

Tim Berners-Lee, Robert Cailliau, Ari Luotonen, Henrik Frystyk Nielsen, and Arthur Secret

The World-Wide Web

The World-Wide Web (W3) was developed to be a pool of human knowledge, which would allow collaborators in remote sites to share their ideas and all aspects of a common project. Physicists and engineers at CERN, the European Particle Physics Laboratory in Geneva, Switzerland, collaborate with many other institutes to build the software and hardware for high-energy physics research. The idea of the Web was prompted by positive experience of a small “home-brew” personal hypertext system used for keeping track of personal information on a distributed project. The Web was designed so that if it was used independently for two projects, and later relationships were found between the projects, then no major or centralized changes would have to be made, but the information could smoothly reshape to represent the new state of knowledge. This property of scaling has allowed the Web to expand rapidly from its origins at CERN across the Internet irrespective of boundaries of nations or disciplines.

If you haven't yet experienced the Web, the best way to find out about it is to try it. An Appendix to this article gives some recipes for getting hold of W3 clients. Given one of these, you will quickly find out all you need to know, and much more. For hard copy to read on the plane, or if you don't have Internet access from your desktop machine, refer to our paper in *Electronic Networking* (see “Glossary and Further Reading”) for an overview of the project, material which we will not repeat but will summarize here.

A W3 “client” program runs on your computer. When it starts, it displays an object, normally a document with text and possibly images. Some of the phrases and images are highlighted: in blue, or boxed, or perhaps numbered, depending on what sort of a display you have and how your

(“anchor”) causes the client program to retrieve another object from some other computer, a “server.” The retrieved object is normally also in a hypertext format, so the process of navigation continues (see Figure 1).

When viewing some documents, the reader can request a search, by typing in plain text (or complex commands) to send to the server, rather than following a link. In either case, the client sends a request off to the server, often a completely different machine in some other part of the world, and within (typically) a second, the related information, in either hypertext, plain text or multimedia format, is presented. This is done repeatedly, and by a sequence of selections and searches one can find anything that is “out there.” Some important things to note are:

- Whatever type of server, the user interface is the same, so users do not need to understand the differences between the many protocols in common use. Before W3, access to networked information typically involved knowledge of many different access “recipes” for different systems, and a different command language for each. The model of hypertext with text input has proved sufficiently powerful to express all the user interfaces, while being sufficiently simple to require no training for a computer user.
- Links can point to anything that can be displayed, including search result lists. (When a query is applied to an object, the resulting object has an address, defined to be the address of the queried object concatenated with the text of the query. As the result object has an address, one can make links to it. Following the link later leads to a reevaluation of the query.)
- While menus and directories are available, the extra option of hypertext provides a more powerful communications tool. In simple cases, the

ple) the directory structure of an existing file store. This allows existing data to be put “on the Web” without further human effort.

- There is a very extendable system for introducing new formats for multimedia data.
- There are many W3 client programs. As hypertext information is transmitted on the network in logical (mark-up) form, each client can interpret this in a way natural for the given platform, making optimal use of fonts, colors, and other human interface resources available on that platform.

What Does W3 Define?

W3 has come to stand for a number of things, which should be distinguished. These include

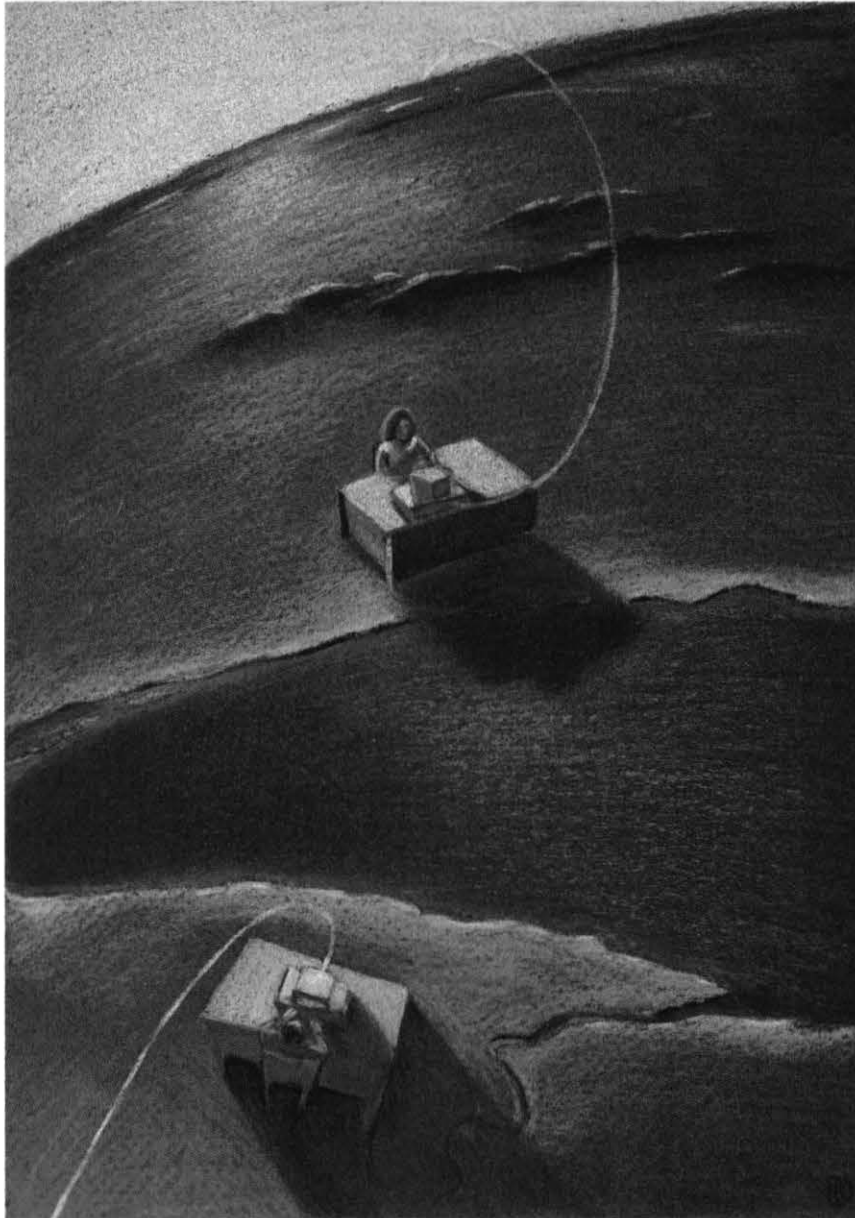
- The idea of a boundless information world in which all items have a reference by which they can be retrieved;
- The address system (URI) which the project implemented to make this world possible, despite many different protocols;
- A network protocol (HTTP) used by native W3 servers giving performance and features not otherwise available;
- A markup language (HTML) which every W3 client is required to understand, and is used for the transmission of basic things such as text, menus and simple on-line help information across the net;
- The body of data available on the Internet using all or some of the preceding listed items.

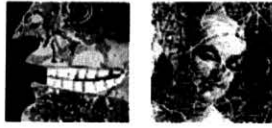
The client-server architecture of the Web is illustrated in Figure 2.

Universal Resource Identifiers

Universal Resource Identifiers¹ (URIs) are the strings used as ad-

¹The Internet Engineering Task Force (IETF) is currently defining a similar and derived syntax known as a Uniform Resource Locator (URL). As this work is not complete, and there is no guarantee that URIs will have the same syntax





dresses of objects (e.g., menus, documents, images) on the Web. For example, the URI of the main page for the WWW project happens to be

<http://info.cern.ch/hypertext/WWW/TheProject.html>

URIs are “Universal” in that they encode members of the universal set of network addresses. For a new network protocol that has some concept of object, one can form an address for any object as the set of protocol parameters necessary to access the object. If these parameters are encoded into a concise string, with a prefix to identify the protocol and encoding, one has a new URI scheme. There are URIs for Internet news articles and newsgroups (the NNTP protocol), and for FTP archives, for telnet destinations, email addresses, and so on. The same can be done for names of objects in a given name space.

The prefix “http” in the preceding example indicates the address space, and defines the interpretation of the rest of the string. The HTTP protocol is to be used, so the string contains the address of the server to be contacted, and a substring to be passed to the server. Different protocols use different syntaxes, but there is a small amount of common syntax. For example, the common URI syntax reserves the “/” as a way of representing a hierarchical space, and “?” as a separator between the address of an object and a query operation applied to it. As these forms recur in several information systems, to allow expression of them in the common syntax allows the features to be retained in the common model, where appropriate. Hierarchical forms are useful for hypertext, where one “work” may be split up into many interlinked documents. Relative names exploit the hierarchical structure and allow links to be made within the work independent of the higher parts of the URI such as the server name.

URI syntax allows objects to be addressed not only using HTTP, but also using the other common networked information protocols in use today (FTP, NNTP, Gopher, and WAIS), and will allow extension when new protocols are developed.

ture. The fact that it is easy to address an object anywhere on the Internet is essential for the system to scale, and for the information space to be independent of the network and server topology.

Hypertext Transfer Protocol

Perhaps misnamed, rather than being a protocol for transferring hypertext, HTTP is a protocol for transferring information with the efficiency necessary for making hypertext jumps. The data transferred may be plain text, hypertext, images, or anything else.

When a user browses the Web, objects are retrieved in rapid succession from often widely dispersed servers. For small documents, the limitations to the response time stem mainly from the number of round trip delays across the network necessary before the rendition of the object can be started. HTTP is therefore a simple request/response protocol.

HTTP does not only transfer HTML documents. Although HTML comprehension is required of W3 clients, HTTP is used for retrieving documents in an unbounded and extensible set of formats. To achieve this, the client sends a (weighted) list of the formats it can handle, and the server replies with data in any of those formats that it can produce. This allows proprietary formats to be used between consenting programs in private, without the need for standardization of those formats. This is important both for high-end users who share data in sophisticated forms, and also as a hook for formats that have yet to be invented. The same negotiation system is used for natural language (English, French, for example) where available, as well as for compression forms.

HTTP is an Internet protocol. It is similar in its readable, text-based style to the File Transfer (FTP) and Network News (NNTP) Protocols that have been used to transfer files and news on the Internet for many years. Unlike these protocols, however, HTTP, is stateless. (That is, it runs over a TCP connection that is held only for the duration of one operation.) The stateless model is efficient when a link from one object may lead

same server, or to another distant server. The purpose of a reference such as a URI is that it should always refer to the “same” (in some sense) object. This also makes a stateless protocol appropriate, as it returns results based on the URI but irrelevant of any previous operations performed by the client.

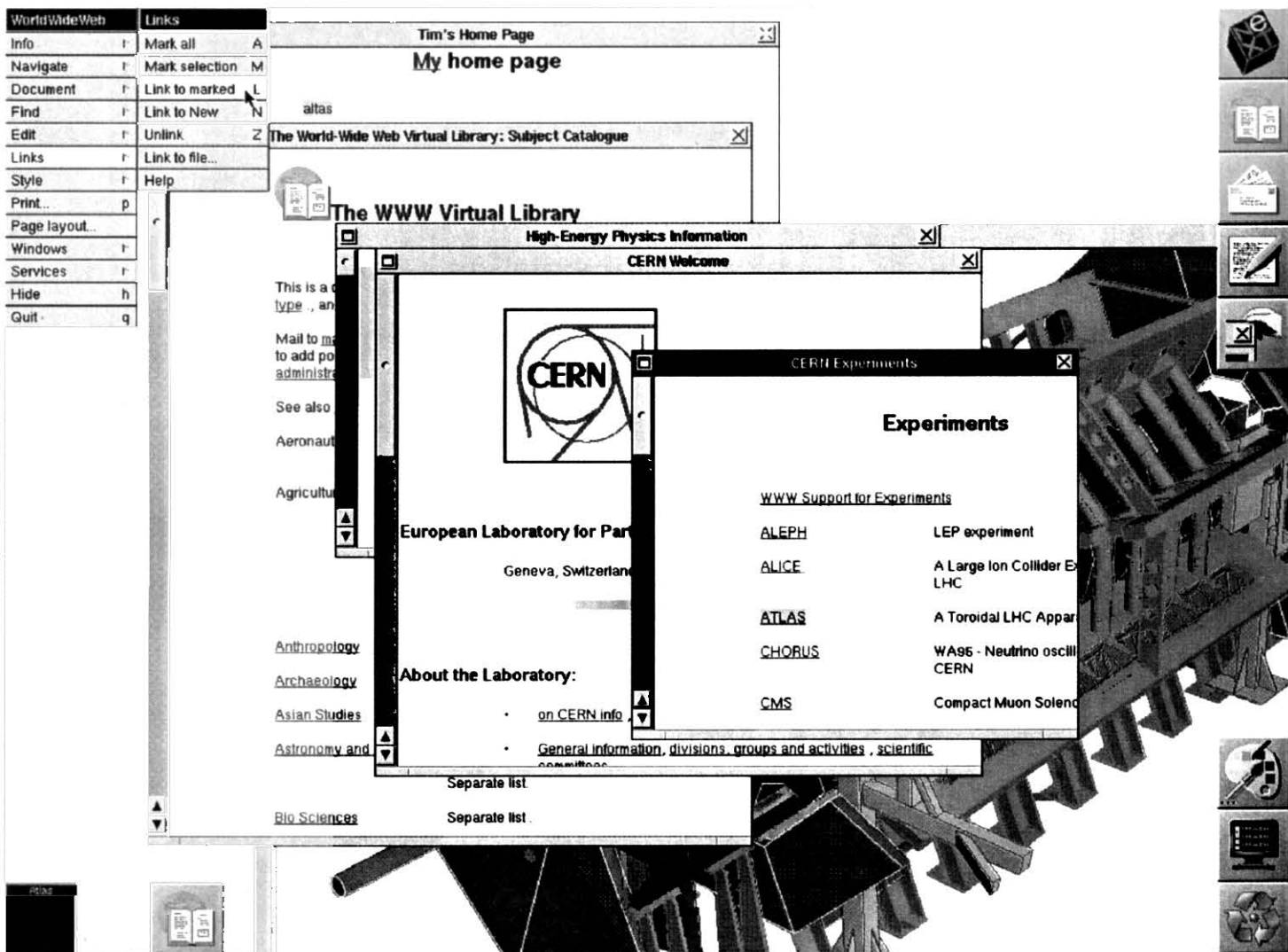
The HTTP request from the client starts with an operation code (known as the method, in conformance with object-oriented terminology) and the URI of the object. The “GET” method used by all browsers is defined to be idempotent in that it should preserve the state of the Web (apart from billing for the information transfer, and statistics). A “PUT” method is defined for front-end update, and a “POST” method for the attachment of a new document to the Web, or submission of a filled-in form or other object to some processor. Use of PUT and POST is currently limited, partly due to scarcity of hypertext editors. The extension to other methods is a subject of study.

When objects are transferred over the network, information about them (“metainformation”) is transferred in HTTP headers. The set of headers is an extension of the Multipurpose Internet Mail Extensions (MIME) set. This design decision was taken to open the door to integration of hypermedia mail, news, and information access. Unlike in email, transfer in binary, and transfer in nonstandard but mutually agreed document formats is possible. This allows, for example, servers to indicate links from, and titles of, documents (such as bit-map images) whose data format does not otherwise include such information.

The convention that unrecognized HTTP headers and parameters are ignored has made it easy to try new ideas on working production servers. This has allowed the protocol definition to evolve in a controlled way by the incorporation of tested ideas.

Hypertext Markup Language (HTML)

Despite the ability of HTTP to negotiate formats, W3 needed a common basic language of interchange for hypertext. HTML is that language, and much of the fabric of the Web is



to be sufficiently simple so as to be easily produced by both people and programs, but also to adhere to the SGML standard in that a valid HTML document, if attached to SGML declarations including the HTML "DTD," may be parsed by an SGML parser. HTML is a markup language that does not have to be used with HTTP. It can be used in hypertext email (it is proposed as a format for MIME), news, and anywhere basic hypertext is needed. It includes simple structure elements, such as several levels of headings, bulleted lists, menus and compact lists, all of which are useful when presenting choices, and in on-line documents.

Under development is a much enriched version of HTML known as HTML+. This includes features for more sophisticated on-line documentation, form templates for the entry of data by users, tables and mathemati-

Figure 1. Using the World-Wide Web. Shown here is the authors' prototype World-Wide Web application for NextStep machines. The application initially displays the user's "home" page (top) of personal notes and links (top). Clicking on underlined text takes the reader to new documents. In this case, the user visited the Virtual Library, and, in the high energy physics department, found a link to CERN. Linked to CERN was the "Atlas" collaboration's web including an engineering drawing (background). To save having to follow the same path again, the link menu (shown) allows a new link to be made, for example from text typed into the home page, directly to the Atlas information.

ers support a subset of the HTML+ features in addition to the core HTML set.

HTML is defined to be a language of communication, which actually flows over the network. There is no requirement that files are stored in HTML. Servers may store files in other formats, or in variations on HTML that include extra information of local interest only, and then generate HTML on the fly with each

W3 and Other Systems

Two other systems, WAIS (from Thinking Machines Corporation and now WAIS, Inc.) and Gopher (from the University of Minnesota), share W3's client-server architecture and a certain amount of its functionality. Table 1 indicates some of the differences.

The WAIS protocol is influenced largely by the z39.50 protocol designed for networking library cata-

Table 1. A comparison of three popular network information projects.

Registered server figures taken April 27, 1993 and April 15, 1994. WAIS: from Thinking Machines Corporation directory, number of distinct hosts. Gopher: from "All the Gophers in the world" register at the University of Minnesota. W3: from Geographical registry at CERN. In all cases many more servers exist which are not directly registered, so these are a very rough guide with no indication of quantity or quality of information at each host.

	WAIS	Gopher	World-Wide Web
Original target application	Text-based information retrieval	Campus-wide information (CWIS)	Collaborative work
Typical objects			
Text	YES	YES	YES
Menus, Graphics	NO	YES	YES
Hypertext	NO	NO	YES
Search functions			
Text search	YES	YES	YES
Relevance feedback	YES	NO	NO
Reference to other servers	NO	YES	YES
Registered servers			
April 1993	113	455	62
April 1994	137	1410	829

and retrieval following a search. Indexes to be searched are found by searching in a master index. This two-stage search has been demonstrated to be sufficiently powerful to cover the current world of WAIS data. There are no navigational tools to allow the reader to be shown the available resources, however, or guided through the data: the reader is "parachuted in" to a hopefully relevant spot in the information world, but left without context.

Gopher provides a free text search mechanism, but principally uses menus. A menu is a list of titles, from which the user may pick one. While gopher space is in fact a web containing many loops, the menu system gives the user the impression of a tree. The Veronica server provides a master index for gopher space.

The W3 data model is similar to the gopher model, except that menus are generalized to hypertext documents. In both cases, simple file servers generate the menus or hypertext directly from the file structure of a server. The W3 hypertext model gives the program more power to communicate the options available to the reader, as it can include headings and various forms of list structure, for

All three systems allow for the provision of graphics, sound and video, although because the WAIS system only has access by text search, text has to be associated with graphics files to allow them to be found.

W3 clients provide access to servers of all types, as a single simple interface to the whole Web is considered very important. Unknown to the user, several protocols are in use behind the scenes. A common code library "libwww" put into the public domain by CERN has promoted this uniformity. Whereas one would not wish to see greater proliferation of protocols, the existence of more than one protocol probably allows for the most rapid progress during this phase in the development of the field. It also allows a certain limited confidence that, if an architecture can encompass older systems and allow transition to current systems, it will, by induction, be able to provide a transition to newer and better ideas as they are invented.

Recent W3 Developments

This article, like others in this issue, was derived from material written in April 1993 for the INET'93 conference. Growth of the Web since that

tion has been completely rewritten. There are now 829 (May: 1,248) rather than 62 registered HTTP servers, and many more client programs available as then.

The initial prototype W3 client was a "wysiwyg" hypertext browser/editor using NeXTStep. We developed a line mode browser, and were encouraging the developments of a good browser for X workstations. One year ago, NCSA's Mosaic W3 browser was in wide use on X workstations. Its easy installation and use was a major reason for the spread of the Web. Today there are many browsers available for workstations, Macintosh and IBM/PC compatible machines, and for users with character-based terminals. Of the latter category, "Lynx" from the University of Kansas provides full-screen access to the Web for users with character terminals or emulators running on personal computers. Since new software is appearing frequently, readers are advised to check the lists on the Web for those most suited to their needs.

The availability of browsers and the availability of quality information have provoked each other. One available indicator of growth has been Merit Inc.'s count of the traffic of var-



Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.