MOBILE ACCESS TO THE INTRANET: WEB CONTENT MANAGEMENT FOR PDAS

Maria Magnusson
Volvo Information Technology
maria.magnusson@volvo.com

Dick Stenmark
Göteborg University
stenmark@informatik.gu.se

Abstract

Although the wealth of information provided by the World-Wide Web is of great value, most of its content is designed for desktop computers. In days when people are becoming increasingly mobile, and hence utilises devises with much smaller screens, much of this information in reality becomes inaccessible. The same is true also for organisational intranets. However, the content management systems (CMS) being used within organisations offer new and unexplored possibilities to present information. In this paper, we describe a CMS-based approach to visualise web information in a PDA, show the benefits of such an approach, and share the lessons learned from deploying this technique in an organisational context.

Keywords: Mobility, content management systems, intranet, PDA

Introduction

The need for and importance of information in today's business context is well documented and since long beyond questioning. The information technology explosion of the last decade is a response to this fact, and the World Wide Web (hereafter the web) has in an unprecedented way lowered the thresholds for information access, outside as well as within organisations. Historically, this information has been designed for and consumed via stationary and increasingly powerful desktop computers. However, in addition to the growing importance of information, we have also witnessed increased mobility amongst office workers and studies report that employees are spending more time away from their desktops (Bellotti and Bly, 1996; Luff and Heath, 1998; Perry *et al.*, 2001). Hence, the need for information access facilitated by mobile devices and wireless connection from wherever you are is – and will continue to be – an issue of great importance within organisations.

When in or around meetings in an organisational context, access to business related information is an expressed need (Lundin and Magnusson, 2002). Much of such information is not found on the public web but on corporate intranets. In our study of Volvo Information Technology (Volvo IT) in Sweden, we have witnessed how the increased mobility is facilitated with artefacts such as GSM phones and PDAs, and how a web infrastructure supporting mobile work is being implemented. Since intranets are implemented in most large organisation, the study of how organisational members interact with the technology is important, however overlooked by many Internet researchers. We aim to contribute in this area.

Accessing web-based information from devices such as PDAs is known to pose difficulties for most users (Jones *et al.*, 1999). Small screens, limited memory, reduced processing power, and low bandwidth constitute special challenges not only for information consumers but equally so for content providers (Freire *et al.*, 2001). In this paper, we focus specifically on the problem of displaying web content on PDA screens. As we show in the related work section, there have been numerous attempts to address this challenge, all with varying approaches and degrees of success. Previous attempts have been focusing on the Internet (Buyukkokten *et al.*, 2000; 2001; Freire *et al.*, 2001; Steinberg and Pasquale, 2002), i.e., a very large and heterogeneous environment. We are instead aiming at the organisation context defined by intranets, which is a more homogenous and, at least in some respects, more manageable environment. In addition, such business environments increasingly use content management systems (CMSs) to host large portions of the available information corpus (Latham, 2002), and we shall argue that this opens new opportunities.

Displaying web content on a PDA, which is the focus of this paper, includes technical as well as design challenges. We will show how the use of a CMS in a novel way can improve the mobile user's access to web based information via a WLAN-connected PDA by keeping sites in dynamic modules rather than static pages, separating information from presentation, and ensuring proper use of metadata. We suggest a new way of presenting web based information based on logical views in a content management tool and show how this affects design. Applied in an intranet environment, our solution gives the end user the option of choosing channel to reach and view the large corpus of existing information. The major strength is that the content itself may remain unaltered.

Next, we shall account for some of the previous work in this field as well as introduce content management as a concept. Thereafter, in section 3, we describe our research approach. Section 4 presents the CMS used at Volvo and describes how we have modified it to comprise also handheld clients. Our experiences from this experiment are presented and generalised in the discussion in section 5. The paper ends with conclusions.

Related Work

Information Visualisation on PDAs

The problem of displaying web information on small devices such as PDAs has not gone undetected, and a number of different attempts to address the problem have been made. Previous research can broadly be sorted into four different categories (cf. Bickmore and Schilit, 1997; Freire *et al.*, 2001); *client-side navigation*; *specialised wrappers*; *duplication of data*, and; *middleware solutions*. Below, each of these approaches is briefly described.

1) Extended *client-side navigation capabilities* was an early approach allowing the user to zoom and/or pan over the displayed document, as in the PAD++ system (Benderson and Hollan, 1994). Although this approach is straightforward and opens up for a great number of different techniques to be used, they all suffer from excessive scrolling requirements (Buyukkokten *et al.*, 2000).

2) Devising *specialised wrappers* allows selected pieces of web information, such as weather information or stock quotes, to be exported to alternative devices. Such wrappers do not require any modifications to the original web page, but they typically have to be individually crafted for each source and updated whenever the source is changed (Freire *et al.*, 2001).

3) *Duplicating the information* by re-engineering the web site to fit a specialised audience is a third way to make information accessible on small screens. For example, Fast Search provides WAP and PDA interfaces to their public search service (see **http://mobile.alltheweb.com**). This allows the sources to display very well in the targeted clients, but requires the content provider to maintain multiple versions of their web sites and the visitors to use different URLs depending on device. It also limits the user to the (small) subset of sites that have chosen this approach.

4) To use a *middleware* or a proxy between the server and the client – allowing customisation without modifying either one of them – is probably the one most commonly used approach. One plausible reason for this popularity is that it gives mobile users immediate access to the vast amount of legacy content already available on the web. Proxies can be entirely remote or used in a combined local/remote configuration (Steinberg and Pasquale, 2002). The local/remote proxy approach has been concerned primarily with bandwidth problems (e.g., Liljeberg *et al.*, 1995; Loon and Bharghavan, 1997), and only partly interested in screen size problems, cf. (Steinberg and Pasquale, 2002). Instead, it is the single proxy approach that most frequently has been used to reformat web pages to better match the display capabilities of client devices, such as in Bickmore and Schilit (1997) or Fox *et al.* (1998). The merit of the single proxy approach is that it gives broad and on-the-fly access to the web, at least in theory. In practice, however, it has been found that complex or badly designed pages translate poorly or not at all (Freire *et al.*, 2001). The technique depends too much on the proxy's ability to interpret the site structure and web sites are just too heterogeneous.

An alternative use of remote proxy is the text summarisation approach where the information is compacted into manageable units that better fit the limited display area (Buyukkokten *et al.*, 2000; 2001). Here, the proxy is used to chop up the web page and order them hierarchically. This approach acknowledges that web pages often have very diverse content and that summarising the entire page does not make sense. A remaining problem is that summarisation always means replacing the real content with an approximation and the correctness of this approximation may vary depending on many variables.

Although our work shows parallels to the single proxy approach, it differs from the above in two distinct ways. First, all of the above efforts have been targeting the Internet, where distributed ownership and large variety in both content and style make many proposed solutions inappropriate. Corporate intranets now represent a significant (and growing) portion of the web and by looking at the problem of accessing web information from small screens from an intranet perspective we contribute to the research field. Second, being in a corporate context, we also contribute by introducing content management systems (CMS) as a feasible way to facilitate web access from PDAs.

Content Management

Content management (CM) is one of those phenomena that have no clear or generally agreed-upon definition. This may be because very little substantiated research on content management is available. Building on the thoughts of Pokorny (2001), we suggest that a content management system (CMS) i) separates content from presentation, ii) provides modularisation of the site, iii) ensures consistent use of metadata, and iv) enables information reuse.

Early claims suggested that web publishing was open to everyone since it did not require technical skills. The way the HyperText Mark-up Language (HTML) has developed has made it increasingly difficult for non computer-savvy users to correctly layout and design web content. The *content/presentation separation* offered by CMSs means that the content provider again is free to author their text without concerns for how it should appear on the screen. The CMS ensures consistent and correct presentation by using predefined templates and style sheets.

The CMS approach also means a *modularisation of the site* since texts are created in smaller "newspaper" blocks rather than as long documents. Content is also separated in different genres such as news, product specs, announcements, reports, or resource collections. A web page typically consists of several such modules that are assembled and rendered at run-time.

For the CMS to know what modules to display on a given page, the system both facilitates and enforces a *consistent use of metadata*. The content provider enters title, genre, or organisational context, which the system validates. Meanwhile, the system itself adds creation date, time and author name. CMSs are often designed so that the authors cannot publish their data without having provided a specified minimum of metadata.

In a CMS, content need not be written for a particular media. Instead, the same piece of content can be output to a printer, a fax, a web page, or any other channel. Even when the web is the primary target, the same information can be displayed on several pages depending on the rules embedded in the templates. This form of *information reuse* means that the same information can be displayed on the Internet site and the intranet site (or sites) without being duplicated. The templates describe the structure of the modules that may appear on the page. Rules control the dynamic relationship between templates and content and define date ranges, genres, and other criteria that must be met for information to be displayed.

The CMS characteristics that we in particular exploit in this work are the modularisation and metadata use. Instead of having a proxy that chops up texts in chunks, the CMS encourage the content provider (i.e., a human) to do this while authoring. The consistent and reliable metadata, which is the result of human entering and machine validation, is then used to correctly filter content in a way that makes it easier for hand-held devices to interact with the information.

Research Site and Method

Both authors are employed by the researched organisation (Volvo IT in Sweden). This means we have inside access to both developers and users, and to the information environment itself, on a daily basis. Our research project may be understood as a mix between action research (Foster, 1972) and case study (Walsham, 1995), since one of the authors has participated actively in the development process as member of the mobility team. We also worked in close cooperation with the team that developed Views (the CM system used within Volvo) since the other author is a member of the content management team. As action researchers our aim has been twofold: on the one hand to contribute on practical concerns of the people in the organisation by being involved in the actual action process (Rapoport, 1970) and, on the other hand, to apply a scientific perspective by simultaneously study the process of the development and implementation of the system (Lindgren, 2002).

Our roles as action researchers in the organisation and in the project can be described in terms of change agents, since problem-solving and change becomes part of the research, which in turn yields new knowledge about the studied phenomena (Foster,

1972). There are dilemmas associated with action research (especially so when the researcher is part of the researched organisation) and they are related to ethics, goals, and initiatives. Each of these means risks of leading away from science, for instance by personal over-involvement or due to project and organisational requirements and goals (Rapoport, 1970). Awareness of the action researcher's dilemmas and a continuous and open dialogue about how we actually influence the direction of projects and thereby the involvement of the organisation have been important for us in ensuring we produce useful scientific results.

A number of different parameters led to the start of the project described in this paper. The need in the organisation to reach business information as well as the increased use of PDAs encouraged the mobility steering committee at Volvo IT to initiate an improvement of the intranet site promoting and advertising mobility in the organisation. Being about mobility, and since the number of PDA's in the organisation now reaches some 3,000 (last count May 2002), the committee wanted this site to be accessible also from PDAs with WLAN-connections. While trying to figure out how to implement this, the development team realised that the emerging solution easily could be generalised to encompass all of the approximately 140 different intranet sites hosted by the VIEWS system. The goal of the project was therefore enlarged and so was the amount of potential users of the application.

System Overview

In this section, we shall describe VIEWS – Volvo's Content Management tool – in a schematic way, followed by a semi-technical description of the add-on module that enables VIEWS information to be displayed in a PDA.

VIEWS – A Content Management System

In VIEWS, the construction of the site is separated from the provision of content. When a site is designed, each *node* (or page) is made up by a set of *logical views* (LVs or partitions, see figure 1), not unlike the concept of frames except that no scrolling is allowed inside a logical view. For each logical view, the site owner decides what *type* of content it should contain; an image, a welcome message, news bulletins, or another category from a provided list. Some logical views contain pre-defined content, e.g., the *menubar*.

The site owner then applies a set of *selection criteria* for each logical view to filter the content, e.g., to only include news bulletins where company equals "Volvo Penta" and where publication date is no older than 7 days.

Nodes may be ordered in hierarchies by using standard file system constructs. VIEWS automatically maps the file structure and creates a corresponding navigation menu that is presented in a specific logical view (the menubar, area B in figure 1). In addition, contact information and search capabilities are automatically generated and incorporated in a consistent way.

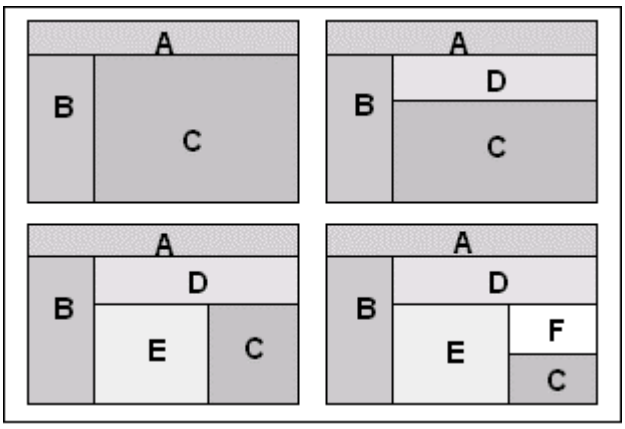

Figure 1. Examples of Four Different Node Types with 3, 4, 5, and 6 Logical Views, Respectively
For example, these may be A) Titlebar, B) Menubar, C) News area, D) Intro area, E) Story area, and F) Image area.

Content is added by content providers who (generally) do not write their information for a particular node, but independent of the various sites, nodes, and LVs. As part of the editing, they are guided through the process of selecting proper metadata for the information they are adding. First they typically identify what information element (IE) they are creating. VIEWS have a number of different IEs, e.g., Story, Bulletin, or Introduction. Though some IEs may have unique meta information (e.g., Alt-text for an image object), most IEs share a set of common metadata variables such as Headline, Language, Creation date, Publication date, Author, and Company name. These metadata variables are used for the selection criteria as described earlier.

When the metadata is in place the author adds whatever content is about to be presented. The content is typically a relatively short piece of information, covering mainly one specific topic. Any time during the process, the author can choose to save the IE as a draft, and whenever the process is completed the author may publish it immediately, or publish it at a future date and time. However, unless all required metadata variables are specified and validated correctly, the IE may only be saved as a draft.

A Quick Run-Trough of the System Logic

When a particular node is requested by a user, the corresponding logical views are populated with content matching the selection criteria and displayed in the user's browser. Figure 2 shows the Mobility project's homepage as rendered in the browser (left) and the underlying LVs and IEs (right).

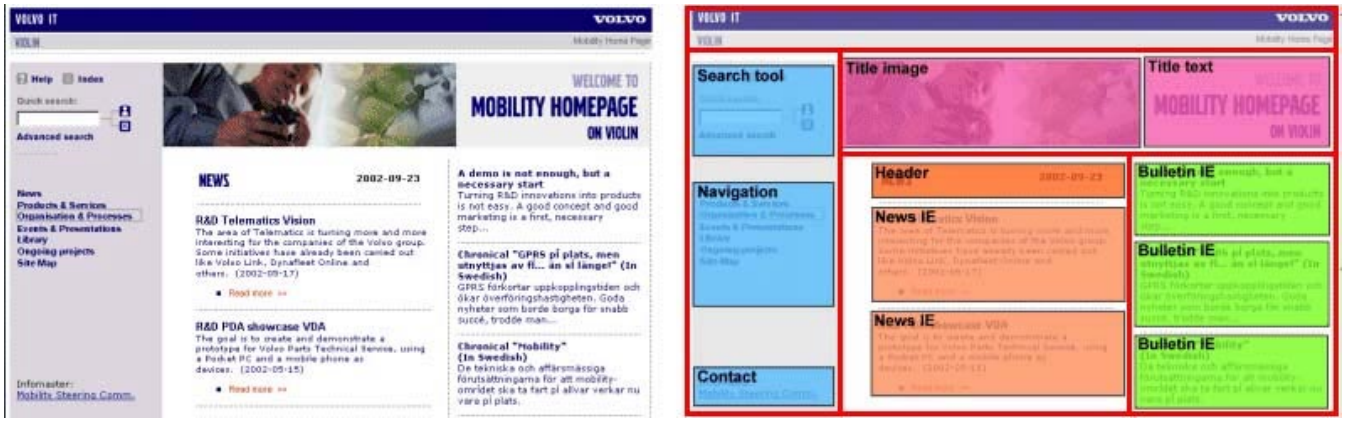

Figure 2. The Mobility Homepage Seen in a Browser (left) and with the Logical Views and Information Element Visualised (right)

VIEWS is built on top of the StoryServer platform from Vignette, Corp. (www.vignette.com) and run in a UNIX environment. In our case we used the 5.0.6 version of StoryServer. Whilst the StoryServer platform provides the basic functionality of a CMS, VIEWS was designed to support the editorial process. The content entered is stored in eXtensible Markup Language (XML) format in a StoryServer internal database. XML was designed to meet the challenges of large-scale electronic publishing, but is also playing an increasingly important role as an independent data exchange of format. The fact that XML was already used helped us implement the required modifications. VIEWS and the modifications added by us are coded in a scripting language called Tcl (pronounced *tickle*). Tcl is intended primarily for issuing commands to interactive programs such as text editors, debuggers, and shells, but it is also programmable, so Tcl users can write command procedures to provide more powerful commands. Tcl also contains its own library package that can be embedded in application programs, thereby allowing applications to extend Tcl with additional commands specific to that application. We used this to invoke built-in StoryServer routines.

Every page in VIEWS is generated in the same way as illustrated in figure 3. When a client requests a node, the call is intercepted by the Authorisation module (1). The Authorisation module verifies that the requested node indeed exist and is a valid VIEWS node. If the node is found valid, control is passed to the Page generation module (2). Page gen calls the database and asks what type the node is, i.e. how many logical views it contains. It then also retrieves the templates for each logical view needed to render this node (3). All this information is then passed to the logical views module (4), which executes all selection criteria and populates the different logical views with content. The page is then sent to the client (5) where it is rendered as a cohesive page.

The separation of design and actual content provided us with the opportunity to easily customise VIEWS to accommodate handheld clients by modifying the above flow as illustrated in figure 4. Code was added to the Authorisation module making it possible to identify the device and type of browser used by a visitor (1). If the client is identified as a PDA, control is passed to the specific PDA page generator that we devised (2). In addition to the actions performed by the standard page gen, our module contains logic to filter out logical frames not containing content. These includes banners (Logical view A, Figure 1), style elements, and the menu (Logical view B, Figure 1). These elements are instead provided for in another, more PDA-friendly way as illustrated below. The templates for the remaining logical frames are collected as before (3). The logical views module finally receives the complete set of information needed to populate the page (4), and sends the resulting page to the browser (5). As is seen in figure 4, only minor modifications to the standard process were necessary.

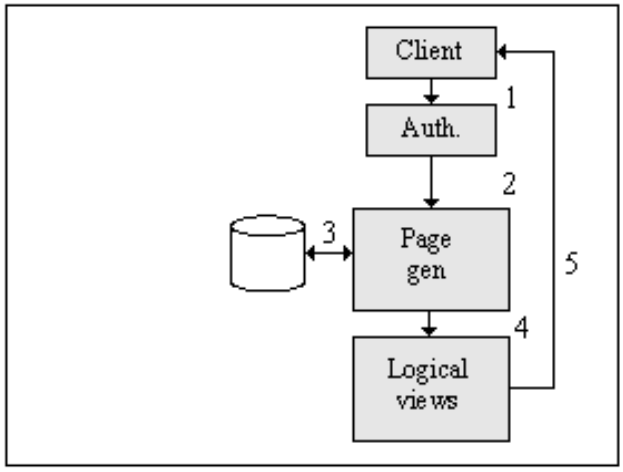

Figure 3. The Original Process Flow in VIEWS when a Stationary Client Requests a Page

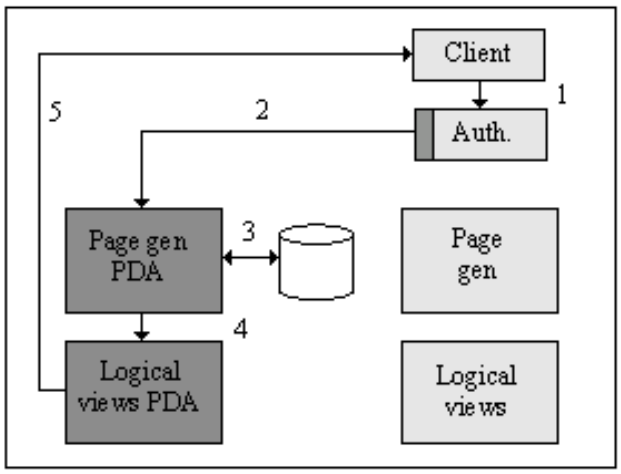

Figure 4. The Modified Flow in VIEWS when a PDA Client Requests a Page

Design and Navigation in the PDA

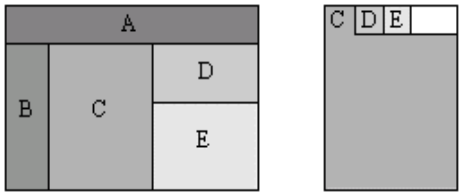

Figure 5. Correspondence between Logical Views in a Regular Web Browser (left) and a PDA (right)

In a regular web browser, all logical views are shown simultaneously as indicated in the left half of figure 5. Since this approach is unsuitable for the PDA's smaller screen, the PDA instead renders these one at a time. The user navigates between the logical frames by means of clickable tabs, as illustrated in the right half of figure 5. Frames A and B are filtered out since they contain no editorial content. However, frame B holds the menu and the functionality usually provided by a menu thus has to be offered by other means. Figure 6 below, which contains a screen dump from our implemented system, illustrates in more detail how the intranet design and navigation appears on a PDA screen.

Discussion

While our implementation is tailored-made for the proprietary VIEWS system, the approach can easily be generalised to most other CMSs and is therefore of general interest. Our primary contribution is the discovery of how CMSs through small means can be used to make large amount of information available to PDA users. The solution is attractive in that it requires a relatively small

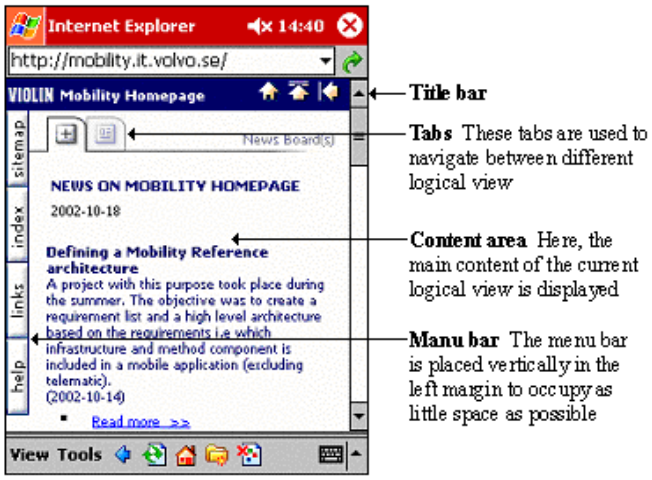

Figure 6. An Intranet Page Viewed in a PDA Browser

manipulation of the CMS code and leaves most of the system intact. Perhaps even more important is the fact that the site owners and the content providers need not do any modifications or adjustments; they continue to design their sites and add their content regardless of the target medium. In addition to the modest modification requirements, our solution has the benefit of instantaneously providing access to many web sites, thereby quickly reaching a critical mass. An approach such as re-engineering of web sites has to be applied to each and every site in turn, meaning that it takes long before the mobile worker has access to the same amount of information as has the desktop user.

Relating our solution to the four categories discussed in the related work section we see that our solution is a bit of a hybrid approach. There are similarities between our implementation and the remote proxy approach inasmuch as the authorisation module sits between the client and the rest of the CMS and intercepts all incoming access requests. However, it is not truly proxy since the authorisation module actually resides within the system and our solution thus requires modifications of the CMS. Although not being a strict proxy implementation, we still manage to maintain the advantages of a proxy solution (i.e., immediate access to already existing sites). Since we are modifying the original CMS, one may argue that our solution belongs to the web site re-engineering category. Undoubtedly, we provide two different interfaces by altering the server and maintaining duplicated sets of page generator modules. Still, there are significant features that separate our approach from web site re-engineering. Modifications are made not on the web server but on the overarching framework, which means that we do not require the site owners to do any modifications; we do not oblige the content providers to adjust their information formats, and; we do not duplicate any data. We consider maintaining two versions of the Page generation module a low cost for the benefits achieved. We have witnessed a growing interest in Content Management systems over the last few years and Gartner Group predicts this trend is accelerating (Latham, 2002). This means that an increasing number of organisations are already relying on CMSs for their web environments, and a significant amount of web-based information is residing in such systems. Consequently, CMSs should be understood not as a marginal phenomenon but as a new and interesting research field and we here show how it can support the mobile PDA user.

The content/presentation separation that CMSs typically offer makes these environments less media dependent. It was therefore easy to PDA-enable the entire content base by only re-writing the Page generation module. We only had to remove the logical frames containing presentation-specific information. The modularising that comes with a CMS affords content providers to keep their contributions short and discourages them from writing long novels. By providing templates for abstracts and paragraphs, the systems assists the author in breaking up the content in small modules that better fits the limited screen area of a PDA. This reduces the need for text summarisation, which is good since summarisation – even if sophisticated – always means that some information is cut out.

The fact that consistent metadata is associated with every information element makes it possible to select information matching specific criteria, e.g., genre=news, author=Magnusson, or info_age<5days, which makes information more accurate and thus more appealing. Although this metadata today is used only on a page level and not individually, it may be possible in a future implementation to filter not only based on device but also on user. In such a system, each user would be able to define his or her

own selection criteria, which would even further increase information precision and decrease the amount of information needed to be transferred to the PDA.

The small modification made to the VIEWS system that we have described in this paper has enabled the organisation to take advantage of one reuse-feature typically offered by CMSs. Seen from an organisational perspective, the whole information base that resides in the system has been made available on a different medium, and the organisation can now reach also their mobile employees. From an individual perspective, our solution has given the mobile worker access to the entire information base that the 140 current web sites contain in a PDA-friendly way.

Conclusions

We have illustrated how Content Management systems can be used to provide content in a form suitable for PDAs and other small screen devices. The growing use of CMSs makes this approach a feasible intranet alternative to more complicated text summarisation or web site re-engineering methods. The filtering out of entire logical views makes this a very cost-effective solution since the content of the site need not be analysed.

We would like to highlight the following:

- Our approach is applicable when the corporate intranet supports the users' everyday activities and when employees are likely to carry out part of their work away from their desktop computer.
- With a small effort, the mobile PDA users are given access to the entire corpus of legacy information already available on the corporate intranet.
- No modifications are required from a content provider's perspective; they can focus on producing information regardless of output media
- No additional code or plug-ins required on the client side; all necessary code alterations are made on middleware level.

References

Bellotti, V. and Bly, S. "Walking Away from the Desktop Computer: Distributed Collaboration and Mobility in a Product Design Team", in Proceedings of CSCW '96, ACM Press, Boston, MA, November 1996.

Benderson, B. and Hollan, J. "PAD++: A Zooming Graphical Interface for Exploring Alternate Interface Physics", in Proceedings of UIST '94, ACM Press, Marina Del Ray, 1994, pp. 17-26.

Bickmore, T.W. and Schilit, B.N. "Digestor: Device-independent Access to the World Wide Web", in Proceedings of WWW6, Santa Clara, 1997.

Buyukkokten, O., Garcia-Molina, H., Paepcke, A., and Winograd, T. "Power Browser: Efficient Web Browsing for PDAs", in Proceedings of CHI 2000, ACM Press, The Hague, 2000, pp. 430-437.

Buyukkokten, O., Garcia-Molina, H., and Paepcke, A. "Seeing the Whole in Parts: Text Summarization for Web Browsing on Handheld Devices", in Proceedings of WWW10, ACM Press, Hong Kong, May 2001, pp. 652-662.

Foster, M. "An Introduction to the Theory and Practice of Action Research", in Work Organizations, in *Human Relations Volume 25, Number 6* (1972), pp. 529-556.

Fox, A., Gribble, S.D., Chawathe, Y. and Brewer, E.A. "Adapting to Network and Client Variation Using Infrastructural Proxies: Lessons and Perspectives", IEEE Personal Communications, Special Issue on Adaptation, vol. 5, issue 4 (August 1998).

Freire, J., Kumar, B., and Lieuwen, D. "WebViews: Accessing Personalized Web Content and Services", in Proceedings of WWW10 (Hong Kong, May 2001), ACM Press, pp. 576-586.

Luff, P. and Heath, C. "Mobility in Collaboration", in proceedings of CSCW (Seattle, November 14-18, 1998), pp. 305-314

Jones, M., Marsden, G., Mohd-Nasir, N., Boone, K., and Buchanan, G. "Improving Web Interaction on Small Displays", in Proceedings of WWW8 (Toronto 1999), ACM Press, pp. 51-59.

Latham, L. "Web Content Management Trends Accelerate", Market analysis, 30 September 2002, Gartner Group.

Liljeberg, M., Alanko, T., Kojo, M., Laamanen, H. and Raatikainen, K. "Optimizing World-Wide Web for Weakly Connected Mobile Workstations: An Indirect Approach", in Proceedings of SDNE '95 (Whistler, June 1995), pp. 132-139.

Lindgren, R. Competence systems, Doctoral Thesis, Gothenburg Studies in Informatics, Göteborg University, 2002, ISSN 1400-741X

Loon, T.S and Bharghavan, V. "Alleivating the Latecy and Bandwidth Problems in WWW Browsing", in Proceedings of the USENIX symposium on Internet technology (Monterey, 1997).

Lundin, J. and Magnusson, M. "Walking & Talking, Sharing Best Practice", in IEEE international workshop on Mobile and Wireless technologies in Education (WMTE), (Växjö, August 29-30, 2002).

Perry, M., O'Hara, K., Sellen, A., Brown, B. and Harper, R. "Dealing with Mobility: Understanding Access Anytime, Anywhere", ACM Transactions on Computer-Human Interaction (TOCHI) December (2001) Volume 8, issue 4, pp. 323-347.

Pokorny, J. "Static pages are dead: How modular approach is changing interaction design", Interactions (September/ October 2001), pp. 19-24.

Rapoport, R. N. "Three Dilemmas in Action Research, With special Reference to the Tavistock Experience", in *Human Relations Volume 23, Number 6* (1970), pp. 499-513.

Steinberg, J. and Pasquale, J. "A Web Middleware Architecture for Dynamic Customization of Content for Wireless Clients", in Proceedings of WWW 2002 (Honolulu, May 7-11, 2002), ACM Press, pp, 639-650.

Walsham, G. "Interpretative Case Studies in IS Research: Nature and Methods", European Journal of Information Systems, 4, (1995), pp. 74-81