E-learning and M-learning: Experiences, a Prototype and First Experimental Results

Anna Trifonova*, Judith Knapp+, Marco Ronchetti*

* University of Trento – Sommarive 14 – 38050 Povo (Trento) – Italy
Tel. +39-0461882033; Fax +39-0461882093; E-mail {Anna.Trifonova, Marco.Ronchetti}@dit.unitn.it
+ European Academy of Bolzano – Drususallee 1 – 39100 Bozen – Italy
Tel. +39-0471055092; Fax +39-0471055089; E-mail Judith.Knapp@eurac.edu

Abstract: The importance of the utilization of modern technologies in education is continuously growing. One of the recently widely discussed fields is the one of mobile learning (or m-learning) where different mobile devices are used for educational purposes. This paper describes a system called Mobile ELDIT (m-ELDIT) that is under development at the University of Trento, Italy. It is a mobile version of an existing language learning platform (ELDIT), developed at the European Academy of Bolzano, Italy. While the ELDIT system can be accessed from any desktop PC having an internet connection, the mobile system can be accessed from PDAs and offers any-time any-where access to the learning base of ELDIT. It means that both online and offline access can be provided to the users. A prototype of m-ELDIT is already being used by some learners as an additional tool for the preparation for the bilingualism exam, but also just for studying the Italian or German languages. The goal of this paper is to share design principles and the practical experience collected from the first experiments with the prototype.

Introduction

The importance of the use of modern technologies in education is continuously growing. Mobile learning (or m-learning) has recently been widely discussed where different mobile devices are used for educational purposes. Our extended review of the research activities in the mobile learning domain in the last few years (Trifonova et al. 03) and the conclusions extracted there motivated our work and influenced our choice of field. The field that we have selected for our experimentations in mobile learning is the one of language learning, as language learning fits well in the boundaries that we placed.

The system that we chose as a starting point is the ELDIT system (http://www.eurca.edu/ELDIT). ELDIT is an innovative electronic language learning system especially designed for the needs of the population of the bilingual region South Tyrol in Italy. The system can be used by anybody interested to study the Italian or German languages, though its mobile version is mainly helpful for preparation for the exams in bilingualism in the mentioned area. This exam must be passed by everybody who wants to work in public administration. The original ELDIT consists of two main data streams – words corpus (learner's dictionary) and texts corpus with comprehension questions. The text corpus has been collected by the "Goethe Institut Milano" (Abel et al. 98). For ELDIT an XML version of the texts has been created. The mobile version of ELDIT concentrates on the second part – the texts. The texts are highly interlinked with corresponding word entries that contain translation, explanations, examples and etc.

This paper will cover details about both the ELDIT and the Mobile ELDIT architectures, namely what the modules are, why they are needed, and how they interact. Details about the practical solutions chosen for the development of m-ELDIT will be also shortly discussed. We show some of our experimental results – interesting observations, positive and negative outcomes. Some points on the future and ongoing work are discussed at the end, followed by conclusions and references.

System Architecture

ELDIT

Figure 1 shows the architecture of the ELDIT system. The basic architecture is a client-server model exploiting the standard Internet and WWW protocols (Gamper et al. 02). The server is implemented using Java Servlet technology

and runs on any Web-Server supporting the Java Servlet API. Java Servlet technology is a powerful and easy to use technology to dynamically create HTML pages. Many useful Java APIs are available for free and can easily be integrated into an existing system. On the client side the system can be accessed by any Web-browser. Currently newer versions of Internet Explorer, Netscape, Mozilla, and Opera are supported.

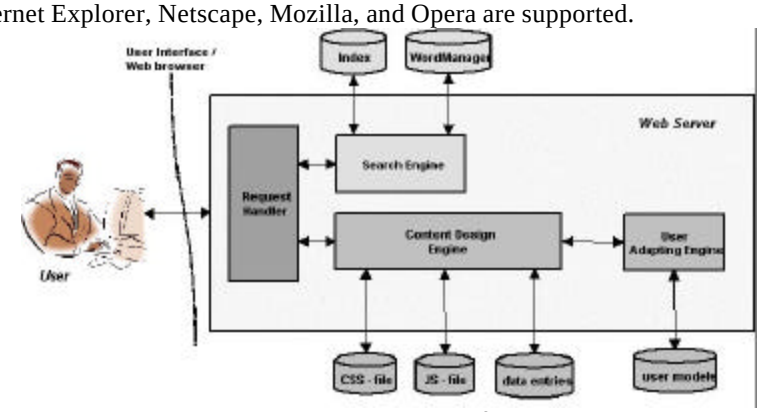

Figure 1: *Architecture of ELDIT*

When a user wants to retrieve a particular dictionary entry, the client sends a request to the server. This request is processed in several steps. First, the Request Handler invokes the Search Engine which is responsible to identify the requested dictionary entry. Inflected word forms are transformed to their base form by modules derived from a morphological database called "Word Manager" (Knapp et al. 03). A fuzzy search algorithm handles spelling mistakes. The data has been indexed and is searched using the Java API Lucene (http://jakarta.apache.org/lucene/docs/index.html).

If the requested word entry is found, the Request Handler invokes the Content Design Engine which is responsible for the compilation of the HTML answer page. We follow a clear separation between content, structure, and presentation: an HTML-page is generated which includes JavaScript arrays containing the data itself as well as a link to an external JavaScript and CSS file. When the browser interprets the answer page, HTML elements such as tables, lists, paragraphs, and layers are generated dynamically by JavaScript functions. The presentation of the information is realized by CSS.

This approach allows great flexibility. The use of JavaScript did not create many problems (what browser differences concerns) since we have always tried to keep things as simple as possible, i.e. from the beginning we restricted the use of JavaScript functions to the basic ones which are understood and interpreted in the same way by almost all graphic-based browsers for desktop PCs.

ELDIT accesses the educational data and user models using DXML and JDOM (http://www.jdom.org/), which are Java packages for handling XML data. XML is a standard for describing and structuring data on the Web. It is a meta-markup language for defining semantic tags that break a document into parts. Such tag-sets are defined via a document type definition (DTD). XML documents can easily be shared and reused among different applications, which is especially important in our case, since we could use the educational content of ELDIT directly for the Mobile ELDIT version.

Mobile ELDIT

For developing the mobile version of ELDIT we applied the principles described in (Trifonova et al. 04), i.e. the three functionalities that are important for a general mobile learning architecture: "Context Discovery", "Specific Adaptation" and "Packaging and Synchronization". We wanted to keep the user experiences during the use of the mobile version as close as possible to the online version, so we used a browser on the mobile device as interface to the Mobile ELDIT. This makes the system very easy for the user to get used to.

Mobile ELDIT consists of two main parts: (1) server side, which we call the m-ELDIT server and (2) client-side, a proxy that serves the cache and collects the log files (see Figure 2). The server is responsible for content redesign, for analyzing the collected information about the user (user modeling) and for predicting the learners' future needs in order to prepare the material that will be used during offline periods (hoarding and pre-fetching).

For the needs regarding content adaptation of the Mobile ELDIT system the only context information that has to be discovered is the device hardware and software limitations. Knowing the screen size, the browser type and the device's browser support for scripts and frames allows the "Content Redesign Engine" module to create the proper 'look' for the Mobile ELDIT pages. As a first step we chose the easiest way to discover the context – through the device browser's HTTP request that is captured on the server site. The HTTP request contains what we need, i.e. what kind of device is used (e.g. Windows CE device), what kind of screen it has (e.g. 240x320), the color resolution, the browser available, etc. In a more advanced version of Mobile ELDIT it will be possible to use other context discovery methods. There are quite a lot of technological solutions nowadays (for example the device independence initiative: www.w3.org/2001/di/). In another scenario the user might receive context-dependant (e.g. location-dependant) language learning material, like for example the system presented in (Jung Li 04). For such scenarios additional equipment and other methods would be necessary, but it is out of the scope of our current work.

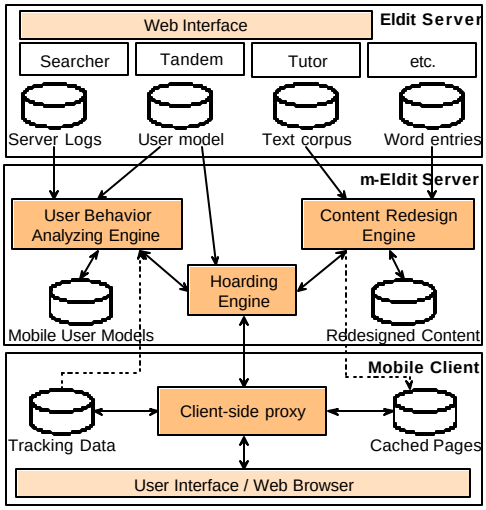

Figure 2: *Architecture of Mobile ELDIT*

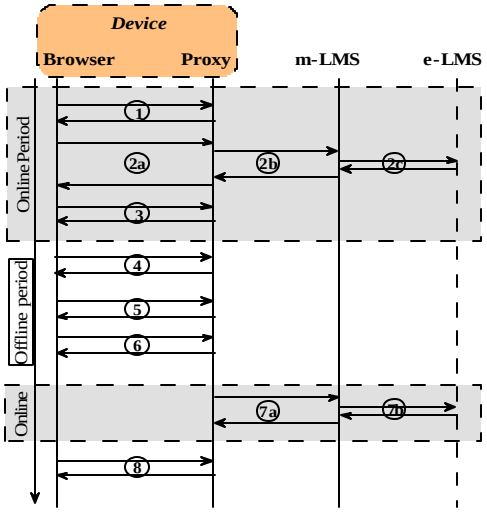

Figure 3: *m-ELDIT transactions*

As said before, in order to keep the Mobile ELDIT users' experiences close to the experiences with the online ELDIT system we chose to use a browser as an interface to the learning material. Most of the browsers on the mobile devices nowadays still do not support frames and have only limited support for script languages. This leads to the need of specific adaptation of the content. The adaptation is also needed because commonly web pages are designed for screen size at least 800x600, hence they are hard to be read and/or navigated from devices with smaller screens. ELDIT does not make an exception. Different adaptation techniques can be used to tackle this problem (Butler 01). The adaptation can be done at server-side, it can be done in a proxy between the server and the client, or it can be done on the client side. Every one of these solutions has its pros and cons.

As mentioned earlier the data of the ELDIT system consists of XML files, both for the texts and for the word entries. For displaying the data on a desktop PC dynamic HTML pages containing frames and JavaScripts are produced on the server site on the fly on every user request. The entries are highly interlinked. For Mobile ELDIT we have decided to use server-side adaptation, namely XSLT transformations of the XML data on a Cocoon server. Our decision was motivated by two facts: (1) on one hand our data was already in XML format, which allows an easy creation of the adaptation rules by using XSLT; (2) the adaptation on the server side is a much better solution in the mobile context, as the adaptation process consumes quite a lot of computational power and does not fit well on a mobile device, as the devices are limited in CPU speed, operational memory and battery.

Figure 4a shows a screenshot of a word entry of the ELDIT system, displayed in a desktop PC browser. One can see that it is made out of three frames. Meaning descriptive information about the selected word is shown in the left-hand frame and additional information in the right-hand frame. The frame on top is dedicated to the searching functionalities of the system.

We have 'converted' the screen on the left (4a) into a series of interconnected screens on the mobile device. When a user wants to see a word entry first the main screen is displayed (4c). Afterwards the user might select to view more detailed information (4b) by clicking the links that were added during the XLST transformations on the server.

For supporting offline use of the learning material and for collecting tracking data a client-side proxy is developed. The proxy is responsible for receiving the browser requests and retrieving the content from the server or from the local store ('cached pages' in Figure 2) when there is no connection available at the moment. The client-side proxy could also seamlessly upload the content that will be used in the future, based on the prediction done in the 'Hoarding Engine'. Uploading might be done on a special user request, where the user might also be given an option of setting different parameters, e.g. provisioned disconnection time, expected duration of time in which the system will be used offline, topics preferred by the user, etc. Different other options could be foreseen, for instance the proxy might be aware of the "cost" of the connection and behave in different ways according to that, i.e. synchronizing the cache when the 'cheap' connection is available (internet through LAN or cradle) and using only the cached content whenever possible on 'expensive' connections.

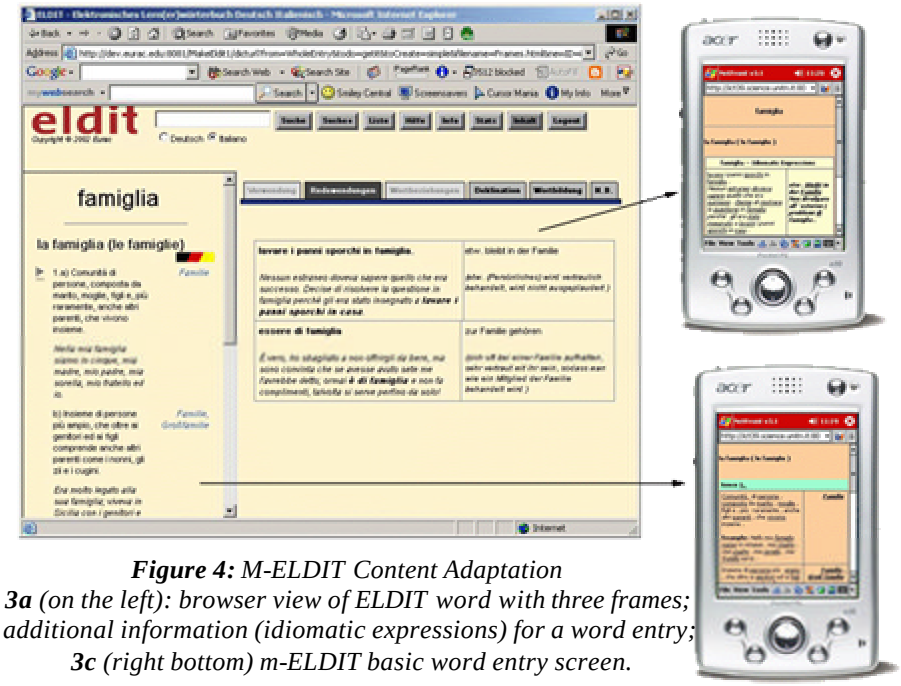

Figure 4: *M-ELDIT Content Adaptation*
3a *(on the left): browser view of ELDIT word with three frames;*
3b *(right top): m-ELDIT additional information (idiomatic expressions) for a word entry;*
3c *(right bottom) m-ELDIT basic word entry screen.*

Another functionality of the proxy is the tracking of the user's activities and on synchronizing the uploading of the log files to the mobile Learning Management System (m-LMS). Thus the user models of the mobile version should be aware of the user's needs and adapt accordingly. The m- LMS should be responsible for calculating and updating the user models, which will differ from the user models in a standard LMS.

Figure 3 shows an exemplary transactional sequence of requests-responses between the mobile device and the two web servers – ELDIT (eLMS) and m-ELDIT (mLMS) during two online periods (in grey) and one offline period. Steps 1 and 3 show the browser request for a page that is captured by the proxy and, after finding what was requested in its own cache, the proxy sends back the response. On the other hand step 2 shows what happens when the requested page is not in the cache – the proxy sends a request to the Mobile ELDIT server (2b), which on its side gets the raw data (2c) from ELDIT (the XML files), redesigns them and sends the response back to the proxy (2a). Step 7 shows that the proxy might decide to contact the server and to update the content of its cache during the online period even when there is no request from the browser. During all offline periods (steps 4-6), even if no cache entry is found, the proxy responds to the requests with a meaningful message.

Results of Experiments

The experimentations with Mobile ELDIT started in June 2004 with three mobile devices - an iPaq H3800 and two Acer n10. All devices are Windows based. Up to now we have observed 9 users.

The ELDIT system consists of about 800 texts – 200 in each language (German, Italian) and level (AB, C). Every text contains about 150 words. Nouns, verbs and adjectives are linked to the word entries. In the mobile version

between 20 and 30 texts are given to the user at a time. On the first access the system offers to the learner texts in both languages and in both difficulty levels, separated thematically into some groups. Afterwards the system presents to the user only the texts he/she is interested in.

Positive Outcomes

- The users found the system very easy to use. Even those that have never used similar devices started using the system almost without problems after a 10 minutes introduction. The users liked the browser interface a lot, as they felt familiar with this way of interaction.
- One thing that almost all users mention is the availability – because of the fact that the device is light and small one can put it in his/her pocket or purse and have it with him/her all the time.
- The users liked a lot the freedom that the mobile device gives them. Some of them often used the system in the train while traveling, others at home or in the office. To the question "*Is there a place where you preferred using the system? Why?*" one user responded "*On the coach. Because it is comfortable ☺*". Another user answered "*None. As I could (and did) use it in any environment.*"
- Two of the persons that used a mobile device as an additional tool for the exam preparation have passed the exam and said that the system has helped them a lot.

Neutral Observations

- Users that were preparing themselves for the exam were (almost always) using additionally a study book and a dictionary. This is due to the fact that in the current version not all word entries of the texts are developed. Moreover these users were also taking paper notes. As we were not giving instructions on how the user could write and save the answers on the device and synchronize it later, most of the users were not doing it, but later they mentioned that this option will be very useful. On the other hand some said that they study better if they take paper notes, so they would do it anyway. Users that were not preparing for the exam generally were not taking paper notes and were less using an additional dictionary.

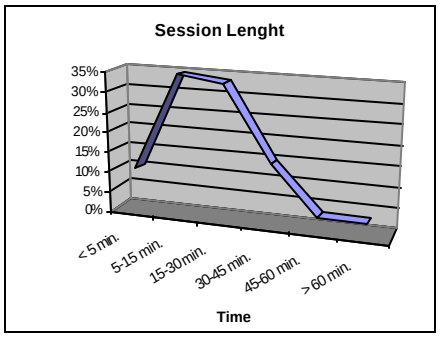

Figure 5: *User session length in Mobile ELDIT*

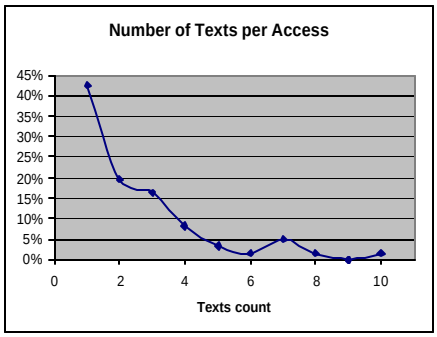

Figure 6: *? of texts read in one m-ELDIT access*

- Generally people were using the system between 10 and 40 minutes (see session length distribution on Fig. 5). Users preparing for the exam were spending longer periods on a text (30-40min) and were picking up more word entries, since they were trying to answer the comprehension questions. This means that on average one text was read in one session. Others that want to improve their language skills might read certain texts in 3-5 minutes, thus reading more texts in a session. Sometimes (rather rarely) a user was reading a large number of texts (see Figure 6) for a longer period (> 1h.)
- Another difference was observed between the behavior of users that were preparing for the exam and the ones that were just studying the target language. As mentioned the texts are both in German and in Italian. For the bilingual exam the student should read a text in one language and answer to questions in the other language. This is done for both languages. Texts have two difficulty levels (AB and C). Users, who were preparing for the exam, were concentrated on one difficulty level, namely the level for which they wanted to conduct the exam (see Figure 7). Typically they read texts in both languages, concentrating slightly more on the texts in their native language, since in this case it is harder to compose correct answers. Alternatively users that were not preparing for the exam concentrate on texts in the target language (e.g. Italian for German speaking user) and this without considering the difficulty (see Figure 8). Only driven by curiosity they also browse texts in their native language.

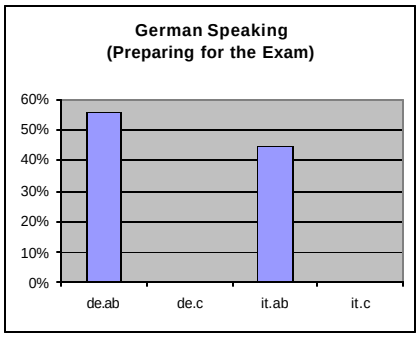

Figure 7: *Typical pattern for a user preparing for the bilingualism exam*

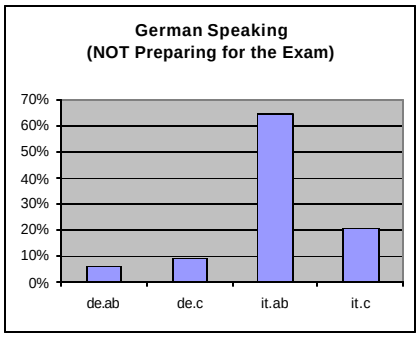

Figure 8: *Typical pattern for a user not preparing for the bilingualism exam*

- The final goal of our experimentation is to support the automatic selection of learning material for offline periods. This process called "hoarding" is described in more details in (Trifonova et al. 05). For this we need to be able to predict what material the user will need in his/her next learning session. Our observations show that during the first use the learner is exploring what is possible to be done in the system, so it is quite unpredictable. For example it is very likely that the user will click on a word that he/she is familiar with just to see what information is available. This leads to the impossibility to exclude even easy (basic) word entries (i.e. essere, *English: to be*). Later on, the users start really studying and do not click on words that they do not need. In our first prototype version we asked the user to grade the importance of missing entries, but later we saw that this is quite useless, as (after the first access) every word that is requested is important for the study process.

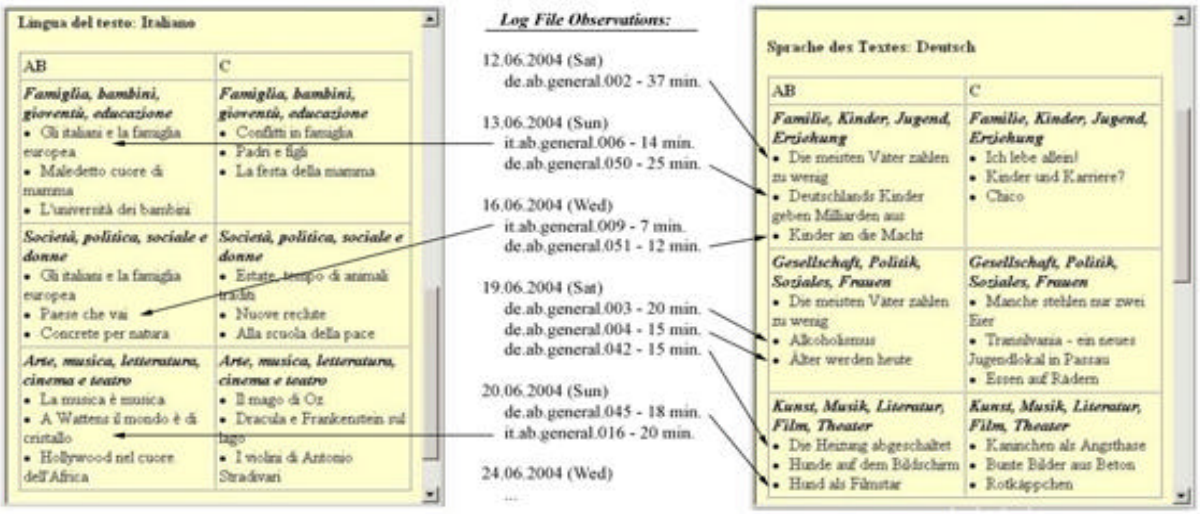

Figure 9: *Example of consecutive browsing behavior of a user*

- Again in the context of hoarding we found out that verbs are a critical point for pruning (excluding). We considered a word to be known to the user if he/she had the possibility to look at its entry from a text, but decided not to revise it. Sometimes we considered a verb to be known to the learner from even several consecutive texts, but later on the same verb could be requested again. This happens because the verbs are linked to their entry based on the infinitive form. So the user might be unfamiliar with a new conjugation, which is more rare or difficult.
- A very interesting observation is that most of the users show a strict consecutive browsing behavior (Figure 9).
- Some users reported changes in their own behavior short time before the exam date. For example a user at the beginning was using the system mainly during the week-ends and more often in the mornings (also this can be seen from the log files). He was reading texts in the language that is more difficult for him (Italian), as the goal was to learn new words. As the exam date was drawing nearer (the last 2 weeks before the exam) the user was using the system much more often – almost every day after work (workdays' evenings) and was reading mainly texts in the mother tongue (German), trying to answer the questions in the target language (Italian).

Problems Found

The problems we found up to now are of a formal nature and are not directly related to our research work. Therefore the positive outcomes reported in the beginning of this section are very important for us and encourage us in our approach. Nevertheless we find it important to report them as they would be common for all research in m-learning domain.

- The battery of Windows based devices discharges quite fast. When a device is frequently used it discharges in 1-2 days, but the main problem comes from the fact that even when not used the battery discharges in about a week time. The discharged device "forgets" the software installed by the user and all user's data. This leads to the necessity to do backups of all important data on an external memory quite often. It is also very inconvenient as all the programs that were installed should be re-installed.
- As our experiment was connected with offline delivery of material we introduced a client-side proxy that should simulate internet access even in offline periods. The problem that appeared was that Internet Explorer (the browser available on windows based mobile devices) does not send requests to the local proxy if it does not find an internet connection by itself. This made it necessary to use another web browser. Unfortunately all other browsers are commercial products.
- Problems were found also considering the presentation in the browser. Special German and Italian letters are not always presented correctly, thus the browser should be chosen carefully.
- The first version of our prototype works with a big number of small files (several thousands). We have observed that the file transfer from the desktop PC to the mobile device is a very slow operation, when lots of files are being copied (e.g. about 5-10 times slower for transferring small files, comparing with transferring one big file). This means that packaging is strongly needed. The process of deleting small files is also very slow.

Further outcomes

In August 2004 an online questionnaire has been made available in the Web-based ELDIT version. About 90 persons have completed it up to October 2004. Some of the outcomes are useful for the development of the mobile version and will therefore be listed here.

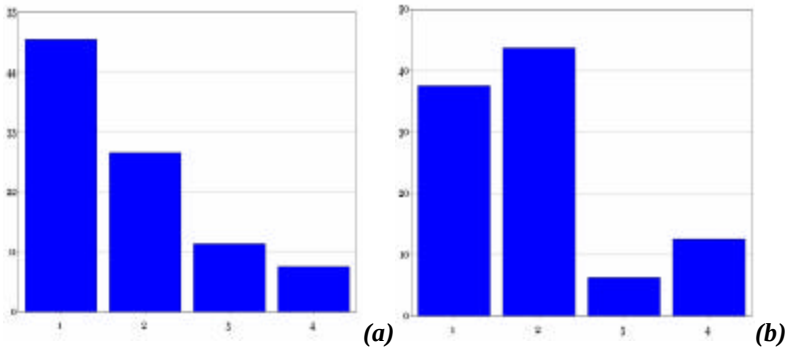

Figure 10: *Do you find ELDIT useful for preparing the bilingual exam (a) level AB and (b) level C?*
[1 – very useful; 2 – quite useful; 3 – not very useful; 4 – useless]

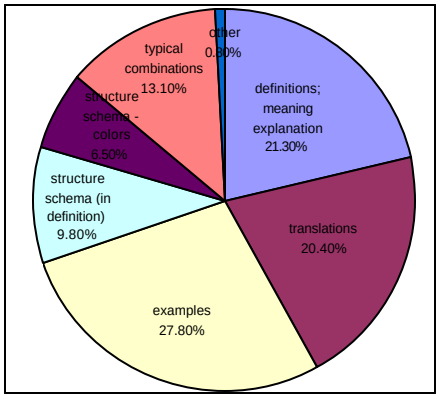

Figure 11: *What do you consider mainly for understanding a word meaning?*

On figure 10 one can see that users that had used ELDIT during their preparation for the bilingualism exams (that are more than 50% of all users – see Figure 12) find the system very or quite useful.

Figure 11 shows that the main consideration in understanding an unknown word falls almost equally on definitions, examples and translations. In the mobile ELDIT they are presented on the first screen for word entry (see Figure 4c).

Figures 13 and 14 show the time usage of ELDIT. Our goal is later to compare the usage of the online desktop and the mobile offline systems, having in mind that the mobile ELDIT complements the main system.

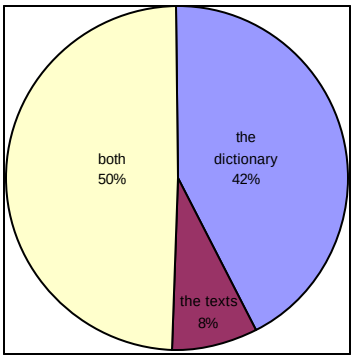

Figure 12: *Do you use the texts or the dictionary of ELDIT?*

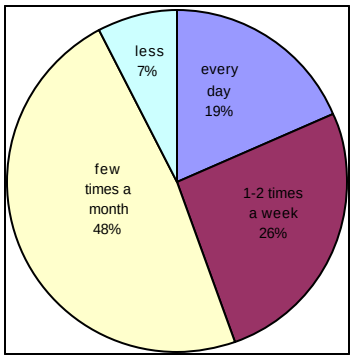

Figure 13: *How often do you use ELDIT?*

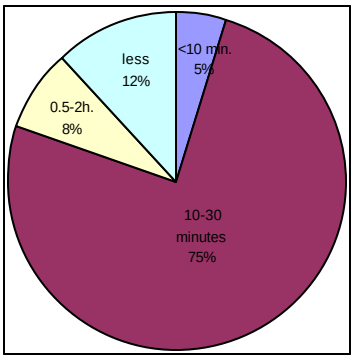

Figure 14: *For how long do you use ELDIT?*

Future Work/Ongoing work

The current evaluation has been conducted in order to collect data about users' behavior in the mobile version of ELDIT. Each user has been given a limited set of texts, recorded on the device. Whenever the user has finished working with these texts, a new set is copied onto the device manually. Beyond suggestions about how to improve interface and interaction, mostly behavioral data has been collected in this way to develop a hoarding algorithm that should allow in the future selecting this limited set of texts automatically whenever a user goes offline (Trifonova et al. 05). An important issue is the packaging and archiving of the data that is being experimented in the next version of the mobile language learning system.

Conclusions

The preliminary conclusions of the experiments are that language learning is a good choice as a field of the use of mobile devices. As people have different learning styles some of them use the system to study in small gaps of waiting time, while others prefer using the PDA just as an electronic dictionary available anytime. Our first results include some interesting findings on user behavior during system exploration, like the distribution of the session length in mobile system usage, consecutive browsing behavior, differences in students' behavior depending on the user goal. Some problems with the usage of mobile devices were also reported and will be considered in the future.

References

Abel, A., Cavagnoli, S., & Putzer, O. (1998). *Zweisprachigkeit - Bilinguismo. Deutsche/Italienische Texte - Testi tedeschi/italiani*. Arcadia Edition.

Butler, M. H. (2001). Current Technologies for Device Independence, *HP Labs Tech. Report HPL-2001-83*.

Gamper J., Knapp J. (2002). XML for an electronic learners' dictionary. In Pedro Isaias, editor, *Proceedings of the IADIS International WWW/Internet 2002 Conference*, p. 427–434, Lisbon, Portugal, 2002. IADIS Press.

Jung Li, (2004). Context-Aware Support for Computer-Supported Ubiquitous Learning. *In proc. of the 2nd IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE'04)*

Knapp J., ten Hacken P., Pedrazzini S. (2003). ELDIT and Word Manager - a powerful partnership. In *Proceedings of World Conference on Educational Multimedia, Hypermedia & Telecommunications (ED-MEDIA 2003*).

Trifonova A., Ronchetti M. (2003). Where is Mobile Learning Going?. *In Proc. The World Conference on E-learning in Corporate, Government, Healthcare, & Higher Education (E-Learn 2003)*, Phoenix, Arizona, USA, November 7-11, 2003.

Trifonova A., Ronchetti M. (2004). A General Architecture to Support Mobility in Learning. In *proc. of the 4th IEEE International Conference on Advanced Learning Technologies (ICALT 2004)*, August 30 - September 1, 2004, Joensuu, Finland.

Trifonova A., Ronchetti M. (2005). Hoarding Content for M-Learning Context. In *Proc. of Workshop on Pervasive eLearning (PerEL 05), in conjunction with 3rd IEEE Int. Conf. on Pervasive Computing and Communications*, Kauai Island, Hawaii