4
Basic strategies

This chapter is devoted to some fundamental questions on the basic strategies of MT systems. These concern decisions which designers of MT systems have to address before any construction can start, and they involve both design issues and theoretical issues. The first to be considered is whether a system is to be designed as a multilingual system or as a bilingual system. Next are decisions about whether the system is to adopt the direct method, the transfer method or the interlingua method. Then there is the computational environment as a whole, and distinctions between non-intervention (or 'batch') systems and interactive systems. Finally, there is the overall organization of lexical data in different system types.

In the previous two chapters we have already discussed issues which affect the basic design of an MT system. The choice of linguistic approach is crucial, determining both the basic theoretical framework and fundamental aspects of linguistic strategies and representations. Ok the computational side, the last chapter covered other crucial aspects, in particular the features of modularity and decoupling (or separation).

The basic design decisions to be outlined here are useful parameters for comparing and contrasting different systems.

4.1 Multilingual versus bilingual systems

Many decisions affecting MT system design hinge on the fundamental differences between multilingual and bilingual systems. **Bilingual** systems are those which

translate between a single pair of languages; **multilingual** systems are designed to translate among more man two languages.

Bilingual systems may be uni-directional or bi-directional; that is to say, they may be designed to translate from one language to another in one direction only, or they may be capable of translating from both members of a language pair. As a further refinement we may distinguish between **reversible** bilingual systems and non-reversible systems. In a reversible bilingual system the process involved in the analysis of a language may be inverted without change for the generation of output in the same language. Thus, a system for English analysis might mirror directly a system for English generation in, say, an English-French bilingual system. The difficulties, both theoretical and practical, in designing truly reversible bilingual systems are so great that nearly all bilingual systems are in effect two perhaps quite similar uni-directional systems running on the same computer. Methods of analysis and generation for either of the languages are designed independently, without attempting structural reversibility. A bilingual system is therefore, typically, one designed to translate from one language into one other in a single direction.

A system involving more than two languages is a multilingual system. At one extreme a multilingual system might be designed for a large number of languages in every combination, as is the case of the European Commission's Eurotra project (Chapter 14), where the aim is to provide for translation from and to the nine languages of the European Communities in all directions (i.e. 72 language-pairs). A more modest multilingual system might translate from English into three other languages in one direction only (i.e. three language pairs). An intermediate type might cover, say, English, French, German, Spanish and Japanese but not all combinations of pairs and directions, for example Japanese into each of the European languages, all the European languages into Japanese, but would not be able to translate between any of the European languages. A potential user of such a system might be a Japanese company interested only in translation involving Japanese as a source or target language.

A 'truly' multilingual system is one in which analysis and generation components for a particular language remain constant whatever other languages are involved. For example, in a multilingual system involving English, French and German, the process of French analysis would be the same whether the translation were into English or German, and the generation process for German would be the same whether the source language had been English or French, and so forth. There have been, in fact, a number of 'multilingual' systems where analysis and generation components have differed according to the other language of the pair. An example is the early versions of Systran, e.g. the language pairs English-French, English-Italian, English-German. Originally, the English analysis modules were developed separately for each pair. Some resources and techniques were shared or copied, but basically the modules were independent: one could almost consider the modules as English-analysis-for-French-as-target, English-analysis-for-German-as-target, and so on. In this respect, Systran was not a 'true' multilingual system, but rather a collection of uni-directional bilingual systems. In particular, Systran's modules were in no sense reversible: the English-French and the French-English systems differed in almost every feature. It should be said that

in recent years Systran's components and structure have become more uniform and compatible, so that today it is more like a genuine multilingual system (cf. Chapter 10).

In a 'true' multilingual system, uniformity would extend further: not only would analysis and generation components for particular languages remain constant whatever the other languages) involved, but there would be a common linguistic approach applied to all the languages in the system, and there would be a common use of software as well. A good example is the GETA Ariane system (Chapter 13).

It should be now apparent that if it is intended to construct a multilingual system then it should be as truly multilingual as possible in all its aspects; all components which can in principle be shared between different languages should in fact be shared. In a system where, for example, the analysis of English for French as target language differs from toe analysis of English for German as target language, it would be impossible to maintain the 'modularity' of the system (cf. Chapter 3). As we shall see, it would also be difficult to distinguish between those parts of analysis which are neutral with respect to the target language and those parts which are oriented towards the target language (cf. the discussion on transfer, section 4.2 below).

An obvious question at this point is whether a truly multilingual system is in practice — as opposed to theory — preferable to a bilingual system designed for a specific language pair. There are arguments on both sides; two of the most successful MT systems illustrate the pros and cons very well: GETA's multilingual Ariane system and TAUM's English-French Météo system (Chapters 13 and 12).

Among the most significant constraints imposed by the decision to adopt a multilingual approach rather than a bilingual one is that the isolation of analysis from generation means that no advantage can be taken of any accidental similarities between languages. By contrast, in a bilingual system similarities of vocabulary and syntax can be exploited. Take an example of potential lexical ambiguity (see section 6.1): when translating English *wall* into German it must be decided whether an external *Mauer* or an internal *Wand* is the correct translation, but such a decision is unnecessary when translating into French where the single word *mur* covers both senses. Likewise translation into French does not require distinguishing between two types of *corner* as does translation into Spanish: *esquina* ('outside corner') and *rincón* ('inside corner'). English analysis in a truly multilingual system would have to make these distinctions at some stage; but an English-French bilingual system would not — in fact it would be perverse not to take advantage of direct lexical correspondences. The Météo system is an example of a bilingual system which, although separating analysis and generation, exploits similarities and regular equivalences of English and French lexicon and syntax at every stage of the translation process.

4.2 Direct systems, transfer systems and interlinguas

There are broadly three basic MT strategies. The earliest historically is the 'direct approach', adopted by most MT systems of what has come to be known as the **first generation** of MT systems. In response to the apparent failure of this strategy,

two types of 'indirect approach' were developed: the 'transfer method', and the use of an 'interlingua'. Systems of this nature are sometimes referred to as second generation systems.

The **direct approach** is an MT strategy which lacks any kinds of **intermediate** stages in translation processes: the processing of the source language input text leads 'directly' to the desired target language output text. In certain circumstances the approach is still valid today — traces of the direct approach are found even in indirect systems such as Météo — but the archetypal direct MT system has a more primitive software design.

In considering the operation of first generation MT systems, it should be borne in mind that computers available in the late 1950s and early 1960s were very primitive even in comparison with the humblest electronic calculators of today. There were no high-level programming languages, most programming was done in assembly code. In broad outline, first generation direct MT systems began with what we might call a morphological analysis phase, where there would be some identification of word endings and reduction of inflected forms to their uninflected basic forms, and the results would be input into a large bilingual dictionary look-up program. There would be no analysis of syntactic structure or of semantic relationships. In other words, lexical identification would depend on morphological analysis and would lead directly to bilingual dictionary look-up providing target language word equivalences. There would follow some local reordering rules to give more acceptable target language output, perhaps moving some adjectives or verb particles, and then the target language text would be produced.
The direct approach is summarized in Figure 4.1.

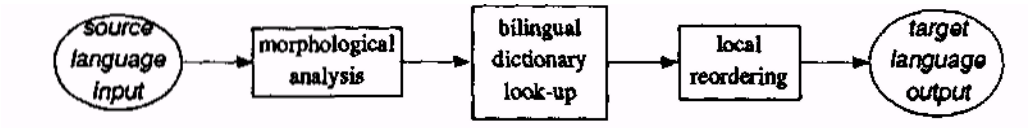

Figure 4.1 Direct MT system

The severe limitations of this approach should be obvious. It can be characterized as 'word-for-word' translation with some local word-order adjustment. It gave the kind of translation quality that might be expected from someone with a very cheap bilingual dictionary and only the most rudimentary knowledge of the grammar of the target language: frequent mistranslations at the lexical level and largely inappropriate syntax structures which mirrored too closely those of the source language. Here are some examples of the output of a Russian-English system of this kind (the correct translation is given second).

(1) *My trebuem mira.*
We require world
'We want peace.'

(2) *Nam nužno mnogo uglja, železa, elektroenergii.*
To us much coal is necessary, gland, electric power.
'We need a lot of coal, iron and electricity.'

(3) *On dopisal stranitsu i otložil ručku v storonu.*
It wrote a page and put off a knob to the side.
'He finished writing the page and laid his pen aside.'

(4) *Včera my tselyi čas katalis' na lodke,*
Yesterday we the entire hour rolled themselves on a boat.
'Yesterday we went out boating for a whole hour.'

(5) *Ona navarila ščei na neskol'ko dnei.*
It welded on cabbage soups on several days.
'She cooked enough cabbage soup for several days.'

The linguistic and computational naivety of this approach was quickly recognized. From a linguistic point of view what is missing is any analysis of the internal structure of the source text, particularly the grammatical relationships between the principal parts of the sentences. The lack of computational sophistication was largely a reflection of the primitive state of computer science at the time, but it was also determined by the unsophisticated approach to linguistics in MT projects of the late 1950s.

Before leaving the direct translation method, it should be noted that it continues to some extent in many uni-directional bilingual systems {cf. previous section): there may be less linguistic naivety than in the past, but the general idea of 'moulding' (restructuring) the source text into a target language output is retained: such systems take advantage of similarities of structure and vocabulary between source and target languages in order to translate as much as possible according to the 'direct' approach; the designers are then able to concentrate most effort on areas of grammar and syntax where the languages differ greatest.

The failure of the first generation systems (cf. section 1.3) led to the development of more sophisticated linguistic models for translation. In particular, there was increasing support for the analysis of source language texts into some kind of intermediate representation — a representation of its 'meaning' in some respect — which could form the basis of generation of the target text. This is in essence the **indirect** method, which has two principal variants.

The first is the **interlingua** method — also the first historically (cf. Chapter 1) — where the source text is analysed in a representation from which the target text is directly generated. The intermediate representation includes all information necessary for the generation of the target text without 'looking back' to the original text. The representation is thus a projection from the source text and at the same time acts as the basis for the generation of the target text; it is an abstract representation of the target text as well as a representation of the source text. The method is interlingual in the sense that the representation is neutral between two or more languages. In the past, the intention or hope was to develop an interlingual representation which was truly 'universal' and could thus be intermediary between any natural languages. At present, interlingual systems are less ambitious.

The interlingua approach is clearly most attractive for multilingual systems. Each analysis module can be independent, both of all other analysis modules and of all generation modules (Figure 4.2). Target languages have no effect on any processes of analysis; the aim of analysis is the derivation of an 'interlingual' representation.

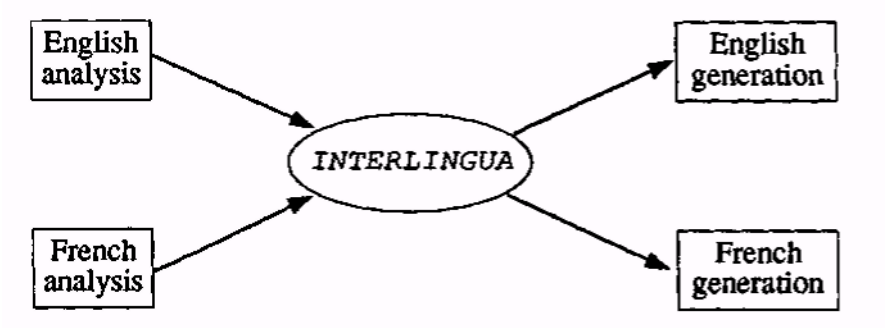

Figure 4.2 Interlingua model with two language pairs

The advantage is that the addition of a new language to the system entails the creation of just two new modules: an analysis grammar and a generation grammar. By adding one analysts module in Figure 4.2, e,g. a German analysis grammar, the number of translation directions is increased from two (English to French, and French to English) to four (by the addition of German to French and German to English). The inclusion of another generation module, a German generation grammar, brings a further two pairs (English to German and French to German). Compare Figure 4.3 with Figure 4.2 above.

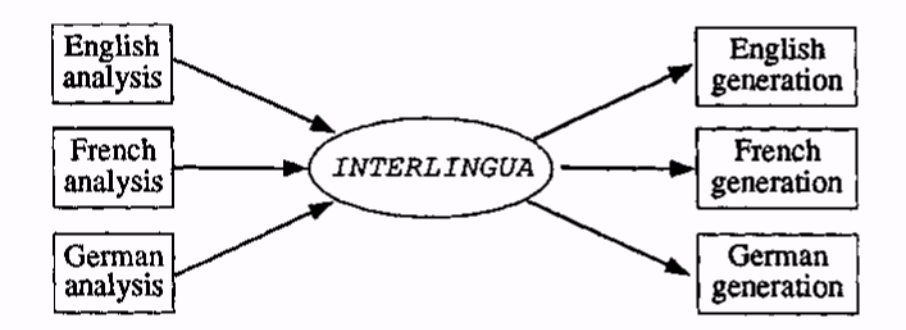

Figure 4.3 Interlingua model with six language pairs

The addition of two further modules for, say, Spanish, would increase the number of language pairs by another six (English, French and German into Spanish, and Spanish into English, French and German), and so on exponentially.

It may also be noted that such a configuration permits 'translation' from and into the same language, for example, the conversion of an English source text into the interlingual representation and then back into an English target text. This seemingly unnecessary 'back-translation' capability could in fact be extremely valuable during system development in order to test analysis and generation modules. Note that you might not necessarily expect the regenerated target text to be identical to the original source text, though you would expect them to be pragmatically equivalent (see Chapter 7 concerning choices in generation).

While the addition of new languages may appear easy in an interlingual system, there are major disadvantages: the difficulties of defining an interlingua, even for closely related languages (e.g. the Romance languages: French. Italian, Spanish, Portuguese). A truly 'universal' and language-independent interlingua has defied the best efforts of linguists and philosophers from the seventeenth century onwards (cf. Chapter 1). The particular problems involved will be considered in more detail in section 6.7.

The second variant of the indirect approach is called the **transfer** method. Strictly speaking all translation systems involve 'transfer' of some kind, the conversion of a source text or representation into a target text or representation. The term 'transfer method' has been applied to systems which interpose bilingual modules between intermediate representations. Unlike those in interlingual systems these representations are **language-dependent**: the result of analysis is an abstract representation of the source text, the input to generation is an abstract representation of the target text. The function of the bilingual transfer modules is to convert source language (intermediate) representations into target language (intermediate) representations, as show in Figure 4.4. Since these representations link separate modules (analysts, transfer, generation), they are also frequently referred to as **interface representations**.

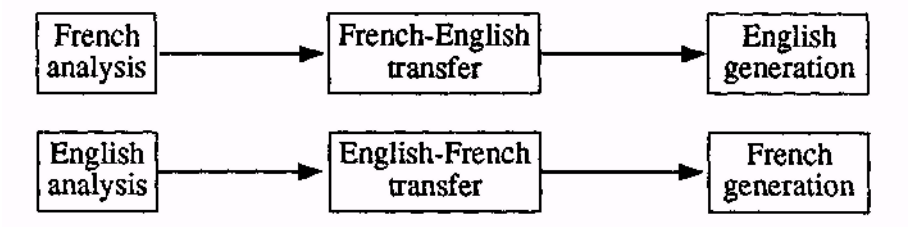

Figure 4.4 Transfer model with two language pairs

In the transfer approach there are therefore no language-independent representations: the source language intermediate representation is specific to a particular language, as is the target language intermediate representation. Indeed there is no necessary equivalence between the source and target intermediate (interface) representations for the same language. This distinction between language independence and language dependence will be examined in more detail in Chapter 6.

In comparison with the interlingua type of multilingual system there are clear disadvantages in the transfer approach. The addition of a new language involves not only the two modules for analysis and generation, but also the addition of new transfer modules, the number of which may vary according to the number of languages in the existing system: in the case of a two-language system, a third language would require four new transfer modules. Compare Figure 4.5 with Figure 4.4., and with the corresponding diagram for the interlingua system (Figure 4.3).

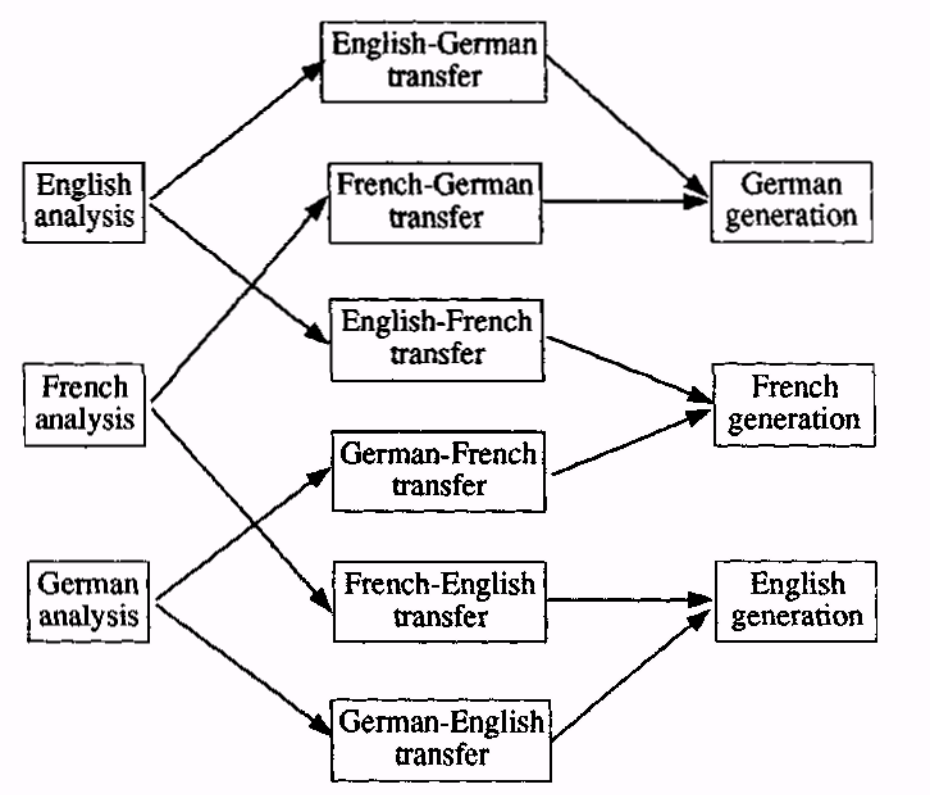

Figure 4.5 Transfer model with six language pairs

The addition of a fourth language would entail the development of six new transfer modules, and so on as illustrated in Table 4.1.

The number of transfer modules in a multilingual transfer system, for all combinations of n languages, is n x (n – 1), i.e. not much less than n^2. Also needed are n analysis and n generation modules, which however would also be needed for an interlingua system.

Why then is the transfer approach so often preferred to the interlingua method? The first reason has already been mentioned: the difficulty of devising language-independent representations. The second is the complexity of analysis and generation grammars when the representations are inevitably far removed from the characteristic features of the source and target texts. By comparison, the relative complexity of the analysis and generation modules in a transfer system is much reduced, because the intermediate representations involved are still language-dependent abstractions (for more on this, see Chapter 6). At the same time, if the design is optimal, the work of transfer modules can be greatly simplified and the creation of new ones can be less onerous than might be imagined.

Number of languages	*Analysis modules*	*Generation modules*	*Transfer modules*	*Total modules*
2	2	2	2	6
3	3	3	6	12
4	4	4	12	20
5	5	5	20	30
...				
9	9	9	72	90
n	*n*	*n*	$n(n-1)$	$n(n+1)$

Table 4.1 Modules required in an all-pairs multilingual transfer system

4.3 Non-intervention vs. on-line interactive systems

The third major design feature to be introduced concerns the users of systems during the translation process, whether they play an active interventionist role, or whether they have no interaction with the system while it is processing the text.

The **non-interventionist** mode of operation has its roots in early computing practice where it was normal for computations to be performed in 'batch' mode: a 'job' was prepared, submitted to the computer and some time later the results would be printed out. The operators and programmers could not influence the operation of the computer once it had started a job. With later computational developments, particularly the advent of mini- and micro-computers, the 'conversational' style of running programs became more widespread, where users are able to intervene and interact directly with the computer. Many of the present-day MT systems were developed at a time when sophisticated human-machine interaction was not possible and only vaguely envisaged for the future. They were designed for 'batch' translation: a source text was input, the MT program was run and a target version output. Any revision could be undertaken only after the whole target text had been produced. More recently, many such MT systems have incorporated text-editing facilities enabling revision at terminals or on micro-computers. There are often also more sophisticated input facilities. However, the translation system itself is still 'non-interventionist': translation jobs are completed in 'batches' and there is no interaction during the translation processes with any human operators, users, translators or revisers.

By contrast the **interactive** mode of translation enables direct involvement on-line during the course of translation. Various types of interaction are possible: the computer may ask the user to supplement its linguistic information, requesting confirmation of its decisions, or selection from among alternatives. Most typical is the case where the source language sentence is ambiguous out of context, i.e. when translated as an independent sentence, which is the normal mode for MT

systems (see Chapter 5 for problems involving lack of contextual information). In other instances, assistance may be sought with the selection of target language expressions when there are distinctions in the target language not made in the source. For example, the English verb *wear* has a number of possible Japanese translations depending on the object concerned (see section 6.1). This interaction could take place in two different ways as illustrated below: (a) assuming that the user knows Japanese, and (b) assuming that the user knows only the source language, English.

(a) To translate wear, please choose one of *haoru*, *haku, kaburu, hameru, shimeru, tsukeru, kakeru, hayasu, kiru*

(b) To translate *wear,* please indicate the type of the (understood) object from the following: coat or jacket, shoes or trousers, hat, ring or gloves, belt or tie or scarf, brooch or clip, glasses or necklace, moustache, general or unspecific

Other types of interaction will be described in more detail in section 8.3.3, where the distinctions made here are shown to be not quite as clear-cut as they seem initially.

4.4 Lexical data

The linguistic data required in MT systems can be broadly divided into lexical data and grammatical data. By 'grammatical data' is understood the information which is embodied in grammars used by analysis and generation routines. This information is stated in terms of acceptable combinations of categories and features (as described in section 3.8 above). By 'lexical data' is meant the specific information about each individual lexical item (word or phrase) in the vocabulary of the languages concerned. In nearly all systems, grammars are kept separate from lexical information, though clearly the one depends on the other.

The lexical information required for MT differs in many respects from that found in conventional dictionaries. As already mentioned (sections 2.4 and 3.6) there is no need in MT for information about pronunciation or etymology, for lists of synonyms, for definitions, or for examples of usage. On the other hand, the information needed for syntactic and semantic processing has to be far more explicit than that found in dictionaries for human consultation: grammatical category, morphological type, subcategorization features, valency information, case frames, semantic features and selection restrictions. Because of these basic differences, the term lexicon is preferred to 'dictionary'.

Each of the design decisions mentioned above has implications for the organization of lexical data. In MT systems of the direct translation design there is basically one bilingual lexicon containing data about lexical items of the source language and their equivalents in the target language. It might typically be a list of source language items (either in full or root forms, cf. sections 2.4 and 5.1), where each entry combines grammatical data (sometimes including semantic features) for the source item, its target language equivalents), relevant grammatical data about each of the target items, and the information considered necessary to select between target language alternatives and in order to change syntactic structures

into those appropriate for the target language (see particularly sections 6.4 and 7.1 below). The result can be a lexicon of considerable complexity.

In indirect systems, by contrast, analysis and generation modules are independent, and such systems have separate monolingual lexicons for source and target languages, and bilingual 'transfer' lexicons. The monolingual lexicon for the source language contains the information necessary for structural analysis and disambiguation (Chapter 5), i.e. morphological inflections, grammatical categories, semantic features, selection restrictions, typically, for homographs and polysemes, each form has its own entry: for example, there will be two entries for *feed,* as a noun and a verb, and the noun *bank* will have two distinct entries for the two readings 'financial institution' and 'side of a river'. The bilingual lexicon for converting lexical items from source to target or to and from interlingual representations may be simpler, being restricted to lexical correspondences (e.g. that in French the first sense of *bank* is *banque* while the second is *rive*), and containing only the minimum of grammatical information for the two languages. The lexicon for generation is also often (though not always) less detailed than the one needed for analysis, since the information required for disambiguation is not necessary; however, as we shall see in Chapter 7, this view of the relative simplicity of generation may be a general weakness in MT system design. In addition, it should be noted that some indirect MT systems incorporate most of the information for the selection of target language forms in their bilingual transfer lexicons rather than in the monolingual generation lexicon (even if the choice of target item is properly a question of target language style or grammar), so that the generation lexicon is typically limited to morphological data.

In practice, the situation is sometimes further complicated in many systems by the division of lexicons into a number of special dictionaries, e.g. for 'high frequency' vocabulary, idiomatic expressions, irregular forms, etc., which are separated from the main or 'core' lexicons. This is because systems often have special routines devoted to handling less 'regular' items.

It is even more common to find separate lexicons for specific subject domains (e.g. biochemistry, metallurgy, economics), with the aim of reducing problems of homography (cf. section 5.2.2 below). For example, instead of a single lexicon containing a number of entries for the word *field,* each specifying different sets of semantic features and co-occurrence constraints, etc, and each relating perhaps to different target language equivalents, there might be a lexicon for physics texts with one entry and one target equivalent, another for agriculture texts, and so on. Such specialised subject lexicons are sometimes called microglossaries. We will return to the question of MT systems for specific subject fields or 'sublanguages' in Chapter 8.

The choice of an 'interactive' mode of operation can permit considerable simplification of lexical databases. If the system itself does not have to make decisions about the choice of target equivalents (whether *wall* is *Wand* or *Mauer),* or if it does not have to resolve ambiguities of grammatical category (e.g. whether *light* is a noun or adjective) or ambiguities of homonymy (e.g. whether *board* is a flat surface or a group of people), then the lexicon does not need to include

complex grammatical, semantic and translational information, but may simply list alternatives for presentation to the human operator for selection.

From this brief outline it should be evident that the organization and content of MT lexicons are highly dependent on the methods used for analysis, transfer and generation and on the degree of human interaction envisaged in practical operation. These differences will emerge in the following chapters and are illustrated particularly in the descriptions of specific systems in Chapters 10 to 17.

4.5 Further reading

Most of the points covered in this chapter are covered in general articles about MT.

The contrast between bilingual and multilingual system design is rather briefly discussed in King (1982:140), and in Bennett *et al.* (1986:78f).

For a discussion of the differences between direct and indirect methods, see Johnson (1984), Tucker (1987), and Lehrberger and Bourbeau (1988:8-38). The Russian-English examples are taken from Knowles (1979), Descriptions of many direct systems are found in Hutchins (1986).

The choice of the interlingua and transfer methodologies is discussed in further detail in Chapter 6 of this book: see there for further references. Likewise, interactive and non-intervention systems are covered in Chapter 8, where further references are also given.