The World-Wide Web as the Universal Interface to the NII[*]

Henning Schulzrinne
GMD Fokus
Hardenbergplatz 2
10623 Berlin
email: schulzrinne@fokus.gmd.de

July 24, 1996

1 Introduction

In a time span of about five years, the World-Wide Web (WWW) [2] has become, next to electronic mail, the most popular Internet application. It has been a major contributor in turning the Internet from an obscure data network for scientists and computer programmers to a household word. The World-Wide Web allows to retrieve text and multimedia objects from servers located throughout the world, with objects connected by hypermedia links. This article aims to provide a snapshot of the World-Wide Web after about half a decade, speculating at the same time where this young medium might be improved and which directions it might take from a technical perspective. Particular attention will be paid to how WWW-derived technology might be used to allow non-specialists access to a wide variety of services.

Like most (successful) Internet technologies, the underlying central functionality of the World Wide Web (WWW) is rather simple: a naming mechanism for files (the universal resource locator or URL), a typed, stateless retrieval protocol (HTTP) and a minimal formatting language with hyperlinks (HTML). All of the basic technologies were around prior to the "invention" of the world-wide web, generally credited to Tim Berners-Lee and Robert Cailliau at CERN around 1989[1]. However, the major accomplishment was not an individual protocol, but rather the integration of disparate pieces into a new, more powerful way of using networks. However, only after the original text-only browser was replaced by one with a graphical user interface (Mosaic from the National Center for Supercomputer Applications at the University of Illinois), did the world-wide web really take off. While it was originally conceived to integrate existing retrieval and access mechanisms, in particular, the file transfer protocol (ftp), gopher as a menu-oriented retrieval system, and telnet for remote login and interaction with databases, the core WWW protocol (http) has far surpassed usage of all three of these. There are other reasons for the rapid proliferation of WWW, making its rise, in hindsight, a bit less surprising than R. Lucky describes [3]: The technology works reasonably well for access speeds from 2400 baud modems on up since it retrievals can be restricted to text only, and newer Internet users, accustomed to graphical user interfaces, are far less tolerant of the command-based interfaces of traditional retrieval mechanisms like ftp or telnet. Also, in the early 90's, the lowest common denominator computing platform shifted from terminals and DOS-based PCs to PCs running windowing systems, X-terminals and workstations, allowing rapid uptake of the WWW-based multimedia content, while the basic functionality remained accessible to those still restricted to ASCII. Also, the Internet itself did not have

[*] Note: This article is based on [1].
[1] See http://www.w3.org/hypertext/WWW/History.html for a timeline of WWW developments.

to offer any new capabilities or "service models", beyond a reliable domain name system. Since WWW retrievals are TCP-based, they share the available bandwidth reasonably fairly, and require no new resource allocation mechanisms in the network. Finally, the cost of entry for "consumers" and "providers" alike was extremely low, because the software was (and is) largely free and Web usage, for corporate and university users, incurs no additional network charges. By now, the sheer volume of WWW transfers has become one of the major forces for the increasing congestion in many areas of the Internet, particularly in Europe and the trans-atlantic connections.

This background paper will try to present a survey of some of the open areas within the WWW framework, both those which are the subject of on-going work and those which may impose longer-term fundamental limitations on the WWW. First, the paper outlines some requirements for generic access to the NII and how the world-wide web fits the requirements in Section 2. Section 3 investigates the role of the browser and how it might become a central part of the operating system. The particular problems of using WWW technology to interface to legacy communications and information systems is touched upon in Section 4. Section 5 points out some more long-term limitations of the WWW model and how other applications could be integrated with the Web. The final section summarizes some of the new applications and alternatives for information delivery that might be viable in the near term. This article does not discuss the important topic of how to organize the resources that can be accessed via the world-wide web; Lynch [4] offers a survey of these more generic issues. Additional perspectives on HTML, HTTP and URLs are contained in [1].

2 Requirements for a Universal Interface

WWW technologies have the potential to be the primary interface to the national information infrastructure, allowing every citizen to have access to information and communication resources

- at any time,
- from any location,
- at low cost,
- reliably,
- regardless of physical handicap,
- while respecting rights of privacy and information autonomy.

However, limitations in current WWW-related technologies and the Internet infrastructure let them fall short of these requirements, as elaborated more fully later.

2.1 Independence of Time and Location

Since the underlying technology is rather simple, web browsers and servers can be built for just about any computing platform and can be integrated into devices not generally considered as networked computers. Currently, web browsers are available for just about any operating system in traditional PCs, workstations or time-sharing terminals, but also appear (or will shortly) in set-top boxes for TV sets[2], information kiosks, screen phones, pay phones and mobile phones or the entertainment systems in the backs of airline and train seats. It is likely that in the near future, personal digital assistants will have built-in web browsers, as well as hotel cable systems or public transport information systems[3] Thus, The ubiquity of web access will also

[2]as a more flexible alternative to the Videotext service popular in Europe

[3]This has been proposed for a German city.

significantly lower the cost of entry, as the existence of public terminals makes it less necessary to own, install and maintain a computer. In many cases, this may also be more convenient than carrying one's own portable computer.

However, public-access web terminals raise a number of technical and policy problems. A user will expect to be able to access information from her company's internal database or her own personal computer at home, not just on-line newspapers. There has to be a means of identifying oneself to information providers. Clearly, the user of such public terminals has to have some assurance that the personal data viewed is not abused by the owner of the web terminal or other parties that may gain access to it.

Ubiquitous access also implies a large diversity of end systems, with widely varying display capabilities. Fortunately, HTML can, in principle, be rendered on anything from a display capable of showing only a few lines of text, to a high-resolution graphics display, However, means beyond a link to a text-only version are needed to tailor the rendition of information to the capabilities of the display device and the available network bandwidth.

2.2 Low Cost

Having a single information delivery mechanism such as the WWW and a uniform network infrastructure, the Internet, can greatly reduce the incremental cost of providing additional information services – this is probably one of the central reasons for its rapid spread. It thus seems likely that private information systems such as automatic teller machines or airline reservation systems will eventually migrate to using a (restricted) web browser.

Currently, a large fraction of the cost of providing content is borne by advertisers, just like for most other non-book media in the United States. The web also allows a different model: A company direct generates or hosts information resources related to its products. A pharmaceutical company, say, could offer general health information on its web pages, as a means to attract visitors to the advertising content.

While many of the applications and innovations are currently driven by commercial interests, the lowered costs of infrastructure and software will also greatly facilitate non-commercial applications, including the interaction of citizens with their government.

[?]However, while information delivery is global, many advertisers would like to restrict the reach of their ads to a certain geographic region. Since IP addresses and domain names often do not indicate the geographic location of the web visitor, provision of such information would be useful to advertisers, yet not significantly affect privacy.

2.3 Universal Access

HTML as "lingua franca" of the web offers the advantage that it is amenable to being rendered not just as on a video display, but also being spoken by text-to-speech engines or translated into Braille. Also, unlike other word processor formats, it can be easily printed or displayed at any font size, making it accessible to those with poor eyesight. The structural information about headings, lists and tables allow for selective rendering, without, say, having to convert the whole text of a web page to speech sequentially. Images can be described with alternate, textual tags. Efforts to allow font hints and font downloading lessen the temptation to show text as bitmaps rather than HTML. However, other current trends in page design and HTML extensions impede non-graphical rendering of web pages. Imagemaps and frames, for example, make sequential rendering to non-graphical output devices very difficult. Tools are needed that allow a designer to create content, which is then automatically converted to both the "house style" and a "linear" style suitable for, say, conversion to Braille or text-to-speech converters. WWW protocols need to be extended so that a visitor to a page can automatically declare his preference for styles of presentation, without having to locate a "text only" switch somewhere on the web page.

2.4 Secure Access

As mentioned above, truly ubiquitous access to information resources also includes the ability to reach proprietary, private or for-fee information across untrusted networks, potentially from abroad. The the basic web protocol suite offers sufficient privacy protection through such protocols as Secure Socket Layer (SSL) or Secure HTTP (SHTTP). However, security in a global network requires a (public) key distribution infrastructure, with unsolved legal and technical problems [5].

It should be noted that the wide distribution of web tools with built-in security offers the hope that at least protocol flaws are quickly discovered. Unfortunately, as experience has shown, most security problems are related not to theoretical weaknesses in the encryption algorithms or transport protocols, but rather faulty implementations. Here, only the widespread availability of source code for security-related functions is likely to help, particularly for browsers on less popular platforms.

2.5 Reliability

For people who are interested in information or entertainment and are not necessarily drawn to the Internet by fascination with technical detail, the current state of the Internet can be rather frustrating, as individual sites or large parts of the Internet can become inaccessible or data transfer speeds fall to a few bytes a second. On a "real" high-way, the motorist stuck in traffic can see the large number of other automobiles or the detour signs, on the Internet, the layman has few reliable clues to distinguish local failures from, say, Internet congestion or temporary domain service unavailability.

3 Browsers

3.1 Development

One of the factors driving the success of the WWW is its ability to attract both content providers and serve as a base for new applications. While in the past, a corporate library may have written its own user interface to its library catalogue, it now appears much easier to base this on a web server and browsers. This avoids having to write a new user interface for each new client platform or operating system, and automatically lets the system participate in advances like security.

There seem to be two contradictory directions for WWW applications: the browser that can do everything and having every application have WWW capabilities. The latter makes it difficult to integrate several data types, but it is certainly desirable to be able to have applications recognize URLs and control a browser to view the URL content. Browsers are already incorporating mail tools, news readers and very primitive file system managers. Soon, they will also feature text editors, at least for HTML. While this integration has the advantage of hiding the difference between local and remote operations to a large extent, it also leads to huge applications and less choice between vendors. Other mechanisms to integrate different applications are currently being deployed, for example, so-called plug-ins, where applications communicate directly with the browser and share some of its window area. However, plug-ins are generally being written only for "popular" operating systems and hardware platforms. As the number of platforms supporting web browsers is going to grow, platform-independent approaches to local program execution seem preferable, in particular platform-independent languages like Java. Downloadable code like Java is also more in the spirit of the web as a transparent storage hierarchy, as they require no downloading and installation. This will only be possible if Java programs can be made to execute sufficiently fast by local compilation.

3.2 A Web-Based View of Computing

Since the 1960's, the dominant metaphor and organizing principle for computer systems has been the hierarchical file system, with directories ("folders") containing files. Various forms of links then create acyclic graphs rather than just trees. There was only a single view for that file system. A single directory or even "drive" was always constrained to reside on a single physical system, with access only semi-transparent in the local area. Except in the Macintosh OS and through crude filename mechanisms, files had no type information. The introduction of Internet web browsers first complicates the picture: in addition to a hierarchical file system, it adds a cross-linked web page structure and a separate bookmark hierarchy. Finding information (and not just a filename) within a large file system is tedious, particularly if it contains objects (files) of different types.

A web-based computer interface offers the opportunity to present a uniform view of both local and remote resources. Instead of a hierarchy of folders and files, objects could be presented organized in various ways, be it metaphors like tables of contents and indices, or loose-form notebooks and topic-specific graphics. Some of these representations might be generated automatically, some crafted by hand. For example, instead of filing correspondence into either a chronological or addressee system, a letter might appear as a link on a map, an orgchart, a personal address book, and a calendar.

Client-based caching is another important aspect of a web-based system. With client-based caching, objects are retrieved for display or execution from a server, but stored locally for faster subsequent retrievals or execution. It can span the gamut between (X) terminal-like system with no local storage, with every interaction going across the network, to a "traditional" stand-alone PC or workstation, where all object references are to local storage. Indeed, the same physical system can assume different roles during its lifecycle. This approach removes the artificial difference between documents and programs, applying the same authentication, encryption and charging mechanism to all. It also avoids the need to explicitly install and uninstall software. This makes client-based caching particularly appropriate for general-audience computing and network devices. Separating the linking of documents from traditional file systems also provides for richer possibilities of semantically linked documents, for example, versions in multiple languages or multiple revisions.

A number of practical problems remain to be solved until a caching-based system is viable: First, the network has to be fast, cheap and reliable enough so that cache entries that are no longer local can be retrieved without unbearable waits. Also, many users are not likely to be willing to pay for each retrieval independently, so that the server has to track client identities. Having multiple references to the same object, as described above, poses particular problems with the one-directional nature of URLs, as the object being linked has no ability to determine which pointers exist to it.

It is now often easier to find something by content on the Internet compared to one's own local file system. A web-based computing system as envisioned here would incorporate one or more search engines to provide a uniform view of local and Internet resources, enhanced with the ability to select based on trust and authorship, "freshness" and accessibility. (Current search engines will only show what is accessible to the search engine, which is likely going to be very different once a search encompasses not just the "public" Internet but the private files.)

4 Access to Legacy Information and Communication Systems

WWW servers can readily make older information systems available to a larger audience, as shown by implementations of interfaces to airline reservation systems, train schedule services, or FAA flight databases, to cite just the travel-related examples. Also, WWW browsers and email can easily serve as interfaces to older modes of communication, such as paging, fax and telex. Here, the browser offers an improved user interface and the Internet can provide a more efficient transport of these data services across the wide area. So far, there has been little integration of web browsers and telephony, although these would offer obvious benefits. As an example, a customer seeking technical support could navigate through a set of web pages on the manufacturer's web site and then, with some straightforward extensions to the browser, press a button to automatically dial the correct extension, possibly with a description of the current set-up already available to the customer service representative. These types of integration, however, have to wait until switching between voice and data use of the same phone line becomes easier, Internet access is provided by means other than the telephone line or until Internet telephony, possibly with gateways to the phone network, is more usable.

Other opportunities to enhance telephony services through web browsers are being explored [11], for example to configure telephone service features such as call forwarding or number blocking.

5 Limitations of the WWW Model

Despite all the press and publicity, the WWW model is currently rather limited: retrieve an object (text, audio or video) and render it. Even with forms, the capabilities of a Web page are roughly that of a page-oriented mainframe terminal, with some graphics spice added. Some inherent capabilities of the web model are just being developed, in particular the ability to store and edit content on the server or annotate existing documents from within a browser. This could be quite useful for collaboration and for maintaining corporate information within intranets[4] and across fire walls, particularly once client authentication is better developed.

Client-side interaction is currently limited to filling out simple forms and clicking on buttons and bitmaps (so-called image maps). There are some efforts to provide more direct feedback to the user rather than having to fill out the whole form and then being told that some field is wrong or clicking on parts of a bit map without any feedback as to what, if anything, might happen. Client-side imagemaps store the coordinates of sensitive areas so that the browser can provide local feedback. Client-side scripts or applets would allow to provision of interactive help or correctness checks. These could also provide richer user interfaces, so that the content of a page can adapt to user action rather than simply by reloading a page from the server.

The integration of multimedia is currently very primitive: A video or audio file is transferred via HTTP and then played with some sufficient buffering or from local temporary storage. Playing audio and video as it arrives from the network avoids waiting minutes for it to download completely (only to discover that it probably was not worth the wait). However, unless the minimum network bandwidth is the access bandwidth, assumed known, the user has no way to choose an appropriate encoding or know ahead of time how long the media content needs to be buffered to assure playout without interruption. A number of solutions can be envisioned. First, non-TCP protocols such as Realtime Transport Protocol (RTP) [6] that provide congestion feedback can be used to adaptively tune their buffering and encoding to the available bit rate. In addition, for both TCP and other protocols, resource reservation could guarantee a minimum bandwidth [7]. Adaptive applications require no changes to the Internet, but are still subject to glitches when adapting quality downward. RSVP, the IETF Internet resource reservation protocol [7, 8], will likely not be deployed widely for a number of years, as it requires router modifications, some form of authentication and usage-based billing to prevent abuse.

Any type of interactive games (often called "twitch" games) will likely bypass the web protocols completely, even though they might use it to locate services or partners to play with. The same is also true for multimedia-on-demand applications. The existing media-on-demand applications use the WWW browser only to present some VCR-style controls and a play list. There is no possibility of embedding "active areas" (i.e., hyperlinks) within the audio or video content or synchronize textual and video content.

For content delivery, the web is "pull-only", that is, the browser has to explicitly check whether there is new content. There are some HTML extensions where the browser can be instructed to do this at preset

[4] Intranets are networks within organizations that may or may not be connected to the Internet, but use some of the same technology and protocols.

intervals. Also, there are tools that gather a predefined set of pages for later (off-line) perusal, as well as services that regularly poll web pages for changes and then send electronic mail to the subscribers of this page. This works well only for pages that change infrequently and at predictable times. An efficient, multi-destination hybrid between messaging (i.e., electronic mail) and WWW presentation is required.

Web browsing is a solitary occupation. A visitor to a web page cannot see if there are other like-minded people reading the same page or interacting with the same 3D-world. There have been a number of proposals to create mutual awareness. The *Virtual Places* [9] protocol uses a separate server to track visitors, represented by small cartoons or pictures displayed on a custom browser. A visitor sees the movement of others on the page and can move her image to touch that of another visitor. As images touch, a packet voice connection is established. Tour buses can be used to visit web sites together, with "passengers" on the bus reachable by audio and video through IP multicast. People that are on the same page can also "chat" with each other by typing messages. However, all of these interactions only use the browser as a platform and bypass WWW protocols. Such web-based interactions might also be feasible by using scripting languages and multicast notifications of presence and location. This offers an enhanced version of the room metaphor employed in some video conferencing systems [10].

Chat rooms are quite popular on on-line services and there have been attempts to use WWW technology. Without some external protocols, this is rather clumsy, as a user would have to reload a page to see what, if anything, others have typed.

6 Conclusion

The success of the World Wide Web has, at least in the eyes of the non-technical public, made it and the Internet nearly synonymous. However, there are many interesting Internet-based services which are largely independent of the Web, and have quite different requirements. As an example, real-time, interactive multimedia services would provide an alternative to the telephone network, immediately offering many of the services promised by the telephone carriers through Advanced Intelligent Networks (AIN), but they require extensive additions to network infrastructure, in particular some form of resource reservation and significantly enhanced bandwidth. A Web-centric Internet would be giving up one of the fundamental strengths of the Internet: the ability to quickly deploy new services.

Overall, the continued growth success of the world-wide web as a global delivery mechanism for multimedia content will ride on technical issues as much as on economic, social and political ones. Some questions that we can only raise here include: Can an advertising-only medium prosper or will there be ways (and willingness) to pay for quality content? Will jurisdictions force Internet service providers to restrict access to parts of the web which are considered in violation of local criminal statutes? Will encryption and user certification be widespread and sufficiently easy to use that electronic commerce can prosper?

References

[1] H. Schulzrinne, "World-wide web: Whence, whither, what next?," *IEEE Network*, vol. 10, March/April 1996.

[2] T. Berners-Lee, R. Cailliau, A. Luotonen, H. F. Nielsen, and A. Secret, "The world-wide web," *Communications ACM*, vol. 37, pp. 76–82, Aug. 1994.

[3] R. W. Lucky, "Where die the web come from?," *IEEE Spectrum*, vol. 32, p. 15, July 1995.

[4] C. A. Lynch, "Networked information resource discovery: An overview of current issues," *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 1505–1522, Oct. 1995.

[5] N. R. C. Committee to Study National Cryptography Policy, *Cryptography's Role in Securing the Information Society*. Washington, DC: National Academy Press, July 1996.

[6] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: a transport protocol for real-time applications," RFC 1889, Internet Engineering Task Force, Jan. 1996.

[7] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "RSVP: a new resource ReSerVation protocol," *IEEE Network*, vol. 7, pp. 8–18, Sept. 1993.

[8] R. Braden, L. Zhang, and S. Berson, "Resource reservation protocol (RSVP) – version 1 functional specification," Internet Draft, Internet Engineering Task Force, Nov. 1995. Work in progress.

[9] E. Shapiro, "Virtual places – a foundation for human interaction," in *Proc. of the Second World Wide Web Conference'94*, (Chicago, Illinois), Oct. 1994.

[10] S. R. Ahuja and J. R. Ensor, "Call and connection management: making desktop conferencing systems a real service," *ACM Computer Communication Review*, vol. 22, pp. 10–11, Mar. 1992.

[11] C. Low, "The internet telephony red herring," in *Proc. of Global Internet'96*, (London, England), Nov. 1996.