

Socratenon and its Application to the Learning of Italian Language

Milicevic, M.¹, Nikolic, N.¹, Milicev, D.¹, Trajkovic, M.¹, Grubic, M.¹, Stoicic, N.¹, Ivkovic, M.¹, Milutinovic, V.¹,
Radojevic, D.², Cekovic, N.², Moderc, S.², Vuco, J.², Raspor, Z.², Stipcevic, N.²,
Desanto, M.³, Salerno, S.³, Marsella, M.³, Cozzolino, A.³, Capuano, N.³, Cirilo, F.³

¹ Department of Computer
Engineering
School of Electrical Engineering
POB 35-54, 11120 Belgrade,
Serbia, Yugoslavia

² Department of Italian Language
and Literature
School of Philology
Studentski Trg 3, 11000
Belgrade, Serbia, Yugoslavia

³ Dipartimento di Ingegneria
dell'Informazione ed Ingegneria
Elettrica
Universita of Salerno
84084 Fisciano (SA) Italy

Socratenon is a Web engine tuned to advanced Web education using the state-of-the-art Internet development technologies and tools. It enables interactive and creative learning/teaching management in four different domains (provider, administrator, teacher, and student). It also includes an interface to an artificial intelligence based tool for off-line improvement or learning curricula (ABITS or Aristotelon). One of the first applications of Socratenon was to help the learning of Italian language for Serbian students (the product was developed through a cooperation between universities in Salerno and Belgrade). This paper presents the basic elements of the Socratenon application and implementation philosophy, and discusses its possibilities in the general language-learning environment. It describes three different experiments and explains the lessons learned. The stress is on the statistical analysis of success of those who used our Web-based product and those who relied on the classical approaches.

1 Introduction

Rapid growth of Internet as a medium and Internet technologies has led to the point when education can be detached from humans and books as the only possessors of knowledge. From the early days, Internet has been exploited in educational institutions for dissemination of research results, and knowledge in general. First shapes were unpolished and required a lot of attention from the users. Such sources of knowledge also included wrestling with several resources of information simultaneously, which actually drove users away from the main subject. Next steps in on-line education were uniform pages of text and images. Then came context sensitive help systems and adaptive hypermedia systems that attempted to tune to user's needs in simplistic fashion. On the other side of the screen, there were common classrooms with teacher facing rows of students. Some students were able to listen carefully to what teacher had learned from

someone else and posed questions. Others were bored by the subject and weren't able to postpone the lecture to a more suitable session. Clearly, both on-line and traditional education approaches have their strengths and weaknesses. Classroom-paradigm is good for collaboration and coordination among students and teachers, but is limited in available knowledge that is hard to upgrade and fails to adapt to each particular student. On-line education is in its childhood, and has a lot to learn to make the most of the huge amounts of on-line information and multimedia, which are also in their pre-adolescence period. This paper will present one doubtlessly good effort to unite advantages of both traditional and cyber educational paradigm.

2 Problem statement

The goal was to design, implement, and study exploitation outcomes of a Web-based training environment. This paper specially covers its application in the field of teaching Italian language. The environment was envisioned to be a user-friendly, easy maintainable intelligent educational environment. First, several design details had to be considered. Primary issues of such a design are data warehousing, distribution, and multifaceted manipulation on one hand, and intelligence incorporation on the other. It was decided to rely on techniques that were broadly applicable, as well as standards and packaged solutions that offer required functionality. For instance, we needed a platform that supports abstract objects and can easily deploy them to a thin-client (any regular Web browser without additional software or hardware needed) in a tangible form. Also, underlying database management should allow easy access to these objects and allow easy functional upgrades

The environment was meant for wider community, and objectives were the improvements in training and learning effectiveness, reduction the training costs, more

sophisticated industrial intellectual capital retention, declined learner training time, and steeper learning

The outcome of the project includes the basic platform, customizable to various users and domains of knowledge, together with well-established methodologies and qualified competencies tested on the pilot-case. Our final purpose is to realize a complete solution (tools, methodologies, and services) that is able to directly perform the training

Pedagogies, and technologies in particular, tend to be evaluated on their ability to adequately and effectively transmit information. A great deal of experiential evidence points, however, to the probability, if not the certainty, that real learning involves something beyond the reception of information (especially the case with language learning). This issue is intended to examine the position that learning takes place in the brain of the learner, and that pedagogies and technologies should be designed and evaluated on the basis of the effect they have on student brains.

3 Existing solutions

On-line training system, concerned with learning in wider sense, exists in numerous flavors. Existing solutions primarily aim at use in education, with some prototypes focused on specific training skills. Majority of universities has put some material for their courses on the Web, making it available for authorized parties or to anyone interested. First and still predominant type of on-line education is static virtual textbook. It is a fixed set of networked web pages, organized as a textbook (sometimes, with some multimedia content) put on-line. Such a way to study at home, at any time, is somewhat similar to using standard books and textbooks. Advantage is that using resources of the Net can make some on-line deepening on the subject. Drawback is that attention may drift away because of something more interesting (and it is easy to find something more interesting than a virtual textbook) in another browser window. Today, this type of on-line education is not regarded as something universally useful.

Other potential end-users of on-line training technologies are industry corporations. In such environments, just-in-time training may significantly drop production costs and improve product quality.

Very soon, the challenge was set to develop Web-based educational applications that can offer some amount of interactivity and adaptability [Brusilovsky97]. Adaptability is important since Web is accessed by a great variety of users that are working from their homes without teachers or colleagues by their side to recognize when their assistance is needed and how should it be

exhibited. Interaction is the key condition to make adaptations possible.

Without strictly categorizing Web-based education systems, we'll describe general techniques that were used, how were they used and their criticism. Adaptive filtering systems help user find relevant drops of knowledge from the knowledge pool. These systems offer some functionality needed for a full-fledged intelligent training system. Adaptive information systems (AIS) personalize information to be presented, according to the knowledge about the user's needs. Having a specific user in mind, access to the right information is performed by using the blend of information labeling, database querying, and some artificial intelligence. However, AIS are oriented on plain information, and use of multimedia and simulation of real-world conditions is hindered.

A more sophisticated approach includes adaptive hypermedia systems (AHS) that are based on applying a set of meta-reasoning rules, utilizing a user model, in order to adapt the content and the links of hypermedia pages to individual user. These systems, just like Intelligent Tutoring Systems (ITS), use knowledge about the domain, the student, and about teaching strategies to provide adaptable individualized learning and tutoring [Brusilovsky97]. Although this seems like a good approach for on-line education, it isn't well suited for general training requirements. Collaboration and cooperation is not embedded into the system, and may be performed outside the system. In addition, we haven't located an AHS that implements needed techniques to the satisfactory degree or plainly lacks robustness.

3.1 Web-based education tools

In fast-paced cyber age, knowledge as a resource is rapidly accumulating and distributing. This calls for creation of a new environment for perception of knowledge, anywhere, anytime, just in time, and most importantly – adjusted to individual user.

These requirements are enabled by technological circumstances – proliferation of Internet technologies, as well as computation and communication capacity.

When compared to accustomed producer-push logic of the delivery of standardized skills, Socratenon proposes a solution for consumer-pull logic of just-in-time retrieval of situation-specific and user-appropriate information.

New learning environment would mean a distributed network of publicly available electronic equipment (e.g., Network Computers, kiosks, etc.). The majority of the environment is transcended in cyberspace with little resemblance to the traditional classrooms with a large teacher's desk facing straight rows of desks. Teacher is no longer the "sage on the stage," but the "guide on the side." They are still present to facilitate group work and to make sure that students ask the right questions and have the resources necessary to answer them. Therefore,

it seems that the fundamental tutor-disciple paradigm is still inevitable. Teachers are not there to tell what formulas will be “on the test.” Formulas may or may not be offered (i.e., hyperlinked), depending on psychopedagogical analysis that is not imposed by Socratenon. Nowadays, in increasing presence of distant training environments, it is even less important to memorize everything. Knowledge is available when needed (just in time) and skills become more approachable. Increased freedom relieves teacher, who now has the time and the position to animate students in collaborative activities, and to answer to questions that were unanticipated during course design. System is meant to prevent teacher from reacting to obvious student questions and uncertainties. Of course, this requires that knowledge base incorporates a kind of FAQ (Frequently Asked Questions) for each course, aside from message board for knowledge exchange that may lead to synergistic results. System monitors student’s behavior, and supplies him/her with material that is most suitable. Thus, teacher can concentrate on his main role, which is not plain delivering of knowledge, but guiding students in unforeseen situations, or plainly, to encourage students. Other benefit resulting from this approach is that a highly qualified person (teacher) can be divided among larger number of students, while still achieving necessary level of competence.

In traditional training environments it is customary for students to pose questions to their tutor, when some uncertainty needs clarification. Socratenon offers means of communication between students and teacher on several levels (from chat and forum to ICQ look-alike; under development is a VRML interface which may heal students with classroom nostalgia). These features would enable synergistic community, which may generate new knowledge, which is the main quality of classical approach.

There are four types of users, inhabiting Socratenon: students, teachers, providers, and administrators. Providers are concerned with putting pieces of knowledge together, by using on-line tools and upload capability (for text objects, raster photos, animations, etc.; Mathematica, Shockwave, RealAudio and similar file types are optional, since they require slim, or “not so thin” client). Since providers contribute content for the courses, they can be also called content providers or knowledge providers since they resemble book publishers. Administrators are concerned with giving access rights to people and deciding on pedagogical and financial policies. Administrator role is needed for Socratenon developers who need to foolproof the system.

Learning process is student-centered, which means that learning material is fitted to student’s needs and ambitions. Socratenon makes use of structured knowledge (rich with metadata), student model (sketch of

student made during student learning sessions by monitoring student’s behavior), and expert model (describing learning strategies). By using these structures, Socratenon is able to perform curriculum customization; decide on how knowledge is sequenced; how the material is presented; when and which hints or help should be offered or available. This means that every student gets individualized course, during which he/she decides when to ask the question and to converge to course mastery. It is like having a well-informed personal Socrates on fingertips or mouse-clicks.

Special attention was given to user interface (UI) so that everything needed for specific learning session occupies rarely more than two screens, avoiding confusion of too many active windows. Hints for the highlighted items are available on a single mouse-click. When help is requested, material is presented in more detail.

It is important to achieve the most comfortable self-education. This is provided by imaginative blend of UI design and interface programming, relying on robust interoperable technologies such as eXtensible Markup Language (XML), eXtensible Stylesheet Language (XSL), Cascading Style Sheets (CSS) and Dynamic HTML, supported by Java programming language.

Web delivery solution has all of the advantages users have come to rely on from a web browser (ease of wide distribution with central update, friendly/familiar user environment), while providing advanced functionality through comprehensive Java development. It uses the [DOM99] to control behavior associated with each element in a document; rendering of the elements is specified using the latest XSL Draft [XSL99]; composing visual presentation with the help of CSS [CSS99].

Stated approaches enable development of innovative environment that would incorporate tangible benefits of both virtual and traditional learning environments, while minimizing the shortcomings of both approaches.

3.2 Classic language learning tools

Traditional language learning tools are courses published in books, on video or audiotapes, as accessory tools, as well as on floppy discs and CD-ROMs.

Contemporary textbooks are based on modern achievements of applied linguistics, on modern psychological learning theories, linguistic theories about language learning, and acquisition applied to modern approaches. Authors of these textbooks, in place of traditional methods, most often mention communicative approach, with the aim of gaining the wanted level of communicative competence. Modern courses are based on theoretical “recipes” made for 16 European languages by experts from the relevant countries, on the bases of initiative and clear theoretical and practical instructions given by linguistic experts from the European Council

(Threshold Level for English, Livello Soglia for Italian, etc.; in 1998; Threshold Level for Russian, based on similar linguistic and sociolinguistic premises was also published, but independently of European Council).

All mentioned tools are of a “closed” type: they represent a complete and a definitely defined system which enables a student to follow a given way in the process of learning and acquiring of a foreign language. Most often the didactic process is based on standardized units scheme. These textbooks are intended for group learning with a teacher, as well as for individual learning. Their scheme is always justifiably rigid, because it introduces new terms by respecting priorities from easier towards more difficult, and from known towards unknown.

4 Essence of the proposed Socratenon approach

The proposed solution has two components, one related to engineering, and the other related to linguistics. Both are discussed next.

4.1 Essence of the technical innovation

One of the demands was to free up the users from the burden of getting familiar with technological prerequisites other than minimum version of Web browser needed to access Socratenon. However, today's Web browsers come packaged with build-in support for most often multimedia standards found today, i.e. GIFs, JPEGs, PNGs, AUs, WAVs, to name the few. In addition, Java applets and JavaScript may be used on client side to improve the overall quality of learning material. Such a set of mostly static graphic and raw audio file standards is typically insufficient for serious education that includes multimedia content, and browsers rely on various helper applications and plug-ins (i.e., RealAudio/RealVideo, Macromedia Shockwave/Flash, QuickTime Movie files, etc.). Current efforts of the main players in Web browser market, supply their products with utilities that automatically download such updates (e.g., Netscape SmartDownload or Microsoft Active Setup). In the following paragraphs conditions and assumptions that enable a low-maintenance system accessible over the Net will be listed:

- **Architecture-related:** Socratenon multi-tier component-based architecture is open to changes on component level. Components are service-based, i.e. have dedicated purpose. It is possible to transfer components throughout middle layer according to load balancing criteria (explained later).
- **System-related:** Expertise of Socratenon may be enhanced with Case-Based Reasoning (CBR) capabilities. CBR module is not integral part of

Socratenon, but can be plugged & played via configurable interface defined with a metadata set.

- **Technology-related:** standards compliance and forward compatibility are issues that greatly reduce future maintenance efforts and need for reengineering. Socratenon exploits implementations of standards' specifications such as [XSL99], [CSS99], [HTML32], ECMAScript ([ECMA_262], which should eliminate differences between JavaScript and JScript), released by the World Wide Web Consortium (W3C). Programmed modules should be kept 100% Pure Java™. Documentation of the project was completed using [UML11], which has already become de facto standard for designing and documenting various software systems. UML diagrams keep system design a lot more understandable and exported Java code more legible, which enables a far greater number of developers and programmers on the same project.
- **Application-related:** We assume that knowledge database has already been provided with application-specific knowledge.

4.2 Essence of the linguistic innovation

The Internet most often presumes individual work, except for selected cases, such as chat, which will be mentioned later, as a new possibility of the Internet-based education process.

The Socratenon training system courses can be used in the traditional way by following the given didactic order. However, in relation to their traditional matrixes, they give a possibility of creating a specifically individualized learning process.

The Socratenon approach to language learning most often presumes an initial possession of a certain level of knowledge of Italian so that it could give the best results. The knowledge of morphological and syntactic categories as well as the knowledge of basic terminology are desirable, but not indispensable. By detecting wrong answers, the program leads to the key, and then the key leads to grammatical explanations related to the mistake (one of the options of the program). It is presumed that the student needs to update his or her knowledge. This way it is possible to create almost automatically an individual didactic curriculum for each student separately. Therefore, the textbook can be used not only horizontally, but also vertically and diagonally.

Consequently, the traditional way loses its characteristics of being closed, and it becomes a tool of an “open” type. The coherence of the scheme of the traditional tool is being destroyed, but we get a tool by students' measure, with prospects of greater success, in shorter period of time, without lost time and repetition.

As to the teaching tools, the program gives a possibility to create groups of students, according to their

level of acquired language skills. The groups are created in the following way: the program detects the same mistakes with different participants subscribed to the Administrator, it connects them and encourages them to do some exercises together. Detection of such tiny details in the situation of traditional language teaching would be much more difficult and much longer, and it would never be so precise.

5 Technical details of interest

Socratenon system will be presented in a two-fold manner, sketching in architectural and design views. This method of outlining Socratenon should give enough details.

5.1 Socratenon Architecture

Initial research was done with a 3-tier Java-based architecture, encompassing client tier, application server tier, and database tier. Idea is to keep the client as thin as possible, without performing any business logic. Java is the chosen language since it offers platform-independence, while saying “write once, test everywhere” should hold. Java Remote Method Invocation (RMI) is used when both client and middle tiers are implemented in Java. HTTP is called to pass the arguments over to the application server tier when processing user’s input. Java DataBase Connectivity (JDBC) is used for communicating with the database tier, while JDBC drivers implement the API for the particular database implementation (Oracle8i was used). Client inhabits the user’s desktop, while the middle and database tiers reside in one or more data centers around the University campus or company information server centers.

We added few architectural enhancements in order to fulfill extra requirements, such as another tier – WebTop server tier (standing between client and application tiers), and reusable services [Gupta98]. WebTop Server tier resides close to the clients, thus reducing communication latency, which is additionally reduced by caching and prefetching mechanisms. Client HTTP requests are processed by Java WebServer (residing on WebTop Server) which decides how the action should be serviced. WebTop Server is also a service locator, determining if service is local or is to be serviced via RMI or CORBA call to remote server. Special features needed by knowledge providers and administrators, i.e. curriculum sequencing design, authoring of knowledge containers, complex user administration, etc., are implemented as stand-alone Java applications. Resulting Socratenon architecture is displayed in Figure 1. In this horizontal representation, number of instances decreases, and computation and communication increases when going to

the right. For example, clients access WebTop Servers on modem speed, while databases serve App. Server queries via T1 links (or optical links for multimedia rich knowledge databases). In alternative Socratenon implementation, databases may reside on separate application servers, when data replication mechanisms are required. Database tier includes three separate databases:

- Knowledge database. Courses defined with metadata, knowledge containers (encompassing textual objects, multimedia objects, applets, and simulations), and curriculum sequencing roadmaps (defines various learning paths).
- User profile database. Complete user model structures, including personal, performance, preferences and portfolio parts.
- Expert model database. Definitions of various teaching strategies

For communication between applets WebTop does the following:

- Creates RMI objects
- Registers objects with the RMI object registry
- Provides security

Clients invoke Java WebServer via URL and Java servlet API is used for managing threads, clients’ requests, and security.

Additional tier solved two problems [Gupta98]:

- Reduced download time. Dynamical downloads at runtime from application server(s), caching applets and static application data closer to the clients.
- Sandbox security model in Java. Prohibiting applets from accessing any local or network resources or from communicating with any machine other than one applets are downloaded from. Signed applets – relaxation of security setting from signed, trusted applets. Problem is that not all browsers support trusted applets yet, and there isn’t substantial infrastructure for their deployment.

* * *

By adding the acoustical component, the Socratenon approach achieves an important new dimension. This component has been already added to the system and enables the following:

- (a) Student can listen to a native speaker pronunciation of all Italian texts,
- (b) By clicking onto an individual words, student obtains the acoustic answer, and

(c) Students can practice the skill of listening comprehension.

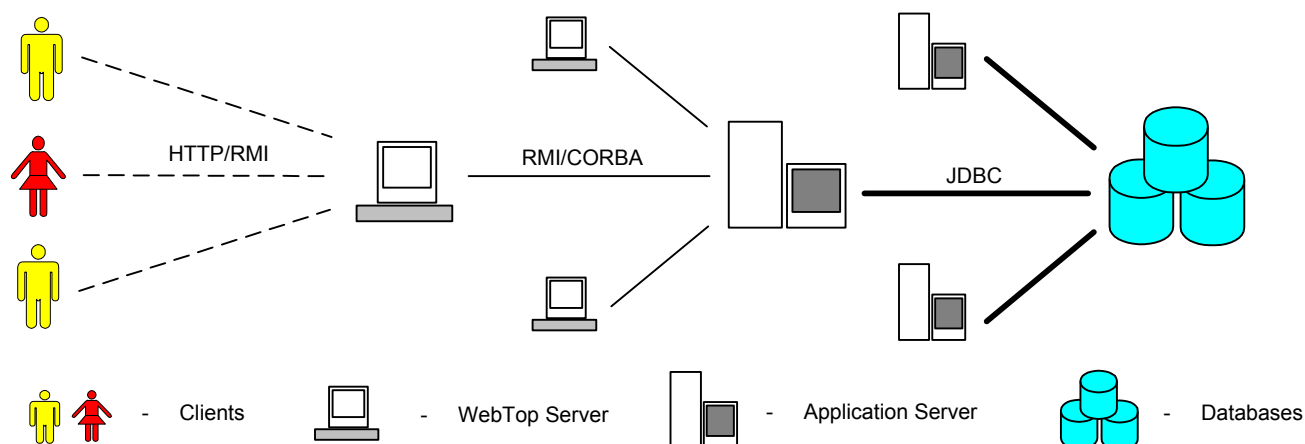


Figure 1 - Socratenon multi-tier architecture

Application tier receives data requests from WebTop server(s) and performs complex data acquisition by issuing sets of database queries.

How can this type of multi-tier architecture be employed in WAN environment for corporate or academic purposes?

After connection with the user is established, Java servlet receives HTTP and RMI requests and determines which service is needed. User ID is passed along from augmented JavaScript code so it is not needed to keep the connections to every user open or to create cookies. For fast WebTop response, user profiles of active users are kept in local database during the session, while performance information exists as an object in JVM memory. This object is semi-static, semi-updated, and remains protected in JVM memory between updates as long as user is alive, i.e. still not dropped out of on-line queue after a time-out.

In more transaction-intense environments two optimization approaches could be considered. Java database applets may be optimized using Java HotSpot™ Performance Engine. For applications that pursue maximum performance, Java may not be good enough and employment of C++ applications should be considered. These applications may also have some intelligence included.

5.2 Socratenon design

With the architecture defined, it was needed to design the programming model. After the components-based programming model (JavaBeans™ components) was chosen, components were designed based on the Socratenon specification, as outlined so far. Further explanation on distinct issues of Socratenon is provided in this section – how the knowledge is represented in the

system, how the courses are adapted to individual students, and more advanced features – i.e. intelligent caching and prefetching, adaptive layout, and load balancing. These mechanisms are implemented in applets that reside on the WebTop server tier, as crucial components of the Intelligent Tutoring System module.

6 The three linguistic experiments

Having in mind that the Socratenon training system has been successfully applied for different purposes in 17 different industrial sites, we thought that this program could be applied successfully also in the field of language learning and acquisition. For this purpose, we have chosen three different Italian language courses.

6.1 The university level course

One of the courses is for the university level, with the possibility to check immediately the effects of its use in the population of the first year students majoring in Italian with the School of Philology of the University of Belgrade [Raspor99].

The course comprises all morphological and syntactic problems of the Italian language, including some lexicological elements planned for that level of knowledge by the curriculum.

The aim of the course is to enable the users to gain adequate linguistic competence with a special stress on enabling them to read and understand, write and speak Italian autonomously, to translate from and into Italian, and to use bilingual dictionaries properly. The course consists of three parts.

The first part of the course consists of 25 didactic units, labeled from 0 to 24. Didactic unit 0 deals with Italian phonemes and their orthographic representations. Each of the didactic units 1 to 24 deals with one of the central problems of Italian morphosyntax. These problems are introduced through Italian and Serbian texts.

The Italian texts are gradually disposed from very simple texts, at the beginning, to the texts taken from contemporary Italian literature (with acoustic simulation).

Because of specific and gradually disposed lexical and morphosyntactic contents, the Serbian texts have not been taken from literary texts, but written by selected authors, in order to respond to the specific needs and possibilities of the student at the described level of knowledge.

Each text is followed by appropriate exercises intended to check reading comprehension and to enable further lexical enrichment and application of the already acquired knowledge. Some didactic units, as an illustration, have texts of popular contemporary Italian songs (also supported by acoustic simulation).

In the second part of the course there are 37 texts which have been used at the written section of the final examinations. They consist of translations in both directions, and they should be treated as a whole. In some cases they complement each other lexically and grammatically.

In the third part of the course there are texts chosen for first readings. These are original Italian texts used to prepare the students for autonomous readings and linguistic analysis of contemporary Italian literature. In this part of the course some contrastive problems, lexical details, etc., have also been pointed at and dealt with. The third part comprises of 10 subparts. Each subpart consists of an original text followed by exercises for recognition, comprehension and acquisition of new lexemes and exercises which enlarge, check and fix the already acquired morphosyntactic knowledge.

All the instructions in the course are given in Italian because the aim language is Italian.

In the second and the third part of the course, where it was possible, the newly introduced lexemes have been explained descriptively in Italian or by using synonyms. Where it was not possible to do this, because of the knowledge level required by the curriculum for the first year students of Italian, the explanations have been given in Serbian. Of course, only the meaning from the given context is covered.

6.1.1 Statistics Questionnaire

The students have been informed that the course could be found on the Internet (namely, [Raspor99] is already used in the teaching process in the standard way – 4 exercise lessons with a teacher and the textbook). During one month, the students had the possibility to use the course autonomously on the Internet. After that, on the

occasion of the written section of the exam, a questionnaire (6.1.2.) has been distributed to the students. Final marks of each student have been decided upon after the oral examination, fifteen days after the written section of the exam. These values have been correlated with the questionnaire data, so that the conclusions have been made 30-45 days after the day when the course started to be used on the Internet.

The major hypothesis underlying the questionnaire are summarized next:

Hypotheses:

1. The Internet is likely to help get a better grade at the final exam, and the question is, what will be the improvement (quantitatively).
2. The Internet is mostly likely to keep attention of the students longer, and especially when they are tired.
3. The Internet is mostly likely to bring better overall effects to younger population.

6.1.2 Linguistic Questionnaire

In order to get concrete values related to the application of the Socratenon system in the field of the Italian language learning, we have prepared a questionnaire with questions that, generally speaking, can be divided into two main groups:

- Ways and possibilities of using the Internet and computer technology (questions 1-4);
- Ways and possibilities of using the Socratenon educational programs for the Italian language learning (questions 5-10).

Each question in the questionnaire is related to one research hypothesis.

Hypotheses:

1. The Internet is mostly used at home or at a friend's or relative's home.
2. The Internet is mostly used for e-mail and for chat.
3. The Internet is mostly used to study the Italian culture.
4. The students know that CD-ROM dictionaries and encyclopedias exist, but they are unreachable to them.
5. The course [Stipèevia95] will be used to the greatest extent (because it gives explanations in Serbian).
6. The most attractive parts of the Socratenon course will be grammatical explanations and topics and subjects in the field of Italian civilization.
7. Closed exercises, connection of terms, and morphological transformations will be considered the most useful parts.
8. Closed exercises, connection of terms, and morphological transformations will be considered the most tiring parts.

9. The students think that the courses should be enriched to a greater extent by authentic material and topics and subjects in the field of Italian civilization.

6.2 The high school experiment

A set of textbooks used in selected elementary and high schools many from Montenegro, but also Serbia [Vuco99] has been ported into the Socratenon system, as well. This helped high school students master the subject using the Internet, but it also helped create statistics to be compared with those of the other experiments. The expectation is that positive effects of using new technology will be much more pronounced in the case of younger population. Details of that experiment are beyond the scope of this paper.

6.3 The continuing education experiment

A popular textbook that enables the general population to learn Italian has been also ported into the Socratenon system. It is the book by Professor Stipcevic [Stipcevic95], the first one and the most famous in Yugoslavia. Experiments related to this textbook will be created after the end of experimentations with the previous two courses.

7 Discussion of essential issues

This section discusses lessons learned and findings generated from two different questionnaires distributed among students. In the first attempt to use the Internet technology for teaching/learning, the response of students was not high enough to create reliable statistical analysis, but their answers definitely indicate a number of important issues, to be discussed next.

For students using Internet to prepare the exam, the percentage of time spent with Socratenon, compared to the total studying time, was maximum 15%. This figure (relatively low) is understandable, because the traditional methods are deeply built into the habits of the typical student population.

The next issue of importance is the distribution of time spent with Socratenon. Students had a tendency to spend either very little time (from 1% to 5%) or a relatively long time (which is about, but not more than, 15%). We expect the total time with Socratenon, as well as the distribution of time spent, to change in the future (we expect future students to be more Internet-oriented).

We were hoping that students that have spent more time with Socratenon will, on average, have better grades. However, we were not able to notice any regularity in that sense. On the other hand, we were able to notice something else, which is probably of a higher importance.

When doing the same exam questions via Socratenon, students were able to achieve a much higher percentage of successful answers. Cases of 80% or 90% were fairly frequent, while the situation with the classical approach (taking the same questions from the classical textbook) was not nearly as good.

Another thing that we were expecting to see was that when a student is tired, it makes easier to follow the material using the Internet technology. That was the case, but not in a way that was dramatic. Percentage of those feeling that Socratenon helped more when they were tired was about 55%. On the other hand, most of the people (over 80%) indicated that learning/studying is more fun when using Socratenon.

Also, most of the students indicated that Socratenon helped save a non-trivial amount of time. This time was in the range going up to 40% (according to their estimate). However, they indicated that a further saving could be achieved if some advanced technical features were incorporated. Analysis of students' suggestions in this domain is the subject of a follow up research.

Students were also asked to tell more about their habits and interests. We feel that technical changes in the Socratenon structure and functionality have to be made compatible with their general habits and interests. Only after such innovations are incorporated, the effectiveness of the Socratenon system will be maximized. Analysis of these issues is the subject of a follow-up paper.

8 Conclusions

The research left to be done and practical implications of the “distant learning” paradigm are still to be evaluated in practice. Revolution in e-education is yet to happen, and our work is one attempt in that direction. There are numerous approaches possible for such application. At least, by the means of diverse utilization of technologies there is a lot space for research. Moreover, social implications that aren't still sufficiently recognized and evaluated, and therefore good evaluation test-bed is also needed. Experiences from the use of Socratenon show that it is one possible approach.

To illustrate the future effort in this research area we'll cite the objectives The International Forum of Educational Technology and Society [IFETS99]:

“[IFETS]... encourages discussions on the issues affecting the educational system developer (including AI) and education communities. While recognising that this brief might be seen as too broad, it is proposed to conduct multiple discussion threads on more specific topics. This approach helps in developing specific aspects concerning the design and implementation of integrated learning environments while sharpening the overall vision about the purpose and processes of education. The discussions are aimed to be definitive and helpful in reaching some

conclusions. The discussions are structured in the form of problem definition, fixed term discussions focusing towards some conclusive end. The conclusions are then put in concrete form for public dissemination.”

Choosing the right blend of good practices in both traditional and existing on-line training environments was a key to success of Socratenon training environment.

Examination of learning and teaching tendencies in both environments has led to creation of a hybrid Web-based training concept, open to future developments and insights in technology, pedagogy and Internet domains.

Making several design decisions was crucial to make the system modular, extensible and above all, useful to both users and future developers of Socratenon. We are looking forward to new creative input from inside and outside Socratenon environment.

Future plans for Socratenon include proving the concept by utilization in a few diverse fields of study. More research & incorporation of adaptable AI techniques are expected to follow. Immersive environments will surely win the battle over the ultimate user interface, so integration of VRML-based interface is also under consideration.

From the linguistic and pedagogic points of view, it has been demonstrated that students hold attention on average longer if high/tech teaching tools are used, plus they score better on exams. On traditional ones they score better and on the Web-based exams even better. Also, effects of high-technology are more pronounced if students are relatively tired after long hours of learning.

9 Note

One can “play” with Socratenon for teaching/learning Italian (for Serbian students, and therefore, some material is in Serbian) at the following URL:

<http://desert.etf.bg.ac.yu:8080/socratenon/>

Over the time, this URL may change. Interested parties are welcome to consult about new updates at vm@etf.bg.ac.yu or julivuco@eunet.yu.

10 References

We have divided references in two paragraphs according to the founding topic: engineering or linguistic.

10.1 Engineering references

[Brusilovsky97] Brusilovsky P. (1997): “Adaptive Educational Systems on The World-Wide-Web: A Review of Available Technologies,” Human-Computer Interaction Institute, Carnegie Mellon University.

[Gupta98] Gupta, A.; Ferris, C.; Wilson, Y.; Venkatasubramanian, K.: “Implementing Java Computing: Sun on Architecture and Applications Deployment,” Internet Computing, Vol. 2, Number 2 (April 1998).

[IBM98] “Enterprise Java Beans & IBM San Francisco - IBM Shareable Framework,” White Paper (<http://www.ibm.com/java/Sanfrancisco/javabean.html>)

[HTML32] “Hypertext Markup Language (HTML) version 3.2” A Web standard. Published by World Wide Web Consortium (1999). (<http://www.w3.org/TR/REC-html32.html>)

[DOM99] “Document Object Model (DOM) Level 1 Specification” Document version 1.0. W3C Recommendation 1 October, 1998 (<http://www.w3.org/TR/REC-DOM-Level-1>)

[XSL99] “Extensible Stylesheet Language (XSL) Specification ” World Wide Web Consortium Working Draft 21 Apr 1999 (<http://www.w3.org/TR/WD-xsl/>)

[CSS99] “Cascading Style Sheets, level 1” World Wide Web Consortium Recommendation (17 Dec 1996, revised 11 Jan 1999). (<http://www.w3.org/TR/REC-CSS1>)

[ECMA_262] “Standard ECMA-262 ECMAScript Language Specification 2nd edition” (June 1998) (<http://www.ecma.ch/stand/ECMA-262.htm>)

[UML11] UML Partners: “Unified Modelling Language v. 1.1,” Object Management Group (August 1997) (<http://www.omg.org/uml/>)

[IFETS99] International Forum of Educational Technology & Society: “Objectives” (<http://ifets.ieee.org/objectives.html>)

10.2 Linguistic references

[Raspor99] Raspor, Z., Vuèò, J. “*Traducioni, Letture, Eserlizi.*” Beograd (1999).

[Stipèevia95] Stipèevia N. :“Osnovi italijanskog jezika” (Bases of the Italian Language), Beograd (1995).

[Vuco99] Vuco, J., Moderc S., “Il mio italiano,” Zavod za školstvo Crne Gore, Podgorica (1995-1111).