

COLORADO TECHNICAL UNIVERSITY

ENABLING A LEGACY MORPHOLOGICAL PARSER
TO USE DATR-BASED LEXICONS

VOLUME 2 of 2

APPENDICES

A DISSERTATION SUBMITTED TO
THE GRADUATE COUNCIL
IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF COMPUTER SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

BY

MICHAEL A. COLBURN

M.C.I.S., University of Denver, Denver, CO, 1991
M.A., University of Texas, Arlington, TX, 1981
B.A., Rockmont College, Denver, CO, 1975

DENVER, COLORADO
DECEMBER 1999

ABSTRACT

These appendices contain foundational work conducted for the author's research. The research is an investigation of the feasibility and desirability of using DATR with AMPLE, a legacy morphology exploration tool developed by the Summer Institute of Linguistics. The research demonstrates the feasibility of using DATR with AMPLE by showing how DATR can be used to encode the lexical information required by AMPLE and by presenting an interface between AMPLE and DATR-based lexicons that does not require modification of AMPLE itself. The desirability of using DATR with AMPLE is shown by demonstrating that there are generalizations that cannot be captured using AMPLE's lexical knowledge representation language (LKRL) that can be captured in DATR, thereby reducing redundancy of lexical information.

Appendix 1 is an overview of Ogea morphophonemics and morphotactics. Appendix 2 is an Ogea AMPLE lexicon that supports the parsing of all Ogea examples found in Appendix 1. Appendix 3 is the DATR version of the Ogea lexicon. Appendix 4 is both an AMPLE and DATR version of a subset of a Yalálag Zapotec AMPLE lexicon. Finally, Appendix 5 lists various Perl scripts used for the research.

TABLE OF CONTENTS

ABSTRACT	II
TABLE OF CONTENTS.....	III
LIST OF TABLES	VI
LIST OF ABBREVIATIONS.....	VIII
APPENDIX 1 AN OVERVIEW OF OGEA MORPHOPHONEMICS AND MORPHOTACTICS	1
1.0 Description	1
1.1 Morphophonemics	4
1.2.1 Rule 1. Syncope.....	7
1.2.2 Rule 2. Glide Formation	8
1.2.3 Rule 3. Epenthetic Consonant Insertion	9
1.2.4 Rule 4. Nasal Assimilation.....	9
1.2 Morphotactics	10
1.2.1 Clitics	10
1.2.2 Adverbial Suffix	12
1.2.3 Verbs.....	12
1.2.4 The Auxiliary Verb <i>bo-</i>	48
1.2.5 Nouns and Adjectives	51
1.2.6 Pronouns.....	55
1.2.7 Templatic Morphology	57
1.3 Summary	62
1.4 References	63

APPENDIX 2 OGEA AMPLE LEXICON.....	64
A2.1 Description	64
A2.2 The Listing.....	64
APPENDIX 3 THE OGEA DATR LEXICON.....	106
A3.1 Overview	106
A3.2 The Ogea DATR Theory File	106
A3.2.1 Description	106
A3.2.2 File Listing	108
A3.3 The Ogea DATR Query File	178
A3.3.1 Description	178
A3.3.2 File Listing	178
A3.4 Batch File to Execute DATR.....	184
A3.4.1 Description	184
A3.4.2 File Listing	184
A3.5 The Ogea DATR Trace File	185
A3.5.1 Description	185
A3.5.2 File Listing	185
APPENDIX 4 THE YALÁLAG ZAPOTEC AMPLE AND DATR LEXICONS ...	267
A4.1 Overview	267
A4.2 The Yalálag Zapotec AMPLE Root Database File.....	268
A4.2.1 Description	268
A4.2.2 File Listing	268
A4.3 The Yalálag Zapotec DATR Theory File.....	274
A4.3.1 Description	274
A4.3.2 File Listing	274

A4.4 The Yalálag Zapotec DATR Query File	283
A4.4.1 Description	283
A4.4.2 File Listing	283
A4.5 Batch File to Execute DATR.....	284
A4.5.1 Description	284
A4.5.2 File Listing	284
A4.6 The Yalálag Zapotec DATR Trace File	284
A4.6.1 Description	284
A4.6.2 File Listing	284
APPENDIX 5 PERL SCRIPTS.....	291
A4.1 Script to Compare Database Files.....	291
A4.1.1 Description	291
A4.1.2 The File Listing	291
A4.2 Script to Generate AMPLE Database File	293
A4.2.1 Description	293
A4.2.2 The File Listing	293
A4.3 Scripts to Compare AMPLE Output.....	294
A4.3.1 Description	294
A4.3.2 The File Listing	295
A4.4 Scripts to Calculate Reduction of Redundancy	296
A4.4.1 For Ogea.....	296
A4.4.2 For Yalalog Zapotec	299

LIST OF TABLES

Table 1 Ogea Phonemes and Orthography.....	5
Table 2 Classes Used for Ogea Morphophonemic Rules.....	6
Table 3 Ogea Verb Endocenter.....	17
Table 4 Ogea Medial, Same Referent, Indicative Verb Exocenter.....	20
Table 5 Ogea Medial, Different Referent, Indicative Exocenter.....	22
Table 6 Ogea Final, Indicative Exocenter.....	23
Table 7 Ogea Contrafactual Verb Exocenter.....	24
Table 8 Ogea Subjunctive Exocenter.....	25
Table 9 Ogea Infinitive Exocenter.....	26
Table 10 Ogea Object Suffixes.....	28
Table 11 Ogea Manner Suffixes.....	30
Table 12 Ogea Benefactive Suffixes.....	30
Table 13 Ogea Remote Past Tense Suffixes.....	32
Table 14 Ogea Immediate Past and Present Tense Suffixes.....	34
Table 15 Ogea Future Tense Suffixes.....	37
Table 16 Ogea Indicative Tense Suffixes - Summary.....	41
Table 17 Ogea Indicative Verb Subject Suffixes.....	41
Table 18 Ogea Indicative Verb Tense and Subject Suffixes Combined.....	42
Table 19 Ogea Contrafactual Subject Suffixes.....	43
Table 20 Ogea Subjunctive Verb Person Suffixes.....	44
Table 21 Ogea Subjunctive Verb Number Suffixes.....	44
Table 22 Ogea Infinitive Verb Number Suffixes.....	45

Table 23 Suppletive Verb Root Forms for 'See'	47
Table 24 Suppletive Verb Root Forms for 'Hit'	47
Table 25 Suppletive Verb Root Forms for 'Get'	47
Table 26 Suppletive Verb Root Forms for 'Give'	47
Table 27 Comparative Matrix for All Suppletive Verb Roots	48
Table 28 Subject Suffixes for the Root <i>bo-</i>	49
Table 29 Ogea Possessive Suffixes.....	52
Table 30 Ogea Personal Pronouns	56
Table 31 Ogea Intensive/Reflexive Pronouns	56
Table 32 Ogea Possessive Pronouns.....	56
Table 33 Various Ogea Verb Roots, Suffixes, and Pronouns	58
Table 34 Paradigmatic Presentation of Person/Number Template.....	59
Table 35 Siroi Object Suffixes, Possessive, and Personal Pronouns.....	61

LIST OF ABBREVIATIONS

The following abbreviations are used to gloss language examples.

1	First person
2	Second person
3	Third person
B	Benefactive
cert	Certainty
Contra	Contrafactive
d	Dual
Hab	Habitual
imp	Imperative
int	Intensive
inf	Infinitive
Nom	Nominative
O	Object
p	Plural (note: a number after the p indicates the 1st or 2nd type of plural)
P	Possessive
perf	Perfective
pred	Predictive
s	Singular
S	Subject
Sj	Subjunctive
SR	Switch Referent
Tf	Future Tense
Tip	Immediate Past Tense
Tp	Present Tense
Trp	Remote Past Tense
TO	Temporal overlap
TS	Temporal Succession
vAux	Verbal Auxilliary
vAuxR	Verbal Auxilliary Root

APPENDIX ONE
AN OVERVIEW OF
OGEA MORPHOPHONEMICS AND MORPHOTACTICS

1.0 Description

The purpose of this appendix is to provide information about Ogea morphophonemics and morphotactics. Information is provided at a level of detail sufficient for the reader to follow the discussion about the Ogea AMPLE-based lexicon and the Ogea DATR-based lexicon found in Appendices Three and Four. An additional purpose of this appendix is to provide a description of the language's morphophonemics and morphotactics since little has been published on the Ogea language.

Ogea¹ is a Papuan (non-Austronesian) language spoken by approximately 700 people living in an area 18 kilometers south of the town of Madang, in the Madang Province of Papua New Guinea. The Ogea live in four villages—Garima, Dogia, Balama, and Erima, bounded by the Golgol and Yawor rivers, and the Astrolabe Bay. Ethnically, the Ogea are Melanesians. There is evidence that Polynesians migrated through New Guinea in pre-European times. Although not an Austronesian language, Ogea contains a number of Austronesian cognates—perhaps loan words—indicating the possibility of interaction between the Ogea and the migrating Polynesians. The first recorded contact with the Ogea by a European was likely made by the Russian scientist, Miloucho-Maclay, who describes visits to several Ogea villages between 1871 and 1883 in his diary [5]. Ogea has at least one Russian loan-word, *sapora* 'axe, hatchet', evidence

perhaps that Mikloucho-Maclay introduced metal to the Ogea. In pre-European times, the Ogea were a stone-age society.

Linguistically, Ogea is classified as follows [3]:

Trans-New Guinea Phylum
 Madang-Adelbert Range
 Madang
 Rai Coast
 Nuru Family
 Ogea

Ogea linguistic data were collected by this author while living among the Ogea between 1977 and 1986 under the auspices of the Summer Institute of Linguistics, and during a visit in 1999. The original grammatical analysis was presented by this author in [1] and was based on a collection of 40 pages of native authored text. The original analysis was reviewed by Dr. Ger Reesink, then of the Summer Institute of Linguistics in Papua New Guinea, and now of the Leiden University in the Netherlands. For purposes of this dissertation, this author did a reanalysis, using a collection of native-translated² texts containing more than 223,000 words. Where paradigms could not be completed using texts, data were elicited from native speakers. The original analysis was based on the minority dialect of Ogea (spoken in Erima village). The analysis presented here is based on the majority dialect (spoken in Garima, Dogia, and Balama). Deviations from

¹ Also known as Erima in the literature. However, speakers of the language refer to it as Ogea, not Erima (which is a village name).

² The use of translated texts is far from ideal, but few native-authored texts are available. The texts used were translated by a native Ogea speaker. The use of such texts for morphological studies is likely more defensible than using them for studies of syntax or discourse. Influence of the source language is less likely to impact morphology than, say, syntax or discourse. To date no morphological contradictions have been found between the large body of translated texts available to this author and the small body of native authored texts that are available. Also, all examples used in this appendix have been verified by a native Ogea speaker.

the original analysis will be noted. To the best of this author's knowledge, no analysis of Ogea phonology, grammar, or morphotactics exists apart from that of this author.

Phonemically, Ogea has a 15 vowel system with 17 consonants. Syntactically, Ogea is a Subject-Object-Verb (SOV) language, with adjectives following nouns, and deictics following adjectives—the reverse of English. Morphologically, Ogea is a highly inflected, suffixing language, with most of the complexity occurring with verbs. There are over 100 basic verbal suffixes, the number of which is significantly multiplied by allomorphic variants. Ogea sentences are often composed of chains of verbs, with suffixes indicating sentence *medial* versus *final* positions. Ogea verbs encode inter-clausal temporality (temporal succession—one action occurs following another—and temporal overlap—actions occur simultaneously). They also encode switch reference. Switch reference indicates whether the referents of the clause in question are referents in the following clause. It is useful to classify Ogea verbal suffixes into two major categories: endocentric and exocentric, following the lead of Staalesen [6] and Wells [7]. Endocentric suffixes occur between the verb root and the exocenter. Endocentric suffixes include manner, object, and benefactive suffixes, among others. The same set of endocentric suffixes are used with varying sets of exocentric suffixes. The endocenter itself is composed of the verb root plus the endocentric suffixes. Exocentric suffixes encode inter-clausal temporality, tense, mood, subject, and switch reference. They are termed exocentric because they may contain suffixes that relate to the clause that follows (i.e., inter-clausal temporality and switch reference relate the current clause to the one that follows it).

In the sections below, Ogea morphophonemics will be presented first, followed by an overview of Ogea morphotactics. Note that in the examples provided throughout Appendix One, when multiple glosses are possible, sometimes only one is selected in order to allow the example to fit on a single line. Appendix Three or Four may be consulted to see the full range of possible glosses for any particular morpheme.

1.1 Morphophonemics

The original analysis of Ogea made by this author did not include morphophonemic rules. Based on the current analysis, there are four morphophonemic rules in Ogea. The rules are as follows:

- Rule 1. Syncope
- Rule 2. Glide Formation
- Rule 3. Epenthetic Consonant Insertion
- Rule 4. Nasal Assimilation

The rules are based on the orthographic representation of Ogea, which also happens to be phonemic (see Table 1). In this scheme, there is partial ordering of the rules in that Rule 2 must apply before Rules 3 or 4. The classes used for Ogea morphophonemic rules are shown in Table 2. Note that the verb roots listed for Morpheme Class One are a closed class, whereas the verb roots in Morpheme Class Two are an open class. In the statement of morphophonemic rules, the '+' symbol is used below to indicate a morpheme boundary, and a 'Ø' indicates the loss of a segment.

Table 1 Ogea Phonemes and Orthography

Type		Values	
Vowels	Normal	Phonemic	a e i o u
		Orthographic	a e i o u
	Nasalized	Phonemic	ã ě ĩ õ ã
		Orthographic	a e i o u
	Lengthened	Phonemic	a: e: i: o: u:
		Orthographic	aa ee ii oo uu
Consonants	Nasals	Phonemic	m n ɲ ŋ
		Orthographic	m n ny ng
	Plosives	Phonemic	t k b d g
		Orthographic	b t k d g
	Fricatives	Phonemic	ɸ s β
		Orthographic	f s w
	Affricates	Phonemic	dʒ
		Orthographic	j
	Trill	Phonemic	r
		Orthographic	r
	Approximants	Phonemic	w l y
		Orthographic	w l y

Table 2 Classes Used for Ogea Morphophonemic Rules

Class	Description		
A	Alveolar plosives		
B	Bilabial plosives		
C	All consonants		
N	Nasals		
V	All vowels		
X	Velar plosives		
MC1	Verb Roots	ati- aule- awa- bisi- biti- dage- heige- huge- inyi- isi- mani- mere- mini- nyi- te- tafa- tani- ume- uye- yafa- Table 23 Table 24 Table 25 Table 26	make carry on the shoulder leave bathe burn tie appear, arrive wash be, remain hear go up cry go down eat get put place inside die dance sit see hit get (a person) give
	Polysyllabic Object and Benefactive Suffixes	-hagi- -hani- -hari- -higi- -hini- -hiri- -nagi- -nari- -nigi- -niri- -tagi- -tari- -tigi- -tiri-	B1p B2s B1d O1p O2s O1d B3p B3d O3s O3d B2p B2d O2p O2d
MC2	Verb Roots	Those not listed in MC1	
	Suffixes	Those not listed in MC1	

1.2.1 Rule 1. Syncope

This rule states that the final vowel of a morpheme is dropped when the morpheme precedes another morpheme that begins with a vowel, and is otherwise retained. The rule only applies to morphemes of Morpheme Class 1 (see Table 2).

$$(1) \quad V]^{MC1} \rightarrow \emptyset / _ +V$$

Some examples of Rule 1 are shown below. The examples are given in pairs. The first example of each pair shows the loss of the vowel, and the second example of each pair shows the retention of the vowel. For each example, the morpheme in bold is the one to which the rule applies.

- (2) a. **yafa-** + -o- + -na → yafona
 sit -Ind.Tf -Ind.S3s
 'he/she will sit'
- b. **yafa-** + -de → yafade
 sit -Ind.S3p
 'he/she sits'
- (3) a. **te**³- + -a- + -i → tai
 get -Ind.Trp -Ind.S3s
 'he/she got'
- b. **te-** + -ni → teni
 get -Ind.S1/2s
 'I get'
- (4) a. le- + **-hini-** + -e- + -ye → lehineye
 speak- -O2s -Ind.Trp -Ind.S1/2p
 'We spoke to you'
- b. le- + **-hini-** + -ni → lehinini
 speak- -O2s -Ind.S1/2s
 'I speak to you'

³ A variant of /te-/ is /tee-/. This appears to be a stylistic lengthening of the vowel.

1.2.2 Rule 2. Glide Formation

This rule states that in the case of verb roots and suffixes whose shape is CVi, the final /i/ becomes /y/ preceding a suffix that begins with a vowel, and remains /i/ otherwise.

$$(5) \quad i \rightarrow y / +CV_ +V$$

Examples of Rule 2 are provided below. As with Rule 1, the examples are given in pairs.

- (6) a. te- + **-hai-** + -o- + -na → tehayona
 get -B1s- -Ind.Tf- -Ind.S3s
 'He/she will get (it) for me'
- b. te- + **-hai-** + -g- + -ou → tehaigou
 get- -B1s- -Hab- -Nom
 'One who get (things) for me'
- (7) a. **goi-** + -o- + -na → goyona
 go- -Ind.Tf- -Ind.S3s
 'He/she will go'
- b. **goi-** + -g- -a- + -I → goigai
 go- -Hab- -Ind.Trp -Ind.S3s
 'He/she habitually went'
- (8) a. **jai-** + -e- + -ye → jayeye
 get.up- -Ind.Trp- -Ind.S1/2p
 'We/you (plural) got up'
- b. **jai-** + -du → jaidu
 get.up- -TS
 'Having gotten up'

Note that Rule 2 does not apply to polysyllabic morphemes that end with /Vi/. Instead, Rule 3 applies.

1.2.3 Rule 3. Epenthetic Consonant Insertion

This rule states that a /w/ is inserted at the beginning of morphemes belonging to Order Class 8⁴ if they begin with a vowel and if they follow a morpheme of Morpheme Class 2 that ends in a vowel. Otherwise, a /w/ is not inserted.

$$(9) \quad V]^{OC8} \rightarrow wV / V+]^{MC2} _$$

Some examples of Rule 3 are:

- (10) a. oofa- + -ou → oofawou
 watch.over- -Nom
 'One who watches over'
- b. oofa- + -g- + -ou → oofagou
 watch.over- -Hab -Nom
 'One who habitually watches over'
- (11) a. ilei- + -a- + -i → ileiwai
 press.down- -Ind.Trp- -Ind.S3s
 'He/she pressed (it) down'
- b. ilei- + -nigi- + -a- + -i → ileinigai
 press.down- -O3p -Ind.Trp -Ind.S3s
 'He/she pressed (it) down'
- (12) a. yafa- + -de- + -e- + -ge → yafadewege
 sit- -well -Contra.S1s-Contra
 'If I had sat properly...'
- b. yafa- + -e- + -ge → yafege
 sit- -Contra.S1s-Contra
 'If I had sat...'

1.2.4 Rule 4. Nasal Assimilation

With few exceptions, Ogea syllables are open. That is, they do not end with a consonant. One exception⁵ occurs with certain verb roots that end with a nasal. When such roots combine with a suffix that begins with a stop, the nasal assimilates the point of

⁴ See Table 5 through Table 9. An order class 8 suffix is the first suffix of the endocenter, if the endocenter does not have a temporality suffix. Otherwise it is the second suffix of the endocenter.

⁵ The other exception is certain subject suffixes in the future tense when they precede the auxiliary verb *bo-* 'intent'. This is discussed in section 1.2.3.5.6 Future Tense Suffixes.

articulation of the stop that begins the morpheme that follows. If the root combines with a suffix that begins with any other consonant, the vowel preceding the nasal becomes nasalized, and the nasal segment is deleted.

- (13) VN → Vm / _ - [B]
 → Vn / _ - [A]
 → Vng / _ - [X]
 → \tilde{V} / _

Some examples, using /tuN-/ 'to poke', are:

- (14) tuN- + -boro- + -Ø- + -na → tumborona
 poke- -fully -Ind.Tp -Ind.S3s
 'He/she pokes all of(it)'
- (15) tuN- + -de- + -Ø- + -na → tundena
 poke- -well -Ind.Tp-Ind.S3s
 'He/she pokes (it) well'
- (16) tuN- + -g- + -a- + -ne- -nga → tungganenga
 poke- -TO -Ind.Trp/Tip/Tp-Ind.S3p-SR
 'While he/she poked (it) (switch referent)...'
- (17) tuN- + -Ø- + -na → t na
 poke- -Ind.Tp-Ind.S3s
 'He/she pokes (it)'
- (18) tuN- + -a- + -ne → t wane
 poke- -Ind.Trp-Ind.S3p
 'They poked (it)'

1.2 Morphotactics

1.2.1 Clitics

Ogea has three clitics: *-fe* 'like', *-nga* 'emphatic', and *-ha* 'pertaining to'. In native authored text, Ogeans concatenate these morphemes with the preceding word rather than writing them as separate words. The clitics *-fe*, *-ha*, and *-nga* appear to attach to a root of any non-verb class. *-nga* also attaches to verb roots and endocentric suffixes, as covered in section **1.2.2 Adverbial Suffix**.

1.2.1.1 -fe 'like'

- (19) buuwafe
 buuwa-fe
 pig -like
 'like a pig'

1.2.1.2 -nga 'just/only'

- (20) wahanga
 wa -ha -nga
 that-pertaining.to-just/only
 'only that'
- (21) nonga
 no -nga
 he/she-just/only
 'just he/she'
- (22) yamenga
 yame-nga
 my -just/only
 'just mine'
- (23) la -nga
 langa
 at -just/only
 'just there'

1.2.1.3 -ha 'pertaining to'

- (24) waha
 wa -ha
 that-pertaining.to
 'regarding that'
- (25) yameha
 yame-ha
 my -pertaining.to
 'of mine'
- (26) mataneha
 matane -ha
 village-pertaining.to
 'of (the) village'
- (27) matane anyakaroha
 matane anyakaro-ha
 village large -pertaining.to
 'of the large village'

1.2.2 Adverbial Suffix

As an adverbial suffix, *-nga* can suffix verb roots or any endocentric suffix.

(28) lengawai
 le -nga -wa -i
 speak-just/only-Ind.Trp-Ind.S3s
 'he/she only spoke'

(29) letungawai
 le -tu -nga -wa -i
 speak-O3s-just/only-Ind.Trp-Ind.S3s
 'he/she spoke only to him/her'

1.2.3 Verbs

1.2.3.1 Derived Verbs

In Ogea, it appears that any adjective may be used as a verb. Using an adjective as a verb semantically encodes a state occurring in the past or future. Present states are encoded using stative constructions which pair demonstratives, pronouns, and noun phrases with an adjective or noun phrase. (There is no copulative verb in Ogea. To say 'the man is big', Ogeans say 'the man big'). The examples, below, of adjectives used as verbs are presented with three sets for each adjective. The first part shows the adjective used in a stative construction. The second and third parts show the adjective suffixed as a verb. The third part also shows the use of the causative suffix *-mu-*.

(30) a. no tutu
 he/she/it short
 'He/she/it is short.'

b. tutuwona
 tutu -wo -na
 short-Ind.Tf -Ind.S3s
 'It will become short.'

- c. tutumutuwau
 tutu -mu -tu -wa -u
 short-causative-O3s-Sj.S2-Sj.SS
 'Make it become short!', or 'Shorten it!'
- (31) a. wa inyaba
 it bad
 'It is bad'
- b. inyabawai
 inyaba -wa -i
 bad -Ind.Trp-Ind.S3s
 'It became bad'
- c. inyabamutudu
 inyaba -mu -tu -du
 bad -causative-O3s-TS
 'Having harmed him/her...'
- (32) a. wa tigini
 he/she/it straight
 'He/she/it is straight'
- b. tiginiwona
 tigini -wo -na
 straight-Ind.Tf-Ind.S3s
 'It will become straight'
- c. tiginimuwai
 tigini -mu -wa -i
 straight-causative-Ind.Trp-Ind.S3s
 'It was made straight'

1.2.3.2 Reduplication

In Ogea, reduplication occurs in three forms. In one form, the first consonant and vowel (CV) of the root is reduplicated. In the second form, the entire stem is reduplicated. It is also possible that CV reduplication may be embedded in a reduplicated stem, as will be seen in the examples below. In the third form, an entire suffix is reduplicated.

1.2.3.2.1 Reduplication of the First CV of a Stem

Although Ogea has object suffixes that encode for person and number, these suffixes are not used for transitive verbs that take inanimate objects. In the case of actions involving more than one inanimate object, Ogea indicates plurality of objects by use of the suffix *-ru-*. In addition, the segments following the first CV of the root are deleted, and the first CV of the verb root is reduplicated. In this case, the reduplication possibly indicates repetitive action since more than one object is involved. Such CV reduplication is always accompanied by the plural suffix *-ru*. Note that the CV reduplication only applies to roots that begin with a consonant.

- (33) a. tauni
 tau -∅ -ni
 plant-Ind.Tp-Ind.S1/2s
 'I am planting'
- b. tataruni
 ta -ta -ru-∅ -ni
 plant-plant-p2-Ind.Tp-Ind.S1/2s
 'I am planting (many things)'
- (34) a. kouni
 kou -∅ -ni
 break.in.two-Ind.Tp-Ind.S1/2s
 'I am breaking into two (pieces)'
- b. kokoruni
 ko -ko -ru-∅ -ni
 break.in.two-break.in.two-p2 -Ind.Tp-Ind.S1/2s
 'I am breaking (many things) into two (pieces)'
- (35) a. karuni
 karu-∅ -ni
 cut -Ind.Tp-Ind.1/2s
 'I am cutting'

- b. kakaruni⁶
 ka -ka -ru-Ø -ni
 cut-cut -p2-Ind.Tp-Ind.S1/2s
 'I am cutting (many things)'

1.2.3.2.2 Reduplication of a Stem

In Ogea, reduplication of an entire stem indicates habitual action. As noted above, CV reduplication involving plural objects only applies to verb roots beginning with a consonant. In (39), the non-reduplicated verb begins with a vowel, and therefore a plural object is only indicated by the plural suffix *-ru*. However, the entire stem is reduplicated to indicate habitual action.

- (36) a. yafani
 yafa-Ø -ni
 sit -Ind.Tp-Ind.S1/2s
 'I am sitting'
- b. yafayafani
 yafa-yafa-Ø -ni
 sit -sit -Ind.Tp-Ind.S1/2s
 'I habitually sit'
- (37) a. karuni
 karu-Ø -ni
 cut -Ind.Tp-Ind.S1/2s
 'I am cut'
- b. karukaruni
 karu-karu-Ø -ni
 cut -cut -Ind.Tp-Ind.S1/2s
 'I habitually cut'
- (38) a. kakaruni
 ka -ka -ru-Ø -ni
 cut-cut -p2-Ind.Tp-Ind.S1/2s
 'I am cutting (many things)'
- b. kakarukakaruni
 ka -ka -ru-ka -ka -ru-Ø -ni
 cut-cut-p2-cut-cut-p2 -Ind.Tp-Ind.S1/2s
 'I habitually cut (many things)'

⁶ The /ru/ at the end of verb root /karu/ is not the plural marker *-ru*.

- (39) a. ayaruni
 aya -ru-∅ -ni
 throw-p2 -Ind.Tp-Ind.S1/2s
 'I am throwing (many things)'
- b. ayaruayaruni
 aya -ru-aya -ru-∅ -ni
 throw-p2 -throw-p -Ind.Tp-Ind.S1/2s
 'I habitually throw (many things)'
- (40) a. tataruni
 ta -ta -ru-∅ -ni
 plant-plant-p2 -Ind.Tp-Ind.S1/2s
 'I am planting (many things)'
- b. tatarutataruni
 ta -ta -ru-ta -ta -ru-∅ -ni
 plant-plant-p2-plant-plant-p2-Ind.Tp-Ind.S1/2s
 'I habitually plant (many things)'

1.2.3.2.3 Reduplication of a Suffix

Reduplication of suffixes indicates a habitual or repeated action (the distinction depending on context)⁷. In Ogea, only endocentric suffixes are reduplicated. In the examples below, an object suffix is reduplicated (41), an manner suffix is reduplicated (42), and a benefactive suffix is reduplicated (43).

- (41) a. letuni
 le -tu -∅ -ni
 speak-O3s-Ind.Tp-Ind.S1/2s
 'I am speaking to him'
- b. letutuni
 le -tu -tu -∅ -ni
 speak-O3s-O3s-Ind.Tp-Ind.S1/2s
 'I spoke repeatedly to him'

⁷ In Ogea, reduplication is but one of two mechanisms to encode habitual action. When it occurs in final verbs, the suffix *-g-* also indicates habitual action. The difference between the use of *-g-* versus reduplication for habitual action is not known, though it may be a matter of focus. For example, reduplicating an object suffix may serve to focus attention on the recipient of the action rather than the actor.

- (42) a. isideni
isi -de -Ø -ni
hear-well-Ind.Tp-Ind.S1/2s
'I am hearing well'
- b. isidedeni
isi -de -de -Ø -ni
hear-well-well-Ind.Tp-Ind.S1/2s
'I always hear well'
- (43) a. huurufuni
huuru-fu -Ø -ni
send -B3s-Ind.Tp-Ind.S1/2s
'I send for him'
- b. huurufufuni
huuru-fu -fu -Ø -ni
send -B3s-B3s-Ind.Tp-Ind.S1/2s
'I always send for him'

1.2.3.3 Verb Endocenter

The structure of the Ogea endocenter is shown in Table 3. As discussed above, endocentric morphemes include the verb stem and various suffixes that may occur between the stem and the verb exocenter. Endocentric suffixes apply solely to the clause containing the verb stem being suffixed.

Table 3 Ogea Verb Endocenter

Morpheme Class	Endocenter						
	Stem		Plurality	Object	Emphatic	Manner	Benefactive
	Root	Causative					
Order Class	0	1	2	3	4	5	6
Required?	Yes	No	No	No	No	No	No
Values	Verb Roots Adjectives Postpositional (bagu-)	-mu-	-gana- -ru-	Table 10	-nga-	Table 11	Table 12

Examples of the causative suffix used with adjectives were given in (30) through (33), and an example of the postpositional *bagu* was given in **Error! Reference source not found.** Examples of the causative suffix used with verb roots are given in (44) and

(45), below. Examples of the plural suffix *-ru-* were given in (33), (34), and (35).

Examples of the plural suffix *-gana-* are given below in (46) and (47).

- (44) heigemuwai
 heige -mu -wa -i
 appear-causative-Ind.Trp-Ind.S3s
 'He/she/it caused (something) to appear'
- (45) usumudu
 usu -mu -du
 finish-causative-TS
 'He/she/it caused it to finish'
- (46) bologanawai
 bolo-gana-wa -i
 full-p1 -Ind.Trp-Ind.S3s
 'It filled them'
- (47) dageganafuwai
 dage -gana-fu -wa -i
 bound-p1 -B3s-Ind.Trp-Ind.S3s
 'He/she bound them for him/her'

The difference between *-ru-* and *-gana-* is not known at present, except that *-gana-* does not suffix stems created by reduplicating the initial CV of a root, whereas *-ru-* does. Note that *-gana-* is glossed as *p1* (plural type one) and *-ru-* as *p2* (plural type two). Examples of the emphatic suffix were given in (28) and (29). In the case of suffix values that have a table number instead of an actual value, examples of the suffixes follow the specified table.

1.2.3.4 Verb Exocenters

As stated in the introduction to this appendix, Ogea verb exocenters include suffixes that encode inter-clausal temporality, tense, mood, subject, and switch reference. They are termed exocentric because they may contain two suffixes that relate to the clause that follows (i.e., inter-clausal temporality and switch reference relate the current clause to the one that follows it). Ogea verb exocenters fall into the following categories: indicative, contrafactual, subjunctive, and infinitive. Each is discussed in turn, below.

1.2.3.4.1 Indicative Exocenters

The indicative exocenters are broken into two major categories: sentence *medial* and sentence *final* forms. The medial exocenters are further categorized into *same* and *different* referents.

The medial exocenters include temporality suffixes. Temporality suffixes are medial verb suffixes that indicate a temporal relationship between the current clause and the one following it. Actions can occur at the same time (temporal overlap) or in sequence (temporal succession). When comparing the tables, below, for the indicative endocenters, note the following. Temporal succession is *marked* for same referent verbs and *unmarked* for different referent verbs. Temporal overlap is *marked* for different referent verbs and *unmarked* for same referent verbs. Also note that the different referent temporal overlap suffix *-g-* can appear as a suffix on final verbs, in which case it indicates habitual action rather than temporal overlap.

1.2.3.4.1.1 Medial, Same Referent, Indicative Exocenter

The medial, same referent, indicative exocenter has but one suffix. The presence of the succession temporality suffix *-du* indicates that the action or state of the verb suffixed by *-du* is completed before the action or state of the subsequent verb. The absence of the succession temporality suffix *-du*, i.e. the occurrence of the zero morpheme $-\emptyset$, indicates that the action of the verb overlaps in time with the action of the subsequent verb.

Table 4 Ogea Medial, Same Referent, Indicative Verb Exocenter

	Exocenter	
Suffix Class	Temporality	
Order Class	7	
Required?	Yes	
Values	Succession	-du
	Overlap	-∅

- (48) no matane awadu fagawai
no matane awa -du faga -wa -i
he/she/it village leave-TS run.away-Ind.Trp-Ind.S3s
'Having left the village, he/she/it ran away.'
- (49) no nere beele beneu leniridu
no nere beele be -neu le -niri-du
he/she/it them words this-like speak-O3d -TS
- bona huurunirai
bo -na huuru-nir-a -i
vAuxR-vAux.S3s send -O3d-Ind.Trp-Ind.S3s
- 'Having said this to them, he/she/it sent them off.'
- (50) no jaidu bona
no jai -du bo -na
he/she/it get.up-TS vAuxR-vAux.S3s
- otodu golowai
oto -du golo-wa -i
stand-TS walk-Ind.Trp-Ind.S3s
- 'Having stood up, he/she/it walked.'
- (51) no yafa bona lewai
no yafa-∅ bo -na le -wa -i
he/she/it sit-TO vAuxR-vAux.S3s speak-Ind.Trp-Ind.S3s
'As he/she/it sat, he/she/it spoke.'
- (52) no fele bona lewai
no fele-∅ bo -na le -wa -i
he/she/it watch-TO vAuxR-vAux.S3s speak-Ind.Trp-Ind.S3s
'As he/she/it watched, he/she/it spoke.'

Note that medial, same referent verbs are often followed by the verbal auxiliary

bo-, discussed in a separate section, below.

1.2.3.4.1.2 Medial, Different Referent, Indicative Exocenter

Whereas the medial, same referent, indicative exocenter is compact, the medial, different referent, indicative exocenter is the opposite. Also, the medial, different referent, indicative exocenter contains suffixes that overlap with those of the final, indicative exocenter. However, the two exocenters differ in two ways. A different meaning is encoded by the initial suffix *-g-*, and the final suffixes differ. The initial suffix *-g-* encodes temporality in the case of the medial, indicative exocenter, but encodes habituality in the case of the final, indicative exocenter. The suffix *-nga* is the final suffix of the medial, same referent, indicative exocenter, and the suffix *-ha* is the final suffix of the final, indicative exocenter. Therefore, except for the final suffix, both the medial, different referent, indicative exocenter, and the final, indicative exocenter have the same form.

The presence of the succession temporality suffix zero morpheme indicates that the action or state of the verb is completed before the action or state of the subsequent verb. See (55) for an example. The succession temporality suffix *-g-* indicates that the action or state of the verb overlaps in time with the action or state of the subsequent verb. An example of overlap in state rather than action is seen in (51).

Note that for the medial, same referent exocenter, the switch referent suffix is mandatory.

Table 5 Ogea Medial, Different Referent, Indicative Exocenter

Exocenter					
Suffix Class	Temporality		Tense	Subject	Switch Referent
Order Class	7		8	9	11
Required?	Yes		Yes	Yes	Yes
Values	Succession	-∅-	Table 16	Table 17	-nga
	Overlap	-g-			

(53) no onou lenigigainga
 no onou le -nigi-g -a -i -nga
 he thus speak-O3p -TO-Ind.Trp.Tip/Tp-Ind.S3s-SR

fai baingaro medu bode
 fai baingaro me -du bo -de
 man many come-TS vAuxR-vAux.S3p

fai waha bagu otowane
 fai waha bagu oto -wa -ne
 man that with stand-Ind.Trp-Ind.S3p

'While he spoke thus to them, many men came and stood with that man.'

(54) ere nere awanigigeyenga
 ere nere awa -nigi-g -e -ye -nga
 we they leave-O3p -TO-Ind.Trp-Ind.S1p-SR

nere ewe onou inyiinyide
 nere ewe onou inyi-inyi-∅ -de
 they idle like.that be -be -Ind.Tp-Ind.S3p

'They remain idly abandoned by us.'

(55) nere no huurutuwanenga
 nere no huuru-tu -∅ -wa -ne -nga
 they he/she/it send -O3s-TS-Ind.Trp/Tip/Tp-Ind.S3p-SR

matane langa minai
 matane la-nga min -a -i
 village at-int go.down-Ind.Trp-Ind.S3s

'They sent him/her/it, and he/she/it went down to the village.'

In (53), the verb for 'speak' (in the first clause of the sentence) is marked for switch reference, since the referent of the first clause (an individual man) is different than the referents of the subsequent clauses (a group of men).

1.2.3.4.1.3 Final, Indicative Exocenter

The final, indicative exocenter is used with a verb occurring in the final clause of a sentence. However, the final form verb may occur in a non-sentence final position if it is part of a sentence embedded as a relative clause⁸. It is none-the-less the final verb of the embedded sentence. An example of this is seen in (58).

Table 6 Ogea Final, Indicative Exocenter

	Exocenter			
Suffix Class	Habituality	Tense	Subject	Perfective
Order Class	7	8	9	10
Required?	No	Yes	Yes	No
Values	-g-	Table 16	Table 17	-ha

An example of the habitual suffix is (56) and of the perfective suffix is (57).

(56) ou hogo fere fere gaigai
 ou hogo fere fere gaigai
 heavy face different different always

ere bagu heigegai
 ere bagu heige-g -a -i
 we with appear-Hab-Ind.Trp-Ind.S3s

'Various kinds of problems repeatedly happened to us.'

(57) abitimituwaneha
 abiti -mu -tu -wa -ne -ha
 show -causative-O3s-Ind.Trp/Tip-Ind.S3s-Perf
 'They have already caused (it to be) shown to him/her'

⁸Structurally, Ogea relative clauses are formed by embedding a sentence into the head noun slot of a definite noun phrase. See Colburn [2].

(58) fai dada nere nage owonege la
 fai dada nere nage owo-nege la
 man thing they themselves hand-P3p with

tafade waha
 tafa-∅ -de wa -ha
 put -Ind.Tp-Ind.S3p that-perf

ada mogetude
 ada moge -tu -∅ -de
 not turn.back.on-O3s-Ind.Tp-Ind.S3p

'Men do not turn their backs on that which they themselves
 have made.'

1.2.3.4.2 Contrafactual Exocenter

The use of contrafactual suffixes indicates the speaker's belief that an event did not actually occur.

Table 7 Ogea Contrafactual Verb Exocenter

Exocenter					
Suffix Class	Temporality		Subject	Contrafactual	Switch Reference
Order Class	7		8	10	11
Required?	Yes		Yes	Yes	No
Values	Succession	-∅-	Table 19	-ge-	-nga
	Overlap	-g-			

Following are examples of Ogea contrafactual clauses:

(59) aririmuhagage
 ariri -mu -hag-a -ge
 deflect-causative-B1p-Contra.2/3S/d/p-Contra
 'if they had caused it to be deflected for us...'

(60) ji yafegenga
 ji yaf-∅ -e -ge -nga
 I sit-TS-Contra.S1s-Contra-SR
 'if I had sat...'

1.2.3.4.3 Subjunctive Exocenter

The term 'subjunctive' is used loosely here. Traditional usage of the term subjunctive contrasts it with the imperative and other moods. In Ogea, the subjunctive

and imperative are not differentiated. The suffixes of the Ogea subjunctive exocenter (see Table 8) encode the deontic mood (obligation and permission⁹) as well as the predictive. As such, the subjunctive suffixes are translated by a variety of English modal auxiliaries (e.g., must, might, will). In the absence of the predictive and intensifier suffixes, the subjunctive person and number suffixes encode the deontic mood (i.e., *I must, you must, he/she must*, etc.). When used with second person subject suffixes, the deontic mood encodes what would in English be called the imperative mood.

Table 8 Ogea Subjunctive Exocenter

Morpheme Class	Exocenter			
	Subject Person	Subject Number	Predictive	Intensifier
Order Class	8	9	10	11
Required?	Yes	Yes	No	No
Value	Table 20	Table 21	-ga-	-nga

By adding the predictive suffix, the speaker is stating a belief that an event is likely to happen. This assertion may be further strengthened by the addition of the intensifier suffix. (In this author's original analysis, *-ga-* was analyzed as encoding apprehension. Apprehension is not contradictory to prediction, but does not capture the full usage of *-ga-*. Also, when clearly indicating apprehension, the word *gamaji* 'lest' will occur in the clause.) Below are examples of Ogea subjunctive clauses. More examples are provided with Table 20 and Table 21.

- (61) yafadewaguganga
 yafa-de -wa -gu -ga -nga
 sit -well-Sj.S2-Sj.Sp-pred-int

⁹ In and of themselves, the Ogea subjunctive suffixes do not encode permission. A clause encoding permission must have both a verb with subjunctive suffixes and the word *dabu* 'may'.

boni leni
 bo -ni le -Ø -ni
 vAuxR-vAux.S1s speak-Ind.Tp-Ind.S1s

'I say this that you might have a good life'

(62) tere huwanyatege ngalengawaguga
 tere huwanya-tege ngalenga-wa -gu -ga
 you.plural inside -P3p true -Sj.S2-Sj.Sp-pred

'that you might believe...'

1.2.3.4.4 Infinitive Exocenter

The term 'infinitive' is used in this author's analysis to refer to verbs whose exocenter only encodes subject number (singular, dual, and plural). Such verbs lack suffixes for temporal succession, tense, switch reference, or subject person. Examples are given following Table 9 and Table 22.

Table 9 Ogea Infinitive Exocenter

Morpheme Class	Exocenter	
	Infinitive	Subject
Order Class	8	9
Required?	Yes	Yes
Value	-e-	Table 22

(63) umeruwegu uwau
 ume-ru-we -gu uwau
 die-p2-Inf-Inf.Sp without
 'without dying'

(64) tere ji awahiyegu uuwa
 tere ji awa -hiy-e -gu uuwa
 you.plural I leave-O1s-Inf-Inf.Sp not
 'you did not leave me'

- (65) haifuwegu nomo bona
 hai -fu -we -gu nomo bo -na
 tell-B3s-Inf-Inf.Sp its vAuxR-vAux.S3s
 'in order to tell on his behalf'
- (66) goloweru nomo edo uuwa
 golo-we -ru nomo edo uuwa
 walk-Inf-Inf.Sd its able not
 'they (dual) were not able to walk'
- (67) yeregei nomo bode
 yereg-e -i nomo bo -de
 write-Inf-Inf.Ss its vAuxR-vAux.S3p
 'in order that they could write'
- (68) lalanigei nomo bona
 lala -nig-e -i nomo bo -na
 light-O3p-Inf-Inf.Ss its vAuxR-vAux.S3s
 'in order to give light to them'

Note that each of the above examples with *bo-* can be transformed to encode an inclusive plural subject. This is done by 1) use of the first person plural subject pronoun, 2) omission of *bo-*, and 3) ending the sentence with an infinitive verb plus *nomo*. For example, (69), a modification of (68), encodes an inclusive plural, whereas (70) encodes an exclusive plural.

- (69) ere lalanigei nomo
 ere lala -nig-e -i nomo
 we light-O3p-Inf-Inf.Ss its
 'Let us (inclusive) give light to them'
- (70) ere lalanigoya
 ere lala -nig-o -ya
 we light -O3p-Ind.Tf-Ind.S1p
 'We(exclusive) will give light to them'

1.2.3.5 Verbal Suffixes

1.2.3.5.1 Object Suffixes

Table 10 Ogea Object Suffixes

		Number		
		Singular	Dual	Plural
Person	1 st	-hi-	-hiri-	-higi-
	2 nd	-hini-	-tiri-	-tigi-
	3 rd	-tu-	-niri-	-nigi-

There is some evidence that the underlying form of the first person singular object suffix *-hi-* is actually *-hii-*. This underlying form is hypothesized based on morphophonemic Rule 2 (discussed above) and as shown in example (71), where */-hii-/* becomes */-hiy-/*. Also, with the exception of third person singular (which often has irregular forms in Ogea), if the underlying form of *-hi-* is actually *-hii-*, then there is a rule to explain the relationship between the object suffixes and the benefactive suffixes. Namely, */Ci/ → /Ca/*. For these reasons, the underlying form of the first person singular object suffix *-hi-* is hypothesized to be *-hii-*. Examples (71) through (80) are paradigmatic in that they all use the same root *le-*, and show the object suffix encoding indirect objects. Examples (81) and (82) show the object suffix encoding direct objects.

(71) lehiyai
 le- + -hii- + -a- + -i → lehiyai
 speak- 01s- -Ind.Trp- -Ind.S3s
 'He/she spoke to me'

(72) lehide
 le -hii-∅ -de
 speak-01s-Ind.Tp-Ind.S3p
 'They are speaking to me'

- (73) lehinina
 le -hini-Ø -na
 speak-O2s -Ind.Tp-Ind.S3s
 'he/she is speaking to you (sing)'
- (74) letugai
 le -tu -g -a -i
 speak-O2s-Hab-Ind.Trp-Ind.S3s
 'he/she habitually speaks to him/her'
- (75) lehiri
 le -hiri-Ø
 speak-O1d -TO
 'while speaking to us (dual)...
- (76) letirinaha
 le -tiri-Ø -na -ha
 speak-O2d -Ind.Tp-Ind.S3s-perf
 'he/she has spoken to you (dual)'
- (77) lenirona
 le -nir-o -na
 speak-O3d-Ind.Tf-Ind.S3s
 'he/she will speak to them (dual)'
- (78) lehigona
 le -hig-o -na
 speak-O1p-Ind.Tf-Ind.S3s
 'he/she will speak to us (plural)'
- (79) letigini
 le -tigi-Ø -ni
 speak-O2s -Ind.Tp-Ind.S1/2s
 'I am speaking to you (plural)'
- (80) lenigini
 le -nigi-Ø -ni
 speak-O2s -Ind.Tp-Ind.S1/2s
 'I am speaking to them (plural)'
- (81) nuutuwona
 nuu -tu -wo -na
 shove-O3s-Ind.Tf-Ind.S3s
 'he/she will shove it'
- (82) hoyotudu
 hoyo-tu -du
 help-O3s-TS
 'having helped him/her...'

1.2.3.5.2 Manner Suffixes

Ogea manner suffixes modify verbs by stating the scope of an action—whether the action was complete in its effect, was done well, or done quickly.

Table 11 Ogea Manner Suffixes

Meaning	Suffix
completely	-boro-
well	-de-
quickly	-ya-

- (83) ayaborowai
 aya -boro -wa -i
 throw-completely-Ind.Trp-Ind.S3s
 'he/she threw (all of it) away'
- (84) fiideganenga
 fii -de -g -a -ne -nga
 smooth-well-TO-Ind.Trp-Ind.S3p-SR
 'while they smoothed (it out) well...'
- (85) owonege ayayadu
 owo -nege aya -ya -du
 hand-P3p throw-quickly-TS
 'having quickly moved their hands...'

1.2.3.5.3 Benefactive Suffixes

Table 12 Ogea Benefactive Suffixes

		Number		
		Singular	Dual	Plural
Person	1 st	-hai-	-hari-	-hagi-
	2 nd	-hani-	-tari-	-tagi-
	3 rd	-fu-	-nari-	-nagi-

As noted above in the discussion of Ogea object suffixes, there appears to be a systematic relationship between the object suffixes and the benefactive suffixes.

Assuming that the first person singular object suffix *-hi-* has an underlying form of */-hii-/*,

then for all persons except third person singular, the initial /Ci/ becomes /Ca/ to encode benefaction. For third person singular, the initial /t/ becomes /f/.

Below are examples of Ogea benefactive suffixes. In the corpora available to this author, there were no examples of first or second person dual benefactive suffixes.

- (86) haiyahaidu
 haiya -hai-du
 prepare-B1s-TS
 'having prepared it for me...'
- (87) uuwamuhaninaha
 uuwa-mu -hani-Ø -na -ha
 not -causative-B2s -Ind.Tp-Ind.S3s-perf
 'he/she has caused it to cease on your behalf'
- (88) uuwamufuwai
 uuwa-mu -fu -wa -i
 not -causative-B3s-Ind.Trp-Ind.S3s
 'he/she caused (it) to cease on his/her behalf'
- (89) fuutunaridu
 fuutu-nari-du
 untie-B3d -TS
 'having untied (it) for them (dual)...'
- (90) jaimuhagihagina
 jai -mu -hagi-hagi-Ø -na
 get.up-causative-B1p -B1p -Ind.Tp-Ind.S3s
 'he/she habitually raises (it up) for us (plural)'
- (91) feutagidu
 feu -tagi-du
 pour.out-B2p -TS
 'having poured (it) out for you (plural)...'
- (92) aawanagidu
 aawa -nagi-du
 leave-B3p -TS
 'having left (it) for them (plural)...'

In Ogea, the benefactive suffixes are used both to encode benefaction and malefaction. The action performed may be for the good or the harm of an individual. An example of a malefactive usage is (93) (from a text in the Erima dialect of Ogea).

- (93) gaigu tehaita
 gaigu te -hai-Ø -ta
 thigh get-B1s-Ind.Tip-Ind.S3s
 'it has gotten (my) thigh for me'

The context of (93) is as follows. A man is trapped in a cave filling with water. He is in immediate danger of drowning. He yells to someone outside that the water has gotten his thigh, meaning that the water has come up to his thigh. In this context, the suffix *-hai-* clearly has a malefactive meaning.

1.2.3.5.4 Indicative Verb Remote Past Tense Suffixes

The remote past refers to actions that occurred before today, and the immediate past tense refers to actions that occurred today.

Table 13 Ogea Remote Past Tense Suffixes

Person	Suffix
1 st and 2 nd	-e-
3 rd	-a-

- (94) ne ji huuruhiyenenga
 ne ji huuru-hii-e -ne -nga
 you.sing I sent -O1s-Ind.Trp/Tip/Tp-Ind.S1/2s-SR
 meene
 me -e -ne
 come.down-Ind.Trp-Ind.1/2s
 'You sent me and I came down.'
- (95) ere nere awanigigeyenga
 ere nere awa -nigi-g -e -ye -nga
 we they leave-O3p -TO-Ind.Trp-Ind.S1/2p-SR
 nere ewe onou inyiinyide
 nere ewe onou inyi-inyi-Ø -de
 they idle thus be -be -Ind.Tp-Ind.S3p
 'We left them, and (now) they are always idle.'

(104) goyane
 goy-a -ne
 goy-Ind.Trp/Tip-Ind.S3p
 'they (plural) went'

1.2.3.5.5 Indicative Verb Immediate Past and Present Tense Suffixes

In Ogea, the immediate past tense refers to actions that occurred today. However, there are times in Ogea discourse where the immediate past tense is used in reference to events that occurred in the distant past. This appears to be a stylistic usage to more fully engage the listener in the narrative.

The present tense not only encodes an action occurring at the present point of time, but is also used when encoding a habitual action that occurred not only in the past, but continues to occur in the present, as shown in (111).

Table 14 Ogea Immediate Past and Present Tense Suffixes

Person	Number	Sentence Medial		Sentence Final	
		Immediate Past and Present		Immediate Past	Present
		Temporal Overlap	Temporal Succession		
1 st & 2 nd	All	-e-	-∅-		
3 rd	Singular	-a-			
	Dual				
	Plural				

(105) nere soonigiganga
 nere soo -nigi-g -a -∅ -nga
 they chase-03p -TO-Ind.Tip-Ind.S3s-SR

ureya
 ure -∅ -ya
 see.03s-Ind.Tp-Ind.S1/2p

'We see them chase them (away).'

(106) ne aita waha
 ne aita wa -ha
 you.sing woman that-pertaining.to

awatuninga
 awa -tu -Ø -ni -nga
 allow-03s-Ind.Tip/Tp-Ind.S1/2s-SR

ne ngare yafana
 ne ngare yafa-Ø -na
 you.sing with.dual sit -Ind.Tp-Ind.S3s

'You are allowing that woman to be with you.'

- (107) ere fai beha
 ere fai be-ha
 we man this-pertaining.to

onotuyarenga golona
 ono -tu -Ø -yare -nga golo-Ø -na
 do.thus-03s-Ind.Tip/Tp-Ind.S1/2d-SR go -Ind.Tp-Ind.S3s

'We did thus to this man and he is walking.'

- (108) no ji haruwe waha
 no ji haruwe wa -ha
 he/she I work that-pertaining.to

hiyanga
 hii -a -Ø -nga
 give.01s-Ind.Tip/Tp-Ind.S3s-SR

ji haruwe waha teni
 ji haruwe wa-ha te -Ø -ni
 I work that.pertaining.to take-Ind.Tip/pT-Ind.S1/2s

'I do the work he/she gives me.'

- (109) tere taate boya nere
 tere taate bo -ya nere
 you.plural what vAuxR-vAux.S1/2p they

awanigiyanga nere
 awa -nigi-Ø -ya -nga nere
 allow-03p -Ind.Tip/Tp-Ind.S1/2p-SR they

inyabamutigegu uwa
 inyaba-mu -tig-e -gu uwa
 bad -causative-02p-Inf-Inf.Sp not

'Why did you not allow them to harm you?' or
 'Why do you not allow them to harm you?'

- (110) ji uulufuni
 ji uulu -fu -Ø -ni
 I peel.off-B3s-Ind.Tp-Ind.S1/2s
 'I peel (it) off for he/she/it'

- (111) ne ada tafaniginigini
 ne ada tafa-nigi-nigi-Ø -ni
 you.sing not put -O3p -O3p -Ind.Tp-Ind.S1/2s
 'You do not habitually put them'

1.2.3.5.6 Future Tense Suffixes

Ogea future tense suffixes are shown in Table 15 and examples are provided below. Two things should be noted about Ogea final form future tense verbs. First, for the future tense, second person subjects are encoded using the subjunctive forms, so final form future tense suffixes are not applicable to the indicative future tense paradigm. However, they are shown anyway to capture the insight that 2nd person medial and final tenses are marked identically. Second, the 1st and 3rd person subject suffixes have allomorphic variants that optionally occur when following a future tense final form suffix, and when preceding the verbal auxiliary *bo-* 'intent'. The variants are optional in that there is variation both within a single speaker's speech and across speakers. The variation may be stylistic. The variants are as follows:

- (112) lewoni + boni → lewon boni (loss of final vowel)
 le -o -ni bo -ni
 speak-Ind.Tf-Ind.S1s vAuxR-vAux.S1s
 'I will speak'
- (113) lewona + bona → lewon bona (loss of final vowel)
 le -o -na bo -na
 speak-Ind.Tf-Ind.S3s vAuxR-vAux.S3s
 'He/she will speak'
- (114) lewoyare + boyare → lewo boyare (loss of suffix)
 le -o -yare bo -yare
 speak-Ind.Tf-Ind.S1d vAuxR-vAux.S1d
 'We.dual will speak'
- (115) lewodere + bodere → lewod bodere (loss of final VCV)
 le -o -dere bo -dere
 speak-Ind.Tf-Ind.S3d vAuxR-vAux.S3d
 'They.dual will come'

(116) lewoya + boya → lewo boya (loss of suffix)
 le -o -ya bo -ya
 speak-Ind.Tf-Ind.S1p vAuxR-vAux.S1p
 'We.plural will speak'

(117) lewode + bode → lewod bode (loss of final vowel)
 le -o -de + bo -de
 speak-Ind.Tf-Ind.S3p vAuxR-vAux.S3p
 'They.plural will speak'

Note that in some cases the final V or the final VCV is lost, and in other cases the entire subject suffix is lost. Also note that the loss of either the entire subject suffix or the final V or final VCV results in ambiguity for a morphological parser that cannot look ahead to the next word to disambiguate the subject suffix. For example, in (126), below, the subject for the verb 'give' is ambiguous between 1st and 3rd person unless information can be obtained by looking ahead to the next word, or by obtaining information from a syntactic parser that identifies the 1st person pronoun *ji* as being the grammatical subject. Note that these variants provide the second exception to the general rule in Ogea that syllables are open rather than closed. In the examples below, future tense verbs are shown with both the normal subject suffix, and the forms that optionally occur when preceding the verbal auxiliary *bo-*.

Table 15 Ogea Future Tense Suffixes

Person	Sentence Position	
	Medial	Final
1 st	-e-	-o-
2 nd	-a ¹⁰	
3 rd	-o-	

¹⁰ Technically the 2nd person future sentence final tense does not belong in a table for the indicative future tenses. However, it is shown here to capture the insight that both the indicative 2nd person medial future tense and the 2nd person subjunctive tense are marked identically.

- (118) ji no huurutuwehenga
 ji no huuru-tu -e -he -nga
 I he/she/it send -O3s-Ind.Tf-Ind.S1s-SR
- tere bagu mona bona
 tere bagu m -o -na bo -na
 you with come.down-Ind.Tf-Ind.S3s vAuxR-vAux.S3s
 'I will send him/her/it to you.plural and
 he/she/it will come'
- (119) ne fasadu mayahenga
 ne fasadu mai -a -he -nga
 you quickly come.up-Ind.Tf-Ind.S2s-SR
- minoyare
 min -o -yare
 go.down-Ind.Tf-Ind.S1/2d
- 'You come up quickly and we will go down.'
- (120) ere megebu ayarudu waredenga
 ere megebu aya -ru-du war -e -de -nga
 we stone throw-p2-TS hit.O3s-Ind.Tf-Ind.S1p-SR
- umona bona
 um -o -na bo -na
 die-Ind.Tf-Ind.S3s vAuxR-vAux.S3s
- 'We will throw stones and hit him/her and he/she will die.'
- (121) tere nere fai ila adai awanigadenga
 tere nere fai ila adai awa -nig-a -de -nga
 you.plural they man some do.not allow-O3p-Ind.Tf-Ind.S2p-SR
- nere yaawatigode
 nere yaawa-tig-o -de
 they lie -O2p-Ind.Tf-Ind.S3p
- 'Do not allow any men to deceive to you.'
- (122) ne taate dada bonahe gau
 ne taate dada bo -nahe gau
 you.plural what thing vAuxR-vAux.vAux.Tf.S2s desire
- hinonga ji
 hin -o -Ø -nga ji
 give.O2s-Ind.Tf-Ind.S3s-SR I
- hinon boni
 hin -o -n bo -ni
 give.O2s-Ind.Tf-Ind.S1/3s vAuxR-vAux.S1s
- 'Whatever you desire, I will give you.'

- (123) nere haruwe usumuworonga
 nere haruwe usu -mu -wo -ro -nga
 they work finish-causative-Ind.Tf-Ind.S3d-SR
- kaafa kui ete no mai heigedu bona
 kaafa kui ete no mai heige -du bo -na
 animal wild one he/she/it come.up arrive-TS vAuxR-vAux.S3s
- nere fai waha bagu
 nere fai wa -ha bagu
 they man that-pertaining.to with.plural
- warawona bona
 wara -wo -na bo -na
 fight-Ind.Tf-Ind.S3s vAuxR-vAux.S3s
- 'After they.dual have finished working, a wild animal will
 come up and fight with them.dual.'
- (124) nere ji yarogonga
 nere ji yar -o -go -nga
 they I hit.O1s-Ind.Tf-Ind.S3p-SR
- umon boni
 um -o -n bo -ni
 die-Ind.Tf-Ind.S1/3s vAuxR-vAux.S1s
- 'They will strike me dead.'
- (125) awatuwoni
 awa -tu -wo -ni
 leave-O3s-Ind.Tf-Ind.S1s
 'I will leave him/her/it'
- (126) ji ne hinon boni
 ji ne hin -o -n bo -ni
 I you.sing give.O2s-Ind.Tf-Ind.S1/3s vAuxR-vAux.S1s
 'I will give (it) to you'
- (127) no yafona
 no yaf-o -na
 he sit-Ind.Tf-Ind.S3s
 'He will sit'
- (128) lenigon bona
 le -nig-o -n bo -na
 speak-O3p-Ind.Tf-Ind.S1/3s vAuxR-vAux.S3s
 'he/she/it will speak to them'
- (129) minoyare
 min -o -yare
 go.down-Ind.Tf-Ind.S1d
 'We.dual will go down'

- (130) letigo boyare
 le-tig -o -Ø bo -yare
 speak-O2p-Ind.Tf-Ind.S1d/p vAuxR-vAux.S1d
 'We.dual will speak to you.plural'
- (131) umeruwodere
 ume-ru-wo -dere
 die-p2 -Ind.Tf-Ind.S3d
 'They.dual will die'
- (132) heigod bodere
 heig -o -d bo -dere
 appear-Ind.Tf-Ind.S3d/p vAuxR-vAux.S3d
 'They.dual will appear'
- (133) nyoya
 ny -o -ya
 eat-Ind.Tf-Ind.S1p
 'We will eat'
- (134) nyo boya
 ny -o -Ø bo -ya
 eat-Ind.Tf-Ind.S1d/p vAuxR-vAux.S1p
 'We will eat'
- (135) tafatuwode
 tafa-tu -wo -de
 put -O3s-Ind.Tf-Ind.S3s
 'They will put him/her/it'
- (136) tod bode
 t -o -d bo -de
 get-Ind.Tf-Ind.S3d/p vAuxR-vAux.S3p
 'They will get (it)'

1.2.3.5.7 Summary of Indicative Verb Tense Suffixes

This section provides a summary table for the Ogea indicative tense suffixes. The abbreviations follow the standard ones used throughout this appendix: Remote Past Tense (RPT), Immediate Past Tense (IPT), Present Tense (PT), Future Tense (FT), Temporal Overlap (TO), and Temporal Succession (TS). Sentence Medial forms are for those involving switch reference. Medial forms for the same referent were given above in section *1.2.3.4.1.1 Medial, Same Referent, Indicative Exocenter*.

Table 16 Ogea Indicative Tense Suffixes - Summary

Person	Number	Sentence Medial						Sentence Final			
		RPT		IPT & PT		FT		RPT	IPT	PT	FT
		TO	TS	TO	TS	TO	TS				
1 st	All	-e-		-∅-		-e-		-∅-		-o-	
2 nd						-a-				-a- ¹¹	
3 rd	Singular	-a-		-∅-		-o-		-a-		-o-	
	Dual					-o-				-o-	
	Plural					-o-				-o-	

1.2.3.5.8 Indicative Verb Subject Suffixes

Below is a table for the Ogea indicative verb subject suffixes. The abbreviations used in the table are the same ones used throughout the appendix and described in the previous section. The sentence final subject suffixes shown for the future tense do not apply to the 2nd person because 2nd person future suffixes are encoded by subjunctive forms (see Table 8). Note that in both sentence medial and final positions the remote past tense subject suffixes are identical.

Table 17 Ogea Indicative Verb Subject Suffixes

Person	Number	Sentence Medial						Sentence Final			
		RPT		IPT & PT		FT		RPT	IPT	PT	FT ¹²
		TO	TS	TO	TS	TO	TS				
1 st & 2 nd	Singular	-ne- ¹³				-he-		-ne-	-na-	-ni-	
	Dual	-re-		-yare-		-dere-		-re-	-yare-		
	Plural	-ye-		-ya-		-de-		-ye-	-ya-		
3 rd	Singular	-i-		-∅-				-i-	-∅-	-na-	
	Dual	-re-				-ro-		-re-		-dere-	
	Plural	-ne-				-go-		-ne-		-de-	

¹¹ Technically the 2nd person future sentence final tense does not belong in a table for the indicative tenses. However, it is shown here to capture the insight that both the indicative 2nd person medial future tense and the 2nd person subjunctive tense are marked identically.

¹² 2nd person does not apply to sentence final indicative future tense. The subjunctive forms are used instead: -a-, -ru-, -gu-.

Examples of indicative verb subject suffixes occur throughout the appendix. The examples given with the various indicative tenses above also provide examples of each subject suffix in combination with each tense suffix.

1.2.3.5.9 Summary of Indicative Verb Tense and Subject Suffixes

By now the reader may have noticed that in isolation, there often is ambiguity in the Ogea tense suffixes. For example, the tense suffix *-e-* is used for both the remote past (in the case of 1st and 2nd person), and in the future (in the case of 1st person medial verbs). In isolation, the indicative verb subject suffixes also have some ambiguity. When the tense and subject suffixes are combined, some, but not all, ambiguity is resolved. For the reader's convenience, a summary table is presented below.

Table 18 Ogea Indicative Verb Tense and Subject Suffixes Combined

Prsn	Nmbr	Sentence Medial						Sentence Final			
		RPT		IPT & PT		FT ¹⁴		RPT	IPT	PT	FT ¹⁵
		TO	TS	TO	TS	TO	TS				
1 st & 2 nd	Sing	-e-ne-		-∅-ne-		-e-he- -a-he-		-e-ne-	-∅-na-	-∅-ni-	-o-ni-
	Dual	-e-re-		-∅-yare-		-e-dere- -a-dere-		-e-re-	-∅-yare-		-o-yare-
	Plural	-e-ye-		-∅-ya-		-e-de- -a-de-		-e-ye-	-∅-ya-		-o-ya-
3 rd	Sing	-a-i-	-a-∅-	-∅-∅-	-o-∅-		-a-i-	-∅-∅-	-∅-na-	-o-na-	
	Dual	-a-re-			-o-ro-			-a-re-		-∅-dere-	-o-dere-
	Plural	-a-ne-			-o-go-			-a-ne-		-∅-de-	-o-de-

Note that the sentence medial subject affixes are always followed by *-nga*, the switch referent suffix. This resolves the ambiguity between the two occurrences of *-∅-*

¹³ Some occurrences of *-ni-* have been observed for medial immediate past or present tense. It is not clear if this is speaker dependent variation, or has significance.

¹⁴ *-e-* is used with 1st person, and *-a-* with 2nd person.

¹⁵ The sentence final indicative future tenses and subjects do not apply to the 2nd person. The subjunctive tense (*-a-*), and the subjunctive subjects (*-a-*, *-ru-*, *-gu-*) are used instead.

yare- and *-Ø-Ø-*. Otherwise the medial occurrence of *-Ø-yare-* would be incorrectly identified as encoding either immediate past or present tense (when it really only encodes present tense in the medial position) and the same for the medial occurrence of *-Ø-Ø-*. When the tense suffixes are combined with the subject suffixes, the only true ambiguity is *-Ø-na-*, which is ambiguous in that it encodes both sentence final 1st and 2nd person singular immediate past tense and sentence final 3rd person singular present tense.

1.2.3.5.10 Contrafactual Subject Suffixes

Contrafaction was described above in section 1.2.3.4.2 Contrafactual Exocenter. Below is a table of the Ogea contrafactual subject suffixes and some examples.

Table 19 Ogea Contrafactual Subject Suffixes

Person	Number	Suffix
1 st	Singular	-e-
	Dual	-edere-
	Plural	-ede-
2 nd , 3 rd	All	-a-

- (137) *ji yafege*
ji yaf-e -ge
 I sit-Contra.S1s-Contra
 'if I had sat'
- (138) *ere yafederege*
ere yaf-edere -ge
 we sit-Contra.S1d-Contra
 'if we (dual) had sat'
- (139) *ere yafedege*
ere yaf-edede -ge
 we sit-Contra.S1p-Contra
 'if we (dual) had sat'
- (140) *yafage*
yaf-a -ge
 sit-Contra.S2/3s/d/p-Contra
 'if you/he/she/they (sing/dual/plural) had sat'

1.2.3.5.11 Subjunctive Person and Number Suffixes

The Ogea subjunctive was described above in section 1.2.3.4.3 Subjunctive Exocenter. Below is a table of the subjunctive verb person suffixes, a table of the subjunctive verb number suffixes, and some examples. Note that subjunctive tenses and subjects do not distinguish between medial and final forms.

Table 20 Ogea Subjunctive Verb Person Suffixes

Person	Suffix
1 st and 3 rd	-o-
2 nd	-a-

Table 21 Ogea Subjunctive Verb Number Suffixes

Person	Number	Suffix
1 st	All	-u
	Singular	
	2 nd & 3 rd	Dual
	Plural	-gu

The following examples of subjunctive suffixes were elicited from a native Ogea speaker. Note that number is undifferentiated in the first person, and that first person uses the singular form for singular, dual, and plural.

(141) ji yafouganga
 ji yaf-o -u -ga -nga
 I sit-Sj.S1/3-Sj.Ss-cert-int
 'I certainly will sit'

(142) ere yaforuganga
 ere yaf-o -ru -ga -nga
 we sit-Sj.S1/3-Sj.Sd-cert-int
 'We (dual) certainly will sit'

(143) ere yafoguganga
 ere yaf-o -gu -ga -nga
 we sit-Sj.S1/3-Sj.Sp-cert-int
 'We (plural) certainly will sit'

- (144) ne yafauganga
 ne yaf-a -u -ga -nga
 you sit-Sj.S2-Sj.Ss-cert-int
 'you (sing) certainly will sit'
- (145) tere yafaruganga
 tere yaf-a -ru -ga -nga
 you sit-Sj.S2-Sj.Sd-cert-int
 'you (dual) certainly will sit'
- (146) tere yafaguganga
 tere yaf-a -gu -ga -nga
 you sit-Sj.S2-Sj.Sp-cert-int
 'you (plural) certainly will sit'
- (147) no yafouganga
 no yaf-o -u -ga -nga
 he/she/it sit-Sj.S3-Sj.Ss-cert-int
 'he/she/it certainly will sit'
- (148) nere yaforuganga
 nere yaf-o -ru -ga -nga
 they sit-Sj.S3-Sj.Sd-cert-int
 'they(dual) certainly will sit'
- (149) nere yafoguganga
 nere yaf-o -gu -ga -nga
 they sit-Sj.S3-Sj.Sd-cert-int
 'they (plural) certainly will sit'

1.2.3.5.12 Infinitive Verb Number Suffixes

Infinitive verbs were discussed in section 1.2.3.4.4 Infinitive Exocenter. Below are a table and examples of infinitive verb number suffixes.

Table 22 Ogea Infinitive Verb Number Suffixes

Number	Suffix
Singular	-i
Dual	-ru
Plural	-gu

- (150) isei uuwa
 is -e -i uuwa
 hear-Inf-Inf.Ss not
 'I/you.sing/he/she did not hear'

- (151) yaurenigei nomo
yaure-nig -e -i nomo
yell.to-03p-Inf-Inf.Ss its
'in order for I/you.sing/he/she to yell to them'
- (152) umugeru uuwa
umug-e -ru uuwa
fear-Inf-Inf.Sd not
'we/you/they.dual were not afraid'
- (153) yafegu uuwa
yaf-e -gu uuwa
sit-Inf-Inf.Sp not
'we/you/they.plural did not sit'
- (154) haawewegu nomo edo uuwa
haawe-we -gu nomo edo uuwa
judge-Inf-Inf.Sp its able not
'we/you/they.plural are not able to judge'
- (155) haawenigegu nomo boya
haawe-nig-e -gu nomo bo -ya
judge-03p-Inf-Inf.Sp its vAuxR-vAux.S1/2p
'in order for we/you.plural to judge them'

1.2.3.6 Suppletive Verb Roots

Ogea has a limited number of suppletive verb roots that vary according to the object person and number. Suppletive forms exist for the verbs meaning 'to see', 'to hit', and 'to get'. It may be argued that a fourth set exists, namely for the verb 'to give', which is formed by using the object suffixes as verb roots. These four verb roots are the only known example of suppletion in Ogea. A paradigm is presented as a separate table for each suppletive verb (Table 23 through Table 26), and then together as a single matrix for comparative purposes in Table 27. Note that there is a relationship between certain segments within each paradigm and across paradigms. This will be discussed in section **1.2.7 Templatic Morphology**, where a comparison will also be made to various pronouns and pronominal suffixes.

Table 23 Suppletive Verb Root Forms for 'See'

		Object Number		
		Singular	Dual	Plural
Object Person	1 st	yeriye-	herire-	herige-
	2 nd	neriye-	terire-	terige-
	3 rd	ure-	nerire-	nerige-

Table 24 Suppletive Verb Root Forms for 'Hit'

		Object Number		
		Singular	Dual	Plural
Object Person	1 st	yari-	harire-	harige-
	2 nd	nari-	tarire-	tarige-
	3 rd	wari-	narire-	narige-

Table 25 Suppletive Verb Root Forms for 'Get'

		Object Number		
		Singular	Dual	Plural
Object Person	1 st	haife-	haire-	haige-
	2 nd	haine-	hatire-	hatige-
	3 rd	hau-	hanire-	hanige-

Table 26 Suppletive Verb Root Forms for 'Give'

		Object Number		
		Singular	Dual	Plural
Object Person	1 st	hi-	hiri-	higi-
	2 nd	hini-	tiri-	tigi-
	3 rd	tu-	niri-	nigi-

Table 27 Comparative Matrix for All Suppletive Verb Roots

		Meaning	Object Number		
			Singular	Dual	Plural
Object Person	1 st	see	yeriye-	herire-	herige-
		hit	yari-	harire-	harige-
		get	haife-	haire-	haige-
		give	hi-	hiri-	higi-
	2 nd	see	neriye-	terire-	terige-
		hit	nari-	tarire-	tarige-
		get	haine-	hatire-	hatige-
		give	hini-	tiri-	tigi-
	3 rd	see	ure-	nerire-	nerige-
		hit	wari-	narire-	narige-
		get	hau-	hanire-	hanige-
		give	tu-	niri-	nigi-

1.2.4 The Auxiliary Verb *bo-*

The auxiliary verb *bo-*¹⁶ occurs both in sentence medial and sentence final positions. In the medial position it sometimes clearly encodes purpose, but in other cases it appears to act as a functor, encoding strictly grammatical rather than semantic information. In sentence final position, *bo-* semantically encodes what appears to be intent or purpose. Note that the exact meaning and function of *bo-* in a sentence final position is not known by this author and is difficult for Ogea speakers to explain. One Ogea speaker suggested that at one time Ogea differentiated immediate future and remote future, and that sentence final *bo-* encoded remote future. Note that *bo-* is suffixed for subject person and number. For all tenses, persons, and number except 2nd person future, the root *bo-* takes present tense indicative subject suffixes. In the examples below, *bo-* is

¹⁶ A speaker dependent variant of *bo-* is *abo-*.

'I desire that you will certainly listen!"
Literally, 'Desire is given to me that you will
certainly listen!'

(159) no bagu goidu bona
no bagu goi-du bo -na
he/she with go -TS vAuxR-vAux.S3s

letuwai
le -tu -wa -i
speak-03s-Ind.Trp-Ind.S3s

'He/she went to (him/her) and spoke to him/her.'

(160) isokinigona bona
isoki-nig-o -na bo -na
ask -03p-Ind.Tf-Ind.S3s vAuxR-vAux.S3s

'In order to ask them...'

(161) onodu bona no
ono -du bo -na, no
do.thus-SR vAuxR-vAux.S3s he/she/it

nomo bemu umai
nomo bemu um -a -i
his/her sibling die-Ind.Trp-Ind.S3s

'Having done thus, his/her sibling died.'

(162) ne haumu beha
ne haumu be -ha
you.sing place this-pertaining.to

awadu bonahe
awa -du bo -nahe
leave-TS vAuxR-vAux.Tf.S2s,

ha fere langa goyau
ha fere la-nga goy-a -u
place different at-int go -Sj.S2-Sj.Ss

'Leave this place and go to a different place!'

(163) ne huwanyate waha bonahe
ne huwanya-te wa -ha bo -nahe
you.sing insides-2sP that-pertaining.to vAuxR-vAux.Tf.S2s

adai ou hinou
adai ou hin -ou
not heavy give.02s-Nom

'Do not be concerned about that!'

(164) tere maidu bonadere uraru
 tere mai -du bo -nadere ur -a -ru
 you.plural come-TS vAuxR-vAux.Tf.S2d see.O3s-Sj.S2-Sj.Sd

'Come and see he/she/it!'

(165) tere dada beha bonade
 tere dada be -ha bo -nade
 you.dual/plural thing this-pertaining.to vAuxR-vAux.Tf.S2p

ninatege isagu
 nina-tege is -a -gu
 thoughts-P2p hear-Sj.S2-Sj.Sp

'Think about this thing!'

1.2.5 Nouns and Adjectives

1.2.5.1 Inalienably Possessed Nouns

In some Papuan languages (e.g., Hua [4]) all noun roots or stems may be suffixed with a possessive suffix. However, in Ogea the possessive suffixes attach only to inalienably possessed nouns such as body parts and kinship terms. Normally, even though an inalienably possessed noun is suffixed with a possessive suffix, the possessed noun will occur as part of a noun phrase, with the possessed noun preceded by a possessive pronoun as in (166).

(166) yame baife
 yame bai -fe
 my sibling.same.sex-P1s
 'my sibling.same.sex'

(167) nebere afenege
 nebere afe -nege
 their leg/foot-P3p
 'their legs'

The possessive suffixes are shown in Table 29.

Table 29 Ogea Possessive Suffixes

Person	Singular	Dual	Plural
1 st	-fe	-re	-ge
2 nd	-ne ~ -te ¹⁸	-tere	-tege
3 rd	-Ø or -u	-nere	-nege

1.2.5.2 Derived Nouns

The nominalizing suffix *-ou* is used to derive nouns from verbs in Ogea. The nominalized verbs are formed by replacing the exocenter with the nominalizing suffix, with the exception of the temporality suffix, which may be retained. This means that nominalized verbs may be fully inflected for object, benefaction, aspect, etc.

In Ogea, adjectives follow nouns. In some cases, the nominalized verb follows the noun, and appears to be functioning as an adjective. However, note that in the majority of cases, the nominalized verb precedes a noun, and thus fills the first noun slot of a compound noun phrase. In such cases the nominalized verb appears to be in an attributive relationship to the head noun. Also note that Ogea forms relative clauses by filling the head noun slot of a definite noun phrase, all the while retaining the full inflection of the exocenter. In example (168), note that (a) shows an independent clause, and (b) shows the same clause filling the head noun slot of a definite noun phrase, thus forming a restrictive relative clause. In the same example, (c) and (d) show the same verb nominalized with *-ou*. Note that in the data available to this author, for the nominalized form of 'die', (c) is the predominant pattern, with the pattern shown in (d)

¹⁸ This is a speaker dependent variation. Most Ogea are at a minimum tri-lingual (Ogea, Tok Pisin, and one other language), and some speakers of Ogea have suggested the variation in the second singular possessive suffix is due to influence by the Uya language.

sometimes occurring. That is, 'umou' typically follows the head noun (as in (c)) rather than follows it (as in (d)). However, for the vast majority of examples of other nominalized verbs in the data (that is, verbs other than 'to die'), the nominalized verb precedes the head noun. Finally, note that relative clause formation as shown in (b) is very productive in Ogea, and is the basis for a variety of semantic relationships between clauses in Ogea (e.g., topicalization, conditions, hypotheticality).

- (168) a. fai umai (an independent clause)
 fai um-a -i
 man die-Ind.Trp-Ind.S3s
 '(A) man died.'
- b. fai umai waha (a relative clause)
 fai um -a -i wa -ha
 man die-Ind.Trp-Ind.S3s that-pertaining.to
 'the man who died'
- c. fai umou waha (a nominalized verb following
 fai um -ou wa -ha the head noun)
 man die-Nom that-pertaining.to
 'that dead man'
- d. umou fai waha (a nominalized verb preceding
 um -ou fai wa -ha the head noun)
 die-Nom man that-pertaining.to
 'that dead man'
- (169) a. fai no isina
 no isi -∅ -na
 he/she understand-Ind.Tp-Ind.S3s
 'he/she understands'
- b. fai no isina waha
 fai no isi-∅ -na wa -ha
 man he understand-Ind.Tp-Ind.S3s that-pertaining.to
 'the man who understands'
- c. isou fai waha
 is-ou fai wa -ha
 understanding-Nom man that-pertaining.to
 'the understanding man'

- d. no isou hiloo bainga bagu
 no is -ou hiloo bai -nga bagu
 he/she understand-nom good very-int with
 'He/she has very good understanding'
- (170) hiiri katiwou menekele
 hiiri kati -wou menekele
 ocean encircle-Nom small
 'small enclosed ocean' (= a lake)
- (171) ne nebere agotetenigigou anyakaro
 ne nebere agotete-nigi-g -ou anyakaro
 you their teach -O3p -Hab-Nom large
 'You are their leading teacher.'
- (172) jirigou mata
 jiri -g -ou mata
 sleep.pl-Hab-Nom house
 'sleeping house' or 'bunkhouse'
- (173) a. oofanigigai
 oofa -nigi-g -a -i
 watch.over-O3p -Hab-Ind.Trp-Ind.S3s
 'he/she habitually watched over'
- b. oofanigigou fai
 oofa -nigi-g -ou fai
 watch.over-O3p -Hab-Nom man
 '(a) man (who) habitually watches over them'
 or 'a watchman'
- (174) a. kurutuna
 kuru -tu -Ø -na
 look.for-O3p-Ind.Tp-Ind.S3s
 'he/she is looking for him/her'
- b. kurutuwou haruwe
 kuru -tu -wou haruwe
 look.for-O3s-Nom work
 '(the) work of looking for (something)'
- (175) a. isiwai
 isi -wa -i
 know-Ind.Trp-Ind.S3s
 'he/she knew'
- b. nomo isou
 nomo is -ou
 his/her know-Nom
 'his/her knowledge'

- (176) a. yeregai
 yereg-a -i
 write-Ind.Trp-Ind.S3s
 'he/she wrote'
- b. nomo yeregou
 nomo yereg-ou
 his/her write-Nom
 'his/her writing'

The suffix *-ou* is also used with adjectives. The motivation for this is not understood by this author. Examples are shown below. In each of the examples, the same phrase would be grammatically correct if *-ou* were deleted. In each case, *-ou* seems to indicate that the noun has the quality of the adjective suffixed by *-ou*.

- (177) ogola giiriwou
 ogola giiri-wou
 cloth red -attribute
 'a red cloth'
- (178) kesebu mutu kobibiwou
 kesebu mutu kobibi -wou
 knife nose crooked-attribute
 'a knife with a crooked point'
- (179) fai tiginiwou
 fai tigini -wou
 man straight-attribute
 'a good (obedient) man'

Finally, note that a nominalized verb is ambiguous with the first and third singular subjunctive. For example, *lewou* may legitimately be parsed as either a nominalized noun ('speaker') or as a 1st or 3rd person subjunctive ('I/he/she/it will speak').

1.2.6 Pronouns

Ogea pronouns are not suffixed and therefore are not properly part of a morphological discussion of Ogea. However, because the Ogea pronouns are discussed in the section below on Ogea templatic morphology, tables are provided below for each type of Ogea pronoun. It should be noted that Ogea pronouns do not encode gender.

1.2.6.1 Personal Pronouns

The same forms of Ogea personal pronouns are used to encode both grammatical subjects and objects. Translations include *I, me, you, he, she, it, they, and them.*

Table 30 Ogea Personal Pronouns

		Number	
		Singular	Dual/Plural
Person	1 st	ji	ere
	2 nd	ne	tere
	3 rd	no	nera

1.2.6.2 Intensive/Reflexive Pronouns

The intensive/reflexive pronouns are translated into English as *myself, yourself, himself, herself, itself, and themselves.*

Table 31 Ogea Intensive/Reflexive Pronouns

		Number		
		Singular	Dual	Plural
Person	1 st	yage	hare	hage
	2 nd	nage	tare	tage
	3 rd	nogo	nare	nage

1.2.6.3 Possessive Pronouns

The possessive pronouns are translated into English as *my, your, his, her, its, and their.*

Table 32 Ogea Possessive Pronouns

		Number		
		Singular	Dual	Plural
Person	1 st	yame	heire	ebere
	2 nd	name	teire	tebere
	3 rd	nomo	neire	nebere

1.2.7 Templatic Morphology

It was noted in [1] that a relationship exists between various forms of the Ogea suppletive verbs. That such a relationship exists between not only the suppletive verbs but also the pronouns and a number of suffix types was not formally noted. Furthermore, no attempt was made to describe these relationships. This section, therefore, attempts to describe the relationships. While the description of such relationships likely has no pragmatic use in the natural language processing of Ogea, such a description is of interest for historical (diachronic) and comparative purposes. That is, the relationships may prove useful in studying the history of the Ogea language, and are certainly useful in comparing Ogea to other related languages such as Siroi (as will be discussed below). Also, these relationships are used elsewhere in this dissertation to illustrate the ability of DATR to capture linguistic generalizations not possible to capture in AMPLE's LKR.

The approach used in the analysis presented here is a templatic one. For convenience, the various tables presented above for suppletive verb roots, various suffixes, and the pronouns are presented as a single table below (Table 33). Examination of the forms for 1st, 2nd, and 3rd person in the dual and plural columns shows an interesting phenomenon. A basic template is evident that has the shape CV(CV)CV, where the initial consonant is a segment

Table 33 Various Ogea Verb Roots, Suffixes, and Pronouns

		Meaning	Number		
			Singular	Dual	Plural
Person	1 st	see	yeriye-	herire-	herige-
		hit	yari-	harire-	harige-
		get	haife-	ha ire-	h aige-
		give	hi-	h iri-	h igi-
		Object	-hi-	-h iri-	-h igi-
		Benefactive	-hai-	-h ari-	-h agi-
		Possessive Suffix	-fe	-re	-ge
		Reflexive Pronoun	yage	h are	h age
		Possessive Pronoun	yame	h eire	ebere
		Personal Pronoun	ji	ere	ere
	2 nd	see	neriye-	terire-	terige-
		hit	nari-	tarire-	tarige-
		get	haine-	hat ire-	hat ige-
		give	hini-	t iri-	t igi-
		Object	-hini-	-t iri-	-t igi-
		Benefactive	-hani-	-t ari-	-t agi-
		Possessive suffix	-ne	-t ere	-t ege
		Reflexive Pronoun	nage	t are	t age
		Possessive Pronoun	name	te ire	tebere
		Personal Pronoun	ne	t ere	t ere
	3 rd	see	ure-	nerire-	nerige-
		hit	wari-	narire-	narige-
		get	hau-	han ire-	han ige-
		give	tu-	n iri-	n igi-
		Object	-tu-	-n iri-	-n igi-
		Benefactive	-fu-	-n ari-	-n agi-
		Possessive Suffix	-mu, -0	-n ere	-n ege
		Reflexive Pronoun	nogo	n are	n age
		Possessive Pronoun	nomo	n eire	nebere
		Personal Pronoun	no	n ere	n ere

that indicates number (dual and plural), the middle V(CV) encodes the core meaning, and the final consonant is a segment that indicates person (1st, 2nd, 3rd). While the current Ogea surface forms contain exceptions to the postulated template, historically the forms may have been regular for all lexemes involved.

Basic generalizations about the dual and plural forms will now be made, followed by a discussion of the exceptions. First, the initial consonant is either /h/, /t/, or /n/, with

/h/ indicating 1st person, /t/ indicating 2nd person, and /n/ indicating 3rd person. Note that it is possible to analyze the initial consonant as a single segment prefix indicating person. However, given that Ogea is an otherwise suffixing language, and that the person segment occurs as the initial segment of certain suffixes, it seems more productive to this author to use a templatic approach to the analysis rather than postulate a prefix that attaches to certain suffixes. The second generalization that may be made is that the final consonant indicates number, with /r/ indicating dual and /g/ indicating plural. However, if the lexeme does not distinguish between dual and plural, /r/ is used to encode both dual and plural (e.g., see the entries for the plural possessive pronouns and personal pronouns). The third observation is that the final vowel is /i/ for morphemes that are suffixes, and /e/ for morphemes that are verb roots, pronouns, or noun suffixes. Thus, the underlying template is {h,t,n}V(CV){r,g}V. This template is presented as a paradigm based on person and number in Table 34.

Table 34 Paradigmatic Presentation of Person/Number Template

		Number	
		Dual	Plural
Person	First	hV(CV)rV	hV(CV)gV
	Second	tV(CV)rV	tV(CV)gV
	Third	nV(CV)rV	nV(CV)gV

Now discussion will turn to exceptions to the basic template. The exceptions include absence of the initial person consonant segment, the presence of a CV before the initial person consonant segment, the absence of an initial CV, and the use of /r/ to indicate both dual and plural.

The initial person consonant segment is missing in the 1st person dual and plural forms for the possessive suffix and the personal pronoun, and in the 1st person plural form of the possessive pronoun. It is, of course, entirely possible that historically the initial /h/ was present, but has been lost. Historically the 1st person dual and plural pronoun may have been /here/.

In the case of the suppletive verb root for 'get' (a person), in the 2nd and 3rd person dual and plural forms, /ha/ precedes the person consonant segment. The segmentation of the verb for 'get' is problematic for the 1st person dual and plural. Historically was the dual form /hahaire/ and the plural form /hahaige/? Was the initial /ha/ lost, or the second?

Regarding the 1st person possessive suffix, the problem is not the presence of a syllable before the person segment, but rather the opposite. No initial syllable occurs before the one indicating number. That is, rather than /-hare/ for dual, and /-hage/ for plural, the forms are /-re/ and /-ge/.

The last exception to the overall template is seen in the forms for the possessive and personal pronoun for all persons in the dual and plural. Rather than the expected /g/ segment occurring as the final consonant, /r/ occurs. When the distinction is only between singular and plural rather than between singular, dual, and plural, the /r/ is used to encode both dual and plural.

A number of interesting questions arise from the data presented here, the answers to which are not known to this author. First, why is it that the first person forms appear to lack the regularity exhibited by the dual and plural forms in terms of a consonant segment encoding person, and a consonant segment encoding number? Second, why do

the suppletive verb roots exist at all? They are the exception to an overwhelming majority of verb roots that merely encode the basic verb meaning, and leave the encoding of object person and number to separate suffixes. Are the suppletive verb roots the result of borrowing from other languages? If they were borrowed, why do they have a systematic relationship to the pronouns and possessive suffix? Or, historically did these particular verbs (see, hit, get, give) encode actions considered of more importance than others? Third, do languages related to Ogea also exhibit such a template as the one described above? This may be the case.

As for the third question, this author has access to data for only one closely related language, namely Siroi, described by Wells [7]. This data indicates that at least one related language has such a template. Wells provides paradigms for the Siroi personal and possessive pronouns and the object suffixes (combined into a single table below), but does not provide a paradigm for benefactive suffixes or reflexive pronouns¹⁹. Also, there is no mention made of suppletive verb roots that vary according to object person in Siroi.

Table 35 Siroi Object Suffixes, Possessive, and Personal Pronouns

		Meaning	Number		
			Singular	Dual	Plural
Person	1 st	Object	-y-	-sik-	-sing-
		Possessive Pronoun	yipe	sikile	singine
		Personal Pronoun	ye	s ile	s ine
	2 nd	Object	-n-	-tik-	-ting-
		Possessive Pronoun	nape	takile	tangine
		Personal Pronoun	ne	t ale	t ane
	3 rd	Object	-Ø-	-nik-	-ning-
		Possessive Pronoun	nupe	nakile	nangine
		Personal Pronoun	nu	n ale	n ane

¹⁹ Per Sjaak and Jacqueline van Kleef (SIL), personal communication, Siroi does not have benefactive suffixes or reflexive pronouns.

In examining Table 35, as with Ogea, in Siroi a pattern may be observed in the dual and plural forms. However, in the case of Siroi, although there is a clear initial segment indicating person (/s/ for 1st, /t/ for 2nd, and /n/ for 3rd), there is not as clear a pattern for a final number consonant segment as there is with Ogea. Either /k/ or /l/ or both may occur for duals, and /ng/ or /n/ or both may occur for plurals in the Siroi data. It is also interesting to note that both Ogea and Siroi share the same segments for the 2nd and 3rd person forms. Both use /t/ to indicate 2nd person, and both use /n/ to indicate 3rd person.

1.3 Summary

This appendix has described Ogea morphophonemics and morphotactics at a level sufficient for the reader to understand discussion of the AMPLE-based and DATR-based lexicons found in other appendices. In addition, this appendix has discussed the relationship between various suppletive verbs, pronouns, and possessive suffixes, and hypothesized an underlying template to describe these relationships.

1.4 References

- [1] M. Colburn, "Erima Grammar Essentials," Summer Institute of Linguistics, Unpublished Manuscript. Ukarumpa, Papua New Guinea 1979.
- [2] M. Colburn, "The Functions and Meanings of the Erima Deictic Articles," *Pacific Linguistics A-69, Papers in New Guinea Linguistics*, vol. 23, pp. 209-272, 1984.
- [3] B. F. Grimes and J. E. Grimes, "Ethnologue: Language Family Index," . Dallas, TX: Summer Institute of Linguistics, 1997.
- [4] J. Haiman, "Hua (Papuan)," in *The Handbook of Morphology*, A. Spencer and A. M. Zwicky, Eds. Oxford: Blackwell Publishers, Ltd., 1998, pp. 539-562.
- [5] N. N. Mikloucho-Maclay, *Mikloucho-Maclay: New Guinea Diaries 1871-1883*. Madang, Papua New Guinea: Kristen Press, 1975.
- [6] P. Staalesen, *Clause Relationships in Iatmul*, vol. A-31:45-69. Canberra: Australian National University, 1972.
- [7] M. A. Wells, *Siroi Grammar*, vol. B-51. Canberra: Australian National University, 1979.

APPENDIX 2

OGEA AMPLE LEXICON

A2.1 Description

This appendix provides a listing of the Ogea AMPLE lexicon file *Ogea.db*. The database file uses a "unified" dictionary, with each record type identifying the entry as either a root ('r') or suffix ('s'). The unique identifier for each record is the morpheme name (\g). In addition to the type field and the morpheme name field, five additional AMPLE field codes were required to properly parse Ogea data. The field codes have the following meaning:

Code	Meaning
\a	allomorph
\c	category
\g	morphname
\ge	English gloss
\mp	morpheme property
\t	type
\u	underlying form

Note that in addition to lexicon (dictionary) database files, AMPLE uses other files, including control files. These files are not presented here since they would not be maintained and generated from DATR. The records entries listed provide the data required to parse all examples found in Appendix 1.

A2.2 The Listing

Filename: ogea.db.

Note: line numbers added for reference only. Not in actual file.

(180) \g able
(181) \ge able
(182) \type r
(183) \a edo
(184) \u edo
(185) \c Adv
(186)
(187) \g allow/leave
(188) \ge allow/leave
(189) \type r
(190) \a awa
(191) \a aw / _ [V]
(192) \u awa
(193) \mp MC1
(194) \c VR
(195)
(196) \g always
(197) \ge always
(198) \type r
(199) \a gaigai
(200) \u gaigai
(201) \c Adv
(202)
(203) \g animal
(204) \ge animal
(205) \type r
(206) \a kaafa
(207) \u kaafa
(208) \c NW
(209)
(210) \g appear/arrive
(211) \ge appear/arrive
(212) \type r
(213) \a heige
(214) \a heig / _ [V]
(215) \u heige
(216) \mp MC1
(217) \c VR
(218)
(219) \g ask
(220) \ge ask
(221) \type r
(222) \a isoki
(223) \u isoki
(224) \mp MC2
(225) \c VR
(226)
(227) \g at
(228) \ge at
(229) \type r
(230) \a la
(231) \u la
(232) \c Prep
(233)
(234) \g Attrib
(235) \ge Attributive marker
(236) \type s

(237) \a ou / [C] _
 (238) \a wou / [V] _
 (239) \u ou
 (240) \o 110
 (241) \c Adj/Atr
 (242)
 (243) \g B1d
 (244) \ge for us (dual)
 (245) \type s
 (246) \a hari
 (247) \a har / _ [V]
 (248) \u hari
 (249) \mp MC1
 (250) \o 60
 (251) \c VR/VR
 (252)
 (253) \g B1p
 (254) \ge for us (plural)
 (255) \type s
 (256) \a hagi
 (257) \a hag / _ [V]
 (258) \u hagi
 (259) \mp MC1
 (260) \o 60
 (261) \c VR/VR
 (262)
 (263) \g B1s
 (264) \ge for me
 (265) \type s
 (266) \a hai
 (267) \a hay / _ [V]
 (268) \u hai
 (269) \mp MC2
 (270) \o 60
 (271) \c VR/VR
 (272)
 (273) \g B2d
 (274) \ge for you (dual)
 (275) \type s
 (276) \a tari
 (277) \a tar / _ [V]
 (278) \u tari
 (279) \mp MC1
 (280) \o 60
 (281) \c VR/VR
 (282)
 (283) \g B2p
 (284) \ge for you (plural)
 (285) \type s
 (286) \a tagi
 (287) \a tag / _ [V]
 (288) \u tagi
 (289) \mp MC1
 (290) \o 60
 (291) \c VR/VR
 (292)
 (293) \g B2s

(294) \ge for you (singular)
 (295) \type s
 (296) \a hani
 (297) \a han / _ [V]
 (298) \u hani
 (299) \mp MC1
 (300) \o 60
 (301) \c VR/VR
 (302)
 (303) \g B3d
 (304) \ge for them (dual)
 (305) \type s
 (306) \a nari
 (307) \a nar / _ [V]
 (308) \u nari
 (309) \mp MC1
 (310) \o 60
 (311) \c VR/VR
 (312)
 (313) \g B3p
 (314) \ge for them (plural)
 (315) \type s
 (316) \a nagi
 (317) \a nag / _ [V]
 (318) \u nagi
 (319) \mp MC1
 (320) \o 60
 (321) \c VR/VR
 (322)
 (323) \g B3s
 (324) \ge for him/her/it
 (325) \type s
 (326) \a fu
 (327) \u fu
 (328) \mp MC2
 (329) \o 60
 (330) \c VR/VR
 (331)
 (332) \g bad
 (333) \ge bad
 (334) \type r
 (335) \a inyaba
 (336) \u inyaba
 (337) \c Adj
 (338)
 (339) \g be
 (340) \ge be
 (341) \type r
 (342) \a inyi
 (343) \a iny / _ [V]
 (344) \u inyi
 (345) \mp MC1
 (346) \c VR
 (347)
 (348) \g become.bad
 (349) \ge become.bad
 (350) \type r

(351) \a inyaba +/ ~_ TO
 (352) \u inyaba_2
 (353) \mp MC2
 (354) \c VR
 (355)
 (356) \g become.crooked
 (357) \ge become.crooked
 (358) \type r
 (359) \a kobibi +/ ~_ TO
 (360) \u kobibi_2
 (361) \mp MC2
 (362) \c VR
 (363)
 (364) \g become.good
 (365) \ge become.good
 (366) \type r
 (367) \a hilou +/ ~_ TO
 (368) \a hiloo +/ ~_ TO
 (369) \u hilou_2
 (370) \mp MC2
 (371) \c VR
 (372)
 (373) \g become.heavy
 (374) \ge become.heavy
 (375) \type r
 (376) \a ou +/ ~_ TO
 (377) \u ou_2
 (378) \mp MC2
 (379) \c VR
 (380)
 (381) \g become.large
 (382) \ge become.large
 (383) \type r
 (384) \a anyakaro +/ ~_ TO
 (385) \u anyakaro_2
 (386) \mp MC2
 (387) \c VR
 (388)
 (389) \g become.many
 (390) \ge become.many
 (391) \type r
 (392) \a baingaro +/ ~_ TO
 (393) \u baingaro_2
 (394) \mp MC2
 (395) \c VR
 (396)
 (397) \g become.nothing
 (398) \ge become.nothing
 (399) \type r
 (400) \a uuwa +/ ~_ TO
 (401) \u uuwa_2
 (402) \mp MC2
 (403) \c VR
 (404)
 (405) \g become.red
 (406) \ge become.red
 (407) \type r

(408) \a giiri +/ ~_ TO
(409) \u giiri_2
(410) \mp MC2
(411) \c VR
(412)
(413) \g become.short
(414) \ge become.short
(415) \type r
(416) \a tutu +/ ~_ TO
(417) \u tutu_2
(418) \mp MC2
(419) \c VR
(420)
(421) \g become.small
(422) \ge become.small
(423) \type r
(424) \a menekele +/ ~_ TO
(425) \a melekene +/ ~_ TO
(426) \u menekele_2
(427) \mp MC2
(428) \c VR
(429)
(430) \g become.straight
(431) \ge become.straight
(432) \type r
(433) \a tigini +/ ~_ TO
(434) \u tigini_2
(435) \mp MC2
(436) \c VR
(437)
(438) \g become.true
(439) \ge become.true
(440) \type r
(441) \a ngalenga +/ ~_ TO
(442) \u ngalenga_2
(443) \mp MC2
(444) \c VR
(445)
(446) \g become.wild
(447) \ge become.wild
(448) \type r
(449) \a kui +/ ~_ TO
(450) \u kui_2
(451) \mp MC2
(452) \c VR
(453)
(454) \g bound
(455) \ge bound
(456) \type r
(457) \a dage
(458) \a dag / _ [V]
(459) \u dage
(460) \mp MC1
(461) \c VR
(462)
(463) \g break.in.two
(464) \ge break.in.two

(465) \type r
(466) \a kou
(467) \a ko
(468) \u kou
(469) \mp MC2 RC1
(470) \c VR
(471)
(472) \g causative
(473) \ge causative
(474) \type s
(475) \a mu
(476) \u mu
(477) \o 10
(478) \c VR/VR
(479)
(480) \g certainty
(481) \ge certainty
(482) \type s
(483) \a nga
(484) \u nga
(485) \o 110
(486) \c VSjS/VSjS
(487)
(488) \g chase
(489) \ge chase
(490) \type r
(491) \a soo
(492) \u soo
(493) \mp MC2
(494) \c VR
(495)
(496) \g cloth
(497) \ge cloth
(498) \type r
(499) \a ogola
(500) \u ogola
(501) \c NW
(502)
(503) \g come.down
(504) \ge come.down
(505) \type r
(506) \a me
(507) \a m / _ [V]
(508) \u me
(509) \mp MC1
(510) \c VR
(511)
(512) \g come.up
(513) \ge come.up
(514) \type r
(515) \a mai
(516) \a may / _ [V]
(517) \u mai
(518) \mp MC2
(519) \c VR
(520)
(521) \g completely

(522) \ge completely
 (523) \type s
 (524) \a boro
 (525) \u boro
 (526) \c VR/VR
 (527)
 (528) \g connect/join
 (529) \ge connect/join
 (530) \type r
 (531) \a bagumu
 (532) \u bagu-mu
 (533) \mp MC2
 (534) \c VR
 (535)
 (536) \g consume
 (537) \ge consume
 (538) \type r
 (539) \a nyi
 (540) \a ny / _ [V]
 (541) \u nyi
 (542) \mp MC1
 (543) \c VR
 (544)
 (545) \g Contra
 (546) \ge Contrafactual
 (547) \type s
 (548) \a ge +/- [Contra.Subj] _
 (549) \u ge
 (550) \o 100
 (551) \c Contra/Contra
 (552)
 (553) \g Contra.S1d
 (554) \ge Contractual first dual subject
 (555) \type s
 (556) \a edere / [C] _ +/- _ Contra
 (557) \a wedere / [V] _ +/- _ Contra
 (558) \u edere
 (559) \o 80
 (560) \c VR/Contra SMV/Contra
 (561)
 (562) \g Contra.S1p
 (563) \ge Contractual first plural subject
 (564) \type s
 (565) \a ede / [C] _ +/- _ Contra
 (566) \a wede / [V] _ +/- _ Contra
 (567) \u ede
 (568) \o 80
 (569) \c VR/Contra SMV/Contra
 (570)
 (571) \g Contra.S1s
 (572) \ge Contractual first singular subject
 (573) \type s
 (574) \a e / [C] _ +/- _ Contra
 (575) \a we / [V] _ +/- _ Contra
 (576) \u e
 (577) \o 80
 (578) \c VR/Contra SMV/Contra

(579)
 (580) \g Contra.S2/3s/d/p
 (581) \ge Contractual second or third singular, dual,
 (582) or plural subject
 (583) \type s
 (584) \a a / [C] _ +/ _ Contra
 (585) \a wa / [V] _ +/ _ Contra
 (586) \u a
 (587) \o 80
 (588) \c VR/Contra SMV/Contra
 (589)
 (590) \g crooked
 (591) \ge crooked
 (592) \type r
 (593) \a kobibi
 (594) \u kobibi
 (595) \c Adj
 (596)
 (597) \g cut
 (598) \ge cut
 (599) \type r
 (600) \a karu
 (601) \a ka
 (602) \u karu
 (603) \mp MC2 RC1
 (604) \c VR
 (605)
 (606) \g deflect
 (607) \ge deflect
 (608) \type r
 (609) \a ariri
 (610) \u ariri
 (611) \mp MC2
 (612) \c VR
 (613)
 (614) \g desire
 (615) \ge desire
 (616) \type r
 (617) \a gau
 (618) \u gau
 (619) \c NW
 (620)
 (621) \g die
 (622) \ge die
 (623) \type r
 (624) \a ume
 (625) \a um / _ [V]
 (626) \u ume
 (627) \mp MC1
 (628) \c VR
 (629)
 (630) \g different
 (631) \ge different
 (632) \type r
 (633) \a fere
 (634) \u fere
 (635) \c Adj

(636)
(637) \g do.not
(638) \ge do.not
(639) \type r
(640) \a adai
(641) \u adai
(642) \c Adv
(643)
(644) \g do.thus
(645) \ge do.thus
(646) \type r
(647) \a ono
(648) \u ono
(649) \mp MC2
(650) \c VR
(651)
(652) \g do.work
(653) \ge do.work
(654) \type r
(655) \a haruwe +/- ~_ TO
(656) \u haruwe_2
(657) \mp MC2
(658) \c VR
(659)
(660) \g dream
(661) \ge dream
(662) \type r
(663) \a wewe
(664) \u wewe
(665) \c NW
(666)
(667) \g encircle
(668) \ge encircle
(669) \type r
(670) \a kati
(671) \u kati
(672) \mp MC2
(673) \c VR
(674)
(675) \g face
(676) \ge face
(677) \type r
(678) \a hogo
(679) \u hogo
(680) \c PNR
(681)
(682) \g fear
(683) \ge fear
(684) \type r
(685) \a umuge
(686) \a umug / _ [V]
(687) \u umuge
(688) \mp MC1
(689) \c VR
(690)
(691) \g fight
(692) \ge fight

(693) \type r
(694) \a wara
(695) \a war / _ [V]
(696) \u wara
(697) \mp MC1
(698) \c VR
(699)
(700) \g fill
(701) \ge fill
(702) \type r
(703) \a bolo
(704) \u bolo
(705) \mp MC2
(706) \c VR
(707)
(708) \g finish
(709) \ge finish
(710) \type r
(711) \a usu
(712) \u usu
(713) \mp MC2
(714) \c VR
(715)
(716) \g fly
(717) \ge fly
(718) \type r
(719) \a fililai
(720) \u fililai
(721) \mp MC2
(722) \c VR
(723)
(724) \g follow
(725) \ge follow
(726) \type r
(727) \a oojo
(728) \u oojo
(729) \mp MC2
(730) \c VR
(731)
(732) \g gather
(733) \ge gather
(734) \type r
(735) \a elege
(736) \a eleg / _ [V]
(737) \u elege
(738) \mp MC1
(739) \c VR
(740)
(741) \g get.Old
(742) \ge get us (dual)
(743) \type r
(744) \a haire
(745) \a hair / _ [V]
(746) \u haire
(747) \mp MC1
(748) \c VR
(749)

(750) \g get.01p
(751) \ge get us (plural)
(752) \type r
(753) \a haige
(754) \a haig / _ [V]
(755) \u haige
(756) \mp MC1
(757) \c VR
(758)
(759) \g get.01s
(760) \ge get me
(761) \type r
(762) \a haife
(763) \a haif / _ [V]
(764) \u haife
(765) \mp MC1
(766) \c VR
(767)
(768) \g get.02d
(769) \ge get you (dual)
(770) \type r
(771) \a hatire
(772) \a hatir / _ [V]
(773) \u hatire
(774) \mp MC1
(775) \c VR
(776)
(777) \g get.02p
(778) \ge get you (plural)
(779) \type r
(780) \a hatige
(781) \a hatig / _ [V]
(782) \u hatige
(783) \mp MC1
(784) \c VR
(785)
(786) \g get.02s
(787) \ge get you (singular)
(788) \type r
(789) \a haine
(790) \a hain / _ [V]
(791) \u haine
(792) \mp MC1
(793) \c VR
(794)
(795) \g get.03d
(796) \ge get them (dual)
(797) \type r
(798) \a hanire
(799) \a hanir / _ [V]
(800) \u hanire
(801) \mp MC1
(802) \c VR
(803)
(804) \g get.03p
(805) \ge get them (plural)
(806) \type r

(807) \a hanige
 (808) \a hanig / _ [V]
 (809) \u hanige
 (810) \mp MC1
 (811) \c VR
 (812)
 (813) \g get.O3s
 (814) \ge get him/her/it
 (815) \type r
 (816) \a hau
 (817) \u hau
 (818) \mp MC2
 (819) \c VR
 (820)
 (821) \g get.up
 (822) \ge get.up
 (823) \type r
 (824) \a jai
 (825) \a jay / _ [V]
 (826) \u jai
 (827) \mp MC2
 (828) \c VR
 (829)
 (830) \g get/take
 (831) \ge get/take
 (832) \type r
 (833) \a te
 (834) \a t / _ [V]
 (835) \u te
 (836) \mp MC1
 (837) \c VR
 (838)
 (839) \g give.O1d
 (840) \ge give us (dual)
 (841) \type r
 (842) \a hiri
 (843) \a hir / _ [V]
 (844) \u hiri
 (845) \mp MC1
 (846) \c VR
 (847)
 (848) \g give.O1p
 (849) \ge give us (plural)
 (850) \type r
 (851) \a hig
 (852) \a hig / _ [V]
 (853) \u hig
 (854) \mp MC1
 (855) \c VR
 (856)
 (857) \g give.O1s
 (858) \ge give me
 (859) \type r
 (860) \a hii
 (861) \a hiy / _ [V]
 (862) \u hii
 (863) \mp MC2

(864) \c VR
(865)
(866) \g give.02d
(867) \ge give you (dual)
(868) \type r
(869) \a tiri
(870) \a tir / _ [V]
(871) \u tiri
(872) \mp MC1
(873) \c VR
(874)
(875) \g give.02p
(876) \ge give you (plural)
(877) \type r
(878) \a tigi
(879) \a tig / _ [V]
(880) \u tigi
(881) \mp MC1
(882) \c VR
(883)
(884) \g give.02s
(885) \ge give you (singular)
(886) \type r
(887) \a hini
(888) \a hin / _ [V]
(889) \u hini
(890) \mp MC1
(891) \c VR
(892)
(893) \g give.03d
(894) \ge give them (dual)
(895) \type r
(896) \a niri
(897) \a nir / _ [V]
(898) \u niri
(899) \mp MC1
(900) \c VR
(901)
(902) \g give.03p
(903) \ge give them (plural)
(904) \type r
(905) \a nigi
(906) \a nig / _ [V]
(907) \u nigi
(908) \mp MC1
(909) \c VR
(910)
(911) \g give.03s
(912) \ge give him/her/it
(913) \type r
(914) \a tu
(915) \u tu
(916) \mp MC2
(917) \c VR
(918)
(919) \g go
(920) \ge go

(921) \type r
 (922) \a goi
 (923) \a goy / _ [V]
 (924) \u goi
 (925) \mp MC2
 (926) \c VR
 (927)
 (928) \g go.down
 (929) \ge go.down
 (930) \type r
 (931) \a mini
 (932) \a min / _ [V]
 (933) \u mini
 (934) \mp MC1
 (935) \c VR
 (936)
 (937) \g good
 (938) \ge good
 (939) \type r
 (940) \a hilou
 (941) \a hiloo
 (942) \u hilou
 (943) \c Adj
 (944)
 (945) \g Hab
 (946) \ge Habitual
 (947) \type s
 (948) \a g
 (949) \u g
 (950) \o 70
 (951) \c VR/SFV VR/Nom
 (952)
 (953) \g hand
 (954) \ge hand
 (955) \type r
 (956) \a owo
 (957) \u owo
 (958) \c PNR
 (959)
 (960) \g he/she/it
 (961) \ge he/she/it
 (962) \type r
 (963) \a no
 (964) \u no
 (965) \c PerPro
 (966)
 (967) \g hear/understand/obey
 (968) \ge hear/understand/obey
 (969) \type r
 (970) \a isi
 (971) \a is / _ [V]
 (972) \u isi
 (973) \mp MC1
 (974) \c VR
 (975)
 (976) \g heavy
 (977) \ge heavy

(978) \type r
 (979) \a ou
 (980) \u ou
 (981) \c Adj
 (982)
 (983) \g help
 (984) \ge help
 (985) \type r
 (986) \a hoyo
 (987) \u hoyo
 (988) \mp MC2
 (989) \c VR
 (990)
 (991) \g himself/herself/itself
 (992) \ge himself/herself/itself
 (993) \type r
 (994) \a nogo
 (995) \u nogo
 (996) \c RflPro
 (997)
 (998) \g his/her/its
 (999) \ge his/her/its
 (1000) \type r
 (1001) \a nomo
 (1002) \u nomo
 (1003) \c PosPro
 (1004)
 (1005) \g hit.Old
 (1006) \ge hit us (dual)
 (1007) \type r
 (1008) \a harire
 (1009) \a harir / _ [V]
 (1010) \u harire
 (1011) \mp MC1
 (1012) \c VR
 (1013)
 (1014) \g hit.Olp
 (1015) \ge hit us (plural)
 (1016) \type r
 (1017) \a harige
 (1018) \a harig / _ [V]
 (1019) \u harige
 (1020) \mp MC1
 (1021) \c VR
 (1022)
 (1023) \g hit.Ols
 (1024) \ge hit me
 (1025) \type r
 (1026) \a yari
 (1027) \a yar / _ [V]
 (1028) \u yari
 (1029) \mp MC1
 (1030) \c VR
 (1031)
 (1032) \g hit.O2d
 (1033) \ge hit you (dual)
 (1034) \type r

(1035) \a tarire
(1036) \a tarir / _ [V]
(1037) \u tarire
(1038) \mp MC1
(1039) \c VR
(1040)
(1041) \g hit.02p
(1042) \ge hit you (plural)
(1043) \type r
(1044) \a tarige
(1045) \a tarig / _ [V]
(1046) \u tarige
(1047) \mp MC1
(1048) \c VR
(1049)
(1050) \g hit.02s
(1051) \ge hit you (singular)
(1052) \type r
(1053) \a nari
(1054) \a nar / _ [V]
(1055) \u nari
(1056) \mp MC1
(1057) \c VR
(1058)
(1059) \g hit.03d
(1060) \ge hit them (dual)
(1061) \type r
(1062) \a narire
(1063) \a narir / _ [V]
(1064) \u narire
(1065) \mp MC1
(1066) \c VR
(1067)
(1068) \g hit.03p
(1069) \ge hit them (plural)
(1070) \type r
(1071) \a narige
(1072) \a narig / _ [V]
(1073) \u narige
(1074) \mp MC1
(1075) \c VR
(1076)
(1077) \g hit.03s
(1078) \ge hit him/her/it
(1079) \type r
(1080) \a wari
(1081) \a war / _ [V]
(1082) \u wari
(1083) \mp MC1
(1084) \c VR
(1085)
(1086) \g I
(1087) \ge I
(1088) \type r
(1089) \a ji
(1090) \u ji
(1091) \c PerPro

(1092)
(1093) \g idle
(1094) \ge idle
(1095) \type r
(1096) \a ewe
(1097) \u ewe
(1098) \c Adj
(1099)
(1100) \g Ind.S1d
(1101) \ge Indicative first dual subject
(1102) \type s
(1103) \a 0 / o _ # +/ Ind.Tf _ ~SR
(1104) \a dere / e _ +/ Ind.Tf _ SR
(1105) \a re / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(1106) \a yare +/ [Ind.Tip.Tp] _
(1107) \a yare / o _ +/ Ind.Tf _ ~SR
(1108) \u yare
(1109) \o 90
(1110) \c SFV/SFV SFV/SFVW SMV/SMV
(1111)
(1112) \g Ind.S1p
(1113) \ge Indicative first plural subject
(1114) \type s
(1115) \a 0 / o _ # +/ Ind.Tf _ ~SR
(1116) \a de / e _ +/ Ind.Tf _ SR
(1117) \a ya +/ [Ind.Tip.Tp] _
(1118) \a ya / o _ +/ Ind.Tf _ ~SR
(1119) \a ye / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(1120) \u ya
(1121) \o 90
(1122) \c SFV/SFV SFV/SFVW SMV/SMV
(1123)
(1124) \g Ind.S1s
(1125) \ge Indicative first singular subject
(1126) \type s
(1127) \a he / e _ +/ Ind.Tf _ SR
(1128) \a n / o _ # +/ Ind.Tf _ ~SR
(1129) \a na +/ Ind.Tip _
(1130) \a ne +/ [Ind.Tip.Tp] _ SR
(1131) \a ne / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(1132) \a ni +/ Ind.Tp _ ~SR +/ [Ind.Tip.Tp] _ SR
(1133) \a ni / o _ +/ Ind.Tf _ ~SR
(1134) \u ni
(1135) \o 90
(1136) \c SFV/SFV SFV/SFVW SMV/SMV
(1137)
(1138) \g Ind.S2d
(1139) \ge Indicative second dual subject
(1140) \type s
(1141) \a dere / a _ +/ Ind.Tf _ SR
(1142) \a re / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(1143) \a yare +/ [Ind.Tip.Tp] _
(1144) \u yare
(1145) \o 90
(1146) \c SFV/SFV SFV/SFVW SMV/SMV
(1147)
(1148) \g Ind.S2p

```

(1149) \ge Indicative second plural subject
(1150) \type s
(1151) \a de / a _ +/ Ind.Tf _ SR
(1152) \a ya +/ [Ind.Tip.Tp] _
(1153) \a ye / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(1154) \u ya
(1155) \o 90
(1156) \c SFV/SFV SFV/SFVW SMV/SMV
(1157)
(1158) \g Ind.S2s
(1159) \ge Indicative second singular subject
(1160) \type s
(1161) \a he / a _ +/ Ind.Tf _ SR
(1162) \a na +/ Ind.Tip _
(1163) \a ne +/ Ind.Tp _ SR
(1164) \a ne / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(1165) \a ni +/ Ind.Tp _ ~SR +/ [Ind.Tip.Tp] _ SR
(1166) \u ni
(1167) \o 90
(1168) \c SFV/SFV SFV/SFVW SMV/SMV
(1169)
(1170) \g Ind.S3d
(1171) \ge Indicative third dual subject
(1172) \type s
(1173) \a d / o _ # +/ Ind.Tf _ ~SR
(1174) \a dere +/ Ind.Tp _ ~SR
(1175) \a dere / o _ +/ Ind.Tf _ ~SR
(1176) \a re / a _ +/ Ind.Tip _ +/ Ind.Tp _ SR +/ Ind.Trp _
(1177) \a ro / o _ +/ Ind.Tf _ SR
(1178) \u dere
(1179) \o 90
(1180) \c SFV/SFV SFV/SFVW SMV/SMV
(1181)
(1182) \g Ind.S3p
(1183) \ge Indicative third plural subject
(1184) \type s
(1185) \a d / o _ # +/ Ind.Tf _ ~SR
(1186) \a de +/ Ind.Tp _ ~SR
(1187) \a de / o _ +/ Ind.Tf _
(1188) \a go / o _ +/ Ind.Tf _ SR
(1189) \a ne / a _ +/ Ind.Tip _ +/ Ind.Tp _ SR +/ Ind.Trp _
(1190) \u de
(1191) \o 90
(1192) \c SFV/SFV SFV/SFVW SMV/SMV
(1193)
(1194) \g Ind.S3s
(1195) \ge Indicative third singular subject
(1196) \type s
(1197) \a 0 / a _ +/ Ind.Tip _ +/ Ind.Tp _ SR
(1198) \a 0 / o _ +/ Ind.Tf _ SR
(1199) \a i / a _ +/ Ind.Trp _
(1200) \a n / o _ # +/ Ind.Tf _ ~SR
(1201) \a na +/ Ind.Tp _ ~SR
(1202) \a ta +/ Ind.Tip _ ~SR
(1203) \a na / o _ +/ Ind.Tf _ ~SR
(1204) \u na
(1205) \o 90

```

(1206) \c SFV/SFV SFV/SFVW SMV/SMV
 (1207)
 (1208) \g Ind.Tf
 (1209) \ge Indicative future tense
 (1210) \type s
 (1211) \a e / [C] _ +/ _ [Ind.S1]
 (1212) \a we / [V] _ +/ _ [Ind.S1]
 (1213) \a a / [C] _ +/ _ [Ind.S2]
 (1214) \a wa / [V] _ +/ _ [Ind.S2]
 (1215) \a o / [C] _ +/ _ [Ind.S1.S3]
 (1216) \a wo / [V] _ +/ _ [Ind.S1.S3]
 (1217) \u o
 (1218) \o 80
 (1219) \c SMV/SMV VR/SMV VR/SFV SFV/SFV
 (1220)
 (1221) \g Ind.Tip
 (1222) \ge Indicative immediate past tense
 (1223) \type s
 (1224) \a e / [C] _ +/ _ [Ind.S1.S2]
 (1225) \a we / [V] _ +/ _ [Ind.S1.S2]
 (1226) \a a / [C] _ +/ _ [Ind.S3]
 (1227) \a wa / [V] _ +/ _ [Ind.S3]
 (1228) \a 0 +/ _ [Ind.S1.S2] +/ _ Ind.S3s
 (1229) \u a
 (1230) \o 80
 (1231) \c SMV/SMV VR/SMV VR/SFV SFV/SFV
 (1232)
 (1233) \g Ind.Tp
 (1234) \ge Indicative present tense
 (1235) \type s
 (1236) \a 0 +/ _ [Ind.S1.S2.S3]
 (1237) \a e / [C] _ +/ _ [Ind.S1.S2]
 (1238) \a we / [V] _ +/ _ [Ind.S1.S2]
 (1239) \a a / [C] _ +/ _ [Ind.S3]
 (1240) \a wa / [V] _ +/ _ [Ind.S3]
 (1241) \u 0
 (1242) \o 80
 (1243) \c SMV/SMV VR/SMV VR/SFV SFV/SFV
 (1244)
 (1245) \g Ind.Trp
 (1246) \ge Indicative remote past tense
 (1247) \type s
 (1248) \a e / [C] _ +/ _ [Ind.S1.S2]
 (1249) \a we / [V] _ +/ _ [Ind.S1.S2]
 (1250) \a a / [C] _ +/ _ [Ind.S3]
 (1251) \a wa / [V] _ +/ _ [Ind.S3]
 (1252) \u e
 (1253) \o 80
 (1254) \c SMV/SMV VR/SMV VR/SFV SFV/SFV
 (1255)
 (1256) \g Inf
 (1257) \ge Infinitive
 (1258) \type s
 (1259) \a e / [C] _
 (1260) \a we / [V] _
 (1261) \u e
 (1262) \o 80

(1263) \c VR/Inf
 (1264)
 (1265) \g Inf.Sd
 (1266) \ge Infinitive subject
 (1267) \type s
 (1268) \a ru
 (1269) \u ru
 (1270) \o 90
 (1271) \c Inf/Inf
 (1272)
 (1273) \g Inf.Sp
 (1274) \ge Infinitive plural subject
 (1275) \type s
 (1276) \a gu
 (1277) \u gu
 (1278) \o 90
 (1279) \c Inf/Inf
 (1280)
 (1281) \g Inf.Ss
 (1282) \ge Infinitive singular subject
 (1283) \type s
 (1284) \a i
 (1285) \u i
 (1286) \o 90
 (1287) \c Inf/Inf
 (1288)
 (1289) \g inside
 (1290) \ge inside
 (1291) \type r
 (1292) \a huwanya
 (1293) \u huwanya
 (1294) \c PNR
 (1295)
 (1296) \g judge
 (1297) \ge judge
 (1298) \type r
 (1299) \a haawe
 (1300) \u haawe
 (1301) \mp MC2
 (1302) \c VR
 (1303)
 (1304) \g just/only
 (1305) \ge just/only
 (1306) \type s
 (1307) \a nga
 (1308) \u nga
 (1309) \c NW/NW RflPro/RflPro PN/PN Adj/Adj PosPro/PosPro
 (1310) PerPro/PerPro Prep/Prep Deictic/Deictic
 VR/VR
 (1311)
 (1312) \g knife
 (1313) \ge knife
 (1314) \type r
 (1315) \a kesebu
 (1316) \u kesebu
 (1317) \c NW
 (1318)

(1319) \g large
 (1320) \ge large
 (1321) \type r
 (1322) \a anyakaro
 (1323) \u anyakaro
 (1324) \c Adj
 (1325)
 (1326) \g leg/foot
 (1327) \ge leg/foot
 (1328) \type r
 (1329) \a afe
 (1330) \u afe
 (1331) \c PNR
 (1332)
 (1333) \g leg/foot.3sP
 (1334) \ge his/her/its leg/foot
 (1335) \type r
 (1336) \a afo
 (1337) \u afo
 (1338) \c PNR
 (1339)
 (1340) \g lie
 (1341) \ge lie
 (1342) \type r
 (1343) \a yaawa
 (1344) \u yaawa
 (1345) \mp MC2
 (1346) \c VR
 (1347)
 (1348) \g light
 (1349) \ge light
 (1350) \type r
 (1351) \a lala
 (1352) \u lala_1
 (1353) \c NW
 (1354)
 (1355) \g like.that
 (1356) \ge like.that
 (1357) \type r
 (1358) \a onou
 (1359) \u onou
 (1360) \c Adj
 (1361)
 (1362) \g like1
 (1363) \ge like1
 (1364) \type s
 (1365) \a fe
 (1366) \u fe
 (1367) \o 100
 (1368) \c NW/NW Deictic/Deictic Adj/Adj
 (1369)
 (1370) \g like2
 (1371) \ge like2
 (1372) \type s
 (1373) \a neu +/- this _
 (1374) \u neu
 (1375) \o 100

(1376) \c Deictic/Deictic
 (1377)
 (1378) \g look.for
 (1379) \ge look.for
 (1380) \type r
 (1381) \a kuru
 (1382) \u kuru
 (1383) \mp MC2
 (1384) \c VR
 (1385)
 (1386) \g man
 (1387) \ge man
 (1388) \type r
 (1389) \a fai
 (1390) \u fai
 (1391) \c NW
 (1392)
 (1393) \g many
 (1394) \ge many
 (1395) \type r
 (1396) \a baingaro
 (1397) \u baingaro
 (1398) \c Adj
 (1399)
 (1400) \g my
 (1401) \ge my
 (1402) \type r
 (1403) \a yame
 (1404) \u yame
 (1405) \c PosPro
 (1406)
 (1407) \g myself
 (1408) \ge myself
 (1409) \type r
 (1410) \a yage
 (1411) \u yage
 (1412) \c RflPro
 (1413)
 (1414) \g N.not
 (1415) \ge not
 (1416) \type r
 (1417) \a uuwa
 (1418) \u uuwa_1
 (1419) \c NomNeg
 (1420)
 (1421) \g Nom
 (1422) \ge Nominalizer
 (1423) \type s
 (1424) \a ou / [C] _
 (1425) \a wou / [V] _
 (1426) \u ou
 (1427) \o 110
 (1428) \c VR/Nom Nom/Nom
 (1429)
 (1430) \g nose
 (1431) \ge nose
 (1432) \type r

(1433) \a mutu
 (1434) \u mutu
 (1435) \c PNR
 (1436)
 (1437) \g O1d
 (1438) \ge to us (dual)
 (1439) \type s
 (1440) \a hiri
 (1441) \a hir / _ [V]
 (1442) \u hiri
 (1443) \mp MC1
 (1444) \o 30
 (1445) \c VR/VR
 (1446)
 (1447) \g O1p
 (1448) \ge to us (plural)
 (1449) \type s
 (1450) \a higi
 (1451) \a hig / _ [V]
 (1452) \u higi
 (1453) \mp MC1
 (1454) \o 30
 (1455) \c VR/VR
 (1456)
 (1457) \g O1s
 (1458) \ge to me
 (1459) \type s
 (1460) \a hi
 (1461) \a hiy / _ [V]
 (1462) \u hii
 (1463) \mp MC2
 (1464) \o 30
 (1465) \c VR/VR
 (1466)
 (1467) \g O2d
 (1468) \ge to you (dual)
 (1469) \type s
 (1470) \a tiri
 (1471) \a tir / _ [V]
 (1472) \u tiri
 (1473) \mp MC1
 (1474) \o 30
 (1475) \c VR/VR
 (1476)
 (1477) \g O2p
 (1478) \ge to you (plural)
 (1479) \type s
 (1480) \a tigi
 (1481) \a tig / _ [V]
 (1482) \u tigi
 (1483) \mp MC1
 (1484) \o 30
 (1485) \c VR/VR
 (1486)
 (1487) \g O2s
 (1488) \ge to you (singular)
 (1489) \type s

(1490) \a hini
 (1491) \a hin / _ [V]
 (1492) \u hini
 (1493) \mp MC1
 (1494) \o 30
 (1495) \c VR/VR
 (1496)
 (1497) \g O3d
 (1498) \ge to them (dual)
 (1499) \type s
 (1500) \a niri
 (1501) \a nir / _ [V]
 (1502) \u niri
 (1503) \mp MC1
 (1504) \o 30
 (1505) \c VR/VR
 (1506)
 (1507) \g O3p
 (1508) \ge to them (plural)
 (1509) \type s
 (1510) \a nig
 (1511) \a nig / _ [V]
 (1512) \u nig
 (1513) \mp MC1
 (1514) \o 30
 (1515) \c VR/VR
 (1516)
 (1517) \g O3s
 (1518) \ge to him/her/it
 (1519) \type s
 (1520) \a tu
 (1521) \u tu
 (1522) \mp MC2
 (1523) \o 30
 (1524) \c VR/VR
 (1525)
 (1526) \g ocean
 (1527) \ge ocean
 (1528) \type r
 (1529) \a hiiri
 (1530) \u hiiri
 (1531) \c NW
 (1532)
 (1533) \g one/a
 (1534) \ge one/a
 (1535) \type r
 (1536) \a ete
 (1537) \u ete
 (1538) \c Adj
 (1539)
 (1540) \g our.dual
 (1541) \ge our.dual
 (1542) \type r
 (1543) \a heire
 (1544) \u heire
 (1545) \c PosPro
 (1546)

(1547) \g our.plural
(1548) \ge our.plural
(1549) \type r
(1550) \a ebere
(1551) \u ebere
(1552) \c PosPro
(1553)
(1554) \g ourselves.dual
(1555) \ge ourselves.dual
(1556) \type r
(1557) \a hare
(1558) \u hare
(1559) \c RflPro
(1560)
(1561) \g ourselves.plural
(1562) \ge ourselves.plural
(1563) \type r
(1564) \a hage
(1565) \u hage
(1566) \c RflPro
(1567)
(1568) \g p1
(1569) \ge plural
(1570) \type s
(1571) \a gana
(1572) \u gana
(1573) \o 20
(1574) \c VR/VR
(1575)
(1576) \g P1d
(1577) \ge first dual possessive
(1578) \type s
(1579) \a re
(1580) \u re
(1581) \o 10
(1582) \c PNR/PN
(1583)
(1584) \g P1p
(1585) \ge first plural possessive
(1586) \type s
(1587) \a ge
(1588) \u ge
(1589) \o 10
(1590) \c PNR/PN
(1591)
(1592) \g P1s
(1593) \ge first singular possessive
(1594) \type s
(1595) \a fe
(1596) \u fe
(1597) \o 10
(1598) \c PNR/PN
(1599)
(1600) \g p2
(1601) \ge plural
(1602) \type s
(1603) \a ru

(1604) \u ru
 (1605) \o 20
 (1606) \c VR/VR
 (1607)
 (1608) \g P2d
 (1609) \ge second dual possessive
 (1610) \type s
 (1611) \a tere
 (1612) \u tere
 (1613) \o 10
 (1614) \c PNR/PN
 (1615)
 (1616) \g P2p
 (1617) \ge second plural possessive
 (1618) \type s
 (1619) \a tege
 (1620) \u tege
 (1621) \o 10
 (1622) \c PNR/PN
 (1623)
 (1624) \g P2s
 (1625) \ge second singular possessive
 (1626) \type s
 (1627) \a ne
 (1628) \a te
 (1629) \u ne
 (1630) \o 10
 (1631) \c PNR/PN
 (1632)
 (1633) \g P3d
 (1634) \ge third dual possessive
 (1635) \type s
 (1636) \a nere
 (1637) \u nere
 (1638) \o 10
 (1639) \c PNR/PN
 (1640)
 (1641) \g P3p
 (1642) \ge third plural possessive
 (1643) \type s
 (1644) \a nege
 (1645) \u nege
 (1646) \o 10
 (1647) \c PNR/PN
 (1648)
 (1649) \g P3s
 (1650) \ge third singular possessive
 (1651) \type s
 (1652) \a 0
 (1653) \a u
 (1654) \u 0
 (1655) \o 10
 (1656) \c PNR/PN
 (1657)
 (1658) \g perf
 (1659) \ge perfective
 (1660) \type s

(1661) \a ha
 (1662) \u ha
 (1663) \o 100
 (1664) \c SFV/SFVW
 (1665)
 (1666) \g pertaining.to
 (1667) \ge pertaining.to
 (1668) \type s
 (1669) \a ha
 (1670) \u ha
 (1671) \c Adj/Adj NW/NW Deictic/Deictic PosPro/PosPro
 (1672)
 (1673) \g pig
 (1674) \ge pig
 (1675) \type r
 (1676) \a buuwa
 (1677) \u buuwa
 (1678) \c NW
 (1679)
 (1680) \g place
 (1681) \ge place
 (1682) \type r
 (1683) \a ha
 (1684) \a haumu
 (1685) \u ha
 (1686) \c NW
 (1687)
 (1688) \g plant
 (1689) \ge plant
 (1690) \type r
 (1691) \a tau
 (1692) \a ta
 (1693) \u tau
 (1694) \mp MC2 RC1
 (1695) \c VR
 (1696)
 (1697) \g poke
 (1698) \ge poke
 (1699) \type r
 (1700) \a tu / _ [O]
 (1701) \a tum / _ [B]
 (1702) \a tun / _ [A]
 (1703) \a tung / _ [X]
 (1704) \u tuN
 (1705) \mp MC2
 (1706) \c VR
 (1707)
 (1708) \g pour.out
 (1709) \ge pour.out
 (1710) \type r
 (1711) \a feu
 (1712) \u feu
 (1713) \mp MC2
 (1714) \c VR
 (1715)
 (1716) \g Prediction
 (1717) \ge Prediction

(1718) \type s
 (1719) \a ga
 (1720) \u ga
 (1721) \o 100
 (1722) \c VSjs/VSjs
 (1723)
 (1724) \g prepare
 (1725) \ge prepare
 (1726) \type r
 (1727) \a haiya
 (1728) \u haiya
 (1729) \mp MC2
 (1730) \c VR
 (1731)
 (1732) \g press
 (1733) \ge press
 (1734) \type r
 (1735) \a ilei
 (1736) \u ilei
 (1737) \mp MC2
 (1738) \c VR
 (1739)
 (1740) \g put
 (1741) \ge put
 (1742) \type r
 (1743) \a tafa
 (1744) \a taf / _ [V]
 (1745) \u tafa
 (1746) \mp MC1
 (1747) \c VR
 (1748)
 (1749) \g quickly
 (1750) \ge quickly
 (1751) \type r
 (1752) \a fasa
 (1753) \u fasa
 (1754) \mp MC2
 (1755) \c VR
 (1756)
 (1757) \g quickly
 (1758) \ge quickly
 (1759) \type s
 (1760) \a ya
 (1761) \u ya
 (1762) \c VR/VR
 (1763)
 (1764) \g red
 (1765) \ge red
 (1766) \type r
 (1767) \a giiri
 (1768) \u giiri
 (1769) \c Adj
 (1770)
 (1771) \g remove
 (1772) \ge remove
 (1773) \type r
 (1774) \a uulu

(1775) \u uulu
 (1776) \mp MC2
 (1777) \c VR
 (1778)
 (1779) \g run.away
 (1780) \ge run.away
 (1781) \type r
 (1782) \a faga
 (1783) \u faga
 (1784) \mp MC2
 (1785) \c VR
 (1786)
 (1787) \g see.01d
 (1788) \ge see us (dual)
 (1789) \type r
 (1790) \a herire
 (1791) \a herir / _ [V]
 (1792) \u herire
 (1793) \mp MC1
 (1794) \c VR
 (1795)
 (1796) \g see.01p
 (1797) \ge see us (plural)
 (1798) \type r
 (1799) \a herige
 (1800) \a herig / _ [V]
 (1801) \u herige
 (1802) \mp MC1
 (1803) \c VR
 (1804)
 (1805) \g see.01s
 (1806) \ge see me
 (1807) \type r
 (1808) \a yeriye
 (1809) \a yeriy / _ [V]
 (1810) \u yeriye
 (1811) \mp MC1
 (1812) \c VR
 (1813)
 (1814) \g see.02d
 (1815) \ge see you (dual)
 (1816) \type r
 (1817) \a terire
 (1818) \a terir / _ [V]
 (1819) \u terire
 (1820) \mp MC1
 (1821) \c VR
 (1822)
 (1823) \g see.02p
 (1824) \ge see you (plural)
 (1825) \type r
 (1826) \a terige
 (1827) \a terig / _ [V]
 (1828) \u terige
 (1829) \mp MC1
 (1830) \c VR
 (1831)

(1832) \g see.02s
 (1833) \ge see you (singular)
 (1834) \type r
 (1835) \a neriy
 (1836) \a neriy / _ [V]
 (1837) \u neriy
 (1838) \mp MC1
 (1839) \c VR
 (1840)
 (1841) \g see.03d
 (1842) \ge see them (dual)
 (1843) \type r
 (1844) \a nerire
 (1845) \a nerir / _ [V]
 (1846) \u nerire
 (1847) \mp MC1
 (1848) \c VR
 (1849)
 (1850) \g see.03p
 (1851) \ge see them (plural)
 (1852) \type r
 (1853) \a nerige
 (1854) \a nerig / _ [V]
 (1855) \u nerige
 (1856) \mp MC1
 (1857) \c VR
 (1858)
 (1859) \g see.03s
 (1860) \ge see him/her/it
 (1861) \type r
 (1862) \a ure
 (1863) \a ur / _ [V]
 (1864) \u ure
 (1865) \mp MC1
 (1866) \c VR
 (1867)
 (1868) \g send
 (1869) \ge send
 (1870) \type r
 (1871) \a huuru
 (1872) \u huuru
 (1873) \mp MC2
 (1874) \c VR
 (1875)
 (1876) \g shine
 (1877) \ge shine
 (1878) \type r
 (1879) \a lala
 (1880) \u lala_2
 (1881) \mp MC2
 (1882) \c VR
 (1883)
 (1884) \g short
 (1885) \ge short
 (1886) \type r
 (1887) \a tutu
 (1888) \u tutu

(1889) \c Adj
 (1890)
 (1891) \g shove
 (1892) \ge shove
 (1893) \type r
 (1894) \a nuu
 (1895) \u nuu
 (1896) \mp MC2
 (1897) \c VR
 (1898)
 (1899) \g show
 (1900) \ge show
 (1901) \type r
 (1902) \a abiti
 (1903) \u abiti
 (1904) \mp MC2
 (1905) \c VR
 (1906)
 (1907) \g sibling.same.sex.his/her/its
 (1908) \ge sibling.same.sex.his/her/its
 (1909) \type r
 (1910) \a bemu
 (1911) \u bai_2
 (1912) \c NW
 (1913)
 (1914) \g sibling.same.sex.possessive
 (1915) \ge sibling.same.sex.possessive
 (1916) \type r
 (1917) \a bai +/ ~_ P3s
 (1918) \u bai_2
 (1919) \c PNR
 (1920)
 (1921) \g sibling.same.sex.referential
 (1922) \ge sibling.same.sex.referential
 (1923) \type r
 (1924) \a babo
 (1925) \u bai_2
 (1926) \c NW
 (1927)
 (1928) \g sit
 (1929) \ge sit
 (1930) \type r
 (1931) \a yafa
 (1932) \a yaf / _ [V]
 (1933) \u yafa
 (1934) \mp MC1
 (1935) \c VR
 (1936)
 (1937) \g Sj.S1/3
 (1938) \ge subjunctive first or third subject
 (1939) \type s
 (1940) \a o / [C] _
 (1941) \a wo / [V] _
 (1942) \u o
 (1943) \o 80
 (1944) \c VR/VSj
 (1945)

(1946) \g Sj.S2
 (1947) \ge subjunctive second subject
 (1948) \type s
 (1949) \a a / [C] _
 (1950) \a wa / [V] _
 (1951) \u a
 (1952) \o 80
 (1953) \c VR/VSj
 (1954)
 (1955) \g Sj.Sd
 (1956) \ge subjunctive dual subject
 (1957) \type s
 (1958) \a ru
 (1959) \u ru
 (1960) \o 90
 (1961) \c VSj/VSjS
 (1962)
 (1963) \g Sj.Sp
 (1964) \ge subjunctive plural subject
 (1965) \type s
 (1966) \a gu
 (1967) \u gu
 (1968) \o 90
 (1969) \c VSj/VSjS
 (1970)
 (1971) \g Sj.Ss
 (1972) \ge subjunctive singular subject
 (1973) \type s
 (1974) \a u
 (1975) \u u
 (1976) \o 90
 (1977) \c VSj/VSjS
 (1978)
 (1979) \g sleep.pl
 (1980) \ge sleep.pl
 (1981) \type r
 (1982) \a jiri
 (1983) \u jiri
 (1984) \mp MC2
 (1985) \c VR
 (1986)
 (1987) \g small
 (1988) \ge small
 (1989) \type r
 (1990) \a menekele
 (1991) \a melekene
 (1992) \u menekele
 (1993) \c Adj
 (1994)
 (1995) \g smooth
 (1996) \ge smooth
 (1997) \type r
 (1998) \a fii
 (1999) \u fii
 (2000) \mp MC2
 (2001) \c VR
 (2002)

(2003) \g some
 (2004) \ge some
 (2005) \type r
 (2006) \a ila
 (2007) \u ila
 (2008) \c Adj
 (2009)
 (2010) \g speak
 (2011) \ge speak
 (2012) \type r
 (2013) \a le
 (2014) \u le
 (2015) \mp MC2
 (2016) \c VR
 (2017)
 (2018) \g SR
 (2019) \ge switch referent
 (2020) \type s
 (2021) \a nga
 (2022) \u nga
 (2023) \o 110
 (2024) \c SMV/SMVW Contra/SMVW
 (2025)
 (2026) \g SR.TO
 (2027) \ge switch referent with temporal overlap
 (2028) \type s
 (2029) \a g +/- _ [Ind.T] +/- _ [Contra.Subj]
 (2030) \u g
 (2031) \o 70
 (2032) \c VR/SMV
 (2033)
 (2034) \g SR.TS
 (2035) \ge switch referent with temporal succession
 (2036) \type s
 (2037) \a 0 +/- _ [Ind.T] +/- _ [Contra.Subj]
 (2038) \u 0
 (2039) \o 70
 (2040) \c VR/SMV
 (2041)
 (2042) \g stand
 (2043) \ge stand
 (2044) \type r
 (2045) \a oto
 (2046) \u oto
 (2047) \mp MC2
 (2048) \c VR
 (2049)
 (2050) \g stick.to
 (2051) \ge stick.to
 (2052) \type r
 (2053) \a fada
 (2054) \u fada
 (2055) \mp MC2
 (2056) \c VR
 (2057)
 (2058) \g stone
 (2059) \ge stone

(2060) \type r
(2061) \a megebu
(2062) \u megebu
(2063) \c NW
(2064)
(2065) \g straight
(2066) \ge straight
(2067) \type r
(2068) \a tigini
(2069) \u tigini
(2070) \c Adj
(2071)
(2072) \g talk/word
(2073) \ge talk/word
(2074) \type r
(2075) \a beele
(2076) \u beele
(2077) \c NW
(2078)
(2079) \g teach
(2080) \ge teach
(2081) \type r
(2082) \a agotete
(2083) \u agotete
(2084) \mp MC2
(2085) \c VR
(2086)
(2087) \g tear
(2088) \ge tear
(2089) \type r
(2090) \a korai
(2091) \u korai
(2092) \mp MC2
(2093) \c VR
(2094)
(2095) \g tell
(2096) \ge tell
(2097) \type r
(2098) \a hai
(2099) \u hai
(2100) \mp MC2
(2101) \c VR
(2102)
(2103) \g that
(2104) \ge that
(2105) \type r
(2106) \a wa
(2107) \u wa
(2108) \c Deictic
(2109)
(2110) \g their.dual
(2111) \ge their.dual
(2112) \type r
(2113) \a neire
(2114) \u neire
(2115) \c PosPro
(2116)

(2117) \g their.plural
 (2118) \ge their.plural
 (2119) \type r
 (2120) \a nebere
 (2121) \u nebere
 (2122) \c PosPro
 (2123)
 (2124) \g theirselves.dual
 (2125) \ge theirselves.dual
 (2126) \type r
 (2127) \a nare
 (2128) \u nare
 (2129) \c RflPro
 (2130)
 (2131) \g theirselves.plural
 (2132) \ge theirselves.plural
 (2133) \type r
 (2134) \a nage
 (2135) \u nage
 (2136) \c RflPro
 (2137)
 (2138) \g they
 (2139) \ge they
 (2140) \type r
 (2141) \a nere
 (2142) \u nere
 (2143) \c PerPro
 (2144)
 (2145) \g thigh
 (2146) \ge thigh
 (2147) \type r
 (2148) \a gaigu
 (2149) \u gaigu
 (2150) \c NW
 (2151)
 (2152) \g thing
 (2153) \ge thing
 (2154) \type r
 (2155) \a dada
 (2156) \u dada
 (2157) \c NW
 (2158)
 (2159) \g this
 (2160) \ge this
 (2161) \type r
 (2162) \a be
 (2163) \u be
 (2164) \c Deictic
 (2165)
 (2166) \g thought
 (2167) \ge thought
 (2168) \type r
 (2169) \a nina
 (2170) \u nina
 (2171) \c PNR
 (2172)
 (2173) \g throw

(2174) \ge throw
 (2175) \type r
 (2176) \a aya
 (2177) \u aya
 (2178) \mp MC2
 (2179) \c VR
 (2180)
 (2181) \g TO
 (2182) \ge temporal overlap
 (2183) \type s
 (2184) \a 0
 (2185) \u 0
 (2186) \o 70
 (2187) \c VR/SMVW
 (2188)
 (2189) \g true
 (2190) \ge true
 (2191) \type r
 (2192) \a ngalenga
 (2193) \u ngalenga
 (2194) \c Adj
 (2195)
 (2196) \g TS
 (2197) \ge temporal succession
 (2198) \type s
 (2199) \a du
 (2200) \u du
 (2201) \o 70
 (2202) \c VR/SMVW
 (2203)
 (2204) \g turn.back.on
 (2205) \ge turn.back.on
 (2206) \type r
 (2207) \a moge
 (2208) \u moge
 (2209) \mp MC2
 (2210) \c VR
 (2211)
 (2212) \g untie
 (2213) \ge untie
 (2214) \type r
 (2215) \a fuutu
 (2216) \u fuutu
 (2217) \mp MC2
 (2218) \c VR
 (2219)
 (2220) \g V.not
 (2221) \ge not
 (2222) \type r
 (2223) \a ada
 (2224) \u ada
 (2225) \c VNeg
 (2226)
 (2227) \g vAux.S1d
 (2228) \ge verbal auxiliary first dual subject
 (2229) \type s
 (2230) \a yare

(2231) \u yare
 (2232) \o 90
 (2233) \c vAuxR/vAuxW
 (2234)
 (2235) \g vAux.S1p
 (2236) \ge verbal auxiliary first plural subject
 (2237) \type s
 (2238) \a ya
 (2239) \u ya
 (2240) \o 90
 (2241) \c vAuxR/vAuxW
 (2242)
 (2243) \g vAux.S1s
 (2244) \ge verbal auxiliary first singular subject
 (2245) \type s
 (2246) \a ni
 (2247) \u ni
 (2248) \o 90
 (2249) \c vAuxR/vAuxW
 (2250)
 (2251) \g vAux.S3d
 (2252) \ge verbal auxiliary third dual subject
 (2253) \type s
 (2254) \a dere
 (2255) \u dere
 (2256) \o 90
 (2257) \c vAuxR/vAuxW
 (2258)
 (2259) \g vAux.S3p
 (2260) \ge verbal auxiliary third plural subject
 (2261) \type s
 (2262) \a de
 (2263) \u de
 (2264) \o 90
 (2265) \c vAuxR/vAuxW
 (2266)
 (2267) \g vAux.S3s
 (2268) \ge verbal auxiliary third singular subject
 (2269) \type s
 (2270) \a na
 (2271) \u na
 (2272) \o 90
 (2273) \c vAuxR/vAuxW
 (2274)
 (2275) \g vAux.Tf.S2d
 (2276) \ge verbal auxiliary second dual subject
 (2277) \type s
 (2278) \a nadere
 (2279) \u nadere
 (2280) \o 90
 (2281) \c vAuxR/vAuxW
 (2282)
 (2283) \g vAux.Tf.S2p
 (2284) \ge verbal auxiliary second plural subject
 (2285) \type s
 (2286) \a nade
 (2287) \u nade

(2288) \o 90
 (2289) \c vAuxR/vAuxW
 (2290)
 (2291) \g vAux.Tf.S2s
 (2292) \ge verbal auxiliary second singular subject, future tense
 (2293) \type s
 (2294) \a nahe
 (2295) \u nahe
 (2296) \o 90
 (2297) \c vAuxR/vAuxW
 (2298)
 (2299) \g vAux.Tnf.S2d
 (2300) \ge verbal auxiliary second dual subject, non-future tense
 (2301) \type s
 (2302) \a yare
 (2303) \u yare
 (2304) \o 90
 (2305) \c vAuxR/vAuxW
 (2306)
 (2307) \g vAux.Tnf.S2p
 (2308) \ge verbal auxiliary second plural subject, non-future tense
 (2309) \type s
 (2310) \a ya
 (2311) \u ya
 (2312) \o 90
 (2313) \c vAuxR/vAuxW
 (2314)
 (2315) \g vAux.Tnf.S2s
 (2316) \ge verbal auxiliary second singular subject, non-future
 (2317) \type s
 (2318) \a ne
 (2319) \u ne
 (2320) \o 90
 (2321) \c vAuxR/vAuxW
 (2322)
 (2323) \g vAuxR
 (2324) \ge verbal auxiliary root
 (2325) \type r
 (2326) \a bo
 (2327) \u bo
 (2328) \c vAuxR
 (2329)
 (2330) \g very
 (2331) \ge very
 (2332) \type r
 (2333) \a bai
 (2334) \u bai
 (2335) \c Adj
 (2336)
 (2337) \g village
 (2338) \ge village
 (2339) \type r
 (2340) \a matane
 (2341) \u matane
 (2342) \c NW
 (2343)
 (2344) \g walk

(2345) \ge walk
 (2346) \type r
 (2347) \a golo
 (2348) \u golo
 (2349) \mp MC2
 (2350) \c VR
 (2351)
 (2352) \g watch
 (2353) \ge watch
 (2354) \type r
 (2355) \a fele
 (2356) \u fele
 (2357) \mp MC2
 (2358) \c VR
 (2359)
 (2360) \g watch.over
 (2361) \ge watch.over
 (2362) \type r
 (2363) \a oofa
 (2364) \u oofa
 (2365) \mp MC2
 (2366) \c VR
 (2367)
 (2368) \g we
 (2369) \ge we
 (2370) \type r
 (2371) \a ere
 (2372) \u ere
 (2373) \c PerPro
 (2374)
 (2375) \g well
 (2376) \ge well
 (2377) \type s
 (2378) \a de
 (2379) \u de
 (2380) \c VR/VR
 (2381)
 (2382) \g what
 (2383) \ge what
 (2384) \type r
 (2385) \a taate
 (2386) \u taate
 (2387) \c IP
 (2388)
 (2389) \g wild
 (2390) \ge wild
 (2391) \type r
 (2392) \a kui
 (2393) \u kui
 (2394) \c Adj
 (2395)
 (2396) \g with
 (2397) \ge with
 (2398) \type r
 (2399) \a bagu +/- ~_ causative
 (2400) \u bagu
 (2401) \c Prep

(2402)
 (2403) \g with.dual
 (2404) \ge with.dual
 (2405) \type r
 (2406) \a ngare
 (2407) \u ngare
 (2408) \c Prep
 (2409)
 (2410) \g without
 (2411) \ge without
 (2412) \type r
 (2413) \a uwau
 (2414) \u uwau
 (2415) \c Prep
 (2416)
 (2417) \g woman
 (2418) \ge woman
 (2419) \type r
 (2420) \a aita
 (2421) \u aita
 (2422) \c NW
 (2423)
 (2424) \g work
 (2425) \ge work
 (2426) \type r
 (2427) \a haruwe
 (2428) \u haruwe
 (2429) \c NW
 (2430)
 (2431) \g write
 (2432) \ge write
 (2433) \type r
 (2434) \a yerege
 (2435) \a yereg / _ [V]
 (2436) \u yerege
 (2437) \mp MC1
 (2438) \c VR
 (2439)
 (2440) \g yell.to/shout
 (2441) \ge yell.to/shout
 (2442) \type r
 (2443) \a yaure
 (2444) \u yaure
 (2445) \mp MC2
 (2446) \c VR
 (2447)
 (2448) \g you.plural
 (2449) \ge you.plural
 (2450) \type r
 (2451) \a tere
 (2452) \u tere
 (2453) \c PerPro
 (2454)
 (2455) \g you.sing
 (2456) \ge you.sing
 (2457) \type r
 (2458) \a ne

(2459) \u ne
(2460) \c PerPro
(2461)
(2462) \g your.dual
(2463) \ge your.dual
(2464) \type r
(2465) \a teire
(2466) \u teire
(2467) \c PosPro
(2468)
(2469) \g your.plural
(2470) \ge your.plural
(2471) \type r
(2472) \a tebere
(2473) \u tebere
(2474) \c PosPro
(2475)
(2476) \g your.sing
(2477) \ge your.sing
(2478) \type r
(2479) \a name
(2480) \u name
(2481) \c PosPro
(2482)
(2483) \g yourselves.dual
(2484) \ge yourselves.dual
(2485) \type r
(2486) \a tare
(2487) \u tare
(2488) \c RflPro
(2489)
(2490) \g yourselves.plural
(2491) \ge yourselves.plural
(2492) \type r
(2493) \a tage
(2494) \u tage
(2495) \c RflPro

APPENDIX 3

THE OGEA DATR LEXICON

A3.1 Overview

This appendix provides listings of the following: The Ogea DATR "theory" file (the lexicon), the Ogea DATR query file, a DOS batch file that invokes DATR, and the result of running DATR using the query file--a query trace file.

The Ogea DATR theory file is the heart of the Ogea DATR lexicon. It is divided into the following sections: function library, interfaces, AMPLE environment constraints, templates, categories, English, Tok Pisin, and Ogea. The Ogea DATR query file is the driver for the interface between DATR and AMPLE. The entries in the query file mirror the entries in the Ogea AMPLE database file. The DOS batch file listing contains two calls to DATR. The first call creates a token file, the second executes the query file against the token file. The Ogea DATR trace file is the output of running the DOS batch file. The trace file provides the Ogea AMPLE lexical entries, combining both roots and suffixes.

A3.2 The Ogea DATR Theory File

A3.2.1 Description

The Ogea DATR theory file is divided into the following sections: function library, interfaces, AMPLE environment constraints, templates, categories, English, Tok Pisin, and Ogea.

The function library loads the standard DATR function library and adds a new function called 'chop'. The 'chop' function is used in the application of various Ogea morphophonemic rules.

The interfaces section provides a one-way interface from DATR to AMPLE and a dictionary. The AMPLE interface supports generation of AMPLE lexical files that use either English, Ogea, or Tok Pisin as the value for the record marker. The dictionary interface generates dictionary entries using plain text labels for each field rather than codes.

The AMPLE environment constraints are a series of nodes that act as variables. Each node defines the detail of a specific string or morpheme constraint. The string constraints begin with a node labeled 'SEC', and the morpheme constraints with a node labeled 'MEC'. These two abstract nodes define the AMPLE makers used to indicate the start of a constraint.

The templates section provides a series of abstract nodes used for two purposes. First, they provide support for multiple allomorph entries in the case that no morphophonemic rules are applied. Second, they provide a mechanism by which morphophonemic rules can be applied to underlying forms to produce the correct allomorphic form. Through the application of morphophonemic rules, only a single underlying form needs to be specified. This reduces lexicon maintenance, and also allows testing of morphophonemic rules.

The categories section provides abstract nodes that represent grammatical and morphological categories. It is through these abstract nodes that much of the redundancy of the Ogea AMPLE lexicon can be reduced.

The English section provides a set of nodes ordered by English glosses. This section supports DATR queries that utilize English words to retrieve the appropriate Ogea. As such it is the entry point into the lexicon through English.

The Tok Pisin section is the entry point into the lexicon through Tok Pisin. Whereas the English section contains a full set of English nodes, the Tok Pisin section has only three

nodes and was developed merely to illustrate how queries may start with Tok Pisin and retrieve Ogea.

The last section is the most comprehensive and is the Ogea section. This section contains the full set of nodes for Ogea roots and suffixes. All entries in the Ogea section are uniquely identified by Ogea morphemes and allomorphs. The morphemes and allomorphs are identified as roots or suffixes by a prefix (R_ or S_) ²⁰. In the case of homophones, a suffix is added to uniquely identify that morpheme or allomorph. The suffix sometimes uses a numeric identifier, and sometimes uses a grammatical gloss. Note that in the case of homophones, an abstract entry is provided to support a query that is ignorant of the uniquely identifying suffixes. For example, the root node R_anyakaro may be queried without knowledge of the two entries for anyakaro. A query will be redirected to R_anyakaro_1 and R_anayakaro_2.

A3.2.2 File Listing

Filename: ogea.dtr

Note: line numbers added for reference only. Not in actual file.

²⁰ DATR node names may not start with a numeral, so zero morphs may not be entered with as a node name starting with '0'. For this reason and pure management of the file, a decision was made to use a leading prefix for Ogea node names.

```

(2496) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(2497) %                                                                                               %
(2498) % File:           Ogea.dtr                                                                 %
(2499) % Purpose:       Lexicon for Ogea Language                                             %
(2500) % Ogea:         A Papuan Language of Papua New Guinea                               %
(2501) % Author:      Michael Colburn                                                         %
(2502) % Email:       mcolburn@sprintmail.com                                               %
(2503) % Created:    October 9, 1999                                                         %
(2504) % Last update: November 13, 1999                                                    %
(2505) % Version:    1.0                                                                     %
(2506) %                                                                                               %
(2507) % Notes:      Supports queries from/to English, Ogea,                               %
(2508) %              and Tok Pisin                                                         %
(2509) %                                                                                               %
(2510) %                                                                                               %
(2511) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(2512)
(2513)
(2514) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Function Library %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(2515) %
(2516)
(2517) #load 'std.dtr' all.
(2518)
(2519) % Chop: Chops off the last byte of a string
(2520) Chop:
(2521) <> ==
(2522) <$a> == Implode:<Reverse:<Rest:<Reverse:<Explode:<$a>>>>.
(2523)
(2524) %
(2525) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
      %%%%%%%%%
(2526)
(2527) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Interfaces %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(2528) %
(2529)
(2530) IAMPLE:
(2531) % Generates records with English as first field
(2532) <amp lex eng> ==      <amp gloss eng>
(2533)                   <amp gloss morpheme name>
(2534) %               <amp gloss national>
(2535)                   <amp rest>
(2536) % Generates records with the national language as first field
(2537) <amp lex nat> ==   <amp gloss national>
(2538)                   <amp gloss morpheme name>
(2539)                   <amp gloss eng>
(2540)                   <amp rest>
(2541) % Generates records with the vernacular language
(2542) % as first field
(2543) <amp lex ver> ==    <amp gloss vernacular>
(2544)                   <amp gloss morpheme name>
(2545)                   <amp gloss eng>
(2546)                   <amp gloss national>
(2547)                   <amp rest>
(2548) % Generates records with the morphname as the first field
(2549) <amp lex morname> == <amp gloss morpheme name>
(2550)                   <amp gloss eng>
(2551) %               <amp gloss national>

```

```

(2552)                                     <amp; rest>
(2553) <amp; rest> ==                         <amp; type>
(2554)                                     "<lex form lex>"
(2555)                                     <amp; underlying form>
(2556)                                     <amp; allo property>
(2557)                                     <amp; order class>
(2558)                                     <amp; category>
(2559)                                     <amp; elsewhere allomorph>
(2560)                                     <amp; mcc>
(2561)                                     <amp; features>
(2562)                                     <amp; infix location>
(2563)                                     <amp; noload>
(2564)                                     <amp; comment>'\n'
(2565) <amp; allomorph> == \a' ' "<allo form>" ' ' "<allo prop>" ' '
(2566)                                     "<allo sec>" ' ' "<allo
    mec>"'\n'
(2567) <amp; category> == \c' ' "<lex cat>"'\n'
(2568) <amp; comment> == \co' ' "<lex com>"'\n'
(2569) <amp; elsewhere allomorph> == \e' ' "<lex else allo>"'\n'
(2570) <amp; features> == \f' ' "<lex feat>"'\n'
(2571) <amp; gloss eng> == \ge' ' "<lex gloss eng>" '\n' % not called
(2572)                                     since used
    as mor name
(2573) <amp; gloss vernacular> == \gv' ' "<lex under>" '\n'
(2574) <amp; gloss national> == \gn' ' "<lex gloss nat>" '\n'
(2575) <amp; infix location> == \loc' ' "<lex loc>" '\n'
(2576) <amp; gloss morpheme name> == \g' ' "<lex gloss eng morname>"
    '\n'
(2577) <amp; allo property> == \mp' ' "<lex prop>"'\n'
(2578) <amp; noload> == \_no' ' "<lex noload>"'\n'
(2579) <amp; order class> == \o' ' "<lex order class>"'\n'
(2580) <amp; mcc> == \mcc' ' "<lex mcc>"'\n'
(2581) <amp; type> == \type' ' "<lex type>"'\n'
(2582) <amp; underlying form> == \u' ' "<lex under>"'\n'.
(2583)
(2584) I_DICT:
(2585) <dict lex eng> == "<lex gloss eng>"'\n'
(2586)     ' Ogea:           ' "<lex under>"'\n'
(2587)     ' Category:      ' "<lex cat>"'\n'
(2588)     ' Property:       ' "<lex prop>"'\n'
(2589)     ' Allomorph(s): \n'
(2590)     "<lex form dict>"'\n'
(2591) <dict allomorph> ==
(2592)     '
(2593)     ' Property:       ' "<allo prop>"'\n'
(2594)     ' String Environment: ' "<allo sec>"'\n'
(2595)     ' Morpheme Environment: ' "<allo
    mec>"'\n'.
(2596)
(2597) %
(2598) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(2599)
(2600) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Templates %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(2601) %
(2602)
(2603) ALLO1:
(2604) <lex form lex> == <lex form lex allo1>

```

```

(2605) <lex form lex allo1> == C_DLT
(2606) <lex form dict> == <lex form dict allo1>
(2607) <lex form dict allo1> == C_DLT.
(2608)
(2609) ALLO2:
(2610) <lex form lex> == <lex form lex allo2>
(2611) <lex form lex allo2> == C_DLT
(2612) <lex form dict> == <lex form dict allo2>
(2613) <lex form dict allo2> == C_DLT.
(2614)
(2615) ALLO3:
(2616) <lex form lex> == <lex form lex allo3>
(2617) <lex form lex allo3> == C_DLT
(2618) <lex form dict> == <lex form dict allo3>
(2619) <lex form dict allo3> == C_DLT.
(2620)
(2621) ALLO4:
(2622) <lex form lex> == <lex form lex allo4>
(2623) <lex form lex allo4> == C_DLT
(2624) <lex form dict> == <lex form dict allo4>
(2625) <lex form dict allo4> == C_DLT.
(2626)
(2627) ALLO5:
(2628) <lex form lex> == <lex form lex allo5>
(2629) <lex form lex allo5> == C_DLT
(2630) <lex form dict> == <lex form dict allo5>
(2631) <lex form dict allo5> == C_DLT.
(2632)
(2633) ALLOMORPH:
(2634) <allo form default> == "<phon under>"
(2635) <allo form elision> == Chop:<"<phon under>">
(2636) <allo form epenthetic_w> == Implode:<w "<phon under>">
(2637) <allo form glide> == Implode:<Chop:<"<phon under>"> y>
(2638) <allo form redup> == Implode:<First:<Explode:
(2639) <"<phon under>">>
First:<Rest:<Explode:
(2640) <"<phon under>">>>>
(2641) <allo form syncope> == Chop:<"<phon under>">
(2642) <allo form n_b> == Implode:<Chop:<"<phon under>"> m>
(2643) <allo form n_a> == Implode:<Chop:<"<phon under>"> n>
(2644) <allo form n_x> == Implode:<Chop:<"<phon under>"> ng>
(2645) <allo form n_v> == Chop:<"<phon under>">
(2646) <allo prop> ==
(2647) <allo mec> ==
(2648) <allo sec> ==
(2649) <allo sec epenthetic_w> == "SEC_02"
(2650) <allo sec glide> == "SEC_01"
(2651) <allo sec syncope> == "SEC_01"
(2652) <allo sec n_b> == "SEC_04"
(2653) <allo sec n_a> == "SEC_05"
(2654) <allo sec n_x> == "SEC_07"
(2655) <allo sec n_v> == "SEC_06".
(2656)
(2657) DEFAULT:
(2658) <lex form lex> == <lex form lex default>
(2659) <lex form dict> == <lex form dict default>
(2660) <lex form lex default> == C_DLT

```



```

(2716) %%%%%%%%%%%%%%% AMPLE Environment Constraints %%%%%%%%%
(2717) %
(2718) % String Environment Constraints
(2719)
(2720) SEC:
(2721) <sec environ> == Implode:<'/' ' "<environ>"' '>
(2722) <allo sec> == <sec environ.>. % if don't cut path,
(2723)                                     must specify each
      extension
(2724)
(2725) SEC_01:
(2726) <> == SEC
(2727) <environ> == '_ [V]'.
(2728)
(2729) SEC_02:
(2730) <> == SEC
(2731) <environ> == '[V] _'.
(2732)
(2733) SEC_03:
(2734) <> == SEC
(2735) <environ> == '[C] _'.
(2736)
(2737) SEC_04:
(2738) <> == SEC
(2739) <environ> == '_ [B]'.
(2740)
(2741) SEC_05:
(2742) <> == SEC
(2743) <environ> == '_ [A]'.
(2744)
(2745) SEC_06:
(2746) <> == SEC
(2747) <environ> == '_ [O]'.
(2748)
(2749) SEC_07:
(2750) <> == SEC
(2751) <environ> == '_ [X]'.
(2752)
(2753) SEC_08:
(2754) <> == SEC
(2755) <environ> == 'o _ #'.
(2756)
(2757) SEC_09:
(2758) <> == SEC
(2759) <environ> == 'e _'.
(2760)
(2761) SEC_10:
(2762) <> == SEC
(2763) <environ> == 'o _'.
(2764)
(2765) SEC_11:
(2766) <> == SEC
(2767) <environ> == 'a _'.
(2768)
(2769) % Morpheme Environment Constraints
(2770)
(2771) MEC:

```

```

(2772)      <mec environ> == Implode:<'+' ' "<environ>"' '>
(2773)      <allo mec> == <mec environ.>.    % if don't cut path,
(2774)                                     must specify each
      extension
(2775)
(2776)      MEC_01:
(2777)      <> == MEC
(2778)      <environ> == '~_ TO'.
(2779)
(2780)      MEC_02:
(2781)      <> == MEC
(2782)      <environ> == '~_ P3s'.
(2783)
(2784)      MEC_03:
(2785)      <> == MEC
(2786)      <environ> == '~_ causative'.
(2787)
(2788)      MEC_04:
(2789)      <> == MEC
(2790)      <environ> == '_ [Contra.Subj]'.
(2791)
(2792)      MEC_05:
(2793)      <> == MEC
(2794)      <environ> == '_ Contra'.
(2795)
(2796)      MEC_06:
(2797)      <> == MEC
(2798)      <environ> == 'Ind.Tf _ ~SR'.
(2799)
(2800)      MEC_07:
(2801)      <> == MEC
(2802)      <environ> == 'Ind.Tf _ SR'.
(2803)
(2804)      MEC_08:
(2805)      <> == MEC
(2806)      <environ> == '[Ind.Tip.Tp] _ SR'.
(2807)
(2808)      MEC_09:
(2809)      <> == MEC
(2810)      <environ> == 'Ind.Trp _'.
(2811)
(2812)      MEC_10:
(2813)      <> == MEC
(2814)      <environ> == '[Ind.Tip.Tp] _'.
(2815)
(2816)      MEC_11:
(2817)      <> == MEC
(2818)      <environ> == 'Ind.Tip _'.
(2819)
(2820)      MEC_12:
(2821)      <> == MEC
(2822)      <environ> == 'Ind.Tp _ ~SR'.
(2823)
(2824)      MEC_13:
(2825)      <> == MEC
(2826)      <environ> == 'Ind.Tp _ SR'.
(2827)

```

```

(2828) MEC_14:
(2829) <> == MEC
(2830) <environ> == 'Ind.Tf _'.
(2831)
(2832) MEC_15:
(2833) <> == MEC
(2834) <environ> == 'Ind.Tip _ ~SR'.
(2835)
(2836) MEC_16:
(2837) <> == MEC
(2838) <environ> == '_ [Ind.S1.S2]'.
(2839)
(2840) MEC_17:
(2841) <> == MEC
(2842) <environ> == '_ Ind.S3s'.
(2843)
(2844) MEC_18:
(2845) <> == MEC
(2846) <environ> == '_ [Ind.S1]'.
(2847)
(2848) MEC_19:
(2849) <> == MEC
(2850) <environ> == '_ [Ind.S2]'.
(2851)
(2852) MEC_20:
(2853) <> == MEC
(2854) <environ> == '_ [Ind.S1.S3]'.
(2855)
(2856) MEC_21:
(2857) <> == MEC
(2858) <environ> == '_ [Ind.S3]'.
(2859)
(2860) MEC_22:
(2861) <> == MEC
(2862) <environ> == '_ [Ind.S1.S2.S3]'.
(2863)
(2864) MEC_23:
(2865) <> == MEC
(2866) <environ> == '_ [Ind.T]'.
(2867)
(2868) MEC_24:
(2869) <> == MEC
(2870) <environ> == 'this _'.
(2871)
(2872) MEC_25:
(2873) <> == MEC
(2874) <environ> == '[Contra.Subj] _'.
(2875)
(2876)
(2877)
(2878) %
(2879) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(2880)
(2881) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Categories %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(2882) %
(2883) % This section contains abstract entries that are
(2884) % language dependent

```

```

(2885)
(2886) C_MORPHEME:
(2887) <> ==
(2888) <amp; lex> == I_AMPLE
(2889) <dict lex> == I_DICT
(2890) <allo> == ALLOMORPH
(2891) <lex under> == "<phon under>"
(2892) <lex prop> ==
(2893) <lex noload> ==
(2894) <lex order class> ==
(2895) <lex form> == DEFAULT
(2896) <lex cat> ==
(2897) <lex loc> ==
(2898) <lex gloss nat> == 'No national language entry found'
(2899) <lex gloss eng> == 'No English entry found'
(2900) <lex gloss eng morname> == 'No English morph name found'.
(2901)
(2902) C_INFIX:
(2903) <> == C_MORPHEME
(2904) <lex type> == i.
(2905)
(2906) C_PREFIX:
(2907) <> == C_MORPHEME
(2908) <lex type> == p.
(2909)
(2910) C_ROOT:
(2911) <> == C_MORPHEME
(2912) <lex type> == r.
(2913)
(2914) C_R_Adj:
(2915) <> == C_ROOT
(2916) <lex cat> == 'Adj'.
(2917)
(2918) C_R_Adj_VR:
(2919) <> == C_R_VR
(2920) <lex under> == Implode: <"<phon under>" _2>
(2921) <allo mec> == "MEC_01".
(2922)
(2923) C_R_Adv:
(2924) <> == C_ROOT
(2925) <lex cat> == 'Adv'.
(2926)
(2927) C_R_Deictic:
(2928) <> == C_ROOT
(2929) <lex cat> == 'Deictic'.
(2930)
(2931) C_R_IP:
(2932) <> == C_ROOT
(2933) <lex cat> == 'IP'.
(2934)
(2935) C_R_NomNeg:
(2936) <> == C_ROOT
(2937) <lex cat> == 'NomNeg'.
(2938)
(2939) C_R_NW:
(2940) <> == C_ROOT
(2941) <lex cat> == 'NW'.

```

```

(2942)
(2943) C_R_PerPro:
(2944) <> == C_ROOT
(2945) <lex cat> == 'PerPro'.
(2946)
(2947) C_R_PNR:
(2948) <> == C_ROOT
(2949) <lex cat> == 'PNR'.
(2950)
(2951) C_R_PosPro:
(2952) <> == C_ROOT
(2953) <lex cat> == 'PosPro'.
(2954)
(2955) C_R_Prep:
(2956) <> == C_ROOT
(2957) <lex cat> == 'Prep'.
(2958)
(2959) C_R_RflPro:
(2960) <> == C_ROOT
(2961) <lex cat> == 'RflPro'.
(2962)
(2963) C_R_vAuxR:
(2964) <> == C_ROOT
(2965) <lex cat> == 'vAuxR'.
(2966)
(2967) C_R_VNeg:
(2968) <> == C_ROOT
(2969) <lex cat> == 'VNeg'.
(2970)
(2971) C_R_VR:
(2972) <> == C_ROOT
(2973) <lex prop> == 'MC2'
(2974) <lex cat> == 'VR'.
(2975)
(2976) C_R_VR_MC1:
(2977) <> == C_R_VR
(2978) <lex form> == DEFAULT SYNCOPE
(2979) <lex prop> == 'MC1'.
(2980)
(2981) C_R_VR_R:
(2982) <> == C_R_VR
(2983) <lex form> == DEFAULT REDUP
(2984) <lex prop> == 'MC2 RC1'.
(2985)
(2986) C_SUFFIX:
(2987) <> == C_MORPHEME
(2988) <lex type> == s.
(2989)
(2990) C_S_ASPECT:
(2991) <> == C_SUFFIX.
(2992)
(2993) C_S_ATTRIB:
(2994) <> == C_S_EPENTHETIC
(2995) <lex order class> == 110
(2996) <lex cat> == 'Adj/Atr'.
(2997)
(2998) C_S_BENEFACTIVE:

```

```

(2999)      <> == C_SUFFIX
(3000)      <lex prop> == 'MC2'
(3001)      <lex order class> == 60
(3002)      <lex cat> == 'VR/VR'.
(3003)
(3004)      C_S_BENEFACTIVE_MC1:
(3005)      <> == C_S_BENEFACTIVE
(3006)      <lex form> == DEFAULT SYNCOPE
(3007)      <lex prop> == 'MC1'.
(3008)
(3009)      C_S_CONTRA:
(3010)      <> == C_SUFFIX.
(3011)
(3012)      C_S_CONTRA_SUBJ:
(3013)      <> == C_S_CONTRA
(3014)      <lex form> == DEFAULT EPENTHETIC_W
(3015)      <allo sec default> == "SEC_03"
(3016)      <allo mec> == "MEC_05"
(3017)      <lex order class> == 80
(3018)      <lex cat> == 'VR/Contra SMV/Contra'.
(3019)
(3020)      C_S_DIFF_REF_TEMPORALITY:
(3021)      <> == C_S_TEMPORALITY
(3022)      <allo mec> == "MEC_23" "MEC_04"
(3023)      <lex cat> == 'VR/SMV'.
(3024)
(3025)      C_S_EPENTHETIC:
(3026)      <> == C_SUFFIX
(3027)      <lex form> == DEFAULT EPENTHETIC_W
(3028)      <allo sec default> == "SEC_03".
(3029)
(3030)      C_S_IND_SUBJ:
(3031)      <> == C_SUFFIX
(3032)      <lex order class> == 90
(3033)      <lex cat> == 'SFV/SFV SFV/SFVW SMV/SMV'.
(3034)
(3035)      C_S_IND_TENSE:
(3036)      <> == C_SUFFIX
(3037)      <allo sec default> == "SEC_03"
(3038)      <lex form> == DEFAULT EPENTHETIC_W
(3039)      <lex order class> == 80
(3040)      <lex cat> == 'SMV/SMV VR/SMV VR/SFV SFV/SFV'.
(3041)
(3042)      C_S_INF:
(3043)      <> == C_SUFFIX.
(3044)
(3045)      C_S_INF_SUBJ:
(3046)      <> == C_S_INF
(3047)      <lex order class> == 90
(3048)      <lex cat> == 'Inf/Inf'.
(3049)
(3050)      C_S_OBJECT:
(3051)      <> == C_SUFFIX
(3052)      <lex prop> == 'MC2'
(3053)      <lex order class> == 30
(3054)      <lex cat> == 'VR/VR'.
(3055)

```

```

(3056) C_S_OBJECT_MC1:
(3057) <> == C_S_OBJECT
(3058) <lex form> == DEFAULT SYNCOPE
(3059) <lex prop> == 'MC1'.
(3060)
(3061) C_S_PLURAL:
(3062) <> == C_SUFFIX
(3063) <lex order class> == 20
(3064) <lex cat> == 'VR/VR'.
(3065)
(3066) C_S_PRONOUN_POSS:
(3067) <> == C_SUFFIX
(3068) <lex order class> == 10
(3069) <lex cat> == 'PNR/PN'.
(3070)
(3071) C_S_SAME_REF_TEMPORALITY:
(3072) <> == C_S_TEMPORALITY
(3073) <lex cat> == 'VR/SMVW'.
(3074)
(3075) C_S_SUBJUNCTIVE:
(3076) <> == C_SUFFIX.
(3077)
(3078) C_S_SUBJUNCTIVE_SUBJ_NUMBER:
(3079) <> == C_S_SUBJUNCTIVE
(3080) <lex order class> == 90
(3081) <lex cat> == 'VSj/VSjs'.
(3082)
(3083) C_S_SUBJUNCTIVE_SUBJ_PERSON:
(3084) <> == C_S_SUBJUNCTIVE
(3085) <lex form> == DEFAULT EPENTHETIC_W
(3086) <lex order class> == 80
(3087) <lex cat> == 'VR/VSj'
(3088) <allo sec default> == "SEC_03".
(3089)
(3090) C_S_TEMPORALITY:
(3091) <> == C_S_VERBAL_SUFFIX
(3092) <lex order class> == 70.
(3093)
(3094) C_S_V_AUX:
(3095) <> == C_SUFFIX
(3096) <lex order class> == 90
(3097) <lex cat> == 'vAuxR/vAuxW'.
(3098)
(3099) C_S_VERBAL_SUFFIX:
(3100) <> == C_SUFFIX
(3101) <lex cat> == 'VR/VR'.
(3102)
(3103) %
(3104) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(3105)
(3106)
(3107) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% English %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(3108) %
(3109)
(3110) ENG_able:
(3111) <> == "R_edo"
(3112) <lex gloss eng> == 'able'.

```

(3113)
(3114) ENG_allow/leave:
(3115) <> == "R_awa"
(3116) <lex gloss eng> == 'allow/leave'.
(3117)
(3118) ENG_always:
(3119) <> == "R_gaigai"
(3120) <lex gloss eng> == 'always'.
(3121)
(3122) ENG_animal:
(3123) <> == "R_kaafa"
(3124) <lex gloss eng> == 'animal'.
(3125)
(3126) ENG_appear/arrive:
(3127) <> == "R_heige"
(3128) <lex gloss eng> == 'appear/arrive'.
(3129)
(3130) ENG_ask:
(3131) <> == "R_isoki"
(3132) <lex gloss eng> == 'ask'.
(3133)
(3134) ENG_at:
(3135) <> == "R_la"
(3136) <lex gloss eng> == 'at'.
(3137)
(3138) ENG_Attrib:
(3139) <> == "S_ou_Attrib"
(3140) <lex gloss eng morname> == 'Attrib'
(3141) <lex gloss eng> == 'Attributive marker'.
(3142)
(3143) ENG_B1d:
(3144) <> == "S_hari"
(3145) <lex gloss eng morname> == 'B1d'
(3146) <lex gloss eng> == 'for us (dual)'.
(3147)
(3148) ENG_B1p:
(3149) <> == "S_hagi"
(3150) <lex gloss eng morname> == 'B1p'
(3151) <lex gloss eng> == 'for us (plural)'.
(3152)
(3153) ENG_B1s:
(3154) <> == "S_hai"
(3155) <lex gloss eng morname> == 'B1s'
(3156) <lex gloss eng> == 'for me'.
(3157)
(3158) ENG_B2d:
(3159) <> == "S_tari"
(3160) <lex gloss eng morname> == 'B2d'
(3161) <lex gloss eng> == 'for you (dual)'.
(3162)
(3163) ENG_B2p:
(3164) <> == "S_tagi"
(3165) <lex gloss eng morname> == 'B2p'
(3166) <lex gloss eng> == 'for you (plural)'.
(3167)
(3168) ENG_B2s:
(3169) <> == "S_hani"

(3170) <lex gloss eng morname> == 'B2s'
 (3171) <lex gloss eng> == 'for you (singular)'.
 (3172)
 (3173) ENG_B3d:
 (3174) <> == "S_nari"
 (3175) <lex gloss eng morname> == 'B3d'
 (3176) <lex gloss eng> == 'for them (dual)'.
 (3177)
 (3178) ENG_B3p:
 (3179) <> == "S_nagi"
 (3180) <lex gloss eng morname> == 'B3p'
 (3181) <lex gloss eng> == 'for them (plural)'.
 (3182)
 (3183) ENG_B3s:
 (3184) <> == "S_fu"
 (3185) <lex gloss eng morname> == 'B3s'
 (3186) <lex gloss eng> == 'for him/her/it'.
 (3187)
 (3188) ENG_bad:
 (3189) <> == "R_inyaba_1"
 (3190) <lex gloss eng> == 'bad'.
 (3191)
 (3192) ENG_be:
 (3193) <> == "R_inyi"
 (3194) <lex gloss eng> == 'be'.
 (3195)
 (3196) ENG_become_bad:
 (3197) <> == "R_inyaba_2"
 (3198) <lex gloss eng> == 'become.bad'.
 (3199)
 (3200) ENG_become_crooked:
 (3201) <> == "R_kobibi_2"
 (3202) <lex gloss eng> == 'become.crooked'.
 (3203)
 (3204) ENG_become_good:
 (3205) <> == "R_hilou_2"
 (3206) <lex gloss eng> == 'become.good'.
 (3207)
 (3208) ENG_become_heavy:
 (3209) <> == "R_ou_2"
 (3210) <lex gloss eng> == 'become.heavy'.
 (3211)
 (3212) ENG_become_large:
 (3213) <> == "R_anyakaro_2"
 (3214) <lex gloss eng> == 'become.large'.
 (3215)
 (3216) ENG_become_many:
 (3217) <> == "R_baingaro_2"
 (3218) <lex gloss eng> == 'become.many'.
 (3219)
 (3220) ENG_become_nothing:
 (3221) <> == "R_uuwa_2"
 (3222) <lex gloss eng> == 'become.nothing'.
 (3223)
 (3224) ENG_become_red:
 (3225) <> == "R_giiri_2"
 (3226) <lex gloss eng> == 'become.red'.

```

(3227)
(3228)   ENG_become_short:
(3229)   <> == "R_tutu_2"
(3230)   <lex gloss eng> == 'become.short'.
(3231)
(3232)   ENG_become_small:
(3233)   <> == "R_menekele_2"
(3234)   <lex gloss eng> == 'become.small'.
(3235)
(3236)   ENG_become_straight:
(3237)   <> == "R_tigini_2"
(3238)   <lex gloss eng> == 'become.straight'.
(3239)
(3240)   ENG_become_true:
(3241)   <> == "R_ngalenga_2"
(3242)   <lex gloss eng> == 'become.true'.
(3243)
(3244)   ENG_become_wild:
(3245)   <> == "R_kui_2"
(3246)   <lex gloss eng> == 'become.wild'.
(3247)
(3248)   ENG_bound:
(3249)   <> == "R_dage"
(3250)   <lex gloss eng> == 'bound'.
(3251)
(3252)   ENG_break_in_two:
(3253)   <> == "R_kou"
(3254)   <lex gloss eng> == 'break.in.two'.
(3255)
(3256)   ENG_causative:
(3257)   <> == "S_mu"
(3258)   <lex gloss eng> == 'causative'.
(3259)
(3260)   ENG_certainty:
(3261)   <> == "S_nga_2"
(3262)   <lex gloss eng> == 'certainty'.
(3263)
(3264)   ENG_chase:
(3265)   <> == "R_soo"
(3266)   <lex gloss eng> == 'chase'.
(3267)
(3268)   ENG_cloth:
(3269)   <> == "R_ogola"
(3270)   <lex gloss eng> == 'cloth'.
(3271)
(3272)   ENG_come_down:
(3273)   <> == "R_me"
(3274)   <lex gloss eng> == 'come.down'.
(3275)
(3276)   ENG_come_up:
(3277)   <> == "R_mai"
(3278)   <lex gloss eng> == 'come.up'.
(3279)
(3280)   ENG_completely:
(3281)   <> == "S_boro"
(3282)   <lex gloss eng> == 'completely'.
(3283)

```

(3284) ENG_connect/join:
(3285) <> == "R_bagumu"
(3286) <lex gloss eng> == 'connect/join'.
(3287)
(3288) ENG_consume:
(3289) <> == "R_nyi"
(3290) <lex gloss eng> == 'consume'.
(3291)
(3292) ENG_Contra:
(3293) <> == "S_ge_1"
(3294) <lex gloss eng morname> == 'Contra'
(3295) <lex gloss eng> == 'Contrafactual'.
(3296)
(3297) ENG_Contra_S1d:
(3298) <> == "S_edere"
(3299) <lex gloss eng morname> == 'Contra.S1d'
(3300) <lex gloss eng> == 'Contractual first dual subject'.
(3301)
(3302) ENG_Contra_S1p:
(3303) <> == "S_ede"
(3304) <lex gloss eng morname> == 'Contra.S1p'
(3305) <lex gloss eng> == 'Contractual first plural subject'.
(3306)
(3307) ENG_Contra_S1s:
(3308) <> == "S_e_Contra"
(3309) <lex gloss eng morname> == 'Contra.S1s'
(3310) <lex gloss eng> == 'Contractual first singular subject'.
(3311)
(3312) ENG_Contra_S2/3s/d/p:
(3313) <> == "S_a_1"
(3314) <lex gloss eng morname> == 'Contra.S2/3s/d/p'
(3315) <lex gloss eng> == 'Contractual second or third singular,
(3316) dual, or plural subject'.
(3317)
(3318) ENG_crooked:
(3319) <> == "R_kobibi_1"
(3320) <lex gloss eng> == 'crooked'.
(3321)
(3322) ENG_cut:
(3323) <> == "R_karu"
(3324) <lex gloss eng> == 'cut'.
(3325)
(3326) ENG_deflect:
(3327) <> == "R_ariri"
(3328) <lex gloss eng> == 'deflect'.
(3329)
(3330) ENG_desire:
(3331) <> == "R_gau"
(3332) <lex gloss eng> == 'desire'.
(3333)
(3334) ENG_die:
(3335) <> == "R_ume"
(3336) <lex gloss eng> == 'die'.
(3337)
(3338) ENG_different:
(3339) <> == "R_fere"
(3340) <lex gloss eng> == 'different'.

(3341)
(3342) ENG_do_not:
(3343) <> == "R_adai"
(3344) <lex gloss eng> == 'do.not'.
(3345)
(3346) ENG_do_thus:
(3347) <> == "R_ono"
(3348) <lex gloss eng> == 'do.thus'.
(3349)
(3350) ENG_do_work:
(3351) <> == "R_haruwe_2"
(3352) <lex gloss eng> == 'do.work'.
(3353)
(3354) ENG_dream:
(3355) <> == "R_wewe"
(3356) <lex gloss eng> == 'dream'.
(3357)
(3358) ENG_encircle:
(3359) <> == "R_kati"
(3360) <lex gloss eng> == 'encircle'.
(3361)
(3362) ENG_face:
(3363) <> == "R_hogo"
(3364) <lex gloss eng> == 'face'.
(3365)
(3366) ENG_fear:
(3367) <> == "R_umuge"
(3368) <lex gloss eng> == 'fear'.
(3369)
(3370) ENG_fight:
(3371) <> == "R_wara"
(3372) <lex gloss eng> == 'fight'.
(3373)
(3374) ENG_fill:
(3375) <> == "R_bolo"
(3376) <lex gloss eng> == 'fill'.
(3377)
(3378) ENG_finish:
(3379) <> == "R_usu"
(3380) <lex gloss eng> == 'finish'.
(3381)
(3382) ENG_fly:
(3383) <> == "R_fililai"
(3384) <lex gloss eng> == 'fly'.
(3385)
(3386) ENG_follow:
(3387) <> == "R_oojo"
(3388) <lex gloss eng> == 'follow'.
(3389)
(3390) ENG_gather:
(3391) <> == "R_elege"
(3392) <lex gloss eng> == 'gather'.
(3393)
(3394) ENG_get/take:
(3395) <> == "R_te"
(3396) <lex gloss eng> == 'get/take'.
(3397)

```

(3398)   ENG_get_0ld:
(3399)   <> == "R_haire"
(3400)   <lex gloss eng morname> == 'get.0ld'
(3401)   <lex gloss eng> == 'get us (dual)'.
(3402)
(3403)   ENG_get_0lp:
(3404)   <> == "R_haige"
(3405)   <lex gloss eng morname> == 'get.0lp'
(3406)   <lex gloss eng> == 'get us (plural)'.
(3407)
(3408)   ENG_get_0ls:
(3409)   <> == "R_haife"
(3410)   <lex gloss eng morname> == 'get.0ls'
(3411)   <lex gloss eng> == 'get me'.
(3412)
(3413)   ENG_get_02d:
(3414)   <> == "R_hatire"
(3415)   <lex gloss eng morname> == 'get.02d'
(3416)   <lex gloss eng> == 'get you (dual)'.
(3417)
(3418)   ENG_get_02p:
(3419)   <> == "R_hatige"
(3420)   <lex gloss eng morname> == 'get.02p'
(3421)   <lex gloss eng> == 'get you (plural)'.
(3422)
(3423)   ENG_get_02s:
(3424)   <> == "R_haine"
(3425)   <lex gloss eng morname> == 'get.02s'
(3426)   <lex gloss eng> == 'get you (singular)'.
(3427)
(3428)   ENG_get_03d:
(3429)   <> == "R_hanire"
(3430)   <lex gloss eng morname> == 'get.03d'
(3431)   <lex gloss eng> == 'get them (dual)'.
(3432)
(3433)   ENG_get_03p:
(3434)   <> == "R_hanige"
(3435)   <lex gloss eng morname> == 'get.03p'
(3436)   <lex gloss eng> == 'get them (plural)'.
(3437)
(3438)   ENG_get_03s:
(3439)   <> == "R_hau"
(3440)   <lex gloss eng morname> == 'get.03s'
(3441)   <lex gloss eng> == 'get him/her/it'.
(3442)
(3443)   ENG_get_up:
(3444)   <> == "R_jai"
(3445)   <lex gloss eng> == 'get.up'.
(3446)
(3447)   ENG_give_0ld:
(3448)   <> == "R_hiri"
(3449)   <lex gloss eng morname> == 'give.0ld'
(3450)   <lex gloss eng> == 'give us (dual)'.
(3451)
(3452)   ENG_give_0lp:
(3453)   <> == "R_higi"
(3454)   <lex gloss eng morname> == 'give.0lp'

```

(3455) <lex gloss eng> == 'give us (plural)'.
 (3456)
 (3457) ENG_give_01s:
 (3458) <> == "R_hii"
 (3459) <lex gloss eng morname> == 'give.01s'
 (3460) <lex gloss eng> == 'give me'.
 (3461)
 (3462) ENG_give_02d:
 (3463) <> == "R_tiri"
 (3464) <lex gloss eng morname> == 'give.02d'
 (3465) <lex gloss eng> == 'give you (dual)'.
 (3466)
 (3467) ENG_give_02p:
 (3468) <> == "R_tigi"
 (3469) <lex gloss eng morname> == 'give.02p'
 (3470) <lex gloss eng> == 'give you (plural)'.
 (3471)
 (3472) ENG_give_02s:
 (3473) <> == "R_hini"
 (3474) <lex gloss eng morname> == 'give.02s'
 (3475) <lex gloss eng> == 'give you (singular)'.
 (3476)
 (3477) ENG_give_03d:
 (3478) <> == "R_niri"
 (3479) <lex gloss eng morname> == 'give.03d'
 (3480) <lex gloss eng> == 'give them (dual)'.
 (3481)
 (3482) ENG_give_03p:
 (3483) <> == "R_nigi"
 (3484) <lex gloss eng morname> == 'give.03p'
 (3485) <lex gloss eng> == 'give them (plural)'.
 (3486)
 (3487) ENG_give_03s:
 (3488) <> == "R_tu"
 (3489) <lex gloss eng morname> == 'give.03s'
 (3490) <lex gloss eng> == 'give him/her/it'.
 (3491)
 (3492) ENG_go:
 (3493) <> == "R_goi"
 (3494) <lex gloss eng> == 'go'.
 (3495)
 (3496) ENG_go_down:
 (3497) <> == "R_mini"
 (3498) <lex gloss eng> == 'go.down'.
 (3499)
 (3500) ENG_good:
 (3501) <> == "R_hilou"
 (3502) <lex gloss eng> == 'good'.
 (3503)
 (3504) ENG_Hab:
 (3505) <> == "S_g_1"
 (3506) <lex gloss eng morname> == 'Hab'
 (3507) <lex gloss eng> == 'Habitual'.
 (3508)
 (3509) ENG_hand:
 (3510) <> == "R_owo"
 (3511) <lex gloss eng> == 'hand'.

(3512)
(3513) ENG_he/she/it:
(3514) <> == "R_no"
(3515) <lex gloss eng> == 'he/she/it'.
(3516)
(3517) ENG_hear/understand/obey:
(3518) <> == "R_isi"
(3519) <lex gloss eng> == 'hear/understand/obey'.
(3520)
(3521) ENG_heavy:
(3522) <> == "R_ou_1"
(3523) <lex gloss eng> == 'heavy'.
(3524)
(3525) ENG_help:
(3526) <> == "R_hoyo"
(3527) <lex gloss eng> == 'help'.
(3528)
(3529) ENG_himself/herself/itself:
(3530) <> == "R_nogo"
(3531) <lex gloss eng> == 'himself/herself/itself'.
(3532)
(3533) ENG_his/her/its:
(3534) <> == "R_nomo"
(3535) <lex gloss eng> == 'his/her/its'.
(3536)
(3537) ENG_hit_Old:
(3538) <> == "R_harire"
(3539) <lex gloss eng morname> == 'hit.Old'
(3540) <lex gloss eng> == 'hit us (dual)'.
(3541)
(3542) ENG_hit_O1p:
(3543) <> == "R_harige"
(3544) <lex gloss eng morname> == 'hit.O1p'
(3545) <lex gloss eng> == 'hit us (plural)'.
(3546)
(3547) ENG_hit_O1s:
(3548) <> == "R_yari"
(3549) <lex gloss eng morname> == 'hit.O1s'
(3550) <lex gloss eng> == 'hit me'.
(3551)
(3552) ENG_hit_O2d:
(3553) <> == "R_tarire"
(3554) <lex gloss eng morname> == 'hit.O2d'
(3555) <lex gloss eng> == 'hit you (dual)'.
(3556)
(3557) ENG_hit_O2p:
(3558) <> == "R_tarige"
(3559) <lex gloss eng morname> == 'hit.O2p'
(3560) <lex gloss eng> == 'hit you (plural)'.
(3561)
(3562) ENG_hit_O2s:
(3563) <> == "R_nari"
(3564) <lex gloss eng morname> == 'hit.O2s'
(3565) <lex gloss eng> == 'hit you (singular)'.
(3566)
(3567) ENG_hit_O3d:
(3568) <> == "R_narire"

```

(3569) <lex gloss eng morname> == 'hit.O3d'
(3570) <lex gloss eng> == 'hit them (dual)'.
(3571)
(3572) ENG_hit_O3p:
(3573) <> == "R_narige"
(3574) <lex gloss eng morname> == 'hit.O3p'
(3575) <lex gloss eng> == 'hit them (plural)'.
(3576)
(3577) ENG_hit_O3s:
(3578) <> == "R_wari"
(3579) <lex gloss eng morname> == 'hit.O3s'
(3580) <lex gloss eng> == 'hit him/her/it'.
(3581)
(3582) ENG_I:
(3583) <> == "R_ji"
(3584) <lex gloss eng> == 'I'.
(3585)
(3586) ENG_idle:
(3587) <> == "R_ewe"
(3588) <lex gloss eng> == 'idle'.
(3589)
(3590) ENG_Ind_S1d:
(3591) <> == "S_Ind_S1d"
(3592) <lex gloss eng morname> == 'Ind.S1d'
(3593) <lex gloss eng> == 'Indicative first dual subject'.
(3594)
(3595) ENG_Ind_S1p:
(3596) <> == "S_Ind_S1p"
(3597) <lex gloss eng morname> == 'Ind.S1p'
(3598) <lex gloss eng> == 'Indicative first plural subject'.
(3599)
(3600) ENG_Ind_S1s:
(3601) <> == "S_Ind_S1s"
(3602) <lex gloss eng morname> == 'Ind.S1s'
(3603) <lex gloss eng> == 'Indicative first singular subject'.
(3604)
(3605) ENG_Ind_S2d:
(3606) <> == "S_Ind_S2d"
(3607) <lex gloss eng morname> == 'Ind.S2d'
(3608) <lex gloss eng> == 'Indicative second dual subject'.
(3609)
(3610) ENG_Ind_S2p:
(3611) <> == "S_Ind_S2p"
(3612) <lex gloss eng morname> == 'Ind.S2p'
(3613) <lex gloss eng> == 'Indicative second plural subject'.
(3614)
(3615) ENG_Ind_S2s:
(3616) <> == "S_Ind_S2s"
(3617) <lex gloss eng morname> == 'Ind.S2s'
(3618) <lex gloss eng> == 'Indicative second singular subject'.
(3619)
(3620) ENG_Ind_S3d:
(3621) <> == "S_Ind_S3d"
(3622) <lex gloss eng morname> == 'Ind.S3d'
(3623) <lex gloss eng> == 'Indicative third dual subject'.
(3624)
(3625) ENG_Ind_S3p:

```


(3626) <> == "S_Ind_S3p"
(3627) <lex gloss eng morname> == 'Ind.S3p'
(3628) <lex gloss eng> == 'Indicative third plural subject'.
(3629)
(3630) ENG_Ind_S3s:
(3631) <> == "S_Ind_S3s"
(3632) <lex gloss eng morname> == 'Ind.S3s'
(3633) <lex gloss eng> == 'Indicative third singular subject'.
(3634)
(3635) ENG_Ind_Tf:
(3636) <> == "S_Ind_Tf"
(3637) <lex gloss eng morname> == 'Ind.Tf'
(3638) <lex gloss eng> == 'Indicative future tense'.
(3639)
(3640) ENG_Ind_Tip:
(3641) <> == "S_Ind_Tip"
(3642) <lex gloss eng morname> == 'Ind.Tip'
(3643) <lex gloss eng> == 'Indicative immediate past tense'.
(3644)
(3645) ENG_Ind_Tp:
(3646) <> == "S_Ind_Tp"
(3647) <lex gloss eng morname> == 'Ind.Tp'
(3648) <lex gloss eng> == 'Indicative present tense'.
(3649)
(3650) ENG_Ind_Trp:
(3651) <> == "S_Ind_Trp"
(3652) <lex gloss eng morname> == 'Ind.Trp'
(3653) <lex gloss eng> == 'Indicative remote past tense'.
(3654)
(3655) ENG_Inf:
(3656) <> == "S_e_Inf"
(3657) <lex gloss eng morname> == 'Inf'
(3658) <lex gloss eng> == 'Infinitive'.
(3659)
(3660) ENG_Inf_Sd:
(3661) <> == "S_ru_Inf"
(3662) <lex gloss eng morname> == 'Inf.Sd'
(3663) <lex gloss eng> == 'Infinitive subject'.
(3664)
(3665) ENG_Inf_Sp:
(3666) <> == "S_gu_Inf"
(3667) <lex gloss eng morname> == 'Inf.Sp'
(3668) <lex gloss eng> == 'Infinitive plural subject'.
(3669)
(3670) ENG_Inf_Ss:
(3671) <> == "S_i_Inf"
(3672) <lex gloss eng morname> == 'Inf.Ss'
(3673) <lex gloss eng> == 'Infinitive singular subject'.
(3674)
(3675) ENG_inside:
(3676) <> == "R_huwanya"
(3677) <lex gloss eng> == 'inside'.
(3678)
(3679) ENG_judge:
(3680) <> == "R_haawe"
(3681) <lex gloss eng> == 'judge'.
(3682)

```

(3683)   ENG_just/only:
(3684)   <> == "S_nga_1"
(3685)   <lex gloss eng> == 'just/only'.
(3686)
(3687)   ENG_knife:
(3688)   <> == "R_kesebu"
(3689)   <lex gloss eng> == 'knife'.
(3690)
(3691)   ENG_large:
(3692)   <> == "R_anyakaro_1"
(3693)   <lex gloss eng> == 'large'.
(3694)
(3695)   ENG_leg/foot:
(3696)   <> == "R_afe"
(3697)   <lex gloss eng> == 'leg/foot'.
(3698)
(3699)   ENG_leg/foot_3sP:
(3700)   <> == "R_afo"
(3701)   <lex gloss eng morname> == 'leg/foot.3sP'
(3702)   <lex gloss eng> == 'his/her/its leg/foot'.
(3703)
(3704)   ENG_lie:
(3705)   <> == "R_yaawa"
(3706)   <lex gloss eng> == 'lie'.
(3707)
(3708)   ENG_light:
(3709)   <> == "R_lala_1"
(3710)   <lex gloss eng> == 'light'.
(3711)
(3712)   ENG_like_that:
(3713)   <> == "R_onou"
(3714)   <lex gloss eng> == 'like.that'.
(3715)
(3716)   ENG_like1:
(3717)   <> == "S_fe_1"
(3718)   <lex gloss eng> == 'like1'.
(3719)
(3720)   ENG_like2:
(3721)   <> == "S_neu"
(3722)   <lex gloss eng> == 'like2'.
(3723)
(3724)   ENG_look_for:
(3725)   <> == "R_kuru"
(3726)   <lex gloss eng> == 'look.for'.
(3727)
(3728)   ENG_man:
(3729)   <> == "R_fai"
(3730)   <lex gloss eng> == 'man'.
(3731)
(3732)   ENG_many:
(3733)   <> == "R_baingaro_1"
(3734)   <lex gloss eng> == 'many'.
(3735)
(3736)   ENG_my:
(3737)   <> == "R_yame"
(3738)   <lex gloss eng> == 'my'.
(3739)

```

(3740) ENG_myself:
(3741) <> == "R_yage"
(3742) <lex gloss eng> == 'myself'.
(3743)
(3744) ENG_N_not:
(3745) <> == "R_uuwa_1"
(3746) <lex gloss eng morname> == 'N.not'
(3747) <lex gloss eng> == 'not'.
(3748)
(3749) ENG_Nom:
(3750) <> == "S_ou_Nom"
(3751) <lex gloss eng morname> == 'Nom'
(3752) <lex gloss eng> == 'Nominalizer'.
(3753)
(3754) ENG_nose:
(3755) <> == "R_mutu"
(3756) <lex gloss eng> == 'nose'.
(3757)
(3758) ENG_Old:
(3759) <> == "S_hiri"
(3760) <lex gloss eng morname> == 'Old'
(3761) <lex gloss eng> == 'to us (dual)'.
(3762)
(3763) ENG_O1p:
(3764) <> == "S_higi"
(3765) <lex gloss eng morname> == 'O1p'
(3766) <lex gloss eng> == 'to us (plural)'.
(3767)
(3768) ENG_O1s:
(3769) <> == "S_hii"
(3770) <lex gloss eng morname> == 'O1s'
(3771) <lex gloss eng> == 'to me'.
(3772)
(3773) ENG_O2d:
(3774) <> == "S_tiri"
(3775) <lex gloss eng morname> == 'O2d'
(3776) <lex gloss eng> == 'to you (dual)'.
(3777)
(3778) ENG_O2p:
(3779) <> == "S_tigi"
(3780) <lex gloss eng morname> == 'O2p'
(3781) <lex gloss eng> == 'to you (plural)'.
(3782)
(3783) ENG_O2s:
(3784) <> == "S_hini"
(3785) <lex gloss eng morname> == 'O2s'
(3786) <lex gloss eng> == 'to you (singular)'.
(3787)
(3788) ENG_O3d:
(3789) <> == "S_niri"
(3790) <lex gloss eng morname> == 'O3d'
(3791) <lex gloss eng> == 'to them (dual)'.
(3792)
(3793) ENG_O3p:
(3794) <> == "S_nigi"
(3795) <lex gloss eng morname> == 'O3p'
(3796) <lex gloss eng> == 'to them (plural)'.

(3797)
(3798) ENG_O3s:
(3799) <> == "S_tu"
(3800) <lex gloss eng morname> == 'O3s'
(3801) <lex gloss eng> == 'to him/her/it'.
(3802)
(3803) ENG_ocean:
(3804) <> == "R_hiiri"
(3805) <lex gloss eng> == 'ocean'.
(3806)
(3807) ENG_one/a:
(3808) <> == "R_ete"
(3809) <lex gloss eng> == 'one/a'.
(3810)
(3811) ENG_our_dual:
(3812) <> == "R_heire"
(3813) <lex gloss eng> == 'our.dual'.
(3814)
(3815) ENG_our_plural:
(3816) <> == "R_ebere"
(3817) <lex gloss eng> == 'our.plural'.
(3818)
(3819) ENG_ourselves_dual:
(3820) <> == "R_hare"
(3821) <lex gloss eng> == 'ourselves.dual'.
(3822)
(3823) ENG_ourselves_plural:
(3824) <> == "R_hage"
(3825) <lex gloss eng> == 'ourselves.plural'.
(3826)
(3827) ENG_p1:
(3828) <> == "S_gana"
(3829) <lex gloss eng morname> == 'p1'
(3830) <lex gloss eng> == 'plural'.
(3831)
(3832) ENG_P1d:
(3833) <> == "S_re_2"
(3834) <lex gloss eng morname> == 'P1d'
(3835) <lex gloss eng> == 'first dual possessive'.
(3836)
(3837) ENG_P1p:
(3838) <> == "S_ge_2"
(3839) <lex gloss eng morname> == 'P1p'
(3840) <lex gloss eng> == 'first plural possessive'.
(3841)
(3842) ENG_P1s:
(3843) <> == "S_fe_2"
(3844) <lex gloss eng morname> == 'P1s'
(3845) <lex gloss eng> == 'first singular possessive'.
(3846)
(3847) ENG_p2:
(3848) <> == "S_ru_1"
(3849) <lex gloss eng morname> == 'p2'
(3850) <lex gloss eng> == 'plural'.
(3851)
(3852) ENG_P2d:
(3853) <> == "S_tere"

(3854) <lex gloss eng morname> == 'P2d'
 (3855) <lex gloss eng> == 'second dual possessive'.
 (3856)
 (3857) ENG_P2p:
 (3858) <> == "S_tege"
 (3859) <lex gloss eng morname> == 'P2p'
 (3860) <lex gloss eng> == 'second plural possessive'.
 (3861)
 (3862) ENG_P2s:
 (3863) <> == "S_ne_2"
 (3864) <lex gloss eng morname> == 'P2s'
 (3865) <lex gloss eng> == 'second singular possessive'.
 (3866)
 (3867) ENG_P3d:
 (3868) <> == "S_nere"
 (3869) <lex gloss eng morname> == 'P3d'
 (3870) <lex gloss eng> == 'third dual possessive'.
 (3871)
 (3872) ENG_P3p:
 (3873) <> == "S_nege"
 (3874) <lex gloss eng morname> == 'P3p'
 (3875) <lex gloss eng> == 'third plural possessive'.
 (3876)
 (3877) ENG_P3s:
 (3878) <> == "S_0_Pro_Poss"
 (3879) <lex gloss eng morname> == 'P3s'
 (3880) <lex gloss eng> == 'third singular possessive'.
 (3881)
 (3882) ENG_perf:
 (3883) <> == "S_ha_1"
 (3884) <lex gloss eng morname> == 'perf'
 (3885) <lex gloss eng> == 'perfective'.
 (3886)
 (3887) ENG_pertaining_to:
 (3888) <> == "S_ha_2"
 (3889) <lex gloss eng> == 'pertaining.to'.
 (3890)
 (3891) ENG_pig:
 (3892) <> == "R_buuwa"
 (3893) <lex gloss eng> == 'pig'.
 (3894)
 (3895) ENG_place:
 (3896) <> == "R_ha"
 (3897) <lex gloss eng> == 'place'.
 (3898)
 (3899) ENG_plant:
 (3900) <> == "R_tau"
 (3901) <lex gloss eng> == 'plant'.
 (3902)
 (3903) ENG_poke:
 (3904) <> == "R_tuN"
 (3905) <lex gloss eng> == 'poke'.
 (3906)
 (3907) ENG_pour_out:
 (3908) <> == "R_feu"
 (3909) <lex gloss eng> == 'pour.out'.
 (3910)

```

(3911)   ENG_Prediction:
(3912)   <> == "S_ga"
(3913)   <lex gloss eng> == 'Prediction'.
(3914)
(3915)   ENG_prepare:
(3916)   <> == "R_haiya"
(3917)   <lex gloss eng> == 'prepare'.
(3918)
(3919)   ENG_press:
(3920)   <> == "R_ilei"
(3921)   <lex gloss eng> == 'press'.
(3922)
(3923)   ENG_put:
(3924)   <> == "R_tafa"
(3925)   <lex gloss eng> == 'put'.
(3926)
(3927)   ENG_quickly:
(3928)   <> == "R_fasa"
(3929)   <lex gloss eng> == 'quickly'.
(3930)
(3931)   ENG_quickly_2:
(3932)   <> == "S_ya_Manner"
(3933)   <lex gloss eng> == 'quickly'.
(3934)
(3935)   ENG_red:
(3936)   <> == "R_giiri_1"
(3937)   <lex gloss eng> == 'red'.
(3938)
(3939)   ENG_remove:
(3940)   <> == "R_uulu"
(3941)   <lex gloss eng> == 'remove'.
(3942)
(3943)   ENG_run_away:
(3944)   <> == "R_faga"
(3945)   <lex gloss eng> == 'run.away'.
(3946)
(3947)   ENG_see_Old:
(3948)   <> == "R_herire"
(3949)   <lex gloss eng morname> == 'see.Old'
(3950)   <lex gloss eng> == 'see us (dual)'.
(3951)
(3952)   ENG_see_O1p:
(3953)   <> == "R_herige"
(3954)   <lex gloss eng morname> == 'see.O1p'
(3955)   <lex gloss eng> == 'see us (plural)'.
(3956)
(3957)   ENG_see_O1s:
(3958)   <> == "R_yeriye"
(3959)   <lex gloss eng morname> == 'see.O1s'
(3960)   <lex gloss eng> == 'see me'.
(3961)
(3962)   ENG_see_O2d:
(3963)   <> == "R_terire"
(3964)   <lex gloss eng morname> == 'see.O2d'
(3965)   <lex gloss eng> == 'see you (dual)'.
(3966)
(3967)   ENG_see_O2p:

```

(3968) <> == "R_terige"
 (3969) <lex gloss eng morname> == 'see.02p'
 (3970) <lex gloss eng> == 'see you (plural)'.
 (3971)
 (3972) ENG_see_02s:
 (3973) <> == "R_neriye"
 (3974) <lex gloss eng morname> == 'see.02s'
 (3975) <lex gloss eng> == 'see you (singular)'.
 (3976)
 (3977) ENG_see_03d:
 (3978) <> == "R_nerire"
 (3979) <lex gloss eng morname> == 'see.03d'
 (3980) <lex gloss eng> == 'see them (dual)'.
 (3981)
 (3982) ENG_see_03p:
 (3983) <> == "R_nerige"
 (3984) <lex gloss eng morname> == 'see.03p'
 (3985) <lex gloss eng> == 'see them (plural)'.
 (3986)
 (3987) ENG_see_03s:
 (3988) <> == "R_ure"
 (3989) <lex gloss eng morname> == 'see.03s'
 (3990) <lex gloss eng> == 'see him/her/it'.
 (3991)
 (3992) ENG_send:
 (3993) <> == "R_huuru"
 (3994) <lex gloss eng> == 'send'.
 (3995)
 (3996) ENG_shine:
 (3997) <> == "R_lala_2"
 (3998) <lex gloss eng> == 'shine'.
 (3999)
 (4000) ENG_short:
 (4001) <> == "R_tutu_1"
 (4002) <lex gloss eng> == 'short'.
 (4003)
 (4004) ENG_shove:
 (4005) <> == "R_nuu"
 (4006) <lex gloss eng> == 'shove'.
 (4007)
 (4008) ENG_show:
 (4009) <> == "R_abiti"
 (4010) <lex gloss eng> == 'show'.
 (4011)
 (4012) ENG_sibling_same_sex_his/her/its:
 (4013) <> == "R_bemu"
 (4014) <lex gloss eng> == 'sibling.same.sex.his/her/its'.
 (4015)
 (4016) ENG_sibling_same_sex_possessive:
 (4017) <> == "R_bai_2"
 (4018) <lex gloss eng> == 'sibling.same.sex.possessive'.
 (4019)
 (4020) ENG_sibling_same_sex_referential:
 (4021) <> == "R_babo"
 (4022) <lex gloss eng> == 'sibling.same.sex.referential'.
 (4023)
 (4024) ENG_sit:

```

(4025)      <> == "R_yafa"
(4026)      <lex gloss eng> == 'sit'.
(4027)
(4028)      ENG_Sj_S1/3:
(4029)      <> == "S_o_Subj"
(4030)      <lex gloss eng morname> == 'Sj.S1/3'
(4031)      <lex gloss eng> == 'subjunctive first or third subject'.
(4032)
(4033)      ENG_Sj_S2:
(4034)      <> == "S_a_2"
(4035)      <lex gloss eng morname> == 'Sj.S2'
(4036)      <lex gloss eng> == 'subjunctive second subject'.
(4037)
(4038)      ENG_Sj_Sd:
(4039)      <> == "S_ru_2"
(4040)      <lex gloss eng morname> == 'Sj.Sd'
(4041)      <lex gloss eng> == 'subjunctive dual subject'.
(4042)
(4043)      ENG_Sj_Sp:
(4044)      <> == "S_gu_Subj"
(4045)      <lex gloss eng morname> == 'Sj.Sp'
(4046)      <lex gloss eng> == 'subjunctive plural subject'.
(4047)
(4048)      ENG_Sj_Ss:
(4049)      <> == "S_u"
(4050)      <lex gloss eng morname> == 'Sj.Ss'
(4051)      <lex gloss eng> == 'subjunctive singular subject'.
(4052)
(4053)      ENG_sleep_pl:
(4054)      <> == "R_jiri"
(4055)      <lex gloss eng> == 'sleep.pl'.
(4056)
(4057)      ENG_small:
(4058)      <> == "R_menekele_1"
(4059)      <lex gloss eng> == 'small'.
(4060)
(4061)      ENG_smooth:
(4062)      <> == "R_fii"
(4063)      <lex gloss eng> == 'smooth'.
(4064)
(4065)      ENG_some:
(4066)      <> == "R_ila"
(4067)      <lex gloss eng> == 'some'.
(4068)
(4069)      ENG_speak:
(4070)      <> == "R_le"
(4071)      <lex gloss eng> == 'speak'.
(4072)
(4073)      ENG_SR:
(4074)      <> == "S_nga_3"
(4075)      <lex gloss eng morname> == 'SR'
(4076)      <lex gloss eng> == 'switch referent'.
(4077)
(4078)      ENG_SR_TO:
(4079)      <> == "S_g_2"
(4080)      <lex gloss eng morname> == 'SR.TO'
(4081)      <lex gloss eng> == 'switch referent with temporal overlap'.

```



```

(4082)
(4083)   ENG_SR_TS:
(4084)   <> == "S_0_1"
(4085)   <lex gloss eng morname> == 'SR.TS'
(4086)   <lex gloss eng> == 'switch referent with temporal succession'.
(4087)
(4088)   ENG_stand:
(4089)   <> == "R_oto"
(4090)   <lex gloss eng> == 'stand'.
(4091)
(4092)   ENG_stick_to:
(4093)   <> == "R_fada"
(4094)   <lex gloss eng> == 'stick.to'.
(4095)
(4096)   ENG_stone:
(4097)   <> == "R_megebu"
(4098)   <lex gloss eng> == 'stone'.
(4099)
(4100)   ENG_straight:
(4101)   <> == "R_tigini"
(4102)   <lex gloss eng> == 'straight'.
(4103)
(4104)   ENG_talk/word:
(4105)   <> == "R_beele"
(4106)   <lex gloss eng> == 'talk/word'.
(4107)
(4108)   ENG_teach:
(4109)   <> == "R_agotete"
(4110)   <lex gloss eng> == 'teach'.
(4111)
(4112)   ENG_tear:
(4113)   <> == "R_korai"
(4114)   <lex gloss eng> == 'tear'.
(4115)
(4116)   ENG_tell:
(4117)   <> == "R_hai"
(4118)   <lex gloss eng> == 'tell'.
(4119)
(4120)   ENG_that:
(4121)   <> == "R_wa"
(4122)   <lex gloss eng> == 'that'.
(4123)
(4124)   ENG_their_dual:
(4125)   <> == "R_neire"
(4126)   <lex gloss eng> == 'their.dual'.
(4127)
(4128)   ENG_their_plural:
(4129)   <> == "R_nebere"
(4130)   <lex gloss eng> == 'their.plural'.
(4131)
(4132)   ENG_theirselves_dual:
(4133)   <> == "R_nare"
(4134)   <lex gloss eng> == 'theirselves.dual'.
(4135)
(4136)   ENG_theirselves_plural:
(4137)   <> == "R_nage"
(4138)   <lex gloss eng> == 'theirselves.plural'.

```

(4139)
(4140) ENG_they:
(4141) <> == "R_nere"
(4142) <lex gloss eng> == 'they'.
(4143)
(4144) ENG_thigh:
(4145) <> == "R_gaigu"
(4146) <lex gloss eng> == 'thigh'.
(4147)
(4148) ENG_thing:
(4149) <> == "R_dada"
(4150) <lex gloss eng> == 'thing'.
(4151)
(4152) ENG_this:
(4153) <> == "R_be"
(4154) <lex gloss eng> == 'this'.
(4155)
(4156) ENG_thought:
(4157) <> == "R_nina"
(4158) <lex gloss eng> == 'thought'.
(4159)
(4160) ENG_throw:
(4161) <> == "R_aya"
(4162) <lex gloss eng> == 'throw'.
(4163)
(4164) ENG_TO:
(4165) <> == "S_0_2"
(4166) <lex gloss eng morname> == 'TO'
(4167) <lex gloss eng> == 'temporal overlap'.
(4168)
(4169) ENG_true:
(4170) <> == "R_ngalenga_1"
(4171) <lex gloss eng> == 'true'.
(4172)
(4173) ENG_TS:
(4174) <> == "S_du"
(4175) <lex gloss eng morname> == 'TS'
(4176) <lex gloss eng> == 'temporal succession'.
(4177)
(4178) ENG_turn_back_on:
(4179) <> == "R_moge"
(4180) <lex gloss eng> == 'turn.back.on'.
(4181)
(4182) ENG_untie:
(4183) <> == "R_fuutu"
(4184) <lex gloss eng> == 'untie'.
(4185)
(4186) ENG_V_not:
(4187) <> == "R_ada"
(4188) <lex gloss eng morname> == 'V.not'
(4189) <lex gloss eng> == 'not'.
(4190)
(4191) ENG_vAux_S1d:
(4192) <> == "S_yare_1_vAux"
(4193) <lex gloss eng morname> == 'vAux.S1d'
(4194) <lex gloss eng> == 'verbal auxiliary first dual subject'.
(4195)

(4196) ENG_vAux_S1p:
(4197) <> == "S_ya_1_vAux"
(4198) <lex gloss eng morname> == 'vAux.S1p'
(4199) <lex gloss eng> == 'verbal auxiliary first plural subject'.
(4200)
(4201) ENG_vAux_S1s:
(4202) <> == "S_ni_vAux"
(4203) <lex gloss eng morname> == 'vAux.S1s'
(4204) <lex gloss eng> == 'verbal auxiliary first singular subject'.
(4205)
(4206) ENG_vAux_S3d:
(4207) <> == "S_dere_vAux"
(4208) <lex gloss eng morname> == 'vAux.S3d'
(4209) <lex gloss eng> == 'verbal auxiliary third dual subject'.
(4210)
(4211) ENG_vAux_S3p:
(4212) <> == "S_de_vAux"
(4213) <lex gloss eng morname> == 'vAux.S3p'
(4214) <lex gloss eng> == 'verbal auxiliary third plural subject'.
(4215)
(4216) ENG_vAux_S3s:
(4217) <> == "S_na_2"
(4218) <lex gloss eng morname> == 'vAux.S3s'
(4219) <lex gloss eng> == 'verbal auxiliary third singular subject'.
(4220)
(4221) ENG_vAux_Tf_S2d:
(4222) <> == "S_nadere"
(4223) <lex gloss eng morname> == 'vAux.Tf.S2d'
(4224) <lex gloss eng> == 'verbal auxiliary second dual subject'.
(4225)
(4226) ENG_vAux_Tf_S2p:
(4227) <> == "S_nade"
(4228) <lex gloss eng morname> == 'vAux.Tf.S2p'
(4229) <lex gloss eng> == 'verbal auxiliary second plural subject'.
(4230)
(4231) ENG_vAux_Tf_S2s:
(4232) <> == "S_nahe"
(4233) <lex gloss eng morname> == 'vAux.Tf.S2s'
(4234) <lex gloss eng> == 'verbal auxiliary second singular subject,
(4235) future tense'.
(4236)
(4237) ENG_vAux_Tnf_S2d:
(4238) <> == "S_yare_2_vAux"
(4239) <lex gloss eng morname> == 'vAux.Tnf.S2d'
(4240) <lex gloss eng> == 'verbal auxiliary second dual subject,
(4241) non-future tense'.
(4242)
(4243) ENG_vAux_Tnf_S2p:
(4244) <> == "S_ya_2_vAux"
(4245) <lex gloss eng morname> == 'vAux.Tnf.S2p'
(4246) <lex gloss eng> == 'verbal auxiliary second plural subject,
(4247) non-future tense'.
(4248)
(4249) ENG_vAux_Tnf_S2s:
(4250) <> == "S_ne_vAux"
(4251) <lex gloss eng morname> == 'vAux.Tnf.S2s'
(4252) <lex gloss eng> == 'verbal auxiliary second singular subject,

```

(4253) non-future'.
(4254)
(4255) ENG_vAuxR:
(4256) <> == "R_bo"
(4257) <lex gloss eng morname> == 'vAuxR'
(4258) <lex gloss eng> == 'verbal auxiliary root'.
(4259)
(4260) ENG_very:
(4261) <> == "R_bai_1"
(4262) <lex gloss eng> == 'very'.
(4263)
(4264) ENG_village:
(4265) <> == "R_matane"
(4266) <lex gloss eng> == 'village'.
(4267)
(4268) ENG_walk:
(4269) <> == "R_golo"
(4270) <lex gloss eng> == 'walk'.
(4271)
(4272) ENG_watch:
(4273) <> == "R_fele"
(4274) <lex gloss eng> == 'watch'.
(4275)
(4276) ENG_watch_over:
(4277) <> == "R_oofa"
(4278) <lex gloss eng> == 'watch.over'.
(4279)
(4280) ENG_we:
(4281) <> == "R_ere"
(4282) <lex gloss eng> == 'we'.
(4283)
(4284) ENG_well:
(4285) <> == "S_de_1"
(4286) <lex gloss eng> == 'well'.
(4287)
(4288) ENG_what:
(4289) <> == "R_taate"
(4290) <lex gloss eng> == 'what'.
(4291)
(4292) ENG_wild:
(4293) <> == "R_kui_1"
(4294) <lex gloss eng> == 'wild'.
(4295)
(4296) ENG_with:
(4297) <> == "R_bagu"
(4298) <lex gloss eng> == 'with'.
(4299)
(4300) ENG_with_dual:
(4301) <> == "R_ngare"
(4302) <lex gloss eng> == 'with.dual'.
(4303)
(4304) ENG_without:
(4305) <> == "R_uwau"
(4306) <lex gloss eng> == 'without'.
(4307)
(4308) ENG_woman:
(4309) <> == "R_aita"

```

```

(4310) <lex gloss eng> == 'woman'.
(4311)
(4312) ENG_work:
(4313) <> == "R_haruwe"
(4314) <lex gloss eng> == 'work'.
(4315)
(4316) ENG_write:
(4317) <> == "R_yerege"
(4318) <lex gloss eng> == 'write'.
(4319)
(4320) ENG_yell_to/shout:
(4321) <> == "R_yaure"
(4322) <lex gloss eng> == 'yell.to/shout'.
(4323)
(4324) ENG_you_plural:
(4325) <> == "R_tere"
(4326) <lex gloss eng> == 'you.plural'.
(4327)
(4328) ENG_you_sing:
(4329) <> == "R_ne"
(4330) <lex gloss eng> == 'you.sing'.
(4331)
(4332) ENG_your_dual:
(4333) <> == "R_teire"
(4334) <lex gloss eng> == 'your.dual'.
(4335)
(4336) ENG_your_plural:
(4337) <> == "R_tebere"
(4338) <lex gloss eng> == 'your.plural'.
(4339)
(4340) ENG_your_sing:
(4341) <> == "R_name"
(4342) <lex gloss eng> == 'your.sing'.
(4343)
(4344) ENG_yourselves_dual:
(4345) <> == "R_tare"
(4346) <lex gloss eng> == 'yourselves.dual'.
(4347)
(4348) ENG_yourselves_plural:
(4349) <> == "R_tage"
(4350) <lex gloss eng> == 'yourselves.plural'.
(4351)
(4352)
(4353) %
(4354) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(4355)
(4356) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Tok Pisin %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(4357) %
(4358)
(4359)
(4360) TP_inap:
(4361) <> == "R_edo"
(4362) <lex gloss nat> == inap.
(4363)
(4364) TP_lek:
(4365) <> == "R_afe"
(4366) <lex gloss nat> == lek.

```

(4367)
(4368) TP_no:
(4369) <> == "R_ada"
(4370) <lex gloss nat> == no.
(4371)
(4372) TP_no_ken:
(4373) <> == "R_adai"
(4374) <lex gloss nat> == 'no ken'.
(4375)
(4376) TP_soim:
(4377) <> == "R_abiti"
(4378) <lex gloss nat> == soim.
(4379)
(4380)
(4381) %
(4382) %%%
(4383)
(4384) %%% Ogea %%%
(4385) %
(4386)
(4387) R_abiti:
(4388) <> == C_R_VR
(4389) <phon under> == abiti
(4390) <lex gloss eng> == ENG_show
(4391) <lex gloss nat> == TP_soim.
(4392)
(4393) R_ada:
(4394) <> == C_R_VNeg
(4395) <phon under> == ada
(4396) <lex gloss eng> == ENG_V_not
(4397) <lex gloss nat> == TP_no.
(4398)
(4399) R_adai:
(4400) <> == C_R_Adv
(4401) <phon under> == adai
(4402) <lex gloss eng> == ENG_do_not
(4403) <lex gloss nat> == TP_no_ken.
(4404)
(4405) R_afe:
(4406) <> == C_R_PNR
(4407) <phon under> == afe
(4408) <lex gloss eng> == ENG_leg/foot
(4409) <lex gloss nat> == TP_lek.
(4410)
(4411) R_afo:
(4412) <> == C_R_PNR
(4413) <phon under> == afo
(4414) <lex gloss eng> == ENG_leg/foot_3sP.
(4415)
(4416) R_agotete:
(4417) <> == C_R_VR
(4418) <phon under> == agotete
(4419) <lex gloss eng> == ENG_teach.
(4420)
(4421) R_aita:
(4422) <> == C_R_NW
(4423) <phon under> == aita

(4424) <lex gloss eng> == ENG_woman.
(4425)
(4426) R_anyakaro:
(4427) <> == "R_anyakaro_1" "R_anyakaro_2".
(4428)
(4429) R_anyakaro_1:
(4430) <> == C_R_Adj
(4431) <phon under> == anyakaro
(4432) <lex gloss eng> == ENG_large.
(4433)
(4434)
(4435) R_anyakaro_2:
(4436) <> == C_R_Adj_VR
(4437) <phon under> == anyakaro
(4438) <lex gloss eng> == ENG_become_large.
(4439)
(4440) R_ariri:
(4441) <> == C_R_VR
(4442) <phon under> == ariri
(4443) <lex gloss eng> == ENG_deflect.
(4444)
(4445) R_awa:
(4446) <> == C_R_VR_MC1
(4447) <phon under> == awa
(4448) <lex gloss eng> == ENG_allow/leave.
(4449)
(4450) R_aya:
(4451) <> == C_R_VR
(4452) <phon under> == aya
(4453) <lex gloss eng> == ENG_throw.
(4454)
(4455) R_babo:
(4456) <> == C_R_NW
(4457) <phon under> == babo
(4458) <lex gloss eng> == ENG_sibling_same_sex_referential
(4459) <lex under> == bai_2.
(4460)
(4461) R_bagu:
(4462) <> == C_R_Prep
(4463) <phon under> == bagu
(4464) <lex gloss eng> == ENG_with
(4465) <allo mec> == "MEC_03".
(4466)
(4467) R_bagumu:
(4468) <> == C_R_VR
(4469) <phon under> == bagumu
(4470) <lex gloss eng> == ENG_connect/join
(4471) <lex under> == bagu-mu.
(4472)
(4473) R_bai:
(4474) <> == "R_bai_1" "R_bai_2".
(4475)
(4476) R_bai_1:
(4477) <> == C_R_Adj
(4478) <phon under> == bai
(4479) <lex gloss eng> == ENG_very.
(4480)

(4481) R_bai_2:
 (4482) <> == C_R_PNR
 (4483) <phon under> == bai
 (4484) <lex gloss eng> == ENG_sibling_same_sex_possessive
 (4485) <lex under> == bai_2
 (4486) <allo mec> == "MEC_02".
 (4487)
 (4488) R_baingaro:
 (4489) <> == "R_baingaro_1" "R_baingaro_2".
 (4490)
 (4491) R_baingaro_1:
 (4492) <> == C_R_Adj
 (4493) <phon under> == baingaro
 (4494) <lex gloss eng> == ENG_many.
 (4495)
 (4496) R_baingaro_2:
 (4497) <> == C_R_Adj_VR
 (4498) <phon under> == baingaro
 (4499) <lex gloss eng> == ENG_become_many.
 (4500)
 (4501) R_be:
 (4502) <> == C_R_Deictic
 (4503) <phon under> == be
 (4504) <lex gloss eng> == ENG_this.
 (4505)
 (4506) R_beele:
 (4507) <> == C_R_NW
 (4508) <phon under> == beele
 (4509) <lex gloss eng> == ENG_talk/word.
 (4510)
 (4511) R_bemu:
 (4512) <> == C_R_NW
 (4513) <phon under> == bemu
 (4514) <lex gloss eng> == ENG_sibling_same_sex_his/her/its
 (4515) <lex under> == bai_2.
 (4516)
 (4517) R_bo:
 (4518) <> == C_R_vAuxR
 (4519) <phon under> == bo
 (4520) <lex gloss eng> == ENG_vAuxR.
 (4521)
 (4522) R_bolo:
 (4523) <> == C_R_VR
 (4524) <phon under> == bolo
 (4525) <lex gloss eng> == ENG_fill.
 (4526)
 (4527) R_buuwa:
 (4528) <> == C_R_NW
 (4529) <phon under> == buuwa
 (4530) <lex gloss eng> == ENG_pig.
 (4531)
 (4532) R_dada:
 (4533) <> == C_R_NW
 (4534) <phon under> == dada
 (4535) <lex gloss eng> == ENG_thing.
 (4536)
 (4537) R_dage:

(4538) <> == C_R_VR_MC1
 (4539) <phon under> == dage
 (4540) <lex gloss eng> == ENG_bound.
 (4541)
 (4542) R_ebere:
 (4543) <> == C_R_PosPro
 (4544) <phon under> == ebere
 (4545) <lex gloss eng> == ENG_our_plural.
 (4546)
 (4547) R_edo:
 (4548) <> == C_R_Adv
 (4549) <phon under> == edo
 (4550) <lex gloss eng> == ENG_able
 (4551) <lex gloss nat> == TP_inap.
 (4552)
 (4553) R_elege:
 (4554) <> == C_R_VR_MC1
 (4555) <phon under> == elege
 (4556) <lex gloss eng> == ENG_gather.
 (4557)
 (4558) R_ere:
 (4559) <> == C_R_PerPro
 (4560) <phon under> == ere
 (4561) <lex gloss eng> == ENG_we.
 (4562)
 (4563) R_ete:
 (4564) <> == C_R_Adj
 (4565) <phon under> == ete
 (4566) <lex gloss eng> == ENG_one/a.
 (4567)
 (4568) R_ewe:
 (4569) <> == C_R_Adj
 (4570) <phon under> == ewe
 (4571) <lex gloss eng> == ENG_idle.
 (4572)
 (4573) R_fada:
 (4574) <> == C_R_VR
 (4575) <phon under> == fada
 (4576) <lex gloss eng> == ENG_stick_to.
 (4577)
 (4578) R_faga:
 (4579) <> == C_R_VR
 (4580) <phon under> == faga
 (4581) <lex gloss eng> == ENG_run_away.
 (4582)
 (4583) R_fai:
 (4584) <> == C_R_NW
 (4585) <phon under> == fai
 (4586) <lex gloss eng> == ENG_man.
 (4587)
 (4588) R_fasa:
 (4589) <> == C_R_VR
 (4590) <phon under> == fasa
 (4591) <lex gloss eng> == ENG_quickly.
 (4592)
 (4593) R_fele:
 (4594) <> == C_R_VR

(4595) <phon under> == fele
 (4596) <lex gloss eng> == ENG_watch.
 (4597)
 (4598) R_fere:
 (4599) <> == C_R_Adj
 (4600) <phon under> == fere
 (4601) <lex gloss eng> == ENG_different.
 (4602)
 (4603) R_feu:
 (4604) <> == C_R_VR
 (4605) <phon under> == feu
 (4606) <lex gloss eng> == ENG_pour_out.
 (4607)
 (4608) R_fii:
 (4609) <> == C_R_VR
 (4610) <phon under> == fii
 (4611) <lex gloss eng> == ENG_smooth.
 (4612)
 (4613) R_fililai:
 (4614) <> == C_R_VR
 (4615) <phon under> == fililai
 (4616) <lex gloss eng> == ENG_fly.
 (4617)
 (4618) R_fuutu:
 (4619) <> == C_R_VR
 (4620) <phon under> == fuutu
 (4621) <lex gloss eng> == ENG_untie.
 (4622)
 (4623) R_gaigai:
 (4624) <> == C_R_Adv
 (4625) <phon under> == gaigai
 (4626) <lex gloss eng> == ENG_always.
 (4627)
 (4628) R_gaigu:
 (4629) <> == C_R_NW
 (4630) <phon under> == gaigu
 (4631) <lex gloss eng> == ENG_thigh.
 (4632)
 (4633) R_gau:
 (4634) <> == C_R_NW
 (4635) <phon under> == gau
 (4636) <lex gloss eng> == ENG_desire.
 (4637)
 (4638) R_giiri:
 (4639) <> == "R_giiri_1" "R_giiri_2".
 (4640)
 (4641) R_giiri_1:
 (4642) <> == C_R_Adj
 (4643) <phon under> == giiri
 (4644) <lex gloss eng> == ENG_red.
 (4645)
 (4646) R_giiri_2:
 (4647) <> == C_R_Adj_VR
 (4648) <phon under> == giiri
 (4649) <lex gloss eng> == ENG_become_red.
 (4650)
 (4651) R_goi:

```

(4652)      <> == C_R_VR
(4653)      <phon under> == goi
(4654)      <lex gloss eng> == ENG_go
(4655)      <lex form> == DEFAULT GLIDE.
(4656)
(4657)      R_golo:
(4658)      <> == C_R_VR
(4659)      <phon under> == golo
(4660)      <lex gloss eng> == ENG_walk.
(4661)
(4662)      R_ha:
(4663)      <> == C_R_NW
(4664)      <phon under> == ha
(4665)      <lex gloss eng> == ENG_place
(4666)      <lex form> == DEFAULT ALLO2
(4667)      <allo form allo2> == haumu.
(4668)
(4669)      R_haawe:
(4670)      <> == C_R_VR
(4671)      <phon under> == haawe
(4672)      <lex gloss eng> == ENG_judge.
(4673)
(4674)      R_hage:
(4675)      <> == C_R_RflPro
(4676)      <phon under> == hage
(4677)      <lex gloss eng> == ENG_ourselves_plural.
(4678)
(4679)      R_hai:
(4680)      <> == C_R_VR
(4681)      <phon under> == hai
(4682)      <lex gloss eng> == ENG_tell.
(4683)
(4684)      R_haife:
(4685)      <> == C_R_VR_MC1
(4686)      <phon under> == haife
(4687)      <lex gloss eng> == ENG_get_01s.
(4688)
(4689)      R_haige:
(4690)      <> == C_R_VR_MC1
(4691)      <phon under> == haige
(4692)      <lex gloss eng> == ENG_get_01p.
(4693)
(4694)      R_haine:
(4695)      <> == C_R_VR_MC1
(4696)      <phon under> == haine
(4697)      <lex gloss eng> == ENG_get_02s.
(4698)
(4699)      R_haire:
(4700)      <> == C_R_VR_MC1
(4701)      <phon under> == haire
(4702)      <lex gloss eng> == ENG_get_01d.
(4703)
(4704)      R_haiya:
(4705)      <> == C_R_VR
(4706)      <phon under> == haiya
(4707)      <lex gloss eng> == ENG_prepare.
(4708)

```

(4709) R_hanige:
(4710) <> == C_R_VR_MC1
(4711) <phon under> == hanige
(4712) <lex gloss eng> == ENG_get_O3p.
(4713)
(4714) R_hanire:
(4715) <> == C_R_VR_MC1
(4716) <phon under> == hanire
(4717) <lex gloss eng> == ENG_get_O3d.
(4718)
(4719) R_hare:
(4720) <> == C_R_RflPro
(4721) <phon under> == hare
(4722) <lex gloss eng> == ENG_ourselves_dual.
(4723)
(4724) R_harige:
(4725) <> == C_R_VR_MC1
(4726) <phon under> == harige
(4727) <lex gloss eng> == ENG_hit_O1p.
(4728)
(4729) R_harire:
(4730) <> == C_R_VR_MC1
(4731) <phon under> == harire
(4732) <lex gloss eng> == ENG_hit_O1d.
(4733)
(4734) R_haruwe:
(4735) <> == C_R_NW
(4736) <phon under> == haruwe
(4737) <lex gloss eng> == ENG_work.
(4738)
(4739) R_haruwe_2:
(4740) <> == C_R_VR
(4741) <phon under> == haruwe
(4742) <lex gloss eng> == ENG_do_work
(4743) <lex under> == haruwe_2
(4744) <allo mec> == "MEC_01".
(4745)
(4746) R_hatige:
(4747) <> == C_R_VR_MC1
(4748) <phon under> == hatige
(4749) <lex gloss eng> == ENG_get_O2p.
(4750)
(4751) R_hatire:
(4752) <> == C_R_VR_MC1
(4753) <phon under> == hatire
(4754) <lex gloss eng> == ENG_get_O2d.
(4755)
(4756) R_hau:
(4757) <> == C_R_VR
(4758) <phon under> == hau
(4759) <lex gloss eng> == ENG_get_O3s.
(4760)
(4761) R_heige:
(4762) <> == C_R_VR_MC1
(4763) <phon under> == heige
(4764) <lex gloss eng> == ENG_appear/arrive.
(4765)

(4766) R_heire:
 (4767) <> == C_R_PosPro
 (4768) <phon under> == heire
 (4769) <lex gloss eng> == ENG_our_dual.
 (4770)
 (4771) R_herige:
 (4772) <> == C_R_VR_MC1
 (4773) <phon under> == herige
 (4774) <lex gloss eng> == ENG_see_O1p.
 (4775)
 (4776) R_herire:
 (4777) <> == C_R_VR_MC1
 (4778) <phon under> == herire
 (4779) <lex gloss eng> == ENG_see_O1d.
 (4780)
 (4781) R_higi:
 (4782) <> == C_R_VR_MC1
 (4783) <phon under> == higi
 (4784) <lex gloss eng> == ENG_give_O1p.
 (4785)
 (4786) R_hii:
 (4787) <> == C_R_VR
 (4788) <phon under> == hii
 (4789) <lex gloss eng> == ENG_give_O1s
 (4790) <lex form> == DEFAULT GLIDE.
 (4791)
 (4792) R_hiiri:
 (4793) <> == C_R_NW
 (4794) <phon under> == hiiri
 (4795) <lex gloss eng> == ENG_ocean.
 (4796)
 (4797) R_hilou:
 (4798) <> == C_R_Adj
 (4799) <phon under> == hilou
 (4800) <lex gloss eng> == ENG_good
 (4801) <lex form> == DEFAULT ALLO2
 (4802) <allo form allo2> == hiloo.
 (4803)
 (4804) R_hilou_2:
 (4805) <> == C_R_Adj_VR
 (4806) <phon under> == hilou
 (4807) <lex gloss eng> == ENG_become_good
 (4808) <lex form> == DEFAULT ALLO2
 (4809) <allo form allo2> == hiloo
 (4810) <allo mec allo2> == "MEC_01".
 (4811)
 (4812) R_hini:
 (4813) <> == C_R_VR_MC1
 (4814) <phon under> == hini
 (4815) <lex gloss eng> == ENG_give_O2s.
 (4816)
 (4817) R_hiri:
 (4818) <> == C_R_VR_MC1
 (4819) <phon under> == hiri
 (4820) <lex gloss eng> == ENG_give_O1d.
 (4821)
 (4822) R_hogo:

(4823) <> == C_R_PNR
 (4824) <phon under> == hogo
 (4825) <lex gloss eng> == ENG_face.
 (4826)
 (4827) R_hoyo:
 (4828) <> == C_R_VR
 (4829) <phon under> == hoyo
 (4830) <lex gloss eng> == ENG_help.
 (4831)
 (4832) R_huuru:
 (4833) <> == C_R_VR
 (4834) <phon under> == huuru
 (4835) <lex gloss eng> == ENG_send.
 (4836)
 (4837) R_huwanya:
 (4838) <> == C_R_PNR
 (4839) <phon under> == huwanya
 (4840) <lex gloss eng> == ENG_inside.
 (4841)
 (4842) R_ila:
 (4843) <> == C_R_Adj
 (4844) <phon under> == ila
 (4845) <lex gloss eng> == ENG_some.
 (4846)
 (4847) R_ilei:
 (4848) <> == C_R_VR
 (4849) <phon under> == ilei
 (4850) <lex gloss eng> == ENG_press.
 (4851)
 (4852) R_inyaba:
 (4853) <> == "R_inyaba_1" "R_inyaba_2".
 (4854)
 (4855) R_inyaba_1:
 (4856) <> == C_R_Adj
 (4857) <phon under> == inyaba
 (4858) <lex gloss eng> == ENG_bad.
 (4859)
 (4860) R_inyaba_2:
 (4861) <> == C_R_Adj_VR
 (4862) <phon under> == inyaba
 (4863) <lex gloss eng> == ENG_become_bad.
 (4864)
 (4865) R_inyi:
 (4866) <> == C_R_VR_MC1
 (4867) <phon under> == inyi
 (4868) <lex gloss eng> == ENG_be.
 (4869)
 (4870) R_isi:
 (4871) <> == C_R_VR_MC1
 (4872) <phon under> == isi
 (4873) <lex gloss eng> == ENG_hear/understand/obey.
 (4874)
 (4875) R_isoki:
 (4876) <> == C_R_VR
 (4877) <phon under> == isoki
 (4878) <lex gloss eng> == ENG_ask.
 (4879)

(4880) R_jai:
(4881) <> == C_R_VR
(4882) <phon under> == jai
(4883) <lex gloss eng> == ENG_get_up
(4884) <lex form> == DEFAULT GLIDE.
(4885)
(4886) R_ji:
(4887) <> == C_R_PerPro
(4888) <phon under> == ji
(4889) <lex gloss eng> == ENG_I.
(4890)
(4891) R_jiri:
(4892) <> == C_R_VR
(4893) <phon under> == jiri
(4894) <lex gloss eng> == ENG_sleep_pl.
(4895)
(4896) R_kaafa:
(4897) <> == C_R_NW
(4898) <phon under> == kaafa
(4899) <lex gloss eng> == ENG_animal.
(4900)
(4901) R_karu:
(4902) <> == C_R_VR_R
(4903) <phon under> == karu
(4904) <lex gloss eng> == ENG_cut.
(4905)
(4906) R_kati:
(4907) <> == C_R_VR
(4908) <phon under> == kati
(4909) <lex gloss eng> == ENG_encircle.
(4910)
(4911) R_kesebu:
(4912) <> == C_R_NW
(4913) <phon under> == kesebu
(4914) <lex gloss eng> == ENG_knife.
(4915)
(4916) R_kobibi:
(4917) <> == "R_kobibi_1" "R_kobibi_2".
(4918)
(4919) R_kobibi_1:
(4920) <> == C_R_Adj
(4921) <phon under> == kobibi
(4922) <lex gloss eng> == ENG_crooked.
(4923)
(4924) R_kobibi_2:
(4925) <> == C_R_Adj_VR
(4926) <phon under> == kobibi
(4927) <lex gloss eng> == ENG_become_crooked.
(4928)
(4929) R_korai:
(4930) <> == C_R_VR
(4931) <phon under> == korai
(4932) <lex gloss eng> == ENG_tear.
(4933)
(4934) R_kou:
(4935) <> == C_R_VR_R
(4936) <phon under> == kou

(4937) <lex gloss eng> == ENG_break_in_two.
(4938)
(4939) R_kui:
(4940) <> == "R_kui_1" "R_kui_2".
(4941)
(4942) R_kui_1:
(4943) <> == C_R_Adj
(4944) <phon under> == kui
(4945) <lex gloss eng> == ENG_wild.
(4946)
(4947) R_kui_2:
(4948) <> == C_R_Adj_VR
(4949) <phon under> == kui
(4950) <lex gloss eng> == ENG_become_wild.
(4951)
(4952) R_kuru:
(4953) <> == C_R_VR
(4954) <phon under> == kuru
(4955) <lex gloss eng> == ENG_look_for.
(4956)
(4957) R_la:
(4958) <> == C_R_Prep
(4959) <phon under> == la
(4960) <lex gloss eng> == ENG_at.
(4961)
(4962) R_lala:
(4963) <> == "R_lala_1" "R_lala_2".
(4964)
(4965) R_lala_1:
(4966) <> == C_R_NW
(4967) <phon under> == lala
(4968) <lex gloss eng> == ENG_light
(4969) <lex under> == lala_1.
(4970)
(4971) R_lala_2:
(4972) <> == C_R_VR
(4973) <phon under> == lala
(4974) <lex gloss eng> == ENG_shine
(4975) <lex under> == lala_2.
(4976)
(4977) R_le:
(4978) <> == C_R_VR
(4979) <phon under> == le
(4980) <lex gloss eng> == ENG_speak.
(4981)
(4982) R_mai:
(4983) <> == C_R_VR
(4984) <phon under> == mai
(4985) <lex gloss eng> == ENG_come_up
(4986) <lex form> == DEFAULT_GLIDE.
(4987)
(4988) R_matane:
(4989) <> == C_R_NW
(4990) <phon under> == matane
(4991) <lex gloss eng> == ENG_village.
(4992)
(4993) R_me:

(4994) <> == C_R_VR_MC1
(4995) <phon under> == me
(4996) <lex gloss eng> == ENG_come_down.
(4997)
(4998) R_megebu:
(4999) <> == C_R_NW
(5000) <phon under> == megebu
(5001) <lex gloss eng> == ENG_stone.
(5002)
(5003) R_menekele:
(5004) <> == "R_menekele_1" "R_menekele_2".
(5005)
(5006) R_menekele_1:
(5007) <> == C_R_Adj
(5008) <phon under> == menekele
(5009) <lex gloss eng> == ENG_small
(5010) <lex form> == DEFAULT_ALLO2
(5011) <allo form allo2> == melekene.
(5012)
(5013) R_menekele_2:
(5014) <> == C_R_Adj_VR
(5015) <phon under> == menekele
(5016) <lex gloss eng> == ENG_become_small
(5017) <lex form> == DEFAULT_ALLO2
(5018) <allo form allo2> == melekene
(5019) <allo mec allo2> == "MEC_01".
(5020)
(5021) R_mini:
(5022) <> == C_R_VR_MC1
(5023) <phon under> == mini
(5024) <lex gloss eng> == ENG_go_down.
(5025)
(5026) R_moge:
(5027) <> == C_R_VR
(5028) <phon under> == moge
(5029) <lex gloss eng> == ENG_turn_back_on.
(5030)
(5031) R_mutu:
(5032) <> == C_R_PNR
(5033) <phon under> == mutu
(5034) <lex gloss eng> == ENG_nose.
(5035)
(5036) R_nage:
(5037) <> == C_R_RflPro
(5038) <phon under> == nage
(5039) <lex gloss eng> == ENG_theirselves_plural.
(5040)
(5041) R_name:
(5042) <> == C_R_PosPro
(5043) <phon under> == name
(5044) <lex gloss eng> == ENG_your_sing.
(5045)
(5046) R_nare:
(5047) <> == C_R_RflPro
(5048) <phon under> == nare
(5049) <lex gloss eng> == ENG_theirselves_dual.
(5050)

(5051) R_nari:
 (5052) <> == C_R_VR_MC1
 (5053) <phon under> == nari
 (5054) <lex gloss eng> == ENG_hit_O2s.
 (5055)
 (5056) R_narige:
 (5057) <> == C_R_VR_MC1
 (5058) <phon under> == narige
 (5059) <lex gloss eng> == ENG_hit_O3p.
 (5060)
 (5061) R_narire:
 (5062) <> == C_R_VR_MC1
 (5063) <phon under> == narire
 (5064) <lex gloss eng> == ENG_hit_O3d.
 (5065)
 (5066) R_ne:
 (5067) <> == C_R_PerPro
 (5068) <phon under> == ne
 (5069) <lex gloss eng> == ENG_you_sing.
 (5070)
 (5071) R_nebere:
 (5072) <> == C_R_PosPro
 (5073) <phon under> == nebere
 (5074) <lex gloss eng> == ENG_their_plural.
 (5075)
 (5076) R_neire:
 (5077) <> == C_R_PosPro
 (5078) <phon under> == neire
 (5079) <lex gloss eng> == ENG_their_dual.
 (5080)
 (5081) R_nere:
 (5082) <> == C_R_PerPro
 (5083) <phon under> == nere
 (5084) <lex gloss eng> == ENG_they.
 (5085)
 (5086) R_nerige:
 (5087) <> == C_R_VR_MC1
 (5088) <phon under> == nerige
 (5089) <lex gloss eng> == ENG_see_O3p.
 (5090)
 (5091) R_nerire:
 (5092) <> == C_R_VR_MC1
 (5093) <phon under> == nerire
 (5094) <lex gloss eng> == ENG_see_O3d.
 (5095)
 (5096) R_neriye:
 (5097) <> == C_R_VR_MC1
 (5098) <phon under> == neriye
 (5099) <lex gloss eng> == ENG_see_O2s.
 (5100)
 (5101) R_ngalenga:
 (5102) <> == "R_ngalenga_1" "R_ngalenga_2".
 (5103)
 (5104) R_ngalenga_1:
 (5105) <> == C_R_Adj
 (5106) <phon under> == ngalenga
 (5107) <lex gloss eng> == ENG_true.

(5108)
 (5109) R_ngalenga_2:
 (5110) <> == C_R_Adj_VR
 (5111) <phon under> == ngalenga
 (5112) <lex gloss eng> == ENG_become_true.
 (5113)
 (5114) R_ngare:
 (5115) <> == C_R_Prep
 (5116) <phon under> == ngare
 (5117) <lex gloss eng> == ENG_with_dual.
 (5118)
 (5119) R_nigi:
 (5120) <> == C_R_VR_MC1
 (5121) <phon under> == nigi
 (5122) <lex gloss eng> == ENG_give_O3p.
 (5123)
 (5124) R_nina:
 (5125) <> == C_R_PNR
 (5126) <phon under> == nina
 (5127) <lex gloss eng> == ENG_thought.
 (5128)
 (5129) R_niri:
 (5130) <> == C_R_VR_MC1
 (5131) <phon under> == niri
 (5132) <lex gloss eng> == ENG_give_O3d.
 (5133)
 (5134) R_no:
 (5135) <> == C_R_PerPro
 (5136) <phon under> == no
 (5137) <lex gloss eng> == ENG_he/she/it.
 (5138)
 (5139) R_nogo:
 (5140) <> == C_R_RflPro
 (5141) <phon under> == nogo
 (5142) <lex gloss eng> == ENG_himself/herself/itself.
 (5143)
 (5144) R_nomo:
 (5145) <> == C_R_PosPro
 (5146) <phon under> == nomo
 (5147) <lex gloss eng> == ENG_his/her/its.
 (5148)
 (5149) R_nuu:
 (5150) <> == C_R_VR
 (5151) <phon under> == nuu
 (5152) <lex gloss eng> == ENG_shove.
 (5153)
 (5154) R_nyi:
 (5155) <> == C_R_VR_MC1
 (5156) <phon under> == nyi
 (5157) <lex gloss eng> == ENG_consume.
 (5158)
 (5159) R_ogola:
 (5160) <> == C_R_NW
 (5161) <phon under> == ogola
 (5162) <lex gloss eng> == ENG_cloth.
 (5163)
 (5164) R_ono:

(5165) <> == C_R_VR
 (5166) <phon under> == ono
 (5167) <lex gloss eng> == ENG_do_thus.
 (5168)
 (5169) R_onou:
 (5170) <> == C_R_Adj
 (5171) <phon under> == onou
 (5172) <lex gloss eng> == ENG_like_that.
 (5173)
 (5174) R_oofa:
 (5175) <> == C_R_VR
 (5176) <phon under> == oofa
 (5177) <lex gloss eng> == ENG_watch_over.
 (5178)
 (5179) R_oarjo:
 (5180) <> == C_R_VR
 (5181) <phon under> == oarjo
 (5182) <lex gloss eng> == ENG_follow.
 (5183)
 (5184) R_oto:
 (5185) <> == C_R_VR
 (5186) <phon under> == oto
 (5187) <lex gloss eng> == ENG_stand.
 (5188)
 (5189) R_ou:
 (5190) <> == "R_ou_1" "R_ou_2".
 (5191)
 (5192) R_ou_1:
 (5193) <> == C_R_Adj
 (5194) <phon under> == ou
 (5195) <lex gloss eng> == ENG_heavy.
 (5196)
 (5197) R_ou_2:
 (5198) <> == C_R_Adj_VR
 (5199) <phon under> == ou
 (5200) <lex gloss eng> == ENG_become_heavy.
 (5201)
 (5202) R_owo:
 (5203) <> == C_R_PNR
 (5204) <phon under> == owo
 (5205) <lex gloss eng> == ENG_hand.
 (5206)
 (5207) R_soo:
 (5208) <> == C_R_VR
 (5209) <phon under> == soo
 (5210) <lex gloss eng> == ENG_chase.
 (5211)
 (5212) R_taate:
 (5213) <> == C_R_IP
 (5214) <phon under> == taate
 (5215) <lex gloss eng> == ENG_what.
 (5216)
 (5217) R_tafa:
 (5218) <> == C_R_VR_MC1
 (5219) <phon under> == tafa
 (5220) <lex gloss eng> == ENG_put.
 (5221)

(5222) R_tage:
 (5223) <> == C_R_RflPro
 (5224) <phon under> == tage
 (5225) <lex gloss eng> == ENG_yourselves_plural.
 (5226)
 (5227) R_tare:
 (5228) <> == C_R_RflPro
 (5229) <phon under> == tare
 (5230) <lex gloss eng> == ENG_yourselves_dual.
 (5231)
 (5232) R_tarige:
 (5233) <> == C_R_VR_MC1
 (5234) <phon under> == tarige
 (5235) <lex gloss eng> == ENG_hit_O2p.
 (5236)
 (5237) R_tarire:
 (5238) <> == C_R_VR_MC1
 (5239) <phon under> == tarire
 (5240) <lex gloss eng> == ENG_hit_O2d.
 (5241)
 (5242) R_tau:
 (5243) <> == C_R_VR_R
 (5244) <phon under> == tau
 (5245) <lex gloss eng> == ENG_plant.
 (5246)
 (5247) R_te:
 (5248) <> == C_R_VR_MC1
 (5249) <phon under> == te
 (5250) <lex gloss eng> == ENG_get/take.
 (5251)
 (5252) R_tebere:
 (5253) <> == C_R_PosPro
 (5254) <phon under> == tebere
 (5255) <lex gloss eng> == ENG_your_plural.
 (5256)
 (5257) R_teire:
 (5258) <> == C_R_PosPro
 (5259) <phon under> == teire
 (5260) <lex gloss eng> == ENG_your_dual.
 (5261)
 (5262) R_tere:
 (5263) <> == C_R_PerPro
 (5264) <phon under> == tere
 (5265) <lex gloss eng> == ENG_you_plural.
 (5266)
 (5267) R_terige:
 (5268) <> == C_R_VR_MC1
 (5269) <phon under> == terige
 (5270) <lex gloss eng> == ENG_see_O2p.
 (5271)
 (5272) R_terire:
 (5273) <> == C_R_VR_MC1
 (5274) <phon under> == terire
 (5275) <lex gloss eng> == ENG_see_O2d.
 (5276)
 (5277) R_tigi:
 (5278) <> == C_R_VR_MC1

(5279) <phon under> == tigi
 (5280) <lex gloss eng> == ENG_give_O2p.
 (5281)
 (5282) R_tigini:
 (5283) <> == C_R_Adj
 (5284) <phon under> == tigini
 (5285) <lex gloss eng> == ENG_straight.
 (5286)
 (5287) R_tigini_2:
 (5288) <> == C_R_Adj_VR
 (5289) <phon under> == tigini
 (5290) <lex gloss eng> == ENG_become_straight.
 (5291)
 (5292) R_tiri:
 (5293) <> == C_R_VR_MC1
 (5294) <phon under> == tiri
 (5295) <lex gloss eng> == ENG_give_O2d.
 (5296)
 (5297) R_tu:
 (5298) <> == C_R_VR
 (5299) <phon under> == tu
 (5300) <lex gloss eng> == ENG_give_O3s.
 (5301)
 (5302) R_tuN:
 (5303) <> == C_R_VR
 (5304) <phon under> == tuN
 (5305) <lex gloss eng> == ENG_poke
 (5306) <lex form> == NASAL.
 (5307)
 (5308) R_tutu:
 (5309) <> == "R_tutu_1" "R_tutu_2".
 (5310)
 (5311) R_tutu_1:
 (5312) <> == C_R_Adj
 (5313) <phon under> == tutu
 (5314) <lex gloss eng> == ENG_short.
 (5315)
 (5316) R_tutu_2:
 (5317) <> == C_R_Adj_VR
 (5318) <phon under> == tutu
 (5319) <lex gloss eng> == ENG_become_short.
 (5320)
 (5321) R_ume:
 (5322) <> == C_R_VR_MC1
 (5323) <phon under> == ume
 (5324) <lex gloss eng> == ENG_die.
 (5325)
 (5326) R_umuge:
 (5327) <> == C_R_VR_MC1
 (5328) <phon under> == umuge
 (5329) <lex gloss eng> == ENG_fear.
 (5330)
 (5331) R_ure:
 (5332) <> == C_R_VR_MC1
 (5333) <phon under> == ure
 (5334) <lex gloss eng> == ENG_see_O3s.
 (5335)

(5336) R_usu:
 (5337) <> == C_R_VR
 (5338) <phon under> == usu
 (5339) <lex gloss eng> == ENG_finish.
 (5340)
 (5341) R_uulu:
 (5342) <> == C_R_VR
 (5343) <phon under> == uulu
 (5344) <lex gloss eng> == ENG_remove.
 (5345)
 (5346) R_uuwa:
 (5347) <> == "R_uuwa_1" "R_uuwa_2".
 (5348)
 (5349) R_uuwa_1:
 (5350) <> == C_R_NomNeg
 (5351) <phon under> == uuwa
 (5352) <lex gloss eng> == ENG_N_not
 (5353) <lex under> == uuwa_1.
 (5354)
 (5355) R_uuwa_2:
 (5356) <> == C_R_Adj_VR
 (5357) <phon under> == uuwa
 (5358) <lex gloss eng> == ENG_become_nothing.
 (5359)
 (5360) R_uwau:
 (5361) <> == C_R_Prep
 (5362) <phon under> == uwau
 (5363) <lex gloss eng> == ENG_without.
 (5364)
 (5365) R_wa:
 (5366) <> == C_R_Deictic
 (5367) <phon under> == wa
 (5368) <lex gloss eng> == ENG_that.
 (5369)
 (5370) R_wara:
 (5371) <> == C_R_VR_MC1
 (5372) <phon under> == wara
 (5373) <lex gloss eng> == ENG_fight.
 (5374)
 (5375) R_wari:
 (5376) <> == C_R_VR_MC1
 (5377) <phon under> == wari
 (5378) <lex gloss eng> == ENG_hit_03s.
 (5379)
 (5380) R_wewe:
 (5381) <> == C_R_NW
 (5382) <phon under> == wewe
 (5383) <lex gloss eng> == ENG_dream.
 (5384)
 (5385) R_yaawa:
 (5386) <> == C_R_VR
 (5387) <phon under> == yaawa
 (5388) <lex gloss eng> == ENG_lie.
 (5389)
 (5390) R_yafa:
 (5391) <> == C_R_VR_MC1
 (5392) <phon under> == yafa

```

(5393) <lex gloss eng> == ENG_sit.
(5394)
(5395) R_yage:
(5396) <> == C_R_RflPro
(5397) <phon under> == yage
(5398) <lex gloss eng> == ENG_myself.
(5399)
(5400) R_yame:
(5401) <> == C_R_PosPro
(5402) <phon under> == yame
(5403) <lex gloss eng> == ENG_my.
(5404)
(5405) R_yari:
(5406) <> == C_R_VR_MC1
(5407) <phon under> == yari
(5408) <lex gloss eng> == ENG_hit_O1s.
(5409)
(5410) R_yaure:
(5411) <> == C_R_VR
(5412) <phon under> == yaure
(5413) <lex gloss eng> == ENG_yell_to/shout.
(5414)
(5415) R_yerege:
(5416) <> == C_R_VR_MC1
(5417) <phon under> == yerege
(5418) <lex gloss eng> == ENG_write.
(5419)
(5420) R_yeriye:
(5421) <> == C_R_VR_MC1
(5422) <phon under> == yeriye
(5423) <lex gloss eng> == ENG_see_O1s.
(5424)
(5425) S_0:
(5426) <> == "S_0_Ind_S1d"
(5427) "S_0_Ind_S1p"
(5428) "S_0_Pro_Poss"
(5429) "S_0_1_Ind_S3s"
(5430) "S_0_2_Ind_S3s"
(5431) "S_0_Ind_Tip"
(5432) "S_0_Ind_Tp"
(5433) "S_0_1"
(5434) "S_0_2".
(5435)
(5436) S_0_1:
(5437) <> == C_S_DIFF_REF_TEMPORALITY
(5438) <phon under> == 0
(5439) <lex gloss eng> == ENG_SR_TS.
(5440)
(5441) S_0_1_Ind_S3s:
(5442) <> == C_S_IND_SUBJ
(5443) <phon under> == 0
(5444) <allo sec> == "SEC_11"
(5445) <allo mec> == "MEC_11" "MEC_13".
(5446)
(5447) S_0_2:
(5448) <> == C_S_SAME_REF_TEMPORALITY
(5449) <phon under> == 0

```



```

(5450) <lex gloss eng> == ENG_TO.
(5451)
(5452) S_0_2_Ind_S3s:
(5453) <> == C_S_IND_SUBJ
(5454) <phon under> == 0
(5455) <allo sec> == "SEC_10"
(5456) <allo mec> == "MEC_07".
(5457)
(5458) S_0_Ind_S1d:
(5459) <> == C_S_IND_SUBJ
(5460) <phon under> == 0
(5461) <allo sec> == "SEC_08"
(5462) <allo mec> == "MEC_06".
(5463)
(5464) S_0_Ind_S1p:
(5465) <> == C_S_IND_SUBJ
(5466) <phon under> == 0
(5467) <allo sec> == "SEC_08"
(5468) <allo mec> == "MEC_06".
(5469)
(5470) S_0_Ind_Tip:
(5471) <> == C_S_IND_TENSE
(5472) <lex form> == DEFAULT
(5473) <allo mec default> == "MEC_16" "MEC_17"
(5474) <allo sec> ==
(5475) <phon under> == 0.
(5476)
(5477) S_0_Ind_Tp:
(5478) <> == C_S_IND_TENSE
(5479) <lex form> == DEFAULT
(5480) <allo mec default> == "MEC_22"
(5481) <allo sec> ==
(5482) <phon under> == 0.
(5483)
(5484) S_0_Pro_Poss:
(5485) <> == C_S_PRONOUN_POSS
(5486) <lex form> == DEFAULT ALLO2
(5487) <phon under> == 0
(5488) <allo form allo2> == u
(5489) <lex gloss eng> == ENG_P3s.
(5490)
(5491) S_a:
(5492) <> == "S_a_1"
(5493) "S_a_2"
(5494) "S_a_Ind_Tf"
(5495) "S_a_Ind_Tip"
(5496) "S_a_Ind_Tp"
(5497) "S_a_Ind_Trp".
(5498)
(5499) S_a_1:
(5500) <> == C_S_CONTRA_SUBJ
(5501) <phon under> == a
(5502) <lex gloss eng> == ENG_Contra_S2/3s/d/p.
(5503)
(5504) S_a_2:
(5505) <> == C_S_SUBJUNCTIVE_SUBJ_PERSON
(5506) <phon under> == a

```

(5507) <lex gloss eng> == ENG_Sj_S2.
(5508)
(5509) S_a_Ind_Tf:
(5510) <> == C_S_IND_TENSE
(5511) <phon under> == a
(5512) <allo mec> == "MEC_19".
(5513)
(5514) S_a_Ind_Tip:
(5515) <> == C_S_IND_TENSE
(5516) <phon under> == a
(5517) <allo mec> == "MEC_21".
(5518)
(5519) S_a_Ind_Tp:
(5520) <> == C_S_IND_TENSE
(5521) <phon under> == a
(5522) <allo mec> == "MEC_21".
(5523)
(5524) S_a_Ind_Trp:
(5525) <> == C_S_IND_TENSE
(5526) <phon under> == a
(5527) <allo mec> == "MEC_21".
(5528)
(5529) S_boro:
(5530) <> == C_S_VERBAL_SUFFIX
(5531) <phon under> == boro
(5532) <lex gloss eng> == ENG_completely.
(5533)
(5534) S_d:
(5535) <> == "S_d_Ind_S3d"
(5536) "S_d_Ind_S3p".
(5537)
(5538) S_d_Ind_S3d:
(5539) <> == C_S_IND_SUBJ
(5540) <phon under> == d
(5541) <allo sec> == "SEC_08"
(5542) <allo mec> == "MEC_06".
(5543)
(5544) S_d_Ind_S3p:
(5545) <> == C_S_IND_SUBJ
(5546) <phon under> == d
(5547) <allo sec> == "SEC_08"
(5548) <allo mec> == "MEC_06".
(5549)
(5550) S_de:
(5551) <> == "S_de_Ind_S1p"
(5552) "S_de_Ind_S2p"
(5553) "S_de_vAux"
(5554) "S_de_1"
(5555) "S_de_1_Ind_S3p"
(5556) "S_de_2_Ind_S3p".
(5557)
(5558) S_de_1:
(5559) <> == C_S_VERBAL_SUFFIX
(5560) <phon under> == de
(5561) <lex gloss eng> == ENG_well.
(5562)
(5563) S_de_1_Ind_S3p:

(5564) <> == C_S_IND_SUBJ
(5565) <phon under> == de
(5566) <allo mec> == "MEC_12".
(5567)
(5568) S_de_2_Ind_S3p:
(5569) <> == C_S_IND_SUBJ
(5570) <phon under> == de
(5571) <allo sec> == "SEC_10"
(5572) <allo mec> == "MEC_14".
(5573)
(5574) S_de_Ind_S1p:
(5575) <> == C_S_IND_SUBJ
(5576) <phon under> == de
(5577) <allo sec> == "SEC_09"
(5578) <allo mec> == "MEC_07"
(5579) <lex gloss eng> == S_Ind_S1p.
(5580)
(5581) S_de_Ind_S2p:
(5582) <> == C_S_IND_SUBJ
(5583) <phon under> == de
(5584) <allo sec> == "SEC_11"
(5585) <allo mec> == "MEC_07".
(5586)
(5587) S_de_vAux:
(5588) <> == C_S_V_AUX
(5589) <phon under> == de
(5590) <lex gloss eng> == ENG_vAux_S3p.
(5591)
(5592) S_dere:
(5593) <> == "S_dere_1_Ind_S3d"
(5594) "S_dere_2_Ind_S3d"
(5595) "S_dere_vAux"
(5596) "S_dere_Ind_S1d"
(5597) "S_dere_Ind_S2d"
(5598) "S_dere_vAux".
(5599)
(5600) S_dere_1_Ind_S3d:
(5601) <> == C_S_IND_SUBJ
(5602) <phon under> == dere
(5603) <allo mec> == "MEC_12".
(5604)
(5605) S_dere_2_Ind_S3d:
(5606) <> == C_S_IND_SUBJ
(5607) <phon under> == dere
(5608) <allo sec> == "SEC_10"
(5609) <allo mec> == "MEC_06".
(5610)
(5611) S_dere_Ind_S1d:
(5612) <> == C_S_IND_SUBJ
(5613) <phon under> == dere
(5614) <allo sec> == "SEC_09"
(5615) <allo mec> == "MEC_07".
(5616)
(5617) S_dere_Ind_S2d:
(5618) <> == C_S_IND_SUBJ
(5619) <phon under> == dere
(5620) <allo sec> == "SEC_11"

```

(5621) <allo mec> == "MEC_07".
(5622)
(5623) S_dere_vAux:
(5624) <> == C_S_V_AUX
(5625) <phon under> == dere
(5626) <lex gloss eng> == ENG_vAux_S3d.
(5627)
(5628) S_du:
(5629) <> == C_S_SAME_REF_TEMPORALITY
(5630) <phon under> == du
(5631) <lex gloss eng> == ENG_TS.
(5632)
(5633) S_e:
(5634) <> == "S_e_Contra"
(5635) "S_e_Ind_Tf"
(5636) "S_e_Ind_Tip"
(5637) "S_e_Ind_Tp"
(5638) "S_e_Ind_Trp"
(5639) "S_e_Inf".
(5640)
(5641) S_e_Contra:
(5642) <> == C_S_CONTRA_SUBJ
(5643) <phon under> == e
(5644) <lex gloss eng> == ENG_Contra_S1s.
(5645)
(5646) S_e_Ind_Tf:
(5647) <> == C_S_IND_TENSE
(5648) <phon under> == e
(5649) <allo mec> == "MEC_18".
(5650)
(5651) S_e_Ind_Tip:
(5652) <> == C_S_IND_TENSE
(5653) <phon under> == e
(5654) <allo mec> == "MEC_16".
(5655)
(5656) S_e_Ind_Tp:
(5657) <> == C_S_IND_TENSE
(5658) <phon under> == e
(5659) <allo mec> == "MEC_16".
(5660)
(5661) S_e_Ind_Trp:
(5662) <> == C_S_IND_TENSE
(5663) <phon under> == e
(5664) <allo mec> == "MEC_16".
(5665)
(5666) S_e_Inf:
(5667) <> == C_S_INF
(5668) <phon under> == e
(5669) <lex form> == DEFAULT EPENTHETIC_W
(5670) <lex order class> == 80
(5671) <lex cat> == 'VR/Inf'
(5672) <allo sec default> == "SEC_03"
(5673) <lex gloss eng> == ENG_Inf.
(5674)
(5675) S_edede:
(5676) <> == C_S_CONTRA_SUBJ
(5677) <phon under> == edede

```

```

(5678) <lex gloss eng> == ENG_Contra_Slp.
(5679)
(5680) S_edere:
(5681) <> == C_S_CONTRA_SUBJ
(5682) <phon under> == edere
(5683) <lex gloss eng> == ENG_Contra_S1d.
(5684)
(5685) S_fe:
(5686) <> == "S_fe_1"
(5687) "S_fe_2".
(5688)
(5689) S_fe_1:
(5690) <> == C_SUFFIX
(5691) <phon under> == fe
(5692) <lex order class> == 100
(5693) <lex cat> == 'NW/NW Deictic/Deictic Adj/Adj'
(5694) <lex gloss eng> == ENG_like1.
(5695)
(5696) S_fe_2:
(5697) <> == C_S_PRONOUN_POSS
(5698) <phon under> == fe
(5699) <lex gloss eng> == ENG_P1s.
(5700)
(5701) S_fu:
(5702) <> == C_S_BENEFACTIVE
(5703) <phon under> == fu
(5704) <lex gloss eng> == ENG_B3s.
(5705)
(5706) S_g:
(5707) <> == "S_g_1" "S_g_2".
(5708)
(5709) S_g_1:
(5710) <> == C_S_VERBAL_SUFFIX
(5711) <phon under> == g
(5712) <lex gloss eng> == ENG_Hab
(5713) <lex order class> == 70
(5714) <lex cat> == 'VR/SFV VR/Nom'.
(5715)
(5716) S_g_2:
(5717) <> == C_S_DIFF_REF_TEMPORALITY
(5718) <phon under> == g
(5719) <lex gloss eng> == ENG_SR_TO.
(5720)
(5721) S_ga:
(5722) <> == C_S_SUBJUNCTIVE
(5723) <phon under> == ga
(5724) <lex gloss eng> == ENG_Prediction
(5725) <lex order class> == 100
(5726) <lex cat> == 'VSjS/Vsjs'.
(5727)
(5728) S_gana:
(5729) <> == C_S_PLURAL
(5730) <phon under> == gana
(5731) <lex gloss eng> == ENG_p1.
(5732)
(5733) S_ge:
(5734) <> == "S_ge_1" "S_ge_2".

```

(5735)
(5736) S_ge_1:
(5737) <> == C_S_CONTRA
(5738) <phon under> == ge
(5739) <lex gloss eng> == ENG_Contra
(5740) <lex form> == DEFAULT
(5741) <lex order class> == 100
(5742) <lex cat> == 'Contra/Contra'
(5743) <allo mec> == "MEC_25".
(5744)
(5745) S_ge_2:
(5746) <> == C_S_PRONOUN_POSS
(5747) <phon under> == ge
(5748) <lex gloss eng> == ENG_P1p.
(5749)
(5750) S_go:
(5751) <> == C_S_IND_SUBJ
(5752) <phon under> == go
(5753) <allo sec> == "SEC_10"
(5754) <allo mec> == "MEC_07".
(5755)
(5756) S_gu:
(5757) <> == "S_gu_Subj"
(5758) "S_gu_Inf".
(5759)
(5760) S_gu_Subj:
(5761) <> == C_S_SUBJUNCTIVE_SUBJ_NUMBER
(5762) <phon under> == gu
(5763) <lex gloss eng> == ENG_Sj_Sp.
(5764)
(5765) S_gu_Inf:
(5766) <> == C_S_INF_SUBJ
(5767) <phon under> == gu
(5768) <lex gloss eng> == ENG_Inf_Sp.
(5769)
(5770) S_ha:
(5771) <> == "S_ha_1" "S_ha_2".
(5772)
(5773) S_ha_1:
(5774) <> == C_S_ASPECT
(5775) <phon under> == ha
(5776) <lex gloss eng> == ENG_perf
(5777) <lex order class> == 100
(5778) <lex cat> == 'SFV/SFVW'.
(5779)
(5780) S_ha_2:
(5781) <> == C_SUFFIX
(5782) <phon under> == ha
(5783) <lex gloss eng> == ENG_pertaining_to
(5784) <lex cat> == 'Adj/Adj NW/NW Deictic/Deictic PosPro/PosPro'.
(5785)
(5786) S_hagi:
(5787) <> == C_S_BENEFACTIVE_MC1
(5788) <phon under> == hagi
(5789) <lex gloss eng> == ENG_B1p.
(5790)
(5791) S_hai:

(5792) <> == C_S_BENEFACTIVE
 (5793) <phon under> == hai
 (5794) <lex gloss eng> == ENG_B1s
 (5795) <lex form> == DEFAULT GLIDE.
 (5796)
 (5797) S_hani:
 (5798) <> == C_S_BENEFACTIVE_MC1
 (5799) <phon under> == hani
 (5800) <lex gloss eng> == ENG_B2s.
 (5801)
 (5802) S_hari:
 (5803) <> == C_S_BENEFACTIVE_MC1
 (5804) <phon under> == hari
 (5805) <lex gloss eng> == ENG_B1d.
 (5806)
 (5807) S_he:
 (5808) <> == "S_he_Ind_S1s"
 (5809) "S_he_Ind_S2s".
 (5810)
 (5811) S_he_Ind_S1s:
 (5812) <> == C_S_IND_SUBJ
 (5813) <phon under> == he
 (5814) <allo sec> == "SEC_09"
 (5815) <allo mec> == "MEC_07".
 (5816)
 (5817) S_he_Ind_S2s:
 (5818) <> == C_S_IND_SUBJ
 (5819) <phon under> == he
 (5820) <allo sec> == "SEC_11"
 (5821) <allo mec> == "MEC_07".
 (5822)
 (5823) S_higi:
 (5824) <> == C_S_OBJECT_MC1
 (5825) <phon under> == higi
 (5826) <lex gloss eng> == ENG_O1p.
 (5827)
 (5828) S_hii:
 (5829) <> == C_S_OBJECT
 (5830) <phon under> == hii
 (5831) <lex gloss eng> == ENG_O1s
 (5832) <lex form> == ELISION GLIDE.
 (5833)
 (5834) S_hini:
 (5835) <> == C_S_OBJECT_MC1
 (5836) <phon under> == hini
 (5837) <lex gloss eng> == ENG_O2s.
 (5838)
 (5839) S_hiri:
 (5840) <> == C_S_OBJECT_MC1
 (5841) <phon under> == hiri
 (5842) <lex gloss eng> == ENG_O1d.
 (5843)
 (5844) S_i:
 (5845) <> == "S_i_Ind_S3s" "S_i_Inf".
 (5846)
 (5847) S_i_Ind_S3s:
 (5848) <> == C_S_IND_SUBJ

(5849) <phon under> == i
 (5850) <allo sec> == "SEC_11"
 (5851) <allo mec> == "MEC_09".
 (5852)
 (5853) S_i_Inf:
 (5854) <> == C_S_INF_SUBJ
 (5855) <phon under> == i
 (5856) <lex gloss eng> == ENG_Inf_Ss.
 (5857)
 (5858) S_Ind_S1d:
 (5859) <> == C_S_IND_SUBJ
 (5860) <lex form> == "S_0_Ind_S1d"
 (5861) "S_dere_Ind_S1d"
 (5862) "S_re_Ind_S1d"
 (5863) "S_yare_1_Ind_S1d"
 (5864) "S_yare_2_Ind_S1d"
 (5865) <phon under> == yare
 (5866) <lex gloss eng> == ENG_Ind_S1d.
 (5867)
 (5868) S_Ind_S1p:
 (5869) <> == C_S_IND_SUBJ
 (5870) <lex form> == "S_0_Ind_S1p"
 (5871) "S_de_Ind_S1p"
 (5872) "S_ya_1_Ind_S1p"
 (5873) "S_ya_2_Ind_S1p"
 (5874) "S_ye_Ind_S1p"
 (5875) <phon under> == ya
 (5876) <lex gloss eng> == ENG_Ind_S1p.
 (5877)
 (5878) S_Ind_S1s:
 (5879) <> == C_S_IND_SUBJ
 (5880) <lex form> == "S_he_Ind_S1s"
 (5881) "S_n_Ind_S1s"
 (5882) "S_na_Ind_S1s"
 (5883) "S_ne_1_Ind_S1s"
 (5884) "S_ne_2_Ind_S1s"
 (5885) "S_ni_1_Ind_S1s"
 (5886) "S_ni_2_Ind_S1s"
 (5887) <phon under> == ni
 (5888) <lex gloss eng> == ENG_Ind_S1s.
 (5889)
 (5890) S_Ind_S2d:
 (5891) <> == C_S_IND_SUBJ
 (5892) <lex form> == "S_dere_Ind_S2d" "S_re_Ind_S2d" "S_yare_Ind_S2d"
 (5893) <phon under> == yare
 (5894) <lex gloss eng> == ENG_Ind_S2d.
 (5895)
 (5896) S_Ind_S2p:
 (5897) <> == C_S_IND_SUBJ
 (5898) <lex form> == "S_de_Ind_S2p" "S_ya_Ind_S2p" "S_ye_Ind_S2p"
 (5899) <phon under> == ya
 (5900) <lex gloss eng> == ENG_Ind_S2p.
 (5901)
 (5902) S_Ind_S2s:
 (5903) <> == C_S_IND_SUBJ
 (5904) <lex form> == "S_he_Ind_S2s" "S_na_Ind_S2s" "S_ne_1_Ind_S2s"
 (5905) "S_ne_2_Ind_S2s" "S_ni_Ind_S2s"

(5906) <phon under> == ni
 (5907) <lex gloss eng> == ENG_Ind_S2s.
 (5908)
 (5909) S_Ind_S3d:
 (5910) <> == C_S_IND_SUBJ
 (5911) <lex form> == "S_d_Ind_S3d" "S_dere_1_Ind_S3d"
 (5912) "S_dere_2_Ind_S3d" "S_re_Ind_S3d"
 "S_ro"
 (5913) <phon under> == dere
 (5914) <lex gloss eng> == ENG_Ind_S3d.
 (5915)
 (5916) S_Ind_S3p:
 (5917) <> == C_S_IND_SUBJ
 (5918) <lex form> == "S_d_Ind_S3p" "S_de_1_Ind_S3p" "S_de_2_Ind_S3p"
 (5919) "S_go" "S_ne_Ind_S3p"
 (5920) <phon under> == de
 (5921) <lex gloss eng> == ENG_Ind_S3p.
 (5922)
 (5923) S_Ind_S3s:
 (5924) <> == C_S_IND_SUBJ
 (5925) <lex form> == "S_0_1_Ind_S3s" "S_0_2_Ind_S3s" "S_i_Ind_S3s"
 (5926) "S_n_Ind_S3s" "S_na_1_Ind_S3s"
 "S_ta"
 (5927) "S_na_2_Ind_S3s"
 (5928) <phon under> == na
 (5929) <lex gloss eng> == ENG_Ind_S3s.
 (5930)
 (5931) S_Ind_Tf:
 (5932) <> == C_S_IND_TENSE
 (5933) <lex form> == "S_e_Ind_Tf" "S_a_Ind_Tf" "S_o_Ind_Tf"
 (5934) <phon under> == o
 (5935) <lex gloss eng> == ENG_Ind_Tf.
 (5936)
 (5937) S_Ind_Tip:
 (5938) <> == C_S_IND_TENSE
 (5939) <lex form> == "S_e_Ind_Tip" "S_a_Ind_Tip" "S_0_Ind_Tip"
 (5940) <phon under> == a
 (5941) <lex gloss eng> == ENG_Ind_Tip.
 (5942)
 (5943) S_Ind_Tp:
 (5944) <> == C_S_IND_TENSE
 (5945) <lex form> == "S_0_Ind_Tp" "S_e_Ind_Tp" "S_a_Ind_Tp"
 (5946) <phon under> == 0
 (5947) <lex gloss eng> == ENG_Ind_Tp.
 (5948)
 (5949) S_Ind_Trp:
 (5950) <> == C_S_IND_TENSE
 (5951) <lex form> == "S_e_Ind_Trp" "S_a_Ind_Trp"
 (5952) <phon under> == e
 (5953) <lex gloss eng> == ENG_Ind_Trp.
 (5954)
 (5955) S_mu:
 (5956) <> == C_S_VERBAL_SUFFIX
 (5957) <phon under> == mu
 (5958) <lex gloss eng> == ENG_causative
 (5959) <lex order class> == 10.
 (5960)

(5961) S_n:
(5962) <> == "S_n_Ind_S1s" "S_n_Ind_S3s".
(5963)
(5964) S_n_Ind_S1s:
(5965) <> == C_S_IND_SUBJ
(5966) <phon under> == n
(5967) <allo sec> == "SEC_08"
(5968) <allo mec> == "MEC_06".
(5969)
(5970) S_n_Ind_S3s:
(5971) <> == C_S_IND_SUBJ
(5972) <phon under> == n
(5973) <allo sec> == "SEC_08"
(5974) <allo mec> == "MEC_06".
(5975)
(5976) S_na:
(5977) <> == "S_na_1_Ind_S3s" "S_na_2" "S_na_2_Ind_S3s"
(5978) "S_na_Ind_S1s" "S_na_Ind_S2s".
(5979)
(5980) S_na_1_Ind_S3s:
(5981) <> == C_S_IND_SUBJ
(5982) <phon under> == na
(5983) <allo mec> == "MEC_12".
(5984)
(5985) S_na_2:
(5986) <> == C_S_V_AUX
(5987) <phon under> == na
(5988) <lex gloss eng> == ENG_vAux_S3s.
(5989)
(5990) S_na_2_Ind_S3s:
(5991) <> == C_S_IND_SUBJ
(5992) <phon under> == na
(5993) <allo sec> == "SEC_10"
(5994) <allo mec> == "MEC_06".
(5995)
(5996) S_na_Ind_S1s:
(5997) <> == C_S_IND_SUBJ
(5998) <phon under> == na
(5999) <allo mec> == "MEC_11".
(6000)
(6001) S_na_Ind_S2s:
(6002) <> == C_S_IND_SUBJ
(6003) <phon under> == na
(6004) <allo mec> == "MEC_11".
(6005)
(6006) S_nade:
(6007) <> == C_S_V_AUX
(6008) <phon under> == nade
(6009) <lex gloss eng> == ENG_vAux_Tf_S2p.
(6010)
(6011) S_nadere:
(6012) <> == C_S_V_AUX
(6013) <phon under> == nadere
(6014) <lex gloss eng> == ENG_vAux_Tf_S2d.
(6015)
(6016) S_nagi:
(6017) <> == C_S_BENEFACTIVE_MC1

(6018) <phon under> == nagi
(6019) <lex gloss eng> == ENG_B3p.
(6020)
(6021) S_nahe:
(6022) <> == C_S_V_AUX
(6023) <phon under> == nahe
(6024) <lex gloss eng> == ENG_vAux_Tf_S2s.
(6025)
(6026) S_nari:
(6027) <> == C_S_BENEFACTIVE_MC1
(6028) <phon under> == nari
(6029) <lex gloss eng> == ENG_B3d.
(6030)
(6031) S_ne:
(6032) <> == "S_ne_1_Ind_S1s" "S_ne_1_Ind_S2s" "S_ne_2"
(6033) "S_ne_2_Ind_S2s" "S_ne_2_Ind_S2s"
"S_ne_Ind_S3p"
(6034) "S_ne_vAux".
(6035)
(6036) S_ne_1_Ind_S1s:
(6037) <> == C_S_IND_SUBJ
(6038) <phon under> == ne
(6039) <allo mec> == "MEC_08".
(6040)
(6041) S_ne_1_Ind_S2s:
(6042) <> == C_S_IND_SUBJ
(6043) <phon under> == ne
(6044) <allo mec> == "MEC_13".
(6045)
(6046) S_ne_2:
(6047) <> == C_S_PRONOUN_POSS
(6048) <phon under> == ne
(6049) <lex gloss eng> == ENG_P2s
(6050) <lex form> == DEFAULT_ALLO2
(6051) <allo form allo2> == te.
(6052)
(6053) S_ne_2_Ind_S1s:
(6054) <> == C_S_IND_SUBJ
(6055) <phon under> == ne
(6056) <allo sec> == "SEC_09"
(6057) <allo mec> == "MEC_08" "MEC_09".
(6058)
(6059) S_ne_2_Ind_S2s:
(6060) <> == C_S_IND_SUBJ
(6061) <phon under> == ne
(6062) <allo sec> == "SEC_09"
(6063) <allo mec> == "MEC_08" "MEC_09".
(6064)
(6065) S_ne_Ind_S3p:
(6066) <> == C_S_IND_SUBJ
(6067) <phon under> == ne
(6068) <allo sec> == "SEC_11"
(6069) <allo mec> == "MEC_11" "MEC_13" "MEC_09".
(6070)
(6071) S_ne_vAux:
(6072) <> == C_S_V_AUX
(6073) <phon under> == ne

(6074) <lex gloss eng> == ENG_vAux_Tnf_S2s.
(6075)
(6076) S_nege:
(6077) <> == C_S_PRONOUN_POSS
(6078) <phon under> == nege
(6079) <lex gloss eng> == ENG_P3p.
(6080)
(6081) S_nere:
(6082) <> == C_S_PRONOUN_POSS
(6083) <phon under> == nere
(6084) <lex gloss eng> == ENG_P3d.
(6085)
(6086) S_neu:
(6087) <> == C_SUFFIX
(6088) <phon under> == neu
(6089) <lex order class> == 100
(6090) <lex cat> == 'Deictic/Deictic'
(6091) <lex gloss eng> == ENG_like2
(6092) <allo mec> == "MEC_24".
(6093)
(6094) S_nga:
(6095) <> == "S_nga_1" "S_nga_2" "S_nga_3".
(6096)
(6097) S_nga_1:
(6098) <> == C_SUFFIX
(6099) <phon under> == nga
(6100) <lex cat> == 'NW/NW RflPro/RflPro PN/PN Adj/Adj PosPro/PosPro
(6101) PerPro/PerPro Prep/Prep
Deictic/Deictic VR/VR'
(6102) <lex gloss eng> == ENG_just/only.
(6103)
(6104) S_nga_2:
(6105) <> == C_S_SUBJUNCTIVE
(6106) <phon under> == nga
(6107) <lex gloss eng> == ENG_certainty
(6108) <lex order class> == 110
(6109) <lex cat> == 'VSjS/Vsjs'.
(6110)
(6111) S_nga_3:
(6112) <> == C_S_VERBAL_SUFFIX
(6113) <phon under> == nga
(6114) <lex gloss eng> == ENG_SR
(6115) <lex order class> == 110
(6116) <lex cat> == 'SMV/SMVW Contra/SMVW' .
(6117)
(6118) S_ni:
(6119) <> == "S_ni_vAux" "S_ni_1_Ind_S1s" "S_ni_2_Ind_S1s"
(6120) "S_ni_Ind_S2s".
(6121)
(6122) S_ni_vAux:
(6123) <> == C_S_V_AUX
(6124) <phon under> == ni
(6125) <lex gloss eng> == ENG_vAux_S1s.
(6126)
(6127) S_ni_1_Ind_S1s:
(6128) <> == C_S_IND_SUBJ
(6129) <phon under> == ni

(6130) <allo mec> == "MEC_12" "MEC_08".
(6131)
(6132) S_ni_2_Ind_S1s:
(6133) <> == C_S_IND_SUBJ
(6134) <phon under> == ni
(6135) <allo sec> == "SEC_10"
(6136) <allo mec> == "MEC_06".
(6137)
(6138) S_ni_Ind_S2s:
(6139) <> == C_S_IND_SUBJ
(6140) <phon under> == ni
(6141) <allo mec> == "MEC_12" "MEC_08".
(6142)
(6143) S_nigi:
(6144) <> == C_S_OBJECT_MC1
(6145) <phon under> == nigi
(6146) <lex gloss eng> == ENG_O3p.
(6147)
(6148) S_niri:
(6149) <> == C_S_OBJECT_MC1
(6150) <phon under> == niri
(6151) <lex gloss eng> == ENG_O3d.
(6152)
(6153) S_o:
(6154) <> == "S_o_Subj" "S_o_Ind_Tf".
(6155)
(6156) S_o_Subj:
(6157) <> == C_S_SUBJUNCTIVE_SUBJ_PERSON
(6158) <phon under> == o
(6159) <lex gloss eng> == ENG_Sj_S1/3.
(6160)
(6161) S_o_Ind_Tf:
(6162) <> == C_S_IND_TENSE
(6163) <phon under> == o
(6164) <allo mec> == "MEC_20".
(6165)
(6166) S_ou:
(6167) <> == "S_ou_Attrib" "S_ou_Nom".
(6168)
(6169) S_ou_Attrib:
(6170) <> == C_S_ATTRIB
(6171) <phon under> == ou
(6172) <lex gloss eng> == ENG_Attrib.
(6173)
(6174) S_ou_Nom:
(6175) <> == C_SUFFIX
(6176) <phon under> == ou
(6177) <lex form> == DEFAULT EPENTHETIC_W
(6178) <lex order class> == 110
(6179) <lex cat> == 'VR/Nom Nom/Nom'
(6180) <allo sec default> == "SEC_03"
(6181) <lex gloss eng> == ENG_Nom.
(6182)
(6183) S_re:
(6184) <> == "S_re_Ind_S1d" "S_re_Ind_S2d" "S_re_2" "S_re_Ind_S3d".
(6185)
(6186) S_re_2:

(6187) <> == C_S_PRONOUN_POSS
(6188) <phon under> == re
(6189) <lex gloss eng> == ENG_P1d.
(6190)
(6191) S_re_Ind_S1d:
(6192) <> == C_S_IND_SUBJ
(6193) <phon under> == re
(6194) <allo sec> == "SEC_09"
(6195) <allo mec> == "MEC_08" "MEC_09".
(6196)
(6197) S_re_Ind_S2d:
(6198) <> == C_S_IND_SUBJ
(6199) <phon under> == re
(6200) <allo sec> == "SEC_09"
(6201) <allo mec> == "MEC_08" "MEC_09".
(6202)
(6203) S_re_Ind_S3d:
(6204) <> == C_S_IND_SUBJ
(6205) <phon under> == re
(6206) <allo sec> == "SEC_11"
(6207) <allo mec> == "MEC_11" "MEC_13" "MEC_09".
(6208)
(6209) S_ro:
(6210) <> == C_S_IND_SUBJ
(6211) <phon under> == ro
(6212) <allo sec> == "SEC_10"
(6213) <allo mec> == "MEC_07".
(6214)
(6215) S_ru:
(6216) <> == "S_ru_1" "S_ru_2" "S_ru_Inf".
(6217)
(6218) S_ru_1:
(6219) <> == C_S_PLURAL
(6220) <phon under> == ru
(6221) <lex gloss eng> == ENG_p2.
(6222)
(6223) S_ru_2:
(6224) <> == C_S_SUBJUNCTIVE_SUBJ_NUMBER
(6225) <phon under> == ru
(6226) <lex gloss eng> == ENG_Sj_Sd.
(6227)
(6228) S_ru_Inf:
(6229) <> == C_S_INF_SUBJ
(6230) <phon under> == ru
(6231) <lex gloss eng> == ENG_Inf_Sd.
(6232)
(6233) S_ta:
(6234) <> == C_S_IND_SUBJ
(6235) <phon under> == ta
(6236) <allo mec> == "MEC_15".
(6237)
(6238) S_tagi:
(6239) <> == C_S_BENEFACTIVE_MC1
(6240) <phon under> == tagi
(6241) <lex gloss eng> == ENG_B2p.
(6242)
(6243) S_tari:

(6244) <> == C_S_BENEFACTIVE_MC1
 (6245) <phon under> == tari
 (6246) <lex gloss eng> == ENG_B2d.
 (6247)
 (6248) S_tege:
 (6249) <> == C_S_PRONOUN_POSS
 (6250) <phon under> == tege
 (6251) <lex gloss eng> == ENG_P2p.
 (6252)
 (6253) S_tere:
 (6254) <> == C_S_PRONOUN_POSS
 (6255) <phon under> == tere
 (6256) <lex gloss eng> == ENG_P2d.
 (6257)
 (6258) S_tigi:
 (6259) <> == C_S_OBJECT_MC1
 (6260) <phon under> == tigi
 (6261) <lex gloss eng> == ENG_O2p.
 (6262)
 (6263) S_tiri:
 (6264) <> == C_S_OBJECT_MC1
 (6265) <phon under> == tiri
 (6266) <lex gloss eng> == ENG_O2d.
 (6267)
 (6268) S_tu:
 (6269) <> == C_S_OBJECT
 (6270) <phon under> == tu
 (6271) <lex gloss eng> == ENG_O3s.
 (6272)
 (6273) S_u:
 (6274) <> == C_S_SUBJUNCTIVE_SUBJ_NUMBER
 (6275) <phon under> == u
 (6276) <lex gloss eng> == ENG_Sj_Ss.
 (6277)
 (6278) S_ya:
 (6279) <> == "S_ya_1_Ind_S1p" "S_ya_2_Ind_S1p" "S_ya_Ind_S2p"
 (6280) "S_ya_Manner" "S_ya_1_vAux" "S_ya_2_vAux".
 (6281)
 (6282) S_ya_1_Ind_S1p:
 (6283) <> == C_S_IND_SUBJ
 (6284) <phon under> == ya
 (6285) <allo mec> == "MEC_10".
 (6286)
 (6287) S_ya_1_vAux:
 (6288) <> == C_S_V_AUX
 (6289) <phon under> == ya
 (6290) <lex gloss eng> == ENG_vAux_S1p.
 (6291)
 (6292) S_ya_2_Ind_S1p:
 (6293) <> == C_S_IND_SUBJ
 (6294) <phon under> == ya
 (6295) <allo sec> == "SEC_10"
 (6296) <allo mec> == "MEC_06".
 (6297)
 (6298) S_ya_2_vAux:
 (6299) <> == C_S_V_AUX
 (6300) <phon under> == ya

(6301) <lex gloss eng> == ENG_vAux_Tnf_S2p.
(6302)
(6303) S_ya_Ind_S2p:
(6304) <> == C_S_IND_SUBJ
(6305) <phon under> == ya
(6306) <allo mec> == "MEC_10".
(6307)
(6308) S_ya_Manner:
(6309) <> == C_S_VERBAL_SUFFIX
(6310) <phon under> == ya
(6311) <lex gloss eng> == ENG_quickly_2.
(6312)
(6313) S_yare:
(6314) <> == "S_yare_1_Ind_S1d" "S_yare_2_Ind_S1d" "S_yare_1_vAux"
(6315) "S_yare_2_vAux" "S_yare_Ind_S2d".
(6316)
(6317) S_yare_1_Ind_S1d:
(6318) <> == C_S_IND_SUBJ
(6319) <phon under> == yare
(6320) <allo mec> == "MEC_10".
(6321)
(6322) S_yare_1_vAux:
(6323) <> == C_S_V_AUX
(6324) <phon under> == yare
(6325) <lex gloss eng> == ENG_vAux_S1d.
(6326)
(6327) S_yare_2_Ind_S1d:
(6328) <> == C_S_IND_SUBJ
(6329) <phon under> == yare
(6330) <allo sec> == "SEC_10"
(6331) <allo mec> == "MEC_06".
(6332)
(6333) S_yare_2_vAux:
(6334) <> == C_S_V_AUX
(6335) <phon under> == yare
(6336) <lex gloss eng> == ENG_vAux_Tnf_S2d.
(6337)
(6338) S_yare_Ind_S2d:
(6339) <> == C_S_IND_SUBJ
(6340) <phon under> == yare
(6341) <allo mec> == "MEC_10".
(6342)
(6343) S_ye:
(6344) <> == "S_ye_Ind_S1p" "S_ye_Ind_S2p".
(6345)
(6346) S_ye_Ind_S1p:
(6347) <> == C_S_IND_SUBJ
(6348) <phon under> == ye
(6349) <allo sec> == "SEC_09"
(6350) <allo mec> == "MEC_08" "MEC_09".
(6351)
(6352) S_ye_Ind_S2p:
(6353) <> == C_S_IND_SUBJ
(6354) <phon under> == ye
(6355) <allo sec> == "SEC_09"
(6356) <allo mec> == "MEC_08" "MEC_09".
(6357)

A3.3 The Ogea DATR Query File

A3.3.1 Description

DATR theory (lexicon) files contain sets of nodes. Queries against a theory file may be made interactively from a DATR command line, or alternatively, may be run in batch mode. In batch mode, a query file may be used so the user does not have to manually type in each query. The Ogea query file listing in this appendix is used to generate AMPLE entries from the Ogea DATR theory file. Each entry in the query file has a corresponding entry in the original Ogea AMPLE root or suffix database file. The path `<amp lex mor name>` is used to direct the query to invoke the AMPLE interface and to use the morph name as the value for the record marker.

A3.3.2 File Listing

Filename: ogea.qry

Note: line numbers added for reference only. Not in actual file.

```
(6360)      ENG_able:<amp lex morname>
(6361)      ENG_allow/leave:<amp lex morname>
(6362)      ENG_always:<amp lex morname>
(6363)      ENG_animal:<amp lex morname>
(6364)      ENG_appear/arrive:<amp lex morname>
(6365)      ENG_ask:<amp lex morname>
(6366)      ENG_at:<amp lex morname>
(6367)      ENG_Attrib:<amp lex morname>
(6368)      ENG_B1d:<amp lex morname>
(6369)      ENG_B1p:<amp lex morname>
(6370)      ENG_B1s:<amp lex morname>
(6371)      ENG_B2d:<amp lex morname>
(6372)      ENG_B2p:<amp lex morname>
(6373)      ENG_B2s:<amp lex morname>
(6374)      ENG_B3d:<amp lex morname>
(6375)      ENG_B3p:<amp lex morname>
```

(6376) ENG_B3s:<amp; lex morname>
(6377) ENG_bad:<amp; lex morname>
(6378) ENG_be:<amp; lex morname>
(6379) ENG_become_bad:<amp; lex morname>
(6380) ENG_become_crooked:<amp; lex morname>
(6381) ENG_become_good:<amp; lex morname>
(6382) ENG_become_heavy:<amp; lex morname>
(6383) ENG_become_large:<amp; lex morname>
(6384) ENG_become_many:<amp; lex morname>
(6385) ENG_become_nothing:<amp; lex morname>
(6386) ENG_become_red:<amp; lex morname>
(6387) ENG_become_short:<amp; lex morname>
(6388) ENG_become_small:<amp; lex morname>
(6389) ENG_become_straight:<amp; lex morname>
(6390) ENG_become_true:<amp; lex morname>
(6391) ENG_become_wild:<amp; lex morname>
(6392) ENG_bound:<amp; lex morname>
(6393) ENG_break_in_two:<amp; lex morname>
(6394) ENG_causative:<amp; lex morname>
(6395) ENG_certainty:<amp; lex morname>
(6396) ENG_chase:<amp; lex morname>
(6397) ENG_cloth:<amp; lex morname>
(6398) ENG_come_down:<amp; lex morname>
(6399) ENG_come_up:<amp; lex morname>
(6400) ENG_completely:<amp; lex morname>
(6401) ENG_connect/join:<amp; lex morname>
(6402) ENG_consume:<amp; lex morname>
(6403) ENG_Contra:<amp; lex morname>
(6404) ENG_Contra_S1d:<amp; lex morname>
(6405) ENG_Contra_S1p:<amp; lex morname>
(6406) ENG_Contra_S1s:<amp; lex morname>
(6407) ENG_Contra_S2/3s/d/p:<amp; lex morname>
(6408) ENG_crooked:<amp; lex morname>
(6409) ENG_cut:<amp; lex morname>
(6410) ENG_deflect:<amp; lex morname>
(6411) ENG_desire:<amp; lex morname>
(6412) ENG_die:<amp; lex morname>
(6413) ENG_different:<amp; lex morname>
(6414) ENG_do_not:<amp; lex morname>
(6415) ENG_do_thus:<amp; lex morname>
(6416) ENG_do_work:<amp; lex morname>
(6417) ENG_dream:<amp; lex morname>
(6418) ENG_encircle:<amp; lex morname>
(6419) ENG_face:<amp; lex morname>
(6420) ENG_fear:<amp; lex morname>
(6421) ENG_fight:<amp; lex morname>
(6422) ENG_fill:<amp; lex morname>
(6423) ENG_finish:<amp; lex morname>
(6424) ENG_fly:<amp; lex morname>
(6425) ENG_follow:<amp; lex morname>
(6426) ENG_gather:<amp; lex morname>
(6427) ENG_get_O1d:<amp; lex morname>
(6428) ENG_get_O1p:<amp; lex morname>
(6429) ENG_get_O1s:<amp; lex morname>
(6430) ENG_get_O2d:<amp; lex morname>
(6431) ENG_get_O2p:<amp; lex morname>
(6432) ENG_get_O2s:<amp; lex morname>

(6433) ENG_get_03d:<amp lex morname>
(6434) ENG_get_03p:<amp lex morname>
(6435) ENG_get_03s:<amp lex morname>
(6436) ENG_get_up:<amp lex morname>
(6437) ENG_get/take:<amp lex morname>
(6438) ENG_give_01d:<amp lex morname>
(6439) ENG_give_01p:<amp lex morname>
(6440) ENG_give_01s:<amp lex morname>
(6441) ENG_give_02d:<amp lex morname>
(6442) ENG_give_02p:<amp lex morname>
(6443) ENG_give_02s:<amp lex morname>
(6444) ENG_give_03d:<amp lex morname>
(6445) ENG_give_03p:<amp lex morname>
(6446) ENG_give_03s:<amp lex morname>
(6447) ENG_go:<amp lex morname>
(6448) ENG_go_down:<amp lex morname>
(6449) ENG_good:<amp lex morname>
(6450) ENG_Hab:<amp lex morname>
(6451) ENG_hand:<amp lex morname>
(6452) ENG_he/she/it:<amp lex morname>
(6453) ENG_hear/understand/obey:<amp lex morname>
(6454) ENG_heavy:<amp lex morname>
(6455) ENG_help:<amp lex morname>
(6456) ENG_himself/herself/itself:<amp lex morname>
(6457) ENG_his/her/its:<amp lex morname>
(6458) ENG_hit_01d:<amp lex morname>
(6459) ENG_hit_01p:<amp lex morname>
(6460) ENG_hit_01s:<amp lex morname>
(6461) ENG_hit_02d:<amp lex morname>
(6462) ENG_hit_02p:<amp lex morname>
(6463) ENG_hit_02s:<amp lex morname>
(6464) ENG_hit_03d:<amp lex morname>
(6465) ENG_hit_03p:<amp lex morname>
(6466) ENG_hit_03s:<amp lex morname>
(6467) ENG_I:<amp lex morname>
(6468) ENG_idle:<amp lex morname>
(6469) ENG_Ind_S1d:<amp lex morname>
(6470) ENG_Ind_S1p:<amp lex morname>
(6471) ENG_Ind_S1s:<amp lex morname>
(6472) ENG_Ind_S2d:<amp lex morname>
(6473) ENG_Ind_S2p:<amp lex morname>
(6474) ENG_Ind_S2s:<amp lex morname>
(6475) ENG_Ind_S3d:<amp lex morname>
(6476) ENG_Ind_S3p:<amp lex morname>
(6477) ENG_Ind_S3s:<amp lex morname>
(6478) ENG_Ind_Tf:<amp lex morname>
(6479) ENG_Ind_Tip:<amp lex morname>
(6480) ENG_Ind_Tp:<amp lex morname>
(6481) ENG_Ind_Trp:<amp lex morname>
(6482) ENG_Inf:<amp lex morname>
(6483) ENG_Inf_Sd:<amp lex morname>
(6484) ENG_Inf_Sp:<amp lex morname>
(6485) ENG_Inf_Ss:<amp lex morname>
(6486) ENG_inside:<amp lex morname>
(6487) ENG_judge:<amp lex morname>
(6488) ENG_just/only:<amp lex morname>
(6489) ENG_knife:<amp lex morname>

(6490) ENG_large:<amp lex morname>
(6491) ENG_leg/foot:<amp lex morname>
(6492) ENG_leg/foot_3sP:<amp lex morname>
(6493) ENG_lie:<amp lex morname>
(6494) ENG_light:<amp lex morname>
(6495) ENG_like_that:<amp lex morname>
(6496) ENG_like1:<amp lex morname>
(6497) ENG_like2:<amp lex morname>
(6498) ENG_look_for:<amp lex morname>
(6499) ENG_man:<amp lex morname>
(6500) ENG_many:<amp lex morname>
(6501) ENG_my:<amp lex morname>
(6502) ENG_myself:<amp lex morname>
(6503) ENG_N_not:<amp lex morname>
(6504) ENG_Nom:<amp lex morname>
(6505) ENG_nose:<amp lex morname>
(6506) ENG_O1d:<amp lex morname>
(6507) ENG_O1p:<amp lex morname>
(6508) ENG_O1s:<amp lex morname>
(6509) ENG_O2d:<amp lex morname>
(6510) ENG_O2p:<amp lex morname>
(6511) ENG_O2s:<amp lex morname>
(6512) ENG_O3d:<amp lex morname>
(6513) ENG_O3p:<amp lex morname>
(6514) ENG_O3s:<amp lex morname>
(6515) ENG_ocean:<amp lex morname>
(6516) ENG_one/a:<amp lex morname>
(6517) ENG_our_dual:<amp lex morname>
(6518) ENG_our_plural:<amp lex morname>
(6519) ENG_ourselves_dual:<amp lex morname>
(6520) ENG_ourselves_plural:<amp lex morname>
(6521) ENG_p1:<amp lex morname>
(6522) ENG_P1d:<amp lex morname>
(6523) ENG_P1p:<amp lex morname>
(6524) ENG_P1s:<amp lex morname>
(6525) ENG_p2:<amp lex morname>
(6526) ENG_P2d:<amp lex morname>
(6527) ENG_P2p:<amp lex morname>
(6528) ENG_P2s:<amp lex morname>
(6529) ENG_P3d:<amp lex morname>
(6530) ENG_P3p:<amp lex morname>
(6531) ENG_P3s:<amp lex morname>
(6532) ENG_perf:<amp lex morname>
(6533) ENG_pertaining_to:<amp lex morname>
(6534) ENG_pig:<amp lex morname>
(6535) ENG_place:<amp lex morname>
(6536) ENG_plant:<amp lex morname>
(6537) ENG_poke:<amp lex morname>
(6538) ENG_pour_out:<amp lex morname>
(6539) ENG_Prediction:<amp lex morname>
(6540) ENG_prepare:<amp lex morname>
(6541) ENG_press:<amp lex morname>
(6542) ENG_put:<amp lex morname>
(6543) ENG_quickly:<amp lex morname>
(6544) ENG_quickly_2:<amp lex morname>
(6545) ENG_red:<amp lex morname>
(6546) ENG_remove:<amp lex morname>

(6547) ENG_run_away:<amp lex morname>
(6548) ENG_see_01d:<amp lex morname>
(6549) ENG_see_01p:<amp lex morname>
(6550) ENG_see_01s:<amp lex morname>
(6551) ENG_see_02d:<amp lex morname>
(6552) ENG_see_02p:<amp lex morname>
(6553) ENG_see_02s:<amp lex morname>
(6554) ENG_see_03d:<amp lex morname>
(6555) ENG_see_03p:<amp lex morname>
(6556) ENG_see_03s:<amp lex morname>
(6557) ENG_send:<amp lex morname>
(6558) ENG_shine:<amp lex morname>
(6559) ENG_short:<amp lex morname>
(6560) ENG_shove:<amp lex morname>
(6561) ENG_show:<amp lex morname>
(6562) ENG_sibling_same_sex_his/her/its:<amp lex morname>
(6563) ENG_sibling_same_sex_possessive:<amp lex morname>
(6564) ENG_sibling_same_sex_referential:<amp lex morname>
(6565) ENG_sit:<amp lex morname>
(6566) ENG_Sj_S1/3:<amp lex morname>
(6567) ENG_Sj_S2:<amp lex morname>
(6568) ENG_Sj_Sd:<amp lex morname>
(6569) ENG_Sj_Sp:<amp lex morname>
(6570) ENG_Sj_Ss:<amp lex morname>
(6571) ENG_sleep_pl:<amp lex morname>
(6572) ENG_small:<amp lex morname>
(6573) ENG_smooth:<amp lex morname>
(6574) ENG_some:<amp lex morname>
(6575) ENG_speak:<amp lex morname>
(6576) ENG_SR:<amp lex morname>
(6577) ENG_SR_TO:<amp lex morname>
(6578) ENG_SR_TS:<amp lex morname>
(6579) ENG_stand:<amp lex morname>
(6580) ENG_stick_to:<amp lex morname>
(6581) ENG_stone:<amp lex morname>
(6582) ENG_straight:<amp lex morname>
(6583) ENG_talk/word:<amp lex morname>
(6584) ENG_teach:<amp lex morname>
(6585) ENG_tear:<amp lex morname>
(6586) ENG_tell:<amp lex morname>
(6587) ENG_that:<amp lex morname>
(6588) ENG_their_dual:<amp lex morname>
(6589) ENG_their_plural:<amp lex morname>
(6590) ENG_theirselves_dual:<amp lex morname>
(6591) ENG_theirselves_plural:<amp lex morname>
(6592) ENG_they:<amp lex morname>
(6593) ENG_thigh:<amp lex morname>
(6594) ENG_thing:<amp lex morname>
(6595) ENG_this:<amp lex morname>
(6596) ENG_thought:<amp lex morname>
(6597) ENG_throw:<amp lex morname>
(6598) ENG_TO:<amp lex morname>
(6599) ENG_true:<amp lex morname>
(6600) ENG_TS:<amp lex morname>
(6601) ENG_turn_back_on:<amp lex morname>
(6602) ENG_untie:<amp lex morname>
(6603) ENG_V_not:<amp lex morname>

(6604) ENG_vAux_S1d:<amp lex morname>
(6605) ENG_vAux_S1p:<amp lex morname>
(6606) ENG_vAux_S1s:<amp lex morname>
(6607) ENG_vAux_S3d:<amp lex morname>
(6608) ENG_vAux_S3p:<amp lex morname>
(6609) ENG_vAux_S3s:<amp lex morname>
(6610) ENG_vAux_Tf_S2d:<amp lex morname>
(6611) ENG_vAux_Tf_S2p:<amp lex morname>
(6612) ENG_vAux_Tf_S2s:<amp lex morname>
(6613) ENG_vAux_Tnf_S2d:<amp lex morname>
(6614) ENG_vAux_Tnf_S2p:<amp lex morname>
(6615) ENG_vAux_Tnf_S2s:<amp lex morname>
(6616) ENG_vAuxR:<amp lex morname>
(6617) ENG_very:<amp lex morname>
(6618) ENG_village:<amp lex morname>
(6619) ENG_walk:<amp lex morname>
(6620) ENG_watch:<amp lex morname>
(6621) ENG_watch_over:<amp lex morname>
(6622) ENG_we:<amp lex morname>
(6623) ENG_well:<amp lex morname>
(6624) ENG_what:<amp lex morname>
(6625) ENG_wild:<amp lex morname>
(6626) ENG_with:<amp lex morname>
(6627) ENG_with_dual:<amp lex morname>
(6628) ENG_without:<amp lex morname>
(6629) ENG_woman:<amp lex morname>
(6630) ENG_work:<amp lex morname>
(6631) ENG_write:<amp lex morname>
(6632) ENG_yell_to/shout:<amp lex morname>
(6633) ENG_you_plural:<amp lex morname>
(6634) ENG_you_sing:<amp lex morname>
(6635) ENG_your_dual:<amp lex morname>
(6636) ENG_your_plural:<amp lex morname>
(6637) ENG_your_sing:<amp lex morname>
(6638) ENG_yourselves_dual:<amp lex morname>
(6639) ENG_yourselves_plural:<amp lex morname>

A3.4 Batch File to Execute DATR

A3.4.1 Description

The batch file listed below is a standard DOS batch file. It contains two statements. The first calls the ZDATR program *zdatrtok*. *Zdatrtok* produces a token file from the Ogea.dtr file and names it Ogea.dtr.tok. The second program called is *zdatrinf* which is the inference program. This program uses both the Ogea.qry file and the Ogea.dtr.tok file as inputs, and then outputs the results of the query to a file named Ogea.dtr.trc. Ogea.dtr.trc is the Ogea trace file. The trace file is cleaned up by the Perl script DToAMPLE.pl, which outputs the file Ogea.db, which is directly runnable by AMPLE.

A3.4.2 File Listing

Filename: go.bat

Note: line numbers added for reference only. Not in actual file.

```
(6640)      zdatrtok -o Ogea.dtr.tok Ogea.dtr
(6641)      zdatrinf -j -u -V -1 -T Ogea.dtr.trc -Q Ogea.qry Ogea.dtr.tok
(6642)      perl DToAMPLE.pl
```


A3.5 The Ogea DATR Trace File

A3.5.1 Description

The result of running the batch file described above is a trace file called *Ogea.dtr.trc*. Each line in the query file *Ogea.qry* results in a corresponding line in the trace file. Note that although the trace file entries are close to AMPLE format, two things must be done before they are usable by AMPLE. First, empty lines and empty fields must be removed. Second, the trailing period must be stripped off. This is handled by a Perl script listed in Appendix Five.

A3.5.2 File Listing

Filename: ogea.dtr.trc

Note: line numbers added for reference only. Not in actual file.

```
(6643)      \g able
(6644)      \ge able
(6645)      \type r
(6646)      \a edo
(6647)      \u edo
(6648)      \mp
(6649)      \o
(6650)      \c Adv
(6651)      \e
(6652)      \mcc
(6653)      \f
(6654)      \loc
(6655)      \_no
(6656)      \co
(6657)      .
```

(6658) \g allow/leave
 (6659) \ge allow/leave
 (6660) \type r
 (6661) \a awa
 (6662) \a aw / _ [V]
 (6663) \u awa
 (6664) \mp MC1
 (6665) \o
 (6666) \c VR
 (6667) \e
 (6668) \mcc
 (6669) \f
 (6670) \loc
 (6671) _no
 (6672) \co
 (6673)
 (6674) .
 (6675) \g always
 (6676) \ge always
 (6677) \type r
 (6678) \a gaigai
 (6679) \u gaigai
 (6680) \mp
 (6681) \o
 (6682) \c Adv
 (6683) \e
 (6684) \mcc
 (6685) \f
 (6686) \loc
 (6687) _no
 (6688) \co
 (6689)
 (6690) .
 (6691) \g animal
 (6692) \ge animal
 (6693) \type r
 (6694) \a kaafa
 (6695) \u kaafa
 (6696) \mp
 (6697) \o
 (6698) \c NW
 (6699) \e
 (6700) \mcc
 (6701) \f
 (6702) \loc
 (6703) _no
 (6704) \co
 (6705)
 (6706) .
 (6707) \g appear/arrive
 (6708) \ge appear/arrive
 (6709) \type r
 (6710) \a heige
 (6711) \a heig / _ [V]
 (6712) \u heige
 (6713) \mp MC1
 (6714) \o

(6715) \c VR
 (6716) \e
 (6717) \mcc
 (6718) \f
 (6719) \loc
 (6720) _no
 (6721) \co
 (6722)
 (6723) .
 (6724) \g ask
 (6725) \ge ask
 (6726) \type r
 (6727) \a isoki
 (6728) \u isoki
 (6729) \mp MC2
 (6730) \o
 (6731) \c VR
 (6732) \e
 (6733) \mcc
 (6734) \f
 (6735) \loc
 (6736) _no
 (6737) \co
 (6738)
 (6739) .
 (6740) \g at
 (6741) \ge at
 (6742) \type r
 (6743) \a la
 (6744) \u la
 (6745) \mp
 (6746) \o
 (6747) \c Prep
 (6748) \e
 (6749) \mcc
 (6750) \f
 (6751) \loc
 (6752) _no
 (6753) \co
 (6754)
 (6755) .
 (6756) \g Attrib
 (6757) \ge Attributive marker
 (6758) \type s
 (6759) \a ou / [C] _
 (6760) \a wou / [V] _
 (6761) \u ou
 (6762) \mp
 (6763) \o 110
 (6764) \c Adj/Atr
 (6765) \e
 (6766) \mcc
 (6767) \f
 (6768) \loc
 (6769) _no
 (6770) \co
 (6771)

(6772) .
 (6773) \g B1d
 (6774) \ge for us (dual)
 (6775) \type s
 (6776) \a hari
 (6777) \a har / _ [V]
 (6778) \u hari
 (6779) \mp MC1
 (6780) \o 60
 (6781) \c VR/VR
 (6782) \e
 (6783) \mcc
 (6784) \f
 (6785) \loc
 (6786) _no
 (6787) \co
 (6788)
 (6789) .
 (6790) \g B1p
 (6791) \ge for us (plural)
 (6792) \type s
 (6793) \a hagi
 (6794) \a hag / _ [V]
 (6795) \u hagi
 (6796) \mp MC1
 (6797) \o 60
 (6798) \c VR/VR
 (6799) \e
 (6800) \mcc
 (6801) \f
 (6802) \loc
 (6803) _no
 (6804) \co
 (6805)
 (6806) .
 (6807) \g B1s
 (6808) \ge for me
 (6809) \type s
 (6810) \a hai
 (6811) \a hay / _ [V]
 (6812) \u hai
 (6813) \mp MC2
 (6814) \o 60
 (6815) \c VR/VR
 (6816) \e
 (6817) \mcc
 (6818) \f
 (6819) \loc
 (6820) _no
 (6821) \co
 (6822)
 (6823) .
 (6824) \g B2d
 (6825) \ge for you (dual)
 (6826) \type s
 (6827) \a tari
 (6828) \a tar / _ [V]

(6829) \u tari
 (6830) \mp MC1
 (6831) \o 60
 (6832) \c VR/VR
 (6833) \e
 (6834) \mcc
 (6835) \f
 (6836) \loc
 (6837) _no
 (6838) \co
 (6839)
 (6840) .
 (6841) \g B2p
 (6842) \ge for you (plural)
 (6843) \type s
 (6844) \a tagi
 (6845) \a tag / _ [V]
 (6846) \u tagi
 (6847) \mp MC1
 (6848) \o 60
 (6849) \c VR/VR
 (6850) \e
 (6851) \mcc
 (6852) \f
 (6853) \loc
 (6854) _no
 (6855) \co
 (6856)
 (6857) .
 (6858) \g B2s
 (6859) \ge for you (singular)
 (6860) \type s
 (6861) \a hani
 (6862) \a han / _ [V]
 (6863) \u hani
 (6864) \mp MC1
 (6865) \o 60
 (6866) \c VR/VR
 (6867) \e
 (6868) \mcc
 (6869) \f
 (6870) \loc
 (6871) _no
 (6872) \co
 (6873)
 (6874) .
 (6875) \g B3d
 (6876) \ge for them (dual)
 (6877) \type s
 (6878) \a nari
 (6879) \a nar / _ [V]
 (6880) \u nari
 (6881) \mp MC1
 (6882) \o 60
 (6883) \c VR/VR
 (6884) \e
 (6885) \mcc

(6886) \f
 (6887) \loc
 (6888) _no
 (6889) \co
 (6890)
 (6891) .
 (6892) \g B3p
 (6893) \ge for them (plural)
 (6894) \type s
 (6895) \a nagi
 (6896) \a nag / _ [V]
 (6897) \u nagi
 (6898) \mp MC1
 (6899) \o 60
 (6900) \c VR/VR
 (6901) \e
 (6902) \mcc
 (6903) \f
 (6904) \loc
 (6905) _no
 (6906) \co
 (6907)
 (6908) .
 (6909) \g B3s
 (6910) \ge for him/her/it
 (6911) \type s
 (6912) \a fu
 (6913) \u fu
 (6914) \mp MC2
 (6915) \o 60
 (6916) \c VR/VR
 (6917) \e
 (6918) \mcc
 (6919) \f
 (6920) \loc
 (6921) _no
 (6922) \co
 (6923)
 (6924) .
 (6925) \g bad
 (6926) \ge bad
 (6927) \type r
 (6928) \a inyaba
 (6929) \u inyaba
 (6930) \mp
 (6931) \o
 (6932) \c Adj
 (6933) \e
 (6934) \mcc
 (6935) \f
 (6936) \loc
 (6937) _no
 (6938) \co
 (6939)
 (6940) .
 (6941) \g be
 (6942) \ge be

```

(6943) \type r
(6944) \a inyi
(6945) \a iny / _ [V]
(6946) \u inyi
(6947) \mp MC1
(6948) \o
(6949) \c VR
(6950) \e
(6951) \mcc
(6952) \f
(6953) \loc
(6954) \_no
(6955) \co
(6956)
(6957) .
(6958) \g become.bad
(6959) \ge become.bad
(6960) \type r
(6961) \a inyaba +/ ~_ TO
(6962) \u inyaba_2
(6963) \mp MC2
(6964) \o
(6965) \c VR
(6966) \e
(6967) \mcc
(6968) \f
(6969) \loc
(6970) \_no
(6971) \co
(6972)
(6973) .
(6974) \g become.crooked
(6975) \ge become.crooked
(6976) \type r
(6977) \a kobibi +/ ~_ TO
(6978) \u kobibi_2
(6979) \mp MC2
(6980) \o
(6981) \c VR
(6982) \e
(6983) \mcc
(6984) \f
(6985) \loc
(6986) \_no
(6987) \co
(6988)
(6989) .
(6990) \g become.good
(6991) \ge become.good
(6992) \type r
(6993) \a hilou +/ ~_ TO
(6994) \a hiloo +/ ~_ TO
(6995) \u hilou_2
(6996) \mp MC2
(6997) \o
(6998) \c VR
(6999) \e

```

```

(7000) \mcc
(7001) \f
(7002) \loc
(7003) \_no
(7004) \co
(7005)
(7006) .
(7007) \g become.heavy
(7008) \ge become.heavy
(7009) \type r
(7010) \a ou +/ ~_ TO
(7011) \u ou_2
(7012) \mp MC2
(7013) \o
(7014) \c VR
(7015) \e
(7016) \mcc
(7017) \f
(7018) \loc
(7019) \_no
(7020) \co
(7021)
(7022) .
(7023) \g become.large
(7024) \ge become.large
(7025) \type r
(7026) \a anyakaro +/ ~_ TO
(7027) \u anyakaro_2
(7028) \mp MC2
(7029) \o
(7030) \c VR
(7031) \e
(7032) \mcc
(7033) \f
(7034) \loc
(7035) \_no
(7036) \co
(7037)
(7038) .
(7039) \g become.many
(7040) \ge become.many
(7041) \type r
(7042) \a baingaro +/ ~_ TO
(7043) \u baingaro_2
(7044) \mp MC2
(7045) \o
(7046) \c VR
(7047) \e
(7048) \mcc
(7049) \f
(7050) \loc
(7051) \_no
(7052) \co
(7053)
(7054) .
(7055) \g become.nothing
(7056) \ge become.nothing

```



```

(7057) \type r
(7058) \a uuwa +/ ~_ TO
(7059) \u uuwa_2
(7060) \mp MC2
(7061) \o
(7062) \c VR
(7063) \e
(7064) \mcc
(7065) \f
(7066) \loc
(7067) \_no
(7068) \co
(7069)
(7070) .
(7071) \g become.red
(7072) \ge become.red
(7073) \type r
(7074) \a giiri +/ ~_ TO
(7075) \u giiri_2
(7076) \mp MC2
(7077) \o
(7078) \c VR
(7079) \e
(7080) \mcc
(7081) \f
(7082) \loc
(7083) \_no
(7084) \co
(7085)
(7086) .
(7087) \g become.short
(7088) \ge become.short
(7089) \type r
(7090) \a tutu +/ ~_ TO
(7091) \u tutu_2
(7092) \mp MC2
(7093) \o
(7094) \c VR
(7095) \e
(7096) \mcc
(7097) \f
(7098) \loc
(7099) \_no
(7100) \co
(7101)
(7102) .
(7103) \g become.small
(7104) \ge become.small
(7105) \type r
(7106) \a menekele +/ ~_ TO
(7107) \a melekene +/ ~_ TO
(7108) \u menekele_2
(7109) \mp MC2
(7110) \o
(7111) \c VR
(7112) \e
(7113) \mcc

```

```

(7114) \f
(7115) \loc
(7116) \_no
(7117) \co
(7118)
(7119) .
(7120) \g become.straight
(7121) \ge become.straight
(7122) \type r
(7123) \a tigini +/ ~_ TO
(7124) \u tigini_2
(7125) \mp MC2
(7126) \o
(7127) \c VR
(7128) \e
(7129) \mcc
(7130) \f
(7131) \loc
(7132) \_no
(7133) \co
(7134)
(7135) .
(7136) \g become.true
(7137) \ge become.true
(7138) \type r
(7139) \a ngalenga +/ ~_ TO
(7140) \u ngalenga_2
(7141) \mp MC2
(7142) \o
(7143) \c VR
(7144) \e
(7145) \mcc
(7146) \f
(7147) \loc
(7148) \_no
(7149) \co
(7150)
(7151) .
(7152) \g become.wild
(7153) \ge become.wild
(7154) \type r
(7155) \a kui +/ ~_ TO
(7156) \u kui_2
(7157) \mp MC2
(7158) \o
(7159) \c VR
(7160) \e
(7161) \mcc
(7162) \f
(7163) \loc
(7164) \_no
(7165) \co
(7166)
(7167) .
(7168) \g bound
(7169) \ge bound
(7170) \type r

```

(7171) \a dage
 (7172) \a dag / _ [V]
 (7173) \u dage
 (7174) \mp MC1
 (7175) \o
 (7176) \c VR
 (7177) \e
 (7178) \mcc
 (7179) \f
 (7180) \loc
 (7181) _no
 (7182) \co
 (7183)
 (7184) .
 (7185) \g break.in.two
 (7186) \ge break.in.two
 (7187) \type r
 (7188) \a kou
 (7189) \a ko
 (7190) \u kou
 (7191) \mp MC2 RC1
 (7192) \o
 (7193) \c VR
 (7194) \e
 (7195) \mcc
 (7196) \f
 (7197) \loc
 (7198) _no
 (7199) \co
 (7200)
 (7201) .
 (7202) \g causative
 (7203) \ge causative
 (7204) \type s
 (7205) \a mu
 (7206) \u mu
 (7207) \mp
 (7208) \o 10
 (7209) \c VR/VR
 (7210) \e
 (7211) \mcc
 (7212) \f
 (7213) \loc
 (7214) _no
 (7215) \co
 (7216)
 (7217) .
 (7218) \g certainty
 (7219) \ge certainty
 (7220) \type s
 (7221) \a nga
 (7222) \u nga
 (7223) \mp
 (7224) \o 110
 (7225) \c VSjs/Vsjs
 (7226) \e
 (7227) \mcc

(7228) \f
 (7229) \loc
 (7230) _no
 (7231) \co
 (7232)
 (7233) .
 (7234) \g chase
 (7235) \ge chase
 (7236) \type r
 (7237) \a soo
 (7238) \u soo
 (7239) \mp MC2
 (7240) \o
 (7241) \c VR
 (7242) \e
 (7243) \mcc
 (7244) \f
 (7245) \loc
 (7246) _no
 (7247) \co
 (7248)
 (7249) .
 (7250) \g cloth
 (7251) \ge cloth
 (7252) \type r
 (7253) \a ogola
 (7254) \u ogola
 (7255) \mp
 (7256) \o
 (7257) \c NW
 (7258) \e
 (7259) \mcc
 (7260) \f
 (7261) \loc
 (7262) _no
 (7263) \co
 (7264)
 (7265) .
 (7266) \g come.down
 (7267) \ge come.down
 (7268) \type r
 (7269) \a me
 (7270) \a m / _ [V]
 (7271) \u me
 (7272) \mp MC1
 (7273) \o
 (7274) \c VR
 (7275) \e
 (7276) \mcc
 (7277) \f
 (7278) \loc
 (7279) _no
 (7280) \co
 (7281)
 (7282) .
 (7283) \g come.up
 (7284) \ge come.up

(7285) \type r
 (7286) \a mai
 (7287) \a may / _ [V]
 (7288) \u mai
 (7289) \mp MC2
 (7290) \o
 (7291) \c VR
 (7292) \e
 (7293) \mcc
 (7294) \f
 (7295) \loc
 (7296) _no
 (7297) \co
 (7298)
 (7299) .
 (7300) \g completely
 (7301) \ge completely
 (7302) \type s
 (7303) \a boro
 (7304) \u boro
 (7305) \mp
 (7306) \o
 (7307) \c VR/VR
 (7308) \e
 (7309) \mcc
 (7310) \f
 (7311) \loc
 (7312) _no
 (7313) \co
 (7314)
 (7315) .
 (7316) \g connect/join
 (7317) \ge connect/join
 (7318) \type r
 (7319) \a bagumu
 (7320) \u bagu-mu
 (7321) \mp MC2
 (7322) \o
 (7323) \c VR
 (7324) \e
 (7325) \mcc
 (7326) \f
 (7327) \loc
 (7328) _no
 (7329) \co
 (7330)
 (7331) .
 (7332) \g consume
 (7333) \ge consume
 (7334) \type r
 (7335) \a nyi
 (7336) \a ny / _ [V]
 (7337) \u nyi
 (7338) \mp MC1
 (7339) \o
 (7340) \c VR
 (7341) \e

```

(7342) \mcc
(7343) \f
(7344) \loc
(7345) \_no
(7346) \co
(7347)
(7348) .
(7349) \g Contra
(7350) \ge Contrafactual
(7351) \type s
(7352) \a ge +/ [Contra.Subj] _
(7353) \u ge
(7354) \mp
(7355) \o 100
(7356) \c Contra/Contra
(7357) \e
(7358) \mcc
(7359) \f
(7360) \loc
(7361) \_no
(7362) \co
(7363)
(7364) .
(7365) \g Contra.S1d
(7366) \ge Contractual first dual subject
(7367) \type s
(7368) \a edere / [C] _ +/ _ Contra
(7369) \a wedere / [V] _ +/ _ Contra
(7370) \u edere
(7371) \mp
(7372) \o 80
(7373) \c VR/Contra SMV/Contra
(7374) \e
(7375) \mcc
(7376) \f
(7377) \loc
(7378) \_no
(7379) \co
(7380)
(7381) .
(7382) \g Contra.S1p
(7383) \ge Contractual first plural subject
(7384) \type s
(7385) \a ede / [C] _ +/ _ Contra
(7386) \a wede / [V] _ +/ _ Contra
(7387) \u ede
(7388) \mp
(7389) \o 80
(7390) \c VR/Contra SMV/Contra
(7391) \e
(7392) \mcc
(7393) \f
(7394) \loc
(7395) \_no
(7396) \co
(7397)
(7398) .

```

(7399) \g Contra.S1s
 (7400) \ge Contractual first singular subject
 (7401) \type s
 (7402) \a e / [C] _ +/ _ Contra
 (7403) \a we / [V] _ +/ _ Contra
 (7404) \u e
 (7405) \mp
 (7406) \o 80
 (7407) \c VR/Contra SMV/Contra
 (7408) \e
 (7409) \mcc
 (7410) \f
 (7411) \loc
 (7412) _no
 (7413) \co
 (7414)
 (7415) .
 (7416) \g Contra.S2/3s/d/p
 (7417) \ge Contractual second or third singular, dual,
 (7418) or plural subject
 (7419) \type s
 (7420) \a a / [C] _ +/ _ Contra
 (7421) \a wa / [V] _ +/ _ Contra
 (7422) \u a
 (7423) \mp
 (7424) \o 80
 (7425) \c VR/Contra SMV/Contra
 (7426) \e
 (7427) \mcc
 (7428) \f
 (7429) \loc
 (7430) _no
 (7431) \co
 (7432)
 (7433) .
 (7434) \g crooked
 (7435) \ge crooked
 (7436) \type r
 (7437) \a kobibi
 (7438) \u kobibi
 (7439) \mp
 (7440) \o
 (7441) \c Adj
 (7442) \e
 (7443) \mcc
 (7444) \f
 (7445) \loc
 (7446) _no
 (7447) \co
 (7448)
 (7449) .
 (7450) \g cut
 (7451) \ge cut
 (7452) \type r
 (7453) \a karu
 (7454) \a ka
 (7455) \u karu

(7456) \mp MC2 RC1
 (7457) \o
 (7458) \c VR
 (7459) \e
 (7460) \mcc
 (7461) \f
 (7462) \loc
 (7463) _no
 (7464) \co
 (7465)
 (7466) .
 (7467) \g deflect
 (7468) \ge deflect
 (7469) \type r
 (7470) \a ariri
 (7471) \u ariri
 (7472) \mp MC2
 (7473) \o
 (7474) \c VR
 (7475) \e
 (7476) \mcc
 (7477) \f
 (7478) \loc
 (7479) _no
 (7480) \co
 (7481)
 (7482) .
 (7483) \g desire
 (7484) \ge desire
 (7485) \type r
 (7486) \a gau
 (7487) \u gau
 (7488) \mp
 (7489) \o
 (7490) \c NW
 (7491) \e
 (7492) \mcc
 (7493) \f
 (7494) \loc
 (7495) _no
 (7496) \co
 (7497)
 (7498) .
 (7499) \g die
 (7500) \ge die
 (7501) \type r
 (7502) \a ume
 (7503) \a um / _ [V]
 (7504) \u ume
 (7505) \mp MC1
 (7506) \o
 (7507) \c VR
 (7508) \e
 (7509) \mcc
 (7510) \f
 (7511) \loc
 (7512) _no

(7513) \co
 (7514)
 (7515) .
 (7516) \g different
 (7517) \ge different
 (7518) \type r
 (7519) \a fere
 (7520) \u fere
 (7521) \mp
 (7522) \o
 (7523) \c Adj
 (7524) \e
 (7525) \mcc
 (7526) \f
 (7527) \loc
 (7528) _no
 (7529) \co
 (7530)
 (7531) .
 (7532) \g do.not
 (7533) \ge do.not
 (7534) \type r
 (7535) \a adai
 (7536) \u adai
 (7537) \mp
 (7538) \o
 (7539) \c Adv
 (7540) \e
 (7541) \mcc
 (7542) \f
 (7543) \loc
 (7544) _no
 (7545) \co
 (7546)
 (7547) .
 (7548) \g do.thus
 (7549) \ge do.thus
 (7550) \type r
 (7551) \a ono
 (7552) \u ono
 (7553) \mp MC2
 (7554) \o
 (7555) \c VR
 (7556) \e
 (7557) \mcc
 (7558) \f
 (7559) \loc
 (7560) _no
 (7561) \co
 (7562)
 (7563) .
 (7564) \g do.work
 (7565) \ge do.work
 (7566) \type r
 (7567) \a haruwe +/ ~_ TO
 (7568) \u haruwe_2
 (7569) \mp MC2

(7570) \o
(7571) \c VR
(7572) \e
(7573) \mcc
(7574) \f
(7575) \loc
(7576) _no
(7577) \co
(7578)
(7579) .
(7580) \g dream
(7581) \ge dream
(7582) \type r
(7583) \a wewe
(7584) \u wewe
(7585) \mp
(7586) \o
(7587) \c NW
(7588) \e
(7589) \mcc
(7590) \f
(7591) \loc
(7592) _no
(7593) \co
(7594)
(7595) .
(7596) \g encircle
(7597) \ge encircle
(7598) \type r
(7599) \a kati
(7600) \u kati
(7601) \mp MC2
(7602) \o
(7603) \c VR
(7604) \e
(7605) \mcc
(7606) \f
(7607) \loc
(7608) _no
(7609) \co
(7610)
(7611) .
(7612) \g face
(7613) \ge face
(7614) \type r
(7615) \a hogo
(7616) \u hogo
(7617) \mp
(7618) \o
(7619) \c PNR
(7620) \e
(7621) \mcc
(7622) \f
(7623) \loc
(7624) _no
(7625) \co
(7626)

(7627) .
 (7628) \g fear
 (7629) \ge fear
 (7630) \type r
 (7631) \a umuge
 (7632) \a umug / _ [V]
 (7633) \u umuge
 (7634) \mp MC1
 (7635) \o
 (7636) \c VR
 (7637) \e
 (7638) \mcc
 (7639) \f
 (7640) \loc
 (7641) _no
 (7642) \co
 (7643) .
 (7644) .
 (7645) \g fight
 (7646) \ge fight
 (7647) \type r
 (7648) \a wara
 (7649) \a war / _ [V]
 (7650) \u wara
 (7651) \mp MC1
 (7652) \o
 (7653) \c VR
 (7654) \e
 (7655) \mcc
 (7656) \f
 (7657) \loc
 (7658) _no
 (7659) \co
 (7660) .
 (7661) .
 (7662) \g fill
 (7663) \ge fill
 (7664) \type r
 (7665) \a bolo
 (7666) \u bolo
 (7667) \mp MC2
 (7668) \o
 (7669) \c VR
 (7670) \e
 (7671) \mcc
 (7672) \f
 (7673) \loc
 (7674) _no
 (7675) \co
 (7676) .
 (7677) .
 (7678) \g finish
 (7679) \ge finish
 (7680) \type r
 (7681) \a usu
 (7682) \u usu
 (7683) \mp MC2

(7684) \o
 (7685) \c VR
 (7686) \e
 (7687) \mcc
 (7688) \f
 (7689) \loc
 (7690) _no
 (7691) \co
 (7692)
 (7693) .
 (7694) \g fly
 (7695) \ge fly
 (7696) \type r
 (7697) \a fililai
 (7698) \u fililai
 (7699) \mp MC2
 (7700) \o
 (7701) \c VR
 (7702) \e
 (7703) \mcc
 (7704) \f
 (7705) \loc
 (7706) _no
 (7707) \co
 (7708)
 (7709) .
 (7710) \g follow
 (7711) \ge follow
 (7712) \type r
 (7713) \a oojo
 (7714) \u oojo
 (7715) \mp MC2
 (7716) \o
 (7717) \c VR
 (7718) \e
 (7719) \mcc
 (7720) \f
 (7721) \loc
 (7722) _no
 (7723) \co
 (7724)
 (7725) .
 (7726) \g gather
 (7727) \ge gather
 (7728) \type r
 (7729) \a elege
 (7730) \a eleg / _ [V]
 (7731) \u elege
 (7732) \mp MC1
 (7733) \o
 (7734) \c VR
 (7735) \e
 (7736) \mcc
 (7737) \f
 (7738) \loc
 (7739) _no
 (7740) \co

(7741)
 (7742) .
 (7743) \g get.O1d
 (7744) \ge get us (dual)
 (7745) \type r
 (7746) \a haire
 (7747) \a hair / _ [V]
 (7748) \u haire
 (7749) \mp MC1
 (7750) \o
 (7751) \c VR
 (7752) \e
 (7753) \mcc
 (7754) \f
 (7755) \loc
 (7756) _no
 (7757) \co
 (7758)
 (7759) .
 (7760) \g get.O1p
 (7761) \ge get us (plural)
 (7762) \type r
 (7763) \a haige
 (7764) \a haig / _ [V]
 (7765) \u haige
 (7766) \mp MC1
 (7767) \o
 (7768) \c VR
 (7769) \e
 (7770) \mcc
 (7771) \f
 (7772) \loc
 (7773) _no
 (7774) \co
 (7775)
 (7776) .
 (7777) \g get.O1s
 (7778) \ge get me
 (7779) \type r
 (7780) \a haife
 (7781) \a haif / _ [V]
 (7782) \u haife
 (7783) \mp MC1
 (7784) \o
 (7785) \c VR
 (7786) \e
 (7787) \mcc
 (7788) \f
 (7789) \loc
 (7790) _no
 (7791) \co
 (7792)
 (7793) .
 (7794) \g get.O2d
 (7795) \ge get you (dual)
 (7796) \type r
 (7797) \a hatire

(7798) \a hatir / _ [V]
 (7799) \u hatire
 (7800) \mp MC1
 (7801) \o
 (7802) \c VR
 (7803) \e
 (7804) \mcc
 (7805) \f
 (7806) \loc
 (7807) _no
 (7808) \co
 (7809)
 (7810) .
 (7811) \g get.02p
 (7812) \ge get you (plural)
 (7813) \type r
 (7814) \a hatige
 (7815) \a hatig / _ [V]
 (7816) \u hatige
 (7817) \mp MC1
 (7818) \o
 (7819) \c VR
 (7820) \e
 (7821) \mcc
 (7822) \f
 (7823) \loc
 (7824) _no
 (7825) \co
 (7826)
 (7827) .
 (7828) \g get.02s
 (7829) \ge get you (singular)
 (7830) \type r
 (7831) \a haine
 (7832) \a hain / _ [V]
 (7833) \u haine
 (7834) \mp MC1
 (7835) \o
 (7836) \c VR
 (7837) \e
 (7838) \mcc
 (7839) \f
 (7840) \loc
 (7841) _no
 (7842) \co
 (7843)
 (7844) .
 (7845) \g get.03d
 (7846) \ge get them (dual)
 (7847) \type r
 (7848) \a hanire
 (7849) \a hanir / _ [V]
 (7850) \u hanire
 (7851) \mp MC1
 (7852) \o
 (7853) \c VR
 (7854) \e

(7855) \mcc
 (7856) \f
 (7857) \loc
 (7858) _no
 (7859) \co
 (7860)
 (7861) .
 (7862) \g get.03p
 (7863) \ge get them (plural)
 (7864) \type r
 (7865) \a hanige
 (7866) \a hanig / _ [V]
 (7867) \u hanige
 (7868) \mp MC1
 (7869) \o
 (7870) \c VR
 (7871) \e
 (7872) \mcc
 (7873) \f
 (7874) \loc
 (7875) _no
 (7876) \co
 (7877)
 (7878) .
 (7879) \g get.03s
 (7880) \ge get him/her/it
 (7881) \type r
 (7882) \a hau
 (7883) \u hau
 (7884) \mp MC2
 (7885) \o
 (7886) \c VR
 (7887) \e
 (7888) \mcc
 (7889) \f
 (7890) \loc
 (7891) _no
 (7892) \co
 (7893)
 (7894) .
 (7895) \g get.up
 (7896) \ge get.up
 (7897) \type r
 (7898) \a jai
 (7899) \a jay / _ [V]
 (7900) \u jai
 (7901) \mp MC2
 (7902) \o
 (7903) \c VR
 (7904) \e
 (7905) \mcc
 (7906) \f
 (7907) \loc
 (7908) _no
 (7909) \co
 (7910)
 (7911) .

(7912) \g get/take
 (7913) \ge get/take
 (7914) \type r
 (7915) \a te
 (7916) \a t / _ [V]
 (7917) \u te
 (7918) \mp MC1
 (7919) \o
 (7920) \c VR
 (7921) \e
 (7922) \mcc
 (7923) \f
 (7924) \loc
 (7925) _no
 (7926) \co
 (7927)
 (7928) .
 (7929) \g give.Old
 (7930) \ge give us (dual)
 (7931) \type r
 (7932) \a hiri
 (7933) \a hir / _ [V]
 (7934) \u hiri
 (7935) \mp MC1
 (7936) \o
 (7937) \c VR
 (7938) \e
 (7939) \mcc
 (7940) \f
 (7941) \loc
 (7942) _no
 (7943) \co
 (7944)
 (7945) .
 (7946) \g give.Olp
 (7947) \ge give us (plural)
 (7948) \type r
 (7949) \a higi
 (7950) \a hig / _ [V]
 (7951) \u higi
 (7952) \mp MC1
 (7953) \o
 (7954) \c VR
 (7955) \e
 (7956) \mcc
 (7957) \f
 (7958) \loc
 (7959) _no
 (7960) \co
 (7961)
 (7962) .
 (7963) \g give.O1s
 (7964) \ge give me
 (7965) \type r
 (7966) \a hii
 (7967) \a hiy / _ [V]
 (7968) \u hii


```

(7969) \mp MC2
(7970) \o
(7971) \c VR
(7972) \e
(7973) \mcc
(7974) \f
(7975) \loc
(7976) \_no
(7977) \co
(7978)
(7979) .
(7980) \g give.02d
(7981) \ge give you (dual)
(7982) \type r
(7983) \a tiri
(7984) \a tir / _ [V]
(7985) \u tiri
(7986) \mp MC1
(7987) \o
(7988) \c VR
(7989) \e
(7990) \mcc
(7991) \f
(7992) \loc
(7993) \_no
(7994) \co
(7995)
(7996) .
(7997) \g give.02p
(7998) \ge give you (plural)
(7999) \type r
(8000) \a tigi
(8001) \a tig / _ [V]
(8002) \u tigi
(8003) \mp MC1
(8004) \o
(8005) \c VR
(8006) \e
(8007) \mcc
(8008) \f
(8009) \loc
(8010) \_no
(8011) \co
(8012)
(8013) .
(8014) \g give.02s
(8015) \ge give you (singular)
(8016) \type r
(8017) \a hini
(8018) \a hin / _ [V]
(8019) \u hini
(8020) \mp MC1
(8021) \o
(8022) \c VR
(8023) \e
(8024) \mcc
(8025) \f

```

(8026) \loc
 (8027) _no
 (8028) \co
 (8029)
 (8030) .
 (8031) \g give.03d
 (8032) \ge give them (dual)
 (8033) \type r
 (8034) \a niri
 (8035) \a nir / _ [V]
 (8036) \u niri
 (8037) \mp MC1
 (8038) \o
 (8039) \c VR
 (8040) \e
 (8041) \mcc
 (8042) \f
 (8043) \loc
 (8044) _no
 (8045) \co
 (8046)
 (8047) .
 (8048) \g give.03p
 (8049) \ge give them (plural)
 (8050) \type r
 (8051) \a nigi
 (8052) \a nig / _ [V]
 (8053) \u nigi
 (8054) \mp MC1
 (8055) \o
 (8056) \c VR
 (8057) \e
 (8058) \mcc
 (8059) \f
 (8060) \loc
 (8061) _no
 (8062) \co
 (8063)
 (8064) .
 (8065) \g give.03s
 (8066) \ge give him/her/it
 (8067) \type r
 (8068) \a tu
 (8069) \u tu
 (8070) \mp MC2
 (8071) \o
 (8072) \c VR
 (8073) \e
 (8074) \mcc
 (8075) \f
 (8076) \loc
 (8077) _no
 (8078) \co
 (8079)
 (8080) .
 (8081) \g go
 (8082) \ge go

(8083) \type r
 (8084) \a goi
 (8085) \a goy / _ [V]
 (8086) \u goi
 (8087) \mp MC2
 (8088) \o
 (8089) \c VR
 (8090) \e
 (8091) \mcc
 (8092) \f
 (8093) \loc
 (8094) _no
 (8095) \co
 (8096)
 (8097) .
 (8098) \g go.down
 (8099) \ge go.down
 (8100) \type r
 (8101) \a mini
 (8102) \a min / _ [V]
 (8103) \u mini
 (8104) \mp MC1
 (8105) \o
 (8106) \c VR
 (8107) \e
 (8108) \mcc
 (8109) \f
 (8110) \loc
 (8111) _no
 (8112) \co
 (8113)
 (8114) .
 (8115) \g good
 (8116) \ge good
 (8117) \type r
 (8118) \a hilou
 (8119) \a hiloo
 (8120) \u hilou
 (8121) \mp
 (8122) \o
 (8123) \c Adj
 (8124) \e
 (8125) \mcc
 (8126) \f
 (8127) \loc
 (8128) _no
 (8129) \co
 (8130)
 (8131) .
 (8132) \g Hab
 (8133) \ge Habitual
 (8134) \type s
 (8135) \a g
 (8136) \u g
 (8137) \mp
 (8138) \o 70
 (8139) \c VR/SFV VR/Nom

(8140) \e
 (8141) \mcc
 (8142) \f
 (8143) \loc
 (8144) _no
 (8145) \co
 (8146)
 (8147) .
 (8148) \g hand
 (8149) \ge hand
 (8150) \type r
 (8151) \a owo
 (8152) \u owo
 (8153) \mp
 (8154) \o
 (8155) \c PNR
 (8156) \e
 (8157) \mcc
 (8158) \f
 (8159) \loc
 (8160) _no
 (8161) \co
 (8162)
 (8163) .
 (8164) \g he/she/it
 (8165) \ge he/she/it
 (8166) \type r
 (8167) \a no
 (8168) \u no
 (8169) \mp
 (8170) \o
 (8171) \c PerPro
 (8172) \e
 (8173) \mcc
 (8174) \f
 (8175) \loc
 (8176) _no
 (8177) \co
 (8178)
 (8179) .
 (8180) \g hear/understand/obey
 (8181) \ge hear/understand/obey
 (8182) \type r
 (8183) \a isi
 (8184) \a is / _ [V]
 (8185) \u isi
 (8186) \mp MC1
 (8187) \o
 (8188) \c VR
 (8189) \e
 (8190) \mcc
 (8191) \f
 (8192) \loc
 (8193) _no
 (8194) \co
 (8195)
 (8196) .

(8197) \g heavy
 (8198) \ge heavy
 (8199) \type r
 (8200) \a ou
 (8201) \u ou
 (8202) \mp
 (8203) \o
 (8204) \c Adj
 (8205) \e
 (8206) \mcc
 (8207) \f
 (8208) \loc
 (8209) _no
 (8210) \co
 (8211)
 (8212) .
 (8213) \g help
 (8214) \ge help
 (8215) \type r
 (8216) \a hoyo
 (8217) \u hoyo
 (8218) \mp MC2
 (8219) \o
 (8220) \c VR
 (8221) \e
 (8222) \mcc
 (8223) \f
 (8224) \loc
 (8225) _no
 (8226) \co
 (8227)
 (8228) .
 (8229) \g himself/herself/itself
 (8230) \ge himself/herself/itself
 (8231) \type r
 (8232) \a nogo
 (8233) \u nogo
 (8234) \mp
 (8235) \o
 (8236) \c RflPro
 (8237) \e
 (8238) \mcc
 (8239) \f
 (8240) \loc
 (8241) _no
 (8242) \co
 (8243)
 (8244) .
 (8245) \g his/her/its
 (8246) \ge his/her/its
 (8247) \type r
 (8248) \a nomo
 (8249) \u nomo
 (8250) \mp
 (8251) \o
 (8252) \c PosPro
 (8253) \e

(8254) \mcc
 (8255) \f
 (8256) \loc
 (8257) _no
 (8258) \co
 (8259)
 (8260) .
 (8261) \g hit.0ld
 (8262) \ge hit us (dual)
 (8263) \type r
 (8264) \a harire
 (8265) \a harir / _ [V]
 (8266) \u harire
 (8267) \mp MC1
 (8268) \o
 (8269) \c VR
 (8270) \e
 (8271) \mcc
 (8272) \f
 (8273) \loc
 (8274) _no
 (8275) \co
 (8276)
 (8277) .
 (8278) \g hit.0lp
 (8279) \ge hit us (plural)
 (8280) \type r
 (8281) \a harige
 (8282) \a harig / _ [V]
 (8283) \u harige
 (8284) \mp MC1
 (8285) \o
 (8286) \c VR
 (8287) \e
 (8288) \mcc
 (8289) \f
 (8290) \loc
 (8291) _no
 (8292) \co
 (8293)
 (8294) .
 (8295) \g hit.0ls
 (8296) \ge hit me
 (8297) \type r
 (8298) \a yari
 (8299) \a yar / _ [V]
 (8300) \u yari
 (8301) \mp MC1
 (8302) \o
 (8303) \c VR
 (8304) \e
 (8305) \mcc
 (8306) \f
 (8307) \loc
 (8308) _no
 (8309) \co
 (8310)

(8311) .
 (8312) \g hit.02d
 (8313) \ge hit you (dual)
 (8314) \type r
 (8315) \a tarire
 (8316) \a tarir / _ [V]
 (8317) \u tarire
 (8318) \mp MC1
 (8319) \o
 (8320) \c VR
 (8321) \e
 (8322) \mcc
 (8323) \f
 (8324) \loc
 (8325) _no
 (8326) \co
 (8327) .
 (8328) .
 (8329) \g hit.02p
 (8330) \ge hit you (plural)
 (8331) \type r
 (8332) \a tarige
 (8333) \a tarig / _ [V]
 (8334) \u tarige
 (8335) \mp MC1
 (8336) \o
 (8337) \c VR
 (8338) \e
 (8339) \mcc
 (8340) \f
 (8341) \loc
 (8342) _no
 (8343) \co
 (8344) .
 (8345) .
 (8346) \g hit.02s
 (8347) \ge hit you (singular)
 (8348) \type r
 (8349) \a nari
 (8350) \a nar / _ [V]
 (8351) \u nari
 (8352) \mp MC1
 (8353) \o
 (8354) \c VR
 (8355) \e
 (8356) \mcc
 (8357) \f
 (8358) \loc
 (8359) _no
 (8360) \co
 (8361) .
 (8362) .
 (8363) \g hit.03d
 (8364) \ge hit them (dual)
 (8365) \type r
 (8366) \a narire
 (8367) \a narir / _ [V]

(8368) \u narire
 (8369) \mp MC1
 (8370) \o
 (8371) \c VR
 (8372) \e
 (8373) \mcc
 (8374) \f
 (8375) \loc
 (8376) _no
 (8377) \co
 (8378)
 (8379) .
 (8380) \g hit.O3p
 (8381) \ge hit them (plural)
 (8382) \type r
 (8383) \a narige
 (8384) \a narig / _ [V]
 (8385) \u narige
 (8386) \mp MC1
 (8387) \o
 (8388) \c VR
 (8389) \e
 (8390) \mcc
 (8391) \f
 (8392) \loc
 (8393) _no
 (8394) \co
 (8395)
 (8396) .
 (8397) \g hit.O3s
 (8398) \ge hit him/her/it
 (8399) \type r
 (8400) \a wari
 (8401) \a war / _ [V]
 (8402) \u wari
 (8403) \mp MC1
 (8404) \o
 (8405) \c VR
 (8406) \e
 (8407) \mcc
 (8408) \f
 (8409) \loc
 (8410) _no
 (8411) \co
 (8412)
 (8413) .
 (8414) \g I
 (8415) \ge I
 (8416) \type r
 (8417) \a ji
 (8418) \u ji
 (8419) \mp
 (8420) \o
 (8421) \c PerPro
 (8422) \e
 (8423) \mcc
 (8424) \f

(8425) \loc
 (8426) _no
 (8427) \co
 (8428)
 (8429) .
 (8430) \g idle
 (8431) \ge idle
 (8432) \type r
 (8433) \a ewe
 (8434) \u ewe
 (8435) \mp
 (8436) \o
 (8437) \c Adj
 (8438) \e
 (8439) \mcc
 (8440) \f
 (8441) \loc
 (8442) _no
 (8443) \co
 (8444)
 (8445) .
 (8446) \g Ind.S1d
 (8447) \ge Indicative first dual subject
 (8448) \type s
 (8449) \a 0 / o _ # +/ Ind.Tf _ ~SR
 (8450) \a dere / e _ +/ Ind.Tf _ SR
 (8451) \a re / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
 (8452) \a yare +/ [Ind.Tip.Tp] _
 (8453) \a yare / o _ +/ Ind.Tf _ ~SR
 (8454) \u yare
 (8455) \mp
 (8456) \o 90
 (8457) \c SFV/SFV SFV/SFVW SMV/SMV
 (8458) \e
 (8459) \mcc
 (8460) \f
 (8461) \loc
 (8462) _no
 (8463) \co
 (8464)
 (8465) .
 (8466) \g Ind.S1p
 (8467) \ge Indicative first plural subject
 (8468) \type s
 (8469) \a 0 / o _ # +/ Ind.Tf _ ~SR
 (8470) \a de / e _ +/ Ind.Tf _ SR
 (8471) \a ya +/ [Ind.Tip.Tp] _
 (8472) \a ya / o _ +/ Ind.Tf _ ~SR
 (8473) \a ye / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
 (8474) \u ya
 (8475) \mp
 (8476) \o 90
 (8477) \c SFV/SFV SFV/SFVW SMV/SMV
 (8478) \e
 (8479) \mcc
 (8480) \f
 (8481) \loc

```

(8482)  \_no
(8483)  \co
(8484)
(8485)  .
(8486)  \g Ind.S1s
(8487)  \ge Indicative first singular subject
(8488)  \type s
(8489)  \a he / e _ +/ Ind.Tf _ SR
(8490)  \a n / o _ # +/ Ind.Tf _ ~SR
(8491)  \a na +/ Ind.Tip _
(8492)  \a ne +/ [Ind.Tip.Tp] _ SR
(8493)  \a ne / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(8494)  \a ni +/ Ind.Tp _ ~SR +/ [Ind.Tip.Tp] _ SR
(8495)  \a ni / o _ +/ Ind.Tf _ ~SR
(8496)  \u ni
(8497)  \mp
(8498)  \o 90
(8499)  \c SFV/SFV SFV/SFVW SMV/SMV
(8500)  \e
(8501)  \mcc
(8502)  \f
(8503)  \loc
(8504)  \_no
(8505)  \co
(8506)
(8507)  .
(8508)  \g Ind.S2d
(8509)  \ge Indicative second dual subject
(8510)  \type s
(8511)  \a dere / a _ +/ Ind.Tf _ SR
(8512)  \a re / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(8513)  \a yare +/ [Ind.Tip.Tp] _
(8514)  \u yare
(8515)  \mp
(8516)  \o 90
(8517)  \c SFV/SFV SFV/SFVW SMV/SMV
(8518)  \e
(8519)  \mcc
(8520)  \f
(8521)  \loc
(8522)  \_no
(8523)  \co
(8524)
(8525)  .
(8526)  \g Ind.S2p
(8527)  \ge Indicative second plural subject
(8528)  \type s
(8529)  \a de / a _ +/ Ind.Tf _ SR
(8530)  \a ya +/ [Ind.Tip.Tp] _
(8531)  \a ye / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(8532)  \u ya
(8533)  \mp
(8534)  \o 90
(8535)  \c SFV/SFV SFV/SFVW SMV/SMV
(8536)  \e
(8537)  \mcc
(8538)  \f

```

```

(8539) \loc
(8540) \_no
(8541) \co
(8542)
(8543) .
(8544) \g Ind.S2s
(8545) \ge Indicative second singular subject
(8546) \type s
(8547) \a he / a _ +/ Ind.Tf _ SR
(8548) \a na +/ Ind.Tip _
(8549) \a ne +/ Ind.Tp _ SR
(8550) \a ne / e _ +/ [Ind.Tip.Tp] _ SR +/ Ind.Trp _
(8551) \a ni +/ Ind.Tp _ ~SR +/ [Ind.Tip.Tp] _ SR
(8552) \u ni
(8553) \mp
(8554) \o 90
(8555) \c SFV/SFV SFV/SFVW SMV/SMV
(8556) \e
(8557) \mcc
(8558) \f
(8559) \loc
(8560) \_no
(8561) \co
(8562)
(8563) .
(8564) \g Ind.S3d
(8565) \ge Indicative third dual subject
(8566) \type s
(8567) \a d / o _ # +/ Ind.Tf _ ~SR
(8568) \a dere +/ Ind.Tp _ ~SR
(8569) \a dere / o _ +/ Ind.Tf _ ~SR
(8570) \a re / a _ +/ Ind.Tip _ +/ Ind.Tp _ SR +/ Ind.Trp _
(8571) \a ro / o _ +/ Ind.Tf _ SR
(8572) \u dere
(8573) \mp
(8574) \o 90
(8575) \c SFV/SFV SFV/SFVW SMV/SMV
(8576) \e
(8577) \mcc
(8578) \f
(8579) \loc
(8580) \_no
(8581) \co
(8582)
(8583) .
(8584) \g Ind.S3p
(8585) \ge Indicative third plural subject
(8586) \type s
(8587) \a d / o _ # +/ Ind.Tf _ ~SR
(8588) \a de +/ Ind.Tp _ ~SR
(8589) \a de / o _ +/ Ind.Tf _
(8590) \a go / o _ +/ Ind.Tf _ SR
(8591) \a ne / a _ +/ Ind.Tip _ +/ Ind.Tp _ SR +/ Ind.Trp _
(8592) \u de
(8593) \mp
(8594) \o 90
(8595) \c SFV/SFV SFV/SFVW SMV/SMV

```

```

(8596) \e
(8597) \mcc
(8598) \f
(8599) \loc
(8600) \_no
(8601) \co
(8602)
(8603) .
(8604) \g Ind.S3s
(8605) \ge Indicative third singular subject
(8606) \type s
(8607) \a 0 / a _ +/ Ind.Tip _ +/ Ind.Tp _ SR
(8608) \a 0 / o _ +/ Ind.Tf _ SR
(8609) \a i / a _ +/ Ind.Trp _
(8610) \a n / o _ # +/ Ind.Tf _ ~SR
(8611) \a na +/ Ind.Tp _ ~SR
(8612) \a ta +/ Ind.Tip _ ~SR
(8613) \a na / o _ +/ Ind.Tf _ ~SR
(8614) \u na
(8615) \mp
(8616) \o 90
(8617) \c SFV/SFV SFV/SFVW SMV/SMV
(8618) \e
(8619) \mcc
(8620) \f
(8621) \loc
(8622) \_no
(8623) \co
(8624)
(8625) .
(8626) \g Ind.Tf
(8627) \ge Indicative future tense
(8628) \type s
(8629) \a e / [C] _ +/ _ [Ind.S1]
(8630) \a we / [V] _ +/ _ [Ind.S1]
(8631) \a a / [C] _ +/ _ [Ind.S2]
(8632) \a wa / [V] _ +/ _ [Ind.S2]
(8633) \a o / [C] _ +/ _ [Ind.S1.S3]
(8634) \a wo / [V] _ +/ _ [Ind.S1.S3]
(8635) \u o
(8636) \mp
(8637) \o 80
(8638) \c SMV/SMV VR/SMV VR/SFV SFV/SFV
(8639) \e
(8640) \mcc
(8641) \f
(8642) \loc
(8643) \_no
(8644) \co
(8645)
(8646) .
(8647) \g Ind.Tip
(8648) \ge Indicative immediate past tense
(8649) \type s
(8650) \a e / [C] _ +/ _ [Ind.S1.S2]
(8651) \a we / [V] _ +/ _ [Ind.S1.S2]
(8652) \a a / [C] _ +/ _ [Ind.S3]

```

(8653) \a wa / [V] _ +/ _ [Ind.S3]
 (8654) \a 0 +/ _ [Ind.S1.S2] +/ _ Ind.S3s
 (8655) \u a
 (8656) \mp
 (8657) \o 80
 (8658) \c SMV/SMV VR/SMV VR/SFV SFV/SFV
 (8659) \e
 (8660) \mcc
 (8661) \f
 (8662) \loc
 (8663) _no
 (8664) \co
 (8665)
 (8666) .
 (8667) \g Ind.Tp
 (8668) \ge Indicative present tense
 (8669) \type s
 (8670) \a 0 +/ _ [Ind.S1.S2.S3]
 (8671) \a e / [C] _ +/ _ [Ind.S1.S2]
 (8672) \a we / [V] _ +/ _ [Ind.S1.S2]
 (8673) \a a / [C] _ +/ _ [Ind.S3]
 (8674) \a wa / [V] _ +/ _ [Ind.S3]
 (8675) \u 0
 (8676) \mp
 (8677) \o 80
 (8678) \c SMV/SMV VR/SMV VR/SFV SFV/SFV
 (8679) \e
 (8680) \mcc
 (8681) \f
 (8682) \loc
 (8683) _no
 (8684) \co
 (8685)
 (8686) .
 (8687) \g Ind.Trp
 (8688) \ge Indicative remote past tense
 (8689) \type s
 (8690) \a e / [C] _ +/ _ [Ind.S1.S2]
 (8691) \a we / [V] _ +/ _ [Ind.S1.S2]
 (8692) \a a / [C] _ +/ _ [Ind.S3]
 (8693) \a wa / [V] _ +/ _ [Ind.S3]
 (8694) \u e
 (8695) \mp
 (8696) \o 80
 (8697) \c SMV/SMV VR/SMV VR/SFV SFV/SFV
 (8698) \e
 (8699) \mcc
 (8700) \f
 (8701) \loc
 (8702) _no
 (8703) \co
 (8704)
 (8705) .
 (8706) \g Inf
 (8707) \ge Infinitive
 (8708) \type s
 (8709) \a e / [C] _

(8710) \a we / [V] _
 (8711) \u e
 (8712) \mp
 (8713) \o 80
 (8714) \c VR/Inf
 (8715) \e
 (8716) \mcc
 (8717) \f
 (8718) \loc
 (8719) _no
 (8720) \co
 (8721)
 (8722) .
 (8723) \g Inf.Sd
 (8724) \ge Infinitive subject
 (8725) \type s
 (8726) \a ru
 (8727) \u ru
 (8728) \mp
 (8729) \o 90
 (8730) \c Inf/Inf
 (8731) \e
 (8732) \mcc
 (8733) \f
 (8734) \loc
 (8735) _no
 (8736) \co
 (8737)
 (8738) .
 (8739) \g Inf.Sp
 (8740) \ge Infinitive plural subject
 (8741) \type s
 (8742) \a gu
 (8743) \u gu
 (8744) \mp
 (8745) \o 90
 (8746) \c Inf/Inf
 (8747) \e
 (8748) \mcc
 (8749) \f
 (8750) \loc
 (8751) _no
 (8752) \co
 (8753)
 (8754) .
 (8755) \g Inf.Ss
 (8756) \ge Infinitive singular subject
 (8757) \type s
 (8758) \a i
 (8759) \u i
 (8760) \mp
 (8761) \o 90
 (8762) \c Inf/Inf
 (8763) \e
 (8764) \mcc
 (8765) \f
 (8766) \loc

(8767) _no
 (8768) \co
 (8769)
 (8770) .
 (8771) \g inside
 (8772) \ge inside
 (8773) \type r
 (8774) \a huwanya
 (8775) \u huwanya
 (8776) \mp
 (8777) \o
 (8778) \c PNR
 (8779) \e
 (8780) \mcc
 (8781) \f
 (8782) \loc
 (8783) _no
 (8784) \co
 (8785)
 (8786) .
 (8787) \g judge
 (8788) \ge judge
 (8789) \type r
 (8790) \a haawe
 (8791) \u haawe
 (8792) \mp MC2
 (8793) \o
 (8794) \c VR
 (8795) \e
 (8796) \mcc
 (8797) \f
 (8798) \loc
 (8799) _no
 (8800) \co
 (8801)
 (8802) .
 (8803) \g just/only
 (8804) \ge just/only
 (8805) \type s
 (8806) \a nga
 (8807) \u nga
 (8808) \mp
 (8809) \o
 (8810) \c NW/NW RflPro/RflPro PN/PN Adj/Adj PosPro/PosPro
 PerPro/PerPro Prep/Prep Deictic/Deictic VR/VR
 (8811) \e
 (8812) \mcc
 (8813) \f
 (8814) \loc
 (8815) _no
 (8816) \co
 (8817)
 (8818) .
 (8819) \g knife
 (8820) \ge knife
 (8821) \type r
 (8822) \a kesebu

(8823) \u kesebu
 (8824) \mp
 (8825) \o
 (8826) \c NW
 (8827) \e
 (8828) \mcc
 (8829) \f
 (8830) \loc
 (8831) _no
 (8832) \co
 (8833)
 (8834) .
 (8835) \g large
 (8836) \ge large
 (8837) \type r
 (8838) \a anyakaro
 (8839) \u anyakaro
 (8840) \mp
 (8841) \o
 (8842) \c Adj
 (8843) \e
 (8844) \mcc
 (8845) \f
 (8846) \loc
 (8847) _no
 (8848) \co
 (8849)
 (8850) .
 (8851) \g leg/foot
 (8852) \ge leg/foot
 (8853) \type r
 (8854) \a afe
 (8855) \u afe
 (8856) \mp
 (8857) \o
 (8858) \c PNR
 (8859) \e
 (8860) \mcc
 (8861) \f
 (8862) \loc
 (8863) _no
 (8864) \co
 (8865)
 (8866) .
 (8867) \g leg/foot.3sP
 (8868) \ge his/her/its leg/foot
 (8869) \type r
 (8870) \a afo
 (8871) \u afo
 (8872) \mp
 (8873) \o
 (8874) \c PNR
 (8875) \e
 (8876) \mcc
 (8877) \f
 (8878) \loc
 (8879) _no

(8880) \co
 (8881)
 (8882) .
 (8883) \g lie
 (8884) \ge lie
 (8885) \type r
 (8886) \a yaawa
 (8887) \u yaawa
 (8888) \mp MC2
 (8889) \o
 (8890) \c VR
 (8891) \e
 (8892) \mcc
 (8893) \f
 (8894) \loc
 (8895) _no
 (8896) \co
 (8897)
 (8898) .
 (8899) \g light
 (8900) \ge light
 (8901) \type r
 (8902) \a lala
 (8903) \u lala_1
 (8904) \mp
 (8905) \o
 (8906) \c NW
 (8907) \e
 (8908) \mcc
 (8909) \f
 (8910) \loc
 (8911) _no
 (8912) \co
 (8913)
 (8914) .
 (8915) \g like.that
 (8916) \ge like.that
 (8917) \type r
 (8918) \a onou
 (8919) \u onou
 (8920) \mp
 (8921) \o
 (8922) \c Adj
 (8923) \e
 (8924) \mcc
 (8925) \f
 (8926) \loc
 (8927) _no
 (8928) \co
 (8929)
 (8930) .
 (8931) \g like1
 (8932) \ge like1
 (8933) \type s
 (8934) \a fe
 (8935) \u fe
 (8936) \mp

```

(8937) \o 100
(8938) \c NW/NW Deictic/Deictic Adj/Adj
(8939) \e
(8940) \mcc
(8941) \f
(8942) \loc
(8943) \_no
(8944) \co
(8945)
(8946) .
(8947) \g like2
(8948) \ge like2
(8949) \type s
(8950) \a neu +/- this _
(8951) \u neu
(8952) \mp
(8953) \o 100
(8954) \c Deictic/Deictic
(8955) \e
(8956) \mcc
(8957) \f
(8958) \loc
(8959) \_no
(8960) \co
(8961)
(8962) .
(8963) \g look.for
(8964) \ge look.for
(8965) \type r
(8966) \a kuru
(8967) \u kuru
(8968) \mp MC2
(8969) \o
(8970) \c VR
(8971) \e
(8972) \mcc
(8973) \f
(8974) \loc
(8975) \_no
(8976) \co
(8977)
(8978) .
(8979) \g man
(8980) \ge man
(8981) \type r
(8982) \a fai
(8983) \u fai
(8984) \mp
(8985) \o
(8986) \c NW
(8987) \e
(8988) \mcc
(8989) \f
(8990) \loc
(8991) \_no
(8992) \co
(8993)

```

(8994) .
 (8995) \g many
 (8996) \ge many
 (8997) \type r
 (8998) \a baingaro
 (8999) \u baingaro
 (9000) \mp
 (9001) \o
 (9002) \c Adj
 (9003) \e
 (9004) \mcc
 (9005) \f
 (9006) \loc
 (9007) _no
 (9008) \co
 (9009)
 (9010) .
 (9011) \g my
 (9012) \ge my
 (9013) \type r
 (9014) \a yame
 (9015) \u yame
 (9016) \mp
 (9017) \o
 (9018) \c PosPro
 (9019) \e
 (9020) \mcc
 (9021) \f
 (9022) \loc
 (9023) _no
 (9024) \co
 (9025)
 (9026) .
 (9027) \g myself
 (9028) \ge myself
 (9029) \type r
 (9030) \a yage
 (9031) \u yage
 (9032) \mp
 (9033) \o
 (9034) \c RflPro
 (9035) \e
 (9036) \mcc
 (9037) \f
 (9038) \loc
 (9039) _no
 (9040) \co
 (9041)
 (9042) .
 (9043) \g N.not
 (9044) \ge not
 (9045) \type r
 (9046) \a uuwa
 (9047) \u uuwa_1
 (9048) \mp
 (9049) \o
 (9050) \c NomNeg

(9051) \e
 (9052) \mcc
 (9053) \f
 (9054) \loc
 (9055) _no
 (9056) \co
 (9057)
 (9058) .
 (9059) \g Nom
 (9060) \ge Nominalizer
 (9061) \type s
 (9062) \a ou / [C] _
 (9063) \a wou / [V] _
 (9064) \u ou
 (9065) \mp
 (9066) \o 110
 (9067) \c VR/Nom Nom/Nom
 (9068) \e
 (9069) \mcc
 (9070) \f
 (9071) \loc
 (9072) _no
 (9073) \co
 (9074)
 (9075) .
 (9076) \g nose
 (9077) \ge nose
 (9078) \type r
 (9079) \a mutu
 (9080) \u mutu
 (9081) \mp
 (9082) \o
 (9083) \c PNR
 (9084) \e
 (9085) \mcc
 (9086) \f
 (9087) \loc
 (9088) _no
 (9089) \co
 (9090)
 (9091) .
 (9092) \g Old
 (9093) \ge to us (dual)
 (9094) \type s
 (9095) \a hiri
 (9096) \a hir / _ [V]
 (9097) \u hiri
 (9098) \mp MC1
 (9099) \o 30
 (9100) \c VR/VR
 (9101) \e
 (9102) \mcc
 (9103) \f
 (9104) \loc
 (9105) _no
 (9106) \co
 (9107)

```

(9108)      .
(9109)      \g O1p
(9110)      \ge to us (plural)
(9111)      \type s
(9112)      \a highi
(9113)      \a hig / _ [V]
(9114)      \u highi
(9115)      \mp MC1
(9116)      \o 30
(9117)      \c VR/VR
(9118)      \e
(9119)      \mcc
(9120)      \f
(9121)      \loc
(9122)      \_no
(9123)      \co
(9124)
(9125)      .
(9126)      \g O1s
(9127)      \ge to me
(9128)      \type s
(9129)      \a hi
(9130)      \a hiy / _ [V]
(9131)      \u hii
(9132)      \mp MC2
(9133)      \o 30
(9134)      \c VR/VR
(9135)      \e
(9136)      \mcc
(9137)      \f
(9138)      \loc
(9139)      \_no
(9140)      \co
(9141)
(9142)      .
(9143)      \g O2d
(9144)      \ge to you (dual)
(9145)      \type s
(9146)      \a tiri
(9147)      \a tir / _ [V]
(9148)      \u tiri
(9149)      \mp MC1
(9150)      \o 30
(9151)      \c VR/VR
(9152)      \e
(9153)      \mcc
(9154)      \f
(9155)      \loc
(9156)      \_no
(9157)      \co
(9158)
(9159)      .
(9160)      \g O2p
(9161)      \ge to you (plural)
(9162)      \type s
(9163)      \a tigi
(9164)      \a tig / _ [V]

```

(9165) \u tigi
 (9166) \mp MC1
 (9167) \o 30
 (9168) \c VR/VR
 (9169) \e
 (9170) \mcc
 (9171) \f
 (9172) \loc
 (9173) _no
 (9174) \co
 (9175)
 (9176) .
 (9177) \g O2s
 (9178) \ge to you (singular)
 (9179) \type s
 (9180) \a hini
 (9181) \a hin / _ [V]
 (9182) \u hini
 (9183) \mp MC1
 (9184) \o 30
 (9185) \c VR/VR
 (9186) \e
 (9187) \mcc
 (9188) \f
 (9189) \loc
 (9190) _no
 (9191) \co
 (9192)
 (9193) .
 (9194) \g O3d
 (9195) \ge to them (dual)
 (9196) \type s
 (9197) \a niri
 (9198) \a nir / _ [V]
 (9199) \u niri
 (9200) \mp MC1
 (9201) \o 30
 (9202) \c VR/VR
 (9203) \e
 (9204) \mcc
 (9205) \f
 (9206) \loc
 (9207) _no
 (9208) \co
 (9209)
 (9210) .
 (9211) \g O3p
 (9212) \ge to them (plural)
 (9213) \type s
 (9214) \a nigi
 (9215) \a nig / _ [V]
 (9216) \u nigi
 (9217) \mp MC1
 (9218) \o 30
 (9219) \c VR/VR
 (9220) \e
 (9221) \mcc

(9222) \f
 (9223) \loc
 (9224) _no
 (9225) \co
 (9226)
 (9227) .
 (9228) \g O3s
 (9229) \ge to him/her/it
 (9230) \type s
 (9231) \a tu
 (9232) \u tu
 (9233) \mp MC2
 (9234) \o 30
 (9235) \c VR/VR
 (9236) \e
 (9237) \mcc
 (9238) \f
 (9239) \loc
 (9240) _no
 (9241) \co
 (9242)
 (9243) .
 (9244) \g ocean
 (9245) \ge ocean
 (9246) \type r
 (9247) \a hiiri
 (9248) \u hiiri
 (9249) \mp
 (9250) \o
 (9251) \c NW
 (9252) \e
 (9253) \mcc
 (9254) \f
 (9255) \loc
 (9256) _no
 (9257) \co
 (9258)
 (9259) .
 (9260) \g one/a
 (9261) \ge one/a
 (9262) \type r
 (9263) \a ete
 (9264) \u ete
 (9265) \mp
 (9266) \o
 (9267) \c Adj
 (9268) \e
 (9269) \mcc
 (9270) \f
 (9271) \loc
 (9272) _no
 (9273) \co
 (9274)
 (9275) .
 (9276) \g our.dual
 (9277) \ge our.dual
 (9278) \type r

(9279) \a heire
(9280) \u heire
(9281) \mp
(9282) \o
(9283) \c PosPro
(9284) \e
(9285) \mcc
(9286) \f
(9287) \loc
(9288) _no
(9289) \co
(9290)
(9291) .
(9292) \g our.plural
(9293) \ge our.plural
(9294) \type r
(9295) \a ebere
(9296) \u ebere
(9297) \mp
(9298) \o
(9299) \c PosPro
(9300) \e
(9301) \mcc
(9302) \f
(9303) \loc
(9304) _no
(9305) \co
(9306)
(9307) .
(9308) \g ourselves.dual
(9309) \ge ourselves.dual
(9310) \type r
(9311) \a hare
(9312) \u hare
(9313) \mp
(9314) \o
(9315) \c RflPro
(9316) \e
(9317) \mcc
(9318) \f
(9319) \loc
(9320) _no
(9321) \co
(9322)
(9323) .
(9324) \g ourselves.plural
(9325) \ge ourselves.plural
(9326) \type r
(9327) \a hage
(9328) \u hage
(9329) \mp
(9330) \o
(9331) \c RflPro
(9332) \e
(9333) \mcc
(9334) \f
(9335) \loc

(9336) _no
 (9337) \co
 (9338)
 (9339) .
 (9340) \g p1
 (9341) \ge plural
 (9342) \type s
 (9343) \a gana
 (9344) \u gana
 (9345) \mp
 (9346) \o 20
 (9347) \c VR/VR
 (9348) \e
 (9349) \mcc
 (9350) \f
 (9351) \loc
 (9352) _no
 (9353) \co
 (9354)
 (9355) .
 (9356) \g P1d
 (9357) \ge first dual possessive
 (9358) \type s
 (9359) \a re
 (9360) \u re
 (9361) \mp
 (9362) \o 10
 (9363) \c PNR/PN
 (9364) \e
 (9365) \mcc
 (9366) \f
 (9367) \loc
 (9368) _no
 (9369) \co
 (9370)
 (9371) .
 (9372) \g P1p
 (9373) \ge first plural possessive
 (9374) \type s
 (9375) \a ge
 (9376) \u ge
 (9377) \mp
 (9378) \o 10
 (9379) \c PNR/PN
 (9380) \e
 (9381) \mcc
 (9382) \f
 (9383) \loc
 (9384) _no
 (9385) \co
 (9386)
 (9387) .
 (9388) \g P1s
 (9389) \ge first singular possessive
 (9390) \type s
 (9391) \a fe
 (9392) \u fe

(9393) \mp
 (9394) \o 10
 (9395) \c PNR/PN
 (9396) \e
 (9397) \mcc
 (9398) \f
 (9399) \loc
 (9400) _no
 (9401) \co
 (9402)
 (9403) .
 (9404) \g p2
 (9405) \ge plural
 (9406) \type s
 (9407) \a ru
 (9408) \u ru
 (9409) \mp
 (9410) \o 20
 (9411) \c VR/VR
 (9412) \e
 (9413) \mcc
 (9414) \f
 (9415) \loc
 (9416) _no
 (9417) \co
 (9418)
 (9419) .
 (9420) \g P2d
 (9421) \ge second dual possessive
 (9422) \type s
 (9423) \a tere
 (9424) \u tere
 (9425) \mp
 (9426) \o 10
 (9427) \c PNR/PN
 (9428) \e
 (9429) \mcc
 (9430) \f
 (9431) \loc
 (9432) _no
 (9433) \co
 (9434)
 (9435) .
 (9436) \g P2p
 (9437) \ge second plural possessive
 (9438) \type s
 (9439) \a tege
 (9440) \u tege
 (9441) \mp
 (9442) \o 10
 (9443) \c PNR/PN
 (9444) \e
 (9445) \mcc
 (9446) \f
 (9447) \loc
 (9448) _no
 (9449) \co

(9450)
 (9451) .
 (9452) \g P2s
 (9453) \ge second singular possessive
 (9454) \type s
 (9455) \a ne
 (9456) \a te
 (9457) \u ne
 (9458) \mp
 (9459) \o 10
 (9460) \c PNR/PN
 (9461) \e
 (9462) \mcc
 (9463) \f
 (9464) \loc
 (9465) _no
 (9466) \co
 (9467)
 (9468) .
 (9469) \g P3d
 (9470) \ge third dual possessive
 (9471) \type s
 (9472) \a nere
 (9473) \u nere
 (9474) \mp
 (9475) \o 10
 (9476) \c PNR/PN
 (9477) \e
 (9478) \mcc
 (9479) \f
 (9480) \loc
 (9481) _no
 (9482) \co
 (9483)
 (9484) .
 (9485) \g P3p
 (9486) \ge third plural possessive
 (9487) \type s
 (9488) \a nege
 (9489) \u nege
 (9490) \mp
 (9491) \o 10
 (9492) \c PNR/PN
 (9493) \e
 (9494) \mcc
 (9495) \f
 (9496) \loc
 (9497) _no
 (9498) \co
 (9499)
 (9500) .
 (9501) \g P3s
 (9502) \ge third singular possessive
 (9503) \type s
 (9504) \a 0
 (9505) \a u
 (9506) \u 0

(9507) \mp
 (9508) \o 10
 (9509) \c PNR/PN
 (9510) \e
 (9511) \mcc
 (9512) \f
 (9513) \loc
 (9514) _no
 (9515) \co
 (9516)
 (9517) .
 (9518) \g perf
 (9519) \ge perfective
 (9520) \type s
 (9521) \a ha
 (9522) \u ha
 (9523) \mp
 (9524) \o 100
 (9525) \c SFV/SFVW
 (9526) \e
 (9527) \mcc
 (9528) \f
 (9529) \loc
 (9530) _no
 (9531) \co
 (9532)
 (9533) .
 (9534) \g pertaining.to
 (9535) \ge pertaining.to
 (9536) \type s
 (9537) \a ha
 (9538) \u ha
 (9539) \mp
 (9540) \o
 (9541) \c Adj/Adj NW/NW Deictic/Deictic PosPro/PosPro
 (9542) \e
 (9543) \mcc
 (9544) \f
 (9545) \loc
 (9546) _no
 (9547) \co
 (9548)
 (9549) .
 (9550) \g pig
 (9551) \ge pig
 (9552) \type r
 (9553) \a buuwa
 (9554) \u buuwa
 (9555) \mp
 (9556) \o
 (9557) \c NW
 (9558) \e
 (9559) \mcc
 (9560) \f
 (9561) \loc
 (9562) _no
 (9563) \co

(9564)
 (9565) .
 (9566) \g place
 (9567) \ge place
 (9568) \type r
 (9569) \a ha
 (9570) \a haumu
 (9571) \u ha
 (9572) \mp
 (9573) \o
 (9574) \c NW
 (9575) \e
 (9576) \mcc
 (9577) \f
 (9578) \loc
 (9579) _no
 (9580) \co
 (9581)
 (9582) .
 (9583) \g plant
 (9584) \ge plant
 (9585) \type r
 (9586) \a tau
 (9587) \a ta
 (9588) \u tau
 (9589) \mp MC2 RC1
 (9590) \o
 (9591) \c VR
 (9592) \e
 (9593) \mcc
 (9594) \f
 (9595) \loc
 (9596) _no
 (9597) \co
 (9598)
 (9599) .
 (9600) \g poke
 (9601) \ge poke
 (9602) \type r
 (9603) \a tu / _ [O]
 (9604) \a tum / _ [B]
 (9605) \a tun / _ [A]
 (9606) \a tung / _ [X]
 (9607) \u tuN
 (9608) \mp MC2
 (9609) \o
 (9610) \c VR
 (9611) \e
 (9612) \mcc
 (9613) \f
 (9614) \loc
 (9615) _no
 (9616) \co
 (9617)
 (9618) .
 (9619) \g pour.out
 (9620) \ge pour.out

```
(9621) \type r
(9622) \a feu
(9623) \u feu
(9624) \mp MC2
(9625) \o
(9626) \c VR
(9627) \e
(9628) \mcc
(9629) \f
(9630) \loc
(9631) \_no
(9632) \co
(9633)
(9634) .
(9635) \g Prediction
(9636) \ge Prediction
(9637) \type s
(9638) \a ga
(9639) \u ga
(9640) \mp
(9641) \o 100
(9642) \c VSjs/Vsjs
(9643) \e
(9644) \mcc
(9645) \f
(9646) \loc
(9647) \_no
(9648) \co
(9649)
(9650) .
(9651) \g prepare
(9652) \ge prepare
(9653) \type r
(9654) \a haiya
(9655) \u haiya
(9656) \mp MC2
(9657) \o
(9658) \c VR
(9659) \e
(9660) \mcc
(9661) \f
(9662) \loc
(9663) \_no
(9664) \co
(9665)
(9666) .
(9667) \g press
(9668) \ge press
(9669) \type r
(9670) \a ilei
(9671) \u ilei
(9672) \mp MC2
(9673) \o
(9674) \c VR
(9675) \e
(9676) \mcc
(9677) \f
```

(9678) \loc
 (9679) _no
 (9680) \co
 (9681)
 (9682) .
 (9683) \g put
 (9684) \ge put
 (9685) \type r
 (9686) \a tafa
 (9687) \a taf / _ [V]
 (9688) \u tafa
 (9689) \mp MC1
 (9690) \o
 (9691) \c VR
 (9692) \e
 (9693) \mcc
 (9694) \f
 (9695) \loc
 (9696) _no
 (9697) \co
 (9698)
 (9699) .
 (9700) \g quickly
 (9701) \ge quickly
 (9702) \type r
 (9703) \a fasa
 (9704) \u fasa
 (9705) \mp MC2
 (9706) \o
 (9707) \c VR
 (9708) \e
 (9709) \mcc
 (9710) \f
 (9711) \loc
 (9712) _no
 (9713) \co
 (9714)
 (9715) .
 (9716) \g quickly
 (9717) \ge quickly
 (9718) \type s
 (9719) \a ya
 (9720) \u ya
 (9721) \mp
 (9722) \o
 (9723) \c VR/VR
 (9724) \e
 (9725) \mcc
 (9726) \f
 (9727) \loc
 (9728) _no
 (9729) \co
 (9730)
 (9731) .
 (9732) \g red
 (9733) \ge red
 (9734) \type r

(9735) \a giiri
 (9736) \u giiri
 (9737) \mp
 (9738) \o
 (9739) \c Adj
 (9740) \e
 (9741) \mcc
 (9742) \f
 (9743) \loc
 (9744) _no
 (9745) \co
 (9746)
 (9747) .
 (9748) \g remove
 (9749) \ge remove
 (9750) \type r
 (9751) \a uulu
 (9752) \u uulu
 (9753) \mp MC2
 (9754) \o
 (9755) \c VR
 (9756) \e
 (9757) \mcc
 (9758) \f
 (9759) \loc
 (9760) _no
 (9761) \co
 (9762)
 (9763) .
 (9764) \g run.away
 (9765) \ge run.away
 (9766) \type r
 (9767) \a faga
 (9768) \u faga
 (9769) \mp MC2
 (9770) \o
 (9771) \c VR
 (9772) \e
 (9773) \mcc
 (9774) \f
 (9775) \loc
 (9776) _no
 (9777) \co
 (9778)
 (9779) .
 (9780) \g see.Old
 (9781) \ge see us (dual)
 (9782) \type r
 (9783) \a herire
 (9784) \a herir / _ [V]
 (9785) \u herire
 (9786) \mp MC1
 (9787) \o
 (9788) \c VR
 (9789) \e
 (9790) \mcc
 (9791) \f

(9792) \loc
 (9793) _no
 (9794) \co
 (9795)
 (9796) .
 (9797) \g see.01p
 (9798) \ge see us (plural)
 (9799) \type r
 (9800) \a herige
 (9801) \a herig / _ [V]
 (9802) \u herige
 (9803) \mp MC1
 (9804) \o
 (9805) \c VR
 (9806) \e
 (9807) \mcc
 (9808) \f
 (9809) \loc
 (9810) _no
 (9811) \co
 (9812)
 (9813) .
 (9814) \g see.01s
 (9815) \ge see me
 (9816) \type r
 (9817) \a yeriye
 (9818) \a yeriy / _ [V]
 (9819) \u yeriye
 (9820) \mp MC1
 (9821) \o
 (9822) \c VR
 (9823) \e
 (9824) \mcc
 (9825) \f
 (9826) \loc
 (9827) _no
 (9828) \co
 (9829)
 (9830) .
 (9831) \g see.02d
 (9832) \ge see you (dual)
 (9833) \type r
 (9834) \a terire
 (9835) \a terir / _ [V]
 (9836) \u terire
 (9837) \mp MC1
 (9838) \o
 (9839) \c VR
 (9840) \e
 (9841) \mcc
 (9842) \f
 (9843) \loc
 (9844) _no
 (9845) \co
 (9846)
 (9847) .
 (9848) \g see.02p

(9849) \ge see you (plural)
 (9850) \type r
 (9851) \a terige
 (9852) \a terig / _ [V]
 (9853) \u terige
 (9854) \mp MC1
 (9855) \o
 (9856) \c VR
 (9857) \e
 (9858) \mcc
 (9859) \f
 (9860) \loc
 (9861) _no
 (9862) \co
 (9863)
 (9864) .
 (9865) \g see.02s
 (9866) \ge see you (singular)
 (9867) \type r
 (9868) \a nerige
 (9869) \a neriy / _ [V]
 (9870) \u nerige
 (9871) \mp MC1
 (9872) \o
 (9873) \c VR
 (9874) \e
 (9875) \mcc
 (9876) \f
 (9877) \loc
 (9878) _no
 (9879) \co
 (9880)
 (9881) .
 (9882) \g see.03d
 (9883) \ge see them (dual)
 (9884) \type r
 (9885) \a nerire
 (9886) \a nerir / _ [V]
 (9887) \u nerire
 (9888) \mp MC1
 (9889) \o
 (9890) \c VR
 (9891) \e
 (9892) \mcc
 (9893) \f
 (9894) \loc
 (9895) _no
 (9896) \co
 (9897)
 (9898) .
 (9899) \g see.03p
 (9900) \ge see them (plural)
 (9901) \type r
 (9902) \a nerige
 (9903) \a nerig / _ [V]
 (9904) \u nerige
 (9905) \mp MC1

(9906) \o
 (9907) \c VR
 (9908) \e
 (9909) \mcc
 (9910) \f
 (9911) \loc
 (9912) _no
 (9913) \co
 (9914)
 (9915) .
 (9916) \g see.03s
 (9917) \ge see him/her/it
 (9918) \type r
 (9919) \a ure
 (9920) \a ur / _ [V]
 (9921) \u ure
 (9922) \mp MC1
 (9923) \o
 (9924) \c VR
 (9925) \e
 (9926) \mcc
 (9927) \f
 (9928) \loc
 (9929) _no
 (9930) \co
 (9931)
 (9932) .
 (9933) \g send
 (9934) \ge send
 (9935) \type r
 (9936) \a huuru
 (9937) \u huuru
 (9938) \mp MC2
 (9939) \o
 (9940) \c VR
 (9941) \e
 (9942) \mcc
 (9943) \f
 (9944) \loc
 (9945) _no
 (9946) \co
 (9947)
 (9948) .
 (9949) \g shine
 (9950) \ge shine
 (9951) \type r
 (9952) \a lala
 (9953) \u lala_2
 (9954) \mp MC2
 (9955) \o
 (9956) \c VR
 (9957) \e
 (9958) \mcc
 (9959) \f
 (9960) \loc
 (9961) _no
 (9962) \co

(9963)
 (9964) .
 (9965) \g short
 (9966) \ge short
 (9967) \type r
 (9968) \a tutu
 (9969) \u tutu
 (9970) \mp
 (9971) \o
 (9972) \c Adj
 (9973) \e
 (9974) \mcc
 (9975) \f
 (9976) \loc
 (9977) _no
 (9978) \co
 (9979)
 (9980) .
 (9981) \g shove
 (9982) \ge shove
 (9983) \type r
 (9984) \a nuu
 (9985) \u nuu
 (9986) \mp MC2
 (9987) \o
 (9988) \c VR
 (9989) \e
 (9990) \mcc
 (9991) \f
 (9992) \loc
 (9993) _no
 (9994) \co
 (9995)
 (9996) .
 (9997) \g show
 (9998) \ge show
 (9999) \type r
 (10000) \a abiti
 (10001) \u abiti
 (10002) \mp MC2
 (10003) \o
 (10004) \c VR
 (10005) \e
 (10006) \mcc
 (10007) \f
 (10008) \loc
 (10009) _no
 (10010) \co
 (10011)
 (10012) .
 (10013) \g sibling.same.sex.his/her/its
 (10014) \ge sibling.same.sex.his/her/its
 (10015) \type r
 (10016) \a bemu
 (10017) \u bai_2
 (10018) \mp
 (10019) \o

(10020) \c NW
 (10021) \e
 (10022) \mcc
 (10023) \f
 (10024) \loc
 (10025) _no
 (10026) \co
 (10027)
 (10028) .
 (10029) \g sibling.same.sex.possessive
 (10030) \ge sibling.same.sex.possessive
 (10031) \type r
 (10032) \a bai +/ ~_ P3s
 (10033) \u bai_2
 (10034) \mp
 (10035) \o
 (10036) \c PNR
 (10037) \e
 (10038) \mcc
 (10039) \f
 (10040) \loc
 (10041) _no
 (10042) \co
 (10043)
 (10044) .
 (10045) \g sibling.same.sex.referential
 (10046) \ge sibling.same.sex.referential
 (10047) \type r
 (10048) \a babo
 (10049) \u bai_2
 (10050) \mp
 (10051) \o
 (10052) \c NW
 (10053) \e
 (10054) \mcc
 (10055) \f
 (10056) \loc
 (10057) _no
 (10058) \co
 (10059)
 (10060) .
 (10061) \g sit
 (10062) \ge sit
 (10063) \type r
 (10064) \a yafa
 (10065) \a yaf / _ [V]
 (10066) \u yafa
 (10067) \mp MC1
 (10068) \o
 (10069) \c VR
 (10070) \e
 (10071) \mcc
 (10072) \f
 (10073) \loc
 (10074) _no
 (10075) \co
 (10076)

```

(10077) .
(10078) \g Sj.S1/3
(10079) \ge subjunctive first or third subject
(10080) \type s
(10081) \a o / [C] _
(10082) \a wo / [V] _
(10083) \u o
(10084) \mp
(10085) \o 80
(10086) \c VR/VSj
(10087) \e
(10088) \mcc
(10089) \f
(10090) \loc
(10091) \_no
(10092) \co
(10093) .
(10094) .
(10095) \g Sj.S2
(10096) \ge subjunctive second subject
(10097) \type s
(10098) \a a / [C] _
(10099) \a wa / [V] _
(10100) \u a
(10101) \mp
(10102) \o 80
(10103) \c VR/VSj
(10104) \e
(10105) \mcc
(10106) \f
(10107) \loc
(10108) \_no
(10109) \co
(10110) .
(10111) .
(10112) \g Sj.Sd
(10113) \ge subjunctive dual subject
(10114) \type s
(10115) \a ru
(10116) \u ru
(10117) \mp
(10118) \o 90
(10119) \c VSj/VSjS
(10120) \e
(10121) \mcc
(10122) \f
(10123) \loc
(10124) \_no
(10125) \co
(10126) .
(10127) .
(10128) \g Sj.Sp
(10129) \ge subjunctive plural subject
(10130) \type s
(10131) \a gu
(10132) \u gu
(10133) \mp

```

```

(10134) \o 90
(10135) \c VSj/VSjS
(10136) \e
(10137) \mcc
(10138) \f
(10139) \loc
(10140) \_no
(10141) \co
(10142)
(10143) .
(10144) \g Sj.Ss
(10145) \ge subjunctive singular subject
(10146) \type s
(10147) \a u
(10148) \u u
(10149) \mp
(10150) \o 90
(10151) \c VSj/VSjS
(10152) \e
(10153) \mcc
(10154) \f
(10155) \loc
(10156) \_no
(10157) \co
(10158)
(10159) .
(10160) \g sleep.pl
(10161) \ge sleep.pl
(10162) \type r
(10163) \a jiri
(10164) \u jiri
(10165) \mp MC2
(10166) \o
(10167) \c VR
(10168) \e
(10169) \mcc
(10170) \f
(10171) \loc
(10172) \_no
(10173) \co
(10174)
(10175) .
(10176) \g small
(10177) \ge small
(10178) \type r
(10179) \a menekele
(10180) \a melekene
(10181) \u menekele
(10182) \mp
(10183) \o
(10184) \c Adj
(10185) \e
(10186) \mcc
(10187) \f
(10188) \loc
(10189) \_no
(10190) \co

```

(10191)
 (10192) .
 (10193) \g smooth
 (10194) \ge smooth
 (10195) \type r
 (10196) \a fii
 (10197) \u fii
 (10198) \mp MC2
 (10199) \o
 (10200) \c VR
 (10201) \e
 (10202) \mcc
 (10203) \f
 (10204) \loc
 (10205) _no
 (10206) \co
 (10207)
 (10208) .
 (10209) \g some
 (10210) \ge some
 (10211) \type r
 (10212) \a ila
 (10213) \u ila
 (10214) \mp
 (10215) \o
 (10216) \c Adj
 (10217) \e
 (10218) \mcc
 (10219) \f
 (10220) \loc
 (10221) _no
 (10222) \co
 (10223)
 (10224) .
 (10225) \g speak
 (10226) \ge speak
 (10227) \type r
 (10228) \a le
 (10229) \u le
 (10230) \mp MC2
 (10231) \o
 (10232) \c VR
 (10233) \e
 (10234) \mcc
 (10235) \f
 (10236) \loc
 (10237) _no
 (10238) \co
 (10239)
 (10240) .
 (10241) \g SR
 (10242) \ge switch referent
 (10243) \type s
 (10244) \a nga
 (10245) \u nga
 (10246) \mp
 (10247) \o 110

(10248) \c SMV/SMVW Contra/SMVW
 (10249) \e
 (10250) \mcc
 (10251) \f
 (10252) \loc
 (10253) _no
 (10254) \co
 (10255)
 (10256) .
 (10257) \g SR.TO
 (10258) \ge switch referent with temporal overlap
 (10259) \type s
 (10260) \a g +/- _ [Ind.T] +/- _ [Contra.Subj]
 (10261) \u g
 (10262) \mp
 (10263) \o 70
 (10264) \c VR/SMV
 (10265) \e
 (10266) \mcc
 (10267) \f
 (10268) \loc
 (10269) _no
 (10270) \co
 (10271)
 (10272) .
 (10273) \g SR.TS
 (10274) \ge switch referent with temporal succession
 (10275) \type s
 (10276) \a 0 +/- _ [Ind.T] +/- _ [Contra.Subj]
 (10277) \u 0
 (10278) \mp
 (10279) \o 70
 (10280) \c VR/SMV
 (10281) \e
 (10282) \mcc
 (10283) \f
 (10284) \loc
 (10285) _no
 (10286) \co
 (10287)
 (10288) .
 (10289) \g stand
 (10290) \ge stand
 (10291) \type r
 (10292) \a oto
 (10293) \u oto
 (10294) \mp MC2
 (10295) \o
 (10296) \c VR
 (10297) \e
 (10298) \mcc
 (10299) \f
 (10300) \loc
 (10301) _no
 (10302) \co
 (10303)
 (10304) .

(10305) \g stick.to
 (10306) \ge stick.to
 (10307) \type r
 (10308) \a fada
 (10309) \u fada
 (10310) \mp MC2
 (10311) \o
 (10312) \c VR
 (10313) \e
 (10314) \mcc
 (10315) \f
 (10316) \loc
 (10317) _no
 (10318) \co
 (10319)
 (10320) .
 (10321) \g stone
 (10322) \ge stone
 (10323) \type r
 (10324) \a megebu
 (10325) \u megebu
 (10326) \mp
 (10327) \o
 (10328) \c NW
 (10329) \e
 (10330) \mcc
 (10331) \f
 (10332) \loc
 (10333) _no
 (10334) \co
 (10335)
 (10336) .
 (10337) \g straight
 (10338) \ge straight
 (10339) \type r
 (10340) \a tigini
 (10341) \u tigini
 (10342) \mp
 (10343) \o
 (10344) \c Adj
 (10345) \e
 (10346) \mcc
 (10347) \f
 (10348) \loc
 (10349) _no
 (10350) \co
 (10351)
 (10352) .
 (10353) \g talk/word
 (10354) \ge talk/word
 (10355) \type r
 (10356) \a beele
 (10357) \u beele
 (10358) \mp
 (10359) \o
 (10360) \c NW
 (10361) \e

(10362) \mcc
(10363) \f
(10364) \loc
(10365) _no
(10366) \co
(10367)
(10368) .
(10369) \g teach
(10370) \ge teach
(10371) \type r
(10372) \a agotete
(10373) \u agotete
(10374) \mp MC2
(10375) \o
(10376) \c VR
(10377) \e
(10378) \mcc
(10379) \f
(10380) \loc
(10381) _no
(10382) \co
(10383)
(10384) .
(10385) \g tear
(10386) \ge tear
(10387) \type r
(10388) \a korai
(10389) \u korai
(10390) \mp MC2
(10391) \o
(10392) \c VR
(10393) \e
(10394) \mcc
(10395) \f
(10396) \loc
(10397) _no
(10398) \co
(10399)
(10400) .
(10401) \g tell
(10402) \ge tell
(10403) \type r
(10404) \a hai
(10405) \u hai
(10406) \mp MC2
(10407) \o
(10408) \c VR
(10409) \e
(10410) \mcc
(10411) \f
(10412) \loc
(10413) _no
(10414) \co
(10415)
(10416) .
(10417) \g that
(10418) \ge that

(10419) \type r
 (10420) \a wa
 (10421) \u wa
 (10422) \mp
 (10423) \o
 (10424) \c Deictic
 (10425) \e
 (10426) \mcc
 (10427) \f
 (10428) \loc
 (10429) _no
 (10430) \co
 (10431)
 (10432) .
 (10433) \g their.dual
 (10434) \ge their.dual
 (10435) \type r
 (10436) \a neire
 (10437) \u neire
 (10438) \mp
 (10439) \o
 (10440) \c PosPro
 (10441) \e
 (10442) \mcc
 (10443) \f
 (10444) \loc
 (10445) _no
 (10446) \co
 (10447)
 (10448) .
 (10449) \g their.plural
 (10450) \ge their.plural
 (10451) \type r
 (10452) \a nebere
 (10453) \u nebere
 (10454) \mp
 (10455) \o
 (10456) \c PosPro
 (10457) \e
 (10458) \mcc
 (10459) \f
 (10460) \loc
 (10461) _no
 (10462) \co
 (10463)
 (10464) .
 (10465) \g theirselves.dual
 (10466) \ge theirselves.dual
 (10467) \type r
 (10468) \a nare
 (10469) \u nare
 (10470) \mp
 (10471) \o
 (10472) \c RflPro
 (10473) \e
 (10474) \mcc
 (10475) \f

(10476) \loc
 (10477) _no
 (10478) \co
 (10479)
 (10480) .
 (10481) \g theirselves.plural
 (10482) \ge theirselves.plural
 (10483) \type r
 (10484) \a nage
 (10485) \u nage
 (10486) \mp
 (10487) \o
 (10488) \c RflPro
 (10489) \e
 (10490) \mcc
 (10491) \f
 (10492) \loc
 (10493) _no
 (10494) \co
 (10495)
 (10496) .
 (10497) \g they
 (10498) \ge they
 (10499) \type r
 (10500) \a nere
 (10501) \u nere
 (10502) \mp
 (10503) \o
 (10504) \c PerPro
 (10505) \e
 (10506) \mcc
 (10507) \f
 (10508) \loc
 (10509) _no
 (10510) \co
 (10511)
 (10512) .
 (10513) \g thigh
 (10514) \ge thigh
 (10515) \type r
 (10516) \a gaigu
 (10517) \u gaigu
 (10518) \mp
 (10519) \o
 (10520) \c NW
 (10521) \e
 (10522) \mcc
 (10523) \f
 (10524) \loc
 (10525) _no
 (10526) \co
 (10527)
 (10528) .
 (10529) \g thing
 (10530) \ge thing
 (10531) \type r
 (10532) \a dada

(10533) \u dada
 (10534) \mp
 (10535) \o
 (10536) \c NW
 (10537) \e
 (10538) \mcc
 (10539) \f
 (10540) \loc
 (10541) _no
 (10542) \co
 (10543)
 (10544) .
 (10545) \g this
 (10546) \ge this
 (10547) \type r
 (10548) \a be
 (10549) \u be
 (10550) \mp
 (10551) \o
 (10552) \c Deictic
 (10553) \e
 (10554) \mcc
 (10555) \f
 (10556) \loc
 (10557) _no
 (10558) \co
 (10559)
 (10560) .
 (10561) \g thought
 (10562) \ge thought
 (10563) \type r
 (10564) \a nina
 (10565) \u nina
 (10566) \mp
 (10567) \o
 (10568) \c PNR
 (10569) \e
 (10570) \mcc
 (10571) \f
 (10572) \loc
 (10573) _no
 (10574) \co
 (10575)
 (10576) .
 (10577) \g throw
 (10578) \ge throw
 (10579) \type r
 (10580) \a aya
 (10581) \u aya
 (10582) \mp MC2
 (10583) \o
 (10584) \c VR
 (10585) \e
 (10586) \mcc
 (10587) \f
 (10588) \loc
 (10589) _no

```

(10590) \co
(10591)
(10592) .
(10593) \g TO
(10594) \ge temporal overlap
(10595) \type s
(10596) \a 0
(10597) \u 0
(10598) \mp
(10599) \o 70
(10600) \c VR/SMVW
(10601) \e
(10602) \mcc
(10603) \f
(10604) \loc
(10605) \_no
(10606) \co
(10607)
(10608) .
(10609) \g true
(10610) \ge true
(10611) \type r
(10612) \a ngalenga
(10613) \u ngalenga
(10614) \mp
(10615) \o
(10616) \c Adj
(10617) \e
(10618) \mcc
(10619) \f
(10620) \loc
(10621) \_no
(10622) \co
(10623)
(10624) .
(10625) \g TS
(10626) \ge temporal succession
(10627) \type s
(10628) \a du
(10629) \u du
(10630) \mp
(10631) \o 70
(10632) \c VR/SMVW
(10633) \e
(10634) \mcc
(10635) \f
(10636) \loc
(10637) \_no
(10638) \co
(10639)
(10640) .
(10641) \g turn.back.on
(10642) \ge turn.back.on
(10643) \type r
(10644) \a moge
(10645) \u moge
(10646) \mp MC2

```

(10647) \o
 (10648) \c VR
 (10649) \e
 (10650) \mcc
 (10651) \f
 (10652) \loc
 (10653) _no
 (10654) \co
 (10655)
 (10656) .
 (10657) \g untie
 (10658) \ge untie
 (10659) \type r
 (10660) \a fuutu
 (10661) \u fuutu
 (10662) \mp MC2
 (10663) \o
 (10664) \c VR
 (10665) \e
 (10666) \mcc
 (10667) \f
 (10668) \loc
 (10669) _no
 (10670) \co
 (10671)
 (10672) .
 (10673) \g V.not
 (10674) \ge not
 (10675) \type r
 (10676) \a ada
 (10677) \u ada
 (10678) \mp
 (10679) \o
 (10680) \c VNeg
 (10681) \e
 (10682) \mcc
 (10683) \f
 (10684) \loc
 (10685) _no
 (10686) \co
 (10687)
 (10688) .
 (10689) \g vAux.S1d
 (10690) \ge verbal auxiliary first dual subject
 (10691) \type s
 (10692) \a yare
 (10693) \u yare
 (10694) \mp
 (10695) \o 90
 (10696) \c vAuxR/vAuxW
 (10697) \e
 (10698) \mcc
 (10699) \f
 (10700) \loc
 (10701) _no
 (10702) \co
 (10703)


```

(10704) .
(10705) \g vAux.S1p
(10706) \ge verbal auxiliary first plural subject
(10707) \type s
(10708) \a ya
(10709) \u ya
(10710) \mp
(10711) \o 90
(10712) \c vAuxR/vAuxW
(10713) \e
(10714) \mcc
(10715) \f
(10716) \loc
(10717) \_no
(10718) \co
(10719) .
(10720) .
(10721) \g vAux.S1s
(10722) \ge verbal auxiliary first singular subject
(10723) \type s
(10724) \a ni
(10725) \u ni
(10726) \mp
(10727) \o 90
(10728) \c vAuxR/vAuxW
(10729) \e
(10730) \mcc
(10731) \f
(10732) \loc
(10733) \_no
(10734) \co
(10735) .
(10736) .
(10737) \g vAux.S3d
(10738) \ge verbal auxiliary third dual subject
(10739) \type s
(10740) \a dere
(10741) \u dere
(10742) \mp
(10743) \o 90
(10744) \c vAuxR/vAuxW
(10745) \e
(10746) \mcc
(10747) \f
(10748) \loc
(10749) \_no
(10750) \co
(10751) .
(10752) .
(10753) \g vAux.S3p
(10754) \ge verbal auxiliary third plural subject
(10755) \type s
(10756) \a de
(10757) \u de
(10758) \mp
(10759) \o 90
(10760) \c vAuxR/vAuxW

```

(10761) \e
 (10762) \mcc
 (10763) \f
 (10764) \loc
 (10765) _no
 (10766) \co
 (10767)
 (10768) .
 (10769) \g vAux.S3s
 (10770) \ge verbal auxiliary third singular subject
 (10771) \type s
 (10772) \a na
 (10773) \u na
 (10774) \mp
 (10775) \o 90
 (10776) \c vAuxR/vAuxW
 (10777) \e
 (10778) \mcc
 (10779) \f
 (10780) \loc
 (10781) _no
 (10782) \co
 (10783)
 (10784) .
 (10785) \g vAux.Tf.S2d
 (10786) \ge verbal auxiliary second dual subject
 (10787) \type s
 (10788) \a nadere
 (10789) \u nadere
 (10790) \mp
 (10791) \o 90
 (10792) \c vAuxR/vAuxW
 (10793) \e
 (10794) \mcc
 (10795) \f
 (10796) \loc
 (10797) _no
 (10798) \co
 (10799)
 (10800) .
 (10801) \g vAux.Tf.S2p
 (10802) \ge verbal auxiliary second plural subject
 (10803) \type s
 (10804) \a nade
 (10805) \u nade
 (10806) \mp
 (10807) \o 90
 (10808) \c vAuxR/vAuxW
 (10809) \e
 (10810) \mcc
 (10811) \f
 (10812) \loc
 (10813) _no
 (10814) \co
 (10815)
 (10816) .
 (10817) \g vAux.Tf.S2s

(10818) \ge verbal auxiliary second singular subject, future tense
 (10819) \type s
 (10820) \a nahe
 (10821) \u nahe
 (10822) \mp
 (10823) \o 90
 (10824) \c vAuxR/vAuxW
 (10825) \e
 (10826) \mcc
 (10827) \f
 (10828) \loc
 (10829) _no
 (10830) \co
 (10831)
 (10832) .
 (10833) \g vAux.Tnf.S2d
 (10834) \ge verbal auxiliary second dual subject, non-future tense
 (10835) \type s
 (10836) \a yare
 (10837) \u yare
 (10838) \mp
 (10839) \o 90
 (10840) \c vAuxR/vAuxW
 (10841) \e
 (10842) \mcc
 (10843) \f
 (10844) \loc
 (10845) _no
 (10846) \co
 (10847)
 (10848) .
 (10849) \g vAux.Tnf.S2p
 (10850) \ge verbal auxiliary second plural subject, non-future tense
 (10851) \type s
 (10852) \a ya
 (10853) \u ya
 (10854) \mp
 (10855) \o 90
 (10856) \c vAuxR/vAuxW
 (10857) \e
 (10858) \mcc
 (10859) \f
 (10860) \loc
 (10861) _no
 (10862) \co
 (10863)
 (10864) .
 (10865) \g vAux.Tnf.S2s
 (10866) \ge verbal auxiliary second singular subject, non-future
 (10867) \type s
 (10868) \a ne
 (10869) \u ne
 (10870) \mp
 (10871) \o 90
 (10872) \c vAuxR/vAuxW
 (10873) \e
 (10874) \mcc

(10875) \f
 (10876) \loc
 (10877) _no
 (10878) \co
 (10879)
 (10880) .
 (10881) \g vAuxR
 (10882) \ge verbal auxiliary root
 (10883) \type r
 (10884) \a bo
 (10885) \u bo
 (10886) \mp
 (10887) \o
 (10888) \c vAuxR
 (10889) \e
 (10890) \mcc
 (10891) \f
 (10892) \loc
 (10893) _no
 (10894) \co
 (10895)
 (10896) .
 (10897) \g very
 (10898) \ge very
 (10899) \type r
 (10900) \a bai
 (10901) \u bai
 (10902) \mp
 (10903) \o
 (10904) \c Adj
 (10905) \e
 (10906) \mcc
 (10907) \f
 (10908) \loc
 (10909) _no
 (10910) \co
 (10911)
 (10912) .
 (10913) \g village
 (10914) \ge village
 (10915) \type r
 (10916) \a matane
 (10917) \u matane
 (10918) \mp
 (10919) \o
 (10920) \c NW
 (10921) \e
 (10922) \mcc
 (10923) \f
 (10924) \loc
 (10925) _no
 (10926) \co
 (10927)
 (10928) .
 (10929) \g walk
 (10930) \ge walk
 (10931) \type r

(10932) \a golo
(10933) \u golo
(10934) \mp MC2
(10935) \o
(10936) \c VR
(10937) \e
(10938) \mcc
(10939) \f
(10940) \loc
(10941) _no
(10942) \co
(10943)
(10944) .
(10945) \g watch
(10946) \ge watch
(10947) \type r
(10948) \a fele
(10949) \u fele
(10950) \mp MC2
(10951) \o
(10952) \c VR
(10953) \e
(10954) \mcc
(10955) \f
(10956) \loc
(10957) _no
(10958) \co
(10959)
(10960) .
(10961) \g watch.over
(10962) \ge watch.over
(10963) \type r
(10964) \a oofa
(10965) \u oofa
(10966) \mp MC2
(10967) \o
(10968) \c VR
(10969) \e
(10970) \mcc
(10971) \f
(10972) \loc
(10973) _no
(10974) \co
(10975)
(10976) .
(10977) \g we
(10978) \ge we
(10979) \type r
(10980) \a ere
(10981) \u ere
(10982) \mp
(10983) \o
(10984) \c PerPro
(10985) \e
(10986) \mcc
(10987) \f
(10988) \loc

(10989) _no
 (10990) \co
 (10991)
 (10992) .
 (10993) \g well
 (10994) \ge well
 (10995) \type s
 (10996) \a de
 (10997) \u de
 (10998) \mp
 (10999) \o
 (11000) \c VR/VR
 (11001) \e
 (11002) \mcc
 (11003) \f
 (11004) \loc
 (11005) _no
 (11006) \co
 (11007)
 (11008) .
 (11009) \g what
 (11010) \ge what
 (11011) \type r
 (11012) \a taate
 (11013) \u taate
 (11014) \mp
 (11015) \o
 (11016) \c IP
 (11017) \e
 (11018) \mcc
 (11019) \f
 (11020) \loc
 (11021) _no
 (11022) \co
 (11023)
 (11024) .
 (11025) \g wild
 (11026) \ge wild
 (11027) \type r
 (11028) \a kui
 (11029) \u kui
 (11030) \mp
 (11031) \o
 (11032) \c Adj
 (11033) \e
 (11034) \mcc
 (11035) \f
 (11036) \loc
 (11037) _no
 (11038) \co
 (11039)
 (11040) .
 (11041) \g with
 (11042) \ge with
 (11043) \type r
 (11044) \a bagu +/- ~_ causative
 (11045) \u bagu

(11046) \mp
 (11047) \o
 (11048) \c Prep
 (11049) \e
 (11050) \mcc
 (11051) \f
 (11052) \loc
 (11053) _no
 (11054) \co
 (11055)
 (11056) .
 (11057) \g with.dual
 (11058) \ge with.dual
 (11059) \type r
 (11060) \a ngare
 (11061) \u ngare
 (11062) \mp
 (11063) \o
 (11064) \c Prep
 (11065) \e
 (11066) \mcc
 (11067) \f
 (11068) \loc
 (11069) _no
 (11070) \co
 (11071)
 (11072) .
 (11073) \g without
 (11074) \ge without
 (11075) \type r
 (11076) \a uwau
 (11077) \u uwau
 (11078) \mp
 (11079) \o
 (11080) \c Prep
 (11081) \e
 (11082) \mcc
 (11083) \f
 (11084) \loc
 (11085) _no
 (11086) \co
 (11087)
 (11088) .
 (11089) \g woman
 (11090) \ge woman
 (11091) \type r
 (11092) \a aita
 (11093) \u aita
 (11094) \mp
 (11095) \o
 (11096) \c NW
 (11097) \e
 (11098) \mcc
 (11099) \f
 (11100) \loc
 (11101) _no
 (11102) \co

(11103)
 (11104) .
 (11105) \g work
 (11106) \ge work
 (11107) \type r
 (11108) \a haruwe
 (11109) \u haruwe
 (11110) \mp
 (11111) \o
 (11112) \c NW
 (11113) \e
 (11114) \mcc
 (11115) \f
 (11116) \loc
 (11117) _no
 (11118) \co
 (11119)
 (11120) .
 (11121) \g write
 (11122) \ge write
 (11123) \type r
 (11124) \a yerege
 (11125) \a yereg / _ [V]
 (11126) \u yerege
 (11127) \mp MC1
 (11128) \o
 (11129) \c VR
 (11130) \e
 (11131) \mcc
 (11132) \f
 (11133) \loc
 (11134) _no
 (11135) \co
 (11136)
 (11137) .
 (11138) \g yell.to/shout
 (11139) \ge yell.to/shout
 (11140) \type r
 (11141) \a yaure
 (11142) \u yaure
 (11143) \mp MC2
 (11144) \o
 (11145) \c VR
 (11146) \e
 (11147) \mcc
 (11148) \f
 (11149) \loc
 (11150) _no
 (11151) \co
 (11152)
 (11153) .
 (11154) \g you.plural
 (11155) \ge you.plural
 (11156) \type r
 (11157) \a tere
 (11158) \u tere
 (11159) \mp

(11160) \o
 (11161) \c PerPro
 (11162) \e
 (11163) \mcc
 (11164) \f
 (11165) \loc
 (11166) _no
 (11167) \co
 (11168)
 (11169) .
 (11170) \g you.sing
 (11171) \ge you.sing
 (11172) \type r
 (11173) \a ne
 (11174) \u ne
 (11175) \mp
 (11176) \o
 (11177) \c PerPro
 (11178) \e
 (11179) \mcc
 (11180) \f
 (11181) \loc
 (11182) _no
 (11183) \co
 (11184)
 (11185) .
 (11186) \g your.dual
 (11187) \ge your.dual
 (11188) \type r
 (11189) \a teire
 (11190) \u teire
 (11191) \mp
 (11192) \o
 (11193) \c PosPro
 (11194) \e
 (11195) \mcc
 (11196) \f
 (11197) \loc
 (11198) _no
 (11199) \co
 (11200)
 (11201) .
 (11202) \g your.plural
 (11203) \ge your.plural
 (11204) \type r
 (11205) \a tebere
 (11206) \u tebere
 (11207) \mp
 (11208) \o
 (11209) \c PosPro
 (11210) \e
 (11211) \mcc
 (11212) \f
 (11213) \loc
 (11214) _no
 (11215) \co
 (11216)

```

(11217) .
(11218) \g your.sing
(11219) \ge your.sing
(11220) \type r
(11221) \a name
(11222) \u name
(11223) \mp
(11224) \o
(11225) \c PosPro
(11226) \e
(11227) \mcc
(11228) \f
(11229) \loc
(11230) \_no
(11231) \co
(11232) .
(11233) .
(11234) \g yourselves.dual
(11235) \ge yourselves.dual
(11236) \type r
(11237) \a tare
(11238) \u tare
(11239) \mp
(11240) \o
(11241) \c RflPro
(11242) \e
(11243) \mcc
(11244) \f
(11245) \loc
(11246) \_no
(11247) \co
(11248) .
(11249) .
(11250) \g yourselves.plural
(11251) \ge yourselves.plural
(11252) \type r
(11253) \a tage
(11254) \u tage
(11255) \mp
(11256) \o
(11257) \c RflPro
(11258) \e
(11259) \mcc
(11260) \f
(11261) \loc
(11262) \_no
(11263) \co
(11264) .
(11265) .

```

APPENDIX 4

THE YALÁLAG ZAPOTEC AMPLE AND DATR LEXICONS

A4.1 Overview

This appendix provides listings of the following: A subset of the Yalálag Zapotec AMPLE root database file, the Yalálag Zapotec DATR "theory" file (the lexicon), the Yalálag Zapotec DATR query file, a DOS batch file that invokes DATR, and the result of running DATR using the query file--a query trace file.

For research purposes, all Order Class 10 root entries and some Order Class 9 entries were extracted from a Yalálag Zapotec AMPLE root database file supplied to this author by Dr. H. Andrew Black. The AMPLE database file shown in this appendix lists the extracted entries.

The Yalálag Zapotec DATR theory file is the heart of the Yalálag Zapotec DATR lexicon. It is divided into the following sections: function library, interfaces, templates, categories, and Yalálag Zapotec root nodes. The theory file was developed specifically to show that there are generalizations not captured in the AMPLE root file that can be captured in DATR, thereby reducing redundancy. The theory file includes an interface between DATR and AMPLE though this was not the primary focus of the DATR work done on Yalálag Zapotec.

The Yalálag Zapotec DATR query file is the driver for the interface between DATR and AMPLE. The order of the entries in the query file mirror the order of the entries in the Yalálag Zapotec AMPLE root database file.

The DOS batch file listing contains two calls to DATR. The first call creates a token file, the second executes the query file against the token file.

The Yalálag Zapotec DATR trace file is the output of running the DOS batch file.

The trace file provides the Yalálag Zapotec AMPLE lexical entries.

A4.2 The Yalálag Zapotec AMPLE Root Database File

A4.2.1 Description

A4.2.2 File Listing

Filename: yal0910.db

Note: line numbers added for reference only. Not in actual file.

```
(11266)    \r cha'a
(11267)    \_no 00129
(11268)    \u cha'
(11269)    \a cha' /~_ #
(11270)    \a cha'a / _ #
(11271)    \c VA
(11272)    \g estar.en.grupo
(11273)    \m 10
(11274)    \mp vi
(11275)    \mp vt
(11276)    \mp class2
(11277)    \mp takes_null_S
(11278)    \mcc estar.en.grupo / ~Pl.pfx _
(11279)
(11280)    \r chach
(11281)    \_no 00130
(11282)    \u cha:ch
(11283)    \a cha:ch
(11284)    \c VA
(11285)    \g aparecer.muchos
(11286)    \m 10
(11287)    \mp vi
(11288)    \mp class2
(11289)    \mp takes_null_S
(11290)    \co llach kwerp kello - effect of something that scares us
```

```

(11291) \r cha'ra'll
(11292) \_no 00138
(11293) \u cha'la'ch
(11294) \a cha'la'ch
(11295) \c VA
(11296) \g engordar
(11297) \m 10
(11298) \mp vi
(11299) \mp class2
(11300) \mp takes_null_S
(11301) \mp takes_:_Caus

(11302) \r chel
(11303) \_no 00158
(11304) \u che:l
(11305) \a che:l
(11306) \c VA
(11307) \g enrollar | take group 2 pns??
(11308) \m 9
(11309) \mp vi
(11310) \mp class2
(11311) \mp takes_null_S
(11312) \mp takes_:_Caus
(11313)
(11314) \r che'z
(11315) \_no 00170
(11316) \u che's
(11317) \a che's +/ [P-or-Modal] _
(11318) \a e's +/ H _
(11319) \a che's +/ [P,H-or-Modal] ~_
(11320) \c VA
(11321) \g romper
(11322) \m 9
(11323) \mp vi
(11324) \mp class2
(11325) \mp takes_:_Caus
(11326)
(11327) \r chia
(11328) \_no 00172
(11329) \u chia
(11330) \a chi / _ [V-hi+cg] +/ _ [1s-or-3r]
(11331) \a chia
(11332) \c VA
(11333) \g coser.vt
(11334) \m 9
(11335) \mp vt
(11336) \mp class2
(11337)
(11338) \r chi'o
(11339) \_no 00195
(11340) \u chi'o
(11341) \a chi' / _ o' +/ _ 2s
(11342) \a chi'kw / _ [V-hi+cg] +/ _ [1s-or-3r]
(11343) \a chi'o
(11344) \c VA
(11345) \g crecer
(11346) \m 9

```

```

(11347) \mp vi
(11348) \mp class2
(11349) \mp takes_:s_Caus
(11350)
(11351) \r labj
(11352) \_no 00495
(11353) \u lapj
(11354) \a lapj
(11355) \c VA
(11356) \g susmergir | group 2 pns??
(11357) \m 10
(11358) \mp vi
(11359) \mp class2
(11360) \mp takes_null_S
(11361)
(11362) \r la'll
(11363) \_no 00498
(11364) \u la'ch
(11365) \a la'ch
(11366) \c VA
(11367) \g vagar
(11368) \m 10
(11369) \mp vi
(11370) \mp class2
(11371) \mp takes_null_S
(11372)
(11373) \r lap
(11374) \_no 00503
(11375) \u la:p
(11376) \a la:p
(11377) \c VA
(11378) \g pizcar
(11379) \m 10
(11380) \mp vt
(11381) \mp class2
(11382) \mp takes_null_S
(11383)
(11384) \r len
(11385) \_no 00536
(11386) \u len
(11387) \a len
(11388) \c VA
(11389) \g asociar
(11390) \m 10
(11391) \mp vi
(11392) \mp class2
(11393) \mp takes_null_S
(11394) \mp takes_:_Caus
(11395)
(11396) \r nit
(11397) \_no 00658
(11398) \u ni:t
(11399) \a ni:t
(11400) \c VA
(11401) \g perder.vi
(11402) \m 10
(11403) \mp vi

```

```

(11404) \mp class2
(11405) \mp takes_null_S
(11406)
(11407) \r nna'lÄllej
(11408) \_no 00666
(11409) \u na':lchej
(11410) \a na':lchej
(11411) \c VA
(11412) \g sorprender
(11413) \m 10 |no Habitual
(11414) \mp vi
(11415) \mp class2
(11416) \mp takes_null_S
(11417)
(11418) \r nnao
(11419) \_no 00667
(11420) \u nao
(11421) \a na under_ao / _ o' +/ _ 2s
(11422) \a nakw / _ [V-hi+cg] +/ _ [1s-or-3r]
(11423) / _ [V-hi-cg]p | where 1s or 3r is followed by 3f or 3a
(11424) \a nao
(11425) \c VA
(11426) \g seguir
(11427) \m 10
(11428) \mp vt
(11429) \mp class2
(11430) \mp takes_null_S
(11431)
(11432) \r nne
(11433) \_no 00670
(11434) \u ne
(11435) \a ni / _ a' +/ _ 1s
(11436) \a n, / _ # +/ [Modal] ... _
(11437) +/ Rep ...
(11438) +/ [Imperatives] ... _
(11439) +/ Pl.pfx ... _
(11440) \a ne / # ([C]) _ #
(11441) / _ ~#
(11442) \c VA
(11443) \g decir2
(11444) \m 10
(11445) \mp vt
(11446) \mp class2
(11447) \mp takes_null_S
(11448)
(11449) \r nnebia'
(11450) \_no 00672
(11451) \u nepia'
(11452) \a nepi / _ [V-hi+cg] +/ _ [1s-or-3r]
(11453) / _ [V-hi-cg]p | where 1s or 3r is followed by 3f or 3a
(11454) \a nepia'
(11455) \c VA
(11456) \g gobernar
(11457) \m 10
(11458) \mp vt
(11459) \mp class2
(11460)

```

(11461) \r nni't
 (11462) _no 00675
 (11463) \u ni':t
 (11464) \a ni':t
 (11465) \c VA
 (11466) \g estar.regado | take group 2 pns?
 (11467) \m 10
 (11468) \mp vi
 (11469) \mp class2
 (11470) \mp takes_null_S
 (11471)
 (11472) \r nnitrao
 (11473) _no 00676
 (11474) \u ni:tlao
 (11475) \a ni:tla under_ao / _ o' +/ _ 2s
 (11476) \a ni:tlakw / _ [V-hi+cg] +/ _ [1s-or-3r]
 (11477) \a ni:tlao
 (11478) \c VA
 (11479) \g borrar.vi
 (11480) \m 10
 (11481) \mp vi
 (11482) \mp class2
 (11483) \mp takes_null_S
 (11484)
 (11485) \r sa'a
 (11486) _no 00752
 (11487) \u sa'
 (11488) \a s / _ a' +/ _ 1s
 (11489) \a se' / _ [V-hi-cg] +/ _ [1s-or-3r]
 (11490) \a sa'a / _ #
 (11491) \a sa' / ~_ #
 (11492) \c VA
 (11493) \g ir/salir/caminar | caminar formerly separate record
 (11494) \m 10
 (11495) \mp vi
 (11496) \mp class2
 (11497) \mp takes_null_S
 (11498) \mp laryngealizes
 (11499)
 (11500) \r sa'ra'll
 (11501) _no 00757
 (11502) \u sa'la'ch
 (11503) \a sa'la'ch
 (11504) \c VA
 (11505) \g sospechar
 (11506) \m 10
 (11507) \mp vi
 (11508) \mp class2
 (11509) \mp takes_null_S
 (11510) \co see also recordar2 wasa'ra'll
 (11511)
 (11512) \r se
 (11513) _no 00759
 (11514) \u se
 (11515) \a si / _ a' +/ _ 1s
 (11516) \a s, / _ # +/ [Modal] ... _
 (11517) +/ Rep ... _


```

(11518) +/ [Imperatives] ... _
(11519) +/ Plural.S ... _
(11520) \a s, / _ :tese| +/ [Modal] ... _
(11521) | +/ Rep ...
(11522) | +/ [Imperatives] ... _
(11523) | +/ Plural.S ... _
(11524) \a se / # ([C]) _ #
(11525) / _ ~#
(11526) \c VA
(11527) \g parar.vi1
(11528) \m 10
(11529) \mp vi
(11530) \mp class2
(11531) \mp takes_null_S
(11532)
(11533) \r sella
(11534) \_no 00761
(11535) \u secha
(11536) \a sech / _ a' +/ _ 1s
(11537) \a seche / _ e' +/ _ 3r
(11538) \a sech / _ #
(11539) \a secha / ~_ #
(11540) \a soch / _ a' +/ _ 1s
(11541) \a soche / _ e' +/ _ 3r
(11542) \a soch / _ #
(11543) \a socha / ~_ #
(11544) \c VA
(11545) \g parar.vi2 | Is -cha really part of root or is it an Adv?
    yes
(11546) \m 10
(11547) \mp vi
(11548) \mp class2
(11549) \mp takes_null_S
(11550) \noload
(11551)
(11552) \r sera'll
(11553) \_no 00764
(11554) \u sela'ch
(11555) \a sela'ch
(11556) \c VA
(11557) \g desear
(11558) \m 10
(11559) \mp vt
(11560) \mp class2
(11561) \mp takes_null_S
(11562)
(11563) \r sill
(11564) \_no 00773
(11565) \u sich
(11566) \a sich
(11567) \c VA
(11568) \g sonar
(11569) \m 10
(11570) \mp vi
(11571) \mp class2
(11572) \mp takes_null_S
(11573) \mp takes_:_Caus

```

A4.3 The Yalálag Zapotec DATR Theory File

A4.3.1 Description

The Yalálag Zapotec DATR theory file is divided into the following sections: function library, interfaces, templates, categories, and Yalálag Zapotec root nodes for Order Class 10 and a subset of Order Class 9.

The function library loads the standard DATR function library. The interfaces section provides a one-way interface from DATR to AMPLE. The templates section provides a series of abstract nodes to support multiple allomorph entries in a root node entry. The categories section provides abstract nodes that potentially represent grammatical and morphological categories. It is through these abstract nodes that redundancy in the Yalálag Zapotec AMPLE lexicon can be reduced for Order Class 9 and 10 root entries. The last section is Yalálag Zapotec root node section. This section contains a subset of Yalálag Zapotec roots from the Yalálag Zapotec AMPLE root database file--a subset of Order Class 9, and all of the Order Class 10 entries.

A4.3.2 File Listing

Filename: yal.dtr

Note: line numbers added for reference only. Not in actual file.

```

(11574) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11575) %
(11576) % File:          yal.dtr
(11577) % Purpose:       Partial lexicon - Yalalag
(11578) %                               Zapotec - Order Class 10 roots
(11579) %                               and some Order Class 9 roots
(11580) % Yalalag Zapotec:  A Language of Mexico
(11581) % Data from:     yalrt.db (from Dr. H. Andrew Black)
(11582) % Author:       Michael Colburn
(11583) % Email:        mcolburn@sprintmail.com
(11584) % Created:     November 12, 1999
(11585) % Last update:
(11586) % Version:      1.0
(11587) %
(11588) % Notes:         Limited entries -- for research and
(11589) %                               illustration only
(11590) %                               Illustrates reduction of redundancy,
(11591) %                               capture of generalizations
(11592) %
(11593) %
(11594) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11595)
(11596) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Function Library %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11597) %
(11598)
(11599) #load 'std.dtr' all.
(11600)
(11601) %
(11602) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11603)
(11604) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Interfaces %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11605) %
(11606)
(11607) I_AMPLE:
(11608) <amp lex morname> == <amp root>
(11609)                   <amp noload>
(11610)                   <amp underlying form>
(11611)                   "<lex form lex>"
(11612)                   <amp category>
(11613)                   <amp gloss morpheme name>
(11614)                   <amp order class>
(11615)                   <amp mor property>
(11616)                   <amp mcc>
(11617)                   <amp comment>'\n'
(11618) <amp allomorph> == \a' ' "<mor form>" ' ' "<mor prop>" ' '
(11619)                   "<mor sec>" ' ' "<mor mec>" '\n'
(11620) <amp category> == \c' ' "<lex cat>" '\n'
(11621) <amp comment> == \co' ' "<lex com>" '\n'
(11622) <amp gloss morpheme name> == \g' ' "<lex gloss eng morname>"
(11623)                   '\n'
(11624) <amp mor property> == \mp' ' "<lex prop>" '\n'
(11625) <amp noload> == \_no' ' "<lex noload>" '\n'
(11626) <amp order class> == \m' ' "<lex order class>" '\n'
(11627) <amp mcc> == \mcc' ' "<lex mcc>" '\n'
(11628) <amp root> == \r' ' "<lex name>" '\n'
(11629) <amp underlying form> == \u' ' "<lex under>" '\n'.

```

```

(11630) %
(11631) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11632)
(11633) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Templates %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11634) %
(11635)
(11636) ALLO1:
(11637) <lex form lex> == <lex form lex allo1>
(11638) <lex form lex allo1> == C_DLT.
(11639)
(11640) ALLO2:
(11641) <lex form lex> == <lex form lex allo2>
(11642) <lex form lex allo2> == C_DLT.
(11643)
(11644) ALLO3:
(11645) <lex form lex> == <lex form lex allo3>
(11646) <lex form lex allo3> == C_DLT.
(11647)
(11648) ALLO4:
(11649) <lex form lex> == <lex form lex allo4>
(11650) <lex form lex allo4> == C_DLT.
(11651)
(11652) ALLO5:
(11653) <lex form lex> == <lex form lex allo5>
(11654) <lex form lex allo5> == C_DLT.
(11655)
(11656) ALLO6:
(11657) <lex form lex> == <lex form lex allo6>
(11658) <lex form lex allo6> == C_DLT.
(11659)
(11660) ALLO7:
(11661) <lex form lex> == <lex form lex allo7>
(11662) <lex form lex allo7> == C_DLT.
(11663)
(11664) ALLOMORPH:
(11665) <mor> ==
(11666) <mor form default> == "<phon under>".
(11667)
(11668) DEFAULT:
(11669) <lex form lex> == <lex form lex default>
(11670) <lex form lex default> == C_DLT.
(11671)
(11672) % Default Lexical Template
(11673) C_DLT:
(11674) <lex form lex> == I_AMPLE:<amp allomorph>.
(11675)
(11676) %
(11677) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11678)
(11679) %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% CATEGORIES %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
(11680) %
(11681) %
(11682)
(11683) C_MORPHEME:
(11684) <> ==
(11685) <lex> ==
(11686) <amp lex> == I_AMPLE

```

```

(11687) <mor> == ALLOMORPH
(11688) <lex name> == "<phon under>"
(11689) <lex under> == "<phon under>"
(11690) <lex order class> ==
(11691) <lex form> == DEFAULT
(11692) <lex add prop> ==.
(11693)
(11694) C_INFIX:
(11695) <> == C_MORPHEME
(11696) <lex type> == i.
(11697)
(11698) C_PREFIX:
(11699) <> == C_MORPHEME
(11700) <lex type> == p.
(11701)
(11702) C_ROOT:
(11703) <> == C_MORPHEME
(11704) <lex type> == r.
(11705)
(11706) V09_10:
(11707) <> == C_ROOT
(11708) <lex cat> == 'VA'
(11709) <lex prop> == 'class2 ' "<lex add prop>".
(11710)
(11711) V09:
(11712) <> == V09_10
(11713) <lex order> == 9.
(11714)
(11715) V9i:
(11716) <> == V09
(11717) <lex prop> == 'vi ' V09:<lex prop>.
(11718)
(11719) V9t:
(11720) <> == V09
(11721) <lex prop> == 'vt ' V09:<lex prop>.
(11722)
(11723) V10:
(11724) <> == V09_10
(11725) <lex order> == 10
(11726) <lex prop> == V09_10:<lex prop> 'takes_null_S '.
(11727)
(11728) V10i:
(11729) <> == V10
(11730) <lex prop> == 'vi ' V10:<lex prop>.
(11731)
(11732) V10ic:
(11733) <> == V10
(11734) <lex prop> == V10i:<lex prop> 'takes::_Caus'.
(11735)
(11736) V10t:
(11737) <> == V10
(11738) <lex prop> == 'vt ' V10:<lex prop>.
(11739)
(11740) %%%%%%%%%%%%%%%%%%%%%%%%%% YALALAG ZAPOTEC %%%%%%%%%%%%%%%%%%%%%%%%%%
(11741)
(11742) R_chaa:
(11743) <> == V10I

```

```

(11744) <lex name> == 'cha\`a'
(11745) <lex noload> == '00129'
(11746) <phon under> == 'cha\`'
(11747) <lex form> == DEFAULT ALLO1
(11748) <mor form allo1> == 'cha\`a / _ #'
(11749) <mor sec default> == '/~_ #'
(11750) <lex gloss> == 'estar.en.grupo'
(11751) <lex add prop> == 'vt '
(11752) <lex mcc> == 'estar.en.grupo / ~Pl.pfx _'.
(11753)
(11754) R_chach:
(11755) <> == V10i
(11756) <lex name> == 'chach'
(11757) <lex noload> == '00130'
(11758) <phon under> == 'cha:ch'
(11759) <lex gloss> == 'aparecer.muchos'
(11760) <lex com> == 'llach kwerp kello - effect of something that
scares us'.
(11761)
(11762) R_charall:
(11763) <> == V10i
(11764) <lex name> == 'cha\`ra\`ll'
(11765) <lex noload> == '00138'
(11766) <phon under> == 'cha\`la\`ch'
(11767) <lex gloss> == 'engordar'
(11768) <lex add prop> == 'takes_:_Caus '.
(11769)
(11770) R_chel:
(11771) <> == V9i
(11772) <lex name> == 'chel'
(11773) <lex noload> == '00158'
(11774) <phon under> == 'che:l'
(11775) <lex gloss> == 'enrollar | take group 2 pns?? '
(11776) <lex add prop> == 'takes_null_S takes_:_Caus'.
(11777)
(11778) R_chez:
(11779) <> == V9i
(11780) <lex name> == 'che\`z'
(11781) <lex noload> == '00170'
(11782) <phon under> == 'che\`s'
(11783) <lex gloss> == 'romper'
(11784) <lex form> == ALLO1 ALLO2 DEFAULT
(11785) <mor form allo1> == 'che\`s +/ [P-or-Modal]
(11786) <mor form allo2> == 'e\`s +/ H
(11787) <mor mec default> == '+/ [P,H-or-Modal] ~_'
(11788) <lex add prop> == 'takes_:_Caus'.
(11789)
(11790) R_chia:
(11791) <> == V9t
(11792) <lex noload> == '00172'
(11793) <phon under> == 'chia'
(11794) <lex gloss> == 'coser.vt'
(11795) <lex form> == ALLO1 DEFAULT
(11796) <mor form allo1> == 'chi / _ [V-hi+cg] +/ _ [1s-or-3r]'.
(11797)
(11798) R_chio:
(11799) <> == V9i

```

```

(11800) <lex name> == 'chi\o'
(11801) <lex noload> == '00195'
(11802) <phon under> == 'chi\o'
(11803) <lex gloss> == 'crecer'
(11804) <lex form> == ALLO1 ALLO2 DEFAULT
(11805) <mor form allo1> == 'chi\' / _ o\' +/ _ 2s'
(11806) <mor form allo2> == 'chi\'kw / _ [V-hi+cg] +/ _ [1s-or-3r]'
(11807) <lex add prop> == 'takes_:s_Caus'.
(11808)
(11809) R_labj:
(11810) <> == V10i
(11811) <lex name> == 'labj'
(11812) <lex noload> == '00495'
(11813) <phon under> == 'lapj'
(11814) <lex gloss> == 'susmergir | group 2 pns??'.
(11815)
(11816) R_lall:
(11817) <> == V10i
(11818) <lex noload> == '00498'
(11819) <phon under> == 'la\'ll'
(11820) <lex gloss> == 'vagar'.
(11821)
(11822) R_lap:
(11823) <> == V10t
(11824) <lex name> == 'lap'
(11825) <lex noload> == '00503'
(11826) <phon under> == 'la:p'
(11827) <lex gloss> == 'pizcar'.
(11828)
(11829) R_len:
(11830) <> == V10i
(11831) <lex noload> == '00536'
(11832) <phon under> == 'len'
(11833) <lex gloss> == 'asociar'
(11834) <lex add prop> == 'takes_:__Caus '.
(11835)
(11836) R_nit:
(11837) <> == V10i
(11838) <lex name> == 'nit'
(11839) <lex noload> == '00658'
(11840) <phon under> == 'li:t'
(11841) <lex gloss> == 'perder.vi'.
(11842)
(11843) R_nnalÄllej:
(11844) <> == V10i
(11845) <lex name> == 'nna\'lÄllej'
(11846) <lex noload> == '00666'
(11847) <phon under> == 'na\':lchej'
(11848) <lex gloss> == 'sorprender'
(11849) <lex order> == '10 |no Habitual'.
(11850)
(11851) R_nnao:
(11852) <> == V10t
(11853) <lex name> == 'nnao'
(11854) <lex noload> == '00667'
(11855) <phon under> == 'nao'
(11856) <lex form> == ALLO1 ALLO2 DEFAULT

```

```

(11857) <mor form allo1> == 'na under_ao / _ o\'      +/ _ 2s'
(11858) <mor form allo2> == 'nakw          / _ [V-hi+cg] +/ _ [1s-or-
      3r]
(11859)          / _ [V-hi-cg]p | where 1s or 3r is followed
      by 3f or 3a'
(11860) <lex gloss> == 'seguir'.
(11861)
(11862) R_nne:
(11863) <> == V10t
(11864) <lex name> == 'nne'
(11865) <lex noload> == '00670'
(11866) <phon under> == 'ne'
(11867) <lex form> == ALLO1 ALLO2 DEFAULT
(11868) <mor form allo1> == 'ni / _ a\' +/ _ 1s'
(11869) <mor form allo2> == 'n, / _ # +/ [Modal] ... _
(11870)          +/ Rep ...
(11871)          +/ [Imperatives] ... _
(11872)          +/ Pl.pfx ... _'
(11873) <mor sec default> == '/ # ([C]) _ #
(11874)          /
          ~#\'
(11875) <lex gloss> == 'decir2'.
(11876)
(11877) R_nnebia:
(11878) <> == V10
(11879) <lex name> == 'nnebia\'\'
(11880) <lex noload> == '00672'
(11881) <phon under> == 'nepia\'\'
(11882) <lex form> == ALLO1 DEFAULT
(11883) <mor form allo1> == 'ni / _ a\' +/ _ 1s'
(11884) <lex prop> == 'vt class2'          % example of
      overriding an inherited value
(11885) <lex gloss> == 'gobernar'.
(11886)
(11887) R_nnit:
(11888) <> == V10i
(11889) <lex name> == 'nni\'t'
(11890) <lex noload> == '00675'
(11891) <phon under> == 'ni\'t'
(11892) <lex gloss> == 'estar.regado | take group 2 pns?'.
(11893)
(11894)
(11895) R_nnitrao:
(11896) <> == V10t
(11897) <lex name> == 'nnitrao'
(11898) <lex noload> == '00676'
(11899) <phon under> == 'ni:tla'
(11900) <lex form> == ALLO1 ALLO2 DEFAULT
(11901) <mor form allo1> == 'ni:tla under_ao / _ o\'      +/ _
      2s'
(11902) <mor form allo2> == 'ni:tlakw          / _ [V-hi+cg] +/ _
      [1s-or-3r]'
(11903) <lex gloss> == 'borrar.vi'.
(11904)
(11905) R_saa:
(11906) <> == V10i
(11907) <lex name> == 'sa\'a'
(11908) <lex noload> == '00752'

```



```

(11909) <phon under> == 'sa\'
(11910) <lex form> == ALLO1 ALLO2 ALLO3 DEFAULT
(11911) <mor form allo1> == 's / _ a\' +/_ 1s'
(11912) <mor form allo2> == 'se\' / _ [V-hi-cg] +/_ [1s-or-3r]'
(11913) <mor form allo3> == 'sa\'a / _ #'
(11914) <lex sec default> == '/ ~_ #'
(11915) <lex add prop> == 'laryngealizes '
(11916) <lex gloss> == 'ir/salir/caminar | caminar formerly separate
      record'.
(11917)
(11918)
(11919) R_sarall:
(11920) <> == V10i
(11921) <lex name> == 'sa\'ra\'ll'
(11922) <lex noload> == '00757'
(11923) <phon under> == 'sa\'la\'ch'
(11924) <lex gloss> == 'sospechar'
(11925) <lex com> == 'see also recordar2 wasa\'ra\'ll'.
(11926)
(11927) R_se:
(11928) <> == V10i
(11929) <lex noload> == '00759'
(11930) <phon under> == 'se'
(11931) <lex form> == ALLO1 ALLO2 ALLO3 DEFAULT
(11932) <mor form allo1> == 'si / _ a\' +/_ 1s'
(11933) <mor form allo2> == 's, / _ # +/_ [Modal] ... _
(11934)           +/_ Rep ...
(11935)           +/_ [Imperatives] ... _
(11936)           +/_ Plural.S ... _'
(11937) <mor form allo3> == 's, / _ :tese| +/_ [Modal] ... _
(11938)           | +/_ Rep ...
(11939)           | +/_ [Imperatives] ... _
(11940)           | +/_ Plural.S ... _'
(11941) <lex sec default> == '/ # ([C]) _ #
(11942)           /
(11943)           _ ~_ #'
(11944) <lex gloss> == 'parar.vi1'.
(11945)
(11946) R_sella:
(11947) <> == V10i
(11948) <lex name> == 'sella'
(11949) <lex noload> == '00761'
(11950) <phon under> == 'secha'
(11951) <lex form> == ALLO1 ALLO2 ALLO3 DEFAULT ALLO4 ALLO5 ALLO6
      ALLO7
(11952) <mor form allo1> == 'sech / _ a\' +/_ 1s'
(11953) <mor form allo2> == 'seche / _ e\' +/_ 3r'
(11954) <mor form allo3> == 'sech / _ #'
(11955) <mor form allo4> == 'soch / _ a\' +/_ 1s'
(11956) <mor form allo5> == 'soche / _ e\' +/_ 3r'
(11957) <mor form allo6> == 'soch / _ #'
(11958) <mor form allo7> == 'socha / ~_ #'
(11959) <lex sec default> == '/ ~_ #'
(11960) <lex gloss> == 'parar.vi2 | Is -cha really part of root or is
      it an Adv? yes'.
(11961)
(11962) R_serall:
(11963) <> == V10t

```


A4.4 The Yalálag Zapotec DATR Query File

A4.4.1 Description

DATR theory (lexicon) files contain sets of nodes. Queries against a theory file may be made interactively from a DATR command line, or alternatively, may be run in batch mode. In batch mode, a query file may be used so the user does not have to manually type in each query. The Yalálag Zapotec query file listing in this appendix is used to generate AMPLE entries from the Yalálag Zapotec DATR theory file. Each entry in the query file has a corresponding entry in the original Yalálag Zapotec AMPLE root database file. The path `<amp lex mor name>` is used to direct the query to invoke the AMPLE interface and to use the morph name as the value for the record marker.

A4.4.2 File Listing

Filename: yal.qry

Note: line numbers added for reference only. Not in actual file.

```
(11978)      R_chaa:<amp lex morname>
(11979)      R_chach:<amp lex morname>
(11980)      R_charall:<amp lex morname>
(11981)      R_chel:<amp lex morname>
(11982)      R_chez:<amp lex morname>
(11983)      R_chia:<amp lex morname>
(11984)      R_chio:<amp lex morname>
(11985)      R_labj:<amp lex morname>
(11986)      R_lall:<amp lex morname>
(11987)      R_lap:<amp lex morname>
(11988)      R_len:<amp lex morname>
(11989)      R_nit:<amp lex morname>
(11990)      R_nnalAllej:<amp lex morname>
(11991)      R_nnao:<amp lex morname>
(11992)      R_nne:<amp lex morname>
(11993)      R_nnebia:<amp lex morname>
(11994)      R_nnit:<amp lex morname>
(11995)      R_nnitrao:<amp lex morname>
(11996)      R_saa:<amp lex morname>
(11997)      R_sarall:<amp lex morname>
(11998)      R_se:<amp lex morname>
(11999)      R_sella:<amp lex morname>
(12000)      R_serall:<amp lex morname>
(12001)      R_sill:<amp lex morname>
```

A4.5 Batch File to Execute DATR

A4.5.1 Description

The batch file listed below is a standard DOS batch file. It contains two statements. The first calls the ZDATR program *zdatrtok*. *Zdatrtok* produces a token file from the *yal.dtr* file and names it *yal.dtr.tok*. The second program called is *zdatrinf* which is the inference program. This program uses both the *yal.qry* file and the *yal.dtr.tok* file as inputs, and then outputs the results of the query to a file named *yal.dtr.trc*. *yal.dtr.trc* is the Yalálag Zapotec trace file.

A4.5.2 File Listing

Filename: goyal.bat

Note: line numbers added for reference only. Not in actual file.

```
(12002)      zdatrtok -o yal.dtr.tok yal.dtr
(12003)      zdatrinf -j -u -V -1 -T yal.dtr.trc -Q yal.qry yal.dtr.tok
```

A4.6 The Yalálag Zapotec DATR Trace File

A4.6.1 Description

The result of running the batch file described above is a trace file called *Yal.dtr.trc*. Each line in the query file *Yal.qry* results in a corresponding line in the trace file. Note that although the trace file entries are close to AMPLE format, two things must be done before they are usable. First, empty lines and empty fields must be removed. Second, the trailing period must be stripped off. This is handled by a Perl script listed in Appendix Five.

A4.6.2 File Listing

Filename: yal.dtr.trc

Note: line numbers added for reference only. Not in actual file.

```

(12004)    \r cha'a
(12005)    \_no 00129
(12006)    \u cha'
(12007)    \a cha' /~_ #
(12008)    \a cha'a /_ #
(12009)    \c VA
(12010)    \g estar.en.grupo
(12011)    \m 10
(12012)    \mp vi class2 vt takes_null_S
(12013)    \mcc estar.en.grupo / ~Pl.pfx _
(12014)    \co
(12015)
(12016)    .
(12017)    \r chach
(12018)    \_no 00130
(12019)    \u cha:ch
(12020)    \a cha:ch
(12021)    \c VA
(12022)    \g aparecer.muchos
(12023)    \m 10
(12024)    \mp vi class2 takes_null_S
(12025)    \mcc
(12026)    \co llach kwerp kello - effect of something that scares us
(12027)
(12028)    .
(12029)    \r cha'ra'll
(12030)    \_no 00138
(12031)    \u cha'la'ch
(12032)    \a cha'la'ch
(12033)    \c VA
(12034)    \g engordar
(12035)    \m 10
(12036)    \mp vi class2 takes_:_Caus takes_null_S
(12037)    \mcc
(12038)    \co
(12039)
(12040)    .
(12041)    \r chel
(12042)    \_no 00158
(12043)    \u che:l
(12044)    \a che:l
(12045)    \c VA
(12046)    \g enrollar          | take group 2 pns??
(12047)    \m 9
(12048)    \mp vi class2 takes_null_S takes_:_Caus
(12049)    \mcc
(12050)    \co

```

```

(12051) .
(12052) \r che'z
(12053) \_no 00170
(12054) \u che's
(12055) \a che's +/ [P-or-Modal] _
(12056) \a e's +/ H _
(12057) \a che's +/ [P,H-or-Modal] ~_
(12058) \c VA
(12059) \g romper
(12060) \m 9
(12061) \mp vi class2 takes_:_Caus
(12062) \mcc
(12063) \co
(12064)
(12065) .
(12066) \r chia
(12067) \_no 00172
(12068) \u chia
(12069) \a chi / _ [V-hi+cg] +/ _ [1s-or-3r]
(12070) \a chia
(12071) \c VA
(12072) \g coser.vt
(12073) \m 9
(12074) \mp vt class2
(12075) \mcc
(12076) \co
(12077)
(12078) .
(12079) \r chi'o
(12080) \_no 00195
(12081) \u chi'o
(12082) \a chi' / _ o' +/ _ 2s
(12083) \a chi'kw / _ [V-hi+cg] +/ _ [1s-or-3r]
(12084) \a chi'o
(12085) \c VA
(12086) \g crecer
(12087) \m 9
(12088) \mp vi class2 takes_:_s_Caus
(12089) \mcc
(12090) \co
(12091)
(12092) .
(12093) \r labj
(12094) \_no 00495
(12095) \u lapj
(12096) \a lapj
(12097) \c VA
(12098) \g susmergir | group 2 pns??
(12099) \m 10
(12100) \mp vi class2 takes_null_S
(12101) \mcc
(12102) \co
(12103)
(12104) .
(12105) \r la'll
(12106) \_no 00498
(12107) \u la'll

```

```

(12108) \a la'll
(12109) \c VA
(12110) \g vagar
(12111) \m 10
(12112) \mp vi class2 takes_null_S
(12113) \mcc
(12114) \co
(12115)
(12116) .
(12117) \r lap
(12118) \_no 00503
(12119) \u la:p
(12120) \a la:p
(12121) \c VA
(12122) \g pizcar
(12123) \m 10
(12124) \mp vt class2 takes_null_S
(12125) \mcc
(12126) \co
(12127)
(12128) .
(12129) \r len
(12130) \_no 00536
(12131) \u len
(12132) \a len
(12133) \c VA
(12134) \g asociar
(12135) \m 10
(12136) \mp vi class2 takes_:__Caus takes_null_S
(12137) \mcc
(12138) \co
(12139)
(12140) .
(12141) \r nit
(12142) \_no 00658
(12143) \u li:t
(12144) \a li:t
(12145) \c VA
(12146) \g perder.vi
(12147) \m 10
(12148) \mp vi class2 takes_null_S
(12149) \mcc
(12150) \co
(12151)
(12152) .
(12153) \r nna'lÄllej
(12154) \_no 00666
(12155) \u na':lchej
(12156) \a na':lchej
(12157) \c VA
(12158) \g sorprender
(12159) \m 10 |no Habitual
(12160) \mp vi class2 takes_null_S
(12161) \mcc
(12162) \co
(12163)
(12164) .

```

```

(12165) \r nnao
(12166) \_no 00667
(12167) \u nao
(12168) \a na under_ao / _ o' +/ _ 2s
(12169) \a nakw / _ [V-hi+cg] +/ _ [1s-or-3r]
(12170) / _ [V-hi-cg]p | where 1s or 3r is followed
        by 3f or 3a
(12171) \a nao
(12172) \c VA
(12173) \g seguir
(12174) \m 10
(12175) \mp vt class2 takes_null_S
(12176) \mcc
(12177) \co
(12178) .
(12179) .
(12180) \r nne
(12181) \_no 00670
(12182) \u ne
(12183) \a ni / _ a' +/ _ 1s
(12184) \a n, / _ # +/ [Modal] ... _
(12185) +/ Rep ... _
(12186) +/ [Imperatives] ... _
(12187) +/ Pl.pfx ... _
(12188) \a ne / # ([C]) _ #
(12189) / _ ~#
(12190) \c VA
(12191) \g decir2
(12192) \m 10
(12193) \mp vt class2 takes_null_S
(12194) \mcc
(12195) \co
(12196) .
(12197) .
(12198) \r nnebia'
(12199) \_no 00672
(12200) \u nepia'
(12201) \a ni / _ a' +/ _ 1s
(12202) \a nepia'
(12203) \c VA
(12204) \g gobernar
(12205) \m 10
(12206) \mp vt class2
(12207) \mcc
(12208) \co
(12209) .
(12210) .
(12211) \r nni't
(12212) \_no 00675
(12213) \u ni':t
(12214) \a ni':t
(12215) \c VA
(12216) \g estar.regado | take group 2 pns?
(12217) \m 10
(12218) \mp vi class2 takes_null_S
(12219) \mcc
(12220) \co

```



```

(12221)
(12222) .
(12223) \r nmitrao
(12224) \_no 00676
(12225) \u ni:tla
(12226) \a ni:tla under_ao / _ o' +/ _ 2s
(12227) \a ni:tlakw / _ [V-hi+cg] +/ _ [1s-or-3r]
(12228) \a ni:tla
(12229) \c VA
(12230) \g borrar.vi
(12231) \m 10
(12232) \mp vt class2 takes_null_S
(12233) \mcc
(12234) \co
(12235)
(12236) .
(12237) \r sa'a
(12238) \_no 00752
(12239) \u sa'
(12240) \a s / _ a' +/ _ 1s
(12241) \a se' / _ [V-hi-cg] +/ _ [1s-or-3r]
(12242) \a sa'a / _ #
(12243) \a sa'
(12244) \c VA
(12245) \g ir/salir/caminar | caminar formerly separate record
(12246) \m 10
(12247) \mp vi class2 laryngealizes takes_null_S
(12248) \mcc
(12249) \co
(12250)
(12251) .
(12252) \r sa'ra'll
(12253) \_no 00757
(12254) \u sa'la'ch
(12255) \a sa'la'ch
(12256) \c VA
(12257) \g sospechar
(12258) \m 10
(12259) \mp vi class2 takes_null_S
(12260) \mcc
(12261) \co see also recordar2 wasarall
(12262)
(12263) .
(12264) \r se
(12265) \_no 00759
(12266) \u se
(12267) \a si / _ a' +/ _ 1s
(12268) \a s, / _ # +/ [Modal] ... _
(12269) +/ Rep ... _
(12270) +/ [Imperatives] ... _
(12271) +/ Plural.S ... _
(12272) \a s, / _ :tese| +/ [Modal] ... _
(12273) | +/ Rep ... _
(12274) | +/ [Imperatives] ... _
(12275) | +/ Plural.S ... _
(12276) \a se
(12277) \c VA

```

```

(12278) \g parar.vi1
(12279) \m 10
(12280) \mp vi class2 takes_null_S
(12281) \mcc
(12282) \co
(12283)
(12284) .
(12285) \r sella
(12286) \_no 00761
(12287) \u secha
(12288) \a sech /_ a' +/_ 1s
(12289) \a seche /_ e' +/_ 3r
(12290) \a sech /_ #
(12291) \a secha
(12292) \a soch /_ a' +/_ 1s
(12293) \a soche /_ e' +/_ 3r
(12294) \a soch /_ #
(12295) \a socha /_ #
(12296) \c VA
(12297) \g parar.vi2 | Is -cha really part of root or is it an Adv?
yes
(12298) \m 10
(12299) \mp vi class2 takes_null_S
(12300) \mcc
(12301) \co
(12302)
(12303) .
(12304) \r sera'll
(12305) \_no 00764
(12306) \u sela'ch
(12307) \a sela'ch
(12308) \c VA
(12309) \g desear
(12310) \m 10
(12311) \mp vt class2 takes_null_S
(12312) \mcc
(12313) \co
(12314)
(12315) .
(12316) \r sill
(12317) \_no 00773
(12318) \u sich
(12319) \a sich
(12320) \c VA
(12321) \g sonar
(12322) \m 10
(12323) \mp vi class2 takes_:_Caus takes_null_S
(12324) \mcc
(12325) \co
(12326)
(12327) .

```

APPENDIX 5

PERL SCRIPTS

A4.1 Script to Compare Database Files

A4.1.1 Description

The following Perl script serves two main purposes. First, it compares the Ogea AMPLE lexicon generated from DATR to the original Ogea AMPLE lexicon. Discrepancies are written to an error file. Second, it splits the generated lexicon into separate root and suffix database files. These separate files may be run directly within AMPLE.

The input files used for the comparison are the *Ogea.dtr.trc file* generated by DATR, and a file named *compare.txt*. Compare.txt is a file that merges the original Ogea AMPLE root database file and the suffix file into a single file. Each AMPLE record in compare.txt is a single line. Line breaks before all non-record marker fields were removed. Also, all whitespace was reduced to a single space character. In this way the original Ogea AMPLE lexicon was normalized for comparison to the DATR generated Ogea AMPLE lexicon. Comparing each record as a single line was easier programmatically than dealing with records where each field begins a new line.

A4.1.2 The File Listing

Filename: compare.pl.

Note: Line numbers not in original. Added for reference only.

```

(12328) # AUTHOR: Michael Colburn, mcolburn@sprintmail.com
(12329) # DATE: November, 1999
(12330) # Filename: Compare.pl
(12331) # Purpose:
(12332) #     1. Cleans up the output of Ogea DATR lexicon queries for
(12333) #         comparison to the original AMPLE verion,
(12334) #     2. Writes discrepencies to an error file,
(12335) #     3. Writes cleaned up version to an AMPLE unified
(12336) #         database file.
(12337)
(12338) # original.db is the original Ogea AMPLE unified database file
(12339) open (AMPLE_IN,
(12340)     'c:\program files\sil\carlastudio\ogea\original.db');
(12341)
(12342) # ogea.dtr.trc is the DATR trace (output) file
(12343) #     resulting from running ogea.gry.
(12344) open (DATR_IN, 'c:\zdatr\zdatr20\program\bin\ogea.dtr.trc');
(12345) open (ERROR_OUT,
(12346)     ">c:\\zdatr\\zdatr20\\program\\bin\\error.txt");
(12347)
(12348) # Output creates an AMPLE unified database file.
(12349) open (AMPLE_OUT,
(12350)     ">c:\\program files\\sil\\carlastudio\\ogea\\ogea.db");
(12351)
(12352) $Error_CNT = 0;
(12353) $/ = "\n\n"; # set record delimiter to two line breaks
(12354)
(12355) while (<AMPLE_IN>) {
(12356)
(12357)     # Normalize the AMPLE input record
(12358)     $AMPL = $_;
(12359)     $AMPL =~ s/\n/ /g; # replace line breaks with spaces
(12360)     $AMPL =~ s/ / /g; # get rid of extra spaces
(12361)     $AMPL =~ s/ / /g; # get rid of extra spaces
(12362)     $AMPL =~ s/ / /g; # get rid of extra spaces
(12363)
(12364)     # Normalize the DATR input record
(12365)     $DATR = <DATR_IN>;
(12366)     $DATR =~ s/^\.\n//g; # get rid of period inserted by DATR
(12367)
(12368)     # get rid of empty fields
(12369)     $DATR =~ s/\\w\\w\\w \\//g;
(12370)     $DATR =~ s/\\w\\w\\w \n\n/g;
(12371)     $DATR =~ s/\\w\\w \\//g;
(12372)     $DATR =~ s/\\w\\w \n\n/g;
(12373)     $DATR =~ s/\\w \\//g;
(12374)     $DATR =~ s/\\w \n\n/g;
(12375)
(12376)     $DATR =~ s/\n/ /g; # replace line breaks with spaces
(12377)     $DATR =~ s/ / /g; # get rid of extra spaces
(12378)     $DATR =~ s/ / /g; # get rid of extra spaces
(12379)     $DATR =~ s/ / /g; # get rid of extra spaces
(12380)
(12381)     # The following code compares the DATR output to the
(12382)     # original AMPLE files and writes out lines that
(12383)     # fail the comparison

```

```

(12384)
(12385)     $DATR_CNT++;
(12386)     if (($AMPLE ne $DATR) && ($DATR ne '')) {
(12387)         print ERROR_OUT "A:$AMPLE"."\\nD:$DATR\\n";
(12388)         $Error_CNT++;
(12389)     }
(12390)
(12391)     # put line breaks back in
(12392)     $DATR =~ s/\\/\\n\\/g;
(12393)     print AMPLE_OUT "$DATR\\n";
(12394) }
(12395)
(12396) close (AMPLE_IN);
(12397) close (DATR_IN);
(12398) close (ERROR_OUT);
(12399) close (AMPLE_OUT);
(12400) print "AMPLE: $AMPLE_CNT DATR: $DATR_CNT Errors: $Error_CNT\\n";

```

A4.2 Script to Generate AMPLE Database File

A4.2.1 Description

The following Perl Script is a version of the previous script (compare.pl) that simply cleans up the DATR trace file and writes out an AMPLE dictionary database file.

A4.2.2 The File Listing

Filename: DToAMPLE.pl.

Note: Line numbers not in original. Added for reference purposes.

```

(12401)     # AUTHOR: Michael Colburn, mcolburn@sprintmail.com
(12402)     # DATE: November, 1999
(12403)     # Filename: DToAMPLE.pl
(12404)     # Purpose: Cleans up DATR trace file, writes to AMPLE
(12405)     #             unified database file.
(12406)
(12407)     # ogea.dtr.trc is the DATR trace (output) file
(12408)     # resulting from running ogea.qry.
(12409)     open (DATR_IN,
(12410)         'c:\zdatr\zdatr20\program\bin\ogea.dtr.trc');
(12411)
(12412)     # Output creates an AMPLE unified database file
(12413)     open (AMPLE_OUT,
(12414)         ">c:\program files\\sil\\carlastudio\\ogea\\ogea.db");
(12415)
(12416)     $/ = "\\n"; # set record delimiter to match DATR's output

```

```

(12417)     while (<DATR_IN>) {
(12418)
(12419)         $DATR = $_;
(12420)
(12421)         $DATR =~ s/\. \n//g; # get rid of period inserted by DATR
(12422)
(12423)         # get rid of empty fields
(12424)         $DATR =~ s/\\w\w\w \\//g;
(12425)         $DATR =~ s/\\w\w\w \n\n/g;
(12426)         $DATR =~ s/\\w\w \\//g;
(12427)         $DATR =~ s/\\w\w \n\n/g;
(12428)         $DATR =~ s/\\w \\//g;
(12429)         $DATR =~ s/\\w \n\n/g;

(12430)         $DATR =~ s/\n/ /g; # replace line breaks with spaces
(12431)         $DATR =~ s/ / /g; # get rid of extra spaces
(12432)         $DATR =~ s/ / /g; # get rid of extra spaces
(12433)         $DATR =~ s/ / /g; # get rid of extra spaces
(12434)
(12435)         # put line breaks back in
(12436)         $DATR =~ s/\\/ \n\\/g;
(12437)         print AMPLE_OUT "$DATR\n";
(12438)     }
(12439)
(12440)     close (AMPLE_IN);
(12441)     close (AMPLE_OUT);

```

A4.3 Scripts to Compare AMPLE Output

A4.3.1 Description

This Perl script compares two AMPLE interlinear files. The interlinear files are the result of running AMPLE in CARLA Studio, using the file `list.txt` as the text file to be parsed. First AMPLE is run using the hand-created Ogea AMPLE database file *original.db*, which is copied to *ogea.db* for the run. The resulting interlinear file, `list.itx` is renamed (by hand) to *old.itx*. Next, the DATR to AMPLE interface is executed using the batch file *go.bat*, which creates a new version of *ogea.db*, generated from DATR (see Appendix 3 for the listing). AMPLE is run again, and this time the resulting interlinear file is named *new.itx*. Finally, the Perl script *C2.pl* is run.

A4.3.2 The File Listing

Filename: C2.pl.

Note: Line numbers not in original. Added for reference purposes.

```
(12442)      # AUTHOR: Michael Colburn, mcolburn@sprintmail.com
(12443)      # DATE:  November, 1999
(12444)      # Filename: C2.pl
(12445)      # Purpose: compares the AMPLE output from using the hand
(12446)      #           developed Ogea AMPLE database file to the
(12447)      #           AMPLE output from using the version generated
(12448)      #           through DATR
(12449)
(12450)      # original.itx is the original Ogea AMPLE output file
(12451)      open (OLD_IN, 'c:\program files\sil\carlastudio\ogea\ERI
(12452)          Interlinear Files\original.itx');
(12453)
(12454)      # new.itx is the interlinear file AMPLE produces using the
(12455)      # database file generated from DATR
(12456)      open (OLD_IN, 'c:\program files\sil\carlastudio\ogea\ERI
(12457)          Interlinear Files\new.itx');
(12458)
(12459)      open (ERROR_OUT,
(12460)          ">c:\\program
(12461)      files\\sil\\carlastudio\\ogea\\ERI
(12462)          Interlinear Files\\error.txt");
(12463)      $Error_CNT = 0;
(12464)
(12465)      while (<OLD_IN>) {
(12466)          $OLD_CNT++;
(12467)          $OLD = $_;
(12468)          $NEW = <NEW_IN>;
(12469)          $NEW_CNT++;
(12470)          if (($OLD ne $NEW) && ($NEW ne '')) {
(12471)              print ERROR_OUT "O:$OLD"."N:$NEW\n";
(12472)              $Error_CNT++;
(12473)          }
(12474)      }
(12475)
(12476)      close (OLD_IN);
(12477)      close (NEW_IN);
(12478)      close (ERROR_OUT);
(12479)      print "OLD: $OLD_CNT NEW: $NEW_CNT Errors: $Error_CNT\n";
```

A4.4 Scripts to Calculate Reduction of Redundancy

A4.4.1 For Ogea

A4.4.1.1 The AMPLE File

A4.4.1.1.1 Description

The Perl script *AMPStat.pl* examines each line of each record in the hand-built Ogea AMPLE database file *original.db*. A counter variable exists for each field type found in a record. The value of each counter is displayed after the entire file has been processed. These values are compared to the output of the file *DATStat.pl*, listed in another section below.

A4.4.1.1.2 The File Listing

Filename: AMPStat.pl.

Note: Line numbers not in original. Added for reference purposes.

```
(12480)      # AUTHOR: Michael Colburn, mcolburn@sprintmail.com
(12481)      # DATE:  November, 1999
(12482)      # Filename: AMPStat.pl
(12483)      # Purpose: Counts the number of occurrences of
(12484)      #           specified fields. This number is compared to
(12485)      #           the number of occurrences
(12486)      #           of the corresponding DATR paths in the
(12487)      #           Ogea DATR lexicon. This provides a measure of the
(12488)      #           reduction of redundancy by using DATR.

(12489)      open (AMPLE_IN, 'c:\program
(12490)      files\sil\carlastudio\ogea\original.db');
```



```

(12491) while (<AMPLE_IN>) {
(12492)     $saf++ if (/\a/); # allomorph
(12493)     $sec++ if (\/ /); # SEC
(12494)     $mec++ if (\/+\/); # MEX
(12495)     $c++   if (\/c/); # Category
(12496)     $co++  if (\/co/); # Comment
(12497)     $f++   if (\/f/); # Feature
(12498)     if (\/g/) { # Gloss
(12499)         if (\/ge/) { # English gloss
(12500)             $ge++;
(12501)         } else {
(12502)             $g++; # Morphname
(12503)         }
(12504)     }
(12505)     $u++   if (\/u/); # underlying form
(12506)     $gn++  if (\/gn/); # national language
(12507)     $loc++ if (\/loc/); # infix location
(12508)     $mp++  if (\/mp/); # morpheme property
(12509)     $o++   if (\/o/); # order class
(12510)     $mcc++ if (\/mcc/); # MCC
(12511)     $type++ if (\/type/); # record type
(12512) }
(12513)
(12514) print "\n af:$saf \n ap:$sap \n sec:$sec \n mec:$mec \n c:$c
(12515)        \n co:$co \n f:$f \n ge:$ge \n u:$u \n gn:$gn
(12516)        \n loc:$loc
(12517)        \n g:$g \n mp:$mp \n o:$o \n mcc:$mcc \n type:$type\n";
(12518) close (DATR_IN);

```

A4.4.1.2 The DATR Ffile

A4.4.1.2.1 Description

The Perl script *DATStat.pl* examines each path of each root node in the Ogea DATR theory file *Ogea.dtr*. A counter variable exists for each path type found in a node. The value of each counter is displayed after the entire file has been processed. These values are compared to the output of the file *AMPSStat.pl*, listed above.

A4.4.1.2.2 The File Listing

Filename: DATStat.pl.

Note: Line numbers not in original. Added for reference purposes.

```

(12519) # AUTHOR: Michael Colburn, mcolburn@sprintmail.com
(12520) # DATE: November, 1999
(12521) # Filename: DATStat.pl
(12522) # Purpose: Counts the number of occurrences of specified paths
(12523) #           This number is compared to the number of
(12524) #           occurrences of the corresponding AMPLE field
(12525) #           markers in the Ogea AMPLE lexicon. This provides
(12526) #           a measure of the reduction of redundancy
(12527) #           by using DATR.
(12528)
(12529) open (DATR_IN, 'c:\zdatr\zdatr20\program\bin\ogea.dtr');
(12530)
(12531) while (<DATR_IN) {
(12532)     if (/^%/) {
(12533)         next;
(12534)     }
(12535)     if (/^ENG_/) {
(12536)         $Hit_ENG = 1;
(12537)         $ECNT++;
(12538)     }
(12539)     if ($Hit_ENG) {
(12540)         $ge++ if (/^<lex gloss eng>/);
(12541)         $g++ if (/^<lex gloss eng morname>/);
(12542)     }
(12543)     if (/^R_/ || /^S_/) {
(12544)         $Hit_ENG = 0;
(12545)         $Hit_R = 1;
(12546)         $RCNT++;
(12547)     }
(12548)     if ($Hit_R) {
(12549)         $af++ if (/^<allo form>/);
(12550)         $af++ if (/^<lex form>/);
(12551)         $ap++ if (/^<allo prop>/);
(12552)         $sec++ if (/^<allo sec>/);
(12553)         $mec++ if (/^<allo mec>/);
(12554)         $c++ if (/^<lex cat>/);
(12555)         $co++ if (/^<lex com>/);
(12556)         $f++ if (/^<lex feat>/);
(12557)         $u++ if (/^<phon under>/);
(12558)         $gn++ if (/^<lex gloss nat>/);
(12559)         $loc++ if (/^<lex loc>/);
(12560)         $mp++ if (/^<lex prop>/);
(12561)         $o++ if (/^<lex order class>/);
(12562)         $mcc++ if (/^<lex mcc>/);
(12563)         $type++ if (/^<lex type>/);
(12564)     }
(12565) }
(12566)
(12567) print "\nRoot or Suffix Nodes: $RCNT \n ENG_ nodes: $ECNT
(12568) \n af:$af \n ap:$ap \n sec:$sec \n mec:$mec \n c:$c
(12569) \n co:$co \n f:$f \n ge:$ge \n u:$u \n gn:$gn \n loc:$loc
(12570) \n g:$g \n mp:$mp \n o:$o \n mcc:$mcc \n type:$type\n";
(12571) close (DATR_IN);

```

A4.4.2 For Yalalog Zapotec

A4.4.2.1 The AMPLE File

A4.4.2.1.1 Description

The Perl script *AMPStat2.pl* examines each line of each record in the Yalalog Zapotec AMPLE database file *yal0910.db*. This file is an extraction from the file *yalrt.db*, and contains all Order Class 10 records, and four Order Class 9 records. In the script, a counter variable exists for each field type found in a record. The value of each counter is displayed after the entire file has been processed. These values are compared to the output of the file *DATStat2.pl*, listed in another section below.

A4.4.2.1.2 The File Listing

Filename: AMPStat2.pl.

Note: Line numbers not in original. Added for reference purposes.

```
(12572) # AUTHOR: Michael Colburn, mcolburn@sprintmail.com
(12573) # DATE: November, 1999
(12574) # Filename: AMPStat2.pl
(12575) # Purpose: Counts the number of occurrences of
(12576) #           specified fields. This number is compared
(12577) #           to the number of occurrences
(12578) #           of the corresponding DATR paths in the
(12579) #           Yalalog Zapotec DATR lexicon. This provides
(12580) #           a measure of the reduction of redundancy
(12581) #           by using DATR.
```

```

(12582)    open (AMPLE_IN, 'c:\zdatr\zdatr20\program\bin\yal0910.db');
(12583)    while (<AMPLE_IN>) {

(12584)        $saf++ if (/\a /); # allomorph
(12585)        $sec++ if (\/ /); # SEC
(12586)        $mec++ if (\/+\/ /); # MEX
(12587)        $c++ if (\/c /); # Category
(12588)        $co++ if (\/co /); # Comment
(12589)        $f++ if (\/f /); # Feature
(12590)        if (\/g /) { # Gloss
(12591)            if (\/ge /) { # English gloss
(12592)                $ge++;
(12593)            } else {
(12594)                $g++; # Morphname
(12595)            }
(12596)        }
(12597)        $no++ if (\/no /); # no load
(12598)        $u++ if (\/u /); # underlying form
(12599)        $gn++ if (\/gn /); # national language
(12600)        $loc++ if (\/loc /); # infix location
(12601)        $mp++ if (\/mp /); # morpheme property
(12602)        $o++ if (\/m /); # order class
(12603)        $mcc++ if (\/mcc /); # MCC
(12604)        $type++ if (\/type /); # record type
(12605)    }
(12606)
(12607)    print "\n af:$saf \n ap:$ap \n sec:$sec \n mec:$mec
(12608)        \n No load:$no \n c:$c \n co:$co \n f:$f \n ge:$ge
(12609)        \n u:$u \n gn:$gn \n loc:$loc \n g:$g \n mp:$mp
(12610)        \n o:$o \n mcc:$mcc \n type:$type\n";
(12611)    close (DATR_IN);

```

A4.4.2.2 The DATR File

A4.4.2.2.1 Description

The Perl script *DATStat2.pl* examines each path of each root node in the Yalalog Zapotec DATR theory file *yal.dtr*. A counter variable exists for each path type found in a node. The value of each counter is displayed after the entire file has been processed. These values are compared to the output of the file *AMPStat2.pl*, listed above.

A4.4.2.2.2 The File Listing

Filename: DATStat2.pl.

Note: Line numbers not in original. Added for reference purposes.

```

(12612) # AUTHOR: Michael Colburn, mcolburn@sprintmail.com
(12613) # DATE: November, 1999
(12614) # Filename: DATStat2.pl
(12615) # Purpose: Counts the number of occurrences of specified paths
(12616) #           This number is compared to the number of
(12617) #           occurrences
(12618) #           of the corresponding AMPLE field markers in the
(12619) #           Yalalog Zapotec AMPLE lexicon. This provides a
(12620) #           measure of the reduction of redundancy by
(12621) #           using DATR.
(12622) open (DATR_IN, 'c:\zdatr\zdatr20\program\bin\yal.dtr');
(12623) while (<DATR_IN>) {
(12624)   if (/^%/) {
(12625)     next;
(12626)   }
(12627)   if (/^R_/) {
(12628)     $Hit_R = 1;
(12629)     $RCNT++;
(12630)   }
(12631)   if ($Hit_R) {
(12632)     $saf++ if (/^<allo form/ || (/^<mor form/));
(12633)     $sap++ if (/^<allo prop/);
(12634)     $c++  if (/^<lex cat/);
(12635)     $co++ if (/^<lex com/);
(12636)     $f++  if (/^<lex feat/);
(12637)     $ge++ if (/^<lex gloss>/);
(12638)     $gn++ if (/^<lex gloss nat/);
(12639)     $loc++ if (/^<lex loc/);
(12640)     $mcc++ if (/^<lex mcc/);
(12641)     $mec++ if (/^<mor mec/);
(12642)     $mp++ if (/^<lex prop/ || /^<lex add prop/);
(12643)     $name++ if (/^<lex name/);
(12644)     $no ++ if (/^<lex no load>/);
(12645)     $o++  if (/^<lex order/);
(12646)     $sec++ if (/^<mor sec/);
(12647)     $type++ if (/^<lex type/);
(12648)     $u++  if (/^<phon under/);
(12649)   }
(12650) }
(12651)
(12652) print "\nRoot Nodes: $RCNT \n Names:$name \n No load:$no
(12653)       \n af:$saf \n ap:$sap \n sec:$sec \n mec:$mec
(12654)       \n c:$c \n co:$co \n f:$f \n ge:$ge \n u:$u
(12655)       \n gn:$gn \n loc:$loc \n g:$g \n mp:$mp \n o:$o
(12656)       \n mcc:$mcc \n type:$type\n";
(12657) close (DATR_IN);

```