

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
5 October 2006 (05.10.2006)

PCT

(10) International Publication Number  
**WO 2006/104657 A1**

(51) International Patent Classification:  
**G06F 17/30** (2006.01)

(21) International Application Number:  
PCT/US2006/008315

(22) International Filing Date: 9 March 2006 (09.03.2006)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:  
60/664,816 24 March 2005 (24.03.2005) US

(71) Applicant (for all designated States except US):  
**HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.** [US/US]; 20555 S.h. 249, Houston, Texas 77070 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **KINNAN, Keith, Robert** [US/US]; Hewlett-Packard Company, 20555 Sh 249, Houston, TX 77070 (US). **NATRAJAN, Balaji** [US/US]; Hewlett-Packard Company, 20555 Sh 249, Houston, TX 77070 (US).

(74) Agent: **LAWS, Gerald E.**; Hewlett-Packard Company, Intellectual Property Administration, P.o. 272400 ; Mail Stop 35, Ft. Collins, Texas 80527-2400, (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

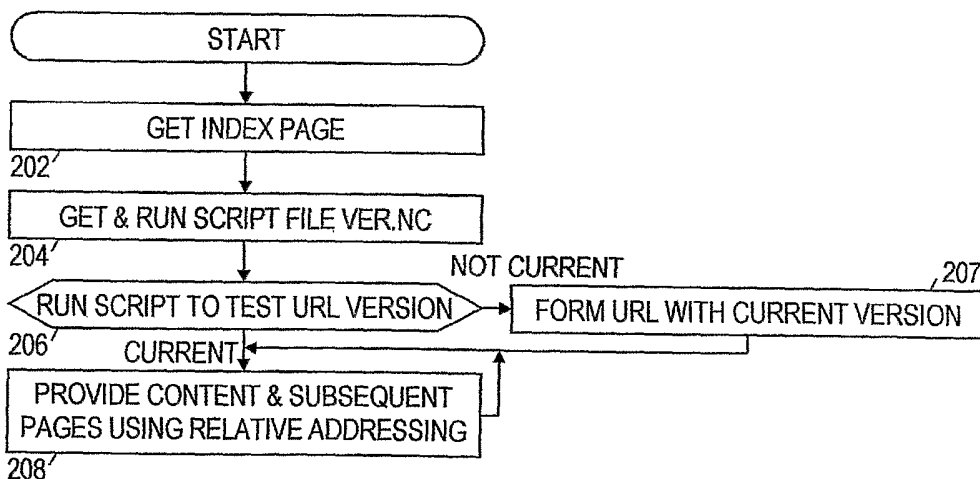
(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Declarations under Rule 4.17:**

- as to the identity of the inventor (Rule 4.17(i))
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))
- as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))

[Continued on next page]

(54) Title: EMBEDDED WEB-BASED MANAGEMENT METHOD



(57) Abstract: Methods and systems for embedded web-based management are disclosed. In some embodiments, an index web page (302) is provided (202) with a link (304) to an uncacheable version control file (402), and with a script (308) to cause redirection to a uniform resource locator (URL) having a version indicator determined from the version control file. The version indicator allows web caching of current files while ensuring that the cache be updated when the software on the target device is updated. The target device (114) may have an embedded web server (124) configured to enforce caching policies through expiration dates in HTTP response headers. The embedded web server may also be configured to remove version information from the URLs in requests before servicing the requests.

WO 2006/104657 A1



**Published:**

— *with international search report*

*For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

## EMBEDDED WEB-BASED MANAGEMENT METHOD

### CROSS-REFERENCE TO A RELATED APPLICATION

**[0001]** The present application claims the benefit of, and incorporates by reference, provisional application Serial No. 60/664,816, filed March 24, 2005, and entitled "Method Of Improving Performance Of A Web-Browser-Based, Target Based Management GUI With Support For Version Upgrades."

### BACKGROUND

**[0002]** The Internet confers many benefits due to its connectivity. Users are able to access public content from any web-enabled computer. People throughout the world are able to interact one-on-one and in groups. Businesses are able to market their goods for minimal costs. Employees are able to tap into their employers' networks to work outside the office. These benefits are just some examples of why the Internet has been almost universally embraced.

**[0003]** As the Internet continues to expand, new applications are being discovered. Storage area networks are being equipped with switches and storage controllers that have embedded web-based management utilities. Such devices ("target" devices) are accessible using Internet Protocol (IP) communications, allowing a user to configure and review the operation of the target devices via a conventional web browser. Proposals have even been made to connect home control systems and even home appliances to the Internet using such embedded web-based management utilities.

**[0004]** Constraints for such devices are expected to include memory and processor power. Processors built in to target devices are expected to be inexpensive (i.e., to have limited processing power) and to have most of their

processing power devoted to coordinating device operation. Memory is also expected to be limited. Visually complex web-based interfaces generally manifest themselves as large files and/or a large number of files that need to be transferred by the utility to the user's web browser. Due to the limited resources available to the utility, such transfers may take an unreasonable amount of time to complete, testing the user's patience if many such transfers are needed.

**[0005]** Conventional web browsers offer caching of web pages, so that repeated downloads of a given file may be avoided. When caching is enabled, a web browser downloads a given file once and stores it in a cache, giving the cached file an expiration date. Subsequent accesses to the file, so long as they occur before the expiration date, are satisfied by retrieving the file from cache rather than downloading the file again.

**[0006]** However, caching creates a potential problem when, e.g., a web-based management utility is updated. Depending on the caching policy employed, the utility may continue to use out-of-date files, rendering the update ineffective, or may fail to operate properly because the browser employs a mix of old and new files. Various solutions to this dilemma exist that involve a tradeoff in performance vs. update risk.

#### BRIEF DESCRIPTION OF THE DRAWINGS

**[0007]** For a detailed description of illustrative invention embodiments, reference will now be made to the accompanying drawings in which:

**[0008]** Figure 1 shows a user device coupled to a target device via a network that supports internet protocol (IP) communications;

**[0009]** Figure 2 shows a flowchart of an illustrative embedded web-based management method;

**[0010]** Figure 3 shows an illustrative web page to implement the method of Fig. 2;

**[0011]** Figure 4 shows an illustrative version control file; and

**[0012]** Figure 5 shows an illustrative redirection script.

#### NOTATION AND NOMENCLATURE

**[0013]** Certain terms are used throughout the following description and claims to refer to particular system components. As one skilled in the art will appreciate, companies may refer to a component by different names. This document does not intend to distinguish between components that differ in name but not function. In the following discussion and in the claims, the terms “including” and “comprising” are used in an open-ended fashion, and thus should be interpreted to mean “including, but not limited to...” Also, the term “couple” or “couples” is intended to mean either an indirect or direct electrical connection. Thus, if a first device couples to a second device, that connection may be through a direct electrical connection, or through an indirect electrical connection via other devices and connections.

#### DETAILED DESCRIPTION

**[0014]** The following discussion is directed to various invention embodiments. Although one or more of these embodiments may be preferred, the embodiments disclosed should not be interpreted, or otherwise used, as limiting the scope of the disclosure, including the claims. In addition, one skilled in the art will understand that the following description has broad application, and the discussion of any embodiment is meant only to be illustrative of that embodiment, and not intended to suggest that the scope of the disclosure, including the claims, is limited to that embodiment.

**[0015]** Fig. 1 shows an illustrative network environment in which a network 102 interconnects a computer 104, a network storage array 106, network-enabled consumer electronics 108, 110, a network-enabled thermostat 112, a target device 114 shown in block-diagram form, and a user device 126 shown in block-diagram form. Computer 104 and device 126 are illustrative of user devices, i.e., devices having a web browser that provides a graphical user interface (GUI) that a user may employ to access files on other network-enabled devices. Other examples of user devices include personal digital assistants (PDAs), mobile phones, and video game consoles.

**[0016]** Devices 106-114 are illustrative of target devices, i.e., devices that include an embedded web server that allows user devices to access files or

other locally available data. Target devices may take the form of network switches, network storage devices, network-enabled appliances, etc. Target device 114 includes a network interface 116, a processor 118, memory 120, and a special purpose module 122. Network interface 116 couples to network 102. Processor 118 executes web server software 124 in memory 120 to receive hypertext transfer protocol (HTTP) requests via interface 116 and to send HTTP responses via interface 116. The processor 118 may be further configured to control the operation of special purpose module 122 which carries out the particular design function of the target device, e.g., information storage, printing, audio-video playback, temperature control, etc.

**[0017]** User device 126 includes a network interface 128, a processor 130, a memory 132, a video interface 134, and an I/O interface 136. Network interface 128 couples to the network 102. Processor 130 executes web browser software in memory 132 to send HTTP requests and receive HTTP responses via interface 128. The web browser software configures processor 130 to generate a visual display via video interface 134 and to operate on user input received via I/O interface 136. The web browser software provides a standardized platform that is employed by the web server software 124 to create a GUI for managing target device 114. Via the GUI the user may, for example, view and change configuration parameters that affect the target device's operation and/or the user may simply monitor the device's performance.

**[0018]** Depending on the design budget and available hardware resources, an embedded web server's responses to requests may be slow. A web browser GUI may require many requests to the embedded web server, making the overall interface appear slow. For example, an embedded web-based management utility for storage system configuration and management may use several hyper-text mark-up language (HTML) files, JavaScript files, and image files to create an interactive and visually dynamic GUI that runs in a web browser on a user device. The GUI may employ a graphical tree that can have thousands of storage devices, with a user-friendly image for each device. Each image resides in a separate image file that must be retrieved using an HTTP

request to the web server. The many HTTP requests slow the browser's operation, burden the target device with additional processing resource demands, and make the embedded web server a bottleneck, unless caching is enabled. The methods disclosed herein leverage the caching abilities of the web browser to improve performance while ensuring that upgrades to the target device are properly run without any requirement for manual user intervention.

**[0019]** The user may be able to greatly speed up the GUI by turning on caching in the browser and preventing the browser for checking for newer file versions on each request. However, such actions mean that if the target device's firmware or software is upgraded, the user will be using obsolete file versions in the browser cache. In this situation, the device software upgrade is carried out in an inconsistent manner. If the user does not clear the browser cache after upgrading, the GUI application may not run properly and may even crash because potentially old and out of date application files are being used.

**[0020]** Fig. 2 shows a flowchart of an illustrative embedded web-based management method that leverages the performance gains offered by browser caching without the foregoing disadvantages. A user types in a uniform resource locator (URL) or network address for the target device in an address field of a web browser. In block 202 the web browser requests an index file from the target device. The index file may take the form of a cacheable hypertext markup language (HTML) page. In responding to this request and other requests for files deemed to be cacheable, the embedded web server may provide a response header with a future expiration date, e.g., 3 months, 6 months, or a year from the present, or "infinity." The web browser will examine the expiration in making a decision as to whether or not to cache the file.

**[0021]** The index page includes a link to a version control file. In some embodiments, the version control file link may be embedded in a script (i.e., a set of instructions for execution by the browser) so that the version control file is received and its contents are treated as executable instructions. Thus, in block 204, the browser requests the version control file from the target device and executes the instructions contained in the file when the file is received. The

instructions may store version information in one or more variables for subsequent use. The version control file is meant to be non-cacheable. In responding to this request and other requests for files deemed to be non-cacheable, the embedded web server may provide a response header with a current or past expiration date. The web browser should decline to cache file from such "expired" responses.

**[0022]** The index page further includes a script that, in block 206, causes the browser to test that the URL of the index page contains a version indicator that matches information in the version control file. If the version indicator is missing or not current, in block 207 the script causes the browser to issue a request with a URL that has a current version indicator, thereby "redirecting" the browser to the file specified by the new URL. (As will be explained in further detail below, the file specified by the new URL may be the same image page.) If the version indicator is correct, in block 208 the script causes the browser to display the content of the index page. The index page may employ relative links that preserve the version indicator in any URLs used to request the pages or files accessible via the index page.

**[0023]** The embedded web server may be configured to ignore the version indicator. That is, in responding to a request having a URL with a version indicator, the embedded web server may remove the version indicator from the URL to determine a version-free URL. The embedded web server may then provide in response to the request a file identified by the version-free URL. In embodiments employing this option, the embedded web server possesses the version information only in the version control file. The scripts do all of the work associated with version control to make sure the user does not inadvertently use obsolete files from the browser cache.

**[0024]** The caching policy (i.e., the determination of which files are not cacheable and therefore are to be provided with HTTP response headers indicating an imminent or past expiration date) may be based on a file naming convention. For example, files such as the version control file that are deemed non-cacheable may be given a file name with a ".nc" extension (e.g.,

"version.nc"). All other files may be deemed cacheable by default, and given a future expiration date, e.g., 1 year in the future. To maximize performance, files would be cacheable by default, with only a minimum number of small files for all dynamic content (i.e., device status and device data), which would be accessible as special files with the ".nc" extension, e.g., "data.nc." When the target device's software or firmware is upgraded, the version control file is updated to hold a new version number. The browser will request the version control file and use an updated version indicator as part of the current and future URL requests. The web browser will look in its cache for these requests and will not find any of these URLs, forcing the browser to send requests to the embedded web server. As the requests are satisfied, the updated (cacheable) files will be saved in the browser cache. Consequently, the GUI application will be upgraded properly and no old cached files will be used.

**[0025]** Fig. 3 shows an illustrative entry level page 302. When a user specifies a device address with no specific file, the targeted web server provides a default page such as "index.htm", so index.htm (the "index page") is a good example of an entry level page. Entry level page 302 is shown as HTML code that may appear in a file named "index.htm." In accordance with the HTML standard, HTML tags are used to delineate components of the page. Each page component appears between a pair of tags, e.g., <HTML> and </HTML>; <HEAD> and </HEAD>; <SCRIPT> and </SCRIPT>. Every HTML page may consist of a head portion and a body portion. The head portion commonly includes metadata about the page, while the body commonly includes the content. In the present example, the head portion includes a title element, identifying a title for the web page.

**[0026]** In the body of the illustrative page 302, three scripts appear. A first script 304 includes a link to a version control file "ver.nc," which causes the version control file to be retrieved and executed by the browser. Fig. 4 shows an illustrative version control file 402 containing three script commands to store version information in three variables, MAJOR\_VERSION, MINOR\_VERSION, and BUILD\_NUMBER. After the script file has been executed, the version

information is available in these three variables for use by subsequent script commands.

**[0027]** With reference to Fig. 3, a second script 306 includes a link to a file “app\_ver.js” containing a class with functions to enforce the use of a version indicator in the URL. Fig. 5 shows illustrative file 502 containing script commands that create a “CVersion” class (lines 5-41). The class includes a redirection function “RedirectToVersionURL” (lines 6-39) that determines the URL of the current page (line 8), tests the URL to determine if it includes a version indicator (lines 11-13), determines whether the version indicator is correct (lines 17-18), constructs a URL with the correct version indicator if none existed (lines 28-35) or it was incorrect (lines 19-24), and in either case redirects the browser to a URL with the correct version indicator (line 37). A supplemental function “versionUrlEncoding” (line 40) has been omitted, but it simply converts the version information into a fixed-length version indicator.

**[0028]** With reference again to Fig. 3, the second script 306 defines a class. The third script 308 creates a class object “version” in statement 310. In statement 312, the third script 308 invokes the redirection function described previously. If the redirection function returns an indication that redirection is unneeded (i.e., an indication that the current page is not obsolete), the script 308 may display the page content in the browser. Alternatively, the page content may be placed after the scripts. In any event, the page content may include links 312 to other files on the target device. Such links 312 are provided in relative form “config.htm” rather than absolute form “http://1.2.3.4:80/config.htm” to ensure that only up-to-date versions of cached files are accessible to the user.

**[0029]** The above discussion is meant to be illustrative of the principles and various embodiments of the present invention. Numerous variations and modifications will become apparent to those skilled in the art once the above disclosure is fully appreciated. It is intended that the following claims be interpreted to embrace all such variations and modifications.

**FOREIGN CLAIMS**

1. A web-based management method comprising:  
providing a web page (302) that includes:  
a link (304) to a version control file (402); and  
a script (306) to cause redirection to a uniform resource locator (URL) having a version indicator determined from the version control file if the URL of the web page does not include a version indicator that matches version information in the version control file; and  
providing the version control file with a response header that prevents caching of the version control file.
2. The method of claim 1, further comprising:  
receiving a request with a URL having a version indicator;  
determining a version-free URL having said version indicator deleted;  
and  
providing a file identified by the version-free URL, wherein the file identified by the version-free URL is said web page.
3. The method of claim 1, wherein said web page includes links using relative addressing to incorporate the version indicator in the URL of corresponding requests to an embedded web server.
4. The method of claim 1, wherein said web page is provided with a response header that allows caching of the web page.
5. The method of claim 1, wherein the version control file has a name to indicate that the version control file should not be cached.
6. The method of claim 1, further comprising:  
receiving a request with a URL having a caching indicator;

providing a file identified by the URL with a response header indicating an expiration date in the past if the caching indicator indicates that the file should not be cached; and  
providing the file identified by the URL with a response header indicating an expiration date in the future if the caching indicator indicates that the file should be cached.

7. A device that comprises:
  - a network interface (116) configured to couple the device to a network having a computer with a web browser;
  - a memory (120) configured to store web server software (124); and
  - a processor (118) coupled to the memory to execute the web server software, wherein the web server software configures the processor to:
    - receive a request for an entry level web page via the network interface; and
    - respond to said request with a cacheable web page that includes:
      - a link to an uncacheable version control file;
      - a script that redirects to a URL having a version identifier determined from the version control file if the URL of the web page does not include a version identifier that matches information in the version control file.
8. The device of claim 7, wherein the web server software further configures the processor to:
  - determine a version-free URL for each request having a URL with a version identifier; and
  - respond to each said request with a file identified by the version-free URL.

9. The device of claim 7, wherein the web server software further configures the processor to:

determine a cacheability status for each file request received via the network interface;

respond to each said request having a cacheable status with a response having a response header indicating an expiration date at least six months in the future; and

respond to each said request having a non-cacheable status with a response having a response header indicating an expiration date in the past.

10. The device of claim 7, wherein the redirection URL causes the web browser to request and cache the entry level web page under a new URL, wherein the entry level web page uses relative links, and wherein the web browser caches cacheable files accessed via the entry level web page under corresponding URL addresses that include a current version identifier.

1 / 3

FIG. 1

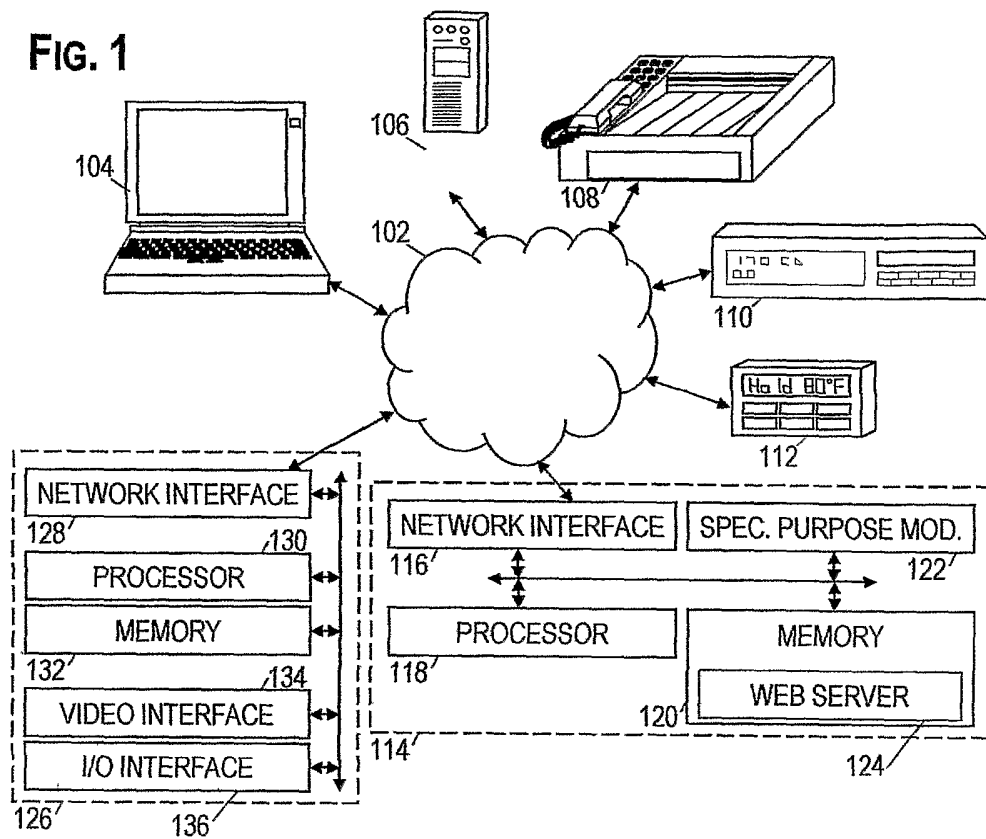
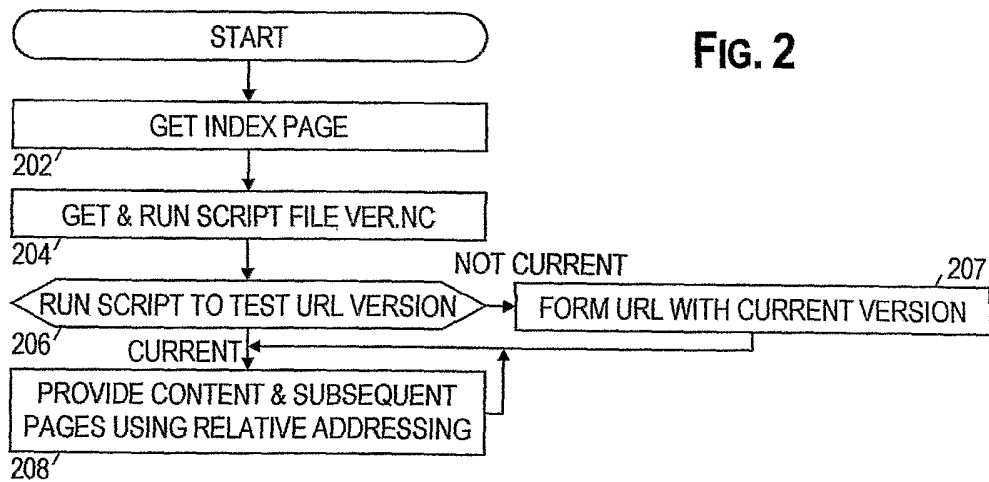
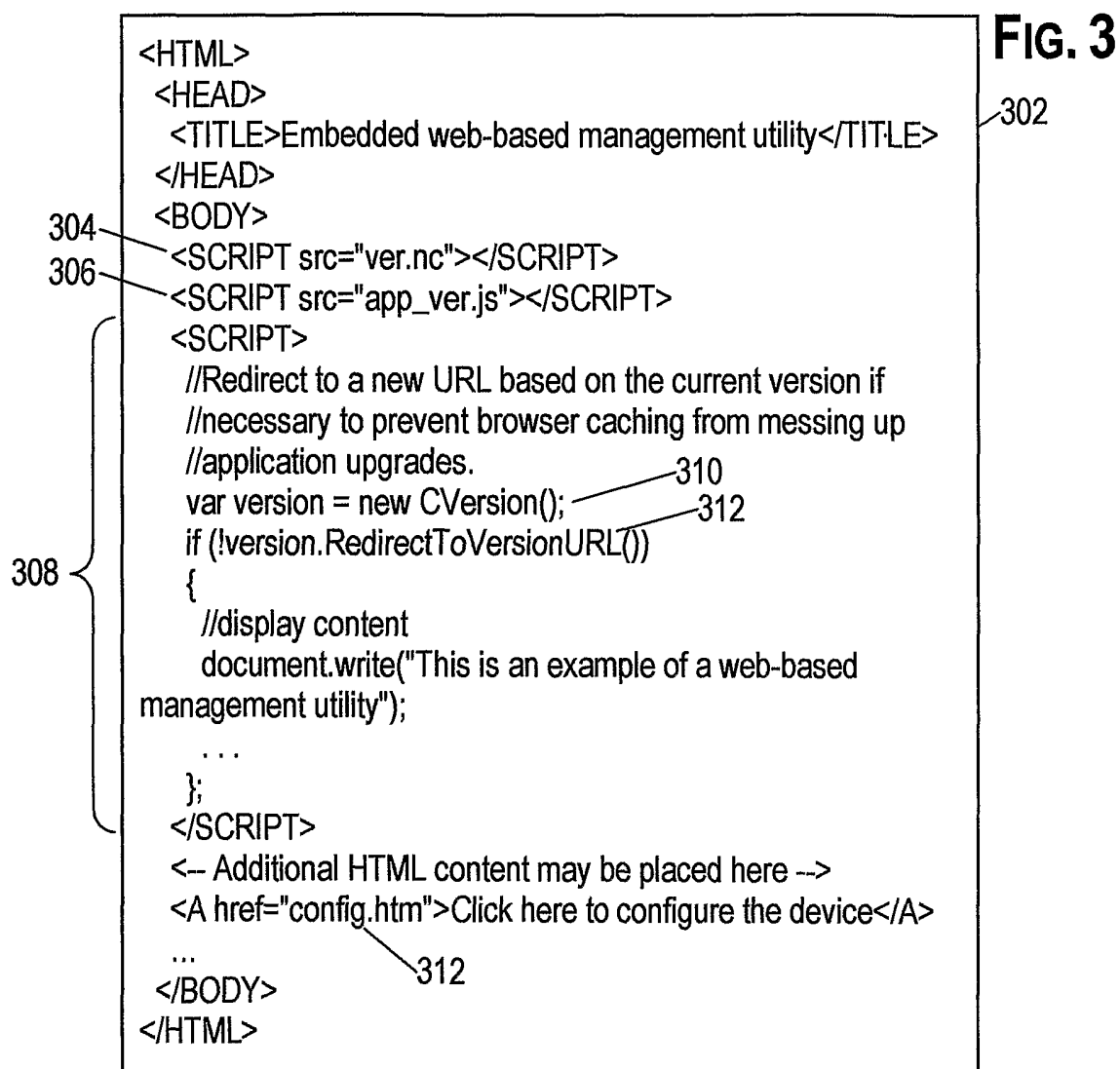


FIG. 2





**FIG. 4**

```
MAJOR_VERSION=2;
MINOR_VERSION=1;
BUILD_NUMBER=15;
```

402

502

```

1 //CVersion - Class to get the application version
2 //public method:
3 // RedirectToVersionURL() - redirects application to a URL based on current version
4 // if already not using a current version URL. Returns true if the redirect will occur.
5 function CVersion() {
6   this.RedirectToVersionURL = function() {
7     var isRedirect = false;
8     var urlString = "" + document.location;
9     //Test to see if there is a version marker in the URL
10    var newUrl = "";
11    var versionRegExp = new RegExp("/v[0-9][0-9][0-9][0-9][0-9][0-9][0-9]/");
12    var ar = versionRegExp.exec(urlString);
13    if (ar != null) {
14      //There is a version marker in the URL. Test to see if the version marker
15      //matches current version. If it does, do nothing - the URL is fine. If it does
16      //not match, then replace it with the current version
17      var thisVersionRegExp = new RegExp("/" + versionUrlEncoding() + "/");
18      if (!thisVersionRegExp.test(urlString)) {
19        //The new URL will have the /#####/ in it with the current version
20        var urlStart = urlString.substr(0, ar.index + 1);
21        var urlEnd = urlString.substr(ar.index + 10);
22        if (urlEnd.charAt(0) != "/") { urlEnd = "/" + urlEnd; }
23        newUrl = urlStart + versionUrlEncoding() + urlEnd;
24        isRedirect=true;
25      }
26    }
27    else {
28      //There is no version marker in the URL. Put one in there.
29      //The new URL will have the /#####/ in it with the current version
30      var lastSlashIndex = urlString.lastIndexOf("/");
31      var urlStart = urlString.substr(0, lastSlashIndex + 1);
32      var urlEnd = urlString.substr(lastSlashIndex);
33      if (urlEnd.charAt(0) != "/") { urlEnd = "/" + urlEnd; }
34      newUrl = urlStart + versionUrlEncoding() + urlEnd;
35      isRedirect = true;
36    }
37    if (isRedirect) { document.location = newUrl; }
38    return isRedirect;
39  }
40  function versionUrlEncoding() { ... }
41 }

```

FIG. 5

# INTERNATIONAL SEARCH REPORT

International application No  
PCT/US2006/008315

**A. CLASSIFICATION OF SUBJECT MATTER**  
INV. G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)  
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data, INSPEC

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category* | Citation of document, with indication, where appropriate, of the relevant passages                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Relevant to claim No. |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| A         | <p>SIMONSON J ET AL: "Version augmented URIs for reference permanence via an Apache module design"</p> <p>COMPUTER NETWORKS AND ISDN SYSTEMS, NORTH HOLLAND PUBLISHING. AMSTERDAM, NL, vol. 30, no. 1-7, April 1998 (1998-04), pages 337-345, XP004121413</p> <p>ISSN: 0169-7552</p> <p>abstract</p> <p>page 338, right-hand column, line 30 -</p> <p>page 339, right-hand column, line 16</p> <p>page 339, right-hand column, line 31 -</p> <p>page 340, right-hand column, line 24</p> <p>page 343, left-hand column, line 6 - page 343, right-hand column, line 15</p> <p style="text-align: center;">-----</p> <p style="text-align: center;">-/--</p> | 1-10                  |

☒ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

\* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

14 June 2006

Date of mailing of the international search report

22/06/2006

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
Fax: (+31-70) 340-3016

Authorized officer

Boyadzhiev, Y

## INTERNATIONAL SEARCH REPORT

International application No

PCT/US2006/008315

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Relevant to claim No. |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| A         | <p>TAY TENG TIOW ET AL: "Distributed web caching with incremental update"<br/> COMMUNICATION SYSTEMS, 2002. ICCS 2002.<br/> THE 8TH INTERNATIONAL CONFERENCE ON NOV.<br/> 25-28, 2002, PISCATAWAY, NJ, USA, IEEE,<br/> vol. 2, 25 November 2002 (2002-11-25),<br/> pages 1147-1151, XP010629400<br/> ISBN: 0-7803-7510-6<br/> abstract<br/> page 1147, right-hand column, line 16 -<br/> page 1147, right-hand column, line 45<br/> page 1148, right-hand column, line 41 -<br/> page 1149, right-hand column, line 26</p>             | 1-10                  |
| A         | <p>US 2002/198956 A1 (DUNSHEA ANDREW ET AL)<br/> 26 December 2002 (2002-12-26)<br/> abstract<br/> paragraph [0018]<br/> paragraph [0036] - paragraph [0037]<br/> paragraph [0055]<br/> paragraph [0061]</p>                                                                                                                                                                                                                                                                                                                            | 1-10                  |
| A         | <p>ZANDER S ET AL: "Dynamics and cachability<br/> of web sites: implications for inverted<br/> capacity networks"<br/> NETWORKS, 2003. ICON2003. THE 11TH IEEE<br/> INTERNATIONAL CONFERENCE ON SEPT. 28-OCT.<br/> 1, 2003, PISCATAWAY, NJ, USA, IEEE,<br/> 28 September 2003 (2003-09-28), pages<br/> 45-50, XP010683500<br/> ISBN: 0-7803-7788-5<br/> abstract<br/> page 47, left-hand column, line 6 - page<br/> 47, left-hand column, line 41<br/> page 48, left-hand column, line 4 - page<br/> 48, left-hand column, line 19</p> | 1-10                  |
| A         | <p>JIM AMSDEN ET AL: "Goals for Web<br/> Versioning;<br/> draft-ietf-webdav-version-goals-01.txt;"<br/> IETF STANDARD-WORKING-DRAFT, INTERNET<br/> ENGINEERING TASK FORCE, IETF, CH,<br/> vol. webdav, no. 1,<br/> 26 June 1999 (1999-06-26), XP015029961<br/> ISSN: 0000-0004<br/> abstract<br/> page 4, line 2 - page 5, line 31<br/> page 8, line 39 - page 9, line 43<br/> page 16, line 8 - page 17, line 2</p>                                                                                                                   | 1-10                  |

### Information on patent family members

PCT/US2006/008315

Patent document  
cited in search report

Publication  
date

Patent family member(s)

Publication date

|               |    |            |      |
|---------------|----|------------|------|
| US 2002198956 | A1 | 26-12-2002 | NONE |
|---------------|----|------------|------|