



Working Paper

Forecast of External Developments

Editor/Author:	Simon McLeish
Project Document Number:	HL-1998-07
Publication Date:	16 July 1998
Task:	1.2
Status:	Adopted
Version:	0.2
Keywords:	World Wide Web, Internet environment developments
Abstract:	- Developments that we might expect during the HeadLine project lifetime in: the Internet and related standards; WWW servers; WWW browsers and the desktop; network computing; Java and WWW programming-
Confidentiality:	Public
Copyright:	LSE, LBS, UH
Edit source file:	H:\headline documents\documents folder\HL-1998-07a.doc
Reference URL/URI:	
Last saved:	20 July 1999

London School of Economics • London Business School • University of
Hertfordshire

The HeadLine project is partly funded by the Joint Information Systems Committee (JISC) of the Higher Education Funding Councils, as part of its Electronic Libraries Programme (eLib)

Contents

Introduction	3
Internet and Related Standards	4
Service Integration	4
Personalisation	5
Usability Studies	5
World Wide Web Browsers and the Desktop	6
Web Browser Development and the Microsoft/US DOJ Legal Actions	6
HTML Development	8
XML	8
DHTML	11
World Wide Web Servers	12
HTTP Development	12
Java and Web Programming	13
Client/Server Computing	15
Network Computing	15
Software Agents	15
CD-ROMs	18
Bibliography	19

Introduction

This paper will look at the developments that we might expect during the project lifetime in: the Internet and related standards; WWW servers; WWW browsers and the desktop; network computing; Java and WWW programming. The sources used will mainly be research papers and briefings available on the Internet and elsewhere (as suggested by those working on the project), and will also look at the IT strategies (if available) of the institutions involved in HeadLine.

There is no way that predictions of the future can be expected to be 100% accurate. "Everyone agrees that some technologies are fated to become dogs. The hard part is knowing which will be dogs while they're still being hyped by their makers and the press. How do you tell?" [16].

A useful glossary containing the terms discussed in this document is [26].

Additions made during approval cycle are mainly in the form of extra footnotes, which are marked as such. A short section has been added on CD-ROMs and an additional paragraph on "action sheets" and "behaviors" in the section on Java and Web Programming.

I'm going to try to make reasonably regular updates to this document, probably in the form of e-mails to the HeadLine mailing list.

Internet and Related Standards

“To see the Internet’s future, look to its past” [8].

Service Integration

Although to most people the terms “Internet” and “World Wide Web” are now synonymous, this does not mean that all the other Internet protocols have died out completely. In fact, e-mail is still the most commonly used Internet application. One trend which is likely to continue is the increasing integration between the different services, as clients become more adept at handling different kinds of files; users now expect all kinds of applications to recognise a URL, highlight it and seamlessly open a Web browser when it is clicked on. Increasing integration with the Web has also affected e-mail servers; newer versions of Microsoft Exchange and Lotus Notes, for example, can allow users to read their e-mail with a Web browser from anywhere in the world, with the minimum of work from a system administrator. (In practice, these services are still difficult to access and administer.) In the last year, Web-based free e-mail systems such as Hotmail (now owned by Microsoft), Rocket and My Yahoo – which is also a personalised news service – have been among the most successful Web sites.

Successful integration between different clients has led to the idea of developing a completely integrated desktop. Such integration is the reason behind one of the most important developments likely to affect the Internet in the foreseeable future: the US Department of Justice case against Microsoft. The original action alleged that the close integration between Internet Explorer and Windows 95 made possible by Microsoft’s ownership and non-disclosure of the code of both amounted to an unfair attack on their competitors. Turns taken by the action toward the end of May [35] suggest that a rather different line of attack will be used as the case progresses – that Microsoft have used their dominant position in the marketplace in a manner unfair to competition [39]. (See below: *Web browser development*.)

Personalisation

The email sites mentioned above point up another current Internet trend: personalisation is in. More and more sites are setting up as (or including) personalised news services, personalised Web directories, personalised e-mail, and so on. The user can customise all of these, so they see only news items they are likely to be interested in (for example). This is a trend that the currently envisaged HeadLine system will fit in well with, with personalised bookmarks and personal areas. Web content is expected to become “self-optimising” [28] – in other words, usage information will be fed back into the system to dictate default choices.

Usability Studies

It is interesting to see that users' ideas of what they want from the Web haven't changed very much over the last three or four years. The following list, summarised from [27], gives findings from user studies in 1994 that matched findings from similar user studies in 1997:

- Users don't read on the Web: they scan the text. Paper versions are printed for reading.
- Users don't like bland impersonal corporate sites. (This is probably not particularly relevant to a project like HeadLine; users of academic library sites are going to expect – and probably want – a site that is impersonal.)
- Users are impatient; they want to get their answers immediately and do not want to be slowed down by “cool” features, mission statements, or self-promotion. (Or, in the HeadLine context, multiple Z39.50 queries to slow servers.)
- Users often print pages because they don't trust sites to be stable enough to have the page again at a later date, as well as to make them easier to read.
- Download times are becoming ever more critical and sites need to design for speed. (Tolerance for slow downloads decreased significantly between 1994 and 1997.)
- Search was always liked by users, and has now become mandatory for any large site.
- Users are intolerant of downtime and crashes.

Apart from changes caused by the introduction of new technology (animation, frames and garish backgrounds are intensely disliked), the main change is that users are now prepared to scroll down long pages, though the need to scroll (e.g. to find important navigation buttons) still reduces usability. It seems likely that none of the listed findings are going to change in the near future, so they should certainly be borne in mind when designing the HeadLine interface.

World Wide Web Browsers and the Desktop

Web Browser Development and the Microsoft/US DOJ Legal Actions

The current generation (version 4.x) of the major browsers consists of huge pieces of software, unwieldy to download and install, and memory hungry in operation. This is a situation caused by the competition between Microsoft and Netscape to get the browser with the most features out fastest, regardless of the quality of the software. The 5.0 browsers are likely to be smaller, faster and more robust [4].

The competition between the two companies is likely to dominate the immediate future of the World Wide Web, as both gear up to a fifth version of their browser software and attempt to anticipate the outcome of the Microsoft anti-trust legal action.

“The major client-side battles of '98 will be fought over browser/operating system integration, Java and competing versions of non-standard technologies such as DHTML” [8].

The outcome of the legal action against Microsoft is likely to be extremely important to the future of the Web browser and the desktop, and exercises the minds of many commentators on the Web. It is difficult to know how it will turn out, though there is an element of wishing against expectation that Microsoft will be destroyed in many of the analyses. Most of the “industry gurus” questioned in [7] felt that the action would eventually fail, mainly because of a lack of political will on the part of the Clinton administration.

The current move from the technical issues of desktop/browser integration to Microsoft's alleged anti-competitive business strategies in general is thought by commentators to greatly increase the strength of the Department of Justice's case [35].¹ The announcement of the charges by the attorney general, Janet Reno, also included an indication of the US government's attitude to the Internet:

“No firm should be permitted to use its monopoly power to develop a chokehold on the browser software to access the Internet.”

If Microsoft wins the action, then the next generations of Windows will see greatly increased integration with the Internet Explorer (IE) browser. In some ways this could be a good thing, as it will make the distinction between items on the local machine and items everywhere else a transparent one. Users will grow to expect everywhere they go to have the same kind of interface.

¹ [Added during approval cycle] Microsoft has indeed now won a significant victory in the first part of the suit, obtaining a reversal of an earlier decision forbidding the shipping of Windows '98 with integrated copies of Internet Explorer. This decision used a particular definition of integration (a product is integrated if “the integration of the two pieces creates a product that was different and better than if an OEM tried to put the pieces together”) which led the court to believe that Internet Explorer was, in fact, a separate product [22,38]. The author of the definition (known as the “Areeda Treatise”) has since stated that the court misused it in coming to their decision, and that it implies that Windows '95 and Internet Explorer are separate products [9].

It will pretty much lock Netscape out of the Windows world², though there are noises being made from the Netscape camp that indicate that Netscape will try to do the same with Communicator and Linux as Microsoft will have done with IE and Windows [19]. Free software products can be successful. The Apache WWW server has a greater share of that market than Microsoft and Netscape between them; the Linux UNIX OS is "the only non-Microsoft OS that is gaining market share" [*ibid.*]. The rapid improvements made in the source code for Communicator show that there is sufficient goodwill in the computing community to enable Netscape's product to remain competitive [29]. Marc Andriessen of Netscape is trying to talk other software manufacturers into following Netscape's lead, citing Sun's problems keeping Java development under its control [19].

If Microsoft loses the action it becomes much more difficult to predict what might happen. To a large extent, the specific actions taken by the court will determine the fate of the company. If these are light enough, Microsoft may well have almost as much power over the desktop as it would have done if it had won. External legal sources have criticised the 1995 agreement that Microsoft is alleged to have breached, "saying that the agreement has done little to rein in the...software company" [12], and indeed, can do little.

One option would be "to prohibit or require certain activities" [*ibid.*]; the different measures falling within this option are those favoured by Microsoft's competitors. These measures are discussed in detail in [12], and are favoured by the US government as interim measures: it is asking for an injunction to require the incorporation of Netscape Navigator into Windows '98 as an alternative to Explorer. The feeling among commentators is, however, that "demands regarding Windows 98, for example, or even specific concessions to Microsoft competitors like Netscape...will probably be outdated in a matter of months as technology continues to change at its usual breakneck pace" [6].

The other possibility is to split Microsoft up; various ways to do this have been suggested (separating the software and operating systems parts and setting up smaller companies retaining the same structure as the parent as was done with AT&T in 1982 are examples). A report that looks at the possibilities in detail is discussed in [6]; its conclusion is that a break-up of Microsoft is the only way the US government can ensure the competitive market they seek in the software industry comes into being.

Vint Cerf and Robert Kahn, the developers of the TCP/IP protocol, think that whatever happens, it won't be very important to the future of computing. Microsoft will fall from its current leading position because new ideas will come along; historically, computing ideas have come from young outsiders rather than large corporations. Also, they expect that Java and intelligent agents will quickly reduce Windows' dominance [24].

It is likely enough that the complicated nature of the suits involved (particularly if the number of them is increased by individual US states – currently 20 of them – bringing suits) will mean that the issue will be protracted beyond the lifetime of the HeadLine project. "Microsoft can be expected to fight a war of delays and attrition in the hope that the next administration will take a more docile attitude" [35]. However, the US judicial system seems to be attempting to prevent the cases dragging on forever and getting bigger and bigger. The two federal cases are to be merged into one, and the judge for the September hearing is limiting the number of in-court witnesses to twelve on each side [36].

Microsoft is not the only company that is pursuing increased integration of the browser and the desktop. "The new Mozilla 5.0 Navigation Center looks like it's taking a big step in this direction"

² This won't happen immediately: "Neither side is going to do away with the other in 1998", according to Jim Balderston of Zona Research Inc. [8].

[4], and this will likely form the core of the Netscape Communicator 5.0. This system will in addition use XML to find information about parts of the Web that you have yet to visit. (The Navigation Centre is rumoured to be part of the Communicator 4.5 release, scheduled for late June '98.³)

HTML Development

The last five years have seen the Internet evolve extremely rapidly following on the massive interest created by the invention of the World Wide Web. It has become apparent that the pace of change is slowing somewhat, as the technologies become maturer. The evidence for this includes the lengthy gap between versions 4 and 5 of the Netscape and Internet Explorer browsers, as compared to that between versions 3 and 4. This is apparently caused by a change in attitude at Netscape and Microsoft; it is not enough to produce the browser with the most features, they want one which runs better and more economically [4].

Much of the early development was designed to make it possible to do things which HTML originally did not allow; it was driven by the perceived needs of Web page authors and designers rather than of users. (Examples of technologies not wanted by most users include inline video and audio, and push.) Technologies fail to gain the attention of users because "in general, users are more concerned with usability than with technology. Hence, if an application is more difficult to use or configure because it relies on one of these technologies, its popularity will be minimal" [23]. So there will be limited acceptance of any development that's proprietary, or that requires installation of a plug-in, or that's not completely supported by all browsers.

The most obvious paradigm, push, failed because it is annoying to users, and hogs bandwidth slowing other Internet applications down. 60% of Microsoft's corporate beta testers said they would not have any uses for push [16], but that did not prevent the advent of the active desktop and Netscape's Netcaster. A year after the advent of the 4.0 browsers, with their emphasis on push technology, "push is a word used only by Web columnists (and then always in the company of sarcastic quote marks)" [4].

XML

There were huge leaps forward in user interface design in this period; not all the "advances" turned out to be rapidly outdated gimmicks. The perceived limitations of HTML which are now causing concern are not so much to do with appearance as with content – the Web currently "does not provide the extensibility, structure and data checking needed for large scale...publishing" [3].

The next generation of HTML – or rather, the related standards such as XML and RDF – supports what Tim Berners-Lee calls "evolvability". This means that its syntax allows for customising and extending the standard, and for the interpretation by clients of such customised additions. Thus it will make it possible for new, customised, features to be added without leaving behind millions of users unable or unwilling to support the newest version of a particular browser or download yet another helper application needing installation and configuration. "XML and its related technologies will let users make sense of...data in a way that is comfortable to them" [23]. XML is complementary to HTML; "browsers will be able to process both, and future HTML standards will

³ [Correction added in approval stage] In fact, it was only a pre-beta form of the software that was released to developers in late June; the full release is scheduled for September 1998.

likely allow mixing HTML and XML in the same document" [18]⁴; the idea is to separate the content and organisation of data rather than set up a new text markup language. XML is a reduced subset of SGML, but has the aim of allowing extensibility while keeping the ease of use of HTML and avoiding the complexity of SGML.⁵ Content developers will be able to define new tags as they need them; the only restriction they must conform to is that the new tags must be correctly defined and used.

So far as the Web is concerned, there are three important parts to the XML specification [5]. The idea of the DTD (Document Type Definition) is inherited from SGML; this is the part that means that XML is extensible, by imposing a syntax for the definition of new tags and their relationships to each other. The rendering of these tags can then be customised through the XSL (eXtensible Style Language), which is a language allowing style sheet definitions. XSL is rather behind the other components of XML in terms of development [18]. Developers need only use one of XSL and CSS, as it is intended that mechanical mapping from one to the other should be possible. XSL is likely to be a subset of the Document Style Semantics and Specification Language (ISO/IEC 10179). XSL will have the following properties [3]:

- Free extensibility so that stylesheet designers can define an unlimited number of treatments for an unlimited variety of tags.
- Full functionality as a programming language so that stylesheet designers can arbitrarily extend the available procedures
- Ability to address the entire tree structure of an XML document so that contextual relationships between elements of a document can be expressed to any level of complexity
- International language/script support
- Sophisticated rendering support to allow complex page layouts
- Partial rendering support to allow efficient delivery across networks (i.e. parts of documents can be displayed before the whole is downloaded)

The third major part of XML is XLL (eXtensible Link Language). HTML contains only a very small part of the functionality traditionally associated with linking in hypertext systems, and XLL should bring in the linkage methods familiar to users of more sophisticated SGML applications. XLL will support as basic standardised link mechanisms location-independent naming, bi-directional links, links that can be specified and managed outside the documents to which they will apply, multiple link systems (such as rings), aggregate links (to multiple sources), transclusion (including linked document within linking document), attributes on links (link types).

The benefits of XML are expected to be [18]:

- More meaningful results from search engines (since they have access to a more machine understandable document structure)
- Easier maintenance of Web content by developers (since it's easy to perform such tasks as updating lists of links to be included in every page on a site)

⁴ At the moment, XML is incompatible with HTML and cannot be read by current Web browsers, and HTML documents will need modification to be read by (purist) XML readers.

⁵ Because HTML has a fixed set of tags, the SGML DTD that defines HTML doesn't need to be included in each page, which makes it easier to learn and implement than full SGML. XML developers will be able to use public DTDs or none at all instead of having to develop their own. In addition, client applications are allowed to assume a document is valid without checking it against the DTD, to save time and bandwidth. Nevertheless, it is suggested that file compression likely to be included in new versions of HTTP will be needed to transfer the comparatively lengthy files efficiently across the Internet [18].

- More stability in Web content (better handling of moved and missing material, using XLL)
- More intelligence in clients (so that tasks like privacy negotiation – see information on P3P below – can be undertaken without constant intervention from the user)
- More user tailored content (using XSL)

Peter Flynn, co-author of the XML FAQ, has said [38]:

“XML is going to provide an unequalled opportunity to break the mould of the “word-processor mindset”. Under that ethos, you typed stuff for one purpose only: to print it. Word-processors – in the main – are useable for that one purpose only: a heading is only identifiable visually by a human, to the word-processor it’s just a bunch of letters in a font. XML/SGML/HTML provide the ability to say *why* these letters are where they are, what role they play in the document. When markup captures *why* we use text in particular ways, documents can be used for multiple purposes – printing, the Web, databases, searching, Braille – and can still be re-used in other forms and by other people regardless of the computing environment. This all holds true because SGML (in which HTML and XML are grounded) is a non-proprietary format.”

The version 4 browsers support some parts of the earlier drafts of XML. For example, Microsoft’s Channel Definition Format used to manage the personalisation of the Explorer push system is defined using XML. Other companies, including Adobe, Sun and Xerox not immediately involved in browser software but providing related products have announced their support for the standard.

For the purposes of HeadLine, it might be more important to know which resources support XML browsing. Any well-formed SGML (such as that which can be produced by a Z39.50 server) should be fairly easily readable in XML. For such formats, it’s just a case of creating a version of the SGML DTD being used which contains only those parts of SGML used by XML (or getting hold of someone else’s XML-compliant DTD)⁶. DTDs for HTML written in XML should soon be generally available HTML. The problem with HTML is that today’s browsers are pretty sloppy about how conformant the HTML they render needs to be, so that many applications do not produce valid HTML. XML readers will almost certainly need to be stricter about this. (See [11] for more details on this issue.)

The chair of the W3C’s XML Working Group, Jon Bosak, has suggested four categories of application [3] which are likely to find XML of particular interest as an alternative to server or client coding. Of these, three seem to fit in with HeadLine, making it a technology worth looking at in some detail.

- Applications that require the Web client to mediate between two or more heterogeneous databases (including applications requiring the use of digital signatures)
- Applications in which intelligent Web agents attempt to tailor information discovery to the needs of individual users
- Applications that require the Web client to present different views of the same data to different users (particularly dynamic tables of contents generated on the fly)

⁶ There does not seem to be a working XML/Z39.50 gateway already in existence, but given the comparative simplicity of such a gateway compared to an HTML/Z39.50 gateway, it should only be a matter of time.

- Applications that attempt to distribute a significant proportion of the processing load from the Web server to the Web client (particularly applications which interface to intelligent Web agents)

An additional benefit of XML is that "Microsoft has announced its intention to support seamless, lossless translation from all the proprietary binary formats of MS Office to XML and back again" [38]. This could make the provision of internally published material (pre-prints, course materials etc.), which clearly falls within the remit of HeadLine and is often in Word format, a great deal simpler. This could be particularly important as the HTML created by the "Save as HTML" function of Office is not in fact conformant even to the HTML 2.0 standard (this is the case up to at least Office '97).⁷

DHTML

While XML is sufficiently different from HTML to force sites to migrate material from one format to the other, DHTML is designed (by Microsoft and Netscape in probably their last major collaborative venture) to extend the capabilities of HTML without changing the syntax with which developers are familiar [18]. The idea is basically to continue in the direction that HTML development has already taken – provide richer graphics capabilities and at the same time make fewer and faster page downloads possible. The idea that drives this, CSS (Cascading Style Sheets), is similar in concept to the XSL component of XML. As was the case with HTML development, the versions of DHTML implemented by the two companies in the version 4 browsers were significantly different and incompatible. The disagreement is sufficiently strong that XML is likely to be finalised before the less ambitious DHTML. (Some of the disagreement is due to the differing states of the official DHTML standard as the version 4 browsers were being released; the later Explorer is more compliant with the current ideas of the W3C.)

⁷ [Added during approval cycle] The initial draft of the next version of HTTP, known as HTTP-ng was released on 10.7.98 [15], too late for consideration for this paper.

World Wide Web Servers

HTTP Development

It is difficult to see any way in which HTML/XML can be superseded in the next few years; it has gained such a huge market share that any replacement would need to support it as a legacy system for the foreseeable future. Instead, HTTP is likely to be expanded to meet the perceived needs of the online community in much the same way as HTML has been. Over the last few years, HTML has developed far faster than HTTP, mainly because of the impetus provided by the changes introduced to HTML browsers by Netscape and Microsoft in advance of changes in the standard. The development of style sheets, dynamic HTML, XML and other metadata standards, and scripting languages (will) allow the Web author to do almost anything he/she might want to do in HTML. Thus, many of the remaining reasons why the Web has major limitations are to do with HTTP, including upgraded login and security, payment mechanisms and so on.

Ad-hoc developments in HTTP are not so easily carried out as changes to HTML; promoting new tags supported by your browser is one thing, but changing the way in which servers present pages involves much more work and co-operation with a browser manufacturer. Users are not willing to move to a system where reading Web pages on a Netscape server requires a Netscape browser; attempts by Web sites to promote one browser over the others are largely ignored by readers. Interoperability is one of the main strengths of the WWW, and will not be willingly abandoned. This is why HTML has developed so fast compared to HTTP; it has gone from version 1.0 to version 3.2 (with 4.0 on the way) in the time HTTP has moved from 0.9 to 1.1. The 1.1 specification has in fact been around for some time, but its implementation has proved slow. The current versions of both the Netscape Enterprise server (3.5.1) and the Apache server (1.3) fully support HTTP 1.1.

HTTP 1.0 was designed to be as simple as possible while supporting "more functionality than simple retrieval" [1, Section 1.1]; this functionality was provided by the ability to use forms as an interface to server-based programs. Statelessness is not just accidental; it was something inherited from earlier experimental versions of HTTP [*ibid.*, Section 1.3], where it was probably specified because of distrust of the integrity of Internet connections.

Each request to the server includes any authentication information, in unencrypted format [*ibid.*, Section 11.1]. This is acknowledged as insecure both for actual authentication and in that authentication information is passed over the network as clear text [*ibid.*, Section 12.1]. The need for persistent connections is acknowledged in the introduction to the HTTP 1.1 rfc [10, Section 1.1]. The HTTP 1.1 specification is open-ended; new authentication schemes can be used apart from the basic one outlined in the rfc [*ibid.*, Section 11]. The security flaws of the basic scheme are discussed at some length [*ibid.*, Section 15.1], particularly with respect to proxy servers. Further clarification is given in the P3P specification working draft, released by the W3C in May '98 [20]; here security is seen more to be content related than a data transport issue, which is why P3P is an XML specification rather than an extension to the HTTP standard.

P3P is designed to facilitate the passage of private data between Web browsers and servers in such a way that the browser and server negotiate the amount of sensitive information to be passed according to sites' security policies. This will mean that users can "tailor their relationship to specific sites" and "delegate decisions to their computer when appropriate" [20, Section 1]. One prediction for the future is that eventually every network transaction will be encrypted, though this will only happen over the long term because of political difficulties with the idea [28] –

governments want to be able to intercept and read material when they feel it is necessary to do so.

Many of the reasons that make the Internet currently unsuitable for commerce apply to the potential HeadLine system as well. Peter Van Kamp, president of CompuServe Network Services, described the problem [21]:

"What we really need is fast, reliable, secure communications and the ability to manage users' access to the enterprise system. This is not a reality now nor in the near future."

Replacing the word "enterprise" with "library", and you could use the sentence as a description of the problems facing HeadLine in its use of HTTP.

We could move as fast as HTTP develops, and it is certainly likely that Web servers in general will support the next generation of changes to that standard by the end of the project. The same issues, however, affect this as changes in HTML have affected designers over the last few years. Adopting the latest standards inevitably means leaving someone out. In this case, sites required to make expensive upgrades to proprietary Web servers will probably have the most lag, so we should certainly think hard about what is impossible with the current version of Notes, for example. Where we can make use of external developments that can be freely added on to existing HTTP 1.0 or 1.1 Web servers, such as suites of CGI scripts, it makes sense to do so.

Java and Web Programming

Web programming has been rather dominated by fads, which have encouraged serious programmers to act in a conservative way, ignoring the latest ideas (Java, push, etc.) unless they really have to use them. Some of these fashions have basically died before they really got going (push); others seem to be growing more and more popular (Java, despite its security flaws). The vision behind Java is incredibly attractive, even when presented with more than a hint of sarcasm [16]:

"Imagine a world without computer incompatibilities. You can write a program on one platform, then run it on a Sun workstation, a Mac, and a PC with any version of Windows. The cost of developing software plummets, software becomes cheaper to buy, and a new age of computing dawns. The lion lies down with the lamb. The land flows with milk and honey."

Java's big problem is the attitude of Microsoft. They seem to see Java as another mechanism for "luring" people into a Windows environment, by creating a version of Java that is tied back into the Windows operating system alone. (Apple is doing the same sort of thing with Rhapsody, but that is less important for the global computer market.) This is the reason for current legal challenges by Sun (who developed Java) against Microsoft using the term Java in connection with its Windows implementation of the language [16]. Gina Smith, ABC technology correspondent, thinks that Microsoft is too big for Sun to take on, and the case spells the end for Java [7], though most other commentators don't agree.⁸

Even in the early stages of the case, Sun were promising that users would not need to write ports for Java code so that they would run in Microsoft environments, though they were cagey about what that would mean exactly [8]. Microsoft were also keen to play down the importance of the

⁸ [Added during approval cycle] The initial hearing of this case has now been postponed for three months and will not take place until 4 September 1998. Both sides petitioned for postponement, as they felt unready to testify at the original date (July 31). This indicates that the case is likely to drag on for some time. See [32].

suit, saying that users should ignore them entirely. Independent analysts were rather more concerned, suggesting that user confusion and uncertainty were likely to slow Java implementation.

Server-side programming is not likely to go away (despite the developments discussed in the next section), because there are concerns about the security and speed of client side scripting. Java is becoming more and more the language of choice for this, though many people still want to hold onto Perl. Java is seen as a better alternative to C++, with its "automatic memory and easier thread management" reducing programming time and bugs. There are currently "technical, performance and hardware capacity issues that have stalled some efforts and complicated others" [17]. These problems are likely to disappear, particularly as Sun continues to develop the Enterprise JavaBeans specification for server-side applications. This is designed to bring in such important infrastructural requirements as transaction monitoring, though "many server-side enterprise requirements will remain outside the realm of formal JavaSoft APIs for months, even years to come" [*ibid.*]. Even when the specification is settled, implementations will still need to be built. In addition, "experienced pioneers warn that the choice of development tools remains crucial" [*ibid.*].

Apart from Java, the major competing technologies for Web scripting which offer event-driven responses beyond that of CGI (e.g. actions conditioned by mouse movement) are Netscape's JavaScript and Microsoft's ActiveX. These technologies are being subsumed into a new scripting standard by the ECMA. However, following the trend set by XML's separation of Web content and appearance, both companies have decided to create a mechanism to separate the Web content from scripting events (at the moment, they use small programs contained in the HTML coding). Both mechanisms, Microsoft's "Behaviors" and Netscape's "Action Sheets" (allowing scripting in any event-driven Web scripting language) will be supported by the version 5 browser released by the respective company. However, the technologies used, though based on existing technological standards (DHTML and CSS) are not compatible with each other, and so programmers and users face yet another choice dictated by continuing rivalry between the browser standards [30].

Client/Server Computing

Network Computing

This is something already being tried in a pilot scheme at BLPES, and provides a potential solution to many of the problems faced by system administrators. Maintaining and supporting a standard desktop is made very simple. But does such a system meet the desires of users?

“Whether you talk to desktop users or server administrators, you’ll find that most folks want more, not fewer, capabilities — and, as always, they want the systems to perform better while providing those capabilities. Intelligent client systems won’t go away, at least not for the vast majority of us” [25].

A crucial development in the intelligent/dumb terminal debate going on here is the possible developments in intelligent agents. “In the era of mass customisation we will ask our computers to represent our interests online” [25].

In general, on-line papers are dismissive of network computing as it is at the moment, but feel that it could play an important role in the future. Network computers “run applications more slowly than PCs, are largely unavailable, [and] have interoperability problems...But this will change soon” [34]. As the problems — particularly those of interoperability — are solved, take-up will increase. The network computer is listed as one of the technologies with no chance of success in [16], but this is a reaction to the idea of thin clients. Managers are apparently more interested now in installing fully functional PC client systems and putting the software that makes Network PCs easy to manage on them.

Naturally, no one wants to invest in a technology that is going nowhere. It shouldn’t be too long before it becomes clear whether network computing is here to stay.

“Network computers will either flourish or remain a niche product that mostly just replaces terminals. It’s been two-plus years, and it doesn’t usually take that long for technology to take off” (International Data Corp. analyst Eileen O’Brien, quoted in [8]).

Part of the reason for the widely differing opinions on the future of the network computer are due to some confusing issues of what the term refers to. The network PC is a specific piece of hardware, often referred to as a network computer, which is IBM’s thin client system. The network computer is a wider concept, taking in diverse thin client systems — just about every hardware manufacturer has a range — as well as fully functional clients used as terminals to access a central server (which is the model used in the BLPES pilot scheme).

Software Agents

One trend that has been forecast to play a major role on the future Internet is the use of “digital agents”, touted as “the backbone of the next generation of computer interfaces” [2]. Hyacinth S. Nwana is even more positive (quoted in [13]):

“Agents are here to stay...As we move further and further into the information age, any information-based organisation which does not invest in agent technology may be committing commercial hara-kiri.”

The increase in the amount of information available as the Internet has become generally accessible, and the greater opportunities for data discovery afforded by electronic information have not made it easier to discover the information required by a user. Traditional search methods require precise knowledge of all the material available to search, and such methods fail for large amounts of rapidly evolving information such as that on the Internet.

Agents are small programs that support a user with the accomplishment of some task or activity (user delegation)⁹. Hermans [13] lists two sets of characteristics of agents, one giving a “weak notion” of the agent concept, and consisting of the properties most researchers are agreed agents should exhibit, and the other giving a “strong notion”, containing more contentious properties. The weak properties of relevance to HeadLine include:

- autonomy (operation without direct intervention)
- social ability (interaction with other agents and humans)
- reactivity (perception of environment and response to changes in it)

The idea is that these programs are easy to relate to and will automate some of the initial data processing (e.g. which e-mail is urgent and which is to be left till later) that takes up lots of time for today’s user. Agents can also improve performance in some areas; in particular, the lack of sophistication in result sorting exhibited by current Web search engines comes under fire [14]. The information abundance is not just Web pages themselves; the number and size of online databases has also increased rapidly, and the problems studies have shown users encounter with these services are also candidates for the tasks agents are meant to be good at [40, 14]:

- finding appropriate search terms
- navigation of large quantities of hits and inability to re-phrase searches to reduce the retrieval sets
- avoiding zero hit searches
- failure to understand the cataloguing rules
- badly formatted searches (e.g. incorrect hyphenation, use of stop words, avoidance of field-based queries)

Typically, agents, or “bots” as they are sometimes more colloquially known, are to be used to filter large quantities of data and guide users. Other applications are beginning to be discussed, including virtual salespeople, robot butlers, shopping agents and so on. Agents have been touted as the future of computing for some years now, but nothing very much has come of them as yet. Reading between the lines of the information available seems to indicate that they are currently more like toys than commercial business products. As in previous years, researchers are predicting that the breakthrough will come at any moment [2]:

⁹ It is actually quite difficult to come up with a definition that fits in with the various ways in which the word has been used. One, due to G. W. Lecky-Thompson and quoted in [13], is:

“A piece of software which performs a given task using information gleaned from its environment to act in a suitable manner so as to complete the task successfully. The software should be able to adapt itself based on changes occurring in its environment, so that a change in circumstances will still yield the intended result.”

“Semi-intelligent software will soon be commonplace both online and on computer desktops, researchers predicted, used routinely to extract any useful information or service out of the Web’s chaos of information.”

Agents are seen as software responses to the changing nature of information [13]. The move toward electronic information means that more information is more easily available, and the small number of easily identifiable providers (publishers) has changed to a large number of extremely diverse potential information sources. Agents mean that access to information can become demand driven rather than supply driven.

Agents will not mean the end of the librarian in the near future [13, introduction]:

“Agents will be a highly necessary tool in the process of information supply and demand. However, agents will not yet be able to replace skilled human information intermediaries. In the forthcoming years their role will be that of a valuable personal assistant that can support all kinds of people with their information activities.”

Current research in agent development is aimed at producing modest, low-level applications. Because of this, agents are likely to become part of the software environment in a fairly stealthy manner [13, Chapter 2]:

“Users will not start to use agents because of their benevolence, proactivity or adaptivity, but because they like the ways agents help and support them in doing all kinds of tasks; soon users will use all sorts of convenient (i.e. “intelligent”) applications, without them realising they are using agents by doing so.”

Use of agents will be determined by two major factors, *trust* (the user is confident that the agent will do only what it is intended to do) and *competence* (the user is confident that the agent can do what it is intended to do).

Specific uses for which agents have been designed which may be of interest to HeadLine include:¹⁰

- Access and networking management.
- Information access and management – searching and filtering, categorisation, prioritisation, selective dissemination, annotation and sharing of information and documents.
- Adaptive user interfaces – system use monitoring, user activity modelling, adaptive “help-file”.

¹⁰ This is a summary of part of a list in [13].

CD-ROMs

The continued use of CD-ROMs has been considered to be assured for some years [33], following speculation that on-line systems would completely supercede them as sources of information [31]. This is because they provide an efficient mechanism for servicing applications requiring infrequent access to static information, as a secondary storage medium (one containing data which can be delivered to the user following a short delay). In fact, instead of competing with these media, the Internet "has resulted in unprecedented use of such services" [33]. As long ago as 1995, Bill Gates said "CD-ROM technology will continue to play a role in the content business for some time. But it is important to note that...the CD-ROM...and the Internet...will no longer be viewed as independent" [37]. The integration between the different storage methods has long been seen as the principal trend in the CD-ROM world, and there is no reason why this trend will not continue.

There is not a great deal of information available about the future of the CD-ROM. This is due to a combination of the feeling that existing goals will continue to be important with the fact that much CD-ROM development is corporate rather than academic, and is therefore kept secret.

Bibliography

- [1] Berners-Lee, T.; Fielding, R. & Frystyk, H.: *Hypertext Transfer Protocol – HTTP/1.0*. Internet RFC 1945, 5.96; <http://www.ics.uci.edu/pub/ietf/http/rfc1945.html>
- [2] Borland, John: *"Bots" Gaining Personality*, Biz Smarts. TechWeb, 18.5.98; <http://www.techweb.com/wire/story/TWB19980518S0017>
- [3] Bosak, Jon: *XML, Java and the Future of the Web*. World Wide Web Journal; Vol II No 4: XML: Principles, Tools & Techniques, 19.12.97. Excerpts available at <http://webreview.com/wr/pub/97/12/19/xml/index.html>
- [4] Boutin, Paul: *5.0 Browser Wish List Update*. Hot Wired Web Monkey, 4.98; <http://www.hotwired.com/webmonkey/98/16/index0a.html>
- [5] Bray, Tim; Paoli, Jean & Sperberg-McQueen, C.M.: *Extensible Markup Language (XML) 1.0*. W3C Recommendation, 10.2.1998; <http://www.w3.org/TR/1998/REC-xml-19980210>; current version <http://www.w3.org/TR/REC-xml>
- [6] Caruso, Denise: *'Baby Softs'? Imagining a breakup of Microsoft*. New York Times, 10.6.98; <http://thestate.com/technews/docs/10breakup.htm>
- [7] C-Net staff: *Forecast 1998 - Smart People Predict This Year's Technology*. C-Net Digital Life feature article, 5.1.98; <http://www.cnet.com/Content/Features/Dlife/Forecast98/index.html>
- [8] Computerworld staff: *Forecast 1998*. 27.12.97; <http://www.computerworld.com/forecast98/971229toc.html>
- [9] Elhaug, Einer: *Microsoft Gets an Undeserved Break*. New York Times, 29.6.98.
- [10] Fielding, R.; Gettys, J.; Mogul, J.; Frystyk, H. & Berners-Lee, T.: *Hypertext Transfer Protocol – HTTP/1.1*. Internet RFC 2068, 1.97; <http://www.cis.ohio-state.edu/htbin/rfc/rfc2068.html>
- [11] Flynn, Peter *et al.*: *Frequently Asked Questions about the Extensible Markup Language (The XML FAQ)*. Version 1.3, 1.6.98; <http://www.ucc.ie/xml/>
- [12] Goodin, Dan: *DOJ explores curbs on Microsoft*. C-Net news.com news analysis article, 3.4.98; <http://www.news.com/News/Item/0%2C4%2C20770%2C00.html>
- [13] Hermans, Björn: *Intelligent Software Agents on the Internet: An Inventory of Currently Offered Functionality in the Information Society and a Prediction of (Near) Future Developments*. First Monday, Vol II No 3, 11.4.97; http://www.firstmonday.dk/issues/issue2_3/ch_123/
- [14] Jansen, James: *Using an Intelligent Agent to Enhance Search Engine Performance*. First Monday, Vol. II No. 3, 11.4.97; http://www.firstmonday.dk/issues/issue2_3/jansen/
- [15] Janssen, Bill; Nielsen, Henrik Frystyk; and Spreitzer, Mike: *HTTP-ng Architectural Model*. W3C Working Draft 10.7.98; <http://www.w3.org/TR/1998/WD-HTTP-NG-architecture-19980710>; latest version <http://www.w3.org/TR/WD-HTTP-NG-architecture>
- [16] Lake, Matthew: *10 Technologies That Don't Stand a Chance*. C-Net Digital Life feature article, 16.10.97; <http://www.cnet.com/Content/Features/Dlife/Failure/index.html>
- [17] Levin, Rich: *Server-Side Java Takes Off*, TechWeb 2.4.98; <http://www.techweb.com/wire/story/TWB19980402S0010>
- [18] Mace, Scott; Flohr, Udo; Dobson, Rick & Graham, Tony: *Weaving a Better Web. What's next?* Byte cover article, Mar 98; <http://www.byte.com/art/9803/sec5/art1.htm>
- [19] Maclachlan, Malcolm: *Andreessen Sees Mozilla-Linux Upset of Windows*. TechWeb, 2.4.98; <http://www.techweb.com/wire/story/TWB19980402S0013>

- [20] Marchiori, Massimo & Jaye, Dan: *Platform for Privacy Preferences (P3P) Syntax Specification*. W3C Working Draft, 19.5.98; <http://www.w3.org/TR/1998/WD-P3P10-syntax-19980519.html>; current version <http://www.w3.org/TR/WD-P3P10-syntax>
- [21] Mateyaschuk, Jennifer: *CompuServe Exec: Internet Is Not Enterprise-Ready*. Information Week, 21.4.98; <http://www.techweb.com/wire/story/TWB19980421S0002>
- [22] Moeller, Michael: *Analysts Pick Over MS Victory*. PC Week Online, 23.6.98; http://www.zdnet.com/zdnn/stories/zdnn_lggraph_display/0,3442,331693,00.html
- [23] Morgenthal, J. P.: *Keeping Up With the Experts' Predictions*. Internet Week, 9.3.98; <http://techweb.cmp.com/internetwk/columns/logic0309.htm>
- [24] Mosquera, Mary: *Fathers of Net Gush About Their Baby* (report of press conference given by Vint Cerf and Robert Kahn, developers of TCP/IP). The Internet Daily, 23.3.98; <http://www.techweb.com/wire/story/TWB19980323S0022>
- [25] Name, Mark L. van & Catchings, Bill: *Think Distributed, Not Client or Server*. Internet Computing, 13.3.98; <http://www.zdnet.com/icom/content/columns/1998/03/think.dist/index.html>
- [26] *NetLingo: The Internet Language Dictionary*; <http://www.netlingo.com>
- [27] Nielsen, Jakob: *Changes in Web Usability Since 1994*. Alertbox, 1.12.97; <http://www.useit.com/alertbox/9712a.html>
- [28] Nielsen, Jakob: *Predictions for the Web in 1998*. Alertbox, 1.1.98; <http://www.useit.com/alertbox/980101.html>
- [29] Phillips, Paul: *Why Mozilla Matters*. 3.4.88; <http://www.mozilla.org/why-mozilla-matters.html>
- [30] Rein, Lisa: *Browser Battles Script On*. Wired News, 8.7.98; <http://www.wired.com/news/news/technology/story/13546.html>
- [31] Reisman, Richard R.: *All Power to the Web – CD-ROM is Dead – or is it?* New York New Media Association News, 8.96; <http://www.teleshuttle.com/tsdistrib.htm>
- [32] Reuters news wire: *Java Court Date Pushed Back*. 7.7.98; <http://www.news.com/News/Item/Textonly/0,25,23941,00.html?pfv>
- [33] Smart Storage white paper: *CD Technology: The Standard for Secondary Storage*. SIGART archive, 1996; <http://www.sigcat.org/articles/white/smart1.htm>
- [34] Taschek, John: *Is There a Future for Network Computers*. PC Week, 13.3.98; <http://www.zdnet.com/zdnn/content/pcwk/1511/293417.html>
- [35] Tran, Mark: *Microsoft Charged with Attempt to Crush Rival*. Guardian, 20.5.98
- [36] Trott, Bob: *Judge looking for tight Microsoft trial*. Info World Electric, 9.6.98; <http://www.infoworld.com/cgi-bin/displayStory.pl?98069.echearing.htm>
- [37] Vieux, Alex S.: *Interview with Bill Gates*. The Red Herring, 11.95.
- [38] Wiggins, Richard: *XML: What Every Webmaster Should Know* (Interview with Peter Flynn, co-author of the XML FAQ). Net Buzz Vol. 1 No. 19, 18.3.98; <http://webreference.com/outlook/column19/>
- [39] Wilke, John R.: *Officials Think Evidence Supports a Broad Charge on Extending Monopoly*. Wall Street Journal, 5.4.98
- [40] Yee, Martha M.: *System Design and Cataloging Meet the User: User Interfaces to On-line Public Access Catalogs*. Journal of the American Society for Information Science, vol. 42, no. 2, pp. 78-98, 1991

For more information - see the Document Standards document