Published in the Proceedings of the Workshop on Multilinguality in the Software Industry: the AI Contribution (MULSAIC'96)
organised in conjunction with the European Conference on Artificial Intelligence (ECAI'96), ECCAI, Budapest, 1996, pp. 21-26.

Multilinguality in Electronic Commerce - Research Issues

Aarno Lehtola[1] and **Timo Honkela**[2]

Abstract. We outline how language technology could be used to support multilinguality in electronic commerce. The growth in international trade and the increasing use of EDI in trade transactions brings new technical challenges also to language technology. Nowadays, trade transactions frequently cross language borders. While ever larger number of users get connected to electronic trading processes, it is no longer possible to presume that everyone knows e.g. English language. The need for native language support is quickly growing in electronic commerce.

In this paper we first present a model of electronic bargaining. After that we consider what multilinguality support means to software construction. OsiCon[3] EDIFACT software is used as an example. In the end we examine how language technology could be used in the various phases of the reference model to enable multilinguality.

1 INTRODUCTION

In recent years the use of data networks and computerised systems in commerce has been quickly growing. Computerised trading systems are in ever wider use also in international commerce. For instance, in the area of electric power sales the Nordic countries are soon starting a common computerised stock exchange which will be accessed via data networks. Similar exchanges are likely to emerge also on other types of commodities. Electronically handled and open competitive bids are coming, as well. For example, the EC projects ELPRO and TAPPE develop information systems support for public procurement in Europe. The target markets are annually around 600 GigaECU with 16 million participating enterprises and up to 500.000 purchasing entities. Support for local language is becoming increasingly important, while user groups of these systems get wider.

One way to model bargaining processes is to divide them into three phases, which are (1) investigation phase, (2) competitive bid and negotiation phase, and (3) bilateral bargaining phase. Figure 1 illustrates these phases from the viewpoint of a client and a supplier organisation. The figure describes a simple process, where extra negotiations are not needed after receiving bids. This applies to many standard commodities. The black boxes in the information flow arrows facilitate third parties that may be involved.

In the investigation phase a client organisation specifies its needs, conducts market search and analysis, and evaluates previous purchases. A supplier organisation disperses information, analyses market situation and keeps up to date with calls for bids. In this phase services of information brokers and agents can be applied e.g. in finding contacts.

In the competitive bid and negotiation phase the client organisation announces competitive bid, selects most suitable supplier and negotiates bilaterally. The seller organisation analyses the calls for bids, prepares bids, collects more information from the buyer, and negotiates bilaterally about the conditions of the contract.

The final phase, i.e. bilateral bargaining, contains closing of the contract and carrying over the measures implied by it. This includes e.g. logistic processes and money transfers. This phase may involve third parties such as carriers, customs, forwarding agencies, banks, and insurance companies. After care and follow up services are included, as well.

This presentation contains results found by VTT in the GLOSSASOFT project (Telematics LRE 61003) and in the TransCoop project (ESPRIT P8012) of the EC. The former developed methods and guidelines for constructing international software [8,13]. In the latter VTT has analysed interorganisational transactions [14]. VTT has also been investigating the use of intelligent agents in bargaining [12].

This paper introduces next requirements and effects which multilinguality poses to information processing systems. We also outline what multilinguality support means to software construction. An EDIFACT product called OsiCon is used as an example. After that, we study ways how language technology could be used in each phase of the trading model to enable multilinguality.

2 MULTILINGUALITY IN INFORMATION PROCESSING SYSTEMS

In order to reach a large international user base it is necessary that globalisation of an information processing system is done properly. The culturally and linguistically dependent parts of the software must be isolated, a process referred to as internationalisation. These parts include e.g. text manipulation and display, character-encoding methods, collation sequences, hyphenation and morphological rules, formats used for numbers and dates, as well as more subtle cultural conventions such as the use of icons, symbols and colour. The local market requirements for these items are encapsulated in the term locale.

Localisation is the opposite of internationalisation, taking an internationalised system and adding features to match it to the language and culture of the target market. It does not just consist of translating menus, commands and messages into the required language, but includes adaptations for the culture and business practices of the target country.

[1] VTT Information Technology, P.O.Box 1201, FIN-02044 VTT, Finland, email: aarno.lehtola@vtt.fi, tel: +358 0 456 6032, fax: +358 0 456 6027.

[2] On leave from VTT. Current address: Helsinki University of Technology, Neural Networks Research Centre, Rakentajanaukio 2 C, FIN-02150 Espoo, Finland, email: timo.honkela@hut.fi

[3] A commercial EDIFACT software developed at VTT Information Technology. More information on the organisation at the address http://www.vtt.fi/tte

ECAI 96. 12th European Conference on Artificial Intelligence
Edited by W. Wahlster
Published in 1996 by John Wiley & Sons, Ltd.

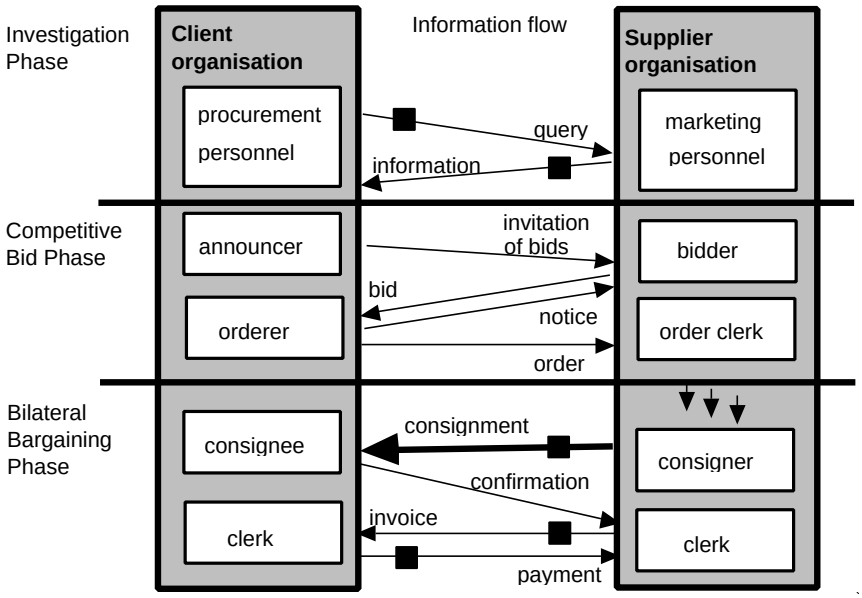

Figure 1. A simplified model of electronic trading

The term globalisation means the whole process starting from feasibility studies, producing an internationalised base version of a software system and then deriving multiple localised versions or a multilingual version from it.

The idealised model of a multilingual information system consists of a locale independent kernel software using locale specific services provided by locale specific program modules. Multilinguality is introduced into this architecture by providing versions of each locale specific service module to each supported language. The kernel software itself is constructed so that it can dynamically keep track and solve the right locales for each service call. The Framework for Global Software (FGS) [8,13] proposes a service architecture with international application programming interfaces (IAPI) suitable for constructing multilingual software (Figure 2). In the FGS, application functionality have been clustered into six groups of interrelated services. A new concept, LocaleContext, addresses the problems of parametering locale specific behaviour of services and realising locale conscious data structures. A LocaleContext is an object which can be defined from scratch or by overriding some properties of an already existing LocaleContext. Hyphenation, spelling, and grammar checking of multilingual documents with varying script systems are examples of internationalised service calls.

The user interface of the OsiCon EDIFACT program was made multilingual following the principles of the FGS. OsiCon provides a graphical user interface for presenting and editing contents of EDIFACT messages, which are widely used in various interorganisational business or administrative transactions. OsiCon is a multiplatform application. It was made multilingual in three respects:

1. GUI controls belonging to the program are dynamically adjusted to right locale,
2. form definitions are dynamically selected according to the selected locale,
3. parts of the form data are translated by simple means, such as correspondence tables and rules for date formats. Locale independent internal data values are separated from locale dependent external presentations.

We also designed facilities for dynamic change of the locale of the form. An active application is able to have simultaneously different locales in different windows with respect to the locales of the forms. The internationalisation solutions are transportable across multiple platforms. Figure 3 illustrates the outlook of the result that satisfied the requirements.

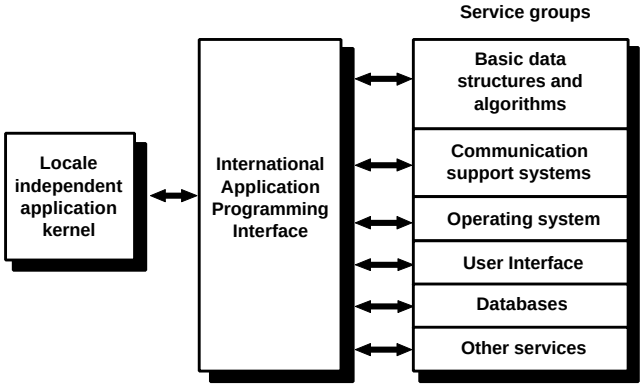

Figure 2. Framework for Global Software.

The case study with OsiCon proved the soundness of the four central design principles of the Framework for Global Software:

1. An internationalised application consists of an application kernel that has been designed in locale independent manner.
2. The application kernel can use internationalised external services via an International Application Programming Interface.
3. An International Application Programming Interface lets parameter locale specific behaviours of external services using LocaleContext structures.
4. Locale dependent data structures are made locale conscious by associating (statically or dynamically) LocaleContext structures to them.

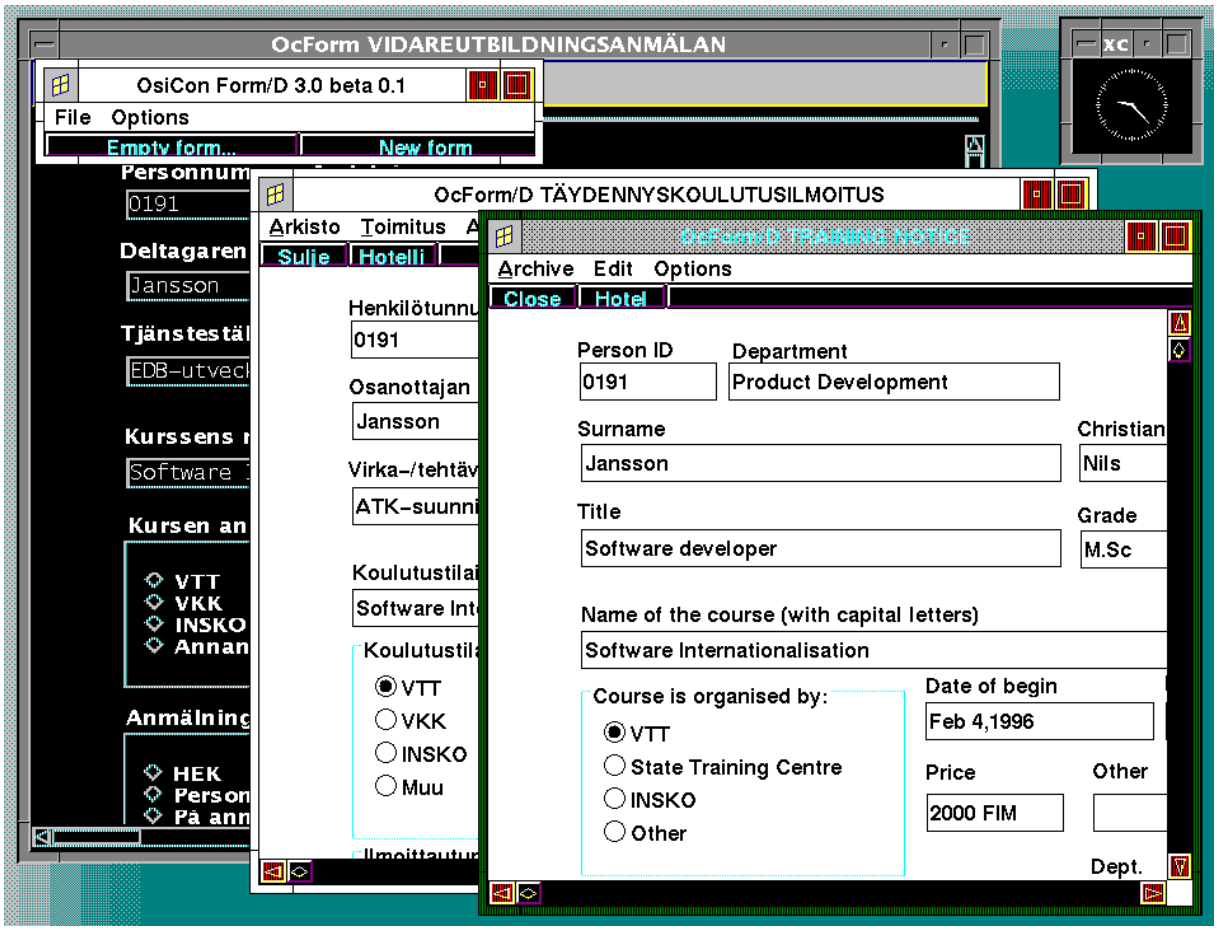

Figure 3. The multilingual user interface of OsiCon EDIFACT program. The same form opened for Swedish, Finnish, and English locale.

Currently there are no comprehensive industry standards available for an IAPI. Fortunately, standards with limited coverage are available (e.g. Unicode [30,31,32], NLS of X/Open) and can be used to implement parts of an IAPI. In everyday work, solutions have to be based on the available APIs provided by systems software and licensed third party tools such as database access and GUI libraries. Many internationalised services can be implemented by introducing extra mapping layers on top of the existing services. However, some of the services are beyond this layering approach and need profound functional extensions to become internationalised (e.g. an internationalised text edit control).

There are several works which cover some or most parts of construction of multilingual software. Some of those that concentrate on a specific platform are [1,3,4,22,25,29], although many of these cover aspects which are important in any platform. A more general approach is taken by [28,33]. Taylor discusses localisation issues rather straightforwardly presenting examples using C. On the other hand, Uren et al. perhaps have their weak points in technical details related to practical programming. [23] concentrates on user interface aspects. [35] describes multilinguality in multimedia being in many aspects rather superficial. Software internationalisation and software engineering aspects are the main focus of [6,10, 34]. A comprehensive collection of Glossasoft project results will appear in [8].

3 LANGUAGE TECHNOLOGY IN MULTILINGUAL TRADING PROCESSES

The need for multilinguality support varies in different phases of bargaining, and may depend much on the commodities that are dealt with. For instance, level of stardardisation (tailor made software vs. shrink wrap software) and price (ship vs. pencil) affect on the number of possible suppliers and on the need for hard negotiating.

Next we consider the activities in multilingual electronic commerce which would benefit of the use of language technology. In the end we outline the use of the technologies in Table 1 to help some of the tasks.

The investigation phase involves many tasks that would justify the use of language technology. Such tasks include the supply of multilingual product information by suppliers, market analysis covering suppliers and buyers in different locales, accessing information on products and suppliers in different locales, and following calls for bids issued in different locales.

The competitive bid and negotiation phase includes, e.g., multilingual access to information on locale specific business rules or legislation, contract models in native languages (tailored, standard), and negotiation messages which may need language translation.

The bilateral bargaining needs also multilingual services, such as multilingual access to the audit trail of the events in the transaction (e.g. checking money transfers afterwards, and customer ac-

count information), and multilingual access of information on the assisting parties (banks, carriers etc.).

Table 1. Language engineering technologies to apply in electronic trading.

Relating two languages	Machine Aided Human Translation	Electronic dictionaries (bi/multilingual; general, special domain, customised, e.g. term banks)
	(MAHT)	Translation memory (db of already translated phrases, sentences, paragraphs; simple/sophisticated search, e.g. fuzzy matching, neural nets)
	Human Aided Machine Translation (HAMT)	Systems requiring pre- and/or post-editing. Batch processing and interactive systems available.
	Fully Automated Machine Translation (FAMT)	Narrow domain, e.g. weather forecasts, or sacrificing accuracy
Within one language	Electronic thesaurus, synonym lists	E.g. synonym lists can be used as building blocks in search engines.
	Morphological modules	Word form analysis and generation, spell checking, hyphenation
	Sentence level syntax analysis	E.g. grammar checkers
	Message generators	E.g. (extended) templates, knowledge based
	Full text search engines/ text filtering agents	May be incorporating morphology processing, knowledge based/fuzzy matching

Machine translation [5,15,24] in its different forms could be applied in multilingual trading processes. Here we outline some of the uses. Fully automated machine translation might be accurate enough for analysing market situation and following up bids in different locales via data networks (Internet, CompuServe, Videotex type of networks etc.). Systems with support for multiple language pairs and with large bilingual general dictionaries, such as the old SYSTRAN system, could be sufficient for an information filtering system. However, the user of such system must be well familiar with the domain of interest, in order to be able to compensate the semantic gaps caused by inaccurate translations. Term dictionaries and translation memories suit well to machine aided human translation of product information for those suppliers who provide it multilingually. Machine translation can help in producing bids in multiple languages when an international competitive bid is issued.

Multilingual user interfaces have many uses in international electronic commerce. Contents of the structured messages (e.g. EDIFACT) in the transactions must be presented to the user through a localised user interface. The user must be given access to the audit trails in the local language. The same applies to many other tasks in electronic trading. Construction of multilingual user interfaces may benefit from machine translation when the static parts are designed. There is also need for dynamic message generation. The GLOSSASOFT project tested two approaches for dynamic message generation [27]. The use of extended message templates and the use of a language independent knowledge base in the generation. The extended templates of the first approach contain calls to the appropriate morphological generation services and this way support synthetic languages, such as Finnish. For each language and each message there is a predefined template. The needed software components are readily available and can be exploited easily. For example, in generating Finnish sentences we have applied the commercially available word form generator FINGEN [19,20]. The extended message catalogue system could be enhanced to generate phrasal constituents, as well. The second approach is more complex. It uses a language independent knowledge base which can be used to generate the same message in different languages and ways according to the users level of expertise, the users task, and the specific style.

Language analysis services could be used e.g. in generating inverted indexes of texts written in inflectional languages. Such preprocessing would enable full text search based on basic word forms. Language syntax analysis and generation are nowadays provided by many sources. The English language constraint grammar parser of University of Helsinki is an example of a publicly accessible service (email: ENGCG@ling.helsinki.fi).

The investigation phase and accessing information on products and suppliers can benefit from advanced information retrieval techniques [2,7,9,16,18,21,26]. The artificial neural network techniques enable development of adaptive systems that can also be used to easily relate partially matching needs and supplies. The Self-Organizing Map (SOM) [17], developed by Teuvo Kohonen in 1982, is a unsupervised neural network algorithm that suits well, among other things, to analysis and visualisation of data acquired from complex processes. In the WEBSOM method [11], Self-Organizing Maps form a system for explorative information retrieval. The neural network algorithm positions encoded documents onto a map that provides a general view into the text collection. The general view visualises similarity relations between the documents on a map display, which can be used in exploring the material rather than having to rely on traditional search expressions. Similar documents become mapped close to each other. The potential of the WEBSOM method is demonstrated in case studies that are available for exploration at the WWW address http://websom.hut.fi/websom/. The WEBSOM method is used to analyse even large document collections of tens of thousands entities. In addition to textual data, analysis of numerical data using SOM is naturally possible.

4 CONCLUSIONS

Natural language processing has many promising application areas in multilingual electronic commerce. Successful application is heavily dependent on how well the language engineering solutions can be integrated or embedded in the main stream of information processing system development. The standards and conventions must be taken into account as well as the compatibility issues. Internet may provide suitable platform for low-cost implementations. On the other hand, the openness of the Internet has often been considered to be a risk, and therefore the main thrust of the electronic commerce is based, e.g., on EDIFACT or on proprietary solutions. To achieve concrete results, co-operation with corporate developers and standardisation bodies is essential.

Data networks give new opportunities to small enterprises that provide specialised language technology based services such as machine translation and multilingual information search and filtering. Large number of customers could take benefit of this kind of services when doing their international trade.

We are continuing to extend the multilinguality support in OsiCon EDIFACT software. Our goal is to make it a good component in multilingual trading systems. However, we are also interested to experiment with more extensive language support to the tasks in trading processes. An international project for demonstrating the wide applicability of language technology in electronic commerce could be well-grounded. For example, a suitable test domain could be tourism.

ACKNOWLEDGEMENTS

The author wishes to thank Prof. Pat Hall, Ray Hudson, Dr. Costas Spyropoulos, Dr. Evangelos Karkaletsis, George Vouros, Dr. Stavros Kokkotos, Prof. Seppo Linnainmaa, Tor Lillqvist, Sakari Kalliomäki, Krista Lagus, Tuula Käpylä and all who have been helping to realise this work.

REFERENCES

[1] Apple Corporation, 1992. *Guide to Macintosh Software Localisation*, Addison-Wesley Publishing Company.

[2] Broglio, J., Callan, J.P., Chmaj, J., Croft, W.B., and Derosa, 1993. Information Retrieval for Customer support, *Proceedings of the 11th National Conference on AI: AI in Service and Support - Bridging the Gap Between Research and Applications*, P. 17-23.

[3] Carter, D.R., 1992. *Writing Localizable Software for the Macintosh*. Addison-Wesley.

[4] Digital Equipment Corporation, 1991. *Digital Guide to Developing International Software*. Digital Press, 381 p.

[5] Dorr, B.J., 1993. Machine Translation: *A View from the Lexicon*. MIT Press, Cambridge, 408 p.

[6] Farid, M. 1990. *Software Engineering Globalization and Localization*. 1990 IEEE International Conference on Systems, Man and Cybernetics, Conference Proceedings, P. 491-495.

[7] Guardalben, G. and Lucarella, D, 1993. Information Retrieval Based on Fuzzy Reasoning, *Data & Knowledge Engineering*, Vol. 10, P. 29-44.

[8] Hall, P., and Hudson, R. (eds.), 1996. *Software Sans Frontieres*, John Wiley and Sons (to be published in 1996).

[9] Hardy, D.R., Schwartz, M.F., 1994, *Customized Information Extraction as a Basis for Resource Discovery*, Technical Report, CU-CS-707-94, Department of Computer Science, University of Colorado at Boulder.

[10] Hewlett Packard, 1992. *How to Develop Internationalised Software*. Hewlett Packard.

[11] Honkela, T., Kaski, S., Lagus, K., and Kohonen, T. (1996): *Newsgroup Exploration with WEBSOM Method and Browsing Interface*. Report A32, Helsinki University of Technology, Laboratory of Computer and Information Science.

[12] Honkela, T., Lehtola, A., and Linnainmaa S., 1994. On the Use of Intelligent Agents in Business Transactions. *Proceedings of the Conference on Artificial Intelligence in Finland*. Finnish Artificial Intelligence Society, P. 137-139.

[13] Hudson, R., and Lehtola, A., 1995. *GLOSSASOFT: Methods and Guidelines for Software Internationalisation and Localisation*. Proceedings of the Language Engineering Convention, London, October 1995, 8 p.

[14] Juopperi, J., Lehtola, A, Pihlajamaa, O., Sladek, A., and Veijalainen J., 1995. Usability of Some Workflow Products in an Inter-organizational Setting. *Proceedings of the IFIP Working Conference on Information System Development for Decentralised Organisations*, Chapman & Hall, London, P. 19-34.

[15] Hutchins, W.J. and Somers, H.L., 1992. *An Introduction to Machine Translation*, Academic Press, San Diego, 362 p.

[16] Knoblock, C.A., Arens, Y., and Hsu, C.N., 1994. Cooperating agents for Information Retrieval, *Proceedings of the Second International Conference on Cooperative Information Systems*.

[17] Kohonen, T., 1995. *Self-Organizing Maps*. Springer-Verlag, Berlin, 362 p.

[18] Lauzon, D. and Rose, T., 1994. Task-Oriented and Similarity-Based Retrieval, *Proceedings of the Ninth Knowledge-based Software engineering Conference*, P. 98-107.

[19] Lingsoft Ltd., 1994. *FINGEN Reference Manual*.

[20] Lingsoft Ltd., 1996. http://www.lingsoft.fi/

[21] Loeffen, A., 1994. Text Databases: A Survey of Text Models and Systems, *SIGMOD Record*, Vol. 23, No. 1, March 1994, P. 97-106.

[22] Microsoft Corporation, 1992. *The Windows Interface: International Handbook for Software Design*. Microsoft Press, 239 p.

[23] Nielsen J (ed.) 1990. *Designing user interfaces for international use*. Elsevier, 230 p.

[24] Nirenburg, S. (ed.) 1987. Machine Translation: Theoretical and methodological issues. Cambridge University Press, Cambridge, 350 p.

[25] O'Donnell, S.M., 1994. *Programming for the World*. Prentice Hall.

[26] Salton, G., Allan J., and Buckley, C., 1994. Automatic Structuring and Retrieval of Large text files, *Communication of the ACM*, Vol. 37, No. 2, February 1994, P. 97-108.

[27] Spyropoulos, C.D., Karkaletsis, E.A., Vouros, G.A., Honkela, T., Lagus, K., and Lehtola, A., 1995. *Investigating On-Line Message Generation in Software Applications: The GLOSSASOFT Results.* ERCIM Workshop on "Towards User Interfaces for All: Current Efforts and Future Trends", Heraklion, Greece, October 1995, 13 p.

[28] Taylor, D., 1992. *Global Software*. Springer-Verlag, 319 p.

[29] Ternasky, J. & Ressler, B.K., 1993. *Writing localizable applications*. Apple Technical Journal, Issue 14, P. 7-33.

[30] Unicode Consortium, 1990. *The Unicode Standard, Version 1.0*. Volume 1, Addison-Wesley, 682 p.

[31] Unicode Consortium, 1992. *The Unicode Standard, Version 1.0*. Volume 2, Addison-Wesley, 439 p.

[32] Unicode Consortium, Caldwell, J.T., 1991. *Unicode.* in Characters and Computers, Main V.H. and Liu Y. (ed.), IOS Press.

[33] Uren, E., Howard, R., Perinotti, T., 1993. *Software Internationalisation and Localisation - An Introduction*. Van Nostrand Reinhold, 300 p.

[34] X/Open Company, 1993. *Internationalisation Guide Version 2*. X/Open Company Ltd., 180 p.

[35] Yazdani, M., 1993. *Multilingual Multimedia*. Intellect, 210 p.