

METHOD AND APPARATUS FOR COMPRESSION AND DECOMPRESSION OF DATA

5 FIELD OF INVENTION

This invention relates to a method and apparatus for compression and decompression of data, more particularly but not exclusively to such a method and apparatus for use in an Internet or other data network environment.

10

BACKGROUND OF THE INVENTION

The proliferation of the Internet over the past few years and the development of new and innovative Internet devices like (wireless) personal digital assistants (PDAs), mobile cellular phones (GSM, PCS, PHS, GPRS, 3G, CDMA, UMTS, W-CDMA), cable modem, digital satellite receivers, wireless LAN (802.11), Bluetooth and infrared protocol (IRDA) has brought about an 'information age' era where larger and larger amount of data are being communicated across the world every second using different communication protocols. However, despite rapid progress in Internet access technologies (e.g. from 28.8kbps to 56kbps to broadband over cable, ADSL and beyond) and computer processing speeds, demand for data bandwidth continues to outstrip the capabilities of available technologies. The need for more bandwidth and speed of access is getting more important for information delivery everyday. This is because most Internet data which consists of text, images, audio and video data requires considerable transmission bandwidth. The majority of successful web sites tend to use color-rich graphic content for appeal. These graphics can consume more than 50% of the data volume found within the web site and html pages. Digital audio and video files however require millions of bytes per second for transmission are transmitted in raw uncompressed form. The recent growth of 'content-rich' multimedia-based web applications (e.g. video-on-demand, video conferencing, e-Education) have only further pushed the need for more efficient ways to

transport this high bandwidth data across the vastly diversified and limited Internet bandwidths to the end user. The network "pipe" as it is known has been the primary focus of the Internet where the speed of access and bandwidth is as fast as the "pipe" size. The bigger the "pipe" the more the data can be
5 transported. All data pipes whether large or small have inherent limitations related to performance and throughput.

Proposals has been made to increase throughput by modifying and compressing the communication protocols, performing overall data compression
10 on the text/character based data and/or optimising the color palettes of the graphics content of indexed colour images, such as GIF files, but even with these improvements, limitations on data throughput are still a problem.

It is an object of the present invention to provide a novel data compression
15 method and apparatus which is usable to alleviate the problem of the prior art and/or provides the public with a useful choice.

SUMMARY OF INVENTION

20 According to the invention in a first aspect, there is provided data compression apparatus arranged to compress data according to at least two data compression techniques by dividing the data into portions and applying a selected said technique to each portion in dependence upon at least one property of the portion.

25

The portions preferably each comprise at least one data object, a said property preferably being object type and/or size.

The two data compression techniques may comprise respective different
30 compression algorithm such as a lossless compression algorithm and an adaptive compression algorithm.

The data is preferably divided in dependence upon content, for example a text portion and an image portion. The image portion may be further sub-divided into at least one index-colour image portion and at least one natural image portion and separate techniques applied to each sub-divided portion. The
5 separate techniques may be further chosen in dependence upon the size of the sub-divided portion.

Each compressed portion is preferably marked with an identity tag identifying the compression technique.

10

The invention may be used for real time compression and decompression of web objects over a data communications network.

The data compression apparatus in a first preferred form is associated with an
15 Internet service provider and arranged to compress data to be sent from the Internet service provider to a client of the Internet service provider and/or vice versa. More preferably the apparatus is functionally integrated with a proxy server, the arrangement being such that the data is received from the Internet via the proxy server, is passed to the apparatus for compression and passed
20 back to the proxy server for transmission to a said client. The proxy server may cache a copy of each data portion.

In another preferred form, the data compression apparatus is arranged to form part of a corporate network or virtual private network or is associated with a
25 server arranged to compress and send data to a further server.

A data decompression apparatus arranged to decompress data compressed by the data compression apparatus is also envisaged, which may be disposed at the client and/or server side of an Internet connection or connected to the
30 corporate network or virtual private network, by way of example.

According to the invention in a second aspect, there is provided apparatus for

compression and de-compression of data comprising a data compression means arranged to compress data according to at least two data compression techniques by dividing the data into portions and applying a selected said technique to each portion in dependence upon at least one property of the portion and a data de-compression means for de-compressing data compressed by the data compression means and wherein the data compression and de-compression means are arranged to compress and de-compress data to be transmitted over a data transmission link between an Internet service provider and a client of the Internet service provider.

10

According to the invention in a third aspect, there is provided apparatus for compression and de-compression of data comprising a data compression means arranged to compress data according to at least two data compression techniques by dividing the data into portions and applying a selected said technique to each portion in dependence upon at least one property of the portion and a data de-compression means for compressing data compressed by the data compression means and wherein the data compression and de-compression means are arranged to compress and de-compress data to be transmitted over a data transmission link between a network data server and a network client.

20

According to the invention in a fourth aspect, there is provided apparatus for compression and de-compression of data comprising a data compression means arranged to compress data according to at least two data compression techniques by dividing the data into portions and applying a selected said technique to each portion in dependence upon at least one property of the portion and a data de-compression means for de-compressing data compressed by the data compression means and wherein the data compression and de-compression means are arranged to compress and de-compress data to be transmitted over a data transmission link between a first server or peer and a second server or peer.

25
30

According to the invention in a fifth aspect, there is provided a method of data compression comprising the steps of dividing the data into portions and applying a selected one of at least two data compression techniques to each portion in dependence upon at least one property of the portion.

The described embodiment of the invention provides a transport and compression mechanism that reduces the data size and thus 'speeds up' the flow of data in a bandwidth limited network with little change being needed to the existing communication protocol and network infrastructure. The described embodiment adopts an object based approach to compress the data at the individual web object level (e.g. text, images, gif, jpeg, html files) using the most suitable and efficient compression algorithm for each object. This results in a more efficient compression system that compresses the web data content before transmission to the client machine (personal computer, PDA, mobile phone, terminal etc.)

A client-server architecture and methodology is provided in the described embodiment to reduce data transmission over a bandwidth limited area (e.g. the 'last mile') of a data communications network such as the Internet. At the server or higher bandwidth end of the bandwidth limited data communication network resides an acceleration server which splits up the data at an individual object (content type) level and applies an intelligent, content type based compression algorithm to compress and optimize the data content (e.g. text, picture etc) with lossless and lossy (minimal perception loss) quality. At the other end of the bandwidth limited network a client application is provided which receives the compressed data from the server and performs the task of de-compression and reassembles the data back to its original format. The client application is used in conjunction with the client's data communication software such as web browser, email software etc, to provide a seamless (i.e. transparent) connection in the data communication link.

BRIEF DESCRIPTION OF THE DRAWINGS

An embodiment of the invention will now be described by way of example with reference to the accompanying drawings in which:

5

Figure 1 illustrates an embodiment of the invention applied to a low bandwidth data link;

10

Figure 2 is a block diagram illustrating the acceleration server of Figure 1;

Figure 3 is a block diagram illustrating the client application of Figure 1;

15

Figure 4 illustrates implementation of the described embodiment within an Internet Service Provider (ISP) access network; and

Figure 5 illustrates implementation of the described embodiment within a typical corporate network.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

20

The overall data flow in a client-server based network typically has a bottleneck in the form of a low bandwidth data link at the 'last mile' nearest to the client's end .

25

With reference to Figure 1 an embodiment of the invention is illustrated which attempts to increase the apparent bandwidth of the 'last mile' link by using an acceleration server 100 (denoted VIPS_S) at the server end to compress the data before feeding the compressed data via the low bandwidth data link 110 to the client 130 via a client application 120 for decompressing the compressed data (denoted by VIPS_C). The net effect of the server 100 and application 120 is an apparent increase in data transfer rate for the user because more data can be transferred in less time.

30

The server 100 is shown in more detail in Figure 2 and includes an object parser 200 which separates the data into individual objects based on its content type, like text and images. This is then fed to an adaptive compressor 210 (for images) and a lossless compressor 220 (for text and characters) depending on object property. After compression, each compressed object is tagged with an identifier which identifies the compression method and the objects are recombined using the native protocol for the data link (HTTP for an Internet access network) to form compressed data at block 230 for transmission and delivery to the end user. All these processes are carried out in real time.

10

The adaptive compression algorithm block 210 in Figure 2 handles image formats like GIF, bitmaps and Jpegs. For index-colour images like GIFs, a colour optimization algorithm using a standard colour quantization technique such as the median cut algorithm (Heckbert P, Proc of SIG-Graph, ACM 1982), or variance minimization (Wu X, Graphics Gems, vol II, 1991) is employed to re-compress the GIF encoded image without sacrificing image quality (since GIF is a lossless compressed format). As for other natural colour image formats like bitmaps and Jpegs, the adaptive compression algorithm attempts to compress the image to a certain target quality factor (visually lossless) and compression ratio (file size) base on the original image size, spatial and colour attributes. The natural colour image compression algorithm (for Jpegs and Bitmap objects) uses a wavelet compression algorithm such as a standard bi-orthogonal (9,7) and (5,3) wavelet algorithm (Daubechies, Communications on Pure and Applied Mathematics, Volume 41, 1988). The upcoming Jpeg 2000 standard, which is also based on wavelet compression technology, may also be used.

25

A feature of the described embodiment is that the decision of the kind of "compression" that is used for images is not solely dependent on image type. In this respect, the block 210, before compression/optimization, sorts images, firstly into index or natural colour and then categorizes the images according to size so that for larger images of whatever type, the wavelet compression algorithm is chosen and for smaller images, the colour quantization method is

30

used. The reason for this is that a standard colour optimization algorithm only achieves 20-25 percent "compression" improvement. This is sufficient for small images of, say, less than a threshold size of 150 x 150 pixels but above this size, a more substantial compression technique is desirable. Similarly, for small
5 natural colour images of less than a threshold size of 80 x 80 pixels, the colour optimization method may produce a better result than the Wavelet method. Thus, in deciding the compression algorithm to use a decision based on type and size is made.

10 This can be achieved by estimating the approximate size of a final compressed object based on the known effectiveness of the algorithms and choosing the algorithm that provides the smaller estimate, this analysis being performed by the block 210 before compression to determine the most suitable algorithm. The block 210 also performs a check that the estimated size of the final
15 compressed object is smaller than the original object (at a target image quality) before making a decision to compress the object. The information of the approximate size of the final compressed object may be in the form of a look-up table for size and type of file or a calculation algorithm can be used. Alternatively, simple file size/type thresholds such as those mentioned in the
20 preceding paragraph may be used to decide which algorithm is employed.

For the text and character based objects these are compressed using a lossless data compression algorithm such as the LZ77 (J. Ziv, A. Lempel, IEEE Trans. Inform. Theory, 1977, vol. 23) which is a high speed and efficient data
25 compression algorithm.

The file name of each compressed object is marked at block 230 with an ID tag identifying the type of compression (e.g. *.html.t for html text with "t" identifying a lossless data compression algorithm, *.jpg.z for JPG, *.bmp.z for bitmaps with
30 "z" identifying an adaptive compression algorithm) so that the algorithm can be readily identified for subsequent decompression. Objects which are not compressed or are subject to a colour optimization algorithm, for example large

GIF and small natural colour image objects are not marked with an ID tag since, as they still remain in their original format, no decompression is required.

At the user/client side, the client application 120 is a software module that runs
5 between the client's other applications (e.g. internet browser, e-mail) and the physical communication port. The application 120 acts as a local proxy server to communicate with the client's communication software, via a designated IP address.

10 To interface to the client 130, for example in the user's web browser software, the web browser's proxy setting is set to the designated IP address of the application 120. In this way all the browser traffic (e.g. HTTP request, download) is re-directed to the client application 120 before going to the actual communication port. Once the application 120 is enabled, this would connect to
15 the appropriate server 100 forming a 'virtual private connection' on top of the existing data communication link.

The client application 120 is illustrated in Figure 3 and includes object passer 300, de-compressor 310, control and admin 320 and network control 330
20 modules. The client application 120 intercepts the compressed data, extracts its content, and performs decompression of the various object types before feeding them to the client's application software. All the compressed objects will be converted back to their respective original formats so that the compression is transparent to the end user.

25

The process of decoding the various compressed objects (e.g. html text, bitmaps, Jpegs) is first achieved by identifying the ID. Next, the decompression of the various compressed objects takes place. All compressed objects are decompressed back to their respective original format based on information on
30 the (original) file extension and ID tag. For GIF images, as they still remain in GIF format after optimization (at the server 100) they do not require any decompression. Therefore the final objects that appear at the user's web

browser would be exactly the same format as the original content.

The acceleration server 100 may also function as a proxy server to cache all the previously compressed objects in order to save repeat processing time if the
5 object is requested again by the user, conserving the incoming data link bandwidth and to increasing the workload of the server. In this way the number of simultaneous client accesses to the server can be increased considerably.

In an Internet environment, the server 100 may be functionally integrated to a
10 Internet proxy server software (e.g. Squid for the Linux operation system). This will further increase the performance of the server 100 by taking advantage of the caching capability of the proxy server to support large number of on-line user requests. As all processed web objects will be cached by the proxy server (after the first request), any subsequent request of the same web object (e.g. a
15 second user which request a similar web page) can be delivered very quickly by the proxy server software without going through the compression process again.

Another function of the server 100 is to serve as a main control centre for the
20 remote clients whereby the activities of the remote clients can be monitored (e.g. the sites visited) and controlled (e.g. by enabling and disabling the remote client application 120). All the control information between the server 100 and client application 120 are encrypted.

25 For control and monitoring purposes, the client application 120 also performs the task of user log-in and authentication in conjunction with the server 100. Upon logging in the client application 120 begins to monitor the user's on-line usage time by communicating with the clock of the server 100. If the user exceeds the allowed usage time limit the client application 120 would alert the
30 user (for commercial purposes) or disconnect the user from the server 100 (i.e. discontinue the service) by re-routing back to the original proxy setting.

At the same time, the client application 120 is also capable of monitoring the user's surfing activity, acting like a front-line proxy server. This has very useful application for example in monitoring the user's surfing habits, preference, etc as well as blocking access to certain undesirable sites. The monitored data and sites blocking list are updated in encrypted form by the server 100 periodically.

In practice the client application 120 and server 100 architecture can be configured in a number of ways to achieve the desired effect. The most preferred application to accelerate the user's Internet surfing speed is to implement this architecture within a Internet Service Provider (ISP) access network, as shown in Figure 4.

In Figure 4, the server 100 is connected to a main ISP backbone which links a dial-up access server 410 to a proxy server 420, firewalls 430 etc. More than one server 100 can be connected to this backbone to in proportion to the number of user base. The server 410 is connected to a plurality of clients 130, each having the client application 120.

When a user is connected to the ISP and requests a Internet web page, the web page would be fetched from the Internet 450 via an ISP gateway 440 and compressed by the server 100 therefore effectively increases the apparent surfing speed for the end user as less data are downloaded. Besides increasing the last-mile connection bandwidth (for the end user), this system also increases the capacity of the ISP dial-up network as more users can be served for a given fixed bandwidth capacity.

The single ISP architecture concept can be potentially expanded to a large scale, global implementation by putting the server(s) 100 in major nodes, ISPs or gateways, for example in every country. The server 100 can then be configured to function as an intelligent router/proxy to fetch the appropriate compressed data (upon user request) from the server 100 located nearest to the original source data. For example, if a user located in Asia requests a web

page hosted at a US site, the user's client application 120 would fetch the compressed data directly from a server 100 located nearest to the US host, and subsequently cache it on the local server 100 to which the user dialed-up to. In this way the bandwidth reduction is extended to the whole Internet network
5 resulting in considerable savings in bandwidth cost and delay time.

On a smaller scale, the client application 120 and server 100 can be applied in a corporate network or virtual private network to provide accelerated access to information hosted within the network, as illustrated in Figure 5. For example, a
10 stock broking house, hospital or news agency could install the server 100 within a network backbone 600 in order for their staff to enjoy faster access to the information stored in their corporate servers 610, 620 as well as from the Internet 630. The information can be accessed in a number of ways by using the client application 120, for example through the LAN 600 directly (not shown)
15 or by means of dial-up 640 or wireless access 650 from home or an outside location via a remote access server 660.

The described embodiment is not to be construed as limitative. For example, the data may be separated into portions other than objects if a property of the
20 portions is identifiable to allow selection of an appropriate compression technique. Furthermore the property need not be object type and/or size but could be based on any suitable selection criterion. The objects themselves may be split before compression for example into data packet-size portions. Other forms/protocols of data communication may be used, such as (but not limited
25 to) email (smtp protocol), newsgroup (nntp protocol), File Transfer Protocol (FTP), Short Message Service (SMS protocol) or Internet Relay Chat (RFC 1459 protocol).

The invention is not limited to server to client data transmission and as well as
30 or instead, compressed client to server data transmission may also be provided with the client performing the compression and the server the decompression, rather than vice versa as described. Server to server and peer to peer data

communication is also within the scope of the invention. Both servers/peers may be unidirectional or have effectively the functionality of both a server and a client as described to allow bi-directional compression and decompression. Server to server communication is mainly used for example in international IPX
5 networks and trunk networks etc in order to maximize international data bandwidth. For example, if a pair of servers as described are located between the US and Singapore effectively all data traffic between the servers (which could be part of a gateway for a larger national data networks) would be compressed hence substantial bandwidth savings can be achieved.

CLAIMS

1. Data compression apparatus arranged to compress data according to at least two data compression techniques by dividing the data into portions and
5 applying a selected said technique to each portion in dependence upon at least one property of the portion.
2. Apparatus as claimed in claim 1 wherein the portions each comprise at least one data object.
- 10 3. Apparatus as claimed in claim 2 wherein a said property is object type.
4. Apparatus as claimed in claim 2 or claim 3 wherein a said property is object size.
- 15 5. Apparatus as claimed in any one of the preceding claims wherein the two data compression techniques comprise respective different compression algorithms.
- 20 6. Apparatus as claimed in claim 5 wherein the algorithms comprise a lossless compression algorithm and an adaptive compression algorithm.
7. Apparatus as claimed in any one of the preceding claims wherein the data is divided in dependence upon content.
- 25 8. Apparatus as claimed in claim 7 wherein the data is generally divided into a text portion and an image portion.
9. Apparatus as claimed in claim 8 wherein the image portion is further sub
30 divided into at least one index-colour image portion and at least one natural image portion and separate techniques are applied to each sub-divided portion.

10. Apparatus as claimed in claim 9 wherein said separate techniques applied to each sub-divided portion are further chosen in dependence upon the size of the sub-divided portion.

5

11. Apparatus as claimed in any one of the preceding claims further comprising means for marking each compressed portion with an identity tag identifying the compression technique.

10 12. A data compression apparatus as claimed in any one of the preceding claims associated with an Internet service provider and arranged to compress data to be sent from the Internet service provider to a client of the Internet service provider and/or vice versa.

15 13. An apparatus as claimed in claim 12 wherein the apparatus is functionally integrated with a proxy server, the arrangement being such that the data is received from the Internet via the proxy server, is passed to the apparatus for compression and passed back to the proxy server for transmission to a said client.

20

14. Apparatus as claimed in claim 13 wherein the proxy server caches a copy of each data portion.

15. A data compression apparatus as claimed in any one of claims 1 to 11
25 associated with a server and arranged to compress data to be sent from the server to a further server.

16. A data compression apparatus as claimed in any one of claims 1 to 11 arranged to form part of a corporate network or virtual private network.

30

17. A data decompression apparatus arranged to decompress data compressed by the data compression apparatus of any one of the preceding

claims.

18. A data decompression apparatus arranged to decompress data
compressed by the data compression apparatus of any one of claims 12 to
5 14 at the client and/or server side.

19. A data decompression apparatus arranged to decompress data compressed
in the data compression apparatus of claim 15 and associated with the
further server.

10

20. A data decompression apparatus arranged to decompress data
compressed in the data compression apparatus of claim 16 and connected
to the corporate network or virtual private network.

15 21. Apparatus for compression and de-compression of data comprising a data
compression means arranged to compress data according to at least two
data compression techniques by dividing the data into portions and applying
a selected said technique to each portion in dependence upon at least one
property of the portion and a data de-compression means for de-
20 compressing data compressed by the data compression means and wherein
the data compression and de-compression means are arranged to compress
and de-compress data to be transmitted over a data transmission link
between an Internet service provider and a client of the Internet service
provider.

25

22. Apparatus for compression and de-compression of data comprising a data
compression means arranged to compress data according to at least two
data compression techniques by dividing the data into portions and applying
a selected said technique to each portion in dependence upon at least one
30 property of the portion and a data de-compression means for compressing
data compressed by the data compression means and wherein the data
compression and de-compression means are arranged to compress and de-

compress data to be transmitted over a data transmission link between a network data server and a network client.

23. Apparatus for compression and de-compression of data comprising a data
5 compression means arranged to compress data according to at least two data compression techniques by dividing the data into portions and applying a selected said technique to each portion in dependence upon at least one property of the portion and a data de-compression means for de-compressing data compressed by the data compression means and wherein
10 the data compression and de-compression means are arranged to compress and de-compress data to be transmitted over a data transmission link between a first server or peer and a second server or peer.
24. A method of data compression comprising the steps of dividing the data
15 into portions and applying a selected one of at least two data compression techniques to each portion in dependence upon at least one property of the portion.