15
METAL

In 1989 the METAL system for German–English translation was introduced to the commercial market by the German electronics company Siemens of Munich. The product of many years of research stretching back to the mid 1960s, the METAL system is essentially based on the transfer approach; it is written in the Lisp programming language, and it represents one of the most advanced operational MT systems at the present time.

15.1 Historical background

The origins of the MT research for METAL go back to the establishment of the Linguistics Research Center (LRC) at the University of Texas (Austin, Texas) under the direction of Winfred Lehmann in 1961. Research on an MT system for German to English had already begun in 1959 with a contract from the US Army; but the major funder was the US Air Force Rome Air Development Center until 1979. Unlike many other MT groups at the time, the emphasis at LRC was long-term fundamental linguistic research. Basic research on English and German syntax led to the proposed development of a bidirectional transfer system, with reversibility of the syntactic transfer rules. This initial research phase ended in 1968.

A second phase of LRC research extended from 1970 to 1975 with the exploration of an interlingua-based approach, still concentrating on German and English. At an early stage, the LRC group convened a conference to assess the direction of MT after the ALPAC report of a few years earlier (section 1.3) and concluded that current linguistic research gave encouragement to the pursuit of deeper levels of analysis and to efforts towards interlingual representations. Like the contemporary CETA system at the University of Grenoble (section 13.1), the LRC interlingua design was interlingual only with respect to syntactic representations: translation of lexical items was handled by rules of lexical transfer.

In 1978 a new phase began with financial support from the Munich-based company Siemens, who became sole supporters from 1980. Siemens' motivation was both the need to increase the productivity of its own translation service and the desire to produce a translation system for others. The METAL system (the name originally comes from the acronym 'Mechanical Translation and Analysis of Language') changed from an interlingua design to an essentially transfer-based approach, not intended to operate fully automatically but to be augmented by sophisticated text-editing facilities and access to large terminology databanks, in particular Siemens' own TEAM database. In 1989 the first METAL system appeared on the market, for German–English translation, and is now installed at a number of large corporate users. Like Systran (Chapter 10), it is a system for mainframe computers; METAL runs on Symbolics™ 36-series Lisp machines, with batch processing and post-editing on workstations (Siemens SINIX™-based machines MX-2, MX-300 or MX-500), linked to printing facilities. There are as yet no plans for microcomputer-based versions.

The German–English version is to be followed by one for English into German and by systems for Dutch–French, French–Dutch, German–Spanish, German–French and German–Danish. Research on METAL is coordinated in Munich, with research on the English–German system conducted at LRC, on Dutch and French at the University of Leuven (Belgium), on German–Spanish at the University of Barcelona, and at the Handelshøjskole Syd in Kolding (Denmark) on German–Danish.

15.2 The basic system

METAL is intended to deal with high volumes of technical documentation, which is typically full of tables, diagrams, flow charts, etc. For economic operation a system has to be able to extract the material to be translated and to put it back in the correct locations. The basic METAL translation system has therefore been integrated into a chain of pre-translation processes, text acquisition and automatic deformatting, and post-translation processes, automatic reformatting, post-editing and transfer to typesetting and printing. These pre- and post-translation stages are performed at present on Siemens SINIX™ equipment, while the core translation system operates on a Symbolics™ Lisp machine.

The core processes of the METAL system are illustrated in Figure 15.1.

1. Text acquisition is from telecommunication links and various input facilities, e.g. magnetic tapes, floppy disks, OCRs, etc.

2. Deformatting programs separate textual data from diagrams, tables, charts, etc. and mark the latter for later re-insertion (stage 6 below); this stage includes identification and marking of 'translation units' (from single word headlines to complete sentences).

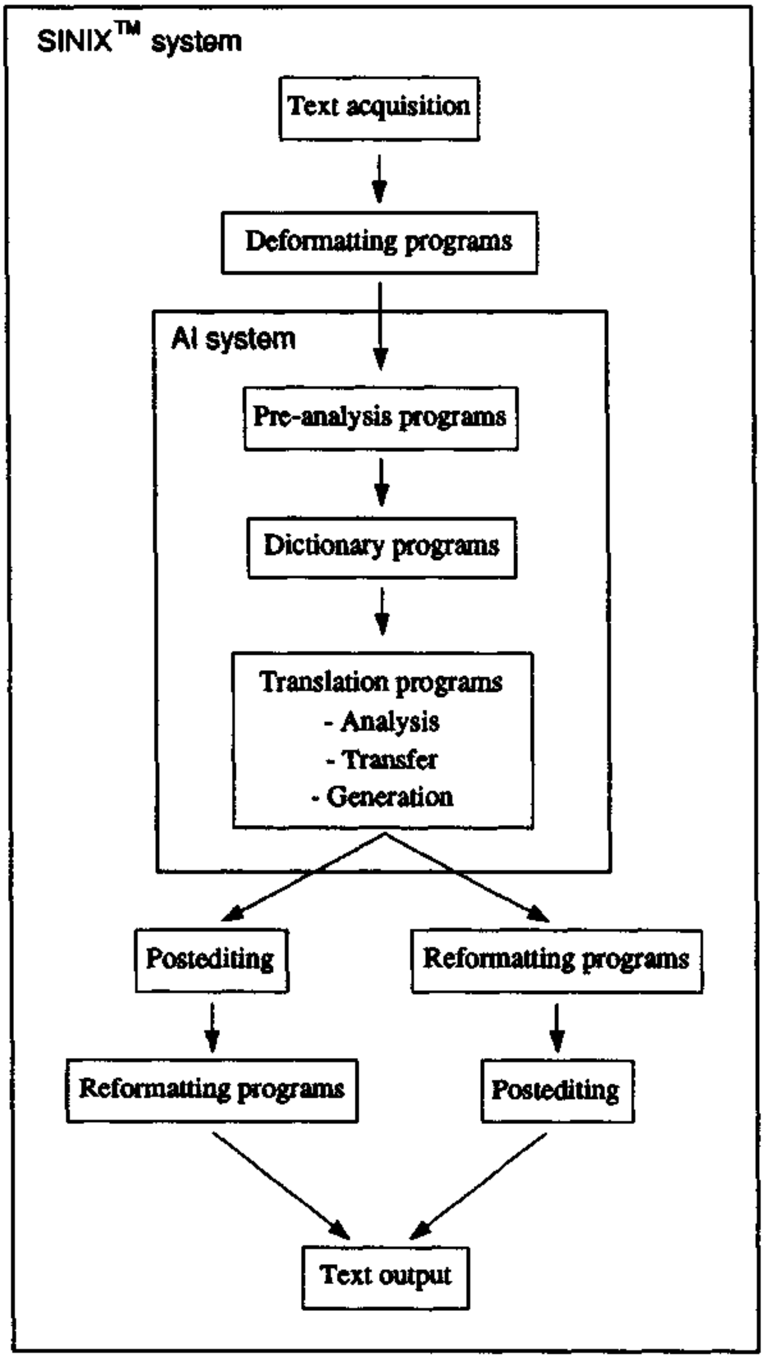


Figure 15.1 METAL core processes

3. Pre-analysis: search of the dictionary database, producing three lists: (a) unknown words listed alphabetically; (b) unknown compounds (likely to be frequent for German), with suggested possible English translations on the basis of components already present in the dictionary, e.g. for *Prüfangaben* it could suggest *testing specification*; and (c) a list of already known technical words, for the user to check for accuracy and appropriateness for the text in hand.

4. Dictionary programs: monolingual dictionaries for source and target languages contain morphological, syntactic and semantic information; the bilingual transfer dictionaries specify translational equivalences and include contextual information.

5. The METAL translation programs (described in section 15.3 below)

6. Reformatting: merging of textual and non-linguistic data

7. Post-editing: revision of translation can be performed either sentence by sentence (interlinearly) or with longer segments, and either with or without sight of the original text (on a split screen); revision can be done either before or after re-insertion of non-linguistic data (stage 6).

8. Output: word processing system, printer output, typesetting, transfer to other text processing and/or publishing systems.

The following description is based essentially on detailed accounts of the German–English system in the mid-1980s. In recent years, the METAL researchers have been introducing a number of modifications, largely in order to extend the system to more languages. These developments will be treated mainly in section 15.6 after descriptions of the databases and translation programs in the German–English version.

15.3 The linguistic databases

In the METAL system there is a clear separation between the operations of translation (analysis, transfer, generation) and the data which are called upon during operations. These data comprise monolingual lexical information for source and target languages, bilingual lexical information for a specific language pair, and sets of grammatical rules. The latter embrace rules applied at all stages of translation, to various kinds of linguistic objects (morphemes, words, phrases, clauses, sentences), and involving morphological, syntactic, and semantic features.

15.3.1 Dictionaries

The monolingual source and target dictionaries of METAL contain the basic morphological, syntactic and semantic information. METAL provides a hierarchy of vocabularies: the basic modules are the three lowest levels (function words, general vocabulary, and common technical vocabulary), which apply whatever the subject matter. To these standard modules users can add as many specialised glossaries as they wish and specify the order(s) in which they should be consulted. As well as subject-specific glossaries users can define country-specific glossaries (e.g. in order that German *Lastwagen* translates as *truck* in the United States and as *lorry* in Britain.)

Lexical entries in monolingual dictionaries are lists of features with values, e.g. root form, grammatical category, morphological variants, number, person, etc. Dictionaries include not only stem forms (typically word-initial elements) but also affixes (typically word-final elements); entries also include a 'preference' value (PRF) to enable selection between alternative analyses, indications of any lexical collocations (e.g. for discontinuous verb forms: *look up*, *zurückgeben*), and an indication of subject areas (TAG). Entries for nouns indicate their inflectional classes and add a semantic feature ('entity', 'living', 'commodity', etc.) used for co-occurrence restrictions; entries for verbs include case-frame specifications (in terms of 'deep case' relations: agent, target, benefactive, etc.), with attached features restricting the semantic values of arguments (e.g. *murder* requires the agent to be +RSP ('responsible')), the constituent type of arguments (e.g. noun phrase, prepositional phrase, clause, etc.), and the surface syntactic function of arguments (e.g. subject, object, *that*-complement, infinitival complement, etc.); entries for verbs include, finally, a specification of their valency pattern ('transitivity type'): intransitive with a single argument (agent), intransitive with two arguments (agent and location), transitive with two arguments (agent and target), etc.

The input of lexical entries is facilitated by the 'Intercoder' which interactively prompts the user for grammatical and translational information. The Intercoder includes a 'lexical default' program which accepts minimal information (e.g. root form and grammatical category) and automatically generates morphological variants and encodes syntactic features and values. Information in bilingual dictionaries can include the specification of particular frames, the presence of arguments of a certain semantic type, changing active phrases into impersonal constructions, adding and deleting elements, etc. The Intercoder maintains lexicon consistency and integrity with automatic validation programs which identify errors of form or syntax. This tool can be used both by system developers and, if desired, by end-users developing their own dictionaries. (Reproductions of Intercoder screens are shown in Figures 15.2 and 15.3.)

The monolingual dictionaries are designed to be neutral and independent, employed for either source language analysis or target language generation and whatever other language may be involved. The bilingual transfer dictionaries, by contrast, are designed for a specific pair in one translation direction.

Typical monolingual entries for the German *Ausgabe* and English *output* are given (somewhat simplified) in (1).

```
(1) (Ausgabe  CAT  (NST)          (output  CAT  (NST)
              ALO  (Ausgabe)               ALO  (output)
              PLC  (WI)                    PLC  (WI)
              TAG  (DP)                    TAG  (DP)
              CL   (P-N S-0)               CL   (P-S S-01)
              GD   (F)                     ON   (VC)
              SX   (N)                     SX   (N)
              TY   (ABS DUR)
```

In both cases, the entries for these 'noun stems' (NST) give a single variant (ALO = 'allomorph'), which must be word initial (WI), specify a data processing (DP) subject field, and state that they are N[eutral] in sex (SX). The other

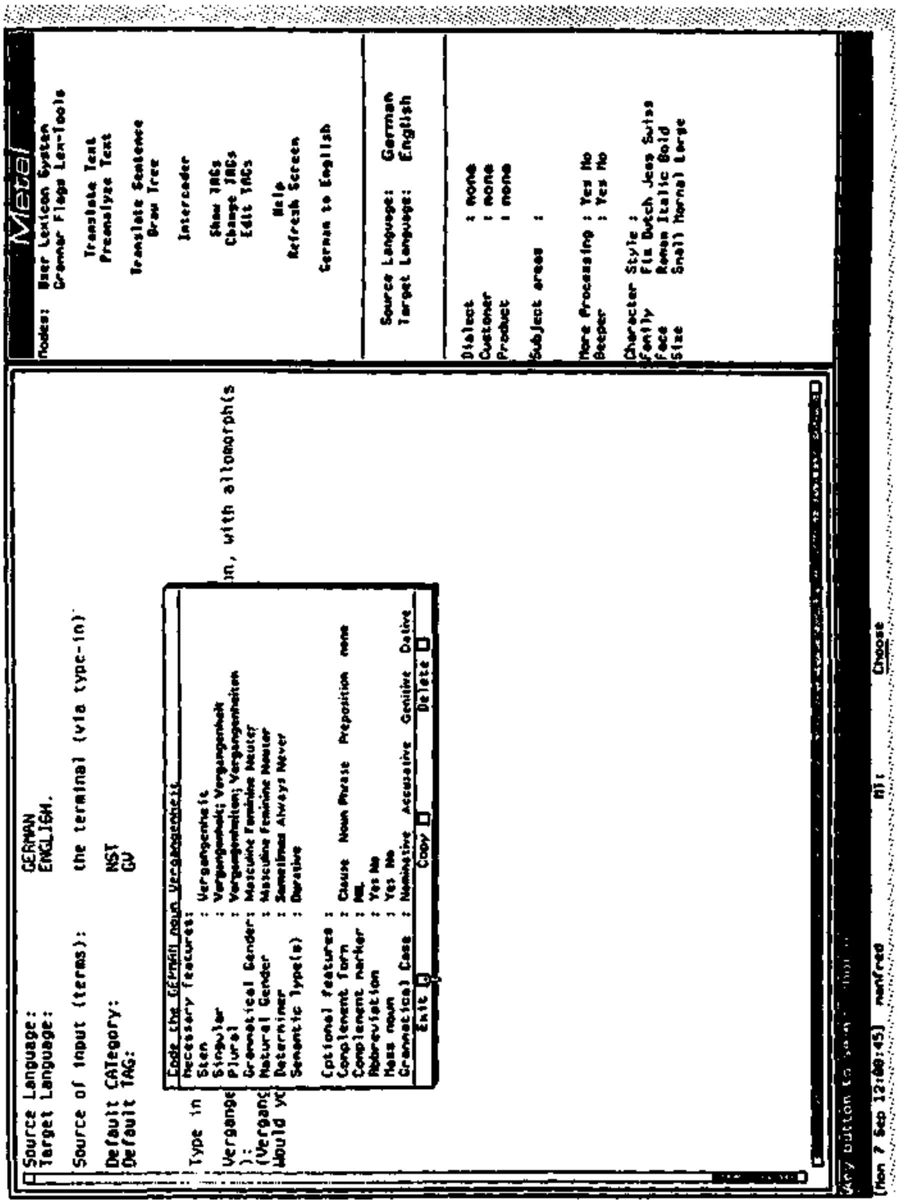


Figure 15.2 Intercoder screen: coding the German noun *Vergangenheit*

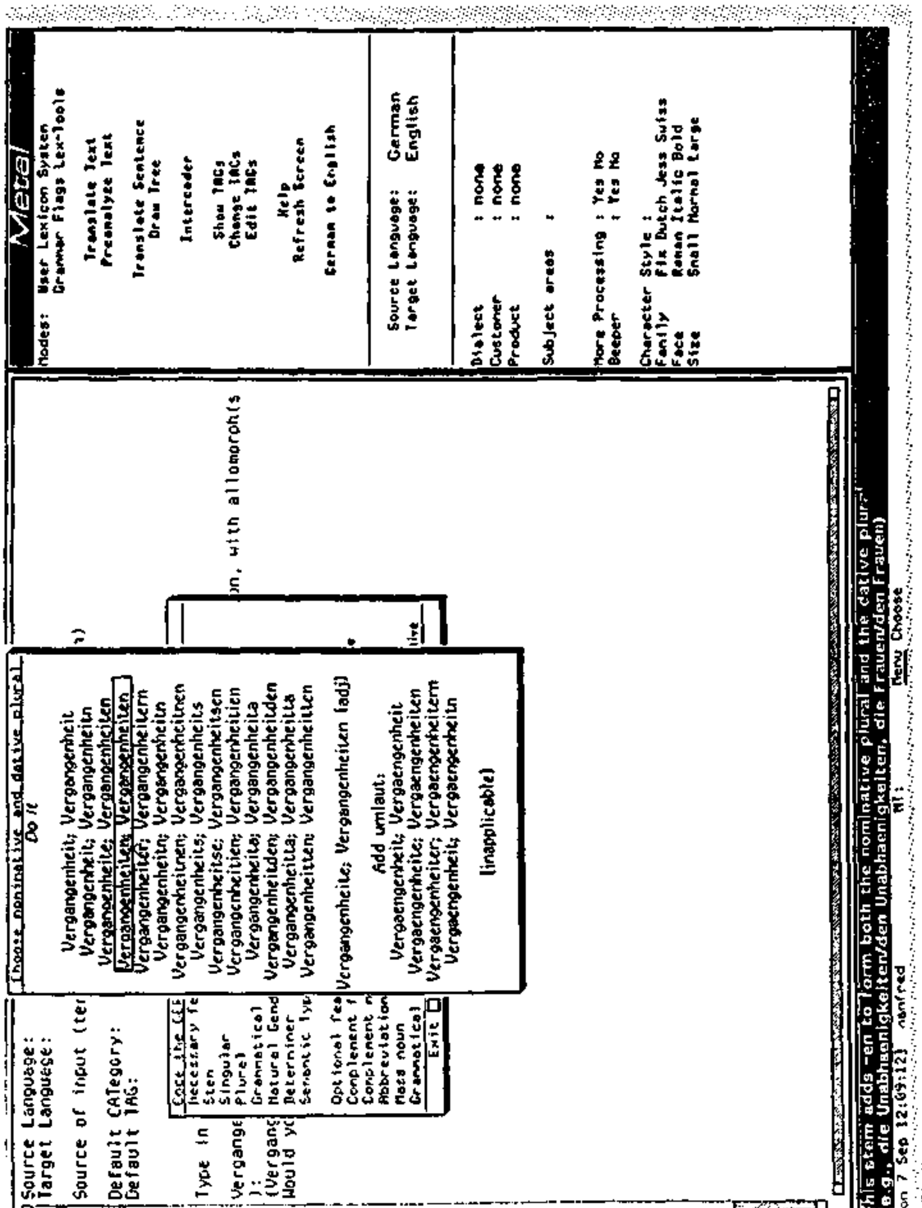


Figure 15.3 Intercoder screen with pop-up window showing possible inflection paradigms

specifications are, of course, language-dependent: the inflectional CL[asses], the grammatical gender (GD) in the German entry and the noun type (ON) in the English entry. The semantic features (TY) attached to the German entry are ABS[tract] and DUR[ative].

The entry in the bilingual dictionary which links the two items is straightforward (2).

```
(2) (Ausgabe (NST DP) 0    output (NST DP) 0)
```

The only condition attached is the restriction to the data processing field. More complex entries are found when there is more than one possible target form, e.g. (3) for the preposition *vor*.

```
(3) (vor (PREP ALL) 30 in front of (PREP ALL) 0
         OPT TY * ABS DUR PNT)
    (vor (PREP ALL) 20 before      (PREP ALL) 0
         GC D
         TY ABS PNT)
    (vor (PREP ALL) 10 ago         (PREP ALL) 0
         GC D
         TY DUR)
    (vor (PREP ALL) 0  in front of (PREP ALL) 0)
```

The English target form is determined by restrictions on semantic features (TY) or grammatical cases (GC) of following German nouns: *in front of* is produced if nouns are not (*) ABS[tract], DUR[ative] or 'punctual' (PNT); *before* if nouns are abstract or punctual and in the D[ative] case; and *ago* if nouns are durative and in the dative, the position of *ago* is later moved after the English noun. If none of these conditions apply, then the default translation is *in front of*. There are no subject field restrictions, and the order in which the options are examined is given by the PRF values (30, 20, 10, 0).

15.3.2 Grammatical rules

The grammars of METAL consist of unordered sets of context free phrase structure rules augmented by tests and conditions and by specifications of the structures to be output. All rules are formulated as Lisp functions. They include rules for inflectional morphology as well as syntactic structures; and they combine operations to be performed during analysis with operations to be performed during transfer.

Just as the lexicographers' task of building up the lexical databases is facilitated by the use of the Intercoder, there is a sophisticated software support to help grammar writers, namely the Metalshop syntax development tool. This enables linguists to see the structures being built at the various stages in the translation process, and to locate quickly the relevant rules which have applied for a given part of the structure, for example by clicking with a mouse on the appropriate node in a tree structure (Figures 15.4 and 15.5). Although this facility is not available (or even particularly useful) to the end user, it certainly supports a more controlled

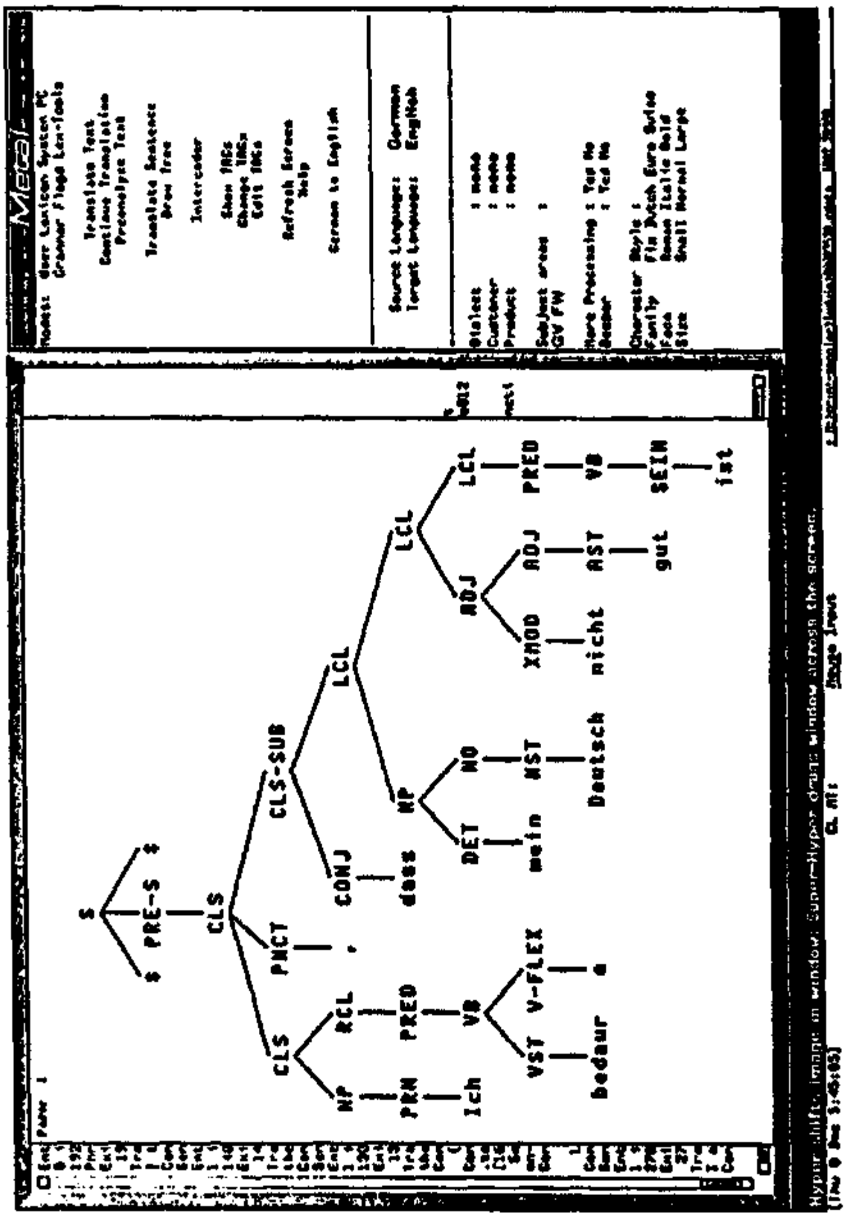


Figure 15.4 Metalshop screen showing result of parse

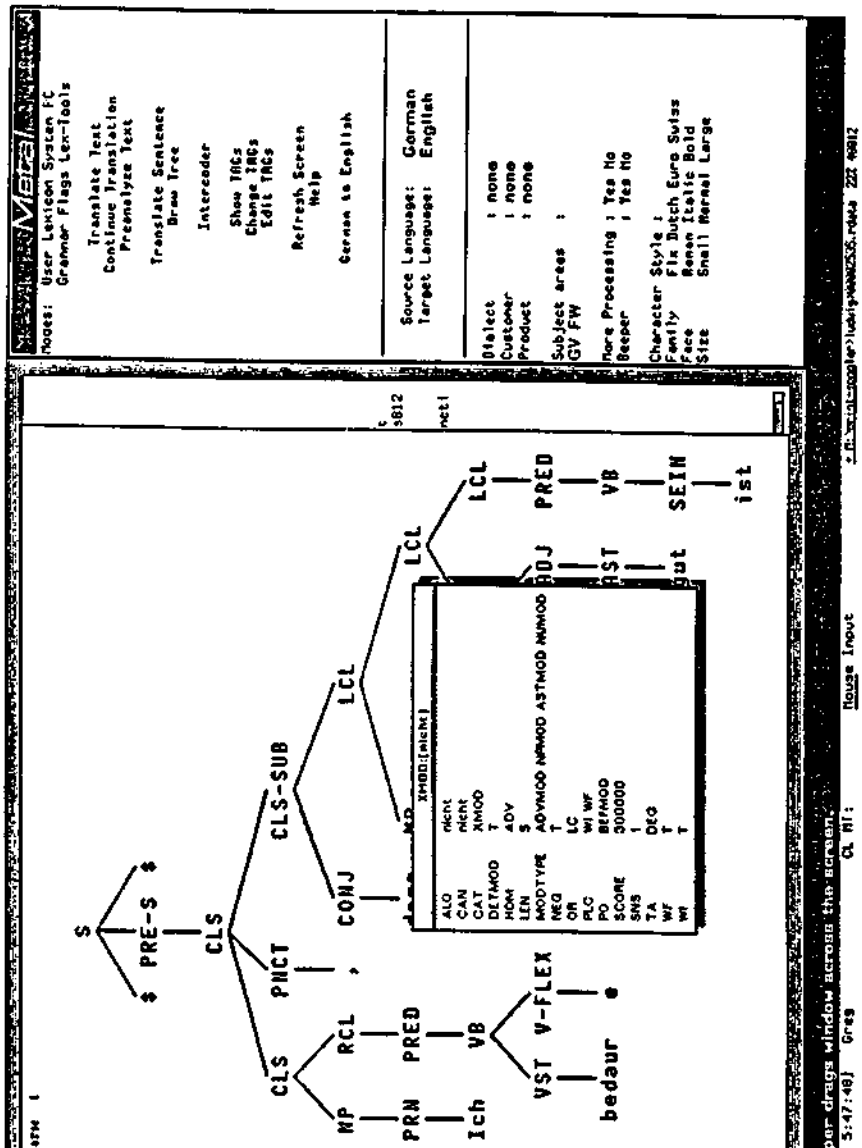


Figure 15.5 Pop-up window showing full specification of XMOD node

grammar development environment, making METAL as a whole less prone to some of the degradation problems encountered in systems like Systran (Chapter 10).

The context free rules are in the familiar rewrite form. There are two types of test. The first specifies morphological and syntactic conditions, e.g. (4).

```
(4)  VB   GE-VB     VST            V-FLEX
     0    1         2              3
          (REQ WI)  (NRQ WI)       (NRQ WI)
          --        (OPT PX NIL)   (REQ PF PAPL)
```

This rule specifies the well-formedness of a German past participle, e.g. *gemacht*. It states that a verb (VB) may consist of a past-participle marker (GE-VB) *ge*, followed by a verb stem (VST) *mach*, followed by a verb ending (V-FLEX) *t*; the GE-VB is REQuired to be word initial (WI), the VST must not be (NRQ) word initial, and nor must the V-FLEX; although the VST element may OPTionally have a 'prefix' feature (PX), it must not be present (NIL) in this construction (i.e. the rule prohibits **geausmacht* from *ausmachen*); and V-FLEX must bear the paradigmatic form (PF) for the past participle (PAPL).

The second test specifies the features required of elements having a particular constituency relation to the item in question; it operates by examining the results of intersections and unions of feature sets (5).

```
(5) ADJ  ADJ  PP
    0    1    2

    TEST (COND ((INT 1 FC PP) (INT 1 MC 2 PR = X1))
               (T NIL))
```

If an ADJ[ective] permits as its complement (FC) a prepositional phrase (PP) then the feature (MC) of the adjective specifying the permissible syntactic functions of complements should be compared ('INT[ersected]') with the values of the PR[eposition] feature on the prepositional phrase which is actually present. The result of this comparison ('intersection') is recorded in a variable (X1) for later use.

Following the tests come the specifications of the structures to be built by rules during analysis. The CONSTR parts of rules specify the features and values to be attached to the governing nodes, and may also modify trees by applying transformational rules. In the familiar fashion, the (non-terminal) category on the left-hand side is the governing node and the categories on the right-hand side are its dependents. The basic processes are those of adding features and values, copying features and values, and assigning preference (PRF) values: for example, a noun phrase comprising an adjective and a noun (6).

```
(6) NO     ADJ                    NO
    0      1                      2
           (REQ PO ATR)           --

    TEST   (INT 1 NU 2 NU = X1)
           (INT 1 CA 2 CA = X2)
           (INT 1 GD 2 GD = X3)
```

```
CONSTR (ADX X1)
       (ADX X2)
       (ADX X3)
       (CPX 2 IN NU CA GD)
       (CPY 1 IN DG WI)
       (ADD PNM)
       (AND (INT 0 CA G) (ADD GEN))
```

The TEST function identifies the values for NU[mber], CA[se] and gender (GD) shared by the adjective and the noun, as variables X1, X2 and X3 respectively. The first three lines of CONSTR add these values (ADX) to the governing node. Then, from the subordinate node (2) it copies all features and values except (CPX) those for IN[flection], number, case and gender. It then copies (CPY) to the governing node the features inflection, degree (DG) and word-initial (WI) from the adjective (node 1), and adds the feature PNM ('prenominal modifier'). Finally if the CA[se] of the governing node is specified as being G[enitive] then the feature GEN[itive] is also added.

For verb structures, the CONSTR part of the rule defines the case-frames which must be satisfied and the construction which should be formed. For example, the structure for intransitive verbs with two arguments (I2AL, e.g. *gehen*) is given as in (7).

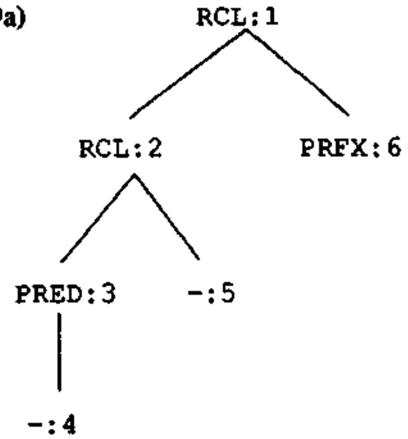
```
(7) (DEXPR I2AL (VC MD)
        (COND ((SYNTAX)
                (COND ((AND (ACTIVE)
                            (NON-COMMAND)
                            (FRAME N NP AGT)
                            (FRAME NIL NIL LOC)
                        T)
                      ((AND (ACTIVE)
                            (COMMAND)
                            (FRAME NIL NIL LOC))
                        T)))
```

The first frame is activated if the verb is active and declarative (NON-COMMAND). It has two arguments, AGT and LOC: the AGT must be a noun phrase (NP) in the N[ominative] case, the form and features of LOC are specific to the particular verb (i.e. NIL indicates that these values have to be derived from the dictionary entries). The second frame applies if the verb is active and imperative (COMMAND); the only argument is LOC, with features again specific to the particular verb.

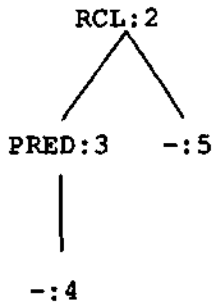
Transformational rules apply generally to the CONSTR parts of rules involving non-terminal categories, for example (8), where RCL indicates 'right-branching clause':

```
(8) XFM (RCL:1 ((RCL:2 ((PRED:3(-:4))-:5)) PRFX:6))
        (RCL:2 ((PRED:3 (-:4) (CPY 6 CAN)
                              (ADD VC A))
                 (-:5) (ADD CLF) (ADD SPX)))
```

(9a)



(9b)



Rule (8) has the effect of transforming (9a) into (9b).

At the same time, the feature CAN[onical form] is copied from PRFX ('prefix') to PRED; the feature VC ('voice') with value A is added to PRED; and the features CLF ('clause final') and SPX ('separable prefix') are added to RCL:2.

The final sections of grammatical rules specify the operations to be performed during transfer. These embrace both lexical transfer and structural transfer. A rule such as (10) applies to verbs.

(10) (TLX (PF NU TN PS) (V-FLEX CL PF NU TN PS))

Rule (10) takes from the V-FLEX node of the source verb the values attached to the features PF ('paradigmatic form'), NU[mber], TN ('tense') and PS ('person') and copies these onto the target verb form. As an example of structural transfer, the entry for intransitive verbs given in (7) continues as in (11).

```
(11) ...
        ((AND (ACTIVE) (NON-COMMAND) (PRED AGT LOC))
              (ROL-ORDER (AGT) (PRED) (LOC))
        ((AND (ACTIVE) (COMMAND) (PRED LOC))
              (ROL-ORDER (PRED) (LOC))))
```

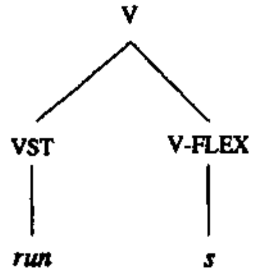
The first line applies if the structure identified during analysis has been active and declarative; the second if it has been active and imperative. In the first case, the target language order of predicates and arguments is specified (by ROL-ORDER) as the sequence AGT PRED LOC; and in the second case as PRED LOC.

15.4 The translation programs

As a transfer-based system, METAL has three basic phases: Analysis, Transfer and Generation. In some accounts a fourth phase of 'Integration' appears between Analysis and Transfer, which has the task of resolving problems of inter- and extra-sentential anaphora. As we shall see, however, the divisions between these components do not conform to the 'pure' form described in earlier chapters. Hence, METAL is usually characterised as a 'modified transfer' system.

The first stage of analysis (invoked by the function PARSE) is lexical and morphological analysis (USER-WORD): the extraction of potential roots (stems) and endings or inflections (affixes) for every word in a sentence, and the production of constructs of stems and affixes, e.g. (12).

(12)



It is followed by two subsidiary functions: USER-ADD is invoked if the item is a number or a literal found on a special list, and USER-ERROR attempts to interpret words not found in dictionaries (e.g. assuming something is a proper noun).

Morphological analysis is followed by the application of a phrase structure grammar to produce alternative parsings ranked according to 'scores' assigned to particular rules or constituents. It operates on the CONSTR parts of grammatical rules (as described above), with constraints on rule application determined by the various tests included in rules. These constraints are defined in the familiar way by syntactic features. There is no use of semantic features at this stage. Syntactic analysis involves the invocation of various procedures specified in grammatical rules, e.g. the identification of case-frames (the FIND-FRAME function), the carrying out of transformations, as illustrated above in (7) and (9a,b). The scores used for choosing between potential interpretations are based on the PRF ('preference') values attached to lexical items and grammatical configurations. Most cases of homograph ambiguity are resolved by lexical preferences. For syntactic analyses the highest scoring tree is selected.

The METAL parser operates in the familiar parallel bottom-up fashion, but producing only some of the potential analyses, i.e. neither all possible interpretations nor only the single first or 'best' analysis. It uses a prioritised chart parser, with unlikely paths eliminated via preferential weightings. Grammar rules are grouped into 'levels'; the parser attempts to apply all rules at a lower level before going to rules at a higher level, and it stops as soon as one or more interpretations are found; if surface structure can be interpreted using lower level rules then the more complex and less likely rules are disregarded, if lower level rules are unsuccessful then progressively higher level rules are attempted. Thus, the system does not produce all possible interpretations. This 'some paths' approach has been extensively tested and investigated: producing faster processing times with no exponential explosions or degradation of quality. It permits also the implementation of 'fail-soft' routines: even if a complete analysis has not been achieved, transfer can still proceed with partial analyses of sentence fragments. The parser's chart is examined for the shortest path, representing the fewest and longest spanning phrases constructed during analysis, and in cases of doubt selecting the highest scoring interpretations. Often the results of such 'phrasal analyses' are in fact grammatically correct. Partial parsing is also applied when the input is itself incomplete, e.g. parenthetical expressions and chapter headings.

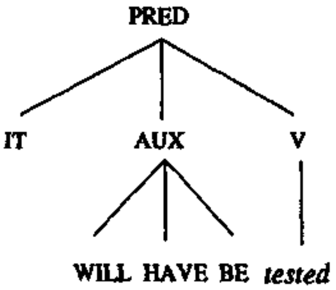
Transfer procedures are invoked by the next function TRANSFER. There is no central control, the process being determined by the TRANSF parts of grammatical rules. During analysis, these have been attached to appropriate nodes of structural representations, but are in effect 'suspended'. During transfer, these parts of grammatical rules are carried out (or 'evaluated' in Lisp terminology). Whereas analysis proceeds in a bottom-up fashion, transfer operates from the head nodes of trees and subtrees, working downwards to all dependent nodes.

Transfer involves complex interactions of lexical transfer rules (from the bilingual dictionary) and structural transfer rules (from the TRANSF parts of grammatical rules). Both types of rules can affect the operation of the other. The inputs to transfer in METAL are relatively shallow phrase structure representations with case role assignments and some semantic features. Outputs are surface representations with full specification of word order (e.g. the rule in (11) for *go*) and morphological constituents. As a consequence, the final generation stage (invoked by the function GENERATE) is concerned solely with producing morphologically correct target language strings.

In later versions of METAL, generation has a greater role. Some of the transformations previously applied during transfer are now invoked by GENERATE. For example, in the English–German system under development, analysis of (13) might be the (somewhat simplified) shallow representation (14).

(13) It will have been tested.

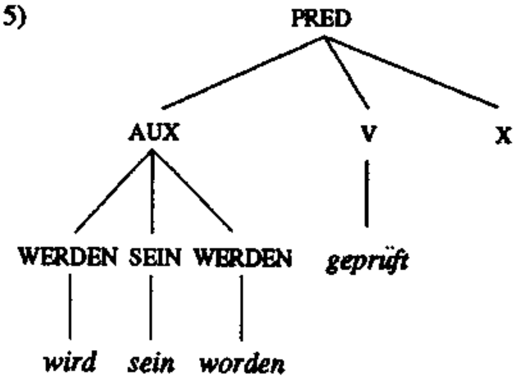
(14)



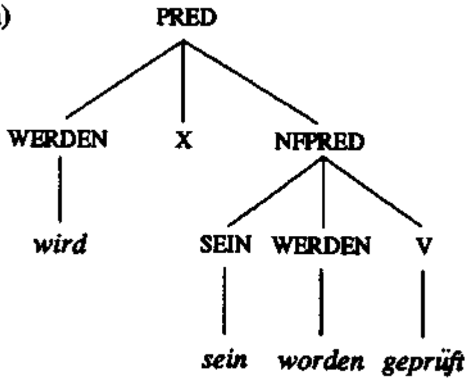
Transfer rules would produce a German structure such as (15).

Lexical transfer rules replace the future auxiliary *WILL* in (14) by the left *WERDEN* in (15), the passive auxiliary *BE* by the right *WERDEN*, and the perfective *HAVE* by the perfective *SEIN*, where selection of the latter is determined by features attached to the second *WERDEN*. In the generation phase this is transformed first into (16a) and then (16b).

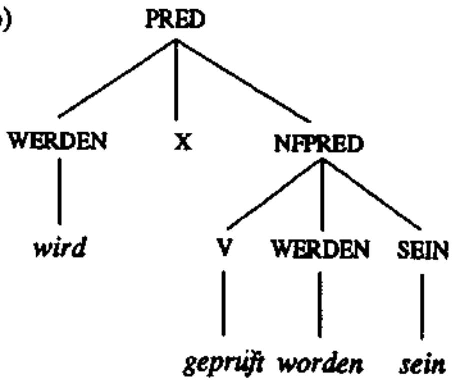
(15)



(16a)



(16b)



From (16b) is produced the correct German sentence (17) corresponding to (13).

(17) *Es wird geprüft worden sein.*

Details of recent developments in the design of transfer and generation components for the new METAL language pairs are not as well documented as the earlier German–English version. It is a system which is still evolving towards a multilingual design (section 15.6).

15.5 Characteristics of the German–English system

The METAL system is generally described as a 'modified transfer' system. Its transfer features are clear in a number of respects: the separation of source, target and transfer dictionaries, the separation of linguistic data (dictionaries and grammar rules) and computational methods (the Lisp evaluations and the chart parser), the completion of all stages of analysis before transfer and generation. Unlike most direct systems (including Systran, Chapter 10), there are full interpreted representations of source language sentences as output from analysis and as input to transfer.

On the other hand, some features of the German–English system, as described above, show divergence from the 'pure' form of transfer-based MT. As already pointed out, there is no clear separation of transfer and generation processes: much of the target language production is contained within the transfer phase, with generation limited to morphology. The most serious objection to the characterisation of the prototype as a true transfer-based system is, however, the mixture and close inter-relationship of analysis and transfer–generation rules in the linguistic database. There is doubt about the independence of the stages of analysis and transfer–generation; there is no doubt that all analytical procedures are completed before transfer begins and that there is an identifiable intermediate representation; what is in doubt is the independence of the rules themselves. To what extent are the analysis rules (the CONSTR parts) autonomous? And to what extent are the forms of transfer rules (the TRANSF parts) determined by source language characteristics?

Transfer operations in the German–English system are not formulated as general procedures which search representations for matching elements and structures. Transfer rules are directly tied to the rules which have been applied in analysis. Instead of one set of general rules transforming, for example, German case-frame patterns into English case-frame patterns, there are specific transformational rules for the case-frames of each German verb type. In other words, the generation of the English structures is not independent of the German-specific classification of case-frame patterns. More generally, the order in which transfer–generation rules are applied is determined by analysis structures and since these structures are relatively shallow (phrase structures with surface case relations) generation itself is determined by source-language-oriented phenomena.

There are computational advantages in tying transfer rules to analysis. There is no need for transfer to incorporate in effect a fresh 'analysis' of the intermediate representations; there do not have to be decision procedures to determine which transfer rules are applicable; there need be no wasteful application of rules which lead to dead ends. Instead, the process is expedited because it is already known which specific transfer rules must be activated. This computational advantage outweighs the relatively low level of duplication in the transfer rules present in the database. The question remains, however, whether these practical benefits may not damage the extendibility of the system.

Despite some reservations, the METAL German–English system represents the most linguistically and computationally advanced MT system available at present on the commercial market. Particularly attractive for potential purchasers is the sophistication of the text acquisition, preparation, reformatting and postediting facilities. To these should be added the excellent interactive 'expert system' for inputting and changing dictionary entries (the Intercoder) and the capacity to produce reasonable translations even when the linguistic data is deficient or absent (the phrasal analysis procedure).

15.6 Recent developments towards a multilingual system

The METAL researchers began work on new language pairs in the mid 1980s, first English–German, then Dutch and French (in both directions), German to French, and English and German to Spanish. With these additions, and various others as future possibilities also envisaged, it became evident that changes had to be made to the basic 'demonstrator' system architecture.

The German–English system, as we have seen, relies to a large extent on similarities of English and German. This is seen in the analysis and generation of case-frames, the treatment of verbal aspect and tense, and the relatively shallow structural representations. The need to develop new modules to generate from analysis representations of German meant that transfer and generation could not continue to be mixed in the same way: distinctive transfer and generation programs had to be developed. A further consequence was that the rule-based design in which transfer–generation rules were closely tied to analysis rules could not continue: there had to be completely independent analysis and generation components.

As a result, METAL can now be characterised as a 'true' transfer-based system: language-specific analysis and generation with bilingual transfer modules. Generation modules for the languages are now quite independent of source language analysis rules. METAL does not use the same grammatical database for analysis and generation, i.e. there is no reversibility of components. It is argued that the coverage of generation grammars can be narrower than those for analysis; whereas the latter must be capable of dealing with all stylistic variants, for generation it is sufficient to default to single acceptable forms.

The need for more explicit intermediate representations in multilingual systems than in bilingual ones has led the METAL researchers to the design of common 'METAL Interface Representation' (MIR) for transfer components. In contrast to representations in the German–English system, structures are simpler and more abstract. Rather than the structural representation of, for example, a complex of finite, modal and auxiliary verbs there is now a single node with features, i.e. aspect, voice, determiners, case particles, adverbs of time, etc. are now represented as feature–values on noun and verb nodes. There is a conscious move towards the kind of abstract interface representations found in the multilingual Eurotra system (Chapter 14). The implication in some accounts is that MIRs could become interlingual representations of structure for the languages in question (but not 'universal' for all languages in any sense); but lexical transfer will remain definitely bilingual and will continue to have effects on structure, i.e. there will still be lexically-conditioned structural transfer.

METAL continues to be somewhat cautious about the use of semantic features. Lexical transfer is generally determined as far as possible by syntactic information (e.g. German *Schuld* is English *guilt* if singular, or *debt(s)* if plural; English *allow* is German *erlauben* if in the active voice, or *dürfen* if in the passive) or by reference to lexical items, e.g. (18).

(18) German *bestehen* English *consist of* (prep-obj = *aus*)
insist on (prep-obj = *auf*)
consist in (prep-obj = *in*)
pass (dobj-lu = *Examen*)
exist (dobj = nil)

There has always been some use of semantic features (TY), as we have seen in (3) above. In recent years there have been more intensive investigations (the 'Semantics Control Project') on the use of features to aid anaphora resolution, to represent verbal aspect and to classify semantic properties of verbs. The position of METAL researchers continues to be somewhat 'minimalist' in this regard by comparison with those other MT projects which aim for full semantic representations. Just as the approach of METAL to grammatical methods is eclectic, the application of semantic features is pragmatic. Nevertheless, it is recognised that multilinguality demands deeper (semantic) treatment than required for the bilingual uni-directional German–English METAL system.

The detailed analysis of technical terminology in terms of semantic features is not in any case considered to be feasible for a practical operational system. The basic motivation for a large MT system such as METAL is the treatment of massive corpora of technical documentation. Semantic analysis on this scale is impractical. More pertinently, most terminology is sufficiently standardised for lexical transfer to be based on easily formulated equivalences and contexts. The large dictionaries of METAL are derived mainly from the relatively simply structured TEAM term bank of Siemens. As we have seen, users are encouraged to develop more specific microglossaries (e.g. reflecting company practice) on similar lines. The accent so far has been on the technological, and one of the basic METAL dictionaries represents common technical vocabulary. With more users wishing to translate administrative and economic documentation, there is now under development a common social vocabulary.

15.7 Evaluations, users and performance

Detailed evaluations of the METAL prototype were undertaken in the mid 1980s, which demonstrated a satisfactory improvement of translation quality, both in terms of the increase in the number of 'correct' sentences (from 45% to 75% over a five-year period) and in increases in the rate of post-editing (from 15–20 pages a day to over 40 pages a day). Experience of METAL in practice is relatively short as yet, but a number of users have indicated reduced costs of up to 40% as a result of faster through-put. Raw METAL output text can be produced at a rate of one word per second, an estimated 200 pages a day. In addition, translation quality is good enough for 20% or more of text to be passed without post-editing.

The main applications of METAL have been in the fields of data processing and telecommunications, not only at Siemens, but also at a Swiss translation company Compulex and at the Philips Kommunikations-Industrie AG, a major supplier for the German PTT. Some twenty organisations have installed METAL. The vendors are commendably cautious about raising expectations. They warn that in the first few months of installation the productivity of a translation service may decrease. Translators need to be trained in new methods and acquire new skills, the dictionaries have to be augmented and revised to suit local circumstances, and administrative practices have to change. After this initial period (which may last up to a year), users of METAL have reported considerable gains in productivity and decreases in turn-around times, by factors of two to three in most cases, and an overall improvement in the quality of their translations through the greater consistency of terminology.

In coming years, practical experience with METAL will permit valid comparisons with systems which have been in longer operational use, particularly Systran (Chapter 10) and Logos. And as a system founded on clearly articulated linguistic principles, METAL will also represent a benchmark against which the performance of newer experimental systems may be assessed.

15.8 Sources and further reading

The basic description of the METAL system is to be found in Bennett and Slocum (1985) and Slocum (1987). Summaries are given in Whitelock and Kilby (1983) and Hutchins (1986), where details of earlier LRC systems are recorded. For recent information see Schneider (1989), Thurmair (1990) and Alonso (1990); and for the views of large users, Compulex and Philips, see Shah (1989) and Little (1990) respectively.

We are grateful to Tom Schneider and Siemens Nixdorf AG for permission to reproduce the Intercoder and Metalshop screens in this chapter.