

Introduction to Holmes Loglan Parsers

Randall Holmes

November 5, 2016

1 What is this File?

This is intended to give a little orientation if you want to try to use either of my Loglan parsers (the Moscow ML version or the Python version).

The bulk of the file is the actual collection of parses of the corpus of Loglan utterances in Notebook 3 (edited as necessary to parse with the new version of the grammar). This may have uses for any Loglan student independently of introduction to features of the parser; it is also the reason why the file is so large (the introduction to the parsers is not long).

10/29/2016 updating this document, it was severely outdated

4/19/2014: the parser output is a lot prettier than in the first version, and the file is a lot larger because I threw in all the parses from the NB3 corpus that I have done so far as an appendix.

2 Where are the parsers?

To run the Standard ML parser, you would need to install Moscow ML (instructions on my page) then get the files `pegparser.sml` (the PEG parser engine) and `loglangrammar.sml` (the file of commands that build the actual Loglan grammar) from my web page.

To run the Python version, you would need to install Python (much easier to find than Moscow ML) then get the files `peg.py` (the PEG parser engine), `loglanpreamble.py` (context file for the parser), and `loglan.py` (the file of commands that build the actual Loglan grammar) from my web page. You might also want `corpus.py`, which contains commands that will parse a large

initial segment of the NB3 teaching corpus of sample Loglan sentences. This file contains comments on where my parser differs from LIP.

The PEG for Loglan is in the file `Loglan.peg` also on my web page. If you want to build a Loglan parser using a different PEG engine, this is what you will want.

Currently I edit the grammar under the ML parser, which has a command which builds the file `loglan.py` for the Python version (a variation of the command which builds `Loglan.peg`), so the two parsers should agree. I find myself using the ML parser only when I want to actually edit the grammar; it seems much more convenient to use the Python parser on actual utterances I want to parse.

3 Relations to LIP

The “official” parser LIP is now quite old. Its *grammar* is good (though now somewhat different from ours). Its morphology (construction of compound little words, notably) is opaque and sometimes demonstrably buggy. Its phonology is completely opaque. It is basically impossible for us to maintain (or at least, for me to maintain) and changes have been made in the language which are not reflected in LIP.

The grammatical portion of LIP is based on a BNF grammar found in the file `trial.85` which is verified to be unambiguous with yacc (I hope I said this correctly on a technical level).

I have built a PEG parser instead of a BNF to be used with yacc. You can read about PEG (Phrase Expression Grammars) by looking at Bryan Ford’s original article, linked from my web site. A PEG is unambiguous no matter what; unlike a BNF, where alternative parses are presented, an order is given in which they are to be tested, and one always takes the first alternative that works. This doesn’t mean that we do not have the same potential problems of unintended parses that a BNF grammar does: in a PEG, problems will exist if the alternatives are given in the wrong order so that an intended parse is actually passed over, whereas in a BNF an unintended parse would happen because of an overlooked ambiguity. What is true is that the ambiguity criterion for a BNF can be checked mechanically for a certain class of grammars, which has been used to test the Loglan grammar in the past; I do not yet have a precise definition of a criterion for “threatened unintended parse” which I can check the PEG for, though I

do have some ideas of conditions that would at least suggest an unintended parse was possible which could be checked mechanically. It should also be noted that PEGs hate left recursion, which has caused some awkwardness in preparing the Loglan PEG.

My parser works from the level of letters upward. In LIP, the phonology and the morphology of little words are hardcoded in mysterious ways into a preparser, and they are not ambiguity checked in the way the grammar is. In my parsers, there is no preparser at all: everything is parsed from the level of letters upward, though if you look at my Loglan agenda document, which includes the PEG with comments in English, you will find it divided into a phonology section, a lexer section (morphology), and a grammar section.

The formal definition of the phonology is more or less given in NB3, and I have implemented it in my PEG. The formal definitions of important classes of little words is not given anywhere (except implicitly in LIP in a format a human being probably cannot read, and which is known to have bugs in it) so I wrote my own, which appears to correlate reasonably well with the existing language definition. I then ported the trial.85 grammar into the PEG. There are both approved and provisional changes to the language built into my PEG. It is however true that my parser handles almost all sentences in the NB3 corpus as JCB intended.

There are a couple of changes which do need to be noted to avoid frustration if you start parsing sentences yourself. Unmarked vocatives are no longer allowed (you must precede a vocative with `hoi`). I impose a phonology on names, quite close to that on borrowings which I deduced from NB3, which has the one notable effect that syllabic (“vocalic”) consonants must be doubled. The name `Rl` becomes `Rr1` and `Suzn` becomes `Suznn` for example. Our change in the vocabulary when we abandoned the `slinkui` test may occasionally affect predicate forms. Pause/GU equivalence is now completely disabled. There are other changes in the grammar, but my experience with the corpus suggests that you will not often encounter them.

In spite of the seemingly radical difference in approach, my reimplementations of the TLI Loglan grammar is basically conservative. My intention is that the PEG parser should handle almost exactly the same set of sentences that LIP does in almost exactly the same way; places in which its approach is fundamentally different should either be rare, or if actually rather common (the way that the PEG handles names is quite different) nonetheless invisible to the user (almost all names in use parse with my PEG without difficulty, but it insists on checking them for pronounceability).

4 Using the Python parser

The file `peg.py` contains the PEG parser engine. The file `loglan.py` calls the previous file as a module, sets some display options defined in `peg.py`, and builds the Loglan grammar.

When the parser comes up, it will be in an interface which allows repeated entry of utterances to parse. If you do type `?` you will get an extensive menu of curious additional things you can do in the interface, which I will not discuss here. Some of them are dangerous: you can edit the grammar live from the interface!

Type an utterance to parse (or `...` to exit, `?` for command help):

```
Mi cluva tu
```

```
Mi cluva tu
```

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      tu]]]
```

```
Parser cache size 5675
```

```
-----
```

The example above shows how to use it. If you fall out of the interface, you can type `interface()` to recover it.

The settings I alluded to above are listed in the comments in `peg.py`. They are mostly toggles (the same command turns a feature on that turns it off). I would be disinclined to mess with the settings as a user at this point: the current configuration works reasonably well. To issue these commands from the command line, you need to break out of the interface (enter `...` as it instructs you).

```

#useful settings:

# Verbose() toggles display of grammar class names
# Shallow() makes the grammar display shallower in a way that is best
# found out by using it
# Numberbrackets() puts numbers <n> next to brackets indicating depth
# Indent() turns on indented display
# Compact (<name of grammar class>) will cause it to display that class
# as an unlabelled unparsed string
# Decompact() turns off all Compact settings

```

If one executes `Verbose()` one turns off and on the display of the names of grammar classes. The LIP display of a sentence is not verbose; the default setting of this parser is verbose (you can see names of grammar classes above). If one executes `Shallow()` one turns on or off the option to display *all* grammar classes. The default state is shallow: roughly speaking, if a string belongs to more than one grammar class, only outermost ones are displayed, along with innermost ones for single words. If one executes `Numberbrackets()` one turns on and off a feature which numbers brackets by depth. `Indent()` turns the indentation you see above on and off. The `Compact` command, which takes the name of a grammar class as an argument, suppresses all internal grammar of a string in that class in the display – notice that we do not see parses of predicate words (which have considerable internal “grammar”!) in the display above. `Decompact()` will turn off all compact settings at once. You can try these commands out, but I think the default settings are likely the most useful for you.

The current state is verbose (class names are shown), indented, so one can see parse depth visually, shallow – not all grammar classes are shown, but I think the current programming of the shallow setting is correct, in that one sees what one needs to see and not too much, and predicate and name words are compacted. Some of the `cmapua` classes should probably be compacted too.

The command `Showrule()` will show you the PEG rule defining a class.

4/21/14 a new command `Greedy()` has been added which toggles a state which modifies the PEG behaviour: when alternatives are presented, it takes the first among the parses which are longest. Any difference in parses done

with the `Greedy()` setting probably indicates a formal problem with the grammar. It runs quite slowly on this setting; it is just a testing device.

```
>>> Showrule('predicate')
((predicate1/identpred) freemod?)
>>> Showrule('predicate1')
(((predicate2 AGE freemod? predicate1)/predicate2) freemod?)
>>> Showrule('predicate2')
(!GE ((backpred (A1 predicate2)+ termset?)/backpred))
>>>
```

Above see an investigation using `Showrule`. You can harvest names of grammar classes from parses and from rules you have displayed. You can read about PEG notation on my web site (I have a link to Ford's original paper).

The command `Utteras` will allow you to parse a string using a specific grammar class.

```
>>> Utteras('predicate', 'cluva tu')
cluva tu

cluva tu
```

```
    [predicate1:
      [sentpred:
        cluva]
      [termset:
        [DA1:
          tu]]]
-----
```

```
>>> Utteras('Predicate', 'cluva')
cluva
```

```

[Primitive:
  [CCV:
    [InitialCC:
      c1]
    [V2:
      u]]
  [CV:
    [C:
      v]
    [V2:
      a]]]
-----

```

In the second example you can see the usually hidden phonological parse of a predicate word.

5 Using the ML parser

I suggest using the Python parser!

You need to install Moscow ML and start the interpreter with the files `pegparser.sml` and `loglanggrammar.sml` in the right places.

Enter the following commands:

```

compile "pegparser.sml"; (* only the first time *)
load "pegparser";
open pegparser;
use "loglanggrammar.sml";

```

Ignore the vast amounts of feedback that you get from some of these.

```
- utter "Da blanu";
```

```
utterance:statement:[terms:DAO:'Da',predicate:PRED:  
[' ',Predicate:CCVCV:[CCV:[InitialCC:'bl',V2:'a'],CV:[C:letter:'n',V2:'u']  
> val it = () : unit
```

```
- niceprec();  
> val it = () : unit  
- utter "Da blanu";
```

Da blanu

```
utterance:  
[  
  uttF: statement:  
    [  
      subject: argument:  
        Da  
      predicate: predunit:  
        blanu  
    ]  
end:  
  !!!check!!!  
]
```

next?

The `niceprec()` command causes the parser to display about the same information in about the same format as the Python parser. Notice that the unedited parse given first includes detailed information down to the level of letters and phonological structure of individual words, which we do not want to see when we parse a large utterance!

There is a sample of the syntax needed to parse a sentence.

I follow this with the description of user commands from the preamble to `pegparser.sml`. Notice that display options can be reset in much the same way as with the Python parser.

Here are user commands you might want to know about.


```

versiondate(); (* version of this parser generator that you have *)

showdef "<name>"; (* look at the grammar rule called <name> *)

Parsedef "<definition>"; (* add a line to the current PEG grammar,\
                           using the exact formalism of Ford's paper.\
                           Not for the faint of heart *)

removedef "<name>"; (* remove a name from the grammar -- you can't
                    change a definition that is present with Parsedef *)

                    (* use of removedef is deprecated -- you really
                    should use clear() and reload the file every time
                    you make a change. Termination checking and
                    ambiguity checking functions are quite meaningless
                    once you start making
                    manual changes to the grammar. *)

parsetest "<class>" "<string>"; (* attempt to parse <string> using the rule
                                called <name> *)

verbose(); (* turn on the class labels in parser output *)

unverbose(); (* turn off the class labels in parser output *)

reallyverbose(); (* turn on *all* the class labels in parser output *)
                (* this is actually independent of "verbose"; if "verbose"
                is off it wont turn on labels, but when you do turn them on
                again there will be lots more *)

unreallyverbose(); (* turn off the excessive class labels in parser output *)

hide "<name>"; (* terms of grammar class <name> will be just shown as strings,
                all structure hidden *)

unhide "<name>";

```

```

tag "<name>"; (*when parsing, all failures of rule <name> will be shown --
              including expected ones in NOT tests, so don't assume this
              always signals errors *)

untag "<name>"; (* obvious related commands *)

cleartags();

(* new 8/23/2013 *)

numberbrackets(); (* turns on numerical marking of brackets in parses
                  that should make matching clear but of course is cumbersome *)

unnumberbrackets(); (* turns bracket numbering off *)

setdepth <number>; (* inserts pretty-printing breaks and indentations
                  down to the <number>th level of bracketing. setdepth 0; turns this off. *)

clear(); (* clears all grammar rules except the internally reserved
         ones that it uses to parse PEG definitions. Usually typed just before
         use "loglangrammar.sml" when you are trying to change something. It
         is actually much better if you want to modify the grammar rules to
         change them in the file then clear the grammar and rerun the entire
         file. The termination checks are not meaningful if you start editing
         random commands in random orders. This also clears caches used by
         the ambiguity checker. *)

setnopause(); (* causes it not to pause at error messages *)

setpause(); (* much better for you *)

savegrammar "<filename>"; (* saves the current grammar to <filename>.peg
                          as a text file in PEG format *)

savegrammar2 "<filename>"; (* saves the current grammar to <filename>.py
                           as a set of commands to build the grammar under
                           my Python PEG engine *)

```

```

loadgrammar "<filename>"; (* read grammar from <filename>.peg --
                           its more graceful than it was at first
                           but still stops with an error, which you
                           can ignore. Comment lines beginning with
                           # can be put in saved theory files. *)

settop "<class>"; (* specify the top level class of your working grammar *)

utter "<string>"; (* parse string as an item of the top level class *)

viewcache(); (* issue this command immediately after a parsetest
              or utter command and you will get to page through all the successful
              partial parses of bits of the string in the parser cache. Ones
              starting earlier are displayed first.
              Of course it is mainly useful (?) to try
              to diagnose reasons for a parse failure.
              The volume of what is displayed may be
              large. You need to hit enter repeatedly to page through it. *)

postcomment "<class>" "<comment>"; (* this will add a comment
                                     to the saved grammar file just above the entry
                                     for the class. Inserts # after returns so that
                                     it will be a comment for PEG grammar and my
                                     loadgrammar command. The command needs to be issued
                                     after the class is defined. *)

```

If the parser generator gives you messages that a rule fails a termination check, it *has* parsed and installed the rule; it is warning you that it can't prove that the rule terminates. Very often this is due to a typo in the rule, but not always. In any event, you should not be installing your own grammar rules unless you understand PEG grammars; they do NOT necessarily work the way you expect if you just copy over a BNF grammar.

The parser generator currently checks for termination of the grammar.

It is intended to have ambiguity checking functions as well, but these do not work at the moment.

6 Appendix: the NB3 corpus so far

Just for fun, I include the entire output of `corpus.py` so far, a lot of parses. One of them fails, I know, and I think it should (I think we should count `ne, i to, i te, i fo...`

This has been updated with the new parser version of 9/24/2016.

```
Python 3.5.2 (v3.5.2:4def2a2901a5, Jun 25 2016, 22:18:55) [MSC v.1900 64 bit (AMD64)]
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\mrand\Desktop\Loglan\corpus.py =====
Hapci!
```

```
[uttF: uttD:
      [sentence: predunit:
            Hapci]
      [period:
            !]]
```

```
Parser cache size 1461
-----
Eo skitu
```

```
[freemod:
      Eo]
[utterance: predunit:
            skitu]
```

```
Parser cache size 3170
-----
```

Stali eo

```
[uttF: predunit:
      Stali
      [freemod:
        eo]]
```

Parser cache size 3527

Hompi ti

```
[uttF: barepred:
      [sentpred: predunit:
        Hompi]
      [termset: argument:
        ti]]
```

Parser cache size 3854

Ao kerju tu

```
[freemod:
      Ao]
[utterance: barepred:
      [sentpred: predunit:
        kerju]
      [termset: argument:
        tu]]
```

Parser cache size 5427

Ai mi kerju mi

```
[freemod:
```

```
    Ai]
  [utterance: statement:
    [subject: argument:
      mi]
    [predicate: barepred:
      [sentpred: predunit:
        kerju]
      [termset: argument:
        mi]]]
```

Parser cache size 7567

Eo helba mi

```
  [freemod:
    Eo]
  [utterance: barepred:
    [sentpred: predunit:
      helba]
    [termset: argument:
      mi]]]
```

Parser cache size 5825

Ao no mi helba tu

```
  [freemod:
    Ao]
  [utterance: statement:
    [subject: argument:
      no
      [argument1: arg1a:
        mi]]
    [predicate: barepred:
      [sentpred: predunit:
        helba]
```

```
[termset: argument:  
tu]]]
```

Parser cache size 9700

No takna

```
[uttF: predunit:  
No  
[predunit1:  
takna]]]
```

Parser cache size 3480

Ae no takna mi

```
[freemod:  
Ae]  
[utterance: barepred:  
[sentpred: predunit:  
no  
[predunit1:  
takna]]  
[termset: argument:  
mi]]]
```

Parser cache size 7947

Eo no godzi

```
[freemod:  
Eo]  
[utterance: predunit:  
no  
[predunit1:
```

godzi]]

Parser cache size 5493

No totco mi

```
[uttF: barepred:
  [sentpred: predunit:
    No
    [predunit1:
      totco]]
  [termset: argument:
    mi]]
```

Parser cache size 5910

No mi totco tu

```
[uttF: statement:
  [subject: argument:
    No
    [argument1: arg1a:
      mi]]
  [predicate: barepred:
    [sentpred: predunit:
      totco]
    [termset: argument:
      tu]]]
```

Parser cache size 8073

Ae no mi puncko tu

```
[freemod:
  Ae]
```



```
[utterance: statement:
  [subject: argument:
    no
    [argument1: arg1a:
      mi]]
  [predicate: barepred:
    [sentpred: predunit:
      puncko]
    [termset: argument:
      tu]]]
```

Parser cache size 12247

Eo no nordri mi

```
[freemod:
  Eo]
[utterance: barepred:
  [sentpred: predunit:
    no
    [predunit1:
      nordri]]
  [termset: argument:
    mi]]]
```

Parser cache size 10104

No eo nordri mi

```
[freemod:
  No eo]
[utterance: barepred:
  [sentpred: predunit:
    nordri]
  [termset: argument:
    mi]]]
```

Parser cache size 9144

No nordri eo mi

```
[uttF: barepred:
  [sentpred: predunit:
    No
    [predunit1:
      nordri
      [freemod:
        eo]]]
  [termset: argument:
    mi]]
```

Parser cache size 10338

No nordri mi eo

```
[uttF: barepred:
  [sentpred: predunit:
    No
    [predunit1:
      nordri]]
  [termset: argument:
    mi
    [freemod:
      eo]]]
```

Parser cache size 10338

Hoi Djim

```
[freemod: voc:
  Hoi
```

[name:
Djim]]

Parser cache size 4850

Loi

[freemod:
Loi]

Parser cache size 2691

Eo nenkaa

[freemod:
Eo]
[utterance: predunit:
nenkaa]

Parser cache size 5190

Sia ai

[freemod:
Sia
[freemod:
ai]]

Parser cache size 4902

Loi, hoi Fred

[freemod:
Loi

```
[freemod:
  [comma:
    ,]
  [freemod: voc:
    hoi
    [name:
      Fred]]]]
```

Parser cache size 7110

Hoi Djin, loi

```
[freemod: voc:
  Hoi
  [name:
    Djin,
    [freemod:
      loi]]]
```

Parser cache size 7938

Sia, hoi Ted

```
[freemod:
  Sia
  [freemod:
    [comma:
      ,]
    [freemod: voc:
      hoi
      [name:
        Ted]]]]]
```

Parser cache size 7148

Oi resto

```
[freemod:
  Oi]
[utterance: predunit:
  resto]
```

Parser cache size 2948

Sia ao no

```
[freemod:
  Sia
  [freemod:
    ao]]
[utterance: uttC:
  no]
```

Parser cache size 7229

Oi titci ti

```
[freemod:
  Oi]
[utterance: barepred:
  [sentpred: predunit:
    titci]
  [termset: argument:
    ti]]
```

Parser cache size 5397

Sia ao

```
[freemod:
```

```
Sia
[freemod:
  ao]]
```

Parser cache size 4899

Siu, hoi Selis

```
[freemod:
  Siu
  [freemod:
    [comma:
      ,]
    [freemod: voc:
      hoi
      [name:
        Selis]]]]]
```

Parser cache size 8204

Sia ae no

```
[freemod:
  Sia
  [freemod:
    ae]]
[utterance: uttC:
  no]
```

Parser cache size 7420

Ea mu godzi

```
[freemod:
  Ea]
```

```
[utterance: statement:
  [subject: argument:
    mu]
  [predicate: predunit:
    godzi]]
```

Parser cache size 5521

Ao

```
[freemod:
  Ao]
```

Parser cache size 2165

Eo mi titci ti

```
[freemod:
  Eo]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: barepred:
    [sentpred: predunit:
      titci]
    [termset: argument:
      ti]]]
```

Parser cache size 7989

Oi

```
[freemod:
  Oi]
```

Parser cache size 2127

Eo no mi titci ti

```
[freemod:
  Eo]
[utterance: statement:
  [subject: argument:
    no
    [argument1: arg1a:
      mi]]
  [predicate: barepred:
    [sentpred: predunit:
      titci]
    [termset: argument:
      ti]]]
```

Parser cache size 10107

Oi no

```
[freemod:
  Oi]
[utterance: uttC:
  no]
```

Parser cache size 4355

Loa, hoi Tcet

```
[freemod:
  Loa
  [freemod:
    [comma:
      ,]
```



```
[freemod: voc:
  hoi
  [name:
    Tcet]]]]
```

Parser cache size 7447

Hoi Djan, santi

```
[freemod: voc:
  Hoi
  [name:
    Djan, ]]
[utterance: predunit:
  santi]
```

Parser cache size 9030

La Djan, santi

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: predunit:
    santi]]
```

Parser cache size 9239

Prano, hoi Mel

```
[uttF: predunit:
  Prano
  [freemod:
```

```
[comma:
  ,]
[freemod: voc:
  hoi
  [name:
    Mel]]]]
```

Parser cache size 5619

Prano la Mel

```
[uttF: barepred:
  [sentpred: predunit:
    Prano]
  [termset: argument:
    la
    [name:
      Mel]]]]
```

Parser cache size 6055

Prano ta, hoi Mel

```
[uttF: barepred:
  [sentpred: predunit:
    Prano]
  [termset: argument:
    ta
    [freemod:
      [comma:
        ,]
      [freemod: voc:
        hoi
        [name:
          Mel]]]]]]
```

Parser cache size 8073

Prano ta hoi Mel, e ti hoi Pol

```
[uttF: barepred:
  [sentpred: predunit:
    Prano]
  [termset: argument:
    [arg7: arg1a:
      ta
      [freemod: voc:
        hoi
        [name:
          Mel, ]]]
    e
    [arg7: arg1a:
      ti
      [freemod: voc:
        hoi
        [name:
          Pol]]]]]]
```

Parser cache size 16409

Hoi Keit, skesa mi

```
[freemod: voc:
  Hoi
  [name:
    Keit, ]]
[utterance: barepred:
  [sentpred: predunit:
    skesa]
  [termset: argument:
    mi]]
```

Parser cache size 11316

Skesa hoi Keit, mi

```
[uttF: barepred:
  [sentpred: predunit:
    Skesa
    [freemod: voc:
      hoi
      [name:
        Keit, ]]]
  [termset: argument:
    mi]]
```

Parser cache size 8600

Skesa mi hoi Keit

```
[uttF: barepred:
  [sentpred: predunit:
    Skesa]
  [termset: argument:
    mi
    [freemod: voc:
      hoi
      [name:
        Keit]]]]]
```

Parser cache size 8159

Skesa la Bab, hoi Keit

```
[uttF: barepred:
  [sentpred: predunit:
    Skesa]
```

```
[termset: argument:
  la
  [name:
    Bab,
    [freemod: voc:
      hoi
      [name:
        Keit]]]]]
```

Parser cache size 10441

Hoi Karl, kukra la Dik

```
[freemod: voc:
  Hoi
  [name:
    Karl, ]]
[utterance: barepred:
  [sentpred: predunit:
    kukra]
  [termset: argument:
    la
    [name:
      Dik]]]
```

Parser cache size 14150

La Karl, kukra la Dik

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Karl, ]]
  [predicate: barepred:
    [sentpred: predunit:
```

```
      kukra]
      [termset: argument:
        la
        [name:
          Dik]]]]
```

Parser cache size 14359

La Stivnn, sunho hoi Ruf

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Stivnn, ]]
  [predicate: predunit:
    sunho
    [freemod: voc:
      hoi
      [name:
        Ruf]]]]]
```

Parser cache size 13848

La Stivnn, sunho la Ruf

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Stivnn, ]]
  [predicate: barepred:
    [sentpred: predunit:
      sunho]
    [termset: argument:
      la
```

```
[name:
  Ruf]]]]
```

Parser cache size 14322

La An, takna la Ritcrrd, mi

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      An, ]]
  [predicate: barepred:
    [sentpred: predunit:
      takna]
  [termset: terms:
    [argumentA: argument:
      la
      [name:
        Ritcrrd, ]]
    [argumentB: argument:
      mi]]]]
```

Parser cache size 18142

Hoi An, takna la Ritcrrd, mi

```
[freemod: voc:
  Hoi
  [name:
    An, ]]
[utterance: barepred:
  [sentpred: predunit:
    takna]
  [termset: terms:
    [argumentA: argument:
```

```
    la
    [name:
      Ritcrrd, ]]
  [argumentB: argument:
    mi]]]
```

Parser cache size 17879

La Djeik, farfu hoi Djan, mi la Suznn

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djeik, ]]
  [predicate: barepred:
    [sentpred: predunit:
      farfu
      [freemod: voc:
        hoi
        [name:
          Djan, ]]]]
  [termset: terms:
    [argumentA: argument:
      mi]
    [argumentB: argument:
      la
      [name:
        Suznn]]]]]
```

Parser cache size 21545

La Djeik, farfu la Djan, mi hoi Suznn

```
[uttF: statement:
  [subject: argument:
```



```

La
  [name:
    Djeik, ]]
[predicate: barepred:
  [sentpred: predunit:
    farfu]
  [termset: terms:
    [argumentA: argument:
      la
      [name:
        Djan, ]]
    [argumentB: argument:
      mi
      [freemod: voc:
        hoi
        [name:
          Suznn]]]]]]]

```

Parser cache size 21591

Hoi Mel, prano la Djan Pol Djonz

```

[freemod: voc:
  Hoi
  [name:
    Mel, ]]
[utterance: barepred:
  [sentpred: predunit:
    prano]
  [termset: argument:
    la
    [name:
      Djan
      Pol
      Djonz]]]

```

Parser cache size 20140

Prano la Djan Pol Djonz, hoi Mel

```
[uttF: barepred:
  [sentpred: predunit:
    Prano]
  [termset: argument:
    la
    [name:
      Djan
      Pol
      Djonz,
      [freemod: voc:
        hoi
        [name:
          Mel]]]]]
```

Parser cache size 16951

Bleka le nirda

```
[uttF: barepred:
  [sentpred: predunit:
    Bleka]
  [termset: argument:
    le
    [descpred: predunit:
      nirda]]]
```

Parser cache size 4673

Eo penso le nable

```
[freemod:
  Eo]
```

```
[utterance: barepred:
  [sentpred: predunit:
    penso]
  [termset: argument:
    le
    [descpred: predunit:
      nable]]]
```

Parser cache size 6859

Eo penso ne nable

```
[freemod:
  Eo]
[utterance: barepred:
  [sentpred: predunit:
    penso]
  [termset: argument:
    ne
    [descpred: predunit:
      nable]]]
```

Parser cache size 7121

Eo penso su nable

```
[freemod:
  Eo]
[utterance: barepred:
  [sentpred: predunit:
    penso]
  [termset: argument:
    su
    [descpred: predunit:
      nable]]]
```

Parser cache size 7141

Ai mi ckano le bunbo

```
[freemod:
  Ai]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: barepred:
    [sentpred: predunit:
      ckano]
    [termset: argument:
      le
      [descpred: predunit:
        bunbo]]]]]
```

Parser cache size 8340

Mi driki le purda

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      driki]
    [termset: argument:
      le
      [descpred: predunit:
        purda]]]]]
```

Parser cache size 6715

La Celdnn, godzi le sitci

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Celdnn, ]]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: argument:
      le
      [descpred: predunit:
        sitci]]]]]
```

Parser cache size 13390

Le rodlu ga gudbi le fitrua

```
[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      rodlu]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        gudbi]
      [termset: argument:
        le
        [descpred: predunit:
          fitrua]]]]]]]
```

Parser cache size 12892

Le ditca ga fundi mi le groda

```

[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      ditca]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        fundi]
      [termset: terms:
        [argumentA: argument:
          mi]
        [argumentB: argument:
          le
          [descpred: predunit:
            groda]]]]]]]]

```

Parser cache size 12397

Le troli ga plizo la Frenk, ne mipli

```

[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      troli]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        plizo]
      [termset: terms:
        [argumentA: argument:
          la
          [name:

```

```
Frenk, []
[argumentB: argument:
  ne
  [descpred: predunit:
    mipli]]]]]]
```

Parser cache size 15673

Le tcaro ga djipo le ponsu le kolro

```
[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      tcaro]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        djipo]
      [termset: terms:
        [argumentA: argument:
          le
          [descpred: predunit:
            ponsu]]
        [argumentB: argument:
          le
          [descpred: predunit:
            kolro]]]]]]]]
```

Parser cache size 13343

tu he

```
[uttF: statement:
  [subject: argument:
```

```
    tu]
  [predicate: predunit:
    he]]
```

Parser cache size 5151

Mi djela

```
  [uttF: statement:
    [subject: argument:
      Mi]
    [predicate: predunit:
      djela]]
```

Parser cache size 3244

Ta he

```
  [uttF: statement:
    [subject: argument:
      Ta]
    [predicate: predunit:
      he]]
```

Parser cache size 5154

Da muzgi

```
  [uttF: statement:
    [subject: argument:
      Da]
    [predicate: predunit:
      muzgi]]
```

Parser cache size 3484

Le cersi ga he

```
[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      cersi]]
  [predicate: markpred:
    ga
    [barepred: predunit:
      he]]]
```

Parser cache size 8297

Da komfu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    komfu]]]
```

Parser cache size 3483

Da he komfu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      he]
    [despredD: predunit:
      komfu]]]
```

Parser cache size 5813

Da nurmue komfu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      nurmue]
    [despredD: predunit:
      komfu]]]
```

Parser cache size 6490

Le nurtci he

```
[uttF: argument:
  Le
  [descpred: despredE:
    [despredD: predunit:
      nurtci]
    [despredD: predunit:
      he]]]
```

Parser cache size 8543

Le nurtci herba

```
[uttF: argument:
  Le
  [descpred: despredE:
    [despredD: predunit:
      nurtci]
```

```
[despredD: predunit:  
herba]]]
```

Parser cache size 7048

Le botci ga he sucmi

```
[uttF: statement:  
  [subject: argument:  
    Le  
    [descpred: predunit:  
      botci]]  
  [predicate: markpred:  
    ga  
    [barepred: sentpred:  
      [senpred3: predunit:  
        he]  
      [despredD: predunit:  
        sucmi]]]]]
```

Parser cache size 9006

Le botci ga tarle sucmi

```
[uttF: statement:  
  [subject: argument:  
    Le  
    [descpred: predunit:  
      botci]]  
  [predicate: markpred:  
    ga  
    [barepred: sentpred:  
      [senpred3: predunit:  
        tarle]  
      [despredD: predunit:  
        sucmi]]]]]
```

Parser cache size 7558

Leva he tcaro

```
[uttF: argument:
  Le
  va
  [descpred: despredE:
    [despredD: predunit:
      he]
    [despredD: predunit:
      tcaro]]]
```

Parser cache size 8062

Leva tcaro he

```
[uttF: argument:
  Le
  va
  [descpred: despredE:
    [despredD: predunit:
      tcaro]
    [despredD: predunit:
      he]]]
```

Parser cache size 8154

Levi ri herba ga he

```
[uttF: statement:
  [subject: argument:
    Le
    vi]
```

```
      ri
      [descpred: predunit:
        herba]]
[predicate: markpred:
  ga
  [barepred: predunit:
    he]]]
```

Parser cache size 12487

Da vendu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    vendu]]]
```

Parser cache size 3483

Leva ro junti na he

```
[uttF: statement:
  [subject: argument:
    Le
    va
    ro
    [descpred: predunit:
      junti]]
  [predicate: markpred:
    na
    [barepred: predunit:
      he]]]
```

Parser cache size 13055

Da na takna

```
[uttF: statement:  
  [subject: argument:  
    Da]  
  [predicate: markpred:  
    na  
    [barepred: predunit:  
      takna]]]
```

Parser cache size 6318

Da mrenu

```
[uttF: statement:  
  [subject: argument:  
    Da]  
  [predicate: predunit:  
    mrenu]]]
```

Parser cache size 3251

Ie da

```
[uttF: argument:  
  Ie  
  [arg5: arg1a:  
    da]]]
```

Parser cache size 5420

La Bab

```
[uttF: argument:
```

```
La
  [name:
    Bab]]
```

Parser cache size 5502

Ie la Bab

```
[uttF: argument:
  Ie
  [arg5: LANAME:
    la
    [name:
      Bab]]]
```

Parser cache size 7832

La Bab, bi le hapci

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Bab, ]]
  [predicate: identpred:
    bi
    [termset: argument:
      le
      [descpred: predunit:
        hapci]]]]]
```

Parser cache size 11278

Da hapci hu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      hapci]
    [termset: argument:
      hu]]]
```

Parser cache size 5898

Le ckela

```
[uttF: argument:
  Le
  [descpred: predunit:
    ckela]]
```

Parser cache size 3819

Ie le ckela

```
[uttF: argument:
  Ie
  [arg5: descriptn:
    le
    [descpred: predunit:
      ckela]]]
```

Parser cache size 6152

Le cninu ckela

```
[uttF: argument:
  Le
```



```
[descpred: despredE:
  [despredD: predunit:
    cninu]
  [despredD: predunit:
    ckela]]]
```

Parser cache size 4256

Ta bi le cninu ckela

```
[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: identpred:
    bi
    [termset: argument:
      le
      [descpred: despredE:
        [despredD: predunit:
          cninu]
        [despredD: predunit:
          ckela]]]]]]]
```

Parser cache size 8597

La Selis, cluva

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Selis, ]]
  [predicate: predunit:
    cluva]]]
```

Parser cache size 9964

Da cluva hu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      hu]]]
```

Parser cache size 5626

Da cluva la Pit

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      la
      [name:
        Pit]]]]]
```

Parser cache size 8030

I de cluva hu

```
I
[uttF: statement:
  [subject: argument:
    de]
```

```
[predicate: barepred:
  [sentpred: predunit:
    cluva]
  [termset: argument:
    hu]]]
```

Parser cache size 8329

La Alis

```
[uttF: argument:
  La
  [name:
    Alis]]]
```

Parser cache size 6838

Ie la Alis

```
[uttF: argument:
  Ie
  [arg5: LANAME2:
    la
    [name:
      Alis]]]]]
```

Parser cache size 9171

La Muhamed Alis

```
[uttF: argument:
  La
  [name:
    Muhamed
    Alis]]]
```

Parser cache size 11154

Da godzi la Italias

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: argument:
      la
      [name:
        Italias]]]]]
```

Parser cache size 10693

Da godzi de hu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: terms:
      [argumentA: argument:
        de]
      [argumentB: argument:
        hu]]]]]
```

Parser cache size 8179

Da godzi de la Danmark

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: terms:
      [argumentA: argument:
        de]
      [argumentB: argument:
        la
        [name:
          Danmark]]]]]]
```

Parser cache size 12422

Hu matma leva botci

```
[uttF: statement:
  [subject: argument:
    Hu]
  [predicate: barepred:
    [sentpred: predunit:
      matma]
    [termset: argument:
      le
      va
      [descpred: predunit:
        botci]]]]]
```

Parser cache size 9024

Ti dui

```
[uttF: statement:
```

```
[subject: argument:
  Ti]
[predicate: predunit:
  dui]]
```

Parser cache size 5423

Mi gudbi tu

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      gudbi]
    [termset: argument:
      tu]]]
```

Parser cache size 5939

Tu gudbi mi hu

```
[uttF: statement:
  [subject: argument:
    Tu]
  [predicate: barepred:
    [sentpred: predunit:
      gudbi]
    [termset: terms:
      [argumentA: argument:
        mi]
      [argumentB: argument:
        hu]]]]]
```

Parser cache size 8233

Mi gudbi tu ti

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      gudbi]
    [termset: terms:
      [argumentA: argument:
        tu]
      [argumentB: argument:
        ti]]]]]
```

Parser cache size 8273

Ei ti breba

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    ti]
  [predicate: predunit:
    breba]]]
```

Parser cache size 4877

Ia da breba

```
[freemod:
  Ia]
[utterance: statement:
  [subject: argument:
    da]]]
```

```
[predicate: predunit:
      breba]]
```

Parser cache size 4836

Ei ti ckela

```
[freemod:
      Ei]
[utterance: statement:
      [subject: argument:
            ti]
      [predicate: predunit:
            ckela]]
```

Parser cache size 4879

No. I da hasfa

```
[uttF:
      [uttE: uttC:
            No
            [period:
                  .]]
      I
      [uttF: statement:
            [subject: argument:
                  da]
            [predicate: predunit:
                  hasfa]]]
```

Parser cache size 7673

Ei ta hasfa


```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    ta]
  [predicate: predunit:
    hasfa]]
```

Parser cache size 5156

Ia no. I da vemsia

```
[freemod:
  Ia]
[utterance: uttF:
  [uttE: uttC:
    no
    [period:
      .]]
  I
  [uttF: statement:
    [subject: argument:
      da]
    [predicate: predunit:
      vemsia]]]
```

Parser cache size 11587

Ei tu cidja

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: predunit:
    cidja]]
```

cidja]]

Parser cache size 5139

No. I mi sonli

```
[uttF:
  [uttE: uttC:
    No
    [period:
      .]]
  I
  [uttF: statement:
    [subject: argument:
      mi]
    [predicate: predunit:
      sonli]]]
```

Parser cache size 7685

Ei tu cluva mi

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      mi]]]
```

Parser cache size 7305

Ia mi cluva tu

```
[freemod:
  Ia]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      tu]]]
```

Parser cache size 7296

Ei toi tradu

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    toi]
  [predicate: predunit:
    tradu]]
```

Parser cache size 5153

Ei toa logla

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    toa]
  [predicate: predunit:
    logla]]
```

Parser cache size 5656

Ei tu na bleka mi

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    na
  [barepred:
    [sentpred: predunit:
      bleka]
    [termset: argument:
      mi]]]]
```

Parser cache size 10137

No. Ibuo mi fa bleka tu

```
[uttF:
  [uttE: uttC:
    No
    [period:
      .]]
  I
  [freemod:
    buo]
  [uttF: statement:
    [subject: argument:
      mi]
    [predicate: markpred:
      fa
    [barepred:
```

```
[sentpred: predunit:
  bleka]
[termset: argument:
  tu]]]]
```

Parser cache size 14483

Ei tu pa tidjo mi

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    pa
  [barepred:
    [sentpred: predunit:
      tidjo]
    [termset: argument:
      mi]]]]
```

Parser cache size 10401

Ia mi pa mutce tidjo tu

```
[freemod:
  Ia]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: markpred:
    pa
  [barepred:
    [sentpred: senpred4:
      [senpred3: predunit:
```

```
mutce]
[despredD: predunit:
  tidjo]]
[termset: argument:
  tu]]]]
```

Parser cache size 11091

Eo tu fa helba mi

```
[freemod:
  Eo]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    fa
  [barepred:
    [sentpred: predunit:
      helba]
    [termset: argument:
      mi]]]]]
```

Parser cache size 10825

Ai mi fa helba tu

```
[freemod:
  Ai]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: markpred:
    fa
  [barepred:
    [sentpred: predunit:
```

```
        helba]
      [termset: argument:
        tu]]]
```

Parser cache size 10419

Mi groda tu

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      groda]
    [termset: argument:
      tu]]]
```

Parser cache size 5699

Tu nahu groda mi

```
[uttF: statement:
  [subject:
    [argument: arg1a:
      Tu]
    [modifiersx: modifier:
      na
      [argument1: arg1a:
        hu]]]
  [predicate: barepred:
    [sentpred: predunit:
      groda]
    [termset: argument:
      mi]]]
```

Parser cache size 9552

Mi groda tu fa ti

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      groda]
    [termset: terms:
      [argumentA: argument:
        tu]
      [modifiersx: modifier:
        fa
        [argument1: arg1a:
          ti]]]]]
```

Parser cache size 10082

Mi penso toi fa

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      penso]
    [termset: terms:
      [argumentA: argument:
        toi]
      [modifiersx: modifier:
        fa]]]
```

Parser cache size 8541

Na gi mi penso da


```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Na]
    gi]
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: barepred:
      [sentpred: predunit:
        penso]
      [termset: argument:
        da]]]]
```

Parser cache size 11404

Na mi, penso da

```
[uttF: sentence:
  [modifiers: modifier:
    Na
    [argument1: arg1a:
      mi
      [freemod: comma:
        ,]]]
  [predicate: barepred:
    [sentpred: predunit:
      penso]
    [termset: argument:
      da]]]
```

Parser cache size 11421

Na mi penso da

```
[uttF: sentence:
  [modifiers: modifier:
    Na
    [argument1: arg1a:
      mi]]
  [predicate: barepred:
    [sentpred: predunit:
      penso]
    [termset: argument:
      da]]]
```

Parser cache size 7938

Na gi la Djan, totco tu

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Na]
    gi]
  [sentence: statement:
    [subject: argument:
      la
      [name:
        Djan, ]]
    [predicate: barepred:
      [sentpred: predunit:
        totco]
      [termset: argument:
        tu]]]]]
```

Parser cache size 17020

Na la Djan, totco tu

```
[uttF: sentence:
  [modifiers: modifier:
    Na
    [argument1: LANAME:
      la
      [name:
        Djan, ]]]
  [predicate: barepred:
    [sentpred: predunit:
      totco]
    [termset: argument:
      tu]]]
```

Parser cache size 13548

Mi na totco tu

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    na
    [barepred:
      [sentpred: predunit:
        totco]
      [termset: argument:
        tu]]]]]
```

Parser cache size 8774

Mi ji na gui, totco tu

```
[uttF: statement:
  [subject: argument:
    [arg1a:
      Mi]]]
```

```

[argmod: argmod1:
  ji
  [modifier: mod1:
    na]
  [gui:
    gui
    [freemod: comma:
      ,]]]
[predicate: barepred:
  [sentpred: predunit:
    totco]
  [termset: argument:
    tu]]

```

Parser cache size 16653

Mi na fundi tu ji pa

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    na
    [barepred:
      [sentpred: predunit:
        fundi]
      [termset: argument:
        [arg1a:
          tu]
        [argmod: argmod1:
          ji
          [modifier: mod1:
            pa]]]]]]]

```

Parser cache size 13392

Mi no fundi tu pa

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      no
      [predunit1:
        fundi]]
    [termset: terms:
      [argumentA: argument:
        tu]
      [modifiersx: modifier:
        pa]]]]]
```

Parser cache size 10424

Ei tu na saadja toi

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    na
    [barepred:
      [sentpred: predunit:
        saadja]
      [termset: argument:
        toi]]]]]
```

Parser cache size 12550

Ia mi dui

```
[freemod:
  Ia]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: predunit:
    dui]]
```

Parser cache size 7020

No mi dui

```
[uttF: statement:
  [subject: argument:
    No
  [argument1: arg1a:
    mi]]
  [predicate: predunit:
    dui]]
```

Parser cache size 7531

Ai mi ditca

```
[freemod:
  Ai]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: predunit:
    ditca]]
```

Parser cache size 5139

Nahu

```
[uttF: modifier:
      Na
      [argument1: arg1a:
        hu]]
```

Parser cache size 4920

Fa la Netomen

```
[uttF: modifier:
      Fa
      [argument1: LANAME:
        la
        [name:
          Netomen]]]
```

Parser cache size 10022

Pa la Torin

```
[uttF: modifier:
      Pa
      [argument1: LANAME:
        la
        [name:
          Torin]]]
```

Parser cache size 8967

Na le monza

```
[uttF: modifier:
      Na
```

```
[argument1: descriptn:  
  le  
  [descpred: predunit:  
    monza]]]
```

Parser cache size 6502

Pahu

```
[uttF: modifier:  
  Pa  
  [argument1: arg1a:  
    hu]]]
```

Parser cache size 4920

Pa le natli

```
[uttF: modifier:  
  Pa  
  [argument1: descriptn:  
    le  
    [descpred: predunit:  
      natli]]]
```

Parser cache size 6517

Fahu

```
[uttF: modifier:  
  Fa  
  [argument1: arg1a:  
    hu]]]
```

Parser cache size 4921

Fa le cimra

```
[uttF: modifier:  
      Fa  
      [argument1: descriptn:  
        le  
        [descpred: predunit:  
          cimra]]]
```

Parser cache size 6516

Fazi

```
[uttF: modifier:  
      Fazi]
```

Parser cache size 4110

Fa la Nevesonin

```
[uttF: modifier:  
      Fa  
      [argument1: LANAME:  
        la  
        [name:  
          Nevesonin]]]
```

Parser cache size 11071

Na tio gi mi fa clafo

```
[uttF: uttC:  
      [headterms:
```

```
[terms: modifier:
      Na
      [argument1: arg1a:
        tio]]
gi]
[sentence: statement:
  [subject: argument:
    mi]
  [predicate: markpred:
    fa
    [barepred: predunit:
      clafo]]]]
```

Parser cache size 14337

Fa ta gi ti rodja

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Fa
      [argument1: arg1a:
        ta]]
    gi]
  [sentence: statement:
    [subject: argument:
      ti]
    [predicate: predunit:
      rodja]]]
```

Parser cache size 11477

Fa ta, rodja

```
[uttF: sentence:
  [modifiers: modifier:
```

```
Fa
  [argument1: arg1a:
    ta
    [freemod: comma:
      ,]]]
[predicate: predunit:
  rodja]]
```

Parser cache size 8812

Fa gi ti rodja

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Fa]
    gi]
  [sentence: statement:
    [subject: argument:
      ti]
    [predicate: predunit:
      rodja]]]
```

Parser cache size 9013

Ti fa rodja

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: markpred:
    fa
    [barepred: predunit:
      rodja]]]
```

Parser cache size 6353

Mi kicmu pazu tu

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      kicmu]
    [termset: modifier:
      pazu
      [argument1: arg1a:
        tu]]]]
```

Parser cache size 9005

Ai mi fa traci na la Nenimen

```
[freemod:
  Ai]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: markpred:
    fa
    [barepred:
      [sentpred: predunit:
        traci]
      [termset: modifier:
        na
        [argument1: LANAME:
          la
          [name:
            Nenimen]]]]]]]
```

Parser cache size 16729

Na la Nenimen, gi mi fa traci la Frans

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Na
      [argument1: LANAME:
        la
        [name:
          Nenimen, ]]]
    gi]
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: markpred:
      fa
      [barepred:
        [sentpred: predunit:
          traci]
        [termset: argument:
          la
          [name:
            Frans]]]]]]]
```

Parser cache size 24275

Mi fa traci da pa la Netomen

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    fa
    [barepred:
      [sentpred: predunit:
```

```

traci]
[termset: terms:
  [argumentA: argument:
    da]
  [modifiersx: modifier:
    pa
    [argument1: LANAME:
      la
      [name:
        Netomen]]]]]]]]

```

Parser cache size 17365

Mi fa traci pa la Netomen, da

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    fa
    [barepred:
      [sentpred: predunit:
        traci]
      [termset: terms:
        [modifiersx: modifier:
          pa
          [argument1: LANAME:
            la
            [name:
              Netomen, ]]]]
      [argumentA: argument:
        da]]]]]]

```

Parser cache size 17891

La Djan, titci vi

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: barepred:
    [sentpred: predunit:
      titci]
    [termset: modifier:
      vi]]]
```

Parser cache size 11678

Ei tu stolo vi

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      stolo]
    [termset: modifier:
      vi]]]
```

Parser cache size 7433

No. I mi stolo va

```
[uttF:
  [uttE: uttC:
    No
    [period:
      .]]]
```

```
I
[uttF: statement:
  [subject: argument:
    mi]
  [predicate: barepred:
    [sentpred: predunit:
      stolo]
    [termset: modifier:
      va]]]]
```

Parser cache size 9968

Ei tu stude vi

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      stude]
    [termset: modifier:
      vi]]]
```

Parser cache size 7433

No. I mi stude vu

```
[uttF:
  [uttE: uttC:
    No
    [period:
      .]]
  I
  [uttF: statement:
```



```
[subject: argument:
  mi]
[predicate: barepred:
  [sentpred: predunit:
    stude]
  [termset: modifier:
    vu]]]]
```

Parser cache size 9974

Ei tu ji vi, stude

```
[freemod:
  Ei]
[utterance: argument:
  [arg1a:
    tu]
  [argmod: argmod1:
    ji
    [predicate: markpred:
      vi
      [freemod: comma:
        ,]
      [barepred: predunit:
        stude]]]]]
```

Parser cache size 13301

Vihu tu sonli

```
[uttF: statement:
  [subject:
    [modifiers: modifier:
      Vi
      [argument1: arg1a:
        hu]]]
```

```
[argument: arg1a:
      tu]]
[predicate: predunit:
      sonli]]
```

Parser cache size 7848

Va tu

```
[uttF: modifier:
      Va
      [argument1: arg1a:
            tu]]
```

Parser cache size 5543

Vi le hasfa

```
[uttF: modifier:
      Vi
      [argument1: descriptn:
            le
            [descpred: predunit:
                  hasfa]]]
```

Parser cache size 6503

Vahu

```
[uttF: modifier:
      Va
      [argument1: arg1a:
            hu]]
```

Parser cache size 4896

Va la Rain

```
[uttF: modifier:  
  Va  
  [argument1: LANAME:  
    la  
    [name:  
      Rain]]]
```

Parser cache size 8041

Vuhu

```
[uttF: modifier:  
  Vu  
  [argument1: arg1a:  
    hu]]
```

Parser cache size 4903

Vu le mursi

```
[uttF: modifier:  
  Vu  
  [argument1: descriptn:  
    le  
    [descpred: predunit:  
      mursi]]]
```

Parser cache size 6497

Vi la Ditroit, gi da pa malbi

```

[uttF: uttC:
  [headterms:
    [terms: modifier:
      Vi
      [argument1: LANAME:
        la
        [name:
          Ditroit, ]]]
    gi]
  [sentence: statement:
    [subject: argument:
      da]
    [predicate: markpred:
      pa
      [barepred: predunit:
        malbi]]]]]

```

Parser cache size 18950

Da pa zbuma va le vrici

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        zbuma]
      [termset: modifier:
        va
        [argument1: descriptn:
          le
          [descpred: predunit:
            vrici]]]]]]]

```

Parser cache size 11431

Da na genbalci de vu le monca

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    na
  [barepred:
    [sentpred: predunit:
      genbalci]
    [termset: terms:
      [argumentA: argument:
        de]
      [modifiersx: modifier:
        vu
        [argument1: descriptn:
          le
          [descpred: predunit:
            monca]]]]]]]]]
```

Parser cache size 17158

Da na genbalci de ji vu le monca

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    na
  [barepred:
    [sentpred: predunit:
      genbalci]
    [termset: argument:
      [arg1a:
        de]
```

```
[argmod: argmod1:
  ji
  [modifier: mod1:
    vu
    [argument1: descriptn:
      le
      [descpred: predunit:
        monca]]]]]]]]
```

Parser cache size 19714

Ba najda

```
[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: predunit:
    najda]]
```

Parser cache size 3471

Ei ba smano

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    ba]
  [predicate: predunit:
    smano]]
```

Parser cache size 4880

Ia ba smano

```
[freemod:
  Ia]
[utterance: statement:
  [subject: argument:
    ba]
  [predicate: predunit:
    smano]]
```

Parser cache size 4876

Ei ba breba

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    ba]
  [predicate: predunit:
    breba]]
```

Parser cache size 4827

Ia. I ba breba vi

```
[freemod:
  Ia]
[period:
  .]
[utterance:
  I
  [uttF: statement:
    [subject: argument:
      ba]
    [predicate: barepred:
      [sentpred: predunit:
        breba]]
```

```
[termset: modifier:  
vi]]]]
```

Parser cache size 10094

Ei ba bukcu vi

```
[freemod:  
  Ei]  
[utterance: statement:  
  [subject: argument:  
    ba]  
  [predicate: barepred:  
    [sentpred: predunit:  
      bukcu]  
    [termset: modifier:  
      vi]]]
```

Parser cache size 7598

Uu no ba bukcu vi

```
[freemod:  
  Uu]  
[utterance: statement:  
  [subject: argument:  
    no  
    [argument1: arg1a:  
      ba]]  
  [predicate: barepred:  
    [sentpred: predunit:  
      bukcu]  
    [termset: modifier:  
      vi]]]
```

Parser cache size 9696

No ba cutri vina

```
[uttF: statement:
  [subject: argument:
    No
    [argument1: arg1a:
      ba]]
  [predicate: barepred:
    [sentpred: predunit:
      cutri]
    [termset: modifier:
      vina]]]
```

Parser cache size 9216

Vihu ba najda

```
[uttF: statement:
  [subject:
    [modifiers: modifier:
      Vi
      [argument1: arg1a:
        hu]]
    [argument: arg1a:
      ba]]
  [predicate: predunit:
    najda]]]
```

Parser cache size 7807

Ba najda vi levi drara

```
[uttF: statement:
  [subject: argument:
```

```
Ba]
[predicate: barepred:
  [sentpred: predunit:
    najda]
  [termset: modifier:
    vi
    [argument1: descriptn:
      le
      vi
      [descpred: predunit:
        drara]]]]]
```

Parser cache size 10829

Nahu ba nilboi

```
[uttF: statement:
  [subject:
    [modifiers: modifier:
      Na
      [argument1: arg1a:
        hu]]
    [argument: arg1a:
      ba]]
  [predicate: predunit:
    nilboi]]]
```

Parser cache size 10118

Ba nilboi na la Nevevonin

```
[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: barepred:
    [sentpred: predunit:
```

```
nilboi]
[termset: modifier:
  na
  [argument1: LANAME:
    la
    [name:
      Nevevonin]]]]]
```

Parser cache size 15782

Ei raba cninu

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    ra
    [arg2: arg1a:
      ba]]
  [predicate: predunit:
    cninu]]
```

Parser cache size 6743

No. I ba no cninu

```
[uttF:
  [uttE: uttC:
    No
    [period:
      .]]
  I
  [uttF: statement:
    [subject: argument:
      ba]
    [predicate: predunit:
```

```
no
[predunit1:
  cninu]]]]
```

Parser cache size 9504

Ei raba cluva rabe

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    ra
    [arg2: arg1a:
      ba]]
  [predicate: barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      ra
      [arg2: arg1a:
        be]]]]]
```

Parser cache size 11052

No. I ba no cluva rabe

```
[uttF:
  [uttE: uttC:
    No
    [period:
      .]]
  I
  [uttF: statement:
    [subject: argument:
      ba]]]
```

```
[predicate: barepred:
  [sentpred: predunit:
    no
    [predunit1:
      cluva]]
  [termset: argument:
    ra
    [arg2: arg1a:
      be]]]]]
```

Parser cache size 13813

Ifeu ba no cluva be

```
I
[freemod:
  feu]
[uttF: statement:
  [subject: argument:
    ba]
  [predicate: barepred:
    [sentpred: predunit:
      no
      [predunit1:
        cluva]]
    [termset: argument:
      be]]]
```

Parser cache size 12543

Ei raba kunci be

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
```

```
ra
  [arg2: arg1a:
    ba]]
[predicate: barepred:
  [sentpred: predunit:
    kunci]
  [termset: argument:
    be]]]
```

Parser cache size 9413

No. I ba kunci ni be

```
[uttF:
  [uttE: uttC:
    No
    [period:
      .]]
  I
  [uttF: statement:
    [subject: argument:
      ba]
    [predicate: barepred:
      [sentpred: predunit:
        kunci]
      [termset: argument:
        ni
        [arg2: arg1a:
          be]]]]]
```

Parser cache size 12522

Raba gudbi be bo

```
[uttF: statement:
  [subject: argument:
```

```
Ra
  [arg2: arg1a:
    ba]]
[predicate: barepred:
  [sentpred: predunit:
    gudbi]
  [termset: terms:
    [argumentA: argument:
      be]
    [argumentB: argument:
      bo]]]]
```

Parser cache size 10080

Ba pasylinkui rabe bo

```
[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: barepred:
    [sentpred: predunit:
      pasylinkui]
    [termset: terms:
      [argumentA: argument:
        ra
        [arg2: arg1a:
          be]]
      [argumentB: argument:
        bo]]]]
```

Parser cache size 13571

Ba vegri be rabo

```
[uttF: statement:
  [subject: argument:
```

```
Ba]
[predicate: barepred:
  [sentpred: predunit:
    vegri]
  [termset: terms:
    [argumentA: argument:
      be]
    [argumentB: argument:
      ra
      [arg2: arg1a:
        bo]]]]]
```

Parser cache size 10080

La Djan, godzi la Frans

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: argument:
      la
      [name:
        Frans]]]]]
```

Parser cache size 14173

Inusoa, de godzi da ba

```
Inusoa
[freemod: comma:
  ,]
```



```
[uttF: statement:
  [subject: argument:
    de]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: terms:
      [argumentA: argument:
        da]
      [argumentB: argument:
        ba]]]]
```

Parser cache size 14307

Ti groda

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: predunit:
    groda]]
```

Parser cache size 3280

Inusoa, da groda ba be

```
Inusoa
[freemod: comma:
  ,]
[uttF: statement:
  [subject: argument:
    da]
  [predicate: barepred:
    [sentpred: predunit:
      groda]
    [termset: terms:
```

```
[argumentA: argument:
  ba]
[argumentB: argument:
  be]]]]
```

Parser cache size 14070

Ti bitsa

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: predunit:
    bitsa]]
```

Parser cache size 3522

Inusoa, da bitsa ba be

```
Inusoa
[freemod: comma:
  ,]
[uttF: statement:
  [subject: argument:
    da]
  [predicate: barepred:
    [sentpred: predunit:
      bitsa]
    [termset: terms:
      [argumentA: argument:
        ba]
      [argumentB: argument:
        be]]]]
```

Parser cache size 14312

Ti vedma

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: predunit:
    vedma]]
```

Parser cache size 3520

Inusoa, da vedma ba be bo

```
Inusoa
[freemod: comma:
  ,]
[uttF: statement:
  [subject: argument:
    da]
  [predicate: barepred:
    [sentpred: predunit:
      vedma]
    [termset: terms:
      [argumentA: argument:
        ba]
      [argumentB: argument:
        be]
      [argumentC: argument:
        bo]]]]
```

Parser cache size 16602

Ti racyketpi

```
[uttF: statement:
  [subject: argument:
```

```
Ti]
[predicate: predunit:
  racyketpi]]
```

Parser cache size 6785

Inusoa, da ketpi ba be bo bu

```
Inusoa
[freemod: comma:
  ,]
[uttF: statement:
  [subject: argument:
    da]
  [predicate: barepred:
    [sentpred: predunit:
      ketpi]
    [termset: terms:
      [argumentA: argument:
        ba]
      [argumentB: argument:
        be]
      [argumentC: argument:
        bo]
      [argumentD: argument:
        bu]]]]]
```

Parser cache size 18711

Mi no vizka

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: predunit:
    no
```

```
[predunit1:
  vizka]]]
```

Parser cache size 5609

Inusoa, mi no vizka raba rabe

```
Inusoa
[freemod: comma:
  ,]
[uttF: statement:
  [subject: argument:
    mi]
  [predicate: barepred:
    [sentpred: predunit:
      no
      [predunit1:
        vizka]]]
  [termset: terms:
    [argumentA: argument:
      ra
      [arg2: arg1a:
        ba]]]
    [argumentB: argument:
      ra
      [arg2: arg1a:
        be]]]]]
```

Parser cache size 20230

La Frans, grada gunti

```
[uttF: statement:
  [subject: argument:
    La
    [name:
```

```
Frans, ]]  
[predicate: sentpred:  
  [senpred3: predunit:  
    grada]  
  [despredD: predunit:  
    gunti]]]
```

Parser cache size 10015

Da he grada gunti

```
[uttF: statement:  
  [subject: argument:  
    Da]  
  [predicate: sentpred:  
    [senpred3: predunit:  
      he]  
    [despredD: predunit:  
      grada]  
    [despredD: predunit:  
      gunti]]]
```

Parser cache size 6270

Da bilca grada gunti

```
[uttF: statement:  
  [subject: argument:  
    Da]  
  [predicate: sentpred:  
    [senpred3: predunit:  
      bilca]  
    [despredD: predunit:  
      grada]  
    [despredD: predunit:  
      gunti]]]
```

Parser cache size 4637

La Spat, simba janto kangu

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Spat, ]]
  [predicate: sentpred:
    [senpred3: predunit:
      simba]
    [despredD: predunit:
      janto]
    [despredD: predunit:
      kangu]]]
```

Parser cache size 10678

Da he simba janto kangu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      he]
    [despredD: predunit:
      simba]
    [despredD: predunit:
      janto]
    [despredD: predunit:
      kangu]]]
```

Parser cache size 7219

Da frika simba janto kangu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      frika]
    [despredD: predunit:
      simba]
    [despredD: predunit:
      janto]
    [despredD: predunit:
      kangu]]]
```

Parser cache size 5342

Da cmalo ge janto kangu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      cmalo]
    [despredD: predunit:
      ge
      [descpred: despredE:
        [despredD: predunit:
          janto]
        [despredD: predunit:
          kangu]]]]]
```

Parser cache size 6717

Da cmalo janto ci kangu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      cmalo]
    [despredD: despredA:
      [predunit: predunit1:
        janto]
      ci
      [predunit: predunit1:
        kangu]]]]]
```

Parser cache size 6576

La Mmbelis, dorja cefli

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Mmbelis, ]]
  [predicate: sentpred:
    [senpred3: predunit:
      dorja]
    [despredD: predunit:
      cefli]]]
```

Parser cache size 10760

Da he dorja cefli

```
[uttF: statement:
```

```
[subject: argument:
  Da]
[predicate: sentpred:
  [senpred3: predunit:
    he]
  [despredD: predunit:
    dorja]
  [despredD: predunit:
    cefli]]]
```

Parser cache size 6512

Da he ge dorja cefli

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      he]
    [despredD: predunit:
      ge
      [descpred: despredE:
        [despredD: predunit:
          dorja]
        [despredD: predunit:
          cefli]]]]]]]
```

Parser cache size 8614

Da ckano ge dorja cefli

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
```

```
[senpred3: predunit:
  ckano]
[despredD: predunit:
  ge
  [descpred: despredE:
    [despredD: predunit:
      dorja]
    [despredD: predunit:
      cefli]]]]]
```

Parser cache size 6715

Da junti ge terla famvo bilra

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      junti]
    [despredD: predunit:
      ge
      [descpred: despredE:
        [despredD: predunit:
          terla]
        [despredD: predunit:
          famvo]
        [despredD: predunit:
          bilra]]]]]]]
```

Parser cache size 7676

Da kusti ge slano torkrilu ci dampa

```
[uttF: statement:
  [subject: argument:
```

```

    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      kusti]
    [despredD: predunit:
      ge
    [descpred: despredE:
      [despredD: predunit:
        slano]
      [despredD: despredA:
        [predunit: predunit1:
          torkrilu]
        ci
        [predunit: predunit1:
          dampa]]]]]]]

```

Parser cache size 12395

Da junti famva ce terla ci sadji

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      junti]
    [despredD:
      [despredB: predunit:
        famva]
      ce
      [despredB: despredA:
        [predunit: predunit1:
          terla]
        ci
        [predunit: predunit1:
          sadji]]]]]]]

```

Parser cache size 9849

Da junti ke famva ki terla ci sadji

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      junti]
    [despredD: kekpredunit:
      ke
      [predicate: predunit:
        famva]
      ki
      [predicate: sentpred:
        [predunit: predunit1:
          terla]
        ci
        [predunit: predunit1:
          sadji]]]]]
```

Parser cache size 12485

Da junti ce terla ci famva sadji

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3:
      [senpred2: predunit:
        junti]
      ce
      [despredB: despredA:
        [predunit: predunit1:
          sadji]]]]]
```

```
                terla]
                ci
                [predunit: predunit1:
                  famva]]]
        [despredD: predunit:
          sadji]]]
```

Parser cache size 9849

Da kubra kanpi ci tcaro rodlu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      kubra]
    [despredD: despredA:
      [predunit: predunit1:
        kanpi]
      ci
      [predunit: predunit1:
        tcaro]]]
    [despredD: predunit:
      rodlu]]]
```

Parser cache size 7324

La Djan, prano go kukra la Djek

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: sentpred:
```

```
[senpred4: predunit:
  prano]
go
[barepred:
  [sentpred: predunit:
    kukra]
  [termset: argument:
    la
    [name:
      Djek]]]]]
```

Parser cache size 16654

Da botsu go vedji groda

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred4: predunit:
      botsu]
    go
    [barepred: sentpred:
      [senpred3: predunit:
        vedji]
      [despredD: predunit:
        groda]]]]]
```

Parser cache size 6837

Da slano takna go staryfoa

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
```

```
[senpred4:
  [senpred3: predunit:
    slano]
  [despredD: predunit:
    takna]]
go
[barepred: predunit:
  staryfoa]]]
```

Parser cache size 8352

Da bilti cmalo ce nirli

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD:
      [despredB: predunit:
        cmalo]
      ce
      [despredB: predunit:
        nirli]]]]]
```

Parser cache size 6922

Da bilti ke cmalo ki nirli

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
```



```
[despredD: kekpredunit:
  ke
  [predicate: predunit:
    cmalo]
  ki
  [predicate: predunit:
    nirli]]]]
```

Parser cache size 9559

Da bilti ge cmalo nirli geu ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD: predunit:
      ge
      [descpred: despredE:
        [despredD: predunit:
          cmalo]
        [despredD: predunit:
          nirli]]]
    [geu:
      geu]]
  [despredD: predunit:
    ckela]]]
```

Parser cache size 9926

Da bilti cmalo ge nirli ckela

```
[uttF: statement:
  [subject: argument:
```

```

    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD: predunit:
      cmalo]
    [despredD: predunit:
      ge
      [descpred: despredE:
        [despredD: predunit:
          nirli]
        [despredD: predunit:
          ckela]]]]]

```

Parser cache size 7148

Da bilti cmalo nirli ci ckela

```

  [uttF: statement:
    [subject: argument:
      Da]
    [predicate: sentpred:
      [senpred3: predunit:
        bilti]
      [despredD: predunit:
        cmalo]
      [despredD: despredA:
        [predunit: predunit1:
          nirli]
        ci
        [predunit: predunit1:
          ckela]]]]]

```

Parser cache size 7007

Da bilti ge cmalo ge nirli ckela

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD: predunit:
      ge
      [descpred: despredE:
        [despredD: predunit:
          cmalo]
        [despredD: predunit:
          ge
          [descpred: despredE:
            [despredD: predunit:
              nirli]
            [despredD: predunit:
              ckela]]]]]]]]]]

```

Parser cache size 9244

Da bilti cmalo ci nirli ci ckela

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD: despredA:
      [predunit: predunit1:
        cmalo]
      ci
      [predunit: predunit1:
        nirli]
      ci

```

```
[predunit: predunit1:
          ckela]]]]
```

Parser cache size 8968

Da bilti ce cmalo nirli ckela

```
[uttF: statement:
      [subject: argument:
        Da]
      [predicate: sentpred:
        [senpred3:
          [senpred2: predunit:
            bilti]
          ce
          [despredB: predunit:
            cmalo]]
        [despredD: predunit:
          nirli]
        [despredD: predunit:
          ckela]]]
```

Parser cache size 7353

Da bilti cmalo ca nirli ckela

```
[uttF: statement:
      [subject: argument:
        Da]
      [predicate: sentpred:
        [senpred3: predunit:
          bilti]
        [despredD:
          [despredB: predunit:
            cmalo]
          ca
```

```
[despredB: predunit:
  nirli]]
[despredD: predunit:
  ckela]]]
```

Parser cache size 7346

Da bilti ka cmalo ki nirli ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD: kekpredunit:
      ka
      [predicate: predunit:
        cmalo]
      ki
      [predicate: sentpred:
        [senpred3: predunit:
          nirli]
        [despredD: predunit:
          ckela]]]]]]]
```

Parser cache size 9984

Da bilti cmalo nirli ce ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
```

```
[despredD: predunit:
  cmalo]
[despredD:
  [despredB: predunit:
    nirli]
  ce
  [despredB: predunit:
    ckela]]]
```

Parser cache size 7353

Da bilti ge cmalo ce nirli ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD: predunit:
      ge
      [descpred: despredE:
        [despredD:
          [despredB: predunit:
            cmalo]
          ce
          [despredB: predunit:
            nirli]]
        [despredD: predunit:
          ckela]]]]]]]
```

Parser cache size 9455

Da bilti ge cmalo nirli ce ckela

```
[uttF: statement:
```

```

[subject: argument:
  Da]
[predicate: sentpred:
  [senpred3: predunit:
    bilti]
  [despredD: predunit:
    ge
  [descpred: despredE:
    [despredD: predunit:
      cmalo]
    [despredD:
      [despredB: predunit:
        nirli]
      ce
      [despredB: predunit:
        ckela]]]]]]]

```

Parser cache size 9456

Da cui bilti cmalo ce nirli ckela

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: senpred2:
      cui
    [despredC:
      [despredB: predunit:
        bilti]
      [despredB: predunit:
        cmalo]]
    ce
    [despredB: predunit:
      nirli]]
  [despredD: predunit:
    ckela]]]

```

Parser cache size 10135

Da bilti ci cmalo ce nirli ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3:
      [senpred2: senpred1:
        [predunit: predunit1:
          bilti]
        ci
        [predunit: predunit1:
          cmalo]]
      ce
      [despredB: predunit:
        nirli]]
    [despredD: predunit:
      ckela]]]
```

Parser cache size 9313

Da bilti canoi cmalo ce nirli ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3:
      [senpred2: predunit:
        bilti]
      canoi
      [despredB: predunit:
        cmalo]]]
```



```
ce
[despredB: predunit:
  nirli]]
[despredD: predunit:
  ckela]]
```

Parser cache size 10388

Da bilti ce cmalo ge nirli ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3:
      [senpred2: predunit:
        bilti]
      ce
      [despredB: predunit:
        cmalo]]
    [despredD: predunit:
      ge
      [descpred: despredE:
        [despredD: predunit:
          nirli]
        [despredD: predunit:
          ckela]]]]]]
```

Parser cache size 9455

Da bilti ce cmalo nirli ci ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
```

```

[senpred3:
  [senpred2: predunit:
    bilti]
  ce
  [despredB: predunit:
    cmalo]]
[despredD: despredA:
  [predunit: predunit1:
    nirli]
  ci
  [predunit: predunit1:
    ckela]]]]

```

Parser cache size 9314

Da bilti cui cmalo nirli ce ckela

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD: despredB:
      cui
      [despredC:
        [despredB: predunit:
          cmalo]
        [despredB: predunit:
          nirli]]
      ce
      [despredB: predunit:
        ckela]]]]]

```

Parser cache size 9857

Da bilti cmalo ci nirli ce ckela

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD:
      [despredB: despredA:
        [predunit: predunit1:
          cmalo]
        ci
        [predunit: predunit1:
          nirli]]
      ce
      [despredB: predunit:
        ckela]]]]]

```

Parser cache size 9314

Da bilti cmalo ce ge nirli ckela

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD:
      [despredB: predunit:
        cmalo]
      ce
      [despredB: predunit:
        ge
        [descpred: despredE:
          [despredD: predunit:

```

```
        nirli]
    [despredD: predunit:
        ckela]]]]]]
```

Parser cache size 9362

Da bilti ce cmalo ca nirli ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3:
      [senpred2: predunit:
        bilti]
      ce
      [despredB: predunit:
        cmalo]
      ca
      [despredB: predunit:
        nirli]]
    [despredD: predunit:
      ckela]]]
```

Parser cache size 9653

Da bilti ce kanoi cmalo ki nirli ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred2: predunit:
      bilti]
    ce
    [despredB: kekpredunit:
```

```
kanoi
[predicate: predunit:
  cmalo]
ki
[predicate: sentpred:
  [senpred3: predunit:
    nirli]
  [despredD: predunit:
    ckela]]]]
```

Parser cache size 12954

Da bilti ce cmalo nirli ce ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3:
      [senpred2: predunit:
        bilti]
      ce
      [despredB: predunit:
        cmalo]]
    [despredD:
      [despredB: predunit:
        nirli]
      ce
      [despredB: predunit:
        ckela]]]]]
```

Parser cache size 9660

Da bilti cmalo ce nirli ca ckela

```
[uttF: statement:
```

```
[subject: argument:
  Da]
[predicate: sentpred:
  [senpred3: predunit:
    bilti]
  [despredD:
    [despredB: predunit:
      cmalo]
    ce
    [despredB: predunit:
      nirli]
    ca
    [despredB: predunit:
      ckela]]]]
```

Parser cache size 9653

Da bilti cmalo ce ka nirli ki ckela

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      bilti]
    [despredD:
      [despredB: predunit:
        cmalo]
      ce
      [despredB: kekpredunit:
        ka
        [predicate: predunit:
          nirli]
        ki
        [predicate: predunit:
          ckela]]]]]]
```

Parser cache size 12223

Da zbuma

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    zbuma]]
```

Parser cache size 3281

Da poi zbuma

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    poi
    [predunit3: predunit1:
      zbuma]]]
```

Parser cache size 5867

Da pa clado poi zbuma

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
    [barepred: sentpred:
      [senpred3: predunit:
        clado]
      [despredD: predunit:
```

```
poi
[predunit3: predunit1:
  zbuma]]]]]
```

Parser cache size 8835

Da poi mrenu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    poi
    [predunit3: predunit1:
      mrenu]]]
```

Parser cache size 5837

Da pa corta poi mrenu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
    [barepred: sentpred:
      [senpred3: predunit:
        corta]
      [despredD: predunit:
        poi
        [predunit3: predunit1:
          mrenu]]]]]
```

Parser cache size 9064

Da pa corta poi mrenu bukcu


```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
    [barepred: sentpred:
      [senpred3: predunit:
        corta]
      [despredD: predunit:
        poi
        [predunit3: predunit1:
          mrenu]]]
      [despredD: predunit:
        bukcu]]]]]

```

Parser cache size 9763

Da pa corta ge poi mrenu bukcu

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
    [barepred: sentpred:
      [senpred3: predunit:
        corta]
      [despredD: predunit:
        ge
        [descpred: despredE:
          [despredD: predunit:
            poi
            [predunit3: predunit1:
              mrenu]]]
          [despredD: predunit:

```

bukcu]]]]]]

Parser cache size 11808

Da po corta mrenu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    po
    [sentence: sentpred:
      [senpred3: predunit:
        corta]
      [despredD: predunit:
        mrenu]]]]]
```

Parser cache size 6523

Da po, corta mrenu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    po
    [freemod: comma:
      ,]
    [sentence: sentpred:
      [senpred3: predunit:
        corta]
      [despredD: predunit:
        mrenu]]]]]
```

Parser cache size 10003

Da pa mutce po corta mrenu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
  [barepred: sentpred:
    [senpred3: predunit:
      mutce]
    [despredD: predunit:
      po
    [sentence: sentpred:
      [senpred3: predunit:
        corta]
      [despredD: predunit:
        mrenu]]]]]]]
```

Parser cache size 9931

Da po mi corta la Djan

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    po
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: barepred:
      [sentpred: predunit:
        corta]
    [termset: argument:
      la
    [name:
```

Djan]]]]]]

Parser cache size 13296

Da po la Pit, mutce corta la Djan

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    po
    [sentence: statement:
      [subject: argument:
        la
        [name:
          Pit, ]]
      [predicate: barepred:
        [sentpred: senpred4:
          [senpred3: predunit:
            mutce]
          [despredD: predunit:
            corta]]
        [termset: argument:
          la
          [name:
            Djan]]]]]]]]
```

Parser cache size 19606

Da pu de fotli di

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    pu
```

```
[sentence: statement:
  [subject: argument:
    de]
  [predicate: barepred:
    [sentpred: predunit:
      fotli]
    [termset: argument:
      di]]]]]
```

Parser cache size 10912

Da pu de fotli

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    pu
    [sentence: statement:
      [subject: argument:
        de]
      [predicate: predunit:
        fotli]]]]]
```

Parser cache size 8359

Da zo de blanu di

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    zo
    [sentence: statement:
      [subject: argument:
        de]
```

```
[predicate: barepred:
  [sentpred: predunit:
    blanu]
  [termset: argument:
    di]]]]]
```

Parser cache size 10637

Da zo de blanu

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    zo
    [sentence: statement:
      [subject: argument:
        de]
      [predicate: predunit:
        blanu]]]]]
```

Parser cache size 8084

Ei do fundi lo malna

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    do]
  [predicate: barepred:
    [sentpred: predunit:
      fundi]
    [termset: argument:
      lo
      [descpred: predunit:
```

malna]]]]

Parser cache size 8594

Ei tu pa titci lo nikri

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    pa
  [barepred:
    [sentpred: predunit:
      titci]
    [termset: argument:
      lo
    [descpred: predunit:
      nikri]]]]]]
```

Parser cache size 11449

Ei tu fa janto lo simba

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    fa
  [barepred:
    [sentpred: predunit:
      janto]
    [termset: argument:
```

```
lo
[descpred: predunit:
  simba]]]]]
```

Parser cache size 11458

Ia mi fa janto da

```
[freemod:
  Ia]
[utterance: statement:
  [subject: argument:
    mi]
  [predicate: markpred:
    fa
  [barepred:
    [sentpred: predunit:
      janto]
    [termset: argument:
      da]]]]]
```

Parser cache size 10366

Ei lo nirda vi sucmi

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    lo
  [descpred: predunit:
    nirda]]
  [predicate: markpred:
    vi
  [barepred: predunit:
    sucmi]]]]]
```


Parser cache size 8946

Ei lo nirda vi, sucmi

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    lo
    [descpred: predunit:
      nirda]]
  [predicate: markpred:
    vi
    [freemod: comma:
      ,]
    [barepred: predunit:
      sucmi]]]
```

Parser cache size 12155

Ei lo nirda ji vi, sucmi

```
[freemod:
  Ei]
[utterance: argument:
  [arg1a: descriptn:
    lo
    [descpred: predunit:
      nirda]]
  [argmod: argmod1:
    ji
    [predicate: markpred:
      vi
      [freemod: comma:
        ,]
    ]]
```

```
[barepred: predunit:  
sucmi]]]]
```

Parser cache size 14539

Ei lovi nirda ga sucmi

```
[freemod:  
  Ei]  
[utterance: statement:  
  [subject: argument:  
    lo  
    vi  
    [descpred: predunit:  
      nirda]]  
  [predicate: markpred:  
    ga  
    [barepred: predunit:  
      sucmi]]]
```

Parser cache size 10571

No. I da fletu

```
[uttF:  
  [uttE: uttC:  
    No  
    [period:  
      .]]  
  I  
  [uttF: statement:  
    [subject: argument:  
      da]  
    [predicate: predunit:  
      fletu]]]
```

Parser cache size 7418

Leva sonda ga he

```
[uttF: statement:
  [subject: argument:
    Le
    va
    [descpred: predunit:
      sonda]]
  [predicate: markpred:
    ga
    [barepred: predunit:
      he]]]
```

Parser cache size 10382

Lopo zbuma

```
[uttF: argument:
  Lo
  po
  [sentence: predunit:
    zbuma]]
```

Parser cache size 5524

Ei tu pa hirti lepo zbuma

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
```

```
pa
[barepred:
  [sentpred: predunit:
    hirti]
  [termset: argument:
    le
    po
    [sentence: predunit:
      zbuma]]]]]
```

Parser cache size 12872

Ia mi pa hirti da. I lopo zbuma ga clado

```
[freemod:
  Ia]
[utterance: uttF:
  [uttE: uttD:
    [sentence: statement:
      [subject: argument:
        mi]
      [predicate: markpred:
        pa
        [barepred:
          [sentpred: predunit:
            hirti]
          [termset: argument:
            da]]]]]
    [period:
      .]]
  I
  [uttF: statement:
    [subject: argument:
      lo
      po
      [sentence: predunit:
        zbuma]]]
```

```
[predicate: markpred:
  ga
  [barepred: predunit:
    clado]]]]
```

Parser cache size 20090

Ei tu pa hirti le clado po zbuma

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        hirti]
      [termset: argument:
        le
        [descpred: despredE:
          [despredD: predunit:
            clado]
          [despredD: predunit:
            po
            [sentence: predunit:
              zbuma]]]]]]]]]]]
```

Parser cache size 14152

Ei tu hirti lo clado po zbuma

```
[freemod:
  Ei]
[utterance: statement:
```

```

[subject: argument:
  tu]
[predicate: barepred:
  [sentpred: predunit:
    hirti]
  [termset: argument:
    lo
  [descpred: despredE:
    [despredD: predunit:
      clado]
    [despredD: predunit:
      po
    [sentence: predunit:
      zbuma]]]]]]]

```

Parser cache size 11325

Ei tu fundi lopo sucmi

```

[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      fundi]
    [termset: argument:
      lo
      po
    [sentence: predunit:
      sucmi]]]]]

```

Parser cache size 10247

Ei lepo prano pa nardu

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    le
    po
    [sentence: predunit:
      prano]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      nardu]]]
```

Parser cache size 10620

Ei lopo nilboi ga terci

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    lo
    po
    [sentence: predunit:
      nilboi]]
  [predicate: markpred:
    ga
    [barepred: predunit:
      terci]]]
```

Parser cache size 12464

Ei lopo mormao lo simba ga nardu

```
[freemod:
```

```

    Ei]
  [utterance: statement:
    [subject: argument:
      lo
      po
      [sentence: barepred:
        [sentpred: predunit:
          mormao]
        [termset: argument:
          lo
          [descpred: predunit:
            simba]]]]]
    [predicate: markpred:
      ga
      [barepred: predunit:
        nardu]]]

```

Parser cache size 16119

Lo ficli ga spalii lo mursi

```

  [uttF: statement:
    [subject: argument:
      Lo
      [descpred: predunit:
        ficli]]
    [predicate: markpred:
      ga
      [barepred:
        [sentpred: predunit:
          spalii]
        [termset: argument:
          lo
          [descpred: predunit:
            mursi]]]]]

```

Parser cache size 11987

Lo tongu ga trime lopo takna

```
[uttF: statement:
  [subject: argument:
    Lo
    [descpred: predunit:
      tongu]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        trime]
      [termset: argument:
        lo
        po
        [sentence: predunit:
          takna]]]]]
```

Parser cache size 11748

Lo tcina ga hasfa lo tongu

```
[uttF: statement:
  [subject: argument:
    Lo
    [descpred: predunit:
      tcina]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        hasfa]
      [termset: argument:
        lo
        [descpred: predunit:
```

tongu]]]]]

Parser cache size 10126

Lopo takna ga proju lo sonda

```
[uttF: statement:
  [subject: argument:
    Lo
    po
    [sentence: predunit:
      takna]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        proju]
      [termset: argument:
        lo
        [descpred: predunit:
          sonda]]]]]]]
```

Parser cache size 12008

Lopo dirлу ga ckozu lopo kecri

```
[uttF: statement:
  [subject: argument:
    Lo
    po
    [sentence: predunit:
      dirлу]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
```

```
      ckozu]
[termset: argument:
  lo
  po
  [sentence: predunit:
    kecri]]]]]
```

Parser cache size 13575

Lepo kanpi sucmi pa valna

```
[uttF: statement:
  [subject: argument:
    Le
    po
    [sentence: sentpred:
      [senpred3: predunit:
        kanpi]
      [despredD: predunit:
        sucmi]]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      valna]]]
```

Parser cache size 9928

Le poi kanpi sucmi pa valna

```
[uttF: statement:
  [subject: argument:
    Le
    [descpred: despredE:
      [despredD: predunit:
        poi
        [predunit3: predunit1:]]]]]
```

```

                                kanpi]]
      [despredD: predunit:
        sucmi]]]
[predicate: markpred:
  pa
  [barepred: predunit:
    valna]]]
```

Parser cache size 10572

Lo, poi kanpi sucmi pa valna

```

[uttF: statement:
  [subject: argument:
    Lo
    [freemod: comma:
      ,]
    [descpred: despredE:
      [despredD: predunit:
        poi
        [predunit3: predunit1:
          kanpi]]
      [despredD: predunit:
        sucmi]]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      valna]]]
```

Parser cache size 11647

Ei tu djano lepo mi stude

```

[freemod:
  Ei]
[utterance: statement:
```

```
[subject: argument:
  tu]
[predicate: barepred:
  [sentpred: predunit:
    djano]
  [termset: argument:
    le
    po
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: predunit:
      stude]]]]]
```

Parser cache size 12087

Tu danza lepo jmite hu

```
[uttF: statement:
  [subject: argument:
    Tu]
  [predicate: barepred:
    [sentpred: predunit:
      danza]
    [termset: argument:
      le
      po
    [sentence: barepred:
      [sentpred: predunit:
        jmite]
      [termset: argument:
        hu]]]]]
```

Parser cache size 10717

Le nirli pa takna mi lopo da traci

```

[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      nirli]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        takna]
      [termset: terms:
        [argumentA: argument:
          mi]
        [argumentB: argument:
          lo
          po
          [sentence: statement:
            [subject: argument:
              da]
            [predicate: predunit:
              traci]]]]]]]]]]

```

Parser cache size 16832

Le farfu pa takna le detra lepo de nu fatru

```

[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      farfu]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        takna]

```

```

[termset: terms:
  [argumentA: argument:
    le
    [descpred: predunit:
      detra]]
  [argumentB: argument:
    le
    po
    [sentence: statement:
      [subject: argument:
        de]
      [predicate: predunit:
        nu
        fatru]]]]]]]]

```

Parser cache size 20703

Ei tu danza lepo mi tcihea tu

```

[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      danza]
  [termset: argument:
    le
    po
    [sentence: statement:
      [subject: argument:
        mi]
      [predicate: barepred:
        [sentpred: predunit:
          tcihea]
        [termset: argument:

```

tu]]]]]]

Parser cache size 16969

Mi krido lepo la Paris, garsitci la Frans

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      krido]
    [termset: argument:
      le
      po
      [sentence: statement:
        [subject: argument:
          la
          [name:
            Paris, ]]
        [predicate: barepred:
          [sentpred: predunit:
            garsitci]
          [termset: argument:
            la
            [name:
              Frans]]]]]]]]]]
```

Parser cache size 25120

Mi djacue lepo la Marz, redro vi levi curta

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
```



```

le
[descpred: predunit:
  surva]]
[argumentB: argument:
  le
  po
  [sentence: statement:
    [subject: argument:
      da]
    [predicate: barepred:
      [sentpred: predunit:
        pluci]
      [termset: argument:
        de]]]]]]]]]]

```

Parser cache size 22152

La Ruprrt, djacue lepo la Denvrr, bitsa le vrici lepo kamda

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Ruprrt, ]]
  [predicate: barepred:
    [sentpred: predunit:
      djacue]
    [termset: argument:
      le
      po
      [sentence: statement:
        [subject: argument:
          la
          [name:
            Denvrr, ]]
        [predicate: barepred:
          [sentpred: predunit:

```


Parser cache size 30560

Lepo le mrenu pa trati ga fekto

```
[uttF: statement:
  [subject: argument:
    Le
    po
    [sentence: statement:
      [subject: argument:
        le
        [descpred: predunit:
          mrenu]]
      [predicate: markpred:
        pa
        [barepred: predunit:
          trati]]]]
  [predicate: markpred:
    ga
    [barepred: predunit:
      fekto]]]
```

Parser cache size 14567

Lepo le botci pa fundi lopo sucmi ga gudcae

```
[uttF: statement:
  [subject: argument:
    Le
    po
    [sentence: statement:
      [subject: argument:
        le
        [descpred: predunit:
          botci]]
      [predicate: markpred:
        ]]]]
```

```

    pa
    [barepred:
      [sentpred: predunit:
        fundi]
      [termset: argument:
        lo
        po
        [sentence: predunit:
          sucmi]]]]]]
  [predicate: markpred:
    ga
    [barepred: predunit:
      gudcae]]]

```

Parser cache size 22753

Lepo lepo le matma pa kamla pa tradu pa fatru

```

[uttF: statement:
  [subject: argument:
    Le
    po
    [sentence: statement:
      [subject: argument:
        le
        po
        [sentence: statement:
          [subject: argument:
            le
            [descpred: predunit:
              matma]]
          [predicate: markpred:
            pa
            [barepred: predunit:
              kamla]]]]]
      [predicate: markpred:
        pa

```

```

                [barepred: predunit:
                  tradu]]]]
    [predicate: markpred:
      pa
      [barepred: predunit:
        fatru]]]

```

Parser cache size 23165

Mi pa viadja lepo lepo prano pa kukra

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        viadja]
      [termset: argument:
        le
        po
        [sentence: statement:
          [subject: argument:
            le
            po
            [sentence: predunit:
              prano]]
          [predicate: markpred:
            pa
            [barepred: predunit:
              kukra]]]]]]]]]

```

Parser cache size 21184

Mi jupni lepo lepo mi danza lepo helba la Meris, pa gudcae


```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      penso]
    [termset: argument:
      lo
      po
      [sentence: barepred:
        [sentpred: predunit:
          firpa]
        [termset: argument:
          lo
          po
          [sentence: predunit:
            crina]]]]]]]]

```

Parser cache size 13383

Mi pa rulkao lepo santi gu, lepo helba la Bab

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        rulkao]
      [termset: terms:
        [argumentA: argument:
          le
          po
          [sentence: barepred:

```



```

                                cluva]
                                [termset: guu:
                                  gu
                                  [freemod: comma:
                                    ,]]]]
[argumentB: argument:
  lo
  po
  [sentence: predunit:
    clivi]]]]]]

```

Parser cache size 19849

Lo humni ga cnida lopo nu cluva gu, lopo hapci clivi

```

[uttF: statement:
  [subject: argument:
    Lo
    [descpred: predunit:
      humni]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        cnida]
      [termset: terms:
        [argumentA: argument:
          lo
          po
          [sentence: barepred:
            [sentpred: predunit:
              nu
              cluva]
            [termset: guu:
              gu
              [freemod: comma:
                ,]]]]]]]]]]]]

```

```
[argumentB: argument:
  lo
  po
  [sentence: sentpred:
    [senpred3: predunit:
      hapci]
    [despredD: predunit:
      clivi]]]]]]]
```

Parser cache size 23033

Da pa djano lepo fa crina gu, pa lepo le neri drida fa felda

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        djano]
      [termset: terms:
        [argumentA: argument:
          le
          po
          [sentence: statement:
            fa
            [barepred:
              [sentpred: predunit:
                crina]
              [termset: guu:
                gu
                [freemod: comma:
                  ,]]]]]
          [modifiersx: modifier:
            pa
            [argument1: abstractn:
```


Parser cache size 13435

Ue, tu nu dupma

```
[freemod:
  Ue
  [freemod: comma:
    ,]]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: predunit:
    nu
    dupma]]
```

Parser cache size 8049

Ua tu no nu dupma

```
[freemod:
  Ua]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: predunit:
    no
    [predunit1:
      nu
      dupma]]]
```

Parser cache size 9574

Tu nu vegri de di

```
[uttF: statement:
```

```
[subject: argument:
  Tu]
[predicate: barepred:
  [sentpred: predunit:
    nu
    vegri]
  [termset: terms:
    [argumentA: argument:
      de]
    [argumentB: argument:
      di]]]]
```

Parser cache size 10611

Da gutra nu begco de

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: senpred4:
      [senpred3: predunit:
        gutra]
      [despredD: predunit:
        nu
        begco]]
    [termset: argument:
      de]]]
```

Parser cache size 8862

Uu tu nu fatru

```
[freemod:
  Uu]
[utterance: statement:
```



```
[subject: argument:
  tu]
[predicate: predunit:
  nu
  fatru]]
```

Parser cache size 7493

Eo no nu fatru mi

```
[freemod:
  Eo]
[utterance: barepred:
  [sentpred: predunit:
    no
    [predunit1:
      nu
      fatru]]]
[termset: argument:
  mi]]
```

Parser cache size 10403

Ui tu no nu fatru

```
[freemod:
  Ui]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: predunit:
    no
    [predunit1:
      nu
      fatru]]]]
```

Parser cache size 9576

Da nu gudbi de ba

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      nu
      gudbi]
    [termset: terms:
      [argumentA: argument:
        de]
      [argumentB: argument:
        ba]]]]
```

Parser cache size 10559

Di no nu gudbi raba rabe

```
[uttF: statement:
  [subject: argument:
    Di]
  [predicate: barepred:
    [sentpred: predunit:
      no
      [predunit1:
        nu
        gudbi]]
    [termset: terms:
      [argumentA: argument:
        ra
        [arg2: arg1a:
          ba]]
      [argumentB: argument:
```

```
ra
[arg2: arg1a:
      be]]]]]
```

Parser cache size 16461

No nu gudbi

```
[uttF: predunit:
      No
      [predunit1:
        nu
        gudbi]]
```

Parser cache size 5943

No ge kukra prano

```
[uttF: predunit:
      No
      [predunit1:
        ge
        [descpred: despredE:
          [despredD: predunit:
            kukra]
          [despredD: predunit:
            prano]]]]]
```

Parser cache size 6013

No, kukra prano

```
[uttF: sentpred:
      [senpred3: predunit:
        No
```

```
[freemod: comma:
    ,]
[predunit1:
    kukra]]
[despredD: predunit:
    prano]]
```

Parser cache size 7391

No kukra prano

```
[uttF: sentpred:
    [senpred3: predunit:
        No
        [predunit1:
            kukra]]
    [despredD: predunit:
        prano]]
```

Parser cache size 3949

Da fu ge briga stuci

```
[uttF: statement:
    [subject: argument:
        Da]
    [predicate: predunit:
        fu
        ge
        [despredE:
            [despredD: predunit:
                briga]
            [despredD: predunit:
                stuci]]]]
```

Parser cache size 7986

Da fu brigystuci

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    fu
    brigystuci]]
```

Parser cache size 8824

Ei da nu kamla de

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    da]
  [predicate: barepred:
    [sentpred: predunit:
      nu
      kamla]
    [termset: argument:
      de]]]
```

Parser cache size 9900

Da fu lerci de

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
```

```
fu
  lerci]
[termset: argument:
  de]]]
```

Parser cache size 8259

Ei da sanpa de di

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    da]
  [predicate: barepred:
    [sentpred: predunit:
      sanpa]
    [termset: terms:
      [argumentA: argument:
        de]
      [argumentB: argument:
        di]]]]]
```

Parser cache size 9840

Ia de nu sanpa da di

```
[freemod:
  Ia]
[utterance: statement:
  [subject: argument:
    de]
  [predicate: barepred:
    [sentpred: predunit:
      nu
      sanpa]
```

```
[termset: terms:
  [argumentA: argument:
    da]
  [argumentB: argument:
    di]]]
```

Parser cache size 12208

Ia di fu sanpa de da

```
[freemod:
  Ia]
[utterance: statement:
  [subject: argument:
    di]
  [predicate: barepred:
    [sentpred: predunit:
      fu
      sanpa]
    [termset: terms:
      [argumentA: argument:
        de]
      [argumentB: argument:
        da]]]]]
```

Parser cache size 12187

No fu sanpa mi ta

```
[uttF: barepred:
  [sentpred: predunit:
    No
    [predunit1:
      fu
      sanpa]]
  [termset: terms:
```

```
[argumentA: argument:
  mi]
[argumentB: argument:
  ta]]]
```

Parser cache size 10694

No nu takna

```
[uttF: predunit:
  No
  [predunit1:
    nu
    takna]]]
```

Parser cache size 5945

No fu takna

```
[uttF: predunit:
  No
  [predunit1:
    fu
    takna]]]
```

Parser cache size 5922

Da fu gudbi

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    fu
    gudbi]]]
```


Parser cache size 5834

Mi fu namci da

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      fu
      namci]
    [termset: argument:
      da]]]
```

Parser cache size 8263

Ta fu garti

```
[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: predunit:
    fu
    garti]]]
```

Parser cache size 5874

Nu garti mi ta

```
[uttF: barepred:
  [sentpred: predunit:
    Nu
    garti]
  [termset: terms:
    mi ta]]]
```

```
[argumentA: argument:
  mi]
[argumentB: argument:
  ta]]]
```

Parser cache size 8571

Uu no ba nu hompi vi

```
[freemod:
  Uu]
[utterance: statement:
  [subject: argument:
    no
    [argument1: arg1a:
      ba]]
  [predicate: barepred:
    [sentpred: predunit:
      nu
      hompi]
    [termset: modifier:
      vi]]]
```

Parser cache size 12073

Da nu vizka ba be

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      nu
      vizka]
    [termset: terms:
      [argumentA: argument:
```

```
        ba]
[argumentB: argument:
        be]]]]
```

Parser cache size 10550

De no nu vizka raba rabe

```
[uttF: statement:
  [subject: argument:
    De]
  [predicate: barepred:
    [sentpred: predunit:
      no
      [predunit1:
        nu
        vizka]]
  [termset: terms:
    [argumentA: argument:
      ra
      [arg2: arg1a:
        ba]]
    [argumentB: argument:
      ra
      [arg2: arg1a:
        be]]]]]]
```

Parser cache size 16458

Da no nu kukra rutma de ti raba

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: senpred4:
```

```

[senpred3: predunit:
  no
  [predunit1:
    nu
    kukra]]
[despredD: predunit:
  rutma]]
[termset: terms:
  [argumentA: argument:
    de]
  [argumentB: argument:
    ti]
  [argumentC: argument:
    ra
    [arg2: arg1a:
      ba]]]]]

```

Parser cache size 17595

Da no nu plizo raba lopo roadru

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      no
      [predunit1:
        nu
        plizo]]]
  [termset: terms:
    [argumentA: argument:
      ra
      [arg2: arg1a:
        ba]]
    [argumentB: argument:
      lo]

```

```
po
[sentence: predunit:
roadru]]]]]
```

Parser cache size 19422

Konduo le batpi

```
[uttF: barepred:
      [sentpred: predunit:
        Konduo]
      [termset: argument:
        le
        [descpred: predunit:
          batpi]]]
```

Parser cache size 7182

Ne. I to. I te. I fo. I foba batpi.

```
[uttF:
  [uttE: uttC:
    [uttA:
      Ne]
    [period:
      .]]
  I
  [uttF:
    [uttE: uttC:
      [uttA:
        to]
      [period:
        .]]
    I
    [uttF:
      [uttE: uttC:
```

```

        [uttA:
          te]
        [period:
          .]]
I
[uttF:
  [uttE: uttC:
    [uttA:
      fo]
    [period:
      .]]
I
[uttF: uttD:
  [sentence: statement:
    [subject: argument:
      fo
      [arg2: arg1a:
        ba]]
    [predicate: predunit:
      batpi]]
  [period:
    .]]]]]]

```

Parser cache size 24416

Bleka le fo batpi

```

[uttF: barepred:
  [sentpred: predunit:
    Bleka]
  [termset: argument:
    le
    fo
  [descpred: predunit:
    batpi]]]

```

Parser cache size 6916

Totco to le fo batpi

```
[uttF: barepred:
  [sentpred: predunit:
    Totco]
  [termset: argument:
    to
    [arg2: descriptn:
      le
      fo
      [descpred: predunit:
        batpi]]]]]
```

Parser cache size 9578

Kambei mi ne le to batpi

```
[uttF: barepred:
  [sentpred: predunit:
    Kambei]
  [termset: terms:
    [argumentA: argument:
      mi]
    [argumentB: argument:
      ne
      [arg2: descriptn:
        le
        to
        [descpred: predunit:
          batpi]]]]]
```

Parser cache size 14162

Su levi fe batpi ga veslo lo viski

```

[uttF: statement:
  [subject: argument:
    Su
    [arg2: descriptn:
      le
      vi
      fe
      [descpred: predunit:
        batpi]]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        veslo]
      [termset: argument:
        lo
        [descpred: predunit:
          viski]]]]]]

```

Parser cache size 17047

Iesu de dui

```

[uttF: statement:
  [subject: argument:
    Ie
    [arg5: arg3:
      su
      [arg2: arg1a:
        de]]]
  [predicate: predunit:
    dui]]

```

Parser cache size 9804

Levi to de dui


```
[uttF: statement:
  [subject: argument:
    Le
    vi
    to
    [arg1a:
      de]]
  [predicate: predunit:
    dui]]
```

Parser cache size 12102

Foba tugle levi tobme

```
[uttF: statement:
  [subject: argument:
    Fo
    [arg2: arg1a:
      ba]]
  [predicate: barepred:
    [sentpred: predunit:
      tugle]
    [termset: argument:
      le
      vi
      [descpred: predunit:
        tobme]]]]
```

Parser cache size 10907

Hova tugle leva tobme

```
[uttF: statement:
  Hova
```

```
[barepred:
  [sentpred: predunit:
    tugle]
  [termset: argument:
    le
    va
    [descpred: predunit:
      tobme]]]]
```

Parser cache size 10449

Leva tobme ga nu tugle soba

```
[uttF: statement:
  [subject: argument:
    Le
    va
    [descpred: predunit:
      tobme]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        nu
        tugle]
      [termset: argument:
        so
        [arg2: arg1a:
          ba]]]]]]
```

Parser cache size 15423

Mi nu tugle hoba

```
[uttF: statement:
  [subject: argument:
```

```
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      nu
      tuggle]
    [termset: argument:
      ho
      [arg2: arg1a:
        ba]]]]
```

Parser cache size 10137

Ne uu

```
  [uttF: uttC:
    Ne
    [freemod:
      uu]]
```

Parser cache size 5459

Ho mu nu tuggle toba

```
  [uttF: statement:
    [subject: argument:
      Ho
      [arg2: arg1a:
        mu]]
    [predicate: barepred:
      [sentpred: predunit:
        nu
        tuggle]
      [termset: argument:
        to
        [arg2: arg1a:
          ba]]]]
```

Parser cache size 12595

Ho le se tobme ga nu tuggle soba

```
[uttF: statement:
  [subject: argument:
    Ho
    [arg2: descriptn:
      le
      se
      [descpred: predunit:
        tobme]]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        nu
        tuggle]
      [termset: argument:
        so
        [arg2: arg1a:
          ba]]]]]
```

Parser cache size 18044

Iene da

```
[uttF: argument:
  Ie
  [arg5: arg3:
    ne
    [arg2: arg1a:
      da]]]
```

Parser cache size 7467

Ho le bekti ji vi levi kruma, ga nu tugle teba

```
[uttF: statement:
  [subject: argument:
    Ho
    [arg2:
      [arg1a: descriptn:
        le
        [descpred: predunit:
          bekti]]
      [argmod: argmod1:
        ji
        [modifier: mod1:
          vi
          [argument1: descriptn:
            le
            vi
            [descpred: predunit:
              kruma
              [freemod: comma:
                ,]]]]]]]]
    [predicate: markpred:
      ga
      [barepred:
        [sentpred: predunit:
          nu
          tugle]
        [termset: argument:
          te
          [arg2: arg1a:
            ba]]]]]]
```

Parser cache size 26340

Ho le bekti ji vi, nu tugle

```
[uttF: argument:
  Ho
  [arg2:
    [arg1a: descriptn:
      le
      [descpred: predunit:
        bekti]]
    [argmod: argmod1:
      ji
      [predicate: markpred:
        vi
        [freemod: comma:
          ,]
        [barepred: predunit:
          nu
          tuggle]]]]]]
```

Parser cache size 14826

Ho le bekti ji vi nu tuggle

```
[uttF: argument:
  Ho
  [arg2:
    [arg1a: descriptn:
      le
      [descpred: predunit:
        bekti]]
    [argmod: argmod1:
      ji
      [predicate: markpred:
        vi
        [barepred: predunit:
          nu
          tuggle]]]]]]
```

Parser cache size 14283

Ho le bekti vi levi kruma ga nu tugle foba

```
[uttF: statement:
  [subject:
    [argument: arg3:
      Ho
      [arg2: descriptn:
        le
        [descpred: predunit:
          bekti]]]
    [modifiersx: modifier:
      vi
      [argument1: descriptn:
        le
        vi
        [descpred: predunit:
          kruma]]]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        nu
        tugle]
      [termset: argument:
        fo
        [arg2: arg1a:
          ba]]]]]
```

Parser cache size 23144

Ne mrenu pa kamla

```
[uttF: statement:
  [subject: argument:
```

```
Ne
  [descpred: predunit:
    mrenu]]
[predicate: markpred:
  pa
  [barepred: predunit:
    kamla]]]
```

Parser cache size 7351

Su mrenu pa kamla

```
[uttF: statement:
  [subject: argument:
    Su
    [descpred: predunit:
      mrenu]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      kamla]]]
```

Parser cache size 7373

Ru mrenu pa kamla

```
[uttF: statement:
  [subject: argument:
    Ru
    [descpred: predunit:
      mrenu]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      kamla]]]
```


Parser cache size 7348

Le fo mrenu pa kamla

```
[uttF: statement:
  [subject: argument:
    Le
    fo
    [descpred: predunit:
      mrenu]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      kamla]]]
```

Parser cache size 9342

Te le fo mrenu pa kamla

```
[uttF: statement:
  [subject: argument:
    Te
    [arg2: descriptn:
      le
      fo
      [descpred: predunit:
        mrenu]]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      kamla]]]
```

Parser cache size 11802

Ru le fo mrenu pa kamla

```
[uttF: statement:
  [subject: argument:
    Ru
    [arg2: descriptn:
      le
      fo
      [descpred: predunit:
        mrenu]]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      kamla]]]
```

Parser cache size 11793

To le te le fo mrenu pa kamla

```
[uttF: statement:
  [subject: argument:
    To
    [arg2: descriptn:
      le
      te
      [arg1a: descriptn:
        le
        fo
        [descpred: predunit:
          mrenu]]]]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      kamla]]]
```

Parser cache size 16509

Le to le te le fo mrenu pa kamla

```

[uttF: statement:
  [subject: argument:
    Le
    to
    [arg1a: descriptn:
      le
      te
      [arg1a: descriptn:
        le
        fo
        [descpred: predunit:
          mrenu]]]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      kamla]]]

```

Parser cache size 18757

Ho le te mrenu

```

[uttF: argument:
  Ho
  [arg2: descriptn:
    le
    te
    [descpred: predunit:
      mrenu]]]

```

Parser cache size 8529

Le ho mrenu

```

[uttF: argument:

```

```
Le
  ho
  [descpred: predunit:
    mrenu]]
```

Parser cache size 6081

Te le ho mrenu

```
[uttF: argument:
  Te
  [arg2: descriptn:
    le
    ho
    [descpred: predunit:
      mrenu]]]
```

Parser cache size 8550

Iete le mrenu

```
[uttF: argument:
  Ie
  [arg5: arg3:
    te
    [arg2: descriptn:
      le
      [descpred: predunit:
        mrenu]]]]]
```

Parser cache size 8253

Ie le ho mrenu

```
[uttF: argument:
```

```
Ie
  [arg5: descriptn:
    le
    ho
  [descpred: predunit:
    mrenu]]]
```

Parser cache size 8411

Iesu le ho mrenu

```
[uttF: argument:
  Ie
  [arg5: arg3:
    su
  [arg2: descriptn:
    le
    ho
  [descpred: predunit:
    mrenu]]]]]
```

Parser cache size 10495

Ieho le ho mrenu

```
[uttF: argument:
  Ie
  [arg5: arg3:
    ho
  [arg2: descriptn:
    le
    ho
  [descpred: predunit:
    mrenu]]]]]
```

Parser cache size 10442

Se sorme pa mercea le se brudi

```
[uttF: statement:
  [subject: argument:
    Se
    [descpred: predunit:
      sorme]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        mercea]
      [termset: argument:
        le
        se
        [descpred: predunit:
          brudi]]]]]]
```

Parser cache size 15771

Se sorme pa mercea ne le se brudi

```
[uttF: statement:
  [subject: argument:
    Se
    [descpred: predunit:
      sorme]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        mercea]
      [termset: argument:
        ne
        [arg2: descriptn:
```

```
le
se
[descpred: predunit:
brudi]]]]]]
```

Parser cache size 18237

Ti dalri lio te

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: barepred:
    [sentpred: predunit:
      dalri]
    [termset: argument:
      lio
      te]]]
```

Parser cache size 9213

Ti dalri

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: predunit:
    dalri]]
```

Parser cache size 3520

Ta gramo lio tema

```
[uttF: statement:
  [subject: argument:
```

```
Ta]
[predicate: barepred:
  [sentpred: predunit:
    gramo]
  [termset: argument:
    lio
    tema]]]
```

Parser cache size 9959

Ti langa ta lio tomumeimei

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: barepred:
    [sentpred: predunit:
      langa]
    [termset: terms:
      [argumentA: argument:
        ta]
      [argumentB: argument:
        lio
        tomumeimei]]]]]
```

Parser cache size 14463

Ti skabubra lio nemumei

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: barepred:
    [sentpred: predunit:
      skabubra]
    [termset: argument:
      ]]]]
```


lio
nemuemei]]]

Parser cache size 13726

La Djan, pa donsu le botci le to dalri

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        donsu]
      [termset: terms:
        [argumentA: argument:
          le
          [descpred: predunit:
            botci]]
        [argumentB: argument:
          le
          to
          [descpred: predunit:
            dalri]]]]]]]
```

Parser cache size 18381

La Djan, pa penti lio totomuedai, le mrenu le torkrilu

```
[uttF: statement:
  [subject: argument:
    La
    [name:
```

```

    Djan, ]]
[predicate: markpred:
  pa
  [barepred:
    [sentpred: predunit:
      penti]
    [termset: terms:
      [argumentA: argument:
        lio
        [term: argument:
          totomuedai,
          [arg2: descriptn:
            le
            [descpred: predunit:
              mrenu]]]]]
      [argumentB: argument:
        le
        [descpred: predunit:
          torkrilu]]]]]]]]

```

Parser cache size 27762

Lepa ckemo pa sekmi lio to

```

[uttF: statement:
  [subject: argument:
    Le
    pa
    [descpred: predunit:
      ckemo]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        sekmi]
      [termset: argument:
        lio

```

to]]]]

Parser cache size 14867

Lepo da clivi pa nirne lio voto

```
[uttF: statement:
  [subject: argument:
    Le
    po
    [sentence: statement:
      [subject: argument:
        da]
      [predicate: predunit:
        clivi]]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        nirne]
      [termset: argument:
        lio
        voto]]]]
```

Parser cache size 18001

Ta pa miksekmi lio to

```
[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        miksekmi]
```

```
[termset: argument:
  lio
  to]]]]
```

Parser cache size 15098

Mi pa takna da ta

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        takna]
      [termset: terms:
        [argumentA: argument:
          da]
        [argumentB: argument:
          ta]]]]]]
```

Parser cache size 11083

Hu bi da

```
[uttF: statement:
  [subject: argument:
    Hu]
  [predicate: identpred:
    bi
    [termset: argument:
      da]]]
```

Parser cache size 7319

Da bi le fremi je le botci

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: identpred:
    bi
  [termset: argument:
    le
  [descpred: predunit:
    [predunit2: predunit1:
      fremi]
  [linkargs: jelink:
    je
  [term: argument:
    le
  [descpred: predunit:
    botci]]]]]]]]
```

Parser cache size 13124

Ie le botci

```
[uttF: argument:
  Ie
  [arg5: descriptn:
    le
  [descpred: predunit:
    botci]]]
```

Parser cache size 6420

Le botci ji le brudi je le merfua je le furvea

```
[uttF: argument:
```


hu]]]]

Parser cache size 10785

Le furvea je le hasfa jue la Djonz, jue lio nemoatomuedai

[uttF: argument:

Le

[descpred: predunit:

[predunit2: predunit1:
furvea]

[linkargs: linkargs1:

[jelink:

je

[term: argument:

le

[descpred: predunit:
hasfa]]]

[links: links1:

[juelink:

jue

[term: argument:

la

[name:

Djonz,]]]

[juelink:

jue

[term: argument:

lio

nemoatomuedai]]]]]]]

Parser cache size 32607

Ie la Djonz

[uttF: argument:

```
Ie
  [arg5: LANAME:
    la
    [name:
      Djonz]]]
```

Parser cache size 8191

La Djonz, ji le farfu je la Meris, ze la Selis

```
[uttF: argument:
  [arg1a: LANAME:
    La
    [name:
      Djonz, ]]
  [argmod: argmod1:
    ji
    [argument1: descriptn:
      le
      [descpred: predunit:
        [predunit2: predunit1:
          farfu]
        [linkargs: jelink:
          je
          [term: argument:
            [arg3: LANAME:
              la
              [name:
                Meris, ]]
            ze
            [arg3: LANAME:
              la
              [name:
                Selis]]]]]]]]]]]
```

Parser cache size 26975

La Djonz, ji le farfu je la Meris, e je la Selis

```
[uttF: argument:
  [arg1a: LANAME:
    La
    [name:
      Djonz, ]]
  [argmod: argmod1:
    ji
    [argument1: descriptn:
      le
      [descpred: predunit:
        [predunit2: predunit1:
          farfu]
        [linkargs:
          [linkargs1: jelink:
            je
            [term: argument:
              la
              [name:
                Meris, ]]]
          e
          [linkargs1: jelink:
            je
            [term: argument:
              la
              [name:
                Selis]]]]]]]]]]]
```

Parser cache size 28919

Dai bi le farfu sui je le matma je la Bab, jue la ditca

```
[uttF: statement:
  [subject: argument:
    Dai]
```



```
    sorme]
[linkargs: linkargs1:
  [jelink:
    je
    [term: argument:
      Pai]]
[links: juelink:
  jue
  [term: argument:
    hu]]]]]
```

Parser cache size 12647

Jue la Rabrrt, ze la Celis

```
[uttF: juelink:
  Jue
  [term: argument:
    [arg3: LANAME:
      la
      [name:
        Rabrrt, ]]
  ze
  [arg3: LANAME:
    la
    [name:
      Celis]]]]]
```

Parser cache size 16300

Inusoa, Dai bi le farfu je Mai jue la sorme je Pai jue Rai ze Cai

```
Inusoa
[freemod: comma:
  ,]
[uttF: statement:
```

```

[subject: argument:
  Dai]
[predicate: identpred:
  bi
  [termset: argument:
    le
    [descpred: predunit:
      [predunit2: predunit1:
        farfu]
      [linkargs: linkargs1:
        [jelink:
          je
          [term: argument:
            Mai]]
        [links: juelink:
          jue
          [term: argument:
            la
            [descpred: predunit:
              [predunit2: predunit1:
                some]
              [linkargs: linkargs1:
                [jelink:
                  je
                  [term: argument:
                    Pai]]
                [links: juelink:
                  jue
                  [term: argument:
                    [arg3: arg
                      Rai
                      ze
                      [arg3: arg
                        Cai

```

Parser cache size 38317

Inusoa, Dai bi le farfu je Mai jue la some je Pai jue Rai, e jue Cai

```

Inusoa
[freemod: comma:
  ,]
[uttF: statement:
  [subject: argument:
    Dai]
  [predicate: identpred:
    bi
    [termset: argument:
      le
      [descpred: predunit:
        [predunit2: predunit1:
          farfu]
        [linkargs: linkargs1:
          [jelink:
            je
            [term: argument:
              Mai]]]
          [links: juelink:
            jue
            [term: argument:
              la
              [descpred: predunit:
                [predunit2: predunit1:
                  sorme]
                [linkargs: linkargs1:
                  [jelink:
                    je
                    [term: argument:
                      Pai]]]
                  [links:
                    [links1: juelink:
                      jue
                      [term: argume
                        Rai
                        [freemo

```

e
[links1: juelink:
jue
[term: argume
Cai]]]

Parser cache size 40878

Dai bi le farfu je Mai jue la sorme je Pai jue Rai, e la Celis

```
[uttF: statement:
  [subject: argument:
    Dai]
  [predicate: identpred:
    bi
    [termset: argument:
      le
      [descpred: predunit:
        [predunit2: predunit1:
          farfu]
        [linkargs: linkargs1:
          [jelink:
            je
            [term: argument:
              Mai]]]
        [links: juelink:
          jue
          [term: argument:
            la
            [descpred: predunit:
              [predunit2: predunit1:
                sorme]
              [linkargs: linkargs1:
                [jelink:
                  je
                  [term: argument:
```



```

    Pai]]
[links: juelink:
    jue
[term: argument:
    [arg7: arg
        Rai
        [fre
            e
            [arg7: LAN
                la
                [nam
                    C

```

Parser cache size 34989

Dai bi le farfu je Mai jue la sorme je Pai jue Rai gu, e je Cai

```

[uttF: statement:
    [subject: argument:
        Dai]
    [predicate: identpred:
        bi
    [termset: argument:
        le
    [descpred: predunit:
        [predunit2: predunit1:
            farfu]
    [linkargs: linkargs1:
        [jelink:
            je
            [term: argument:
                Mai]]
    [links: juelink:
        jue
        [term: argument:
            la

```



```

[utterance: barepred:
  [sentpred: predunit:
    vedma]
  [termset: terms:
    [argumentA: argument:
      mi]
    [argumentB: argument:
      le
    [descpred: predunit:
      [predunit2: predunit1:
        ketpi]
      [linkargs: linkargs1:
        [jelink:
          je
          [term: argument:
            la
            [name:
              Paris, ]]]]
      [links: links1:
        [juelink:
          jue
          [term: argument:
            la
            [name:
              Romas, ]]]]
        [juelink:
          jue
          [term: argument:
            le
            [descpred: despredE:
              [despredD: predunit:
                kukra]
              [despredD: predunit:
                trena]]]]]
        [juelink:
          jue
          [term: argument:
            lio

```

Da po nigro ze no nigro

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    po
    [sentence: sentpred:
      [senpred2: predunit:
        nigro]
      ze
      [despredB: predunit:
        no
        [predunit1:
          nigro]]]]]]
```

Parser cache size 11028

Le mrenu jio pa merji na la Somen, haimro de jio pa merji na la Nenimen

```
[uttF: statement:
  [subject: argument:
    [arg1a: descriptn:
      Le
      [descpred: predunit:
        mrenu]]
    [argmod: argmod1:
      jio
      [sentence: statement:
        pa
        [barepred:
          [sentpred: predunit:
            merji]
          [termset: modifier:
            na
```



```
        [name:
          Franes, ]]
[argumentB: argument:
di]]]]]
```

Parser cache size 19537

La Meris, jao la Pit, ditca la Franes, di

```
[uttF: argument:
  [arg1a: LANAME:
    La
    [name:
      Meris, ]]
  [argmod: argmod1:
    jao
    [sentence: statement:
      [subject: argument:
        la
        [name:
          Pit, ]]
      [predicate: barepred:
        [sentpred: predunit:
          ditca]
        [termset: terms:
          [argumentA: argument:
            la
            [name:
              Franes, ]]
          [argumentB: argument:
            di]]]]]]]
```

Parser cache size 27244

Raba jio katma ga titci be jio ratcu

```

[uttF: statement:
  [subject: argument:
    Ra
    [arg2:
      [arg1a:
        ba]
      [argmod: argmod1:
        jio
        [sentence: predunit:
          katma]]]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        titci]
      [termset: argument:
        [arg1a:
          be]
        [argmod: argmod1:
          jio
          [sentence: predunit:
            ratcu]]]]]]]

```

Parser cache size 17016

Raba jio redro nu herfa, e pa brana na la Nenemen Nevovoton, pa cenja be jio na

```

[uttF: statement:
  [subject: argument:
    Ra
    [arg2:
      [arg1a:
        ba]
      [argmod: argmod1:
        jio
        [sentence: predicate:
          [backpred: sentpred:

```

```

[senpred3: predunit:
  redro]
[despredD: predunit:
  nu
  herfa
  [freemod: comma:
    ,]]]
e
[backpred: markpred:
  pa
  [barepred:
    [sentpred: predunit:
      brana]
    [termset: modifier:
      na
      [argument1: LANAME:
        la
        [name:
          Nenemen
          Nevovoton, ]]]]]]]]
[predicate: markpred:
  pa
  [barepred:
    [sentpred: predunit:
      cenza]
    [termset: argument:
      [arg1a:
        be]
      [argmod: argmod1:
        jio
        [sentence: predunit:
          narmykoi]]]]]]]

```

Parser cache size 44501

La Djan, pa no gudbi prano

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [barepred: sentpred:
      [senpred3: predunit:
        no
        [predunit1:
          gudbi]]
      [despredD: predunit:
        prano]]]]]

```

Parser cache size 11619

La Djan, no pa gudbi prano

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: backpred1:
    [N02:
      no]
    [markpred:
      pa
      [barepred: sentpred:
        [senpred3: predunit:
          gudbi]
        [despredD: predunit:
          prano]]]]]

```

Parser cache size 11762

La Djan, no pa godzi la Paris

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: backpred1:
    [N02:
      no]
    [markpred:
      pa
    [barepred:
      [sentpred: predunit:
        godzi]
      [termset: argument:
        la
        [name:
          Paris]]]]]]]
```

Parser cache size 17131

La Djan, no pa titci le pligo

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: backpred1:
    [N02:
      no]
    [markpred:
      pa
    [barepred:
      [sentpred: predunit:
```

```
        titci]
    [termset: argument:
      le
      [descpred: predunit:
        pligo]]]]]]
```

Parser cache size 14534

La Djan, pa titci ni le pligo

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        titci]
      [termset: argument:
        ni
        [arg2: descriptn:
          le
          [descpred: predunit:
            pligo]]]]]]]]
```

Parser cache size 14902

La Djan, pa godzi ni la Paris

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
```

```
[predicate: markpred:
  pa
  [barepred:
    [sentpred: predunit:
      godzi]
    [termset: argument:
      ni
      [arg2: LANAME:
        la
        [name:
          Paris]]]]]]
```

Parser cache size 17495

Ni la Djan, pa godzi la Paris

```
[uttF: statement:
  [subject: argument:
    Ni
    [arg2: LANAME:
      la
      [name:
        Djan, ]]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        godzi]
      [termset: argument:
        la
        [name:
          Paris]]]]]]
```

Parser cache size 17483

La Djan, no pa kukra godzi la Paris


```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: backpred1:
    [N02:
      no]
    [markpred:
      pa
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          kukra]
        [despredD: predunit:
          godzi]]
      [termset: argument:
        la
        [name:
          Paris]]]]]]]]

```

Parser cache size 17830

La Djan, pa no kukra godzi la Paris

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          no

```

```
        [predunit1:
          kukra]]
      [despredD: predunit:
        godzi]]
    [termset: argument:
      la
      [name:
        Paris]]]]]]
```

Parser cache size 17687

La Djan, pa kukra no godzi la Paris

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          kukra]
        [despredD: predunit:
          no
          [predunit1:
            godzi]]]
      [termset: argument:
        la
        [name:
          Paris]]]]]]]]
```

Parser cache size 17800

Ni la Djan, pa kukra godzi la Paris, la Romas

```

[uttF: statement:
  [subject: argument:
    Ni
    [arg2: LANAME:
      la
      [name:
        Djan, ]]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          kukra]
        [despredD: predunit:
          godzi]]
      [termset: terms:
        [argumentA: argument:
          la
          [name:
            Paris, ]]
        [argumentB: argument:
          la
          [name:
            Romas]]]]]]]]

```

Parser cache size 23900

No, la Djan, pa kukra godzi la Paris, la Romas

```

[uttF: statement:
  [subject: argument:
    No
    [freemod: comma:
      ,]
    [argument1: LANAME:
      la

```

```

        [name:
          Djan, ]]
[predicate: markpred:
  pa
  [barepred:
    [sentpred: senpred4:
      [senpred3: predunit:
        kukra]
      [despredD: predunit:
        godzi]]
    [termset: terms:
      [argumentA: argument:
        la
        [name:
          Paris, ]]
      [argumentB: argument:
        la
        [name:
          Romas]]]]]]]

```

Parser cache size 23784

La Djan, no pa kukra godzi la Paris, la Romas

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: backpred1:
    [N02:
      no]
    [markpred:
      pa
      [barepred:
        [sentpred: senpred4:
          [senpred3: predunit:

```

```

        kukra]
    [despredD: predunit:
      godzi]]
[termset: terms:
  [argumentA: argument:
    la
    [name:
      Paris, ]]
  [argumentB: argument:
    la
    [name:
      Romas]]]]]]]]

```

Parser cache size 23548

La Djan, no ga kukra godzi la Paris, la Romas

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: backpred1:
    [N02:
      no]
    [markpred:
      ga
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          kukra]
        [despredD: predunit:
          godzi]]
      [termset: terms:
        [argumentA: argument:
          la
          [name:

```

```
Paris, ]]  
[argumentB: argument:  
  la  
  [name:  
    Romas]]]]]]]]
```

Parser cache size 23065

La Djan, no kukra godzi la Paris, la Romas

```
[uttF: statement:  
  [subject: argument:  
    La  
    [name:  
      Djan, ]]  
  [predicate: barepred:  
    [sentpred: senpred4:  
      [senpred3: predunit:  
        no  
        [predunit1:  
          kukra]]  
      [despredD: predunit:  
        godzi]]  
    [termset: terms:  
      [argumentA: argument:  
        la  
        [name:  
          Paris, ]]  
      [argumentB: argument:  
        la  
        [name:  
          Romas]]]]]]
```

Parser cache size 20742

La Djan, pa cutse li, Ai, lu la Tam

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        cutse]
      [termset: terms:
        [argumentA: argument:
          li
          ,
          [utterance0: freemod:
            Ai
            [freemod: comma:
              ,]]
          lu]
        [argumentB: argument:
          la
          [name:
            Tam]]]]]]]]

```

Parser cache size 21510

La Tam, gi la Djan, pa cutse liu Ai

```

[uttF: uttC:
  [headterms:
    [terms: argument:
      La
      [name:
        Tam, ]]
    gi]
  [sentence: statement:

```

```
[subject: argument:
  la
  [name:
    Djan, ]]
[predicate: markpred:
  pa
  [barepred:
    [sentpred: predunit:
      cutse]
    [termset: argument:
      liu Ai]]]]]
```

Parser cache size 21827

Liu Ai, la Tam, gi la Djan, pa cutse

```
[uttF: uttC:
  [headterms:
    [terms:
      [argumentA: argument:
        Liu Ai, ]
      [argumentB: argument:
        la
        [name:
          Tam, ]]]
    gi]
  [sentence: statement:
    [subject: argument:
      la
      [name:
        Djan, ]]
    [predicate: markpred:
      pa
      [barepred: predunit:
        cutse]]]]]
```

Parser cache size 21682

La Tam, gi la Djan, pa cutse

```
[uttF: uttC:
  [headterms:
    [terms: argument:
      La
      [name:
        Tam, ]]
    gi]
  [sentence: statement:
    [subject: argument:
      la
      [name:
        Djan, ]]
    [predicate: markpred:
      pa
      [barepred: predunit:
        cutse]]]]
```

Parser cache size 17654

La Frans, gi la Djan, pa takna

```
[uttF: uttC:
  [headterms:
    [terms: argument:
      La
      [name:
        Frans, ]]
    gi]
  [sentence: statement:
    [subject: argument:
      la
      [name:
        Djan, ]]]
```

```
[predicate: markpred:
  pa
  [barepred: predunit:
    takna]]]]
```

Parser cache size 18026

La Frans, gi la Djan, pa takna da

```
[uttF: uttC:
  [headterms:
    [terms: argument:
      La
      [name:
        Frans, ]]
    gi]
  [sentence: statement:
    [subject: argument:
      la
      [name:
        Djan, ]]
    [predicate: markpred:
      pa
      [barepred:
        [sentpred: predunit:
          takna]
        [termset: argument:
          da]]]]]
```

Parser cache size 20442

La Tam, la Frans, gi la Djan, pa takna

```
[uttF: uttC:
  [headterms:
    [terms:
```

```

        [argumentA: argument:
          La
          [name:
            Tam, ]]
        [argumentB: argument:
          la
          [name:
            Frans, ]]]
    gi]
[sentence: statement:
  [subject: argument:
    la
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      takna]]]]

```

Parser cache size 22693

Ti gi lo redro litla ga sanpa lo denro raba

```

[uttF: uttC:
  [headterms:
    [terms: argument:
      Ti]
    gi]
  [sentence: statement:
    [subject: argument:
      lo
      [descpred: despredE:
        [despredD: predunit:
          redro]
        [despredD: predunit:
          litla]]]
    [predicate: markpred:

```

```

ga
[barepred:
  [sentpred: predunit:
    sanpa]
  [termset: terms:
    [argumentA: argument:
      lo
      [descpred: predunit:
        denro]]
    [argumentB: argument:
      ra
      [arg2: arg1a:
        ba]]]]]]]]

```

Parser cache size 21003

Lopo stise gu, ti gi lo redro litra ga sanpa

```

[uttF: uttC:
  [headterms:
    [terms:
      [argumentA: argument:
        Lo
        po
        [sentence: barepred:
          [sentpred: predunit:
            stise]
          [termset: guu:
            gu
            [freemod: comma:
              ,]]]]
      [argumentB: argument:
        ti]]
    gi]
  [sentence: statement:
    [subject: argument:
      lo

```

```

[descpred: despredE:
  [despredD: predunit:
    redro]
  [despredD: predunit:
    litra]]]
[predicate: markpred:
  ga
  [barepred: predunit:
    sanpa]]]

```

Parser cache size 21156

Liu tceму, mela Djenis, rutma purda, e sackaa liu mutce

```

[uttF: statement:
  [subject: argument:
    Liu tceму, ]
  [predicate: predicate2:
    [backpred: sentpred:
      [senpred3: predunit:
        me
        [argument1: LANAME:
          la
          [name:
            Djenis, ]]]]
    [despredD: predunit:
      rutma]
    [despredD: predunit:
      purda
      [freemod: comma:
        ,]]]
  e
  [backpred: barepred:
    [sentpred: predunit:
      sackaa]
    [termset: argument:
      liu mutce]]]]]

```

Parser cache size 24906

Liu tcemu jao sackaa liu mutce, mela Djenis, rutma purda

```
[uttF: statement:
  [subject: argument:
    [arg1a: arg1:
      Liu tcemu]
    [argmod: argmod1:
      jao
      [sentence: barepred:
        [sentpred: predunit:
          sackaa]
        [termset: argument:
          liu mutce, ]]]]
  [predicate: sentpred:
    [senpred3: predunit:
      me
      [argument1: LANAME:
        la
        [name:
          Djenis, ]]]
    [despredD: predunit:
      rutma]
    [despredD: predunit:
      purda]]]
```

Parser cache size 25200

Ba pa meliu me forma holdu le lengu

```
[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: markpred:
```

```

pa
[barepred:
  [sentpred: senpred4:
    [senpred3: predunit:
      me
      [argument1: arg1:
        liu me]]
    [despredD: predunit:
      forma]
    [despredD: predunit:
      holdu]]
  [termset: argument:
    le
    [descpred: predunit:
      lengu]]]]]]

```

Parser cache size 16415

Ra ba ra bua goi, ko ba melo bua, ki ba bua

```

[uttF: uttC:
  [headterms:
    [terms:
      [argumentA: argument:
        Ra
        [arg2: arg1a:
          ba]]
      [argumentB: argument:
        ra
        [descpred: predunit:
          bua]]]
    goi]
  [freemod: comma:
    ,]
  [sentence: keksent:
    ko
    [sentence: statement:

```

```

[subject: argument:
  ba]
[predicate: predunit:
  me
  [argument1: descriptn:
    lo
    [descpred: predunit:
      bua
      [freemod: comma:
        ,]]]]]
ki
[uttA1: statement:
  [subject: argument:
    ba]
  [predicate: predunit:
    bua]]]

```

Parser cache size 35106

Ra ba ra bua goi, ko ba melea bua, ki ba bua

```

[uttF: uttC:
  [headterms:
    [terms:
      [argumentA: argument:
        Ra
        [arg2: arg1a:
          ba]]
      [argumentB: argument:
        ra
        [descpred: predunit:
          bua]]]
    goi]
  [freemod: comma:
    ,]
  [sentence: keksent:
    ko]

```



```

[sentence: statement:
  [subject: argument:
    ba]
  [predicate: predunit:
    me
    [argument1: descriptn:
      lea
      [descpred: predunit:
        bua
        [freemod: comma:
          ,]]]]]
ki
[uttA1: statement:
  [subject: argument:
    ba]
  [predicate: predunit:
    bua]]]

```

Parser cache size 35151

Ra ba ra bua goi, ko ba bie lea bua, ki ba bua

```

[uttF: uttC:
  [headterms:
    [terms:
      [argumentA: argument:
        Ra
        [arg2: arg1a:
          ba]]
      [argumentB: argument:
        ra
        [descpred: predunit:
          bua]]]
    goi]
  [freemod: comma:
    ,]
  [sentence: keksent:

```

```

ko
[sentence: statement:
  [subject: argument:
    ba]
  [predicate: identpred:
    bie
    [termset: argument:
      lea
      [descpred: predunit:
        bua
        [freemod: comma:
          ,]]]]]]

```

```

ki
[uttA1: statement:
  [subject: argument:
    ba]
  [predicate: predunit:
    bua]]]

```

Parser cache size 36646

Li, Da melo preda, lu durduo snola li, Da preda, lu

```

[uttF: statement:
  [subject: argument:
    Li
    ,
    [utterance0: statement:
      [subject: argument:
        Da]
      [predicate: predunit:
        me
        [argument1: descriptn:
          lo
          [descpred: predunit:
            preda
            [freemod: comma:
              ,]]]]]]]]

```

```

,]]]]]
lu]
[predicate: barepred:
 [sentpred: senpred4:
  [senpred3: predunit:
   durduo]
  [despredD: predunit:
   snola]]
 [termset: argument:
  li
 ,
 [utterance0: statement:
  [subject: argument:
   Da]
  [predicate: predunit:
   preda
  [freemod: comma:
   ,]]]
lu]]]

```

Parser cache size 25515

Ta meda po movdo

```

[uttF: statement:
 [subject: argument:
  Ta]
 [predicate: sentpred:
  [senpred3: predunit:
   me
  [argument1: arg1a:
   da]]
 [despredD: predunit:
  po
  [sentence: predunit:
   movdo]]]]]

```

Parser cache size 9939

Ta memi tcaro

```
[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: sentpred:
    [senpred3: predunit:
      me
      [argument1: arg1a:
        mi]]
    [despredD: predunit:
      tcaro]]]
```

Parser cache size 7259

Le kangu pa meli, Mi danza lepo hasfa godzi, lu bleka mi

```
[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      kangu]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          me
          [argument1: LI:
            li
            ,
            [utterance0: statement:
              [subject: argument:
                Mi]]]]]]]]]
```

```

[predicate: barepred:
  [sentpred: predunit:
    danza]
  [termset: argument:
    le
    po
  [sentence: sentpred:
    [senpred3: predunit:
      hasfa]
    [despredD: predunit:
      godzi
      [freemod: com
        ,]]]]]]

```

```

    lu]]
  [despredD: predunit:
    bleka]]
  [termset: argument:
    mi]]]]

```

Parser cache size 25696

Le kangu pa bleka je mi go meli, Mi danza lepo hasfa godzi, lu

```

[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      kangu]]
  [predicate: markpred:
    pa
    [barepred: sentpred:
      [senpred4: predunit:
        [predunit2: predunit1:
          bleka]
        [linkargs: jelinke:
          je
          [term: argument:

```

```

mi]]]
go
[barepred: predunit:
  me
  [argument1: LI:
    li
    ,
    [utterance0: statement:
      [subject: argument:
        Mi]
      [predicate: barepred:
        [sentpred: predunit:
          danza]
        [termset: argument:
          le
          po
          [sentence: sentpred:
            [senpred3: predunit:
              hasfa]
            [despredD: predunit:
              godzi
              [freemod: comma:
                ,]]]]]]
lu]]]]]

```

Parser cache size 29455

Raba be goi, be matmaa ba

```

[uttF: uttC:
  [headterms:
    [terms:
      [argumentA: argument:
        Ra
        [arg2: arg1a:
          ba]]
      [argumentB: argument:

```

```

        be]]
    goi]
  [freemod: comma:
    ,]
  [sentence: statement:
    [subject: argument:
      be]
    [predicate: barepred:
      [sentpred: predunit:
        matmaa]
      [termset: argument:
        ba]]]]

```

Parser cache size 18219

Be raba goi, be matmaa ba

```

  [uttF: uttC:
    [headterms:
      [terms:
        [argumentA: argument:
          Be]
        [argumentB: argument:
          ra
          [arg2: arg1a:
            ba]]]
      goi]
    [freemod: comma:
      ,]
    [sentence: statement:
      [subject: argument:
        be]
      [predicate: barepred:
        [sentpred: predunit:
          matmaa]
        [termset: argument:
          ba]]]]

```

Parser cache size 18219

Ba lea humni goi, ba pasylinkui da

```
[uttF: uttC:
  [headterms:
    [terms:
      [argumentA: argument:
        Ba]
      [argumentB: argument:
        lea
        [descpred: predunit:
          humni]]]
    goi]
  [freemod: comma:
    ,]
  [sentence: statement:
    [subject: argument:
      ba]
    [predicate: barepred:
      [sentpred: predunit:
        pasylinkui]
      [termset: argument:
        da]]]]
```

Parser cache size 19767

Ba rabe ji humni goi, ba pasylinkui da

```
[uttF: uttC:
  [headterms:
    [terms:
      [argumentA: argument:
        Ba]
      [argumentB: argument:
```



```

        ra
      [arg2:
        [arg1a:
          be]
        [argmod: argmod1:
          ji
          [predicate: predunit:
            humni]]]]]
    goi]
  [freemod: comma:
    ,]
  [sentence: statement:
    [subject: argument:
      ba]
    [predicate: barepred:
      [sentpred: predunit:
        pasylinkui]
      [termset: argument:
        da]]]]]

```

Parser cache size 22886

Raba goi cluva ba

```

  [uttF: uttC:
    [headterms:
      [terms: argument:
        Ra
        [arg2: arg1a:
          ba]]
      goi]
    [sentence: barepred:
      [sentpred: predunit:
        cluva]
      [termset: argument:
        ba]]]

```

Parser cache size 10966

Lea humni goi cluva da

```
[uttF: uttC:
  [headterms:
    [terms: argument:
      Lea
      [descpred: predunit:
        humni]]
    goi]
  [sentence: barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      da]]]
```

Parser cache size 10661

Raba ji humni goi cluva ba

```
[uttF: uttC:
  [headterms:
    [terms: argument:
      Ra
      [arg2:
        [arg1a:
          ba]
        [argmod: argmod1:
          ji
          [predicate: predunit:
            humni]]]]]
    goi]
  [sentence: barepred:
    [sentpred: predunit:
      cluva]
```

```
[termset: argument:
  ba]]
```

Parser cache size 13761

Raba ji humni go clivi goi cluva ba

```
[uttF: uttC:
  [headterms:
    [terms: argument:
      Ra
      [arg2:
        [arg1a:
          ba]
        [argmod: argmod1:
          ji
          [predicate: sentpred:
            [senpred4: predunit:
              humni]
            go
            [barepred: predunit:
              clivi]]]]]]
    goi]
  [sentence: barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      ba]]]
```

Parser cache size 16392

Raba ji clivi humni goi ba gi cluva

```
[uttF: uttC:
  [headterms:
    [terms: argument:
```

```

Ra
  [arg2:
    [arg1a:
      ba]
    [argmod: argmod1:
      ji
      [predicate: sentpred:
        [senpred3: predunit:
          clivi]
        [despredD: predunit:
          humni]]]]]]
  goi
  [terms: argument:
    ba]
  gi]
[sentence: predunit:
  cluva]]

```

Parser cache size 16387

La Djan, e la Meris, fundi la Bab

```

[uttF: statement:
  [subject: argument:
    [arg7: LANAME:
      La
      [name:
        Djan, ]]
    e
    [arg7: LANAME:
      la
      [name:
        Meris, ]]]
  [predicate: barepred:
    [sentpred: predunit:
      fundi]
    [termset: argument:

```

```
la
[name:
  Bab]]]]
```

Parser cache size 21449

La Djan, efafi la Meris, fundi la Bab

```
[uttF: statement:
  [subject: argument:
    [arg7: LANAME:
      La
      [name:
        Djan, ]]
    efafi
    [arg7: LANAME:
      la
      [name:
        Meris, ]]]
  [predicate: barepred:
    [sentpred: predunit:
      fundi]
    [termset: argument:
      la
      [name:
        Bab]]]]
```

Parser cache size 24600

Ke la Djan, ki la Meris, fundi la Bab

```
[uttF: statement:
  [subject: argument:
    Ke
    [argument1: LANAME:
      la
```

```

        [name:
          Djan, ]]
ki
[argx: LANAME:
  la
  [name:
    Meris, ]]]
[predicate: barepred:
  [sentpred: predunit:
    fundi]
  [termset: argument:
    la
    [name:
      Bab]]]]]

```

Parser cache size 24327

Di, e de ke fundi da ki tsodi la Pit

```

[uttF: statement:
  [subject: argument:
    [arg7: arg1a:
      Di
      [freemod: comma:
        ,]]]
    e
    [arg7: arg1a:
      de]]
  [predicate: barepred:
    ke
    [predicate: barepred:
      [sentpred: predunit:
        fundi]
      [termset: argument:
        da]]]
    ki
  [predicate: barepred:

```

```
[sentpred: predunit:
  tsodi]
[termset: argument:
  la
  [name:
    Pit]]]]
```

Parser cache size 20443

Di, e da fundi da, epafi tsodi la Pit

```
[uttF: statement:
  [subject: argument:
    [arg7: arg1a:
      Di
      [freemod: comma:
        ,]]
    e
    [arg7: arg1a:
      da]]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        fundi]
      [termset: argument:
        da
        [freemod: comma:
          ,]]]
    epafi
    [backpred: barepred:
      [sentpred: predunit:
        tsodi]
      [termset: argument:
        la
        [name:
          Pit]]]]]
```

Parser cache size 21386

Di, e da fundi da, e tsodi gu, la Pit

```
[uttF: statement:
  [subject: argument:
    [arg7: arg1a:
      Di
      [freemod: comma:
        ,]]
    e
    [arg7: arg1a:
      da]]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        fundi]
      [termset: argument:
        da
        [freemod: comma:
          ,]]]
    e
    [backpred: barepred:
      [sentpred: predunit:
        tsodi]
      [termset: guu:
        gu
        [freemod: comma:
          ,]]]
    [termset: argument:
      la
      [name:
        Pit]]]]]
```

Parser cache size 20865

Di, e de fundi, e tsodi gu, la Pit


```

[uttF: statement:
  [subject: argument:
    [arg7: arg1a:
      Di
      [freemod: comma:
        ,]]
    e
    [arg7: arg1a:
      de]]
  [predicate: predicate2:
    [backpred: predunit:
      fundi
      [freemod: comma:
        ,]]
    e
    [backpred: barepred:
      [sentpred: predunit:
        tsodi]
      [termset: guu:
        gu
        [freemod: comma:
          ,]]]
    [termset: argument:
      la
      [name:
        Pit]]]]]

```

Parser cache size 18367

La Pol, farfu la Bab, efafi la Pit

```

[uttF: statement:
  [subject: argument:
    La
    [name:

```

```

    Pol, ]]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: argument:
      [arg7: LANAME:
        la
        [name:
          Bab, ]]
      efafi
      [arg7: LANAME:
        la
        [name:
          Pit]]]]]]

```

Parser cache size 23565

La Pol, farfu la Bab, la Selis, onoi la Bet

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Pol, ]]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: terms:
      [argumentA: argument:
        la
        [name:
          Bab, ]]
      [argumentB: argument:
        [arg7: LANAME:
          la
          [name:
            Selis, ]]

```

```
onoi
[arg7: LANAME:
    la
    [name:
        Bet]]]]]]
```

Parser cache size 28470

La Pol, farfu la Bab, la Selis, guu, onoi la Bet

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Pol, ]]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: termset2:
      [termset1:
        [terms:
          [argumentA: argument:
            la
            [name:
              Bab, ]]
          [argumentB: argument:
            la
            [name:
              Selis, ]]]
        [guu:
          guu
          [freemod: comma:
            ,]]]
      onoi
      [termset1: argument:
        la
        [name:
```

Bet]]]]]

Parser cache size 32438

La Pol, jao farfu la Bab, e la Pit

```
[uttF: argument:
  [arg1a: LANAME:
    La
    [name:
      Pol, ]]
  [argmod: argmod1:
    jao
    [sentence: barepred:
      [sentpred: predunit:
        farfu]
      [termset: argument:
        [arg7: LANAME:
          la
          [name:
            Bab, ]]
        e
        [arg7: LANAME:
          la
          [name:
            Pit]]]]]]]
```

Parser cache size 20890

La Pol, jao farfu la Bab, gu, e la Pit

```
[uttF: argument:
  [arg1a: LANAME:
    La
    [name:
      Pol, ]]
```

```

[argmod: argmod1:
  jao
  [sentence: barepred:
    [sentpred: predunit:
      farfu]
    [termset: argument:
      [arg7: LANAME:
        la
        [name:
          Bab, ]
        [gap:
          gu
          [freemod: comma:
            ,]]]
      e
      [arg7: LANAME:
        la
        [name:
          Pit]]]]]]]

```

Parser cache size 23384

La Pol, jao farfu la Bab, gugu, e la Pit

```

[uttF: argument:
  [arg1a: LANAME:
    La
    [name:
      Pol, ]]
  [argmod: argmod1:
    jao
    [sentence: barepred:
      [sentpred: predunit:
        farfu]
      [termset: termset2:
        [termset1:
          [terms: argument:

```

```

        la
        [name:
          Bab, ]
        [gap:
          gu]]
    [guu:
      gu
      [freemod: comma:
        ,]]]
    e
    [termset1: argument:
      la
      [name:
        Pit]]]]]]

```

Parser cache size 25972

La Pol, jao farfu la Bab, gui, e la Pit

```

[uttF: argument:
  [arg7: arg2:
    [arg1a: LANAME:
      La
      [name:
        Pol, ]]
    [argmod: argmod1:
      jao
      [sentence: barepred:
        [sentpred: predunit:
          farfu]
        [termset: argument:
          la
          [name:
            Bab, ]]]]
      [gui:
        gui
        [freemod: comma:

```

,]]]]

e

```
[arg7: LANAME:
  la
  [name:
    Pit]]]
```

Parser cache size 24808

La Pol, farfu, e perdia la Bab, e la Pit

[uttF: statement:

[subject: argument:

```
La
  [name:
    Pol, ]]
```

[predicate: predicate2:

```
[backpred: predunit:
  farfu
  [freemod: comma:
    ,]]]
```

e

[backpred: barepred:

```
[sentpred: predunit:
  perdia]
```

[termset: argument:

```
[arg7: LANAME:
  la
  [name:
    Bab, ]]
```

e

[arg7: LANAME:

```
la
  [name:
    Pit]]]]]]]
```

Parser cache size 26334

La Pol, farfu, e perdia gu, la Bab, e la Pit

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Pol, ]]
  [predicate: predicate2:
    [backpred: predunit:
      farfu
      [freemod: comma:
        ,]]
    e
    [backpred: barepred:
      [sentpred: predunit:
        perdia]
      [termset: guu:
        gu
        [freemod: comma:
          ,]]]
    [termset: argument:
      [arg7: LANAME:
        la
        [name:
          Bab, ]]
      e
      [arg7: LANAME:
        la
        [name:
          Pit]]]]]
```

Parser cache size 28864

Da farfu ke la Bab, ki la Pit, la Meris


```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: terms:
      [argumentA: argument:
        ke
        [argument1: LANAME:
          la
          [name:
            Bab, ]]
        ki
        [argx: LANAME:
          la
          [name:
            Pit, ]]]
      [argumentB: argument:
        la
        [name:
          Meris]]]]]]

```

Parser cache size 23540

Da farfu kanoi la Bab, guu, ki la Pit, la Meris

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: termset1:
      kanoi
      [termset2: termset1:
        [terms: argument:

```

```

        la
        [name:
          Bab, ]]
    [guu:
      guu
      [freemod: comma:
        ,]]
ki
[termset1: terms:
  [argumentA: argument:
    la
    [name:
      Pit, ]]
  [argumentB: argument:
    la
    [name:
      Meris]]]]]]

```

Parser cache size 28407

Da farfu kanoi la Bab, ki la Pit, la Meris

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: terms:
      [argumentA: argument:
        kanoi
        [argument1: LANAME:
          la
          [name:
            Bab, ]]
        ki
        [argx: LANAME:

```

```

                                la
                                [name:
                                    Pit, ]]
[argumentB: argument:
    la
    [name:
        Meris]]]]]
```

Parser cache size 24271

Da farfu la Bab, guu, e la Pit, la Meris, a la Betis

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: termset2:
      [termset1:
        [terms: argument:
          la
          [name:
            Bab, ]]
        [guu:
          guu
          [freemod: comma:
            ,]]]
      e
      [termset1: terms:
        [argumentA: argument:
          la
          [name:
            Pit, ]]
        [argumentB: argument:
          [arg7: LANAME:
            la
```

```
                [name:
                  Meris, ]]
a
[arg7: LANAME:
  la
  [name:
    Betis]]]]]]]]
```

Parser cache size 31804

Da farfu la Bab, anoi la Pit, la Meris, a la Betis

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: terms:
      [argumentA: argument:
        [arg7: LANAME:
          la
          [name:
            Bab, ]]
        anoi
        [arg7: LANAME:
          la
          [name:
            Pit, ]]]
      [argumentB: argument:
        [arg7: LANAME:
          la
          [name:
            Meris, ]]
        a
        [arg7: LANAME:
          la
```

```
[name:
  Betis]]]]]]
```

Parser cache size 30178

Da farfu la Bab, anoi la Pit, la Meris, guu, a la Betis

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: termset2:
      [termset1:
        [terms:
          [argumentA: argument:
            [arg7: LANAME:
              la
              [name:
                Bab, ]]
            anoi
            [arg7: LANAME:
              la
              [name:
                Pit, ]]]
          [argumentB: argument:
            la
            [name:
              Meris, ]]]
        [guu:
          guu
          [freemod: comma:
            ,]]]
        a
        [termset1: argument:
          la
```

```
[name:  
  Betis]]]]]
```

Parser cache size 34146

Da farfu la Bab, e la Pit, ka la Meris, ki la Betis

```
[uttF: statement:  
  [subject: argument:  
    Da]  
  [predicate: barepred:  
    [sentpred: predunit:  
      farfu]  
    [termset: terms:  
      [argumentA: argument:  
        [arg7: LANAME:  
          la  
          [name:  
            Bab, ]]  
        e  
        [arg7: LANAME:  
          la  
          [name:  
            Pit, ]]]  
      [argumentB: argument:  
        ka  
        [argument1: LANAME:  
          la  
          [name:  
            Meris, ]]  
      ki  
      [argx: LANAME:  
        la  
        [name:  
          Betis]]]]]]]
```

Parser cache size 31200

Da farfu ke la Bab, ki la Pit, ka la Meris, ki la Betis

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: terms:
      [argumentA: argument:
        ke
        [argument1: LANAME:
          la
          [name:
            Bab, ]]
        ki
        [argx: LANAME:
          la
          [name:
            Pit, ]]]
      [argumentB: argument:
        ka
        [argument1: LANAME:
          la
          [name:
            Meris, ]]
        ki
        [argx: LANAME:
          la
          [name:
            Betis]]]]]]]
```

Parser cache size 34342

Da farfu la Bab, guu, e la Pit, la Meris, e lendia

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        farfu]
      [termset: termset2:
        [termset1:
          [terms: argument:
            la
            [name:
              Bab, ]]
          [guu:
            guu
            [freemod: comma:
              ,]]]
        e
        [termset1: terms:
          [argumentA: argument:
            la
            [name:
              Pit, ]]
          [argumentB: argument:
            la
            [name:
              Meris, ]]]]]
    e
    [backpred: predunit:
      lencia]]]

```

Parser cache size 30196

Da farfu la Bab, e la Bet, e ditca la Franses, di, e do

```

[uttF: statement:

```



```

[subject: argument:
  Da]
[predicate: predicate2:
  [backpred: barepred:
    [sentpred: predunit:
      farfu]
    [termset: argument:
      [arg7: LANAME:
        la
        [name:
          Bab, ]]
      e
      [arg7: LANAME:
        la
        [name:
          Bet, ]]]]
  e
  [backpred: barepred:
    [sentpred: predunit:
      ditca]
    [termset: terms:
      [argumentA: argument:
        la
        [name:
          Franses, ]]
      [argumentB: argument:
        [arg7: arg1a:
          di
          [freemod: comma:
            ,]]
        e
        [arg7: arg1a:
          do]]]]]]]

```

Parser cache size 31168

Da farfu, e ditca la Franses, guu, di, e do

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predicate2:
    [backpred: predunit:
      farfu
      [freemod: comma:
        ,]]
    e
    [backpred: barepred:
      [sentpred: predunit:
        ditca]
      [termset: termset1:
        [terms: argument:
          la
          [name:
            Franeses, ]]
        [guu:
          guu
          [freemod: comma:
            ,]]]]
    [termset: argument:
      [arg7: arg1a:
        di
        [freemod: comma:
          ,]]
      e
      [arg7: arg1a:
        do]]]]

```

Parser cache size 23867

Da farfu, e ditca la Franeses, di, e do

```

[uttF: statement:
  [subject: argument:

```

```

Da]
[predicate: predicate2:
  [backpred: predunit:
    farfu
    [freemod: comma:
      ,]]
e
[backpred: barepred:
  [sentpred: predunit:
    ditca]
  [termset: terms:
    [argumentA: argument:
      la
      [name:
        Franses, ]]
    [argumentB: argument:
      [arg7: arg1a:
        di
        [freemod: comma:
          ,]]
      e
      [arg7: arg1a:
        do]]]]]]]

```

Parser cache size 20015

Da farfu, e ditca la Franses, di guu, e do

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predicate2:
    [backpred: predunit:
      farfu
      [freemod: comma:
        ,]]
    e

```

```

[backpred: barepred:
  [sentpred: predunit:
    ditca]
  [termset: termset2:
    [termset1:
      [terms:
        [argumentA: argument:
          la
          [name:
            Franes, ]]
        [argumentB: argument:
          di]]
      [guu:
        guu
        [freemod: comma:
          ,]]]
    e
    [termset1: argument:
      do]]]]]

```

Parser cache size 22857

Da farfu ba be, noa ditca la Franes, ba, e be

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        farfu]
      [termset: terms:
        [argumentA: argument:
          ba]
        [argumentB: argument:
          be
          [freemod: comma:
            ,]]]]]]]

```

```

,]]]]
noa
[backpred: barepred:
  [sentpred: predunit:
    ditca]
  [termset: terms:
    [argumentA: argument:
      la
      [name:
        Franses, ]]
    [argumentB: argument:
      [arg7: arg1a:
        ba
        [freemod: comma:
          ,]]
      e
      [arg7: arg1a:
        be]]]]]]]

```

Parser cache size 26464

Da kanoi farfu ba be, ki ditca la Franses, ba, e be

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    kanoi
    [predicate: barepred:
      [sentpred: predunit:
        farfu]
      [termset: terms:
        [argumentA: argument:
          ba]
        [argumentB: argument:
          be
          [freemod: comma:
            ,]]]]]]]

```

```

,]]]]
ki
[predicate: barepred:
  [sentpred: predunit:
    ditca]
  [termset: terms:
    [argumentA: argument:
      la
      [name:
        Franses, ]]
    [argumentB: argument:
      [arg7: arg1a:
        ba
        [freemod: comma:
          ,]]
      e
      [arg7: arg1a:
        be]]]]]]]

```

Parser cache size 28232

Liu tceму kanoi sackaa liu mutce, ki me la Djenis, rutma purda

```

[uttF: statement:
  [subject: argument:
    Liu tceму]
  [predicate: barepred:
    kanoi
    [predicate: barepred:
      [sentpred: predunit:
        sackaa]
      [termset: argument:
        liu mutce, ]]
  ki
  [predicate: sentpred:
    [senpred3: predunit:
      me

```

```
[argument1: LANAME:
    la
    [name:
        Djenis, ]]]
[despredD: predunit:
    rutma]
[despredD: predunit:
    purda]]]]
```

Parser cache size 28477

Mi pa godzi moi

```
[uttF: statement:
    [subject: argument:
        Mi]
    [predicate: markpred:
        pa
        [barepred:
            [sentpred: predunit:
                godzi]
            [termset: modifier:
                moi]]]]]
```

Parser cache size 9076

Mi moipa godzi

```
[uttF: statement:
    [subject: argument:
        Mi]
    [predicate: markpred:
        moipa
        [barepred: predunit:
            godzi]]]
```

Parser cache size 7797

Mi ji moi, pa godzi

```
[uttF: argument:
  [arg1a:
    Mi]
  [argmod: argmod1:
    ji
    [predicate: markpred:
      moi, pa
      [barepred: predunit:
        godzi]]]]]
```

Parser cache size 10603

Mi ji moipa, godzi

```
[uttF: argument:
  [arg1a:
    Mi]
  [argmod: argmod1:
    ji
    [predicate: markpred:
      moipa
      [freemod: comma:
        ,]
      [barepred: predunit:
        godzi]]]]]
```

Parser cache size 13214

Moi mi, godzi

```
[uttF: sentence:
```



```
[modifiers: modifier:
  Moi
  [argument1: arg1a:
    mi
    [freemod: comma:
      ,]]]
[predicate: predunit:
  godzi]]
```

Parser cache size 9033

Moi gi mi godzi

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Moi]
    gi]
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: predunit:
      godzi]]]
```

Parser cache size 9215

Soa ta gi da bi de

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Soa
      [argument1: arg1a:
        ta]]
    gi]
  [sentence: statement:
```

```
[subject: argument:
  da]
[predicate: identpred:
  bi
  [termset: argument:
    de]]]]
```

Parser cache size 15977

Soa lepo da bi de gi, da tsidru

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Soa
      [argument1: abstractn:
        le
        po
        [sentence: statement:
          [subject: argument:
            da]
          [predicate: identpred:
            bi
            [termset: argument:
              de]]]]]]
    gi]
  [freemod: comma:
    ,]
  [sentence: statement:
    [subject: argument:
      da]
    [predicate: predunit:
      tsidru]]]]
```

Parser cache size 23153

Da tsidru soa lepo da bi de

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      tsidru]
    [termset: modifier:
      soa
      [argument1: abstractn:
        le
        po
        [sentence: statement:
          [subject: argument:
            da]
          [predicate: identpred:
            bi
            [termset: argument:
              de]]]]]]]]

```

Parser cache size 19113

Mi pa tokna ti moi ta

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        tokna]
      [termset: terms:
        [argumentA: argument:
          ti]
        [modifiersx: modifier:

```

```
moi
[argument1: arg1a:
  ta]]]]]]
```

Parser cache size 13415

Moi ta gi mi pa tokna ti

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Moi
      [argument1: arg1a:
        ta]]
    gi]
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: markpred:
      pa
      [barepred:
        [sentpred: predunit:
          tokna]
        [termset: argument:
          ti]]]]]]
```

Parser cache size 17085

Mi pa danza ta numoi ti

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pa
    [barepred:
```

```
[sentpred: predunit:
  danza]
[termset: terms:
  [argumentA: argument:
    ta]
  [modifiersx: modifier:
    numoi
    [argument1: arg1a:
      ti]]]]]]
```

Parser cache size 13908

Ti pa rodja kou lepo tu cuidru da

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        rodja]
      [termset: modifier:
        kou
        [argument1: abstractn:
          le
          po
          [sentence: statement:
            [subject: argument:
              tu]
            [predicate: barepred:
              [sentpred: predunit:
                cuidru]
              [termset: argument:
                da]]]]]]]]]]
```

Parser cache size 20874

Da pa rodja kou lepo cuidru

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        rodja]
      [termset: modifier:
        kou
        [argument1: abstractn:
          le
          po
          [sentence: predunit:
            cuidru]]]]]]]
```

Parser cache size 16128

Da pa rodja kou lo cutri

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        rodja]
      [termset: modifier:
        kou
        [argument1: descriptn:
          lo
          [descpred: predunit:
            cuidru]]]]]]]
```

cutri]]]]]]

Parser cache size 12414

Ti pa rodja kouhu

```
[uttF: statement:
  [subject: argument:
    Ti]
  [predicate: markpred:
    pa
  [barepred:
    [sentpred: predunit:
      rodja]
    [termset: modifier:
      kou
      [argument1: arg1a:
        hu]]]]]
```

Parser cache size 10881

Tu pa cuidra da moi

```
[uttF: statement:
  [subject: argument:
    Tu]
  [predicate: markpred:
    pa
  [barepred:
    [sentpred: predunit:
      cuidra]
    [termset: terms:
      [argumentA: argument:
        da]
      [modifiersx: modifier:
        moi]]]]]
```

```
[argument1: arg1a:  
  hu]]]]]]
```

Parser cache size 15083

Ti pa rodja nukouhu

```
[uttF: statement:  
  [subject: argument:  
    Ti]  
  [predicate: markpred:  
    pa  
    [barepred:  
      [sentpred: predunit:  
        rodja]  
      [termset: modifier:  
        nukou  
        [argument1: arg1a:  
          hu]]]]]]
```

Parser cache size 11919

Kouhu da pa rodja

```
[uttF: statement:  
  [subject:  
    [modifiers: modifier:  
      Kou  
      [argument1: arg1a:  
        hu]]  
    [argument: arg1a:  
      da]]  
  [predicate: markpred:  
    pa  
    [barepred: predunit:  
      rodja]]]
```


Parser cache size 11200

Nukouhu da pa rodja

```
[uttF: statement:
  [subject:
    [modifiers: modifier:
      Nukou
      [argument1: arg1a:
        hu]]
    [argument: arg1a:
      da]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      rodja]]]
```

Parser cache size 11689

Moihu tu pa cuidru da

```
[uttF: statement:
  [subject:
    [modifiers: modifier:
      Moi
      [argument1: arg1a:
        hu]]
    [argument: arg1a:
      tu]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        cuidru]
      [termset: argument:
```

da]]]]

Parser cache size 15581

Moihu tu pa danza lepo da rodja

```
[uttF: statement:
  [subject:
    [modifiers: modifier:
      Moi
      [argument1: arg1a:
        hu]]
    [argument: arg1a:
      tu]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        danza]
      [termset: argument:
        le
        po
        [sentence: statement:
          [subject: argument:
            da]
          [predicate: predunit:
            rodja]]]]]]]]
```

Parser cache size 18456

Kou ta gi ti pa rodja

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Kou
```

```
          [argument1: arg1a:
            ta]]
    gi]
[sentence: statement:
  [subject: argument:
    ti]
  [predicate: markpred:
    pa
    [barepred: predunit:
      rodja]]]]
```

Parser cache size 14969

Mi moi ditca

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    moi
    [barepred: predunit:
      ditca]]]
```

Parser cache size 6568

Moi gi mi ditca

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Moi]
    gi]
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: predunit:
```

ditca]]

Parser cache size 9230

Moi mi, ditca

```
[uttF: sentence:
  [modifiers: modifier:
    Moi
    [argument1: arg1a:
      mi
      [freemod: comma:
        ,]]]
  [predicate: predunit:
    ditca]]
```

Parser cache size 9121

Moi ba gi mi ditca

```
[uttF: uttC:
  [headterms:
    [terms: modifier:
      Moi
      [argument1: arg1a:
        ba]]
    gi]
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: predunit:
      ditca]]]
```

Parser cache size 11643

Mi ditca moi ba

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      ditca]
    [termset: modifier:
      moi
      [argument1: arg1a:
        ba]]]]

```

Parser cache size 8213

Mi ditca moi lepo mi snire le junti

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      ditca]
    [termset: modifier:
      moi
      [argument1: abstractn:
        le
        po
        [sentence: statement:
          [subject: argument:
            mi]
          [predicate: barepred:
            [sentpred: predunit:
              snire]
            [termset: argument:
              le
              [descpred: predunit:

```



```
          [name:
            Djan, ]]]
[predicate: markpred:
  pa
  [barepred: predunit:
    godzi]]]
```

Parser cache size 18655

Mi, efafi la Djan, pa godzi

```
[uttF: statement:
  [subject: argument:
    [arg7: arg1a:
      Mi
      [freemod: comma:
        ,]]
    efafi
    [arg7: LANAME:
      la
      [name:
        Djan, ]]]
  [predicate: markpred:
    pa
    [barepred: predunit:
      godzi]]]
```

Parser cache size 16465

Mi pa vizka la Meris, enukoufi la Pit

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pa
```



```
[barepred:
  [sentpred: predunit:
    vizka]
  [termset: argument:
    [arg7: LANAME:
      la
      [name:
        Meris, ]]
    enukoufi
    [arg7: LANAME:
      la
      [name:
        Pit]]]]]]
```

Parser cache size 23945

Da farfu la Djek, la Meris, enukoufi ni la Selis

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: terms:
      [argumentA: argument:
        la
        [name:
          Djek, ]]
      [argumentB: argument:
        [arg7: LANAME:
          la
          [name:
            Meris, ]]
        enukoufi
        [arg7: arg3:
          ni
```

```
[arg2: LANAME:
      la
      [name:
        Selis]]]]]]]]
```

Parser cache size 29145

Mi pa godzi, enukoufi pa vizka la Djan

```
[uttF: statement:
      [subject: argument:
        Mi]
      [predicate: predicate2:
        [backpred: markpred:
          pa
          [barepred: predunit:
            godzi
            [freemod: comma:
              ,]]]
        enukoufi
        [backpred: markpred:
          pa
          [barepred:
            [sentpred: predunit:
              vizka]
            [termset: argument:
              la
              [name:
                Djan]]]]]]]]
```

Parser cache size 22626

Mi pa godzi, evafi pa vizka la Djan

```
[uttF: statement:
      [subject: argument:
```

```

Mi]
[predicate: predicate2:
  [backpred: markpred:
    pa
    [barepred: predunit:
      godzi
      [freemod: comma:
        ,]]]
evafi
[backpred: markpred:
  pa
  [barepred:
    [sentpred: predunit:
      vizka]
    [termset: argument:
      la
      [name:
        Djan]]]]]]]

```

Parser cache size 20384

Mi pa godzi, epafi vizka la Djan

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: predicate2:
    [backpred: markpred:
      pa
      [barepred: predunit:
        godzi
        [freemod: comma:
          ,]]]
    epafi
    [backpred: barepred:
      [sentpred: predunit:
        vizka]

```

```
[termset: argument:
  la
  [name:
    Djan]]]]]
```

Parser cache size 17784

Mi pa godzi, enumoifi, vizka la Djan

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: predicate2:
    [backpred: markpred:
      pa
      [barepred: predunit:
        godzi
        [freemod: comma:
          ,]]]
    enumoifi
    [freemod: comma:
      ,]
    [backpred: barepred:
      [sentpred: predunit:
        vizka]
      [termset: argument:
        la
        [name:
          Djan]]]]]
```

Parser cache size 23434

Tu danza lo skafi, ha lo tcati

```
[uttF: statement:
  [subject: argument:
```

```

    Tu]
  [predicate: barepred:
    [sentpred: predunit:
      danza]
    [termset: argument:
      [arg7: descriptn:
        lo
        [descpred: predunit:
          skafi
          [freemod: comma:
            ,]]]
      ha
      [arg7: descriptn:
        lo
        [descpred: predunit:
          tcati]]]]]]

```

Parser cache size 12688

Enoi eo

```

[uttF: uttC:
  Enoi
  [freemod:
    eo]]

```

Parser cache size 6536

Tu farfu la Djein, ha la Alis

```

[uttF: statement:
  [subject: argument:
    Tu]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]

```

```
[termset: argument:
  [arg7: LANAME:
    la
    [name:
      Djein, ]]
ha
[arg7: LANAME2:
  la
  [name:
    Alis]]]]]
```

Parser cache size 16618

E

```
[uttF: uttC:
  E]
```

Parser cache size 2744

Tu farfu la Djein, e la Alis, la Meris, ha la Betis

```
[uttF: statement:
  [subject: argument:
    Tu]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: terms:
      [argumentA: argument:
        [arg7: LANAME:
          la
          [name:
            Djein, ]]
        e
        [arg7: LANAME2:
```

```

        la
        [name:
          Alis, ]]]
[argumentB: argument:
  [arg7: LANAME:
    la
    [name:
      Meris, ]]
ha
  [arg7: LANAME:
    la
    [name:
      Betis]]]]]]]]

```

Parser cache size 29770

Noenoi. I mi farfu la Djein, la Meris, guu, e la Alis, la Betis

```

[uttF:
  [uttE: uttC:
    [uttA:
      Noenoi]
    [period:
      .]]
I
[uttF: statement:
  [subject: argument:
    mi]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
  [termset: termset2:
    [termset1:
      [terms:
        [argumentA: argument:
          la
          [name:

```

```

                Djein, ]]
            [argumentB: argument:
                la
                [name:
                    Meris, ]]]
        [guu:
            guu
            [freemod: comma:
                ,]]]
    e
    [termset1: terms:
        [argumentA: argument:
            la
            [name:
                Alis, ]]
        [argumentB: argument:
            la
            [name:
                Betis]]]]]]]]]]

```

Parser cache size 37322

Inusoa, tu farfu la Djein, a la Alis, la Meris, e la Betis

```

Inusoa
[freemod: comma:
    ,]
[uttF: statement:
    [subject: argument:
        tu]
    [predicate: barepred:
        [sentpred: predunit:
            farfu]
        [termset: terms:
            [argumentA: argument:
                [arg7: LANAME:
                    la

```



```

                [name:
                  Djein, ]]
a
  [arg7: LANAME2:
    la
      [name:
        Alis, ]]]
[argumentB: argument:
  [arg7: LANAME:
    la
      [name:
        Meris, ]]
e
  [arg7: LANAME:
    la
      [name:
        Betis]]]]]]

```

Parser cache size 35129

Ibuo no, tu farfu la Djein, e la Alis, la Meris, a la Betis

```

I
[freemod:
  buo]
[uttF: statement:
  [subject: argument:
    no
    [freemod: comma:
      ,]
    [argument1: arg1a:
      tu]]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
  [termset: terms:
    [argumentA: argument:

```

```

    [arg7: LANAME:
      la
      [name:
        Djein, ]]
  e
    [arg7: LANAME2:
      la
      [name:
        Alis, ]]]
[argumentB: argument:
  [arg7: LANAME:
    la
    [name:
      Meris, ]]
  a
    [arg7: LANAME:
      la
      [name:
        Betis]]]]]]]]

```

Parser cache size 36325

Da grobou go kukra lo litra

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred4: predunit:
      grobou]
    go
  [barepred:
    [sentpred: predunit:
      kukra]
    [termset: argument:
      lo
    [descpred: predunit:

```

litra]]]]]

Parser cache size 11892

Da kukra je lo litla gue grobou

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
      [predunit2: predunit1:
        kukra]
      [linkargs: linkargs1:
        [jelink:
          je
          [term: argument:
            lo
            [descpred: predunit:
              litla]]]
        [gue:
          gue]]]
      [despredD: predunit:
        grobou]]]
```

Parser cache size 13750

Da kukra je lo litla, grobou

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predunit:
    [predunit2: predunit1:
      kukra]
    [linkargs: jelink:
```

```
je
[term: argument:
  lo
  [descpred: despredE:
    [despredD: predunit:
      litla
      [freemod: comma:
        ,]]
    [despredD: predunit:
      grobou]]]]]]
```

Parser cache size 13204

Da penso go kukra mi

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred4: predunit:
      penso]
    go
    [barepred:
      [sentpred: predunit:
        kukra]
      [termset: argument:
        mi]]]]
```

Parser cache size 8831

Da kukra je mi penso

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
```

```
[senpred3: predunit:
  [predunit2: predunit1:
    kukra]
  [linkargs: jelink:
    je
    [term: argument:
      mi]]]
[despredD: predunit:
  penso]]]
```

Parser cache size 8269

Ta rutma je la Mineapolis, go kukra ti

```
[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: sentpred:
    [senpred4: predunit:
      [predunit2: predunit1:
        rutma]
      [linkargs: jelink:
        je
        [term: argument:
          la
          [name:
            Mineapolis, ]]]]
    go
    [barepred:
      [sentpred: predunit:
        kukra]
      [termset: argument:
        ti]]]]]
```

Parser cache size 19213

Ta rutma je la Mineapolis, jue la Cikagos, go kukra ti

```

[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: sentpred:
    [senpred4: predunit:
      [predunit2: predunit1:
        rutma]
      [linkargs: linkargs1:
        [jelink:
          je
          [term: argument:
            la
            [name:
              Mineapolis, ]]]]
        [links: juelink:
          jue
          [term: argument:
            la
            [name:
              Cikagos, ]]]]]]
    go
    [barepred:
      [sentpred: predunit:
        kukra]
      [termset: argument:
        ti]]]]]

```

Parser cache size 29027

Ta kukra je ti rutma la Mineapolis, la Cikagos

```

[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: barepred:

```

```

[sentpred: senpred4:
  [senpred3: predunit:
    [predunit2: predunit1:
      kukra]
    [linkargs: jelink:
      je
      [term: argument:
        ti]]]
  [despredD: predunit:
    rutma]]
[termset: terms:
  [argumentA: argument:
    la
    [name:
      Mineapolis,]]]
  [argumentB: argument:
    la
    [name:
      Cikagos]]]]]

```

Parser cache size 23250

Da farbru je la Djan, go plumro la Pit, la Djan

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred4: predunit:
      [predunit2: predunit1:
        farbru]
      [linkargs: jelink:
        je
        [term: argument:
          la
          [name:
            Djan, ]]]]

```

```

go
[barepred:
  [sentpred: predunit:
    plumro]
  [termset: terms:
    [argumentA: argument:
      la
      [name:
        Pit, ]]
    [argumentB: argument:
      la
      [name:
        Djan]]]]]]

```

Parser cache size 26648

Da plumro je la Pit, jue la Djan, farbru la Djan

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: senpred4:
      [senpred3: predunit:
        [predunit2: predunit1:
          plumro]
        [linkargs: linkargs1:
          [jelink:
            je
            [term: argument:
              la
              [name:
                Pit, ]]]]
          [links: juelink:
            jue
            [term: argument:
              la

```



```

[name:
  Djan, ]]]]
[despredD: predunit:
  farbru]]
[termset: argument:
  la
  [name:
    Djan]]]]]

```

Parser cache size 28855

Ba pa korfro je liu me holdu le lengu

```

[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          [predunit2: predunit1:
            korfro]
          [linkargs: jelink:
            je
            [term: argument:
              liu me]]]
        [despredD: predunit:
          holdu]]
      [termset: argument:
        le
        [descpred: predunit:
          lengu]]]]]]]

```

Parser cache size 19127

Ba pa holdu je le lengu gue go korfro liu me

```

[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: markpred:
    pa
  [barepred: sentpred:
    [senpred4: predunit:
      [predunit2: predunit1:
        holdu]
      [linkargs: linkargs1:
        [jelink:
          je
          [term: argument:
            le
            [descpred: predunit:
              lengu]]]
        [gue:
          gue]]]
      go
    [barepred:
      [sentpred: predunit:
        korfro]
      [termset: argument:
        liu me]]]]]]

```

Parser cache size 23343

Ba pa holdu je le lengu, go korfro liu me

```

[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: markpred:
    pa
  [barepred:

```

```

[sentpred: predunit:
  [predunit2: predunit1:
    holdu]
  [linkargs: jelinke:
    je
    [term: argument:
      le
      [descpred:
        [despredE: predunit:
          lengu
          [freemod: comma:
            ,]]
        go
        [descpred: predunit:
          korfro]]]]]]
[termset: argument:
  liu me]]]]

```

Parser cache size 21392

Da sanpa de di do du

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: barepred:
    [sentpred: predunit:
      sanpa]
    [termset: terms:
      [argumentA: argument:
        de]
      [argumentB: argument:
        di]
      [argumentC: argument:
        do]
      [argumentD: argument:
        du]]]]]

```

Parser cache size 12647

Zua da sanpa

```
[uttF: statement:
  [subject: argument:
    Zua
    [argument1: arg1a:
      da]]
  [predicate: predunit:
    sanpa]]
```

Parser cache size 5896

Zue de sanpa

```
[uttF: statement:
  [subject: argument:
    Zue
    [argument1: arg1a:
      de]]
  [predicate: predunit:
    sanpa]]
```

Parser cache size 5905

Zui di sanpa

```
[uttF: statement:
  [subject: argument:
    Zui
    [argument1: arg1a:
      di]]
  [predicate: predunit:
    sanpa]]
```

sanpa]]

Parser cache size 5909

Zuo do sanpa

```
[uttF: statement:
  [subject: argument:
    Zuo
    [argument1: arg1a:
      do]]
  [predicate: predunit:
    sanpa]]
```

Parser cache size 5922

Zuu du sanpa

```
[uttF: statement:
  [subject: argument:
    Zuu
    [argument1: arg1a:
      du]]
  [predicate: predunit:
    sanpa]]
```

Parser cache size 5927

Zuu lopo tcaro bapra ga sanpa zua lo redro litla zue lo humni

```
[uttF: statement:
  [subject: argument:
    Zuu
    [argument1: abstractn:
      lo
```

```

      po
      [sentence: sentpred:
        [senpred3: predunit:
          tcaro]
        [despredD: predunit:
          bapra]]]]
[predicate: markpred:
  ga
  [barepred:
    [sentpred: predunit:
      sanpa]
    [termset: modifiersx:
      [argxx: argument:
        zua
        [argument1: descriptn:
          lo
          [descpred: despredE:
            [despredD: predunit:
              redro]
            [despredD: predunit:
              litla]]]]]
      [argxx: argument:
        zue
        [argument1: descriptn:
          lo
          [descpred: predunit:
            humni]]]]]]]]]]

```

Parser cache size 23698

Lopo tcaro bapra ga nufe sanpa zua lo redro litla zue lo humni

```

[uttF: statement:
  [subject: argument:
    Lo
    po
    [sentence: sentpred:

```



```

denra]]]
[predicate: markpred:
  ga
  [barepred:
    [sentpred: senpred4:
      [senpred3: predunit:
        danri]
      [despredD: predunit:
        sanpa]]
    [termset: argument:
      zui
      [argument1: descriptn:
        lo
        [descpred: predunit:
          nimla]]]]]]]

```

Parser cache size 15802

Zuo lopo prano ga danri sanpa zue lo denro

```

[uttF: statement:
  [subject: argument:
    Zuo
    [argument1: abstractn:
      lo
      po
      [sentence: predunit:
        prano]]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          danri]
        [despredD: predunit:
          sanpa]]
      [termset: argument:

```



```
zue
[argument1: descriptn:
  lo
  [descpred: predunit:
    denro]]]]]]
```

Parser cache size 17222

Lopo prano ga danri ju sanpa zue lo denro

```
[uttF: statement:
  [subject: argument:
    Lo
    po
    [sentence: predunit:
      prano]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          danri]
        [despredD: predunit:
          ju
          sanpa]]
      [termset: argument:
        zue
        [argument1: descriptn:
          lo
          [descpred: predunit:
            denro]]]]]]]]
```

Parser cache size 17172

Lo nimla ga fu sanpa zuu lopo do clivi


```
        godzi]]
[termset: argument:
  zui
  [argument1: LANAME:
    la
    [name:
      Seint
      Pol]]]]]
```

Parser cache size 25503

Ba godzi la Miniapolis, la Seint Pol

```
[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: terms:
      [argumentA: argument:
        la
        [name:
          Miniapolis, ]]
      [argumentB: argument:
        la
        [name:
          Seint
          Pol]]]]]
```

Parser cache size 19221

Zuo ta vedma

```
[uttF: statement:
  [subject: argument:
```

```
Zuo
  [argument1: arg1a:
    ta]]
[predicate: predunit:
  vedma]]
```

Parser cache size 5947

Ta ju vedma

```
[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: predunit:
    ju
    vedma]]
```

Parser cache size 5864

Ba vedma be bo ta

```
[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: barepred:
    [sentpred: predunit:
      vedma]
    [termset: terms:
      [argumentA: argument:
        be]
      [argumentB: argument:
        bo]
      [argumentC: argument:
        ta]]]]
```

Parser cache size 10509

Le farfu je zui la Meris

```
[uttF: argument:
  Le
  [descpred: predunit:
    [predunit2: predunit1:
      farfu]
    [linkargs: jelink:
      je
      [term: argument:
        zui
        [argument1: LANAME:
          la
          [name:
            Meris]]]]]]]
```

Parser cache size 13800

Le farfu je ba jue la Meris

```
[uttF: argument:
  Le
  [descpred: predunit:
    [predunit2: predunit1:
      farfu]
    [linkargs: linkargs1:
      [jelink:
        je
        [term: argument:
          ba]]
      [links: juelink:
        jue
        [term: argument:
          la
          [name:
```



```
[predicate: predunit:
  [predunit2: predunit1:
    kukra]
  [linkargs: jelink:
    je
    [term: argument:
      zui
      [argument1: arg1:
        lio
        [term: argument:
          nenimuemeikuasei,
          [descpred: predunit:
            tcaro]]]]]]]]
```

Parser cache size 19353

Zui lio nenimuemeikuasei, fu kukra

```
[uttF: argument:
  Zui
  [argument1: arg1:
    lio
    [term: argument:
      nenimuemeikuasei,
      [descpred: predunit:
        fu
        kukra]]]]]
```

Parser cache size 17383

Ba kukra be lio nenimuemeikuasei

```
[uttF: statement:
  [subject: argument:
    Ba]
  [predicate: barepred:
```

```
[sentpred: predunit:
  kukra]
[termset: terms:
  [argumentA: argument:
    be]
  [argumentB: argument:
    lio
    nenimuemeikuasei]]]]
```

Parser cache size 17435

Mi fapa sucmi

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    fapa
  [barepred: predunit:
    sucmi]]]
```

Parser cache size 7453

Mi nepa sucmi

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    nepa
  [barepred: predunit:
    sucmi]]]
```

Parser cache size 7551

Mi nipa sucmi


```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    nipa
    [barepred: predunit:
      sucmi]]]
```

Parser cache size 7545

Mi nina sucmi

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    nina
    [barepred: predunit:
      sucmi]]]
```

Parser cache size 7545

Mi rana sucmi

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    rana
    [barepred: predunit:
      sucmi]]]
```

Parser cache size 7493

Mi no na sucmi

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: backpred1:
    [N02:
      no]
    [markpred:
      na
      [barepred: predunit:
        sucmi]]]]
```

Parser cache size 8440

Mi pacenoina sucmi

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pacenoina
    [barepred: predunit:
      sucmi]]]
```

Parser cache size 9970

Mi no pacana sucmi

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: backpred1:
    [N02:
      no]
```

```
[markpred:
  pacana
  [barepred: predunit:
    sucmi]]]]
```

Parser cache size 10715

Mi pacefa bragai

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pacefa
    [barepred: predunit:
      bragai]]]
```

Parser cache size 10204

Mi no pacanafa bragai

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: backpred1:
    [N02:
      no]
    [markpred:
      pacanafa
      [barepred: predunit:
        bragai]]]]]
```

Parser cache size 13415

Facenoina mi, garni levi landi

```

[uttF: sentence:
  [modifiers: modifier:
    Facenoina
    [argument1: arg1a:
      mi
      [freemod: comma:
        ,]]]
  [predicate: barepred:
    [sentpred: predunit:
      garni]
    [termset: argument:
      le
      vi
      [descpred: predunit:
        landi]]]]]

```

Parser cache size 18178

Kanoi no tu fa kamla, ki no mi hijra va

```

[uttF: sentence:
  Kanoi
  [sentence: statement:
    [subject: argument:
      no
      [argument1: arg1a:
        tu]]
    [predicate: markpred:
      fa
      [barepred: predunit:
        kamla
        [freemod: comma:
          ,]]]]
  ki
  [uttA1: statement:
    [subject: argument:

```

```
no
[argument1: arg1a:
  mi]]
[predicate: barepred:
  [sentpred: predunit:
    hijra]
  [termset: modifier:
    va]]]]
```

Parser cache size 23165

No tu fa kamla, inoca no mi hijra va

```
[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      No
      [argument1: arg1a:
        tu]]
    [predicate: markpred:
      fa
      [barepred: predunit:
        kamla
        [freemod: comma:
          ,]]]]
  inoca
  [sen1: statement:
    [subject: argument:
      no
      [argument1: arg1a:
        mi]]
    [predicate: barepred:
      [sentpred: predunit:
        hijra]
      [termset: modifier:
        va]]]]
```

Parser cache size 21946

No, tu fa kamla, inoca no, mi hijra va

```
[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      No
      [freemod: comma:
        ,]
      [argument1: arg1a:
        tu]]
    [predicate: markpred:
      fa
      [barepred: predunit:
        kamla
        [freemod: comma:
          ,]]]]
  inoca
  [sen1: statement:
    [subject: argument:
      no
      [freemod: comma:
        ,]
      [argument1: arg1a:
        mi]]
    [predicate: barepred:
      [sentpred: predunit:
        hijra]
      [termset: modifier:
        va]]]]
```

Parser cache size 23371

No mi fa hijra va, icanoi no tu kamla

```

[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      No
      [argument1: arg1a:
        mi]]
    [predicate: markpred:
      fa
      [barepred:
        [sentpred: predunit:
          hijra]
        [termset: modifier:
          va
          [freemod: comma:
            ,]]]]]]

```

icanoi

```

[sen1: statement:
  [subject: argument:
    no
    [argument1: arg1a:
      tu]]
  [predicate: predunit:
    kamla]]]

```

Parser cache size 22495

No, mi fa hijra va, icanoi no, tu kamla

```

[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      No
      [freemod: comma:
        ,]
      [argument1: arg1a:
        mi]]
    [predicate: markpred:

```

```

fa
[barepred:
  [sentpred: predunit:
    hijra]
  [termset: modifier:
    va
    [freemod: comma:
      ,]]]]
icanoi
[sen1: statement:
  [subject: argument:
    no
    [freemod: comma:
      ,]
    [argument1: arg1a:
      tu]]
  [predicate: predunit:
    kamla]]]

```

Parser cache size 23920

Kanoi no tu fa kamla ki ke no mi hijra va ki tu fa kecdri

```

[uttF: sentence:
  Kanoi
  [sentence: statement:
    [subject: argument:
      no
      [argument1: arg1a:
        tu]]
    [predicate: markpred:
      fa
      [barepred: predunit:
        kamla]]]
  ki
  [uttA1: keksent:
    ke

```



```

[sentence: statement:
  [subject: argument:
    no
    [argument1: arg1a:
      mi]]
  [predicate: barepred:
    [sentpred: predunit:
      hijra]
    [termset: modifier:
      va]]]
ki
[uttA1: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    fa
    [barepred: predunit:
      kecdri]]]]]

```

Parser cache size 36312

No mi fa hijra va, icanoi no tu fapa kamla

```

[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      No
      [argument1: arg1a:
        mi]]
    [predicate: markpred:
      fa
      [barepred:
        [sentpred: predunit:
          hijra]
        [termset: modifier:
          va
          [freemod: comma:

```

```

,]]]]]
icanoi
[sen1: statement:
  [subject: argument:
    no
    [argument1: arg1a:
      tu]]
  [predicate: markpred:
    fapa
    [barepred: predunit:
      kamla]]]]]

```

Parser cache size 26444

Tu fapa kamla, icanoi mi hijra va

```

[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      Tu]
    [predicate: markpred:
      fapa
      [barepred: predunit:
        kamla
        [freemod: comma:
          ,]]]]]

```

```

icanoi
[sen1: statement:
  [subject: argument:
    mi]
  [predicate: barepred:
    [sentpred: predunit:
      hijra]
    [termset: modifier:
      va]]]]]

```

Parser cache size 19384

Mi fa hijra va, inoca tu fapa kamla

```
[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      Mi]
    [predicate: markpred:
      fa
    [barepred:
      [sentpred: predunit:
        hijra]
      [termset: modifier:
        va
      [freemod: comma:
        ,]]]]]
  inoca
  [sen1: statement:
    [subject: argument:
      tu]
    [predicate: markpred:
      fapa
    [barepred: predunit:
      kamla]]]]]
```

Parser cache size 21630

Kanoi mi fa hijra va ki tu pa kamla

```
[uttF: sentence:
  Kanoi
  [sentence: statement:
    [subject: argument:
      mi]
    [predicate: markpred:
      fa
```

```

        [barepred:
          [sentpred: predunit:
            hijra]
          [termset: modifier:
            va]]]]
ki
[uttA1: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
    pa
    [barepred: predunit:
      kamla]]]]

```

Parser cache size 21129

Ka no mi fa hijra va ki tu pa kamla

```

[uttF: sentence:
  Ka
  [sentence: statement:
    [subject: argument:
      no
      [argument1: arg1a:
        mi]]
    [predicate: markpred:
      fa
      [barepred:
        [sentpred: predunit:
          hijra]
        [termset: modifier:
          va]]]]
  ki
  [uttA1: statement:
    [subject: argument:
      tu]
    [predicate: markpred:

```

```
pa
[barepred: predunit:
  kamla]]]]
```

Parser cache size 22531

No mi fa hijra va, ica tu pa kamla

```
[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      No
      [argument1: arg1a:
        mi]]
    [predicate: markpred:
      fa
      [barepred:
        [sentpred: predunit:
          hijra]
        [termset: modifier:
          va
          [freemod: comma:
            ,]]]]]]
```

```
ica
  [sen1: statement:
    [subject: argument:
      tu]
    [predicate: markpred:
      pa
      [barepred: predunit:
        kamla]]]]
```

Parser cache size 21155

Tu fapa kamla, ica no mi hijra va

```
[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      Tu]
    [predicate: markpred:
      fapa
      [barepred: predunit:
        kamla
        [freemod: comma:
          ,]]]]
```

ica

```
[sen1: statement:
  [subject: argument:
    no
    [argument1: arg1a:
      mi]]
  [predicate: barepred:
    [sentpred: predunit:
      hijra]
    [termset: modifier:
      va]]]]
```

Parser cache size 19440

Tu fapa kamla, ica ke no hijra va ki tu kecdri

```
[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      Tu]
    [predicate: markpred:
      fapa
      [barepred: predunit:
        kamla
        [freemod: comma:
          ,]]]]
```

ica

```

[sen1: keksent:
  ke
  [sentence: barepred:
    [sentpred: predunit:
      no
      [predunit1:
        hijra]]
    [termset: modifier:
      va]]
  ki
  [uttA1: statement:
    [subject: argument:
      tu]
    [predicate: predunit:
      kecdri]]]]

```

Parser cache size 28198

Ice mi sui fa hijra va

```

Ice
[uttF: statement:
  [subject: argument:
    mi
    [freemod:
      sui]]
  [predicate: markpred:
    fa
    [barepred:
      [sentpred: predunit:
        hijra]
      [termset: modifier:
        va]]]]

```

Parser cache size 16222

Ti pa rodja, ikou, tu cuidru da

```

[uttF:
  [uttE: statement:
    [subject: argument:
      Ti]
    [predicate: markpred:
      pa
      [barepred: predunit:
        rodja
        [freemod: comma:
          ,]]]]
  ikou
  [freemod: comma:
    ,]
  [uttF: statement:
    [subject: argument:
      tu]
    [predicate: barepred:
      [sentpred: predunit:
        cuidru]
      [termset: argument:
        da]]]]

```

Parser cache size 19896

Nukouki ti pa rodja, ki tu cuidru da

```

[uttF: sentence:
  Nukouki
  [sentence: statement:
    [subject: argument:
      ti]
    [predicate: markpred:
      pa
      [barepred: predunit:
        rodja

```



```

                                [freemod: comma:
                                    ,]]]]
ki
[uttA1: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      cuidru]
    [termset: argument:
      da]]]]

```

Parser cache size 23767

Tu pa cuidru ti, inukou, da rodja

```

[uttF:
  [uttE: statement:
    [subject: argument:
      Tu]
    [predicate: markpred:
      pa
      [barepred:
        [sentpred: predunit:
          cuidru]
        [termset: argument:
          ti
          [freemod: comma:
            ,]]]]]]

```

```

inukou
[freemod: comma:
  ,]
[uttF: statement:
  [subject: argument:
    da]
  [predicate: predunit:
    rodja]]]

```

Parser cache size 21224

Mi pa cuidru da, imoi, da pafa rodja

```
[uttF:
  [uttE: statement:
    [subject: argument:
      Mi]
    [predicate: markpred:
      pa
    [barepred:
      [sentpred: predunit:
        cuidru]
      [termset: argument:
        da
        [freemod: comma:
          ,]]]]]
  imoi
  [freemod: comma:
    ,]
  [uttF: statement:
    [subject: argument:
      da]
    [predicate: markpred:
      pafa
    [barepred: predunit:
      rodja]]]]]
```

Parser cache size 23605

Kouki tu pa cuidru ti ki da rodja

```
[uttF: sentence:
  Kouki
  [sentence: statement:
```

```

[subject: argument:
  tu]
[predicate: markpred:
  pa
  [barepred:
    [sentpred: predunit:
      cuidru]
    [termset: argument:
      ti]]]]
ki
[uttA1: statement:
  [subject: argument:
    da]
  [predicate: predunit:
    rodja]]]

```

Parser cache size 22017

Mi pa danza ti, imoi, mi pafa ponsu ta

```

[uttF:
  [uttE: statement:
    [subject: argument:
      Mi]
    [predicate: markpred:
      pa
      [barepred:
        [sentpred: predunit:
          danza]
        [termset: argument:
          ti
          [freemod: comma:
            ,]]]]]]
  imoi
  [freemod: comma:
    ,]
  [uttF: statement:

```

```
[subject: argument:
  mi]
[predicate: markpred:
  pafa
  [barepred:
    [sentpred: predunit:
      ponsu]
    [termset: argument:
      ta]]]]]
```

Parser cache size 24000

Mi ditca, imoi, mi bremao lo junti

```
[uttF:
  [uttE: statement:
    [subject: argument:
      Mi]
    [predicate: predunit:
      ditca
      [freemod: comma:
        ,]]]
  imoi
  [freemod: comma:
    ,]
  [uttF: statement:
    [subject: argument:
      mi]
    [predicate: barepred:
      [sentpred: predunit:
        bremao]
      [termset: argument:
        lo
        [descpred: predunit:
          junti]]]]]]]
```

Parser cache size 17506

Eo kambei mi lae li, la Loglan Nen, lu

```
[freemod:
  Eo]
[utterance: barepred:
  [sentpred: predunit:
    kambei]
  [termset: terms:
    [argumentA: argument:
      mi]
    [argumentB: argument:
      lae
    [arg5: LI:
      li
      ,
      [utterance0: argument:
        la
        [name:
          Loglan
          Nen, ]]
      lu]]]]
```

Parser cache size 25677

Eo kambei mi lae lie War y and y Peace

```
[freemod:
  Eo]
[utterance: barepred:
  [sentpred: predunit:
    kambei]
  [termset: terms:
    [argumentA: argument:
      mi]
    [argumentB: argument:
```

```
    lae
    [arg5: LIE:
      lie
      War
      y
      and
      y
      Peace]]]]
```

Parser cache size 13622

La Djan, melaelie stingy

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: predunit:
    me
    [argument1: argx:
      lae
      [arg5: LIE:
        lie
        stingy]]]]]
```

Parser cache size 15727

Ei tu pa ridle laeli, la Loglan Nen, lu

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: markpred:
```

```

pa
[barepred:
  [sentpred: predunit:
    ridle]
  [termset: argument:
    lae
    [arg5: LI:
      li
      ,
      [utterance0: argument:
        la
        [name:
          Loglan
          Nen, ]]
      lu]]]]]

```

Parser cache size 25173

Ei tu fundi laeli, la Loglan Nen, lu

```

[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      fundi]
    [termset: argument:
      lae
      [arg5: LI:
        li
        ,
        [utterance0: argument:
          la
          [name:
            Loglan

```

Nen,]]
lu]]]]

Parser cache size 22341

Da pa cutse lue lepo la Djan, pa prano

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: markpred:
    pa
  [barepred:
    [sentpred: predunit:
      cutse]
    [termset: argument:
      lue
    [arg5: abstractn:
      le
      po
    [sentence: statement:
      [subject: argument:
        la
        [name:
          Djan, ]]
      [predicate: markpred:
        pa
        [barepred: predunit:
          prano]]]]]]]]]]
```

Parser cache size 21534

Ei tu vizka lue la Djan

```
[freemod:
  Ei]
```



```

[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      vizka]
    [termset: argument:
      lue
      [arg5: LANAME:
        la
        [name:
          Djan]]]]]]

```

Parser cache size 12171

Ei tu vizka laeli, la Djan, lu

```

[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      vizka]
    [termset: argument:
      lae
      [arg5: LI:
        li
        ,
        [utterance0: argument:
          la
          [name:
            Djan, ]]
        lu]]]]]]

```

Parser cache size 18010

Ei tu vizka lae la Djan

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      vizka]
    [termset: argument:
      lae
      [arg5: LANAME:
        la
        [name:
          Djan]]]]]
```

Parser cache size 12440

Ei tu vizka lo sanpa je la Djan

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      vizka]
    [termset: argument:
      lo
      [descpred: predunit:
        [predunit2: predunit1:
          sanpa]
        [linkargs: jelink:
```

```
je
[term: argument:
  la
  [name:
    Djan]]]]]]]]
```

Parser cache size 15038

Ei tu vizka lo nu sanpa je la Djan

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      vizka]
    [termset: argument:
      lo
      [descpred: predunit:
        [predunit2: predunit1:
          nu
          sanpa]
        [linkargs: jelink:
          je
          [term: argument:
            la
            [name:
              Djan]]]]]]]]]]
```

Parser cache size 17522

Ei tu vizka lue lue lue la Djan

```
[freemod:
```

```

    Ei]
  [utterance: statement:
    [subject: argument:
      tu]
    [predicate: barepred:
      [sentpred: predunit:
        vizka]
      [termset: argument:
        lue
        lue
        lue
      [arg5: LANAME:
        la
        [name:
          Djan]]]]]

```

Parser cache size 16621

Ei tu vizka lo sanpa je lo sanpa je lo sanpa je la Djan

```

  [freemod:
    Ei]
  [utterance: statement:
    [subject: argument:
      tu]
    [predicate: barepred:
      [sentpred: predunit:
        vizka]
      [termset: argument:
        lo
      [descpred: predunit:
        [predunit2: predunit1:
          sanpa]
        [linkargs: jelink:
          je
          [term: argument:
            lo

```

```

[descpred: predunit:
  [predunit2: predunit1:
    sanpa]
  [linkargs: jelin:
    je
    [term: argument:
      lo
      [descpred: predunit:
        [predunit2: pred
          sanpa]
        [linkargs: jelin:
          je
          [term: arg
            la
            [nam
              D

```

Parser cache size 24978

Ei tu vizka lae lue lae la Djan

```

[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      vizka]
    [termset: argument:
      lae
      lue
      lae
    [arg5: LANAME:
      la
      [name:
        Djan]]]]]

```

Parser cache size 17159

Mi farfu jo la Loglan

```
[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      farfu
      [freemod:
        jo]]
    [termset: argument:
      la
      [name:
        Loglan]]]]]
```

Parser cache size 11756

Ia no. I tu matma jo la Loglan

```
[freemod:
  Ia]
[utterance: uttF:
  [uttE: uttC:
    no
    [period:
      .]]
  I
  [uttF: statement:
    [subject: argument:
      tu]
    [predicate: barepred:
      [sentpred: predunit:
        matma
```

```
        [freemod:
          jo]]
[termset: argument:
  la
  [name:
    Loglan]]]]]]
```

Parser cache size 17563

I la Uorf, pa farfu jo de

```
I
[uttF: statement:
  [subject: argument:
    la
    [name:
      Uorf, ]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        farfu
        [freemod:
          jo]]
      [termset: argument:
        de]]]]
```

Parser cache size 18393

Da smina huigro tojo

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: sentpred:
    [senpred3: predunit:
```

```
        smina]
    [despredD: predunit:
        huigro
    [freemod:
        tojo]]]]
```

Parser cache size 8923

Da smina huicma jo

```
[uttF: statement:
    [subject: argument:
        Da]
    [predicate: sentpred:
        [senpred3: predunit:
            smina]
        [despredD: predunit:
            huicma
        [freemod:
            jo]]]]
```

Parser cache size 7924

La Djan, pa brahea rajo na lepo la Loglan, brana

```
[uttF: statement:
    [subject: argument:
        La
        [name:
            Djan, ]]
    [predicate: markpred:
        pa
        [barepred:
            [sentpred: predunit:
                brahea
            [freemod:
```



```

        rajo]]
[termset: modifier:
    na
    [argument1: abstractn:
        le
        po
        [sentence: statement:
            [subject: argument:
                la
                [name:
                    Loglan, ]]
            [predicate: predunit:
                brana]]]]]]]]

```

Parser cache size 30504

Ia no. I la Djan, pa bremao ckela ditca tejo la Loglan

```

[freemod:
    Ia]
[utterance: uttF:
    [uttE: uttC:
        no
        [period:
            .]]
    I
    [uttF: statement:
        [subject: argument:
            la
            [name:
                Djan, ]]
        [predicate: markpred:
            pa
            [barepred:
                [sentpred: senpred4:
                    [senpred3: predunit:
                        bremao]

```

```

[despredD: predunit:
  ckela]
[despredD: predunit:
  ditca
  [freemod:
    tejo]]]
[termset: argument:
  la
  [name:
    Loglan]]]]]]

```

Parser cache size 26910

La Djan, pa bremao ckela ditca la Loglan, rajo

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          bremao]
        [despredD: predunit:
          ckela]
        [despredD: predunit:
          ditca]]
      [termset: argument:
        la
        [name:
          Loglan,
          [freemod:
            rajo]]]]]]]

```

Parser cache size 22081

La Djan, pa rajo bremao ckela ditca la Loglan

```
[uttF: statement:
  [subject: argument:
    La
    [name:
      Djan, ]]
  [predicate: markpred:
    pa
    [freemod:
      rajo]
    [barepred:
      [sentpred: senpred4:
        [senpred3: predunit:
          bremao]
        [despredD: predunit:
          ckela]
        [despredD: predunit:
          ditca]]
      [termset: argument:
        la
        [name:
          Loglan]]]]]]
```

Parser cache size 21343

Le mrenu pa vedma le kangu le cmalo nirli

```
[uttF: statement:
  [subject: argument:
    Le
    [descpred: predunit:
      mrenu]]
  [predicate: markpred:
```

```

pa
[barepred:
  [sentpred: predunit:
    vedma]
  [termset: terms:
    [argumentA: argument:
      le
      [descpred: predunit:
        kangu]]
    [argumentB: argument:
      le
      [descpred: despredE:
        [despredD: predunit:
          cmalo]
        [despredD: predunit:
          nirli]]]]]]]]

```

Parser cache size 14487

Nei fa cluva kei

```

[uttF: statement:
  [subject: argument:
    Nei]
  [predicate: markpred:
    fa
    [barepred:
      [sentpred: predunit:
        cluva]
      [termset: argument:
        kei]]]]

```

Parser cache size 8912

La Ned, farfu nei

```

[uttF: statement:
  [subject: argument:
    La
    [name:
      Ned, ]]
  [predicate: barepred:
    [sentpred: predunit:
      farfu]
    [termset: argument:
      nei]]]

```

Parser cache size 11845

Nai cluva nei, e fundi lo kangu, enumoi, fa cluva kei

```

[uttF: statement:
  [subject: argument:
    Nai]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        cluva]
      [termset: argument:
        nei
        [freemod: comma:
          ,]]]
    e
    [backpred: barepred:
      [sentpred: predunit:
        fundi]
      [termset: argument:
        lo
        [descpred: predunit:
          kangu
          [freemod: comma:
            ,]]]]
    enumoi

```

```
[freemod: comma:
  ,]
[backpred: markpred:
  fa
  [barepred:
    [sentpred: predunit:
      cluva]
    [termset: argument:
      kei]]]]]
```

Parser cache size 26664

Nai merji la Meris, ice Mai no fundi lo kangu

```
[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      Nai]
    [predicate: barepred:
      [sentpred: predunit:
        merji]
      [termset: argument:
        la
        [name:
          Meris, ]]]]
  ice
  [sen1: statement:
    [subject: argument:
      Mai]
    [predicate: barepred:
      [sentpred: predunit:
        no
        [predunit1:
          fundi]]
      [termset: argument:
        lo
        [descpred: predunit:
```

kangu]]]]]

Parser cache size 22423

Bai merji la Alis, ice Ziama fundi lo kangu

```
[uttF: sentence:
  [sen1: statement:
    [subject: argument:
      Bai]
    [predicate: barepred:
      [sentpred: predunit:
        merji]
      [termset: argument:
        la
        [name:
          Alis, ]]]]
  ice
  [sen1: statement:
    [subject: argument:
      Ziama]
    [predicate: barepred:
      [sentpred: predunit:
        fundi]
      [termset: argument:
        lo
        [descpred: predunit:
          kangu]]]]]
```

Parser cache size 21706

Inusoa, ba nu dakli lepo Mai no fa cluva kei

```
Inusoa
[freemod: comma:
  ,]
```



```
[arg5: LIE:
  lie
  Central
  y
  Intelligence
  y
  Committee]]]]
```

Parser cache size 17645

Nai nu perpli la Caiziza

```
[uttF: statement:
  [subject: argument:
    Nai]
  [predicate: barepred:
    [sentpred: predunit:
      nu
      perpli]
    [termset: argument:
      la
      [name:
        Caiziza]]]]]
```

Parser cache size 15242

Mai nu perpli la Zunai

```
[uttF: statement:
  [subject: argument:
    Mai]
  [predicate: barepred:
    [sentpred: predunit:
      nu
      perpli]
    [termset: argument:
```

```
la
[name:
  Zunai]]]]
```

Parser cache size 13410

Nei danza lepo fa nu perpli la Zunaizesaicaizo

```
[uttF: statement:
  [subject: argument:
    Nei]
  [predicate: barepred:
    [sentpred: predunit:
      danza]
    [termset: argument:
      le
      po
      [sentence: statement:
        fa
        [barepred:
          [sentpred: predunit:
            nu
            perpli]
          [termset: argument:
            la
            [name:
              Zunaizesaicaizo]]]]]]]]]]
```

Parser cache size 26354

Ibuo nei danza lepo na nu perpli la Saipaicaiza

```
I
[freemod:
  buo]
[uttF: statement:
```

```

[subject: argument:
  nei]
[predicate: barepred:
  [sentpred: predunit:
    danza]
  [termset: argument:
    le
    po
  [sentence: statement:
    na
    [barepred:
      [sentpred: predunit:
        nu
        perpli]
      [termset: argument:
        la
        [name:
          Saipaicaiza]]]]]]]]

```

Parser cache size 27853

Meo ponsu ba jio kasni

```

[uttF: statement:
  [subject: argument:
    Meo]
  [predicate: barepred:
    [sentpred: predunit:
      ponsu]
    [termset: argument:
      [arg1a:
        ba]
      [argmod: argmod1:
        jio
        [sentence: predunit:
          kasni]]]]]]

```

Parser cache size 9701

Lomela Haitozo, bi lo cutri

```
[uttF: statement:
  [subject: argument:
    Lo
    [descpred: predunit:
      me
      [argument1: LANAME:
        la
        [name:
          Haitozo,]]]]
  [predicate: identpred:
    bi
    [termset: argument:
      lo
      [descpred: predunit:
        cutri]]]]]
```

Parser cache size 17237

Ta mela haitosaizofo

```
[uttF: statement:
  [subject: argument:
    Ta]
  [predicate: predunit:
    me
    [argument1: LANAME:
      la
      [name:
        haitosaizofo]]]]]
```

Parser cache size 14024

Zia groda bei cei

```
[uttF: statement:
  [subject: argument:
    Zia]
  [predicate: barepred:
    [sentpred: predunit:
      groda]
    [termset: terms:
      [argumentA: argument:
        bei]
      [argumentB: argument:
        cei]]]]]
```

Parser cache size 8542

Hoi Mrenu

```
[freemod: voc:
  Hoi
  [descpred: predunit:
    Mrenu]]]
```

Parser cache size 3415

Hoi Ganfua go Redro nu Herfa

```
[freemod: voc:
  Hoi
  [descpred:
    [despredE: predunit:
      Ganfua]
    go
    [descpred: despredE:
      [despredD: predunit:
```

```
Redro]
[despredD: predunit:
  nu
  Herfa]]]]
```

Parser cache size 12198

Ei tu vizka Hoi Ganbra, lepo mi pana nu ganble turka letu fothaa

```
[freemod:
  Ei]
[utterance: statement:
  [subject: argument:
    tu]
  [predicate: barepred:
    [sentpred: predunit:
      vizka
      [freemod: voc:
        Hoi
        [descpred: predunit:
          Ganbra
          [freemod: comma:
            ,]]]]
    [termset: argument:
      le
      po
      [sentence: statement:
        [subject: argument:
          mi]
        [predicate: markpred:
          pana
          [barepred:
            [sentpred: senpred4:
              [senpred3: predunit:
                nu
                ganble]
              [despredD: predunit:
```



```

        Garfua]
    go
    [descpred: despredE:
      [despredD: predunit:
        Redro]
      [despredD: predunit:
        nu
        Herfa]]]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        matma]
      [termset: argument:
        mi]]]]

```

Parser cache size 18501

La Blabi Garfua pa korji la Redro Garfua lepo godzi la Hatro Sitfa

```

[uttF: statement:
  [subject: argument:
    La
    [descpred: despredE:
      [despredD: predunit:
        Blabi]
      [despredD: predunit:
        Garfua]]]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        korji]
      [termset: terms:
        [argumentA: argument:
          la
          [descpred: despredE:

```



```

                [subject: argument:
                  mi]
                [predicate: predunit:
                  sucmi]]
        [period:
          .]]
Ifa
[freemod: comma:
  ,]
[uttF:
  [uttE: uttD:
    [sentence: statement:
      [subject: argument:
        mi]
      [predicate: predunit:
        fleti]]
    [period:
      .]]
  Ifa
  [freemod: comma:
    ,]
  [uttF: uttD:
    [sentence: statement:
      [subject: argument:
        mi]
      [predicate: predunit:
        dzoru]]
    [period:
      .]]]]]]

```

Parser cache size 26158

Mi pa prano, e sucmi, e fleti, e dzoru

```

[uttF: statement:
  [subject: argument:
    Mi]

```

```

[predicate: predicate2:
  [backpred: markpred:
    pa
    [barepred: predunit:
      prano
      [freemod: comma:
        ,]]]
  e
  [backpred: predunit:
    sucmi
    [freemod: comma:
      ,]]
  e
  [backpred: predunit:
    fleti
    [freemod: comma:
      ,]]
  e
  [backpred: predunit:
    dzoru]]]

```

Parser cache size 16362

Mi pa takna da, e de, e di, efafi do

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: markpred:
    pa
    [barepred:
      [sentpred: predunit:
        takna]
      [termset: argument:
        [arg7: arg1a:
          da
          [freemod: comma:

```

```

,]]
e
[arg7: arg1a:
de
[freemod: comma:
,]]
e
[arg7: arg1a:
di
[freemod: comma:
,]]
efafi
[arg7: arg1a:
do]]]]]

```

Parser cache size 24296

Muvdo, icanoi da redro, icaci de nigro

```

[uttF: uttE:
[uttD: predunit:
Muvdo
[freemod: comma:
,]]
icanoi
[uttD:
[uttC: statement:
[subject: argument:
da]
[predicate: predunit:
redro
[freemod: comma:
,]]]
icaci
[uttD: statement:
[subject: argument:
de]

```

```
[predicate: predunit:  
nigro]]]]
```

Parser cache size 21085

Muvdo, icanoi da redro, ica de nigro

```
[uttF: sentence:  
  [sen1: predunit:  
    Muvdo  
    [freemod: comma:  
      ,]]  
  icanoi  
  [sen1: statement:  
    [subject: argument:  
      da]  
    [predicate: predunit:  
      redro  
      [freemod: comma:  
        ,]]]  
  ica  
  [sen1: statement:  
    [subject: argument:  
      de]  
    [predicate: predunit:  
      nigro]]]
```

Parser cache size 18152

Muvdo, icanoige da redro, ica de nigro, ica di verdi

```
[uttE: predunit:  
  Muvdo  
  [freemod: comma:  
    ,]]  
icanoige
```

```

[utterance: sentence:
  [sen1: statement:
    [subject: argument:
      da]
    [predicate: predunit:
      redro
      [freemod: comma:
        ,]]]
ica
[sen1: statement:
  [subject: argument:
    de]
  [predicate: predunit:
    nigro
    [freemod: comma:
      ,]]]
ica
[sen1: statement:
  [subject: argument:
    di]
  [predicate: predunit:
    verdi]]]

```

Parser cache size 27803

Da movdo, anoi redro, aci nigro

```

[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predicate2:
    [backpred: predunit:
      movdo
      [freemod: comma:
        ,]]
    anoi
    [backpred:

```

```
[backpred1: predunit:
  redro
  [freemod: comma:
    ,]]
aci
[backpred1: predunit:
  nigro]]]
```

Parser cache size 14666

Da movdo, anoi redro, a nigro

```
[uttF: statement:
  [subject: argument:
    Da]
  [predicate: predicate2:
    [backpred: predunit:
      movdo
      [freemod: comma:
        ,]]
    anoi
    [backpred: predunit:
      redro
      [freemod: comma:
        ,]]
    a
    [backpred: predunit:
      nigro]]]
```

Parser cache size 12922

Da movdo, anoige redro, a nigro, epafi vegri

```
[uttF: statement:
  [subject: argument:
    Da]
```



```

[predicate: predicate1:
  [predicate2: predunit:
    muvdo
    [freemod: comma:
      ,]]
  anoige
  [predicate1: predicate2:
    [backpred: predunit:
      redro
      [freemod: comma:
        ,]]
    a
    [backpred: predunit:
      nigro
      [freemod: comma:
        ,]]
    epafi
    [backpred: predunit:
      vegri]]]]

```

Parser cache size 21135

Mi godzi da, anoi de, e di

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: argument:
      [arg7: arg1a:
        da
        [freemod: comma:
          ,]]
      anoi
      [arg7: arg1a:

```

```
de
[freemod: comma:
,]]
```

```
e
[arg7: arg1a:
di]]]]
```

Parser cache size 16504

Mi godzi da, anoi de, eci di

```
[uttF: statement:
[subject: argument:
Mi]
[predicate: barepred:
[sentpred: predunit:
godzi]
[termset: argument:
[arg7: arg1a:
da
[freemod: comma:
,]]
anoi
[arg7:
[argx: arg1a:
de
[freemod: comma:
,]]
eci
[argx: arg1a:
di]]]]]
```

Parser cache size 18163

Mi godzi da, anoifage de, a di, a do

```

[uttF: statement:
  [subject: argument:
    Mi]
  [predicate: barepred:
    [sentpred: predunit:
      godzi]
    [termset: argument:
      [arg8: arg1a:
        da
        [freemod: comma:
          ,]]
      anoifage
      [argument: arg8:
        [arg7: arg1a:
          de
          [freemod: comma:
            ,]]
        a
        [arg7: arg1a:
          di
          [freemod: comma:
            ,]]
        a
        [arg7: arg1a:
          do]]]]]]

```

Parser cache size 24499

Liu artomi purkaa lie atom, e nu leasri liu Zareiteizomeizi

```

[uttF: statement:
  [subject: argument:
    Liu artomi]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        purkaa]

```

```

        [termset: argument:
          lie
          atom, ]]
e
[backpred: barepred:
  [sentpred: predunit:
    nu
    leasri]
  [termset: argument:
    liu Zareiteizomeizi]]]]]

```

Parser cache size 32126

Liu atmo sui purkaa lie atom, e nu leasri liu Zateimeizo

```

[uttF: statement:
  [subject: argument:
    [arg1:
      Liu atmo]
    [freemod:
      sui]]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        purkaa]
      [termset: argument:
        lie
        atom, ]]
    e
    [backpred: barepred:
      [sentpred: predunit:
        nu
        leasri]
      [termset: argument:
        liu Zateimeizo]]]]]

```

Parser cache size 31081

Ie le to po purkaa ga nu fundi tu

```
[uttF: statement:
  [subject: argument:
    Ie
    [arg5: descriptn:
      le
      to
      [descpred: predunit:
        po
        [sentence: predunit:
          purkaa]]]]
  [predicate: markpred:
    ga
    [barepred:
      [sentpred: predunit:
        nu
        fundi]
      [termset: argument:
        tu]]]]
```

Parser cache size 20505

Liu protoni, purkaa lie proton, e nu leasri liu peireizoteizonei

```
[uttF: statement:
  [subject: argument:
    Liu protoni, ]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        purkaa]
      [termset: argument:
        lie
        proton, ]]]
```

```
e
[backpred: barepred:
  [sentpred: predunit:
    nu
    leasri]
  [termset: argument:
    liu peireizoteizeinei]]]]
```

Parser cache size 32722

Liu purkaa purkaa li purda kamla lu, e nu leasri liu peizureikeizaza

```
[uttF: statement:
  [subject: argument:
    Liu purkaa]
  [predicate: predicate2:
    [backpred: barepred:
      [sentpred: predunit:
        purkaa]
      [termset: argument:
        [arg1: LI:
          li
          [utterance0: sentpred:
            [senpred3: predunit:
              purda]
            [despredD: predunit:
              kamla]]]
          lu]
        [freemod: comma:
          ,]]]
    ]]]
```

```
e
[backpred: barepred:
  [sentpred: predunit:
    nu
    leasri]
  [termset: argument:
    liu peizureikeizaza]]]]
```

Parser cache size 34848

Ga seidjo lea no nu trecymro bekti ji vi lo rardza ga lea surpernova

```
[uttF: statement:
  Ga
  [barepred:
    [sentpred: predunit:
      seidjo]
    [termset: argument:
      [arg1a: descriptn:
        lea
        [descpred: despredE:
          [despredD: predunit:
            no
            [predunit1:
              nu
              trecymro]]
          [despredD: predunit:
            bekti]]]
      [argmod: argmod1:
        ji
        [modifier: mod1:
          vi
          [argument1: descriptn:
            lo
            [descpred: predunit:
              rardza]]]]]]]
    ga
    [subject: argument:
      lea
      [descpred: predunit:
        surpernova]]]
```

Parser cache size 36008

Ga gudbi lo cutci lopo pucblo lo naldi ga lo mroza

```
[uttF: statement:
  Ga
  [barepred:
    [sentpred: predunit:
      gudbi]
    [termset: terms:
      [argumentA: argument:
        lo
        [descpred: predunit:
          cutci]]
      [argumentB: argument:
        lo
        po
        [sentence: barepred:
          [sentpred: predunit:
            pucblo]
          [termset: argument:
            lo
            [descpred: predunit:
              naldi]]]]]]]]
  ga
  [subject: argument:
    lo
    [descpred: predunit:
      mroza]]]
```

Parser cache size 22929

Na levi delnai ga bilti lo blabi flora ga rui re lea floryclu lo redro

```
[uttF: statement:
  [modifiers: modifier:
    Na
    [argument1: descriptn:
```



```

        le
        vi
        [descpred: predunit:
          delnai]]]
[gasent: gasent1:
  ga
  [barepred:
    [sentpred: predunit:
      bilti]
    [termset: argument:
      lo
      [descpred: despredE:
        [despredD: predunit:
          blabi]
        [despredD: predunit:
          flora]]]]]
  ga
  [subject:
    [modifiers: modifier:
      rui
      [argument1: arg3:
        re
        [arg2: descriptn:
          lea
          [descpred: predunit:
            floryclu]]]]]
    [argument: descriptn:
      lo
      [descpred: predunit:
        redro]]]]]

```

Parser cache size 32791

>>>