Mobile Support for Collaborative Work

Luis A. Guerrero[1], José A. Pino[1], César A. Collazos[2],
Andres Inostroza[1], and Sergio F. Ochoa[1]

[1] Department of Computer Science
Universidad de Chile
Blanco Encalada 2120, Santiago, Chile
{luguerre,jpino,neinostr,sochoa}@dcc.uchile.cl
[2] Department of Systems
Universidad del Cauca
FIET-Sector Tulcan, Popayán, Colombia
{ccollazo}@unicauca.edu.co

Abstract. An attempt is made to characterize situations in which the use of mobile devices can be useful for the development of collaborative systems. Mobile devices have advantages, such as small size, low cost, portability. They also have disadvantages, such as small viewing screen, little storage capacity, slow processor, unreliable communication facilities. The idea is to use them when advantages are most relevant and disadvantages do not affect the system under development. A collaborative system for text co-authoring is presented as an example of design for the best conditions of mobile devices inclusion. This system uses the mobile devices for individual tasks performed while away from normal work site in uncomfortable or congested places.

1 Introduction

Many people need to move themselves to accomplish their jobs. For them, hand-held computer devices have convenient small size and there are simple but useful applications already running on these gadgets, such as telephone directories, to-do lists, and calendars.

Hand-helds have recently improved the portability introduced by notebooks some years earlier. The newest models include relatively larger memories than the first ones, better graphic resolution and wireless communication. Therefore, many system developers increasingly look at hand-helds as tentative devices to build with them new applications which might take advantage of their features. In particular, some developers may be interested on designing systems to support people doing cooperative tasks. However, a few relevant questions should be asked: are hand-helds appropriate components of collaborative applications? If they are, which are the tasks they support in the best way?

Of course, it is not obvious hand-helds – also called Personal Data Assistants, or PDAs – may be adequate for ambitious systems involving several people with many complex interactions among them. One of the main advantages of handheld computers is that they are portable. Also, initial vendor-supplied software encouraged individual rather than collaborative use. Even now, most handheld applications often reinforce the idea of a handheld computer as a *personal* digital assistant. Neverthe-

less, recent articles describe group or collaborative applications based on this kind of devices [20, 26].

Developing collaborative applications including traditional computers is already a hard task [13]. Trying to design useful collaborative applications using PDAs present further challenges, since these devices have several difficulties for group work when compared to normal computers. These restrictions include small screens for visualization and interaction, limited input facilities, short-life batteries and slow processors.

On the other hand, when we observe collaborative work we may notice there are some tasks or activities which are performed individually in many cases. We may hypothesize PDAs as potentially useful devices for these individual tasks within the group context. This hypothesis may be reinforced if these tasks are to be done by people with high mobility.

The purpose of this paper is to attempt to characterize collaborative situations in which PDAs could be used with advantage. It will also present a case in which such conditions are held for one of the most traditional collaborative applications: text co-authoring.

2 Cases Reported in the Literature

The mobile computing concept is quite new and has no clear definition, although some studies have already tried to survey this fast-growing area of information technology. Mobile computing does not only involve mobile computing devices (notebooks, cellular phones, PDAs and wearable computers), which are designed to be carried around, but also the mobile networks to which these computers are connected. Mobile services are the third component, rounding out this definition of mobile computing.

In this sense, mobile computing has been discussed in just a few recent papers from certain typical points of view. Wireless network service problems in so-called wireless personal area networks (PAN) are a fundamental issue [38]. In his paper, Zimmerman examines wireless technologies appropriate for PANs, and reviews promising research in resource discovery and service utilization, including data formats. Zimmerman already emphasizes the role of Extensible Markup Language (XML) as a standard for structured document interchange. Some interesting further problems of mobile network services are discussed by Chalmers [4].

We agree with Zimmerman in emphasizing the evolution of PDAs as the engine driving mobile computing. Pocket-size, low-price units with long-lasting power autonomy and broad functionality guarantee the critical mass of buyers necessary to motivate the industry to produce and further develop this type of devices. (The growth of mobile phone use is a good example of such a stimulant loop). Zimmerman's list begins with the legendary but rather clumsy Newton, introduced by Apple Computer in 1993. The Nokia 9001 communicator, introduced in 1998, was another milestone, representing the first successful fusion of a handheld PC with a mobile phone. Unfortunately, it ran on a non-standard operating system, and it was therefore short of software. The Sagem Pocket PC, introduced by Sagem and Microsoft in 2000, was a hybrid of the same kind, but with a standard operating system: Windows CE.

The opposite of integrating functions in a single device is device modularization, which has some additional positive effects. A small mobile phone with data transmission capability can be connected to a PDA or another more specialized device, thus

offering them networking capabilities. Bluetooth wireless connectivity [2] simplifies and automates the linking of devices in a very convenient way, thereby supporting the concept of modularity. This technology enables users to connect to a wide range of computing and telecommunications devices without the need to carry, buy or connect cables. The new generation of mobile phones, notebooks and PDAs already comes with Bluetooth. However, integration versus modularization is an old dilemma, one which is not very significant in the present case.

A paper by Held and Ziegert describes some of the characteristics of mobile computing, presents a system model, and shows in more detail how one of the features of mobile computing – service mobility – can be realized [15]. Some earlier papers addressed the specific problems of mobile computing, like designing mobile computing systems using distributed objects [6], but the most relevant research reports can be found only in recent publications. Interesting topics include: mobile client-server computing, including mobile-aware adaptation, an extended client-server model, and mobile data access [18], interaction with the World Wide Web via wireless-connected PDAs, problems with bandwidth limitations, screen real-estate shortage, battery capacity, and the time costs of a pen-based search keyword input [3], and client/agent/server adaptation of a framework to enable performance improvement, which is very important on slow or expensive networks [29]. Other subjects include context-aware collaboration [24], and data management, including data dissemination over limited bandwidth channels, location-dependent data querying, and advanced interfaces for mobile computers [1].

A field which could much benefit from mobile devices is electronic commerce (M-commerce). Rao and Minakis examine location-based services and their importance in M-commerce, taking advantage of mobility rather than repackaging old applications in a new format [32]. While old methods can be adapted and retooled to create applications and explain M-commerce successes and failures, new methods, tools, and ways of thinking must be developed and refined to take advantage of mobility and its potential. The same could be said about other PDA application fields.

Although the number of research papers addressing mobile computing is modest, there is no doubt that much research is going on, perhaps even too fast for papers to be published. As one technology overtakes another [17] it is probably wiser to concentrate on more general concepts and problems. One such problem is the adaptation of existing information systems suitable for efficient integration with mobile computing. In this regard, we agree with Vizard that “until then, mobile computing will just remain a troublesome niche application for those who can afford to pay for it” [37].

We believe it is important to gain an understanding of how collaborative processes can be structured using handheld computers. Handheld computers have the potential to impact both the individual and social processes and we need to understand how to best design the inclusion of these devices to support these activities.

Sarker and Wells describe a framework for mobile device usage and acceptance identifying the relationships between the variables motivating people not only to purchase devices but actually to use them for M-commerce [33]. Their model presents an integrated framework for use and adoption of PDAs according to an IPO model. IPO (Input-Process-Output) consists of Inputs (User characteristics, technology characteristics, modality of mobility, and the surrounding context); Process (exploration and experimentation), and Output (outcome of the process). Our strategies have similar goals, but they rather try to characterize favorable contexts for the use of PDAs.

3 A Favorable Case

We consider next a case in which a group of scientific researchers is trying to write a joint technical paper. Scientific papers [35] are an important --although poorly understood--method of publication. A scientific paper is written for the scientific community at large. The contents may be, e.g., a survey, a tutorial, present a theoretical model or discuss new experimental results. The typical paper needs to introduce the subject, place any results in context with other scientific works, and suggest future possibilities for research.

Some roles need to be specified, e.g., scribe, reviewer, coordinator. Also, some goal achievement strategies and social protocols are required. Let us assume not all co-authors are co-located at all times. This may occur because some of the co-authors work in various places. For instance, one author may wish to do part of his work while being at an airport lounge waiting for a plane, or in the plane itself. Another co-author may need to work in a rather crowded underground train while returning home at evening.

Let us also assume the group has decided to divide the initial writing task among some of the co-authors. Each of these writers then has to produce a draft of a part of the article. The writers will need meetings for information sharing and synchronization after they finish their divergent tasks. During or after these meetings, it is expected co-authors will do material reviewing and re-writing and also some planning.

This set-up seems appropriate to incorporate handheld computers into the solution: perhaps the co-author working in an airport or plane may use a laptop computer, but clearly the one working in a crowded underground train needs something smaller and more portable than a laptop. However, the restrictions stated in the previous sections make us aware of the limitations of PDAs. Their small screen and limited viewing angle makes it difficult for a group of persons to collaborate around a shared display when co-located. Also, the handheld input clumsiness is relevant in this case. Current handheld technology introduces other problems, such as the involvement of users in the transfer of information, and the fact communication is primarily peer-to-peer. Thus, users must switch their focus from the productive task to the act of transferring the information [36]. In conclusion, even when we guess PDAs may be useful, we must carefully design their introduction into the solution. Next section describes the strategies we have designed and implemented to take advantage of this technology.

4 A Design Strategy to Support Collaboration with PDAs

Designing work with PDAs means to take into account both opportunities and restrictions. We have already mentioned restrictions due to the reduced screen size, available memory, input devices and processor speed. We should now add small communications bandwidth and eventual interruptions in information interchange as a consequence of wireless communication intermittence [19]. Therefore, it seems reasonable to consider the following design aspects as influenced by these restrictions:

- Design simple user interfaces, including few elements.
- Consider little memory and storage. Most data stored in the PDA is volatile.
- Consider alternative data input devices. PDAs are slow and somehow unfit for large amounts of data to be input.

On the other hand, opportunities provided by handhelds include: great portability, short start-up and response times and ability for gathering and presenting small pieces of information. Thus, the following design characteristics are suggested by these opportunities:

- PDAs may be useful when user mobility is a main consideration during the operation of the system.
- Use PDAs when individual work is required in unusual, congested or uncomfortable places.
- Consider the use of PDAs when users can make timely short annotations which can be expanded into larger contributions afterwards.
- Consider the use of PDAs during periods of individual divergent work. Each of these periods will probably be followed by a period of group convergent work, in which the previous achievements are appropriately merged and improved.

The role of the individual in collaborative work should not be overlooked, as many activities within a complex undertaking may be better done by individuals than by groups. An interesting study exploring the opposing individual vs. collective views in the foundations of group work in the context of an organization has been done by DeSanctis [9].

5 A Solution to the Case

Our solution to the case presented in Sect. 3 considers to support co-authors in order to work in the most convenient place and time. It also distinguishes work which may be done individually from the work which needs synchronous joint participation. Furthermore, we make another design decision by specifying that the synchronous joint participation be a face-to-face meeting. Individual work output does not have to wait for a meeting to be shared: it may be communicated as individual work proceeds, as explained below.

The technology needed to support this solution includes workstations or notebooks to do normal text and multimedia authoring and editing, and PDAs for some of the individual text creation and edition. It is assumed a wireless local network and Internet connection are available. All of this hardware is used by MoSCoW (Mobile Support for Collaborative Writing), our software composed of four modules: a user management component, a Web editing module, a PDA text editing module, and a communication and synchronization component.

Work supported by PDAs may be done in two ways: network-connected and off-line. When working network-connected, the user works in a way resembling workstation use, i.e., document synchronization is automatic. Off-line PDA work occurs when the co-author steps outside the range of the wireless network. In this latter case, the PDA stores a “copy” of the original document; the co-author then does all text editing as desired. Of course, after off-line work, the performed changes must be synchronized with the master document. When this is done, the master document is stored as a new version. In turn, this means all stored versions must be merged (synchronized) at some time. A coordinator must do this merging, and usually this involves discussion with the other co-authors in order to keep document coherence.

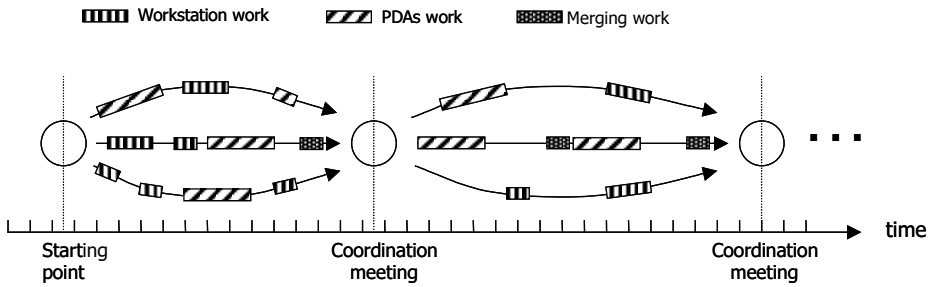

Fig. 1. A sample activities sequence

Figure 1 shows a diagram of a collaborative text-editing job supported by PDAs. Represented activities are editing from a workstation or notebook, editing from PDAs, and merging processes. Large nodes represent coordination meetings. The three depicted co-authors may do divergent work in the most convenient way according to their needs. We may guess that perhaps many of the contributions generated from PDAs are annotations or brief statements which are developed in full when working from workstations afterwards. In Figure 1, a single co-author (coordinator role) is responsible for merging the various existing versions. This latter type of task must be done with the other co-authors being aware and agreeing.

It should be noted the merging process is only needed to incorporate changes made from off-line PDAs because the other ways of work (from workstations or network-connected PDAs) consider an instantly updated shared document. The shared document includes a locking mechanism: only one user can be updating it at the same time. When one user is updating the shared document, the others have reading access. Although the locking mechanism may not be desirable when several users want to update at the same time, the system is actually intended for mostly asynchronous work, and thus, update rights will seldom be denied. Annotations, however, may be done concurrently with one user doing updates on the master document (annotations are actually made on a copy of the master document). Annotations are visible by all group members. Figure 2 shows a diagram of concurrent use of the system.

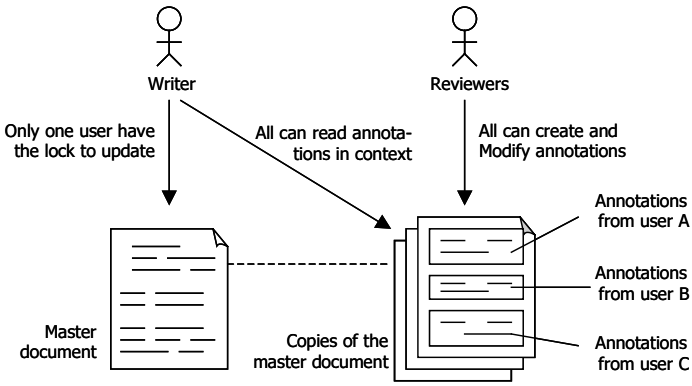

Fig. 2. Concurrent work on a document

The software system consists of four modules. They allow static or mobile operation, as described in the next subsections:

- Web editing module: it is intended as the main program to create, edit and share documents. Roles within the group are also assigned with this module.
- PDA editing module: it also allows creating, editing and sharing documents. It can also manage local documents to the PDA itself.
- Communication module: it manages communication between the PDA and the shared documents database.
- User management module: it handles user creation and privileges within the system. Only an administrator can enter this module.

The system only allows users who have been created with the User management module. The rest of the modules are discussed below.

5.1 Web Editing Module

This module lets users to create, edit and share documents through the Web. When a group member creates a new document, she must provide the list of co-authors and the roles assigned to each of them (the current implementation just considers *reader* and *reader/writer*). A co-author can modify a document by first blocking it; after making changes, she must unblock it. This module also lets co-authors to generate a new document version. Furthermore, the same module allows co-authors to add own annotations and see annotations provided by other users. Figure 3 shows a document being edited via Web.

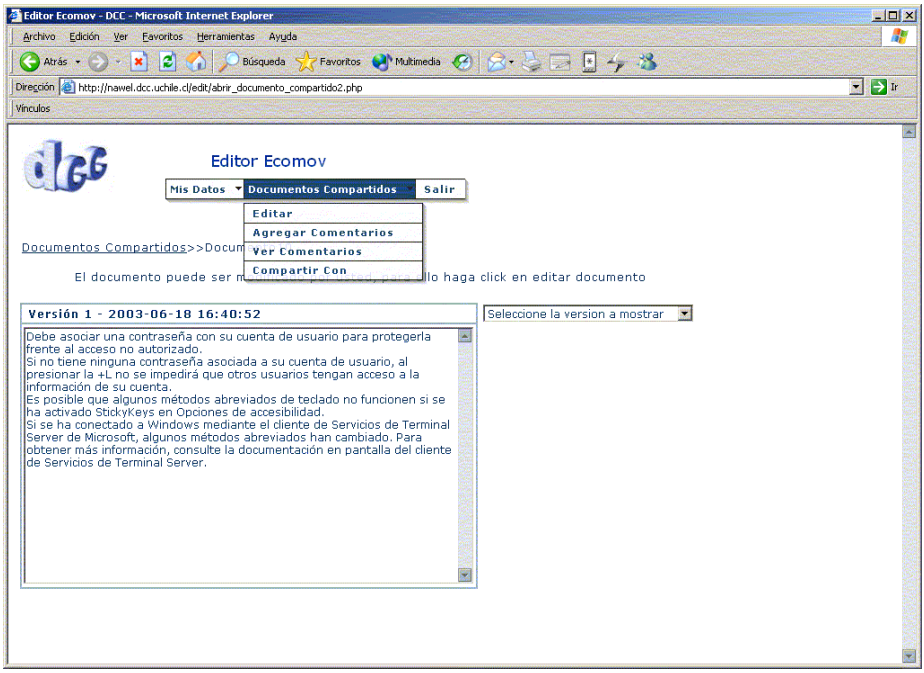

Fig. 3. Editing a document from a Web browser

5.2 PDA Editing Module

The design of our solution tried to make the Web and PDA editing modules as similar as possible. However, taking into consideration the strategy outlined in Sect. 4, the PDA module should be very compact, with a simple user interface, and using little storage. Despite this austere design, the PDA module should still provide the main functionality of the Web module.

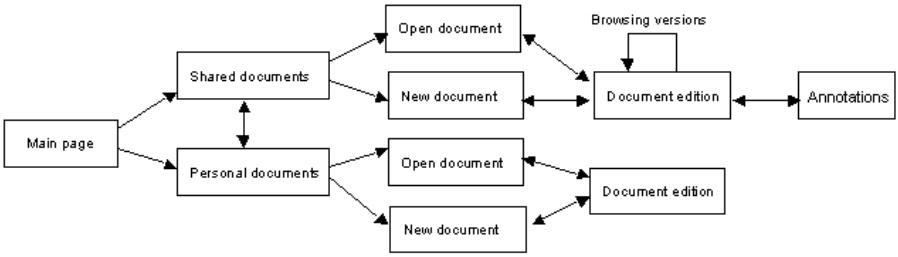

Fig. 4. Navigation model for the PDA editing module

Figure 4 depicts the navigation model of the PDA editing component. Entrance to the system is done through the main page. Here, basic data for connection to the server is initialized. Then, a choice must be made by the user: work on shared documents or personal ones. Shared documents will be the normal choice; personal documents will not be shared when the PDA will get synchronized with the server.

The co-author may create new documents or open previous ones. These may be personal or shared. A typical use may be to create a personal document with an outline of ideas; these are expanded later in a shared document. Shared documents may have several versions, which can be navigated by the co-author. The user can also place annotations on any document from this software module.

Figures 5a and 5b show the PDA editing module user interface. The upper part of the screen has information on the current document. The middle part of the screen presents the document, and the lower part contains the application menu. Fig. 5a shows the “File” menu options. The editing menu has an option to work on the various versions: buttons allow to move forward or backwards on the local document versions.

Annotations can be added to personal or shared documents. In the case of shared documents, a co-author is permitted to include annotations only if he has the corresponding privileges. Annotations creation privileges also include permits to delete them. Annotations are entered as text comments enclosed within braces ({}); see Figure 5b.

5.3 Communication and Synchronization Module

This module allows communication and synchronization between PDAs and the server database. The database is also accessed by the Web editing modules. The main difficulty solved by the communication module concerns concurrency, since several co-authors could be editing the same document at the same time. The module also

solves the document versions management problem. Keys for a simple solution to these problems are the locking mechanism already mentioned and a time stamp associated by the system to each document version. Time stamps are then used by the system itself to guide co-authors on which versions are appropriate for merging.

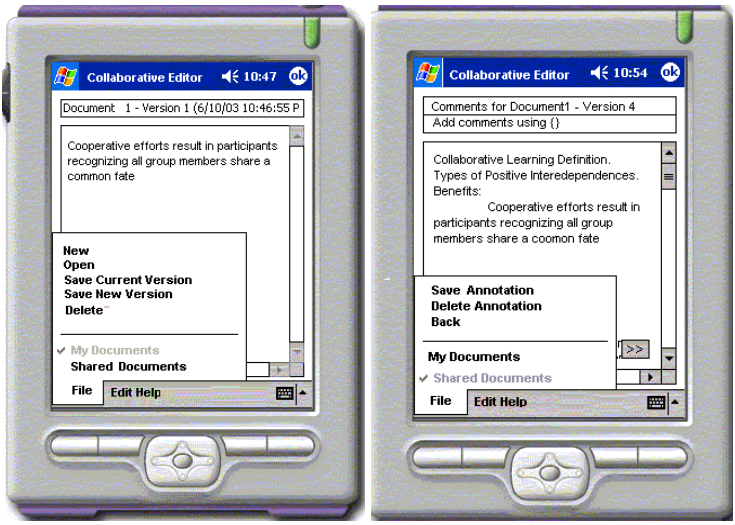

Fig. 5. (a) PDA editing module user interface and (b) adding annotations

The locking mechanism is paired with a unique version of the document, called the master document, as introduced above. When a co-author has blocked the master document, the other co-authors can make annotations over copies of the master document. At a later time, a co-author can modify the master document based on her colleagues' annotations. For such task, the system lets visualize all document copies associated to a master document as separate windows (Fig. 6). Annotations are shown in color to make them easily distinguishable.

6 Related Work

Collaborative text editing has been widely studied. Several tools have also been developed to support it, both for synchronous and asynchronous work, and with various approaches to role assignment, versioning, awareness and editing mechanisms. Some of these tools support our strategy of separating individual tasks from group activities [14,5].

Decouchant et al. have developed a system called Alliance, which allows group participants to ease asynchronous distributed documents edition in a structured way. Stored documents may be remotely accessed through the definition of parts or segments within a document. Such fragments are the basic sharable units [8]. In relation to PREP [28], the approach is to provide information usability and visual representation, emphasized with the use of annotations. Roles are not explicitly supported by this system. GROVE is a distributed synchronic text editor with blocking at a granularity of one character [10].

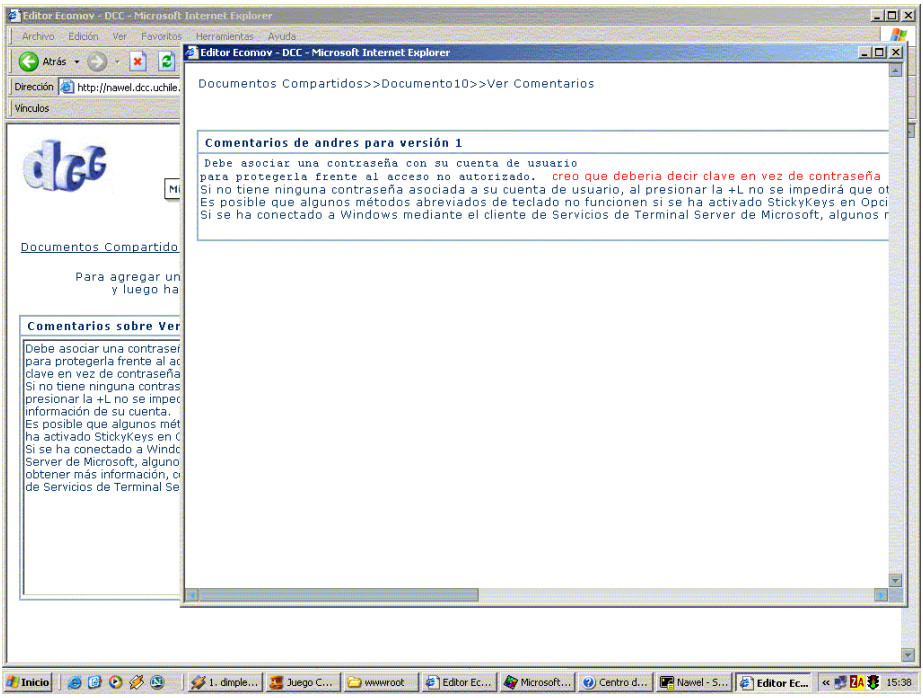

Fig. 6. Improving the master document with a copy containing an annotation

Asynchronous collaborative text co-authoring has been supported by many tools, such as QUILT. This editor bases its support on providing services to very clear roles [22]. Duplex is another such tool, providing support to users connected by Internet [30]. Finally, IRIS allows synchronous or asynchronous collaborative work [21]. Document versioning has long been known. Also, several proposals have been made to present *local versions*: suggested changes to parts of the text. Some of them are change bars [16], flexible diffs [27], active diffs [23] and Stick-Ons [31].

Supporting collaborative work using PDAs is achieved with Shared Notes [12], which allows information interchange among group members. NotePals [7] is a lightweight meeting support system that automatically combines individuals' meeting notes into a shared record. Shared records are essential to maintaining organizational knowledge. Pebbles [26] supports collaborative work both with PDAs and personal computers. Finally, Fieldwise [11] focuses on solving mobility problems; it allows group members to work from their PDAs, coordinating group members for decision making and task performing. None of the previous experiences refers to collaborative text editing.

7 Discussion

We are no longer tied to our desktop computer, since we now have wireless, mobile networking capabilities, and a plethora of new computer technologies. New advances will allow us to access and utilize technology in new ways. Given we often want or need to collaborate with others in many scenarios, it is important we consider the

potential of each technology. In the case of handheld computers and other mobile support devices, each of them has strengths and weaknesses, which should be taken into account to support the collaborative activity. This not only requires communication among devices, but more importantly, it requires an understanding of "how" users can best take advantage of multiple technologies. How should information be distributed across devices? What information is best displayed where? How do users interact with the devices as well as with each other? Which task types should be performed with each device? How should tasks performed in various devices be synchronized?

Of course, some the previous questions may be ignored. It is possible to develop systems which do not take advantage of the devices strengths or without much consideration to the weaknesses. However, the result may be sub-optimal. For instance, it is possible to build a system based on PDAs to be used within a classroom to support typical teaching/learning tasks. In such a case, why not to use a wireless network of notebooks instead, with less problems of batteries running out, transmission interruptions and without so many visualization/input deficiencies? Perhaps the shortcomings outweigh the low cost and small device size of the PDA network in this case.

Joint scientific papers authoring imposes restrictions on the co-authors. They have to be available for discussions, be willing to change some (or all) of the text they have written, etc. One way to facilitate this may be allowing them to work in any place they wish and at any time. It is not an exaggeration to let them do some work while travelling in a crowded underground train: people are increasingly conscious of their time both for work and leisure. PDAs then have something to contribute in this field, since they have such a small size and light weight. Other advantages and disadvantages must be considered, as we have argued, to get a successful system.

Will PDAs evolve to eventually overcome the difficulties we have identified? That cannot be assured, although it is probable some of these current shortcomings will be easied or forgotten. Candidate features to be improved in the short term are batteries, storage capacity and communications reliability.

8 Conclusions

PDAs are appealing and offer many advantages, not the least of which is sustainability. It is their cost effectiveness which makes PDAs an exciting option. The capital investment is low since there is no need for specialized labs, additional wiring, or intensive training. PDA technology is portable, requires little infrastructure and can therefore be readily transferred from one site to the next. It is not platform dependent and readily interacts with any operating system. All these advantages make handheld devices attractive for developers to integrate them in their computer-based systems. However, our argument is PDAs advantages must be considered as well as their disadvantages when designing computer-based systems, in particular, collaborative systems. PDAs must not be used where they are not effective, at least with current technology. We showed an example: the design of a collaborative system for joint document authoring. Although many studies have been done on this subject, none has considered the inclusion of PDAs as part of the authoring process. Our design does not pretend to use these devices in most of the process, but just in those in which they are well suited. Of course, the developed tools are just that: they may be used in the creative process with the frequency or intensity the co-authors themselves consider

most appropriate. As Sharples has pointed out, writing is an open-ended, under-constrained, recursive design task, without formal transitions between states [34].

The example system we developed can certainly be improved: additional awareness, a chat and messaging tool, and blocking granularity at the paragraph level are all new features we plan to incorporate in a second version. Furthermore, experimentation is needed with actual users. Experiments may allow us to answer the question stated in the Introduction concerning PDAs relevance for building collaborative applications. In particular, we wish to have feedback from users concerning various document sizes in order to know whether or not that variable is significant for the application usefulness.

Acknowledgments

This work was partially supported by Fondecyt (Chile) grants No. 1040952 and 1030959 and by MECESUP (Chile) project No. UCH0109.

References

1. Barbara, D., Mobile computing and databases – a survey. IEEE Transactions on Knowledge and Data Engineering 11, (1), (1999), 108-117
2. Bluetooth Website, http://www.bluetooth.com
3. Buyukkokten, O., Garcia-Molina, H., Paepcke. A., Focused Web searching with PDAs. Computer Networks - the International Journal of Computer and Telecommunications Networking 33, (1-6), (2000),. 213-230
4. Chalmers, D., Sloman, M., A survey of quality of service in mobile computing environments. IEEE Communications Survey, (1999)
5. Chambers, R., Crockett, D., Griffing, G., Paris, J., A Java tool for collaborative editing over the Internet. In Proceedings of the ETCE '98, Houston, TX, (1998)
6. Chen L, Suda T. Designing mobile computing systems using distributed objects. IEEE Communications Magazine 35(2), (1997), 62-70
7. Davis, R., Landay, J., Chen, V., NotePals: Lightweight note sharing by the group, for the group. In Proceedings of the CHI'99. ACM Press, (1999), 3388-345
8. Decouchant, D., Quint, V., Romero-Salcedo, M., Structured and distributed cooperative editing in a large scale work. In Groupware and Authoring; R. Rada (Ed.), Academic Press, UK, (1996), pp.265-295
9. DeSanctis, G., Shifting Foundations in Group Support System Research. In Group Support Systems – New Perspectives, L. Jessup, J. Valacich (eds.), MacMillan Pub. Co., New York, (1993), 97-111
10. Ellis, C., Gibbs, S., Rein, G., Design and Use of a Group Editor. In: Engineering for Human-Computer Interaction. G. Cockton (Ed.), Elsevier Science Publisher, North Holland, (1990)
11. Fagrell, H., Forsberg, K., Sanneblad, J., FieldWise: A mobile Knowledge Management Architecture. Proceedings of the ACM Conference of CSCW2000, , (2000), 211-220
12. Greenberg, S., Boyle, M., Laberge, J., PDAs and Shared Public Displays: Making Personal Information Public, and Public Information Personal. Personal Technologies, USA, (1999)
13. Guerrero, L.A., Fuller D. A, Pattern System for the Development of Collaborative Applications. Information and Software Technology Vol.43, No.7, (2001), 457-467
14. Hodel, T., Dubacher, M., Dittrich, K., using database management systems for collaborative text editing European Conference of Computer-supported Cooperative Work, ECSCW CEW (2003), Helsinki, Finland
15. Held, A., Ziegert, T., Service mobility – a new feature of mobile computing. Integrated Computer-aided Engineering 6, (2), (1999), 131-142

16. Irish, P., Trigg, R., Supporting collaboration in hypermedia: Issues and experiences. In Barrett, E. (ed.) The Society of Text: Hypertext, Hypermedia and the Social Construction of Information, MIT Press, (1989), 90-106
17. Jefferson, S., Orubeondo, A., Mobile computing advances on reality. InfoWorld 22, (39), (2000), 57-59
18. Jing, J., Helal, A.S., Elmagarmid, A. Client-server computing in mobile environments. ACM Computing Surveys 31, (2), (1999), 117-157
19. Joseph A. D., Tauber J. A., Kaashoek M. F., Mobile Computing with the Rover Toolkit, IEEE Transactions on Computers Vol. 46, No. 3, (1997), 337-352
20. Kirda, E., Fenkam, P., Reif, G., Gall, H., A service architecture for mobile teamwork. In Proceedings of the 14th international conference on Software engineering and knowledge engineering, Ischia, Italy, (2002)
21. Koch.,M., Design issues for a distributed multi-user editor. Computer Supported Cooperative Work - An International Journal 3(3-4), (1995), 359-378
22. Leland, M., Fish, R., Kraut, R., Collaborative Document Production Using Quilt. Proceedings of the CSCW'88, (1988), 206-215
23. Minor, S., Magnusson, B., A model for semi (a) synchronous collaborative editing. In Proc. ECSCW'93: 3rd. European Conf. Computer Supported Cooperative Work, Milano, Italy, (1993), 219-231
24. Muñoz, M., Gonzalez, V., Rodriguez, M., Favela, J., Supporting Context-Aware Collaboration in a Hospital: An Ethnographic Informed Design, CRIWG 2003. Lecture Notes in Computer Science 2806, (2003), 330-344
25. Myers, B.A., Stiel, H., Gargiulo, R., Collaboration using multiple PDAs connected to a PC, Proc. ACM Conf. on Computer-Supported Cooperative, CSCW'98), (1998), 285-294
26. Myers, B., Using Handhelds and PCs Together. Communication of the ACM 44(11), (2001), pp. 34-41
27. Neuwirth, C., Chandhok, R., Kaufer, D., Erion, P., Morris, J., Miller, D., Flexible diff-ing in a collaborative writing system. Proc. of CSCW'92, the Conf. ofn Computer-Supported Cooperative Work, Toronto, Canada, (1992), 147-154
28. Neuwirth, C., Chandhok, R., Charney, D., Wojahn, P., Kim, L., Distributed Collaborative: A Comparation of Spoken and Written Modalities for Reviewing and Revising Documents. Proceedings of the Conference on Computer Human Interactions, CHI'94, (1994), 51-57
29. Papastavrou, S., Samaras, G., Pitoura, E., Mobile agents for World Wide Web distributed database access. IEEE Transactions on Knowledge and Data Engineering 12, (5), (2000), 802-820
30. Pacull, F., Sandoz, A., Schiper, A., Duplex: A Distributed Collaborative Editing Environment in Large Scale. Departement d'Informatique. Ecole Polytechnique Federale de Lausanne, CH-1015, (1994)
31. Pino, J.A., A visual approach to versioning for text co-authoring. Interacting with Computers 8 (4), (1996), 299-310
32. Rao, B., Minakakis, L., Evolution of Mobile Location-based Services. Communications of the ACM 46 (12), (2003), 61-65
33. Sarker, S., Wells, J.,Understanding Mobile Handheld Device Use and Adoption. Communications of the ACM 46(12), (2003), 35-40
34. Sharples, M., Goodlet, J. S., Beck, E. E., Wood, C. C., Easterbrook, S. M., Plowman, L., Research issues in the study of computer supported collaborative writing, Sharples, M. (ed) - Computer Supported Collaborative Writing, Springer-Verlag, London, (1993)
35. Schulman, E., How to Write a Scientific Paper Journal of the American Association of Variable Star Observers 17, 130, (1988)
36. Stewart, J., Bederson, B.B., Druin, A., Single display groupware: A model for co-present collaboration. In Proceedings of CHI 99 , (1999), 286-293
37. Vizard, M., The way things work is the fundamental problem with mobile computing. InfoWorld 22, (40), (2000), 83-87
38. Zimmerman, T. G., Wireless networked digital devices: a new paradigm for computing and communication. IBM Systems Journal 38, (4), (1999), 566-574