Market research for requirements analysis using linguistic tools[1]

Luisa Mich*, Mariangela Franch°, Pierluigi Novi Inverardi°
** Department of Computer and Telecommunication Technology, University of Trento*
Via Sommarive 14, I 38050 Trento (I) - Tel. +39-0461-882087 – Fax +39-0461-882093
E-mail: mich@dit.unitn.it
° Department of Computer and Management Sciences, University of Trento
Via Inama 5, 38100 Trento (I) - Tel: +39 0461 88213/2287 - Fax: +39 0461 882124
E-mail: franch@cs.unitn.it, inverard@cs.unitn.it

Abstract

Numerous studies in recent months have proposed the use of linguistic instruments to support requirements analysis. There are two main reasons for this: (i) the progress made in natural language processing, (ii) the need to provide the developers of software systems with support in the early phases of requirements definition and conceptual modelling. This paper presents the results of an online market research intended (a) to assess the economic advantages of developing a CASE tool that integrates linguistic analysis techniques for documents written in natural language, and (b) to verify the existence of potential demand for such a tool. The research included a study of the language – ranging from completely natural to highly restricted – used in documents available for requirements analysis, an important factor given that on a technological level there is a trade-off between the language used and the performance of the linguistic instruments. To determine the potential demand for such tool, some of the survey questions dealt with the adoption of development methodologies and consequently with models and support tools; other questions referred to activities deemed critical by the companies involved. Through statistical correspondence analysis of the responses, we were able to outline two "profiles" of companies that correspond to two potential market niches which are characterised by their very different approach to software development.

Keywords: Market research, Potential demand, NLP-based CASE tools, Requirements Analysis, Conceptual modelling

1. Objectives and structure of the paper

Premise

This paper presents the results of an online market research conducted in the spring and summer of 1999 by the Department of Computer and Management Sciences of Trento University, Italy. The study is part of a larger project whose principal aim is to identify the advantages and disadvantages of market research done online with respect to traditional methods and channels, and to look at its applicability in diverse product markets[2]. In methodological terms the objective of the research presented in this paper was to demonstrate the benefits of conducting online market studies for innovative products. Problems with such innovative products derive firstly from the fact that their characteristics cannot be thoroughly defined before conducting the research, and secondly their availability in commercial form usually requires further sizeable investments in research and trialling. Both of these issues are critical for CASE (Computer Aided Software Engineering) tools, which use linguistic instruments to analyse documents in natural language, and are therefore based on technologies for natural language processing (NLP) developed in the field of Artificial

[1] Submitted to REJ.
[2] Multi-year project funded by the Department of Computer and Management Sciences of Trento University.

Intelligence. Working from the perspective of a company attempting to decide which products to develop (from among different projects related to NLP-based applications), our objective was to evaluate potential demand for NLP-based CASE tools. In conducting the study we made the reasonable assumption that the respondents (people involved in developing software systems) could be contacted easily by Internet; this prerequisite could not be guaranteed principally at a national level for other sectors studied previously (e.g., tourism or electronic commerce of groceries)[3]. At the same time, a certain predisposition not to participate in the study was to be expected, whether because of time constraints (noted even at the initial explorative interviews) or because of an already high level of saturation. In fact, both of these assumptions were confirmed during the course of the research. Nonetheless, we emphasise that this paper focuses on the results of the actual content of the research, and hereinafter we describe only methodological aspects that are pertinent to the interpretation of the results obtained[4].

Objectives

As previously mentioned, the aim of the research was to analyse the potential demand for a CASE tool integrating linguistic instruments as a support to requirements analysis [2]. To give the context in which such a tool could be designed and used, the following paragraph first describes the role of natural language in requirements engineering and then classifies the possible applications of linguistic instruments, making reference to the architecture of an ideal NLP system and to the three fundamental activities of requirements analysis: Elicitation, Modelling and Validation [3]. Our market research refers principally to the support of conceptual modelling, an activity that to benefit from the use of linguistic instruments requires the design of a modelling module. The other activities could be supported by existing functionalities of an NLP system, with varying levels of performance.

It was found early in the study that none of the commercial CASE tools exploited linguistic instruments to support requirements modelling [4]; this meant, therefore, that the market research was to focus on a new product whose features could not be defined in relation to similar existing products (analysis of the competition). Numerous research projects do exist in this area, however, and serve as a testimony of the considerable interest in the use of linguistic instruments in requirements engineering[5]. The common objective is to carry out a linguistic analysis of requirements documents in order to produce conceptual models of them[6]. Among the most recent projects, as an example, we can cite those described in [8,9]. While a complete review is beyond the scope of this paper, it is worth noting how different approaches can be analysed by looking at two principal aspects (depending on the characteristics of the linguistic tools adopted):

a) how “natural” the input language is, which is normally subject to restrictions regarding grammar, vocabulary, or both;
b) how much intervention by an analyst is needed in order to process "semi-automatically" the text or to identify the key elements for conceptual modelling.

The survey described in this paper focuses on the first of these points, one that we deem of vital importanc because whatever the approach adopted, the "naturalness" of the language

[3] Some comparisons deriving from our research are described in [1].
[4] For further study of issues related to online market research, the interested reader can refer to the literature (see for example, the publications found at ESOMAR - European Society for Opinion and Marketing Research - http://www.esomar.nl/).
[5] See [5,6]. A bibliography is available at http://nl-oops.cs.unitn.it.
[6] The first proposals to use linguistic criteria for the extraction of entities and relations, and then objects and associations, from narrative descriptions of requirements date from the 1980s [7].

directly affects the amount of effort needed to extract useful information from the documents. First, it was necessary to establish whether the documents gathered in the requirements elicitation phase were in 'real' natural language or in some type of restricted language, and if they were in natural language, whether the user or customer could be asked to describe the requirements using a more restricted language. In fact, if the documents are written in a 'controlled' language (restrictions on grammar or vocabulary), information can be extracted using syntactic or 'shallow' techniques, such as parse trees[7]. To obtain equivalent performances with documents in unrestricted natural language it is necessary to have a semantic representation of knowledge that embeds reasoning techniques. Such applications are currently being studied[8]. Moreover, the language used in the documents can be more or less linked to a particular application domain (for example, software for telecommunications), thus determining the degree of specialisation of the support linguistic tool to be used in the conceptual analysis, and therefore of its knowledge base. In other words, hypothesizing that the basic NLP technologies are available, for a company that must decide whether or not to invest in the development of an NLP-based tool for requirements analysis, it is important to establish first if it is possible to design and realise a general-purpose tool to support software development for different application domains or if instead it is necessary to make further investments later to customize the tool for the different companies or customers it will eventually serve. These are all essential considerations in determining the investment necessary to convert a research prototype - like those developed in the existing research projects - into a commercial tool.

Results of preliminary interviews as well as the state of the art of existing prototypes led us to decide not to investigate the degree of analyst intervention requested nor performance requested of the tool (point b: we limit ourselves on this point to giving some general findings that emerged while conducting the research). To do so would have required further investment in a more extensive market research; such study would be justifiable only with a positive outcome, certainly not guaranteed, relative to the issues related to point a). Moreover, to assess the potential market for an NLP-based tool for requirements analysis, we studied aspects related to the diffusion of methods and instruments of software engineering. In particular, we intended to verify whether requirements analysis is in fact considered critical in relation to other important activities in software development (testing, documentation, etc.).

Structure of the paper

The paper is organised as follows: the next section describes the context of an NLP-enabled CASE tool and summarises possible applications of linguistic tools for requirements engineering. This provides information on the design of the questionnaire and the eventual interpretation of the results. The third section outlines the plan of the market research, noting the different phases and focusing on the questionnaire and on the characteristics of the respondents. The main results of the online survey are presented in the fourth section, where they are analysed using a statistical technique referred to as correspondence analysis. The profiles obtained have revealed the existence of two market niches characterised by their diverse approaches to software development. Finally, some observations are given regarding the characteristics of the survey and the extendibility of the results. The conclusions summarise how the results of the survey can be used by those who develop software in general, and by those who design tools and environments for requirements analysis in particular.

[7] Included in this category are, for example, the instruments described in [10] and [11].

[8] For example, to recognise if *Washington* is the name of a person, of an airport, or of a city in a given document requires a semantic approach. Limitations on space do not permit a deeper discussion of this issue here; see for example [12].

2. The role of natural language in requirements engineering

Much has been written on the importance of requirements analysis. In order to show why environments and tools to support such analysis are less satisfactory than those available for the other phases of the software life-cycle, we shall briefly review the distinctive features of requirements engineering, defined as:

> "the systematic approach of developing requirements through an iterative cooperative process of analysing the problem, documenting the resulting observations in a variety of representation formats, and checking the accuracy of the understanding gained".
> [3, p 13].

Thus evident is the central importance of communication[9] and knowledge. Compared with other phases of software engineering, requirements analysis and conceptual modelling [15] present unique difficulties. Many of the activities involved are cognitive and require creativity as well as knowledge about information technologies and the application domain. Moreover, the recent advances brought about by business process re-engineering (BPR) and the inclusion of innovative components in information systems are broadening the scope of projects. As a consequence, the number of the actors, interactions and languages involved have increased. Completing the picture are the needs of companies, which operate at ever higher levels of competitiveness and which demand increasingly flexible information systems.

In this context, the use of linguistic tools – more precisely of NLP systems – to support the development of software systems in general and requirements analysis in particular, may help the analyst to:

- concentrate on the problem rather than on the modelling;
- interact with other actors;
- take into account the various kinds of requirements (organisational, functional, etc.);
- achieve traceability as from the first documents produced;
- manage more efficiently the problem of the changing user requirements.[10]

As regards the possible applications of NLP systems to requirements engineering, it is worth noting that they are able to process both vocal and textual input, sometimes imposing restrictions such as limiting the vocabulary or the grammar.

NLP systems can be used to obtain, with different levels of performance, essentially three types of output:

- syntactic, semantic or pragmatic analysis;
- text either in the same language or another one, natural or artificial;
- syntheses in the form of differently structured summaries or templates.

Figure 1 is a simplified scheme of an ideal general-purpose NLP system. It is important to remember that the systems for real applications are usually highly dependent on the task and on the domain[11].

---

[9] "The hard part, and the true essence of requirements, is trying to understand your customer's needs. A person involved in requirements needs human skills, communication skills, understanding skills, feeling skills, listening skills" [13]. See also [14].

[10] For a recent study on why it is impossible for users to know their requirements beforehand, see [16].

[11] On this point, see, for example, the tasks required by the MUC competitions (*Message Understanding Competition)* organised by the DARPA (Defense Advanced Research Projects Agency) [17].

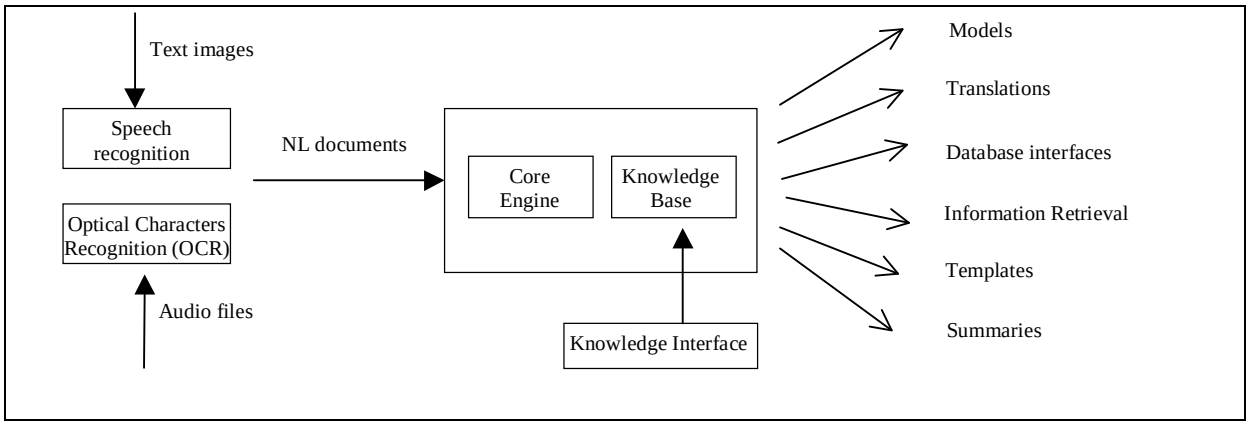


Figure 1 – The architecture of a general-purpose NLP system

With reference to this scheme, linguistic tools of differing complexity and especially of differing maturity can be used:

a) in the requirements elicitation phase:
   - to facilitate the digitising of requirements documents using speech recognition systems or NLP-based interrogation interfaces;
   - to reveal ambiguities and contradictions in documents describing user needs (see for example, [12,18,19];
   - to design questionnaires or interviews, by verifying the ambiguity of the questions;
   - for automatic analysis of replies to open-ended questions, interpreting and classifying their contents [20].

b) to model requirements by extracting (directly from the text) the descriptions of the elements to include in the conceptual models envisaged by the development method adopted, in particular UML (Unified Modelling Language)[12] diagrams (see Figure 2).

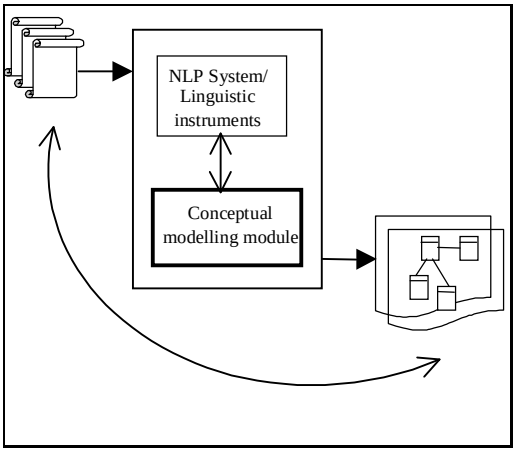


Figure 2 – The models generation process

c) to support requirements validation, by exploiting the generation functionality of NLP systems to produce descriptions in natural language based on the structures used to represent knowledge.

A complete vision requires noting that NLP tools can also be used for documentation,

[12] The official documents of the UML's specifications can be find on the OMG (Object Management Group) web site: http://www.omg.org.

generating reports on the various stages of requirements collection and modelling; for traceability, allowing a link to be maintained between the texts used and the models produced; and for the translation of documents into various languages, something that becomes increasingly necessary in the design of international information systems.

The survey described in this paper concerns the second of these points, that is, the use of NLP techniques to support the development of conceptual models, given that it requires the design of a modelling module. All the other activities could be supported by existing functionalities of an ideal NLP system, albeit with different performances. The most important assumption is that the requirements documents, once analysed, can contribute to a "knowledge base" from which to extract elements deemed useful for modelling activities. There are two important aspects to note regarding projects for developing this type of instrument: i) many of these projects are based on *ad hoc* NLP systems, and therefore do not appear to correspond to the requirements for scalability and robustness of real applications; ii) given the complexity of natural language, almost all of them expect that documents will be written in restricted language or that some revision of the text will have taken place before undergoing the automatic analysis. These two facts are worth remembering when interpreting the results of market research and when estimating potential investments in NLP technologies, and certainly when developing a CASE module to support requirements analysis.

3. Plan and realisation of the market research

The decision to investigate the market for an NLP-based tool for requirements analysis was taken in the context of a joint research project with the Department of Computer Sciences of Durham University (UK) in which a prototype was developed of a CASE tool - called NL-OOPS -,[13] for requirements modelling according to the object-oriented approach [21,22].

The market research described here was based on the administration of a questionnaire whose design required consideration of the experience gained throughout the development of NL-OOPS, and of the methodology and techniques of online market research. Specifically, the research progressed in the following phases:

- preliminary survey
- identification of interview subjects
- designing and testing of the questionnaire
- selection of the contact method
- distribution of the questionnaire and reminders
- collection and analysis of the data.

A description of each phase follows, giving greater emphasis to the third phase (designing the questionnaire) and to the final stage (analysis of data).

Preliminary survey The first step in the research project was to create a focus group composed of both companies that develop linguistic instruments as well as big and small businesses that develop software or offer services linked to the introduction of information technologies in the workplace. The goal of this phase was to collect information about the users' needs that could be satisfied with an NLP-based CASE tool and to gather other information useful in designing the questionnaire. The researchers were immediately confronted with pessimistic views of tools which use NLP techniques to support requirements analysis. In particular, some focus group members expressed serious doubts that the language in the documents gathered for requirements analysis was sufficiently ‘natural’ to justify the adoption of a tool based on NLP techniques. Others questioned the technical feasibility of

[13] Natural Language – Object-Oriented Production System, http://nl-oops.cs.unitn.it.

such tools, citing their own unsatisfatory experiences with other NLP applications such as translation programs.

Identification of interview subjects In accordance with the objective of the study, the questionnaire was directed principally to persons involved in software development, and in addition to managers responsible for important decisions regarding the process of software development, including the decision to adopt methodologies and support instruments. From a statistical viewpoint, when dealing with a survey conducted via Internet, one of the main problems is to establish the degree to which the sample is representative of the target population, in this case the people or companies involved in software development. On one hand, it is reasonable to assume that the intended respondents are reachable by Internet, while on the other hand the population has characteristics (number, size, geographic distribution, etc.) that are not documented. Given this and also considering the chosen methods of contact, the approach to the study is conceptually similar to a sequential sampling. Statistically, this would classify it as a descriptive study, and as such requires caution when extending the results outside of the survey sample.

Designing and testing of the questionnaire Again considering the objectives of the study, in terms of both methodology and content, the survey was conducted only via Internet and it consisted of a questionnaire on a Web page[14] (see appendix A). This choice was the driving force during the design and testing stage, the aim being to have a concise questionnaire with closed-ended questions in language as clear as possible.[15] As for the questions themselves, the choices were made as logical and pertinent issues emerged throughout the course of the focus group. After a phase of testing in which the questionnaire underwent the scrutiny - first directly and then online - of a select group of analysts and project managers, the final version was produced. The final questionnaire was divided into two sections, for a total of eighteen questions, and a final open question for further observations. The first group consisted of questions relating to the company (questions 1 – 4) and to the respondent (questions 5 and 6). The second part investigated processes of software production, so that one group of questions concerned the use of methodologies (questions 7 – 10) and tools (questions 13 and 14) in software development; another group dealt with documents used in requirements analysis (questions 11, 12 and 15) and the last three were about the efficiency of the development process (questions 16, 17 and 18). The respondents were also asked if they were interested in obtaining the results of the research or in viewing a demonstration of a prototype of an NLP-based CASE tool. The decision to introduce questions associated with an engineering approach to software development was made after verifying the possibility of using existing data. Surprisingly,[16] only a small amount of data was found, whether for the diffusion of object-oriented methodology or for the use of 'classic' models such as the entity-relationships models. These are important because the early research and conceptual models for linguistic analysis of requirements [7] looked to produce entity-relationships diagrams; moreover, these models can be seen as a particular case of the class models foreseen by the object-oriented approach. As regards the market for CASE tools,[17] in many cases they did not meet expectations and as a consequence did not have the desired market success [25]. We will have to wait for the adoption of the UML – developed about one year before the present research project began – as a standard for conceptual modelling by the OMG (Object Management Group); only then will there be a significant growth in the market for CASE tools, repackaged

[14] The questionnaire is available along with the data gathered and other related research material at http://on-line.cs.unitn.it.

[15] For example, a questionnaire like the one used for the survey described in [23] would have to be radically altered to be used on-line.

[16] In light of the observations in [24], this may not be so surprising.

[17] The choice of tools for question 14 was made on the basis of sales data for a period prior to the study.

and renamed as object modelling tools or visual modelling tools. In short, the scarcity of data on the penetration and role of an engineering approach to software development influenced the choice of questions for the survey, but also, as we shall see, the ability to validate and extend the results.

The questions considered most important to verifying the existence of a market niche for an NLP-based CASE tool are those related to the documents used to collect requirements. In fact, as we have already seen, if documents are in real NL, an even more sophisticated (and costly) technology is needed to develop an environment that effectively supports analysis using linguistic instruments. It is therefore useful to establish whether the company is in a position to require clients or analysts to describe requirements in a restricted language. Typical restrictions can regard: a) grammar - aiming to have syntactic constructions that are easier to analyse by requiring, for example, shorter phrases, using the active voice, by avoiding anaphorical references, etc.; b) vocabulary - aiming to reduce ambiguity of terms. Moreover, in order to determine the degree of customisation required of a possible NLP-based tool, further questions dealt with the level of specialisation of the terminology and the domain knowledge required to develop the software.

In the questions related to the efficiency of production processes, respondents were asked in particular about the improvements that they would like to see (choosing from a list of eight possible activities considered critical, two of which are fundamental for the phase of requirements analysis) and how they could be achieved, the choice being among 'internal delegation', 'outsourcing' and 'automation'. The final question was designed to ascertain whether the company was able to deliver the software systems or products without delays. Finally, in keeping with the general rule of market research, an incentive to participate was provided in the form of a random drawing among respondents for tickets to an opera performance at the Arena in Verona.[18]

Selection of the contact method The objectives of the research and the characteristics of the tool inherently required a contact method that would permit efficient use of time and resources while at the same time reach the largest number of potential respondents. On this point, to take into account the fact that there is a high level of saturation - due to the large number of such survey requests that the respondents receive - we had initially thought to send the questionnaire to some specialised newsgroups,[19] highlighting the academic nature of the research. In the first phase we identified three newsgroups whose work is related to the research topic (comp.object, comp.software-eng, alt.comp.software-tools); another twenty-one newsgroups were later added to the list (the complete list is available at http://on-line.cs.unitn.it). Nonetheless, after this method of contact proved less successful than expected,[20] we decided to contact the companies directly by email, supplying them with the address of the Web page where they could find and complete the questionnaire. The companies' addresses were acquired online using search engines, in particular a directory of Yahoo![21] (Computer > Software > Developers).

Distributing the questionnaire and reminders As described above, the questionnaire was administered in two different ways. In a first phase it was publicised on a number of newsgroups devoted to software development (resulting in 44 completed questionnaires and 39 software companies) and in the second, requests to take part in the survey were sent by e-

[18] Because the survey concluded at the end of the Arena opera season, the tickets were replaced by CDs of opera music by Verdi.

[19] One of the aims of the survey, in fact, was to investigate the conditions under which newsgroups can be used to carry out online surveys.

[20] Limited number of questionnaires obtained (44) and accusations of spamming.

[21] http://www.yahoo.com.

mail to 1541 addresses corresponding to 1234 software companies. By means of this second method, 107 completed questionnaires corresponding to 103 companies, were obtained. To get these results, it was necessary in many cases to send a message reminding the receiver to participate in the study, yet at the same time allowing him or her to explain the decision not to complete the questionnaire. Reasons given for not completing the questionnaire frequently referred to a lack of time and the large number of requests of this kind received (the email messages sent are accessible online at http://on-line.cs.unitn.it). In addition, several addresses were incorrect, although the percentage was rather low (7.6%, 6.1% if calculated by number of companies).[22] Consequently, the number of valid contacts was 1424, corresponding to 1159 companies.

Collection and analysis of the data A total of 151 questionnaires were returned, 91% within five days of sending the initial request or the questionnaire itself. The response rate calculated for the questionnaires sent via email was around 8%. This can be regarded as a satisfactory result when compared with traditional surveys conducted by post or fax, and with other surveys of software development, for which the response rate has been 3% [25].[23] In strictly statistical terms, the group of companies contacted – while constituting in itself a large number - cannot be taken as a representative sample of the population of software development companies. Given this, it is important that the results be interpreted in a descriptive mode, thus requiring caution in extending them. We shall see, however, that for some questions the quality of the survey results can be evaluated by comparing them with those obtained from other surveys and with data relative to the CASE market. The results of these comparisons are provided at the end of the next paragraph.

On a methodological level, the use of newsgroups confirmed that little effort was required to ask respondents to participate, but the low number of questionnaires completed may nullify this advantage. Furthermore, the use of newsgroups should be evaluated on the basis of the following factors: level of specialisation,[24] number of messages, and presence of a moderator. In light of the results of our survey, in the case of very specialised newsgroups, even if the contents of the survey are relevant to them, in order to increase the response rate it is advisable to ask for the moderator's consent, or to identify one or more newsgroup leaders who can legitimate the survey with their participation.

The initial analysis noted the geographic distribution of the respondents, most of whom are residents of European states or of North America (see Figure 3). This first result of the research is supported by the analysis of similarities among different geographic distributions (using appropriate indices) showing, in fact, that these markets have similar characteristics. Given this, we present here results of the survey in its entirety, highlighting only those aspects where geographic area of residence influenced the responses.

[22] This is a rather high percentage, bearing in mind that they were collected from the homepages of official company websites. Another survey carried out in the same period on winter tourism, where the addresses were provided by a specialized magazine, found a very similar percentage of wrong addresses (8.9%), but the amount can be much higher. For example, in a survey of Internet users carried out in 1996, 35% of a total 1221 addresses were found to be wrong [26].

[23] This was the minimum value for the traditional-type surveys, which achieved a maximum response rate of 20%. In the survey described by Glass and Howard [25], the percentage rose to 17% after the questionnaire mailings were supplemented by telephone contacts with fax follow-up.

[24] For a survey on virtual supermarkets, a message was sent to 6 newsgroups obtaining 100 completed questionnaires.

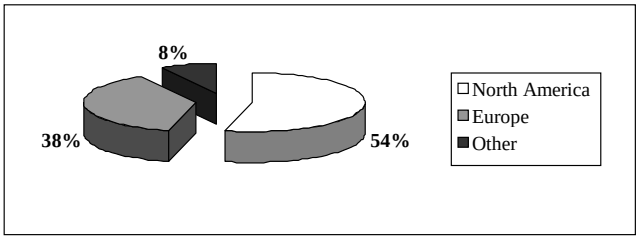


Figure 3 - The respondents by geographical area of residence

Eighty-six percent of the respondents fill roles relating to software development projects, 68% having occupied the role for more than six years.[25] Moreover, as to be expected, length of service influenced the position occupied in the company, so that programming work was more frequently performed by persons employed for the shortest periods, while those who had worked in their companies for 6-10 years were almost uniformly distributed among roles. To be noted is that the majority of European respondents selected 'System Engineer/Architect' but their American counterparts selected 'Project Manager', which may have been because different terms are used to denote the same role in the two areas. Some 29% of the respondents worked in companies with more than one hundred employees, although small-sized companies were also well represented (Table 1).

How many employees and consultants are there in your company?				
1-5	**6-20**	**21-50**	**51-100**	**More than 100**
27%	24%	15%	5%	29%

Table 1 – Company size

The core business of the companies surveyed in 77% of the cases is 'SW development' and in 23% is 'Web sites' or 'Other'. As expected, the highest percentage of companies engaged in other types of business (or rather, *also* in other types of business) consisted of larger-sized ones. As regards the type of software produced, 42% of the companies developed software for niche markets (Figure 4), with a high 48% for North America. This may be due to the presence of a larger number of small-sized companies, given that 59% of companies with five or fewer employees, and 24% of those with more than 100, operated in niche markets. Software products were mostly sold to the end-user: 84%;[26] only 13% sold to another software company, and 3% to software shops. Interestingly, all the companies that developed Web sites sold their products directly to the end-users, given the nature of this type of product.

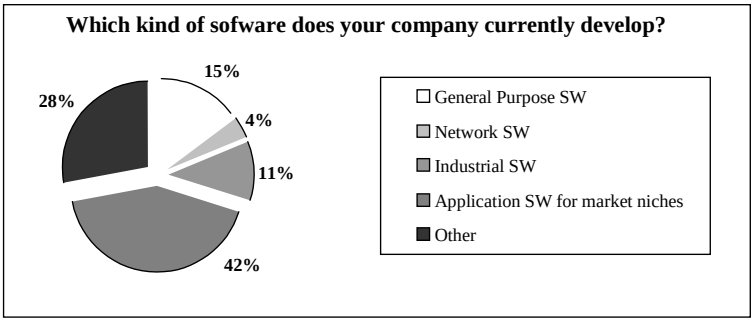


Figure 4 – Type of software

[25] All the percentages were calculated on the total number of respondents who answered the relative questions, with non-replies omitted.

[26] Further investigation of this aspect would require knowledge of the number and size of the companies' customers. This, however, is beyond the scope of our survey.

The next paragraph provides a detailed analysis of the results of research into the existence of a potential market for an innovative tool to support conceptual analysis—a tool that has the capability to analyse documents written in varying levels of natural language.

4. The results of the survey and the potential demand for an NLP-based tool to support requirements analysis

We can identify three groups of elements that are useful in evaluating potential demand[27] for a CASE tool to support requirements analysis for documents written in natural language. They can be described as follows, taking into account their interrelatedness:

[1] The market for instruments supporting software development and requirements modelling. How extensive is the market? How much competition is there? Do software developers use CASE tools? If so, which ones? (Normally the use of a CASE tool presupposes the adoption of a development methodology.) This last point was important both for establishing which conceptual models the tool should support (an aspect that became less important with the diffusion of UML[28]), and for reasons of compatibility and integration with existing tools[29]. Some information on this point could be obtained by means of the data on sales of CASE tools, but one question on this topic was inserted regarding the tools supporting requirements analysis and top-level design.

[2] Features of the tool. The requirements principally influencing the investments necessary to develop a tool for requirements analysis based on linguistic instruments are (a) the language found in the documents gathered in the elicitation of requirements phase, crucial in identifying appropriate techniques and linguistic instruments, and (b) the degree of specialised domain knowledge required of the tool, which determines the degree of specialisation required of the producer of the CASE tool (generality). Also**,** given the state of the art of linguistic instruments, an important consideration is the performance required of the tool; in other words, how 'good' does it have to be to merit purchase?[30]

[3] Requirements analysis viewed as crucial. This is a vital element in identifying potential market niches and in ascertaining the propension of users to invest in a tool that supports requirements analysis, as well as their willingness and ability to accept the changes that accompany the adoption of a new tool. Companies that have an engineering approach to software development have highly standardised processes and should therefore consider the activities lacking structure or support as crucial points demanding attention. A company employing a more informal or 'craft' process would not necessarily share this concern but would, however, be more interested in the use of natural language.

To glean the most useful information on these three points, we analysed the completed questionnaires in two phases. In the first phase we looked at individual answers, studying reciprocal relationships and dependencies. In the second phase we applied correspondence analysis [28], aiming to unveil the existence of profiles corresponding to potential market niches for an innovative CASE tool.

---

[27] For an introduction to the evaluation of potential demand, see for example, [27].

[28] In the past, the need to support different graphic notations was a drawback to the market for CASE, in that it required producers to choose which notation to support with their own tools, or to absorb the higher cost of developing different versions.

[29] A CASE based on linguistic techniques for object-oriented analysis does not necessarily require the realisation of an entire support environment, but rather can be seen as a module that can be integrated with an existing product.

[30] A study of the 'robustness' is of utmost importance also to establish the degree of analyst intervention required in developing requirements models, and should be conducted using a prototype of the tool. See also point (b) of the introduction and conclusion.

[1] As for the use of a tool supporting requirements analysis and top-level design, only 30% replied positively. As was expected, greater use was made of these tools in large-sized companies, reaching 51% in those with more than one hundred employees, as is shown in the table of conditional distributions (Table 2). Not surprisingly, the use of these tools increases with length of service (rising from 17% to 36%) with analysts as the category of employee using them most frequently.

Table 2 - Use of tools for requirements analysis and top-level design by company size

	How many employees and consultants are there in your company?				
Do you use any tool supporting requirements analysis and top-level design?	**1 – 5**	**6 – 20**	**21 – 50**	**51 – 100**	**More than 100**
Yes	16%	18%	33%	33%	51%
No	84%	82%	67%	67%	49%

Moreover, 84% of the respondents stated that they used specific methodologies for software development. Size was a determining characteristic here, 78% of companies with five or fewer employees using specific methodologies and 93% for those with more than 100. The type of software or the sales channel does not significantly influence the use of methodologies, although role and experience seems to do so to some extent.

The best known diagrams for data modelling, entity-relationship (E-R) diagrams were used by 63% of respondents who adopted a methodology. Moreover, smaller company size corresponded to their more infrequent use (52% in companies with fewer than five employees, 73% in those with more than 100). The use of E-R diagrams was substantially greater among respondents who had worked longer in the computer business (increasing from 35% among those who had worked in the field for less than three years to 66% among those who had done so for more than ten). Finally, as regards the type of software, E-R diagrams were used to very different extents by respondents who developed general-purpose software (93%) and by those who developed network software (25%), while there were no substantial differences as far as the other items are concerned.

The percentage of respondents who used an object-oriented (OO) method was 68%, a percentage similar to that of E-R diagram users. The classification by company size shows a difference between companies with five or fewer employees (60% of which used OO methods) and those with more than 100 (74% of which do so). There are no significant variations with respect to years of experience, while there is a closer association with the position occupied within the company: the percentages ranged from 45% for programmers to 78% for system engineers/architects. An interesting comparison can be made in Table 3, where one notes that those who adopt OO methods were already accustomed to using E-R diagrams, thus indicating that they seemed more inclined to use an OO approach.

Table 3 – Entity-Relationship diagrams and Object-Oriented Methods

	Do you use Entity-Relationship diagrams to model your data requirements?	
Do you use an OO Method?	**Yes**	**No**
Yes	69%	63%
No	31%	37%

As far as the most widely used OO method, 77% of respondents who replied in the affirmative to the previous question declared that they use UML. This is a result which confirms the affirmation of UML as the industrial standard for OO modelling. It is worth

mentioning that the survey was carried out approximately one and a half years after the adoption of UML by the OMG.

It also emerged that the great majority of the respondents who said that they did not use methodologies did not use tools for requirements analysis and top-level design either (90%): indeed, there is an association between the use of methodologies and CASE tools. Another finding to be emphasised is the connection between the use of CASE tools for requirements analysis or top-level design and the type of language employed in documents. Not unexpectedly, these tools were used more frequently when the language was more formal (24% with 'common natural language' and 63% with 'formalised language'). Even if these results should be treated with caution, given the low number of companies surveyed, they seemingly confirm the inability of currently available CASE tools to meet the needs of natural language processing by yielding environments that are effectively useful. As far as the tools used are concerned, 52% of respondents who replied in the affirmative to the previous question declared that they used Rational Rose.[31] Rational Rose was the tool with the highest market share both worldwide and in Europe.[32] In 1998 it accounted for 33% of the market, with an increase of 79% on the previous year.[33] For this reason, the percentage found by our survey (52% for the year 1999) appears to be as one would expect.[34]

[2] As noted, the type of language used in requirements documents determines the complexity of the linguistic instruments and of the NLP techniques to be used. When documents are written in a constrained language (a subset of NL) – which imposes restrictions on the grammar or the vocabulary, or both – simpler and more mature linguistic tools can be used. However, it is not usually possible to impose restrictions on the language employed. Firstly, because it is necessary to adopt a customer-oriented approach in the development of software applications. Secondly, because it is necessary to reduce the risk that the restrictions imposed on the language and the formalisms adopted will force the user, or even the analyst, to express what the models permit to be represented, rather than the real requirements of the system. The survey shows that, in both Europe and North America, requirements documents are furnished directly by the customer and integrated with interviews in around two-thirds of projects. The main difference between the two regions considered was the percentage of companies that conducted interviews with customers: 73% in North America and 58% in Europe, without significant differences of behaviour between small- and large-sized companies.

With regard to the level of the terminology in requirements documents, one finds that 79% of the latter are couched in natural language (Figure 5). For the correspondence analysis, the final two modalities (structured and formalised language) have been merged.

[31] None of the tools indicated by those choosing the option 'Other' was selected more than twice.
[32] International Data Corporation (IDC) data.
[33] These figures seem to contradict the results of the survey by Glass and Howard [25], where CASE technologies are described as being in decline. However, it should be pointed out that where back-end or 'lower' CASE are concerned, many of the functions offered by these tools are by now part of the development environment. Moreover, other expressions are often used instead of 'CASE': for example, the IDC surveys use OOAMDC (Object-Oriented Analysis, Modelling, Design and Construction) tools. On the other hand, in 1998 the market for OOAMDC grew by more than 10% (24% in Europe), See also the results in [29].
[34] It should be pointed out, however, that the data of our survey are expressed in terms of units of output by the companies surveyed, while the sales figures are calculated on invoices and consequently depend on the prices charged by vendors.

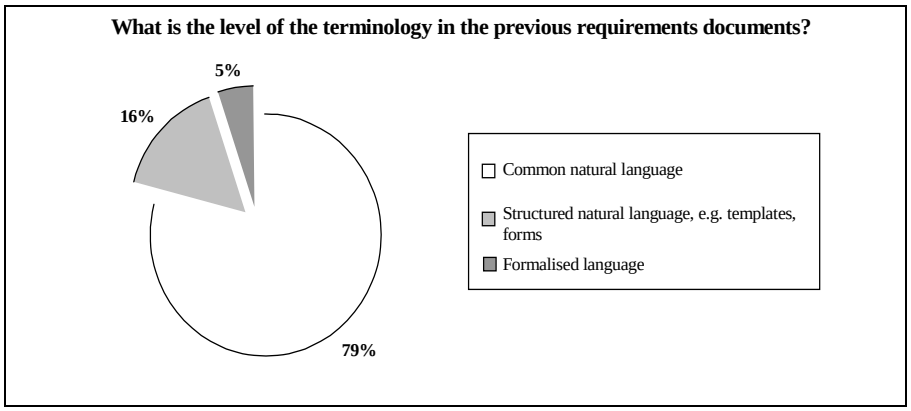


Figure 5 – Level of terminology in the requirements documents

An analysis of the interdependence of the use of natural language with the other factors examined did not show any significant association with type of company, nor with the adoption of a methodology.

Another important aspect concerning both the potential demand for an NLP-based CASE tool in particular and software development in general, is the domain knowledge required for adequate understanding of the problem so that the user's requirements can be defined. In fact, in the presence of high levels of specialist knowledge, the tool must be adapted to the needs of every customer if it is to operate efficiently in different corporate settings. By contrast, a very low level permits the development of a single standard tool able to operate in different fields of application. In this regard, it was found that respondents required an average (54%) to high (34%) level of domain knowledge. It also emerged, that the higher the level of domain knowledge required to develop the software, the greater the use of methodologies (9% for low levels, 53% for average ones, and 38% for high ones) and of tools for requirements analysis and top-level design (2%, 56% and 42% respectively).

[3] As regards the efficiency of production processes, upon conclusion of the market study it was important to determine which software activities were viewed as crucial, as well as their weight relative to requirements (question 16).

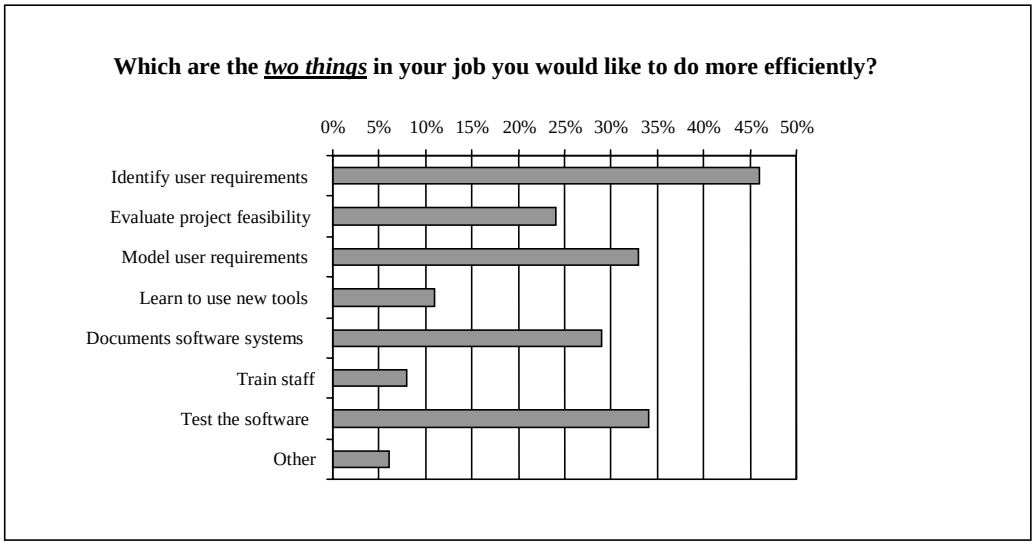


Figure 6 – Activities perceived as crucial in software development

In interpreting the answers to this question, it is worth noting that two selections were requested, thus having results above 100 percent. Figure 6 shows that 'Identify user

requirements' and 'Model user requirements' were cited as priorities by a high percentage of respondents.[35] Unlike in the case of 'Identify user requirements' – which was largely independent of the language used to model requirements (46% for common natural language, 37% for structured natural language and 50% for formalised language) and for 'Testing the software' (35%, 32%, 38% respectively) – for 'Model user requirements' the percentages were 38% for common natural language and 13% for formalised language, in accordance with expectations. Another noteworthy finding is that testing was viewed as crucial by higher percentages (ranging from 19% to 46%) of the respondents who used no tools at all. A similar pattern is displayed by the level of domain knowledge necessary, where at low levels of knowledge, testing was perceived as more important than all the other activities (63%, compared to 32% and 30% for medium to high levels of knowledge). Also of interest is the fact that 'Learn to use a new tool' was selected by a higher percentage of respondents declaring that they did not use a tool for requirements analysis than by those who instead said that they used a tool of this kind.[36]

The importance of this question requires a comparison of the results for Europe and North America (see Figure 7). Also the correspondence analysis - reported in the second part of this section - was done taking into account the centrality of this question with respect to the objectives of the market research, in which the activities considered most critical become determinative when identifying profiles.

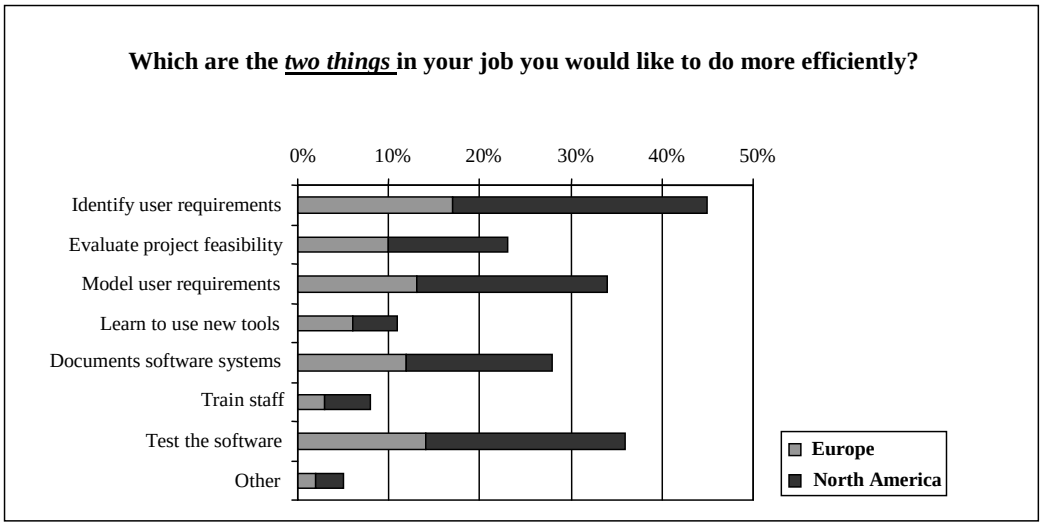


Figure 7 – Activities perceived as crucial in software development (Europe vs. North America)

To the question 'What would be the most useful thing to improve general day-to-day efficiency?', the majority (64%) chose the option 'automation', while 'outsourcing' was selected by 7% and 'internal delegation' by 29%. Contrary to expectations, no particular differences emerged among the replies to this question with respect to company size, where the only significant difference concerned companies with 6 to 20 employees, where the percentage selecting 'internal delegation' was nearly double that for other company groups, a

[35] To be noted is that also around one-third of the final observations concerned the role and importance of requirements. Taking into account of the different goals of the surveys described in [30,31], we can compare these results with those obtained for a question therein on the perceived relative importance of software problems in Europe (most of the software problems are in the area of requirements specification and managing customer requirements; following documentation and testing) and on the perceived scope of a generic process model (defining system requirements, 78%).

[36] In this regard we quote a remark made in one of the questionnaires: "*I hate to be a cynic, but there are hardly any worthwhile tools. The overhead in learning to use them is too great for the payoff*".

difference which may be due to organisational shortcomings. Interestingly, the percentage of respondents who used a methodology or a requirements analysis tool and believed it less important to increase the level of internal delegation was above the average of the entire sample. Instead, there were no differences regarding the documents available for requirements analysis.

Joint analysis of the two questions on the efficiency of software production processes shows that a larger percentage of respondents who believed it important to increase the level of automation had previously selected 'Learn to use a new tool' and 'Model user requirements' (Table 4).

Table 4 – Efficiency of software development processes

	What would be the most useful thing to improve general day-to-day efficiency?		
Which are the two things in your job you would like to do more efficiently?	**Automation**	**Outsourcing**	**Internal delegation**
Identify user requirements	69%	9%	22%
Evaluate project feasibility	44%	12%	44%
Model users requirements	**75%**	4%	21%
Learn to use new tool	**86%**	0%	14%
Documents software systems	71%	5%	24%
Train staff	18%	0%	82%
Test the software	67%	4%	29%
Other	43%	14%	43%

For the final question, regarding the average delay in delivery of the software, the best performances were achieved by companies with 6-20 employees (29% of which delivered with less than one week of delay and 59% with less than one month) and by those who sold directly to the end-consumer (probably for contractual reasons). Though not to a statistically significant extent, companies using formalised language delivered with the least delay, although there were no substantial differences as regards delays of more than one month (26% for common natural language, 33% for structured natural language, 25% for formalised language). A fair interpretation of these results requires one to remember that the answers do not factor in the length of the projects. Nonetheless, assuming that an average delay of less than one week corresponds to companies which on average deliver the software within the designated time, similar findings are reported in [32], where more than 80% of the respondents stated that their projects were sometimes or usually late.

Considering the purpose of this study, and particularly the question of whether there is a market for an NLP-based CASE tool for requirements analysis, the results presented thus far confirm the perception of requirements analysis as crucial for the development of systems, the widespread use of the object-oriented approach and of UML, and the important role of natural language. Specifically:

- More than 80% of the companies adopt a methodology to develop their software, and nearly 68% of them adopt an object-oriented method (UML or one of the methods merged into UML).
- The majority of the documents available for requirements analysis are in natural language and are either furnished by the customer or obtained by means of interviews.
- The domain knowledge required is medium to high.

- Tools supporting requirements analysis and top-level design are used in less than one-third of cases.
- However, identifying and modelling requirements are perceived as being at least as important as testing the software.
- A higher level of automation is indicated by around 64% of the respondents as the most useful means to improve day-to-day efficiency.

All of these elements work together to confirm the existence of a potential demand for a CASE tool based on NLP. To justify this claim, we undertook a correspondence analysis (CA) study. This meant using a statistical technique suited for the study of relationships between modalities with two or more distinguishable variables, usually qualitative. The main steps of correspondence analysis are concisely described as follows:

1) define a cloud of points (rows and columns of a contingency table) in a multidimensional vector space;
2) choose the metric structure on this space;
3) produce the fit of the cloud in 1) to a variable low-dimensional subspace onto which the points (row and column profiles) are projected for display;
4) give an interpretation of the clusters of points corresponding to the projections of the rows and columns of the original contingency table; analyse their absolute contributions as guides to the interpretation of the underlying dimensions and their relative contributions (the so-called squared correlations) to indicate how well the points are described along the considered dimension.

The geometry of CA is very similar to Karl Pearson's [33] geometric description of Principal Components Analysis. The closeness of the points to a line, plane, or in general to a low-dimensional subspace, is defined as the sum of squared distances from the points to the subspace. In general, it is important to avoid the direct comparison of the distances among the projections of row and column profiles because they belong to different low dimensional subspaces and the raw interpretation of their distances may produce misleading conclusions.

Here we have considered a CA involving one of the items of the questionnaire (what should be done more efficiently) as dependent variable and some other collected variables (number of employees, core business, kind of software produced, use of any methodology, starting documentation, level of terminology, use of any tool, knowledge of domain, thing to improve the day-to-day efficiency, average delay in delivering the software) as independent variables in order to verify whether and how much the answer to this item is influenced by the modalities of the other variables and to identify some relevant aggregations of modalities which can reveal the potential market demand for a CASE tool based on NLP.

We present here the result of the application of the CA based on the responses to the question regarding which activities are considered most critical (see Figure 8).[37]

An initial interpretation of the graph can be reached by looking at the axes. Specifically, one can interpret the vertical axis in organisational terms, assuming that the request for more automation rather than internal delegation is due to an already more or less solid organisational structure. The horizontal axis, meanwhile, corresponds to an engineering or to a more informal approach to software development depending on the use or not of methodologies and instruments to support analysis and designing.

[37] The contingency table is available at http://on-line.cs.unitn.it.

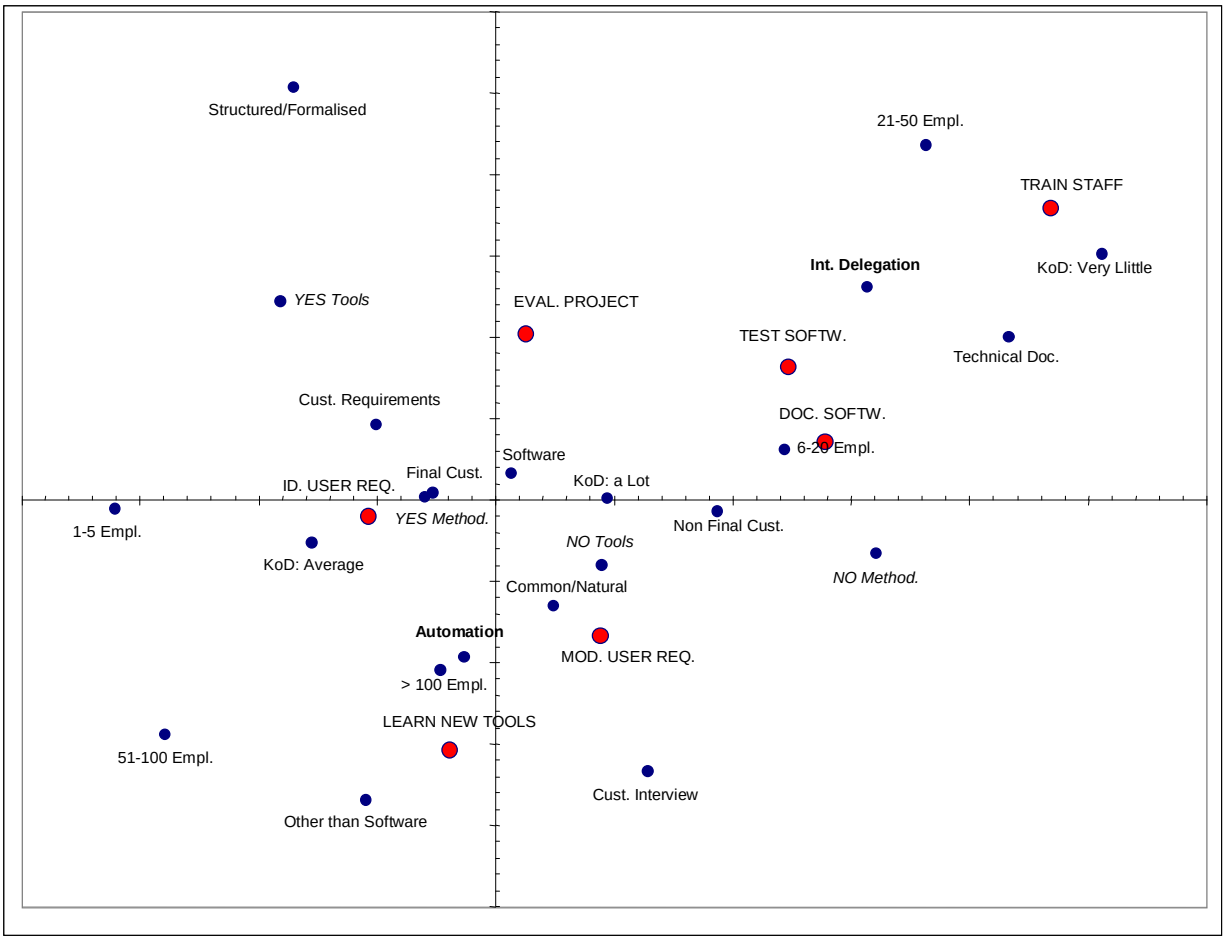


Figure 8 - Output of the correspondence analysis[38]

According to this interpretation of the graph, there are two potential market niches.

The first market niche corresponds to companies that adopt methodologies and instruments to support requirements analysis and top-level design. We can safely assume that they use an 'industrial' rather than 'craft' software development process. For this type of company, project evaluation is considered a critical activity, along with requirements identification. These two activities, among the possible activities listed in the questionnaire, are the most interdisciplinary and at the same time the most difficult to structure. In particular, for purposes of our study, requirements identification can be efficiently supported by tools able to analyse documents in natural language. Moreover, for this type of company, the tool should be specialised to have an appropriate level of domain knowledge for the given area of software development. The client provides requirements documents and the software produced is in turn delivered to the client. For a customer-oriented approach, this means having only a limited possibility to ask the client to write the documents in a restricted form of natural language; however, these companies sometimes receive the documents in a somewhat structured (formalised) form. In these cases it is possible to envision the use of less sophisticated linguistic techniques to analyse requirements documents in order to produce conceptual models using the object-oriented approach.

[38] Two points ('Other' in question 16 regarding critical activities, and 'Outsourcing' for question 17) have not been represented because of their great distance from the centre (low frequency), thereby making the graph more comprehensible.

The second market niche includes medium- or large-sized companies that use neither methodologies nor instruments to support requirements analysis and top-level design. They do, however, perceive requirements modelling as critical, along with other activities such as software documentation and testing, which are already supported in varying ways by existing CASE tools. One can reasonably conclude that also this second group of companies constitutes a market niche for a CASE tool enabled by linguistic instruments. In fact, a CASE of this type could integrate the functionalities of a traditional CASE, favouring the adoption of an engineering approach in software development. Another activity deemed critical is to learn new tools, an obstacle that could be surmounted by adopting a CASE that makes extensive use of natural language. The indication of requirements modelling rather than identification brings to light the fact that a problem at the level of requirements specification can hide deeper problems related to requirements elicitation (these can be supported by speech recognition systems and by all the functionalities envisaged in point *a* of section 2.). This is confirmed to some extent by the fact that identification, rather than modelling, of requirements is considered critical by the companies that adopt a more structured approach to software development.

An important aspect of this research is the broader application of the results. As noted, this research is descriptive, based on a large number of questionnaires (among the highest we have seen in our studies[39]), yet not fully representative of the population. The fact is that for the software industry, there simply is not enough information on the reference population to permit a meaningful and statistically correct extension of the results.

Having said this, we maintain that it is useful to make a comparison with data available in the literature. The following table summarises the most significant of these (Table 5). Worth noting is the scarcity of existing data. Although the surveys to which these results refer[40] are very different, their similarities do stand out.

Table 5 – Comparison with results relative to other surveys and the CASE market

	NLP-based CASE tool online Market Research – 1999 (142 companies)	**State of the practice Survey on RE - 1999[41] (12 companies)**	**SW Development - State-of-the Practice - 1997[42] (78 companies)**	***Market share OO CASE tools – 1998[43]***
Sell to the end-user	**84%**	83%	-	
SW as core business	**82%**	66%	-	
Use OO methods	**68%**	50%	39% (Use O-orientation) 53% (Use a formal life cycle methodology)	
Use UML	**77%**	-	-	*> 48%*
Natural language requirements	**79%**	100%	-	
Use RA tools	**30%** (& top level design tools)	0%	29% (Use front-end CASE tool)	
Use Rational Rose	**52%**	-	-	*33%*
Identify user requirements	**46%**	66%	-	

[39] Notable exceptions are the surveys conducted by the European Software Institute: http://www.esi.es.

[40] These surveys were carried out with different objectives and using different methods and samples. The survey described in [25] used 78 questionnaires compiled mainly by directors or managers of information systems development in companies operating outside the software field, while the Finnish one reports results relative to 12 Finnish companies, 8 of which worked exclusively in the software field.

[41] See [23].

[42] See [25]. Note that when this survey was carried out, UML had only just been adopted as standard by OMG.

We can also cite here some data found in [34], which contains detailed indications of the percentage of pages in natural language or similar forms – text with keywords, hierarchical enumeration and table – for three projects, having values ranging from 82% to 99% (73%, 43.9% and 34.4% respectively, only for natural language text).

Another aspect that enables positive assessment of the outcome of the survey is the low percentage of non-replies (1.65%) and the fact that in the case of replies for which the option 'other' was selected, in 91% of cases a specification was given.

5. Conclusions

As the principal aim of this research project was to assess if there is a market for NLP-enabled CASE tools, the most important finding is that the majority of the documents available for requirements analysis are provided by the customer and couched in 'real' natural language, leading to the conclusion that the use of linguistic techniques and tools may perform a crucial role in providing support for requirements analysis.

Because an engineering approach suggests the use of linguistic tools suited to the language employed in the narrative description of user requirements, we find that in a majority of cases it is necessary to use NLP systems capable of analysing documents in full natural language. If the language used in the documents is controlled (giving a subset of natural language), it is possible to use simpler and therefore less costly linguistic tools, which in some cases are already available. Instruments of this type can also be used to analyse documents in full natural language, even if in this case more analyst consultation is required to reduce the complexity of the language used in input documents or to intervene automatically in the models produced as output. Moreover, needed in many cases, besides an adequate representation of the shared/common knowledge, is specialised knowledge of the domain. Once again, the management of expert knowledge requires more substantial investments to adapt the tool to the company's needs.

As for the potential demand for NLP-based CASE tools, two company profiles have been identified, corresponding to two distinct market niches. The first is composed of companies having an engineering approach to software development and that indicated - of the two activities linked to requirements analysis - the identification of requirements as the more critical. In this case the tool could be configured as a module to integrate with the CASE tool already used by the company, and would provide support for phases where existing tools are insufficient. In the second market niche, the technologies of natural language are used to facilitate the adoption of a CASE tool and more generally of 'best practises' of software development, given that along with requirements modelling, these companies have also indicated as crucial activities in which the contribution of software engineering is well developed (testing or software documentation, for example).

We can also make some preliminary observations here regarding the features expected of a tool based on NLP, proceeding from interviews with systems analysts/engineers and project managers in both small- and medium-sized companies. Specifically, they confirm assumptions made regarding potential demand and interest in the following features:

– The possibility to accelerate the production of analysis models and to rapidly create models to be used in interactions with users and in project groups. The fact that, for example, the class models may contain spurious classes or that some classes may be missing was regarded as less important if the models are produced automatically.

[43] International Data Corporation (IDC) data.

- The tool was also regarded as useful for the training of analysts, with the presentation of texts and the corresponding models, both for junior analysts and for the retraining of those unfamiliar with the object-oriented approach (the latter problem seems to be more important for small-sized companies).
- The possibility of integrating the tool with CASE tools for drawing diagrams using the elements singled out by the algorithm and using tools for documents management.

Finally, for some questions in the survey (e.g., the use of methodologies and E-R models, the use of support tools in the initial phases of development) the contributions this paper makes to the field go beyond the confines of the market research as described by the title. It confirmed some expectations (the diffusion of the object-oriented approach), which on the surface could appear obvious, yet have not been sufficiently supported by hard data. It also confirmed the presence of significant possibilities for the adoption of instruments and methods of software engineering [35].

References

[1] Franch M, Mich L, Osti L. Online Research as Decision Tool for Marketing and Management Strategies. In Proc. Information Technology for Business Management - ITBM2000, 16th IFIP WCC, Beijing, China, 21-25 August 2000, Gan R (ed), Beijing, 2000, pp 737-743.

[2] D'Elia M. On-line Market Research: an Application to the Software Domain. Degree Thesis, University of Trento (In Italian), 2000.

[3] Loucopoulos P, Karakostas V. System Requirements Engineering. McGraw-Hill 1995

[4] Chiocchetti N, Mich L. The Market for Object-Oriented CASE tools. Tech Report, Department of Computer and Management Sciences, University of Trento (In Italian), 31, 2000.

[5] Burg J.F.M. Linguistic Instrument in Requirements Engineering, IOS, Amsterdam, 1997.

[6] Ryan K. The Role of Natural Language in Requirements Engineering. IEEE 1992; 240-242.

[7] Chen PP-S. English Sentence Structure and Entity-Relationships diagrams. Information Sciences, 1983; 29: 127-149.

[8] Ambriola V, Gervasi V. An Environment for cooperative construction of natural-language requirements bases. In Proc.8th ICRE, IEEE Computer Society Press, 1999, pp 124-130.

[9] Juristo N, Moreno AM, Lòpez M. How to use Linguistic Instruments for OO Analysis. IEEE SW, May/June 2000, 80-89.

[10] Fuchs NE, Schwitter R. Attempto Controlled English. In: CLAW'96, 1st Int Workshop on Controlled Language Applications, Katholieke Universiteit, Leuven - Belgium 1996.

[11] Delisle S, Barker K, Biskri I. Object-Oriented Analysis: Getting Help from Robust Computational Linguistic Tools. In G Friedl, HC Mayr (eds) Application of Natural Language to Information Systems, OCG, pp 167-172, 1999.

[12] Mich L, Garigliano R. Ambiguity Measures in Requirements Engineering. In Proc. ICS2000 16th IFIP WCC, Beijing, China, 21-25 August 2000, pp 39-48.

[13] Davis AM. The Harmony in Rechoirments. IEEE Software, March/April 1998, pp 6-8.

[14] Nitto E Di, Fuggetta A. Change vs Consolidation: a Challenge for SW development organisations. Rivista di Informatica, AICA 1995, 25 (4): 267-279.

[15] Mylopoulos J. Information Modeling in the Time of the Revolution, Information Systems, May 1998; 23(3-4): 127-156.

[16] Rugg G, Hooper S. Knowing the unknowable: the Causes and Nature of Changing Requirements. In J Eder, N Maiden, M Missikoff (eds) Proc. 1st Int. Workshop EMRPS'99, Venice, September 25-27, 1999, pp 183-192.
[17] AAA Proceedings Message Understanding Conference. MUC-3, MUC-4, MUC-5, MUC-6, MUC-7 Morgan Kaufmann 1991, 1992, 1993, 1995, 1998, http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html.
[18] Fabbrini F, Fusani M, Gervasi V, Gnesi S, Ruggieri S. Achieving Quality in Natural Language Requirements. In: Proc Int SW Quality Week, S.Francisco CA, May 1998
[19] Laitenberg O, Atkinson C, Schlich M, El Emam K. An Experimental Comparison of Reading Techniques for defect detection in UML design documents. J of S&SW, North Holland-Elsevier 2000; 53: 183-204.
[20] Canzano G. Natural Language Processing in Market Research: Automatic Analysis of Replies to open-ended Questions. Degree Thesis, University of Trento (In Italian), 1999.
[21] Mich L. NL-OOPS: From Natural Language to OO Requirements using the Natural Language Processing System LOLITA. In: J of Natural Language Engineering, Cambridge University Press 1996; 2 (2): 161-187.
[22] Mich L, Garigliano R. The NL-OOPS project: OO modelling using the NLPS LOLITA. In Proc. 4th Int. Conf. NLDB'99, Klagenfurt, June 17-19 1999, Friedl G, Mayr HC (eds), Application of Natural Language to Information Systems, Wien 1999, 215-218.
[23] Nikula U, Sajaniemi J, Kaelviaeinen H. A State-of-the-Practice survey on Requirements Engineering in Small- and Medium-Sized Enterprises. Research Report 1. Lappeenranta University of Technology, 2000.
[24] Zvegintzov N. Frequently Begged Questions and how to answer them. IEEE SW March/April 1998; 93-96.
[25] Glass R, Howard A. Software Development State-of-the-Practice. Managing System Development, June 1998, 7-8.
[26] Comley P. The Use of the Internet as a Data Collection Method. SGA Market Research, 1996.
[27] Wheelwright SC, Makridakis S. Forecasting Methods. John-Wiley & Sons, New York 1985.
[28] Greenacre JM. Theory and Application of Correspondence Analysis. Academic Press, New York 1984.
[29] Dutta S, Lee M, Wassenhove L Van. Software Engineering in Europe: A Study of Best Practices. IEEE SW May-June, 82-90, 1999.
[30] ESI, ESPITI - European User Survey Analysis. European Software Insitute, Spain, Nov 1996.
[31] ESI, System Engineering in Europe. Survey: Summary of Results. European Software Insitute, Spain, Aug 1998.
[32] Genuchten M van. Why is Software Late? An Empirical Study of Reasons for Delay in Software development. IEEE Trans on SWE, June 1991; 17 (6): 582-590.
[33] Pearson K. On Lines and Planes of closest Fit to Systems of Points in Space. Philosophical Magazine, 1901, ser. 6 (2): 559-572.
[34] Melchisedech R. Investigation of Requirements Documents Written In natural Language, Requirements engineering. Springer-Verlag 1998, 3:91-97.
[35] ESI,Software Best Practice Questionnaire, Analysis of Results. European Software Insitute, Spain, Dec 1997.

Figures

Tables

On-line material (http://on-line.cs.unitn.it)

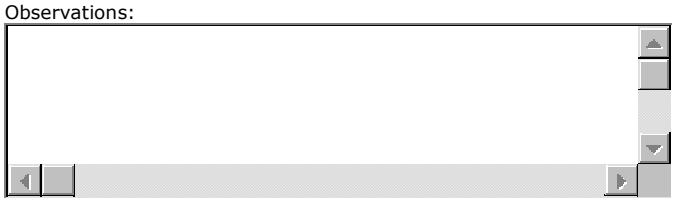
Appendix A

Questionnaire for a new CASE tool

1. How many employees and consultants are there in your company?
   - 1-5
   - 6-20
   - 21-50
   - 51-100
   - more than 100
2. Which is the core business for your company?
   - Software
   - Web-sites (go to question 4)
   - Other: _____ (go to question 4)
3. Which kind of software does your company currently develop?
   - General-purpose software
   - Network software
   - Industrial software
   - Application software for market niches
   - Other: _____
4. Does your company usually sell its products ...
   - to final customer
   - to another software company
   - to software shops
5. What is your current prevalent role in the company?
   - Analyst
   - Designer
   - Programmer
   - System Engineer/Architect
   - Project Manager
   - Other: _____
6. How many years have you been working as computer scientist?
   - Less than 3
   - From 3 to 5
   - Form 6 to 10
   - More than 10
7. Do you use any methodology to develop your software?
   - Yes
   - No (go to question 10)
8. Do you use Entity-Relationship Diagrams to model your data requirements?
   - Yes
   - No
9. Do you use an Object Oriented method?
   - Yes
   - No (go to question 11)
10. Which Object Oriented Method do you use? (max. 2 answers)
    - UML (Unified Modeling Language)
    - OMT
    - Booch
    - OOSE (Jacobson)
    - Other: _____
11. Which document do you start from with in the very first step of system analysis? (max. 2 answers)
    - A requirements document given by the customer
    - One or more interviews to the customer/user
    - A technical document

12. What is the level of the terminology in the previous requirements documents?
    - Common natural language
    - Structured natural language, e.g., templates, forms
    - Formalised language
13. Do you use any tool supporting requirements analysis and top level design?
    - Yes
    - No (go to question 15)
14. Which tool do you use? (max. 2 answers)
    - Rose
    - Stp/UML
    - Paradigm Plus
    - ObjectTeam
    - Other: _____
15. How much knowledge of the domain do you use to develop software applications?
    - Very little
    - Average
    - A lot
16. Which are the two things in your job you would like to do more efficiently?
    - Identify user requirements
    - Evaluate project feasibility
    - Model users requirements
    - Learn to use new tools
    - Documents software systems
    - Train staff
    - Test the software
    - Other: _____
17. What would be the most useful thing to improve general day to day efficiency?
    - Automation
    - Outsourcing
    - Internal delegation
18. What is the average delay in delivering your software systems or products? (behind schedule)
    - Less than one week
    - One month
    - More

Observations:



- Are you interested in receiving the final results of the questionnaire?
- Are you interested to see a demo of the tool?

Personal Information:

First Name _____ Surname _____

E-mail Address _____

Company _____

Address _____

City _____

Country Please Select

Email and country fields are required

Your personal information will be used only for this questionnaire
according to the Italian law 675/96

Submit Reset