical Basis of Linguistic Performance.* Cambridge: MIT.

Clark, Robin 1987. *Rules and Parsing.* paper presented at MIT.

Fong, Sandway 1990. *The computational implementation of principle-based parsers.* in The MIT Parsing Volume, 1989-90.

Gibson, Edward 1987. *Garden path effects in a parser with parallel architecture.* paper presented at the Eastern States Conference on Linguistics.

Huang C.-T. James 1982. *Logical Relations in Chinese and the Theory of Grammar.* MIT doctoral dissertation.

Johnson, Mark. 1988. *Parsing as deduction: The use of knowledge of Language.* in The MIT Parsing Volume, 1987-88.

Kashket, Michael. 1988. *Parsing Warlpiri, a free-word order language.* in The MIT Parsing Volume, 1987-88.

Pritchett, Bradley (forthcoming). *Principle-based Parsing and Processing Breakdown.* (title tentative), University of Chicago Press.

Pritchett, Bradley (in press). *Head-driven parsing and the CED.*

Pritchett, Bradley. 1990. *Subjacency in a principle-based parser.* in The MIT Parsing Volume, 1988-89.

Pritchett, Bradley. 1988. *Garden Path Phenomena and the Grammatical Basis of Language Processing.* LANGUAGE 64.3.

Stabler, Edward (forthcoming). *The Logical Approach to Syntax.* MIT Press.