```
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
%                                                                       %
%    File:           chukchi_deponents.dtr                              %
%    Purpose:        This is a formal account of the antipassive        %
%                    related deponency in Chukchi and is a              %
%                    deliverable of the ESRC funded project             %
%                    'Extended Deponency' (RES-000-23-0375)             %
%    Author:         Dunstan Brown June 06                              %
%    Email:          d.brown@surrey.ac.uk                               %
%    Address:        University of Surrey, Guildford GU2 7XH            %
%    Documentation: Chukchi_report.pdf                                  %
%    Related files: chukchi_deponents_theorem.pdf                       %
%    Version:        1.19                                               %
%                                                                       %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

% This Network Morphology/DATR fragment addresses the issue of         %
% how there can be a mismatch between the syntactic and                %
% morphological properties of a Chukchi verb. This theory assumes      %
% that syntax passes to the morphology  the person and number          %
% information associated with the absolutive and ergative              %
% arguments. This syntactic information takes the form of ‘query       %
% nodes’ at the end of this file for each of the 92 word forms.        %
% Query node Word92 (at the end of this file), for example, looks      %
% like this:                                                           %
%                                                                      %
% Word92:                                                              %
%    <> == Wiriŋ                                                       %
%    <syn transitivity> == trans                                       %
%    <syn tns> == pres-2                                               %
%    <syn abs-arg person> == third                                     %
%    <syn abs-arg number> == plural                                    %
%    <syn erg-arg person> == third                                     %
%    <syn erg-arg number> == plural.                                   %
%                                                                      %
% While it inherits from the one lexical entry Wiriŋ ‘defend’,         %
% this node should otherwise be interpreted as a query which           %
% syntax makes to morphology, "Give me the third plural on third       %
% plural form of the verb." Hence, the status of the query nodes       %
% is different from the parts of the fragment which come before        %
% them. They are an approximation of what would be generated           %
% by syntax.                                                           %
%                                                                      %
% The theory itself makes use of a distinction between                 %
% syntactic transitivity and morphological transitivity.               %
% Deponency comes about where syntactically transitive verbs           %
% are morphologically antipassive (using either tku or ine).           %
% In addition to the information provide by syntax about               %
% absolutive and, for transitives, ergative arguments, the theory      %
% requires the following distinctions:                                 %
%                                                                      %
%   <syn subj> - the subject                                           %
%   <syn obj> - the object                                             %
%   <syn arg1>                                                         %
%                                                                      %
% The theory determines the <syn subj> values for person and           %
% number on the basis of syntactic transitivity: if the verb           %
% is intransitive the values will be identical with those for          %
% <syn abs-arg>. If the verb is transitive the values will be          %
% identical with those for <syn erg-arg>.                              %
%                                                                      %
```

```
% For <syn obj> if the verb is intransitive  the values will be     %
% ‘undefined’. If the verb is transitive, the  values will be       %
% indentical with those for <syn abs-arg>.                          %
%                                                                   %
% By default <syn arg1> has values identical with those for         %
% <syn abs-arg>. However, if there is a mismatch such that          %
% <syn transitivity> is ‘trans’, but <mor transitivity> is          %
% ‘antipass1’ (ine) or ‘antipass2’ (tku), then <syn arg1> has       %
% values identical with <syn erg-arg>.                              %

% Additional comments precede the nodes to which they refer.        %

% The node VERB: if information is not provided by this node, then  %
% there is a default to the ‘undefined’ value. The syntactic        %
% category of verbs <syn cat> is ‘verb’. The underspecified path    %
% <syn> will determine the values for both <syn subj> and           %
% <syn obj>. It does this by evaluating <syn transitivity> and      %
% looking up the answer at ARGS.                                    %
%                                                                   %
% <syn arg1>’s values are determined by evaluating                  %
% <syn transitivity> and looking up the answers at ARG_1.           %
%                                                                   %
% <mor transitivity> is determined by evaluating <syn transitivity> %
% and the person value of the ergative argument, and looking up the %
% answer at the node VOICE. Recall that in the typological database %
% for Chukchi person is the conditioning feature for deponency.     %
%                                                                   %
% The stem (<stem>) is by default the same as the root (<root>).    %
% The antipassive 1 stem <stem antipass1> prefixes ine- to the      %
% root. The antipassive 2 stem affixes tku at the node AFFIX_TKU.   %
% (The result of affixation will depend on other information.)      %



VERB:
    <> == undefined
    <syn cat> == verb
    <syn> == ARGS:<syn "<syn transitivity>" >
    <syn arg1> == ARG_1:<syn "<syn transitivity>" >
    <mor> == MOR_VERB
    <mor transitivity> == VOICE:< "<syn transitivity>"
                            subj "<syn erg-arg person>" >
    <stem> == "<root>"
    <stem antipass1> == ine - "<root>"
    <stem antipass2> == AFFIX_TKU.

% The node ARGS: if information is not provided by this node, then  %
% there is a default to the ‘undefined’ value. This node specifies  %
% that intransitive subjective and transitive object share          %
% identical values with the absolutive argument. The transitive     %
% subject shares identical values with the ergative argument, and   %
% the two antipassives behave as intransitives, sharing person      %
% and number values in the same way as the intransitive.            %



ARGS:
    <> == undefined
    <syn intrans subj> == "<syn abs-arg>"
    <syn trans obj> == "<syn abs-arg>"
    <syn trans subj> == "<syn erg-arg>"
```

```
    <syn antipass1> == <syn intrans>
    <syn antipass2> == <syn intrans>.

% The node ARG_1: if information is not provided by this node, then %
% there is a default to the 'undefined' value. This node specifies  %
% the values for <syn arg1>. For the intransitive and antipassives  %
% the values of <syn arg1> are always identical with those of the   %
% intransitive subject.                                             %
%                                                                   %
% If the verb is syntactically transitive, however, then the value  %
% of <syn arg1> will depend on the morphological transitivity. If   %
% a transitive verb is <mor antipass1> or <mor antipass2> (i.e.     %
% (there is a mismatch) then the values of <syn arg1> will be       %
% identical to those of the transitive subject, which will be the   %
% ergative argument values, of course.                              %
%                                                                   %
% The number of arg1 if a transitive is morphologically an ine      %
% antipassive is determined by the equation with LHS                %
% <mor antipass1 number>. The person and number of the ergative     %
% argument are evaluated, as well as the person of the absolutive   %
% argument. If these evaluate to <mor subj third sing on obj third> %
% then the number of <syn arg1> will be that of the object.         %
% Otherwise (i.e. if they evaluate to <mor subj>), then the number  %
% value of <syn arg1> will be the number of the subject. That is,   %
% <syn subj number>. This accounts for the small corner of the      %
% syntactically transitive-morphologically antipassive deponency    %
% where the antipassive form agrees with the number of the third    %
% person object.                                                    %
% Note: the evaluable path on the RHS of the equation with LHS      %
% <mor antipass1 number> has added attributes subj, on and obj.     %
% This has been done with the intention of enhancing the            %
% readability of the paths to which the RHS evaluates.              %
% That is, <mor subj third sing on obj third> is considered more    %
% readable than <mor third sing third>.                             %
%                                                                   %
% Finally, if a syntactically transitive verb is morphologically    %
% transitive (i.e. <mor trans>), then <syn arg1> will have the      %
% person and number values of the object.                           %
%                                                                   %

ARG_1:
    <> == undefined
    <syn antipass1> == ARGS:<syn intrans subj>
    <syn antipass2> == ARGS:<syn intrans subj>
    <syn intrans> == ARGS:<syn intrans subj>
    <syn trans> == <mor "<mor transitivity>" >
    <mor antipass1> == ARGS:<syn trans subj>
    <mor antipass2> == ARGS:<syn trans subj>
    <mor antipass1 number> ==
                   <mor subj "<syn erg-arg person>"
                             "<syn erg-arg number>" on
                         obj "<syn abs-arg person>" >
    <mor subj third sing on obj third> == "<syn obj number>"
    <mor subj> == "<syn subj number>"
    <mor trans> == ARGS:<syn trans obj>.

% The node VOICE: this is used to determine morphological           %
% transitivity. By default (i.e. <>) this is the same as syntactic  %
% transitivity. Only if a verb is transitive is further evaluation  %
% required. If a verb is transitive (i.e. <trans>), but does not    %
% have a second person subject or third person subject, then the    %
```

```
% TAM (i.e. <tns>) of the verb is evaluated.                          %
%                                                                    %
% If the verb has a second person subject, then the person and       %
% number of the absolutive argument are evaluated and the value      %
% looked up at the node SECONDONTHIRDSUBJ.                           %
%                                                                    %
% If the verb has a third person subject, then the person and        %
% number of the absolutive argument, as well as the number value of  %
% the ergative argument, are evaluated, and the value looked up at   %
% the node SECONDONTHIRDSUBJ.
% Note: the use of the attributes on, if and third in the evaluable  %
% paths is again intended to enhance the readability of the paths    %
% to which they evaluate.                                            %
%                                                                    %

VOICE:
    <> == "<syn transitivity>"
    <trans> == TNS:< "<syn tns>" >
    <trans subj second> ==
      SECONDORTHIRDSUBJ:<on "<syn abs-arg person>"
                            "<syn abs-arg number>" >
    <trans subj third> ==
      SECONDORTHIRDSUBJ:<on "<syn abs-arg person>"
                            "<syn abs-arg number>"
                         if third "<syn erg-arg number>" >.

% The node SECONDONTHIRDSUBJ: this is used to determine              %
% morphological transitivity of second and third person subjects,    %
% mainly on the basis of the absolutive argument information         %
% evaluated by VOICE, but also on the basis of the ergative          %
% argument number, if if that is third person.                       %
%                                                                    %
% In the absence of other information (i.e. <>), TAM (i.e.           %
% <syn tns>) is evaluated and the value looked up at the node TNS.   %
%                                                                    %
% If the object is first plural, the verb is morphologically         %
% antipass2 (tku).                                                   %
%                                                                    %
% If the object is first singular, then the verb is morphologically  %
% antipass1 (ine).                                                   %
%                                                                    %
% If the object is first singular and the subject third plural,      %
% morphological transitivity is the same as the syntactic            %
% transitivity (i.e. there is no mismatch).                          %
%                                                                    %
% If the object is first plural and the subject third person,        %
% morphological transitivity is the same as the syntactic            %
% transitivity (i.e. there is no mismatch).                          %
%                                                                    %
% If the object is second person, then  morphological transitivity   %
% is the same as the syntactic transitivity (i.e. there is no        %
% mismatch).                                                         %
%                                                                    %

SECONDORTHIRDSUBJ:
    <> == TNS:< "<syn tns>" >
    <on first plural> == antipass2
    <on first sing> == antipass1
    <on first sing if third plural> == "<syn transitivity>"
    <on first plural if third> == "<syn transitivity>"
    <on second> == "<syn transitivity>".
```

```
% The node TNS: this is used to determine morphological             %
% transitivity which is determined by TAM. Recall that for the      %
% and third person subject paradigms, the person feature takes      %
% precedence.                                                       %
%                                                                   %
% Morphological transitivity defaults to syntactic transitivity.    %
% If <tns> evaluates to present 2 (pres-2) then syntactically       %
% transitive verbs will be antipass1 (ine).                         %
% However, if there is a third person object (<pres-2 on third>),   %
% then morphological transitivity will only be antipass1, if the    %
% subject is singular or second person plural. Otherwise it will    %
% default to syntactic transitivity via the empty path <>.          %

TNS:
    <> == "<syn transitivity>"
    <pres-2> == antipass1
    <pres-2 on third> == <if subj "<syn erg-arg number>"
                                  "<syn erg-arg person>" >
    <if subj sing> == antipass1
    <if subj plural second> == antipass1.


% The node AFFIX_TKU: this says that tku is suffixed by default,    %
% but prefixed in the past 1 tense and <syn arg1> is first          %
% person.                                                           %


AFFIX_TKU:
    <tku> == tku
    <suffix> == "<root>" - <tku>
    <prefix> == <tku> - "<root>"
    <stem antipass2> == <suffix>
    <stem antipass2 past-1 first> == <prefix>.


% The node MOR_VERB: this says that a word (<mor word>) consists    %
% of a prefix, a stem and a suffix. The form of the prefix requires %
% evaluation of TAM (<tns>), subject person, subject number,        %
% morphological transitivity, object person and object number.      %
% (Sometimes, in the case of intransitives and real antipassives,   %
% the object number will be undefined.)                             %
%                                                                   %
% The form of the stem depends on morphological transitivity, TAM   %
% and the person of <syn arg1>. Recall, that <syn arg1> is usually  %
% the absolutive argument, but is the ergative argument             %
% when verbs are syntactically transitive but morphologically       %
% antipassive. It should also be noted that the form of the stem    %
% is either the same as the root, or will involve affixation of the %
% antipassive markers. The information on morphological             %
% transitivity, TAM and <syn arg1 person> is required for when the  %
% the stems are antipassive, either purely morphologically or both  %
% morphologically and syntactically.                                %
%                                                                   %
% Further information related to the node MOR_VERB is given before  %
% the relevant equations.                                           %


MOR_VERB:
    <mor> ==
    <mor word> == "<mor prefix
```

```
                    "<syn tns>"
                    "<syn subj person>"
                    "<syn subj number>"
                    "<mor transitivity>"
                    "<syn obj person>"
                    "<syn obj number>" >"

                "<stem
                    "<mor transitivity>"
                    "<syn tns>"
                    "<syn arg1 person>" >"


                "<mor suffix
                    "<syn tns>"
                    "<syn arg1 person>"
                    "<syn arg1 number>"
                    "<mor transitivity>"
                    "<syn erg-arg person>"
                    "<syn erg-arg number>" >"


% MOR_VERB (cont): the specification of prefixes is self-explanatory. %
% The attributes follow the order defined by the evaluation for       %
% <mor prefix> given above: <syn tns>, <syn subj person>,             %
% <syn subj number>, <mor transitivity>, <syn obj person>,            %
% <syn obj number>.                                                   %


    <mor prefix pres-2> == n-
    <mor prefix past-1 first sing> == t-
    <mor prefix past-1 first plural> == mət-
    <mor prefix past-1 third plural trans> == ne-
    <mor prefix past-1 third sing trans> ==
                                        <mor prefix past-1 third plural trans>
    <mor prefix past-1 third sing trans third> ==
    <mor prefix pres-2 third plural trans first plural> == ɤe-


% MOR_VERB (cont): the specification of suffixes is self-explanatory. %
% The attributes follow the ordered defined by the evaluation for     %
% <mor suffix> given above: <syn tns>, <syn arg1 person>,             %
% <syn arg1 number>, <mor transitivity>, <syn obj person>,            %
% <syn erg-arg person>, <syn erg-arg number>.                         %

    <mor suffix past-1> == -ɤʔi
    <mor suffix past-1 first sing> == -ɤʔek
    <mor suffix past-1 first plural> == -mək
    <mor suffix past-1 second plural> == -tək
    <mor suffix past-1 third plural> == -ɤʔet

% MOR_VERB (cont): transitive suffixes                                %

    <mor suffix past-1 second sing trans> == -ɤət


    <mor suffix past-1 third sing trans> == -ɤʔen
    <mor suffix past-1 third plural trans> == -net
    <mor suffix past-1 third sing trans second plural> == -tkə
```

```
    <mor suffix past-1 third plural trans second plural> ==
                                <mor suffix past-1 third sing trans second plural>
    <mor suffix past-1 first sing trans> == -ʏəm
    <mor suffix past-1 third sing trans third sing> == -nin
    <mor suffix past-1 third plural trans third sing> == -ninet

% MOR_VERB (cont): pres-2 suffixes                                        %


    <mor suffix pres-2 first sing> == -iʏəm
    <mor suffix pres-2 first plural> == -muri
    <mor suffix pres-2 second sing> == -iʏət
    <mor suffix pres-2 second plural> == -turi
    <mor suffix pres-2 third sing> == -qin
    <mor suffix pres-2 third plural> == -qinet.


% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
%                                                                       %
%                                LEXICAL ENTRY                          %
%                                                                       %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %


Wiriŋ:
   <> == VERB
   <gloss> == defend
   <root> == wiriŋ.


% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
%                                                                       %
%                            SYNTACTIC QUERY NODES                      %
%                                                                       %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

% As we noted at the beginning of this file, while they inherit        %
% from the one lexical entry Wiriŋ 'defend', these nodes should        %
% otherwise be interpreted as a query which syntax makes to            %
% morphology. Hence, the status of the query nodes                      %
% is different from the parts of the fragment which come before        %
% them. They are an approximation of what would be generated           %
% by syntax.                                                            %

Word1:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == sing.



Word2:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == sing.
```

```
Word3:
    <> == Wiriŋ


    <syn transitivity> == intrans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == plural.



Word4:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == plural.



Word5:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == sing.



Word6:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == sing.



Word7:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == plural.



Word8:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == plural.



Word9:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == past-1
```

```
    <syn abs-arg person> == third
    <syn abs-arg number> == sing.


Word10:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == sing.


Word11:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == plural.


Word12:
    <> == Wiriŋ
    <syn transitivity> == intrans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == plural.


Word13:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == sing.


Word14:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == sing.


Word15:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == plural.


Word16:
```

```
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == plural.


Word17:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == sing.


Word18:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == sing.


Word19:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == plural.


Word20:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == plural.


Word21:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == sing.


Word22:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == sing.
```

```
Word23:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == plural.


Word24:
    <> == Wiriŋ
    <syn transitivity> == antipass1
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == plural.


Word25:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == sing.


Word26:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == sing.


Word27:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == plural.


Word28:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == plural.


Word29:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == sing.
```

```
Word30:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == sing.


Word31:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == plural.


Word32:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == plural.


Word33:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == sing.


Word34:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == sing.


Word35:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == plural.


Word36:
    <> == Wiriŋ
    <syn transitivity> == antipass2
    <syn tns> == pres-2
```

```
    <syn abs-arg person> == third
    <syn abs-arg number> == plural.


Word37:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == sing
    <syn erg-arg person> == second
    <syn erg-arg number> == sing.


Word38:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == sing
    <syn erg-arg person> == second
    <syn erg-arg number> == sing.


Word39:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == sing
    <syn erg-arg person> == second
    <syn erg-arg number> == plural.


Word40:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == sing
    <syn erg-arg person> == second
    <syn erg-arg number> == plural.


Word41:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.


Word42:
```

```
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.


Word43:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word44:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word45:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == plural
    <syn erg-arg person> == second
    <syn erg-arg number> == sing.


Word46:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == plural
    <syn erg-arg person> == second
    <syn erg-arg number> == sing.


Word47:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1


    <syn abs-arg person> == first
```

```
    <syn abs-arg number> == plural
    <syn erg-arg person> == second
    <syn erg-arg number> == plural.


Word48:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == plural
    <syn erg-arg person> == second
    <syn erg-arg number> == plural.


Word49:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.


Word50:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.


Word51:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == first
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word52:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == first
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.
```

Word53:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == sing
    <syn erg-arg person> == first
    <syn erg-arg number> == sing.

Word54:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == sing
    <syn erg-arg person> == first
    <syn erg-arg number> == sing.

Word55:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == sing
    <syn erg-arg person> == first
    <syn erg-arg number> == plural.

Word56:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == sing
    <syn erg-arg person> == first
    <syn erg-arg number> == plural.

Word57:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.

Word58:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2

```
    <syn abs-arg person> == second
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.


Word59:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word60:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word61:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == plural
    <syn erg-arg person> == first
    <syn erg-arg number> == sing.


Word62:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == plural
    <syn erg-arg person> == first
    <syn erg-arg number> == sing.


Word63:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == plural
    <syn erg-arg person> == first
    <syn erg-arg number> == plural.
```

```
Word64:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == plural
    <syn erg-arg person> == first
    <syn erg-arg number> == plural.


Word65:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.


Word66:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.


Word67:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == second
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word68:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == second
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word69:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
```

```
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == first
    <syn erg-arg number> == sing.


Word70:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == first
    <syn erg-arg number> == sing.


Word71:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == first
    <syn erg-arg number> == plural.


Word72:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == first
    <syn erg-arg number> == plural.


Word73:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == second
    <syn erg-arg number> == sing.


Word74:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == second
    <syn erg-arg number> == sing.
```

Word75:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == second
    <syn erg-arg number> == plural.

Word76:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == second
    <syn erg-arg number> == plural.

Word77:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.

Word78:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.

Word79:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.

Word80:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third

```
    <syn abs-arg number> == sing
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word81:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == first
    <syn erg-arg number> == sing.


Word82:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == first
    <syn erg-arg number> == sing.


Word83:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == first
    <syn erg-arg number> == plural.


Word84:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == first
    <syn erg-arg number> == plural.


Word85:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == second
    <syn erg-arg number> == sing.
```

Word86:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == second
    <syn erg-arg number> == sing.

Word87:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == second
    <syn erg-arg number> == plural.

Word88:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == second
    <syn erg-arg number> == plural.

Word89:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == past-1
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.

Word90:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == sing.

Word91:
    <> == Wiriŋ

    <syn transitivity> == trans
    <syn tns> == past-1

```
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.


Word92:
    <> == Wiriŋ
    <syn transitivity> == trans
    <syn tns> == pres-2
    <syn abs-arg person> == third
    <syn abs-arg number> == plural
    <syn erg-arg person> == third
    <syn erg-arg number> == plural.




# show

<mor word>

<syn transitivity>

<syn tns>

<syn subj person>

<syn subj number>

<syn obj person>

<syn obj number>

<syn abs-arg person>

<syn abs-arg number>

<syn erg-arg person>

<syn erg-arg number>

<mor transitivity>

<syn arg1 person>

<syn arg1 number>.




# hide

MOR_VERB

SECONDORTHIRDSUBJ

ARG_1
```

ARGS

AFFIX_TKU

TNS

VERB

VOICE

Wiriŋ.