

Design of User Interfaces for the Web

Alan Dix

aQtive limited
Birmingham Research Park
Vincent Drive, Birmingham, B15 2SQ, UK

and

Staffordshire University
School of Computing
Beaconside, Stafford, ST21 6JU, UK

alan@hiraeth.com
<http://www.hiraeth.com/alan/topics/web/>

Abstract

This paper examines several facets of user interface design for the world-wide web. First it considers the use of the web as a platform for prototyping or deploying user interface. Second it looks at guidelines and issues for the design of web sites and the nature of the web medium. Third it examines the way in which interfaces can help users to manage the complexity of the web including history mechanisms and diagrammatic overviews. Leading directly from the last of these is a discussion of the multiple geometries of the web given by links, content and people's browsing. Finally the paper looks at the future of the Internet as an integrative phenomenon. This includes a description of aQtive onCue, an application that integrates Internet services, desktop applications and the user's current work. The long-term future is PopuNET, the network for everyone, everywhere and everywhen, requiring a yet more radical approach.

Keywords. world-wide web, user interfaces, browsers, design guidelines, visualisation, hypertext history

1. Introduction

The disadvantage of an invited paper is that one ends up with a title that it is so broad it would take a book to do justice to. The advantage is that one can follow one's interests, and possibly prejudices, without an eye to the reviewer! One either ends up with a boring survey or a polemic.

I'll try to do a bit of both: give an overview of user interfaces and the web, but also link this to some aspects that I've been particularly involved with and some of my own view to the future.

I'll initially look at three aspects of design for the web now:

- using the web as a tool for creating user interfaces
- designing usable pages and sites

- managing the complexity of the web as a whole

The latter leads on to looking at the large-scale geometric structure of the web and cyberspace (a subject for a book in itself!). Finally I'll look ahead at the future of the web and find, perhaps not surprisingly, that central to this vision are the products I have personally been involved in the designing of!

Please look at the web page (above) for links to URLs in this paper and other related links and information.

2. Using the web as a user interface

2.1 Catching the dream

As a user-interface researcher, one of the frustrations has always been both the complexity of building user interfaces and the lack of portability when they are done. With a commercial system one is often building for a specific platform and the goal is the production of the entire system. For UI research, the goal is usually the demonstration of some small technique or widget and the rest of the system is just noise – but essential noise. If the surrounding system is too bad, people can't see the novel feature, just the irrelevant problems; if the surrounding system is not substantial it is often hard to test. Furthermore, having produced a prototype system to test one then finds that not all the world uses X-windows, HyperCard etc., and that one's lovely demo is not quite so useful after all.

Joy of joys when the web changed this! At last it is possible to create relatively large test beds where the broad interface is understood and one can focus on the critical novel aspects. Then, when one has demonstrated the point, demos can be made available to everyone! Not surprisingly, a vast proportion of prototype UI research systems are now built either using the web or using Java.

These advantages for the researcher are of course equally true for large-scale commercial use. Even within

organisations hardware/software platforms are not uniform and the cross-platform, ubiquitous nature of the web means it has become the way to deliver interfaces both within organisational Intranets and to end-users over the Internet. Examples of the latter include many on-line banking systems, UK rail information, on-line currency conversion, and, of course, ever growing numbers of e-commerce sites such as Amazon [1].

Of course, anyone who has built such systems will know that a lot of the hype is somewhat overblown. As soon as you attempt to deploy 'rich' web interfaces (detailed layout, DHTML, JavaScript) you find that the interpretation of the technology is far from standard – even Java!

Furthermore, as soon as one uses 'advanced' features, from rollovers to Flash or Shockwave animations [2], one cuts out large numbers of potential users who haven't got the latest browsers and plug-ins.

Rapid user interface construction can rapidly degenerate into a random search to find a work-around for yet another browser bug. The silver bullet has a heart of lead.

Nevertheless, the web is clearly the platform of choice for so many applications, despite its flaws. This is evident in many papers and web and HCI conferences as well as the response to special events and publications such as the 'CSCW and the Web' special editions of CSCW Journal (now in book form [3]) ; the special issue of IJHCS on WWW Usability [4]; and the day conference on the 'Active Web' [5] that I recently co-organised.

2.2 Architectures

Software architecture has always been a concern of user interface researchers, but the distributed nature of the web and limited local functionality increase the importance of effective architectural design.

The central question of software architecture, especially on a distributed platform, is 'what goes where'. In user interface terms the 'what' question has well rehearsed answers. Going back some time, the Seeheim model [6] has three main components: the presentation or display management (which determines the appearance and low-level behaviour of the interface), the dialogue control (which determines the order of interaction) and a link to the underlying data structures and semantics. This layered presentation/semantics distinction is evident in other abstract architectures and implementation frameworks such as MVC [7], PAC [8] and Arch/Slinky [9].

Historically, these distinctions have been largely aimed at program modularisation for maintenance and re-use. However, for distributed systems, like the web,

how these conceptual units are allocated makes an enormous difference to performance.

On the web we have a number of places that computation can occur. The most obvious are the user's browser or web client (e.g. forms, Javascript, Java) and the web server (for example CGI scripts). However, this is not simply a matter of client-server design. The nature of web technology puts string limits on what is possible or desirable, for example the security limitations of applets. In addition, many web systems now use database backends – another site for computation. For an overview of these as sites for interaction see my *Interfaces* articles on Active Web technology [10].

Finally (although that is probably a foolish word to use when talking about the web) several systems are now making use of computation at proxy servers. Instead of just using these for caching visited pages, proxies or other intermediaries are now being used for a variety of purposes, for example to filter and transform pages for display on small portable devices [11] or to counter transmission protocol problems [12].

The way an application is split over these sites affects resources and efficiency, but most important it affects the rate of feedback for a user. There are neurologically fixed targets for this – for some kinds of hand-eye coordination feedback must come within 100-200 ms or our ability to control collapses, for lip-sound synchronisation even tighter synchronisation is needed. Also, there are well documented limits on how long people are prepared to wait (5-10 seconds) for 'bigger' actions without feeling frustrated [13,14].

In collaborative systems, there is the additional issue that there may be many views of the same shared data, with some shared control (e.g. locking) and some individual interaction. This inevitably leads to more complex decisions. Also collaborative systems have to consider not only feedback (seeing the effect of one's own actions) but also feedthrough (how long it takes to see the effects of other users' actions) [15]. To some extent feedthrough is intractable – other users are physically remote and communication will take time. However, when users are physically close together, their interactions are often still mediated by distant servers. Proxies and similar entities (such as routers) offer an excellent opportunity for 'local' shared computation [16].

2.3 Large data sets?

Applications can often be split into some parts that need rapid interaction and other parts that require heavy back-end data manipulation. Placement is easy, interaction-rich parts are executed close to the user, data-intensive parts executed close to the data (on a server).

However, many recent interfaces for exploration of large data sets depend on rapid user interaction with representations of (apparently) the entire data set. Examples include various systems from Maryland, such as Starfield displays [17], Xerox work including Cone-trees [18], and Hyperbolic trees [19].



Figure 1. Hyperbolic Tree – Inxight Software

These systems are often implemented on high-end workstations with a complete data set stored locally. For the web (and indeed for very large data sets), this is not possible, but neither is the normal split – both interaction and data are required together!

Obviously enough of the data has to be downloaded to enable local interaction, but here 'enough' does not mean a subset of the data – these systems gain their power by allowing the user to see some representation of the *whole* data set. Instead, 'enough' means some abstraction of the data (index, aggregate, etc.) that is sufficiently detailed to allow interaction. Note that this need not be an entirely accurate abstraction, perhaps simply clusters of the original data with averages of attributes. As the user drills in to see more detail new aggregates can be downloaded. The goal is to produce representations that the user can use to examine the data, not 100% accuracy. This seems an obvious lesson, but it is one it took the graphics community many years to learn.

3. Designing usable webs

Now let's take a look at designing for the web itself. There are many sets of web design guidelines, design case studies and studies of web users and use. Although, I would not agree with every guideline, the most comprehensive by far are the Yale guidelines [20]. For other similar resources see the SIGCHI HCI and the Web pages [21] and the web page for this paper.

3.1 Design issues

To some extent designing a web is just like designing any other computer system. Consider any set of user-interface design guidelines or principles (e.g. consistent navigation structure, icon design, use of metaphor, learnability) and they apply equally to the web. However, the web poses UI design problems of its own.

Who are the users? The most widely preached and important UI design principle is to understand who your users are and what they want to do. With the web there are so many users with so many different purposes. Typically they all hit the same home page. Think of a university department's web site. There will be potential students: post-18, mature students, part-time, full-time. There may be commercial users looking for consultancy. There may be job applicants checking the department's research and teaching portfolio. The list continues. In fact, it is not quite as bad as it seems – often it is possible to identify the most significant user group and/or design the site to funnel different types of user to different areas, but it is certainly a challenge!

Who is this user? The transactional nature of HTTP means that it is hard to even know where a particular user has been before or what they've done before on your site. One of the clearest examples of this is when content changes. Typically sites change at the leaves of the site, but people enter at the root. Repeated visits give the same content – it is not surprising that few sites are revisited! A traditional information system can accommodate this by highlighting areas that have changed since a user has last seen them, but this only works because the system knows who the user is! This is partly a technological issue – there are many means of authentication and identification (including cookies). But a combination of technological limitations and (understandable) user worries about privacy mean that there is currently no reliable means to tailor web interfaces or sites for individual users.

Where is the start? The programmer usually has the ultimate say on where a user enters his/her program and, barring crashes, where they leave. With a web site we have no such control! Many web designers naively assume that people will start at the home page and drill down from there. In reality people will bookmark pages in the middle of a site, or even worse enter a site for the first time from a link and find themselves at an internal page. Just imagine if someone were able to freeze your program half way through executing, distribute it globally to friends and acquaintances, who then started

off where it was frozen. Even the easiest interface would creak under that strain!

Where does it end? When a user exits your program, your responsibility ends. In the web they are just as likely to leave your site via a link to a third-party site. Your clean, easy to understand navigation model breaks down when someone leaves your site, but, of course, for them it is a single experience. To some extent this is similar to any multi-window interface. This is why Apple's guidelines have been so important in establishing a consistent interface on the Macintosh [22] (with similar, but somewhat less successful initiatives on other platforms). However, it would neither be appropriate, not welcomed by the web community, were one to suggest a single web look and feel!

In short, the difference between traditional interface design and web design is that the latter seems totally out of control.

3.2 Design of pages/documents

To caricature slightly there are two main types of professionally designed web page: content-rich, heavy-text pages vs. image-rich, CDROM-style pages, or to the typical home user: the quick and the dead! The former are more common on sites for service providers such as search engines, the latter in 'corporate' sites.

The most vociferous proponent of low-bandwidth web pages is Jacob Nielsen (also known for his opposition to frames) [23]. However, looking at other media suggests that the use of graphics and images significantly increases the accessibility of information.

There are good criteria for adopting either style: who are the users, what will appeal to them, what equipment/network connectivity do they possess, will there be visually impaired visitors to the site? However, it is apparent that many page formats have not been chosen based on such criteria.

One of the reasons for this is that many web designers (again to caricature) are either computing graduates with limited graphic design skills (lots of 3D buttons and moving graphics) or graphic designers with little understanding of interaction (pages where you have to hunt to work out what to click).

The problem is that a perfectly designed computer screen, that gives its message to you, that interests you and appeals to your aesthetic sense, no matter how good it looks on the designer's screen, is not necessarily a good web page. The web is a different medium, sharing many features with computer interfaces, CDROMS, print media and broadcast media, but distinct.

3.3 Understanding the medium

The creation of webs depends on an understanding of the medium and its affordances.

Time. Not only do pages take time to download, there are also delays as plug-ins initialise, the browser lays out tables, etc. This is the main reason cited for adopting graphics free (or low-graphics) pages. However, this can over-simplify the issue; for example, reuse of graphics within a site can mean that the first page takes a while to load, but subsequent pages are much faster as the graphics are cached. Also full-text sites are no panacea. Often they are designed with small pages with a few 'menu' options to move deeper into the site. As the network delay on web page loading is often dominated by the initial handshake with the server, a small page may take nearly as long to download as a larger one with more information! On the other hand, the nature of table layout algorithms may mean that text-rich (or graphic-rich) with large tables may take a long time to format.

Space. We have all seen pages that scale badly:

- where deep frame structures mean that the text we are trying to read is in a 2 cm square box at the bottom-right-hand corner of the screen,
- where the content is formatted to a fixed size and sits at the top or centre of a large otherwise empty screen,
- where a page on a small screen needs to be endlessly scrolled from side to side,
- where a page laid out on a large screen has lines so long it is hard to work out the next line as your eye moves back.

Good layout is no easy task at the best of times, but with the web you don't know what size screen you are formatting in, what fonts will be available, what the line spacing will be, or precisely where images will be placed! This is not simply a problem for the web, graphical toolkits such as X-Motif and Java AWT have complex layout rules to allow dynamic window resizing and there has been endless work on pretty printers and automatic text layout. It is probably fair to say that no existing solutions are perfect.

These problems are not helped by the fact that many web-authoring tools produce fixed size pages and extensive editing is required to make them appropriate for multiple platforms and screens. Although authoring tools often tackle the technological issues for multiple browsers (e.g. tables or frames, different styles of JavaScript), the issue of size is often completely ignored.

Again appropriate understanding of the medium can help. Graphically it is possible to produce strong vertical or horizontal features, but techniques suitable for fixed

layout, such as graphics breaking through horizontals and verticals almost always lead to misalignments on some browsers. Several sites use a relatively narrow, fixed-width layout, but stretch downwards – not perfect, but an effective compromise.

Colour and graphics. It is surprising how few designers understand the limitations of electronic media: colour reproduction: poor colour matching, dithering, etc. This is not helped by the fact that many applications do not interact correctly with the operating system's palette management. Even the 216 colour web-safe palette is not totally 'safe', especially for background images – in fact, colour problems in backgrounds seem so intractable I have largely given up using them! This being said, there are simple ways to avoid the worst of the pitfalls.

Standard colour guidelines (for any interface) hold. Remember colour blindness, especially the most common, red–green. In fact, if you rely on colour alone to distinguish things you are usually in trouble. The colour part of our vision is not nearly as precise or fast acting as the black-and-white (or to be precise contrast) part, so make sure that there is a difference in contrast not just colour between text (or line art) and background. Also some colours (in particular bright red or blue) are bad for picking out detail. Of course, red is also (as a *redundant* cue for the non-colour blind) the ultimate attention grabber! Also, for good reasons, browsers do not print background graphics (do you really want to use up all your ink cartridge on one page?), so white text on a black background may look alright on screen...

Time interacts with colour in compression technology. A little understanding of this can make an enormous difference. Many graphics programs will save in web-safe and optimised colour palettes, but using a small, web-safe, palette when an image is created may make the difference between a dithered (and usually large) image and a sharper, smaller image. It is common to see pages now with graphics split into many small pieces. There are good reasons to do this, for example, it may be possible to create mixed text/graphics layouts where the text appears quickly on screen; also the small images can use different palettes and graphics formats. However, many designers do not realise that the encoding used in GIF images means that large areas of flat colour compress to almost nothing. A single large image can often be smaller in total size and have less overhead than many small images.

Overall, the lesson is clear: do not adopt a simplistic attitude (e.g. "no graphics on my site"), but do use the medium to its best advantage to achieve the graphical and functional goals of the site within the technological limitations of the web.

3.4 Design of sites

The heart of interface design is not screen layout (despite what many outside the field think!), but interaction for a purpose. These skills are increasingly required within pages as they become more dynamic; however, the greatest interaction challenge of most sites is still inter-page navigation.

To some extent this is again just like any interface!

- knowing where you are
- knowing what you can do
- knowing where you are going

Aspects of these simple navigation rules can be found in many usability guidelines. Looking at the first.

Knowing where you are. How often have you gone to a previously bookmarked page, only to realise that it has no indication of context, no links to the site's main page. You may not have noticed when you navigated to the site, because you were operating within the context of previous pages on the site, but the page is meaningless on its own. Given any page may be bookmarked one must either design all pages to be accessed out of context, or deliberately prevent bookmarking with frames or server scripts.

Knowing what you can do. The scourge of 'designer' sites – a random movement of the cursor over icons (and even blank space), watching for the cursor to change from arrow to pointing finger, made harder by time delays as the cursor switches. Sometimes intriguing, sometimes appropriate, but certainly not easy!

Knowing where you are going. Think of sites where the navigation consists of a menu of single, slightly obscure terms. You click on a link, wait for the page, and then discover it was not what you wanted – back and try again! This is perhaps even more of a problem on touch-screens than the web – large buttons encourage large fonts, few words and many mistakes. It is not difficult to add a few words of explanation for each option – the screen is usually large enough! Where text would really spoil that beautiful graphic there are lots of techniques now for making explanations (e.g. pop-ups on rollovers).

Broad vs. deep. This gets us on to another issue, which has been touched on earlier. There are three pressures that have led to the frequent design of narrow–deep sites, that is where there are few options on each page, leading to long interactions. Two of these are associated with the different schools of web design and one with HCI itself. The first pressure is download time – less items shown means a smaller page. The second is graphic design – a small set of headings to navigate looks so much nicer.

The third is human processing capacity – the famous Miller 7 ± 2 result for short-term memory [24] is often misapplied to everything including the number of items on a page. So everyone is likely to fall for this one, but the evidence is that, for the web, broad–shallow structures are often better [25]. This is because we can scan lists quite quickly by eye, especially if they have some structure (alphabetic, numeric, hierarchical), and so if the download time is at all slow it is better to get deeper into the site from a single page. For CDROM and other interactive media the refresh time is faster and so different advice applies.

Warp drive. Of course the crucial difference between the web and other media is its openness. Users must eventually leave the safe world of your well-designed site for the wilds of the web beyond.

4. Taming the web

Users do not just navigate web sites, they navigate The Web. There are two sides to this.

- local structure – knowing where you've been and what's nearby.
- global structure – knowing what's out there.

4.1 Local structure

Even within sites or conventional, closed hypertext systems users can become disorientated. The problem has become known as 'lost in hyperspace'.

There are two design extremes. The first is design within rigid hierarchies, such as the librarian's catalogue or a typical book layout. The other is the free use of hyperlinks to represent the semantic net of knowledge. Indeed Ted Nelson's classic exposition on the power of hypertext [26] takes the hypertext paradigm back into print giving us pages full of semi-related snippets – textual photomontage.

In another classic paper, Thimbleby contrasted *over-determined* and *under-determined* interfaces [27]. Over-determined interfaces, such as a question-and-answer style, only offer you one thing you can do at a time. Under-determined interfaces, such as the UNIX or DOS command line, give you a blank screen and let you decide what to do next, or perhaps flounder. The ideal is a well-determined interface that gives you guidance and support, but does not unnecessarily limit you. With the structured vs. ad hoc extremes we can see shades of over and under-determination. In fact, studies do confirm that structure is important and that far from giving control, ad hoc structures leave users feeling confused and powerless.

4.2 Back and history

One way users can reassert their control over the web is by the tools they use to browse. Web studies have shown that the 'back' button accounts for over 30% of the actions performed in a browser [28,29] (this is compared with 50% link following). If you do the sums this means that about 2/3 of the times a user visits a page, they leave by going back rather than following links forward.

So, why so much going back?

- correcting mistakes – the user gets somewhere they don't want to be. Again the curse of terse labels!
- dead ends – the user gets where they want to go, but there is no way to go on.
- exploratory browsing – the user is just taking a look
- depth first traversal – the user is trying to visit all of a site, so is expanding links one by one.

Given the back button is so common one would hope it is clear to use. In fact, the semantics of 'back' are not entirely clear.

For one step it is pretty consistent – it takes you to the previous page. Unfortunately, as the web gets more complex there are many more types of interaction: frames, redirection, CGI scripts, applets and JavaScript. For the user, it may seem as if they are following a normal web link, but does the browser regard it as such? Of these, perhaps redirects are the most confusing (many browsers behave better in frames now). The user goes to a page, hits 'back' and the same page re-appears. What is really happening is that the browser has the extra 'redirect' page in its history list: when the user presses 'back', the browser goes back to the redirect, which then redirects them back to the page!

Multi-step 'back' is far less clear. Web use studies show few people using these or history mechanisms. One reason is that the 'back' menu (called 'Go' in Netscape!) depends on the visited pages having meaningful 'title' tags. Some title pages are useful for distinguishing pages within a site, but poor at telling which site they refer to. Some sites have very similar titles on all pages. Some have no titles whatsoever.

Another reason is that the meaning of multi-step 'back' is very unclear even for hypertext browser designers. Although web browsers are (reasonably) consistent in their model, a comparison of several different hypertext browsers showed that they all had different behaviour when dealing with multi-step 'back', especially when the path involved multiple hits to the same page [30]. In particular, the 'back' button on several hypertext systems does not adequately support depth-first traversal.

The semantics of full histories get even more confusing – do you record the backward paths? Do you

record all the pages visited within a site? Do you record repeat visits to the same page? It's no wonder that users rarely use these features. However, when Tauscher and Greenberg [28] analysed revisitation patterns they found that, although many pages are only visited once, a significant number are revisited. So, there is great potential for well-designed histories. See their paper for a short review of graphical history mechanisms.

4.4 Global structure

With something like 3 million web sites and countless individual pages, how on earth does one get an overview of the material? No wonder so many people feel utterly overwhelmed by the web – you know the information is there, but how to find it?

Although people may have a similar feeling about the Library of Congress or the Bibliothèque nationale de France or the Bodleian, for some reason we feel less guilty about not having found all the relevant books on a subject than we do in missing a vital web page. Electronic omniscience appears just within our grasp, but this dream is hubris. It could be that the challenge is the need not so much to access all available information, but to accept the incompleteness of information.

5. The geometry of the web

In fact, it is possible to capture the entire web, as the recent snapshot of the web by the Alexa project has shown [31]. The problem is to understand the structure of it. You can see a beach at sunset, and somehow grasp it all, but it would be foolish to try to somehow understand each grain of sand, or even to find the smallest. Similarly, it is reasonable and valuable to view the overall structure of parts of the web.

Maps of the web, both site maps and representations of larger bodies of pages can help give us such an overview, but they are usually portrayed in 2D or 3D, and the web just isn't like that. We need to understand the geometry of cyberspace itself [32]

We are used to the geometry of 2D and 3D space – we've lived in it all our lives! However, it doesn't take much to confuse us. This has been part of the mystery and fascination of mazes throughout history [33]. One of the biggest problems with mazes is that two points that appear close are, in fact, long way apart. In cyberspace, not only does this happen, but also distant points can suddenly be joined – magic.

The most obvious geometry of cyberspace is that of the links. This gives a directed graph structure. Actually the directedness in itself is a problem. Just like driving round a one-way system! This is another reason the

'back' button is so useful: it gives us the official permit to back up the one-way street after we've taken the wrong turning.

Lots of systems, including most site management tools, use this link geometry to create sitemaps. Different algorithms are used which attempt to place the pages in 2 or 3 dimensions so as to preserve some idea of link-closeness. The difficulty (as with any graph layout algorithm) is two-fold: (i) how to deal with remote links, and (ii) how to manage the fact that the number of pages distance n from a given page increases exponentially whereas the space increases linearly (2D space) or quadratically (3D space). The first problem (i) is fundamentally intractable, but in practice is solved by either simply repeating such nodes or marking some sort of 'distant reference', effectively reducing a directed graph to a tree. The second problem (ii) is intractable in normal space, even for trees. The Hyperbolic tree (see figure 1, [19]) gets round this by mapping the web structure into a non-Euclidean space (although beware: some papers describing this work confuse hyperbolic and projective geometries). Of course, they then have to map this into a 2D representation of hyperbolic space!!

The second form of web geometry is that defined by its content. This is the way search engines work. You look for all pages on or close to some hyperplane of a high dimensional space (where the dimensions are occurrences of different words). Alexa operates on a similar principle, indicating the closest page to a given one using similar content as a distance metric [31] and there are several web mappers, very similar to the link mappers, but using this form of semantic distance as the metric [34].

The third kind of geometry is that given indirectly by the people who view the pages. Two pages are close if the same people have viewed them. A whole battery of 'recommender' systems have arisen which use this principle [35,36].

Of course, these are not independent measures. If pages share some common content, it is also likely that they will link to one another. If pages link to one another it is likely that the people will follow these paths and hence visit the same pages. If search engines throw up the same two pages together for certain classes of query it is likely they will have common visitors.

Ideas of 'space' and navigation in the web and other virtual and cyber-spaces have been explored in several recent workshops [37,38,39].

6. Future trends

So much for the current state of interfaces for the web, but where is the Internet going in the future and what will this mean for interfaces?

Ubiquitous computing emphasises the fading of computers into the background of our everyday lives, until they become invisible [40]. Similarly, the Internet must become less visible and simply become the glue between things. The future is thus the way the Internet breaks barriers between things, making parts of life that used to be disparate come together – bringing the world to you, removing the distinction between your computer desktop and the rest of the electronic world, linking people to one another.

6.1 No boundaries

The Internet has already brought many aspects of the physical world closer: you can watch satellite pictures of the earth or see into someone's flat with a web cam [41]. Not only can you see the world, but you can also control it. Many network devices (routers, printers etc.) are now configured by web interfaces, potentially from any place in the world. Further, you can also control radio telescopes and model robots on the net [42].

6.2 aQtive onCue

At aQtive we are producing Internet software that is looking towards this integrated future. One product is aQtive onCue (available for free download from the aQtive web site [43]).

To see how aQtive onCue works, let's imagine a simple scenario – Sarah is in her office and starts up aQtive onCue. Initially a small floating window appears with a few information icons in it (Fig 2a).

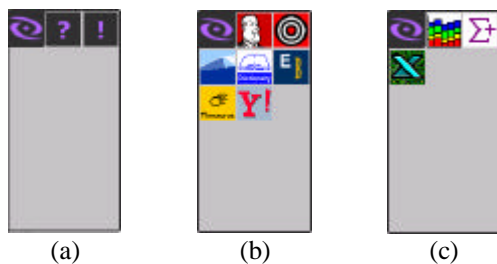


Fig 2. Sarah opens her email

Sarah then starts to look at her email, she finds a message from a colleague. The message contains text and also a table (laid out with spaces) as well as the URL of a web page.

She first selects and copies the word "histograms" in the text. When she does so, the aQtive onCue window changes (Fig 2b). Several icons appear in it representing several things she may want to do with the words "histograms". aQtive onCue suggests looking up "histograms" in various online search engines, an online thesaurus and dictionary and the online Encyclopedia Britannica (eBlast).

She double clicks the thesaurus icon and aQtive onCue launches a web browser and directs it to the thesaurus service which then returns a web page listing similar words such as chart, diagrams, etc.

After looking at this for a while Sarah selects the table in the text (Fig 3, lower left). This time the search engines are not suggested. Instead aQtive onCue suggests three desktop programs: Dancing Histograms, SumIt! and Microsoft Excel (Fig 2c) She double clicks the Dancing Histograms icon and aQtive onCue launches the Dancing Histogram application (Fig 3 centre).

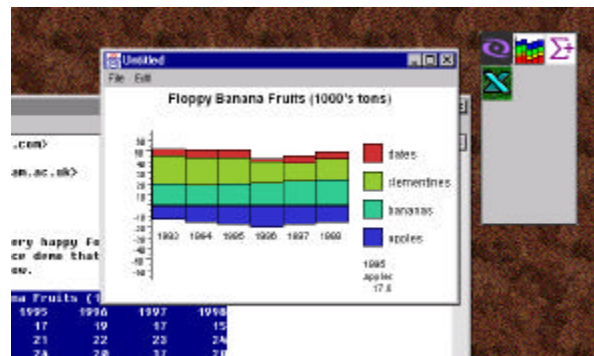


Fig 3. Sarah clicks histogram icon

She interacts with this for a while. She then clicks the Excel icon, which reveals a menu of possible things to do with Excel including drawing a 3D chart. She selects this and aQtive onCue responds by opening Excel, pasting the data into a new worksheet, and then telling Excel to draw the chart. Sarah just has to watch it appear.

Sometimes both Internet and desktop services are recommended for the same data, sometimes one, sometimes the other – the crucial thing is that it doesn't matter. Both Internet and desktop services are treated equally and furthermore are seamlessly linked into the user's day-to-day activities.

As aQtive onCue is constructed from plug-in components, it is also an excellent platform for developing both commercial software and student projects. It is freely available for this purpose – see the developer section of the aQtive web site [43].

6.3 PopuNET

Integration of the desktop and the Internet is a first step, but the changes to the Internet need to be more radical than that. Surveys such as GVVU's annual WWW User Survey [44] are beginning to see a shift from the professional, often academic, web user to a much broader social and cultural mix. If Internet usage grows as expected we will see not just a quantitative growth, but a qualitative change in the net. I have previously called this PopuNET [45] and is characterised by the availability of the net by *everyone, everywhere and everywhen*. Of these the last is perhaps the most important

Everyone. We have all seen the growth statistics! There are three stages to cultural change. First when all one's friends, family and co-workers are 'connected' – you experience an individual change of perspective. The second is when a whole country is largely connected – leading to a national cultural shift. Third, is universal global connectivity. At present Scandinavia and the US are fast approaching stage two (penetration over 25%), but still with a strong bias towards more affluent social groups. As Internet use climbs society will restructure itself (as occurred with car use in the West) and the dangers of cultural and economic exclusion grow.

Everywhere. In the UK the principal reason for purchasing a home PC is now to use the Internet. However, a PC is not the only entrance to the Internet. Many libraries have public Internet access, recasting their role as information (not simply book) providers. Mobile Internet access is growing using proxies to reformat pages for small screens [11]. Perhaps most significant is the fact that games consoles and TV set-top boxes are beginning to have Internet connectivity. This has the potential to really break the social barriers to connectivity within developed countries (in the UK more households have television sets than bathrooms!),

Everywhen. Finally, and most important, is the ability to interact with network resources at any time. Having Internet capable devices to hand (be it a PDA or games console) means that it is possible to *connect* at any time, but this is not the same as *being connected*. Many academic and office users will have relatively fast and permanent Internet connections. In contrast, a home user has to *get* connected using a modem to use the Internet. In some countries (including the UK), the telephone charges have been a real barrier to heavy use. This is changing. In the USA free local calls (albeit now under threat) make connectivity easier and some households even have an additional phone line dedicated to the

Internet. In the UK cable modems and free Internet Service Providers are having an equally profound effect.

It is only when the network is as available as the television or telephone, that it can become a part of everyday life.

Although PopuNET is not here in its entirety, we are approaching stage one of 'everyone' connectivity within certain social groups in several western countries, and it is clear that this techno-cultural shift is fast approaching. What is less clear is what network 'applications' will be appropriate. Do we really see a world of web surfers instead of TV watchers, or will it simply be a world of interactive TV participants?

To some extent web community services such as Yahoo clubs [46] seem to be catching this wave. At aQtive we too are working to create products that look towards this new Internet world – watch this space!

7. Summing up

If you can keep up with the endless technology shifts, the web is an excellent, but far from trouble free, platform for prototyping and deploying user interfaces. Designing for the web itself needs all the same care as any other interface, but with additional issues. It is a new medium and, as with all media, you need to understand it and work with it to produce the best results. The web is fast drawing a large proportion of the population into cyberspace. It is a territory with many guides, but few maps. Representations and tools to help us navigate this strange land are very necessary, but by no means ready now.

The future for the web must be in drawing parts of our disparate pre-millennium world together. We saw how aQtive onCue takes a step towards this integration. The massive change will come as the net becomes continually available to a large proportion of (at least significant groups of) the population. As an academic and as a product developer I have some inklings of the changes that this PopuNET will bring, but the changes are so radical it would be foolish to predict the ultimate direction this will take us.

8. References

- [1] Amazon Books. <http://www.amazon.com/>
- [2] Macromedia. <http://www.macromedia.com/>
- [3] R. Bentley, U. Busbach, D. Kerr and K. Sikkel (Eds.), *Groupware and the World Wide Web*, Dordrecht, Kluwer, 1997.
- [4] S. Buckingham Shum and C. McKnight (editors), "Special Issue on World Wide Web Usability", *International journal of Human-Computer Studies*, 47(1), 1997.
<http://www.hbuk.co.uk/ap/ijhcs/webusability>

- [5] D. Clarke, and A. Dix, *Proceedings of The Workshop on the Active Web*, Staffordshire University 20th January 1999.
<http://www.hiraeth.com/conf/ActiveWeb/>
- [6] G. Pfaff, and P.Hagen (Eds.), *Seeheim Workshop on User Interface Management Systems*, Springer-Verlag, Berlin, 1985.
- [7] S. Lewis, *The Art and Science of Smalltalk*. Prentice Hall, 1995.
- [8] J. Coutaz, "PAC, an object oriented model for dialogue design". *Human-Computer Interaction – INTERACT'87*, Eds. H.-J. Bullinger and B. Shackel. Elsevier (North-Holland), pp. 431-436, 1987.
- [9] "The UIMS tool developers workshop: A metamodel for the runtime architecture of an interactive system", *SIGCHI Bulletin*, **24**(1), pp 32-37, 1992.
- [10] A. Dix, "The Active Web", *Interfaces*, Part 1 – **38**:18–21, Part 2 – **39**:22–25, 1998.
- [11] A. Fox, S.D. Gribble, E.A. Brewer, and E. Amir. "Adapting to Network and Client Variation via On-Demand, Dynamic Distillation", *Proceedings of ASPLOS-VII*, Boston, Massachusetts, U.S., 1996
- [12] T. Alanko, K. Kojo, M. Liljeberg and K. Raatikainen. "Mobile access to the Internet: a mediator-based solution", *Internet Research*, **9**(1): 58–65, 1999.
- [13] S. K. Card, T. P. Moran, and A. Newell, *The Psychology of Human Computer Interaction*, Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1983.
- [14] B. Shneiderman, "Response time and display rate in human performance with computers", *ACM Computing Surveys*, **16**(3), pp. 265-286, September 1984
- [15] D. Ramduny and A. Dix, "Why, What, Where, When: Architectures for Cooperative Work on the World Wide Web", *Proceedings of HCI'97*, Springer-Verlag, pp. 283–301, 1997.
- [16] A. Dix, D. Ramduny, T. Rodden, N. Davies, Places to stay on the move: software architectures for mobile user interfaces, 1999. <http://www.hiraeth.com/alan/topics/mobile/>
- [17] C. Ahlberg and B. Shneiderman, "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays", *Proceedings of CHI94*, ACM Press, pp. 313-317, 1994.
- [18] G. Robertson, S. Card, and J. Mackinlay, "Cone trees: Animated 3D visualisation of hierarchical information", *Proceedings of CHI'91*, ACM Press, pp. 184–194. 1991.
- [19] *Hyperbolic Tree*, Inxight Software, 1999.
<http://www.inxight.com/>
- [20] *Yale C/AIM Web Style Guide*
<http://info.med.yale.edu/caim/manual/>
- [21] *ACM SIGCHI forum on HCI and the Web*.
<http://www.acm.org/sigchi/webhci/>
- [22] *Macintosh Human Interface Guidelines*, Apple Computer, 1996. <http://developer.apple.com/techpubs/mac/>
- [23] J. Nielsen, *AlertBox*, <http://www.useit.com/alertbox/>
- [24] G. A. Miller, "The magical number seven, plus or minus two: some limits on our capacity to process information", *Psychological Review*, **63**(2):81-97, 1956.
- [25] K. Larson. And M. Czerwinski, "Web Page Design: Implications of Memory, Structure and Scent for Information Retrieval", *Proceedings of CHI98, Human Factors in Computing Systems*, ACM Press, pp. 25-32, 1998.
- [26] T. Nelson, *Literary Machines: The Report on, and of, Project Xanadu, Concerning Word Processing, Electronic Publishing, Hypertext, Thinkertoys, Tomorrow's Intellectual Revolution, and Certain other Topics Including Knowledge, Education and Freedom*, 1981.
- [27] H. Thimbleby, Dialogue determination. *International Journal of Man-Machine Studies* 13: 295-304, 1980.
- [28] L. Tauscher and S. Greenberg, "How people revisit web pages: empirical findings and implications for the design of history systems", *International Journal of Human Computer Studies*, **47**(1), 1997.
<http://www.cpsc.ucalgary.ca/grouplab/papers/1997/>
- [29] L. Catledge and J. Pitkow, "Characterizing browsing strategies in the World-Wide Web",. *Proceedings of the Third International World Wide Web Conference*, Darmstadt, Germany, 1995. <http://www.igd.fhg.de/www/www95/papers/>
- [30] A. Dix and R. Mancini, "Specifying history and backtracking mechanisms", *Formal Methods in Human-Computer Interaction*, P. Palanque and F. Paterno. (Eds.), Springer-Verlag. London, pp. 1-24, 1997.
<http://www.soc.staffs.ac.uk/~cmtajd/papers/histchap97/>
- [31] C. Lohr, *Alexa Internet Donates Archive of the World Wide Web to Library of Congress*. Alexa Internet Press Release, 13th Oct. 1998.
<http://www.alexa.com/company/inthenews/loc.html>
- [32] <http://www.hiraeth.com/alan/topics/cyberspace/>
- [33] A. Fisher, *Labyrinth - solving the riddle of the maze*, New York: Harmony Books, 1990.
- [34] C. Chen and M. Czerwinski, "From Latent Semantics to Spatial Hypertext – An Integrated Approach", *Hypertext'98*, ACM Press, pp. 77–86, 1998.
- [35] Resnick, P. and H.R. Varian (guest editors), Special Issue on Recommender Systems. *CACM*. **40**(3):56–89, 1997.
- [36] AAAI-98 Workshop on Recommender Systems, 1998.
<http://rsv.ricoh.com/~marko/rec98/>
- [37] S. Jul and G. Furnas, "Navigation in electronic worlds: a CHI 97 workshop", *SIGCHI Bulletin*, **29**(4), pp. 44-49, 1997.
<http://www-personal.engin.umich.edu/~sjul/nav97/>
- [38] Workshop on Personalised and Social Navigation in Information Space, Stockholm, March 1998.
<http://www.sics.se/humle/projects/persona/web/workshop/proceedings.html>
- [39] T. Kindberg (organiser), *Changing Places: workshop on workspace models for collaboration*, QMW, London, 1999.
<http://www.dcs.qmw.ac.uk/research/distrib/Mushroom/workshop/>
- [40] M. Weiser, "Some computer science issues in ubiquitous computing", *CACM*, **36**(7): 75-84, 1993.
- [41] N. Roussel, "Beyond Webcams and Videoconferencing: Informal Video Communication on the Web", in [5]
- [42] Rita the Robot. <http://www.cs.bham.ac.uk/~axg/Rita.html>
- [43] *aQtive limited*. <http://www.aqtive.com/>
- [44] *GVU's WWW User Surveys*.
http://www.cc.gatech.edu/gvu/web_surveys/
- [45] *PopuNET*. <http://www.magisoft.co.uk/PopuNET/>
- [46] Yahoo! Clubs. <http://clubs.yahoo.com/>