US 20020078087A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2002/0078087 A1**

Stone (43) **Pub. Date: Jun. 20, 2002**

(54) **CONTENT INDICATOR FOR ACCELERATED DETECTION OF A CHANGED WEB PAGE**

(76) Inventor: **Alan E. Stone**, Morristown, NJ (US)

Correspondence Address:
ANTONELLI, TERRY, STOUT & KRAUS, LLP
Suite 1800
1300 North Seventeenth Street
Arlington, VA 22209 (US)

(21) Appl. No.: **09/737,946**

(22) Filed: **Dec. 18, 2000**

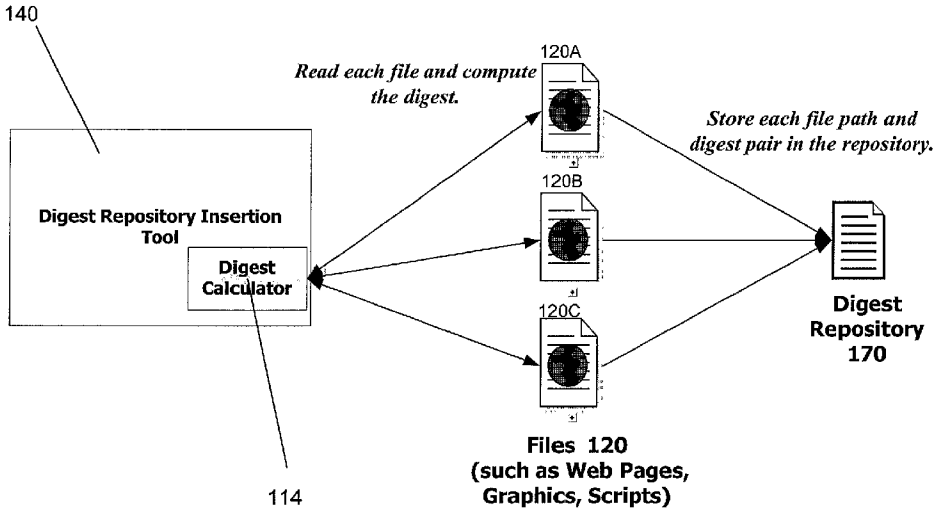
Publication Classification

(51) **Int. Cl.**[7] **G06F 17/21**

(52) **U.S. Cl.** **707/511**; 707/530

(57) **ABSTRACT**

Various embodiments are described for content indicators to accelerate detection of a changed web page or other file.



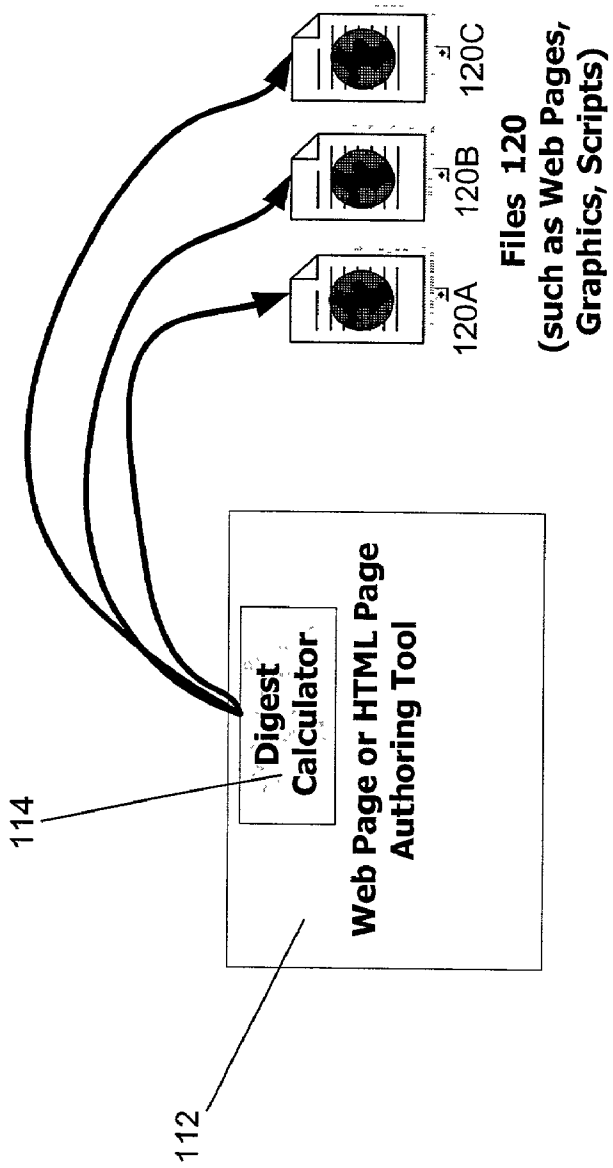

Files 120
(such as Web Pages,
Graphics, Scripts)
120A
120B
120C
Digest
Calculator
Web Page or HTML Page
Authoring Tool
114
112

FIG. 1

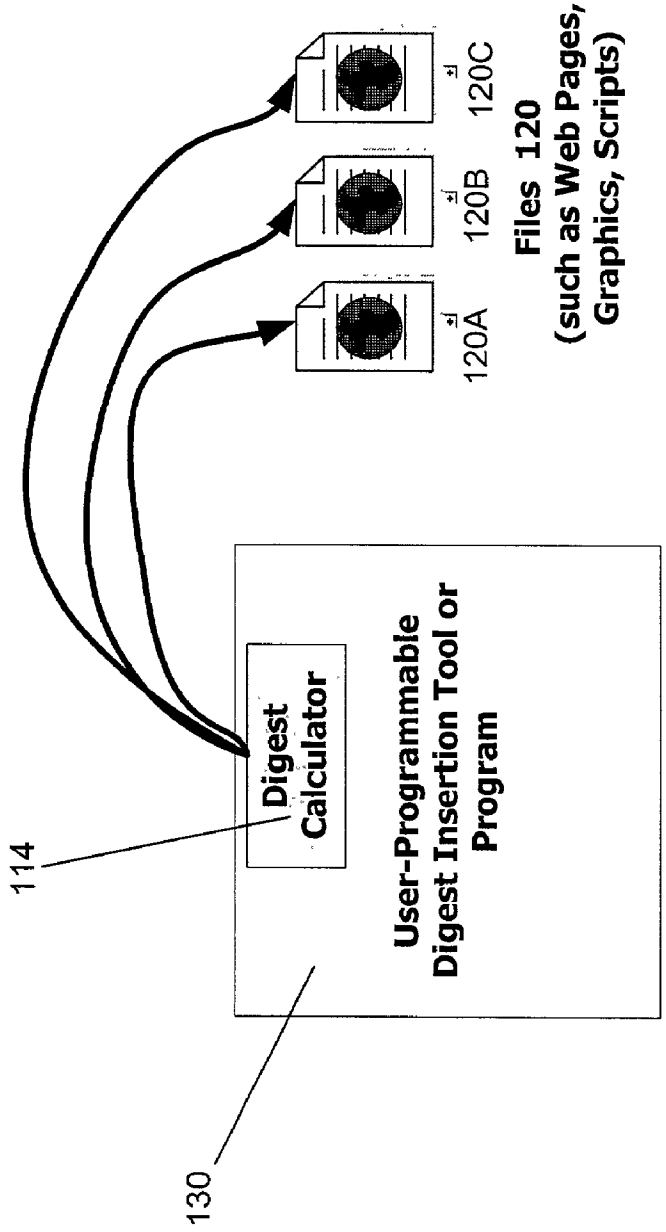

Files 120
(such as Web Pages, Graphics, Scripts)
120A
120B
120C
114
Digest Calculator
User-Programmable Digest Insertion Tool or Program
130

FIG. 2

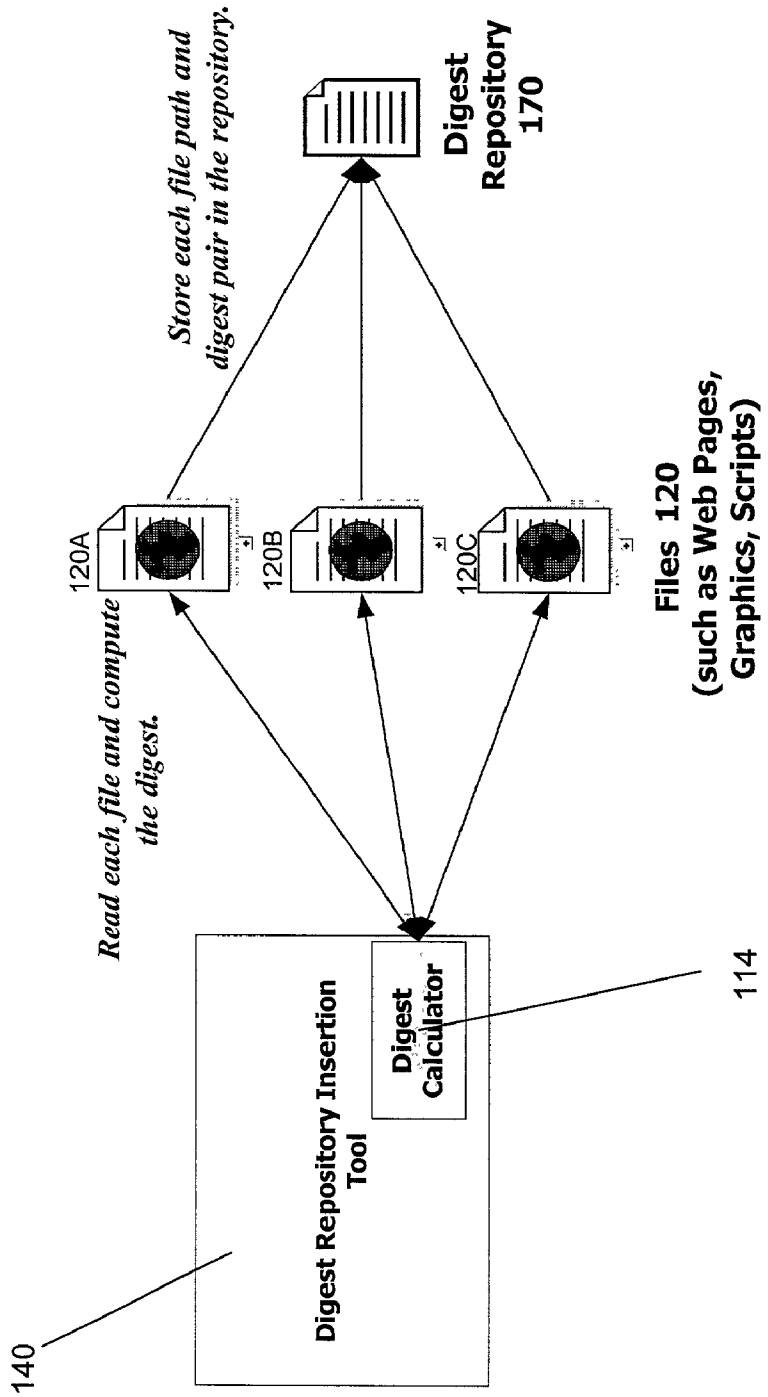

Store each file path and digest pair in the repository.
Digest Repository 170
120A
120B
120C
Files 120 (such as Web Pages, Graphics, Scripts)
Read each file and compute the digest.
Digest Repository Insertion Tool
Digest Calculator
114
140

FIG. 3

EXAMPLE HTML DOCUMENT

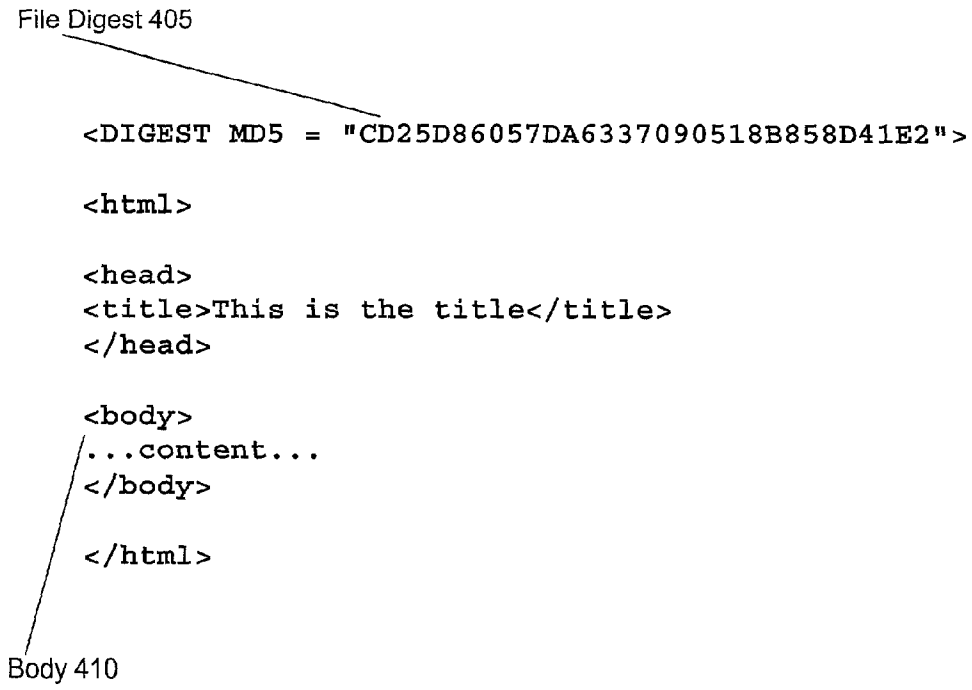


FIG. 4

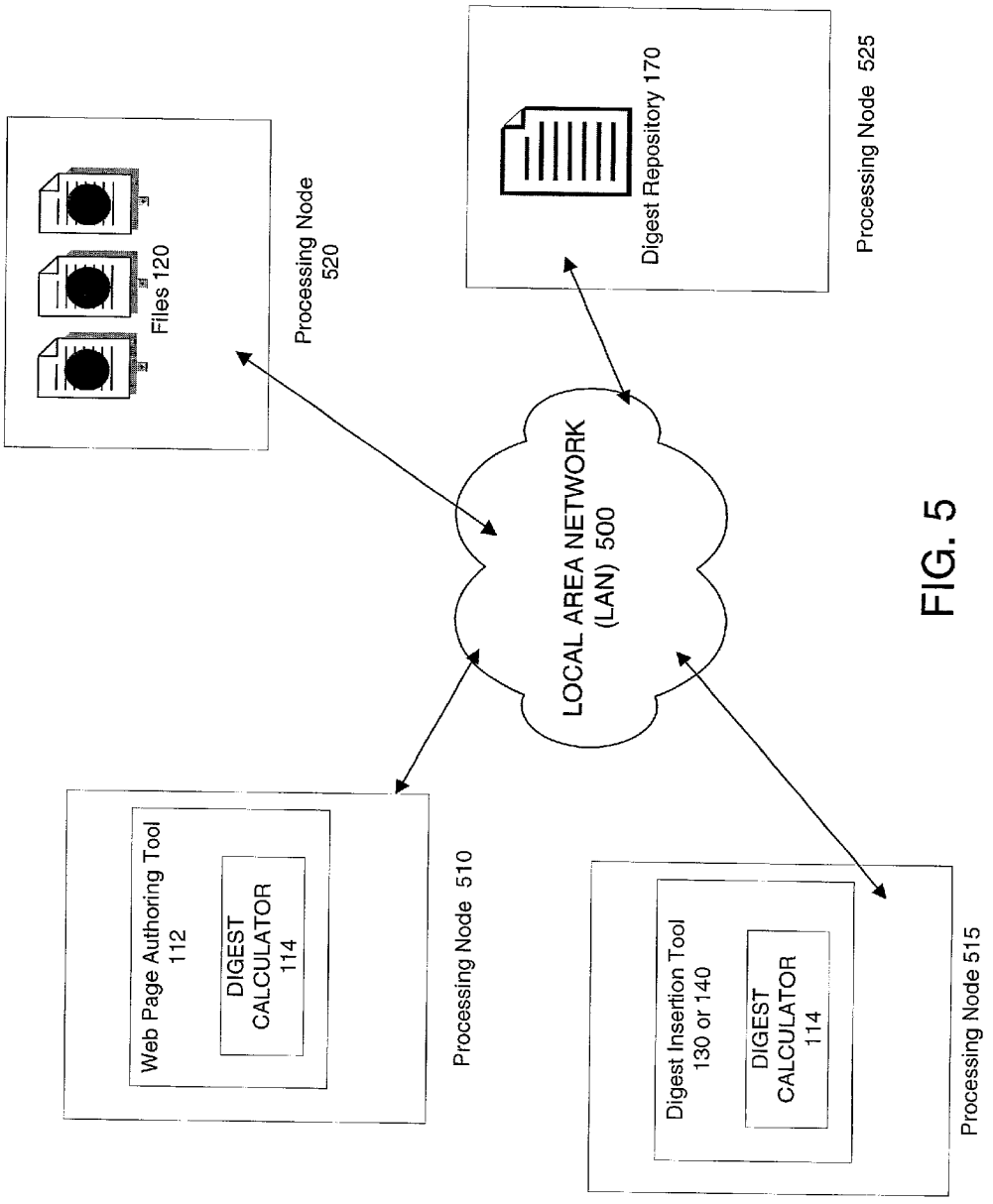

Files 120
Processing Node 520
Digest Repository 170
Processing Node 525
LOCAL AREA NETWORK (LAN) 500
Web Page Authoring Tool 112
DIGEST CALCULATOR 114
Processing Node 510
Digest Insertion Tool 130 or 140
DIGEST CALCULATOR 114
Processing Node 515

FIG. 5

CONTENT INDICATOR FOR ACCELERATED DETECTION OF A CHANGED WEB PAGE

FIELD

[0001] The invention generally relates to web pages, browsers and search engines, and in particular, to a content indicator for accelerated detection of a changed web page.

BACKGROUND

[0002] Today, web pages are commonly stored on web servers. A web server is a server that stores or provides web pages, typically in Hypertext Markup Language (HTML) format, and makes these web pages available to clients upon request, such as in response to a "Get" request using Hypertext Transfer Protocol (HTTP)—HTTP/1.1, Request For Comments 2616, June 1999. A client may be any software program that may request access the web pages. Two common web clients include a web browser and search engine indexers. A web browser is a program which can retrieve web pages from remote web servers and display the web page for the user.

[0003] The Internet is typically indexed via search engine indexers, also known as web "spiders." Typically, these spiders may be dedicated machines that relentlessly visit all the publicly addressable Internet addresses to gain access to the HyperText Transfer Protocol (HTTP) port number 80 to find "home pages" or "web pages." Once found, the spider navigates through the content of each 'page', indexing both content and hyperlinks. The index may provide, for example, a correspondence between the subject matter of a web page and an address or Universal Resource Identifier for each web page. This information is then provided to a search engine, to allow the search engine to identify addresses or locations of pertinent web pages in response to a particular search.

[0004] Changes to web pages can create problems for browsers and search engine indexers or spiders. Web content is frequently changed, by adding new content to pages, removing or adding new pages, or changing a hyperlink to another page, etc. When a browser retrieves a web page, a copy of the web page is stored in a local cache. When a second request for the cached web page is received at the browser from a user, the browser determines whether to use the cached copy of the web page, or whether to retrieve the web page from the web server. In HTTP/1.1 protocol, RFC 2068, a technique is described for the web server to provide a page content change indication. The content change indication is provided by either file size, file date, or a file digest specified by MD5 message digest algorithm, described in RFC 1321. The client can request one or more of these values from the web server for a particular page. The web server then retrieves the page from memory, calculates the file digest, file size or file date, and then returns this information to the client, where the client may use this information to decide whether to use the cached copy or request a copy from the web server. However, this is a slow and inefficient technique. Also, in some instances, web pages may be stored at a location where a web server is not available. For example, it is common to store web pages on a server or a network accessible drive, without the additional burden of an HTTP server. Thus, in such cases, it is desirable to obtain a page content change indication without querying the web server.

[0005] For the search engine, the changes in the web content can cause the web index to become outdated, which may create search results that include stale pages, pages that have moved or disappeared, broken links, etc. As a result, the web spider usually indexes web content relentlessly, constantly downloading indexing the same web content over and over again in attempt to provide updated indexes. This is very inefficient because this repetitive downloading of web pages consumes a large amount of bandwidth. As a result, it is desirable to provide a technique to obtain a page content change indication so that only the changed pages would be necessary to download and re-index.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] The foregoing and a better understanding of the present invention will become apparent from the following detailed description of exemplary embodiments and the claims when read in connection with the accompanying drawings, all forming a part of the disclosure of this invention. While the foregoing and following written and illustrated disclosure focuses on disclosing example embodiments of the invention, it should be clearly understood that the same is by way of illustration and example only and is not limited thereto. The spirit and scope of the present invention is limited only by the terms of the appended claims.

[0007] The following represents brief descriptions of the drawings, wherein:

[0008] **FIG. 1** is a diagram illustrating insertion of a digest or other content indicator into a file according to an example embodiment.

[0009] **FIG. 2** is a diagram illustrating insertion of a digest into a file according to another example embodiment.

[0010] **FIG. 3** is a diagram illustrating use of a digest according to yet another example embodiment.

[0011] **FIG. 4** is a diagram illustrating a HTML document according to an example embodiment.

[0012] **FIG. 5** is a block diagram that illustrates a network according to an example embodiment.

DETAILED DESCRIPTION

[0013] Referring to the Figures in which like numerals indicate like elements, **FIG. 1** is a diagram illustrating insertion of a digest or other content indicator into a file according to an example embodiment. As shown in **FIG. 1, a** web page or HTML page authoring tool **112** is provided to author or generate web pages or HTML pages. HTML authoring tool **112** typically may be a software program running on a processing node, such as a computer. The processing node or computer may include a processor, memory and other components. Web page authoring tool **112** may be, for example, software programs such as Front Page or Word, both available from Microsoft Corporation, Redmond, Washington.

[0014] According to the embodiment shown in **FIG. 1, a** page-resident content indicator may be provided for each page to allow programs or clients to detect web page changes. For example, the authoring tool **112** may include an additional program that calculates or generates a content

indicator for each file. The files may be, for example, a web page or HTML page, a graphic, a script, etc.

[0015] According to an example embodiment, a content indicator is calculated or generated for each web page. The content indicator may then be stored in or with the file or web page. A content indicator may be anything that allows a client or other program to detect a change or update to the content of the web pages. According to an example embodiment, a content indicator, when compared to another content indicator for the same web page, provides an indication as to whether or not the content of the web page has been changed or updated.

[0016] A content indicator may include, for example, a file size of the web page, a date and time that the web page was last modified or changed, and a file digest. When a file digest is calculated for a web page, a digest function takes an arbitrary sized message or file, such as a web page, and generates a number, which is typically a fixed length quantity. A hash algorithm or hash function, also known as a message digest is typically a one-way function. It is considered a function because it takes an input message and produces an output. It may be considered one-way because it is not practical to figure out what input corresponds to a given output. If it is cryptographically secure, it should be impossible to find two messages or files that have the same file digest. Thus, if a change is made to a web page, the digest for that page will change. The digest may be calculated, for example, using message digest algorithms, including MD2, MD4 and MD5, and documented in Request for Comments 1319, 1320, 1321, respectively. Other algorithms, such as hash functions or Cyclic Redundancy Checks (CRC) algorithms, etc. may be used to generate the file digests. The term digest will be used hereinbelow in the various embodiments and examples. However, other types of content indicators may be used as well.

[0017] Therefore, as shown in **FIG. 1**, the page authoring tool **112** includes a digest calculator **114** to calculate or generate a digest for each file or web page each time a web page or file is generated or created or updated, and then to store this digest with the corresponding web page. Files **120** includes files **120**A, **120**B and **120**C, which may be web pages, HTML pages or other types of files. Thus, according to an example embodiment, the digests may be page-resident, since the digests may reside with the corresponding web pages or files **120**.

[0018] **FIG. 4** is a diagram illustrating a HTML document according to an example embodiment. The web page authoring tool **112** (**FIG. 1**) generates or updates, or is used to generate or update, the HTML web page shown in **FIG. 4**, including the head and title of the message and the body of the message **410**. The digest calculator **114** (**FIG. 1**) then calculates or generates the file digest **405** based on the HTML page shown in **FIG. 4**. The file digest **405** may then be prepended or attached to or stored within the HTML file. Thus, each file or web page may include a corresponding digest that is encoded onto the file or web page.

[0019] The page-resident file digests for each of the files or web pages allows web indexers to quickly index the web pages since the indexer can identify which pages have changed, and then update the index using only changed web pages. For example, the indexer can read the file digest for each web page. If the digest for a web page matches the digest for a previous version of the web page that has already been indexed, then the indexer can skip this page and move on to the next web page without downloading the web page. If the digest for a web page is different from a previous digest for that web page, this indicates that the web page has changed, and the indexer can download and index that page. This allows the indexer to selectively download only those web pages that have changed, resulting in a significant decrease in bandwidth usage to index a set of web pages.

[0020] The page-resident digests for each of the stored web pages or files are also beneficial to the browsers that may be accessing these web pages. For example, in the event that the web pages are stored on a local storage drive or if a web server is not available, the browser may compare a digest from the cache-stored page to the digest from the page stored on the storage drive to determine if the cache-stored web page is invalid. If the cached copy of the page is invalid, as indicated by different digests, then the browser will retrieve the web page from the storage device. Otherwise, if the digests are the same, then this indicates that the cached copy of the page is still valid, and the browser may then use the cached copy, and need not download the entire web page from the network drive.

[0021] **FIG. 2** is a diagram illustrating insertion of a digest into a file according to another example embodiment. As shown in **FIG. 2**, as user-programmable digest insertion tool **130**, or a content indicator insertion tool in the general case, is provided. Rather than calculating a digest each time a file or web page is created, updated or saved, the digest insertion tool **130** can be programmed or directed to calculate updated digests for a plurality of files or web pages **120**, and then replace the existing digest in each file with the updated digest. The digest insertion tool **130** may also include or use the digest calculator to calculate or generate a digest for each file or web page.

[0022] **FIG. 3** is a diagram illustrating use of a digest according to yet another example embodiment. As shown in **FIG. 3, a** digest repository insertion tool **140** is provided to read each file or web page **120** and the file path. The file path for each file may be the path that identifies the location or address of the file in a network, for example. The file path may be a Universal Resource Identifier (URI) or a Universal Resource Location (URL), for example. The digest repository insertion tool **140** includes a digest calculator **114**. The digest repository insertion tool **140** then calculates or computes a digest for each web page or file, or uses the digest calculator **114** to perform these calculations. The digest repository insertion tool **140** then stores a file path and digest in a digest repository or storage **170**, for each file or web page. Two example file path and digest pairs are shown below:

[0023] 1) home/stonea/new.html MD5= "CD25D86057DA6337090518B858D41E2"

[0024] 2) home/stonea/improved.html home/stonea/new.html

[0025] Where "home/stonea/new.html" is the file path and "CD25D86057DA6337090518B858D41 E2" is the digest for file 1), shown above as an example.

[0026] It may be advantageous to store such an array or listing of file path and digest pairs for each of a plurality of files or web pages. This would allow a web indexer or a

browser to retrieve entries from the digest repository **170**, rather than retrieve portions of the web pages or files, to quickly obtain a current digest for each page or file. The client, indexer or web browser, may then compare the digest from the repository to a local copy of the digest for the same page to determine if the web page has changed, which would typically be indicated by digests that are different.

[**0027**] The page authoring tool **112**, digest insertion tools **130** or **140**, the files or web pages **120** and the digest repository **170** may be provided on a single processing node, or spread across multiple processing nodes, where a processing node may be a computer, a server or similar system.

[**0028**] **FIG. 5** is a block diagram that illustrates a network according to an example embodiment. For example, as shown in **FIG. 5**, web page authoring tool **112** may be a software program running on processing node **510**, digest insertion tool **130** or **140** may be a software program running on processing node **515**, files **120** may be stored in processing node **520**, while digest repository may be stored on processing node **525**. This is just an example network, however, the invention is not limited in scope to such a network or arrangement.

[**0029**] Several embodiments of the present invention are specifically illustrated and/or described herein. However, it will be appreciated that modifications and variations of the present invention are covered by the above teachings and within the purview of the appended claims without departing from the spirit and intended scope of the invention.

What is claimed is:

1. An apparatus comprising:

an authoring tool to generate files;

a calculator to calculate a content indicator for one or more of the files, the page authoring tool to store each of the calculated content indicators with a corresponding file.

2. The apparatus of claim 1 wherein the authoring tool comprises a HTML authoring tool or program.

3. The apparatus of claim 1 wherein the calculator comprises a digest calculator to calculate digests for each of the files.

4. The apparatus of claim 1 wherein the apparatus encodes each of the content indicators within a corresponding file.

5. An apparatus comprising:

an insertion tool comprising a calculator to calculate a content indicator for each of a plurality of files, the insertion tool to insert each of the calculated content indicators within a corresponding one of the files.

6. The apparatus of claim 5 wherein the calculator comprises a digest calculator to calculate digests for each of the files.

7. The apparatus of claim 5 wherein the apparatus encodes each of the content indicators within a corresponding file.

8. The apparatus of clam **6** wherein the insertion tool comprises an insertion tool to insert each of the calculated digests within a corresponding one of the files.

9. The apparatus of claim 5 wherein the files comprise one or more of the following:

web pages;

HTML pages;

Graphics; and

Scripts.

10. An apparatus comprising:

an insertion tool comprising a calculator to calculate a content indicator for each of a plurality of files, the insertion tool to obtain a path for each file and store a file path and the content indicator in a repository for each of a plurality of files.

11. The apparatus of claim 10 wherein the calculator comprises a digest calculator to calculate digests for each of a plurality of files.

12. The apparatus of claim 11 wherein the calculator comprises a digest calculator to calculate digests for each of a plurality of web pages.

13. The apparatus of claim 10 wherein the path file indicates a location or address of the file.

14. A method comprising:

generating a file;

calculating a content indicator for the file;

storing the content indicator with the file to provide content change indication for the file when compared to another content indicator.

15. The method of claim 14 wherein the calculating comprises calculating a digest for the file.

16. The method of claim 15 wherein the storing comprises storing the digest with the file to provide content change indication for the file when compared to another digest.

17. The method of claim 14 and further comprises:

retrieving the content indicator for the file;

comparing the content indicator for the file to a content indicator corresponding to a previous version of the file; and

determining whether the file has changed based on the comparing.

18. The method of claim 14 and further comprises:

retrieving the content indicator for the file;

comparing the content indicator for the file to a content indicator corresponding to a previous version of the file; and

retrieving the file if the content indicator for the file and the content indicator corresponding to a previous version of the file do not match.

19. The method of claim 18 and further comprising updating an index based on the retrieved file if there was not a match.

20. A method comprising:

obtaining a path for a file;

retrieving the file;

calculating a content indicator for the file;

storing the file path and the content indicator for the file in a repository or storage.

21. The method of claim 20 wherein the calculating comprises calculating a digest for the file.

22. The method of claim 21 wherein the storing comprises storing the file path and the digest in a digest repository or storage.

23. An apparatus comprising a readable media having instructions thereon, the instructions resulting in the following when executed:

generating a file;

calculating a content indicator for the file;

storing the content indicator with the file to provide content change indication for the file when compared to another content indicator.

24. The apparatus of claim 23 wherein the calculator comprises calculating a digest for the file.

25. The apparatus of claim 23 wherein the storing comprises storing the digest with the file to provide content change indication for the file when compared to another digest.

26. An apparatus comprising a readable media having instructions thereon, the instructions resulting in the following when executed:

obtaining a path for a file;

retrieving the file;

calculating a content indicator for the file;

storing the file path and the content indicator for the file in a repository or storage.

27. The apparatus of claim 26 wherein the calculating comprises calculating a digest for the file.

28. The apparatus of claim 27 wherein the storing comprises storing the file path and the digest in a digest repository or storage.

* * * * *