# EXHIBIT 3

SERIAL NUMBER	7085	NAN	3018	<b>39</b>	PATENT NUMBER	3.			
SERVAL NUMBER 08/727,085	PUNG DÁTE 10/08/96	CLASS	-	SVBCLAS	101	GR	OUP ART UNIT	EXAMINER	
GILBERT BORMA	W Bloom	****/c.>	7 1	. 50	0 T SUO	TNOGO	**************************************	HONG.	<del></del>
ARUL SEBASTIA	IN, STAMFO	ORD, CT;	ANUP I	MATHUF	, SUNN	IYVALE	. CA; VIN	AY WADHWA,	
GURAGON THO	4-cv-703	66-JAC	-RSW	טיייין	ocum	ent 54	4-5 Fil	ed 09/27/2	005 Pa
**CONTINUING	DATES***	******	· ******	***	•			EXHI	BITTE
VERIFIED		:- j			•		•	30	
NONE-SA	+	<i>*</i>		•				V- 2. 05	
*		/		•		•			the desired and
***FOREIGN/PC	T APPLZCA	rions***	****	***		. ••	•		
VERIFIED							÷	•	
NOVZ-SS	#	: -						<u>.</u>	
		11/2			:-				
FOREIGN FILI	NG LICENS	E GRANTEI	04/1	0/97	•	****	SMALL EN	TITY ****	
J.									
Foreign priority claimed		AS	STATE OR	SHEETS	TOTAL	INDEP.	FILING FEE RECEIVED	ATTORNEY'S DOCKET NO.	·
35 USC 119 conditions met Verilled and Adknowledged		FILED	STATE OR COUNTRY MI	SHEETS DRWGS.	TOTAL CLAMS 26	INDEP. CLAIMS		DOCKETNO.	2
Verilled and Acknowledged CHARLES C C	ARY INI GOODR	FILED	MI	DRWGS.	CLAIMS	CLAIMS	RECEIVED	DOCKETNO.	•
35 USC 118 conditions met   Verified and Acknowledged   CHARLES C C	ARY INI GOODR LE ROAD	ICH & RO	MI	DRWGS.	CLAIMS	CLAIMS	RECEIVED	DOCKETNO.	
SS USC 116 conditions met Verilled and Acknowledged CHARLES C C WILSON SONS CONSTRUCTION OF PALO ALTO CO	ARY INI GOODR LE ROAD	ICH & RO	MI SATI	DRWGS.	CLAMS 26	GLAIMS 6	\$636.0	DOCKETNO. 0 19041.701	<u> </u>
Verified and Admonfedged CHARLES C C WILSON SONS 650 PAGE MI	ARY INI GOODR LE ROAD A 94304-1	ICH & RO	MI SATI	DRWGS.	CLAMS 26	6 FOR A	\$636.0	DOCKETNO. 19041.701  DATA FROM	Ā
Verified and Acknowledged CHARLES C C WILSON SONS 650 PAGE MII PALO ALTO C	ARY INI GOODR LE ROAD A 94304-1	ICH & RO	MI SATI	DRWGS.	CLAMS 26	6 FOR A	\$636.0	DOCKETNO. 0 19041.701	A 6124
SS USC 118 conditions materially in the state of the stat	ARY INI GOODR LE ROAD A 94304-1 ARCH TOCK	ICH & RO	MI SATI	DRWGS.	CLAMS 26	6 FOR A	\$636.0	DOCKETNO. 19041.701  DATA FROM	A 1228
SS USC 119 conditions materially stated and Acknowledged CHARLES C C WILSON SONS SO PAGE MIL PALO ALTO CO TIME TO THE SET OF APPLICATION SONS PALO ALTO CO TIME TO THE SET OF APPLICATION SONS PARTS OF APPLICATION SONS PARTS OF APPLICATION SONS FILED SEPARATELY	ARY INI GOODR LE ROAD A 94304-1  ARCH TOCK JG LOCATT	ICH & RO	MI SATI	DRWGS.	CLAMS 26	6 FOR A	RECEIVED \$636.0  RETRIEVING TOF COMMUPAT	DATA FROM  a TIM—PTO-456L (Rem  Columns Examiner. /	A 4124
SS USC 118 conditions materially stated and Acknowledged CHARLES C C. S. WILSON SONS SECTION SONS SECTION SONS FOR PAGE MILES PALO ALTO CO. TINTERNET SECTION SONS WEST WAS ALTO CO. TINTERNET SECTION SO	ARY INI GOODR LE ROAD A 94304-1  ARCH TOCK JG LOCATT	ICH & RO	MI SATI	DRWGS.	CLAMS 26	FOR A	RECEIVED \$636.0  RETRIEVING TOF COMMUPAT	DATA FROM  a TIM-PTO-456L (Page	A 4124
SS USC 118 conditions materially stated and Acknowledged CHARLES C C. S. WILSON SONS SONS PAGE MII PALO ALTO C. S. T.	ARY INI GOODR LE ROAD A 94304-1  ARSH TOOL JG LOCATT	ICH & RO	MI SATI	DRWGS.	CLAMS 26	FOR A	#636.0  #636.0  PETRIEVING TOF COMMIPAT	DATA FROM  a TM—PTO-456L (Recommendations Examiner.)	A .12-1
SSUSC 118 conditions materially in the conditions will be conditioned by the conditions of the conditi	ARY INI GOODR LE ROAD A 94304-1 ARCH TOCK JG LOCATI  ANCE MAILED 1998	ICH & RO	MI SATI	DRWGS.	CLAMS 26	CLAIMS 6	RECEIVED \$636.0  RECEIVED \$636.0  RECEIVED  ADDRES CLAIMS CLAIMS CLAIMS	DOCKET NO. 19041.701  DATA FROM  & TM-PTO-438L (Rev  Hons Examiner.  ALLOWED  Print Claim/  WING	A 4124
SSUSC 118 conditions materially in the conditions with the conditions and acknowledged the conditions of the conditions	ARY INI GOODR LE ROAD A 94304-1 ARCH TOCK JG LOCATI ATTON Y VANCE MAILED	ICH & ROUDS OF THE Assistant Ex	MI SATI  D AND FIERS  TERRES	APP	CLAMS 26	CLAIMS 6	#636.0  #636.0  #636.0  #636.0	DOCKET NO. 19041.701  DATA FROM  & TM-PTO-438L (Rev  Hons Examiner.  ALLOWED  Print Claim/  WING	A 4124
SSUSC 118 conditions materially in the conditions will be conditioned by the conditions of the conditi	ARY INI GOODR LE ROAD A 94304-1 ARCH TOCK JG LOCATI  ANCE MAILED 1998	ICH & ROUDS OF THE Assistant Ex	MI SATI D AND FIERS	APP	CLANAS 26 ARATUS	CLAIMS 6	RECEIVED \$636.0  RECEIVED \$636.0  RECEIVED ADDRES CLAIMS C	DOCKET NO. 19041.701  DATA FROM  & TM-PTO-458L (Rev  Hons Examiner.  ALLOWED  Print Claim/  WING:  Qrivg. Print Fig. J	A 412-4
SSUSC 118 conditions materially in the conditions with the conditions of the conditi	ARY INI GOODR LE ROAD A 94304-1 ARCH TOCK JG LOCATI  ATTON Y VANCE MAILED  1998	ICH & ROS	MI SATI  D AND FIERS  TERRES	APP)  APP)  Printing	CLANS 26  ALATUS	CLAIMS 6	PECENED \$636.0	DOCKET NO. 19041.701  DATA FROM  & TM-PTO-458L (Rev  Hons Examiner.  ALLOWED  Print Claim/  WING:  Qrivg. Print Fig. J	A 4124
SSUECTIBE conditions materially and Acknowledged  CHARLES C C WILSON SONS  SOO PAGE MIT PALO ALTO CO  INTERNET SE WATERNET SE	ARY INI GOODR LE ROAD A 94304-1 ARCH TOGL JG LOCATI  AND LOCATI  A	ICH & ROS	MI SATI  D AND FIELS  ATENIEAS	APP)  APP)  Prior Por IS	THE SUE	U.S. DEP	Applica CLAIMS I Claims DRA els/Drwg. Figs.	DOCKET NO.  18041.701  DATA FROM  ATM—PTO-436L (Recommendations Examiner:  ALLOWED  Print Claim/  WING*  Orwig. Print Fig. 7  5 C.	
SSUSC 118 conditions materially in the second part of the second part	ARY INI GOODR LE ROAD A 94304-1 ARCH TOGL JG LOCATI  AND LOCATI  A	ICH & ROS	D AND FIELS  TEPAREL  The information the United	Principal Princi	THE THE STATE OF T	CLAMS 6  FOR A U.S. DEP  Tota  Sha iner NU y be restrict Sections 1:	Applica CLAIMS I Claims DRA els/Drwg. Figs.	DOCKET NO.  13041.701  DATA FROM  TIM-PTO-436L (Residents Examiner.  ALLOWED  Print Claim/  WING  Oring. Print Fig. 1	

		APRINOVED FOR LICENSE
	PATENT APPLICATION	INITIALS
	08727085	
Date (**	CONTENTS	Date Received
Counted	CONTENTS	or Minied
	in the second se	RECEIVED
	Il I	MR 24 1997
	1. Applicationpapers.	
<u> </u>	2 stupe for & view	GRUUP/23/00
Case	2:04-cz-70366-dAC-RSW Docume	ent <b>54,5.5</b> 7 Filed 09/27/20
2-24-98	* Ras(Birley)	3.3.98828
a-a+-75	( C) ( CO. )	<u> </u>
	Est of Time I many	JUN 0 8 1998
<u> </u>	5. Amost A w/ Appendices A-D	6-8-98
8-27-98=	T. Notice of Allowance	AUG 3 1 1998
12/1/98	8. Drawings ( V shife) set (	1917/13
	DOOR OF GARAGINE STAR	
	••	
1	<u>o</u> .	
	1.	·
	2	•
	3	_
	4	
1	<b>5.</b>	
	6	
	7	<i>M</i> <sup>2</sup>
	B	
		The second secon
	3	
	20	
<b></b> :	21.	
1704	2.	
	•	
<b>33</b> 5	3.	
<u> </u>	24	
	25	
	26	
	27	
	28.	<del></del>
<u> </u>	29.	<u>:</u>
1	30.	
	31.	
	31.	
	32	
		a <u>. (≛**) - (-\$**</u>

Rage 3 of

## United States Patent [19]

Borman et al.

[11] Patent Number:

5,890,172

[45] Date of Patent:

· Mar. 30, 1999

[54] METHOD AND APPARATUS FOR RETRIEVING DATA FROM A NETWORK USING LOCATION IDENTIFIERS

Cip Shreston: Giffer Bornes Shoondate Mile SW Mich; Bajas Bhatagar; Aruj Sebastian, both of Stamford, Cood.; Assap Mather, Sannyvale, Calif.;

Assp Mathur, Sumyvale, Calif.; Vinny Wadhwa, New Delhi, India; Mulesh Kumar, Pasadena, Calif.; C. Vinny Kumar Singh, Geragon, India

[73] Assignoe: Tenretni Dynamics, Inc., Southfield, Mich.

[21] Appl. No.: 727,085

[22] Filed: Oct. 8, 1996

[51] Int. CL\* \_\_\_\_\_\_\_ G06F 15/16 [52] U.S. Cl. \_\_\_\_\_\_ 767/501; 707/513; 345/339

58] Field of Search 707/501, 513, 707/514, 10, 102, 104; 345/329, 333, 335, 338, 339, 340, 346

[56] References Cited

**PUBLICATIONS** 

CNN Interactive, "http://cnn.com/index", Feb. 20, 1998, screen printed pp. 1-7.

screen primost pp. 1-1.

AltaVism, "http://www.altavism.com", Feb. 20, 1998, screen printosts pp. 1-2.

Primary Examiner—Stephes S. Hong
Attorney, Agent, or Firm—Wilson Sonsini Goodrich &
Rosati; Paul Davin; Charles C. Cary

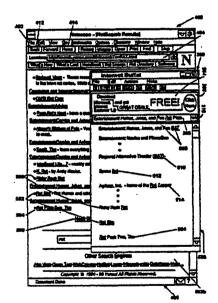
Document 54-5

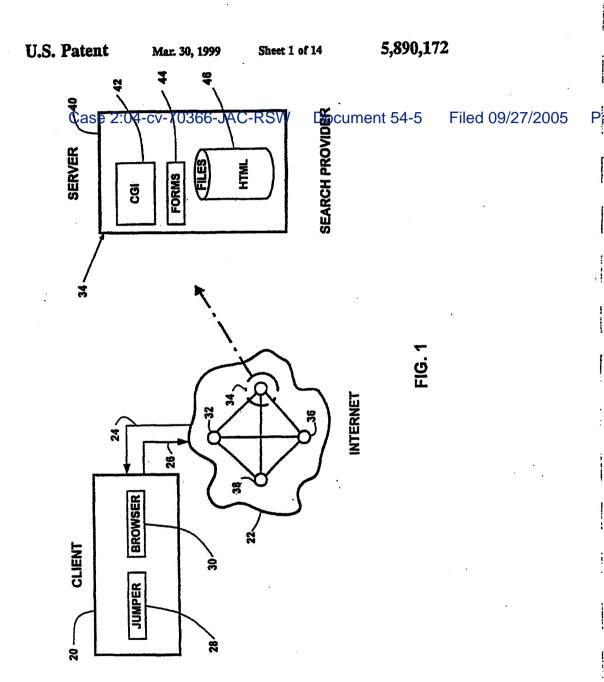
Filed 09/27/2005

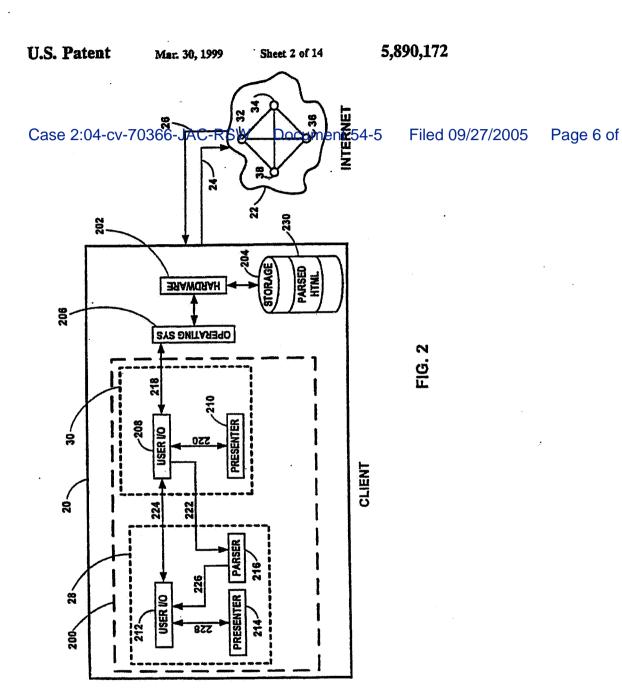
Page 4 of

mputer implemented method and system for retrieving information through a browser connected to a network. A first file of information is received which includes a first mark-up language to identify contents of the information. which contents include site identifiers. The site identifiers corresponding for example to file locations on the Internet. The first file is displayed is a browser window. Responsive to receiving the first file of information by the browser, the first file of information is persod by a jumper to generate a list of site identifiers. This list of site identifiers is then stored by the jumper and displayed in a jumper window. Responsive to an activation by the user, a computer is directed to perform the following steps. The jumper determines which of the stored site identifiers is carmotly selected and sutometically selects the next. Next the jumper directs the between to access the file at the site corresponding to automatically selected sits identifier. Finally, the browser is directed to display the file the browser has retrieved in the browser window.

18 Claims, 14 Drawing Sheets





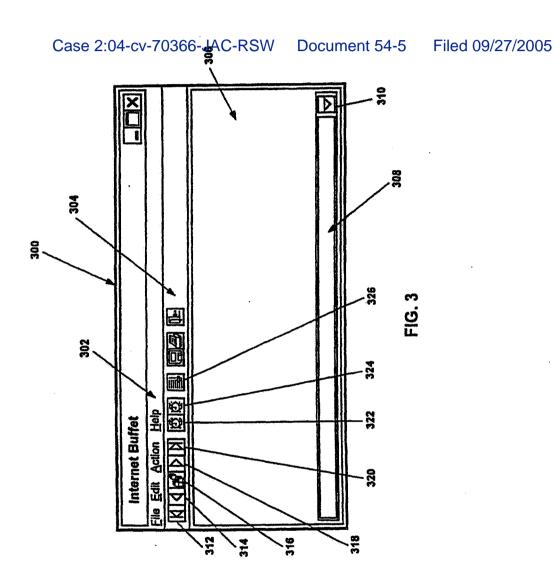


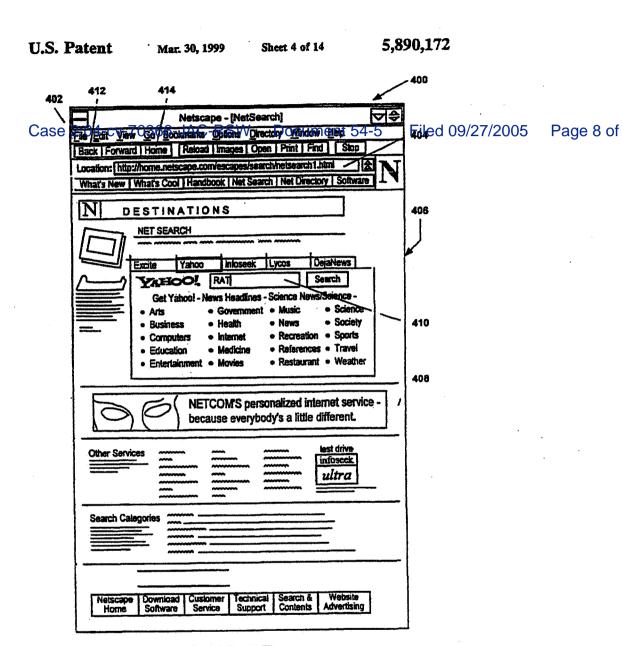
U.S. Patent

Mar. 30, 1999

Sheet 3 of 14

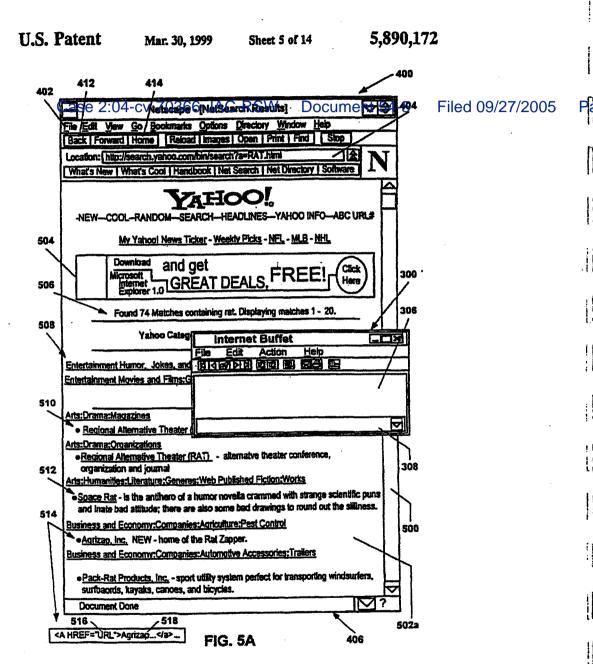
5,890,172



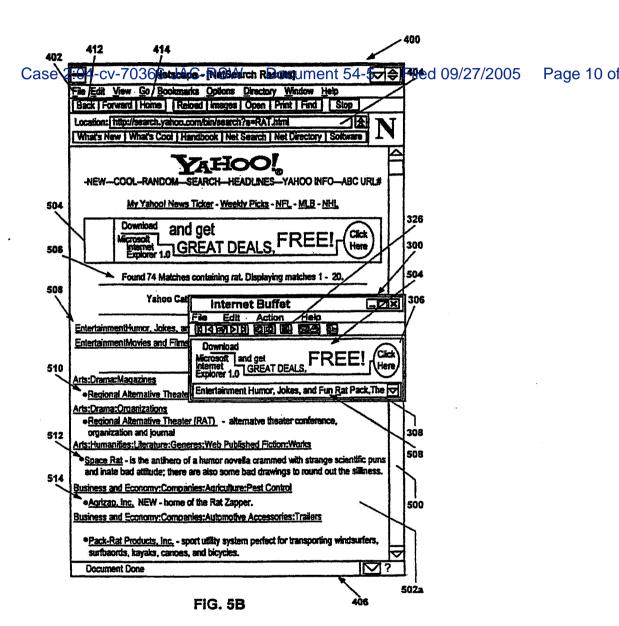


(PRIOR ART) FIG. 4

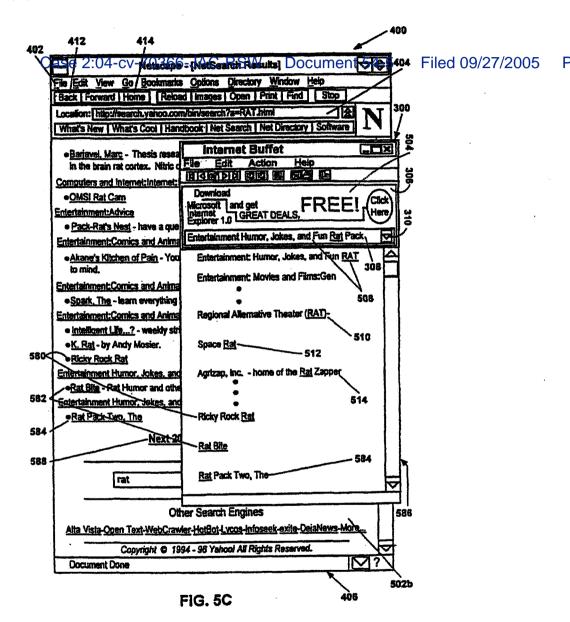
ء ند



5,890,172



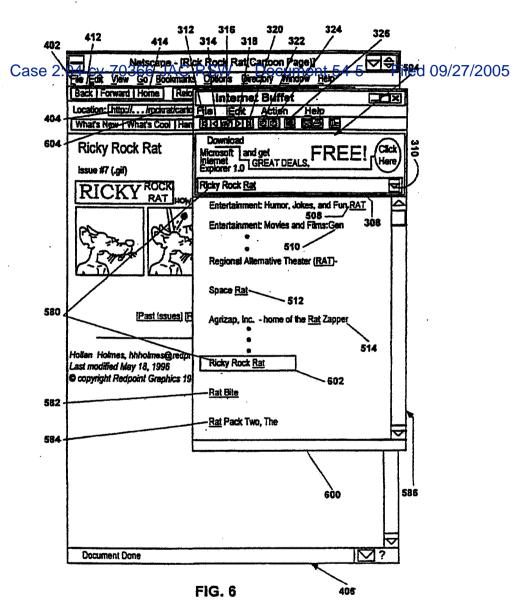
5,890,172

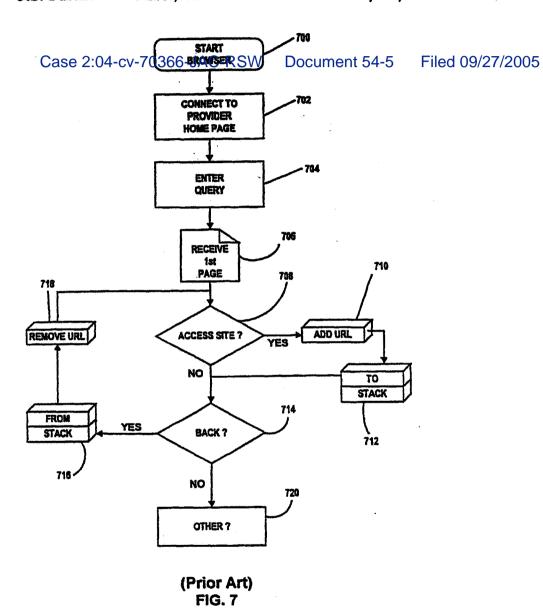


Sheet 8 of 14

5,890,172

Page 12 of





i de di

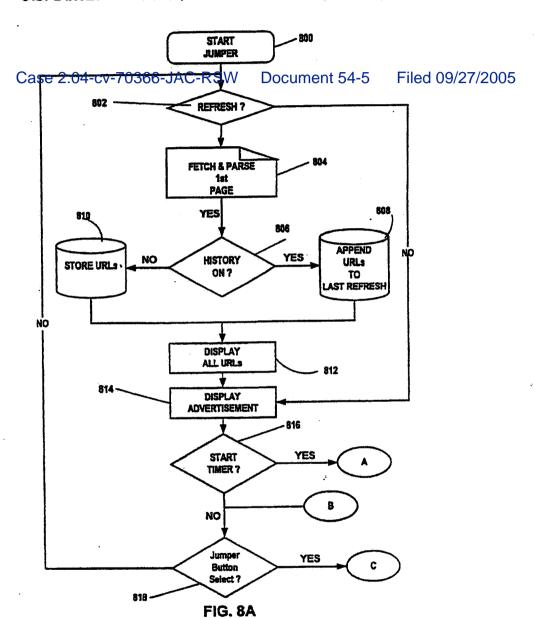
U.S. Patent

Mar. 30, 1999

Sheet 10 of 14

5,890,172

Page 14 of



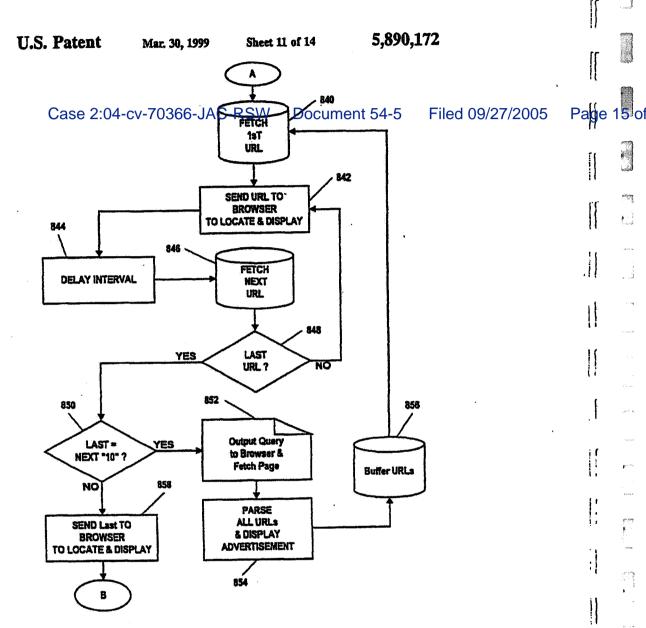
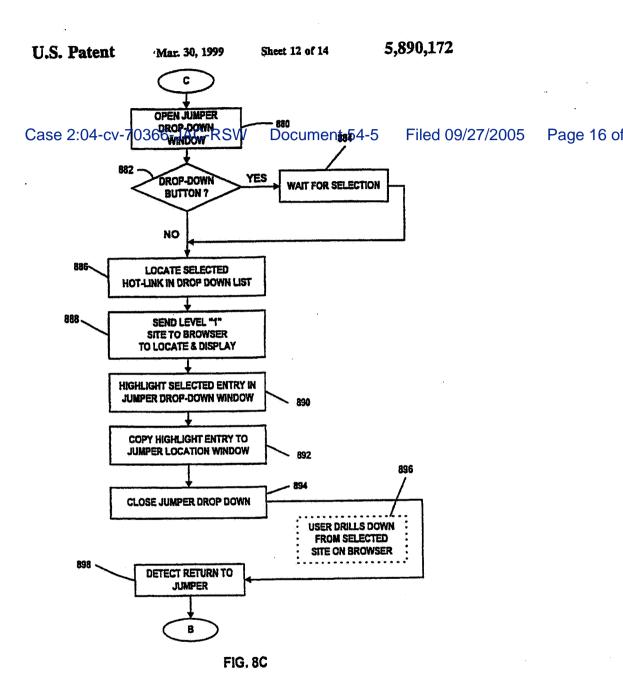
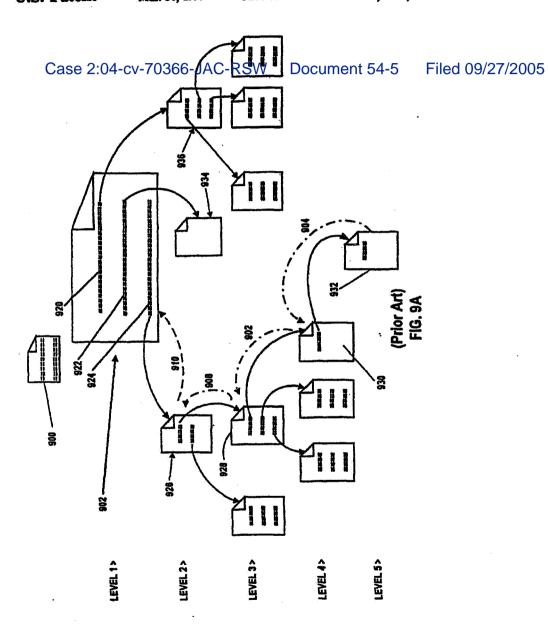
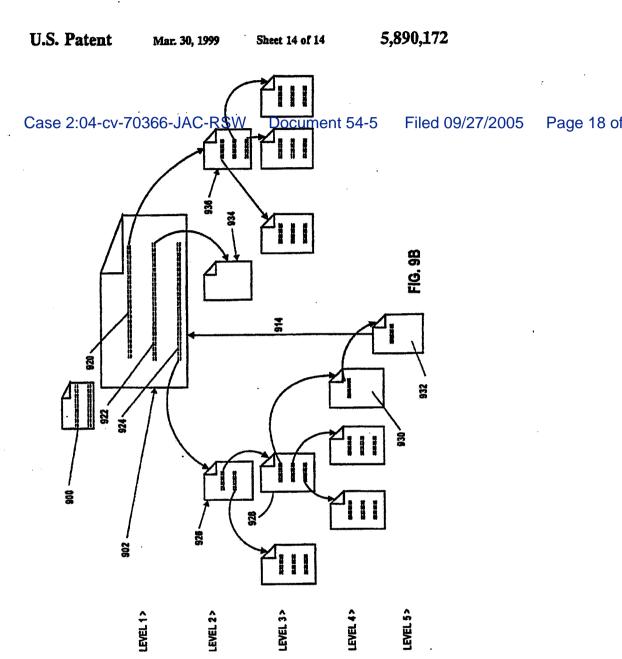


FIG. 8B







#### METHOD AND APPARATUS FOR RETRIEVING DATA FROM A NETWORK USING LOCATION IDENTIFIERS

at invention relates to information retrieval. More specifically, the invention relates to tools for traversing hypertext data.

#### BACKGROUND INFORMATION

The development of computerized information reso The development of computerment information resources such as the Internet and various on-hise services, such as Computeree's, America On-Line<sup>110</sup>, Prodigy<sup>110</sup> and other services has lead to a proliferation of electrosically-svailable information. This electrosic information increasingly displacing more conventional means of information transmission, such as newspapers, magazines and

The largest information resource in existence today is the laternet. The internet is a group of client and server com-puters limited one to another and each having a unique Internet. The internet is a group of client and server computers linited one to another and each having a unique
identifier, DNS (distributed actwork server), assigned by the
Internet authority in Cambridge, Mans. and Geneva, Switzerland. In order the information to be found on the Internet
every file is given a specific address by which it may be
located. To access the internet a user employs what is called
a browner. Curroutly the most popular browner is Neucape
Navigator the browner developed by Neusape Communications Corporation of Moustain View, Call. A wide array of
browners is available for just about every platform. The
browner's job is twofold. First, given a pointer to a piece of
information on the set it has to be able to access the
information or operate in some way based on the contents of
that pointer, Second, if the document/file (bertinafter file) is
encoded the browner has to translate that to a suitable format
for display to the user. The display may include multimedia for display to the user. The display may include multimedia effects, e.g. sound and animation.

The most popular encoding of Internet files commun cated between client and server is the HTML (hypertext markey) language). The WWW (World Wide Web) or simply the "Web" includes all the servers adhering to this Each page loaded from the internet is a single file encoded in HTML. HTML describes the structure of a file. The structure of the file includes title, paragraphs, images and any pointers to other files.

The pointers to a specific site is called an URL (Uniform Resource Locator). The URL provides a universal, consistent method for finding and accessing information for a Web however. The URL comprises a file type, a server LD. (DNS), one or more directories and subdirectories, and a file (DNS), one of most circumes and subsections, and a me name. URLs are also used as part of a hypertax link within a file to another file. These URLs then provide the browser with a way to navigate the Web. URLs constain information about a file: including file type (FIP, Gopbe, HTTF), the interset server on which the file is located (WWW.NCSA.UIUC.EDU, or FDP.AFPLE.COM, or Net Com 16.Net.Com, and so on), the directory of the file, and the file nem

In order to speed the process of finding relevant informa-tion on the internet several servers on the internet provide an tion on the internet several servers on the internet provide at index to the internet and a search origins. These information indexers such as Yakoo <sup>31</sup>, Excise <sup>34</sup>, Lycos <sup>34</sup>, Informi <sup>34</sup>, at and Alta Vista <sup>34</sup> perform two valuable functions. First, using their own Internet links, they continually search the

Interpet and index all files on the laternet into subjecategories and store this index on their own database. The information indexers also allow a client to connect to their USING LOCATION IDENTIFIERS

BACKGROUND OF THE INVENTION

CASELFOR THE INVENTION

STREET OF THE I

a client in should be noted that the information that is retrieved from Yahoo to contains only a general topic identifier and the file location on the location for that specific

It would indeed be a combergone process for the client seeking specific information if the indexes that were retrieved from Yahoo <sup>30</sup> only told the searcher where to look. If this were the case, the considerable task remaining to the eld be to manually enter the network addres of each file and the go through the process of retrieving that file. To overcome this problem, the search result retrieved by Yahoo<sup>ra</sup> is escoded in HTML as a bot-link which makes every "footnoss" an active rather than a passive reference. These hot-lights appear to the user in the browser window as bold face tent which is easily distinguished from the other sext based information is the file. A hot-link comprises a text description and a corresponding URL. When the user selects a bot-link the browser detects that selection and outputs the a bot-link the browse detects that selection and outputs the URL on the laternet to retrieve the file corresponding to that URL and display it to the user. Therefore, by merely selecting with a mouse a specific flootnote in a file encoded in a markey language, a client is immediately given access to the remote web server that contains the specific file referred to in the footsots.

With a markup language such as HTML (hypertext write a marking language such as rilivil. (hypericit marking language) both anatisms and profusionals become authors and the footnotes on the printed page become the hypericit of the electronic page. What was a passive reference new becomes as accessible link to a related file. A marking language describes the structure of a file including backings, persympts, images and what are called not-links.

A hos-link displays at the user level as tent or graphic and is processed for communication purposes as an URL. It. is these hos-links which provide the interactive footnotes eribed above.

Even with the indexing provided by Yahoo Te, Lycos Te.
Excite Te, Inktomi Te. Alta Vista Te. etc., the process of finding the exact topic is still extremely time consuming and can involve visiting literally bandreds of Web Sites.

Typically, a user will retrieve a file from an information Typically, a mer will retrieve a net that an instantation indexer and will not only look at the files retrieved by selecting the indexer's hot-links, but will also select other hot-links in the retrieved documents. This process of starting a search that begins with an initial hot-link and following a search trail that leads to successive files each increasingly displaced from the starting point is known as a drill-do The problem with current browsers is that when a user has the problem with current provinces at teat when a user that drilled-down through many levels of sites, the cally way to return to the original HTML file is to thit the browser's beek key which moves the user up one level at a time through the original search tree back to level "1." Only then can the user ess other hot-links retrieved in the original search

What is needed is a more efficient way to conduct a

#### SUMMARY

A first object of the invention is to provide operational controls for simplified Internet navigation from various sites and back again.

**(...** 

t

A second object of the invention is to save users time and effort in finding information on the lotemet.

A third object of the invention is to enable the publication

A third object of the invention is to enable the publication of a new format for on-line magazines, called Notatines, which allow publishest to leaverage the newigational controls a provided in the invention product to foster a new paradigm (or the white industry, style infinitely style industry). A comparer implemented method and system for retrieving information from a network. In a first embodiment a first

information from a network. In a first embodiment a first get mark-up language to identify contents of the information is received which may include a first mark-up language to identify contents of the information, which contents include site identifiers. The site identifiers corresponding for example to file locations on the Internet. The first file is displayed in a browner window. Responsive to receiving the first file of information by the however, the 15 first file of information by the however, the 25 first file of information by the pumper to generate a list of site identifiers. This list of site identifiers is then stored by the jumper and displayed in a jumper window. Responsive to an extivation by the user, a computer is directed to determine which of the stored site identifiers is currently as selected and automatically selects an other. The other includes the first, the prior, the next, or the last on the list.

In a second preferred embodiment a first file of information in a hypertaxt markup language is received and displayed in the browser window. The first file of information also contains sits identifiers and other information. The first file is displayed in the browser and is passed, and the site identifiers from that file is ear stored by the jumper in a list. The stored list of sits identifiers is displayed in the jumper window. Responsive to a selection by a user of automatic amode, a computer is discreted to perform the following stops. The jumper selects a first sits identifier from the stored list of sits identifiers. The jumper directs the browser to access the file at the sits corresponding to the untomatically selected sits identifier. The browser, is directed by the jumper, to display the file in the browser, window. Thus the jumper inditiates a delay of a predetermined interval. At the end of the interval the jumper selects the next site identifier from the stored list of site identifiers. The jumper directs the browser to access the file at the site corresponding to the automatically selected next site identifier. The browser, is then directed by the jumper, to display that file in the browser window. The jumper, to display that file in the browser window. The jumper, to display that file in the browser window. The jumper continues to stpeat this animation loop from one site identifier in the stored list of site of identifiers has been sent to the browser and displayed, or until the steer terminates the process. In this manner, an animated tour of all files having site identifiers parsed from the flext file is displayed in the browser window under the

direction of the jumper.

In a third embodiment, a first file of information is a hypertext markup language is received and displayed in a browser window. The first file of information also contains a site identifier and other information. The first file of information is displayed by the browser in the browser window. Responsive to receiving the first file of information, the jumper perses the first file and extracts and stores a list comprised of first file site identifiers. The stored list of site identifiers is then displayed in the jumper window. The user is allowed to select from the browser window a specific site identifier, in mown as a search level "1" site identifier, and responsive thereto to cause the browser to access and display a second file retrieved from the site corresponding to the select few 1" site identifier. The user is again allowed to select from the browser window a specific site identifier and transponder to cause the browser window a specific site identifier and transponder the cause the browser to access the browser to access the browser to the property of the select file and transponder the property causes the browser to access the second to access t

: .

: د د access and display a third file retrieved from the site corresponding to the selected site identifier. This process can be repeated in the browser, until the user has completed their search, or dell-down.

Responsive to the completion of the drill-down in the browner, as indicated to the jumper by receipt of a single jump life@04 [170-016] take the gamputer is dicated to 9/27/2005 perform the following steps. The jumper determines which of the stored level "1" site identifiers is currently selected and automatically selects the single jump level "1" site identifier. Next the jumper directs the browser to access the file at the site corresponding to the single jump selected level "1" site identifier. Finally, the browser is directed to display the file the browser has retrieved in the browser window. This allows the ware to retern to the files pointed to by the original level "1" site identifiers without having to traverse acriel order the site identifiers selected in the drill-down.

Page 20 of

#### BRIEF DESCRIPTIONS OF DRAWINGS

The present invention is illustrated by way of example and not limitation in the figures in which like references indicate libs elements and in which:

FIG. 1 shows a client server system including embodiments of the jumper invention.

FIG. 2. shows a more detailed view of the client server system illustrated in FIG. 1.

FIG. 3. shows a preferred embodiment of the screen interface and tool bar of the jumper.

FIG. 4 shows a prior art browser user interface, and the query form of an information index provider.

FIGS. SA-C shows a the initial stages of a search session utilizing a prior art browser and the jumper.

FIG. 6, shows a later stage in the search session of FIGS. SA-C.

FIG. 7, shows the steps in the search process in prior art browsers.

FIGS. 8A-C shows the processes of a search sension on the jumper.

FIGS. 9A-B show the search traversal patterns of a prior art browser and the jumper.

#### DETAILED DESCRIPTION

One embodiment of the invention is a method and apparatus for retrieving information through a browser connected to a setwork, and providing the information to a user. Although the following will be described with reference to certain particular embodiments, including data structures, flow of steps, hardwars configurations, menu configurations, etc. . . . implementation of the invention can be practiced without these specific details.

One embodiment of the invention allows a search to be conducted in non-linear order. This may be done in either single jump or automatic jump mode utilizing a jumper. A search tree does not therefore need to be traversed in revenuserial order after a drill-down as is required utilizing prior art browsees. The invention allows a searcher to jump across multiple levels at a time.

Search Levels:

a second file retrieved frost the site corresponding to the selected level "I" site identifier. This second file includes a bypertext marking language and site identifiers. The user is at bypertext marking language and site identifiers. The user is at large in allowed to select from the browser window a specific site identifier and responsive thereto causes the browser to

as linked to four levels of files. Selecting hot-link 924 causes the browser to access file 926, which is a level 2 file. g the first bot-link on file 926 causes the browser to rescuent use the noveless on the 1.00 cases the browner to access the 528, which is a level 3 file. Selecting the first hot-link on file 528 causes the browner to access the 528, which is a level 5 file. Fightly, selecting the first hot-link on file 530 causes the browner to access the 532, which is a level 5 file. Selecting hot-link 522 causes the browner to access file 934, which is a level 2 file. Selecting bot-link 920

causes the browser to account file 356, which is a level 2 file.

The meversal steps from host-link 924 through 926, 926, 936 and 932 constitute a drill-down from level 1 to level 5. Prior act between require that the user, in order to return to the original file 902, must trevene the search tree in revene the original file 902, must reverse the search tree in reverse social order. As shown serial return 904 traverses from 932 15 to 932, serial return 906 traverses from 938 to 928, serial return 906 traverses from 928 to 926, and finally serial setten 918 traverses from 928 to 926, the initial starting point. This requires the seedless steps of visiting each of the branches traversed in the drill-down. This process is unaccessarily to time communing when compared with the expediilities of the

current jumper invention.
FIG. 9B shows the same tree structures as does FIG. 9A. FIG. 3B shows the same tree structures as coss Fig. 37.
FIG. 3B shows the use of one embodiment of the invention to provens many levels at a time in either single jump mode when the searcher has reached level 5 file 332. If the user wants to return to level 1 they do not need to do so in revenue serial order, as is required by prior art browsers. Instead, th order, as is required by prior art between finstead, the user directs the jumper to access the sent level 1 hot-link. In response, the jumper centers the user in a slogic jump 914 from level 5 file 932 to the level 2 files pointed to by the hot-links in the level 1 file 902. Much time is saved and the user can process to access other level 1 hot-links. In automatic jump mode, the user instructs the jumper to access all hot-links on the level 1 file 902. In reasoness the improved as on the level 1 file 902. In response, the jumper directs the browser to access and display in the browser window each of level 2 files 926, 934 and 936. These files are accessed by the browser in response to the jumper sending at timed intervals each of bot-links 924, 922, and seconing at timest mistreats each of not-links \$24, \$22, and \$20 from a periad hor-link list to the browner. The browner accesses each of the files \$26, \$34, and \$36 to which hor-links \$24, \$22, and \$20 respectively point. This allows the user to see an animation in the browner window for a timed interval.

In order to bener understand the operation of a jumper, it is necessary to first understand an environment in which the jumper is practiced. FIGS. 1–2, show the hardware envirecognition of the pumper operatus. As shown in FiG. 1, 20 as internst 22 is connected to client 29 by an incoming laternst line 24 and an outgoing lostenest line 26. Jumper 28 and browser 30 are part of client 29, bettere 22 comprises survers 32, 34, 36, and 38. Each server contains HTML files. The client and all servers are linked electronically. Server 34 31 in the part of the server contains the server servers are linked electronically. is shown as the server of an information indexer. The server comprises a CGI 42 (common gateway interface), forms 44 comprises a ULI 45 (common gateway interface), forms 44 on which a user query is entered, and storage containing an updated topical index 46 of all files on the usb. To access a specific size on the network, browner 30 outputs an URL so across outgoing internet line 26. Routers on the internet establish an electronic link between the client 29 and the appropriate across 22-38. Communications with the internet are received on incoming internst line 24.

Hardware:

The client hardware architecture for both the browser and the jumper is shown in greater detail in FIG. 2. The client 20 contains storage 204, computational hardware 202, operat-

ing system 206, and GUI (Graphical User Interface) 200. The storage 204 cos e storage 204 contains a storage segment 236 in which t oper stores persod HTML files. Hardware 202 compris number stores person HTML miss. Harrowns 202 comprises
a microprocessor, as I/O interface, a display and a bus
connection to storage 204. The operating system 204 may
conscious a biol and up preceding convincation such as | ed | 09/27/2005
System 774, Unix 74 or Windows 74. The GUI 200 comprises

System 77th, Unix 1st or Windows 1st. The UUI 200 comprises user I/O 208 and browner 30. Between 30 comprises user I/O 208 and presenter 219. The jumper 28 comprises user I/O 212, presenter 214, and parser 214.

The operation of hardware 202 is controlled by the operating system 204. The operating system 204. The operating system 205 are instrument (GUI) 200, counts the local servironment in which the jumper 28 and browner 30 crist. The browner user I/O 208 handles the tasks of sanding and retrieving communications 218 vin the operating system 206 and hardware I/O 208 handles the tasks of sending and netrieving com-munications 218 via the operating system 226 and hardware 202 to the Internet 22. The browner handles Internet com-munications for the jumper user I/O 212 via jumps-parser I/O protocol 224. The browner user I/O 212 via jumps-parser I/O protocol 224. The browner user I/O also handles the task of translating files received from the internet, e.g., HTML, encoded files, to the presenter 210 via presenter protocol 229. Additionally, browner user I/O uploads at the direction of jumper user I/O 212, HTML encoded files to presse 216 via browner I/O-jumper parser protocol 222. Jumper parser 216 handles the task of converting an HTML encoded file uploaded from browner user I/O 208 into a format suitable for a single jump or automatic jump mode search, as discussed above in connection with FIG. 9B. The parser extracts from the HTML file: an

9B. The parser extracts from the HTML file: an 9B. The parser extracts from the HTML file: an advantisament, a plansity of URLs and topic descriptors corresponding to each URL. The parser passes these via jumper-passer I/O protocol 226 to jumper user I/O 212. Jumper user I/O 212 communicates with jumper presenter 214 via jumper I/O-presenter protocol 228 to display to the user available jump sizes and to provide a user intenface by which the user may input commands. Jumper user I/O also communicates with browser I/O 208 via browser I/O-jumper I/O protocol 224.

As embodiment of the jumper user interface is shown in FIG. 3. This interface allows the user to see all the hot-links on a given level and to select a bot link in either single jump or automatic jump mode as described above is connection with FIG. 7B. The jumper thus cohences the search capebilities of the brow

Jumper window 300 comprises a mens har 302, a button Jumper window 360 comprises a mens har 362, a button har 364, an advertisement area 366, a jump sile window 306, and a chop-down list button 310. The jumper mens har contains aftest entry button 310. The jumper button bar contains a first entry button 312, a pravious entry button 314, a random jump button 316, a next entry button 318, a last entry button 320, a start timer button 325, a stop timer button 326 and a refresh/apdate button 326.

The jump site window 368 has an associated drop down list accessible by selecting drop-down-down list button 310. This list comprises passed hot-links. These hot-links are extracted from a file initially retrieved by the browser. Any one of the hot-links in the parsed list case be selected by the server with a moses. This drop-down list provides one method

ours in the motionax in the purson has can be structed by the user with a mouse. This drop-down list provides one method for selecting a specific internet sits to jump to. Advertise-ment area 306 contains an advertisement period from an HTML page received by the browser. Button ber 304 prowides controls for single jump and seatomatic jump mode. All buttons relate to the bot-links in the jumpus: paged list segardless of how far the user has drilled-down in the browser. All buttons return the user to at least one of the

Page 21 of

First entry betton 312 allows the user to return fro drill-down to the first entry on persed list. Previous entry Counce 316 returns the most of the other too title of the list.

Random jump betton 316 rotums the user to a randomly selected hot-link on the persed list. Next entry betton 318 selects the next hot-link on the list, and last entry betton 320 accesses the last entry on the list. Start timer button 322 initiates the automatic and successive selection of every hot-link on the parsed list and the boover responsive thereto, displays for the user a file retrieved from the URL corresponding to each bot-link. The animation initiated by 15 start timer button is terminated by stop timer button 324.
Refresh/spdate button 326 causes all hot-links in a file Refresh/spdate betton 326 causes all hot-lieks in a file-which the browser has ratrieved to be parsed and opicaded and displayed in the drop-down list of jumper window 380. Additionally, this parsed list is stored in HTML storage segment 230 as shown in FIG. 2. In alternate embodiments the jumper window may take any of several forms. The user interface may include populy or persistent window, a toolber, a menta modification of the browser window, a toolber, a menta modification of the browser window, or the taylor of the browser window, or the use of accelerator keys on the keyboard. FIG. 4 shows a color at browser were inserface and a

3 :

FIG. 4 shows a prior art browner user interface and a query form of an information index provider. This interface allows the user to access web files and also displays the file s the user to access web files and also displays the file mis to the user. The browser interface 460 is that of Netscape Navigator<sup>16</sup>. The browser interface includes a browser menu ber 462, a site window 484, and a window browser means but 402, a site window 404, and a window 406 for viswing a file. The browser means har includes a back button 412 and a home button 414. The file shown in the visw window is the query form 406 of Yahoo 121. The query 35 form has a query field 416.

The browser mean bar 402 generally provides for editing and saving files. Back batton 412 allows the user to move back a search tree in reverse serial order as shown in FIG. 9A. Home button 414 allows the user to retain to their home page 900 in the browser window. The browser site window 404 displays the URL corresponding to the file carrently being displayed, in this case the query form 406 of Yahoo \*\*\*. nems comparison; is miscase insteady own were in lands. The query form is exceeded in a marking language and contains a query field 410 in which the user may enter the search topic for which a topical of the web is desired. In this case, the search topic is "Rat." When the search topic is entered, it is sent by the browser over the web to Server 34 owned by Yahoo. \*\*A and shown in FIG. 1. ng a search:

FIGS. 5A-C show the initial stages of a search session conducted with the jumper and browser. A browser interface 400, a file retrieved by the browser, and a jumper window 300 are shown. These are the basic tools for the demonstrative search. The initial file retrieved from Yabooms in response to the query "Rat" is shown as an upper page portion SUZz and a lower page portion SUZz in FIGS. 5A-B and SC, respectively. These are accessed by scrollable window bar 500. The retrieved file doubtims an advertisement 504, a search result summary 506, a plurality of bot-links of which 508-514 and 588-584 are reference representative hot-link 514, is shown to contain both an URL 516 and a text portion 518. In FIG. 5C a jumper drop down list 586 is shown. In addition, the retrieved file also contains a next feature 588.

se to the entry of the query "Rat" as sh FIG. 4, the query is sent by browser to Yahoo'. The topical

index 46 at Yahoo ", shown in PIG. 1, is sean file 302e-6 containing all related level 1 hot-links and other information is returned to the client 20 and displayed to the user by the browser at 502s-b. Page 502s is captioned with saur by the blowest as 3425-0, rage state in captioned with an advertisement is a march result summary 306, which indicates 74 matches permissing the way real descriptory of the heart 1000/27/2005 links for those matches. Various bot-links with subject

Page 22 of

links for those matches. Various bot-links with subject descriptions appear on the span in bold tent with underlining. These are, for example, 506, "Entertainment, Humor...", 518, "Regional Alternative Theater...", 512, "Space RAT...", 514, "Agrizap, Inc...".
Each bot-link and discussed above is an active reference. For example the bot-link 514 comprises a hyperext identifier "A", an URL 516, a text portion 518 and a closing identifier. The between displays to the uner only text portion 518 and any images that may accompany it. These are highlighted and underlined on the user display to indicate to the user of the way that when they are salected with a mouse a correthe usor that when they are selected with a mouse a corresponding URL will be sent by the browner to the Internet to ses the specific file referred to by the URL

As shown in FIG. 5B, jumper window 300 has been refreshed via refreshysdate button 326, discussed above in connection with FIG. 3. Responsive to selection of refresh/spdate button 326, the file 522-b is passed from the update button 326, the file 362e-b is passed from the browner 38 to the jumper perser 216 as shown in FIG. 2. All hot-links including the referenced hot-links 508-514 and 589-584 are extracted and passed to jumper user I/O 212. These are passed to jumper greatoner 214 and displayed in the jumper window 300. Specifically the advertisement 504 is displayed in jumper advertisement area 366, and all hot-links 508-514 and 588-584 are put in the jumper drop down list. The first of these hot-links 506 is displayed in the jumper 366. jumper site window 346.

jumper site window 368.

In FIG. 5C, the lower portion 502b of the retrieved HTML page is shown. It contains hot-links 590, 582, 584. In jumper window 360 and specifically jumper site window 368 thereof, the jumper drop-down window, 586, is shown. This window has been accessed by the user's selection of jumper drop-down list button 318. The drop-down list contains all hot-links that were obtained from the original file 502a-b including the reference to FIG. 6. a later stam of the search.

including the referenced bot-links 508–514 and 580–584. Now, by reference to FiG. 6, a later stage of the search session is shown. FIG. 6 shows a file 600 in browser view window 406, an URL corresponding to file 600 in browser site window 404, and a bightlight 602 around hot-link 580 in both the jumper drop-down list window 586 and the jumper site window 308. The file 600 was obtained in a drill-down conducted in the browner window 406. The steps in that drill-down intervening between FIG. 5C and FIG. 6 have not been shown. The drill-down however mannhles that shown been shown. The drill-down however resembles that shown and discussed above in connection with FIG. 9A. The file 600, shown in browser window 406 is four levels removed from the initial level 1 bot-liok 580. The drill-down was conducted on the browser view screen by selecting the hot-links presented in the files retrieved by the browser. If the user desires to return to the as other hot-link from the level 1 file, shown as 5020-b in FIGS. 5A-C, from this file 600, they could use the browser menu bar to do so but it would be slow. They would have to select back button 412 four times to return them one level at a time in reverse serial four times to return them one level at a time in reverse serial order through their entire search back to the level 1 file. The process is time consuming. Alternately, if they selected the browser bome key 414 they would return to their home page and loss the level 1 search results. Neither of these alternatives is acceptable. Both of these alternatives are cumber-

automatic and successive selection at a timed interval of every bot-link on the parsed list. In response to the automatic jump mode selection, each file corresponding to the selection catch file corresponding to the selected hot-link will be retrieved and displayed by the browner for the timed interval. As a third option, the user may select desp-down window beaton 310 and select a given entry from the drop down list by clicking on it with a mouse. In response to their selection the file corresponding to the selected bot-link will be retrieved and displayed in the browner window 466. Alternately, another instance of browner can be opened to display the selected file. All of these choices enhance the search, and most require multiple coverne serial order steps to return to the level I hot-links. Process Flow:

Process Flow: FIG. 7 shows the proces FIG. 7 shows the process connected with conducting a search willking a prior art browner. Commencing with start process 700, a connection process 702 results in Yahoo's 12 bones page 406 and queey from being displayed is browner window 406, as shown in FIG. 4. Then in query process 704, the user enters a search topic, e.g., "Rat", is information indexer query field 410. In response to the entry of that search topic, the browner sends the query to server 34 of the search topic, the browner sends the query to server 34 of the search index provider, e.g., "Ahoo 12, the relavant topical identifiers and URLs are uploaded on an HTML encoded file/page and presented to the user in page receipt process 786. Subsequently, when a user utilizing a mouse selects one of the host-links shown on that page, e.g., 1922—6, then that 35 selection is detected in site access decision 708. The URL corresponding to the last selected host-link is output over the Inserest to access the file indicated by that URL and that file is uploaded and displayed in access process 710 to the user. as connected with condu inserest to access the file indicated by that URL and that file is uploated and displayed in access process 712, that URL is added to a stack comprising in LIFO order the most recently accessed files, including the complete URL for those files. Control is then returned to back-one decision 714 which decision is also reached if a negative determination is made in site access decision 788. In back-one decision 714, a determination is made as to whether the user has selected ack betten 412 on the browner mean 402. If thet selectio back common \*1.6 on his occurrent mann was, it that selection has been cade, then control is peased to repeat history process 71.6 in which the previous sits visited in popped from the history stack and in access process 71.6 the browser accesses the sits and file indicated by the prior URL and presents that in its view window to the neer. Control is

equently returned to site access decision 762. ternately, if in back-one decision 714 a negative deter-tion is reached the browser awaits the next user comination is to mand 729.

in FIGS. 8A-C the process con ment of the current invention is shown. It contrasts with the prior art process, in that the user may return to the level of files pointed to by any level one bot-link immediately. The composition to op any level one bot-link immediately. The combessome process and many steps of reverse serial order traversal discussed above in Fig. 7, are not therefore necessary. Fig. 2A is an overall view of the process connected with either the single jump or automatic jump mode. Fig. 3B details the specific steps connected with the submatic jump mode. Fig. 3C details the specific steps connected with the single jump mode. 10

Commencing with FIG. 8A and start jumper process 800 nigol is passed to refresh decision 802. In the event, a Commencing with FIG. 8A and start jumper process 800 control is passed to refresh decision 802. In the event, a determination is made that refresh/update button 326 shown in FIG. 3 has been selected, then control is passed to frich and purse process 804 in which an HTML encoded page control of the buyerser where valued in episoded and end of the buyerser where valued in episoded and end of the played in the jumper advertisement and bot-links. These are displayed in the jumper advertisement area 306 and the jumper cop-down window 386, enspectively. Then is history decision 806, the determination is made as to whether the jumper listorer maintain only in malerted. This notion is drop-down window SE6, respectively. Then in binory decision 896, the determination is made as to whether the jamper history maintain option has been selected. This option is found under the edit portion of the messe ber 302. In the event that this option has been selected, then control is passed to appead and store process 883 in which the hot-links entracted in fastch and purse process 804 are appended and stored with previous hot-links in the pursed HTMC Size in storage segment 230 shows on FIG. 2. Coutrol is then passed to display URL process 812. Alternately, if a negative determination is made in history decision 806, then countrol is passed to store process 812 in which the hot-links from the fauth and purse process 812 in which the hot-links from the fauth and purse process 812 in which the hot-links ground hot-links. Subsequent control also is passed to display URL process 812. In display URL process 812, the hot-links are displayed in the jumper drop-down window 586 as shown in FIG. 6. Control subsequently passes to display advertisement process 814 in which an image corresponding to an advertisement passed from the browner page is placed in the jumper advertisement area. 306.

As discussed above in connection with refresk decision

As discussed above is connection with refresh decision 842, if a negative determination is made that the refresh button has not been pressed, then control passes also to display advertisement process \$1.4 in which the advertisemanny arrentments process 91.4 m which the advertise-ment from the page currently displayed on browner window 406 is uploaded and displayed in the jumper advertisement area 306. Control subsequently passes to start timer decision \$16. If the user has selected start timer button 322 from the alls, it lies terr man watered with their could are into the jumper means but, then automatic jump mode commences in a manner set forth in Fig. 8B. Alternately, if a orgative determination is reached, then control is passed to jumper button select decision SIB. A determination is made as to whether the user has selected say one of buttons 310–320 in the jumper window 300. In this event, control is passed to the jumper window 309. In this event, control is passed to processes set forth on FIG. 8C and corresponding to single jump mode. The return from either the automatic jump or single jump process is via splice black B to jumper button select decision 818. Finally, when a negative determination is reached in jumper button select decision, coatrol returns to refresh decision 802. This completes the overall process-

to refresh docision 802. This completes the overall processing connected with the jumpers tree traversal.

In FIG. 88, the automatic jump mode of the search process is shown in greater detail. This process allows a user to view an animated tour of the level 2 files pointed to by the level 1 but lists in the browser window. Communicing with splice block A, control passes to memory fatch process 846 in which the first URL, of the hot-links in storage segment 230 shown in FIG. 2, is fatched by the jumper, in the next process, jumper-browser process 842 the first URL is seet to the browser, user I/O 208 caused the browser to access the specific site and file on the site, to upload the file and to display the file in the browser window 406. In one embodiment, if the browser is still leading a servious file cusping the in the second second as previous file when the jumper-browner process 842 is succeed, then sending the browser the URL interrupts the leading of the previous file. In another embodiment, the jumper-browser process \$42 does not interrupt the leading of the previous file. In a third embodiment, whether the jemper

Page 23 of

process set interrupts the loading of the previous file is a user configurable option.

Courol is then passed to delay process \$44 in which a user selectable display interval causes the page image to be displayed on between window 446 for the selected time interval. At the end of that time interval control is a selected time. displayed on browner window was fir the statement time interval. At the end of that time interval, control is persued to parend by facily being \$46 in price the play (BL) in parend HTML file is uploaded and in last of list decision \$48 a determination is made as to whether that URL is the last among the secred URLs. In the event that determination is in among the source URLLs at the event that constitution is in the negative, control returns to jumper-between process \$42 for the timed display of the sent accessed URL. This loop from jumper-browner process \$42 through last of list deci-sion \$48 constitutes a weer configurable animation, in which a set of URLs which have been passed and stored are now 15 used to drive an automatic retrieval and display process.

Alternatory, in last URL decision 848, if a determination is made in the affirmative, coatrol is passed to "seat" link decision, 859. In sext link decision 838 a determination is made as to whether the last URL also includes a query for the next 18 corrise corresponding to the example shown in FIG. 5C and indicated by 58S. In the event that determina-FIG. SC and indicated by SSE. In the event that determina-tion in its the affirmative, control is passed to jumper-browser-jumper process SSZ. In process SSZ the query for the next 10 hot-links is sent to Yahoo<sup>181</sup>. In response the next 13 HTML encoded page containing the 10 hot-links is retrieved. Control is passed to purse process SS4 in which all the URLs and corresponding text identifiers are passed along with the image corresponding text identifiers are passed along with the image corresponding to the advertisement. Then in buffer process SS6, the parage cutyet is stored for this 10 session. Control thee passes to memory fatch process S48 for entomatic retrieval and display. Alternately, if is "max" link decision SS9, a sequive dearmination is made, then in for estomatic retrieval and display. Alternatally, if in "mext" link decision \$50, a negative determination is made, then in jumper-browser process \$52, the last URL is sent to the browser to both access and display the file corresponding to 35 the last URL. Subsequently, council is passed to splice block. B for a return to bot-link select decision \$55.

The process corresponding to the single jump mode is shown in FIG. \$C. This process allows the user to select a contribution of the single jump mode in the state of the select and the selec

wife level 1 hot-link and have the browser retrieve and specific level 1 hot-link and nave us occurs to the selected level 1 hot-link. In window process \$25, the jumper deep-down window 556 is automatically operad, as is above in FIG. SC. Then in drop-down betton process \$82 a determination is made at to whether the betton selected by the user and detected in jumper button select decision \$16. was thoughdown list button 310. If the user did select button 310. drop-down list buston 310. If the user did select buston 310 than the user has not yet selected a specific hot-link from the parsed hot-links in the jumper drop-down window 586. Control is then pessed to wait process 884. When the user so selects a specific entry from the drop-down list control passes to locate select process 886. Alternately, if a segative determination is reached in jumper buston select decision 618 these control is pessed directly to locate select process 886. A negative determination in jumper select decision 818 means that one of jumper bustons 312, 314, 316, 318 or 328, was selected thereby indicating a specific user choice for the single jump location. These buttons were discussed above in connection with FIG. 3.

2

In locate select process 886 a determination is made as to which of the hot-links in the jumper drop-down window was d. If, for example, first entry button 312 were selected then the first of the hot-links in the list is selected.

Alternately, if the user selected drop-down window button 318 and then mouse selected an entry e.g. the footh entry on 45 the drop-down list, then that entry would be highlighted. In the example shown in FIG. 6 the user has selected hot-link 12

588 and that selection, the "Ricky Rock Rat"hot-link is detected by the jumper.

Subsequent to selection detection, control is passed to jumper-browner process 888. The URL corresponding to the

jumper-browser process \$88. The URL corresponding to the selected hot-link is passed by the jumper to the browser causing the browser to locate and subsequently display a file corresponding to the REAL The jumper continues to display 9/27/2005 all level "I" make at hot-links, in the next step highlight 9/27/2005 process \$90 the selected hot-link is highlighted, as shown for exampte in Fig. 6 as highlight 602 around selected hot-link \$88. "Richy Rock Rut". After the highlight process, the selected hot-link on the drop-down list is copied to the jumper sits window 368 in copy process \$92. Abbsequently, in process \$94 the jumper drop-down window is closed.

When the browser displays the above mentioned file, the user may conduct a drill-down on the hot-links in that file in the browser window. In that case, the browser window is

user may conduct a drill-down on the bet-inks in that his in the between window. In that case, the browser window is active and the jumper window is inactive. When the drill-down is complete and the user reactivates the jumper window by mouse clicking on it, that reactivation is detected in percess \$96 and control is pessed to jumper miact decision \$1.5 via the splice block B, shown in FIG. \$A. The following describes additional embodiments of the invention. Except where stated in alternative form, each of these ambodiments include features that can be combined

se embodiments include features that can be see with the factures discussed above.

Some alternative embediments provide better integration Some atternative embodiments provide belier integration of the jemper's functions and the browsers functions. In one embodiment of the invention, whon a user initiates a search in the browser, the jumper automatically starts and begins parsing the results of the search. This saves the user from paring the results of the search. This saves the war knowledge paring to start the jumper separately from the browser. In another embodiment, the jumper functions are bell directly into the browser. In a different embodiment, the jumper into the browser. In a different embodiment, the jumper is a servicesting, such as an applet, which is mso the browser. In a different embodiment, the jumper is implemented as an application, such as an applet, which is sent to the browser by the search engine. All of these embodiments provide a more integrated jumper/browser environment for the user.

Some alternative embodiments provide the user with Some alternative embodiments provide the user with more powerful tools for traversing the search results, it one embodiment, the categories in the search results are specially tagged (e.g., with a previously unused HTML tag) to indicate category fields as opposed to simple URLs. Alternatively, the categories can be passed given their location in the search result (e.g., not indented). In an embodiment that includes category identification, the jumper includes functions for jumping from one category to the next category, in addition to being able to jump from one site identifies to the sais identifier. As an enhancement to this subportionest, the user is necessaried with additional buttons. embodiment, the user is presented with additional bettoms for jumping from category to category. In another embodiment, the user can define how many site identifiers should be persed from the search results. For example, if the search results provide twenty site identifiers, but the user may only want the first five identifiers, the user can specify that only the first five identifiers be provided. Similarly, in another embodiment, the user can specify what types of results should be parsed (e.g., only categories). Alternatively, in another embodiment, the jumper partes all the site identifiers, but the user specifies how many or what type to display.

In another embodiment, the user will be able to invoke the product from within their electronic e-stall box simply by double-clicking on attached files. These files may be encoded in markey language. In another embodiment, the HTML parsed files for both transmission and receipt by users would be able to be compressed and decompressed by

Page 24 of

users with simple click commands common to normal software operations. In another embodiment, the product will not require an additional instance of the presenter to be will not require an additional instance of the presenter to be invoked upon return to the original home page. In still another embodiment of the product, the list box will contain 3 a drug and drug apprinterface which will allow the weet to cut and page Will a from one list to mobiled that compile their own individual lists to their preferred selection and taste, in still one more embodiment of the product, the uner will be able to adjust and modify the overall interface of 19 various URLS, but links and other files viewable within the research to both highlight the various objects mentioned various URLS, not-links and other files viewable within the presenter to both highlight the various objects sentioned previously for marking the users piece in the list that they are working from, and similarly be able to change the nature of the highlight of such previously mentioned item to be noted 15 as important by a permanent highlighting of the particular

object.

Thus, a method and appearants for retrieving info has been described. Note that though the foregoing has particular stillty and has been described with reference to certain specific embediments in the figures and the text, that one may practice the present invention without implement-ing all of these specific details. Thus, the figures and the text are to be viswed in an illustrative sense only, and not limit the present invention. The present invention is only to be 25 limited by the appended claims which follow.

#### DEFINITIONS

- 1. NETWORK: A network is a hard drive, a local network, a wide area network, an intranet, the intermet or any series.
- a wide area network, an intrust, the minute or any sensor or combination of computers or computing hardware.

  2. FILE: A file is a collection of data that may be coded or mescoded. Coded files may contain the HTML, SGML or other mark up language. Unescoded files comprise; satio, visual, graphics, and/or video. Files may be encrypted or 35 reconcrepted during transmission. An electronic mail mes-sage is also considered to be a file. In addition, anach-ments to electronic mail are also considered files.
- PRESENTER: A presenter is an interactive information media either visual or sadio, animated or static, graphical or tentual, andio or silent.
- 4. SITE IDENTIFIER: A site identifier is a pointer to a file. What is claimed is:
- What is Games in:

  1. A computer implemented method for searching on a
  local computer a natwork of nucles with data files stored at
  corresponding cases of the nodes and such of the data files
  identifiable by a location identifier and several of the data
  files, and the method for searching comprising the acts performed on the local computer of:
  - constructing a search window on a display screen of the local com
  - displaying a first and a second icon separate from the search window on said display screen;
  - retrieving as initial data file from the network together with displaying the initial data file in the search window, and the initial data file including location identifiers;
  - persons the location identifiers from the initial data file to go form an initial list of location identifiers together with storing the initial list, responsive to a selection of the
  - restroving a first data file corresponding to a selected one of the location identifiers in the stored initial list as together with displaying the first data file in the search window, responsive to a selection of the second icon.

14

- had of claim 1 wherein 2. The computer imple said initial data file comprises information is a markup language; and

- language; and
  said location identifiers comprise URLs.
  3. The computer implemented method of claim 1 wherein:
  Subjected file and said first data file comprise information | ed 09/27/2005
  in a methop language; and
  said location identifiers comprise URLs.
  4. The computer implemented method of claim 1 wherein

Page 25 of

- said retrieving act farther comprises;
  - retrieving the first data file corresp location identifiers in the stored ing to the one of the trieving the first daits the corresponding to use one of the location identifiers in the stored initial list selected from a group consisting of: a next location identifier, a prior location identifier, a first location identifier and a last location identifier, together with displaying the first data file in the search window, responsive to a assection of the second icon.
- 5. A computer reable medium having computer readable program code means embodied therein for searching on a program code means embodied therein for searching on a local computer a network of nodes with dut files attend at corresponding ones of the nodes and each of the data files identifiable by a location identifier and saveral of the data files containing location identifiers for others of the data files, the computer readable program code means in said article of manufacture comprising:
  - computer readable program code means for causing a computer to construct a search window on a display computer to con screen of the local computer,
- computer medable program code means for causing a computer to display a first and a second icon separate from the search window on said display screen;
- computer readable program code means for causing a computer to retrieve an initial data file from the activities and displaying the initial data file in the search window, and the initial data file including location dentifiers;
- computer readable program code means for causing a computer to passe the location identifiers from the initial data file to form an initial list of location idenis together with storing the initial list, respon a salection of the first icon; and
- emputer readable program code means for causing a computer to reciseve a first data file corresponding to a selected one of the location identifiers in the stored initial list together with displaying the first data file in the search window, respon ive to a selection of the
- second ione.

  6. The computer machible program code means in said article of manufacture of claim 5 comprising:
- computer readable program code means for causing a computer to retrieve the initial data file, wherein said initial data file, comprises information in a mariosp language and said location identifiers comprise URLs.
- 7. The computer readable program code means in said nicle of meanfacture of claim 5 comprising:
- computer madable program code means for causing a computer to retrieve the initial data file and the first data file, wherein each of said initial and said first data files, ne, women cach of said must and said mist dass lies, comprise information in a markets language and said location identifiers comprise URLs.

  3. The computer readable program code masse in said article of manufacture of claim 5 comprising:
- computer readable program code means for causing a computer to retrieve the first data file corresponding to the one of the location identifiers in the stored initial list

selected from a group consisting of: a next location identifier, a prior location identifier, a first location identifier and a last location identifier together with displaying the first data file in the search window,

responsive to a selection of the second icon.

2. A computer implemented sected for exacting on a computer implemented sected for exacting on a computer interest of nodes with detaillier storage of the computer interest of nodes with detaillier storage of the computer interest of nodes with detaillier storage of the computer interest of nodes with detaillier storage of the computer interest of nodes with detaillier storage of the computer in ding ones of the nodes and each of the data files identifiable by a location identifier and several of the data files containing location identifiers for others of the data 10 files, and the method for searching, comprising the acts performed on the local computer of:

constructing a search window on a display acrees of the local computer;

displaying a first and a second score separate from the 15 

retrieving as initial data fils from the network together with displaying the initial data file in the search window, and the initial data file including location

parsing the location identifiers from the initial data file to form an initial list of location identifiers together with storing the initial list, responsive to a selection of the

atomatically retrieving at a prodefined time interval data files corresponding to each of the location identifiers in the stored initial list, together with successively dis-playing the data files in the search window, responsive to a single selection of the second icon.

10. The computer implemented method of claim 9

said initial data file comprises information in a markup

said location identifiers comprise URLs.

11. The competer implemented method of claim 9

said initial data file and said first data file comprise information is a markup language; and

said location identifiers comprise URLs.

12. A computer usable medium having co nguter readable ogram gode means embodied therein for for search per a perwork of nodes with data files stored at corresponding ones of the nodes and each of the data files. dentifiable by a location identifier and several of the data files containing location identifiers for others of the data files, and the computer readable program code means in said article of manufacture comprising:

computer readable program code means for causing a computer to construct a search window on a display screen of the local computer;

computer readable program code means for causing a computer to display a first and a second icon separate from the search window on said display screen;

computer readable program code means for causing a computer to retrieve an initial data file from the network together with displaying the initial data file in the search window, and the initial data file including loca-

computer readable program code means for causing a computer to parse said initial data file to form an initial list of location identifiers together with storing the initial list, responsive to a selection of the first icon;

computer readable program code means for causing a 65 computer to automatically retrieve at a predefined time rval data files correspon oding to each of the location

16

identifiers in the stored initial list, together with suc-cessively displaying the data files in the search window, responsive to a single selection of the second icon.

13. The computer readable program code means in said

article of magazacture of claim 12 comprising:

computer readable program code means for causing a computer readable program code means for causing a computer professional distribution for the code of the code

Page 26 of

14. The computer readable program code mesos in said article of manufacture of claim 12 comprising:

computer readable program code means for causing a date file, wherein said initial date file and said first data file, comprise information in a market language and said location identifiers comprise URLs. 15. A computer-implemented method for searching on a

local computer a network of nodes with data files stored at corresponding ones of the nodes and each of the data files identifiable by a location identifier and several of the data files containing location identifiers for others of the data flies, and the method for searching comprising the acts performed on the local computer of:

constructing a browser window on a display screen of the local com

displaying a first icon and a list window separate from the browser window on said display scree

retrieving into the browser window an initial data file from the network together with displaying the initial data file in the browser window, and the initial data file including location identifiers;

parsing the location identifiers from the initial data file to form an initial list of location identifiers together with storing and displaying the initial list in the list window, responsive to a selection of the first icon; and

retrieving a first data file corresponding to a one of the together with displaying the first data file in the browser window, responsive to a selection of the corresponding one of the location identifiers displayed in the list

16. A computer-implemented method of of claim 15.

wherein said initial data file and said first data file comprise information in a markup language and said

location identifiers comprise URLs.

17. A computer usable medium having o program code means embodied therein for for searching on a local computer a network of nodes with data files store corresponding ones of the nodes and each of the data files identifiable by a location identifier and several of the data files containing location identifiers for others of the data files, the computer readable program code means in said article of manufacture comprising:

computer readable program code means for constructing a browser window on a display screen of the local computer; computer readable program code means for displaying a first icon and a list window separate from the browser window on said display screen;

computer readable program code means for retrieving into the browser window an initial data file from the network together with displaying the initial data file in the browser window, and the initial data file including location identifiers;

computer readable program code means for parsing the location identifiers from the initial data file to form an

initial list of location identifiers together with storing and displaying the initial list in the list window, respon-

computer rectable program code means for retrieving a first data file corresponding to a case of the location identified digitaged in the file window together with displaying the first data file in the because window, responsive to a selection of the corresponding one of the location identifiers displayed in the list window. 18

15. The computer restable program code means in said

computer readable program code means to retrieve said initial data file of information, wherein said initial data

-RS to comprise information in a merbyp language and illed 09/27/2005

9 -



CERTIFICATE OF MAILING BY "EXPERSS MAIL"

"Expens Mail" Mailing Label No. 2040203309UX.

Date of Chopsell's Counter & 1976. Linearly earlify that
this paper or fine is being deposited with sufficient pentage with the
United States Pound Services "Express Mail Pout Office to Addresses"
surplys under 37 CPR. Life on the date indicated above and is subsessed to:

But Part of Parties and Counter Cou

Document 54-5

Filed 09/27/2005

Page 28 of

Homel Lindows
(Cyped or French Mining Pages or Fee)

NOT WALTON (6/8/96

Streets of Name Making Engine (Fee)

PATENT Attorney Docket No. 18041.701

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

## UTILITY PATENT APPLICATION TRANSMITTAL LETTER

Box Patent Application Assistant Commissioner for Patents Washington, D.C. 20231

Şir:

Enclosed for filing is [x] an original patent application or, [] a continuation-in-part patent application, by Gilbert Borman, Rajat Bhatnagar, And Sebastian, Anun Mathur.

## Vinay Wadhwa and Mukesh Kimar, for INTERNET SEARCH TOOLS

#### Also enclosed are:

- [X] 14 sheet(s) of [] formal [X] informal drawing(s);
  a claim for foreign priority under 35 U.S.C. §§ 119 and/or 365 in
  [] a separate document [] the declaration;
  a certified copy of the priority document;
  an Associate Power of Attorney;
- [] \_\_\_\_ verified statement(s) claiming small entity status; and
- [] an Assignment document and form PTO-1595.

The declaration of the inventor(s) [] also is enclosed [X] will follow.

0108488.01

## Attorney Docket No. 18041.701

\$1142.00

The fee has been calculated as follows:

Case	2:04-ev-1	70366-JAU	HRSW.	Dasúme	ot 54-5 Filed	09/27/200
	NO. OF CLAIMS		EXTRA CLAIMS	RATE .	· FEE	
Basic Applicati	ion Fæ				\$770.00	
Total Claims	26 -	MINUS 20	6	\$22.00-	\$132.00	
Independent Claims	6	MINUS 3	3	\$80.00=	\$240.00	
If multiple dep	endent claim	-0-				
						A T
If verified state 50% of Total	ement claimir	-0-				
Add Recordin		-0-				

۲٦	À	chacle	in the	amount	of S	is At	closed
11	А	cneck	in the	amount	OL 3	_ I3 CI	いいつうじゅん

TOTAL APPLICATION FEE DUE

- [] Charge \$\_\_\_\_ to Deposit Account No. 23-2415.
- [X] The fees will be paid with the response to Notice of Missing Parts of Application.

Respectfully submitted,

WILSON, SONSINI, GOODRICH & ROSATI

Bv:

Registration No. 36,764

650 Page Mill Road Palo Alto, CA 94304-1050 (415) 493-9300

Date: October 8, 1996

0108488.01

-2

THE PROPERTY OF THE PROPERTY O

A Justee 11 727085

Document 54-5

Filed 09/27/2005

Page 30 of

Sale of Opposit Ordinar 6, 1980; I hardy centify that this paper or he fit in dependent with mediatest produce with the United States Potati States of States Potati States of States Potati States of States Potati States of Sta

(Nuating 10/s/pr

PATENT Attorney Docket No. 18041.701

### INTERNET SEARCH TOOLS

Inventors: Gilbert Borman Rajat Bhatnagar Arul Sebastian Anup Mathur -Vinay Wadhwa Mukesh Kumar

### BACKGROUND OF THE INVENTION

#### Field of the Invention

The present invention relates to information retrieval. More specifically, the invention relates to tools for traversing hypertext data.

#### Background Information

5

10

15

D8727085.100896

The development of computerized information resources such as the Internet and various on-line services, such as CompuServe™, America On-Line™, Prodigy™ and other services has lead to a proliferation of electronically-available information. This electronic information is increasingly displacing more conventional means of information transmission, such as newspapers, magazines and even television.

The largest information resource in existence today is the Internet.

The Internet is a group of client and server computers linked one to another and each having a unique identifier, DNS (distributed network server), assigned by the Internet authority in Cambridge, Mass. and Geneva, Switzerland. In order for information to be found on the Internet every file is given a specific address by which it may be located. To access the Internet a

I:\PUBLIC\CC7\0108301.07

2

15

20

.25

30

## Case 2:04-cv-70366-JAC-RSW Document 54-5

user employs what is called a browser. Currently the most popular browser is Netscape Navigator<sup>TM</sup> browser developed by Netscape Communications
Corporation of Mountain View, California. A wide array of browsers is available for just about every platform. The browser's job is twofold. First, given a pointer to a piece of information on the net it has to be able to access the information or operate in some way based on the contents of that pointer. Second, if the document/file (hereinafter file) is encoded the browser has to translate that to a suitable format for display to the user. The display may include multimedia effects, e.g. sound and animation.

The most popular encoding of Internet files communicated between client and server is the HTML (hypertext markup language). The WWW (World Wide Web) or simply the "Web" includes all the servers adhering to this standard. Each page loaded from the Internet is a single file encoded in HTML HTML describes the structure of a file. The structure of the file includes title, paragraphs, images and any pointers to other files.

The pointer to a specific site is called an URL (Uniform Resource Locator). The URL provides a universal, consistent method for finding and accessing information for a Web browser. The URL comprises a file type, a server I.D. (DNS), one or more directories and subdirectories, and a file name. URLs are also used as part of a hypertext link within a file to another file. These URLs then provide the browser with a way to navigate the Web. URLs contain information about a file: including file type (FTP, Gopher, HTTP), the Internet server on which the file is located (WWW.NCSA.UIUC.EDU, or FDP.APPLE.COM, or Net Com 16.Net.Com, and so on), the directory of the file, and the file name.

In order to speed the process of finding relevant information on the Internet several servers on the Internet provide an index to the Internet and a search engine. These information indexers such as Yahoo™, Excite™, Lycos™, Inktomi™, and Alta Vista™ perform two valuable functions. First,

L1PUBLIC\CC7\0106301.07

Filed 09/27/2005

B6800T

5

10

15

20

25

using their own internet links, they continually search the Internet and index all files on the Internet into subject categories and store this index on their own database. The information indexers also allow a client to connect to their server and enter a search query. In response the information indexer provides a list of all files on the web that meet the search criteria. Therefore, the information indexer such as Yahoo<sup>TM</sup> not only updates and maintains a topical index for all files on the web, but also makes that index searchable by a client. It should be noted that the information that is retrieved from Yahoo<sup>TM</sup> contains only a general topic identifier and the file location on the Internet for that specific topic.

it would indeed be a cumbersome process for the client seeking specific information if the indexes that were retrieved from Yahoo™ only told the searcher where to look. If this were the case, the considerable task remaining to the client would be to manually enter the network address, URL, of each file and the go through the process of retrieving that file. To overcome this problem, the search result retrieved by Yahoo™ is encoded in HTML as a hot-link which makes every "footnote" an active rather than a passive reference. These hot-links appear to the user in the browser window as bold face text which is easily distinguished from the other text based information in the file. A hot-link comprises a text description and a corresponding URL When the user selects a hot-link the browser detects that selection and outputs the URL on the internet to ratrieve the file corresponding to that URL and display it to the user. Therefore, by merely selecting with a mouse a specific footnote in a file encoded in a markup language, a client is-Immediately given access to the remote web server that contains the specific file referred to in the footnote.

With a markup language such as HTML (hypertext markup language) both amateurs and professionals become authors and the footnotes on the printed page become the hypertext of the electronic page. What was a passive reference now becomes an accessible link to a related file. A markup

EVPUBLICYCC7/0106301.07

4

10

15

20

25

30

## Case 2:04-cv-70366-JAC-RSW Document 54-5

language describes the structure of a file including headings, paragraphs, images and what are called hot-links. A hot-link displays at the user level as text or graphic and is processed for communication purposes as an-URL. It is these hot-links which provide the interactive footnotes described above.

Even with the indexing provided by Yahoo TM, Lycos TM, Excite TM, Inklomi TM, Alfa Vista TM, etc., the process of finding the exact topic is still extremely time consuming and can involve visiting literally hundreds of Web Sites. Typically, a user will retrieve a file from an information indexer and will not only look at the files retrieved by selecting the indexer's hot-links, but will also select other hot-links in the retrieved documents. This process of starting a search that begins with an initial hot-link and following a search trail that leads to successive files each increasingly displaced from the starting point is known as a drill-down. The problem with current browsers is that when a user has drilled-down through many levels of sites, the only way to return to the original HTML file is to hit the browser's back key which moves the user up one level at a time through the original search tree back to level "1." Only then can the user access other hot-links retrieved in the original search.

What is needed is a more efficient way to conduct a search.

#### SUMMARY

A first object of the invention is to provide operational controls for simplified internet navigation from various sites and back again.

A second object of the invention is to save users time and effort in finding information on the internet.

A third object of the invention is to enable the publication of a new format for on-line magazines, called Netazines, which allow publishers to leverage the navigational controls provided in the invention product to foster a new paradigm for browsing magazine style information on the Internet.

tyPUBLICYCC7/0108301.07

Filed 09/27/2005

15

20

25

A computer implemented method and system for retrieving information from a network. In a first embodiment a first file of information is received which may include a first mark-up language to identify contents of the information, which contents include site identifiers. The site identifiers corresponding for example to file locations on the Internet. The first file is displayed in a browser window. Responsive to receiving the first file of information by the browser, the first file of information is parsed by a jumper to generate a list of site identifiers. This list of site identifiers is then stored by the jumper and displayed in a jumper window. Responsive to an activation by the user, a computer is directed to determine which of the stored site identifiers is currently selected and automatically selects an other. The other includes the first, the prior, the next, or the last on the list.

In a second preferred embodiment a first file of information in a hypertext markup language is received and displayed in the browser window. The first file of information also contains site identifiers and other information. The first file is displayed in the browser and is parsed, and the site identifiers from that file are stored by the jumper in a list. The stored list of site identifiers is displayed in the jumper window. Responsive to a selection by a user of automatic mode, a computer is directed to perform the following steps. The jumper selects a first site identifier from the stored list of site identifiers. The jumper directs the browser to access the file at the site corresponding to the automatically selected site identifier. The browser, is directed by the jumper, to display the file in the browser window. Then the jumper initiates a delay of a predetermined interval. At the end of the interval the jumper selects the next site identifier from the stored list of site identifiers. The jumper directs the browser to access the file at the site corresponding to the automatically selected next site identifier. The browser, is then directed by the jumper, to display that file in the browser window. The jumper continues to repeat this animation loop from one site identifier in the stored list of site identifiers to the next, until either the entire list of site identifiers has been sent to the browser and displayed, or until the user terminates the process. In this manner, an

1:1PUBLIC\CC7\0105301.07

15

20

25

30

Case 2:04-cv-70366-JAC-RSW Document 54-5

animated tour of all files having site identifiers parsed from the first file is displayed in the browser window under the direction of the jumper.

In a third embodiment, a first file of information in a hypertext markup language is received and displayed in a browser window. The first file of information also contains a site identifier and other information. The first file of information is displayed by the browser in the browser window. Responsive to receiving the first file of information, the jumper parses the first file and extracts and stores a list comprised of first file site identifiers. The stored list of site identifiers is then displayed in the jumper window. The user is allowed to select from the browser window a specific site identifier, known as a search level "1" site identifier, and responsive thereto to cause the browser to access and display a second file retrieved from the site corresponding to the selected level "1" site identifier. This second file includes a hypertext markup language and site identifiers. The user is again allowed to select from the browser window a specific site identifier and responsive thereto causes the browser to access and display a third file retrieved from the site corresponding to the selected site identifier. This process can be repeated in the browser, until the user has completed their search, or drill-down.

Responsive to the completion of the drill-down in the browser, as indicated to the jumper by receipt of a single jump selection from the user, the computer is directed to perform the following steps. The jumper determines which of the stored level "1" site identifiers is currently selected and automatically selects the single jump level "1" site identifier. Next the jumper directs the browser to access the file at the site corresponding to the single jump selected level "1" site identifier. Finally, the browser is directed to display the file the browser has retrieved in the browser window. This allows the user to return to the files pointed to by the original level "1" site identifiers without having to traverse in reverse serial order the site identifiers selected in the drill-down.

INPUBLICICCT/0106301.07

G 000106

Filed 09/27/2005

## BRIEF DESCRIPTIONS OF DRAWINGS

The present invention is illustrated by way of example and not limitation in the figures in which like references indicate like elements and in which:

Fig. 1 shows a client server system including embodiments of the jumper invention.

Fig. 2. shows a more detailed view of the client server system Illustrated in Fig. 1. .

Fig. A. shows a preferred embodiment of the screen interface and tool bar of the jumper.

Fig. 4 shows a prior art browser user interface, and the query form of an information index provider.

Figs. 5A-C shows a the initial stages of a search session utilizing a prior art browser and the jumper.

Fig. 6 shows a later stage in the search session of Figs. 5A-C.

Fig. 7/shows the steps in the search process in prior art browsers.

Figs. 8A-C shows the processes of a search session on the jumper.

Figs. 9A-B show the search traversal patterns of a prior art browser and the jumper.

## DETAILED DESCRIPTION

One embodiment of the invention is a method and apparatus for retrieving information through a browser connected to a network, and providing the information to a user. Although the following will be described with

L'PUBLICYCC7/0108301.07

G 000107

08727085,100896

10

15

25

15

20

25

30

# Case 2:04-cv-70366-JAC-RSW

Document 54-5

Filed 09/27/2005

reference to certain particular embodiments, including data structures, flow of steps, hardware configurations, menu configurations, etc..., implementation of the invention can be practiced without these specific details.

One embodiment of the invention allows a search to be conducted in non-linear order. This may be done in either single jump or automatic jump mode utilizing a jumper. A search tree does not therefore need to be traversed in reverse serial order after a drill-down as is required utilizing prior art browsers. The invention allows a searcher to jump across multiple levels at a time.

#### Search Levels:

The operation of prior art browsers is best described by reference to Fig. 9A. Fig. 9A shows a series of hypertext files. The files may be resident on a hard drive, a local network, a wide area network or the Internet. Three hot-links are shown 920, 922, 924 on file 902. Hot-link 924 is shown as linked to four levels of files. Selecting hot-link 924 causes the browser to access file 928, which is a level 2 file. Selecting the first hot-link on file 926 causes the browser to access file 928, which is a level 3 file. Selecting the first hot-link on file 928 causes the browser to access file 930, which is a level 4 file. Finally, selecting the first hot-link on file 930 causes the browser to access file 932, which is a level 5 file. Selecting hot-link 922 causes the browser to access file 934, which is a level 2 file. Selecting hot-link 920 causes the browser to access file 934, which is a level 2 file. Selecting hot-link 920 causes the browser to access file 938, which is a level 2 file.

The traversal steps from hot-link 924 through 926, 928, 930 and 932 constitute a drill-down from level 1 to level 5. Prior art browsers require that the user, in order to return to the original file 902, must traverse the search tree in reverse serial order. As shown serial return 904 traverses from 932 to 930, serial return 906 traverses from 930 to 928, serial return 908 traverses from 928 to 926, and finally serial return 910 traverses from 926 to 902, the initial starting point. This requires the needless steps of visiting each of the

ENPUBLICICC7/0108301.07

G 000108

branches traversed in the drill-down. This process is unnecessarily lime consuming when compared with the capabilities of the current jumper invention.

Fig. 9B shows the same tree structures as does Fig. 9A. Fig. 9B shows the use of one embodiment of the invention to traverse many levels at a time in either single jump or automatic jump mode. In the single jump mode when the searcher has reached level 5 file 932. If the user wants to return to level 1 they do not need to do so in reverse serial order, as is required by prior art browsers. Instead, the user directs the jumper to access the next level 1 hot-link. In response, the jumper returns the user in a single jump 914 from level 5 file 932 to the level 2 files pointed to by the hot-links in the level 1 file 902. Much time is saved and the user can proceed to search other level 1 hotlinks. In automatic jump mode, the user instructs the jumper to access all hotlinks on the level 1 file 902. In response, the jumper directs the browser to access and display in the browser window each of level 2 files 926, 934 and 936. These files are accessed by the browser in response to the jumper sending at timed intervals each of hot-links 924, 922, and 920 from a parsed hot-link list to the browser. The browser accesses each of the files 926, 934, and 938 to which hot-links 924, 922, and 920 respectively point. This allows the user to see an animation in the browser window for a timed interval.

#### Hardware:

08727085

Sesont.

10

15

20 .

25

30

In order to better understand the operation of a jumper, it is necessary to first understand an environment in which the jumper is practiced. Figs. 1-2, show the hardware environment in which the jumper operates. As shown in Fig. 1, an Internet 22 is connected to client 20 by an incoming Internet line 24 and an outgoing Internet line 26. Jumper 28 and browser 30 are part of client 20. Internet 22 comprises servers 32, 34, 36, and 38. Each server contains HTML files. The client and all servers are linked electronically. Server 34 is shown as the server of an information indexer. The server comprises a CGI 42 (common gateway interface), forms 44 on which a user query is entered,

HPUBLICICCT/0105301.07

15

20

25

30

# Case 2:04-cv-70366-JAC-RSW Docum

Document 54-5

Filed 09/27/2005

and storage containing an updated topical index 46 of all files on the web. To access a specific site on the network, browser 30 outputs an URL across outgoing internet line 26. Routers on the Internet establish an electronic link between the client 20 and the appropriate server 32-38. Communications with the Internet are received on incoming Internet line 24.

The client hardware architecture for both the browser and the jumper is shown in greater detail in Fig. 2. The client 20 contains storage 204, computational hardware 202, operating system 206, and GUI (Graphical User Interface) 200. The storage 204 contains a storage segment 230 in which the jumper stores parsed HTML files. Hardware 202 comprises a microprocessor, an I/O interface, a display and a bus connection to storage 204. The operating system 208 may comprise a bios and an operating environment such as. System 7<sup>TM</sup>, Unix<sup>TML</sup> or Windows<sup>TML</sup>. The GUI 200 comprises jumper 28 and browser 30. Browser 30 comprises user I/O 208 and presenter 210. The jumper 28 comprises user I/O 212, presenter 214, and parser 216.

The operation of hardware 202 is controlled by the operating system 208. The operating system and graphical user interface (GUI) 200, create the local environment in which the jumper 28 and browser 30 exist. The browser user I/O 208 handles the tasks of sending and retsieving communications 218 via the operating system 208 and hardware 202 to the Internet 22. The browser handles Internet communications for the jumper. The browser user I/O communicates with the jumper user I/O 212 via jumper-parser I/O protocol 224. The browser user I/O also handles the task of translating files received from the Internet, e.g., HTML encoded files, to the presenter 210 via presenter protocol 220. Additionally, browser user I/O uploads at the direction of jumper user I/O 212, HTML encoded files to parser 216 via browser I/O-jumper parser protocol 222.

Jumper parser 216 handles the task of conventing an HTML encoded file uploaded from browser user I/O 208 into a format suitable for a single jump

ENPUBLICACCT/0106301.07

G 000110

#### User interface:

10

15

- 20

25

30

8727085.100896

An embodiment of the jumper user interface is shown in Fig. 3. This interface allows the user to see all the hot-links on a given level and to select a hot link in either single jump or automatic jump mode as described above in connection with Fig. 9B. The jumper thus enhances the search capabilities of the browser.

Jumper window 300 comprises a menu bar 302, a button bar 304, an advertisement area 306, a jump site window 308, and a drop-down list button 310. The jumper menu bar contains file, edit, action, and help items. Jumper button bar contains a first entry button 312, a previous entry button 314, a random jump button 316, a next entry button 318, a last entry button 320, a start timer button 322, a stop timer button 324 and a refresh/update button 328.

The jump site window 308 has an associated drop down list. accessible by selecting drop-down-down list button 310. This list comprises parsed hot-links. These hot-links are extracted from a file initially retrieved by the browser. Any one of the hot-links in the parsed list can be selected by the user with a mouse. This drop-down list provides one method for selecting a specific internet site to jump to. Advertisement area 306 contains an advertisement parsed from an HTML page received by the browser. Button bar 304 provides controls for single jump and automatic jump mode. All

EVPUBLIC/CC7/0108301.07

# Case 2:04-cv-70366-JAC-RSW

# Document 54-5

Filed 09/27/2005

buttons relate to the hot-links in the jumper's parsed list regardless of how far the user has drilled-down in the browser. All buttons return the user to at least one of the hot-links on the list and direct the browser 30 shown in Figs. 1-2 to retrieve a file from the site corresponding to the URL associated with the hot-link and further direct the browser to display that file to the user.

First entry button 312 allows the user to return from a drill-down to the first entry on parsed list. Previous entry button 314 returns the user to the prior hot-link on the list. Random jump button 316 returns the user to a randomly selected hot-link on the parsed list. Next entry button 318 selects the next hot-link on the list, and last entry button 320 accesses the last entry on the list. Start timer button 322 initiates the automatic and successive selection of every hot-link on the parsed list and the browser responsive thereto, displays for the user a file retrieved from the URL corresponding to each hot-link. The animation initiated by start timer button is terminated by stop timer button 324. Refresh/update button 326 causes all hot-links in a file which the browser has retrieved to be parsed and uploaded and displayed in the drop-down list of jumper window 300. Additionally, this parsed list is stored in HTML storage segment 230 as shown in Fig. 2.

20

15

10

In alternate embodiments the jumper window may take any of several forms. The user interface may include popup or persistent window, a toolbar, a menu modification of the browser window, a toolbar modification of the browser window, or the use of accelerator keys on the keyboard.

25

Fig. 4 shows a prior art browser user interface and a query form of an information index provider. This interface allows the user to access web files and also displays the file contents to the user. The browser interface 400 is that of Netscape Navigator<sup>ns</sup>. The browser interface includes a browser menu bar 402, a site window 404, and a window 406 for viewing a file. The browser menu bar includes a back button 412 and a home button 414. The

HPUBLICICC710108301.07

The browser menu bar 402 generally provides for editing and saving files. Back button 412 allows the user to move back a search tree in reverse serial order as shown in Fig. 9A. Home button 414 allows the user to return to their home page 900 in the browser window. The browser site window 404 displays the URL corresponding to the file currently being displayed, in this case the query form 408 of Yahoo<sup>TM</sup>. The query form is encoded in a markup language and contains a query field 410 in which the user may enter the search topic for which a topical of the web is desired. In this case, the search topic is "Rat." When the search topic is entered, it is sent by the browser over the web to Server 34 owned by Yahoo<sup>TM</sup> 40 and shown in Fig. 1.

#### Conducting a search:

5

10

15

20

25

08727085

SEBOOT

Figs. 5A-C show the initial stages of a search session conducted with the jumper and browser. A browser interface 400, a file retrieved by the browser, and a jumper window 300 are shown. These are the basic tools for the demonstrative search. The initial file retrieved from Yahoo in response to the query "Rat" is shown as an upper page portion 502a and a lower page portion 502b in Figs 5A-B and 5C, respectively. These are accessed by scrollable window bar 500. The retrieved file contains an advertisement 504, a search result summary 506, a plurality of hot-links of which 508-514 and 580-584 are referenced. A representative hot-link 514, is shown to contain both an URL 516 and a text portion 518. In Fig. 5C a jumper drop down list 586 is shown. In addition, the retrieved file also contains a next feature 588.

In response to the entry of the query "Rat" as shown on Fig. 4, the query is sent by browser to Yahoo™. The topical index 46 at Yahoo™, shown in Fig. 1, is searched and the file 502a-b containing all related level 1 hot-links and other information is returned to the client 20 and displayed to the user by the browser as 502a-b. Page 502a is captioned with an advertisement 504.

1:\PUBLIC\CC7\0106301.07

G 000113

Page 42 of

15

20

25

30

Case 2:04-cv-70366-JAC-RSW Document 54-5

Immediately below the advertisement is a search result summary 508, which indicates 74 matches pertaining to the word "Rat" and display of the first 20 hot links for those matches. Various hot-links with subject descriptions appear on the page in bold text with underlining. These are, for example, 508, "Entertainment, Humor...", 510, "Regional Alternative Theater...", 512, "Space RAT...", 514, "Agrizap, Inc...".

Each hot-link as discussed above is an active reference. For example the hot-link 514 comprises a hypertext identifier "<A", an URL 516, a text portion 518 and a closing identifier. The browser displays to the user only text portion 518 and any images that may accompany it. These are highlighted and underlined on the user display to indicate to the user that when they are selected with a mouse a corresponding URL will be sent by the browser to the internet to access the specific file referred to by the URL.

As shown in Fig. 5B, jumper window 300 has been refreshed via refresh/update button 326, discussed above in connection with Fig. 3.

Responsive to selection of refresh/update button 326, the file 502a-b is passed from the browser 30 to the jumper parser 216 as shown in Fig. 2. All hot-finks including the referenced hot-links 508-514 and 580-584 are extracted and passed to jumper user I/O 212. These are passed to jumper presenter 214 and displayed in the jumper window 300. Specifically the advertisement 504 is displayed in jumper advertisement area 306, and all hot-links 508-514 and 580-584 are put in the jumper drop down list. The first of these hot-links 508 is displayed in the jumper site window 308.

In Fig. 5C, the lower portion 502b of the retrieved HTML page is shown. It contains hot-links 580, 582, 584. In jumper window 300 and specifically jumper site window 308 thereof, the jumper drop-down window, 586, is shown. This window has been accessed by the user's selection of jumper drop-down list button 310. The drop-down list contains all hot-links

EXPUBLICICCT/0108301.07

G 000114

Filed 09/27/2005

10

15

20

25

08727085

.10089

that were obtained from the original file 502a-b including the referenced hotlinks 508-514 and 580-584.

Now, by reference to Fig. 8, a later stage of the search session is shown. Fig. 6 shows a file 600 in browser view window 406, an URL corresponding to file 600 in browser site window 404, and a highlight 602 around hot-link 580 in both the jumper drop-down list window 586 and the jumper site window 308. The file 600 was obtained in a drill-down conducted in the browser window 406. The steps in that drill-down intervening between Fig. 5C and Fig. 6 have not been shown. The drill-down however resembles that shown and discussed above in connection with Fig. 9A. The file 600, shown in browser window 406 is four levels removed from the initial level 1 hot-link 580. .The drill-down was conducted on the browser view screen by selecting the hot-links presented in the files retrieved by the browser. If the user desires to return to the an other hot-link from the level 1 file, shown as 502a-b in Figs. 5A-C, from this file 600, they could use the browser menu bar to do so but it would be slow. They would have to select back button 412 four times to return them one level at a time in reverse serial order through their entire search back to the level 1 file. The process is time consuming. Alternately, if they selected the browser home key 414 they would return to their home page and lose the level 1 search results. Neither of these alternatives is acceptable. Both of these alternatives are cumbersome.

If the jumper product is utilized instead, the user has several options. As a first option, the user may select one of jumper buttons 312, 314, 316, 318 or 320 to return to respectively: a first, a previous, a random, a next, or a last of the hot-links shown in Fig. 6 in jumper drop-down list window 586. In response to user selection, the file corresponding to the selected hot-link will be retrieved and displayed in the browser window 406. As a second option, the user may select start timer button 322 to initiate the automatic and successive selection at a timed interval of every hot-link on the parsed list. In response to the automatic jump mode selection, each file corresponding to the

1:\PUBLIC\CC7\0106301.07

# Case 2:04-cv-70366-JAC-RSW Document 54-5

selected hot-link will be retrieved and displayed by the browser for the timed interval. As a third option, the user may select drop-down window button 310 and select a given entry from the drop down list by clicking on it with a mouse. In response to their selection the file corresponding to the selected hot-link will be retrieved and displayed in the browser window 406. Alternately, another instance of browser can be opened to display the selected file. All of these choices enhance the search, and none require multiple reverse serial order steps to return to the level 1 hot-links.

Process Flow:

10

15

20

25

30

Fig. 7 shows the process connected with conducting a search utilizing a prior art browser. Commencing with start process 700, a connection process 702 results in Yahoo's™ bome page 408 and query form being displayed in prowser window 406, as shown in Fig. 4. Then in query process 704, the user enters a search topic, e.g., "Rat", in information indexer query field 410. In response to the entry of that search topic, the browser sends the query to server 34 of the search index provider, e.g., Yahoo™, the relevant topical identifiers and URLs are uploaded on an HTML encoded file/page and presented to the user in page receipt process 706. Subsequently, when a user utilizing a mouse selects one of the hot-links shown on that page, e.g., 502a-b, then that selection is detected in site access decision 708. The URL corresponding to the last selected hot-link is output over the internet to access the file indicated by that URL and that file is uploaded and displayed in access process 710 to the user. Subsequently, in history creation process 712, that URL is added to a stack comprising in LIFO order the most recently accessed files, including the complete URL for those files. Control is then returned to back-one decision 7.14 which decision is also reached if a negative determination is made in site access decision 708. In back-one decision 714, a determination is made as to whether the user has selected back button 412 on the browser menu 402. If that selection has been made, then control is passed to repeat history process 716 in which the previous site visited is popped from the history stack and in access process 718 the browser

INPUBLICICOTO 106301.07

G 000116

Filed 09/27/2005

accesses the site and file indicated by the prior URL and presents that in its view window to the user. Control is subsequently returned to site access decision 708.

Alternately, if in back-one decision 714 a negative determination is reached the browser awaits the next user command 720.

In Figs. 8A-C the process connected with one embodiment of the current invention is shown. It contrasts with the prior art process, in that the user may return to the level of files pointed to by any level one hot-link immediately. The cumbersome process and many steps of reverse serial order traversal discussed above in Fig. 7, are not therefore necessary. Fig. 8A is an overall view of the process connected with either the single jump or automatic jump mode. Fig. 8B details the specific steps connected with the automatic jump mode. Fig. 8C details the specific steps connected with the single jump mode.

Commencing with Fig. 8A and start jumper process 800 control is passed to refresh decision 802. In the event, a determination is made that refresh/update button 326 shown in Fig. 3 has been selected, then control is passed to fetch and parse process 804 in which an HTML encoded page displayed in the browser view window is uploaded and parsed into an advertisement and hot-links. These are displayed in the jumper advertisement area 306 and the jumper drop-down window 586, respectively. Then in history decision 806, the determination is made as to whether the jumper history maintain option has been selected. This option is found under the edit portion of the menu bar 302. In the event that this option has been selected, then control is passed to append and store process 808 in which the hot-links extracted in fetch and parse process 804 are appended and stored with previous hot-links in the parsed HTML files in storage segment 230 shown on Fig. 2. Control is then passed to display URL process 812. Alternately, if a negative determination is made in history decision 806, then control is passed

HPUBLIC/CC7/0106301.07

18

8727085

SEBUOL

10

15

20

25

15

20

25

30

# Case 2:04-cv-70366-JAC-RSW Document 54-5

to store process 810 in which the hot-links from the fetch and parse process 804 replace any existing stored hot-links. Subsequent control also is passed to display URL process 812. In display URL process 812, the hot-links are displayed in the jumper drop-down window 586 as shown in Fig. 6. Control subsequently passes to display advertisement process 814 in which an image corresponding to an advertisement parsed from the browser page is placed in the jumper advertisement area 306.

As discussed above in connection with refresh decision 802, if a negative determination is made that the refresh button has not been pressed, then control passes also to display advertisement process 814 in which the advertisement from the page currently displayed on browser window 406 is uploaded and displayed in the jumper advertisement area 308. Control subsequently passes to start timer decision 816. If the user has selected start timer button 322 from the jumper menu bar, then automatic jump mode commences in a manner set forth in Fig. 8B. Alternately, if a negative determination is reached, then control is passed to jumper button select decision 818. A determination is made as to whether the user has selected any one of buttons 310-320 in the jumper window 300. In this event, control is passed to processes set forth on Fig. 8C and corresponding to single jump mode. The return from either the automatic jump or single jump process is via splice block B to jumper button select decision 818. Finally, when a negative determination is reached in jumper button select decision, control returns to refresh decision 802. This completes the overall processing connected with the jumpers tree traversal.

In Fig. 8B, the automatic jump mode of the search process is shown in greater detail. This process allows a user to view an animated tour of the level 2 files pointed to by the level 1 hot links in the browser window.

Commencing with splice block A, control passes to memory fetch process 840 in which the first URL, of the hot-links in storage segment 230 shown in Fig. 2, is fetched by the jumper. In the next process, jumper-browser process 842 the

InPUBLICIÓC7/0108301.07

Filed 09/27/2005

Case 2:04-cv-70366-JAC-RSW

Filed 09/27/2005 Page 48 of

2:04-cv-70366-JAC-RSW Document 54-5 first URL is sent to the browser, user I/O 208 caused the browser to access the specific site and file on the site, to upload the file and to display the file in the browser window 408. In one embodiment, if the browser is still loading a previous file when the jumper-browser process 842 is reached, then sending the browser the URL interrupts the loading of the previous file: In another embodiment, the jumper-browser process 842 does not interrupt the loading of the previous file. In a third embodiment, whether the jumper-browser process 842 interrupts the loading of the previous file is a user configurable option.

10

Control is then passed to delay process 844 in which a user selectable display interval causes the page image to be displayed on browser window 408 for the selected time interval. At the end of that time interval, control is passed to memory fetch process 846 in which the next URL in the parsed HTML file is uploaded and in last of list decision 848 a determination is made as to whether that URL is the last among the stored URLs. In the event that determination is in the negative, control returns to jumper-browser process 842 for the timed display of the next accessed URL. This loop from jumperbrowser process 842 through last of list decision 848 constitutes a user configurable animation, in which a set of URLs which have been parsed and stored are now used to drive an automatic retrieval and display process.

20

25

30

Alternately, in last URL decision 848, if a determination is made in the affirmative, control is passed to "next" link decision, 850. In next link decision 850 a determination is made as to whether the last URL also includes a query for the next 10 entries corresponding to the example shown in Fig. 5C and indicated by 588. In the event that determination is in the affirmative, control is passed to jumper-browser-jumper process 852. In process 852 the query for the next 10 hot-links is sent to Yahoo™. In response the next HTML encoded page containing the 10 hot-links is retrieved. Control is passed to parse process 854 in which all the URLs and corresponding text identifiers are parsed along with the image corresponding to the advertisement. Then in buffer process 856, the parser output is stored for this session. Control then

1:\PUBLIC\CC7\0108301.07

15

#### Case 2:04-cv-70366-JAC-BSW Document 54-5

Filed 09/27/2005

passes to memory fetch process 849 for automatic retrieval and display. Alternately, if in "next" link decision 850, a negative determination is made, then in jumper-browser process 858, the last URL is sent to the browser to both access and display the file corresponding to the last URL. Subsequently, control is passed to splice block B for a return to hot-link select decision 818.

The process corresponding to the single jump mode is shown in Fig. 8C. This process allows the user to select a specific level 1 hot-link and have the browser retrieve and display the level 2 file corresponding to the selected level 1 hot-link. In window process 880, the jumper drop-down window 586 is automatically opened, as is shown in Fig. 5C. Then in dropdown button process 882 a determination is made at to whether the button selected by the user and detected in jumper button select decision 818, was the drop-down list button 310. If the user did select button 310 then the user has not yet selected a specific hot-link from the parsed hot-links in the jumper drop-down window 588. Control is then passed to wait process 884. When the user selects a specific entry from the drop-down list control passes to locate select process 888. Alternately, if a negative determination is reached in jumper button select decision 818 then control is passed directly to locate select process 886. A negative determination in jumper select decision 818 means that one of jumper buttons 312, 314, 316, 318 or 320, was selected thereby indicating a specific user choice for the single jump location. These buttons were discussed above in connection with Fig. 3.

In locate select process 888 a determination is made as to which of the hot-links in the jumper drop-down window was selected. If, for example, first entry button 312 were selected then the first of the hot-links in the list is selected. Alternately, if the user selected drop-down window button 310 and then mouse selected an entry e.g. the fourth entry on the drop-down list, then that entry would be highlighted. In the example shown in Fig. 6 the user has selected hot-link 580 and that selection, the "Ricky Rock Rat" hot-link is detected by the jumper.

MPUBLICIC::7/0106301.07

G 000120

08727085.100896

25

30

20

Page 49 o

•

10

15

20

25

08727085

3680DT

Subsequent to selection detection, control is passed to jumperbrowser process 888. The URL corresponding to the selected hot-link is passed by the jumper to the browser causing the browser to locate and subsequently display a file corresponding to that URL. The jumper continues to display all level "1" sites as hot-links. In the next step highlight process 890 the selected hot-link is highlighted, as shown for example in Fig. 6 as highlight 802 around selected hot-link 580, "Ricky Rock Rat". After the highlight process, the selected hot-link on the drop-down list is copied to the jumper site window 308 in copy process 892. Subsequently, in process 894 the jumper drop-down window is closed.

When the browser displays the above mentioned file, the user may conduct a drill-down on the hot-links in that file in the browser window. In that case, the browser window is active and the jumper window is inactive. When the drill-down is complete and the user reactivates the jumper window by mouse clicking on it, that reactivation is detected in process 898 and control is passed to jumper select decision 818 via the splice block B, shown in Fig. 8A.

The following describes additional embodiments of the invention.

Except where stated in alternative form, each of these embodiments include features that can be combined with the features discussed above.

Some alternative embodiments provide better integration of the jumper's functions and the browser's functions. In one embodiment of the invention, when a user initiates a search in the browser, the jumper automatically starts and begins parsing the results of the search. This saves the user from having to start the jumper separately from the browser. In another embodiment, the jumper functions are built directly into the browser. In a different embodiment, the jumper is implemented as an application, such as an appliet, which is sent to the browser by the search engine. All of these

E-PUBLIC/CC7/0108301.07

22