MACHINE TRANSLATION OF KLINGON

TRACY CANFIELD

L466 FINAL PROJECT

CONTENTS

OVERVIEW

PROJECT ARCHITECTURE

The machine translation project presented in this paper was developed using three existing programs: PC-PATR, an open-source feature-grammar parser developed by SIL International; ExpandLex, written by George Wilson, which uses user-created morphological rules to generate lexicon files to be read by PC-PATR; and SyntacticTransfer, also by George Wilson, which applies user-created rules to transform the source-text parse trees generated by PC-PATR into sentences in the target text. To make use of this framework, a set of morphological rules, a base lexicon, and a feature-structure grammar were created, as well as a postprocessor to further modify the lexicon file before it is used by PC-PATR.

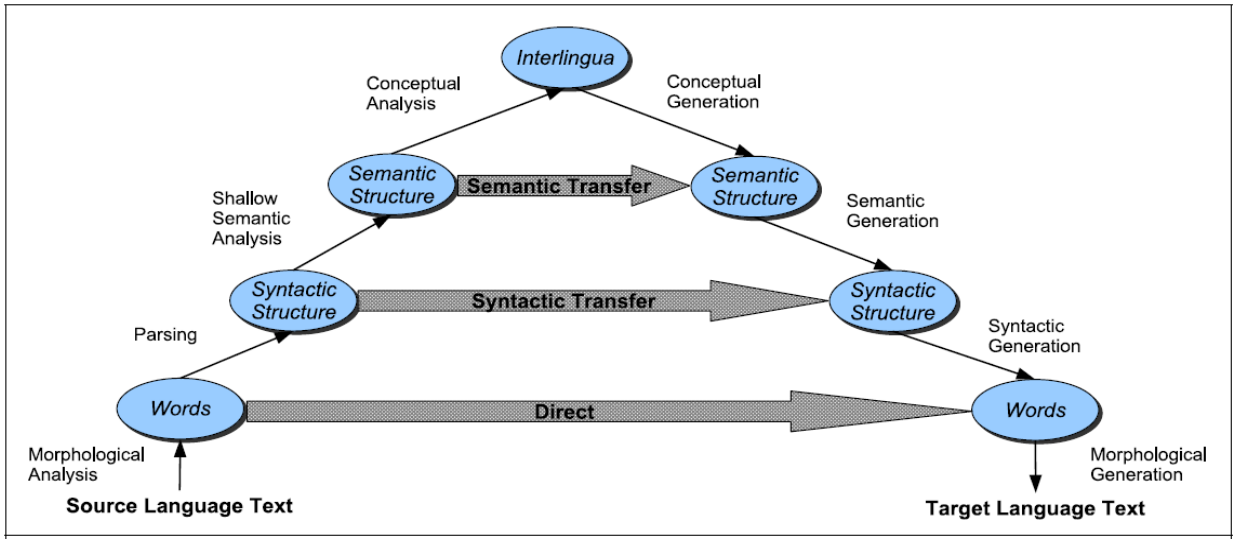

Figure 1: The Vauquois Triangle, from Jurafsky and Martin (2006).

The architecture corresponds in some, but not all, respects to the Vauquois Triangle, proposed by Bernard Vauquois in 1968 as a model for the task of machine translation (Dorr et al, 2005). It may be thought of as corresponding to the syntactic transfer of analysis, which incorporates syntactic information to determine the relations between lexical items and to disambiguate homonyms, but does not attempt to extract any semantic information from the source text.

In the classic Vauquois model, we would expect to see three distinct phases: a parsing phase, which analyzes the source text and produces a tree or some other representation of the syntactic structure; a syntactic transfer phase, which transforms this structure into a tree in the target language; and a syntactic generation phase, which uses this tree to produce a sentence in the target language. In this architecture, however, the generation task is partially performed during parsing, during which glosses are assigned to the source-text lexical items, and partially during the syntactic phase, during which elements such as articles or pronouns which do not correspond to lexical items in the source text are inserted by SyntacticTransfer. This process will be discussed in greater detail below.

THE LEXICON

The base pre-lexicon file contains information that cannot be predicted, such as the part of speech and the base target-language gloss. A typical entry is shown in Figure 2.

```
\w toQDuj
\c NOUN
\g Bird of Prey
\e IRREGULAR
\f <prep> = on
```

Figure 2: A pre-lexicon entry.

The line beginning with `\w` specifies the root word form before inflectional affixes are added. `\g` is the base English gloss. The feature `\f` indicates that this noun does not take the default preposition *at*, and specifies which preposition will be used. `\e` introduces a tag which is used by ExpandLex to determine which rules apply to this lexical item. (*Bird of Prey* is irregular because the plural cannot be formed by simply adding *-s*[1]; the plural is *Birds of Prey*, not **Birds of Prey*.) While the class `\c` might appear to be the part of speech, it is not in fact used directly by PC-PATR. As can be seen in the feature-structure grammar in Appendix B, a sentence may never include words with the part of speech NOUN. Instead, the NOUN entries are used by the ExpandLex morphological rules to generate lexical entries with the part of speech N, which will be used to parse user input.

A typical ExpandLex rule which would apply to the input from Figure 2 is shown in Figure 3.

<Morph>NOUN, , , N, , mey, , , irreg-plur-, , <head-agr-number> = plural <head-case> = unmarked, BODY_PART|NON_COUNT|INHERENT_PLURAL, IRREGULAR </Morph

Figure 3: An ExpandLex rule

This rule selects lexical entries with the category NOUN and the ExpandLex tag IRREGULAR and creates a new lexical entry with the part of speech N. It appends the plural ending **-mey** to the root, adds several features to the lexical entry, and prepends "irreg-plur-" to the English gloss. NOUN entries with the ExpandLex tags BODY_PART, NON_COUNT, or INHERENT_PLURAL do not participate in this rule. (As we shall see when we treat Klingon grammar in greater detail, the plural ending **-mey** may not be used with certain classes of nouns.) The complete ExpandLex file has not been included due to its length – it is 8753 lines in length and implements 6741 rules.

Originally, the ExpandLex rules incorporated certain English spelling rules, such as ensuring that nouns with an English gloss ending in *y*, when pluralized, had glosses ending in *ies*. However, this led to a large number of nearly-identical ExpandLex rules. In the current version of the software, rewrite rules of this sort have been moved to the postprocessing section of the SyntacticTransfer phase, described below.

The `irreg-plural` prefix on the gloss is used by the lexical postprocessor to identify irregular forms and substitute the correct forms. The code for the postprocessor is included in Appendix C. It might seem unintuitive to implement the realization of irregular forms as a set of rules, rather than as part of the original lexicon. As we shall see, however, a Klingon root may combine with a large number of inflectional affixes. Entering each of these forms in the lexicon would be tedious and error-prone. Currently, the only irregular forms which are hard-coded in the lexicon, rather than generated by a later step of the process, are suppletive pronoun forms such as *I* and *me*, and idiomatic forms such as **tu'lu'**, which literally means "(something) is found" but is used with the sense of *There is* or *There are.*

---

[1] A Bird of Prey is a type of Klingon warship.

tlhIngan Suvwl'pu' qan tu'lu'be'

Klingon fight-AGENT-ANIMATE-PL old find-SUBJ-IMPERS-OBJ-3rd-PL-NEG

There are no old Klingon warriors

A NOTE ABOUT THE EXAMPLES

All English glosses in the examples are the output of a computer running the software described above.

THE FEATURE STRUCTURE GRAMMAR

The feature structure grammar includes both features which are necessary to ensure that source sentences are grammatical, such as verb agreement, and features which are necessary to ensure that the correct English text is generated, such as <verb-of-motion> (used to ensure that the correct English preposition is used in the translation). Some of the more notable constructions used in the file will be discussed alongside the grammatical features they model. The complete file is included as Appendix B.

THE SYNTACTIC TRANSFER PHASE

The syntactic transfer phase accomplishes several tasks. First, SyntacticTransfer applies a series of pre-processing rules to the input. It then launches a PC-PATR process to attempt to parse the input. If an input sentence is rejected by PC-PATR because one or more words in the input are not in the lexicon, it attempts to assign a reasonable category and any predictable features to the word. (This will be discussed further below once we have considered the characteristics of Klingon that apply to this process.) More importantly, it re-orders the constituents within the parse and inserts additional lexical items (such as pleonastic subjects and articles) based on the structure and features assigned by PC-PATR. Finally, it applies a series of post-processing rules to the output, which perform surface-level changes such as substituting *an* for *a* before a word beginning with a vowel.

CHALLENGES PRESENTED BY THIS ARCHITECTURE

PC-PATR does not support non-ASCII characters. Other writing systems must be transliterated into ASCII before they can be processed. While this is not necessarily a difficult task, it does potentially introduce problems for coding and debugging lexicon and grammar files for a user who is more accustomed to reading the original script than its transliteration, without raising interesting theoretical challenges.

Furthermore, no morphological analysis is implemented. Inflected forms must be pre-generated and included in the lexicon. Languages such as English or Chinese, with few or no inflectional endings, are more easily implemented than languages such as Turkish, in which a single root may appear with numerous inflectional endings.

The fictional language Klingon was used for this implementation because it is written in ASCII, making a transliteration phase unnecessary, and because it has a large number of inflectional endings affixes whose meanings may interact when they co-occur. Furthermore, as we shall see, its syntax is significantly different from that of English.

THE FICTIONAL KLINGON LANGUAGE

The primary reference used for Klingon is the 1992 edition of *The Klingon Dictionary* by Marc Okrand, the creator of the language[2].

Klingon is a constructed, or created[3], language with a tiny community of users outside of the context of its fictional representations in the Star Trek television programs and movies. Linguist Anita Okrent has estimated that approximately 300 people can read and write Klingon well enough to converse on electronic forums, and perhaps 20 or 30 speak it well enough to converse aloud in real time (Okrent, 2009). While there are, obviously, no practical applications for the translation of Klingon into English, this serves as a convenient demonstration of more general principles of machine translation.

Klingon was designed by a linguist, Marc Okrand, who drew elements from numerous real-world languages without closely following any particular syntax. Thus, any given syntactic or morphological feature of Klingon may be found in an actual language which might to be translated; however, Klingon's status as a deliberate linguistic isolate ensures that existing feature grammars cannot simply be re-implemented for the particular requirements of PC-PATR.

While neither the grammar nor the lexicon of Klingon are as large as those of actual spoken languages, it is the largest and most complete fictional language, surpassing even Tolkien's Elvish languages Sindarin and Quenya (Conley and Cain, 2007). Klingon is sufficiently flexible and expressive that the complete text of Hamlet has been translated into it (Schoen et al, 2000)

Klingon phonology and morphology are entirely regular. There are no allophonic or neutralization rules to cause inflected forms to differ from a simple adjunction of root and affix. Furthermore, there are no irregular noun or verb forms. It is true that irregular forms occur in most if not all natural languages, and even an MT implementation designed as a demonstration should not avoid them altogether; however, it is also true that there is little to be learned from them, compared with what can be done with syntax and morphology. As we have seen, this MT implementation does handle irregular English forms.

KLINGON MACHINE TRANSLATION AS A DEMONSTRATION OF REAL-WORLD MT CHALLENGES

At first glance it might appear that Klingon is too simple to make use of the full power of the suite of programs used for this project. As we shall see, however, the specific characteristics of English – such as the raising of deep-structure objects to surface subjects in passive clauses, subject-auxiliary inversion in yes-no questions, and pleonastic subjects, not to mention the numerous spelling rules and irregular forms which must be captured – require careful design to ensure that correct English glosses are generated for a wide range of inputs. By making one half of the language pair fairly regular – if quite dissimilar in syntax and morphology to the target language – it

---

[2] David Trimboli, d'Armand Speers, and Steven Boozer were of invaluable assistance in clarifying gaps left by TKD and identifying other materials by Okrand which expanded on TKD's description of Klingon syntax.

[3] Ria Cheyne, noting that the term "artificial language" is considered derogatory among Esperantists, argues that fictional languages are better described as "created languages" that "constructed languages". "Constructed" implies a degree of regularity and completeness which is not found in all fictional languages, and its use suggests a value hierarchy in which less completely described languages are of less interest. For Cheyne, the important question is not how large a fictional language is compared with a natural language, but, rather, what the evocation of a fictional language communicates to the reader.

becomes possible to demonstrate that even though the ExpLex/PC-PATR/SyntacticTransfer suite is intended for use as a teaching tool, rather than an enterprise-level MT solution, it can in fact be used to implement projects of considerable size and complexity.

KLINGON SYNTACTIC FEATURES NOT PRESENT IN ENGLISH

We shall be treating Klingon morphology and syntax in much more detail in the pages which follow. The following brief list highlights some of the features of Klingon which differ from English and must be taken into account by a machine translation implementation.

- OVS word order
- Pro-drop for both subjects and objects
- Plurals not always explicitly marked
- No copula
- Meanings corresponding to adjectives in English expressed by conjugated one-place verbs
- No articles
- Ambiguity[4]

TERMINOLOGY AND CONVENTIONS

Throughout this document, “ST” is used to mean “source text” and “TT” to mean “target text”. “TKD” refers to *The Klingon Dictionary*.

"Grammar" is used to refer to the feature grammar used by PC-PATR. To avoid ambiguity, the grammars of Klingon and English are referred to as "syntax" or "morphology", as appropriate.

Klingon is given in **bold**, and English glosses in *italic*. Morphological analyses are in SMALL CAPS.

DESIGN GOALS

The grammar should accept well-formed sentences and reject poorly-formed ones; however, no attempt is made to rule out semantically odd sentences.

The emphasis is on producing reasonable English output for a wide range of possible inputs, rather than the optimal output for a smaller number of possible inputs[5]. In some cases, this results in the loss of information that

[4] In at leas one instance, producers re-used a line of Klingon dialogue in a film, giving it a different subtitle. Okrand, who was a consultant on the film, re-worked the Klingon grammar to allow both interpretations. The line was originally **qama'pu' jonta' neH**, "I wanted to capture prisoners." (In one spoken register of Klingon, not treated at length in this paper, verbal prefixes may be dropped; thus, this sentence does not correspond to the verbal inflection rules treated below.) The second translation, "Engines only", was incorporated by allowing the perfective verb form **jonta'** to be homophonous with a noun meaning "engines", and the verb form **neH** to be homophonous with an adverbial form meaning "just". **qama'pu',** "prisoners", became a conjugated verb form with a verbal prefix **qa-** and a perfective suffix **–pu.** The word means "I have accommodated you", and is believed to be a message from Okrand to the producers. I am indebted to Michael Roney Jr. for bringing this example to my attention (Roney, 2009).

[5] In many cases, even the idea of a single optimal output is questionable.

is normally underspecified in English, such as the distinction between plural and singular *you* (note, however, that forms such as *yourself* and *yourselves* are marked for number), or the distinction between inclusive and exclusive *or* (both normally rendered as *or* in English except when unusual precision is needed).

Output sentences are written in Standard English, using a conversational register which includes contractions where appropriate. Most "canonical" Klingon texts – i.e., those written by Okrand – consist of dialogue, where this usage would be expected.

In general, predictable information is captured by rules, and unpredictable information is restricted to the lexicon. As noted above, this principle was used as a guideline, and did not always dictate the final implementation.

In many cases, it is impossible for even a human translator to produce a single TT sentence which is unquestionable the best translation of the ST. Klingon lacks articles, for example (as do many human languages), and does not distinguish gender in the third person (Turkish shows the same pattern). (As Brown et al. note, "Often, knowing the broader context in which *e* occurs may serve to winnow the field of acceptable … translations, but even so, many acceptable translations will remain") (Brown et al, 2003). Given these limitations, no attempt was made to determine whether (for example) "a" or "the" would be most logical in a larger context. Although the implementation discussed in this paper includes no statistical analyses, it does attempt to incorporate Brown's insight that the probability of a given TT as a translation of a given ST increases along with the probability of the TT as a sentence in English.

We will see numerous examples of this strategy in the examples that follow. The handling of article insertion is typical, and worth considering in some depth.

ARTICLES

In English, we can easily construct sentences such as *I saw the man* or *I saw a man*, in which either the definite or the indefinite article is equally natural. With constructions that are typically used to introduce new discourse referents, however, definite articles are permitted but unusual: *I am the officer* is less probable than *I am an officer*. *I have the knife* is acceptable, but *I have the hands* is extremely strange.

As is the case with isolated sentences in many real languages which do not use articles, multiple English translations may be acceptable, and only context or semantic information – neither of which is available to us in a syntactic transfer system – would allow one translation to be objectively more correct than another. In such cases, following Brown, we attempt to produce the most probable English sentence given the component NPs and VPs and their syntactic relationships.

In order to produce natural English target sentences, the following rules are used:

1. If the ST includes an element that would correspond to an English determiner, such as a demonstrative or possessive, no article is added.
2. If the ST is the direct object of a verb with the <predicative> feature equal to "y"[6], an indefinite article is inserted if the NP is singular. No article is inserted if the NP is plural or non-count.
3. If the NP is preceded by a number, or an expression such as *all* or *no*, no article is inserted.
4. Otherwise, the definite article is inserted.

[6] This feature was originally added to handle predicative NPs, but was later extended to the objects of certain verbs.

Effectively, this means that most NPs take the definite article.

'avwl'vaD De' nobbe' qama'

guard-DAT information give-NOT prisoner

The prisoner doesn't give the guards the information

Singular NPs following the copula take the indefinite article.

ml' ghaHbe'

number COPULA-3RD-ANIMATE-SING-NEG

He is not a number

loD tlhab ghaH

man free COPULA-3RD-ANIMATE-SING

He is a free man

Plural NPs following the copula take no article.

Qel maHbe'

doctor COPULA-1ST-PLUR-NEG

We are not doctors

Non-count NPs following the copula take no article.

'Iw SuD 'oH[7]

blood green COPULA-3RD-INANIMATE-SING

It is green blood

Otherwise, they take the definite article.

bIQ bIr luneH

water cold want-SUBJ-3RD-PLUR-OBJ-3RD-SING

They want the cold water

The objects of verbs which typically introduce new discourse items take the indefinite article if singular and no article if plural.

ghItlh chu' Daghajchugh bIbel

---

[7] Klingon does mark third-person singular pronouns as animate or inanimate, which, for this ST. excludes the ambiguous parses we often see with the third-person singular: 'oH must be translated as "it", not "he" or "she".

manuscript new have-SUBJ-2ND-SING-OBJ-3RD

If you have new manuscripts, I am pleased

wep yISuq

coat get-SUBJ-2ND-SING-OBJ-3RD-SING-IMPERATIVE

Get a coat

See also the further discussion of articles in the sections on numbers and noun-noun phrases.

OTHER ENGLISH TRANSLATION CHALLENGES

While it is rare for case to be morphologically realized in English, this marking is obligatory for pronouns. (The English genitive case ending in *'s* can be replaced with a prepositional phrase using *of* for our purposes here; this specific case is discussed in more detail in the section on noun-noun phrases below.)

Klingon nouns do show inflection that indicates their role within a sentence. However, this use of case does not map isomorphically onto English case.

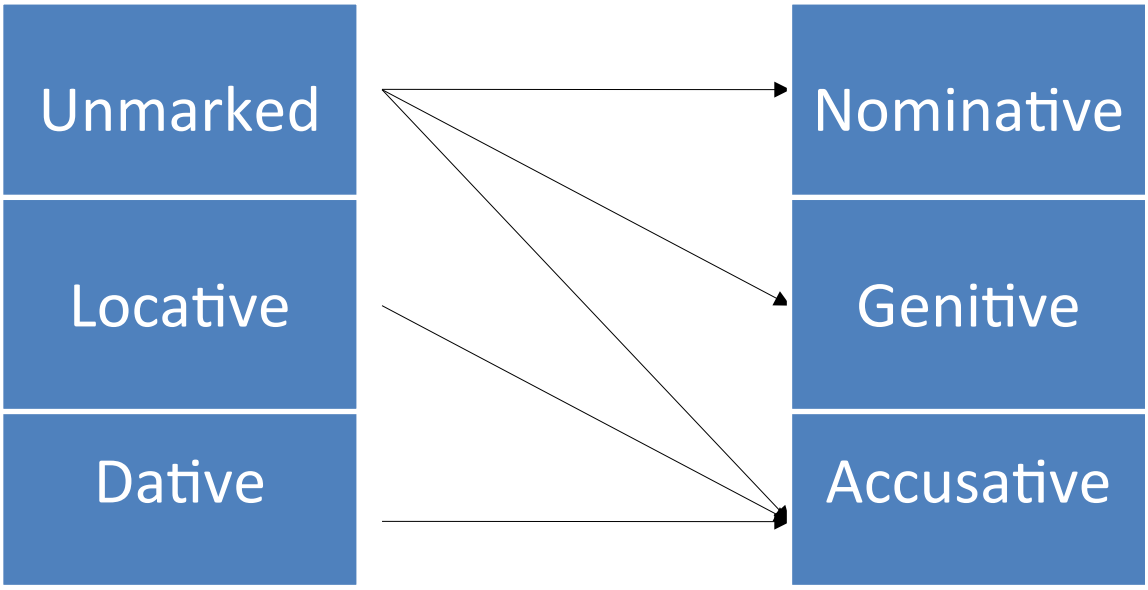

Figure 4. The mapping between Klingon and English case systems.

We must track Klingon case to ensure that sentences are parsed correctly, but this information is not sufficient to determine the case of the TT gloss. The challenge is to distinguish among the possible uses of the unmarked Klingon forms.

The target gloss case may always be determined positionally. The subject of a Klingon verb will correspond to a nominative form in the English TT, the direct object of a Klingon verb will correspond to an accusative form, and the first noun in a noun-noun construction will correspond to a genitive form.

This allows us to use a feature, <gloss-case>, which in our lexicon is underspecified for Klingon nouns with the exception of those pronouns which display morphological alternation in English. This feature is assigned to NPs which are subjects, objects, or occur initially in a noun-noun construction, and is unified throughout the NP branch down to the head noun.

This also allows us to correctly determine the case of nouns followed by adjectives. These nouns are not marked with the Type 5 case suffixes, which appear on the adjectives. For these nouns, the <head-case> feature is set to unmarked, and the <gloss-case> feature is set to the appropriate English case. (In practice, pronouns have not been observed in these positions.)

OUT-OF-VOCABULARY ITEMS

The total set of Klingon root forms in TKD and Okrand's supplemental works[8] contains approximately three thousand entries. It would certainly be possible to include all of them in the lexicon; however, given the long initialization times with the current lexicon of approximately 600 root forms, there seemed to be no particular gain from doing so. (Even in this case, there might be a need to handle out-of-vocabulary items such as proper names; furthermore, Okrand occasionally introduces new Klingon vocabulary.)

Because Klingon has a large number of noun and verb suffixes, it is sometimes possible to identify the part of speech of unrecognized vocabulary items even if they cannot be matched to an English gloss. **–mey**, for example, is a plural noun ending, and we may assume that an OOV item containing **–mey** is a plural noun. (The <head-case> feature is specified as "unmarked" because of other rules, applying before this one, which act on OOV terms ending in case-marking suffixes. Case marking in Klingon will be discussed in greater detail below.)

<OOV> mey, N, <head-agr-number> = plur <head-case> = unmarked</OOV>

Figure 5. An out-of-vocabulary rule using a plural noun ending

Klingon syllable structure is overwhelmingly CVC, and no onsetless syllables are attested. Verbal prefixes, however, are always CV. This allows us to conclude that an unknown vocabulary item beginning in a known verbal prefix followed by a CV sequence is very probably a verb, and to assign it not only the correct part of speech, but also the person and number features for (in most cases) both its subject and its object.

<OOV> ^qa(b|ch|D|gh|H|j|l|m|n|ng|p|q|Q|r|S|t|tlh|v|w|y|')[aeIou], V, <agr-subj-pers> = 1st <agr-subj-number> = sing <agr-obj-pers> = 2nd <agr-obj-number> = sing <mood> = indicative</OOV

Figure 6. An out-of-vocabulary rule for a transitive verb prefix.

If even these methods fail, the out-of-vocabulary word is treated as a noun.

Unicode yaj De'wI'

The computer understands the UnIcoDe

The odd capitalization of "UnIcoDe" is caused by the pre-processor. Lower-case *i* and *d* never occur in Klingon, so these letters are automatically uppercased before processing. (This is discussed further under "Preprocessing" below.)

[8] Including *The Klingon Way* (Okrand, 1996), *Klingon for the Galactic Traveler* (Okrand, 1997), television and movie scripts on which Okrand has consulted, a blessing in Klingon written by Okrand for the non-Star Trek television series Frazier, and private e-mail to Klingon enthusiasts.

PREPROCESSING

Klingon orthography is case-sensitive. **q** and **Q**, for example, represent different phonemes; they are not variants of a single letter. Other letters, such as **D**, **I**, and **S**, never occur in lower-case. **tlh** is an unusual sequence for English-speaking typists and is easily transposed to **thl**. The transfer preprocessor corrects these common typographical errors but is by no means a comprehensive spellchecker.

Sentence-final punctuation is stripped out before processing begins.

THE KLINGON WRITING SYSTEM

The writing system presented as Klingon in the Star Trek films and television shows was not created to reflect Okrand's Klingon phonology, and Klingon writing from these sources is typically a random selection of characters. An employee of Paramount released an informal mapping of these characters to Klingon phonemes (Klingonska Akademien, 2007), and it has been adopted by some users. Formal proposals to include this script in Unicode failed (Unicode, Inc. 2010), but a private mapping is occasionally used (Everson, 2004). A different symbol-to-phoneme mapping of the same characters was used in at least one licensed Star Trek product, a set of trading cards, for texts written by Okrand (Wikipedia, 2009).

Because there are so few texts written in this script, support for it has not been included. It would certainly be possible to translate the first mapping to the ASCII input accepted by PC-PATR with a preprocessor. The second mapping, which has a many-to-one mapping of characters to phonemes, would be non-trivial to implement.

A sample of this writing system is shown below.

 [9]

POSTPROCESSING

The initial letters of the English gloss sentence are capitalized. Commas are added after sentence-initial subordinate clauses and before coordinate conjunctions. Question marks are added after questions, and exclamation points are added to commands. (The syntactic transfer rules use the features <question> and <mood> to identify these sentences.) At present, exclamation points and periods are not added to non-questions which are not commands. Since, in general, the same syntactic rules apply to both exclamations and statements, distinguishing the two would require two sets of transfer rules for every sentence-level rule. It was felt that this step would be best done after all other features have been implemented.

Some English spelling rules are also applied during postprocessing – for example, ensuring that nouns ending in *y* have plural forms ending in *–ies* rather than *–ys*.

GRAMMAR INTRODUCTION

A more detailed treatment of Klingon syntax is found in the section on syntax below. The following is intended to be sufficient to understand the example sentences before that point.

[9] A line from a song by Lieven Litaer, who performs under the name Klenginem. (Litaer, 2006)

Klingon has OVS word order. Nouns which are morphologically unmarked for number may be singular or plural. (This can result in multiple legitimate parses for a single ST.) Both objects and subjects may be dropped.

quS luDel chaH.[10]

chair describe-SUBJ-3RD-PL-OBJ-3RD-SING PRON-3RD-PL-ANIMATE

They describe the chair

quS luDel.

chair describe-SUBJ-3RD-PL-OBJ-3RD-SING

They describe the chair

luDel.

describe-SUBJ-3RD-PL-OBJ-3RD-SING

They describe her

(This could also be *they describe it* or *they describe him*, since the ST object is marked for person and number, but not gender.)

NOUNS

Nouns may take up to one suffix apiece from five suffix classes, which occur in a fixed order.

Klingon arguably has no overt case marking. The equivalent of the English dative case is a noun with the Type 5 suffix **–vaD**, but this suffix is also used for beneficiaries of other actions. Despite this lack of case marking, the analysis grammar does assign case positionally; this ensures that the correct English glosses are chosen for pronouns and that NPs ending in **–vaD** do not occur in impermissible positions (i.e., subjects, direct objects, or the first element in a noun-noun construction.) As we shall see, the **–vaD** ending does not necessarily occur on the head noun of an NP; this is discussed in more detail below.

Unmarked nouns may be interpreted as singular or plural. Although the PC-PATR documentation states that multiple entries may be created from a single lexical item (giving the example of English words such as "deer" yielding both singular and plural) (McConnel, 1995), in practice this feature did not work well enough to be used; correct sentences were correctly parsed, but incorrect sentences yielded both error messages *and parses*. Therefore, this feature was implemented via ExpandLex.pl rules which generated both singular and plural lexical entries for forms lacking explicit plural morphology (discussed below).

SUFFIXES

[10] Punctuation is included on the example sentences to aid clarity. Since it is stripped out by the preprocessor, however, it is not necessary to include it in order to generate a correct parse.

TYPE 1: Augmentative/Diminutive

Type 1 suffixes modify the root noun by indicating that it is more minor or major than its unmarked counterpart. They are not currently implemented, because often the English equivalent of the Klingon noun plus the type 1 suffix is not derivable from the gloss for the root. For example, **juHHom** and **juH'a'** may be derived from the Klingon **juH**, but their glosses *cottage* and *mansion* cannot be derived from the English *home.*

Because Type 1 suffixes always occur immediately after the root, it is easy to include words which use these suffixes and have known glosses in the prelexicon. The normal ExpandLex rules will add Type 2-5 suffixes to them exactly as if they were bare roots.

juHHomwIjDaq chaHtaH puqpu'chaj'e'

house-DIMIN-POSS-INANIMATE-1ST-SING-LOC COP-3RD-PL-ANIMATE-CONTINUOUS child-PL-ANIMATE-POSS-3RD-PL-TOPIC

Their children are in my cottage

It would also be possible to construct a preprocessor for ExpandLex that created nouns with **–'a'** and **–Hom** in only those cases where the resulting compound did not already exist in the lexicon, either adding *great*/*minor* to the gloss or setting a feature that would allow it to be added during transfer. This would help prevent OOV errors even if it might not always produce the most natural English gloss.

TYPE 2: Number

Klingon nouns may belong to three noun classes: beings capable of language[11], body parts, and other. Body parts form plurals exclusively with **-Du'**. Nouns other than body parts are pluralized with **–mey**. Beings capable of language typically are pluralized with **–pu'**. They may also be pluralized with **–mey**, but this carries a sense of *scattered all over* or *all over the place*. Because English would not necessarily convey this meaning as part of the NP, this nuance has been omitted from the gloss.

qamDu' tInqu' boghaj

foot-PL-BODY big-INTENS have-SUBJ-2ND-SING-OBJ-3RD

You have very big feet.

But the following are completely unacceptable:

*** qampu' tInqu' boghaj**

*** qammey tInqu' boghaj**[12]

[11] Although "beings capable of language" is more restricted than "animate", the term "animate" will be used hereafter in the interests of simplicity.

[12] TKD notes that "Klingon poets often violate this grammatical rule" but adds that "Until the subtle nuances of such constructions are firmly grasped … it is suggested that students of Klingon stick to the rules."

SyntacticTransfer treats **qampu'** and **qammey** as out of vocabulary items. It recognizes them as nouns because they contain **-pu'** and **-mey**, but does not associate them with the gloss *feet*, because no plurals for *feet* were generated with these endings.

TYPE 3: Qualification

These suffixes convey such senses as s*o-called*, *apparent*, and *definite*. They are not currently implemented. If they are added later, they will probably be handled by adding features to the lexical items so that the transfer process can add English ADJ tokens to the TT, rather than by changing the gloss.

TYPE 4: Determiners

All Type 4 suffixes correspond to English determiners – either demonstratives or possessives. Definite and indefinite articles are not inserted in NPs which contain Type 4 suffixes.

-vam and **-vetlh** are demonstratives which correspond to *this* and *that*. The transfer rules correctly assign *these* and *those* to plural NPs.

mIDmeyvam chargh Suvwl'pu'vetlh

colony-INANIMATE-PL-DEMONSTR/THIS conquer fight-AGENT-ANIMATE-PL-DEMONSTR/THAT

Those warriors conquer these colonies

Although the ordering of the morphemes in the ST is N+DET ADJ, the transfer rules correctly order the elements in the TT, yielding DET ADJ N.

ylHmeyvetlh tIn DaleghlaH'a'?

tribble-PL-DEMONSTR/THAT big see-SUBJ-3rd-SING-OBJ-3rd-QUES

Can you see those big tribbles?

Klingon has two series of possessive suffixes for the first and second person. One series - **wl'** *my*, **ll'** *your* (singular), **ma'** *our*, and **ra'** *your* (plural) - is only used with animate nouns. The other - **wIj**, **lIj**, **maj**, and **raj** – are usually used with non-animate nouns. Using these suffixes with animate nouns is considered insulting. This is handled with a <derogatory> = "y" feature that percolates[13] up to the sentence level, but does not currently affect the gloss[14]. The third-person possessives, **Daj** *his, her, its* and **chaj** *their*, do not have alternate forms for animate objects.

nuHlIj lutlhap bu'wl'

weapon-INANIMATE-POSS-2ND-SING take-SUBJ-3RD-PL-OBJ-3RD-SING sergeant-ANIMATE-POSS-1ST-SING

My sergeants take your weapon

---

[13] Strictly speaking, of course, features do not percolate in the sense in which this is commonly used in government and binding models. It would be more accurate to say that the <derogatory> feature is underspecified for most items, allowing it to unify with other structures for which it is set.

[14] SyntacticTransfer.pl does not show this feature, but it may be viewed using PCPATR.

nuHwIj lutlhap bu'lIj

weapon-INANIMATE_POSS-1ST-SING take-SUBJ-3RD-PL-OBJ-3RD-SING sergeant-INANIMATE-POSS-2ND-SING

Your sergeants take my weapon (feature <derogatory> set to "y" for sentence, but not reflected in gloss)

The possessives present an interesting contrast between syntactic transfer systems and semantic transfer systems. Although they are clearly 1st, 2nd, or 3rd person, and singular or plural, there is no construction in which they need to be checked for agreement. There is no modeling of co-reference with other NPs in the sentence.

TYPE 5: Syntactic Markers

At present this implementation only handles two suffixes from this class, and those only partially, in order to keep the lexicon file at a manageable size. (The ablative/illative suffix **–vo'**, for example, which expresses movement out of or away from a thing or place, would use rules very similar to those provided for the locative **-Daq**.)

-Daq is a locative marker. It may indicate either the place where an event takes place or a destination, depending on the verb with which it co-occurs.

maSDaq jaghpu' tu'lu'

moon-LOC enemy-ANIMATE-PL find-IMPERS

There are enemies on the moons

maSDaq yIjaH

moon-LOC go-2nd-SING-IMPERATIVE

Go to the moons

Note that the preposition may vary. This is handled by setting features in the lexicon: <prep> for nouns which do not use the default preposition "at", and <verb-of-motion> for verbs which require the preposition "to".

yaHDaq maleS

station-LOC relax-SUBJ-1st-PL-OBJ-NONE

We relax at the stations

Klingon does not use prepositions to express concepts such as "beside" or "below". Instead, a locative NP, with a head noun corresponding to "area beside", "area below" is used.

yuQ DungDaq Duj tu'lu'

planet area-above-LOC ship find-SUBJ-IMPERS-OBJ-3rd-SING

There is a ship above the planet

A literal gloss might be "A ship is found at the area above the planet".

-vaD indicates a beneficiary. It is used for indirect objects. They are assigned dative case, which allows us to ensure that the **–vaD** marker occurs where it is required and not where it is forbidden.

-'e' is a topic marker[15]. It is obligatory in marking what becomes the English subject of certain sentences, as we shall see later. This is currently the only use our contrastive grammar makes of it. Because of its limited distribution – it may only occur on a subject or object NP, or on a topicalized sentence-initial NP – it would be possible for a later release to incorporate this feature by capitalizing the topicalized NP or using a cleft construction.

In Klingon, Type 5 noun suffixes attach to the final lexical item in the NP. As adjectives occur after the noun, the Type 5 suffix may in fact occur on the adjective rather than the head noun. (This is discussed further in the section on adjectives below.)

mIDDaq 'oHtaH Duj'e'

colony-LOC COPULA-INANIMATE-3rd-SING-CONTINUOUS ship-TOPIC

The ship is at the colony

mID chu'Daq 'oHtaH Duj chu''e'

colony new-LOC COPULA-INANIMATE-3rd-SING-CONTINUOUS ship new-TOPIC

The new ship is at the new colony

THE NOUN-NOUN CONSTRUCTION

Two nouns may be juxtaposed so that one modifies the other. Our grammar marks the first noun as genitive, but this is a convenience for the translation rather than a reflection of the ST syntax; there is no overt marking of the relationship between the two.

TKD specifies that both the first and the second NP may take suffixes of types one through four. Type 5 suffixes may only attach to the second noun. TKD often glosses the first noun as a possessive, but this construction cannot be generalized to all possible noun-noun constructions, as the resulting NP is only grammatical in English if the second Klingon noun does not have a Type 4 suffix. In other words, **nuHmey peghmey** may be rendered in perfectly good English as *the weapons' secrets*, but **the weapons' these secrets* for **nuHmey peghmeyvetlh** is ungrammatical. The transfer uses *of* rather than *'s* for most English glosses[16] to prevent the generation of English parses with doubled determiners. (It would be possible to modify the rewrite rules to use the possessive when the second noun does not have a determiner, and restrict *of* to the cases where it does.)

nuHmey peghmey vIyaj

weapon-INANIMATE-PL secret-INANIMATE-PL understand-SUBJ-1st-SING-OBJ-3rd

[15] TKD uses the terms "topic" and "focus" interchangeably when discussing **–'e'**.

[16] As discussed in the section on **–Daq** above, locative NPs in which the head noun is an abstract noun expressing location are translated with prepositional phrases such as "below" rather than expressions using "of": **quS retlhDaq 'ISjaH tu'lu'** *There is a calendar next to the chair* rather than "There is a calendar at the area next to of the chair".

I understand the secrets of the weapons

Type 5 suffixes attach to the second noun in a noun-noun phrase. This allows dative case and focus features to be derived from the second noun.

ghojwl' chaH juppu'wl' puqbe''e'

learn-AGENT COPULA-ANIMATE-3rd-PL friend-ANIMATE-PL-POSS-1st-SING daughter-TOPIC

The daughters of my friends are students

In the current implementation, noun-noun constructions are limited to two component NPs. Three-NP compounds are attested in the canon, but too infrequently to determine whether they are exclusively right-branching.

tlhIngan, *Klingon* or *Klingons*, is an NP in Klingon. An expression such as **tlhIngan Hol**, *the Klingon language*, literally means "the language of the Klingons." This MT implementation treats **tlhIngan** as a special case and inserts an adjective when it is used as the first element of a noun-noun construction.

paw tlhIngan Duj

The Klingon ship arrives

ADJECTIVES

According to TKD, "There are no adjectives as such in Klingon" – one-place verbs serve this function. In practice, this grammar treats attributive adjectives as ADJ rather than V. (We will consider predicative adjectives in the section on verbs.) In the interest of clarity, we will refer to these lexical items as adjectives.

Adjectives follow the noun they modify:

tlhIngan Suvwl'pu' qan tu'lu'be'

There are no old Klingon warriors

Type 5 suffixes (such as –**vaD**) attach to the adjective, rather than the noun. In the sentence bewlow, **puq** *child* is a noun, **QaQ** an attributive adjective. Note that -**vaD** follows **QaQ** rather than **puq**.

puq QaQvaD mu'ghom tIn nob wa' vav Hem

child good-DAT dictionary big give-SUBJ-3rd-SING-OBJ-3rd-PL one father proud

One proud father gives the good child the big dictionaries

According to TKD, attributive adjectives may only take the verbal intensifier suffix **–qu'**, here treated as *very*.

Hom pe'laH taj jejqu'

bone cut-CAN knife sharp-INTENS

The very sharp knives can cut the bones

NUMBERS

Articles are not inserted if the NP includes a numeric quantifier.

jav maS ghajlaH wa' yuQ

moon six have-CAN one planet

One planet can have six moons

Hoch (*all*, *every*) and **pagh** (*no*) are treated as numbers. **Hoch** may be singular or plural; **pagh** is treated as singular.

Dal pagh jagh

no enemy be-boring-SUBJ-3rd-SING-OBJ-NONE

No enemy is boring

COMPUND NPS

When combining NPs, Klingon distinguishes between exclusive *or* **ghap** and inclusive *or* **joq**. The English *or* can cover both cases, though due to the Gricean Maxim of Quantity it is usually understood to mean exclusive *or*. (Chierchia and McConnell-Ginet, 2000) Because spoken English rarely uses this level of precision, "or" is used in both cases.

Compound NPs are always plural. Individual lexical items select certain persons as first or last argument so that the resulting NP will have the correct person.

taD nItlhDu'maj qamDu'maj je

be-frozen-SUBJ-3RD-PL-OBJ-NONE finger-BODYPART-PL-POSS-1st-PL foot-BODYPART-PL-POSS-1st-PL and-NP

Our fingers and our feet are frozen

Dujvetlh DIlegh maH puqpu' je

ship-DEMONSTR/THAT see-SUBJ-1st-PL-OBJ-3rd-PL we child-ANIMATE-PL and-NP

We and the children see those ships

Note that the first-person plural verbal prefix, which unifies with the VP to restrict the person of the subject NP, combines correctly with a compound NP containing one first-person and one third-person NP.

PRONOUNS

Klingon pronouns are marked for person and number. Third-person pronouns distinguish between animate and inanimate referents, but not male and female referents. A complete list is provided in the appendix.

ADVERBS

Most Klingon adverbs are sentence-initial. Two, **neH** *merely, just* and **je** *also, too* occur following the verb.[17] A complete list of adverbs is included in the appendix.

not jegh tlhInganpu'

never surrender-SUBJ-3rd-OBJ-NONE Klingon-ANIMATE-PL

The Klingons never surrender

qama' vIqIp neH

prisoner hit-SUBJ-1st-SING-OBJ-3rd just

I just hit the prisoner

DaH maSagh

now be-serious-SUBJ-1st-PL-OBJ-NONE

We now are serious

toQDujDaq not maSagh

Bird-of-prey-LOC never be-serious-SUBJ-1st-PL-OBJ-NONE

We never are serious on the Bird of Prey

DaHjaj Suvwl''e' jIH

today fight-AGENT-TOPIC COPULA-1st-SING

Ttoday I am a warrior

Not all English adverbs are equally natural in all positions – some are most natural at the beginning of a clause, some at the end, and some before the verb (but after any co-occurring auxiliary). In a future release it would be possible to assign the adverbs to classes based on these categories and use the transfer process to insert them at the correct points. This would, obviously, result in a much larger number of transfer rules.

INTERJECTIONS

There are about a dozen interjections. They can be treated as stand-alone sentences.

HISlaH

Yes

majQa'

[17] The sentence **not qama' vIqIp neH,** *I never just hit the prisoner*, which has both a sentence-initial and a VP-final adverb, is parsed correctly by PC-PATR, but not re-ordered by SyntacticTransfer despite the presence of apparently legal syntactic transfer rules. This seems to be caused by a bug in Syntactic Transfer.

Very good

A complete list may be found in the appendix.

VERBS

GENERAL NOTES

Klingon does not explicitly mark tense. The current implementation of the software uses present tense so as not to create numerous parses which are legitimate, but cannot be distinguished from each other. In a later release, it would be possible to default to present tense while using past tense if a past adverbial expression is present. This would require much larger number of verb rules, however.

Klingon verbs are inflected with prefixes indicating subject and object agreement, nine ordered suffix categories, and a small number of scoped suffixes which may occur anywhere after the verb.

THE COPULA

Klingon does not have a copula per se. In the simplest cases, the appropriate pronoun and verb may co-occur:

tej po' chaH

scientist expert PRON-ANIMATE-3rd-PL

They are expert scientists

DaH qama'ma' tlhIH

now prisoner-POSS-1st-PL PRON-2nd-PL

Now you are our prisoners

This might appear to be a null copula, which occurs in many real-world languages. However, the pronoun may also take verbal endings. For this reason, when this form occurs in a context which would be translated in English as either the copula or the copula with a pronominal subject, the morphological analysis has been marked as COPULA.

loD Quch ghaHbe'

man happy COPULA-ANIMATE-3rd-SING-NEG

He is not a happy man[18]

ta' jIHlaHbe'chugh yaS SoHlaHbe'

emperor COPULA-CAN-1st-SING-NEG-IF officer COPULA-2nd-SING-CAN-NEG

If I can't be an emperor, you can't be an officer

[18] Contrast the placement of the negative morpheme **-be'** in **Quchbe' ghaH**, *He is not happy*.

If the subject is a noun, the pronominal verb must still be included, and the subject NP must take the focus/topic ending. A more literal (and less natural) English gloss might be *As for the weapon, it is a sword.*

'etlh 'oH nuH'e'

sword COPULA-INANIMATE-3rd-SING weapon-TOPIC

The weapon is a sword

'etlh bIH nuH'e'

sword COPULA-INANIMATE-3rd-PL weapon-TOPIC

The weapons are swords

Note that even without explicit plural morphology, the number of the subject and object NP is disambiguated by the number of the pronoun used as a verb[19].

The copula may also be used to indicate a location.

pa'wIjDaq 'oHtaH De'wI''e'

room-POSS-1st-SING-LOC COPULA-INANIMATE-3rd-SING-CONTINUOUS computer-TOPIC

The computer is in my room

The **–taH** ending is a Type 7 verbal suffix denoting aspect. In Okrand's texts, equivalents of "to be" denoting identity most often occur without this marker, and equivalents of "to be" denoting spatial location most often occur without it. However, Stephen Boozer was able to find examples of the opposite usage in both cases. They have therefore been treated as equivalent for the purposes of English translation. (If we were translating from English into Klingon, of course, we would want to take this pattern into account.)

PREFIXES

Klingon verbs are marked for subject and object agreement via prefixes.

INTRANSITIVE

Subject	**Prefix**
I	**jI-**
You (singular)	**bI-**

[19] There does not seem to be any grammatical reason that the subject and predicate NPs must agree in number; **pu'beq chaH**, *they are a phaser crew*, might be acceptable. In practice, allowing a mismatch in number between the two NPs leads to a large number of unwanted parses, due to the fact that nouns unmarked for plurality may be either singular or plural. In the interest of excluding these frequent multiple interpretations, agreement in number between subject and predicate has been enforced even though this might exclude the occasional legitimate parse. This problem is treated in more detail in the "Limitations of the Current Software" section below.

He/she/it	**-**
We	**ma-**
You (plural)	**Su-**
They	**-**

Table 1. Intransitive verbal prefixes (also used with transitive verbs for which no object is specified).

Note that the null prefix indicates a third-person subject, but is underspecified for number.

The transfer rules insert subject pronouns if no subject is specified.

maqeq

practice-SUBJ-1st-PL-OBJ-NONE

We practice

Suratlh

remain-SUBJ-2nd-PL-OBJ-NONE

You remain

They do not insert pronouns if an explicit subject is specified.

maqeq maH

practice-SUBJ-1st-PL-OBJ-NONE PRON-1st-PL

We practice

Suratlh tlhIH

remain-SUBJ-2nd-PL-OBJ-NONE PRON-2nd-PL

You remain.

jegh

surrender-SUBJ-3rd-OBJ-NONE

She surrenders

jegh nuch

surrender-SUBJ-3rd-OBJ-NONE coward

The coward surrenders

If the verb's prefix does not agree with a noun which must be interpreted as a subject, the sentence will not parse. SyntacticTransfer automatically provides the glosses of all identified words even if PC-PATR cannot derive a parse tree from the input.

Supum Sor

* fall-SUBJ-2nd-PL-OBJ-NONE tree

VERBS GLOSSED WITH "TO BE"

As discussed above, predicative adjectives are one-place verbs. In Klingon grammar, they are indistinguishable from other intransitive verbs; however, they must be handled differently in translation to ensure that the resulting gloss has the correct form of the copula. (This step is handled by the syntactic transfer rules.)

Quch ghojwI'pu' Qatlhbe'chugh pab

happy-SUBJ-3RD-PL-OBJ-NONE learn-AGENT-ANIMATE-PL be-difficult-SUBJ-3rd-OBJ-NONE-NEG-IF grammar

the students are happy if the grammar is not difficult

TRANSITIVE

Object	Me	You	Him/her/it	Us	You (plural)	Them
Subject						
I	[blocked]	**qa-**	**vI-**	[blocked]	**Sa-**	**vI-**
You (singular)	**cho-**	[blocked]	**Da-**	**ju-**	[blocked]	**Da-**
He/she/it	**mu-**	**Du-**	**-**	**nu-**	**lI-**	**-**
We	[blocked]	**pI-**	**wI-**	[blocked]	**re-**	**DI-**
You (plural)	**tu-**	[blocked]	**bo-**	**che-**	[blocked]	**bo-**
They	**mu-**	**nI-**	**lu-**	**nu-**	**lI-**	**-**

Table 2. Transitive verb prefixes.

Transitive verbs agree with both their subjects and their objects in number and person. Combinations of subject and object labeled "blocked" in Table 2 are expressed using the Type 1 reflexive suffixes. Both subject and object pronouns are inserted if these NPs are not specified.

choyaj

understand-SUBJ-2nd-SING-OBJ-1st-SING

You understand me.

jurur

resemble-SUBJ-2nd-SING-OBJ-1st-PL

You resemble us.

ta' magh yaS

emperor betray-SUBJ-3rd-SING-OBJ-3rd-PL officer

The officer betrays the emperors

magh

betray-SUBJ-3rd-SING-OBJ-3rd-SING

She betrays her

lubej

watch-SUBJ-3rd-PL-OBJ-3rd-SING

They watch her

matHa' lubej yaS

gunner watch-SUBJ-3rd-PL-OBJ-3rd-SING officer

The officers watch the gunner

NON-SPECIFIED OBJECTS

The intransitive prefixes may be used with transitive stems "when an object is possible, but unknown or vague" (TKD). In some cases, English transitive verbs also have an intransitive sense which is equivalent. This has been used when available.

jIyajbe'

understand-SUBJ-1st-SING-OBJ-NONE-NEG

I don't understand

This contrasts with

vIyajbe'

understand-SUBJ-1st-SING-OBJ-3rd-NEG

I don't understand her

in which some specific person is intended, even if no object NP is supplied.

When the English gloss verb may not be used without an object, *something* is added to the gloss.

maSeH

control-SUBJ-1st-PL-OBJ-NONE

We control something (indefinite object)

wISeH

control-SUBJ-1st-PL-OBJ-3rd-SING

We control her (definite, but unspecified, object)

IMPERATIVE

Subject	Object	None	Me	Him/her/it	Us	Them
You (sing.)		**yI-**	**HI-**	**yI-**	**gho-**	**tI-**
You (pl.)		**pe-**	**HI-**	**yI-**	**gho-**	**tI-**

Table 3. Imperative prefixes.

Imperative verbs use a separate set of verb prefixes. The subject pronoun "you" is not inserted in imperative sentences. Note that all imperative prefixes which indicate objects are identical in the singular and plural – thus, many imperative sentences will have two parses.

Because, in general, the English glosses do not reflect the number of the subject, this feature is left unspecified for prefixes which are identical for both singular and plural subjects. This prevents an unnecessary multiplication of morphological rules and resulting parses.

The English glosses do reflect number when the verb is combined with a reflexive or reciprocal object, expressed in Klingon with a Type 1 verbal suffix. Examples are provided in the section on Type 1 verbal suffixes below.

ghoDev

lead-SUBJ-2nd-OBJ-1st-PL-IMPERATIVE

Lead us

yaSpu' tIHoH

officer-ANIMATE-PL kill-SUBJ-2nd-OBJ-3rd-PL-IMPERATIVE

Kill the officers

The suffix **–Qo'** negates imperatives. It follows all other verb suffixes except for Type 9 suffixes.

yISaQQo'

cry-SUBJ-2nd-SING-OBJ-NONE-IMPERATIVE

Don't cry

SUFFIXES

Verbs may take suffixes of nine types, which occur in a strict order. At most one suffix of any type may appear. Certain other suffixes are officially classified as "rovers". For example, **–Qo'**, which negates imperatives as discussed above, follows any suffixes of Types 1 through 8 and may be followed by a Type 9 suffix.

Note that not all of the nine classes are currently implemented, and in some cases, only a subset of the possible suffixes in a given class is supported. In general, suffixes were selected because they posed interesting problems for implementation.

Although TKD does not classify the ending **–Ha'** *undo* in one of the nine numbered classes, it always occurs immediately after the verb, changing its meaning – for example, **jot** *be calm* becomes **jotHa'** *be uneasy.* As was the case with the augmentative and diminutive noun suffixes, the English gloss is unpredictable, and the derived form – with the derivational morpheme occurring immediately after the root – may be treated as any other verb. Therefore, verbs with the **–Ha'** suffix have separate lexical entries[20].

The negative morpheme **–be'** may occur anywhere in the suffix sequence and has leftward scope, as seen in the following examples.

Suvangvlp

act-SUBJ-2nd-PL-OBJ-NONE-AFRAID

You are afraid to act.

Suvangvlpbe'

act-SUBJ-2nd-PL-OBJ-NONE-AFRAID-NEG

You aren't afraid to act.

Suvangbe'vlp

act-SUBJ-2nd-PL-OBJ-NONE-NEG -AFRAID

You are afraid not to act.

The placement of -**be'** with respect to other suffixes can cause significant changes in the English gloss, as we will see.

TYPE 1: REFLEXIVES AND RECIPROCALS

Reflexive verbs are formed by adding the **–'egh** *oneself* and **–chuq** *each other* endings. These may only be added to verbs which are transitive or ditransitive in their root forms and may be thought of as saturating the PATIENT theta rôle. Verbs with Type 1 suffixes, despite being two-place predicates, must take intransitive suffixes.

[20] As was the case with Type 1 noun suffixes, it would be possible for a lexical pre-processor to generate lexical entries in **–Ha'** for verbs which lacked them. Glosses with *mis-* or *un-* would be awkward, but still more understandable than treating these words as OOV.

yIbach'eghQo'

shoot-SUBJ-2nd-SING-REFLEXIVE-IMPERATIVE-NEG

Don't shoot yourself

pebach'eghQo'

shoot-SUBJ-2nd-PL-REFLEXIVE-IMPERATIVE-NEG

Don't shoot yourselves (Note difference between singular and plural, indicated in Klingon by the verb prefix, and in English by the plural marking on the pronoun.)

pebachchuqQo'

shoot-SUBJ-2nd-PL-RECIPROCAL-IMPERATIVE-NEG

Don't shoot each other

Existing canonical sources do not specify how reciprocal or reflexive meanings are expressed in positions other than that of direct object. "They give each other chocolate" is not currently translatable.

TYPE 2: VOLITION

Three verbs from this class are currently implemented. (The other two would not present any particular technical challenge and were only omitted to keep the lexicon manageable.)

-nIS indicates necessity. Note its scope interactions with **-be'**.

mavoqchuqnISbe'

trust-SUBJ-1st-PL-RECIPROCAL-NEED-NEG

We don't need to trust each other

mavoqchuqbe'nIS

trust-SUBJ-1st-PL-RECIPROCAL-NEG-NEED

We should not trust each other

-qang indicates willingness.

Sujeghqang

surrender-SUBJ-2nd-PL-OBJ-NONE-WILLING

You are willing to surrender.

Sujeghqangbe'

surrender-SUBJ-2nd-PL-OBJ-NONE-WILLING-NEG

You aren't willing to surrender.

Sujeghbe'qang

surrender-SUBJ-2nd-PL-OBJ-NONE-NEG-WILLING

You are willing to refrain from surrendering.

-vIp indicates fear, as seen in the examples for **–be'** above[21].

TYPE 3: CHANGE OF STATE

-choH indicates a change of state. With action verbs we gloss it as *begin*, and with verbs of qualities as *become*. These glosses are generated by the appropriate ExpandLex rules.

maDo'choH

lucky-SUBJ-1st-PL-OBJ-NONE-CHANGE

We are becoming lucky

veng boQaw'choH

city destroy-SUBJ-2nd-PL-OBJ-3rd-PL-CHANGE

You begin to destroy the cities.

Type 1 and 2 suffixes may co-occur with **–choH**. Type 2 suffixes in particular show interactions which must be reflected in the gloss.

veng boQaw'qangchoH

city destroy-SUBJ-2nd-PL-OBJ-3rd-PL-WILLING-CHANGE

You become willing to destroy the cities.

SuHoHchuqnISchoH

kill-SUBJ-2nd-PL-RECIPROCAL-NEED-BEGIN

You begin to need to kill each other.

TYPE 4: CAUSATION

[21] TKD notes that there is a cultural taboo against the use of **–vIp** with first-person subjects; however, the enforcement of cultural taboos is not a design goal of this software.

-moH indicates causation. There are two types of English glosses which typically correspond to verbs with **–moH**. The first is *to cause someone to*; this is not currently implemented.

The second may be thought of as converting one-place predicates into two-place predicates. Compare the intransitive **tet** *melt* and the transitive **tetmoH** *melt* in the following examples:

tet yuch

melt-SUBJ-3rd-SING-OBJ-NONE chocolate

the chocolate melts

yuch tetmoH tuj

chocolate melt-SUBJ-3rd-SING-OBJ-3rd-SING-CAUSE heat

the heat melts the chocolate

In many cases, the English gloss cannot be automatically derived from the bare root. For example, **ghoj** is *learn*, and **ghojmoH** is *teach*.

Note that Type 1 through Type 3 suffixes occur between the root and **–moH**.

yuch tetnISmoH vutwl'

chocolate melt-SUBJ-3rd-SING-OBJ-3rd-SING-NEED-CAUSE cook-AGENT

the cook needs to melt the chocolate

TYPE 5: IMPERSONAL SUBJECT/ABILITY

The **–lu'** suffix expresses an "unknown, indefinite, and/or general" subject and is "often translated into the English passive voice" (TKD). Verbs with **–lu'** may never occur with a subject NP; an ungrammatical sentence such as ***Daqawlu' ta'** will not parse.

-lu' occurs only with the subset of transitive prefixes which indicate a third-person singular object, and the grammatical subject of the prefix (in its usual usage) agrees with the *object* of the verb taking **–lu'**. This is illustrated in the following examples:

vIvan

salute-SUBJ-1st-SING-OBJ-3rd-SING

I salute her

vIvanlu'

salute-SUBJ-1st-SING-OBJ-3rd-SING-IMPERS

I am saluted

If the verb occurs with an NP object, that object must be raised to the subject position in the English gloss[22], and no subject pronoun may be inserted. This may be seen in the following sentences with and without **–lu'**.

Suvwl'pu' qan jeybe' jaghpu' qu'

fight-AGENT-ANIMATE-PL old defeat-SUBJ-3rd-PL-OBJ-3rd-PL-NEG enemy-ANIMATE-PL fierce

The fierce enemies do not defeat the old warriors (**Suvwl'pu' qan** *old warriors* occurs first and is the grammatical object; it remains the object in the English gloss)

Suvwl'pu' qan lujeylu'be'

fight-AGENT-ANIMATE-PL old defeat-SUBJ-3rd-PL-OBJ-3rd-PL--IMPERS-NEG

The old warriors aren't defeated (**Suvwl'pu' qan** *old warriors* is still the grammatical object; it becomes the subject in the English gloss)

lujeylu'be'

defeat-SUBJ-3rd-PL-OBJ-3rd-PL--IMPERS-NEG

They aren't defeated (Pronoun insertion only occurs if there is no subject NP present)

-lu' may co-occur with other suffixes.

DaHoHnISlu'

kill-SUBJ-2nd-SING-OBJ-3rd-SING-NEED-IMPERS

You need to be killed

The other Type 5 suffix, **-laH**, conveys the ability to do something.

bIlujlaHbe'chugh bIQaplaHbe'

fail-SUBJ-2nd-SING-OBJ-NONE-CAN-NEG-IF succeed-SUBJ-2nd-SING-OBJ-NONE-CAN-NEG

If you can't fail, you can't succeed

Note that unlike the roving suffix **–be'**, fixed-order suffixes do not necessarily have leftward scope.

choSuvnISlaH

---

[22] Strictly speaking, of course, this process is not raising in a government and binding sense. The constituent is not being moved to an empty position in a hierarchical tree, but re-ordered with respect to other consituents at the same level. (As can be seen in Appendix B, the feature-structure grammar does not produce binary-branching trees.) However, the raising model is useful in several respects. First, the syntactic transfer process may be thought of as a transformation of an underlying form (the ST) to a surface form (the TT). Second, by using an existing model for human language as a guideline when creating an MT grammar, we can make use of existing work on complex behavior in English to ensure that our computational system works as well as the pre-existing theory.

fight-SUBJ-2nd-SING-OBJ-1st-SING-NEED-CAN

You need to be able to fight me.

TYPE 6: QUALIFICATION

The type 6 suffixes convey the speaker's certainty concerning the proposition expressed by the clause. No suffixes of this type are currently implemented. If they are added later, they will be glossed with adverbs such as *clearly* or *undoubtedly*, which will probably be inserted in the transfer phase based on features generated by the ExpandLex rules.

TYPE 7: ASPECT

The Type 7 suffixes convey perfective and continuative aspect. They are not implemented (except in the case of pronominal verbs, described above). Verbs which are unmarked for aspect – i.e., all the verbs handled by this program – "are translated by the English simple present tense" (TKD). Implementing these suffixes would require a careful analysis of the existing corpus.

TYPE 8: HONORIFIC

In contrast with the Type 7 aspect suffixes, the honorific –**neS** suffix would be trivial to implement, but doing so would double the number of lexical entries for verbs without changing any of their glosses[23]. A preprocessor rule could be added to strip out **-neS**.

TYPE 9: SYNTACTIC MARKERS

-chugh creates a subordinate clause. The English gloss uses *if*.

cheHoHqangchugh ghoHIv

kill-SUBJ-2nd-PL-OBJ-1st-PL-WILLING-IF attack-SUBJ-2nd-OBJ-1st-PL-IMPERATIVE

If you are willing to kill us, attack us!

Subordinate clauses will be discussed further in the section on syntax.

The suffix **-'a'** marks a yes/no question. The transfer process inserts an appropriate auxiliary item, and, if necessary, a subject pronoun.

bItlhuHnIsbe''a'

breathe-SUBJ-2nd-SING-OBJ-NONE-NEED-NEG-QUES

Don't you need to breathe? (Note that the negation indicated by the suffix **–be** is preserved in the English gloss.)

veS lutlv'a' tlhIngan

---

[23] Furthermore, TKD notes that **–neS** is rarely used.

war enjoy-SUBJ-3rd-PL-OBJ-3rd-SING-QUES Klingon

Do the Klingons enjoy the war?

Dojbe'nIS'a' bortaS?

be-impressive-SUBJ-3rd-SING-OBJ-NONE-NEED-QUES revenge

Shouldn't the revenge be impressive?

qIj'a' 'oH?

be-black-SUBJ-3rd-SING-OBJ-NONE-QUES PRON-INANIMATE-3rd-SING

Is it black?

SuHeghqangbe''a' tlhIH?

die-SUBJ-2nd-PL-OBJ-NONE-WILLING-NEG-QUES PRON-2nd-PL

Aren't you willing to die?

DIlon'a' neH?

abandon-SUBJ-1st-PL-OBJ-3rd-PL-QUES just

Do we just abandon them? (Note the insertion of the auxiliary and pronoun and their ordering with respect to the adverb, which is moved from VP-final position.)

SYNTAX

DITRANSITIVE VERBS

Klingon has two syntactic structures for expressing ditransitivity. In the long form, the indirect object is realized as an NP with the **–vaD** Type 5 suffix, discussed in more detail above; in the other, the indirect object is realized on the verb prefix.

LONG FORM

In the long form, the direct object is optional. If there is no explicit direct object NP, the syntactic transfer rules addsan appropriate pronoun.

pInvaD 'ISjaH bonob

boss-DAT calendar give-SUBJ-2nd-PL-OBJ-3rd-PL

You give the boss the calendars.

pInvaD bonob

boss-DAT give-SUBJ-2nd-PL-OBJ-3rd-PL

You give them to the bosses.

SHORT FORM

In the long form, the direct object is optional. The feature which is usually used to ensure agreement with the direct object is instead used by the syntactic transfer rules to generate an appropriate indirect object pronoun.

'ISjaH tunob

calendar give-SUBJ-2nd-PL-OBJ-3rd-PL

You give me the calendars.

SUBORDINATE CLAUSES

A subordinate clause may precede or follow the matrix clause. If it precedes the matrix clause, the syntactic transfer rules insert a comma following it.

Sujeghchugh SaHoHbe'qang

surrender-SUBJ-2nd-PL-OBJ-NONE-IF kill-SUBJ-1st-SING-OBJ-2nd-PL-NEG-WILLING

If you surrender, I am willing to refrain from killing you

SaHoHbe'qang Sujeghchugh

kill-SUBJ-1st-SING-OBJ-2nd-PL-NEG-WILLING surrender-SUBJ-2nd-PL-OBJ-NONE-IF

I am willing to refrain from killing you if you surrender

EMBEDDED SENTENCES

Embedded sentences are followed by the complementizer **'e'**[24].

vubpu' DIbach 'e' DaSov

hostage-ANIMATE-PL shoot-SUBJ-1st-PL-OBJ-3rd-PL COMPL known-SUBJ-2nd-SING-OBJ-3rd-SING

The expected gloss would be

You know that we shoot the hostages

Although these sentences are currently parsed correctly by PC-PATR, they are only partially re-ordered by SyntacticTransfer. This appears to be a bug rather than an error in the transfer rules.

[24] A second complementizer with a more restricted distribution is not yet implemented, because it would presumably be subject to the problems with **'e'** described above. Three verbs, **neH**, **ja'**, and **jatlh**, do not use the complementizer; they are not currently implemented.

Many Klingon sentences using **'e'** are equivalent to English sentences using big PRO. Klingon, unlike English, uses a finite tense in such clauses. **majaH 'e' wIwuqpu'**, "we decided to go", is marked with first-person plural subject agreement on both verbs. Although this construction is not currently supported, it would be possible to extend the grammar to handle such sentences correctly once the source of the reordering problems affecting embedded are identified.

LIMITATIONS OF THE CURRENT SOFTWARE

Numerous constructions, such as relative clauses, wh-questions, and comparatives, are simply not implemented at this time. In some cases, this is because of as-yet unresolved problems (for example, the syntactic transfer rules dealing with comparatives appear to be correct, but are not applied by SyntacticTransfer.pl; this might be related to other known issues with constructions containing two of the same constituent at the same level.)

A larger problem, however, is the production of multiple legitimate but unwanted parses which cannot be pruned syntactically.

Consider, for example, the unremarkable sentence

lojchugh yuch yuch yIje'

The desired parse would be something along the lines of *if the chocolate is all gone, buy chocolate*. However, the grammar finds four different parses for this sentence (and it found even more before **je'** *feed*, a homonym for **je'** *buy*, was removed from the lexicon). **yuch yuch** may be parsed as a noun-noun construction, and if so, it could be either the subject of **yuch** or the object of **yIje'**. While this is semantically ruled out in this case, other sentences could be constructed for which this would be the desired parse.

In part this is because Klingon allows both subjects and objects to be dropped, and because there is little explicit case marking to tie the nouns to the subject, object, or modifier positions[25]. Furthermore, the vast majority of Klingon root words are CVC monosyllables, resulting in a limited sound space in which most possible syllables have been used as roots and new words are likely to be homophonous with existing ones. As a result, even small vocabularies quickly develop lexical ambiguity.

However, it must be noted that these problems exist for real-world language pairs as well. It is well-known that as grammars are enlarged, they find more legitimate but low-probability parses. This also suggests the solution: If probabilities can be assigned to parses, the most probable parse may be selected from the set of possible parses.

Re-implementing the feature-structure grammar using a probabilistic CFG parser, such as the chart-parsing modules provided with the Natural Language Toolkit, would permit the assignment of weights to rules, allowing only the most probable parse to be selected.

It is not clear that there is enough canonical Klingon data to train a statistical parser; however, without some means of ranking parses for probability, syntactic (and lexical) ambiguity will continue to limit the usefulness of these translations.

PERFORMANCE

[25] Real-world spoken languages often have prosodic marking of these boundaries, and many writing systems use punctuation to delimit them. TKD does not use commas at these boundaries.

The lexicon file for this implementation is over 80MB in size, and file I/O is a major bottleneck for performance. A different file format might speed initialization. This might, however, be an argument for augmenting PC-PATR with a two-level morphological analyzer that extracts features from roots.

The Stuttgart Finite-State Transducer software package, SFST, is one open-source package that implements these features. The SFST compiler reads one or more source files to produce a highly efficient compiled finite-state transducer. Re-implementing the current implementation using a finite-state morphological parser clearly goes beyond the original scope of the project, but would be possible. The sample SFST code included in Appendix D represents a more complete modeling of Klingon morphology than would be reasonably possible with the current architecture, given its performance limitations.

BIBLIOGRAPHY

Boozer, Stephen, writing as Voragh. 2009. Personal communication.

Brown, Peter F., et al. "The Mathematics of Machine Translation: Parameter Estimation." *Computational Linguistics* 19.2 (1993): 263-311. Print.

Cheyne, Ria. "Created Languages in Science Fiction." *Science Fiction Studies* 35.3 (2008): 386-403. Print.

Chierchia, Gennaro, and Sally McConnell-Ginet. *Meaning and Grammar: An Introduction to Semantics*. 2nd ed. Boston: MIT, 2000. Print.

Conley, Tim, and Stephen Cain. *Encyclopedia of Fictional and Fantastic Languages.* Westport, CT: Greenwood P, 2006. Print.

Dorr, Bonnie J., Eduard H. Hovy, and Lori S. Levin. "Natural Language Processing and Machine Translation." *Encyclopedia of Language and Linguistics*. Boston: Elsevier, 2006. Print.

Everson. Michael. "Klingon: U+F8D0 - U+F8FF." *Conscript Unicode Registry*. 2004. Web. 11 Mar. 2010. <http://www.evertype.com/standards/csur/klingon.html>.

Jurafsky, Daniel, and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2006. Print.

Klingonska Akademien. *pIqaD, And How To Read It*. 2007. Web. 11 Mar. 2010. <http://klingonska.org/piqad/ >.

Litaer, Lieven, writing as Klenginem. "Lyrics." *Klenginem Official Website*. 2006. Web. 11 Mar. 2010. <http://www.klenginem.de/e/lyrics10.html >.

McConnel, Stephen. *PC-PATR Reference Manual*. 1995. Web. 11 Mar. 2010. <http://www.sil.org/pcpatr/manual/pcpatr.html>.

Okrand, Marc. *The Klingon Dictionary*. New York: Pocket Books, 1992. Print.

---. *Star Trek: The Klingon Way*. New York: Pocket Books, 1996. Print.

---. *Star Trek: Klingon for the Galactic Traveler*. New York: Pocket Books, 1997. Print.

Roney, Michael Jr., writing as naHQun. 2009. Personal communication.

Schoen, Lawrence, and the Klingon Language Institute. *The Klingon Hamlet*. New York: Pocket Books, 2000. Print.

Speers, d'Armand, writing as Holtej. 2009. Personal communication.

Trimboli, David, writing as SuStel. 2009. Personal communication.

Unicode, Inc. "Archive of Notices of Non-Approval." 25 Feb. 2009. Web. 11 Mar. 2010. <http://unicode.org/alloc/nonapprovals.html >.

Wikipedia. "Klingon Writing Systems." 22 Aug. 2009. Web. 11 Mar. 2010. < http://en.wikipedia.org/wiki/Klingon_writing_systems#Skybox_pIqaD>.

APPENDIX A: ROOT FORMS

In the interest of clarity, not all possible glosses are shown. Multiple glosses have sometimes been omitted (*I*, *me*) or combined (*he*, *she*). Parenthetical information is intended for clarification and is not part of the lexicon's glosses.

Entries are in Klingon alphabetical order:

a b ch D e gh H j l m n o ng p q Q r S t tlh u v w y ‘

Note that **‘** is a separate letter, as are **q** and **Q**. **ch**, **gh**, **ng** and **tlh** are treated as single letters.

PRONOUNS

jIH	I	**ghaH**	he, she
maH	we	**'oH**	it
SoH	you (singular)	**chaH**	they (animate only)
tlhIH	you (plural)	**bIH**	they (inanimate only)

CONJUNCTIONS

These may only join NPs. They may not join sentences.

je and

joq or

ghap or

NOUNS

ANIMATE NOUNS

avwl'	guard	**bu'**	sergeant	**jup**	friend
baHwI'	gunner	**DeghwI'**	helmsman	**loD**	man
bang	loved one	**ghojwI'**	student	**loDHom**	boy
be'	woman	**ghot**	person	**loDnal**	husband
be'Hom	girl	**HoD**	captain	**loDnI'**	brother
be'nal	wife	**jagh**	enemy	**matHa'**	gunner
be'nI'	sister	**joH**	lord	**ngan**	inhabitant

nuch coward

nuv person

pIn boss

puq child

puqbe' daughter

puqloD son

qama' prisoner

qoH fool

Qel doctor

SoS mother

Suvwl' warrior

ta' emperor

tej scientist

tera'ngan human

tlhIngan Klingon

vaj warrior

vav father

vavnl' grandfather

verengan Ferengi

vub hostage

vunqangan Vulcan

vutwl' cook

wIgh genius

yaS officer

INANIMATE COUNTABLE NOUNS

bach shot

chal sky

chav achievement

cha'pujqut dilithium crystal

DaS boot

Degh helm

DeS arm

De'wl' computer

Dip noun

Dir skin

Doch thing

DoS target

Duj ship

Du' farm

ghIch nose

ghItlh manuscript

ghom group

ghop hand

ghum alarm

Hal source

Hatlh countryside

HIvje' glass

jar month

juH house

juH'a' mansion

mu'tlhegh sentence

juHHom cottage

Ha'DIbaH animal

Hogh week

Hol language

Ho' tooth

jaj day

laH ability

latlh other one

lojmIt door

lut story

maS moon

may' battle

meq reason

mer surprise

mID colony

ml' number

mu' word

mu'ghom dictionary

nab plan

nach head

nav paper

nep lie

nIQ breakfast

nItlh finger

nob gift

nov alien

nuH weapon

nuj mouth

ngach debate

ngem forest

ngeng lake

nger theory

ngoD fact

ngoQ goal

pab grammar

pa' room

paq book

pat system

pegh secret

peng torpedo

pong name

porgh body

pov afternoon

puch toilet

pu'beq phaser crew

qach building

qam foot

qech idea

qep meeting

qevpob cheek

qIv knee

qoS birthday

qun history

quS chair

Qapla' success

Qe' restaurant

QumwI' communicator

Qu' duty

ram night

ra' order

rav floor

rep hour

roj peace

Saj pet

Sor tree

SuS wind

tach bar

taj knife

ta' accomplishment

tup minute

tlhaq chronometer

tlhegh rope

tlhon nostril

veng city

vergh dock

veS war

vIj thruster

vuD opinion

wanI' event

waq shoe

wep coat

wot verb

yab brain

yaH station

yay victory

yIH tribble

yIn life

yoD shield

yuQ planet

'eb opportunity

'eng cloud

'etlh sword

'Ip oath

'ISjaH calendar

'uQ dinner

'uS leg

NON-COUNT NOUNS

batlh honor

bel pleasure

bIQ water

boQ aid

bortaS revenge

chuch ice

Dap nonsense

De' information

ghong abuse

mu'tay' vocabulary

pagh nothing

qeS advice

qul fire

QeD science

Qob danger

Seng trouble

Sip gas

Soj food

veQ garbage

yuch chocolate

'Iw blood

VERBS

INTRANSITIVE VERBS

ba' sit

chagh drop

chegh return

choH drop

chung accelerate

Ding spin

ghor break

Hagh laugh

Hegh die

HuS hang

jach scream

jaH go

jegh surrender

leng travel

leS relax

mej leave

moD hurry

ngab disappear

ngor cheat

ngoS dissolve

nup decrease

paw arrive

pum fall

qaS happen

qeq practice

qet run

qIm concentrate

Qagh err

Qam stand

Qoch disagree

Qong sleep

Qum communicate

ratlh remain

SaQ cry

SIH bend

So' hide

tet melt

vang act

vI' accumulate

wav divide

wew glow

yIn live

yIt walk

'ov compete

STATES AND QUALITIES

batlh be honored

bel be pleased

bIr be cold

bIt be nervous

buD be lazy

chep be prosperous

chIS be white

chu' be new

Dach be absent

Daj be interesting

Dal be boring

Doq be red

Doj be impressive

Doy' be tired

Do' be lucky

ghegh be rough

ghIH be sloppy

ghung be hungry

Hab be smooth

Hem be proud

Hop	be far	**po'**	be expert	**Sey**	be excited
HoS	be strong	**puj**	be weak	**SoQ**	be closed
jej	be sharp	**puS**	be few	**Soy'**	be clumsy
jot	be calm	**qab**	be bad	**SuD**	be green
lam	be dirty	**qan**	be old	**taD**	be frozen
law'	be many	**qay'**	be a problem	**tam**	be quiet
le'	be special	**qej**	be mean	**taQ**	be weird
let	be hard	**qetlh**	be dull	**tay'**	be together
ll'	be useful	**qlj**	be black	**teH**	be true
loj	be all gone	**QaD**	be dry	**tIn**	be big
lugh	be right	**QaQ**	be good	**tuj**	be hot
mach	be small	**Qatlh**	be difficult	**tun**	be soft
maw'	be crazy	**QeH**	be angry	**tlhab**	be free
mIS	be confused	**Qav**	be last	**tlhaQ**	be funny
mob	be alone	**QIp**	be stupid	**val**	be intelligent
nap	be simple	**QIv**	be inferior	**wIb**	be sour
nIb	be identical	**Qob**	be dangerous	**wov**	be bright
nIv	be superior	**Quch**	be happy	**yap**	be sufficient
noy	be famous	**ram**	be unimportant	**yep**	be careful
ngeb	be fake	**rotlh**	be tough	**yoH**	be brave
ngeD	be easy	**rop**	be sick	**yon**	be satisfied
ngo'	be old	**ror**	be fat	**yuD**	be dishonest
ngoy'	be responsible	**ru'**	be temporary	**'eH**	be ready
pIm	be different	**Sagh**	be serious	**'IH**	be good-looking
pIv	be healthy	**SaH**	be present	**'Itlh**	be advanced
pI'	be fat	**Sar**	be various	**'IQ**	be sad
poS	be open	**Say'**	be clean	**'It**	be depressed

'oj be thirsty

'oy' be sore

'ugh be heavy

'uH be hung over

'um be qualified

'ut be necessary

TRANSITIVE VERBS

baH fire (in the sense of "fire a weapon")

chargh conquer

chav achieve

chaw' allow

cher establish

chIj navigate

chop bite

Del describe

ghaj have

ghIj scare

ghItlh write

ghom meet

ghoj learn

ghong abuse

ghoS approach

ghov recognize

HaD study

Har believe

Hev receive

HIv attack

HoH kill

jab serve

jang answer

jatlh speak

je' buy

jol beam

laD read

laH accept

legh see

lIj forget

lon abandon

lo' use

magh betray

maS prefer

much translate

muS hate

neH want

nej seek

nuQ annoy

ngas contain

ngIp borrow

ngu' identify

par dislike

pIch blame

poj analyze

pol save

poQ demand

qaw remember

qeng carry

qem bring

qIH meet

qIp hit

QaH help

Qan protect

Qoy hear

rur resemble

SeH control

SIch reach

SIj slit

SIm calculate

Sop eat

Sov know

So' hide

Such visit

Suq get

Suv fight

tem deny

teb fill

teq remove

tIch insult

tIj board

tIv enjoy

tI' fix

toj trick

toS climb

toy' serve

tung discourage

tu' find

tuQ wear

tlhap take

tlha' follow

tlhej accompany

tlhob ask

tlhutlh drink

van salute

voq trust

vut cook

vuv respect

wam hunt

wIv choose

yaj understand

ylv chew

yuv push

yu' question

'av guard

'el enter

TRANSITIVES IN **-MOH**

See the section on Type 3 verb endings for a discussion of these verbs.

chaghmoH drop

chenmoH make

choHmoH change

chungmoH accelerate

DingmoH spin

HuSmoH hang

nupmoHdecrease

ngoSmoH dissolve

SIHmoH bend

So'moH hide

tetmoH melt

vl'moH accumulate

wavmoH divide

DITRANSITIVES

bach shoot

bej watch

Dev lead

ghItlh write

ja' tell

je' buy

nob give

ngeH send

ngev sell

qem bring

VERBS WHICH TAKE SENTENCE COMPLEMENTS

chaw' allow

Hon doubt

legh see

Qub think

Sov know

tul hope

tu' find

IDIOMATIC VERBS

tu'lu' there is, there are

tu'lu'be' there isn't, there aren't

ADVERBS

batlh with honor

bong accidentally

chaq perhaps

chIch on purpose

DaH now

DaHjaj today

Do' luckily

loQ a little bit

nom fast

not never

pay' suddenly

plj often

QIt slowly

reH always

rut sometimes

tugh soon

vaj accordingly

wej not yet

POST-VERBAL ADVERBS

je also

neH just

SPATIAL ADVERBS

Dat everywhere

naDev here

pa' there

NUMBERS

wa' one

cha' two

wej three

loS four

vagh five

jav six

Soch seven

chorgh eight

Hut nine

Hoch each, all

pagh no

INTERJECTIONS

ghobe' no

Ha' let's go, come on

HIja' yes

HISlaH yes

lu', luq yes, okay I will (Note: not used for answering yes/no questions)

maj good

majQa' very good, well done

pItlh done

Qo' no, I won't, I refuse

SuH ready

Su' ready

toH well, so

wejpuH charming

'eH ready

COMPLEMENTIZERS

'e' that

HOMONYMS

The following words were removed from the lexicon because they are homonymous with other words, leading to a large number of legitimate but indistinguishable parses.

qab face (homophonous with *bad*)

Quch forehead (homophonous with *happy*)

je' feed (homophonous with *buy*)

APPENDIX B: FEATURE GRAMMAR

```
; Grammar rules

; OVS word order

RULE
S = { (LOCP) (ADV) VP (ADV-PV) (NP) } / { SC (LOCP) (ADV) VP (ADV-PV) (NP) } /
{ (LOCP) (ADV) VP (ADV-PV) (NP) SC } / COMPARATIVE

    <NP head-agr-number> = <VP agr-subj-number>
    <NP head-agr-pers> = <VP agr-subj-pers>
    <NP head-case> = unmarked
    <NP gloss-case> = nominative

    <S derogatory> = <NP derogatory>
    <S derogatory> = <VP derogatory>
    <S question> = <VP head-question>
    <S mood> = <VP mood>

    <VP head-subordinate> = n

    ; These rules are meant to ensure that verbs with unspecified
    ; subjects do not co-occur with subject NPs.

    <NP specified-subject> = y
    <NP predicative> = n
    <VP head-specifiedsubject> = <NP specified-subject>

    <LOCP verb-of-motion> = <VP verb-of-motion>

; Two types of copula sentences
; First has pronoun subject
; Second has noun subject, which must be marked with 'e'

; I'm not crazy about allowing pronouns as predicates, but I suppose
; it's permitted in English, so I won't exclude it for the moment

RULE
S = { (LOCP) (ADV) COPVP (ADV-PV) (NP) (SC) } / { SC (LOCP) (ADV) COPVP (ADV-
PV) }
    <NP head-case> = unmarked
    <NP predicative> = n
    <NP gloss-case> = nominative
    <NP head-agr-number> = <COPVP agr-subj-number>
    <NP head-agr-animate> = <COPVP agr-subj-animate>
    <NP focus> = y
    <NP pred-focus> = n
    <COPVP pred-focus> = <NP pred-focus>

    <COPVP head-specifiedsubject> = n
    <COPVP head-specifiedsubject> = <NP specified-subject>
    <S derogatory> = <COPVP derogatory>
    <S derogatory> = <NP derogatory>
    <COPVP head-subordinate> = n
    <NP head-agr-pers> = <COPVP agr-subj-pers>
```

```
    ; If an NP is the subject,we know the whole thing is third-person
    <NP head-agr-pers> = 3rd

RULE
COPVP = NP COPV
    <NP head-agr-number> = <COPV agr-subj-number>
    <COPV agr-subj-number> = <COPVP agr-subj-number>
    <COPV head-agr-animate> = <COPVP head-agr-animate>
    <COPV agr-subj-pers> = <COPVP agr-subj-pers>
    <COPV aux> = <COPVP aux>
    <COPV neg> = <COPVP neg>
    <COPV mood> = <COPVP mood>
    <COPV complementizer> = <COPVP complementizer>
    <COPV head-question> = <COPVP head-question>
    <COPV head-subordinate> = <COPVP head-subordinate>
    <COPV derogatory> = <COPVP derogatory>

    ; Predicate NP doesn't need to agree in animacy

    <NP focus> = <COPV pred-focus>
    <NP predicative> = y
    <NP head-case> = unmarked
    <NP gloss-case> = nominative

    ; Don't care for this case
    <NP pronoun> = n

    ; Do any rewrite rules still use this?

    <COPV insert-subj> = <COPVP insert-subj>

; Spillover of sentences with copula

RULE
S = INTRJ / { COPS-LOC } / { SC COPS-LOC } / { COPS-LOC SC }

RULE
COPS-LOC = { LOCP (ADV) COPV_1 (ADV-PV) } / { LOCP (ADV) COPV_2 (ADV-PV) NP }
    <NP focus> = y
    <NP pronoun> = n
    <LOCP verb-of-motion> = n

    ; This should be 3rd
    <COPV_2 agr-subj-pers> = <NP head-agr-pers>
    <COPV_2 agr-subj-number> = <NP head-agr-number>

    <COPS-LOC derogatory> = <NP derogatory>
    <COPS-LOC head-question> = <COPV_1 head-question>
    <COPS-LOC head-subordinate> = <COPV_1 head-subordinate>
    <COPS-LOC complementizer> = <COPV_1 complementizer>
    <COPS-LOC insert-subj> = n
    <COPS-LOC aux> = <COPV_1 aux>
    <COPS-LOC neg> = <COPV_1 neg>

    <COPS-LOC head-question> = <COPV_2 head-question>
    <COPS-LOC head-subordinate> = <COPV_2 head-subordinate>
```

```
    <COPS-LOC complementizer> = <COPV_2 complementizer>
    <COPS-LOC insert-subj> = n
    <COPS-LOC aux> = <COPV_2 aux>
    <COPS-LOC neg> = <COPV_2 neg>

    <COPV_1 agr-subj-number> = <COPS-LOC agr-subj-number>
    <COPV_1 agr-subj-pers> = <COPS-LOC agr-subj-pers>
    <COPV_2 agr-subj-number> = <COPS-LOC agr-subj-number>
    <COPV_2 agr-subj-pers> = <COPS-LOC agr-subj-pers>

    <COPS-LOC head-agr-animate> = <COPV_1 head-agr-animate>
    <COPS-LOC head-agr-animate> = <COPV_2 head-agr-animate>

RULE
SC = { (LOCP) (ADV) VP (ADV-PV) (NP) } / COPS-LOC
    <VP head-subordinate> = y

    <NP head-agr-number> = <VP agr-subj-number>
    <NP head-agr-pers> = <VP agr-subj-pers>
    <NP head-case> = unmarked
    <NP gloss-case> = nominative

    <SC derogatory> = <VP derogatory>
    <SC derogatory> = <NP derogatory>
    <SC derogatory> = <COPS-LOC derogatory>
    <COPS-LOC head-subordinate> = y

    <NP predicative> = n

; Verbs are inflected to agree with person and number of both subject and
object

RULE
SC = (LOCP) (ADV) COPVP (ADV-VP) (NP)
    <NP head-case> = unmarked
    <NP predicative> = n
    <NP gloss-case> = nominative
    <NP head-agr-number> = <COPVP agr-subj-number>
    <NP head-agr-animate> = <COPVP agr-subj-animate>
    <NP focus> = y
    <NP pred-focus> = n
    <COPVP pred-focus> = <NP pred-focus>

    <COPVP head-specifiedsubject> = n
    <COPVP head-specifiedsubject> = <NP specified-subject>
    <S derogatory> = <COPVP derogatory>
    <S derogatory> = <NP derogatory>
    <COPVP head-subordinate> = y
    <NP head-agr-pers> = <COPVP agr-subj-pers>

    ; If an NP is the subject, we know the whole thing is third-person
    <NP head-agr-pers> = 3rd

RULE
VP = TVP / IV / DTVP / V-SP
    <VP agr-subj-number> = <IV agr-subj-number>
    <VP agr-subj-pers> = <IV agr-subj-pers>
```

```
<VP agr-subj-number> = <TVP agr-subj-number>
<VP agr-subj-pers> = <TVP agr-subj-pers>
<VP agr-obj-number> = <TVP agr-obj-number>
<VP agr-obj-pers> = <TVP agr-obj-pers>

<VP agr-subj-number> = <DTVP agr-subj-number>
<VP agr-subj-pers> = <DTVP agr-subj-pers>
<VP agr-obj-number> = <DTVP agr-obj-number>
<VP agr-obj-pers> = <DTVP agr-obj-pers>
<VP agr-indirobj-number> = <DTVP agr-indirobj-number>
<VP agr-indirobj-pers> = <DTVP agr-indirobj-pers>

<VP head-question> = <IV head-question>
<VP head-question> = <TVP head-question>
<VP head-question> = <DTVP head-question>
<VP head-question> = <V-SP head-question>

<VP head-subordinate> = <IV head-subordinate>
<VP head-subordinate> = <TVP head-subordinate>
<VP head-subordinate> = <DTVP head-subordinate>
<VP head-subordinate> = <V-SP head-subordinate>

<VP mood> = <TVP mood>
<VP mood> = <IV mood>
<VP mood> = <DTVP mood>
<VP mood> = <V-SP mood>

<VP complementizer> = <TVP complementizer>
<VP complementizer> = <IV complementizer>
<VP complementizer> = <DTVP complementizer>
<VP complementizer> = <V-SP complementizer>

<VP head-specifiedsubject> = <TVP head-specifiedsubject>
<VP head-specifiedsubject> = <IV head-specifiedsubject>
<VP head-specifiedsubject> = <DTVP head-specifiedsubject>
<VP head-specifiedsubject> = <V-SP head-specifiedsubject>

<VP head-specifiedobject> = <TVP head-specifiedobject>

<VP raising> = <TVP raising>
<VP raising> = <IV raising>
<VP raising> = <DTVP raising>
<VP raising> = <V-SP raising>

<VP aux> = <TVP aux>
<VP aux> = <IV aux>
<VP aux> = <DTVP aux>
<VP aux> = <V-SP aux>

<VP neg> = <TVP neg>
<VP neg> = <IV neg>
<VP neg> = <DTVP neg>
<VP neg> = <V-SP neg>

<VP verb-of-motion> = <TVP verb-of-motion>
<VP verb-of-motion> = <IV verb-of-motion>
```

```
    <VP verb-of-motion> = <DTVP verb-of-motion>
    <VP verb-of-motion> = <V-SP verb-of-motion>

RULE
TVP = { NP TV_1 } / TV_2
    <NP head-agr-number> = <TV_1 agr-obj-number>
    <NP head-agr-pers> = <TV_1 agr-obj-pers>
    <NP head-case> = unmarked
    <NP gloss-case> = accusative
    <NP predicative> = <TV_1 predicative>

    <TVP derogatory> = <NP derogatory>

    <TVP head-subordinate> = <TV_1 head-subordinate>
    <TVP head-question> = <TV_1 head-question>
    <TVP agr-subj-pers> = <TV_1 agr-subj-pers>
    <TVP agr-subj-number> = <TV_1 agr-subj-number>
    <TVP agr-obj-pers> = <TV_1 agr-obj-pers>
    <TVP agr-obj-number> = <TV_1 agr-obj-number>

    <TVP head-subordinate> = <TV_2 head-subordinate>
    <TVP head-question> = <TV_2 head-question>
    <TVP agr-subj-pers> = <TV_2 agr-subj-pers>
    <TVP agr-subj-number> = <TV_2 agr-subj-number>
    <TVP agr-obj-pers> = <TV_2 agr-obj-pers>
    <TVP agr-obj-number> = <TV_2 agr-obj-number>

    <TVP complementizer> = <TV_1 complementizer>
    <TVP mood> = <TV_1 mood>
    <TVP head-specifiedsubject> = <TV_1 head-specifiedsubject>
    <TVP raising> = <TV_1 raising>
    <TVP aux> = <TV_1 aux>
    <TVP neg> = <TV_1 neg>
    <TVP verb-of-motion> = <TV_1 verb-of-motion>

    <TVP complementizer> = <TV_2 complementizer>
    <TVP mood> = <TV_2 mood>
    <TVP head-specifiedsubject> = <TV_2 head-specifiedsubject>
    <TVP aux> = <TV_2 aux>
    <TVP neg> = <TV_2 neg>
    <TVP raising> = <TV_2 raising>
    <TVP verb-of-motion> = <TV_2 verb-of-motion>

    ; This is an ugly but effective way of giving TVP a feature
    ; that lets us check whether it co-occurred with an object.
    ; This feature percolates upwards so that raising verbs know
    ; whether there's an object NP to raise to subject or not.

    <TV_1 head-specifiedobject> = y
    <TV_2 head-specifiedobject> = n
    <TVP head-specifiedobject> = <TV_1 head-specifiedobject>
    <TVP head-specifiedobject> = <TV_2 head-specifiedobject>

; Ditransitives have two syntactic variants:
; INDIR-OBJ DIR-OBJ VB, in which the verb's object prefix agrees with the
direct object
```

```
; DIR-OBJ VB, in which the verb's object prefix agrees with the indirect
object

; Splitting the two here is intended to increase readability later

RULE
DTVP = DTVP-LONGFORM / DTVP-SHORTFORM
    <DTVP agr-subj-pers> = <DTVP-LONGFORM agr-subj-pers>
    <DTVP agr-subj-number> = <DTVP-LONGFORM agr-subj-number>
    <DTVP agr-obj-pers> = <DTVP-LONGFORM agr-obj-pers>
    <DTVP agr-obj-number> = <DTVP-LONGFORM agr-obj-pers>
    <DTVP agr-indirobj-number> = <DTVP-LONGFORM agr-indirobj-number>
    <DTVP agr-indirobj-pers> = <DTVP-LONGFORM agr-indirobj-pers>

    <DTVP agr-subj-number> = <DTVP-SHORTFORM agr-subj-number>
    <DTVP agr-subj-pers> = <DTVP-SHORTFORM agr-subj-pers>
    <DTVP agr-obj-number> = <DTVP-SHORTFORM agr-obj-number>
    <DTVP agr-obj-pers> = <DTVP-SHORTFORM agr-obj-pers>
    <DTVP agr-indirobj-number> = <DTVP-SHORTFORM agr-indirobj-number>
    <DTVP agr-indirobj-pers> = <DTVP-SHORTFORM agr-indirobj-pers>

    <DTVP derogatory> = <DTVP-LONGFORM derogatory>
    <DTVP head-question> = <DTVP-LONGFORM head-question>
    <DTVP head-subordinate> = <DTVP-LONGFORM head-subordinate>

    <DTVP derogatory> = <DTVP-SHORTFORM derogatory>
    <DTVP head-question> = <DTVP-SHORTFORM head-question>
    <DTVP head-subordinate> = <DTVP-SHORTFORM head-subordinate>

    <DTVP-LONGFORM complementizer> = <DTVP complementizer>
    <DTVP-SHORTFORM complementizer> = <DTVP complementizer>

    <DTVP-LONGFORM mood> = <DTVP mood>
    <DTVP-SHORTFORM mood> = <DTVP mood>

    <DTVP-LONGFORM head-specifiedsubject> = <DTVP head-specifiedsubject>
    <DTVP-SHORTFORM head-specifiedsubject> = <DTVP head-specifiedsubject>

    <DTVP-LONGFORM raising> = <DTVP raising>
    <DTVP-SHORTFORM raising> = <DTVP raising>

    <DTVP-LONGFORM aux> = <DTVP aux>
    <DTVP-SHORTFORM aux> = <DTVP aux>

    <DTVP-LONGFORM neg> = <DTVP neg>
    <DTVP-SHORTFORM neg> = <DTVP neg>

    <DTVP-LONGFORM verb-of-motion> = <DTVP verb-of-motion>
    <DTVP-SHORTFORM verb-of-motion> = <DTVP verb-of-motion>

RULE
DTVP-LONGFORM = NP_1 (NP_2) DTV
    <DTVP-LONGFORM agr-subj-number> = <DTV agr-subj-number>
    <DTVP-LONGFORM agr-subj-pers> = <DTV agr-subj-pers>
    <DTVP-LONGFORM agr-obj-number> = <DTV agr-obj-number>
    <DTVP-LONGFORM agr-subj-pers> = <DTV agr-subj-pers>
    <DTVP-LONGFORM agr-indirobj-number> = <DTV agr-indirobj-number>
```

```
    <DTVP-LONGFORM agr-indirobj-pers> = <DTV agr-indirobj-pers>

    <DTV derogatory> = <DTVP-LONGFORM derogatory>
    <DTV head-question> = <DTVP-LONGFORM head-question>
    <DTV head-subordinate> = <DTVP-LONGFORM head-subordinate>
    <DTV agr-indirobj-number> = <NP_1 head-agr-number>
    <DTV agr-indirobj-pers> = <NP_1 head-agr-pers>

    <NP_1 head-case> = dative

    ; Note: it may not ultimately be necessary to set gloss-case
    ; where it's not ambiguous.  For now, adding it to prevent
    ; non-existant features from being checked during debugging.

    <NP_1 gloss-case> = dative
    <DTV agr-obj-number> = <NP_2 head-agr-number>
    <DTV agr-obj-pers> = <NP_2 head-agr-pers>
    <NP_2 head-case> = unmarked
    <NP_2 gloss-case> = accusative
    <NP_1 predicative> = n
    <NP_2 predicative> = n

    <DTVP-LONGFORM complementizer> = <DTV complementizer>
    <DTVP-LONGFORM mood> = <DTV mood>
    <DTVP-LONGFORM head-specifiedsubject> = <DTV head-specifiedsubject>
    <DTVP-LONGFORM raising> = <DTV raising>
    <DTVP-LONGFORM aux> = <DTV aux>
    <DTVP-LONGFORM neg> = <DTV neg>
    <DTVP-LONGFORM verb-of-motion> = <DTV verb-of-motion>

; For now we're just handling the complementizer 'e', so we don't have to
restrict
; the subject agreement on V-S the way we would for net (which can only be
used with
; third-person subject agreement).

; head-specifiedsubject will also help us with net.

; Note that V-S can never take an aspect marker - not important right now,
because
; aspect isn't currently implemented, but may need to be done later

RULE
V-SP = S COMP V-S
    <S mood> = indicative

    <V-SP complementizer> = <V-S complementizer>
    <V-SP mood> = <V-S mood>
    <V-SP head-specifiedsubject> = <V-S head-specifiedsubject>
    <V-SP raising> = <V-S raising>
    <V-SP aux> = <V-S aux>
    <V-SP neg> = <V-S neg>
    <V-SP verb-of-motion> = <V-S verb-of-motion>

RULE
DTVP-SHORTFORM =  NP DTV
    <DTVP-SHORTFORM agr-subj-number> = <DTV agr-subj-number>
```

```
    <DTVP-SHORTFORM agr-subj-pers> = <DTV agr-subj-pers>

    ; Note that in the short form, the DTVP's indirect object
    ; agreement is realized on the verb prefix - that is, <DTV agr obj>

    ; This also kept us from just copying over everything in agr
    ; between the two.  Or did, back when agr was a nested data
    ; structure.

    <DTVP-SHORTFORM agr-indirobj-pers> = <DTV agr-obj-pers>
    <DTVP-SHORTFORM agr-indirobj-number> = <DTV agr-obj-number>

    <DTV derogatory> = <DTVP-SHORTFORM derogatory>
    <DTV head-question> = <DTVP-SHORTFORM head-question>
    <DTV head-subordinate> = <DTVP-SHORTFORM head-subordinate>
    <DTV derogatory> = <NP derogatory>

    <NP head-case> = unmarked
    <NP gloss-case> = accusative
    <NP head-agr-number> = <DTVP-SHORTFORM agr-obj-number>
    <NP head-agr-pers> = <DTVP-SHORTFORM agr-obj-pers>

    <NP predicative> = n

    <DTV complementizer> = <DTVP-SHORTFORM complementizer>
    <DTV mood> = <DTVP-SHORTFORM mood>
    <DTV head-specifiedsubject> = <DTVP-SHORTFORM head-specifiedsubject>
    <DTV raising> = <DTVP-SHORTFORM raising>
    <DTV aux> = <DTVP-SHORTFORM aux>
    <DTV neg> = <DTVP-SHORTFORM neg>
    <DTV verb-of-motion> = <DTVP-SHORTFORM verb-of-motion>

RULE
NP = { (NUM_1) NBAR_1 } / { (NUM_2) NBAR_2 ADJ } / COMP-NP / PR
    <NP pronoun> = <PR pronoun>

    ; Pronoun features are straightforward enough
    <NP head-agr-number> = <PR head-agr-number>
    <NP head-agr-pers> = <PR head-agr-pers>
    <NP head-case> = <PR head-case>
    ; This is where gloss-case really becomes important
    <NP gloss-case> = <PR gloss-case>
    <NP head-agr-animate> = <PR head-agr-animate>

    <NP head-case> = <COMP-NP head-case>
    <NP gloss-case> = <COMP-NP gloss-case>
    <NP head-agr-number> = <COMP-NP head-agr-number>
    <NP head-agr-pers> = <COMP-NP head-agr-pers>
    <NP derogatory> = <COMP-NP derogatory>
    <NP head-agr-animate> = <COMP-NP head-agr-animate>

    <NP head-agr-number> = <NBAR_1 head-agr-number>
    <NP head-agr-pers> = <NBAR_1 head-agr-pers>
    <NP head-case> = <NBAR_1 head-case>
    <NP gloss-case> = <NBAR_1 gloss-case>
    <NP derogatory> = <NBAR_1 derogatory>
    <NUM_1 head-agr-number> = <NBAR_1 head-agr-number>
```

```
    <NP focus> = <NBAR_1 focus>
    <NP head-agr-animate> = <NBAR_1 head-agr-animate>

    ; In an NP with an attributive adjective, the dative case
    ; ending is attached to the adjective, not the noun.
    ; The simplest way to handle this is to place case agreement
    ; on the adjective.  The noun is specified as nominative
    ; for convenience's sake - it doesn't take a case ending,
    ; but needs to be specified for case to prevent unification
    ; with a noun having a dative ending.

    ; Adjectives never need gloss-case.

    <NUM_2 head-agr-number> = <NBAR_2 head-agr-number>
    <NBAR_2 head-case> = unmarked
    <NBAR_2 gloss-case> = nominative
    <NP head-case> = <ADJ head-case>
    <NP head-agr-number> = <NBAR_2 head-agr-number>
    <NP head-agr-pers> = <NBAR_2 head-agr-pers>
    <NP derogatory> = <NBAR_2 derogatory>
    <NUM_2 head-agr-animate> = <NBAR_2 head-agr-animate>

    ; Any Type 5 noun endings must go on the adjective, not the noun.
    <NP focus> = <ADJ focus>
    <NBAR_2 focus> = n

    ; Keep numbers from unifying with non-count nouns
    <NUM non-count> = <NBAR non-count>
    <NUM non-count> = n

    <NBAR_1 predicative> = <NP predicative>
    <NBAR_2 predicative> = <NP predicative>
    <COMP-NP predicative> = <NP predicative>

    <NP non-count> = <NBAR_1 non-count>
    <NP non-count> = <NBAR_2 non-count>

    <NP prep> = <NBAR_1 prep>
    <NP prep> = <NBAR_2 prep>

RULE
NBAR = N_1 / { N_2 N_3 }

    ; This rule permits noun-noun compounds.  In these compounds,
    ; the first noun is always in genitive case.  (Since Klingon
    ; only has overt case marking in the dative, this is mostly
    ; a matter of choosing the right English gloss.)

    ; In noun-noun constructions, Type 5 suffixes - notably dative
    ; case (-vaD) and focus (-'e') - can only occur on the secon
    ; element.

    <NBAR head-agr-number> = <N_1 head-agr-number>
    <NBAR head-agr-number> = <N_3 head-agr-number>

    ; This has no excuse for not being 3rd
    <NBAR head-agr-pers> = <N_1 head-agr-pers>
```

```
    <NBAR head-agr-pers> = <N_3 head-agr-pers>

    <NBAR head-agr-animate> = <N_1 head-agr-animate>
    <NBAR head-agr-animate> = <N_3 head-agr-animate>

    <NBAR head-case> = <N_1 head-case>
    <NBAR head-case> = <N_3 head-case>
    <NBAR gloss-case> = <N_1 gloss-case>
    <NBAR gloss-case> = <N_3 gloss-case>
    <N_2 head-case> = unmarked
    <N_2 gloss-case> = genitive

    <NBAR derogatory> = <N_1 derogatory>
    <NBAR derogatory> = <N_2 derogatory>
    <NBAR derogatory> = <N_3 derogatory>
    <NBAR focus> = <N_1 focus>
    <NBAR focus> = <N_3 focus>
    <N_2 focus> = n

    <NBAR non-count> = <N_1 non-count>
    <NBAR non-count> = <N_3 non-count>

    <NBAR definite> = <N_1 definite>
    <NBAR definite> = <N_3 definite>

    <NBAR determiner> = <N_1 determiner>
    <NBAR determiner> = <N_3 determiner>

    <NBAR prep> = <N_1 prep>
    <NBAR prep> = <N_3 prep>

RULE
COMPARATIVE = NP_1 ADJ_1 COMP-C_1 NP_2 ADJ_2 COMP-C_2
    <COMP-C_1 lex> = law'
    <COMP-C_2 lex> = puS
    <ADJ_1 lex> = <ADJ_2 lex>
    <NP_1 head-case> = unmarked
    <NP_1 gloss-case> = nominative
    <NP_2 head-case> = unmarked
    <NP_2 gloss-case> = nominative

RULE
IV = V
    <IV agr-subj-pers> = <V agr-subj-pers>
    <IV agr-subj-number> = <V agr-subj-number>

    <V arity> = 1

    <IV head-question> = <V head-question>
    <IV head-subordinate> = <V head-subordinate>

    <IV complementizer> = <V complementizer>
    <IV mood> = <V mood>
    <IV head-specifiedsubject> = <V head-specifiedsubject>
    <IV raising> = <V raising>
    <IV aux> = <V aux>
    <IV neg> = <V neg>
```

```
    <IV verb-of-motion> = <V verb-of-motion>

RULE
TV = V
    <TV agr-subj-pers> = <V agr-subj-pers>
    <TV agr-obj-pers> = <V agr-obj-pers>
    <TV agr-subj-number> = <V agr-subj-number>
    <TV agr-obj-number> = <V agr-obj-number>

    <V arity> = 2
    <TV head-question> = <V head-question>
    <TV head-subordinate> = <V head-subordinate>

    <TV complementizer> = <V complementizer>
    <TV mood> = <V mood>
    <TV raising> = <V raising>
    <TV aux> = <V aux>
    <TV neg> = <V neg>
    <TV head-specifiedsubject> = <V head-specifiedsubject>
    <TV predicative> = <V predicative>
    <TV verb-of-motion> = <V verb-of-motion>
    <TV gloss-1p> = <V gloss-1p>

RULE
DTV = V
    <V arity> = 3
    <DTV agr-subj-pers> = <V agr-subj-pers>
    <DTV agr-obj-pers> = <V agr-obj-pers>
    <DTV agr-subj-number> = <V agr-subj-number>
    <DTV agr-obj-number> = <V agr-obj-number>
    <DTV head-question> = <V head-question>
    <DTV head-subordinate> = <V head-subordinate>

    <DTV complementizer> = <V complementizer>
    <DTV mood> = <V mood>
    <DTV raising> = <V raising>
    <DTV aux> = <V aux>
    <DTV neg> = <V neg>
    <DTV head-specifiedsubject> = <V head-specifiedsubject>

    ; I'm not convinced this is possible
    <DTV verb-of-motion> = <V verb-of-motion>

RULE
V-S = V
    <V agr-subj-pers> = <V-S agr-subj-pers>
    <V agr-subj-number> = <V-S agr-subj-number>

    ; We probably don't need these, since the V-S lexical rules
    ; exclusively generate third-person singular object prefixes.

    <V agr-obj-pers> = <V-S agr-obj-pers>
    <V agr-obj-number> = <V-S agr-obj-number>

    <V head-question> = <V-S head-question>
    <V head-subordinate> = <V-S head-subordinate>
    <V arity> = S
```

```
    <V-S complementizer> = <V complementizer>
    <V-S mood> = <V mood>
    <V-S head-specifiedsubject> = <V head-specifiedsubject>
    <V-S raising> = <V raising>
    <V-S aux> = <V aux>
    <V-S neg> = <V neg>

    ; can these ever be verbs of motion?

    <V-S verb-of-motion> = <V verb-of-motion>

RULE
COMP-NP = NP_1 NP_2 CONJ-N
    ; Right now this is underspecified for animacy, which could
    ; be solved by adding more arg-agr features for CONJ-N

    <NP_1 head-agr-pers> = <CONJ-N first-arg-agr-pers>
    <NP_2 head-agr-pers> = <CONJ-N second-arg-agr-pers>
    <COMP-NP head-agr-number> = <CONJ-N head-agr-number>
    <COMP-NP head-agr-pers> = <CONJ-N head-agr-pers>
    <COMP-NP head-case> = <CONJ-N head-case>
    <CONJ-N head-case> = <NP_1 head-case>
    <CONJ-N head-case> = <NP_2 head-case>
    <COMP-NP gloss-case> = <CONJ-N gloss-case>
    <CONJ-N gloss-case> = <NP_1 gloss-case>
    <CONJ-N gloss-case> = <NP_2 gloss-case>

    <CONJ-N derogatory> = <NP_1 derogatory>
    <CONJ-N derogatory> = <NP_2 derogatory>

    <NP_1 predicative> = <COMP-NP predicative>
    <NP_2 predicative> = <COMP-NP predicative>

RULE
LOCP = ADV-LOC / NP_1 / { NP_2 POSTP }
    <NP_1 head-case> = locative
    <NP_1 gloss-case> = locative
    <NP_1 predicative> = n

    <NP_2 head-case> = unmarked
    <NP_2 gloss-case> = accusative
    <NP_2 predicative> = n

    ; Make sure "between" takes a plural NP
    <NP_2 head-agr-number> = <POSTP head-agr-number>

    ; We need to know at higher levels if an adverb or a noun
    ; filled this slot, because only nouns take prepositions

    ; ADV-LOC has a prep value to prevent error messages from
    ; unifying a nonexistent value.  All preposition-adding
    ; rules will test the value of prep, so it should be okay
    ; to just set ADV-LOC's to n.

    <LOCP prep> = <NP_1 prep>
    <LOCP prep> = <NP_2 prep>
```

```
    <LOCP prep> = <ADV-LOC prep>
    <LOCP wh-elem> = <ADV-LOC wh-elem>

    ; Awful and unintuitive as this might seen, NP is the only
    ; thing on its side of the rewrite rule, so if we want to
    ; check this feature, we have to pass it along to the NP

    <NP_1 spatial-movement> = <LOCP verb-of-motion>
    <NP_2 spatial-movement> = <LOCP verb-of-motion>

Let N be <head-agr-number> = !sing
         <definite> = !n
         <head-agr-animate> = !n
         <focus> = !n
         <non-count> = !n
         <head-agr-pers> = 3rd
         <pronoun> = n
         <prep> = !at
         <case> = !unmarked
         ; No default for gloss-case as it is always set positionally

Let V be <head-subordinate> = !n
         <raising> = !n
         <head-specifiedsubject> = !y
         ; Will this be a problem when we add wh-questions?
         <head-question> = !n
         <aux> = !do
         <predicative> = !n
         <neg> = !n
         <verb-of-motion> = !n
         <gloss-1p> = !n

Let COPV be <head-subordinate> = !n
            <neg> = !n
            <head-question> = !n

Let PR be <focus> = !n

; Do we need these defaults?
Let ADV be <wh-elem> = !n

Let ADV-LOC be <prep> = n
               <wh-elem> = !n

; To keep them from unifying with V-SP
Let INTRJ be <mood> = n

; Not getting good results with this
;Let nom/acc be {[case:nominative]
                ;[case:accusative]}
```

APPENDIX C: THE POSTPROCESSOR CODE

```
#!/usr/bin/perl

my $filename = shift;

open (FILE, $filename) or die "Can't open input file: $!";

my (%global_replacements, %gloss_replacements, %restricted_replacements);

while (<DATA>) {
      chomp;
      my ($restriction, $pattern, $replacement) = (split(','));
      if ($restriction =~ /GLOBAL/) {
            $global_replacements{$pattern} = $replacement;
        } elsif ($restriction =~ /GLOSS/) {
            $gloss_replacements{$pattern} = $replacement;
      } else {
            $restricted_replacements{$restriction}{$pattern} = $replacement;
      }
}

my $counter;

FILE:while (<FILE>) {
    foreach $global (keys %global_replacements) {
        s/$global/$global_replacements{$global}/g;
    }

    if (/^\\g/) {
      $counter++;

        if (/[^aeou]y[ds]$/) {
            s/([^aeou])y([ds])$/$1ie$2/;
        }

      GLOSS:foreach $gloss (keys %gloss_replacements) {
            s/$gloss/$gloss_replacements{$gloss}/g;
        }

      PATTERN:foreach $pattern (keys %restricted_replacements) {
            @irregulars = keys %{$restricted_replacements{$pattern}};

          if (/$pattern/) {
              s/$pattern//;

            foreach $irregular (@irregulars) {
                  # Doubles match within word
                s/\b$irregular$/$restricted_replacements{$pattern}-
>{$irregular}/;
            }
            last PATTERN;
            }
      }
```

```
        if (/change-copula-/) {
            s/change-copula-//;
            s/(am|are|is) can't/can't be/;
            s/(am|are|is) can/can be/;
        }

      if (/double-passive-/) {
          s/double-passive-//;
            s/(.)ed$/$1$1ed/;
        }
    }
    print $_;
}

warn "Wrote $counter words\n";

__DATA__
GLOBAL,%,
GLOSS, haves$, has
GLOSS, ieing$, ying
irreg-plur-,Bird of Preys?,Birds of Prey
irreg-plur-,childs?,children
irreg-plur-,foots?,feet
irreg-plur-,helmsman?,helmsmen
irreg-plur-,lifes?,lives
irreg-plur-,knifes?,knives
irreg-plur-,mans?,men
irreg-plur-,persons?,people
irreg-plur-,tooths?,teeth
irreg-plur-,wifes?,wives
irreg-plur-,womans?,women
irreg-passive-,bended,bent
irreg-passive-,biteed,bitten
irreg-passive-,breaked,broken
irreg-passive-,bringed,brought
irreg-passive-,buyed,bought
irreg-passive-,chooseed,chosen
irreg-passive-,cuted,cut
irreg-passive-,drinked,drunk
irreg-passive-,eated,eaten
irreg-passive-,feeded,fed
irreg-passive-,fighted,fought
irreg-passive-,finded,found
irreg-passive-,forgeted,forgotten
irreg-passive-,geted,got
irreg-passive-,giveed,given
irreg-passive-,hanged,hung
irreg-passive-,haveed,had
irreg-passive-,heared,heard
irreg-passive-,hideed,hid
irreg-passive-,hited,hit
irreg-passive-,hurted,hurt
irreg-passive-,knowed,known
irreg-passive-,leaded,led
irreg-passive-,makeed,made
irreg-passive-,meeted,met
irreg-passive-,readed,read
```

```
irreg-passive-,seeed,seen
irreg-passive-,seeked,sought
irreg-passive-,selled,sold
irreg-passive-,sended,sent
irreg-passive-,shooted,shot
irreg-passive-,slited,slit
irreg-passive-,speaked,spoken
irreg-passive-,spined,spun
irreg-passive-,takeed,taken
irreg-passive-,thinked,thought
irreg-passive-,telled,told
irreg-passive-,understanded,understood
irreg-passive-,weared,worn
irreg-passive-,writeed,written
```

APPENDIX D: AN SFST PARSER FOR KLINGON MORPHOLOGY

```
#include "klingon-symbols-include.fst"

% Type 1 noun suffixes
$nouns-type-one$ = <N>:<> (<aug-dim=aug>:{'a'} | <aug-dim=dim>:{Hom} |\
     <aug-dim=affectionate>:{oy})?

% Type 2 noun suffixes
$nouns-unmarked-type-two$ = (<num=pl>:{mey})?
$nouns-body-type-two$ = (<num=pl>:{Du'})?
$nouns-lang-type-two$ = ({<num=pl><scattered=y>}:{mey} | <num=pl>:{pu'})?

% Type 3 noun suffixes
$nouns-type-three$ = (<qual=so-called>:{qoq} | <qual=apparent>:{Hey} |\
     <qual=definite>:{na'})?

% Type 4 noun suffixes
$nouns-type-four-non-lang$ = ({<poss-pers=1st><poss-num=sg>}:{wIj} |\
                              {<poss-pers=2nd><poss-num=sg>}:{lIj} |\
                              {<poss-pers=1st><poss-num=pl>}:{maj} |\
                              {<poss-pers=2nd><poss-num=pl>}:{raj} )

$nouns-type-four-lang$ = ({<poss-pers=1st><poss-num=sg>}:{wI'} |\
                          {<poss-pers=2nd><poss-num=sg>}:{lI'} |\
                          {<poss-pers=1st><poss-num=pl>}:{ma'} |\
                          {<poss-pers=2nd><poss-num=pl>}:{ra'} )

$nouns-type-four-lang-derog$ = ({<poss-pers=1st><poss-num=sg><derog=y>}:{wIj}
|\
                              {<poss-pers=2nd><poss-num=sg><derog=y>}:{lIj} |\
                              {<poss-pers=1st><poss-num=pl><derog=y>}:{maj} |\
                              {<poss-pers=2nd><poss-num=pl><derog=y>}:{raj} )

$nouns-type-four-common$ = ({<poss-pers=3rd><poss-num=sg>}:{Daj} |\
                            {<poss-pers=3rd><poss-num=pl>}:{chaj} |\
                            {<deictic=this>}:{vam} |\
                            {<deictic=that>}:{vetlh})

$nouns-type-four-non-lang-all$ = ($nouns-type-four-non-lang$ |\
     $nouns-type-four-common$)?
$nouns-type-four-lang-all$ = ($nouns-type-four-lang$ |\
     $nouns-type-four-lang-derog$ | $nouns-type-four-common$)?

% Type 5 noun suffixes

$nouns-type-five$ = (<case=loc>:{Daq} | <case=vo'>:{vo'} | <case=mo'>:{mo'} |\
     <case=dat>:{vaD} | <topic=y>:{'e'})?

% Verb prefixes

% To do: re-order so these come after verb root

$prefixes-object$ = ({<subj-pers=1st><subj-num=sing><obj-pers=2nd><obj-
num=sing>}:{qa} |\
```

```
                    {<subj-pers=1st><subj-num=sing><obj-pers=3rd><obj-
num=sing>}:{vI} |\
                    {<subj-pers=1st><subj-num=sing><obj-pers=3rd><obj-
num=pl>}:{vI} |\
                    {<subj-pers=1st><subj-num=sing><obj-pers=2nd><obj-
num=pl>}:{Sa} |\
                    {<subj-pers=2nd><subj-num=sing><obj-pers=1st><obj-
num=sing>}:{cho} |\
                    {<subj-pers=2nd><subj-num=sing><obj-pers=3rd><obj-
num=sing>}:{Da} |\
                    {<subj-pers=2nd><subj-num=sing><obj-pers=1st><obj-
num=pl>}:{ju} |\
                    {<subj-pers=2nd><subj-num=sing><obj-pers=3rd><obj-
num=pl>}:{Da} |\
                    {<subj-pers=3rd><subj-num=sing><obj-pers=1st><obj-
num=sing>}:{mu} |\
                    {<subj-pers=3rd><subj-num=sing><obj-pers=2nd><obj-
num=sing>}:{Du} |\
                    {<subj-pers=3rd><subj-num=sing><obj-pers=3rd><obj-
num=sing>}:<> |\
                    {<subj-pers=3rd><subj-num=sing><obj-pers=1st><obj-
num=pl>}:{nu} |\
                    {<subj-pers=3rd><subj-num=sing><obj-pers=2nd><obj-
num=pl>}:{lI} |\
                    {<subj-pers=3rd><subj-num=sing><obj-pers=3rd><obj-
num=pl>}:<> |\
                    {<subj-pers=1st><subj-num=pl><obj-pers=2nd><obj-
num=sing>}:{pI} |\
                    {<subj-pers=1st><subj-num=pl><obj-pers=3rd><obj-
num=sing>}:{wI} |\
                    {<subj-pers=1st><subj-num=pl><obj-pers=2nd><obj-num=pl>}:
{re} |\
                    {<subj-pers=1st><subj-num=pl><obj-pers=3rd><obj-num=pl>}:
{DI} |\
                    {<subj-pers=2nd><subj-num=pl><obj-pers=1st><obj-
num=sing>}:{tu} |\
                    {<subj-pers=2nd><subj-num=pl><obj-pers=3rd><obj-
num=sing>}:{bo} |\
                    {<subj-pers=2nd><subj-num=pl><obj-pers=1st><obj-num=pl>}:
{che} |\
                    {<subj-pers=2nd><subj-num=pl><obj-pers=3rd><obj-num=pl>}:
{bo} |\
                    {<subj-pers=3rd><subj-num=pl><obj-pers=1st><obj-
num=sing>}:{mu} |\
                    {<subj-pers=3rd><subj-num=pl><obj-pers=2nd><obj-
num=sing>}:{nI} |\
                    {<subj-pers=3rd><subj-num=pl><obj-pers=3rd><obj-
num=sing>}:{lu} |\
                    {<subj-pers=3rd><subj-num=pl><obj-pers=1st><obj-num=pl>}:
{nu} |\
                    {<subj-pers=3rd><subj-num=pl><obj-pers=2nd><obj-num=pl>}:
{lI} |\
                    {<subj-pers=3rd><subj-num=pl><obj-pers=3rd><obj-
num=pl>}:<> \
)
```

```
$prefixes-object-imperative$ = ({<subj-pers=2nd><obj-pers=1st><obj-
num=sing><mood=imperative>}:{HI} |\
                      {<subj-pers=2nd><obj-pers=3rd><obj-
num=sing><mood=imperative>}:{yI} |\
                      {<subj-pers=2nd><obj-pers=1st><obj-
num=pl><mood=imperative>}:{gho} |\
                      {<subj-pers=2nd><obj-pers=3rd><obj-
num=sing><mood=imperative>}:{tI} \
)

$prefixes-indef-subj$ = ({<subj-pers=UNSPEC><subj-num=UNSPEC><obj-
pers=1st><obj-num=sing>}:{vI} |\
                      {<subj-pers=UNSPEC><subj-num=UNSPEC><obj-pers=2nd><obj-
num=sing>}:{Da} |\
                      {<subj-pers=UNSPEC><subj-num=UNSPEC><obj-pers=1st><obj-
num=pl>}:{wI} |\
                      {<subj-pers=UNSPEC><subj-num=UNSPEC><obj-pers=2nd><obj-
num=pl>}:{bo} |\
                      {<subj-pers=UNSPEC><subj-num=UNSPEC><obj-pers=3rd><obj-
num=pl>}:{lu} |\
                      {<subj-pers=UNSPEC><subj-num=UNSPEC><obj-pers=3rd><obj-
num=sing>}:<>)

% Need to group these into prefixes-trans and prefixes-intrans

% Not specifying object is NONE - we want to be able to re-use this with Type
1
% suffixes
$prefixes-no-object$ = ({<subj-pers=1st><subj-num=sing>}:{jI} |\
                        {<subj-pers=2nd><subj-num=sing>}:{bI} |\
                        {<subj-pers=3rd><subj-num=sing>}:<> |\
                        {<subj-pers=1st><subj-num=pl>}:{ma} |\
                        {<subj-pers=2nd><subj-num=pl>}:{Su} |\
                        {<subj-pers=3rd><subj-num=pl>}:<>)

$prefixes-no-object-imperative$ = ({<subj-pers=2nd><subj-
num=sing><mood=imperative>}:{yI} |\
                        {<subj-pers=2nd><subj-num=pl><mood=imperative>}:{pe})

$prefixes-no-object-plural$ = ({<subj-pers=1st><subj-num=pl>}:{ma} |\
                               {<subj-pers=2nd><subj-num=pl>}:{Su} |\
                               {<subj-pers=3rd><subj-num=pl>}:<>)

$prefixes-no-object-imperative-plural$ = ({<subj-pers=2nd><subj-
num=pl><mood=imperative>}:{pe})

% Type 1 suffixes
% Need to be separate because they can't occur with the same prefixes

$suffixes-type-one-egh$ = ({<obj=refl>}:{'egh})
$suffixes-type-one-chuq$ = ({<obj=recip>}:{chuq})

% Type 2 suffixes

$suffixes-type-two$ = (<vol=need>:{nIS} |\
                       <vol=will>:{qang} |\
                       {<vol=ready><animate=y>}:{rup} |\
```

```
                        {<vol=ready><animate=n>}:{beH} |\
                        <vol=afraid>:{vIp})?

% Type 3 verb suffixes

$suffixes-type-three$ = (<change=incip>:{choH} |\
                         <change=resum>:{qa'})?

%  Type 4 verb suffixes

$suffixes-type-four$ = (<cause=y>:{moH})?
$suffixes-type-four-moh$ = (<cause=y>:{moH})

% Type 5 suffixes

$suffixes-type-five-lah$ = (<ability=y>:{laH})?
$suffixes-type-five-lu$ = (<specified-subj=n>:{lu'})

% Type 6 suffixes

$suffixes-type-six$ = (<qual=perfectly>:{chu'} |\
                       <qual=certainly>:{bej} |\
                       <qual=seemingly>:{law'} |\
                       <qual=obviously>:{ba'})?

% Type 7 suffixes
% Note: not optional

$suffixes-type-seven$ = (<aspect=unmarked>:<> |\
                         <aspect=perf>:{pu'} |\
                         {<aspect=perf><intent=y>}:{ta'} |\
                         <aspect=imperf>:{taH} |\
                         {<aspect=imperf><intent=y>}:{lI'})
$suffixes-type-seven-tah$ = (<aspect=imperf>:{taH})

% Type 8 suffixes

$suffixes-type-eight$ = (<honorific=y>:{neS})?

% Type 9 suffixes

$suffixes-type-nine-no-vis$ = (<conj=as-soon-as>:{DI'} |\
                               <conj=if>:{chugh} |\
                               <conj=before>:{pa'} |\
                               <conj=for>:{meH} |\
                               <mood=question>:{'a'} |\
                               <relative=y>:{bogh})?

$suffixes-type-nine-vis$ = <conj=while>:{vIS}

% Rovers

$rovers-base$ = (<base-neg=y>:{be'} |\
                 <base-intens=y>:{qu'} |\
                 {<base-neg=y><base-intens-neg=y>}:{be'qu'} |\
                 <base-neg-intens=y>:{qu'be'})?
$rovers-type-one$ = (<one-neg=y>:{be'} |\
```

```
                    <one-intens=y>:{qu'} |\
                    {<one-neg=y><base-intens-neg=y>}:{be'qu'} |\
                    <one-neg-intens=y>:{qu'be'})?
$rovers-type-two$ = (<two-neg=y>:{be'} |\
                    <two-intens=y>:{qu'} |\
                    {<two-neg=y><base-intens-neg=y>}:{be'qu'} |\
                    <two-neg-intens=y>:{qu'be'})?
$rovers-type-three$ = (<three-neg=y>:{be'} |\
                      <three-intens=y>:{qu'} |\
                      {<three-neg=y><base-intens-neg=y>}:{be'qu'} |\
                      <three-neg-intens=y>:{qu'be'})?
$rovers-type-four$ = (<four-neg=y>:{be'} |\
                     <four-intens=y>:{qu'} |\
                     {<four-neg=y><base-intens-neg=y>}:{be'qu'} |\
                     <four-neg-intens=y>:{qu'be'})?
$rovers-type-five$ = (<five-neg=y>:{be'} |\
                     <five-intens=y>:{qu'} |\
                     {<five-neg=y><base-intens-neg=y>}:{be'qu'} |\
                     <five-neg-intens=y>:{qu'be'})?
$rovers-type-six$ = (<six-neg=y>:{be'} |\
                    <six-intens=y>:{qu'} |\
                    {<six-neg=y><base-intens-neg=y>}:{be'qu'} |\
                    <six-neg-intens=y>:{qu'be'})?
$rovers-type-seven$ = (<seven-neg=y>:{be'} |\
                      <seven-intens=y>:{qu'} |\
                      {<seven-neg=y><base-intens-neg=y>}:{be'qu'} |\
                      <seven-neg-intens=y>:{qu'be'})?
$rovers-type-eight$ = (<eight-neg=y>:{be'} |\
                      <eight-intens=y>:{qu'} |\
                      {<eight-neg=y><base-intens-neg=y>}:{be'qu'} |\
                      <eight-neg-intens=y>:{qu'be'})?
$rovers-base-imperative$ = (<base-intens=y>:{qu'})?
$rovers-type-one-imperative$ = ( <one-intens=y>:{qu'} )?
$rovers-type-two-imperative$ = ( <two-intens=y>:{qu'} )?
$rovers-type-three-imperative$ = ( <three-intens=y>:{qu'} )?
$rovers-type-four-imperative$ = ( <four-intens=y>:{qu'} )?
$rovers-type-five-imperative$ = ( <five-intens=y>:{qu'} )?
$rovers-type-six-imperative$ = ( <six-intens=y>:{qu'} )?
$rovers-type-seven-imperative$ = ( <seven-intens=y>:{qu'} )?
$rovers-type-eight-imperative$ = ( <eight-intens=y>:{qu'} )?
$rovers-ha$ = (<undo=y>:{Ha'})?
$rovers-qo-imperative$ = (<imp-neg=y>:{Qo'})?
$rovers-qo-non-imperative$ = (<refusal=y>:{Qo'})?

$suffixes-specified-subject-gen$ = $suffixes-type-two$ $rovers-type-two$ \
    $suffixes-type-three$ $rovers-type-three$ $suffixes-type-four$ \
    $rovers-type-four$ $suffixes-type-five-lah$ $rovers-type-five$ \
    $suffixes-type-six$ $rovers-type-six$ $suffixes-type-seven$ \
    $rovers-type-seven$ $suffixes-type-eight$ $rovers-type-eight$ \
    $rovers-qo-non-imperative$ $suffixes-type-nine-no-vis$
$suffixes-specified-subject-moh$ = $suffixes-type-two$ $rovers-type-two$ \
    $suffixes-type-three$ $rovers-type-three$ $suffixes-type-four-moh$ \
    $rovers-type-four$ $suffixes-type-five-lah$ $rovers-type-five$ \
    $suffixes-type-six$ $rovers-type-six$ $suffixes-type-seven$ \
    $rovers-type-seven$ $suffixes-type-eight$ $rovers-type-eight$ \
    $rovers-qo-non-imperative$ $suffixes-type-nine-no-vis$
$suffixes-imperative-gen$ = $suffixes-type-two$ $rovers-type-two-imperative$ \
```

```
    $suffixes-type-three$ $rovers-type-three-imperative$ $suffixes-type-
four$ \
    $rovers-type-four-imperative$ $suffixes-type-five-lah$ $rovers-type-
five$ \
    $suffixes-type-six$ $rovers-type-six-imperative$ $suffixes-type-seven$ \
    $rovers-type-seven-imperative$ $suffixes-type-eight$ \
    $rovers-type-eight-imperative$ $rovers-qo-imperative$
$suffixes-imperative-moh$ = $suffixes-type-two$ $rovers-type-two-imperative$ \
    $suffixes-type-three$ $rovers-type-three-imperative$ \
    $suffixes-type-four-moh$ $rovers-type-four-imperative$ \
    $suffixes-type-five-lah$ $rovers-type-five$ $suffixes-type-six$ \
    $rovers-type-six-imperative$ $suffixes-type-seven$ \
    $rovers-type-seven-imperative$ $suffixes-type-eight$ \
    $rovers-type-eight-imperative$ $rovers-qo-imperative$
$suffixes-unspecified-subject-gen$ = $suffixes-type-two$ $rovers-type-two$ \
    $suffixes-type-three$ $rovers-type-three$ $suffixes-type-four$ \
    $rovers-type-four$ $suffixes-type-five-lu$ $rovers-type-five$ \
    $suffixes-type-six$ $rovers-type-six$ $suffixes-type-seven$ \
    $rovers-type-seven$ $suffixes-type-eight$ $rovers-type-eight$ \
    $rovers-qo-non-imperative$ $suffixes-type-nine-no-vis$
$suffixes-specified-subject-vis$ = $suffixes-type-two$ $rovers-type-two$ \
    $suffixes-type-three$ $rovers-type-three$ $suffixes-type-four$ \
    $rovers-type-four$ $suffixes-type-five-lah$ $rovers-type-five$ \
    $suffixes-type-six$ $rovers-type-six$ $suffixes-type-seven-tah$ \
    $rovers-type-seven$ $suffixes-type-eight$ $rovers-type-eight$ \
    $rovers-qo-non-imperative$ $suffixes-type-nine-vis$
$suffixes-specified-subject-vis-moh$ = $suffixes-type-two$ $rovers-type-two$ \
    $suffixes-type-three$ $rovers-type-three$ $suffixes-type-four-moh$ \
    $rovers-type-four$ $suffixes-type-five-lah$ $rovers-type-five$ \
    $suffixes-type-six$ $rovers-type-six$ $suffixes-type-seven-tah$ \
    $rovers-type-seven$ $suffixes-type-eight$ $rovers-type-eight$ \
    $rovers-qo-non-imperative$ $suffixes-type-nine-vis$
$suffixes-unspecified-subject-vis$ = $suffixes-type-two$ $rovers-type-two$ \
    $suffixes-type-three$ $rovers-type-three$ $suffixes-type-four$ \
    $rovers-type-four$ $suffixes-type-five-lu$ $rovers-type-five$ \
    $suffixes-type-six$ $rovers-type-six$ $suffixes-type-seven-tah$ \
    $rovers-type-seven$ $suffixes-type-eight$ $rovers-type-eight$ \
    $rovers-qo-non-imperative$ $suffixes-type-nine-vis$

"k-nouns-lang.lex" <animate=y>:<> $nouns-type-one$ $nouns-lang-type-two$ \
    $nouns-type-three$ $nouns-type-four-lang-all$ $nouns-type-five$ |\
"k-nouns-unmarked.lex" <animate=n>:<> $nouns-type-one$ \
    $nouns-unmarked-type-two$ $nouns-type-three$ \
    $nouns-type-four-non-lang-all$ $nouns-type-five$ |\
"k-nouns-body.lex" <animate=n>:<> $nouns-type-one$ $nouns-body-type-two$ \
    $nouns-type-three$ $nouns-type-four-non-lang-all$ $nouns-type-five$ |\
($prefixes-object$ |\
    ($prefixes-no-object$ {<obj-pers=UNSPEC><obj-num=UNSPEC>}:<>)) \
    "k-verbs-trans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-specified-subject-gen$ |\
($prefixes-object$ |\
    ($prefixes-no-object$ {<obj-pers=UNSPEC><obj-num=UNSPEC>}:<>)) \
    "k-verbs-trans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-specified-subject-vis$ |\
$prefixes-object-imperative$ "k-verbs-trans.lex" $rovers-base-imperative$ \
     $rovers-ha$ $suffixes-imperative-gen$ |\
$prefixes-object-imperative$ "k-verbs-intrans.lex" $rovers-base-imperative$ \
```

```
     $rovers-ha$ $suffixes-imperative-moh$ |\
$prefixes-no-object$ {<obj-pers=NONE><obj-num=NONE>}:<> \
    "k-verbs-intrans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-specified-subject-gen$ |\
$prefixes-object$ "k-verbs-intrans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-specified-subject-moh$ |\
$prefixes-no-object-imperative$ "k-verbs-intrans.lex" $rovers-base-imperative$
\
     $rovers-ha$ $suffixes-imperative-gen$ |\
$prefixes-no-object$ "k-verbs-trans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-type-one-egh$ $suffixes-specified-subject-gen$ |\
$prefixes-no-object$ "k-verbs-intrans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-type-one-egh$ $suffixes-specified-subject-moh$ |\
$prefixes-no-object-imperative$ "k-verbs-trans.lex" $rovers-ha$ \
    $rovers-base-imperative$ $suffixes-type-one-egh$ \
    $suffixes-imperative-gen$ |\
$prefixes-no-object-imperative$ "k-verbs-intrans.lex" $rovers-ha$ \
    $rovers-base-imperative$ $suffixes-type-one-egh$ \
    $suffixes-imperative-moh$ |\
$prefixes-no-object$ "k-verbs-trans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-type-one-egh$ $suffixes-specified-subject-vis$ |\
$prefixes-no-object-plural$ "k-verbs-trans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-type-one-chuq$ $suffixes-specified-subject-gen$ |\
$prefixes-no-object-plural$ "k-verbs-intrans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-type-one-chuq$ $suffixes-specified-subject-moh$ |\
$prefixes-no-object-imperative-plural$ "k-verbs-trans.lex" $rovers-ha$ \
    $rovers-base-imperative$ $suffixes-type-one-chuq$ \
    $suffixes-imperative-gen$ |\
$prefixes-no-object-imperative-plural$ "k-verbs-intrans.lex" $rovers-ha$ \
    $rovers-base-imperative$ $suffixes-type-one-chuq$ \
    $suffixes-imperative-moh$ |\
$prefixes-no-object-plural$ "k-verbs-trans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-type-one-chuq$ $suffixes-specified-subject-vis$ |\
$prefixes-indef-subj$ "k-verbs-trans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-unspecified-subject-gen$ |\
$prefixes-indef-subj$ "k-verbs-trans.lex" $rovers-ha$ $rovers-base$ \
    $suffixes-unspecified-subject-vis$
```