Coping with Consistency under Multiple Design Constraints: The Case of the Nokia 9000 WWW Browser

Heli Hjelmeroos
Nokia Telecommunications Oy
P.O. Box 735
FIN-33101 Tampere
Finland
+358-3-2574721
Heli.Hjelmeroos@ntc.nokia.com

Pekka Ketola
Nokia Mobile Phones
P.O. Box 68
FIN-33721 Tampere
Finland
+358-10-5056827
Pekka.Ketola@nmp.nokia.com

Kari-Jouko Räihä
University of Tampere
Department of Computer Science
P.O. Box 607
FIN-33101 Tampere
Finland
+358-3-2156952
kjr@cs.uta.fi

ABSTRACT
Consistency is a commonly accepted but sometimes problematic design goal. External and internal consistency may conflict, and sometimes the best solution is inconsistent in both respects. We describe user interface design issues and several usability studies for the Nokia 9000 Communicator WWW browser and for WWW pages optimized for the browser. The results show how within the same, restricted design domain, various forms of consistency have to be favored over others in solving various design problems.

Keywords
Consistency, design constraints, WWW browser, web page design, PDA, mobile phone

INTRODUCTION
Graphical browsers for the World-Wide Web (WWW) have been available for only about six years. In that relatively short time, they have become one of the most common tools of information workers. Browsing the web can take several hours of a work day, and browsing habits and patterns are already deeply rooted in the way we carry out our daily work.

WWW browsers are typically used in personal computers. However, new implementations for mobile devices are rapidly becoming available, for instance in PDA devices [11]. Designing a WWW browser for such a platform is faced with many problems caused by the small size of the device, the low data transfer rate, small memory, and limited processing capacity.

Although people are highly adaptive to new platforms, their previous experiences are, nevertheless, critical in shaping their expectations [7]. A web browser on a new platform should behave in a reasonably similar manner as browsers in other environments, to achieve *external consistency*. On the other hand, few devices are built for a single application only; the browser should also work in harmony with the other applications to guarantee *internal consistency*. The designer is often faced with a difficult design issue: how to weigh the different factors, or design constraints, that point in different directions.

It is precisely for this reason why consistency is a somewhat controversial design principle. While it is generally agreed that consistency is a worthwhile goal to strive for [8, 9], many authors [1, 2, 7] warn about the pitfalls of taking only one aspect of consistency as the guiding force in design. The overall goal, of course, should be to produce a usable product.

Nokia 9000 Communicator is a mobile device that incorporates a WWW browser. In this paper we discuss several issues that were encountered and had to be solved during the design process. We often had to weigh the relative importance of external consistency, internal consistency, and customized solutions for difficult problems. Most decisions had to be made without the possibility for a usability test, because the schedule was extremely tight and the project highly confidential to retain the first mover advantage.

For each design issue we explain the reasoning that led to the chosen solution. We then discuss whether the solution turned out to be successful in various evaluations and tests that were carried out after the release of the product.

It turned out that some solutions worked and some caused unexpected problems. Sometimes opting for consistency would have been the right choice, sometimes not. Even well justified design decisions proved wrong because it was difficult to foresee how deeply rooted some browsing patterns already were. Thus we expect our story to be useful for developers of new mobile browsers: they should benefit by becoming aware of some pitfalls to avoid.

To make things even more complicated, designing and studying the browser is not enough: its usability depends on the content that is browsed. We designed and tested several versions of the same web site, optimized for different browsing platforms. Here we have yet another dimension of consistency: whether the site should consistently look the same on all platforms, or whether it should be changed to make use of the idiosyncrasies of a particular environment (e.g., to avoid problems caused by small screen size or low data transfer rate). Some findings on this issue will also be discussed.

NOKIA 9000 COMMUNICATOR

The physical device

Nokia 9000 Communicator (Figure 1) was released in March 1996 [5, 6]. It is a portable device that unifies the functionalities of a mobile phone and a computer.

Figure 1. Nokia 9000 Communicator.

The Nokia 9000 device has two user interfaces. Figure 1 shows it in a state where it is opened up; this interface is called the Communicator. When the cover is closed, a normal mobile phone interface (located on the back side of the cover) can be used.

The physical interface of the Communicator contains a graphical grayscale LCD display. The size of the screen is 640*200 pixels (120*38 mm) and the working area for applications is 540*200 pixels. To the left of the screen there are two physical buttons for scrolling the view. To the right there are four physical command buttons. The meaning of each command button is shown in the rightmost column of the screen.

The device also contains a small but fairly complete keyboard. Keys for accessing the applications are located in the top row of the keyboard (thus replacing ordinary function keys). Arrow keys can be seen in the lower right hand corner of the keyboard. No pointing device (neither a stylus nor a trackball) is available.

The applications

The principal task of the Nokia 9000 is to communicate with the environment. All other features are designed to support this task. Technically, several communication methods are supported, including a voice call, data call, fax call and messaging (SMS).

The primary application of the Nokia 9000 is the mobile phone. However, in addition to the phone it contains several other applications, all in one physical package. The applications include an address book, a note book, a calendar, a fax viewer and composer, an e-mail client, and a WWW browser. In this paper we focus on the design of the browser, referred to as the N9000 Browser, and on the design of web pages to be viewed by the browser.

DESIGN AND EVALUATION PROCESS

The N9000 Browser

The mobile phone market is highly competitive. At the time of its release, Nokia 9000 was a revolutionary product. It took years before other products with matching functionality became available.

To develop such products, exceptional security measures need to be taken. Most of the work was done inside the company by the development team. When outside evaluators were used, strict nondisclosure policies were applied. This fact alone had the consequence that extensive usability evaluation in the traditional sense simply could not be done.

Another reason for only limited pre-release evaluation was the fact that Nokia 9000 is an embedded system. Because of design and implementation restrictions, it was not possible to release versions for beta testing.

Thus, although the design went through several iterations with paper mock-ups, simulations, and prototypes of increasing sophistication, the evaluations at that phase were still informal, involved a small group of people, and are not available to be discussed here. Furthermore, most of the studies focused on the general user interface of the Communicator, not particularly on the N9000 Browser.

In the following, we concentrate on the released design. To evaluate how successful the design decisions were, two rounds of evaluation have been carried out under different circumstances and for different user types. Both studies included participants from several European countries.

The first study was carried out by different persons non-systematically in 1996. These can be called friendly-user tests because all participants were eager to see what possibilities the Communicator has to offer. The observations were made in normal use situations.

Users were asked to mention positive and negative impressions about the Communicator after having used the device for some weeks. The answers were recorded using videotapes and questionnaires. The participants in this study were novice Communicator users. The study focused on finding the questions that users have when they start using the Communicator and on the concepts and functions that are not easy to learn.

The first study was used as a guideline for the second, more systematic study, which was carried out among critical Communicator error hunters and developers of third-party software. A questionnaire was sent at the beginning of 1997 via e-mail to the readers of two mailing lists. The goal of the second study was to find out what topics were causing usability problems in the N9000 Browser and also to find out if usage rate affects the acceptance of the system. We also wanted to see specifically whether problematic design decisions had caused usability problems. 53 persons took part in this study. Their experience with the N9000 Browser varied from 0 to 12 months. The frequency of use varied, too, between daily use and no use at all.

Optimized web pages

The web browsing experience depends not only on the user interface of the N9000 Browser, but also on the contents of the web pages that are browsed. To study the effect of web page design, we implemented three versions of a web site that was assumed to be particularly useful for owners of the Communicator: Club Nokia. It is an on-line magazine targeted at mobile people. The early Club Nokia had five sections: Contest, Backroom, Magazine, Support, and Locator. Contest and Backroom provided visual attraction and a place for technical experiments. Magazine contained short stories about, e.g., the wireless lifestyle. The Support section contained product information and support for Communicator users. Finally, Locator was a collection of links for mobile users.

In our study we created three different versions of the web site with almost the same content. One version was for PC users with a fast connection and normal size display, one for laptop users with a slower connection but a reasonable sized display and one for Communicator users with a slow connection and a small display.

Information about the usefulness of the optimized pages was collected in two ways. First, users of the N9000 Browser were interviewed to find out their browsing habits. In addition, a semi-formal usability test was carried out in a usability laboratory with 14 participants. The majority of the test users were partly mobile professionals, 25-30 years old, and familiar with the Communicator.

Although the focus of the test was in browsing using the Communicator, each test session also contained a part where the users browsed the normal pages of the web site using a PC. This made it possible to compare the browsing behavior of the same users under different conditions.

DESIGN CONSTRAINTS

The Nokia 9000 is a collection of several applications. The killer application of the Communicator is e-mail. The N9000 Browser was assumed to appeal to a certain fraction of the market, but it was not envisioned as a heavy duty application. The figures on use frequency given above confirm the validity of this assumption. The situation is rapidly changing, though, as mobile Internet usage becomes more common [11].

There is an in-house style guide that the application interfaces are assumed to follow. Because of the role of the N9000 Browser, it did not have a large impact on the original style guide. Naming and navigation principles that worked well with applications that had less functionality could cause difficult problems for the N9000 Browser, as we shall see. Occasionally it would have been tempting to opt for other solutions and abandon internal consistency, but this was not possible: the style guide had to be followed.

External consistency, i.e., providing functionality and behavior that was familiar from PC-based browsers, would clearly have been desirable. This was difficult, though, because of the physical constraints of the device. Furthermore, the fact that the operating system of the Communicator is not in common use on the PC platform necessarily makes it difficult to achieve this goal.

WWW actions without a pointing device

The Communicator does not have a pointing device. This causes some design problems for the browser. A selection method for both text and graphics has to be implemented. Hypertext navigation in the N9000 Browser is selecting rather than pointing.

Link selection is implemented in the following way. The user selects a link in the document by using the scroll buttons. Scroll up moves the focus to the previous link, or scrolls the document up if the previous link is not visible. Scroll down moves the focus to next link or scrolls the document down. If the next or previous link is not in the visible area, it is not selected until it is scrolled to the visible area.

While this is simple, it clearly created a problem: how to distinguish between normal scrolling of the page and scrolling from link to link.

Data transfer rate

The data transfer rate in cellular connection is currently limited to 9600 bps which is significantly lower than the rate used in fixed network connections. In wireless devices also establishing the connection is slow (from a few seconds to half a minute) – in fact, from 10 to 20 times as slow as in wired networks.

Even though the majority of web users is used to low data transfer rates [3, 4], the time needed for establishing the connection is a new phenomenon particular to the mobile environment. Since staying connected is costly, the user has the option to close the connection while s/he works locally. Thus, there may be a need to establish the connection several times during a working session.

Moreover, the users of the Communicator are not average web users. They are used to high speed connections in the office, and the change from megabytes to kilobytes is dramatic. In fact, it creates a situation where two forms of external consistency compete with each other: whether the pages should look the same on various platforms, or whether they should load with roughly the same speed on various platforms. Both cannot be achieved simultaneously.

DESIGN ISSUES AND SOLUTIONS

We now turn to individual design problems and discuss how the constraints described above were taken into consideration in each case.

Where to start?

When normal web browsers are launched, they open a default document or “homepage”. External consistency would require similar behavior from the N9000 Browser.

However, because of the nature of the wireless environment it was decided that the default state of the browser is off-line (no connection). Forcing the user to wait for half a minute or more before seeing anything

useful was not a tempting design solution. Therefore we abandoned the approach where a default document is always fetched. Instead, a Hotlist view containing a user customizable list of bookmarks is shown. This is a natural view because it offers a clear starting point without a need to establish a connection. Moreover, the Hotlist as the main view increases internal consistency, since other Communicator applications also launch with various list views, typically the list of contacts.

This design decision was estimated to be a possible usability problem because it violates external consistency. This turned out not to be the case: all evaluations indicated that the Hotlist as a starting point is easily understood and accepted.

Providing the necessary functionality

In well designed web sites, much of the time users just click-click-scroll-click.However, the browser provides a wealth of operations that may not be used quite so often but that still are essential for serious use of the web. These include the history view, ability to save documents on disk, ability to enter the URL using the keyboard, support for multiple browser windows, etc.

In the N9000 Browser, some of the functionality (such as multiple windows) can safely be ignored, since it would not be useful given the small size of the screen. However, there are still far too many operations that need to be supported to make them fit on the screen at once.

In the Communicator in general, and in the N9000 Browser in particular, functions are placed in separate views, not for example in menus. Recall that there are only four command buttons available in the Communicator, and that their meaning changes from view to view. Moreover, the style guide reserves (in most cases) the lowest button for returning to the previous level in the view hierarchy. Thus there are, in essence, at most three buttons available for moving down and one button for moving up in the view hierarchy. Some views may provide a combination of operations and movement buttons, thereby reducing the number of buttons available for downward movement.

It is obvious, then, that users of the N9000 Browser will have to navigate more than the users of conventional browsers to reach all desired operations. What operations should be available with each view, and how should the views be arranged?

We decided to distribute the functionality on three levels, each level containing a set of views. Figure 2 shows the three levels as a simplified state diagram. The grouping of states is based on the services that each state offers and on the services needed in the state. For instance, when the user starts the browser, s/he can either edit the Hotlist, fetch a document or change the browser settings. In this view it is not necessary to use the saving functionality.

Our idea is that when the user moves from layer 2 to layer 3, s/he makes a logical transition from working locally within the device to being connected to the Internet. The transition is reflected in the functionality offered by the command buttons. When the user is in the “within device” state s/he can use local operations, such as saving pages locally, and when s/he is in the “connected” state, the navigation tools are available.

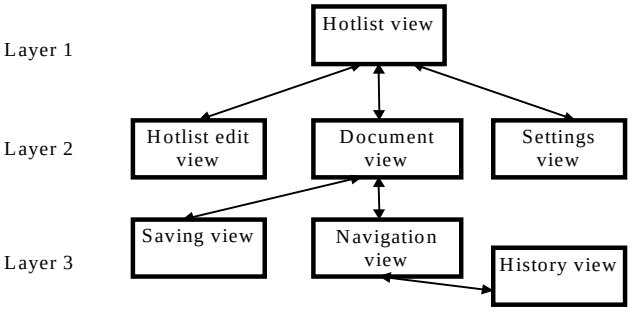

Figure 2. Layering the functionality.

Typically, browsing starts by selecting a link from the Hotlist. There is a clear state transition from the Hotlist view to the Document view (Figure 3). This view appears on layer 2 in Figure 2.

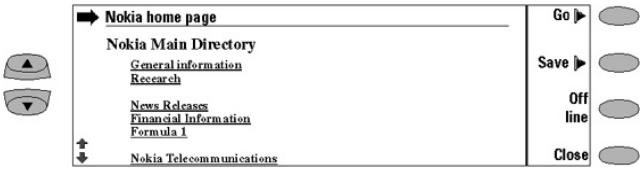

Figure 3. The Document view.

The design of this view was based on the idea that a document has two logical functions: it is a document that the user wants to read and it is also a navigation platform. The command buttons in the Document view are selected to support the reading task. This view contains buttons for saving the document, for closing the connection for the reading period in order to save expensive on-line time, and for closing the document. Consequently, there are two buttons for moving down, one command button, and one button for moving up in the view hierarchy.

The other function of a document, navigation, is supported in the Navigation view (Figure 4). This view appears on layer 3 in Figure 2. The view contains tools for using the hypertext functionality in the document and for fetching documents from the history list. The state transition from the Document view is indicated by changing the command button names and by activating a link in the document.

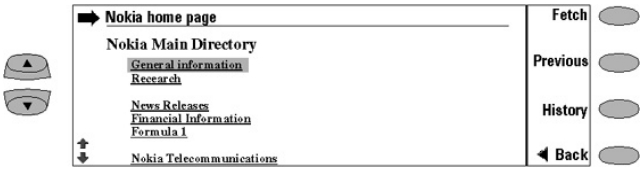

Figure 4. The Navigation view.

Although the layering of the functionality was considered carefully and the result is logical, the separate reading and navigation states of the document were still expected to be the hardest part for the user to understand in this browser. After all, no other implementation uses the same logic. User studies confirmed this expectation.

The *concept* of distributing the browser functionality on many levels did not cause major problems to the users. It was easily understood and accepted. However, the use of separate reading and navigation modes had the effect that it was not always clear what operations could be executed in the different states. It was difficult for the users to understand that the modes were separate and that there are different commands available in these views.

A common confusion situation was met when the user had fetched a document from the Hotlist view. After the operation s/he wanted to start to navigate immediately without extra mode changes. Figure 5 shows the state transitions that are needed to activate the navigation mode from the Hotlist view.

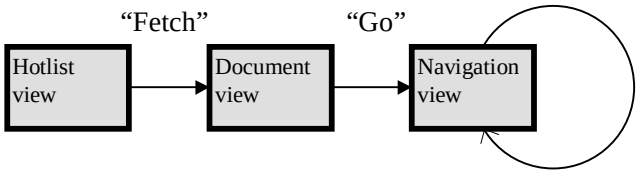

Figure 5. State transitions from the Hotlist to the navigation mode.

The conceptual model that the users typically expected was different: more straightforward and more natural (Figure 6). In this view, only one state transition would be needed to enable the navigation tools. In this case our design model is too complex.

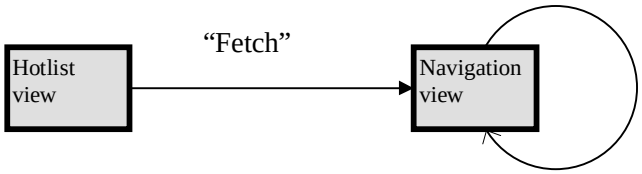

Figure 6. The functionality expected by the users.

In retrospect, the logical and balanced layering of the functionality should not have been the key design principle. Instead, we should have identified the most common operations and grouped them together. This, however, would require some other solution to the problem of separating page scrolling from hopping from link to link.

The Back button

One difficult design problem is visible in the Navigation view in Figure 4. As explained earlier, the lowermost command button is used for moving upwards in the view hierarchy. The style guide requires that this button is called "Back" – a perfectly reasonable choice for all other purposes but *not* for the N9000 Browser. How should the common Back operation be made available?

The solution was to use a different name, "Previous", for the command that moves back to the previous page in the history list. This was a decision that was forced by design constraints that enforced internal consistency. We expected problems with external consistency, and indeed, were not let down. Even after users realize the meaning of the two buttons, they still consistently struggle with them. This is not surprising considering the fact that the Back operation is the most commonly used single navigation mechanism in normal web browsing [10].

Optimization of the web pages for the mobile browser

When web sites are optimized for mobile use, at least the following issues have to be considered:

1. content of the optimized site (is all material equally relevant, which pictures are important enough) – important to optimize download times;
2. page layout – important to make use of the small screen area; and
3. navigation (how to structure the site, how to create descriptive links) – important to minimize the number of file transfers.

Choosing the content

We made the choice based on interviews of the users. Travel information seemed to be the most interesting section to the users of Club Nokia. Information about the Communicator and product support were also expected to be interesting. These parts of the site were designed with special care to fit the N9000 Browser.

Again, it turned out that user behavior is difficult to predict. In spite of the possibility to use the web while travelling, users said that they would like to look for travel information mainly *before* the trip. On the other hand, if users had a need for a specific information and they knew in advance where to find it using the WWW, they were likely to use the Communicator for browsing the web also while travelling.

Contrary to what we expected, users were quite interested also in entertaining stories. They said that short, amusing tales would be adequate to read for example when waiting at an airport lounge.

Page layout

Communicator test users did not complain about the size of the N9000 Browser window. They did not need graphics or animation – they wanted to get to the text information fast. When comparing the optimized pages with the graphical, large display site of Club Nokia, users often said that it is almost more comfortable to read text from the Communicator's screen.

Links and navigation

There is no external pointing device in Communicator to move freely within page and links. Links and pages should be designed so that users can move easily with the browser's command buttons (next link, previous link). Introducing the content, navigation support and descriptive text links should be designed to fit into the communicator's display.

Although users could wait to get to the information they needed, predictability of the links became very important. Especially when the data transfer rate is low, it is essential to use descriptive links to avoid the frustration of down-

loading useless pages. More research on link naming policies is clearly needed.

The most often used way to navigate was to use the navigation tools of the N9000 Browser (the Previous command and the History view). The reason is probably that the command button is always there, consistently and independently of individual page design. A further advantage is that it is conveniently near the thumb. In addition, most of our pages in the tests were more than one screenful long, so that the navigation links within the pages were not constantly accessible.

DISCUSSION

In the second user study that was carried out for the N9000 Browser, the users were asked the question "*What was the major problem for you?"* We tried to identify problems that are bothering users after the initial difficulties and during the daily use. Most problems concerned the low data transfer rate (19%) and connectivity problems (21%). These problems are typically caused by slow network systems and not the device. The second group of problems contained the general difficulty of understanding the user interface (13%), complexity of settings (13%) and software bugs (13%).

Contrary to what we assumed, the two modes of viewing the document caused a relatively small number of complaints (8%). This indicates that in the long run, users are willing to learn new patterns of behavior, but problems that are out of their control (low data transfer rate) are frustrating.

Studies of the web pages optimized for the N9000 Browser showed that visual or even structural external consistency of the pages was not important to the users – making good use of the platform was much more crucial.

CONCLUSIONS

The cases discussed above contain (a) a situation where internal consistency and device specific solutions were the right design choice (the start-up view), (b) a situation where careful design under hard physical constraints produced a solution with usability problems (layering the functionality), and (c) a case where internal consistency drove to a design that caused confusion in the users (the Back button). In each case, the success or problem rate of the chosen solution was hard to predict accurately in advance by logical reasoning.

The evaluations showed that the layering of the document functionality causes problems in the beginning of use but not in the long run. Speed and connectivity problems were not usability problems in the beginning but they are problems in the long run. Table 1 lists what usability problems were anticipated during the design process and what problems the users reported. The severity is indicated by the number of plus signs: two plus signs are a serious problem.

The results indicate that when an application is designed in a way that differs from a generally accepted and used style, users may have difficulties in adopting to the new functionality, especially in the learning phase. Users don't easily accept differences to systems they have already learned.

Problem	*Expected*	*Study 1*	*Study 2*
Connection establishment time		++	+++
Data transfer rate	++	++	+++
Connectivity problem	+	+	+
Layered functionality	++	+++	+++
External consistency	+	+	+
Settings		+++	++

Table 1. Expected and found problems.

With the increased use of PDAs, wearable computers and other smart products, the need for mobile web browsers is growing fast, and we hope that our experiences are useful in their development. Future versions of the N9000 browser have already solved differently many of the problems discussed here.

REFERENCES

1. Grudin, J. The case against user interface consistency. *Comm. ACM 3*2, 10 (October 1989), 1164-1173.
2. Grudin, J. Consistency, standards, and formal approaches to interface development and evaluation: A note on Wiecha, Bennett, Boies, Gould, and Greene. *ACM Transactions on Information Systems 10*, 1 (January 1992), 103-111.
3. GVU Center's WWW User Surveys. Available at http://www.cc.gatech.edu/gvu/user_surveys/.
4. Nielsen, J. The Need for Speed. Jakob Nielsen's Alertbox for March 1, 1997. Available at http://www.useit.com/alertbox/9703a.html.
5. Nokia's innovations offer a new dimension to mobile communication. Press release, March 1996. Available at http://www.nokia.com/news/news_htmls/nmp_960313a.html.
6. NOKIA 9000 Communicator info page, 1998. Available at http://www.nokia.com/com9000/n9000.html.
7. Norman, Donald A. *The Design of Everyday Thing*s. Basic Books, 1988.
8. Preece, J. *et al. Human-Computer Interaction.* Addison-Wesley, 1994.
9. Shneiderman, B. *Designing the User Interface: Strategies for Effective Human-Computer Interaction (Third Edition).* Addison-Wesley, 1998.
10. Spool, J. M. *et al. Web Site Usability: A Designer's Guide*. User Interface Engineering, 1997.
11. W3C: Mobile Access web page. Available at http://www.w3.org/Mobile/.