[*Terminologie et Traduction*, no.1, 1986]

ENSURING COMPATIBILITY IN THE SYSTRAN ENVIRONMENT

Thomas Pahl,
codework Software GmbH, Hennef, FRG

Abstract

The compatibility issue is related to the growing development of Systran. To some extent, lack of compatibility has been triggered by the desire of some users to go it alone without seeking technical cooperation and exchange. However, the widening distribution of Systran, for example in the European Community countries, requires steps towards standardization, modernization and reunification. The paper will discuss the current situation, show what has been undertaken so far in the European systems, and try to present technical and organizational solutions aiming at an evolutionary approach to systems unification for the benefit of all users.

****

1. What could compatibility mean with respect to Systran?

The compatibility question only arises in a dynamic environment where there is change, exchange, movement. Whereas Systran and its use have developed rather calmly for a long period, the system has now reached a state of rapid growth both with respect to language pairs and in the type and number of applications. An enormous development effort is being spent and will continue to be spent on Systran. For this to be of maximum efficiency and to materialize in wide-spread user benefits, the time would appear to be right for consideration of what actually exists, what could or rather should be done, and how the different existing development forces could be united.

Besides the linguistic developments, two major technical tasks have to be solved for the further growth - and, in my opinion, for the very survival of Systran as the leading MT system:

- unification of the existing system versions;
- rationalization and modernization of the system, both in internal architecture and technology and in external behaviour.

Compatibility in this context would mean - very generally speaking - the availability of most existing and new developments from the different locations accompanied by protection of the existing investment in dictionaries, linguistic programming, text interfacing and other aspects.

Speaking as a systems engineer, it is beyond my scope to talk about the political or organizational environment necessary for the successful execution of these tasks, but this conference has already showed that there is a lot of positive movement in this area.

Thus I will concentrate on the technical aspects, try to give an overview of the existing technical situation, sketch possible ways towards technical fulfillment of the above tasks and discuss some of the impacts on existing translation production and ongoing development (which is another example of the compatibility question).

For today's audience, I will restrict myself to rather general technical statements, keeping the deeper technical details for the workshops on Friday and for the Technical Coordination Group that will hopefully be founded or rather re-founded by end of this week.

2. Systran system structure

In order to clarify which of the many aspects of Systran we are talking about and in order to relate these to the other papers relating to compatibility, let us briefly recall some global system structure information.

The basic functional model of Systran as an MT system is strikingly simple: a Systran production system (i.e. an operational Systran package of programs and data ready to translate in one or more language pairs) can be seen as a functional unit bound into a document production line:

There are many ways in which to provide the hungry Systran machine with input data and to further process the output of the translation mill. We have heard of some very impressive examples in the course of this conference.

Such integration of Systran, and the ability of Systran to be integrated, are a substantial part of the usefulness of Systran for the effective users. These facilities must not only be maintained but should be collected, combined and expanded.

Internally, the Systran box splits into four major components areas:

- the basic operational software;
- dictionaries;
- linguistic programs;
- text interfaces.

And each area has its compatibility problems.

3. Systran Incompatibilities

Systran has a rather peculiar development story, different from other MT systems and untypical for most large software systems:

- a very long development history, implying an aging basic technical design (however, still proving powerful in its concepts);
- for a long time in the past and still to a large extent development was and is triggered by few major users;
- not a single powerful supplier, but various development centres (which have come and gone), with insufficient cooperation and coordination;
- local developments often for short-term user satisfaction (contract fulfillment) with less regard to long-term concepts.

Thus we now face a variety of Systran versions around the world, with the effect that an exchange of data, programs and dictionaries is not easily possible.

Trying to draw a genealogical tree of the Systran versions would take more time than is available - it would also annoy any gardener as a real tree would probably not have survived if it were grown that under such strange circumstances. Systran has survived up to now with two main stems and a little stick:

- The "European" systems, represented by the CEC's system as the master. This line goes back to the so-called "Systran II" of the former WTCC, Ottawa, which itself stems from the non-Russian La Jolla systems.
  Systran Institut uses a slightly older version of the CEC's basic system for its language pairs, especially the German systems.
- The "Universal" systems.
  Primary representatives of this line are the systems of the USAF/FTD at Dayton.
  Other systems are at XEROX (with its special MCE approach) and at FESTO (which, to my knowledge, uses a version this is only half-way converted to the "Universal" structures).
  Finally, the Japanese systems belong to this group.
- the solitary stick to me seems to be the system of EDS Canada (GM Canada) derived from an early WTCC (pre-European) version. It is not considered further in this paper.

The differences between these systems exist in all four main areas of the Systran architecture as mentioned before. One cannot say that one of these systems is generally better than the other, each one has its extra good facilities which would be desirable in a unified system. However, these systems have diverted so much that it is simply not possible to take a good piece from one system and plug it into the other - it would no longer fit.

Some concrete examples:

- Exchange of dictionaries:

  The exchange of dictionaries, especially subject-field oriented subsets, could play an important role in the rapid development of dictionaries covering a wide area of terminology. Whereas questions of style and meaning selection will always require personal judgment, at least the purely technical exchange of dictionary entries should present no problem - it does now due to slight variations in dictionary record formats, different expression coding facilities, and - to give a more sophisticated example - due to different semantic taxonomies.

- Exchange of linguistic programs:

  As I understand, there are at least three variations of English analysis. It may well be that one of these, let us assume the CEC's, is the best, and would solve many of the linguistic problems, encountered, for example, in the English-Japanese system. However, since the linguistic programming environment, as provided by the basic software and the linguistic macro instructions, differs between the "European" and the "Universal" system, it cannot be done simply by exchanging the modules.

- As we have heard in the user presentations, intelligent text pre- and postprocessing programs contribute to a great extent to the acceptance and economy of Systran. Whereas some of these are very specific, many others would be candidates for exchange, provided they can rely on a standardized Systran text interface format.

At a first glance, these discrepancies between the systems look discouraging. One could say, "Well, we got along somehow with these different systems until now, why not try to go on, more or less separately, for another five years or so." There are two technical arguments against this fatalistic approach:

- When we look deeper into the technical details of the system differences, we find that a great number of these are not half as serious, technically, as their effects are on the user surface. Finally, all these systems have a common ancestor.

- On the other hand, the growing use of Systran and its expansion into more language pairs will call for larger development activities not only with respect to dictionaries but also with respect to the basic functional facilities, the linguistic machine of Systran. Without consideration of compatibility, this would lead to a faster diversion of the systems with an exponential growth in the complexity and cost of a later attempt at reunification.

So now is the time to unify the Systran versions into one single group of systems with the following attributes:

- combination of all good features from both the "European" and "Universal" systems;

- a single common basic operational software;
- no unnecessary duplication in languages, neither in linguistic programs nor in dictionaries (some special alternatives might continue to exist, e.g. MCE).

4. Compatibility-oriented aspects of the European systems

The CEC was faced with the compatibility problems quite early on. On the one hand, the Commission's system has been continuously developed to fulfill growing needs, especially in linguistic programming, dictionary coding support, and text interfacing. On the other hand, new language pairs were acquired from La Jolla which were originally developed in the more and more diverging "Universal" system. Integration times for new language pairs are increasing.

Over the last six years, the "European" system has undergone initial modernization steps. Some technical highlights which influence compatibility are:

- Increased operating system independence:
  Operating system references, have been removed from the Systran programs proper and are centralized in a system environment module. Operational problems like blocksizes are handled centrally and uniformly. Systran can be run under IBM/MVS and Siemens BS2000 without any change in either the basic system or the linguistic programs (a VM/CMS version would be no problem).
- Initial orthogonalization and identification of internal interfaces:
  Centralized access functions are replacing individual hand-coded access to tables and files like the loaded dictionaries, text files, semantic code tables. These centralized functions can also fulfill adaptor functions in case of format changes (evolutionary approach maintaining compatibility).
- Increased symbolic programming (instead of bits and bytes) in linguistic programs, supported by more intelligent and more problem-oriented macro instructions.
- Simplified dictionary coding, especially on the target side, where a lot of morphological and other detail codes are generated automatically.
- Exclusive and radical use of the standardized TDCS intermediate text format (true upper lower case, single-byte codes for accented characters, protected user format codes) as the only exchange between the Systran translation subsystem and the text interfaces.

Resulting from the previously mentioned need for integration of new linguistic programs coming from La Jolla, the "European" system already contains adaptions or alternatively conversion aids for some of the features of the "Universal" system. The "European" system still fully supports all pre-"Universal" linguistic software.

Among the features of the "Universal" system which have no counterpart yet in the "European" systems are:

- multi-target dictionaries;
- extended expression coding facilities (this is an area where the "Universal" system provides more symbolic coding than the "European");
- execution of lexical routines at various stages of analysis.

5. How to achieve unification and compatibility

All technical strategy has to be based on the fact that the existing translation production and ongoing linguistic development must not seriously be disturbed by unification activities. A delay in linguistic improvements for a language pair may be tolerable for a month: it certainly is not tolerable for a year.

Furthermore, individual improvements should become beneficial as soon as possible - and to all interested users.

This calls for an evolutionary approach, which starts with one of the existing systems, gracefully modifying it in the direction of unification while keeping it fully operational at all times.

The technical viability of such an approach has already been shown by the above mentioned first steps towards modernization and unification undertaken in the "European" systems. The matter is somewhat facilitated by the fact - especially in the basic software - that many programs are rather loosely coupled (e.g. via intermediate files), which often will allow their renovation without affecting other programs. In other cases, a temporary adapter program could help to massage parameters, etc.

A technical action plan would look roughly as follows:

a. Survey of existing versions.

b. Definition of the features of the unified system (including possible by-products with respect to modernization).

c. Definition/documentation of the official internal interfaces between the major components and data structures, e.g.

    linguistic programs - basic system;
    dictionary coding;
    common function library.

d. Components-wise and, probably, language-wise modification.

It is not necessary, however, to do the complete survey first, then the complete definition, and so on, approving everything at a committee meeting twice a year. The technical activities can be executed virtually in parallel on most of the Systran software components and could start as soon as the general decision is made.