code for tags which designate objects, such as Anchor tags HREFs and image SRCs having the appropriate extensions such as GIF, JPEG and (Portable Network Graphics (PNG) that can be compressed and replaced with compressed content tags. Once the editing is complete the edited tags and images will reflect proprietary image type ".trans".

[0038] The Editor **422** gathers the image source URLs into a list along with their location and then submits them for compression at compressor **418**. The compressor **418** is, by way of example but not limited to, a JAVA application that is responsible for pacing the Black Box Compressor **432**. The Black Box Compressor **432** provides a corresponding JAVA native common interface to the compressors **418** and may support 1 to n compressor server **418**. The actual image files to be compressed are placed in a directory **420** that is equally writeable by both the Editor **422** and the Compression Server **418**. The Editor **422** writes the image file to the compression directory **420** using a temporary name. The Compression Server **418** then causes the Black Box Compressor **432** to create a new file with the ".trans" extension in the compression directory. Based on the success of the compression, the Editor **422** moves the edited and compressed URLS to the Web Server cache **414** where they are available for delivery on subsequent requests. Simultaneously with the actions described above the n-Depth Compressor **424** acting upon notification from the Redirector **416** parses out all of the links in the requested URL and sends the links that can be edited to the Editor **422**. These URLs are edited, compressed, and placed in the Web Server cache. The depth of this pre-fetching is configurable. Requests subsequent to an unedited page being returned to the requester will be delivered from the cache in an edited and compressed form, thereby greatly increasing the speed as well as continuity of content delivery.

[0039] The Cache Manager **426** is, by way of example, a JAVA object that handles interfacing with the Web Server cache **414**. Entries in the cache are hashed using a Cyclic Redundancy Check (CRC) algorithm, e.g., by calculating the Adler32 checksum of the characters in the URL. This provides 8 hexadecimal digits for forming a two-tier directory structure for each file. The original file extension is preserved for ease in identifying the contents of the file.

[0040] The Cache Server **428** is a C++ application that runs periodically to check the Web Server cache **414**. The Cache Server **428** uses a URL sorting algorithm and high- and low-watermark hysteresis to maintain the most active edited pages in the Web Server cache **414**. Because the Proxy Server **408** also caches these pages once they have been requested, some duplication of disk space is unavoidable. However, by utilizing the Internet Cache Protocol (ICP) supported by the Proxy Server **408**, the Redirector **416** can make use of the Proxy's cache **410** as well as the Web Server's cache **414**.

[0041] **FIG. 5** is a diagram of the components of the invention with Scanner according to an alternative embodiment of the invention **500**. The critical difference between this diagram and the diagram of **FIG. 4** is the replacement of the Editor with a Scanner. The Scanner **522** scans uncompressed requested pages for those that can be compressed and submits such pages to the Compressor. The scanner interoperates with the components of the network in a similar manner as does the editor described with reference to **FIG. 4**. However, using the scanner, the content/pages are only scanned to be identified if it needs compression or not. The content/pages are not edited but are only compressed and stored. The components of the network, in this embodiment, function in a similar fashion as described with reference to the embodiment of **FIG. 4**.

[0042] **FIG. 6** illustrates an overview diagram of components assembled on Distribution/Control Servers according to an alternative embodiment of the invention. Certain functions contained on servers in **FIG. 6** retain the same functions as illustrated and set out in **FIG. 4**. For example, the Proxy Server function illustrated in **FIG. 6608** is the same function shown by Proxy Serve **408** in **FIG. 4**. Names and designations of elements between **FIGS. 4 and 6** have generally been kept symmetrical so that one may refer back and forth between component descriptions where applicable. In this embodiment of the invention, Distribution/Control Server(s) **634** may reside on separate networks from the Control Server(s) and Compression Server(s) thereby acting as subsets of the Control Server **642**. The Control Server **642** permits the Distribution/Control Servers **634** to share compressed content, thereby increasing the cache hit rate of the Distribution/Control Servers **634** by allowing the Distribution/Control Servers **634** to access the Control Servers' **642** content, in effect increasing the size of the compressed content cache.

[0043] The requester, shown as the HTTP Client **602** makes a request **604** for information from a Content Server on the Internet **606**. The Proxy Server **608** residing on a Distribution/Control Server **634** intercepts the request and forwards the request to the Redirector **616**. If an edited and compressed version of the content is available locally then the URL is returned to the Proxy Server **608** which either fetches the URL from the Web Servers' Cache **614** and sends the edited and compressed content to the requestor where it is viewed with a browser having a Plugin **630**. However, if the Edited and Compressed URL exists in the Cache Server **628**, the Proxy Server **608** bypasses the fetch routine and returns the edited and compressed content to the requester. In the event that the request is not available locally, a request is sent from the Distribution/Control Server **634** to the Control Server **642** for the requested content. The Control Server **642** allows Distribution/Control Servers **634** to access the Control Server's cache **636**, thereby increasing the size of the network of compressed content. The Control Server's cache **636** is indexed with an associated Search Engine **638** to allow for searching of content or sites. URLs may be submitted to the Control Server **642** to be compressed, cached, indexed and stored in compressed or original form, or both. When the Distribution/Control Server(s) **634** make requests or submit URLs for n-depth compression **624** caching and indexing **636** may take place depending on the content. In the event that the Control Server **642** does not have a cached copy of the to content available, such content will be retrieved from the Content Server **640** on the Internet and may be sent back to the requestor **602** the first time in an uncompressed format. The requested content is contemporaneously edited, compressed, cached and placed in the Web Server's cache **614**. Thus upon subsequent requests the Proxy Server **608** may retrieve and deliver the previously cached compressed content to the requester **602**. The control server may be connected to the Internet and the distribution server, and can be accessed through either of the two connections. The main function of the control server is to