# A Doubtless Incomplete and Provisional Account of Loglan, as Practiced at the Loglan Institute, with sidelong glances at its History and its Sister Language

Randall Holmes, with homage to John Cowan

December 3, 2017

# Contents

# Chapter 1

# Introduction

The artificial language Loglan was conceived by the sociologist James Cooke Brown in 1955. His intention was to experimentally test the Sapir-Whorf hypothesis, that language influences the shape of human cognition. His not unambitious plan was to design a language, teach it to experimental subjects, and find out what it did to their minds. No such experiment has officially been carried out, but a language, or two languages, or even more than two, do exist.

His experimental design called for the language to have a "small" grammar (so as to be easy to learn), and extreme in some respect. He decided to make the language extremely logical. The sense(s) in which it is extremely logical can be misunderstood. It is logical in that its logical operations are (to some degree) parallel to those of the first-order predicate logic used by mathematicians. It is also logical in a quite different respect (also shared by mathematical notations): it is intended to be syntactically unambiguous and in fact parsable by computer. This is intended to go below the level of grammar, to the level of lexicography: Loglan text or speech is supposed to be parseable into words on purely phonetic grounds without ambiguity.

There are two senses in which Loglan is not logical. It is not semantically precise. Its root words (other than those explicitly representing logical operations) have meanings no more precise than those of English, and it supports metaphorical constructions which do not have precisely defined semantics and allow for plenty of poetic vagueness. This segues naturally to another sense in which Loglan is not "logical" (in which *logic* is not logical, in the opinion of this writer): it is not a language for Vulcans. It does not lack facility for emotional nuance. It is not even particularly weak at expressing emotional nuance by comparison with natural languages, in the opinion of this writer.

The intention of this book is not narrative. We intend to give an account of our current, provisional version of the Loglan language, while providing our readers with information about how our language differs from past versions (mostly, from the language described in the 1989 edition of *Loglan 1, a Logical Language*) and on how it differs from the much more popular sister language, Lojban. We do not intend to tell the story of the origins of Loglan or Lojban.

We treat the sister language (**le sorme lengu**), Lojban, as a linguistic fact (with a much larger footprint in the world than our own version, a fact which we cheerfully acknowledge). We think that our own language is better in some ways, but both languages are realizations of the same vision.

The form of the book is to some extent inspired by that of John Cowan's *The Complete Lojban Language*, a book which we have read and thoroughly enjoyed. We cannot be faulted for using it as a model to some extent, as Cowan himself has suggested this.

## 1.1 A Technical Aside on Grammar, Ambiguity, and the PEG formalism

Loglan was originally supposed to have a provably unambiguous BNF grammar. There was indeed a BNF grammar (in many versions) which was verified as unambiguous using computer tools. However, it did not achieve its aims. The difficulty was that all it analyzed was grammar. Lexicography and phonetics were not (and could not) be computer verified in this way, and we (and we think others) have established that 1989 Loglan had various demonstrable ambiguities caused by problems with lexicography. We are no more engaged in a polemic against 1989 Loglan than we are in a polemic against Lojban, so we are not going to create a catalogue of such problems here: some may be alluded to in passing when explaining the reasons for details of our grammar.

We have abandoned this strategy for computer parsing. We have instead adopted the PEG (Phrase Expression Grammar) program of Bryan Ford. A PEG is by definition unambiguous, as whenever a set of alternative forms is presented, an order on the various alternatives is given, and the first one which works is chosen. There are failures in design of a PEG analogous to those revealed in a BNF grammar when it is found to be ambiguous: they generally arise out of mistakes in the ordering of alternatives, and can be described as "unintended parses" rather than "ambiguities". We do not present automated tests for non-existence of unintended parses: we are not sure that there is a mathematical definition of an unintended parse, or that if such a definition were given that this would imply the ability to test reliably for the possibility of unintended parses.

Further, we have used the PEG formalism to parse Loglan text for phonetic and lexicographic correctness as well as grammatical correctness: all syntactical features of correct Loglan utterances are checked by the parser.

In this section we briefly describe the PEG formalism, since we use it in formalizing grammar rules. The reader is free to skip this until such time as they actually try to puzzle out the meaning of a grammatical rule stated in this notation.

A PEG consists of a list of rules of the form `identifier <- expression`, defining a class of strings-in-context named by `identifier` using the definition `expression`.

We say strings-in-context because the elements of the class defined by a PEG rule are not merely strings, but substrings sitting in a longer string possibly having additional relations to the longer string. Formally, a string-in-context is a substring of a larger string of a stated length starting at a stated position. Strings-in-context of length 0 are of interest and indeed of considerable use.

We now describe the building blocks of the definition bodies that can play the role `expression` in the above scheme.

`'<string>'` or `"<string>"` is an example of the simplest variety of string class. `'cat'` or `"cat"` is literally the string `cat` with no conditions on context: that is, it is the class of all strings-in-context of that exact length and containing those exact characters with no context conditions.

Another simple variety of class is a class of one-character strings defined by a list of permitted characters. `[abc]` for example is inhabited by strings-in-context of length one character, that character being either `a`, `b`, or `c`. Some characters can be represented by escape sequences, such as `\'` for the single quote or `\"` for the double quote (the need for these is somewhat diminished by the availability of two different sets of quotes). A notable class is `[]`, the empty class, expressed as a class of characters with no valid alternatives. Another notable related class is `.` which is the class of all single character strings-in-context. Shorthand notations such as `[a-dA-D]` for a single character which is one of the first four letters, either upper or lower case, should be noted.

If E and F are notations for classes, (E F) is the class of concatenations of a string-in-context belonging to E and a string-in-context belonging to F: we know what a concatenation is, surely, but we state explicitly that what we mean is that the string of class F starts just after the end of the string of class E, and the length of the string of class (E F) is the sum of the lengths of the component strings. Similarly (E1 E2 ... En) is the class of concatenations of $n$ strings, the $i$th string belonging to class Ei.

If E and F are notations for classes, (E / F) is the class of strings-in-context which are either of class E or which are of class F, without either having an initial segment of class E or being an initial segment of a class E string. Similarly (E1 / E2 / ... / En) is the class of strings which are of some class Ei without either being an initial segment of a class Ej string for $j < i$ or having an initial segment of class Ej for $j < i$. Note that we regard a string-in-context as an initial segment of itself. The idea is that when one reads a string of this class, one chooses the first class that applies.

A string of class E? is a string which is either of class E or of length 0 and not an initial segment of a class E string (recalling that though it is of length 0, it does have a position in the larger string which serves as its context). A string of class E* is a concatenation of class E strings (possibly of length 0) which is not followed by a class E string. A string of class E+ is a string of class E* which is not of length 0.

A string &E is a string of length 0 which is followed by a string of class E. A string !E is a string of length 0 which is not followed by a string of class E. These classes allow us to apply logical tests (within certain limits). Notice that `!.` expresses the concept end-of-string! For example, a string of class (&E F)

is a string-in-context of class F which has an initial segment of class E or is an initial segment of a string-in-context of class E. A string of class (E &F) is a string of class E which is followed by a string of class F [the string of class F is not included in it, merely present after it].

A computer program is readily written to read a string starting at a given position and find a string-in-context of a given class starting at that position.

Further, PEG rules can be recursive: the definition of a class may mention a class which has not yet been defined, and classes may be defined in terms of themselves or one another. It is quite possible to define a class recursively in such a way that parsing with it will lead into an infinite loop. The ML version of our implementation of PEG formalism contains a test for nontermination which has allowed us mostly to avoid such problems.

# Chapter 2

# A High Level Look at the Simple Loglan Sentence

In the language of first-order logic, atomic sentences are of the form $Px$, $Qxy$, or more generally $Rx_1 \ldots x_n$, where $P, Q, R$ are predicates (roughly analogous to verbs) and $x, y, x_i$ are arguments (roughly speaking, pronouns or noun phrases). First order logic (and Loglan) admit more complex sentences, but the analysis of the Loglan implementation of atomic sentences provides a nice starting point for understanding relationships between Loglan and both logic on the one hand and natural languages on the other.

The default form of the Loglan simple sentence is not $Px_1, \ldots x_n$, but $x_1\, P\, x_2 \ldots x_n$. In linguistic terms, Loglan is primarily an SVO language rather than a VSO language as its relationship to first order logic might have suggested it would be. It is worth noting that $x\, R\, y$ is often found in logic as a permitted variant notation for $Rxy$: predicate logic has its own SVO tendencies. Further (and with odd effects in our opinion) the fine grammatical structure of this sentence is $x_1\, [Px_2 \ldots x_n]$: it does not fall apart as one might suppose into three components at the same level, the first argument, followed by the predicate, followed by the list of subsequent arguments. Instead, it falls apart into a subject and a complex predicate with the string of further arguments as a component.

Simple examples of such sentences follow.

1. **Da mrenu**

   X is a man

2. **Ti blanu**

   This is blue

3. **Ti blanu ta**

   This is bluer than that

4. **Da donsu de di**

X gives Y to Z

Our translations in some cases reveal respects in which Loglan violates our elementary expectations, and in some cases conceal them.

We have kept things simple in that all our arguments are Loglan "pronouns" and all of our predicates are single words (Loglan arguments can be more complex "noun phrases", and Loglan predicates can be more complex "verb phrases").

In logic, each predicate has a numerical arity: an $n$-ary predicate always appears with exactly $n$ arguments. A Loglan predicate word has an arity as well. When you look up `blanu` in the dictionary, you will find an entry "X is bluer than Y". When you look up `donsu`, you will find that it means "X gives Y to Z". Unlike a predicate in logic, a Loglan predicate may appear without its complete list of arguments. The sentence `Ti blanu` means "This is bluer (than something)", which expresses the idea that it is at least to some extent blue. The sentence `Da donsu` means "X is a giver" or perhaps even "X is generous": its literal meaning is "X gives (something to someone)".

The translations of the pronouns unavoidably give away something about Loglan. Loglan has no pronouns along the lines of *he, she, it* which are classified by some semantic property of their referents. A Loglan pronoun is capable of referring to anything; the ways in which the pronouns **da**, **de**, **di** acquire reference are quite unfamiliar to the English speaking mind. The best we can do is suggest their meaning using mathematical variables: we will explain how they work later.

The pronouns **ti** and **ta** are quite analogous to the familiar English *this* and *that*.

The use of tense in the English translations is deceptive. The predicates in the given Loglan sentences do not have any specific tense: to specify a time, we would have to use additional words: we will see how to do this later.

A point to notice is that while we have suggested that the role of the predicate words in these sentences is analogous to that of the verb in an English sentence, the words actually used to translate them in the examples are of different parts of speech. In fact, Loglan predicates play the roles of nouns, verbs, adjectives, and adverbs. This will become evident as we reveal more of the grammatical constructions in Loglan and get past the very simple sentence construction we have allowed ourselves so far. It is already to some extent evident in the dictionary: if we look up *man* we will find a word **mrenu** one of whose meanings is the intransitive (i.e, 1-ary) verb ... *is a man*; if we look up *green* we will find a word **vegri** one of whose meanings is the transitive (i.e. 2-ary) verb ... *is greener than* .... When we discover noun-like constructions which let us turn **mrenu** into an argument meaning roughly "the man", it will be just as usable to construct arguments meaning roughly "the blue thing" or "the giver" from **blanu** and **donsu** respectively. The noun/verb/adjective distinction which is fundamental for the English speaker is nonexistent in Loglan.

## 2.1 Variations in order of subject, verb and object(s)

One thing we will do while we are at this very simple level is point out that the SVO order of Loglan is default, but not obligatory.

1. **Da gio de donsu di**

   X gives Y to Z

2. **Da gio de di donsu**

   X gives Y to Z

3. **Ga donsu de di ga da**

   X gives Y to Z

4. **Ga donsu ga da de di**

   X gives Y to Z

We can put some of the arguments which would normally appear after the predicate between the subject and the predicate, using the particle **gio** to separate the subject from these original arguments (**gio** was not used in 1989 Loglan and no analogous particle is used in Lojban; it is an initiative of ours to introduce this marker, based on the desire to avoid characteristic parsing accidents we discovered in trying to parse existing large Loglan texts).

We imposed a restriction on the use of `ga` to move the subject after the verb: we require that the second `ga` is followed either by the first argument alone or by all the arguments of the predicate. This condition was not imposed in 1989 Loglan. I do not know if this construction is provided in Lojban.

Thus, we have shown how to implement SOV, VOS, and VSO word orders (with further variations if there is more than one object). The construction which gives us OSV and OVS word orders is more complex, and will be introduced later.

A further form which is allowed is

1. **Ga donsu**

   It gives (Someone gave something to someone)

2. **Ga donsu de di**

   (Someone) gave Y to Z

If the subject is omitted without the marker `ga`, what results is an imperative:

1. **Donsu!**

   Give!

2. **Donsu de di!**

   Give Y to Z!

Lojban uses a special subject pronoun to mark the imperative forms. In this, Loglan is more like English than Lojban.

# Chapter 3

# Simple Noun Phrases

The arguments (noun phrases) which we have already introduced are the demonstratives **ti**, **ta** (English *this*, *that*). Additional demonstratives are **tio**, **tao** referring to situations, and **toi**, **toa** referring to items of speech or text. There seems to be some confusion in the sources about which demonstrative in the latter two pairs is "this" and which is "that".

We also mentioned the pronouns **da**, **de**, **di**, **do**, **du**. These always acquire reference from some argument mentioned earlier (in a precise way to be described).

A further set of pronouns which have a parallel role as names of the letters (putting the variables of mathematics in mind) is provided: they have their own distinctive way of acquiring reference.

Additionally, the pronouns **mi** (I, we) and **tu** (you), and a cloud of related first and second person pronouns are available.

All of these pronouns and demonstratives can be further diversified with numerical suffixes. Loglan has an amazing abundance of demonstratives and pronouns. We will later describe all of them.

The principal business of this section is to introduce simple descriptive noun phrases, and to give a first glimpse of the use of the Loglan predicate to construct arguments (use of these words as nouns rather than verbs). We will also introduce names.

1. **le mrenu ga sadji**

   The man is wise

2. **le blanu ga cmalo**

   The blue thing is small

3. **le cmalo je la Djan, ga bilti**

   The thing (person?) smaller than John is beautiful

4. **le donsu je le bakso jue la Meris, ga zavlo**

   The giver of the box to Mary is bad

The phrase **Le mrenu** refers to an object X which is understood by the hearer to satisfy **X mrenu**. Similarly, **Le cmalo je la Djan** refers to an object X which is understood by the hearer to be smaller than John, and **le donsu je le bakso je la Meris** refers to one who is understood by the hearer to have given the box to Mary. These sentences do not assert that the X under consideration has the property which is used to locate it: the property, already understood by the hearer to hold of that object, is used by the hearer to establish what object is being referred to. This can lead to paradox.

1. **Mi bleka le mrenu, i le mrenu ga fumna!**

   I looked at the man. The man is a woman!

I think heavier weather has been made of this point in the Loglan literature (made in Loglan 1) than is required. The hearer *does* understand that **le mrenu** refers to a man, and this is the expectation. It is simply that this fact about **le mrenu** is a presupposition in the environment of the sentences using it rather than part of what the sentences using it are asserting. The sentence **le mrenu ga fumna** is not false, but it does overthrow expectations: it indicates that something that the hearer believed was false, and after this utterance the designation **le mrenu** for the individual in question is likely to be retired.

# Chapter 4

# Loglan Sounds, Phonetics, and Word Forms

(in this section I'm trying to give a high level account of the phonetic rules which I am hoping to use to actually develop simplified PEG rules. If I can do this for phonetics, perhaps I can do it for grammar anon.)

The Loglan vowels are the *regular vowels* **a,e,i,o,u** and the *irregular vowel* **y**. The pronunciations of the regular vowels are typical Continental European (not English!)) pronunciations. The sound of **y** is officially the schwa, but we think there is something to be said for it being another sound easily distinguished from the regular vowels, such as the *oo* in English *look*. We have also considered ö and the Cyrillic letter that looks like bI as implementations of Loglan **y**.

The Loglan consonants are **b,d,f,g,h,j,k,l,m,n,p,r,s,t,v,z**. The pronunciations of these are standard European pronuniciations, except that **c** is English *sh* and **j** is the corresponding voiced sound found in English a*z*ure. **g** is always "hard". **h** has an alternative pronunciation as *ch* in Scottish English lo*ch* when final in a syllable [this is new: **h** does not occur in syllable final position in 1989 Loglan]. *n* is pronounced as *ng* in English si*ng* when it appears before **k,g** or syllable final *h*.

The vowels **i** and **u** are sometimes pronounced as the English consonants *y* and *w*.

The continuants **m,n,l,r** can appear as syllabic consonants (functioning as the vowel in a syllable). In this role, these consonants are doubled, **mm, nn, ll, rr**. The requirement that continuants be doubled is NEW, but it is actually suggested by JCB in Loglan 1.

There are some mandatory and some optional pairs of vowels which form diphthongs, which can serve as the vowel component of a single syllable.

The mandatory diphthongs are **ai, ao, ei, oi**. The pronunciations are as one would expect from the values of the vowels, except that **ao** is as in English c*ow*.

The optional diphthongs are the pairs of vowels beginning with **i** or **u**. The

pronunciation of these diphthongs is as if the initial **i** were the English consonant *y* or the initial **u** were the English consonant *w*. When a pair of adjacent vowels is not pronounced as a diphthong, one may flow into the other without pause, or a glottal stop (not expressed in writing except possibly indirectly by a hyphen or stress mark) may be inserted (NEW: we do not allow the glottal stop to be an allophone of the pause, as earlier versions of Loglan did).

A string of three or more vowels (appearing in the context of a name or predicate word: in a cmapua word made up of VV units, such a stream is resolved into pairs, each of which is pronounced as a monosyllable or disyllable as appropriate; an odd length stream of vowels parsed as cmapua will consist of a V word followed by a stream of VV units) which is not marked with explicit syllable breaks (a hyphen or a stress marker is used for this purpose, as we will discuss below) is resolved into syllables (possibly including consonants adjacent to the string of vowels) following a priority order reading left to right:

1. group the first two vowels and continue if they make up a mandatory diphthong.

2. pronounce the first vowel as a single syllable and continue if the second two make up a mandatory diphthong.

3. optionally, take the first vowel as a syllable and continue or group the first two vowels and continue if they make up an optional diphthong; the parser will always take the second alternative.

4. pronounce the first vowel as a single syllable and continue.

5. by "continue", we mean "apply the same set of rules to the remainder of the stream of vowels".

This is NEW, not the same as the rule in 1989 Loglan but it appears to have similar effects in practice.

A optional diphthong not appearing in a stream of three or more vowels may be pronounced either as a single syllable or two syllables: sometimes other factors will force the monosyllabic pronunciation. We believe that it is not possible to force the disyllabic pronunciation of an optional diphthong without explicit indication of a syllable break.

Where doubled vowels are not separated by a pause and not pronounced as a diphthong, one of them must be stressed: this always applies to **aa, ee, oo**, and applies to **ii**, **uu** unless they are pronounced *yee, woo*.

We now discuss the Loglan syllable. Each Loglan syllable is either unstressed, stressed, or emphatically stressed. Syllables may be separated by a hyphen. A marker ' of stress or * of emphatic stress may terminate a syllable: a stress marker may not be followed by a hyphen, as the stress marker itself serves the purpose of separating the syllable from a following syllable. Syllable breaks and stress markers do not have to be written explicitly, though it can be useful to do this. The precise definition of the syllable that we give does not appear in the sources, but every component of it is found there, and all words in

the dictionary are successfully resolved using this definition; we do not regard this definition as new except in detail. The use of the hyphen and the explicit stress markers is NEW.

Each Loglan syllable contains a vocalic unit, which is either a single vowel, a diphthong pronounced as such, or a doubled continuant.

A doubled continuant may not appear adjacent to another occurrence of the same consonant.

Each Loglan syllable consists of up to three parts. The first part (which is optional) is a consonant cluster called the initial consonant group. There is a list of pairs of consonants called permissible initial pairs. An initial consonant group will be either a single consonant, an initial pair, or a group of three consonants in which each adjacent pair of consonants is a permissible initial pair.[1]

The second part is the mandatory vocalic unit.

The third part, which is optional, consists of one or two final consonants. There is a list of impermissible medial pairs and a list of impermissible medial triples.[2] A final consonant may not be followed by a vowel and may not stand at the beginning of an impermissible medial pair or triple (regardless of whether the other consonants are in the same syllable, and ignoring stress marks and hyphens). A continuant pair is an impermissible medial pair: a final consonant may not participate in such a pair, even with a syllable break intervening. Every Loglan word must resolve into syllables. Some classes of words must also resolve into other small units which are not exactly syllables, though they do not usually cross syllable boundaries. A pair of final consonants may not consist of a non-continuant followed by a continuant [the last sentence is NEW, but seems self-evident: such a pair would basically have to be pronounced as a separate syllable, and no violations occur in the dictionary].

There are four phonetic classes of Loglan words: these are (1) structure words (**cmapua**), (2) name words, (3) borrowed predicates and (4) complex predicates.

Words (in the phonetic sense) end at whitespace, at a comma or mark of terminal punctuation (periods and some other punctuation marks; we do not give the list here, and in general remark that there are some subtleties of punctuation rather than phonetics in the provisional parser which we do not cover here), or sometimes without any explicit indication at all (where phonetics are sufficient to recognize where one word ends and another begins).

A comma in Loglan marks an explicit pause (and is followed by whitespace; the close comma used to indicate unusual syllable breaks between vowels in

---

[1] The initial pairs are **bl br ck cl cm cn cp cr ct dj dr dz fl fr gl gr jm kl kr mr pl pr sk sl sm sn sp sr st sv tc tr ts vl vr zb zl zv**

[2] The impermissible medial pairs consist of all doubled consonants, any pair beginning with **h**, any pair both of which are taken from **cjsz**, **fv**, **kg**, **pb**, **td**, any of **fkpt** followed by either of **jz**, **bj**, and **sb**.

There is a list of impermissble medial triples as well, consisting of **cdz**, **cvl**, **ndj**, **ndz**, **dcm**, **dct**, **dts**, **pdz**, **gts**, **gzb**, **svl**, **jdj**, **jtc**, **jts**, **jvr**, **tvl**, **kdz**, **vts**, and **mzb**. All of these consist of a consonant followed by an initial pair, but they are not permitted to occur with the juncture between syllables in either of the two positions.

earlier versions of Loglan is replaced by the hyphen which we use to represent syllable breaks in general).

Whitespace is sometimes an explicit pause and sometimes a word boundary which is not marked by any actual phonetic feature. Where whitespace does not appear, it should never be possible to pause. Where a pause is mandatory at whitespace, a comma should always be permitted (The old parser LIP does not always support this, but we regard this as debugging, not a novelty). There are situations where whitespace is allowed due to a word break but an actual comma pause would change the parse (and so in speech such a whitespace is not expressible as a pause).

Vowel initial words are always preceded by a pause if they are not at the start of a text or utterance. Consonant final words are always followed by a pause if they are not at the end of a text or utterance. Thus, whitespace preceded by a consonant or whitespace followed by a vowel must represent an actual pause. So we also regard whitespace preceded by consonants or followed by vowels as an explicit pause. It appears to be NEW that we must pause before the first in a stream of VV words, but it is also clearly necessary, as experiments with phonetic transcripts have revealed.

We note the subtle point that the end of a predicate word may have to be indicated by whitespace if the stressed syllable is not explicitly marked. So in this case whitespace may have no local phonetic meaning but will have the definite phonetic effect of signalling the presence of an earlier stressed syllable.

It is possible to resolve a stream of Loglan phonemes into words unambiguously, with the qualification that the resolution of streams of Loglan grammatical particles is actually done by the grammar proper. We indicate how to do this.

A structure word (in the phonetic sense: some phonetic structure words are actually semantically names or predicates) is a word which resolves into a stream of V, VV, CV, CVV, and Cvv-V units (where vv denotes a diphthong). These words are called **cmapua** in Loglan. The CVV units do not have to be syllables (there is no requirement that the VV be a diphthong, or that it be pronounced as such if it is an optional diphthong). A V unit can only appear initially; if any unit in a cmapua is of the shape VV, all units are of the shape VV. We recall the rule that one must pause before a word which begins with a vowel, so one must pause before an initial V- unit; it is not necessary to pause between VV units in a phonetic word made up of such units, but it is necessary to pause before a lone VV unit or the first in a stream of such units (the necessity of pausing before the first in a stream of VV units seems not to have been recognized in NB3). Except in the case of the end of a string of VV units, it is impossible to recognize the ends of individual words in a stream of cmapua phonetic units on phonetic grounds alone: the grammar proper allows us to resolve streams of unit cmapua into words, but for phonetic purposes we may regard streams of VV units and streams of non–VV units as "words".[3] It is part of the definition of a cmapua unit that it cannot be an initial segment of a predicate (either

[3]There is a further subtlety that certain streams of unit cmapua which are broken by explicit pauses may be recognized semantically as words of class PA or NI: this is a semantic rather than a phonetic proposal, not discussed here

complex or borrowed): a more precise statement of this is that a cmapua unit may not be followed without an intervening explicit pause by **y** or by **CyC**, and may only be followed without an intervening explicit pause by CC if the CC itself is initial in a predicate word. Where the final unit of a phonetic cmapua immediately precedes a predicate word and is stressed, it must be followed by a pause: if the stress is marked in writing and the following predicate is consonant initial, the pause must be explicitly written as a comma. [4]

A name is a stream of Loglan syllables which ends with a consonant followed by end of text, terminal punctuation, or a pause (noting that whitespace after a consonant always represents an explicit pause). In most but not all cases a name must be followed by a comma if not final, under the current parser, though the fact that whitespace after a consonant is an explicit pause suggests that this could be relaxed. A name must begin with a pause unless it is immediately preceded by one of a list of cmapua, the name marker words[5]. The right boundary of a name word is always readily recognized by the consonant followed by end of text, terminal punctuation, a comma, or whitespace. The left boundary is as far to the left as the stated conditions permit: it either (1) starts at a pause indicated by whitespace or a comma, which must further be recognizable as a mandatory pause, either because it is a comma or because of a preceding consonant or following vowel, or (2) starts right after a name marker word. We do not forbid a "false name marker" from occurring in a name word, as earlier versions of Loglan did and as Lojban does: a false name marker is a phonetic copy of a name marker word such that the remainder of the name word following it would itself be a legal name word. The thing to note is that if the tail is actually the name which one intends to utter, then one must pause explicitly before or after the false name marker to enforce this reading (whereupon it becomes a true name marker). In order to avoid unintended miswriting or misspeaking of name words, we structure the grammar to strongly limit the contexts in which a name word may appear which is not preceded (with or without pause) by a name marker word: for example, we outlawed unmarked vocatives (a reform made by the current Academy) , and we generally require that a name word which contains a false name marker always follow a name marker (sometimes without pause between the marker and the name, in order to forbid a bad reading).

The parser further will raise an error if it finds a name marker word followed at an interval of any length by an actual name word with a non-explicit pause

[4] We note a subtle point about the articulation of acronyms in Loglan: these are semantically names but phonetically cmapua. The legacy vowel names are of the weird shape VCV; they can occur without initial pause following a CVV unit because the (CVV)(VCV) shape (when articulated as letters) can be rearticulated as (Cvv-V)(CV) for purposes of articulation as cmapua units. We also acknowledge a frank irregularity, though it will not often occur: the CVV letters of the common sorts have the VV actually a diphthong, but there are C**eo** letters, and these do work in acronyms preceding a legacy vowel.

[5] These are **la, hoi, hue, ci, liu, gao, loi, loa, sie, sia, siu**. The first five are for one reason or another unavoidably name markers. **gao** is here because of a proposal to allow it to construct letter names from name words. The last five (the words of social lubrication) are NEW on this list. We could continue to say **Loi hoi Djan** instead of **loi Djan**, I suppose.

but no explicit pause intervening between the name marker word and the end of the name word. There are other uses of name marker words: it is the obligation of the speaker to explicitly pause at some point after such an occurrence of a name marker and before the next occurrence of an actual name word, and Loglan orthography requires this to be indicated explicitly. Note that whitespace before a vowel or after a consonant does suffice, but where this doesn't happen, an explicit comma pause (of the form V, C) may be required. Actually complying with this rule is best implemented by style directives such as "always pause after a predicate name", rather than by attention to this esoteric rule as such.

Names are the only Loglan words which end with consonants.

The rule that false name markers cannot occur in names has already been abandoned (in the 1990s) by TLI Loglan; it still obtains in Lojban, which has very few name markers. Precise definition of what you do with false name markers was hard to think about before phonetic parsing was available. The requirement that names resolve into syllables is NEW (I seem to recall that Lojban does something like this?), and interacts with the definition of a false name marker as indicated above. The addition of the words of social lubrication to the list of name markers is NEW. The automatic detection of dangerous situations where a non-explicit pause should be made explicit is NEW, and we can report from extensive experience in parsing Alex Leith's Visit to Loglanida that there are straightforward ways to correct problems it detects (and that it really does detect things which are potential problems).

A borrowed predicate is a stream of Loglan syllables which contains just one stressed syllable, whose vocalic unit is not a continuant pair, the stressed syllable being followed by at least one and no more than two syllables, the first of these, if there are two, having a continuant vocalic unit, and the final one not having a continuant vocalic unit: in other words, it is penultimately stressed, ignoring one possible intervening unstressed continuant syllable. A borrowed predicate contains at least one pair of distinct adjacent consonants; it is permissible for one of these to be in a continuant pair. A borrowed predicate may not contain two successive syllables with continuant pairs nor may it start with such a syllable (or, as noted, end with such a syllable), nor may a continuant pair follow a vowel in a borrowed predicate. A borrowed predicate may not contain a doubled vowel unless it is pronounced as a diphthong. A borrowed predicate may not contain **y**. The part of the borrowed predicate before the first pair of distinct adjacent consonants must have the property that omitting it will not leave a legal borrowed predicate. A borrowing cannot begin VCCV with the CC permissible initial, and there can be no CCVV or CCCVV borrowed predicates[6]. (C)VVV...CCV (where the dots represent zero or more additional vowels and the CC pair is permissible initial) cannot be the shape of a borrowing: this restriction is caused by a technical problem with borrowing djifoa (see below).[7]

[6]The CCVV predicates are forbidden so that CVCCCV complex predicates do not have to be **y**-hyphenated; the CCCVV predicates would, if allowed, greatly complicate our parsing algorithm.

[7]What needs to be averted is the possibility of reading a borrowing djifoa ending with CCV**y** as a stream of phonetic cmapua units followed by a predicate beginning with the

The end of a borrowed predicate is recognized either by seeing a explicitly stressed syllable and counting the one or two allowed following unstressed syllables, or by whitespace, a comma, end of text, or terminal punctuation: in the latter case, the fact that one is in a borrowed predicate is recognized by the occurrence of two adjacent distinct consonants.

A borrowing must not resolve into *djifoa* (see the next section), even ones with badly placed internal syllable boundaries or lacking required hyphens (noting that the five-letter djifoa and'the borrowing djifoa, when not final, *include* their **y** hyphens and so will not be involved in such resolutions, since a borrowing candidate cannot include **y**.). This both ensures that complex predicates are read as complex predicates, and ensures that certain illegal complexes are not read as legal borrowed predicates.

There is nothing actually new in the description of borrowed predicates: some features are points worked out in the 1990's (all details of borrowing djifoa are late and not in 1989 Loglan 1). Forbidding doubled vowels in borrowings is the most recent change, made by the current Academy in the last few years. Everything else is explicit in the sources somewhere (there may be some guesswork about the exact rules for use of continuants for gluing, but they fit actual practice). The precise definition of the syllable was made in order to make it possible to implement the description of borrowings in NB3 and L1.

A complex predicate is a stream of units distinct from syllables, called *djifoa*, with the additional property that any syllable breaks respect the djifoa boundaries (testing for resolution into djifoa requires that djifoa with badly placed internal syllable boundaries be recognized; predicates which resolve into djifoa with badly placed boundaries are to be rejected as borrowings as well).

The djifoa take the forms

1. CVV (legal syllable forms Cvv or CVV or CV-V)
2. CVC
3. CCV (the CC must be permissible initial).
4. CCVCV (when this is not final, the final V is replaced with **y**). The CC must be permissible initial. The only legal syllable break is CCV-CV.
5. CVCCV (when this is not final, the final V is replaced with **y**). The CC must not be impermissible medial. CV-CCV (if the CC is permissible initial) and CVC-CV are legal syllable breaks. Either break is permitted if the CC is permissible initial.
6. a borrowing predicate (modified to have final rather than penultimate stress) with appended **y** (in an unstressed syllable by itself) if not in final position.
7. When attempting to resolve a predicate into djifoa to establish that it cannot be a borrowed predicate, the forms to consider are CVV, CVV**r**,

CCV**y** djifoa instead of the intended borrowing djifoa.

> CVV**n** (only when followed by **r**), CVC, CCV, and the five letter forms in final position only, plus the illegal C-CV, C-CVCV (the last in final position, the hyphen indicating an explicit syllable break or stress marker). A mechanical resolution into these forms, without any side conditions, shows that a string cannot be a borrowed predicate.

Note that a complex predicate may not contain any continuant pair, except in the context of a borrowing djifoa.

The only complexes consisting of a single djifoa are the five-letter djifoa (these could also be viewed as a separate species of primitive predicates).

Djifoa may have phonetic hyphens appended, which may take the shapes **n**, **r**, or **y**. Only one phonetic hyphen can be appended. The consonant hyphens can only be appended to CVV cmapua. A phonetic hyphen **n** appears only when followed by **r** in the next djifoa. A phonetic hyphen is never final in a complex predicate nor will it follow a five letter djifoa. A phonetic hyphen **y** is always unstressed, and appears by itself in a syllable or in a C**y** syllable in the context CV-C**y** of a CVC djifoa (the **y** in a five letter djifoa in non-final position may participate in a C**y** or CC**y** syllable, which is also always unstressed). A borrowing djifoa is always preceded by **y** if not initial (the **y** will be a hyphen or a consitituent of a five letter or borrowing djifoa) and includes an appended **y** if not final.

An initial CVV followed by a djifoa beginning CV must be hyphenated with a consonant. An initial CVC followed by C in a way which would make a permissible initial pair must be hyphenated with **y** if the entire word is not of length 6. These are rules to allow recognition of the start of a predicate word.

There must be a CC pair (or a C**y**C pair) in a complex predicate (this is not a separate stipulation: it follows from what we have already said).

A complex predicate must end with a vowel (so the last djifoa will not be CVC). Any adjacent pair of consonants in a complex predicate may not be impermissible medial.

A complex predicate may contain stress only in the penultimate position (where it must be stressed; in determining the penultimate stress, a syllable with **y** may be ignored) or in the final position of a borrowing djifoa (where stress is optional, unless such a syllable is also penultimate in the predicate). It is also permitted to contain a pause (in explicit comma form) after the **y** following a stressed borrowing djifoa (violating our expectations about word boundaries, but it is there in the sources, and it may be practical, as borrowing djifoa are large). We think that the stress shift in borrowing djifoa may serve as a useful marker that something odd is going on when this happens. Note that stress strongly constrains where a CV-V djifoa may appear if the VV is a doubled vowel.

Nothing in the description of complex predicates is new, though all language about borrowing djifoa comes from decisions taken in the 1990s after the 1989 edition of Loglan 1. Considerations about explicit stress markers and syllable breaks are new but consistent with the logic of complex predicates as already defined: we do not want an illegal complex to become a legal borrowing by

moving a syllable break so that it doesn't conform with a djifoa boundary. We expect that directly implementing the resolution procedure described above will allow us to relax some rather odd rules about placement of explicit syllable breaks which are currently enforced in borrowings under our parser.