INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

Conceptual Multi-Device Design: Improving Theoretical Foundations

Rodrigo de Oliveira   *Heloísa Vieira da Rocha*

Technical Report - IC-07-001 - Relatório Técnico

February - 2007 - Fevereiro

Conceptual Multi-Device Design: Improving Theoretical Foundations

Rodrigo de Oliveira* Heloísa Vieira da Rocha

Abstract

This work presents the Conceptual Multi-Device Design (CMDD) with a deeper discussion about its theoretical assumptions. The proposal suggests multi-device design by maintaining the application's conceptual model (wider perspective, including navigational and presentation models) on every interface to avoid ambiguities on the user's mental model. This consistency gives support to decision making problems, allowing users to behave according to their previous experience while executing one task on different interfaces of a given application. The CMDD framework that provides mobile access (with pocket PCs or smartphones) to desktop web interfaces is improved and the first impressions with beta prototypes are presented. We expect to conduct complete user evaluations sooner for a better identification of this proposal's advantages.

1 Introduction

Mobile devices introduced a great challenge for Human Computer Interaction: to develop multi-device interfaces for existent applications. Some have tried device oriented designs with linear transformations, creating mobile interfaces from scratch, like Avantgo (`www.avantgo.com`) and Usable Net (`www.usablenet.com`); others looked for dynamic and automatic adaptations, but still focusing on the device [2, 5, 8]. These and other related approaches were well received by many mobile users who could finally access applications on their handhelds with better usability. The main reason is that the application had its whole interface restructured according to each device's feature (smaller screen space, ability to talk, no keyboard available, etc.). However, the new interfaces generated are usually different from the original and lack in usability when users need to change from one interface to another (e.g. desktop computer to a cell phone), especially for refinding and/or comparing information [9, 11]. Isolated usability tests on these new adapted interfaces guarantee the desired goals, but they can't do it when the user needs to interact with all of them to execute the same task. This happens because the original interface was built under a certain conceptual model which is forgotten on the next interfaces development, overlooking many of the user's cognitive processes.

These observations led to the Conceptual Multi-Device Design (CMDD) proposal [15], defending the hypothesis that one application shall not demand as many conceptual models

*Institute of Computing, University of Campinas, 13081-970 Campinas, SP.

as the final media devices to achieve its maximum usability. Since then, many contributions were received from researchers, students and handheld users, which demanded better explanations about the CMDD assumptions and application on real case studies. This work presents a deeper discussion about the main theoretical issues concerning the CMDD and also improves the web system transformation framework [16] proposed on a previous work for dynamic adaptation of desktop web interfaces to smaller screens. The first prototypes were constructed and some informal results point to the acceptance of this approach.

2 Discussing the Conceptual Multi-Device Design

The multi-device interface design approach proposed by Oliveira & Rocha [15] states that one application should have the same conceptual model presented on the n interfaces available but also ensuring good usability. Here, the term conceptual model is in accordance with the definition given by Preece, Rogers & Sharp [18]: *it's a description of the proposed system in terms of a set of integrated ideas and concepts about what it should do, behave and look like, that will be understandable by the users in the manner intended.* From the definition, it is clear that this conceptual model has a much wider perspective, concerning not just class diagrams with attributes and relationships between them [1, 20], but also the interface's behavior (navigational model), look and feel and implementation (presentation model).

Affirming that CMDD suggests not changing the conceptual model between the interfaces of an application means that the user's model for this application should always be the same. It doesn't matter if he/she wants to check an account balance on an ATM machine or by phone. The interaction should be as close as possible to his/her previous experience with other interfaces for the same application. On this sense, CMDD is much more oriented to the user than to the device.

This last paragraph is enough to start a long discussion about the CMDD assumptions. In the next subsections, many gaps between theory and practice for this design methodology will be filled in order to make things less obscure within the CMDD proposal.

2.1 The User Mental Model

The main question about this topic raises the doubt if users will build a mental model of an application domain on the first device they use and, if so, if they will have difficulty in adapting these mental models to a new and different platform [19]. According to the logic definitions for inductive inference, decisions are made based on previous experiences, which means they must, somehow, be stored in the brain. These internal constructions that can be manipulated enabling predictions are called mental model [6]. If users weren't able to build this mental model from a first interaction, they would be like a RAM memory, loosing its context state after every shutting down. In fact, humans not just build this mental model, but also adapt it. The problem is that bad results will be obtained by using this gift on a context full of different devices to access the same application. Different conceptual models demand individual maintenance, generating new inconsistencies to which the user will have to readapt. Hence, it doesn't matter how good people are to adapt their mental model;

misunderstandings will arise from these improvements, causing frustration, uncertainty and distrust.

2.2 Maintaining the Conceptual Model on Contrasting Devices

Consider the example of a system designed for users with many diversified needs, like mobile, static and speech interactions. Hence, in this case, the desktop PC (graphical user interface), the pocket PC (pen-based interface) and the telephone (speech user interface) are reasonable devices to attend the requirements. However, each one has different conceptual models and it seems like the CMDD user centered approach got itself into trouble letting the user choose the access medias with conflicting interaction modes (instructing, manipulating, conversing, etc. [18]). The best way to solve this misunderstanding is to forget the devices' conceptual model and focus on the application's conceptual model. Users will get all the interaction modes they need, but these will be externally consistent. In other words, the voice commands may be discursive on the phone, which is different from the desktop, but the system will interpret them as if they were mouse "clicks". In the user's mental model, the tasks always follow the same action flow, despite being words written, typed or said.

2.3 Does Different Contexts Suggest Different Applications?

The argument of different tasks on different contexts requiring different conceptual models has a similar discussion as in the latter subsection. The key assumption about maintaining the application's conceptual model on each interface doesn't prevent additional or less features on each device, but suggests the same action, behavior and visual appearance for similar tasks. Consider an example of browsing web pages on a handheld and on a desktop PC. Will users see them as the same kind of browsing? People are very different from each other and it's probably a consensus that mobile users have different perspectives than other regular users. The CMDD proposal doesn't stay against this assumption, but there is a subtle misunderstanding here. If we could affirm that none of these mobile users would ever had to browse web pages on a desktop nor the regular users would browse on a handheld, there would be no problem on changing the conceptual models. But if any of these users have to access the system through more than one of its interfaces, than the task should be accomplished according to his/her previous experience. No doubt there will be different programs performing the same task, some with additional features and others with less, but common tasks should always share the user's model developed on the first interactions with the referent device.

2.4 Moving from Theory to Practice

Although these questions emphasize some important theoretical issues, there is still the need to fill gaps between knowing and applying the theory. In other words, to check if the application's conceptual model hasn't changed on its interfaces. First, it's important to remember that maintaining the conceptual model is a mean to achieve an end: good usability for multi-device interfaces. Hence, checking the usability through common user evaluations and other testing methods is enough considering the main interests for Interaction Design.

Anyway, the designer will probably need guidelines or formal methods to maintain the same conceptual model. The best approach is analogous to black-box tests; it doesn't matter what technologies are involved or how the system processes the user instructions: the interaction flow to execute a certain task and the system's look and feel should be as close as possible on every interface. According to Oliveira & Rocha [15], this can be accomplished with a lifecycle model that considers the interaction design an endless process beyond the scope of a single product development, which is just one of the interfaces that will be available for the application. Designers concentrate on prospective user needs ensuring new potential related products will be identified and built according to the same design model. Figure 1 presents a CMDD adaptation of Norman's framework [14] illustrating the relationship between a system's design and what the user understands of it.

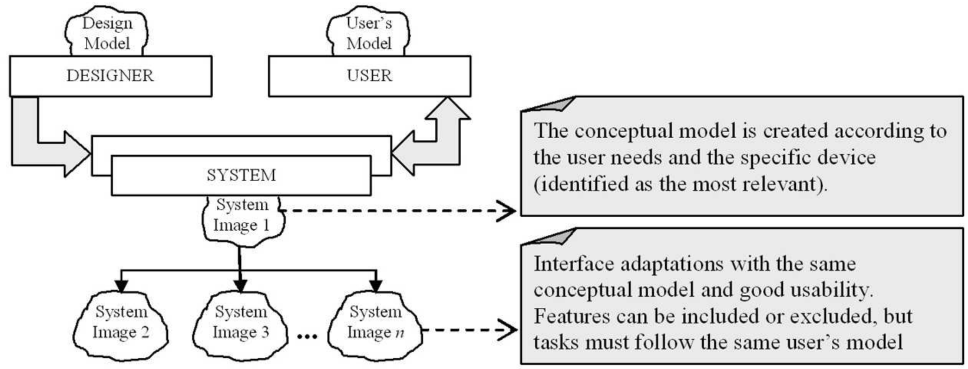

Figure 1: Interactive components from Norman [14] adapted for CMDD. Changing the conceptual model would result in n models for each component.

Following the best idea of checking the conceptual model through user evaluations, it could be argued that opposing approaches, defending a more device centric design, realized these evaluations on their works with success. But again, these evaluations tend to be conducted with just one of the interfaces for a given application and not with all of them. Without such additional analysis, the users' satisfaction with the given product has nothing to do with the application, but with one of its interface instances. Therefore, it would be better to apply a mix of diagnostic and definitive evaluation techniques using task oriented interaction experiments together with inspections. For example, cognitive walkthroughs [17] might help to identify problems related to conceptual model changes and also contrast results from evaluations of each interface. We plan to conduct these user studies for the CMDD prototypes discussed on the next section.

These are just a few but important theoretical issues concerning the CMDD that should be clarified for a better proposal maturing process. There is no expectation to put a closure on the subject, but to improve the arguments towards user centered approaches for applications with multi-device access needs.

3 Towards An Empirical Validation

Oliveira & Rocha [16] presented an example to implement the CMDD proposal through a framework able to adapt desktop web pages to handheld devices with smaller screens, like pocket PCs and smartphones. This framework is readdressed here after some improvements identified for its implementation and application. The e-learning domain was chosen to illustrate the prototypes due to the fact that schools and universities are actual great sources for spreading technologies and, consequently, future multi-device access needs (electronic boards, projectors, laptops, pocket PCs, cell phones, etc.). The following subsections describe the theory behind the framework, the conditions proposed to implement/use it and the impressions identified on informal tests realized with the first prototypes.

3.1 The Framework For Web System Transformation

According to Mackay, Watters & Duffy [11], web page transformations can be divided into three categories:

- *Direct Migration* - No transformations are made to the web page. The user generally navigates using extensively both horizontal and vertical scrolling. Although the same conceptual model is maintained, the interface design lacks in visibility and efficiency of use;
- *Linear Transformation* - The original web site is changed to a long linear list that fits within the width constraints of the small display. Used by sites like Avantgo and Usable Net, it usually breaks the original application's conceptual model;
- *Overview Transformation* - An overview of the original page is provided and, for the most, content remains the same.

Among these categories, the latter is the one with the closest works related to the CMDD proposal. Following are listed three of them:

- *Smartview* [12] - A thumbnail view of the original web page in zoom-out, fitting the screen horizontally. As a result of this shrinking, texts become illegible and the approach tries to overcome this problem partitioning the page in logical regions bounded with lines (Figure 2); when one of these regions is selected, content is presented with good visibility inside the screen space on a detailed view;
- *Gateway* [11] - Similar to Smartview, but without the region bounds. Also, the detailed view uses a focus-plus-context technique, enlarging the selected region over the detailed view, as shown on Figure 2;
- *Summary Thumbnail* [9] - Preserves the page layout using the same thumbnail approach of Smartview and Gateway, but the texts are summarized enabling a good legibility (fonts are enlarged to a legible size and characters are cropped from right to left until the sentence fits on the respective area). However, the detailed view with full

text (accessed through one click to a clean area of the page) is a direct migration and has no adaptation to the screen size. Moreover, the summary is language dependent and may get undesirable results, as can be seen on Figure 2.

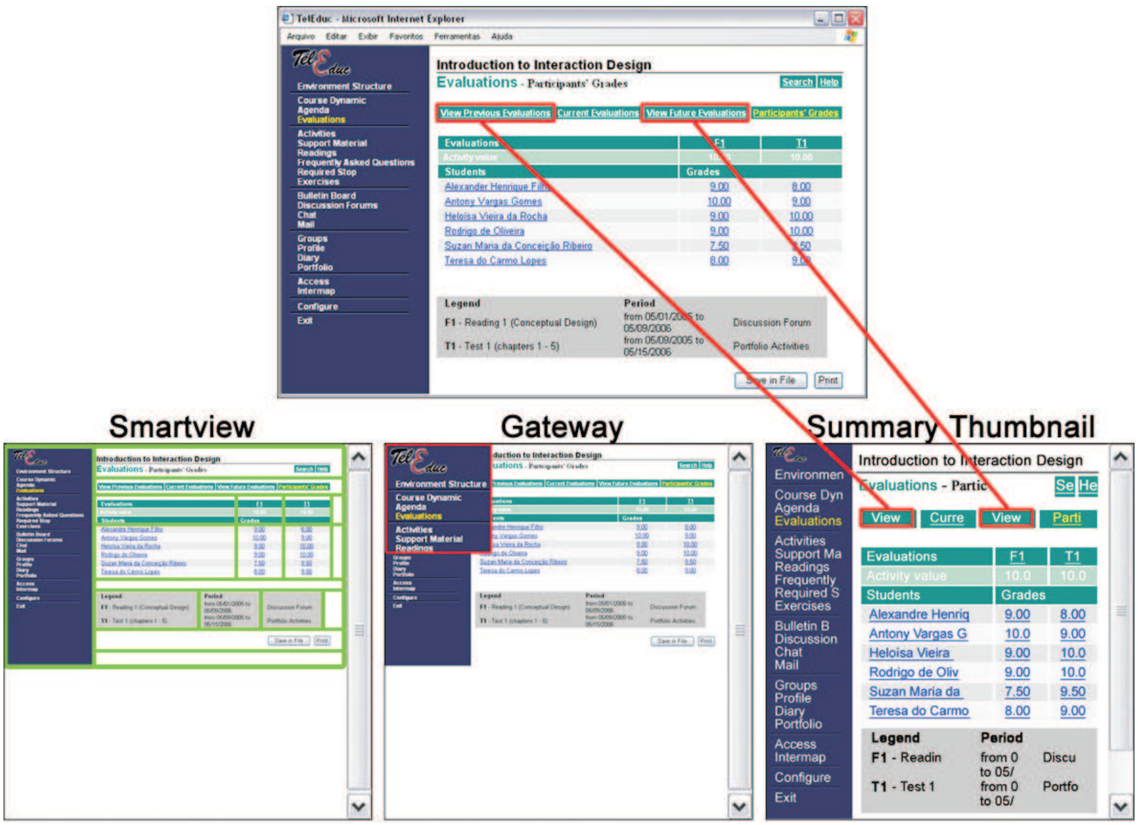

Figure 2: Comparison between three overview transformation approaches: Smartview is completely illegible until a region is selected; Gateway has a cleaner aspect, but also requires an interaction to read any text; Summary Thumbnail has the best visibility, but the text reduction generates ambiguities (in the example, two different links with the same label). Also, the detailed view access requires pointing a non-hyperlink object (causing interaction fear) and then a Direct Migration approach is applied, with extensive scrolling.

Among the transformation techniques presented, Summary Thumbnail has the best usability trade-offs, ensuring good visibility and still providing almost the same conceptual model. Basically, this isn't the same because of:

1. A simpleton summarization approach generating ambiguities on the sentences reduced for the navigational links;

2. The new concepts of *thumbnail view* and *detailed view* along with their access procedures, resulting interaction fear and context lost with Direct Migration.

If localization testing [13] is considered, the first problem can lead to even worse results. In English, adjectives come before nouns, which is fine on most of the cases for the right-to-left cropping approach. For example, anchors named "previous evaluations" and "future evaluations" are cropped to something like "previous" and "future", much better than two links with the same label "evaluations" (at least when these links are available on an evaluation context). However, for languages such as Portuguese, Spanish, Italian and others which nouns tend to appear before their adjectives, this text reduction approach won't work.

Figure 2 shows an example in which the summarization generates two links with the same label even for English sentences. This could be solved by a summarization process based on lexical or semantic analysis. Lexical analysis chooses the most important words based on their occurrences on the document and the whole collection of documents. This approach can solve most of the problems, but doesn't handle grammar issues, like synonymy (different words with the same meaning) and polysemy (several meanings for the same word). For example, while analyzing a text with the words *teacher* and *professor*, the lexical analysis would consider them as distinct words, giving each one a different weight. A better approach would identify these words have the same meaning and would consider them as the same word. This improvement can be done by a semantic analysis like the LSA [7]. However, this approach reduces the computational efficiency with too many matrices products and decompositions [10].

In order to adequate efficiency on runtime web page transformations and the semantic analysis, we suggest applying lexical analysis like the one given by Buyukkokten et al. [4] with some restrictions. This method uses the TF/IDF technique (term frequency / inverse document frequency) to calculate the importance of each word and chooses the appropriate ones to be extracted from the document. The word is important if it occurs frequently within the document but infrequently in the larger collection. This collection may be a database containing web pages from a specific domain (e.g. sports news, e-learning environments, etc.). Equation 1 shows the formula used to calculate each word's importance.

$$w_{ij} \quad = \quad tf_{ij} \times log_2 \frac{N}{n} \tag{1}$$

w_{ij} is the weight of term T_j in document D_i;
tf_{ij} is the frequency of term T_j in document D_i;
N is the number of documents in collection;
n is the number of documents where T_j occurs at least once.

According to Buyukkokten et al. [4], the TF/IDF should be used together with their within-sentence clustering technique. In summary, the TF/IDF is used to identify relevant words and the within-sentence clustering to choose the relevant text fragments according to each word's weight. As stated before, we suggest using this method with some restrictions, which are explained following by dividing the texts to be summarized in two major groups:

- *Long texts summarization* - the results of long texts summarization are always questionable. We suggest using the right-to-left cropping approach that will also avoid additional processing time waste with complex text reduction methods;

- *Navigational links summarization* - anchors, buttons, hyperlinks and other access structures are the only interface objects that we expect to be considered for summarization as their corresponding actions may be critical and shouldn't be misunderstood. And as they are usually composed by short sentences with no more than a few words, there is no need to use the within-sentence clustering, but only the TF/IDF technique. Additionally, it should be extended with domain orientation using a database collection with documents from a specific domain. This means that, for e-learning applications, the database will have pages of many web learning environments, like the TelEduc (`http://teleduc.nied.unicamp.br/teleduc`), Moodle (`http://moodle.org`), Sakai (`http://sakaiproject.org`), among others. However, to achieve a more generic purpose for this web system interface transformation framework, the database should comprise other domains, but the text reduction process would still have to be domain oriented. In this sense, different dictionary domain files could be generated remotely, each one containing information of term occurrences in each particular domain, and used by the interface adapter according to the web page being summarized.

Attempting to improve the lexical analysis with the semantic benefits, we suggest a brute force stemming approach (process for reducing inflected/derived words to their stem form using a lookup table). For example, the words *teacher*, *teachers*, *professor* and *professors* can all be related to the same stem inside the lookup table. As a result, each one will be considered as the same word by the TF/IDF method. Improvements will contemplate not just the handling of synonymies, but also grammatical inflections: gender, number and case.

The Summary Thumbnail's second problem concerning the conceptual model change when the user needs to move from the thumbnail view to the detailed view and backwards is a more complicated one. This concept doesn't exist on the original application's interface but is fully necessary on the approach. Even considering a fast learning curve for the user to master the concept, the detailed view should be more elaborated than just applying Direct Migration and letting the user deal with extensive vertical and horizontal scrolling. Probably this problem wasn't considered that important because of thinking the user will first scan the thumbnail and, at last, move to the desired detailed view to read the full text. That's probably what he/she will do, but as the whole content is summarized, the need for reading full text here and there must be considered. What's the best way to bring users back to the thumbnail view? Will they get lost on the detailed view, trying to find other full texts? We believe user evaluations should be taken towards finding adequate solutions for these questions and identifying the best approach to smooth the transition between the thumbnail and detailed views. Some ideas could come from combining the approaches given by Gateway and Smartview, inserting their concepts of detailed view adapted to the screen size. The first prototypes developed for this proposed framework follow this direction and, even for simple implementations of the detailed view, good impressions could be perceived by its use. These issues will be addressed on a later subsection describing the prototypes.

3.2 Directions For Applying The Framework

The framework proposed for mobile access to web system interfaces relies upon the Internet client/server architecture: the server software runs on powerful computers to provide services for the client software installed on any Internet enabled device. Although generic client/server architectures are two-tier, many actual application servers store data on a third machine, known as the database server. Although the processing core of this three-tier client/server paradigm is generally attributed to the server side, the complexity transition to the network's edge experienced on the last years has proved this isn't a mandatory rule. In fact, we propose another logic tier on the client side to perform every interface adaptation needed for the web page transformation framework. Figure 3 presents an application example running on this architecture.

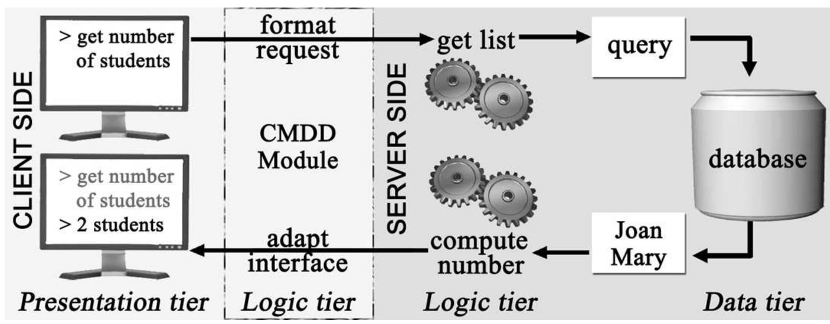

Figure 3: Example of an application running on the framework architecture. The logic tier on the client side delivers the user's requests to the server and adapts its responses. This approach contributes to an easier installation of the CMDD module, better personalization of the user's preferences, higher efficiency on the HTTP requests and good portability.

This enhancement of the presentation tier is more suitable to this proposal because many usability and technology issues will be fulfilled, like the following:

- *Ease of installation and personalization* - if it was the other way around (interface adaptation on the server side), every web server should improve its logic tier by installing the web system interface adapter. Once it is the mobile user's interest, one local installation shall enable the whole web access through his/her handheld. Also, many personal choices can be done easily and safely on the client side (e.g. minimum font size, image cropping, text summarization, etc.);
- *Better efficiency* - The network congestion can be decreased avoiding unnecessary further server requests. For example, when the client tries to see a detailed view from a certain thumbnail region, the adaptations can be done faster on its side, without having to resend request messages to the server;
- *Portability* - Cross-platform solutions for many operational systems (e.g. Microsoft Windows, Linux, AIX, Solaris, MacOS, BSD, HP-UX, OpenVMS) and portable devices can be obtained using, for example, the XPCOM open source technology (`http://`

mozilla.org/projects/xpcom). Besides that, there is no need to concern with server side programming languages as the client presentation tier will always deal with the resulting HTML web page delivered by the server.

3.3 First Impressions With First Prototypes

The web page transformation framework proposed on this work is still on its early stages of implementation. The interface adaptation process doesn't require any additional Internet traffic, but is performed locally by the browser script interpreter. Although the first results indicate an acceptable delay on the interface adaptation, we expect better outcomes by implementing the prototype as part of the web browser, just like the Opera Fit To Screen[1] feature. The hardware used for testing is the HP iPAQ Pocket PC h2400 running Windows Mobile 2003 operating system, but could be any other pocket PC or smartphone with a CSS, DHTML and JavaScript compatible browser, like Opera Mobile, Opera Mini (www.opera.com/products/mobile) or Access NetFront (http://nfppc.access.co.jp/english). Even the outdated model used for testing is able to process the web pages in less than two seconds, which has being considered acceptable for an undisturbed navigation. This is a good indicator that the logic tier doesn't have to be implemented on the server side or on a proxy server to avoid processing delays on computationally weak mobile devices [9, 11].

As mentioned on a previous subsection, the Summary Thumbnail project has the closest ideas to the CMDD proposal for a web system transformation framework, but still has some issues to be considered in order to maintain the same conceptual model. With the first prototype generation, the main focus was on smoothing the transition between the thumbnail and detailed views, leaving the summarizing process aside (the simple right-to-left cropping approach was used). The idea was to provide a faster detailed view right over the thumbnail using the focus-plus-context technique to prevent localization loss with frequent zoom-in and zoom-out. In this sense, the hint concept present on almost every graphical user interface was used to reveal full texts and normal sized images whenever the user points to any object on the page. This approach also removes the interaction fear of pointing to something and wondering if a zoom-in or a link navigation will be performed (when the user points to a hyperlink, the hint presents both the full hyperlink text and a button to visit it). Recapturing the personalization advantage of a logic tier implemented on the client side, the hint approach is used with an additional button so the user may choose if the full text should be on the page. Figure 4 shows a sequence of the prototype screens to give a good idea about the hint detailed view.

[1] The Fit to Screen is a feature provided by the Opera Mobile and Opera Mini web browsers that uses the Smart-Screen Rendering technology to reformat web pages fitting them in the pocket PCs and smartphones' screens. This commercial solution for automatic transformation has received many awards (www.opera.com/products/mobile/reviews/).

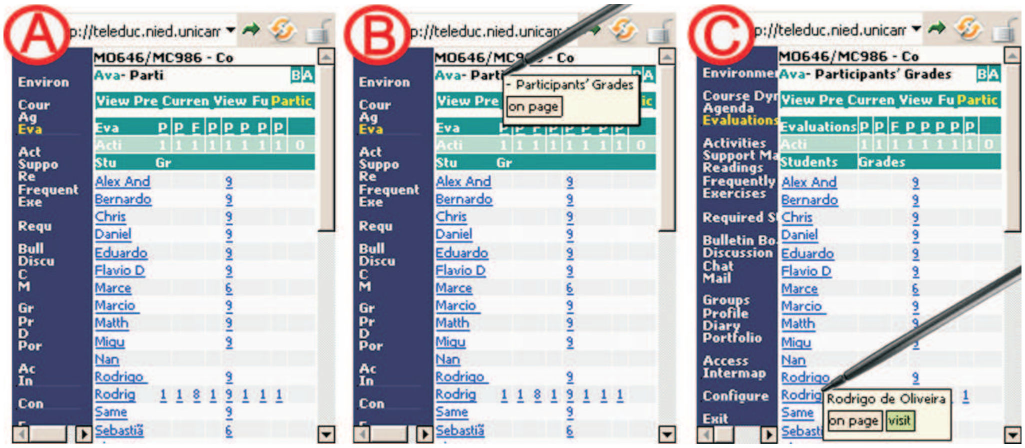

Figure 4: Detailed view on first prototype generation: (a) The TelEduc desktop interface is shrunk, texts are summarized (right-to-left cropping) and fonts are sized up. (b) When the user points to summarized text, detailed view appears over the thumbnail without loosing context (full text can be shown on the thumbnail using the "on page" button and this information is stored for future accesses). (c) If the user points to any navigational structure, an additional button is provided on the detailed view to visit that link.

Although the detailed view presented on Figure 4 preserves layout without strong transitions between the thumbnail and the Direct Migration, it loses formatting attributes that may be useful on systems more iconic than TelEduc. To solve this problem, it was used a mix of the Direct Migration complete context and the hint smoothing feature. Figure 5 presents this approach used on the second prototype generation.

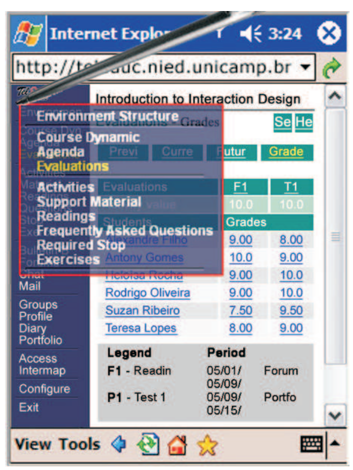

Figure 5: Detailed view on second prototype generation: similar to the focus-plus-context technique used on the Gateway proposal but with a lower opacity level to improve the context visualization. As this feature isn't available on current pocket PC browsers (requires CSS3 compatibility), evaluation studies will probably be conducted using Tablet PC simulations.

Recently, the first prototype generation was presented and informally tested on a few institutions, including the Campinas State University (`www.unicamp.br`), CPqD (`www.cpqd.com.br/usa/`) and a workshop demonstration for the Tidia-Ae project (`http://tidia-ae.incubadora.fapesp.br/portal`). These informal evaluations revealed some good impressions, indicating several users interested on the prototype with a clear majority preference for this proposal instead of the well established commercial solution given by Opera. Figure 6 compares screens generated by both approaches.

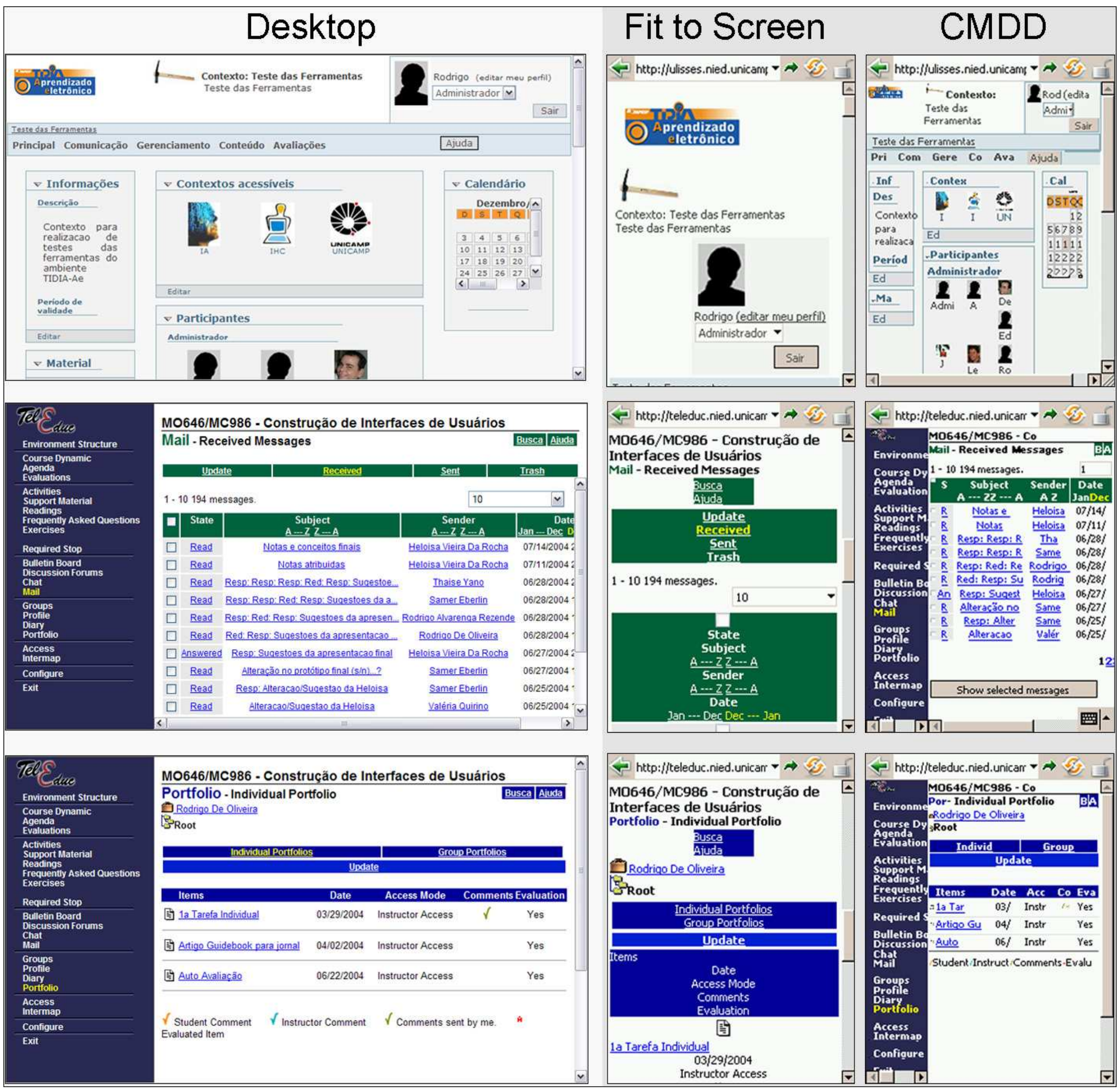

Figure 6: Comparison between the interfaces generated by the framework prototype and the Opera Fit to Screen solution. The informal tests revealed a clear preference for the CMDD approach.

It's important to state that, by no means, the comparisons presented on Figure 6 are used to validate the CMDD ideas, but just to start moving on this direction. As soon as the second prototype generation is ready, user evaluations will be conducted to better identify the pros and cons of this proposal. We plan to follow an experiment protocol similar to the one used by Botherel and Karsenty [3], where the devices alternative use was tested by a different group than the referent devices constant use. This seems to be an interesting approach to evaluate the impacts on usability when the conceptual model is changed.

4 Conclusions

Many arguments were presented using well known HCI concepts on behalf of a user-centered approach for any kind of design, especially for those applications predisposed to multi-device needs. These ideas were used to clarify the Conceptual Multi-Device Design proposed on a previous work [15], improving its theoretical foundations and providing a healthy discussion concerning the pros and cons of user and device oriented designs. Also, the web system interface transformation framework [16] was enhanced with better directions for applying the text reduction and switching between thumbnail and detailed views. The first prototypes revealed some prosper impressions with a simpler detailed view for full text presentation. Next prototypes are improving this visualization with the Direct Migration approach inside the hint detailed view, which shall prevent the user from loosing the context. User evaluations will be taken sooner with the second prototype generation to investigate the advantages of CMDD over design approaches that change the application's conceptual model on each access media.

References

[1] Andrade, A. R.; Munson, E. V.; Pimentel, M. G. C. Engineering web applications with xml and xslt. In Proc. LA-Web 2004, IEEE CS Press, Ribeirão Preto, SP, pp. 86-93.

[2] Berti, S.; Correani, F.; Mori, G.; Paternò, F.; Santoro, C. Teresa: a transformation-based environment for designing and developing multi-device interfaces. In Proceedings of the CHI 2004 Extended Abstracts, pp. 793-794, Vienna, Austria, April 2004.

[3] Botherel, V.; Karsenty, L. How inconsistencies between speech and graphical user interfaces may be detrimental... or desirable. In Proceedings of 10th Conference on Speech and Computers, SPECOM'05, Patras, Greece, 2005.

[4] Buyukkokten, O.; Kaljuvee, O; Garcia-Molina, H.; Paepcke, A.; Winograd, T. Efficient web browsing on handheld devices using page and form summarization. In ACM Transactions on Information Systems (TOIS). New York, USA, v. 20, 2002. pp. 82-115.

[5] Coninx, K.; Luyten, K.; Vandervelpen, C.; Bergh, J. V. D.; Creemers, B. Dygimes: dynamically generating interfaces for mobile computing devices and embedded systems. Proc. MHCI 2003, Udine, Italy, pp. 256-270.

[6] Craik, K. J. W. The nature of explanation. Cambridge University Press, 1943.

[7] Furnas, G. W.; Deerwester, S.; Dumais, S. T.; Landauer T. K., Harshman, R. A.; Streeter, L. A.; Lochbaum, K. E. Information retrieval using a singular value decomposition model of latent semantic structure. In Proc. 11th International Conference on Research & Development in Information Retrieval, 1988, pp. 465-480.

[8] Hinz, M. Fiala, Z. Wehner, F. Personalization-based optimization of web interfaces for mobile devices. In Proc. MHCI 2004, Glasgow, Scotland, pp. 204-215.

[9] Lam, H.; Baudisch, P. Summary thumbnails: readable overviews for small screen web browsers. In Proceedings of CHI 2005, Portland, OR, pp. 681-690.

[10] Macedo, A. A.; Pimentel, M. G. C.; Camacho-Guerrero, J. A. Latent semantic linking over homogeneous repositories. In Proc. ACM Symposium on Document Engineering, Atlanta, USA, 2001.

[11] Mackay, B.; Watters, C.; Duffy, J. Web page transformation when switching devices. In Proc. of the Mobile HCI 2004, Glasgow, Scotland, pp. 228-239.

[12] Milic-Frayling, N.; Sommerer, R. SmartView: enhanced document viewer for mobile devices. MSR Technical Report, 2002.

[13] MSDN Localization Testing. Available at: http://msdn2.microsoft.com/en-gb/library/aa292138(VS.71).aspx. Accessed on December 2006.

[14] Norman, D. The design of everyday things. Basic Books, New York, 1988.

[15] Oliveira, R.; Rocha, H. V. Towards an approach for multi-device interface design. Proc. Webmedia 2005, Poços de Caldas, Brazil, vol. 125. ACM Press, 1-3.

[16] Oliveira, R.; Rocha, H. Mobile Access to Web Systems Using a Multi-Device Interface Design Approach. In Proceedings of the 2006 International Conference on Pervasive Systems and Computing (PSC'06). Las Vegas, USA: CSREA Press, 2006. pp. 37-43.

[17] Po, S.; Howard, S.; Vetere, F.; Skov, M. B. Heuristic Evaluation and Mobile Usability: Bridging the Realism Gap. Proc. MHCI 2004, Glasgow, Scotland, pp. 49-60.

[18] Preece, J.; Rogers, Y.; Sharp, H. Interaction design: beyond human-computer interaction. Wiley & Sons, 2002.

[19] Pyla, P. S.; Tungare, M., Pérez-Quiñones, M. A. Multiple user interfaces: why consistency is not everything, and seamless task migration is key. In Proceedings of CHI 2006 Workshop on The Many Faces of Consistency in Cross-Platform Design, 2006.

[20] Rossi, G.; Schwabe, D.; Lyardet, F. Web application models are more than conceptual models. In Proc. of ER'99, Springer, Paris, France, 1999, pp. 239-252.