TOWARD INTEGRATED DICTIONARIES FOR M(a)T:
motivations and linguistic organisation

presented for COLING-86
Bonn, 25-29 August 1986

by

Ch.BOITET & N.NEDOBEJKINE

GETA, BP 68
Université de Grenoble
38402 Saint-Martin-d'Hères, FRANCE

ABSTRACT

In the framework of Machine (aided) Translation systems, two types of lexical knowledge are used, "natural" and "formal", in the form of on-line terminological resources for human translators or revisors and of coded dictionaries for Machine Translation proper.

A new organization is presented, which allows to integrate both types in a unique structure, called "fork" integrated dictionary, or FID. A given FID is associated with one natural language and may give access to translations into several other languages.

The FIDs associated to languages L1 and L2 contain all information necessary to generate coded dictionaries of M(a)T systems translating from L1 into L2 or vice-versa. The skeleton of a FID may be viewed as a classical monolingual dictionary, augmented with one (or several) bilingual dictionary. Each item is a tree structure, constructed by taking the "natural" information (a tree) and "grafting" onto it some "formal" information.

Various aspects of this design are refined and illustrated by detailed examples, several scenarii for the construction of FIDs are presented, and some problems of organization and implementation are discussed. A prototype implementation of the FID structure is under way in Grenoble.

Key-words : Machine (aided) Translation, Fork Integrated Dictionary, Lexical Data Base, Specialized Languages for Linguistic Programming.

Abbreviations : M(a)T, MT, HAMT, MAHT, FID, LEXDB, SLLP.

INTRODUCTION

Integrated Machine (aided) Translation ("M(a)T") systems include two types of translator aids. First, there is a sort of translator "workstation", relying on a text processing system augmented with special functions and giving access to one or several "natural" on-line "lexical resources" [4,7], such as dictionaries, terminology lists or data banks, and thesauri. This constitutes the Machine Aided Human Translation ("MAHT") aspect. Second, there may be a true Machine Translation ("MT") system, which "lingware" consists of "coded" grammars and dictionaries. This is the (human aided) MT aspect, abbreviated as "HAMT", or simply "MT", because human revision is necessary even more for machine translations than for human translations.

The term "coded" doesn't only mean that MT grammars and dictionaries are written in Specialized Languages for Linguistic Programming ("SLLP"), but also that the grammatical and lexical information they contain is of a more "formal" nature. In some systems, the formal lexical information is a reduction (and perhaps an oversimplification) of the information found in usual dictionaries. But, in all sophisticated systems, it is far more detailed, and relies on some deep analysis of the language. Moreover, the access keys may be different: classical dictionaries are accessed by lemmas, while formal dictionaries may be accessed by morphs (roots, affixes...), lemmas, lexical units, and even other linguistic properties. In many systems written in ARIANE-78 [1], lemmas are not directly used.

Efforts have been made to devise data base systems for the natural or the formal aspect, separately. Multilingual terminological data bases, such as TERMIUM [5] or EURODICAUTOM, illustrate the first type.

On the other hand, the Japanese and the French National MT projects have developed specialized lexical data base systems ("LEXDB"), in which the (formal) information is entered, and from which MT dictionaries are produced. More precisely, there is a data base for each language (L), and for each pair of languages (L1,L2) handled by the MT system. From the first LEXDB, analysis and synthesis MT dictionaries for L are automatically constructed, while transfer dictionaries for (L1,L2) are produced from the second.

In an integrated M(a)T system, it would be useful to maintain the two types of dictionaries in a unique structure, in order to ensure coherency. This structure would act as a "pivot", being the source of the "natural" view as well as of the "formal" dictionaries. Moreover, it would be interesting, for the same reasons, to reduce the number of LEXDBs. With the technique mentioned above, there are n**2 for n languages.

The authors have begun a research along those lines in 1982 [6]. In 1985, this has led to a tentative (small-scale) implementation of a first prototype, adapted to the aims of a Eurotra contract.

At the time of revision of this paper, work on specification and implementation was being continued by a small team trying to construct a Japanese-French-English LEXDB, for a particular domain. This is why some details given in this paper are already obsolete. However, the spirit has remained the same.

The main idea of the new organization is to integrate both types of dictionaries in a unique structure, called "fork" integrated dictionary, or "FID". A given FID is associated with one natural language and may give access to translations into several other languages.

Hence, there would be only n FIDs for n languages. The form of the "natural" part has been designed to reflect the organization of current modern usual dictionaries. This is why we have limited ourselves to the "fork" architecture", and have not attempted to construct a unique structure for n languages.

In the first part, we present the "skeleton" of a FID item. Part II shows how to "graft" codes onto it, and discusses the nature and place of those codes. Finally, some problems of organization and implementation are discussed in part III. An annex gives a complete example for the lemmas associated with the lexical unit COMPTER.

I. USING A "NATURAL" SKELETON

After having studied the structures of several classical dictionaries, including LOGOS, LAROUSSE, ROBERT, HARRAP'S, WEBSTER, SACHS, etc., we have proposed a standard form for the "natural skeleton" of a FID item.

Items are accessed by the lemmas, but the notion of lexical unit ("LU", or "UL" in French) is present. Lemmas are "normal forms" of words (in English, infinitive for verbs, singular for nouns, etc.). A lexical unit is the main element of a derivational family, and is usually denoted by the main lemma of this family. Lexical units are useful in MT systems, for paraphrasing purposes.

1. SOME SIMPLE EXAMPLES

1.1. "atmosphère", "atmosphérique"

```
clé "atmosphère"
   lm cl N.F. ul -- base --
      constr 1 : NON QUANTIFIE
         raff 1 : ASTRONOMIE
            sens 1 :
               def "masse gazeuse qui entoure un astre"
               ex "l'atmosphère terrestre"
               dériv "atmosphérique" cl A
                                         schem RELATIF-A
               trad 1 :
                  ANG "atmosphere"
                  RUS "atmosfera"
                  ALM "Atmosphäre"
         raff 2 : FIGURE
            sens 2 :
               def "ambiance, climat moral"
               ex "une atmosphère déprimante"
               trad 2 :
                  ANG voir trad 1
                  RUS voir trad 1
                  ALM "Stimmung"

      constr 2 : QUANTIFIE
         sens 3 :
            def "unité de pression"
            ex "une pression de 2 atmosphères"
            trad 3 : voir trad 1
```

```
clé "atmosphérique"
   lm cl A. ul "atmosphère" cl orig N.F. voir ul sens 1
      sens
         def "relatif à l'atmophère"
         ex "perturbations atmosphériques"
         trad
            ANG "atmospheric"
            RUS "atmosfernyij"
            ALM "atmosphärisch"
```

1.2. "préméditer", "prémédité", "préméditation"

```
clé "préméditer"
   lm cl V.T.1 ul -- base "prémédit"
         dériv suff "ation" cl N.F. schem ACTION-DE
               direct PPAS  cl A.schem QUI-EST--
      sens
         def "décider, préparer avec calcul"
         ex "le pharmacien avait prémédité la rupture"
         ex "il avait prémédité de s'enfuir"
         trad
            ANG "premeditate" cl V.
            RUS "zamyishlitq"
            ALM "vorsessen"
```

```
clé "prémédité"
   lm cl A. ul "préméditer" cl orig V.T.1
         dériv direct PPAS
      sens
         def "qui est réalisé avec préméditation"
         ex "son crime fut prémédité"
         trad
            ANG "premeditate" cl A.
            RUS "prednamerennyij"
            ALM "vorsässlich"
```

```
clé "préméditation"
   lm cl N.F. ul "préméditer" cl orig V.T.1
         dériv suff "ation" schem ACTION-DE
      sens
         def "dessein réfléchi d'accomplir une action"
         ex "meurtre avec préméditation"
         trad
            ANG "premeditation"
            RUS "prednamerennostq"
            ALM "Vorsass"
```

1.3. Types of elements in the notation

There are three types of elements in the examples. Keywords are underlined. They show the articulation of the standard structure. In case of repetition at the same level, numbers are used (e.g. trad 1).

Identifiers are in uppercase (and should be in italic, but for the limitations of our printer). They correspond to the list of abbreviations which is usually placed at the beginning of a classical dictionary. They may contain some special signs such as "." or "-".

Strings are shown between double quotes. They correspond to the data. We use our "local" transcription, based on ISO-025 (French character set).

2. FORM OF AN ITEM

2.1. Keys, lemmas, lexical units

As illustrated above, an item may consist of several lemmas, because of possible ambiguities between two canonical forms (e.g. LIGHT-noun and LIGHT-adjective).

The corresponding LU is always given. The symbol "--" stands for the key of the item. Confusion should be avoided in the denotation of LUs. For example, for lemmas LIGHT, we could denote the LU correponding to the first (the noun) by "-- lm 1" or "-- cl N."

2.2. Constructions, refinements, meanings

The preceding items have been chosen for their relative simplicity. In general, a lemma may lead to several constructions, a construction to several refinements, each defined as a "meaning", for lack of a better word.

Further refinements may be added, to select various translations for a given meaning. The following diagram illustrates the idea.

```
+-----------------------------------------------------------+
! key                                                       !
! !___lemma                                                 !
! !   !___construction                                      !
! !   !         !___meaning/transl.___ANG_______construction!
! !   !                            !___RUS______construction!
! !   !                            !___ALM______construction!
! !   !___construction                                      !
! !        !___refinement                                   !
! !        !    !___meaning/transl.___ANG                   !
! !        !                       !   !___refinement       !
! !        !                       !   !    !___construction!
! !        !                       !   !___refinement       !
! !        !                       !        !___construction!
! !        !                       !___RUS______construction!
! !        !                       !___ALM______construction!
! !        !___refinement                                   !
! !             !___meaning/transl.___ANG_______construction!
! !                                !___RUS______construction!
! !                                !___ALM______construction!
! !___lemma_________meaning/transl.                         !
+-----------------------------------------------------------+
```

Intuitively, constraints are more local to the left than to the right. The presence of a construction may be tested in a sentence, but the notion of domain of discourse or of level of language is obviously more global.

The notion of construction is fundamental. In particular, predicative words cannot be translated in isolation, and it is necessary to translate expressions of the form P(x,y,z), P being the predicate and x, y, z its arguments, possibly with conditions on the arguments. Note that idioms or locutions are particular forms of constructions.

In general, refinements may be local or global. Local refinements often consist in restrictions on the semantic features of the arguments ("to count on somebody" vs. "to count on something"). Global refinements concern the domain, the style (level of discourse), or the typology (abstract, bulletin, article, ckeck-list...).

In our view, a meaning in L1 is translated by one or several constructions in L2.

We have then avoided to translate a meaning by a meaning, which might seem more logical. But this would have forced us to describe the corresponding cascade of constraints in L2. As a matter of fact, it is usually possible to reconstruct it, from the constraints in L1 and contrastive knowledge about L1 and L2. Hence, we follow the practice of usual dictionaries.

2.3. Translations (1--n): "fork" dictionaries

We have shown how to include in an item its translations into several target languages. Hence the term "fork". The "handle" of the item consists in all information concerning the source language (L1). In order for such an organization to work, we must have at least 2 such dictionaries, for L1 and L2, as no detailed information about L2 is included in the L1-based dictionary. This information may be found in the L2-based dictionary, by looking-up the appropriate item and locating the construction: the path from the key to the construction contains it.

3. FACTORIZATION AND REFERENCE

As seen in the examples, we introduce some possibilities of naming subparts of a given lemma, by simply numbering them (sens 3 refers to trad 1 in "atmosphère").

This allows not only to factorize some information, such as translations, but also to defer certain parts of the item. For example, translations might be grouped at the end of the (linear) writing of an item. The same can be said of the formal part of the information (see below).

II. GRAFTING FORMAL INFORMATION ("CODES")

1. PRINCIPLES

1.1. Attributes and classes

The formalized information may correspond to several distinct linguistic theories. Such a theory is defined by a set of formal attributes, each of a well-defined type. For example, the morphosyntactic class might be defined as a scalar attribute:

CATMS = (VERB, NOUN, ADJECTIVE, ADVERB, CONJUNCTION, etc.)

The gender might be defined as a set attribute:

GENDER = ens (MASCULIN, FEMININ, NEUTRE).

Each theory may give rise to several implementations (lingwares), each of them having a particular notation for representing these attributes and their values. Moreover, in a given lingware, the information relative to an item may be distributed among several components, such as analysis, transfer and synthesis dictionaries.

Usually, combinations of particular properties (or attribute/value pairs) are given names and called classes. For example, in ARIANE-78, there are the "morphological" and "syntactic" "formats", abbreviated as FTM and FTS, in the AM (morphological analysis) dictionaries. Special questionnaires, called "indexing charts", lead to the appropriate class, by asking global questions (vs. one particular question for each possible attribute).

1.2. Form of what is grafted

In the simplest case, there is one theory, and one corresponding lingware. The grafted part will be of the form:

```
app info properties in the theory
    code codes (classes and possibly basic properties)
```

The keyword app means "appended".

In a less simple case, there might be two theories, called A and B, of French. Suppose that there is an analyzer, FR1, and a synthesizer, FRA, corresponding to A, and two analyzers and a synthesizer (FR2, FR3, FRB), relative to B. The grafted part will be of the form:

```
app th A info properties in theory A
         code LS FR1 AM FTM CM001 FTS CS023
         code LC FRA ...    (LS for source language,
                            (LC for target language)
    th B info properties in theory B
         code LS FR2 AM FTM FORM3 FTS SEM25
         code LS FR3 ...
         code LC FRB ...
```

"AM" must be known as an introductor of codes for morphological analysis in ARIANE-78-based lingwares.

1.3. Where to graft: inheritance principle

Formal parts may be attached at all levels of an item, for factorization purposes. The information is supposed to be cumulated along a path from a key to a "meaning" or to a translation. If two bits of information are contradictory, the most recent one (rightmost in our diagrams) has preeminence.

Taking again the example of systems written in ARIANE-78, we may suggest to distribute the codes in the following fashion. One could attach:

- the morphological codes (FTM) and the "morphs" to the roots ("bases") or to the lemmas;
- the "local" syntaxo-semantic codes (FTS) to the lemmas or to the constructions;
- the "global" syntactic codes (concerning the typology) to the various levels of refinement;
- the codes concerning the derivations to the dériv parts, wherever they appear in the item.

2. AN EXAMPLE ("ATMOSPHERE")

```
clé "atmosphère"
  lm cl N.F. ul --
      base --
        app
          th A info FLEXN=S, MORPH="atmosphère",
                    DERIV="atmosphérique"
               code LS FR1 AM FTM FXN1
               code LC FRA GM FAF FXN1
          th B info FLEXN=ES, MORPH="atmosphér",
                    ALTER=GRAVE, SUF=IQUE
               code LS FR2 AM FTM FNESIQ
               code LC FRB GM FAF FNESIQ
    app th A info CATMS=NOUN, GENDER=FEMININ
             code LS FR1 AM FTS NCFEM
             code LC FRA GM FAF NCFEM
        th B info CAT=N, GNR=FEM, N=NC, AMBSEM=3
             code LS FR2 AM FTS NCFEM3
             code LC FRB GM FAF NCFEM
    constr 1 : NON QUANTIFIE
      raff 1 : ASTRONOMIE
        sens 1 :
          def "masse gazeuse qui entoure un astre"
          ex "l'atmosphère terrestre"
          dériv "atmosphérique" cl A
                                schem RELATIF-A
          trad 1 :
             ANG "atmosphere"
             RUS "atmosfera"
             ALM "Atmosphäre"
          app th A info SEM=STRUCT, SEM1=ASTRE,
                        DERPOT=NADJ, SCHEM=13
```

```
                    code LS FR1 AX FAF PNA
                    code LC FRA GX PAF PNA13
               th B info SEM=COLLECT, CLCT=FLUID,
                         SEM1=SPHERE, DERPOT=NA
                    code LS FR2 AX FAF PNA PAF COLF
                    code LC FRB GX FAF DERIQUE
      raff 2 : FIGURE
         sens 2 :
            def "ambiance, climat moral"
            ex "une atmosphère déprimante"
            trad 2 :
               ANG voir trad 1
               RUS voir trad 1
               ALM "Stimmung"
            app th A info SEM=ETAT, SEM1=ACTIVITE
                    code LS FR1 AX PAF SDETAT, V1ACT
                    code LC FRA ...
   constr 2 : QUANTIFIE
      sens 3 :
         def "unité de pression"
         ex "une pression de 2 atmosphères"
         trad 3 : voir trad 1
            app th A info SEM=UNITE
                    code LS FR1 AX PAF SOUNT
                    code LC FRA ...
               th B info SEM=UNITE, SEMZ=POIDS
                    code LS FR1 AX PAF SOUNT, VPPS
                    code LC FRB ...
```

3. CONSTRUCTION OF INTEGRATED DICTIONARIES

Suppose the natural skeleton of an item is obtained by using available dictionaries. There are two main methods for constructing the app parts.

First, one may begin by filling the info parts. This is the technique followed by the two afore-mentioned national projects. For this, people without special background in computer linguistics may be used. They fill questionnaires (on paper or on screen) asking questions directly related to the formal attributes. This information is checked and inserted in the info parts at the proper places, which are determined by knowing the relation between the "natural" information and the "theory".

In a second stage, programs knowing the relation between the theory and a particular lingware will fill the code parts.

The second methods tries to make better use of existing MT dictionaries. First, the relation between the elements of a lingware and the "natural" system is defined, and programs are constructed to extract the useful information from the MT dictionaries and to distribute it at the appropriate places. Then, knowing the relation between the "coded" information and the theory, info parts may be constructed or completed.

At the time this paper was revised, M.DYMETMAN was implementing such a program to construct a FID from our current Russian-French MT system. His results and conclusions should be the theme of a forthcoming paper.

Inconsistencies may be detected at various stages in the construction of a FID, and the underlying DB (data base) system must provide facilities for constructing checks, using them to locate incorrect parts, and modifying the item.

III. PROBLEMS OF DESIGN AND IMPLEMENTATION

The construction of an implemented "mock-up" has led us to identify some problems in the design, to wonder whether there is any available DBMS (data base management system) adequate for our purposes, and to ask what should be done about the representation of characters, in a multilingual setting.

1. RELATION BETWEEN NATURAL AND FORMAL INFORMATION

The relation between the formal information of a theory and the formal information of an implemented model of it (a lingware) is simple: the latter is a notational variant of (a subset of) the former.

By contrast, it is not so easy to define and use the relation between a formal theory and the "natural" information. The theory might ignore some aspects, such as phonology, or etymology, while it would use "semantic" categories (such as COUNTABLE, TOOL, HUMAN, PERSONNIFIABLE, CONCRETE, ABSTRACT...) far more detailed than the "natural" ones (SOMEBODY, SOMETHING...).

In order for the construction of such FID to be possible, we must at least ask that all "selective" information, which guides the choice of a meaning and of a translation, must in some sense be common to the natural and the formal systems.

Hence, these systems must have a certain degree of homogeneity. Dictionaries containing very little grammatical information (e.g. only the class) cannot be used as skeletons for FIDs integrating the lexical data base of a (lexically) sophisticated MT system.

Another problem is just how to express the relation between the systems, in such a way that it is possible:

- to reconstruct (part of) the skeleton of an item from the "coded" information;
- to compute (part of) the formal information on a path of the skeleton.

For the time being, we can write ad hoc programs to perform these tasks, for a particular pair of systems, but we have no satisfactory way to "declare" the relation and to automatically generate programs from it.

2. TYPE OF UNDERLYING DATA-BASE SYSTEM

P.Vauquois (a son of B.Vauquois) and D.Bachut have implemented the above-mentioned mock-up in Prolog-CRISS, a dialect of Prolog which provides facilities for the manipulation of "banks" of clauses. It is possible to represent directly the tree structure of an item by a (complex) term, making it easy to program the functions associated to a FID directly in Prolog.

However, Prolog is not a DBMS, and, at least with the current implementations of Prolog, a large scale implementation would be very expensive to use (in terms of time and space), or perhaps even impossible to realize.

As FIDs would certainly grow to at least 50000 items (perhaps to 200000 or more), it might be preferable to implement them in a commercially available DBMS system, such as DL1, SOCRATE, etc. A numeric simulation made by E. de Boussineau shows that a (1--2) FID of about 100000 lemmas could be implemented in a Socrate DB, of the network type, in one or two "virtual spaces". No experiment has yet been conducted to evaluate the feasibility of the method and its cost.

Other possibilities include relational and specialized DBMS systems. In a relational DBMS, each Socrate entity would give rise to a relation. Specialized DBMS have been developed for terminological data banks, such as TERMIUM or EURODICAUTOM. There is a general tool for building terminological DB, ALEXIS [3].

3. CHARACTER SETS

None of the above-mentioned systems provides facilities for handling multilingual character sets. Hence, all strings representing units of the considered natural languages, including the keys, must be represented by appropriate transcriptions.

This is clumsy for languages written in the Roman alphabet, and almost unacceptable for other languages, alphabetical or ideographic. Supposing that bit-map terminals and printers are available, two solutions may be envisaged:

- define appropriate ASCII or EBCDIC transcriptions, and equip the DBMS with corresponding interfaces;

- modify the DBMS itself to represent and handle several (possibly large) character sets. This is what has been done in Japan, where programming languages, text processing systems and operating systems have been adapted to the 16-bit JIS (or JES) standard.

CONCLUSION

We have presented and illustrated the new concept of FID, or Fork Integrated Dictionary. To our knowledge, this is the first attempt to unify classical and MT dictionaries. However, only a small mock-up has been implemented, and some problems of design and implementation have been detected. It remains to be seen whether large scale FIDs can be constructed and used in an operational setting.

ACKNOWLEDGMENT

We are grateful to ADI (French Agence de l'Informatique) and to the EC (European Community, EUROTRA project) to have given us the opportunity to start this research and to test some of our ideas.

-o-o-o-o-o-o-o-o-

REFERENCES

1. Ch.Boitet & N.Nedobejkine (1981), "Recent developments in Russian-French Machine Translation at Grenoble", Linguistics 19, 199-271 (1981).

2. Ch.Boitet, P.Guillaume, M.Quézel-Ambrunaz (1982), "ARIANE-78: an integrated environment for automated translation and human revision", Proceedings COLING82, North-Holland, Linguistic Series No 47, 19-27, Prague, July 82.

3. Société ERLI (1984), "ALEXIS: présentation générale et manuel d'utilisation", Soc. ERLI, Charenton, octobre 1984.

4. A.Melby (1982), "Multi-level translation aids in a distributed system", Proceedings COLING82, North-Holland, 215-220, Prague, July 82.

5. M.Perrier (1982), "Banque TERMIUM, gouvernement du Canada. Lexique BCF (budgétaire, comptable et financier", Bureau des traductions, Direction de la terminologie, section économique et juridique, Ottawa, juin 1982.

6. Ch.Boitet, N.Nedobejkine (1982), "Base lexicale: organisation générale et indexage", Projet ESOPE, ADI, rapport final, partie D, 1982..

7. D.E.Walker (1985), "Knowledge Resource Tools for Accessing Large Text Files", Proc. of the Conf. on Theoretical and Methodological Issues in Machine Translation of Natural Languages, Colgate Univ., Aug. 14-16, 1985.

ANNEX: "COMPTER"

```
clé "compter"
    lm cl V.T.1. ul --
       app 1        (no ":", hence see forward)
       base "compt" app 2
       constr 1 : QN.x -- QCH.y A QN.z
          app 3
          sens 1 :
             def "faire payer"
             trad 1 :
                ANG "charge" cl V.
                        cstrad S-O.x -- S-O.z FOR S-TH.y
                               S-O.x -- S-TH.y TO S-O.z
                   app 4
                RUS "zakhestq cl V.
                ALM "auszahlen" cl V.
                        cstrad J-D.x -- ETW.y J-M.z
                   app 5
       constr 2 : QN.x -- QN.y  POUR QN.z
          app 6   (further app parts suppressed)
                 QN.x -- QCH.y POUR QCH.z
          sens 2 :
             def "tenir pour"
             trad 2 :
               ANG "consider" cl V.
                 cstrad S-O.x -- S-O/S-TH.y AS S-O/S-TH.z
               RUS "skhitatq" cl V.
                 cstrad KTO.x -- KOGO/KHTO.y KEM/KHEM.z
               ALM "halten" cl V.
                 cstrad J-d.x -- J-N/ETW.y FUER J-N/ETW.z
       constr 3 : QN.x -- QN.y  PARMI QN.z
                  QN.y -- PARMI QN.z  POUR QN.x
                  QN.x -- QCH.y PARMI QCH.z
                  QCH.y -- PARMI QCH.z  POUR QN.x
          sens 3 :
             def "considérer comme faisant partie de"
             trad 3 :
                ANG "count" cl V.
           cstrad S-O.x -- S-O/S-TH.y AMOUNG S-O/S-TH.z
                RUS "skhitatq" cl V.
           cstrad KTO.x -- KOGO/KHTO.y SREDI KOGO/KHEGO.z
                ALM "zählen" cl V.
           cstrad J-D.x -- J-N/ETW.y ZU J-N/ETW.z
       constr 4 : QN.x -- INF/QUE+IND/SUR+QCH.y
          sens 4 :
             def "espérer"
             trad 4 :
                ANG "expect" cl V.
                 cstrad S-O.x -- TO+INF/THAT+IND/S-TH.y
                RUS "rasskhityivatq" cl V.
                 cstrad KTO.x -- INF/KHTO+IND/NA+KHTO.y
                ALM "hoffen" cl V.
                 cstrad J-D.x -- ZU+INF/DASS+IND/AUF+ETW.y
       constr 5 : QN.x -- SUR QN.y
          sens 5 :
             def "avoir confiance"
             trad 5 :
                ANG "rely" cl V.
                     cstrad S-O.x -- ON S-O.y
                RUS "polozhitqsya" cl V.
                     cstrad KTO.x -- NA KOGO.y
                ALM "zählen" cl V.
                     cstrad J-D.x -- AUF J-N.y
       constr 6 : QN.x -- AVEC QN/QCH.y
          sens 6 :
             def "prendre en considération"
             trad 6 :
                ANG "reckon" cl V.
                        cstrad S-O.x -- WITH S-O/S-TH.y
                RUS "skhitatqsya" cl V.
                        cstrad KTO.x -- S KEM/KHEM.y
                ALM "rechnen" cl V.
                        cstrad J-D.x -- MIT J-M/ETW.y
       constr 7 : QCH.x -- TANT-DE.y
          sens 7 :
             def "totaliser"
             ex  "la bibliothèque compte 1000 livres"
             trad 7 :
                ANG "count" cl V.
                        cstrad S-TH.x -- SO-MUCH.y
                RUS "naskhityivatq" cl V.
                        cstrad KHTO.x -- SKOLQKO.y
                ALM "zählen" cl V.
                        cstrad ETW.x -- SOVIEL.y
       constr 8 : QN/QCH.x -- QCH.y
          raff x.PERSONNE/INSTRUMENT & y.NOM-DE-MESURE
             sens 8 :
                def "mesurer, évaluer"
                trad 8 :
                   ANG "count" cl V.
                             cstrad S-O/S-TH.x -- S-TH.y
```

```
              RUS "otskhitatq" cl V.
                        cstrad KTO/KHTO.x -- KHTO.y
              ALM "rechnen" cl V.
                        cstrad J-D/ETW.x -- ETW.y
      raff x.PERSONNE/INSTRUMENT
        & y.NOM-COLLECTIF/PLURIEL-DENOMBRABLE
        sens 9 :
          def "dénombrer"
          ex  "compter les moutons"
          trad 9 :
              ANG "count" cl V.
                     cstrad S-O/S-TH.x -- S-O/S-TH.y
              RUS "skhitatq" cl V.
                     cstrad KTO/KHTO.x -- KOGO/KHTO.y
              ALM "zählen" cl V.
                     cstrad J-D/ETW.x -- J-N/ETW.y
    constr 9 : QN/QCH.x --
      raff x.PERSONNE/INSTRUMENT
        & -- DE-TETE/SUR-SES-DOIGTS/JUSQU'A
        sens 10 :
          def "énumérer"
          trad 10 : voir trad 9  (sans y)
      raff
        sens 11 :
          def "être important"
          trad 11 :
              ANG voir trad 10
              RUS "skhitatqsya" cl V.
                     cstrad NUZHNO -- S KEM/KHEM.x
              ALM "wichtig" cl A.
                     cstrad J-D/ETW.x -- SEIN
      raff x.PERSONNE
        sens 12 :
          def "regarder à la dépense"
          trad 12 :
              ANG "stingy" cl A.
                     cstrad S-O.x BE --
              RUS "yekonomnyij" cl A.
                     cstrad KTO.x (BYITQ) --
              ALM "sparsam" cl A.
                     cstrad J-D.x -- SEIN
    constr 10 : locut A -- DE QCH.x cl PREP.
      sens 13 :
        def "à partir de"
        trad 13 :
              ANG "reckoning" cl PREP.
                     cstrad -- FROM S-TH.x
              RUS "nakhinaya" cl PREP.

                     cstrad -- S KHEGO.x
              ALM "von" cl PREP.
                     cstrad -- ETW.x AN
app 1 : th A info CAT=V, EXPANS=(TRANS,INTRANS),
                       SEM=(ACTION,ETAT)
             code LS FR1 AM FTS VB  AX PAF VT1TR
                  LC FRB GX FAF VB
app 2 : th A info CONJUG=1GR
             code LS FR1 AM FTM VB1A
                  LC FRA GM FAF VB1A
app 3 : th A info PRED=ECHANGE, MODALITE=FACTITIF,
                  VL1=GN, VL2=AGN, VLO=GN
             code LS FR1 AX FAF SCHR11 PAF ECHFC
                  LC FRA GX PCP CSTR1 FAF SCHR11
app 4 : th A info ARG2=FOR, ARGINV=12
             code LS FR1 TL FAF XYFORZ PAF INV12
etc...
```

-o-o-o-o-o-o-o-o-