

$$\frac{z}{\sqrt{\lambda}} \leq z_0 \leq \frac{z}{\sqrt{\lambda}} + \frac{1}{\sqrt{\lambda}}, \quad \frac{z}{\sqrt{\lambda}} \leq z_0 \leq \frac{z}{\sqrt{\lambda}}$$

↑↓ ∅ ∅ z \ Z o L ∞ || e

lel-μ3D10-D1 lel-D1 jʰee-μ3D10

Middle Rymakonian, the language of Rymako

uruwi

$$\partial \partial \Gamma \uparrow = z \perp \partial \partial \Gamma \neq \partial \Gamma \partial z \parallel \partial \infty$$

een^ψ_J-Decbdebe-ጎጋጠ

A complete grammar

2 May 2018

Dedicated to Gufferdk.

Branch: canon

Version: 0.1

Date: 2017-12-22 (29 dia len)

(C)opyright 2017 Uruwi. See README.md for details.

| Contents

0.1	Introduction	5
1	Phonology and orthography	7
1.1	Phoneme inventory	7
1.2	Hacmisation	8
1.3	Neđam script	9
1.4	Phonotactics	10
1.4.1	Prosody	11
1.5	Vowel harmony	12
1.6	Rod signal sandhi	12
2	Syntax	13
2.1	Basic word order	13
2.2	Questions	13
2.3	Multiple clauses	13
3	Nouns	15
3.1	Number	15
3.2	Case	15
3.3	Noun classes	15
3.4	Definiteness	16
3.5	Declension table	16
3.5.1	Countable classes	16
3.5.2	Measurable and uncountable classes	16
3.6	Pronouns	17
3.6.1	Last-clause pronouns	18
3.7	Compounding	18
3.8	Possession	18
4	Verbs	21
4.1	Aspect	23
4.1.1	Simultaneous temporal and spatial aspects	24
4.2	Historically clausal arguments	24
4.2.1	Obliques	25
4.2.2	Conjunctions	25
4.3	Connectors	26
4.4	Comparatives	27

4.5	Ditransitive-like constructions	28
4.6	Transitivisation	28
4.7	The copula	29
5	Descriptors	31
6	Tree mode	33
6.1	Activation	33
6.2	Branch-switching	33
6.3	Anaphoric pronouns in joiner clauses	33
6.4	Errors	33
6.5	Example	34
7	Numerals	35
7.1	Irregular numerals	35
7.2	Double-digit numerals	36
7.3	Numerals up to 4096	36
7.4	Larger numerals	36
8	Derivational morphology	37
8.1	Abstraction	37
8.2	Dematuration	37
8.3	Verb-to-noun conversions	37
8.4	Verb-to-verb conversions	38
9	Names	39
9.1	Nominal names	39
9.2	Clausal names	40
9.3	Station names	41
9.3.1	Common words	41
10	Calendar	43
10.1	Tides	43
10.2	Months	44
10.3	Days within a month	44
10.4	Years	45
10.5	Eras	46
11	Quoting direct speech	47
11.1	Representation of binary data	47
11.2	The stack	47
11.3	Bytecode	48
11.4	Limitations	52
11.5	Example	53
12	Miscellanea	57
12.1	Colour	57
A	Listings of programs	59
A.1	workfiles/7/tides.sage	59
A.2	workfiles/7/bins.pl6	60

0.1. INTRODUCTION	5
B List of glossing abbreviations	63
C Quoting assembler	65
D Listings of programs	67
D.1 7_1/listings/example-nestedquote.qasm	67
D.2 7_1/listings/example-quote.qasm	68
E Dictionary	71

0.1 | Introduction

1 | Phonology and orthography

1.1 | Phoneme inventory

Middle Rymakonian underwent several sound changes from Lek-Tsaro, in the following order:

$s \rightarrow \text{ɬ}$	$(\blacklozenge\{w, j, u, y\})$	<i>NB this is a whistled sibilant.</i>
$\eta \rightarrow \text{j̥}$	$(\square\blacklozenge)$	
$\theta x \rightarrow \theta$	$\neg(\blacklozenge\square)[x = \emptyset]$	
$C_1[+fr] \rightarrow C_1[+v]$	$(V_1\blacklozenge V_2)$	
$\text{ɹ} \rightarrow \text{z}$	$(V_1\blacklozenge V_2)$	
$\{x, u\} \rightarrow \text{ɰ}$		
$V_1[+r] \rightarrow V_1[-r]$		
$k \rightarrow c$	$(\blacklozenge i)$	
$t \rightarrow \text{tʃ}$	$(\blacklozenge i)$	
$r \rightarrow \text{r}$		

Thus Middle Rymakonian has the following phoneme inventory:

Table 1.1: The consonants of Middle Rymakonian.

	Bilabial	Dental	Alveolar	Palatal	Velar	Glottal
Nasal	m		n	j̥	ŋ	
Plosive	p b		t d	c ɟ	k g	ʔ
Fricative	f v	θ ð	s z	ʃ ʒ	x ɣ	
(coarticulated)	fx vɣ	θx ðɣ		fʃ vʒ		
(whistled)			ɬ ʐ			
Affricate			ts	tʃ		
Lateral fricative			ɬ̺ ʐ̺			
Approximant			ɹ	j	w	
Lateral approximant			l			
Tap			r			

Table 1.2: The vowels of Middle Rymakonian.

	Front	Central	Back
High	i	ɤ	u
Mid	ɛ		ʌ
Low		a	

In addition to consonants and vowels, Middle Rymakonian has rod signals, represented by numbers. Rod A is blue and held by one's dominant hand and B is red and held by one's non-dominant hand. Rod signals can occur only at the end of words.

1. Rod A is raised to one's chest, while B is pointed down.
2. Rods A and B are crossed in the front.
3. Rod B is raised upwards in front of the nondominant arm, while rod A is lowered.
4. Rod A is pointed sideways near one's nondominant arm, while rod B is lowered.
5. Rods A and B are extended to the sides.
6. Rods A and B are extended, facing forward.
7. Rod A is raised forward, while B is pointed to the side.
8. Rod B is raised forward, while A is pointed to the side.
9. Rod A is raised besides one's head, while Rod B is extended toward the side of the dominant hand. This rod signal does not exist alone, but rather as a transition to the seventh or eighth rod signal.

In addition, the fourth rod signal has a "halfway" form where Rod A is retracted away from the nondominant arm.

Lowering both rods is interpreted as an absence of a rod signal.

If the use of rods are unavailable, the numerals of the positions may be pronounced.

1.2 | Hacmisation

As using IPA is quite wieldly, we shall use the following hacmisation, with superscript letters to indicate phonemes not found in Arka.

Rod signs are represented by the hacm digits <1 2 3 4 5 6 7 8 9 Δ> attached to the end of the verbs they encompass. Halfway rod signals are represented by a subscript digit: <9_h>. Transitions from the ninth rod signal are written <L⁹ L^Δ>. Proper words are preceded by a backslash <\>.

Note that the hacmisation is slightly different from Lek-Tsaro's use of hacm. Lek-Tsaro's <h s> are now written using <l^h l^s>, for instance.

Table 1.3: The consonants of Middle Rymakonian in hacm.

	Bilabial	Dental	Alveolar	Palatal	Velar	Glottal
Nasal	ɒ		n	nʲ	nʷ	
Plosive	d b		ɾ ɳ	ɽ ɳʲ	ɭ ɰ	.
Fricative	ɑ u	j ^a z ^u	j z	ɭ s	ɭ ^l ɰ ^s	
(coarticulated)	ɑ ^h u ^h	j ^h z ^h		ɑ ^l u ^s		
(whistled)			j ^o z ^o			
Affricate			β	ɽ		
Lateral fricative			ɭ ^l s ^l			
Approximant			ɹ	ɥ	o	
Lateral approximant			ɭ			
Tap			ɾ			

Table 1.4: The vowels of Middle Rymakonian in hacm.

	Front	Central	Back
High	ɕ	ɜ	ə
Mid	e		ɔ
Low		ɪ	

1.3 | Neđam script

The *Neđam* (neʒʲɪɒ / ʰəʒʲəɒ) script, which is used for Middle Rymakonian, is a boring old alphabet. The consonants and vowels are shown in tables 1.5 and 1.6, respectively. Rod signals and punctuation are shown in table 1.7.

Interestingly, most glyphs are either made entirely of straight lines or made entirely of curves. In particular, consonants that were in Lek-Tsaro are made of straight lines, although it is <j^a> rather than <j^h> that corresponds to Lek-Tsaro's <j^h>. Other consonants, as well as vowels, are made of curves.

Table 1.5: The consonants of Middle Rymakonian in the Neđam script.

	Bilabial	Dental	Alveolar	Palatal	Velar	Glottal
Nasal	ɽ		ɳ	ɽ	ɽ	
Plosive	ɽ ɽ		ɽ ɳ	ɽ ɽ	ɽ ɽ	=
Fricative	ɽ ɳ	ɽ ɽ	ɽ ɽ	ɽ ɽ	ɽ ɽ	
(coarticulated)	ɽ ɽ	ɽ ɽ		ɽ ɽ		
(whistled)			ɽ ɳ			
Affricate			ɽ	ɽ		
Lateral fricative			ɽ ɽ			
Approximant			ɽ	ɽ	ɽ	
Lateral approximant			ɽ			
Tap			ɽ			

Note that two phonemes in the word were metathesised when it was derived from Lek-Tsaro. In general, a word with n spoken phonemes cannot have more than $\lceil n/2 \rceil$ phonoruns. Therefore, the following changes are executed in order until an application of one rule reduces the number of phonoruns to an acceptable number, after which the other rules are not executed:

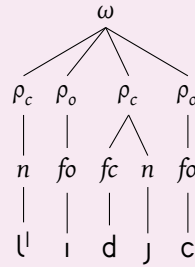
$$\begin{aligned}
 X_1[do]X_2[dc]R[do] &\rightarrow X_2X_1R \\
 X_1[dc]X_2[do]R[dc] &\rightarrow X_2X_1R \\
 X_1[dc]X_2[do]?X_3[do] &\rightarrow X_1?X_2X_3 \\
 X_1[do]?X_2[do]X_3[dc] &\rightarrow X_1X_2?X_3 \\
 X_1[op \geq 0]X_2[dc]X_3[do]X_4[op \leq 0] &\rightarrow X_1X_3X_2X_4 \quad [X_1.op + X_3.op - X_2.op - X_4.op \geq 6] \\
 X_1[op \leq 0]X_2[do]X_3[dc]X_4[op \geq 0] &\rightarrow X_1X_3X_2X_4 \quad [X_2.op + X_4.op - X_1.op - X_3.op \geq 6] \\
 X_1[do]X_2[dc]X_3[do] &\rightarrow X_1X_3X_2 \quad \text{for ever} \\
 X_1[dc]X_2[do]X_3[dc] &\rightarrow X_2X_1X_3 \quad \text{for ever}
 \end{aligned}$$

where R means a rod signal, X represents a spoken phoneme and op stands for *openness* (full-open = 2, neutral = 0, full-closed = -2). do is short for $op > 0$, and dc is short for $op < 0$. (The same rule can occur multiple times within a word, although such invocations may not intersect each other.)

All of the rules above move from right to left and do not occur across compound boundaries. The last two rules are executed in parallel in a loop until the number of phonoruns is reduced to an acceptable number or both rules converge to a fixed point. This process will hereafter be called *phonorun reduction*.

In the example above, $\langle xj^a.en \rangle$ had $4 > \lceil 5/2 \rceil$ phonoruns, so the third rule was applied. This changed the word into $\langle j^a.en \rangle$, which has $2 \leq \lceil 5/2 \rceil$ phonoruns.

An example where phonorun reduction does not result in a word with few enough phonoruns is $\langle l^1dj.c \rangle$ *soup*, which has the starting phonoruns



Obviously, the first four rules do not match anywhere in the word. The sixth rule seems promising because it matches the pattern at $\langle l^1dj- \rangle$, but the required sum is $0 + 2 + 2 + 0 < 6$, so this rule does not match. In addition, the last two rules do not match, and we encounter a fixed point. In such cases, the anomaly is allowed to pass.

The dictionary lists forms of roots *before* the phonorun reduction happens, because affixes can radically affect which phonemes are switched.

1.4.1 | Prosody

The time taken to utter a phonorun is given by the model:

$$t_o = K \cdot (1 + v \cdot \alpha + c \cdot \beta) \quad (\text{phonorun is open}) \quad (1.1)$$

$$t_c = K \cdot \eta \cdot (\gamma + v \cdot \alpha + c \cdot \beta) \quad (\text{phonorun is closed}) \quad (1.2)$$

where K is a constant varying from person to person, v is the number of vowels and c is the number of consonants in the run. α , β , γ and η are also constants such that $\beta < \alpha$, and both γ and η are less than 1. In other words:

- There is a fixed cost for starting a new phonorun. This cost is less for closed phonoruns than open.
- Closed phonoruns are faster to say than open runs with the same number of consonants and vowels.
- Closed phonoruns are also more length-dependent than open runs.
- It takes less time to utter consonants than vowels.

An estimate of the constants for the standard dialect would be $\alpha = 0.37$, $\beta = 0.46$, $\gamma = 0.82$ and $\eta = 0.61$.

1.5 | Vowel harmony

Middle Rymakonian inherits vowel harmony from Lek-Tsaro. Thus $\langle \text{c e} \rangle$ are front vowels, $\langle \text{ə ɔ} \rangle$ are back vowels and $\langle \text{i ɜ} \rangle$ are neutral. Most roots with neither front nor back vowels act as if they had front vowels, though some might behave as if they had back vowels. Many affixes will change depending on which vowels are present.

If by some odd chance a word has both front and back vowels, then the rightmost vowel (before phonorun reduction) takes precedence.

1.6 | Rod signal sandhi

The following rules influence rod signals depending on the previous rod signal (of the current or previous word):

- $\langle \text{ŋ} \rangle$ is realised as $\langle \text{J} \rangle$ after $\langle \text{ŋ} \rangle$ or $\langle \text{ŋ} \rangle$.
- $\langle \text{ŋ} \rangle$ is realised as $\langle \text{L}^{\text{ŋ}} \rangle$ after $\langle \text{ŋ} \rangle$ or $\langle \text{L}^{\text{ŋ}} \rangle$.
- $\langle \Delta \rangle$ is realised as $\langle \text{L}^{\Delta} \rangle$ after $\langle \Delta \rangle$ or $\langle \text{L}^{\Delta} \rangle$.

Rod sandhi does not affect the orthography or phonorun reduction.

2 | Syntax

2.1 | Basic word order

The basic word order is VSO. Descriptors follow what they modify.

However, unlike Lek-Tsaro, Middle Rymakonian has oblique arguments. As these were historically formed from a preclause, all obliques precede V. Likewise, any arguments with conjunctions also precede V. Such arguments that were formed from a clause will be called *historically clausal arguments* (HCAs).

Usually, oblique arguments are prepared by prepositions and fall after what they modify (unless the antecedent is V), but if an oblique argument is a conjunctive phrase or governs an HCA, it uses a postposition instead and precedes its antecedent.

2.2 | Questions

In all questions, the intonation of the second word of the last clause is lowered considerably.

Binary questions have the interrogative polarity marker and no change to syntax.

In wh-questions, the wh-word is pulled to the front (i. e. before the verb). This requires case marking for the wh-word:

ʃezcn ʁeəʒɲɪ dɔʒ
who-ACC speak-FAR.PAST-Q PR.FAR
Whom did you speak to?

This applies only to questions, not interrogative-mood clauses that act as relative clauses:

ʁeəʒɲɪ dɔ ʃel, ɥɪf ʁɔ.
speak-FAR.PAST-Q PR.FAR who, see-NEAR.PAST PR.ANAPH_OBJ
I saw the person whom you talked to.

2.3 | Multiple clauses

A sentence might have multiple clauses. Each clause in a sentence follows the basic VSO order, and clauses are separated with commas.

3 | Nouns

Nouns are declined for number, case and definiteness.

3.1 | Number

Countable nouns come in two numbers: *dual* and *non-dual*.

There are two different conceptualisations of the dual number. Some dialects use the dual number to refer to all cases with two objects (we say that they have the *unpaired dual*); others use it only to refer to objects in pairs (these lack the unpaired dual). In general, dialects without the unpaired dual are more prevalent in cities, as well as northern regions.

Each countable noun has an *inherent number*. A noun whose number agrees with its inherent number receives no marking; a mismatch causes the noun to receive a special affix.

3.2 | Case

In a clause with both the subject and object directly expressed in that order, both the subject and object are declined in the nominative case (and their roles are inferred through word order). In a clause where only one is present, or where both are expressed in the opposite order, the subject will receive the nominative case and the object will receive the accusative case.

3.3 | Noun classes

There are three overarching groups of noun classes.

1. Countable
 - (a) Sentient – such as humans, AIs, deities.
 - (b) Non-sentient – anything else.
2. Measurable
 - (a) Measure – all measurable nouns, especially units of measurement.
3. Uncountable
 - (a) Edible – edible (to humans).

- (b) Inedible – inedible (to humans).
- (c) Abstract – abstract ideas.

3.4 | Definiteness

The definite form of a noun is formed regularly by reduplicating the first syllable (without the coda): <DIZI> “a person” becomes <DIDIZI> “the person”.

3.5 | Declension table

Here, the inflected forms of words are shown both before and after phonorun reduction to illustrate the pattern. The declension patterns for each class is shown, both for roots ending with consonants and those ending with vowels.

Note that noun declensions for countable classes respect vowel harmony. For nouns with back vowels, replace the front vowels with the back vowels of the same height and rounding, and vice versa. (Noun declensions for measurable and uncountable classes do not respect vowel harmony.)

3.5.1 | Countable classes

Table 3.1: Declensions for countable nouns.

	Direct #	Inverse #
Sentient: <xDIZI> “person”		
Nominative	DIZI (DIZI)	DIZI (DIZI)
Accusative	DIZIn (DIZIn)	DIZInIl (DIZInIl)
Sentient: <xj ⁰ .en> “magician”		
Nominative	j ⁰ .en (j ⁰ .en)	j ⁰ .el (j ⁰ .el)
Accusative	j ⁰ .ezcn (j ⁰ .ezcn)	j ⁰ .epcl (j ⁰ .epcl)
(Note that the final consonant is preserved only in the direct nominative form.)		
Non-sentient: <xD3n ⁰ ɔ> “rabbit”		
Nominative	D3n ⁰ ɔ (D3n ⁰ ɔ)	D3n ⁰ ɔ.ə (D3n ⁰ ɔ.ə)
Accusative	D3n ⁰ ɔD (D3n ⁰ ɔD)	D3n ⁰ ɔuə (D3n ⁰ ɔuə)
Non-sentient: <x.cDen> “house”		
Nominative	.cDen (.cDen)	.cDe.c (.cDec.)
Accusative	.cDezCD (.cDezCD)	.cDeɥCUC (.cDeɥCUC)

3.5.2 | Measurable and uncountable classes

Table 3.2: Declensions for measurable and uncountable nouns.

	Direct
Measure: <xɥ3D3> “day (continuous)”	
Nominative	ɥ3D3 (ɥ3D3)
Accusative	ɥ3D3n (ɥ3D3n)
Measure: <xDeI> “volume” (in expressions such as <xDeI-ɥ3ɔ> “cupful”)	
Nominative	DeI (DeI)

	Direct
Accusative	de ^z cn (de ^z cn)
Edible: <xfe _μ .c> “beef”	
Nominative	fe _μ .c (fe _μ .c)
Accusative	fe _μ .cn (fe _μ .cn)
Edible: <xɔɪn> “rice”	
Nominative	ɔɪn (ɔɪn)
Accusative	ɔɪncn (ɔɪncn)
Inedible: <xpəfɔ> “gold”	
Nominative	pəfɔ (pəfɔ)
Accusative	pəfɔbe (pəfɔbe)
Inedible: <xlɔɪɲ> “stone”	
Nominative	lɔɪɲ (lɔɪɲ)
Accusative	lɔɪɲde (lɔɪɲde)
Abstract: <xɑ ^h əɔɔ> “empathy”	
Nominative	ɑ ^h əɔɔ (ɑ ^h əɔɔ)
Accusative	ɑ ^h əɔɔn ^ʰ (ɑ ^h əɔɔn ^ʰ)
Abstract: <xɸɔɔ> “[the number] five”	
Nominative	ɸɔɔ (ɸɔɔ)
Accusative	ɸɔɔzn ^ʰ (ɸɔɔzn ^ʰ)
Here, the final consonant is voiced if it is a fricative.	

(NB: be sure to change any <ɪ> and <ɲ> into <ɪʰ> and <ɲʰ> respectively before <ɔ>.)

3.6 | Pronouns

Personal pronouns are not divided into first, second and third persons as in most languages. Instead, they fall into six categories that exhibit different behaviour depending on whether they occur as the first non-oblique noun in the clause or elsewhere (second noun, verb inflection, oblique):

Table 3.3: Pronoun persons and their functions.

Person	Role in first position	Role elsewhere
Near	The speaker.	The first non-oblique argument of the clause. The person with which the first argument is conversing. An entity that is neither the speaker, the listener nor the first argument.
Far	The listener.	
Other	A third entity.	
Generic	A generic entity (akin to “one”).	
Anaphoric Subject	The subject of the previous clause. Also used on the verb when an oblique or conjunction is present.	
Anaphoric Object	The object of the previous clause.	

In wh-questions, the wh-word assumes the second position and the other argument becomes the first.

If a clause has no explicit arguments, the first argument is understood to be the subject.

Table 3.4: Personal pronouns (before phonorun reduction).

	Nominative		Accusative	
	Non-dual	Dual	Non-dual	Dual
Near	fi	aczc	fin	aczen
Far	dc	bpi	dcn	bpin
Other	nc	lizc	ncn	lizen
Anaph. Sub.	pi	n ⁴ cpc	pin	n ⁴ cpen
Anaph. Obj.	pc	n ⁴ apc	pcn	n ⁴ apcn
Generic	.ə		.ən	

3.6.1 | Last-clause pronouns

The anaphoric pronoun <ebj> (accusative: <bezen>) is grammatically an other pronoun, and it refers to the previous clause said. Likewise, <bdecj> (accusative: <bdecn>) refers to the clause before the previous one. All of these pronouns should undergo phonorun reduction inside a compound.

3.7 | Compounding

Nouns can be compounded together in a head-initial manner. When that happens, only the leftmost noun is the one to be declined.

del-μɜɟɔ-ɑ¹ɜpə-ɸɕɟ
 volume-cup-water-five
 five cupfuls of water

Note that pronouns can modify other nouns, in which personal possession is indicated:

del-μɜɟɔ-ɑ¹ɜpə-ɸɕɟ-fi
 volume-cup-water-five-PR.NEAR.ND
 (arg1)'s five cupfuls of water

Descriptors can also compound on nouns. Unlike in Lek-Tsaro, this is the only way to have descriptors modify nouns.

ɖɪzɪ-lɪəi
 ɖɪzɪ-ləfi
 person-old
 old people

3.8 | Possession

“X’s Y” is translated as <Y=ɖɪ X> (plus phonorun reduction). The possessive construction is also used to create appositives. (Note the head-marking!)

Observe that possession marks the head, and <–Dɪ> is a clitic, not an affix, as in the following example:

D3D3ŋ⁰ɔ–a^lʒpə–Dɪ j^h.ien
 D3D3ŋ⁰ɔ–a^lʒpə–Dɪ j^h.en
 DEF~rabbit-water=GEN magician
 the magician's water rabbit

This construction is also used when compounding would otherwise be used, but the dependent is larger than a single noun or descriptor:

nɣizɪDɪ ɪ.lle an ɲj
 cat=GEN 4096 and two
 4098 cats

4 | Verbs

Verbs are conjugated for person of the subject, tense, polarity and tellicity, in two paradigms. Conjugation respects vowel harmony. In addition, a final <-j> or <-z> in the stem of a first- or second-conjugation verb becomes whistled in the generic form.

The dictionary lists the stem of the verb and the conjugation scheme used.

Table 4.1: Person-tense conjugations for first-conjugation verbs, using <ɔil-> “(S) eats (O)”, before and after phonorun reduction.

	Nonpast	Past
Near	ɔilɪn (ɔilɪn)	ɔilɪf (ɔilɪf)
Far	ɔilɪn (ɔilɪn)	ɔilɜj (ɔilɜj)
Other	ɔilɪ (ɔilɪ)	ɔilɜ (ɔilɜ)
Anaph. Sub.	ɔile (ɔile)	ɔilel (ɔilel)
Anaph. Obj.	ɔilc.e (ɔil.ce)	ɔilc.el (ɔil.cel)
Generic	ɔilc (ɔilc)	ɔilc (ɔilc)

Table 4.2: Person-tense conjugations for second-conjugation verbs, using <nən-> “(S) kills (O), (O) dies”, before and after phonorun reduction.

	Nonpast	Past
Near	nənɪn (nənɪn)	nənɪf (nənɪf)
Far	nənɪn (nənɪn)	nənɜj (nənɜj)
Other	nənɪ (nənɪ)	nənɜ (nənɜ)
Anaph. Sub.	nənɔ (nənɔ)	nənɔl (nənɔl)
Anaph. Obj.	nənə.ɔ (nənə.ɔ)	nənə.ɔl (nənə.ɔl)
Generic	nənə (nənə)	nənə (nənə)

Notes:

- The polarity-tellicity suffix is added after the person-tense ending.
- “Negative atelic” means something akin to “unsuccessfully tried to avoid doing X”.
- The interrogative polarity, in addition to marking questions, is used to mark clauses that may or may not be true but are referred to later in the sentence.
- As an exception, the generic form of <ɣ-> is <ɣə>.

Table 4.3: Person-tense conjugations for third-conjugation verbs, using <μeu-> “(S) spreads (O)”, before and after phonorun reduction.

	Nonpast	Past
Near	μeucn (μeucn)	μeucf (μeucf)
Far	μeuin (μeuin)	μeu3j (μeu3j)
Other	μeui (μeui)	μeu3 (μeu3)
Anaph. Sub.	μeue (μeue)	μeuel (μeuel)
Anaph. Obj.	μeuc.e (μeuc.e)	μeuc.el (μeuc.el)
Generic	μeu3 (μeu3)	μeu3 (μeu3)

Table 4.4: Polarity-telicity suffixes for verbs (before phonorun reduction). The interrogative affix can also follow a negative affix.

	Positive	Negative	Interrogative
Telic	-·	-f ⁴ e / -כ	-ל ¹
Atelic	-DC / -Də	-J	-ל3

Some examples:

דילן ל'רלע ל'כזכ.
eat-NEAR.NONPAST fish flower
Fish eat flowers.

דילן ל'רלע ל'כזכ, דילן חקזי מ.
eat-NEAR.NONPAST fish flower, eat-NEAR.NONPAST cat PR.ANAPH_SUB
Fish eat flowers, and cats eat fish.

דילן ל'רלע ל'כזכ, דילן מילביע.
דילן ל'רלע ל'כזכ, דילן מילביע.
eat-NEAR.NONPAST fish flower, eat-ANAPH_SUB.NONPAST grass-ACC
Fish eat flowers, and they eat grass.
(Grass is inedible to humans, but edible to fish.)

דילנעף ל'כזכ ל'רלע.
דילנל'ע ל'כזכ ל'רלע.
eat-NEAR.NONPAST-NEG flower fish
Flowers don't eat fish.

דפנ נכ ל'פל'פנען, ינן ל' עב.
דפנ נכ ל'פל'פנען, ינן ל' עב.
carry-NEAR.NONPAST PR.OTHER DEF~book, worry-NEAR.NONPAST PR.NEAR
PR.LAST_CLAUSE
He has the book; that worries me.
or: That he has the book worries me.

דפנל' נכ ל'פל'פנען, ינן ל' עב.
דפנל' נכ ל'פל'פנען, ינן ל' עב.
carry-NEAR.NONPAST-Q PR.OTHER DEF~book, worry-NEAR.NONPAST PR.NEAR.INT

PR.LAST_CLAUSE

He might have the book; that worries me.

or: That he might have the book worries me.

4.1 | Aspect

Verbs can also be marked for aspect, either using a rod sign directly on the verb, or a particle with a rod sign, placed anywhere between the verb it modifies and the next verb.

Table 4.5: Aspect markers. Those with hyphens are attached to verb. Those without hyphens are placed as separate particles anywhere after the verb.

Aspect name	Marking	Meaning
Imperfect	–ɿ	An action that is currently going on. Also used to distinguish static actions as opposed to dynamic (e. g. <i>wear</i> as opposed to <i>put on</i>).
Interrupted	ɿʔcɿ	An action that was interrupted.
Perfect	–ɿ	An action that has already finished. Changes present tense to immediate past. Also used to distinguish dynamic actions as opposed to static (e. g. <i>put on</i> as opposed to <i>wear</i>).
Gnomic	–ɿ	A general truth or aphorism, or an action done habitually.
Gnomic dubitative	ɿʔcɿ	A general truth or aphorism that the speaker considers to be false.
Deontic necessity	–ɿ	An action that the speaker insists on happening.
Deontic recommendation	–ɿ	An action that the speaker recommends that happens.
Epistemic necessity	ɿəɸɿ	An action that the speaker infers is happening. (<i>Situational necessitative and potential moods are grouped with their epistemic versions.</i>)
Deontic potential	–ɿ	An action that the speaker permits to occur.
Epistemic potential	ɿəɸɿ	An action that the speaker infers that might happen.
Unexpected	–ɿ	An action that is unexpected (akin to using “but”).
Comparative	deɿ	Indicates an action of greater intensity than what was described in the previous clause.
Nonexclusive subject	ɿʔcɿ	Indicates that the subject comprises not only of what is explicitly mentioned, but also other things.

Aspect name	Marking	Meaning
Nonexclusive object	c^{uo}	Indicates that the object comprises not only of what is explicitly mentioned, but also other things.
Nonexclusive argument	c^{uo}	Combination of both nonexclusive subject and nonexclusive object.
Temporal universal	$-L^q$	The statement is always true (“never true” when negative).
Temporal non-universal	$j^o L^q$	The statement is not always true (“sometimes true” when negative).
Spatial universal	$-L^\Delta$	The statement is true (false) everywhere.
Spatial non-universal	$j^o L^\Delta$	The statement is false (true) somewhere.

An attached rod signal reverts $\langle j^a z^u \rangle$ to $\langle j^h z^h \rangle$, respectively, and might affect phonorun reduction.

An example:

$\text{f}^{\text{ilil}}\text{f}^{\text{dc1}} \text{fi } nc, \text{lcnc.elf } d3n^{\text{u}}\text{iu}\text{a}-\mu.$
 $\text{f}^{\text{ilil}}\text{f}^{\text{dc1}} \text{fi } nc, \text{lcnc.elf } d3n^{\text{u}}\text{iu}\text{a}-\mu.$
 fight-NEAR,PAST-ATELIC-IMPERFECT PR.NEAR PR.OTHER, shoot-ANAPH_OBJ.PAST-UNEXPECTED knee-INV.ACC-PR.ANAPH_SUB
 I tried to fight them, but they shot my knee.

4.1.1 | Simultaneous temporal and spatial aspects

A verb may be modified by both temporal and spatial aspects, in which case their mutual order is significant:

Table 4.6: Behaviour when both temporal and spatial markers exist, where t is a time variable and \vec{x} is a space variable.

Marking	Definition	Equivalent
$-L^{\Delta q}$	$\forall t \forall \vec{x} : P(t, \vec{x})$	$\forall t \forall \vec{x} : P(t, \vec{x})$
$-L^\Delta j^o L^q$	$\neg \forall t \forall \vec{x} : P(t, \vec{x})$	$\exists t \exists \vec{x} : \neg P(t, \vec{x})$
$j^o L^{\Delta q}$	$\forall t \neg \forall \vec{x} : P(t, \vec{x})$	$\forall t \exists \vec{x} : \neg P(t, \vec{x})$
$j^o L^\Delta j^o L^q$	$\neg \forall t \neg \forall \vec{x} : P(t, \vec{x})$	$\exists t \forall \vec{x} : P(t, \vec{x})$
$-L^q \Delta$	$\forall \vec{x} \forall t : P(t, \vec{x})$	$\forall \vec{x} \forall t : P(t, \vec{x})$
$-L^q j^o L^\Delta$	$\neg \forall \vec{x} \forall t : P(t, \vec{x})$	$\exists \vec{x} \exists t : \neg P(t, \vec{x})$
$j^o L^q \Delta$	$\forall \vec{x} \neg \forall t : P(t, \vec{x})$	$\forall \vec{x} \exists t : \neg P(t, \vec{x})$
$j^o L^q j^o L^\Delta$	$\neg \forall \vec{x} \neg \forall t : P(t, \vec{x})$	$\exists \vec{x} \forall t : P(t, \vec{x})$

4.2 | Historically clausal arguments

Historically clausal arguments (HCAs) are arguments of a sentence that are derived from clausal constructions. They include obliques and conjunctions. HCAs precede V.

An HCA that modifies a verb causes it to be conjugated in the anaphoric subject person.

4.2.1 | Obliques

An oblique expresses a relation between the verb of a sentence or some argument thereof.

An oblique phrase that modifies a verb falls before it. An oblique phrase that modifies either S or O pulls it before the verb as well.

If the argument of the oblique phrase is not an HCA, then it uses a preposition and follows its antecedent (unless it is the main verb). If the argument is an HCA, then the phrase uses a postposition and precedes its antecedent.

Consider the preposition <ɪn> *in, on, at (location)* (from Lek-Tsaro <ɪn> (S) *is at* (O)). The sentence *Ryze is hiding from me in the tree* would be translated as:

ɪn ʃuɔl neɹae1 ɪn ʔɜze
in tree hide-ANAPH.SUB.NONPAST-IMPERFECT PR.NEAR.ACC Ryze

Now say that we want to translate *Ryze is hiding from me in the tree with fruit*. *With* would be translated as <ɖɹ> (from Lek-Tsaro <ɖɹɪn> *hold, carry*, which also begets <ɹɪn>), but now we have nested obliques, which means we need to use <ɪn> as a postposition:

ʃuɔl ɹɪn ʔɜze ɪn neɹae1 ɪn ʔɜze
tree with fruit in-POST hide-ANAPH.SUB.NONPAST-IMPERFECT PR.NEAR.ACC Ryze

Deriving a postposition from a preposition is done *after* phonorun reduction. Prepositions that end with a closed phonorun receive <-ɪ>, and those that end with an open phonorun receive <-z>.

The prefix <ɪʔ-> negates an adposition.

4.2.2 | Conjunctions

Conjunctions are derived from verbs as well; for instance, <an> *and* is derived from Lek-Tsaro <acn> *join*. However, in Middle Rymakonian, conjunctions are infixes:

ʔɜze an ʋɪzɪl ɖɪl ʃɛɹ..
ʔɜze an ʋɪzɪl ɖɪl ʃɛɹ.c.
Ryze and Tazyl eat-ANAPH.SUB.NONPAST beef

(Note that as long as S still precedes O, no case marking is needed.)

Unlike Lek-Tsaro's approach, this approach works well with more complex sentences:

ʔɜze an ʋɪzɪl ʃɛɹ. an ɭɪdʒɪ ɖɪl.
ʔɜze an ʋɪzɪl ʃɛɹ.c an ɭɪdʒɪ ɖɪl.
Ryze and Tazyl beef and soup eat-ANAPH.SUB.NONPAST

An entire conjunctive phrase can be modified by treating the conjunction as a nominal antecedent:

ɳɥɪzɪ an-ɭɪl ɖɜɳɔ
ɳɥɪzɪ an-ɭɪl ɖɜɳɔ

Clauses of a connector set are joined by the relation of the connector used therein:

דויל ל'רדע ר'צזז יעי.
 דויל ל'רדע ר'צזז יעי.
 eat-NEAR.PAST fish flower ORDINARY-⟨e⟩-0
 The fish ate the flower.

 יעי נדדדל ר'רעזע ר'טזל.
 ORDINARY-⟨e⟩-0 dance-NEAR.PAST child tree
 Then the child danced around the tree.

 יעי דילל ל'ל'רדע.
 eat-ANAPH_SUB.PAST ORDINARY-⟨e⟩-0 DEF~fish-ACC
 Then the child ate the fish.

 דעלדלל פ'פ'נ' יעז דו.
 דעלדלל פ'פ'נ' יעז דו.
 imitate-NEAR.PAST-IMP frog ORDINARY-⟨e⟩-1 PR.FAR
 At another time, a frog was imitating me. (...)

4.4 | Comparatives

The comparative is a function $\text{cmp} : A \times A \times (A \rightarrow \mathbb{R}) \times (A \times A \rightarrow \{0, 1\}) \rightarrow \{0, 1\}$, where $\text{cmp}(a, b, f, \sqsupset) = f(a) \sqsupset f(b)$.

Consider the following sentences:

Fish eat flowers more than cats.
 More fish eat flowers than cats.

Semantically, they can be translated to:

$$\text{cmp}(\text{fish}, \text{cats}, a \mapsto (\# \text{ of flowers eaten by } a), >) \quad (4.1)$$

$$\text{cmp}(\text{fish}, \text{cats}, a \mapsto (\# \text{ of } a \text{ that eat flowers}), >) \quad (4.2)$$

The heart of comparatives in Middle Rymakonian is the quadrivalent verb $\langle \text{ר'צזז } a \text{ } b \text{ } f \sqsupset \rangle$. Thus:

דוילל ר'צזז-י'ר'נ, ר'צזז ל'רדע נ'פ'זי פ'נ נעל.
 eat-GENERIC-Q flower-ACC-how_many, CMP-NEAR fish cat PR.ANAPH_OBJ >
 Fish eat more flowers than cats.

 דוילל .א-י'ר'נ ר'צזז, ר'צזז ל'רדע נ'פ'זי פ'נ נעל.
 eat-GENERIC-Q PR.GENERIC-how_many flower, CMP-NEAR fish cat PR.ANAPH_SUB >
 More fish eat flowers than cats.

Note that we place a clause whose argument is the generic pronoun before the comparative clause. From the dozan-clause, we refer to the function using the anaphoric pronoun referring to the position of the return value.

the following clause uses an anaphoric subject, it refers to the object of the current clause.

Moreover, the verb does not need to be one that can never take an object. In the above example, <ᵀᶜᵣᵈᶜᵢ> means “(S) falls on (O)”. However, if the verb in question is taking an object, it cannot be transitivised directly and a more roundabout way is required:

ᵀᶜᵣᵈᶜᵢ ᵃᵂᵉᵃᵂᵉᵃᵂᵉ ᵂᵢᵢᵢ.

fall-NEAR.PAST DEF~coin grass

The coins fell on the grass.

ᵢᵢ ᶜᵂᵀᶜᵣᵈᶜᵢ ᵃᵂᵉᵃᵂᵉᵃᵂᵉ, ᵀᶜᵣᵈᵉᵢ ᵂᵢᵢᵢᵇᵉ.

ᵢᵢ ᵂᶜᵀᶜᵣᵈᶜᵢ ᵃᵂᵉᵃᵂᵉᵃᵂᵉ, ᵀᶜᵣᵈᵉᵢ ᵂᵢᵢᵢᵇᵉ.

PR.NEAR TRANS-fall-OTHER.PAST DEF~coin, fall-ANAPH_SUB.PAST grass-ACC

I dropped the coins; they fell on grass.

or: I dropped the coins on grass.

4.7 | The copula

The copula <ᵢ-ᵉ> (v3) can take a noun as an object, in which case it can mean identity or membership. (Location is expressed with <ᵢ-ᵉ> (v1) “be at”.) With no object at all, it is used to denote existence.

It can also accept a descriptor, in which case the descriptor is attached before <ᵢᶜᵢ> in the dictionary form. (This precedes phonorun reduction but forms a compounding boundary.)

5 | Descriptors

Descriptors act as adjectives or adverbs. They follow what they modify, and are inflected for the verbal person of their antecedents.

Modifying nouns is done through compounding, but there are special forms for modifying verbs. These are separate words.

Table 5.1: Descriptor declensions, using the descriptors <ᵐᵉᵈᶠ-⟩ “large” and <ᵐᵉᶠ-⟩ “old”.

Person	Declined form	
<i>Nouns</i>	ᵐᵉᵈᶠᵐ (ᵐᵉᵈᵐᵐ)	ᵐᵉᶠᵐ (ᵐᵉᶠᵐ)
Near	ᵐᵉᵈᶠᵐᵐ (ᵐᵉᵈᵐᵐᵐ)	ᵐᵉᶠᵐᵐ (ᵐᵉᶠᵐᵐ)
Far	ᵐᵉᵈᶠᵐᵐᵐ (ᵐᵉᵈᵐᵐᵐᵐ)	ᵐᵉᶠᵐᵐᵐ (ᵐᵉᶠᵐᵐᵐ)
Other	ᵐᵉᵈᶠᵉᵐᵐ (ᵐᵉᵈᵉᵐᵐᵐ)	ᵐᵉᶠᵉᵐᵐ (ᵐᵉᶠᵉᵐᵐ)
Anaph. Sub.	ᵐᵉᵈᶠᵐᵐᵐ (ᵐᵉᵈᵐᵐᵐᵐ)	ᵐᵉᶠᵐᵐᵐ (ᵐᵉᶠᵐᵐᵐ)
Anaph. Obj.	ᵐᵉᵈᶠᵉᵐᵐᵐ (ᵐᵉᵈᵉᵐᵐᵐᵐ)	ᵐᵉᶠᵉᵐᵐᵐ (ᵐᵉᶠᵉᵐᵐᵐ)
Generic	ᵐᵉᵈᶠᶜᵐᵐ (ᵐᵉᵈᶜᵐᵐᵐ)	ᵐᵉᶠᵉᵐᵐᵐ (ᵐᵉᶠᵉᵐᵐᵐ)

Note that a final <-j> or <-z> in a stem becomes whistled in the generic form.

6 | Tree mode

Anaphoric referents in a linked-list sentence are sometimes insufficient for expressing complex sentence structures. While the easiest method of resolving this issue is using definite nouns, Middle Rymakonian also provides a mode where sentences are not linked lists of clauses, but rather (binary) trees.

6.1 | Activation

Tree mode is enabled automatically when the treeing particle <ᵿᵿ> is used, and disabled at the end of a sentence.

6.2 | Branch-switching

The aforementioned particle <ᵿᵿ> marks the beginning of the right branch of the tree. The right branch is ended by the particle <ᵿᵿΔ>, which causes the next clause to join the left and right branches.

(N. B. <ᵿᵿ> and <ᵿᵿΔ> can occur only between clauses. If the particles are represented by left and right brackets, respectively, then the brackets should match.)

6.3 | Anaphoric pronouns in joiner clauses

In clauses that join two branches, anaphoric pronouns require marking whether the antecedent occurs in the left predecessor <ᵿᵿ> or the right predecessor <ᵿᵿΔ>. This is done by marking the pronoun with <-ᵿ> or <-Δ>.

Likewise, verbs can be modified with <-ᵿ> or <-Δ> to indicate which branch the subject came from.

6.4 | Errors

The following are ungrammatical:

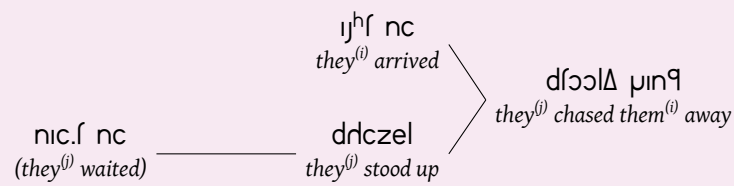
- Using the particle <ᵿᵿΔ> or the branched anaphoric pronouns when tree mode is disabled
- Using the particle <ᵿᵿΔ> other than to close a corresponding <ᵿᵿ>
- Using the unbranched anaphoric pronouns in clauses with two predecessors

- Using the branched anaphoric pronouns in clauses with one predecessor
- Starting a new branch with <ᲛᲣᲗᲚ> when the current branch is empty

6.5 | Example

ᲓᲣᲗᲚ ᲛᲥ, ᲛᲣᲗᲚ ᲛᲥᲥ ᲛᲥ, ᲔᲔᲥᲥᲗᲗᲗ, ᲛᲣᲗᲚ ᲔᲔᲥᲥᲗᲗᲗ ᲡᲗᲛᲗ.
 ᲓᲣᲗᲚ ᲛᲥ, ᲛᲣᲗᲚ ᲛᲣᲥᲥ ᲛᲥ, ᲔᲔᲥᲥᲗᲗᲗ, ᲛᲣᲗᲚ ᲔᲔᲥᲥᲗᲗᲗ ᲡᲗᲛᲗ.
 go-NEAR.PAST PR.OTHER, BRANCH wait-NEAR.PAST PR.OTHER, stand_up-
 ANAPH_SUB.PAST,JOIN chase-ANAPH_SUB.PAST-RIGHT PR.ANAPH_SUB.ACC-LEFT
 When they⁽ⁱ⁾ arrived, they⁽ⁱ⁾ stood up and chased them⁽ⁱ⁾ away.

The resulting tree is shown below:



7 | Numerals

Unlike Lek-Tsaro, which used a downright unusual numbering system, Middle Rymakonian uses base 16 consistently.

7.1 | Irregular numerals

Here are the numerals that do not follow the usual pattern, before phonorun reduction:

Table 7.1: Irregular numerals.

base 10	base 16	word
0	0	μ3D
1	1	a3l
2	J	fiJ
3	?	Ω ⁴ on
4	0	a ⁴ μ
5	†	φCJ
6	?	Dye
7	9	βCJ
8	Δ	dən
9	L	l ^l ed
10	F	b3n ⁹
11	7	nə
12	£	le
13	#	j ^a cd
14	A	yin
15	V	f ⁴ el
16	10	.μ
17	11	l ^l el
18	1J	l ^l ela3l
19	1?	l ^l elfiJ
33	J1	aDil
34	JJ	Dil
119	99	Dlβ
256	100	f ⁴ l ^l a3
323	10?	l ^l izilfin

base 10	base 16	word
4199	10f9	l.ɔlə

Note that digits above 9 use capital haem letters.

7.2 | Double-digit numerals

Numerals of the form $x \cdot 16$ with $1 \leq x < 16$ are formed by concatenating $\langle .\mu \rangle x$. For instance, $128 = 80_{16}$ is written $\langle .\mu d \rangle n \rightarrow \langle .\mu \partial n \rangle$.

Numerals for integers of the form $x \cdot 16 + y$ with both x and y between 1 and 15, inclusive, and not listed in table 7.1, are formed by concatenating $x \langle \mu \rangle y \langle \mu \rangle$ (before PR). For instance, $89 = 59_{16}$ is written $\langle \phi c j \mu l \rangle e d \mu \rightarrow \langle \phi c j \mu l \rangle d e \mu$.

7.3 | Numerals up to 4096

Numerals for integers of the form $x \cdot 256 + y$ with $0 \leq x < 16$ and $0 \leq y < 256$, and not listed in table 7.1, are formed by concatenating $y \langle \mu l \rangle a \rangle x$. This is done after phonorun reduction. For instance, $2018 = 7E2_{16}$ is written $\langle \mu l \rangle a \rangle c \beta j$.

Note that there is no special case for $y = 0$; $512 = 200_{16}$ is written $\langle \mu \partial \mu l \rangle a \rangle n$.

7.4 | Larger numerals

Multiples of 4096 (up to 65536) are written by concatenating $\langle .l \rangle x$ before phonorun reduction: $8192 = 2000_{16}$ is written $\langle .l \rangle e l j$. The exception is 4096 itself, which is $\langle .l \rangle l e$.

Then other numerals up to 65536 are written as a conjunctive phrase: $10000 = 2710_{16}$ is written $\langle .l \rangle e l j \text{ an } .\mu l \rangle a \rangle c \beta j - 2 \cdot 4096 + 16 + 7 \cdot 256$.

8 | Derivational morphology

The following methods are used to derive related terms from existing ones.

8.1 | Abstraction

Abstraction is a derivation that takes a non-abstract noun and returns the abstract noun representing the concept of the argument. Before pronoun reduction, this formation appends $\langle -ne \rangle$ or $\langle -n \rangle$ to the noun. In addition, any final fricatives or lateral fricatives after a vowel are voiced, and a final $\langle \mu \rangle$ after a vowel is changed to $\langle z \rangle$.

Examples:

- $\langle l^l \mu c n \rangle$ *book* \rightarrow $\langle l^l \mu c n e \rangle$ *literature*
- $\langle l^l \nu \rangle$ *cart* \rightarrow $\langle l^l \nu n \rangle$ (\rightarrow $\langle l^l \nu n \rangle$) *transportation*
- $\langle \nu c \mu \rangle$ *hand* \rightarrow $\langle \nu c z n e \rangle$ *technique*

8.2 | Dematuration

Dematuration is a derivation that takes a noun and returns a noun of the same class that represents an immature form of the argument (not necessarily a diminutive). $\langle l^l c - \rangle$ or $\langle f \theta - \rangle$ are prepended to nouns that begin in $\langle j \rangle$, $\langle l \rangle$ or $\langle j^o \rangle$, or $\langle j c - \rangle$ or $\langle j \theta - \rangle$ otherwise. In addition, an initial fricative or lateral fricative before a vowel is voiced, and an initial $\langle \mu \rangle$ before a vowel is changed to $\langle z \rangle$.

Examples:

- $\langle \nu \nu \nu \rangle$ *person* \rightarrow $\langle j c \nu \nu \nu \rangle$ *child*
- $\langle l^l \nu z \theta \rangle$ *fruit* \rightarrow $\langle j \theta l^l \nu z \theta \rangle$ (\rightarrow $\langle j \theta l^l \nu z \theta \rangle$) *unripe fruit*
- $\langle j e r l i c n \rangle$ *essay* \rightarrow $\langle l^l c z e r l i c n \rangle$ *draft*

8.3 | Verb-to-noun conversions

To derive a noun from a verb, an affix is added to the verb stem:

Table 8.1: Conversion affixes.

Name	Affix
Agent	–en ^ϕ / –ɔn ^ϕ
Patient	–ed / –ɔd
Location	–eɥ / –ɔɥ
Instrument	–ɪʃ
Causer	–en ^ɥ d / –ɔɸd

Then the resulting word is declined as an abstract noun, and phonorun reduction happens. After phonorun reductions, the order of phonoruns is reversed, such that the last phonorun becomes the first, for instance. Finally, the final phonorun is continued by appending <–c> or <–ʃ>.

The following words are derived from <nəb–> (v2) *to steal*:

- Agent: <nəbɔn^ϕ> → <ɔn^ϕbnəc> *thief*
- Patient: <nəbɔd> → <nəɔbd> → <bdnəɔc> *stolen goods*
- Location: <nəbɔɥ> → <ɔɥbnəc> *site of theft*
- Instrument: <nəbɪʃ> → <nəɪbʃ> → <bʃnəɪc> *tools used for theft*
- Causer: <nəbɔɸd> → <nəɔbɸd> → <bɸdnəɔc> *person or factor that caused the theft*

Occasionally, a word derived by this method might become lexicalised. In that case, it moves to the noun class of best semantic fit and its declension is regularised (based on the nominative form).

8.4 | Verb-to-verb conversions

The main productive verb-to-verb conversion is the *immediate reversal*, which is expressed with an infix <–ɪDZ–> or <–dɜn–> immediately before the ending, depending on whether the last phonorun of the stem (before phonorun reduction) is open or closed, respectively. This conversion is often associated with the lack of volition.

- <nəD–> *sleep* → <nəDɪDZ–> *be woken up forcibly* (→ <ɸcnəDɪDZ–> *wake someone up forcibly*)
- <Dez–> *ride, board* → <DezɪDZ–> *be forcefully ejected from a ride* (→ <ɸcDezɪDZ–> *eject someone from a ride*)
- <nəɸa–> *hide* → <nəɸadɜn–> *be uncovered* (→ <ɸcnəɸadɜn–> *uncover something*)

9 | Names

Names fall into two grammatical categories:

- *Nominal names* act as nouns. They are usually single words.
- *Clausal names* are entire clauses. These names usually refer to places, although a few people have clausal names. In extreme cases, such a name can span multiple clauses.
- *Station names* are used for places where a medium of transportation stops to exchange passengers according to a regular schedule (e. g. a train station or a bus stop). These are verbs.

9.1 | Nominal names

These names act as nouns, and they are preceded by a backslash <\>. If the name spans multiple words (as common in foreign names), spaces are escaped by backslashes. No distinction is made between native and foreign names.

Only personal names can stand on their own, and even then, only given or full names. Other names must modify a common noun describing the nature of what is named, in the integral number without definiteness.

Table 9.1: Some examples of nominal names.

Name	Type
\p3ze	Personal (native)
\i33l	Personal (native)
\pedcn	Personal (foreign)
dpe3e-\oil.c	Place (foreign)

Native names will usually respect vowel harmony. Children of parents who work in professions demanding physical labour (e. g. bricklaying) will usually have names with back vowels. In contrast, those born to parents of professions that do not demand physical strength (e. g. computer programming) will usually bear names with front vowels.

In addition, native names tend to undergo sound changes from Lek-Tsaro to Middle Rymakonian, but foreign names given when Lek-Tsaro was still spoken retain Lek-Tsaro forms, but with the following vowel replacements (and, of course, orthographic changes) – see table 9.2.

Table 9.2: Lek-Tsaro to Middle Rymakonian correspondences for foreign names.

Middle Rymakonian	From these vowels in Lek-Tsaro
ɪ	ɪ ɔ ^e
e	e e ^ɔ ɔ ^ɔ
c	c c ^a
ɔ	ɔ ə ^ɔ
ə	ə ^c ə

In phonorun reduction, foreign names are not affected by metathesis.

9.2 | Clausal names

These names comprise of one or more clauses. Due to the nature of clausal names, they are all considered native. However, almost all except the newest clausal names are frozen and might not be valid clauses in Middle Rymakonian; usually, they are Lek-Tsaro clauses with the vowel replacements outlined in table 9.2.

Most of these names refer to places; personal clausal names are almost always nicknames or such. Orthographically, they are put into square brackets <[]>.

Clausal names are used by saying them as their own clauses, then using an anaphoric pronoun to backreference the entity described by the name in question. The type of anaphoric pronoun used varies from name to name. It might be the anaphoric subject pronoun, the object pronoun or the last-clause pronoun.

We call the *referent* the subject, the object or the verb of the last clause, respectively depending on the type of anaphoric pronoun used to refer to the name. If the referent is a noun, it must be declined in the integral number without definiteness.

Here, as common in maps and such, the referent will be capitalised. However, other contexts that make the type of anaphoric pronoun to use clear do not use this type of capitalisation.

Table 9.3: Some examples of clausal names.

Name	Type	Literal meaning
[AEX4€ dɪɪʔɪ (ɔaɔɪ)]	Place	The trees covered the ground
[Dəʔɪ ɸXEDƏ ɲɔɪ-\\lenɲe]	Place	The city remembers the Šedri (Šedrŷ) star
[ac ɲɪfə-Dɪ jɔɪ jəɲ jʰɪ.en-Dɪ jɔɪ ɲɪne, ɸoeɲc.ɛɪ ɸXEDƏD]	Place	The city was founded by the warrior of the sun and the wizard of the moon
[Dezi \\ʦETɔF jɔɪ-l'el]	Personal	Gulto takes care of 17 foxes

An example of usage:

[Dəʔɪ dɲede ɲɔɪ-\\lenɲe], n.ɪ bɪnen-cɪɲɪ jʰe dɪdɪzɪ ɲɪ.
 (name), in_time year-next go-ANAPH_SUB DEF~person PR.ANAPH_SUB
 He will go to Muta Pröme Ryk-Šedrŷ next year.

9.3 | Station names

These names describe places where a medium of transportation stops to exchange passengers according to a regular schedule (e. g. a train station or a bus stop). Station names are (usually first- or second-conjugation) verbs whose base meaning is (S) *goes to \$station via \$transportation*. They are marked with a per cent sign <%> before the name.

Unlike with other verbs, the immediate reversal does not necessarily suggest a lack of volition. The reversal of a station name, rather, simply means (S) *goes from \$station*.

The other derivations have the following meanings:

Table 9.4: Derivations of station names.

	(base)	Immediate reversal
(base verb)	(S) goes to \$station	(S) goes from \$station
Agent	A passenger going to \$station	A passenger going from \$station
Patient	(undefined)	
Location	The \$path to \$station	The \$path from \$station
Instrument	The \$transportation going to \$station	The \$transportation going from \$station
Causer	The driver of said \$transportation	

In addition, the aspect marker <–J> changes the meaning from (S) *goes from \$station* to (S) *boards the \$transportation to \$station*. Similarly, combining both the immediate reversal infix and the perfect aspect marker yields (S) *boards the \$transportation from \$station*.

There are several ways a station name can be derived:

- From a nominal name: if it does not end in a verbal affix, then one is attached: e. g. <dpede–\oili.c> → <%oili.–> (v3).
- From a clausal name wherein the referent is S or O: Let C be the nonreferent among S and O (or empty if none). Then the station name comes from a compound of C-V: <[dafi φXEDE μcl–\lenpe]> → <%μcl–\lenpe–dafi–> (v2). (In a name with multiple clauses, ignore those that do not contain the referent.)
- From a clausal name with V as the referent: If neither S nor O exists, let C be empty. If S xor O is a “common word”, then let C be the one that is not common. Otherwise, let C be S. Then the station name comes from a compound of C-V: <[AEXψ€ dñlñl fcaɔ]> → <%dñlñl–nepa–> (v3).

9.3.1 | Common words

“Common words”, in the context of deriving station names, include:

- Numerals
- Sentient nouns that are neither personal names nor modified by <=Dl>
- <dñlñl> *ground*, <fcaɔ> *tree*, <lñlñ> *stone* or <f'izc> *river*, as well as their Lek-Tsaro equivalents in frozen names, not modified by any nominal names (but names for a specific kind of tree, etc., e. g. <ncja> *birch* are not common words)

10 | Calendar

Domain II, which contains *Rymako*, has a day that is 26.99410 hours long. Other figures are given in terms of local days:

Table 10.1: Astronomical measures for Domain II.

Period	Length in local days
Local (synodic) day	1.00000
Sidereal day	0.99699
Tropical year (l_y)	301.94714
Sidereal year	302.03719
Synodic month (l_m)	30.80152
Sidereal month	27.95032

10.1 | Tides

In Domain II, the offset of the sea level due to the tide can be modeled by the following equations:

$$y = y_s + y_m \quad (10.1)$$

$$y_s = A_s \cdot (1 + A_{sa} \cdot \cos(\tau \cdot t)) \cdot \cos(2 \cdot \tau \cdot t) \quad (10.2)$$

$$y_m = A_m \cdot \left(1 + A_{ma} \cdot \cos\left(\frac{\tau \cdot t}{l_m}\right)\right) \cdot \cos\left(\frac{2 \cdot \tau \cdot (1 - l_m) \cdot t}{l_m}\right) \quad (10.3)$$

where:

$$\tau = 2 \cdot \pi$$

$$A_s \approx 0.675$$

$$A_{sa} \approx 0.0532$$

$$A_m \approx 1.267$$

$$A_{ma} \approx 0.176$$

y = offset of sea level in metres

t = time since HAT in local synodic days

An exact solution to $dy/dt = 0$ is not known to exist. However, the solutions to this equation can be found numerically. Consult Section A.1 for a Sage program to do so.

As the calendar used by Middle Rymakonian uses the high and low tides to count time, it is not synchronised even with days. The basic unit of time in the calendar is the *tidal day* $\langle \mu 3 \mathfrak{D} 3 \rangle$ (l_t) – the amount of time between a high tide and the second high tide thereafter, which is, on average, 1.03356 local synodic days, but can vary considerably. Thus:

$$l_m/l_t \approx 29.80148 \quad (10.4)$$

$$\approx 4053/136 \quad (10.5)$$

$$l_y/l_m \approx 9.80299 \quad (10.6)$$

$$\approx 7215/736 \quad (10.7)$$

This suggests that:

1. most months will have 30 days, but every 136 months, 27 months will have only 29.
2. most years will have 10 months, but every 736 years, 145 years will have only 9.

10.2 | Months

Months follow a 136-month cycle wherein the $5n + 2$ -numbered months (zero-indexed, $n \in \mathbb{N}$) have 29 days and the other months have 30.

The names of months, on the other hand, are determined from their positions relative to the first month of the year:

Table 10.2: The months of the year.

#	Name
0	æbæ
1	μ3.εβ
2	βιφɔ
3	ɔzɜɪ
4	ɪɹezɪɔ
5	ɪɔɔɜɪ
6	nʰɜzɜɪ
7	ɪɔɔɪ
8	ɔɔɔɪ
9*	.ɔzeɪ

10.3 | Days within a month

The first nine tidal days have their own names, as follow:

Table 10.3: Day names.

Day #	Year number plus month number	
	Even	Odd
1	j ^o əpɪl	jɪlɔ.ə
2	zɪlɪ	ɑʒl.cneɪ
3	ɾcəlɪ	l'ɪzɪl
4	ɔɪzɛɸ ^s c	ɪ'ɪɟɛz
5	dɪlɔ	nɟɛz
6	ɯ ^h ɪpɪl	ucnɪ
7	ɪɪɪɪc	dec.ʒɑ
8	ɔcɪɪ ^ɶ	ɪɑɪɪl
9	ɪɪz ^h əɪɔ	.czɔɔ

The days after those are referred to by number (one-indexed): the thirteenth day would be called $\langle \mu 3 D 3 - j^a c d \rangle$.

10.4 | Years

The lengths of the year follow a 736-year cycle as specified in Figure 10.1. The code used to generate this table can be found in Section A.2.

Figure 10.1: Table of year lengths in a cycle.

0	123456789		0123456789		38	9XXXX9XXXX		57	XXX9XXXX9X	
1	9XXXX9XXXX		20	XXX9XXXX9X		39	9XXXX9XXXX		58	XXX9XXXX9X
2	9XXXX9XXXX		21	XXX9XXXX9X		40	9XXXX9XXXX		59	XXX9XXXX9X
3	9XXXX9XXXX		22	XXX9XXXX9X		41	X9XXXX9XXX		60	XXXX9XXXX9
4	9XXXX9XXXX		23	XXX9XXXX9X		42	X9XXXX9XXX		61	XXXX9XXXX9
5	9XXXX9XXXX		24	XXX9XXXX9X		43	X9XXXX9XXX		62	XXXX9XXXX9
6	9XXXX9XXXX		25	XXX9XXXX9X		44	X9XXXX9XXX		63	XXXX9XXXX9
7	X9XXXX9XXX		26	XXX9XXXXX9		45	X9XXXX9XXX		64	XXXX9XXXX9
8	X9XXXX9XXX		27	XXX9XXXXX9		46	X9XXXX9XXX		65	XXXX9XXXX9
9	X9XXXX9XXX		28	XXX9XXXXX9		47	XX9XXXX9XX		66	XXXX9XXXXX
10	X9XXXX9XXX		29	XXX9XXXXX9		48	XX9XXXX9XX		67	9XXXX9XXXX
11	X9XXXX9XXX		30	XXX9XXXXX9		49	XX9XXXX9XX		68	9XXXX9XXXX
12	X9XXXX9XXX		31	XXX9XXXXX9		50	XX9XXXX9XX		69	9XXXX9XXXX
13	X9XXXX9XXX		32	XXX9XXXXX9		51	XX9XXXX9XX		70	9XXXX9XXXX
14	XX9XXXX9XX		33	XXXX9XXXXX		52	XX9XXXX9XX		71	9XXXX9XXXX
15	XX9XXXX9XX		34	9XXXX9XXXX		53	XX9XXXX9XX		72	9XXXX9XXXX
16	XX9XXXX9XX		35	9XXXX9XXXX		54	XX9XXXX9XX		73	9XXXX9
17	XX9XXXX9XX		36	9XXXX9XXXX		55	XX9XXXX9XX			
18	XX9XXXX9XX		37	9XXXX9XXXX		56	XX9XXXX9XX			

9: 9 months
X: 10 months

10.5 | Eras

Years are grouped further into *eras* $\langle \text{לִּיָּד} \rangle$, which change on major historical events. The start of a new era resets the month and year cycle. Eras can also start in the middle of a year of the previous era; thus, the start of the year is different for each era. The *crossover date* of an era is the date of the era that coincides with the first day of the next; in other words, it is the date immediately after the last day of the era.

Table 10.4: The months of the year.

Name	Crossover date	Days between	Cumulative
לִּיָּד-טַבִּי	אֶלְדֵּי־טַבִּי		
לִּיָּד-לִּיָּד	אֶלְדֵּי־לִּיָּד	889726	889726
לִּיָּד-לִּיָּד	אֶלְדֵּי־לִּיָּד	642508	1532234
לִּיָּד-לִּיָּד	אֶלְדֵּי־לִּיָּד	207366	1739600
לִּיָּד-לִּיָּד	(to present)		

The first day of $\langle \text{לִּיָּד-לִּיָּד} \rangle$ coincides with the founding of the (not yet named).

11 | Quoting direct speech

Direct speech is not quoted as-is, but rather it is converted into a stack code. A *quotative* begins with $\langle \text{ld}\alpha \rangle$. Quotatives are not affected by phonorun reduction.

11.1 | Representation of binary data

Nine bits are represented with a syllable. Consider $0 \leq n < 512$. Let $r = n \bmod 6$ and $q = \lfloor n/6 \rfloor$. Then r corresponds to the vowel $g(r) = (\text{i}, \text{e}, \text{a}, \text{o}, \text{u}, \text{y})[r]^1$.
 q corresponds to one of 86 consonant clusters. Let

$$P = (\text{f}, \text{j}, \text{d}, \text{a}, \text{l}, \text{l}, \text{a}, \text{b}, \text{z}^{\text{o}}, \text{u}, \text{p}, \text{a}^{\text{h}}, \text{j}^{\text{h}}, \text{a}^{\text{l}}) \quad (14 \text{ entries}) \quad (11.1)$$

$$N = (\text{n}, \text{n}^{\text{u}}, \text{d}, \text{n}^{\text{p}}) \quad (4 \text{ entries}) \quad (11.2)$$

$$L = (\text{p}, \text{l}, \text{u}, \text{o}) \quad (4 \text{ entries}) \quad (11.3)$$

$$M = (\text{l}^{\text{l}}, \text{p}, \text{s}^{\text{l}}, \text{l}, \text{l}^{\text{u}}, \text{s}, \text{a}^{\text{u}}) \quad (7 \text{ entries}) \quad (11.4)$$

Then:

$$f(q) = \begin{cases} \cdot & \text{if } q = 0 \\ P[q - 1] & \text{if } 1 \leq q < 15 \\ N[q - 15] & \text{if } 15 \leq q < 19 \\ P[\lfloor (q - 19)/4 \rfloor] \sim L[(q - 19) \bmod 4] & \text{if } 19 \leq q < 75 \\ L[q - 75] & \text{if } 75 \leq q < 79 \\ M[q - 79] & \text{if } 79 \leq q \end{cases} \quad (11.5)$$

where \sim denotes string concatenation. Therefore, the resulting syllable is $g(r) \sim f(q)$.

Multi-byte numbers are represented in little-endian.

11.2 | The stack

The stack is a LIFO data structure with its entries being either an *atom* or a result from an operation. An atom is one of the following:

¹We use zero-indexing consistently.

- A root, consisting of a string of MR characters and an integer between 0 and 7, inclusive.
- A (computable) real number.
- A pronoun, consisting of a person (near = 0, far = 1, other = 2, anaphoric subject = 4, anaphoric object = 6, generic = 7) and number.

Root atoms are untyped (they can be cast to different parts of speech). The integer part of the root depends on the part of speech it is used as:

- In nouns, this is the noun class (sentient = 0, non-sentient = 2, measure = 3, edible = 5, inedible = 6, abstract = 7).
- In verbs, this is the conjugation pattern (first conjugation = 0, second conjugation = 1, third conjugation = 2).
- In descriptors, this indicates whether the root contains front vowels (0) or back vowels (4).
- The root atom can be cast to an adposition, conjunction or seqid, but in this case the integer part is ignored.

Results from operations (e. g. 001₈, which pushes a noun) are typed. Attempting to use such a value is a syntax error.

11.3 | Bytecode

Note that the bytecode is not *purely* bytecode; some sections are composed of raw Middle Rymakonian characters.

Table 11.1: Bytecodes of direct quotes.

Opcode (octal)	Additional parameters	Effect
000	none	No operation.
001	none	Pop a from the stack and then n from the stack. Treating n as a noun and a as an adjective, push the noun $[n, a]$ to the stack.
002	none	Pop n_a from the stack and then n from the stack. Treating n as a noun and n_a as a noun, push the noun $[n, n_a]$ to the stack.
003	none	Pop n_a from the stack and then n from the stack. Treating n as a noun and n_a as a noun, push the noun phrase $\langle n - \text{DI } n_a \rangle$ to the stack.
004 – 007	none	If bit 1 of <i>opcode</i> is set, then pop o from the stack. If bit 0 of <i>opcode</i> is set, then pop s from the stack. Pop v from the stack. Treating s and o as nouns and v as a verb, push the clause $[v, s, o]$ to the stack.

Opcode (octal)	Additional parameters	Effect
010 – 017	$n : \text{Byte}, \text{root} : \text{Char}[n]$	Reads a size n of one byte and then n raw Middle Rymakonian characters root . Pushes $\text{Root}(\text{root}, \text{opcode} \bmod 8)$ onto the main stack.
020	$n : \text{Byte}$	Push a pronoun onto the stack. The three least significant bits denote the person and bit 3 denotes number (non-dual = 0, dual = 1).
021	$x : \text{Byte}$	Pop a noun n from the stack and push it back with the following properties depending on the various bits of x : <ul style="list-style-type: none"> • Bit 0: Inverse if set, direct if not set. • Bit 1: Definite if set, indefinite if reset.
022	$x : \text{Byte}$	Pop v from the stack as a verb and push it back with the following properties depending on the various bits of x : <ul style="list-style-type: none"> • Bit 0: Past if set, nonpast if reset. • Bit 1: Atelic if set, telic if reset. • Bit 2: Negative if set, positive if reset. • Bit 3: Interrogative if set, affirmative if reset.

Opcode (octal)	Additional parameters	Effect
023	<i>x</i> : Byte, <i>y</i> : Byte, <i>z</i> : Byte	<p>Pop <i>v</i> from the stack as a verb and set its associated aspects according to the various bits of <i>x</i>, <i>y</i> and <i>z</i>:</p> <ul style="list-style-type: none"> • <i>x</i>: <ul style="list-style-type: none"> – Bit 0: Imperfect – Bit 1: Perfect – Bit 2: Gnostic – Bit 3: Deontic necessity – Bit 4: Deontic potential – Bit 5: Unexpected – Bit 6: Left branch – Bit 7: Right branch – Bit 8: (unused) • <i>y</i>: <ul style="list-style-type: none"> – Bit 0: Interrupted – Bit 1: Deontic recommendation – Bit 2: Gnostic dubitative – Bit 3: Epistemic necessity – Bit 4: Epistemic potential – Bit 5: Comparative – Bit 6: Temporal universal – Bit 7: Temporal non-universal – Bit 8: Spatial universal • <i>z</i>: <ul style="list-style-type: none"> – Bit 0: Nonexclusive subject – Bit 1: Spatial non-universal – Bit 2: Nonexclusive object – Bit 3: Nonexclusive argument – Bit 4: Set if both temporal and spatial aspects exist and the spatial marker comes first; otherwise, this bit is not set.

Opcode (octal)	Additional parameters	Effect
024	$x : \text{Byte}$	<p>Push a verb v such that $[v, a, b] \equiv \text{cmp}(a, b, f, \sqsupset)$, where the free variables are set according to the bits of x:</p> <ul style="list-style-type: none"> • Bit 0: set if f refers to the anaphoric object, reset if anaphoric subject. • Bit 1 – 4: index into $(>, <, =, \geq, \leq, \neq, \approx, \gg, \ll)$ for \sqsupset.
025	nothing	Pop o as an adjective and s as a noun and push the clause $[j-, s, o]$.
026	nothing	Pop o as a noun and s as a noun and push the clause $[j-, s, o]$.
030	nothing	Push the last-clause pronoun.
031	nothing	Push the second-to-last-clause pronoun.
040	none	Pop a noun n_p , an adposition p and another noun n . Then push n modified by $[p, n_p]$.
041	none	Pop a noun n_p , an adposition p and a verb v . Then push v modified by $[p, n_p]$.
042	none	Pop an adposition p and push its negated version.
043	none	Pop a conjunction c and two nouns n_2 and n_1 . Push the compound NP $[n_1, c, n_2]$.
050	none	Pop a verb v and push its transitivised version on the stack.
051	none	Pop v from the stack and then n from the stack. Treating v as a verb and a as an descriptor, push the verb $[b, a]$ to the stack.

Opcode (octal)	Additional parameters	Effect
060	$x : \text{Byte}$	<p>Push a connector according to the bits of x:</p> <ul style="list-style-type: none"> • Bits 0 – 2: the type of the connector (0 = ordinary, 1 = analogous, 2 = subversive, 3 = augmentative, 4 = explanatory, 5 = conditional). • Bit 3: the parity (0 = even, 1 = odd). • Bit 4: if set, pop a seqid from the stack. Otherwise, do not pop anything and consult bits 5 – 8 instead. • Bits 5 – 8: one of 16 intrinsic seqids. These depend on the dialect, but in the standard dialect these are $\langle \imath \ d \ e \ f \ c \ \Omega \ \mathfrak{c} \ \wp \ \Theta \ \rangle \ z \ \mathfrak{l} \ u \ j^h \ \imath \mu \ d\mu \rangle$. Unused if bit 4 is not set.
061	none	Pop a connector c and a clause α and push α with c attached.
100	none	Pop a literal and push it with information that it is a nominal name.
200	none	Pop roots b and a from the stack and push the concatenation of a and b on the stack with the attribute being the bitwise xor of those of a and b .
700	$x : \text{Byte}[4], a : \text{Byte}[x]$	Push the two's-complement integer represented by a to the stack.
701	none	Pop b and a from the stack as numbers and pushes $a + b$.
702	none	Pop b and a from the stack as numbers and pushes $a - b$.
703	none	Pop b and a from the stack as numbers and pushes a/b .
704	none	Pop b and a from the stack as numbers and pushes $a \cdot b$.

11.4 | Limitations

The quoting sublanguage has a few limitations compared to Middle Rymakonian proper:

- A sentence with a copula and an adjective as the “object” cannot have any attributes attached to the copula. The workaround is to refrain from using in-

struction 025 and instead create a normal sentence (via 007) whose object is a dummy noun (e. g. <DIZI>, <UCI>) modified by the adjective.

- It is not possible to specify the order of the subject relative to the object. Some nonstandard dialects have extensions that allow this; for instance, the Myðun dialect reserves instruction 633 to be equivalent to 007 but with inverted S-O order. Other dialects might provide an instruction to explicitly mark accusative arguments if needed, as Middle Rymakonian itself does.

11.5 | Example

We shall translate the following:

Impressed by her confidence, I said, “Having known Vladimir for nearly 12 years, twice as long as you have, I found it surprising at the time, though not anymore, that he had let Maria, despite her having been ridiculed by her brothers for daring to listen to him, either take the opportunity to quit the Assembly without repercussions or announce there the following: ‘We shall no sooner compensate all of you than let the man who had allowed the criminal we had let the chairman capture to escape return to his proper role of lifelong slavery.’”

Let us first work on the deepest level of nesting:

‘We shall no sooner compensate all of you than let the man who had allowed the criminal we had let the chairman capture to escape return to his proper role of lifelong slavery.’

This is approximately equivalent to stating “if we compensate all of you, then we would have let the man ...”. This requires the use of the conditional connector between the two main clauses. Let us look at the right clause:

... let the man who had allowed the criminal we had let the chairman capture to escape return to his proper role of lifelong slavery.’

We can break this down into a series of simple clauses:

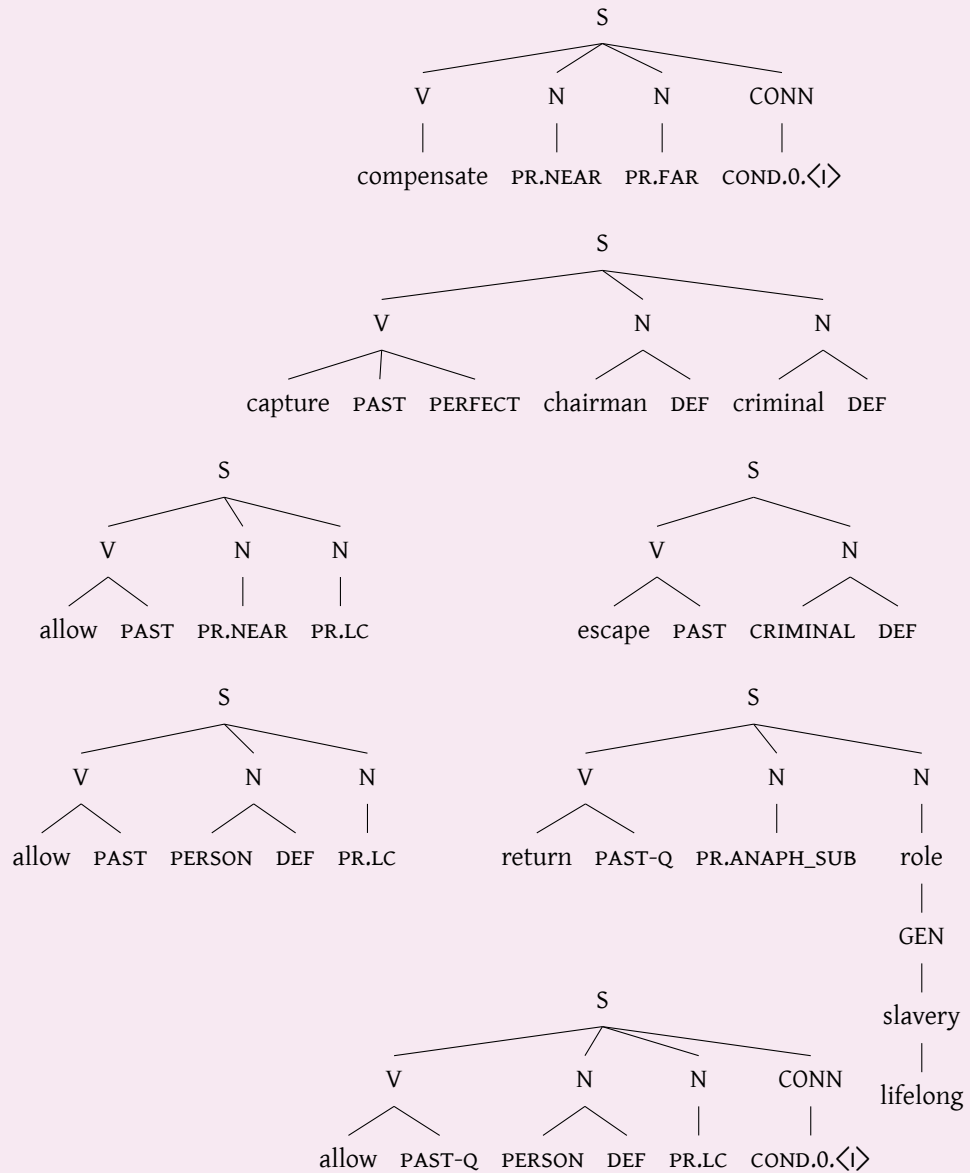
- The chairman captured the criminal.
- We had let the above happen.
- The criminal escaped.
- The man had allowed the above to happen.
- The man would returned to his proper role of lifelong slavery.
- We would let the above happen.

The ASTs for all seven clauses are shown in figure 11.1.

The seven clauses together thus form the word <Iḡḡḡ.enebdi.cjə.ijə.cfcbi.ɔbcfe.eəəzdi.cdc.ɜ.ɪ.ɪfɜcnɪbəjənʰjɜ.ɜfɜ.ɔḡʰɜcnʰjɜ.ɜfcə.cɪdi.cjə.ɪaɪfcɜ.ɛɪɸdi.cɜ.ɔḡʰɜcnʰjɜ.ɜɪfə.cɪdi.cɜ.əɔɪɜjɜ.ɜaɪfcə.cɔdiɛjə.əjə.ɜjɜɔɪjə.əɔɪɪjɪ.əənən.c.ɛfcə.cɪdiɛfɜ.əɔɪɜjɜ.ɜaɪfcbi.ɔbc>. (See 7_1/listings/example-nestedquote.qasm for how this was derived.)

Now we work on the middle clause, which can be unfolded as:

Figure 11.1: ASTs for the clauses at the deepest nesting level. NB: We translate *role* as $\langle j\textcircled{a}l \rangle$, which has a connotation of normalcy, so we need not explicitly state that it is “proper”.



- I have known Vladimir for 12 years.
- You have known Vladimir for $12/2 = 6$ years.
- Maria could have taken the opportunity to quit the Assembly without repercussions.
- She could have announced (...).
- Vladimir let either one of the above to happen. (1)
- She dared to listen to Vladimir.
- Therefore, she was ridiculed by her brothers. This was done despite (1).
- I found the above surprising.
- I do not find it surprising right now.

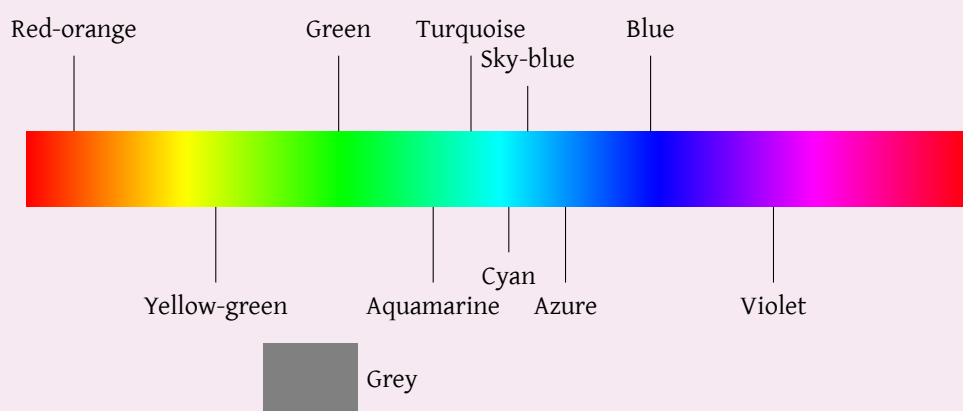
For the sake of brevity, we will not show the ASTs for these clauses, but see `7_1/listings/example-quote.qasm` for how this is derived. The part outside the quotation is straightforward. Thus we have:

desif .c b c p - n c f i , p e u c . e l p i f e . e b a n d i . c d c . c . i . i f 3 . 3 n e f a . e b i n e a ' o
 a . c . i . i . i . j i . 3 e j e . i f 3 z u l i n c b c r t u a c f e . e b a n d i . c d c . c . i . i f 3 . 3 n e f a . e b i n e a ' o
 a . c . i . i . i . j i . a ' o a . c . i . i . i . 3 p c . 3 e j e . c j a f i f c a f c u c p l i j d i . c d c . 3 j e . i f 3 . a d r l e n i ' a f
 3 f i a c l b i d i ' a f 3 i f i d r c . i u a j e . a d i l i n j a . 3 c f 3 . e p e u d i . c d c . 3 j e . i j e . a f 3 i l e f 3 . e
 n e b d i . c j e . i j e . c f c b i . c b c f e . e a a z d i . c d c . 3 . i . i f 3 c n i b a j e n ' j a . 3 f 3 . a a ' a f c n ' o
 j a . 3 f c f a . c r d i . c j e . i a i f c f 3 . e f i p d i . c f 3 . a a ' a f c n ' o j a . 3 f i f a . c r d i . c f 3 . a d i z i j a . 3
 a i f c f a . c d d i f e j e . a j e . a j a l j e . a o i i j i . a e n e n . c . e f c f a . c r d i f e f 3 . a d i z i j a . 3 a i f
 c b i . c b c f c f a . c r l f 3 z u l i n c b c r t u a i a c f 3 . c z l a f c b i l e b c f 3 . a u e n j d i . c j i . a . i a i
 z l ' a f 3 i f i d r c . i u a j e . a f c b i . e b c f 3 . e l ' i . d i . c f 3 . a a 3 l a j a . c j e . a f c b i . e b c b i l e b c
 f 3 . e d e s d i . c a i j e . c f c f 3 . e d e s d i . a f 3 . e d c a l ' a a c j e . c f c .

12 | Miscellanea

12.1 | Colour

Middle Rymakonian has eleven non-derived colour terms:



Note that “grey” refers generically to a loss of chroma. There is no distinction between a decrease in saturation and a decrease in value.

Middle Rymakonian works with colour *transitions*, not static colours, and uses abstract nouns to represent them. See table 12.1.

Table 12.1: Colour transitions in Middle Rymakonian. Each row represents a different starting colour; each column represents a different ending colour.

	RO	YG	Gn	Aq	Tu	Cy	SB	Az	Bl	Vi	Gy
RO	l̥ij	deɪ	ɔ ^h en	aiɪ	j ^o cɥ	dʒɪ	bef	ɔɪɪ	ɑ ^h ɔd	fɛn ^ʰ	ɔc.cd
YG	ɔɔɪ	ɥəɔ	ɪ ^h ɔd	ɔɔɪ	n ^h cɹ	fɛɪ	n ^h ɪɥ	ɑ ^h ef	βen	ɥed	dəɔ
Gn	ɪ ^h ɔn	ɑ ^h ɔɔɔ	l̥eɪ	ɔɔɔ	ɥɔɪ	nəl	bɔn	ɪef	jɔɪɔ	j ^o əɹ	ɪɔɹ
Aq	j ^o ɪɥ	fɔɥ	ɥɔɔ	oɔɪ	ɔcɥɹ	.əɪ	ɪ ^h ɔɥ	ɪɥɔɹ	noed	ɔɪɪ	beɪ
Tu	ɑəɪ	n ^h əɹ	ɪeɪ	ɔ ^h ɹɥ	ɔɥen	ɥcɹ	ɪ ^h ɔɔ	fɥeɪ	j ^o cɪ	ɔəɥ	ɔɔɔ
Cy	bɹɥ	ɔɔɪ	nɔɥ	.cɪ	ɔəɹ	bɔeɪ	ɪ ^h cɪ	n ^h ɪɪ	ɑɥcn	bəɪ	nɔɹ
SB	dɔɪ	n ^h ɪɪ	den	ɪeɪ	ɑ ^h əɔ	ɪəɥ	oɪɪ	j ^o ɔɥd	den	ɔcɪ	βɪɔ
Az	ɔɪɥ	jɔɪ	ɪɔd	ɥɔeɹ	ɔɔɥ	n ^h ɪɪ	ɑɥef	ɑocɹ	fɔɥ	ɥɔd	oɪ
Bl	heɪ	βɔn	ɑ ^h ɹɥ	nɥɔɪ	ɑ ^h c.əɪ	j ^o ɥən	bɔn	ɔəɪ	ɪ ^h ɪɪ	ɪɔn ^ʰ	ɥəɔ
Vi	ɔɔn ^ʰ	oɔɪ	ɑ ^h cɹ	ɔɪɪ	fɔɪ	dɔɪ	ɥəɪ	ɔcɪ	ɥɔn ^ʰ	ɪ ^h ɔn	ai
Gy	ɔɔ	bɔɹ	ɪe	dɔɹ	ɥe	jɔ	βɪ	ɥɪɥɪ	ɑ ^h e	j ^o c	.cɹ

A | Listings of programs

A.1 | workfiles/7/tides.sage

```
1 # How many values to output
2
3 limit = int(sys.argv[1]) if len(sys.argv) > 1 else 1000
4
5 # :P
6
7 tau = 2 * pi
8
9 t = var("t")
10
11 # Constants
12
13 A_s = 0.675; A_sa = 0.0532; A_m = 1.267; A_ma = 0.176; l_m = 30.80152
14
15 # Solar component
16 y_s2 = A_s * (1 + A_sa * cos(tau * t)) * cos(2 * tau * t)
17 # Lunar component
18 y_m2 = A_m * (1 + A_ma * cos(tau * t / l_m)) * cos(2 * tau * t / l_m -
19             2 * tau * t)
20 y = y_s2 + y_m2
21 yp = diff(y, t)
22
23 # High and low tides occur at values of t where dy/dt = 0.
24
25 i = 0
26 time = 0
27 print(0)
28 while i < limit:
29     try:
30         time2 = find_root(yp == 0, time + 0.000000001, time + 0.35)
31         print(time2)
32         time = time2
33         i += 1
34     except:
35         time += 0.01
```

workfiles/7/tides.sage

A.2 | workfiles/7/bins.pl6

```

1  # CONSTANTS
2
3  constant \MONTHS_PER_YEAR_CYCLE = 7215;
4  constant \YEARS_PER_YEAR_CYCLE = 736;
5  constant \AVG_MONTHS_PER_YEAR = MONTHS_PER_YEAR_CYCLE /
6      YEARS_PER_YEAR_CYCLE;
7
8  # COMPUTATION
9  # For each year, take as many months as are needed
10 # in order to cycle to the next.
11
12 my $c = 0;
13 my @k;
14
15 for 0 ..^ YEARS_PER_YEAR_CYCLE -> $i {
16     my $need = 1 - ($c - floor($c));
17     my $objs = ceiling($need * AVG_MONTHS_PER_YEAR);
18     @k[$i] = $objs;
19     $c += $objs / AVG_MONTHS_PER_YEAR;
20 }
21
22 # DISPLAY
23
24 my \cols = 4;
25 my $len = @k.elems;
26
27 say ("    0123456789" xx cols).join(" | ");
28
29 my \total-rows = ceiling($len / 10);
30 my \rows = ceiling(total-rows / cols);
31
32 for 0 ..^ rows -> $j {
33     for 0 ..^ cols -> $p {
34         print(" | ") if $p != 0;
35         my $q = $j + rows * $p;
36         next if $q >= total-rows;
37         printf("%3d ", $q);
38         for 0 ..^ 10 {
39             my $i = 10 * $q + $_;
40             if $i >= $len { print " "; }
41             else {
42                 print "0123456789XE".substr(@k[$i], 1);
43             }
44         }
45     }
46     say " ";
47 }

```

workfiles/7/bins.pl6

Romanisation

In this text, the romanisation is used only to transcribe names into English. Whenever possible, the hacmisation should be used.

Table A.1: The consonants of Middle Rymakonian.

	Bilabial	Dental	Alveolar	Palatal	Velar	Glottal
Nasal	m		n	ɲ	ŋ	
Plosive	p b		t d	tʃ dʒ	k g	ʔ
Fricative	f v	θ ð	s z	ʃ ʒ	h ɦ	
(coarticulated)	fh vɦ	ɸh ðɦ		fʃ vʒ		
(whistled)			ʂ ʐ			
Affricate			c	č		
Lateral fricative			ɬ ɮ			
Approximant			r	j	w	
Lateral approximant			l			
Tap			ɾ			

Table A.2: The vowels of Middle Rymakonian.

	Front	Central	Back
High	i	y	u
Mid	e		o
Low		a	

The digraphs <fh vɦ ɸh ðɦ fʃ vʒ ts tʃ> correspond to coarticulated consonants and affricates. An apostrophe can be placed between the two letters if this is not desired.

Rod signs are represented by the Arabic digits <1 2 3 4 5 6 7 8> attached to the end of the verbs they encompass. Halfway rod signals are represented by subscript digits <₄>. Transitions from the ninth rod signal are written <⁹⁷ ⁹⁸>. Proper words are preceded by a backslash <\>.

<ɲ> should be capitalised as <Ŋ> only if one can depend on the majuscule glyph appearing like an N with a hook. Otherwise, it should be spelled <Ng>.

B | List of glossing abbreviations

An asterisk indicates that this value is unmarked.

Abbr	Definition
<i>Case</i>	
NOM*	Nominative
ACC	Accusative
GEN	Genitive (this is a clitic and not a case <i>per se</i>)
<i>Number</i>	
DIR*	Direct
INV	Inverse
ND*	Nondual
DU	Dual
<i>Person</i>	
NEAR	Near
FAR	Far
OTHER	Other
ANAPH_SUB	Anaphoric subject
ANAPH_OBJ	Anaphoric object
GENERIC	Generic
<i>Definiteness</i>	
INDEF*	Indefinite
DEF	Definite
<i>Tense</i>	
NONPAST*	Nonpast
PAST	Past
<i>Tellicity</i>	
TELIC*	Telic
ATELIC	Atelic
<i>Polarity</i>	
POS*	Positive
NEG	Negative
Q	Interrogative
<i>Aspect and connector type names are not abbreviated.</i>	
<i>Connector parity</i>	
0	Even
1	Odd
<i>Comparisons</i>	

Abbr	Definition
CMP	Comparative
<i>Comparators are glossed using their respective operator symbols.</i>	
<i>Transitivisation</i>	
TRANS	Transitivised
<i>Tree mode</i>	
BRANCH	Create a new branch ($\langle n^u \uparrow \rangle$)
JOIN	Join two branches ($\langle n^u \Delta \rangle$)
LEFT	Refer to left branch ($\langle - \uparrow \rangle$)
RIGHT	Refer to right branch ($\langle - \Delta \rangle$)

C | Quoting assembler

This chapter describes a mnemonic language to assist in the process of quoting as described in chapter 11.

Table C.1: Syntax corresponding to operations of direct quoting. Square brackets denote optional parameters, and pipe characters denote alternation. Non-literals (e. g. parameters) are shown in *italics*. Defaults are shown in **bold**.

Opcode	Syntax	Notes
000	nop	
001	modadj	
002	modnoun	
003	modgen	
004 – 007	sentence [s ns] [o no]	s denotes subject and ns denotes no subject from the stack. o and no are analogously defined for objects.
010 – 017	literal <0–7 v 1–3 nsent nnonsent nmeas nedib ni ned nabst dfront dback > <i>literal</i>	The literal is surrounded by double quotes and uses romanisation from hacm. Superscript letters are denoted by caret characters.
020	pronoun < near far other anaph_sub anaph_obj generic > [nd du]	
021	setnopt [dir inv] [indef def]	
022	setvopt [nonpast past] [telic atelic] [pos neg] [nq q]	

Opcode	Syntax	Notes
023	setaspect [imp] [perf] [gnom] [deon_nec] [deon_pot] [unexp] [left] [right] [inter] [deon_rec] [gnom_dub] [epis_nec] [epis_pot] [cmp] [tmpu] [tmpnu] [spcu] [nxs] [spcnu] [nxo] [nxa] [spcfirst]	
024	cmp <s o> <g l e ge le ne ae gg ll>	
025	copadj	
026	copnoun	
030	pronounlc	
031	pronounllc	
040	adpton	
041	adptov	
042	negadp	
043	conj	
050	trans	
051	modverb	
060	conn <ordinary analogous subversive augmentative explanatory conditional> <0 1> seqid [noimm]	If <i>seqid</i> is one of the intrinsic seqids, then the instruction is converted to not pop a seqid unless the noimm option is set. If a default seqid is not used and <i>seqid</i> is not –, then a literal instruction is generated before this instruction.
061	attachconn	
100	setnname	
200	concat	
700	pushint <i>a</i>	
701	add	
702	sub	
703	div	
704	mul	

D | Listings of programs

D.1 | 7_1/listings/example-nestedquote.qasm

```
1 # Clause #1
2   literal v1 "neb" # = compensate
3   setvopt past
4   pronoun near # Subject
5   pronoun far # Object
6   sentence # Join sentence
7   conn conditional 0 "a" # connector
8   attachconn
9 # Clause #2
10  literal v2 "fuz" # = capture
11  setvopt past
12  setaspect perf
13  literal nsent "nabusun^g" # = chairperson
14  setnopt def
15  literal nsent "f^xoton^g" # = criminal
16  setnopt def
17  sentence
18 # Clause #3
19  literal v3 "c" # = allow
20  setvopt past
21  pronoun near # Subject
22  pronounlc # Object
23  sentence
24 # Clause #4
25  literal v1 "tag" # = escape
26  setvopt past
27  # In this sense, the 0 is the escaper.
28  literal nsent "f^xoton^g" # = criminal
29  setnopt def
30  sentence ns # Omit the subject
31 # Clause #5
32  literal v3 "c" # = allow
33  setvopt past
34  literal nsent "maza" # = person
35  setnopt def
36  pronounlc # Object
37  sentence
38 # Clause #6
39  literal v3 "m" # = return to
40  setvopt past q
41  pronoun anaph_sub # Subject
42  literal nabst "skofl" # = role; Object
```

```

43 # Push a node representing "lifelong slavery"
44 literal nabst "wat^sa" # = slavery
45 literal dback "unun" # = lifelong
46 modadj # Modify
47 # Attach it by genitive
48 modgen
49 sentence
50 # Clause #7
51 literal v3 "c" # = allow
52 setvopt past q
53 literal nsent "maza" # = person
54 setnopt def
55 pronounlc
56 sentence
57 conn conditional 0 "a" # connector
58 attachconn

```

7_1/listings/example-nestedquote.qasm

D.2 | 7_1/listings/example-quote.qasm

```

1 # Clause #1: I have known Vladimir for 12 years.
2 literal v2 "mun" # = know (a person)
3 setvopt past
4 setaspect imp
5 literal 0 "ne" # preposition
6 literal nmeas "bane" # = year
7 pushint 12
8 modnoun
9 adptov
10 pronoun near # Subject
11 literal 0 "vladimic" # Object
12 setnname
13 sentence
14 # Clause #2: You have known Vladimir for 12/2 years.
15 literal v2 "mun" # = know (a person)
16 setvopt past
17 setaspect imp
18 literal 0 "ne" # preposition
19 literal nmeas "bane" # = year
20 pushint 12
21 pushint 2
22 div
23 modnoun
24 adptov
25 pronoun far # Subject
26 pronoun anaph_obj # Object
27 sentence
28 # Clause #3: Maria could have taken the opportunity to quit the
    Assembly without repercussions.
29 literal v3 "virlast" # = leave (an organisation)
30 setvopt past
31 setaspect epis_pot perf
32 literal dfront "pcen" # = by opportunity
33 modverb
34 literal dfront "filmap" # = without repercussions
35 modverb
36 literal 0 "maci.a" # name
37 setnname

```

```

38 literal nabst "malna" # = assembly
39 setnopt def
40 sentence
41 # Clause #4: She could have announced (...).
42 literal v1 "rev" # = say
43 setvopt past
44 setaspect epis_pot perf
45 pronoun anaph_sub
46 quote example-nestedquote.qasm
47 sentence
48 # Clause #5: Vladimir let either one of the above happen.
49 literal v3 "c" # = allow
50 literal 0 "vladimic" # Subject
51 setnname
52 # Object:
53 pronounlc
54 pronounllc
55 literal 0 "z" # = or
56 conj
57 # End
58 sentence
59 conn subversive 0 "p"
60 attachconn
61 # Clause #6: She dared to listen to Vladimir.
62 literal v1 "vens" # = listen
63 setvopt past
64 literal dback ".adaz" # = dare to
65 modverb
66 literal 0 "maci.a" # name, Subject
67 setnname
68 pronoun anaph_sub # Object
69 sentence
70 conn explanatory 0 "a"
71 attachconn
72 # Clause #7: Therefore, she was ridiculed by her brothers.
73 literal v1 "k^xa." # = ridicule
74 setvopt past
75 literal nsent "fqlto" # = sibling (du)
76 setnopt inv
77 pronoun anaph_sub # Object
78 sentence
79 conn explanatory 0 "a"
80 attachconn
81 conn subversive 0 "p"
82 attachconn
83 # Clause #8: I found the above surprising.
84 literal v1 "mej" # = surprise
85 setvopt past
86 pronounlc
87 pronoun far
88 sentence
89 # Clause #9: I do not find it surprising right now.
90 literal v1 "mej" # = surprise
91 setvopt neg
92 literal dfront "mif" # = now
93 modverb
94 pronounllc
95 pronoun far
96 sentence

```


E | Dictionary

An entry looks like this:

ᄁᄇᄇ– v1 (S) eats (O)

From left to right:

1. The entry – the Middle Rymakonian term listed.
2. The part of speech of the corresponding entry:
 - *n* – a noun
 - *-d-* – inherently dual
 - *-sent* – sentient noun
 - *-nonsent* – nonsentient noun
 - *-meas* – measure noun
 - *-edib* – edible noun
 - *-ined* – inedible noun
 - *-abst* – abstract noun
 - *v1, v2, v3* – first-, second- and third- conjugation verbs
 - *desc* – a descriptor
 - *pp* – a preposition
 - *-(b)* – this entry has only neutral vowels but acts as if it had back vowels
 - *-(ŋ)* – this entry came from a word that started with <ŋ⁰-> and thus certain prefixes will revert it back
3. The definition – the gloss for the corresponding entry.
 - (S) – subject
 - (O) – direct object
4. If applicable, any special grammatical or semantic notes for this term.
5. Optionally, examples of usage.

| .

.ᄇᄇZ– *desc(b)* dare to

- j^h**
- j^h pp toward
 j^h– v1 (S) goes toward (O)
 j^hz pp written by
- n**
- n.l pp in, on, at (time)
 n.μ pp through
 nɥɪzi nnonsent cat
 ni.– v3 (S) waits for/until (O), temporal verb, if
 ni.ez– v3 (S) covers, spans (O)
 ni.bəjən^ɸ nanim chairperson
 ncj nnonsent face
 ncja nnonsent birch
 ncd– v3 (S) dances around (O)
 ne pp for (some period of time)
 neuc nined fire
 neb– v1 (S) gives something to (O), (S) rewards, compensates (O)
 nepa– v3 (S) hides from (O)
 nez^uɪɔ nnonsent rose
 nən– v2 (S) kills (O), (O) dies
 nəb– v2 steal
- n^ɥ**
- n^ɥɔz– v1(ɣ) (S) thinks, ponders about (O)
- u**
- ucɪ nnonsent inanimate object
 ucɥɪɣf– v3 leave, resign from (an organisation)
 uenj– v1 hear, listen to
- a**
- an conj and
 aɔɣɪɪ nnonsent stick, rod
 apen^{ɥe} nnonsent coin
 azɪ– v2 (S) intersects (O) meet = paths intersect
 ac.cɪ nabst centre, origin
 acjɪn pp according to
- acz– v1 (S) shines on (O)
 acze nabst light
 aczelɪɔ nabst moonlight
 acɪɔɪɔ– desc without repercussions
- a^l**
- a^l– v3 do what?
 a^lize nabst quote, words, speech
 a^lɔɪɔɪ^ɸ nsent criminal
 a^lɔμə nedib potable water
- a^h**
- a^həɔɔ nabst empathy
- D**
- D– v2 (S) returns, regresses to (O) (figuratively)
 Dɪ pp far away from
 Dɪɪ pp inside
 Dɔɪɔɪ^ɸ nsent nobleman, gentleman
 Dɪɪ– v1 (S) eats (O)
 Dɪɪɪ nabst void
 Dɪɪɪz– desc empty
 Dɪɪ nedib rice
 Dɪz nnonsent corpse
 Dɪzɪ nsent person
 Dɪɪɪ nabst organisation, group, assembly
 Dɪɪ desc far
 Dɪɪ nnonsent border, boundary
 Dɪɪ– desc current, now
 Dɪɪ ndnonsent hand
 Dɪz ne nabst technique
 Dɔɪ– v2 (S) produces, makes (O)
 Dɔɪ^ɸ nnonsent head
 Dɔɪ– v2 (S) dreams of (O)
 Dɔɪɔɪ^ɸ ndsent parent
 Dɪɪɪ nmeas hour
 Dɪɪ– v1 (S) buries (O) in the ground
 Dɪɪ– v1 begin, start

deφ- v1 (S) receives (O)
 ded- v3 err, miss
 debϣ- v1 (S) resembles (O)
 debc nabst shape, structure
 debcdelbe nabst grammar
 debed nabst dream
 dez- v3 (S) raises, takes care of,
 tends to (O)
 dez- v1 (S) boards, rides (O)
 des- v3 (S) gives birth to (O), (O) is
 born (S) is not necessarily the mother;
 this can be either parent
 des- v1 surprise, impressed
 del nmeas volume
 dell- v3 (S) imitates (O)
 dəlɔn^φ nsent servant
 dən- v2 know (a person)
 dɔn^φɔ nnonst rabbit

Ω

Ωnɔφ nnonst wall
 Ωϣu- desc weak
 Ωcl^l- v1 (S) hits, strikes (O)
 Ωcz- v2 comparative verb
 Ωed- v1 (S) sleeps
 Ωed- v1 (S) sleeps
 Ωə.ɔn- desc sick, ill
 Ωəzɔ nnonst pathway, street,
 road

φ

φ^lcl- v3 (S) lowers their own (O)
 φpɪl- v1 (S) raises their own (O)
 φpə.ɪ nnonst earth, soil
 φpən^φ nnonst frog

d

dfj- desc healthy, well (not sick)
 dj^hf- v1 (S) mates with (O)
 dφcj nabst ability, potential, pos-
 sibility
 dɪɪfjɪn^φ nnonst mist, fog
 dɪɪβɪ nabst ground, floor
 dɪcz- v1 stand, get up
 dɪen- desc by opportunity

dɪ pp with (comitative)
 dɪ- v3 hold, carry, instrumental
 verb
 dɪu- v1 (S) passes (O)
 dɪede nnonst city
 dɪfɪ- v1 (S) sits at (O)
 dɪlɔ nnonst pecan
 dɔf- v2 (S) chases away (O), (O)
 flees from (S)
 dez^u- v1 disagree, protest, disap-
 prove of
 delbe nabst sentence, utterance
 dɔn^φɪ ndnonst(b) knee

b

bine nmeas year
 beφ^sɪ nedib sea

ɥ

ɥ- v1 (S) sees (O), because, (S) illu-
 minates (O)
 ɥɪβ- v1 (S) enslaves (O)

ɹ

ɹ- v3 (S) allows (O)

ɹ

ɹɪd pp with (colour) hair
 ɹɪ pp with (instrumental)
 ɹu pp in front of
 ɹɪɪ nined grass
 ɹɪl- v1 (S) climbs, rises in (O)
 ɹɪldɔ nnonst hair
 ɹeu- v3 (S) spreads (O)
 ɹeu- v1 (S) says a phrase (O)
 ɹea- v3 (S) speaks to (O), (S) asks
 (O)
 ɹellɪ nnonst place
 ɹellɪuc.cɪ nnonst hometown,
 home village, (figurative) Rymako
 ɹəɔ nined gold
 ɹəz- v2 (S) ties (O) in a knot
 ɹəɪ nnonst star
 ɹəɔ nabst nighttime

μ3Jɔ *nnon*sent cup
 μ3D3 *nmeas* (tidal) day (contin-
 ous)

oɪʃen^ϕ *nanim* slave

| l

| z

ʎen- v3 (S) obeys (O)

z conj or

| o

| ə

oɪʃɪ *nabst* slavery

ənən- *desc* lifelong, permanent