RESTRICTION LANGUAGE MANUAL

Linguistic String Project

PART 1:
Elementary view of the English Grammar
and The Restriction Language

PART 2:
AN INTRODUCTION TO THE TRANSFORMATIONAL COMPONENT OF
THE LSP ENGLISH GRAMMAR

RESTRICTION LANGUAGE MANUAL

Linguistic String Project

PART 1:

Elementary view of the English grammar
and The Restriction Language

1. STRING GRAMMARS

Our approach to the recognition of the structure of natural language sentences is based on linguistic string theory. This theory sets forth, in terms of particular syntactic categories (noun, tensed verb, etc.) a set of elementary strings and rules for combining the elementary strings to form sentence strings.

The simplest sentences consist of just one elementary string, called a center string. Examples of center strings are noun tensed-verb, such as "Tapes stretch.", and noun tensed-verb noun, such as "Users cause problems." Any sentence string may be made into a more complicated sentence string by inserting an adjunct string to the left or right of an element of some elementary string of the sentence. For example, "Programmers at our installation write lengthy code." is built up by adjoining "at our installation" to the right of "programmers" and "lengthy" to the left of

"code" in the center string "programmers write code." Sentences may also be augmented by the insertion of a conjunct string, such as "and debug" in "Programmers at our installation write and debug lengthy code." Finally, string theory allows an element of a string to be replaced by a replacement string. One example of this is the replacement of noun by what noun tensed-verb to form the sentence "What linguists do is puzzling."

Each word of the language is assigned one or more word categories on the basis of its grammatical properties. The assignment is based on the word's use in the language as a whole, not its use in a particular sentence or text. For example, "users" and "problems" would each be classed as a noun, while "cause" would be assigned the three categories tensed verb, untensed verb, and noun. Every sequence of words is thereby associated with one or more sequences of word categories. Linguistic string theory claims that each sentence of the language has at least one sequence of word categories which is a sentence string, i.e., which can be built up from a center string by adjunction, conjunction, and replacement.

However, not every combination of words drawn from the appropriate categories and inserted into a sentence string forms a valid sentence. Sometimes only words with related grammatical properties are acceptable in the same string, or in adjoined string. For example, one of the sequences of word categories associated with "Tape stretch." is noun tensed-verb, which is a sentence string; this sentence is

ungrammatical, however, because a singular noun has been combined with a plural tensed-verb. To record these properties, we add the subcategory (or attribute) singular to the category noun in the definition of "tape" and the subcategory plural to the category tensed-verb in the definition of "stretch". We then incorporate into the grammar a restriction on the center string noun tensed-verb, to check for number agreement between noun and verb.

A string grammar thus consists of four major components:

1. A set of major categories for the words of the language, where each category may have associated with it a set of subcategories;

2. A lexicon (or word-dictionary) giving for each word of the language its major category assignment(s), and within each major category its subcategory assignment(s);

3. A list of the elementary strings of the language, where each string consists of a particular sequence of the major word-categories. The strings are arranged into subsets according to whether they are center strings, adjunct strings, conjunct strings or replacement strings. The adjunct and conjunct string sets are further subdivided according to the position they occupy within the string they adjoin or conjoin: to the left of a noun, to the right of a verb, etc. This subsetting is equivalent to rules of combination.

4. A set of restrictions on well-formed combinations of the elementary strings, giving further detailed well-

formedness requirements. These usually refer to the sub-categories of the sentence words which constitute occurrences of the elementary strings in the sentence. An example was cited above wherein a restriction would require that the tensed verb of a center string agree in number with its subject noun.

To illustrate the principles of a string grammar we develop here a small string grammar which is sufficient to analyze the example sentences given so far. Let the major word categories be:

CATEGORY	EXAMPLES
N noun	tapes, stretch, users, cause, problems, programmers, code, installation, linguists, puzzle
TV tensed verb	is, are, does, do, tapes, stretch, cause, code, puzzle, write
V untensed (infinitive) verb*	be, do, tape, stretch, cause, code, puzzle, write
P preposition	at
T article	the, a, our**
ADJ adjective	lengthy
VING present participle, i.e. verb with -ing suffix	being, doing, taping, stretching, causing, coding, writing, puzzling
VEN past participle, usually V-en or V-ed	been, done, taped, stretched, caused, coded, puzzled, written

*This category was not used in the example sentences, but with its addition the lexical entries for individual words will be more complete.

**In the LSP English grammar adjectival possessive pronouns my, your, his, her, its, our, their are classed in the category T.

Let the subcategories (also called attributes) be SINGULAR and PLURAL*; these apply only to N and TV.

Some sample lexical entries would be:

```
PROGRAMMERS
     N: (PLURAL).

WRITE
     TV: (PLURAL), V.

CODE
     N: (SINGULAR), TV: (PLURAL), V.

LENGTHY
     ADJ.

AT
     P.

OUR
     T .

INSTALLATION
     N: (SINGULAR).
```

Here, major category assignments of a word are separated by commas. If the word is a member of a subclass of a major category, then that subclass follows the major category symbol, separated from it by a colon and enclosed in parentheses.

---

*SINGULAR and PLURAL are defined in the LSP English grammar as follows: A noun is SINGULAR if it can occur in the sentence environment (a) "This _____ TV" and cannot occur in the environment (b) "These _____ TV". A noun is PLURAL if it can occur in (b) and not in (a). A tensed verb TV is SINGULAR if it occurs with a SINGULAR N as subject; it is PLURAL if it occurs with a PLURAL N as subject. Note that nouns like fish (this fish, these fish) are neither SINGULAR nor PLURAL.

The elementary strings of this small grammar are:

CENTER STRINGS	EXAMPLES
N TV	"Tapes stretch,"
N TV N	"Users cause problems," "Programmers write code"
N TV VING	"Programmers are puzzling"

ADJUNCT STRINGS

Left adjuncts of N	
T	"our"
ADJ	"lengthy"
Right adjuncts of N	
P N	"at installations"

CONJUNCT STRINGS

Conjunct of TV	
'AND' TV	"and debug"

N-REPLACEMENT STRINGS

'WHAT' N TV	" what linguists do"

Each of the example sentences (and many more) can now be represented by a decomposition of the sentence into its component elementary strings. For the sentence to be well-formed according to the string grammar, it must contain the occurrence of one (and only one) center string, and if there are left or right adjunct string occurrences, then each must occur at the proper point in the string it adjoins, that is, to the left or right of the appropriate category (N in this case). Similarly, if there is an occurrence of a conjunct string, then it must appear to the right of stated elements in the string which it conjoins, (in this case to the right of TV in the center string). The N-replacement string should be found only in the position(s) where N appears in other elementary strings.

We can represent the decomposition of a sentence into its elementary strings by writing each component string on a separate numbered line, starting with the center string on the first line. We write the name of each string element (i.e., the word-category) above the word which satisfies it in the sentence. We indicate where each left/right adjunct string string occurs in the string it adjoins (its host string) by writing to the left/right of the word it adjoins in the host string the number of the line on which the adjunct string is written. Thus, the decomposition of sentence 1 becomes:

SENTENCE 1: Programmers at our installation write lengthy code.

1. center = N: Programmers 2. TV: write N: 4. code

2. right adjunct of N = P: at N: 3. installation

3. left adjunct of N = T: our

4. left adjunct of N = ADJ: lengthy

To indicate that a replacement string occurs in place of a noun, we write in place of the noun the number of the line on which the replacement string is written. Thus the decomposition of the sentence "What linguists do is puzzling" will

appear as follows:

SENTENCE 2: What linguists do is puzzling.

1. center =		N	TV	VING	
		2.	is	puzzling	
2. N-replacement string =	'WHAT'	N	TV		
	what	linguists	do		

Without the fourth component, the restrictions, it is clear that this small grammar describes many non-sentences and many nonsense sentences which conform to the same sequences of word-categories as the well-formed sentences described by the grammar. Thus, with our small lexicon, we can construct "Tapes is puzzling" conforming to the center string N TV VING, but violating the restriction, mentioned above, that a tensed verb must agree in number with its subject. We may also construct "Tapes code puzzling" conforming to the same category sequence, since no restriction is placed on the type of verb which occurs preceding VING in the center string N TV VING. A similar situation arises with regard to the center string N TV. We might compose the (supposed) sentences "Tapes are," "Programmers cause," which fail to be sentences because the verbs "are" and "cause" are not of the type which can occur intransitively. The grammar thus requires a restriction on the center string N TV stating that TV must be a verb of the type which can

occur without an object. Just the contrary restriction is required in the N-replacement string 'what' N TV. Here the verb must be of the type which can occur transitively, that is, with a noun object, even though the string does not call for the noun object following the verb. This anomaly is explained by the function of the word "what." A further discussion of this type of construction is deferred until Part 2.

With our small lexicon and restrictionless grammar we may also construct sentences of the type "Installation tapes linguists" which may cryptically convey meaning, but violates acceptable English syntax in omitting an article before the type of noun, called a countable noun ("installation") which in the singular specifically requires a preceding article. By adding the subcategory NCOUNT to our list of noun subcategories, a restriction to this effect could be added to the grammar. Lastly, sentences which border on the nonsensical or ungrammatical (depending on how "grammatical" is defined by the grammar writer) can be constructed, such as "Tapes cause programmers." Here "cause" is not the kind of verb which takes "human" nouns as its object. We might find it possible and desirable, when working with texts in a specific subject matter area, to restrict all transitive verbs with regard to the type of noun which is acceptable as their object, and possibly, in addition, to restrict all verbs with regard to the type of noun which is acceptable as their subject. The subcategory NHUMAN would be useful in such a restriction.

All of these cases suggest restrictions which should or could be a part of this small grammar. However, we postpone until later the exact formulation of these restrictions. We want first to arrange the strings of the grammar in a form which is convenient for computation; it will then be possible to formulate restrictions so that they can easily be translated into restriction tests which can be carried out by a computer in the course of analyzing an input sentence. The execution of the restriction tests eliminates wrong parses which would be obtained by the computer program operating with the context-free grammar alone (i.e. the grammar without the restrictions).

2. THE BNF COMPONENT OF THE GRAMMAR

The small string grammar presented informally above can be expanded to include all the major English sentence types. The restriction component will be correspondingly increased. In adding strings to the grammar, however, one soon finds that restricting oneself to elementary strings composed entirely of word-categories leads to a great proliferation of strings which are partially similar, and which have to be repeated in various parts of the grammar. For example, simply to add the word category PRO (pronoun) as an alternative to N in the various positions where N occurs in the small grammar developed above increases the number of strings by

50%, from 8 to 12.

We therefore choose to define a single "super string" for each set of strings whose members can be segmented into corresponding positions. We name the positions in the super string and we define for each such named position the set of alternative values which, when substituted for the named position, yield the original set of strings. Thus, in place of center strings N TV N, PRO TV N, N TV PRO, and PRO TV PRO, we could have one string CENTER = SUBJECT TV OBJECT where both SUBJECT and OBJECT have the alternative values N or PRO.

A convenient formalism for specifying a grammar which has been formulated in terms of these "superstrings" is Backus-Naur Form (BNF). BNF is a format for writing context-free grammars which was originally developed to describe the syntax of programming languages.

A BNF grammar is a set of definitions (also called productions), each of which defines one syntactic type as a sequence of syntactic types or literals. For example, consider the following grammar for writing letters:

```
<LETTER>     ::=<SALUTATION><BODY><CLOSE>
<SALUTATION>::=<FRIENDLY>|<FORMAL>
<FRIENDLY>   ::=DEAR<NAME>
<NAME>       ::=JOHN|MARY|CON EDISON
<FORMAL>     ::=TO WHOM IT MAY CONCERN
<BODY>       ::=I LOVE YOU|I HATE YOU|
                 PAY UP OR WE WILL SHUT OFF YOUR<UTILITY>
<UTILITY>    ::=WATER|ELECTRICITY|SUNSHINE
<CLOSE>      ::=SINCERELY,<NAME>
```

The syntactic types in a BNF grammar (also called "non-terminal symbols") are enclosed in pointed brackets. The words which are not enclosed in such brackets, such as MARY and ELECTRICITY, are literals (or "terminal symbols"); these are words which can appear in a letter. The first line of the grammar states that a LETTER is a SALUTATION followed by a BODY followed by a CLOSE; SALUTATION, BODY, and CLOSE are called the elements of LETTER. The second line states that a SALUTATION is either a FRIENDLY or a FORMAL; the vertical stroke separates the different options (alternative definitions)

of SALUTATION. The third line of the grammar states that FRIENDLY is the word "DEAR" followed by a NAME.

The syntactic type LETTER is distinguished by the fact that it does not appear on the right side of any BNF definition. We call LETTER the root symbol of our grammar. If we wanted to generate a few letters for our friends, we would start with the root symbol and use the BNF definitions as rewriting rules. The first definition would rewrite

<LETTER>

as

→<SALUTATION><BODY><CLOSE>

We would then choose one option of the second definition to rewrite SALUTATION:

→<FRIENDLY><BODY><CLOSE>

Continuing in this way

→ DEAR<NAME><BODY><CLOSE>

→ DEAR CON EDISON<BODY><CLOSE>

→ DEAR CON EDISON I HATE YOU<CLOSE>

→ DEAR CON EDISON I HATE YOU SINCERELY,<NAME>

we obtain the letter

DEAR CON EDISON I HATE YOU SINCERELY, JOHN

By repeating this process and choosing all possible combinations of options we can generate the set of all possible LETTERs; this set is the language defined by this grammar.

If our circle of acquaintances grew, we could enlarge the definition for NAME accordingly. Alternatively, we could set up a lexicon for use with our grammar, listing all the words with the word category NAME, something like this

WORD	WORD-CATEGORY
JOHN	NAME
MARY	NAME
CON EDISON	NAME
ABEL	NAME
CAIN	NAME
FRITZ	NAME

This would replace the definition for NAME in our original grammar. To indicate that NAME is now a word category, whose possible values are determined by the lexicon rather than another BNF definition, we put an asterisk to the left of NAME in the BNF grammar:

```
<FRIENDLY> ::=  DEAR <*NAME>
<CLOSE>    ::=  SINCERELY, <*NAME>
```

Finally, suppose we wanted to add anonymous letters to our letter language. Specifically, we want to allow a CLOSE to be the null or empty string, i.e., absolutely nothing. We

indicate this by adding an option to CLOSE with the element <*NULL>:

<CLOSE> ::= SINCERELY,<*NAME> | <*NULL>.

We could then complete the sentence generated above somewhat differently, with

DEAR CON EDISON I HATE YOU <CLOSE>
→ DEAR CON EDISON I HATE YOU

In a BNF grammar of English, the syntactic types are the grammatical constructs, e.g., <CENTER>, the lexical types are the major word categories (also called "atomics"), e.g., <*N>, and the literals are punctuation marks and grammatical constants, like the infinitive marker "to."

To illustrate how a string grammar can be written in BNF we will first rewrite the simple string grammar of the preceding section in this form. In order to do this we must introduce a few notational conventions. Adjunct strings, in a string grammar, are optional insertions into other strings. In BNF, this can be expressed by placing a <*NULL> as an option in the definition of each adjunct set. To indicate the position at which adjuncts of a given type can be inserted into another string, say to the left of a word-category <*X> or to

its right, we adopt a standard type of definition, having the form

<LXR> ::= <LX><*X><RX>

where LX and RX are respectively, the sets of left and right adjunct strings of the word category X, and have the form

<LX> ::= <LX1>|<LX2>|...|<*NULL>.
<RX> ::= <RX1>|<RX2>|...|<*NULL>.

Here LXi and RXi stand for the different adjunct strings in the adjunct set definitions. We add these definitions to the grammar and then write LXR in place of *X wherever *X appears in a definition of an elementary string.

With these notational conventions, the string grammar of Section 1 can be written in BNF as follows.*

*BNF

<SENTENCE> ::= <CENTER> ' . ' .

<CENTER> ::= <NSTG><*TV>(<LNR>|<*VING>) .

---

*Henceforth, until Part 3, conjunctional strings will not be included in the grammar.

Because the small string grammar has so few adjunct sets, only *N here has been replaced by LNR in accord with the above discussion. The atomic categories *TV, *VING, *T, *ADJ, *P appear here without adjuncts. In BNF definitions intended for use in the LSP system, literals containing characters other than letters must be enclosed in '' . Every BNF definition must end in a period.

```
<NSTG>    ::= <LNR>|<NREP> .
<LNR>     ::= <LN><*N><RN> .
<LN>      ::= (<*T>|<*NULL>)(<*ADJ>|<*NULL>)* .
<RN>      ::= <PN>|<*NULL> .
<PN>      ::= <*P><LNR> .
<NREP>    ::=  WHAT  <LNR><*TV > .
```

Notice that the BNF string grammar no longer "looks like" a string grammar. We have introduced auxiliary definitions which are neither elementary string definitions nor adjunct set definitions, but rather collections of alternative values for particular string-element positions. This was a necessary consequence of the decision to group partially similar elementary strings into a single "super string" definition. We call such definitions positional variants; an example of such a definition is the noun string NSTG occupying the first position of the CENTER super string.

Despite the seeming opacity of this form of the string grammar, its string character is maintained by separating the definitions into distinct types which are treated differently in subsequent operations, such as restrictions and the printing of parse trees. These types are listed in the grammar after

*The left adjuncts of N are ordered: "The three blue pens," ≠"The blue three pens," ≠"blue the three pens," etc. (The symbol "≠" indicates "is not well-formed.") It is therefore convenient to express them as a sequence of optional elements.

the BNF definitions:

```
*LISTS

    ATTRIBUTE = SINGULAR, PLURAL.

    TYPE STRING = CENTER, NREP, PN.

    TYPE ADJSET = LN, RN.

    TYPE LXR = LNR.
```

Thus we find a list of the definitions which correspond to our original elementary strings. This is the TYPE STRING list. The adjunct set definitions are named in the TYPE ADJSET list. The LXR type definitions are also listed since they play a distinguished role in the treatment of conjunctions. We also list the attributes which will appear in dictionary entries and restrictions.

This small BNF grammar, when augmented in several respects, will serve as a useful base for introducing the restriction component of the grammar. We will want to have more LXR type definitions to allow for richer adjunction, and we wish to introduce the important set SA of sentence adjunct strings. This group is so named because the strings can adjoin at various points within the host string, with the interpretation that the string modifies the whole host string. A prime example is the interjection of words like "however," "moreover" (called INT) and adverbs like "tomorrow", "generally," etc. (adverbs are called D). These may occur at any inter-element position in a center string, and before the first

element and after the last element. For example: "Generally he leaves early," "He generally leaves early," "He leaves generally (quite) early," "He leaves early generally."

We also find it convenient to name the positions in the center string, not surprisingly SUBJECT, VERB, OBJECT, and to name the ASSERTION center string as an option of CENTER so as to make room for other center strings, such as the question, imperative, and inverted forms, which will be defined later. Once the SUBJECT and OBJECT positions are defined, it is natural to add further options to their definitions, for example, pronouns (added within NSTG) and certain object nominalization strings, such as <THATS>, the string which is used to analyze sentences of the type "I know that he is here."* The N TV center string for intransitive verbs can be assimilated into the SUBJECT VERB OBJECT format of an ASSERTION center by defining a null object <*NULLOBJ> especially for this case. The center string N TV VING raises many linguistic questions (e.g., how to treat "be"), which need not concern us at this juncture. We therefore remove it from the small grammar and will take it up again at a later time.

---

*This string contains the string ASSERTION as one of its elements. For this reason ASSERTION appears on the list STGSEG (string segments) in the LSP grammar. This list includes any definition which is on the TYPE STRING list and also occurs as an element of another definition appearing on the TYPE STRING list.

In preparation for the development of a restriction component of the grammar we also wish to augment the ATTRIBUTE list of the grammar. We described earlier the attributes SINGULAR, PLURAL (fn. p. 5), NCOUNT and NHUMAN (p 10). To these we add the verb attribute OBJLIST (p. 32) which is used to specify which object strings can occur with a particular verb, and the pronoun subcategories NOMINATIVE and ACCUSATIVE. The NOMINATIVE pronouns will be "I", "he", "she", "we", "they", and the ACCUSATIVE pronouns will include "me", "him", "her", "us", "them."

With these modifications, the final form of our introductory BNF component of the string grammar has the form shown in Table 1. Here each definition is numbered and a correspondingly numbered word example is provided on the facing page when appropriate.