Group	: Computational Linguistics
Topic	: formalism

Title	: **Transfer Rule Notation**
Author	: René Leermakers

Doc.Nr.	: 0253
Date	: 19-02-88
Status	: concept
Supersedes	: -
Distribution	: Project
Clearance	: Project
Keywords	: Transfer, notation

Institute for Perception Research

Contents

1 Introduction

This document describes the syntax of the file 'LANGUAGE':TRANSFERRULES.TRANS. This file defines the relation between Mrules and ILrules. The values of rule parameters are in principle explicated. If a rule (mrule or ilrule) parameter is not mentioned, a default translation takes place. These default parameter translation are defined in a separate section of the file.

The rule compiler translates the transfer rules to pascal code comprising the modules LDANILRULES and LDGENILRULES. In order to check the rules, build the targets" 'language:'ldanilrules.obj" and "'language':ldgenilrules.obj".

2 Syntax

```
UTT = LANGVERSION.[PARAMETERSTRANS].RULESTRANS

PARAMETERSTRANS = "PARAMETER"."TRANSLATION"."<".{PARAMETERTRANS}.">"

PARAMETERTRANS = PARAMNAME.":".TYPENAME."<"."-".">".PARAMNAME.":".TYPENAME
                  ."=".".(".ELEMENTTRANS.{",".ELEMENTTRANS}.")"

ELEMENTTRANS = (VALUENAME|SETVALUENAME).{"|"(.VALUENAME|SETVALUENAME)}.
                ["<"]."-".[">"].
               (VALUENAME|SETVALUENAME).{"|"(.VALUENAME|SETVALUENAME)}

LANGVERSION = "LANGUAGE" . ":" . IDENTIFIER

RULESTRANS = "RULE"."TRANSLATION"."<" . {RULETRANS} . ">"

RULETRANS = MRULENAME.["{".{ANDCOND}."}"].
              {(","|"|").MRULENAME.["{".{ANDCOND}."}"]}.
            ["<"]."-".[">"].
            ILRULENAME.["{".{ANDCOND}."}"]
              {(","|"|").ILRULENAME.["{".{ANDCOND}."}"]}.
ANDCOND = ORCOND.{ANDCAT.ORCOND}

ORCOND = ELEMENTARYCOND.{ORCAT.ELEMENTARYCOND}

ELEMENTARYCOND = ("(".ANDCOND.")"|PASCALEXPR)
```

```
  PASCALEXPR = EQUALS|SETBOOLEAN|NIBOOLEAN

  EQUALS = PARAMNAME.("="|("<".">")).(VALUENAME|SETVALUENAME)

  SETBOOLEAN = ["NOT"].["("].PARAMNAME.("IN"|"<"."="|">"."=").SETVALUENAME.[")"]

  NIBOOLEAN = ["NOT"].["("].PARAMNAME."NI".VALUENAME.[")"]

  SETVALUENAME = "[".[VALUENAME.{",".VALUENAME}]."]"

  VALUENAME,PARAMNAME,TYPENAME,MRULENAME,ILRULENAME = IDENTIFIER
```

3 Example

```
LANGUAGE:DUTCH

PARAMETER TRANSLATION
   <
     param1:type1 <-> ilparam1:type2 = ( elt1 <- ilelt1, elt2 <-> ilelt2)
     param2:type2 <-> ilparam2:type3 = ( [elt1] <-> [ilelt1],
                              [elt2] -> [ilelt2]|[ilelt22],
                             [elt1,elt2]|[] <-> [ilelt1,ilelt2], [] <-> []
                           )
   >
RULE TRANSLATION
   <
     Mrule1{ par1 = val1
             par2 = val2
           }               <- rule1a{ ilpar2 = ilval2
                                      NOT (ilpar1 NI ilsetval1)
                                         }|
                             rule2b
     Mrule2a{ par1 = val1
             par2 = val2
           },
     Mrule2b               <-> rule2{ ilpar2 = ilval2
                                      }
     Mrule3{ par1 = val1
             par2 = val2
```

```
        }               <- rule3{ ilpar2 IN ilval2 AND
                                      ilpar3 <> TRUE
                                  }
>
```

4 Remarks

- In a transfer rule M-rules appear before the arrows, IL-rules after them.
- Separated by comma's or bar's, any number of M-rules and IL-rules (with their parameter specifications) may appear in a transfer rule.
- The parameter *level* is translated automatically. Hence, in each transfer rule either all or none of the (M and IL) rules can have this parameter.
- Parameter specifications may be simple or not simple. Simple means that it is a number of expressions of the form 'parname=value'. Note that the order of parameter name and its value is obligatory, i.e. 'value=parname' is wrong. Not-simple parameter specifications are boolean expressions build with the operators NOT, IN, NI(inverse of IN), =, <>, AND and OR.
- Double and single arrows are allowed in transfer rules. However, arrows may only point towards rules with simple parameter specifications.
- Each parameter of a rule with an arrow pointing at it, must either be mentioned explicitly in a (simple) parameter specification or appear in the parameter translation section. There it must appear once and only once.
- In the parameter translation section the types of the parameters must be mentioned. These types are the same ones as in the M-rules, in the case of Mrule parameters, or as in the interlingua definition, in the case of IL-rule parameters.
- In the parameter translation section, parameter values may be translated ambiguously. Also double and single arrows may be used. Each value of each parameter must at least have one translation.