(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 827 336 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
06.10.1999 Bulletin 1999/40

(43) Date of publication A2:
04.03.1998 Bulletin 1998/10

(21) Application number: **97306679.8**

(22) Date of filing: **29.08.1997**

(51) Int Cl.6: **H04N 5/44**, H04N 7/16, H04N 7/24

(84) Designated Contracting States:
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE

(30) Priority: **30.08.1996 JP 23001596**
11.03.1997 JP 5668797

(71) Applicant: **MATSUSHITA ELECTRIC INDUSTRIAL CO., LTD.**
Kadoma-shi, Osaka-fu, 571 (JP)

(72) Inventors:
- **Shimoji, Tatsuya**
  Neyagawa-shi, Osaka-fu 572 (JP)
- **Okamura, Kazuo**
  Hirakata-shi, Osaka-fu 573 (JP)
- **Hirai, Junichi**
  Suita-shi, Osada-fu, 564 (JP)
- **Oashi, Masahiro**
  Hirakata-shi, Osaka-fu 573-01 (JP)
- **Kakiuchi, Takashi**
  Toyonaka-shi, Osaka-fu 561 (JP)
- **Kusumi, Yuki**
  Kashiba-shi, Nara-ken 639-02 (JP)
- **Miyabe, Yoshiyuki**
  Osaka-shi, Osaka-fu 532 (JP)
- **Minakata, Ikuo**
  Souraku-gun, Kyoto-fu 619-02 (JP)
- **Kozuka, Masayuki**
  Neyagawa-shi, Osaka-fu 572 (JP)
- **Mimura, Yoshihiro**
  Hirakata-shi, Osaka-fu 573 (JP)
- **Inoue, Shinji**
  Neyagawa-shi, Osaka-fu 572 (JP)
- **Mori, Toshiya**
  Settsu-shi, Osaka-fu 566 (JP)
- **Takao, Naoya**
  Kadoma-shi, Osaka-fu 571 (JP)

(74) Representative: **Crawford, Andrew Birkby et al**
A.A. THORNTON & CO.
Northumberland House
303-306 High Holborn
London WC1V 7LE (GB)

(54) Digital broadcasting system, digital broadcasting apparatus, and associated receiver therefor

(57) A broadcasting system which includes a broadcasting apparatus and a reception apparatus and which achieves interactiveness using a broadcast wave. The broadcasting apparatus includes a content storing unit for storing the plurality of contents, each content including a set of video data and a set of control information that indicates another content that is a link destination for a present content, and a transmitting unit for multiplexing a set of video data and a plurality of sets of the same control information included in a same content as the set of video data, and for transmitting the multiplexed sets of video data and control information. The reception apparatus includes an extracting unit for extracting a set of video data and a set of control information in a same content as the set of video data, a storing unit for storing the extracted set of control information, a reproducing unit for reproducing the extracted set of video data and outputting an image signal, an operation unit for receiving a user operation that indicates a content switching, and a control unit for controlling the extracting unit to extract another content indicated by the set of control information stored in the storing unit, in accordance with the user operation.

Printed by Jouve, 75001 PARIS (FR)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 97 30 6679

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X A	EP 0 680 216 A (THOMSON CONSUMER ELECTRONICS) 2 November 1995 (1995-11-02) * page 3, line 14 - line 35 * * page 4, line 19 - line 30 * * page 4, line 53 - page 5, line 8 * ---	1,29,43, 48 14,28, 47,49	H04N5/44 H04N7/16 H04N7/24
X A	US 5 422 674 A (HOOPER DONALD F ET AL) 6 June 1995 (1995-06-06) * column 5, line 59 - line 68 * * column 7, line 39 - column 8, line 2 * * column 9, line 45 - line 65 * * figure 7 * ---	1,29,43, 48 14,28, 47,49	
A	US 5 541 662 A (ADAMS ROBERT ET AL) 30 July 1996 (1996-07-30) * column 4, line 45 - line 65 * * column 5, line 39 - line 47 * * column 7, line 43 - column 8, line 23 * * column 8, line 64 - column 9, line 10 * --- -/--	1,14,28, 29,43, 47-49	TECHNICAL FIELDS SEARCHED (Int.Cl.6) H04N

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	17 August 1999	Sindic, G

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 97 30 6679

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X A	COLAITIS F: "LA NORME MHEG, VIDEO A LA DEMANDE ET TELEVISION INTERACTIVE. THE MHEG STANDARD FOR VIDEO ON DEMAND AND DIGITAL TELEVISION SERVICES" REE: REVUE GENERALE DE L ELECTRICITE ET DE L ELECTRONIQUE, no. 4, 1 October 1995 (1995-10-01), pages 33-39, XP000533328 ISSN: 1265-6534 * page 36, left-hand column, paragraph 1 - paragraph 3 * * page 38, right-hand column, paragraph 3 *	1,29,43, 48 14,28, 47,49	
A	MEYER-BOUDNIK T ET AL: "MHEG EXPLAINED" IEEE MULTIMEDIA, vol. 2, no. 1, 21 March 1995 (1995-03-21), pages 26-38, XP000500084 ISSN: 1070-986X * page 28, left-hand column, paragraph 4 - right-hand column, paragraph 4 * * page 32, right-hand column, paragraph 4 * * page 33, left-hand column, paragraph 5 - page 34, right-hand column, paragraph 1 *	1,14,28, 29,43, 47-49	TECHNICAL FIELDS SEARCHED (Int.Cl.6)
A	WO 96 05699 A (ACTV INC) 22 February 1996 (1996-02-22) * page 12, line 22 - line 33 * * page 14, line 25 - page 15, line 4 * * page 18, line 25 - page 19, line 7 * -/--	1,14,28, 29,43, 47-49	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	17 August 1999	Sindic, G

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 97 30 6679

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
P,X P,A	ATZORI L ET AL: "TRANSMISSION OF HYPERTEXTUAL INFORMATION USING MPEG-2 PRIVATE DATA" PROCEEDINGS OF THE SPIE, vol. 2952, 7 October 1996 (1996-10-07), pages 582-590, XP000675526 * page 582, paragraph 4 - page 583, paragraph 1 * * page 585, paragraph 5 - paragraph 6 *	1,29,43,48 14,28,47,49	
P,X P,A	WO 96 42144 A (NOKIA OY AB ;SALOMAEKI ARI (FI)) 27 December 1996 (1996-12-27) * page 4, line 24 - line 32 * * page 7, line 1 - line 8 *	1,29,43,48 14,28,47,49	
			TECHNICAL FIELDS SEARCHED (Int.Cl.6)

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	17 August 1999	Sindic, G

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

IPR2022-01227
EXHIBIT 1003 - PAGE 00704

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 97 30 6679

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

17-08-1999

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0680216 A	02-11-1995	US 5548532 A	20-08-1996
		US 5448568 A	05-09-1995
		CN 1114814 A	10-01-1996
		JP 8046950 A	16-02-1996
		SG 30364 A	01-06-1996
US 5422674 A	06-06-1995	DE 69418580 D	24-06-1999
		EP 0660609 A	28-06-1995
		US 5493638 A	20-02-1996
US 5541662 A	30-07-1996	AU 3411595 A	26-04-1996
		EP 0783818 A	16-07-1997
		WO 9610888 A	11-04-1996
		US 5760838 A	02-06-1998
WO 9605699 A	22-02-1996	US 5585858 A	17-12-1996
		AU 688496 B	12-03-1998
		AU 2290895 A	07-03-1996
		BR 9506316 A	05-08-1997
		CA 2173996 A	22-02-1996
		EP 0723729 A	31-07-1996
		JP 9504156 T	22-04-1997
WO 9642144 A	27-12-1996	FI 98175 B	15-01-1997
		AU 6127696 A	09-01-1997
		EP 0872053 A	21-10-1998

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

IPR2022-01227
EXHIBIT 1003 - PAGE 00705

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 859 535 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
19.08.1998 Bulletin 1998/34

(51) Int. Cl.[6]: **H04Q 11/04**

(21) Application number: **98102634.7**

(22) Date of filing: **16.02.1998**

(84) Designated Contracting States:
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **17.02.1997 JP 31610/97**

(71) Applicant:
Matsushita Electric Industrial Co., Ltd.
Kadoma-shi, Osaka 571-0050 (JP)

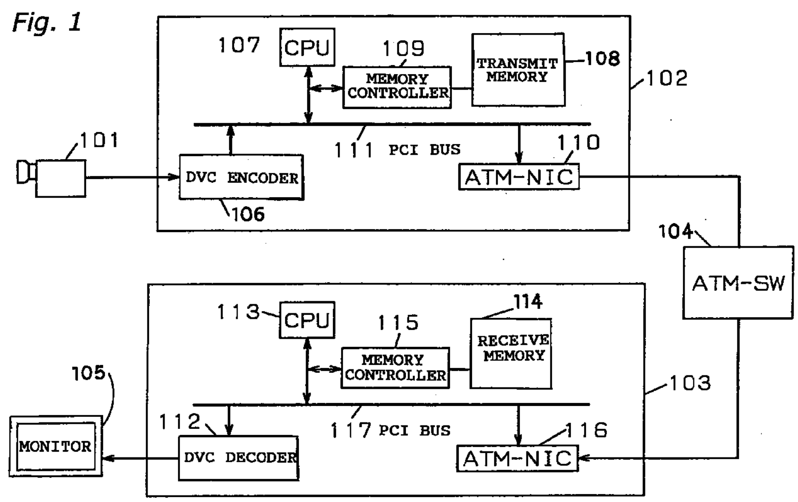
(72) Inventors:
- **Ohyama, Satoshi**
  Ashiya-shi, Hyogo 659-0033 (JP)
- **Higashida, Masaaki**
  Moriguchi-shi, Osaka 570-0011 (JP)
- **Morioka, Yoshihiro**
  Kashiba-shi, Nara 639-0261 (JP)

(74) Representative:
Grünecker, Kinkeldey,
Stockmair & Schwanhäusser
Anwaltssozietät
Maximilianstrasse 58
80538 München (DE)

(54) Communications system

(57) A communications system has a real time signal generating part for generating the real time signal of data rate R with one frame as one unit; a transmission part, having a transmission rate C (C > R), over which the real time signal is transmitted; and a real time signal outputting part for outputting the real time signal stored in a real time signal receive buffer part, wherein after storing the real time signal for a prescribed number, N, of frames in the real time signal receive buffer part, the real time signal outputting part is activated, thereby performing control so that all generated portions of the real time signal can be output at the receiving end without interruption.

Fig. 1



EP 0 859 535 A2

Description

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to an apparatus for transmitting and receiving image data, voice data, and other supplementary data over a network, and more particularly to a communications system capable of maintaining the real time requirement of such data.

2. Description of the Related Art

In recent years, with increasing performance of microcomputers and the advent of OSs equipped with networking capabilities, coupled with increasing speeds of peripheral interfaces and increasing performance of computers, dramatic changes have been occurring in the field of computers, particularly personal computers and work stations, these changes entailing changes in information that computers handle. Earlier, information handled by computers was character code, such as ASCII and JIS, but gradually expanded to include graphics and the like in the field of CAD, and nowadays, handling of multimedia information such as moving images and voice is increasing in importance. The most notable feature of multimedia information is that information occurs continuously in real time (hereinafter referred to as the real time requirement). On the other hand, with the spread of high-speed wide area networks, networks are being commercially implemented that store and manage such multimedia data and that have fast data transfer rates that enable such data to be launched into the networks. What characterizes such networks is that when transmitting signals having the real time requirement over the network, the transmitting end must send out data not later than the time expected at the receiving end.

One example of a network suitable for multimedia communications is the ATM (Asynchronous Transfer Mode) network. With ATM, transmission systems capable of 156 Mbit/s, for example, have been commercially implemented. ATM specifications are being discussed and standardized by the ITU-T (International Telecommunications Union-Telecommunication Standardization Sector), the ATM Forum, etc., and many related books have been published. Besides ATM, there are other techniques, such as 100-Mbps Ethernet (100BASE-T) and Fiber Distributed Data interface (FDDI), that can provide fast transmission capabilities of 100 Mbit/s or higher and can achieve multimedia information communications. With slower versions of Ethernet (IEEE 802.2, IEEE 802.3) also, since switching hubs are now readily available, each terminal connected to the switching hub is capable of a transmission rate of about 10 Mbit/s and is therefore able to transmit real time signals that do not require data rates higher than that. In the Internet also, signals having real time requirements, such as video-conferencing and Internet telephone signals, are being transmitted.

Prior art examples of communications systems for real time signals are disclosed, for example, in Japanese Patent Unexamined Publication Nos. 7-170502, 7-170503, 7-170290, 7-170291, 7-170504, and 7-170292.

Japanese Patent Unexamined Publication No. 7-170502 discloses a system in which, when the amount of video or voice data in a buffer memory at the receiving end exceeds an upper limit value or drops below a lower limit value, the buffer memory is controlled in such a manner as to discard portions of the video or voice data other than critical portions thereof.

Japanese Patent Unexamined Publication No. 7-170503 concerns a system in which, when the amount of video or voice data in the buffer memory at the receiving end exceeds an upper limit value or drops below a lower limit value, the clock rate for buffer read is adjusted.

Japanese Patent Unexamined Publication No. 7-170290 describes a system in which, when the amount of voice data in the buffer memory at the receiving end exceeds an upper limit value or drops below a lower limit value, a control signal is sent to alert the transmitting end, and at the transmitting end the clock rate for memory read is adjusted.

Japanese Patent Unexamined Publication No. 7-170291 proposes a system in which, when the amount of voice data in a buffer memory at the transmitting end exceeds an upper limit value or drops below a lower limit value, video and voice data in the buffer memory are discarded.

Japanese Patent Unexamined Publication No. 7-170504 provides a system in which a frame buffer is placed in front of the buffer memory at the transmitting end and, when the amount of data in the buffer memory exceeds an upper limit value or drops below a lower limit value, the amount of data is adjusted by adjusting the clock rate of the frame buffer memory.

Japanese Patent Unexamined Publication No. 7-170292 discloses a system in which, when the amount of voice data in the buffer memory at the transmitting end exceeds an upper limit value or drops below a lower limit value, the clock rate for memory read is adjusted, and when the amount of voice data in the buffer memory at the transmitting end exceeds an upper limit value or drops below a lower limit value, the amount of communication is controlled by changing the compression ratio.

However, the prior art systems disclosed in Japanese Patent Unexamined Publication Nos. 7-170502, 7-170503, 7-170290, 7-170291, 7-170504, and 7-170292 have had the problem that the system size increases because of the provision of extra circuitry such as a circuit for measuring the amount of buffer contents, a circuit for manipulating the discarding of buffer contents, a circuit for judging mathematical operations, and a circuit for specifying compression parameters necessary for obtaining the desired image compression ratio.

Furthermore, since the degree of network congestion varies over time, there can occur cases where data cannot be handled at one or more exchange points, etc. resulting in overflow. Generally, when a network falls into such a condition, the amount of transmission is reduced to alleviate the condition. The above prior art systems have had the problem that, when the transmission capacity is restricted, it becomes impossible to maintain the real time requirement of real time signals or image quality degrades more than necessary.

The present invention has been devised to overcome the above-outlined problems, and it is an object of the invention to provide a communications system that is simple in configuration and that, when transmitting digital data having a real time requirement such as a moving image (video) signal or voice signal, ensures transmission of all frames without compromising the real time requirement of the data.

SUMMARY OF THE INVENTION

To resolve the above problems, the present invention provides a communications system for communication of a real time signal having a real time requirement, comprising:

real time signal generating means for generating the real time signal of data rate R;
real time signal transmit buffer means for temporarily storing the real time signal;
real time signal transmitting means for transmitting the real time signal;
transmission means, having a transmission rate C (C > R), over which the real time signal is transmitted;
real time signal receiving means for receiving the real time signal transmitted from the real time signal transmitting means over the transmission means;
real time signal receive buffer means for temporarily storing the real time signal received by the real time signal receiving means; and
real time signal outputting means for outputting the real time signal stored in the real time signal receive buffer means.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a diagram showing the configuration of a communications system according to a first embodiment of the present invention;
Figure 2 is a flow chart illustrating a control program in a transmitting terminal 102 according to the first embodiment of the invention;
Figure 3 is a flow chart illustrating a control program in a receiving terminal 103 according to the first embodiment of the invention;
Figure 4 is a conceptual diagram showing a frame data buffer used in the first embodiment of the invention;
Figure 5 is a timing chart showing the operation in a normal condition according to the first embodiment of the invention;
Figure 6 is a timing chart showing the operation in a congested condition according to the first embodiment of the invention;
Figure 7 is a diagram showing a TCP segment format;
Figure 8 is a diagram showing the configuration of a communications system according to a second embodiment of the present invention;
Figure 9 is a diagram showing the configuration of a communications system according to a third embodiment of the present invention;
Figure 10 is a flow chart illustrating a control program in a transmitting terminal 102 according to the third embodiment of the invention;
Figure 11 is a flow chart illustrating a control program in a receiving terminal 103 according to the third embodiment of the invention; and
Figure 12 is a diagram showing the configuration of a communications system according to a fourth embodiment of the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENTS

The preferred embodiments of the present invention will be described below with reference to the accompanying drawings.

The real time signal contemplated by the present invention can be applied to any signal having a real time requirement, such as a moving image signal or a voice signal, but in the embodiments of the present invention, a digitized moving image signal is taken as an example of the real time signal, unless otherwise stated.

Further, the present invention is applicable regardless of the quality of the transmitted moving image which is determined by the number of samples in the horizontal and vertical directions relating to the resolution of the moving image or by whether compression is applied or not and, if compression is applied, then by the type of compression applied. That is, the present invention is applicable regardless of the bit rate of the transmitted real time signal.

In the embodiments of the present invention, an image signal employing the sampling method and compression method used in VTRs manufactured to the DVC standard is taken as an example.

DVC is an acronym for Digital Video Cassette, and the DVC standard was agreed upon by the HD Digital VCR Consortium. This standard is described, for example, in a magazine "National Technical Report", Vol. 41, No. 2, April 1995, pp. 152-159. In the embodiments of the present invention, the NTSC signal compliant with the DVC standard is taken as an example. According to the method defined by the DVC standard (NTSC signal), the data bit rate is 28.8 Mbit/s (hereinafter described as 28.8 Mbps).

Any transmission means may be used as long as it has a transmission capacity greater than the bit rate of the real time signal used, but in the embodiments of the present invention, an ATM network having a transmission rate of 156 Mbit/s is used as the transmission means; further, as a network protocol, a scheme called IP over ATM is used that is defined in RFC 1577 and that realizes a virtual IP network, and as higher layer protocols, TCP/IP (Transmission Control Protocol/Internet Protocol), now the standard suite of protocols for the Internet, is used in the examples described herein.

Generally, the lowest level in a communications network provides unreliable packet delivery. If a transmission error occurs that affects the data, or if the network load becomes too heavy to be handled properly, packets may be lost or destroyed. In a network that dynamically routes packets, the packets may be delivered out of order, may be duplicated, or may even arrive after a large delay. Therefore, at the highest level, programmers are required to incorporate error detection and recovery into application programs to achieve transmission of large volumes of data.

TCP/IP uses the basic technique called the positive acknowledgement with retransmission to provide reliable transmission. This technique requires that the receiver communicate with the sender and return an acknowledgement (ACK) message each time data is received. The sender maintains records of each transmitted packet, and sends the next packet after an acknowledgement arrives. The sender also activates a timer when sending a packet and, if the timer has timed out before arrival of an acknowledgment, retransmits the packet.

The transfer unit used by TCP is called a segment. Figure 7(a) illustrates the TCP segment format. Each segment is divided into two areas, the header and the data area. The header is known as the TCP header and carries expected identification and control information. The SOURCE PORT field and the DESTINATION PORT field contain the TCP port numbers identifying application programs at both ends of the connection. The SEQUENCE NUMBER field identifies the position in a byte stream of the sender's data contained in the segment. The ACKNOWLEDGEMENT NUMBER field identifies the beginning of the sequence number expected to be received next (the next octet to be received). Since the TCP header length varies depending on the options selected, the HLEN field indicates the offset of the data area in the segment. The six-bit field labeled RESERVED is set aside for future use. The TCP segment may be used to carry a connection setup or connection clear request or to carry only an acknowledgement or data. The six-bit field labeled CODE BITS is used to determine the purpose and contents of the segment. More specifically, the CODE BITS field dictates the way how the other fields in the header are to be interpreted in accordance with the table shown in Figure 7(b). The WINDOW field tells the amount of data the receiving end is willing to accept, by specifying the buffer size each time a segment is sent. Data that needs immediate processing is called urgent data. If the urgent code bit is on, that means that this segment contains urgent data. The URGENT POINTER field carries a pointer that indicates the position where the urgent data ends. The CHECKSUM field in the TCP header contains an integer checksum used to verify the integrity of the TCP header and the data. The OPTIONS field is used for expanded TCP specifications, for example, but it is usually used to notify the remote end of the maximum receivable segment size.

As previously described, TCP/IP uses the technique called the positive acknowledgement; however, in a simple positive acknowledgement protocol, since transmission of the next packet has to wait until a positive acknowledgement for the previous packet is received, the available bandwidth of the network is wasted. To address two important problems of efficient transfer and flow control, TCP uses a special mechanism called a sliding window. In this method, a fixed-length window is set in the sequence, and all packets inside the window are transmitted at a time. For example, in a sliding window protocol of window size 8, the sender is allowed to transmit eight packets before an acknowledgement is received. After the sender receives an acknowledgement for the first packet, the sender slides the window and sends another packet. The window continues to slide as long as an acknowledgement is received. In this case, the number of packets that may be remaining unacknowledged at any given time is limited by the window size, and is restricted to a small predetermined value. Further, when the remaining capacity of the receiving buffer becomes low, for example, the amount of data being transmitted from the remote end can be reduced by reducing the window size value. TCP performs flow control by using this value.

There are two major causes for the loss of a TCP segment transferred over a network. One is a sporadic error. This, however, is not common in LANs and WANs where line quality is good. The other is the existence of a bottleneck such as a low-speed line or low-processing capacity router in the path through which the TCP segment is carried. A condition in which a critical delay is caused by overload with datagrams at one or more exchange points such as routers is called congestion. Once congestion occurs, the delay increases and the router begins to queue datagrams until they can be routed out. In the worst case, the total number of datagrams arriving at the congested router reaches the limit of the router's capacity and the router begins to drop datagrams. Usually, an end point has no detailed idea of where and how the congestion has occurred, and the congestion simply manifests itself as an increased delay

IPR2022-01227
EXHIBIT 1003 - PAGE 00709

at the end point. The increased delay, in turn, causes the transport protocol to request retransmissions due to timeouts, further increasing the traffic on the network and exacerbating the congestion. This condition is called congestion collapse.

To avoid congestion collapse, in TCP it is recommended to use two techniques, slow start and multiplicative decrease congestion avoidance.

In multiplicative decrease congestion avoidance, a second restriction called a congestion window is provided to control congestion, and anytime, TCP compares the receiver's window size (buffer size) with the congestion window size and uses the smaller window in transmission. Each time a segment loss is detected, the congestion window size is reduced by one half (the minimum value is 1 segment) and the timeout interval is doubled. This strategy provides a quick and significant decrease in traffic and allows a sufficient time to clear the datagrams already queued at the router.

When it is determined that the congestion has ended, TCP initializes the congestion window to 1, sends the first segment, and waits for an acknowledgement. Thereafter, the congestion window is increased by one segment each time an acknowledgement arrives. This technique is called slow start.

To prevent the window size from increasing too rapidly and thereby causing congestion again, TCP provides a still another restriction. That is, when the congestion window size reaches one half of its original size, TCP enters a congestion avoidance stage and reduces the increasing speed of the congestion window size. During the congestion avoidance period, the congestion window is increased only by 1 even if the acknowledgements for all the segments in the window are received.

The above has described the process in TCP from the occurrence of congestion to the recovery from the congestion. While this process is in progress, the network can only transmit data at a lower transmission rate than its actual transmission capacity and imposes extra loads on end points. This presents a problem, especially when transmitting real time signals. That is, the real time signals cannot be delivered in time and the real time requirement is impaired.

In the present invention, the number of frames of a real time signal accumulated in advance at the receiving end can be applied to any signals having real time requirements. In the embodiments of the present invention, description will be given by taking a digitized moving image signal as an example, unless otherwise stated.

(Embodiment 1)

Figure 1 shows a system configuration according to a first embodiment of the present invention. In Figure 1, reference numeral 101 is a camera for capturing a moving image, 102 is a transmitting terminal, 103 is a receiving terminal, 104 is an ATM switch, and 105 is a monitor.

In the transmitting terminal 102, reference numeral 106 is a DVC encoder for applying DVC compression, 107 is a CPU, 108 is a transmit memory for temporarily storing transmit data and a control program for the transmitting terminal 102, 109 is a memory controller for controlling the transmit memory 108, 110 is an ATM network interface card (ATM-NIC), and 111 is a PCI bus.

In the receiving terminal 103, reference numeral 112 is a DVC decoder, 113 is a CPU, 114 is a receive memory for temporarily storing received data and a control program for the receiving terminal 103, 115 is a memory controller for controlling the receive memory 114, 116 is an ATM network interface card (ATM-NIC), and 117 is a PCI bus.

The operation of the thus configured communications system will be described below.

First, in the transmitting terminal 102, the moving image signal from the camera 101 is input into the DVC encoder 106 where DVC compression is applied on a frame by frame basis. Since the NTSC signal consists of 30 frames per second, data for 30 frames per second is obtained in the embodiment of the present invention. The DVC data output from the DVC encoder is stored frame by frame in the transmit memory 108 via the PCI bus under the control of the memory controller 109. Then, the DVC data stored in the transmit memory 108 is transferred to the ATM-NIC 110 via the PCI bus.

With the above-described processing in the transmitting terminal 102, the moving image data from the ATM-NIC 110 is transmitted over the ATM network to the receiving terminal 103 via the ATM switch 104.

Next, processing at the receiving terminal 103 will be described.

First, at the receiving terminal, the data received by the ATM-NIC is stored in the receive memory 114 via the PCI bus. The DVC data stored in the receive memory 114 is then transferred frame by frame to the DVC decoder 112, where the compressed image data is decoded into a decompressed image signal, which is output to the monitor 105.

In Figure 1, the DVC encoder 106 and the DVC decoder 112 operate with clocks of the same frequency.

Next, the processing at the transmitting terminal 102 described with reference to Figure 1 will be explained from the standpoint of the control program with reference to Figure 2.

Figure 2 is a flow chart of the control program of the transmitting terminal 102. This program is stored in the memory 108 and executed by the CPU 107.

The control program of the transmitting terminal 102 executes two threads, an input thread and a transmit thread, in time division fashion to accomplish the transmitting process. At this time, a transmit buffer is used to temporarily store the DVC data, and the transmit buffer is accessed in accordance with an input pointer pointing to the storage location (address) where data is to be stored and a transmit pointer pointing to the storage location (address) where data to be transmitted is stored. The transmit buffer is implemented, for example, as part of the memory 108.

The thread here is the most basic unit in terms of which the operating system assigns the processor's execution time.

Previously, many multitasking operating systems such as UNIX employed a process as a basic unit, but in recent UNIX and Windows NT, multiple threads can be allocated to one process, and most of the functions implemented by multiprocessing can now be implemented by multithreading. In previous systems using multiprocessing, information on all resources, including virtual memory space, is stored temporarily, and interprocess communication is necessary in order to share the information between processes; here, the performance of the interprocess communication becomes a bottleneck that limits the overall performance of the system. In the case of multithreading, on the other hand, since many memory spaces including virtual memory space are shared, this kind of bottleneck can be avoided.

Figure 4(a) shows a conceptual diagram of the transmit buffer. The buffer has a so-called queue structure. SB1 to SB5 are buffer sections where data are actually stored, and IP indicates the input pointer and SP the transmit pointer. The initial values of the input pointer and transmit pointer are both SB1. The input pointer IP is updated each time data for one frame is read from the DVC encoder 106 and stored in a buffer section. On the other hand, the transmit pointer SP is updated each time data in a buffer section is transmitted out onto the network. More specifically, the input pointer and the transmit pointer both move from one buffer section to the next as they are updated, that is, from SB1 to SB2, then from SB2 to SB3, and returning to SB1 after SB5.

In the present invention, the transmit buffer can be of any size as long as it is larger than the receive buffer. In the example shown here, the transmit buffer size is set to five frames.

The actual transmitting process will be described below with reference to Figure 2. In the input thread, processing is performed in 201, such as establishing a logical connection with the receiving terminal 103 and securing and initializing the transmit buffer. Next, in 202, the transmit thread is created. Then, in 203, frame data generated by the DVC encoder 106 is written to the transmit buffer at the address indicated by the input pointer. In 204, the input pointer is updated to point to the transmit buffer address at which the next data is to be stored. Conditional branches are provided at 205 where it is determined whether the transmitting process is to be continued or not. If it is to be continued, the process from 203 onward is repeated. If it is determined in 205 that the transmitting process is not to be continued, a transmit thread termination notification is generated in 206 to cause the transmit thread to terminate, and at the same time, the execution of the input thread is terminated in 207.

The transmit thread is created in step 202 in the input thread. First, in 208, processing such as initialization of the transmit pointer is performed. If it is determined in 209 that there is frame data at the transmit buffer address indicated by the transmit pointer, then the data is transmitted out in 210. After the data is transmitted out, the contents of the buffer indicated by the transmit pointer are cleared in 211, and in 212 the transmit pointer is updated to point to the transmit buffer address at which the next data to be transmitted is stored. Conditional branches are provided at 213 where it is determined whether the transmit thread is to be continued or not. If it is to be continued, the process from 209 onward is repeated. On the other hand, if the transmit thread termination notification 216 is received, the execution of the thread is stopped, and the transmit thread is terminated in 214.

Next, the processing at the receiving terminal 103 described with reference to Figure 1 will be explained from the standpoint of the control program with reference to Figure 3.

Figure 3 is a flow chart of the control program of the receiving terminal 103. This program is stored in the memory 114 and executed by the CPU 113. The control program of the receiving terminal 103 executes two threads, a receive thread and an output thread, in time division fashion to accomplish the receiving process. At this time, a receive buffer is used to temporarily store the DVC data, and the receive buffer is accessed in accordance with a receive pointer pointing to the storage location (address) where the received data is to be stored and an output pointer pointing to the storage location (address) where data to be output is stored. The receive buffer is implemented, for example, as part of the memory 114.

Figure 4(b) shows a conceptual diagram of the receive buffer. This buffer has the same structure as the previously described transmit buffer structure. RB1 to RB5 are buffer sections where data are actually stored, and RP indicates the receive pointer and OP the output pointer. The initial values of the receive pointer RP and output pointer OP are both RB1. The receive pointer RP is updated each time data is received from the network and stored in a buffer section, and the output pointer OP is updated each time data in a buffer section is output to the DVC decoder 112. More specifically, the receive pointer and the output pointer both move from one buffer section to the next as they are updated, that is, from RB1 to RB2, then from RB2 to RB3, and returning to RB1 after RB5.

In the present invention, the receive buffer size and the number of frames of a real time to be accumulated in advance are arbitrary within the range that does not impair the real time requirement of the signal. In the example shown here, the receive buffer size is set to five frames and the number of frames to be accumulated in advance is set to four frames.

In the receive thread, processing is performed in 301, such as establishing a logical connection with the transmitting terminal 102 and securing and initializing the receive buffer. Next, in 302, frame data is received, the frame data is written to the receive buffer at the address indicated by the receive pointer, and then the receive pointer is updated; this sequence of processing is repeated for four successive frames. Next, the output thread is created in 303. If, in 304, there is frame data to be received, then in 305 the data is received and written to the receive buffer at the address indicated by the receive pointer. Then, the receive pointer is updated in 306.

Conditional branches are provided at 307 where it is determined whether the receiving process is to be continued or not. If it is to be continued, the process from 304 onward is repeated. On the other hand, if it is determined in 307 that the receiving process is not to be continued, an output thread termination notification is generated in 308 to cause the output thread to terminate, and at the same time, the execution of the receive thread is terminated in 309.

The output thread is created in step 303 in the receive thread. First, in 310, processing such as initialization of the output pointer is performed. Next, in 311, frame data at the address indicated by the output pointer is transferred from the receive buffer to the DVC decoder 112 shown in Figure 1. The transferred frame data is decoded and output to the monitor 105 shown in Figure 1. Then, in 312, the contents of the receive buffer indicated by the receive pointer are cleared, and in 313 the output pointer is updated to point to the receive buffer address at which the next data to be output is stored. Conditional branches are provided at 314 where it is determined whether the output thread is to be continued or not. If it is to be continued, the process from 311 onward is repeated. On the other hand, if the output thread termination notification is received, the execution of the thread is stopped, and the output thread is terminated in 315.

Figures 5 and 6 are timing charts according to the embodiment of the present invention. Figure 5 is a timing chart in a normal operating condition, and Figure 6 is a timing chart when the transmission capacity of the network is temporarily restricted due to congestion, etc. In Figures 5 and 6, 501 and 601 indicate the timing for reading data from the DVC encoder at the transmitting terminal, 502 and 602 the timing for transmission to the network, 503 and 603 the state of the transmit buffer, 504 and 604 the timing at the receiving terminal for receiving data from the network, 505 and 605 the timing for outputting frame data to the DVC decoder, and 506 and 606 the state of the receive buffer. The hatched portions in Figures 5 and 6 are data invalid portions where actual real time signal data is not handled. Data valid portions are each shown without breaks, but in actual communications, data is transmitted in an intermittent manner; accordingly, the valid portions in Figures 5 and 6 mean portions during the period of which valid data is transmitted.

Here, since transmission delays, fluctuations, etc. do not affect the essential characteristics of the present invention, for simplicity the timing charts assume conditions of a constant delay and no fluctuations.

First, the embodiment of the present invention will be described with reference to the timing chart of Figure 1 illustrating the normal operating condition.

Symbols FA0 to FA10 in the figure designate frame data being handled at the respective times, the same symbol indicating the same frame data.

Reference numeral 501 shows the timing for reading real time signal data from the DVC encoder 106 at the transmitting terminal 102. As shown by 501, the data is read into the transmit buffer precisely at intervals of one frame period T1 (1000/30 = 33.3 [msec]). T2 denotes a valid data period during which valid data is read from the DVC encoder 106, and T3 an invalid data period. T2 and T3 are both constant values such that $T2 + T3 = T1$. That is, in practice there is no need to read data into the transmit buffer during the period T3.

Reference numeral 502 shows the transmit timing for the real time signal data transmitted out onto the network by the ATM-NIC 110 at the transmitting terminal 102. As shown by 502, T4 is a valid data portion for the transmit data and T5 is an invalid portion ($T4 + T5 = T1$). The ratio of T4 to T5 is determined by the bit rate of the transmit data and the transmission capacity of the network including the NIC. As is apparent, transmission of real time signals requires that the transmission capacity of the network be greater than the bit rate of the data actually transmitted in the normal operating condition. This means that valid data is transmitted in a time interval shorter than one frame time interval ($T4 < T1$).

Shown at 503 is a schematic diagram illustrating the state of the transmit buffer at the transmitting terminal 102, starting from the time at the left edge of the diagram. S1 to S6 designate the buffer sections, and FA4 to FA9 in the diagram indicate the frame data stored in the buffer at the respective times and the position of that data. Further, arrows IP1 to IP6 in the diagram indicate the position of the input pointer at the respective times, and likewise, SP1 to SP6 indicate the position of the transmit pointer.

Reference numeral 504 shows the receive timing for the real time signal data that the ATM-NIC 116 at the receiving terminal 103 receives from the network. Since it is assumed that the transmission delay is constant, and no fluctuations are considered, the valid data and invalid data periods are the same in length as T4 and T5, respectively, in the transmit timing. T6 indicates the transmission delay with respect to the transmit timing.

Reference numeral 505 shows the timing for outputting the real time signal data to the DVC decoder 112 at the receiving terminal 103. As shown by 505, the data is read from the receive buffer precisely at intervals of one frame period T1. As in the read timing at the transmitting terminal 102, T2 denotes a valid data period during which valid data is transferred to the DVC decoder 112, and T3 an invalid data period, T2 and T3 both being constant values. That is, in practice there is no need to transfer data to the DVC decoder 112 during the period T3.

Shown at 506 is a schematic diagram illustrating the state of the receive buffer at the receiving terminal, starting from the time at the left edge of the diagram. R1 to R6 designate the buffer sections, and FA0 to FA8 in the diagram indicate the data stored in the buffer at the respective times and the position of that data. Further, arrows RP1 to RP6 in the diagram indicate the position of the receive pointer at the respective times, and likewise, OP1 to OP6 indicate the position of the output pointer.

Next, the condition where the transmission capacity of the network is temporarily restricted due to congestion, etc. will be described with reference to the timing chart shown in Figure 6. Symbols FB0 to FB10 in the figure designate frame data being handled at the respective times, the same symbol indicating the same frame data.

Reference numeral 601 shows the timing for reading real time signal data from the DVC encoder 106 at the transmitting terminal 102. As in the case of Figure 5, the data is read into the transmit buffer precisely at intervals of one frame period T1. T2 denotes a valid data period during which valid data is read from the DVC encoder 106, and T3 an invalid data period, T2 and T3 both being constant values.

Reference numeral 602 shows the transmit timing for the real time signal data transmitted out onto the network by the ATM-NIC 110 at the transmitting terminal 102. Unlike the case of Figure 5, in Figure 6 time T7 (T7 > T1) is required for the transmission of FB4 because the transmission capacity of the network is restricted during congestion. When the congested condition is cleared, the transmission capacity gradually recovers as a result of the previously described congestion control, so that transmission of the next frame data FB5 takes time T8 (T7 > T8 > T1 > T4). Starting with the transmission of frame data FB6, the network returns to the normal condition and each frame is transmitted in time T4. Further, since data continues to be transmitted as long as there is data in the transmit buffer indicated by the transmit pointer, valid data transmission continues without interposing data invalid portions, unlike the normal operating condition.

Shown at 603 is a schematic diagram illustrating the state of the transmit buffer at the transmit terminal 102, starting from the time at the left edge of the diagram. S7 to S12 designate the buffer sections, and FB4 to FB9 in the diagram indicate the data stored in the buffer at the respective times and the position of that data. Further, arrows IP7 to IP12 in the diagram indicate the position of the input pointer at the respective times, and likewise, SP7 to SP12 indicate the position of the transmit pointer. S7 to S12 in the diagram correspond to S1 to S6 in Figure 5. In S1 to S6, the frame data accumulated in the buffer at any given time is only one that is stored in the buffer section indicate by the transmit pointer; on the other hand, in S7 to S12, since the next frame data is read into the buffer before the current frame data is sent out, data for multiple frames are stored, starting with S8.

Reference numeral 604 shows the receive timing for the real time signal data that the ATM-NIC 116 at the receiving terminal 103 receives from the network. Since it is assumed that the transmission delay is constant, and no fluctuations are considered, the data receiving time is the same as that in the transmit timing. T6 indicates the transmission delay with respect to the transmit timing.

Reference numeral 605 shows the timing for outputting the real time signal data to the DVC decoder 112 at the receiving terminal 103. As in the case of Figure 5, the data is read from the receive buffer precisely at intervals of one frame period T1. As in the read timing at the transmitting terminal 102, T2 denotes a valid data period during which valid data is transferred to the DVC decoder 112, and T3 an invalid data period, T2 and T3 both being constant values.

Shown at 606 is a schematic diagram illustrating the state of the receive buffer at the receiving terminal 103, starting from the time at the left edge of the diagram. R7 to R12 designate the buffer sections, and FB0 to FB8 in the diagram indicate the data stored in the buffer at the respective times and the position of that data. Further, arrows RP7 to RP12 in the diagram indicate the position of the receive pointer at the respective times, and likewise, OP7 to OP12 indicate the position of the output pointer. R7 to R12 in the diagram correspond to R1 to R6 in Figure 5. In R1 to R6, frame data for four frames are accumulated, from the buffer section indicated by the output pointer to the buffer section indicated by the receive pointer, at any given time, and the number of accumulated frames is thus constant. On the other hand, in R7 to R12, since frame data is output before the next frame data is received, the number of accumulated frames at each instant in time begins to decrease at R8 and, when recovered from the congestion at R12, begins to increase and continues to increase until four frames are accumulated, thus returning to the normal operating condition.

The important point of the present invention is that even when the transmission capacity is temporarily restricted due to congestion, etc. rendering it impossible to transmit a real time signal within the real time, since data are accumulated in advance at the receiving terminal the real time signal can be output without interruption as long as the accumulated data are output, and when recovered from the congestion, the valid data accumulated in the transmit buffer are transmitted successively until data for a prescribed number of frames are accumulated in the receive buffer as in the previous normal operating condition.

(Embodiment 2)

Figure 8 is a block diagram showing a second embodiment of the present invention. In the figure, reference numeral 801 is a master tape on which a DVC-compressed digital signal is prerecorded, 802 is a DVC player for playing back the master tape 801 and for outputting the prerecorded digital signal, 803 is a DVC interface (DVC-I/F) for transferring the digital signal output from the DVC deck 802 into the memory 108 via the PCI bus 111, 804 is a DVC-IF for outputting via the PCI bus 117 the DVC data stored in the memory 114 at the receiving terminal, 805 is a subtape for recording the received DVC data, and 806 is a DVC recorder for recording the DVC data on the subtape 805. The same constituent elements as those in Figure 1 are designated by the same reference numerals. In Figure 8, the DVC player 802 and the DVC recorder 806 operate with clocks of the same frequency.

According to the above configuration, between remote terminals interconnected over an ATM network, all frame data on the master tape connected to the transmitting terminal can be copied onto the subtape connected to the receiving terminal without impairing the real time requirement.

(Embodiment 3)

Figure 9 is a block diagram showing a third embodiment of the present invention. In the figure, reference numeral 901 is a disk medium on which DVC-compressed data is prerecorded. Not only can the disk medium 901 be read faster than the real time of the recorded DVC data, but it also permits random access to the recorded data. Otherwise, the configuration is the same as that shown in Figure 1, and the same constituent elements are designated by the same reference numerals between the two figures.

The disk medium 901 may be constructed from a semiconductor memory or the like as long as it permits random access and can be read faster than the real time of the recorded data.

Actual processing in this embodiment will be described with reference to Figures 10 and 11.

Figure 10 is a flow chart illustrating the control program of the transmitting terminal 102. The same constituent elements as those in Figure 2 are designated by the same reference numerals. The flow of this embodiment differs from that of Figure 2 in that an image for transmission is selected before transmission because data at the transmitting terminal is recorded on a randomly accessible medium, and in that each time a frame data request is received, synchronization is established with the receiving terminal 103 by sending data, since the medium can be read faster than the real time.

First, in 1001, an image selection notification is received from the receiving terminal, and in 1002, the selected image is transmitted in accordance with the instruction. The receiving terminal checks the received image, decides whether or not to determine the selection, and sends the decision notification. In 1003, an appropriate branch is chosen in accordance with the decision notification sent from the receiving terminal. When the selection is determined, the transmit thread is created, as in the case of Figure 2. Thereafter, the input thread reads the next frame data each time a frame data request notification is received in 1004. Otherwise, the process is the same in operation as that shown in Figure 2. The transmit thread is the same in operation as that shown in Figure 2.

Figure 11 is a flow chart illustrating the control program of the receiving terminal 103. The same constituent elements as those in Figure 3 are designated by the same reference numerals. The flow of this embodiment differs from that of Figure 3 in that an image for transmission is selected before transmission because data at the transmitting terminal is recorded on a randomly accessible medium, and in that each time a frame data request is received, synchronization is established with the receiving terminal 103 by sending data, since the medium can be read faster than the real time.

First, in 1101, the receiving terminal 103 sends an image selection notification, and in 1102, the image transmitted in accordance with the instruction is displayed. The receiving terminal 103 checks the received image and decides whether or not to determine the selection in 1103, and if the selection is determined, sends the decision notification in 1104. When the selection is determined, then in 1105 a moving image data request notification is sent to the transmitting terminal 102 and moving image data for a prescribed number of frames are received, and the output thread is created, as in the case of Figure 3. Thereafter, the receive thread performs the same operation as that in Figure 3. In the output thread, on the other hand, each time image data is output to the external monitor, a request notification for the next moving image data to receive is sent to the transmitting terminal in 1106. Otherwise, the process is the same in operation as that shown in Figure 3.

According to the above embodiment, moving image data is recorded on a randomly accessible disk medium capable of being read faster than the real time of the recorded DVC data, so that a moving image recorded at any location on the disk medium can be retrieved quickly, and after the selection is determined, the real time signal can be transmitted without interruption while maintaining synchronization with the transmitting terminal via the network.

IPR2022-01227
EXHIBIT 1003 - PAGE 00714

(Embodiment 4)

Figure 12 is a block diagram showing a fourth embodiment of the present invention. In the figure, reference numerals 1201, 1203, and 1205 are cameras corresponding to 101 in Figure 1, 1202, 1204, and 1206 are transmitting terminals which are identical in configuration to 102 shown in Figure 1, and 1207 is a receiving terminal which is identical in configuration to 103 shown in Figure 1.

According to this embodiment, when selecting a transmitting terminal, received data is immediately displayed on the monitor, and after determining the transmitting terminal selection, frame data for a prescribed number of frames are accumulated and then the image is displayed on the monitor; in this way, the desired transmitting terminal can be selected quickly, and once the selection is determined, the received image can be displayed without interruption.

The embodiments of the present invention have been described as using the TCP/IP transmission protocols for example, but since the essential characteristic of the present invention is that data is output without interruption even when the transmission capacity is temporarily restricted due to retransmissions or when transmitting means has a congestion control mechanism, the invention is not limited to using TCP/IP and configurations using other protocols having similar functions should not be excluded from the scope of the present invention.

Further, the embodiments of the present invention have been described as handling only moving images for example, but it will be appreciated that the invention can also be applied to configurations where both moving image and voice signals are transmitted simultaneously, by using, for example, time division multiplexing techniques.

As described above, according to the communications system of the present invention, at the transmitting terminal, processing for reading a real time signal and processing for sending it out on the network are performed by using multithreading techniques, and at the receiving terminal, real time signal data for a prescribed number of frames are accumulated first, and then processing for receiving real time signal data from the network and processing for outputting the accumulated data to the external monitor are performed by using multithreading techniques; this enables the real time signal to be output to the external monitor without interruption, even when the transmission capacity is temporarily restricted due to congestion, etc. on the network and transmission of the real time signal within the real time is rendered impossible.

In one preferred embodiment of the present invention, the real time signal transmitted from the transmitting terminal is a signal reproduced from a master tape on which the real time signal is prerecorded, and the real time signal is then recorded on a subtape at the receiving terminal; in this way, between apparatuses connected via a network it becomes possible to duplicate the tape with the real time signal recorded thereon without causing interruptions in the recorded signal.

In another preferred embodiment of the present invention, the real time signal to be transmitted from the transmitting terminal is prerecorded on a medium that is capable of being read faster than the real time of the recorded data and that permits random access to the recorded data; when searching for the location of desired data for transmission, the received data is immediately displayed on the monitor, and after the location is determined, frame data for a prescribed number of frames are accumulated first and then output for display on the monitor, each time followed by transmission of a data request notification to the transmitting terminal. In this way, a moving image recorded at any location on the medium can be retrieved quickly, and once the location is determined, the real time signal can be transmitted without interruption while maintaining synchronization with the transmitting terminal.

In a further preferred embodiment of the present invention, a plurality of transmitting terminals are provided, and one of the transmitting terminals is selected to transmit data to the receiving terminal; when selecting a transmitting terminal, received data is immediately displayed on the monitor, and after determining the transmitting terminal selection, frame data for a prescribed number of frames are accumulated first and then the image is displayed on the monitor. In this way, the desired transmitting terminal can be selected quickly, and once the selection is determined, the received image can be displayed without interruption.

Thus, in a network having a transmission capacity greater than the transmission rate of a real time signal, the present invention offers an enormous advantage in that the real time signal received from the transmitting end can be output at the receiving end without interruption even when the transmission capacity is temporarily restricted due to congestion, etc.

Claims

1. A communications system for communication of a real time signal having a real time requirement, comprising:

real time signal generating means for generating said real time signal of data rate R with one frame as one unit;
real time signal transmit buffer means for temporarily storing said real time signal;
real time signal transmitting means for transmitting said real time signal;
transmission means, having a transmission rate C (C > R), over which said real time signal is transmitted;
real time signal receiving means for receiving

IPR2022-01227
EXHIBIT 1003 - PAGE 00715

said real time signal transmitted from said real time signal transmitting means over said transmission means;

real time signal receive buffer means for temporarily storing said real time signal received by said real time signal receiving means; and

real time signal outputting means for outputting said real time signal stored in said real time signal receive buffer means, wherein

after storing said real time signal for a prescribed number, N, of frames in said real time signal receive buffer means, said real time signal outputting means is activated, thereby performing control so that all generated portions of said real time signal can be output at the receiving end without interruption.

2. A communications system according to claim 1, further comprising real time signal recording means for recording said real time on a recording medium, wherein said real time signal stored in said receive buffer means is recorded on said recording medium as said real time signal is output by said real time signal outputting means.

3. A communications system for communication of a real time signal having a real time requirement, comprising:

real time signal reproducing means for reproducing a real time signal of data rate R in real time with one frame as one unit by playing back a first information recording medium on which said real time signal is prerecorded;

real time signal transmit buffer means for temporarily storing said real time signal;

real time signal transmitting means for transmitting said real time signal;

transmission means, having a transmission rate C (C > R), over which said real time signal is transmitted;

real time signal receiving means for receiving said real time signal transmitted from said real time signal transmitting means over said transmission means;

real time signal receive buffer means for temporarily storing said real time signal received by said real time signal receiving means; and

real time signal recording means for recording in real time said real time signal stored in said real time signal receive buffer means onto a second information recording medium, wherein

after storing said real time signal for a prescribed number, N, of frames in said real time signal receive buffer means, said real time signal recording means is activated, thereby performing control so that all reproduced portions of said real time signal can be recorded on said second information recording medium without interruption.

4. A communications system according to claim 3, further comprising: real time signal reproduction control means for externally controlling reproduction start timing of said real time signal reproducing means; and real time signal recording control means for controlling recording start timing of said real time signal recording means, wherein control is performed so that all or part of said real time signal prerecorded on said first information recording medium can be recorded on said second information recording means.

5. A communications system comprising:

a real time signal recording medium on which real time signal data having a real time requirement is prerecorded on a frame by frame basis;

real time signal reproducing means capable of reading said real time signal prerecorded on said real time signal recording medium faster than real time, and capable of randomly accessing said real time signal;

real time signal transmit buffer means for temporarily storing said real time signal;

real time signal transmitting means for transmitting said real time signal;

transmission means over which said real time signal is transmitted;

real time signal receiving means for receiving said real time signal transmitted from said real time signal transmitting means over said transmission means;

real time signal receive buffer means for temporarily storing said real time signal received by said real time signal receiving means; and

real time signal outputting means for outputting said real time signal stored in said real time signal receive buffer means, wherein

after storing said real time signal for a prescribed number, N, of frames in said real time signal receive buffer means, said real time signal outputting means is activated, and

each time said real time signal outputting means outputs said real time signal of a prescribed unit, a receive notification is sent to said real time signal transmitting means, and if said receive notification does not reach said real time signal transmitting means within a predetermined time, said real time signal read by said real time signal reproducing means is added to said real time signal transmit buffer means, thereby performing control so that all portions of said real time signal generated from said real time signal transmitting means can be output at the receiving side without interruption.

6. A communications system for communication of a real time signal having a real time requirement, comprising:

a plurality of real time signal generating means, each for generating said real time signal of data rate R with one frame as one unit;
a plurality of real time signal transmitting means respectively connected to said plurality of real time signal generating means;
a plurality of real time signal transmit buffer means for temporarily storing said real time signals respectively generated by said plurality of real time signal generating means;
a plurality of real time signal transmitting means respectively connected to said plurality of real time signal transmit buffer means, for transmitting said real time signals;
transmission means, having a transmission rate C (C > R), over which said real time signals are transmitted;
selecting means for selecting one of said plurality of real time signal transmitting means;
real time signal receiving means for receiving said real time signal transmitted from said real time signal transmitting means selected by said selecting means;
real time signal receive buffer means for temporarily storing said real time signal received by said real time signal receiving means; and
real time signal outputting means for outputting said real time signal stored in said real time signal receive buffer means, wherein
when selecting said real time signal to be received by said real time signal receiving means, the real time signal to be received is output sequentially from said real time signal transmitting means, and once the selection is determined, after storing said real time signal for a prescribed number, N, of frames in said real time signal receive buffer means said real time signal outputting means is activated, thereby performing control so that, after the selection is determined, all generated portions of said real time signal can be output at the receiving end without interruption.

Fig. 1

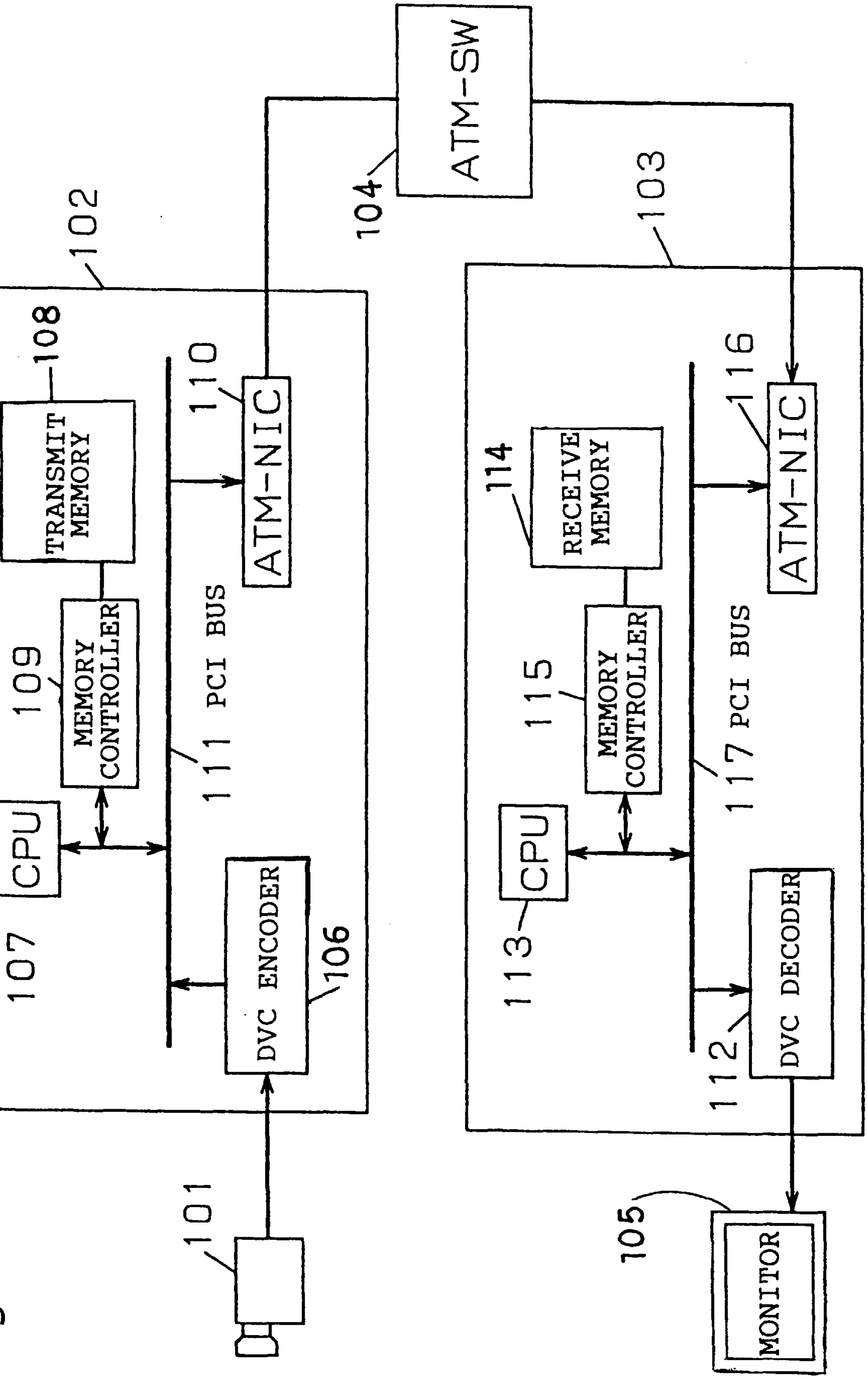
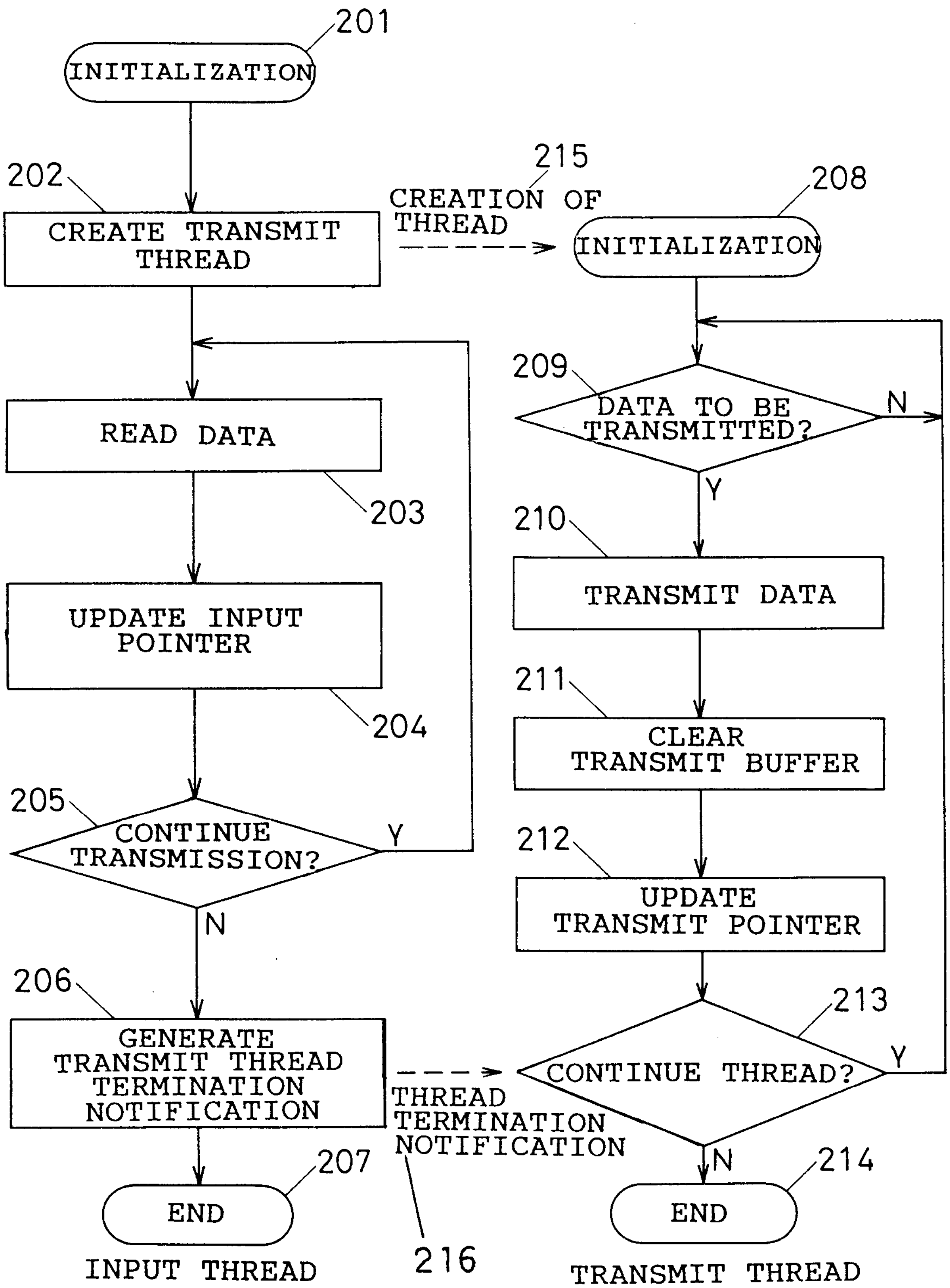
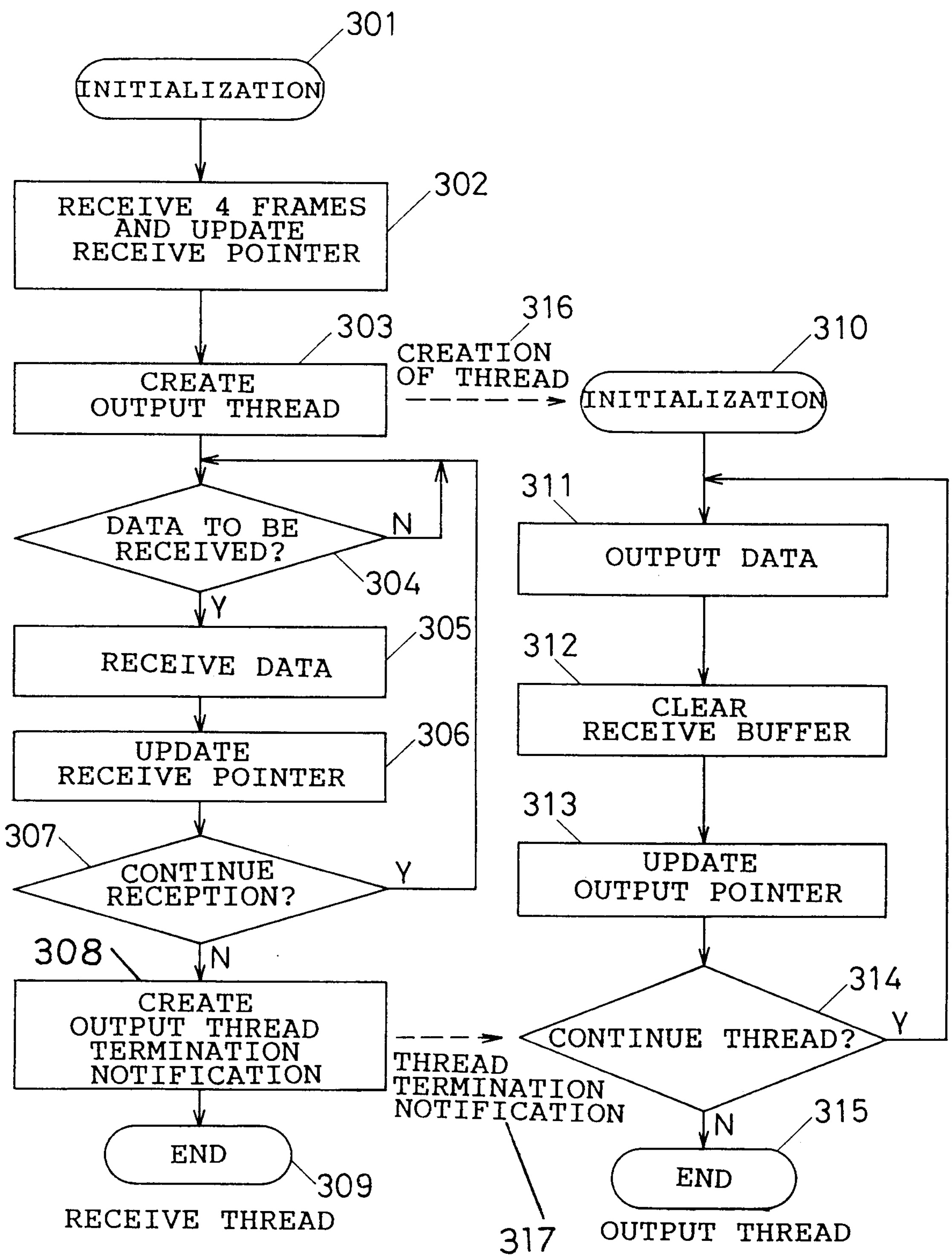

ATM-SW
104
102
103
108
TRANSMIT MEMORY
110
ATM-NIC
PCI BUS
109
MEMORY CONTROLLER
111
CPU
107
DVC ENCODER
106
101
114
RECEIVE MEMORY
116
ATM-NIC
115
MEMORY CONTROLLER
117
PCI BUS
CPU
113
DVC DECODER
112
105
MONITOR

Fig. 2



IPR2022-01227
EXHIBIT 1003 - PAGE 00719

Fig. 3



IPR2022-01227
EXHIBIT 1003 - PAGE 00720

Fig. 4 (a)

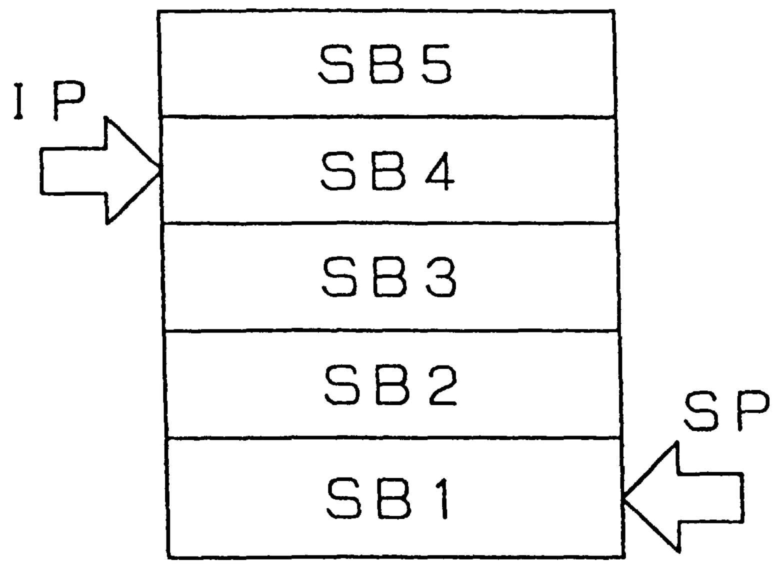
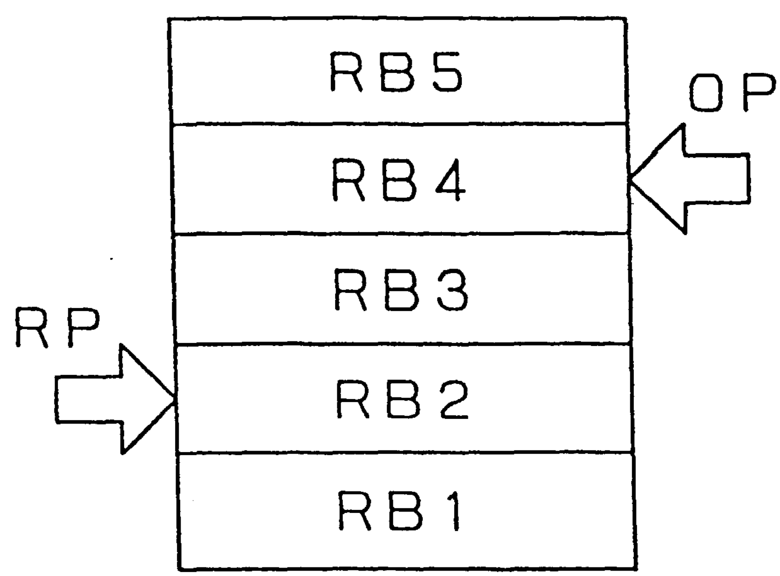
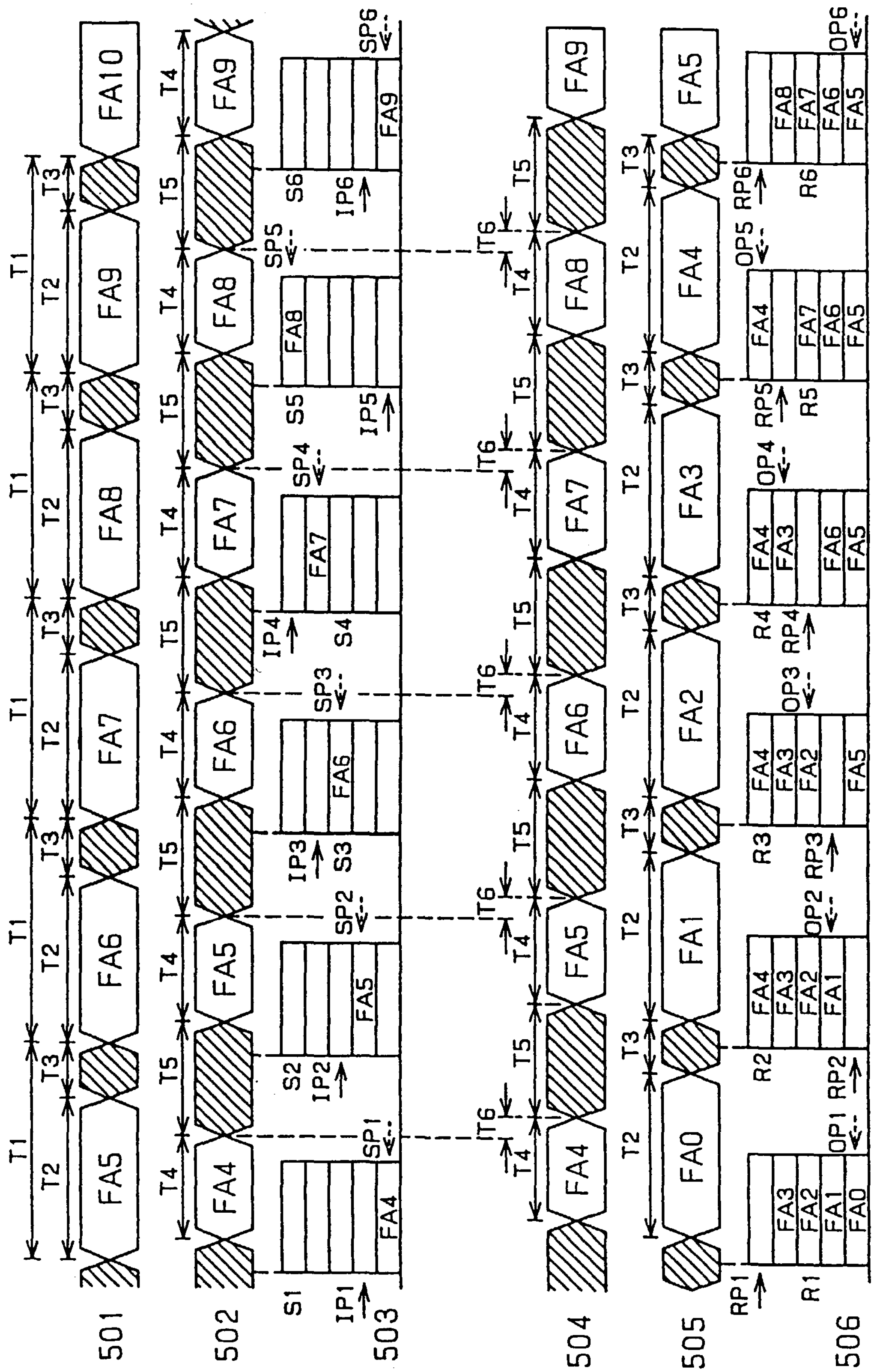


Fig. 4 (b)



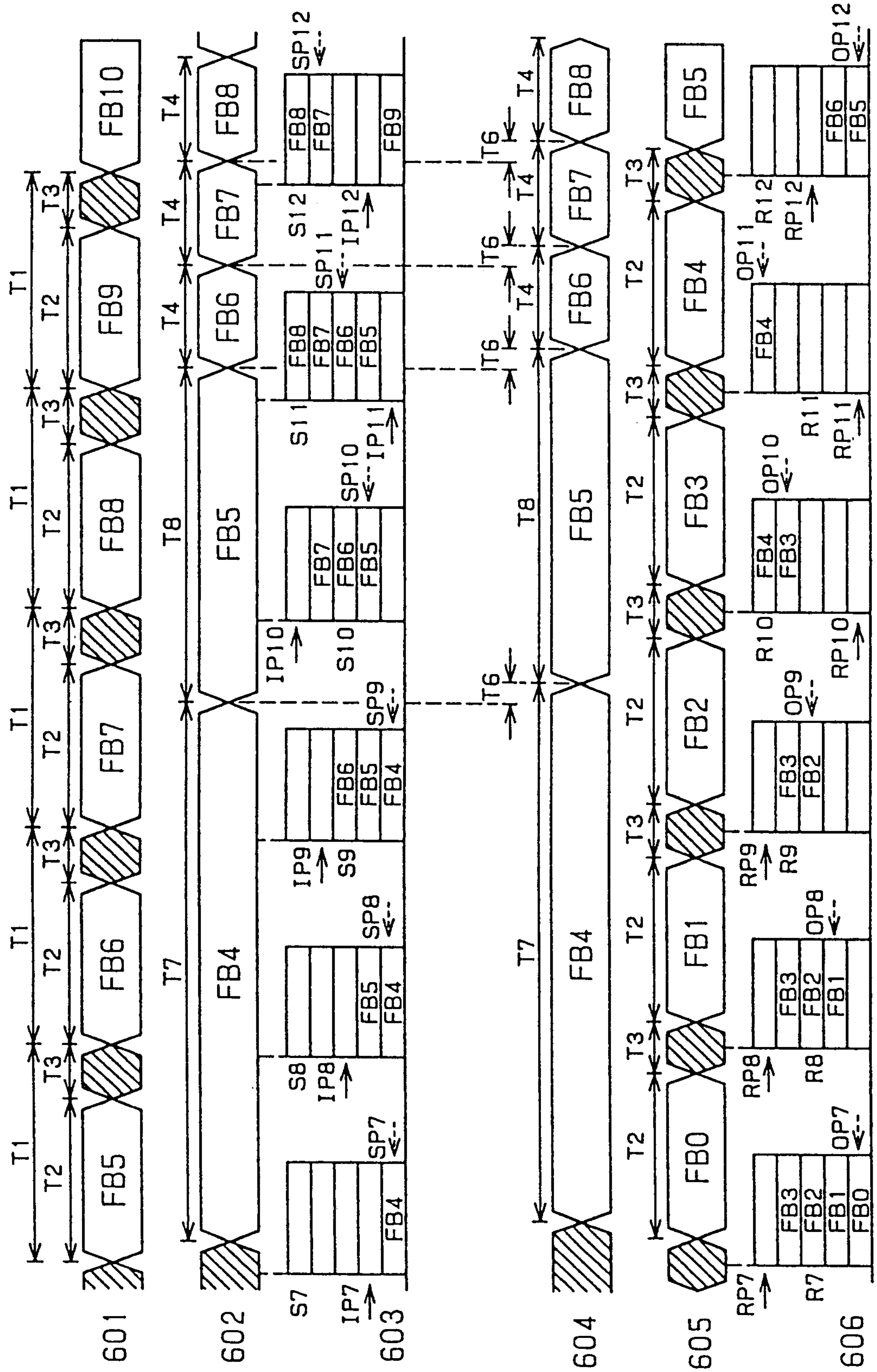
IPR2022-01227
EXHIBIT 1003 - PAGE 00721

Fig. 5



IPR2022-01227
EXHIBIT 1003 - PAGE 00722

Fig. 6



IPR2022-01227
EXHIBIT 1003 - PAGE 00723

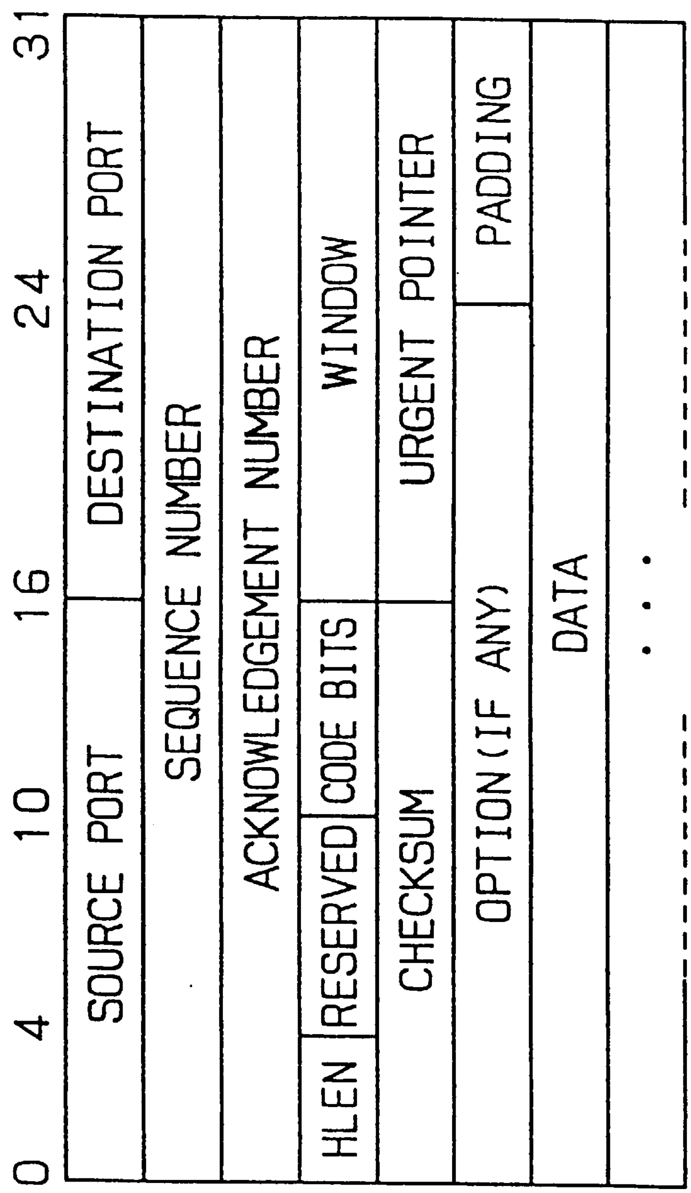


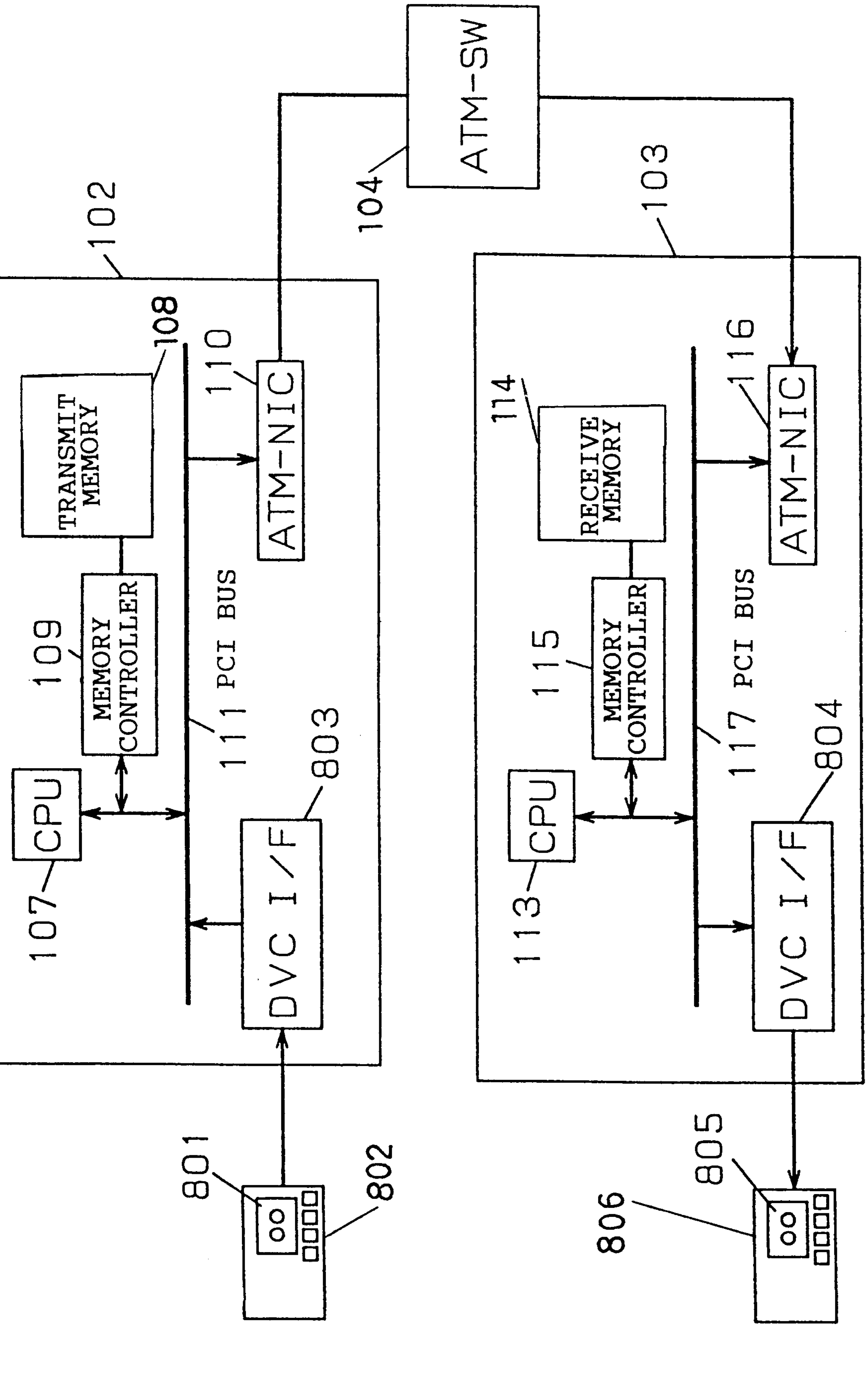
Fig. 7 (a)

IPR2022-01227
EXHIBIT 1003 - PAGE 00724

Fig. 7 (b)

Bit (left to right)	Meaning if bit set to 1
URG	Urgent pointer field is valid
ACK	Acknowledgement field is valid
PSH	This segment requests a push
RST	Reset the connection
SYN	Synchronize sequence number
FIN	Sender has reached end of its byte stream

Fig. 8


ATM-SW
104
102
103
TRANSMIT MEMORY
108
110
ATM-NIC
MEMORY CONTROLLER
109
111 PCI BUS
CPU
107
DVC I/F
803
801
802
RECEIVE MEMORY
114
116
ATM-NIC
MEMORY CONTROLLER
115
117 PCI BUS
CPU
113
DVC I/F
804
805
806

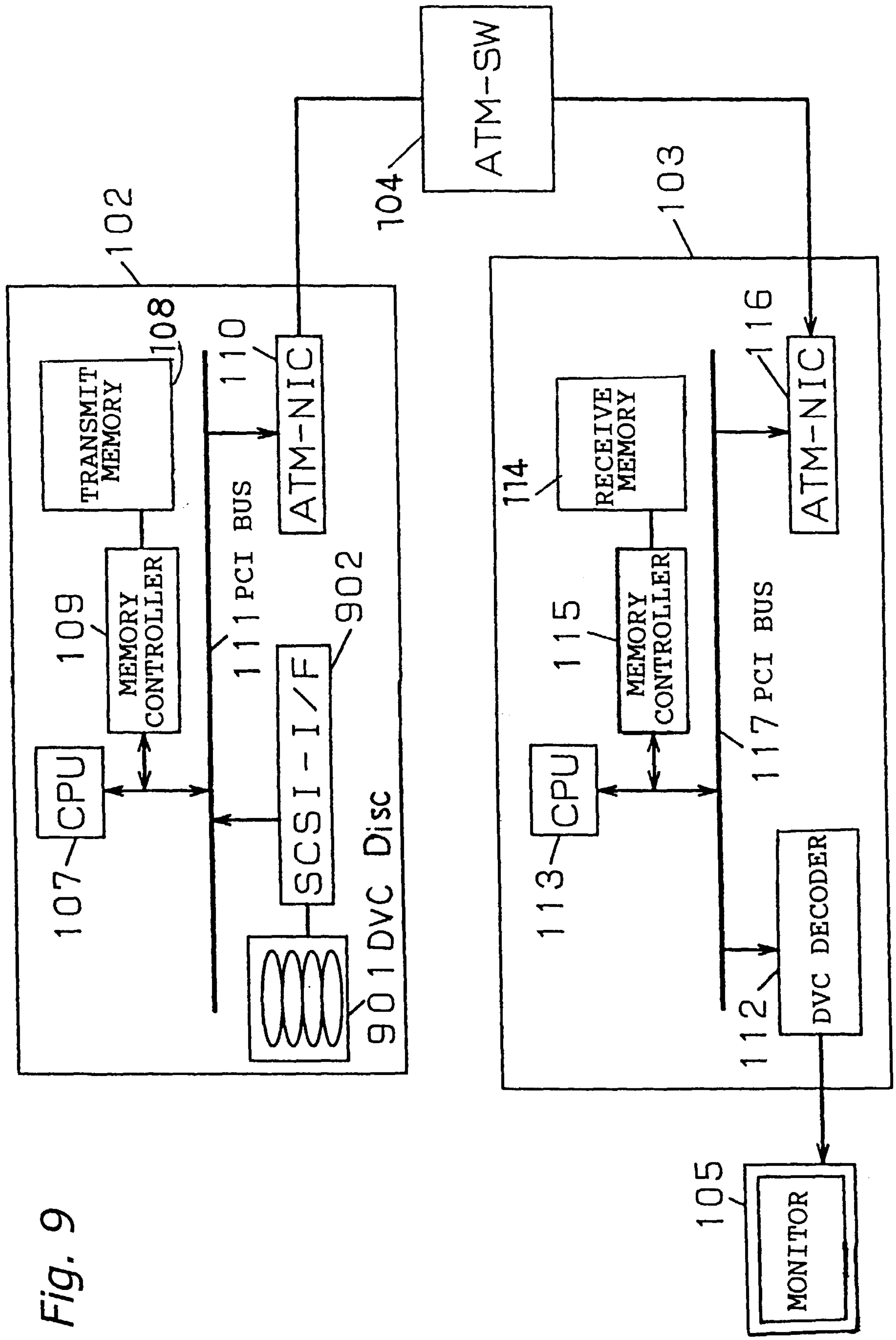
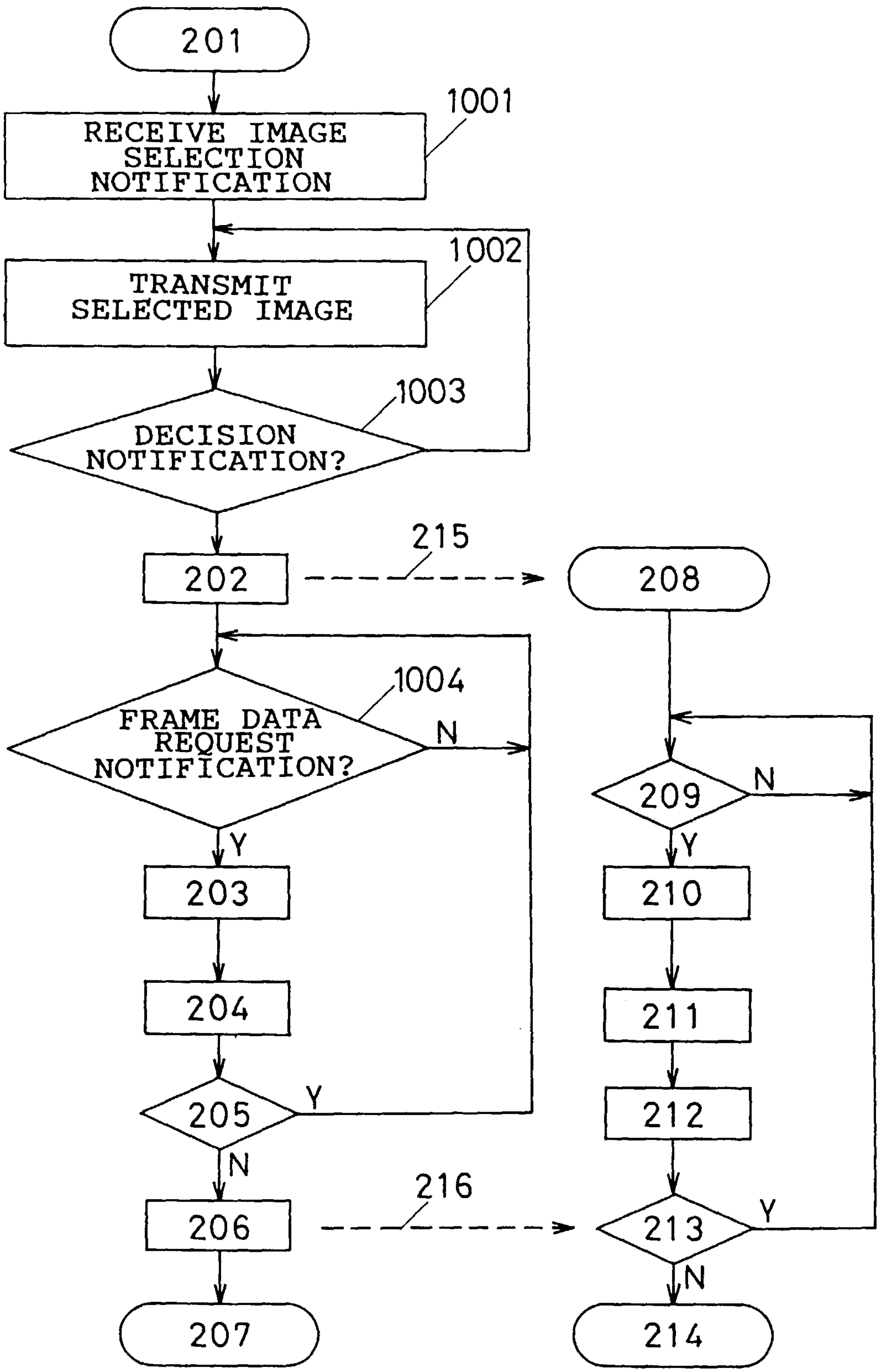

ATM-SW
104
102
TRANSMIT MEMORY
108
110
ATM-NIC
111 PCI BUS
109
MEMORY CONTROLLER
CPU
107
SCSI-I/F
902
901 DVC Disc
103
114
RECEIVE MEMORY
116
ATM-NIC
115
MEMORY CONTROLLER
117 PCI BUS
CPU
113
DVC DECODER
112
105
MONITOR

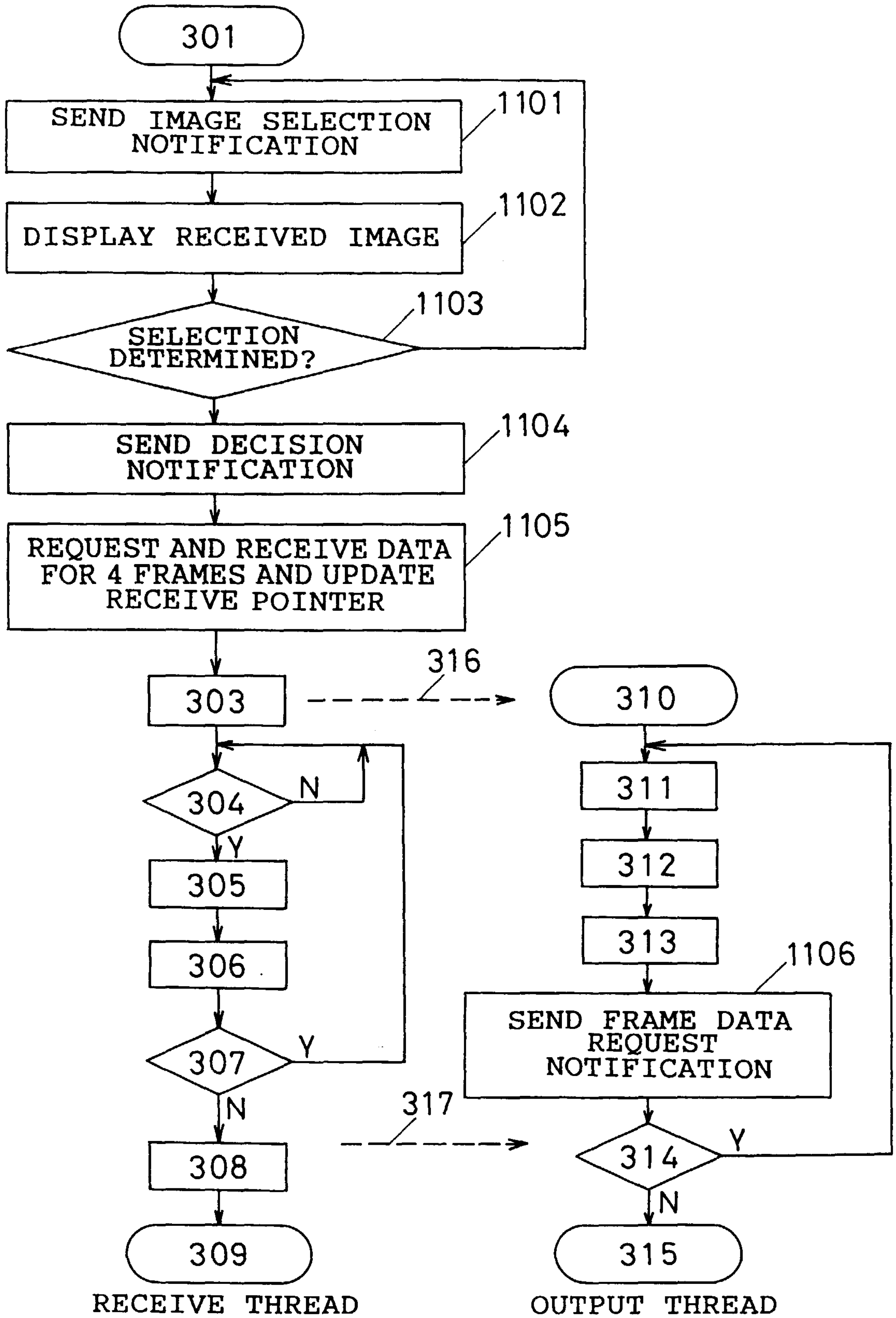
Fig. 9

Fig. 10

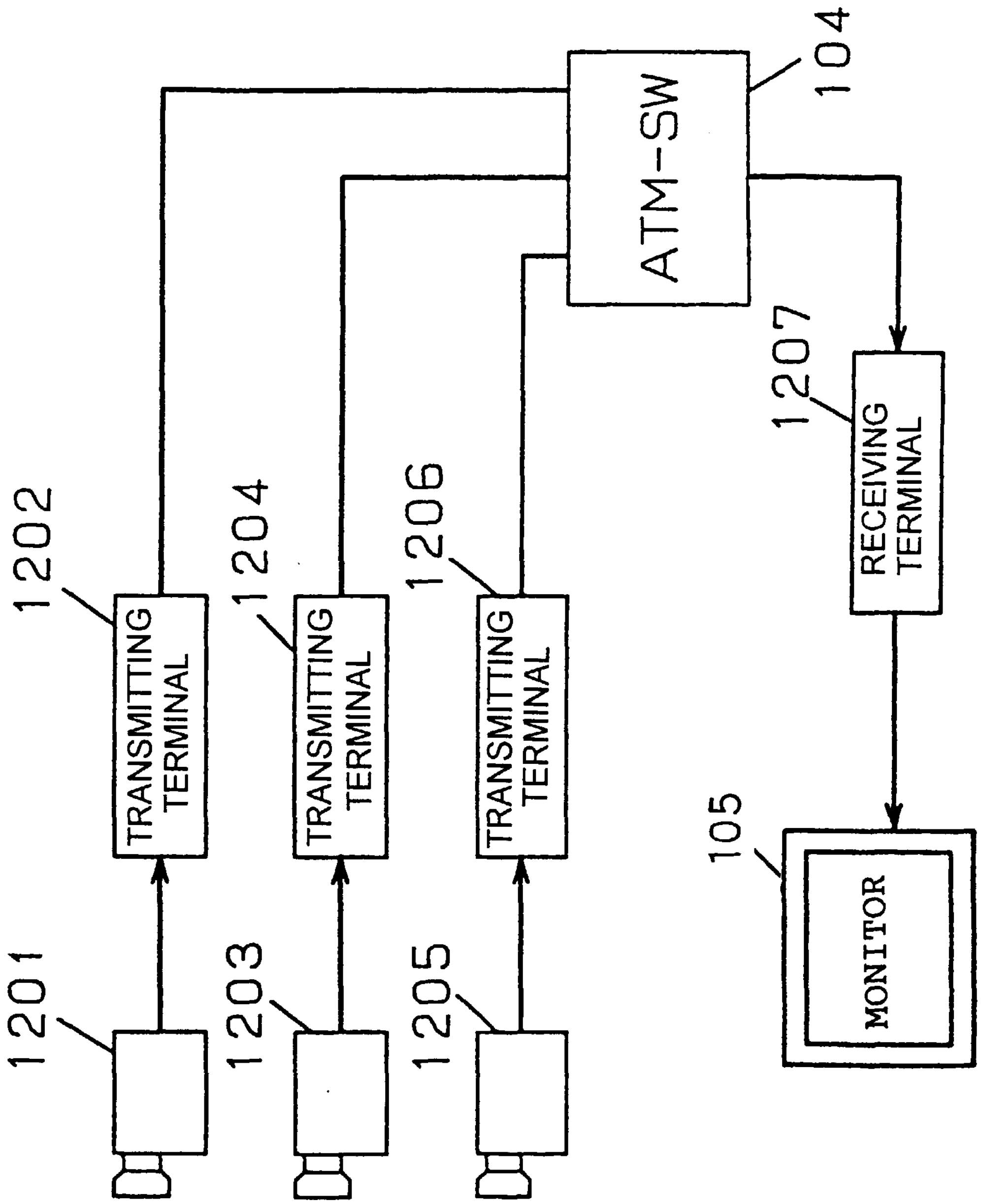


IPR2022-01227
EXHIBIT 1003 - PAGE 00728

Fig. 11



IPR2022-01227
EXHIBIT 1003 - PAGE 00729



ATM-SW
104
1202
1204
1206
1207
TRANSMITTING TERMINAL
TRANSMITTING TERMINAL
TRANSMITTING TERMINAL
RECEIVING TERMINAL
1201
1203
1205
105
MONITOR

Fig. 12

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 895 420 A2

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
03.02.1999 Bulletin 1999/05

(51) Int Cl.[6]: **H04N 7/173**

(21) Application number: **98306129.2**

(22) Date of filing: **31.07.1998**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **01.08.1997 JP 207766/97**
05.09.1997 JP 241280/97

(71) Applicant: **Victor Company of Japan, Ltd.**
Yokohama 221-0022 (JP)

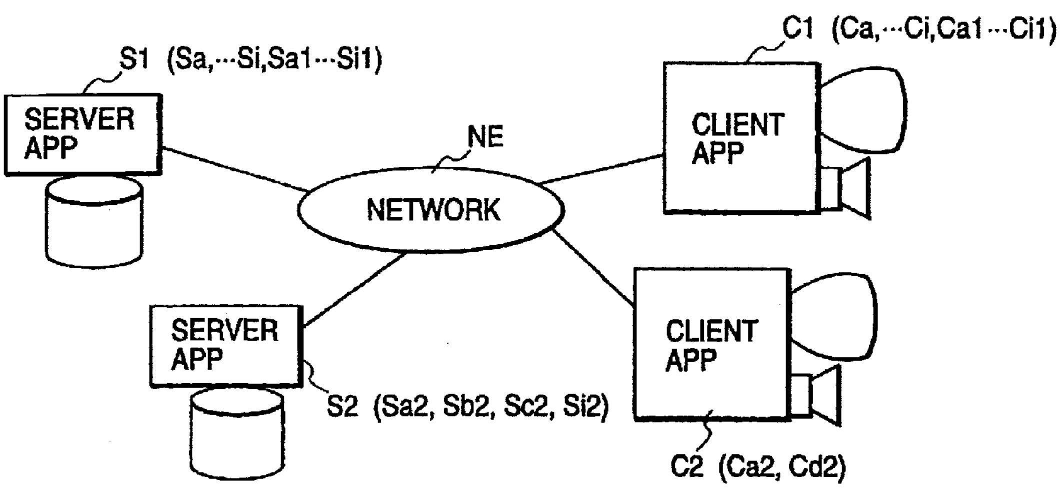
(72) Inventors:
- **Yamada, Yasuhiro**
  Totsuka-ku, Yokohama (JP)
- **Tsutsumi, Masahito**
  Yokosuka-shi, Kanagawa-ken (JP)

(74) Representative:
Dempster, Benjamin John Naftel et al
Withers & Rogers,
4 Dyer's Buildings,
Holborn
London EC1N 2QP (GB)

(54) **A data transmission system and reproducing apparatus**

(57) A data transmission system including: at least one server apparatus for supplying video (moving picture image) data and audio data and at leas client apparatus coupled through a network receiving the video data and audio data is disclosed. The server apparatus transmits a burst including the video data and audio data. The client apparatus receives and reproduce the video and audio data and server's time data indicative of a timing of requesting the next burst. The next burst transmission request is transmitted according to the clock of the client and the server's time data. The video data and audio data may be transmitted independently. The video data and audio data may be transmitted periodically. The next burst transmitting request may be transmitted when an empty data is less than a reference. Data demanding amount data may be transmitted from the client apparatus to said server apparatus. Data transmission number data may be transmitted in the next burst transmission signal. A test signal is transmitted between the client and the server to determined the data demanding amount data and a capacity of the buffer memory and corresponding reproducing apparatus is also disclosed.

FIG. 16A



EP 0 895 420 A2

Description

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0001] This invention relates to a a data transmission system and a reproducing apparatus.

2. Description of the Prior Art

[0002] A data transmission system including a server apparatus supplying video data and audio data to be reproduced continuously in time base to a network and a client apparatus for receiving the video data and the audio data from the network and reproduces the received video data and audio data continuously in time base is known. Moreover, a reproducing apparatus receives video data and audio data from a network and reproduces the received video data and audio data continuously in time base is known.

SUMMARY OF THE INVENTION

[0003] The aim of the present invention is to provide a superior data transmission system and a superior reproducing apparatus.

[0004] According to the present invention a first data transmission system is provided which comprises: at least one server apparatus having a memory storing coded data of at least one of coded moving picture image data and coded audio data and a transmitting circuit for incontinuously transmitting the data read from the memory in every burst to a network system; and at least client apparatus including a receiving circuit for receiving the transmitted data in the burst directing to the client apparatus and a decoding and reproducing circuit for decoding the received data to be reproduced continuously in time base, wherein the server apparatus further comprises a server's time data generation circuit for generating server's time data indicative of time of the next burst to provide continuously reproducing the moving picture image data and the audio data and a data attaching circuit for attaching the server's time data to the data in the burst and the client apparatus further comprises extraction circuit for extracting the server's time data from the received data, a client's time data generation circuit for generating client's time data indicative of present time of the client apparatus, and a data requesting circuit for transmitting a request for the next burst at a request timing determined in accordance with the server's time data from the extraction circuit and present time indicated by the client's time data.

[0005] According to the present invention, a second data transmission system is also provided which comprises at least one server apparatus having a memory storing coded moving picture image data and coded audio data and a transmitting circuit for incontinuously transmitting a video burst including the coded moving picture image data and an audio burst including coded audio data read from the memory to a network system respectively; and at least client apparatus including a receiving circuit for receiving the video and audio bursts directing to the client apparatus and a decoding circui for decoding the coded moving picture image data in the video burst and the coded audio data in the audio burst to be reproduced continuously in time base, wherein the server apparatus further comprises a video time data generation circuit for generating video time data indicative of transmission time of the next video burst to provide continuously reproducing the moving picture image data and the audio time data generation circuit for generating audio time data indicative of transmission time of the next audio burst, and a data attaching circuit for attaching the video time data to the coded moving picture image data in the video burst and for attaching the audio time data to the coded audio data in the audio burst, and the client apparatus further comprises an extraction circuit for extracting the video time data from the video burst from the receiving circuit and the audio time data from the audio burst from the receiving circuit, a client's time data generation circuit for generating a client reference time data, and a first data requesting circuit for transmitting a first request for transmitting the next video burst in accordance with the extracted first server's time data and time indicated by the client's reference time data and a second data requesting circuit for transmitting a second request for transmitting the the next audio burst in accordance with the extracted audio time data and present time indicated by the client's reference time data.

[0006] According to the present invention, a third transmission system is further provided which comprises: at least one server apparatus having a memory storing data of at least one of coded moving picture image data and coded audio data and a transmitting circuit for intermittently transmitting a burst including the data read from the memory to a network; and at least client apparatus including receiving circuit for receiving the transmitted burst directing to the client and a decoding circuit for decoding the received data in the received burst to be reproduced continuously in time base, wherein the server apparatus further comprises a time interval data generation circuit for generating time interval data indicative of time interval of successive coding to provide continuously reproducing the moving picture image data and the audio data and data attaching circuit for attaching the time interval data to the data in the burst and the client apparatus further comprises an extraction circuit for extracting the time interval data from the received data, a clock circuit circuit for generating a reference clock signal, and a data requesting circuit for periodically transmitting a request for transmitting the next burst at a timing determined in accordance with the time interval data and time indicated by the reference clock signal.

[0007] According to the present invention, a fourth data transmission system is also provided which comprises: at least one server apparatus having a memory storing data of at least one of coded moving picture image data and coded audio data and a transmitting circuit for intermittently transmitting a burst including the data read from the memory to a network in response to data request; and at least client apparatus including a data request transmission circuit for transmitting the data request, a receiving circuit for receiving the transmitted data in the burst directing to the client apparatus and a decoding circuit for decoding the data from the receiving circuit to be reproduced continuously in time base, wherein the client apparatus further comprises a network buffer circuit for temporally storing the received data and supplying the temporally stored data to the decoding circuit, a buffer observing circuit for detecting an empty area of the network buffer circuit regarding the stored data, and a request transmitting circuit transmitting the data request to the server when the empty area is higher than a reference and the server apparatus further comprises a request receiving circuit for receiving the data request from the client and operating the transmitting circuit to transmit the next burst in response to the data request.

[0008] In the fourth data transmission system, the request transmitting circuit transmits the data request including empty area size data indicative of a size of the detected empty area and the transmission circuit controls an amount of data of the coded data to be transmitted in accordance with the capacity of the empty area.

[0009] In the fifth data transmission system, the client apparatus may further comprise a request number data generating circuit responsive to the buffer memory observing circuit for generating request number data indicative of successively increasing number when the empty area is higher than the reference, the request transmitting circuit may further transmit the request number data with the transmission request, and the server apparatus may transmit the received request number data with the data requested by the transmission request to the client apparatus.

[0010] In the fifth data transmission system, the buffer observing circuit may generate a test signal and transmit the test signal to the server when the empty area is higher than the reference, the server apparatus may further comprise a test circuit for receiving the test signal and immediately transmitting a response signal at least once, the client apparatus may further comprise a data buffer control circuit for controlling a capacitance of the data buffer circuit, a response signal receiving circuit for receiving the response signal and measuring circuit for measuring an interval between when the test signal is transmitted and when the response signal is received, the data buffer observing circuit further generates amount control data in accordance with the interval and transmits the amount control data with the next burst transmission request, the request receiving circuit in the server apparatus controls transmitting the burst including the coded data at an amount determined in accordance with the amount control data, and the data buffer control circuit controls the capacitance of the data buffer circuit in accordance with the interval.

[0011] In this case, the server apparatus may further comprise a test signal generation circuit for generating a test signal and transmitting the test signal to the server in response to the data request, a calculation circuit, and a compensation circuit, the client apparatus may further comprise a test circuit for receiving the test signal and immediately transmitting a response signal including data transmitting amount data and capacity data of the data buffer circuit, the server apparatus may further comprise a response signal receiving circuit for receiving the response signal and a measuring circuit for measuring an interval between when the test signal is transmitted and when the response signal is received, the calculation circuit may calculate a data rate from the the interval and the capacity data, and the compensation circuit may compensate the data amount data from the response signal receiving circuit in accordance with the data rate from the calculation circuit.

[0012] In the first to fifth data transmission systems, the client apparatus may further comprise a reproducing circuit for reproducing the video data and the audio data continuously in time base as a reproducing apparatus.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] The object and features of the present invention will become more readily apparent from the following detailed description taken in conjunction with the accompanying drawings in which:

Fig. 1 is a block diagram of a client apparatus of the first embodiment;

Figs. 2A to 2D show timing charts of transmitting data requests and transmitted data of the first embodiment;

Figs. 3A and 3B are timing charts showing data requests and data transmission of the first embodiment in detail;

Figs. 4A and 4B are timing charts of a second embodiment showing data transmission between a server apparatus and a client apparatus;

Figs. 5A and 5B shows timing charts of a third embodiment showing transmission timings between a server apparatus and a client apparatus;

Fig. 6 is a block diagram of a server apparatus of a fourth embodiment;

Fig. 7 is a block diagram of the fourth embodiment showing a network buffer memory shown in Fig. 6;

Figs. 8A to 8D are timing charts of the fourth embodiment showing transmission timings between the server apparatus and the client apparatus;

Figs. 9A and 9B are timing charts of a fifth embodiment showing transmission operation between a server apparatus and a client apparatus;

Figs. 10A and 10B are timing charts of a sixth embodiment showing transmission operation between a server apparatus and a client apparatus;

Figs. 11A and 11B are timing charts of a seventh embodiment showing transmission operation between a server apparatus and a client apparatus;

Figs. 12A and 12B are timing charts of a modification in the seventh embodiment showing transmission operation between the server apparatus and the client apparatus;

Figs. 13A and 13B are timing charts of an eighth embodiment showing transmission operation between a server apparatus and a client apparatus;

Fig. 14 is a block diagram of a client apparatus of a ninth embodiment;

Figs. 15A to 15D are timing charts of the ninth embodiment showing the video data transmission operation between the server apparatus and the client apparatus and the audio data transmission operation between the server apparatus and the client apparatus;

Figs. 16A and 16B are block diagrams of data transmission systems of this invention;

Fig. 17 is a block diagram of a server apparatus of the first embodiment;

Fig. 18 is a block diagram of a server apparatus of the second embodiment;

Fig. 19 is a block diagram of a client apparatus of the second embodiment;

Fig. 20 is a block diagram of a server apparatus of the third embodiment;

Fig. 21 is a block diagram of a client apparatus of the third embodiment;

Fig. 22 is a block diagram of a client apparatus of the fourth embodiment;

Fig. 23 is a block diagram of a server apparatus of the fifth embodiment;

Fig. 24 is a block diagram of a client apparatus of the fifth embodiment;

Fig. 25 is a block diagram of a server apparatus of the sixth embodiment;

Fig. 26 is a block diagram of a client apparatus of the sixth embodiment;

Fig. 27 is a block diagram of a server apparatus of the seventh embodiment;

Fig. 28 is a block diagram of a client apparatus of the seventh embodiment;

Fig. 29 is a block diagram of a server apparatus of the eighth embodiment;

Fig. 30 is a block diagram of a client apparatus of the eighth embodiment; and

Fig. 31 is a block diagram of a server apparatus of a ninth embodiment.

[0014] The same or corresponding elements or parts are designated with like references throughout the drawings.

DETAILED DESCRIPTION OF THE INVENTION

(FIRST EMBODIMENT)

[0015] Fig. 1 is a block diagram of a client apparatus of the first embodiment. Figs. 16A and 16B are block diagrams of the data transmission system of this invention. Fig. 17 is a block diagram of a server apparatus of the first embodiment.

[0016] The data transmission system of the first embodiment comprises at least a server apparatus Sa and at least a client apparatus Ca coupled through a network NE comprising a single network as shown in Fig. 16A. The server apparatus Sa may coupled to the client apparatus Ca through a network system NE' including a plurality of different kinds of networks as shown in Fig. 16B.

[0017] A server apparatus Sa includes a memory 1701 for storing data of at least one of coded video (moving picture image) data and coded audio data to be reproduced continuously and a transmission circuit 1702 for intermittently transmitting a burst of the data read from the memory 1701 to a network system NE or NE' through a network interface circuit 1703.

[0018] The client apparatus Ca has a network interface 11, a buffer memory 12 for receiving the transmitted data in the burst directing to the client apparatus Ca and a decoder 18 for decoding the received coded video data and audio (sound) data from the buffer memory 12 in response to client's time data 15a, and a reproducing unit 19 for reproducing the decoded video data and the decoded audio data continuously in time base.

[0019] The server apparatus Sa further comprises a server's time data generation circuit 1704 for generating server's time data indicative of time of next transmitting the burst to provide continuously reproducing the video data and the audio data. The transmission circuit 1702 attaches the server's time data to the coded data from the memory 1701 in the burst to transmit the burst to the network system NE' through the network interface circuit 1703.

[0020] The client apparatus Ca further includes a time data extraction circuit 13 for extracting the server's time data from the received data, a client's time data generation circuit 15 for generating a client's time data in response to a reference clock signal, and a data requesting circuit 21 for transmitting a request for transmitting the next burst at a request timing determined in accordance with the server's time data and client's time data. More specifically, the data request circuit 21 comprises a subtractor 14, a memory 16, a request timing determining circuit 17, and a request transmission circuit 20.

[0021] The subtractor 14 obtains a difference between the server's time data from the time data extraction circuit 13 and the client's time data 15a when the time data is received. The data requesting circuit 17 compensates a relation between the client's time data and the received server's time data in accordance with the difference from the memory 16, and determines the request timing in accordance with the client's time data and the compensated relation. Alternatively, the client's time data generation circuit 15 may be compensated in accordance with the extracted server's time data.

[0022] In the data transmission system of the first embodiment, the server apparatus Sa stores the coded video data and the accompanied audio data, the coded audio data, and the coded video data and transmits in response to a program transmission request from the client apparatus Ca. The server apparatus Sa transmits the program data incontinuously, that is, intermittently transmits a burst including the video data and the audio data to be continuously reproduced in the real time base without lack of data to be reproduced.

[0023] Figs. 2A to 2D show timing charts of transmitting request and transmitted data of the first embodiment.

[0024] The client apparatus Ca transmits a request for transmitting the next burst of the coded data to the sever apparatus S1. In response to this, the server apparatus S1 transmits the coded data read from the memory 1701 to the client apparatus C1. On the other hand, another client apparatus C2 transmits a request for transmitting the next burst of the coded data to the same sever apparatus S1. In response to this, the server apparatus S1 transmits the coded data read from the memory 1701 to the client apparatus C2 at different timing. If there is a collision of requesting data to the same server apparatus S1, one of the client apparatus, (ex. C1), should wait until the other client apparatus (ex. Ca) finishes receiving the transmitted data. However, if there is a collision between data requesting by the client apparatus C1 and C2 to different server apparatus S1 and S2, it is possible to transmit the data at the same time.

[0025] Figs. 3A and 3B are timing charts showing data requests and data transmission of the first embodiment in detail.

[0026] A client Ca transmits the data request rl for the next burst of the coded data to the server apparatus Sa through a network system NE or NE', the server apparatus Sal transmits the coded data read from the memory 1701 in a burst 31. The burst 31 includes coded data d1 (d1-d4) such as the video data and the accompanied audio data or the data to be reproduced in the real time base and the server's time data t1 (t1-t4) indicative of next transmission to be requested by the client apparatus Ca. However, it is also possible to transmit only the coded data d5 in a burst 32. The server's time data t1 is time-division multiplexed with the coded data at the end of the burst 31. However, it is also possible to arrange the server's time data in any position in the burst 31.

[0027] The server's time data t1, t2, t3, and t4 represents the last time information of each data d1 to d4 or time information of the next data d2, d3, d4, and d5 which is being waiting for transmission.

[0028] The client apparatus Ca receives variable length of coded data, that is, the burst 31 by the network interface circuit 11 and temporally stores the coded data dl in the buffer memory 12. On the other hand, the server's time data tl is extracted from the received data from the network interface circuit 11 and supplies the extracted server's time data t1 to the subtractor 14 and to the request timing determining circuit 17. The decoder 18 decodes the coded data d1 in response to a reference clock signal and the client's time data 15a from the client's time data generation circuit 15. The client's time data generation circuit comprises a counter (not shown) for counting the reference clock signal of 90 KHz generated from a clock signal of 27 MHz for digital video signal for example.

[0029] Generally, there is a difference between the server's time data t1 and the client's time data. Therefore, the subtractor 14 obtains the difference between the server's time data tl and the client's time data and the memory 16 stores the difference. The request timing determining circuit 17 compensates the server's time data and compares the compensated server's time data with the client's time data 15a. When there is agreement, the request transmission circuit 20 generates the data request r2 requesting the next transmitting the burst.

[0030] Compensating the server's time data may be executed by the server's time data is preset to the client's time data generation circuit (counter) 15 and then, the client's time data generation circuit 15 self-runs. That is, the client's time data generation circuit 15 may comprise a self-running counter.

[0031] The request timing determining circuit 17 compensates the server's time data t1 in accordance with the difference stored in the memory 16 and at T1 indicated by the compensated tl, the request timing determining circuit 17 informs the request transmission circuit 20 of the requesting timing. The request transmission circuit 20 generates the request r2 and transmits it to the server apparatus Sa through the network interface circuit 11. The server apparatus Sa transmits the next coded data d2 and the server's time data t2 to the client apparatus Ca according to the request r2.

[0032] There is a delay to receive the coded data (ex. d1) from transmission of the request (ex. r1). However, it is also possible to take consideration of the delay in transmission of the data request (ex. r1). Moreover, if the capacity of the data buffer is sufficient, it is possible to transmit the request earlier than the determined timing to provide a margin against busy traffic or erroneous transmission.

[0033] This processing is repeated to reproduce the data to be reproduced in real time base.

[0034] The coded data d1-d5 includes error check data Cr3 (cyclic redundancy check code) and the decoder 18 effects the error check and the error correction in accordance with the error check data Cr3.

[0035] If an error occurs which cannot be corrected in accordance with the error check data Cr3 during decoding, the client apparatus Ca transmits the re-transmission request "e", as shown by Fig. 3, to the server apparatus Sa. Then, in response to the re-transmission request, the server apparatus Sa transmits the same coded data d4 and the server's time data t4 to the client apparatus Ca again.

[0036] The client apparatus Ca decodes the re-transmitted data d4 by the decoder 14 and at the timing T4 indicated by the compensated t4 as described above, the data request r5 for next transmission of the burst is transmitted by the request transmission circuit 20.

[0037] In response to this data request r5, the server apparatus Sa transmits the coded data d5 to the client Ca.

[0038] As mentioned, in response to the data request for next transmitting the coded data to be reproduced in the real time base by the client apparatus Ca, the client apparatus Ca can receives the incontinuously transmitted coded data and the server's time data from the server apparatus Sa thorough the network system NE or NE', so that the client apparatus Ca can continuously reproduce the incontinuously transmitted data as moving picture image and a sound.

(SECOND EMBODIMENT)

[0039] Fig. 18 is a block diagram of a server apparatus of a second embodiment. Fig. 19 is a block diagram of a client apparatus of a second embodiment.

[0040] A data transmission system of the second embodiment includes at least one server apparatus Sb and at least one client apparatus Cb coupled through the network system NE or NE'.

[0041] The data transmission system of the second embodiment is substantially the same as the first embodiment. The difference is that the video data and the audio data is transmitted independently with transmission timings controlled video time data and audio time data transmitted from the server apparatus Sb. That is, in Fig. 18, the server apparatus Sb includes a video time data generation circuit 1804 for generating video time data indicative of next transmission time of the video burst to provide continuously reproducing the video data and an audio time data generation circuit 1805 for generating audio time data indicative of next transmission time of the next audio burst. The transmission circuit 1802 attaches the video time data from the memory 1701 to the coded video data in the video burst and independently attaches the audio time data from the memory 1701 to the coded audio data in the audio burst,

[0042] Correspondingly, in Fig. 19, the client apparatus Cb includes a time data extraction circuit 1913 for extracting the video time data and the audio time data from the video burst and the audio burst respectively, a video data request circuit 1821 and an audio data request circuit 1921.

[0043] The video data request circuit 1821 comprises a subtractor 1814, a memory 1816, a request timing determining circuit 1817, and a request transmission circuit 1820.

[0044] The subtractor 14 obtains a difference between the video time data from the time data extraction circuit 1913 and the client's time data 15a when the video time data is received. The data requesting circuit 1817 compensates a relation between the client's time data and the received video time data in accordance with the difference from the memory 1816, and determines the request timing of the video data in accordance with the client's time data and the compensated relation regarding the video time data.

[0045] The audio data request circuit 1921 comprises a subtractor 1914, a memory 1916, a request timing determining circuit 1917, and a request transmission circuit 1920.

[0046] The subtractor 1914 obtains a difference between the audio time data from the time data extraction circuit 1913 and the client's time data 15a when the audio time data is received. The data requesting circuit 1917 compensates a relation between the client's time data and the received audio time data in accordance with the difference from the memory 1916, and determines the request timing of the audio data in accordance with the client's time data and the compensated relation regarding the audio time data.

[0047] In response to the request transmission circuits 1820 and 1920, the request transmission circuits 1820 and 1920 respectively transmit the video data request and the audio data request independently.

[0048] In the server apparatus Sb, the video time data generation circuit 1804 generates the video time data indicative of next transmission time of the video burst to provide continuously reproducing the video data. On the other hand, the audio time data generation circuit 1805 generates the audio time data indicative of next transmission time of the audio burst independently. The transmission circuit 1802 having a data attaching function attaches the video time data to the coded video data in the video burst and attaches the audio time data to the coded audio data in the audio burst.

[0049] In the client apparatus Cb, the time data extraction circuit 1913 extracts the video time data from the video burst from the network interface circuit 11 and supplies the video time data to the video data request circuit 1821 and extracts the audio time data from the audio burst from the network interface circuit 11 and supplied the extracted audio time data to the audio data request circuit 1921. The client's time data generation circuit 1915 supplies the client's time data to the video data request circuit 1821 and the audio data request circuit 1921.

[0050] The video data requesting circuit 1821 transmits the video data request for transmitting the next video burst in accordance with the extracted video time data and the present time indicated by the client's time data. The audio data requesting circuit 1921 transmits the audio request for next transmitting the audio burst in accordance with the extracted audio time data and present time indicated by the client's time data.

[0051] Figs. 4A and 4B are timing charts of the second embodiment showing data transmission between the server apparatus Sb and the client apparatus Cb.

[0052] The client apparatus Cb transmits the video data request f1 and the audio data request g1 to the server apparatus Sb as shown in Fig. 4A. In response to the video data request f1 and the audio data request g1, the server apparatus Sb transmits a video burst V0 including video data and video time data tv1 and an audio burst A0 including the audio data and the audio time data ta1. At timing Tv1, indicated by the compensated video time data tv1, the client apparatus Cb transmits video data request f2. In response to reception of the video data request f2, the server apparatus Sb transmits the next video burst V1. On the other hand, at the timing Ta1 indicated by the compensated audio time data tal, the video burst V1 is being transmitted from the server apparatus Sb to the client apparatus Cb, so that the client apparatus Cb postpones transmitting audio data request g2 to completion of the video burst V1 and transmits the audio data request g2 at a timing Ta1'. This processing is repeated to reproduce the video data and the audio data continuously in time base.

[0053] In this embodiment, transmitting the video data and the audio data is controlled by independent video time data and audio time data indicating the next transmission video burst and audio burst respectively, wherein the video time data and the audio time data is compensated by the subtractors 1814 and 1914 to obtain the difference between the video time data extracted by the time data extraction circuit 1913 and the client's time data and the difference between the audio time data extracted by the time data extraction circuit 1913 and the client's time data respectively, so that it is possible to receive the video data and the audio data from the different server apparatus Sb1 and Sb2. For example, the client apparatus Cb receives the video data from the server apparatus Sb1 and receives the audio data from the server apparatus Sb2.

(THIRD EMBODIMENT)

[0054] Fig. 20 is a block diagram of a server apparatus Sc of a third embodiment. Fig. 21 is a block diagram of a client apparatus Cc of a third embodiment.

[0055] A data transmission system of the third embodiment includes at least one server apparatus Sc and at least one client apparatus Cc coupled through the network system NE or NE'.

[0056] The data transmission system of the third embodiment is substantially the same as the first embodiment. The difference is that interval data is transmitted from the server apparatus Sc to the client apparatus Cc at least once and the data request of the burst is repeated periodically at the interval indicated by the interval data transmitted from the server apparatus Sc.

[0057] The server apparatus Sc comprises an interval data generation circuit 2004 for generating video time interval data indicative of time interval of successive coding to provide continuously reproducing the video data and audio time interval data indicative of time interval of audio burst for continuously reproducing. The transmission circuit 2002 attaches the video time interval data to the video data from the memory 2001 in the video burst and the audio time interval data to the audio data from the memory 2001 in the audio burst to the client apparatus Cc through the network interface circuit 2003.

[0058] The client apparatus Cc, as shown in Fig. 21, comprises an interval data extraction circuit 2113 for extracting the video time interval data and the audio time interval data from the received data from the network interval circuit 11, a request timing determining circuit 2114 for determining a request timing of the video burst in accordance with the video time interval data from the interval data extraction circuit 2113 and the client's time data from the client's time data generation circuit 15 and a request timing determining circuit 2117 for determining a request timing of the audio burst in accordance with the audio time interval data from the interval data extraction circuit 2113 and the client's time data from the client's time data generation circuit 15, a request transmission circuit 2116 responsive to the request timing determining circuit 2114 for transmitting data request for the next video burst, a request transmission circuit 2120 responsive to the request timing determining circuit 2117 for transmitting data request for the next audio burst through the network interface circuit 11 in addition to the buffer memory 12, the decoder 18, the reproduction circuit 19, the client's time data generation circuit 15, and the network interface circuit 11.

[0059] Figs. 5A and 5B show timing charts showing transmission timings between the server apparatus Sc and the client apparatus Cc.

[0060] In this embodiment, the video time interval data j1 and the audio time interval data k1 is transmitted at the first video burst V0 and the first audio burst A0. In Fig. 21, the request transmission circuit 2116 periodically transmits the data request of the video burst at the interval determined in accordance with the video interval data j1 and the request transmission circuit 2120 periodically transmits the data request of the audio burst at the interval determined in accordance with the audio interval data k1. The buffer memory 12, the decoder 18, and reproducing circuit 19 continuously reproduce the incontinuously received video data and the audio data.

[0061] For example if the video data is coded every frame of 25 Hz, the coding period of the video data is 40 ms (=1/25 Hz) and the audio signal is sampled at 32 kHz and coded every 1024 events of sampling (this is referred to as an audio frame), so that the coding period of the audio data is 32 ms (=1024/(32x 10^3) Hz).

[0062] In Figs. 5A and 5B, the client Cc transmits data requests h1 and i1 to the server apparatus Sc. In response to this, as shown in Fig. 5B, the server apparatus Sc transmits the first video burst V0 including the video interval data j1 of 40 msec coding period and the video data and the first audio burst A0 including the audio interval data k1 of 32 ms coding period and the audio data. That is, the client apparatus Cc transmits the data request of the video burst every 40 ms as shown by the references h2 to h4 and one frame of the video data as shown by the references V1 to V3 respectively. On the other hand, the client apparatus Cc transmits the data request of the audio burst every 32 ms as shown by the references i2 to i4 and one frame of the audio data as shown by the references A1 to A3 respectively.

[0063] In this embodiment, the video time interval data j1 and the audio time interval data k1 is transmitted before the first video data vd at the same first video burst V0 and before the first audio data ad at the same first audio burst A0. However, it is also possible that the video time interval data j1 is independently transmitted before the first video data vd and the audio time interval data k1 is independently transmitted before the first audio data ad.

[0064] In this embodiment, the transmitting data is controlled in accordance with the client's time data from the client's time data generation circuit 15, so that it is possible that different server apparatus Sc1 and Sc2 transmit the video burst and the audio burst.

[0065] Moreover, the capacity of the buffer memory 12 is determined to store N frames of the video data or the audio data, so that the transmission request may be effected every N frames, N being a natural number.

[0066] If the inter-frame prediction such as MPEG2 is used, an interval equal to or more than three frames is necessary to output the first frame. In this case, the data request for the video data is performed first and then, after two frames, the data request for the audio data is effected.

(FOURTH EMBODIMENT)

[0067] Fig. 6 is a block diagram of a client apparatus Cd of a fourth embodiment. Fig. 22 is a block diagram of a server apparatus Sd of a fourth embodiment.

[0068] A data transmission system of the fourth embodiment includes at least one server apparatus Sd and at least one client apparatus Cd coupled through the network system NE or NE'.

[0069] The server apparatus Sd includes a memory 1701 for storing data of at least one of coded video data and coded audio data, a transmitting circuit 2201 for intermittently transmitting a burst including the data read from the memory 1701 to the network system in response to data request, and a network interface circuit 1703, and a request receiving circuit 2202.

[0070] The client apparatus Cd includes a network interface circuit 41, a receiving circuit 47 including network buffer memory 42 for receiving the transmitted data in the burst directing to the client apparatus Cd, the network buffer memory 42 for temporally storing the received data and supplying the temporally stored data, a buffer observing circuit 43 for detecting an empty area of the stored data and transmitting the data request to the server Sd when the empty area is higher than a reference, a decoding buffer 44 for storing the data from the network buffer memory 42, a decoder 45 for decoding the data from the decoding buffer memory 44, a reproducing circuit 19 for reproducing the data from the network buffer memory 42 continuously in time base, and a time data and clock generation circuit 46 for generating time data of the client apparatus Cd and a reference clock signal.

[0071] Fig. 7 is a block diagram of the fourth embodiment showing a network buffer memory 42 shown in Fig. 6.

[0072] The the network buffer memory 42 comprises a FIFO (First-In-First-Out) memory 51 for successively storing the received data and successively reading and outputting the received data and outputting a quarter-occupied signal 52, a half-occupied signal 53, and a three-quarter-occupied signal 54.

[0073] When a quarter of the capacity of the FIFO MEMORY 51 is occupied, the quarter-occupied signal 52 is outputted, when a half of the capacity is occupied, the half-occupied signal 53 is outputted, and when three quarters of the capacity is occupied, the three-quarter-occupied signal 54 is outputted.

[0074] The buffer memory observing circuit 43 detects that the empty area of the network buffer memory 42 is higher than the reference by logic operation among the quarter-occupied signal 52, the half-occupied signal 53, and the three-quarter-occupied signal 54.

[0075] The server apparatus Sd transmits the coded data to the client apparatus Cd in response to the data request from the client apparatus Cd to the server apparatus Sd. The client apparatus Cd receives the coded data from the server apparatus Sd in the network buffer memory 42.

[0076] The coded data in the network buffer memory 42 is read and supplied to the decoder 45 through the decoding buffer memory 44. The decoder 45 decodes the coded data from the decoding buffer memory 44 on the basis of the time data and the reference clock signal from the time data and clock signal generation circuit 46 to output the video data and the audio data continuously in time base with a synchronous relation therebetween. The reproducing circuit 19 including a CRT and a speaker reproduces the video data and the audio data.

[0077] The buffer memory observing circuit 43 responsive to the quarter-occupied signal 52, the half-occupied signal 53, and the three-quarter-occupied signal 54 observes the empty area of the network buffer memory 42 and when the empty is higher than a half of the capacity for example, the buffer memory observing circuit 43 transmits data request to the server apparatus Sd.

[0078] Figs. 8A to 8D are timing charts of the fourth embodiment showing transmission timings between the server apparatus Sd1 and the client apparatus Cd1 and between the server apparatus Sd1 and the client apparatus Cd2.

[0079] For example, as shown in Figs. 8A and 8B, the client apparatus Cd1 transmits a data request to the server apparatus Sd1 by the buffer memory observing circuit 43 because the empty area is higher then a half of the capacity of the network buffer memory 42 (Fig. 8A). In response to this, the server apparatus Sd1 transmits a burst of which amount is less than a half of the capacity of the network buffer memory 42 to the client apparatus Cd1 (Fig. 8B).

[0080] Similarly, as shown in Figs. 8C and 8D, the client apparatus C2d transmits a data request to the server apparatus Sd1 by the buffer memory observing circuit 43 because the empty area is higher then a half of the capacity of the network buffer memory 42 (Fig. 8C). In response to this, the server apparatus Sd1 transmits a burst of which amount is less than a half of the capacity of the network buffer memory 42 to the client apparatus Cd2 (Fig. 8D). The capacity of the network buffer memory 42 may be predetermined or registered in advance. If the capacity of the network buffer memory 42 is registered in advance, the capacity may be different between respective client apparatus.

[0081] As mentioned in the network buffer memory, there is coded data of which amount is higher than a predetermined amount, so that it is possible to reproduce the video data and the audio data in accordance with the client's time data.

(FIFTH EMBODIMENT)

[0082] Fig. 23 is a block diagram of a server apparatus Se of a fifth embodiment. Fig. 24 is a block diagram of a client apparatus Ce of the fifth embodiment.

[0083] The data transmission system of the fifth embodiment is substantially the same as that of the fourth embodiment. The difference is that the client apparatus Ce transmits the data request including transmission requesting amount data corresponding to a size of the detected empty area and the server apparatus transmits the burst including coded data at an amount of data in a burst determined in accordance with the transmission requesting amount data from the client apparatus Ce.

[0084] More specifically, the buffer memory observing circuit 143 transmits the data request including the transmission request amount data and the transmission circuit 2301 controls an amount of data of the coded data to be transmitted in a burst in accordance with the transmission request amount data.

[0085] Figs. 9A and 9B are timing charts of the fifth embodiment showing transmission operation between the server apparatus Se and the client apparatus Ce.

[0086] The client Ce transmits a data request including header hel indicative of data requesting and transmission request amount data r1 indicating an amount A which corresponds to the size of the empty area of the network buffer memory 42 at that instance.

[0087] Then, the client apparatus Ce intermittently transmits following data requests including header he2 to he5 and transmission request amount data r2 to r5. The transmission request amount data r1 to r5 indicates maximums of requested transmission amounts are A to E.

[0088] When the server apparatus Se receives the data request, the server apparatus Se analyses the transmission request amount data r1 and determines the transmission amount d11 of the coded data in the burst by the transmission amount determining circuit 2301 and the transmission circuit 2301 transmits the burst d11 of which amount of data is Ax which is less than the maximum of the requested transmission amount A. In response to the following transmission request amount data r2 to r5, the transmission circuit 2301 transmits the bursts d12 to d15 of which amounts of data are Bx to Ex which are less than the maximums of the requested transmission amounts B to E. Therefore, there is a relation:

$$A > Ax,\ B > Bx,\ C > Cx,\ D > Dx,\ E > Ex$$

[0089] In this embodiment, the client apparatus Ce transmits the data request including the request amount data corresponding to the size of the empty area of the network buffer memory 42 and the server apparatus Se transmits a burst of which data amount is less than the requested transmission amount but corresponding to the requested transmission amount, so that if there is a much empty area in the network buffer memory 42, the more coded data is transmitted from the server apparatus Se to the client apparatus Ce. Therefore, jitter due to delay in transmission data is surely prevented.

[0090] A modification of the fifth embodiment will be described.

[0091] The server apparatus Se measures an amount of the data transferred to the decoding buffer memory 44 from when the client apparatus Ce transmits the data request to when the server apparatus Se transmits the burst. Then, the server apparatus Se adds the measured amount of the data to the transmission request amount (r1-r5) and then, the server apparatus Se transmits a burst of which data amount is less than the added amount.

(SIXTH EMBODIMENT)

[0092] Fig. 25 is a block diagram of a server apparatus Sf of a sixth embodiment. Fig. 26 is a block diagram of a client apparatus Cf of the sixth embodiment.

[0093] The data transmission system of the sixth embodiment is substantially the same as that of the fifth embodiment. The difference is that a buffer memory observing circuit 243 of the client apparatus Cf transmits to a server apparatus Sf a data request including request number data of which number increases every data request and a transmission number data generation circuit 2502 generates transmission number data indicative a natural number which increases every transmission of the burst and the transmission circuit 2501 adds the transmission number data to coded data in the burst.

[0094] Figs. 10A and 10B are timing charts of the sixth embodiment showing transmission operation between the server apparatus Sf and the client apparatus Cf.

[0095] The client apparatus Cf transmits a data request including a header hel indicative of the data request and transmission number data rnl indicative of request number 1 when the empty area of the network buffer memory 42 is higher than the reference. Then, the client apparatus Cf intermittently transmits following data requests including header he2 to he5 and request number data rn2 to rn5 respectively.

[0096] When the server apparatus Sf receives the data request, the transmission number data generation circuit 2502 generates transmission number data n1-n5 and the transmission circuit 2501 adds the transmission number data n1-n5 to coded data d11-d15.

[0097] In this embodiment, the client apparatus Cf can detect which data request does the received burst respond to, so that if one of successive burst of coded data is delayed due to busy traffic, the client apparatus can rearrange the bust of the coded data to correctly reproduce the video data and the audio data. Moreover, there is the case that a data request is transmitted before the coded data is transmitted from the server apparatus to the client apparatus Cf in response to the previous data request. In such a case, the client apparatus Cf can judge which data request does the received burst responds to, so that receiving bursts of data is controlled.

(SEVENTH EMBODIMENT)

[0098] Fig. 27 is a block diagram of a server apparatus Sg of a seventh embodiment. Fig. 28 is a block diagram of a client apparatus Cg of the seventh embodiment.

[0099] The data transmission system of the seventh embodiment is substantially the same as that of the fourth embodiment. The difference is that the buffer memory observing circuit 343 generates a test signal TEST and transmitting the test signal TEST to the server Sg when the empty area is higher than the reference, a transmission control circuit 2802 receives the test signal TEST in addition to data request and immediately transmits a response signal re at least once and the buffer memory observing circuit 343 measures a turnaround interval, that is, from transmission of the test signal TEST to receiving the response signal re and determines an amount of the coded data in a burst in accordance with the measured turnaround interval.

[0100] Figs. 11A and 11B are timing charts of the seventh embodiment showing transmission operation between the server apparatus Sg and the client apparatus Cg.

[0101] The client Cg transmits the test signal TEST with the header he1 when the empty area of the network buffer memory 42 is higher than the reference. Then, the server apparatus Sg transmits a response signal re with a header.

[0102] The buffer memory observing circuit 343 receives the response signal re and measures the turnaround interval and determines the request amount of coded data and transmits request amount data AMT with a header he2 to the server apparatus Sg. The server apparatus Sg transmits the burst of which data amount is determined in accordance with the request amount data AMT.

Modifications will be described.

[0103] In this embodiment, as shown in Fig. 28, the client apparatus Cg may further include a buffer memory control circuit 2801 for controlling a capacity of the network buffer memory 42 in accordance with the measured turnaround interval or a transmission bit rate.

[0104] In this embodiment, the test signal TEST is transmitted when the empty area of the network buffer memory is higher than the reference. However, it is also possible that the test signal TEST is transmitted when the empty area of the network buffer memory is higher than the reference every when a predetermined interval has passed.

[0105] Figs. 12A and 12B are timing charts of a modification in the seventh embodiment showing transmission operation between the server apparatus and the client apparatus.

[0106] The test signal TEST is transmitted from the client apparatus Cg twice and the server apparatus Sg receives every test signal TEST and transmits a response signal re in response to every reception of the test signal TEST. Then, the buffer memory observing circuit 343 averages the measured turnaround intervals. Moreover, the buffer memory observing circuit 343 may effect other statistic operation to obtain a variance for example to determine determine the turn around interval and to estimate an amount of jitter.

[0107] In this modification, the test signal TEST is transmitted from the client apparatus Cg twice. However, it is also possible to the test signal TEST is transmitted from the client apparatus Cg more than twice.

(EIGHTH EMBODIMENT)

[0108] Fig. 29 is a block diagram of a server apparatus Sh of an eighth embodiment. Fig. 30 is a block diagram of a client apparatus Ch of the eighth embodiment.

[0109] The data transmission system of the eighth embodiment is substantially the same as that of the seventh embodiment. The difference is that the test signal TEST is transmitted from the server apparatus Sh to the client apparatus Ch and the response signal re is transmitted from the client apparatus Ch to the server apparatus Sh.

[0110] More specifically, the transmission control circuit 2902 generates a test signal TEST and transmitting the test signal TEST to the Client apparatus Ch in response to a program request signal from the client apparatus Ch. The buffer memory observing circuit 434 transmits a response signal re to the server apparatus Sh in response to the test signal TEST from the server apparatus Sh. The transmission control circuit 2902 receives the response signal re and measures a turnaround interval. This operation is repeated to obtain the turnaround interval 2.

[0111] Figs. 13A and 13B are timing charts of the eighth embodiment showing transmission operation between the server apparatus Sh and the client apparatus Ch.

[0112] The transmission control circuit 2902 generates the test signal TEST and transmitting the test signal TEST to the client apparatus Ch in response to the program request signal from the client apparatus Ch. The buffer memory observing circuit 434 transmits a response signal re with data BU indicative of the capacity of the memory buffer memory 42 or the like to the server apparatus in response to the test signal TEST from the server apparatus Sh. The transmission control circuit 2902 receives the response signal re and measures a turnaround interval 1. This operation is repeated to obtain the turnaround interval by effecting a statistic operation from the turnaround interval 1 and the turnaround interval 2. The transmission control circuit 2902 calculates the transmission bit rate in accordance with data BU indicative of the capacity of network buffer memory 42 and compensates request amount data from the client apparatus Ch in accordance with the transmission bit rate. That is, the transmission control circuit 2902 determines the amount of data in the network buffer memory 42 in response to a data request, wherein the bit rate is limited under the calculated transmission bit rate.

[0113] Setting the bit rate to a suitable value determined in accordance with the delay in the network and the capacity of the network buffer memory 42 prevents the fluctuation of the delay and the jitter.

[0114] When the capacity of the network buffer memory 42 is fixed, a degree of absorbing jitter depends on the bit rate. For example if there is a jitter of one second peak to peak and there is the capacity of 4M bits in the network buffer memory 42, the maximum bit rate is less than 4 Mbit/ sec to absorb jitters. That is, in this embodiment, if the delay interval is long and jitters occur frequently, the bit rate is limited, i.e., the resolution is made low.

[0115] The client Ch transmits the test signal TEST with the header hel when the empty area of the network buffer memory 42 is higher than the reference. Then, the server apparatus Sh transmits a response signal re with a header.

[0116] The buffer memory observing circuit 443 receives the response signal re and measures the turnaround interval and determines the request amount of coded data and transmits request amount data AMT with a header he2 to the server apparatus Sh. The server apparatus Sh transmits the burst of which data amount is determined in accordance with the request amount data AMT.

(NINTH EMBODIMENT)

[0117] Fig. 31 is a block diagram of a server apparatus Si of a ninth embodiment. Fig. 14 is a block diagram of a client apparatus Ci of the ninth embodiment.

[0118] The data transmission system of the ninth embodiment is substantially the same as that of the fourth embodiment. The difference is that the server apparatus Si transmits the video data and the audio data independently and the client apparatus Ci of the ninth embodiment can receive the video data and the audio data from the same or different server apparatus Si at the same time.

[0119] More specifically, the server apparatus Si includes a memory 1701 for storing data of at least one of coded video data and coded audio data, a transmitting circuit 3102 for intermittently transmitting a burst including the data read from the memory 1701 to the network system in response to a data request, and a network interface circuit 1703, a video data transmission control circuit 3103, and an audio data transmission control circuit 3104.

[0120] The video data transmission control circuit 3103 operates the transmission circuit 3102 and the memory 1701 to transmit the video data in response to a video data request and independently the audio data transmission control circuit 3104 operates the transmission circuit 3102 and the memory 1701 to transmit the audio data in response to an audio data request.

[0121] The client apparatus Ci includes a network interface circuit 61 with a separator, clock generation and counter circuit 70, and a PLL circuit and counter 71, a video system, and an audio system. The video system includes a video network buffer memory 62 for receiving the transmitted video data in the burst directing to the client apparatus Ci from the network interface circuit 61, a buffer memory observing circuit 63 for detecting an empty area of the stored video data in the video network buffer memory 62 and transmitting the data request to the server apparatus Si when the empty area is higher than a reference, a decoding buffer 66 for storing the data from the video network buffer memory 62, a video decoder 67 for decoding the video data from the decoding buffer memory 66, and a video reproducing circuit 72 for reproducing the decoded video data from continuously in time base.

[0122] The audio system includes a audio network buffer memory 64 for receiving the transmitted audio data in the burst directing to the client apparatus Ci from the network interface circuit 61, a buffer memory observing circuit 65 for detecting an empty area of the stored audio data in the audio network buffer memory 64 and transmitting an audio data request to the server Si2 when the empty area is higher than a reference, a decoding buffer memory 68 for storing the audio data from the audio network buffer memory 64, and an audio decoder 69 for decoding the audio data from the decoding buffer memory 68, and an audio reproducing circuit 73 for reproducing the decoded audio data continuously in time base.

[0123] The network interface circuit 61 receives the video data and the audio data independently, that is, receives the video data and the audio data from different server apparatus Si1 and Si2 and supplies the received video data to the video network buffer memory 62 and supplies the received audio data to the audio network buffer memory 64, so that the video data and the audio data can be received from the different server apparatus Si1 and Si2.

[0124] Figs. 15A to 15D are timing charts of the ninth embodiment showing the video data transmission operation between the server apparatus Si1 and the client apparatus Ci1 and the audio data transmission operation between the server apparatus Si2 and the client apparatus Ci1.

[0125] In the above-mentioned embodiment the fourth to ninth embodiments and their modifications can be combined each other.

Claims

1. A data transmission system comprising:

   at least one server apparatus having a memory storing coded data of at least one of coded moving picture image data and coded audio data and transmitting means for incontinuously transmitting said data read from said memory in every burst to a network system; and

   at least client apparatus including receiving means for receiving the transmitted data in said burst directing to the client apparatus and decoding means for decoding the received data to be continuously reproduced in time base, wherein said server apparatus further comprises server's time data generation means for generating server's time data indicative of time of the next burst to provide continuously reproducing said moving picture image data and said audio data and data attaching means for attaching said server's time data to said data in said burst and said client apparatus further comprises extraction means for extracting said server's time data from the received data, a client's time data generation means for generating client's time data indicative of present time of said client apparatus, and data requesting means for transmitting a request for the next burst at a request timing determined in accordance with said server's time data from said extraction means and present time indicated by said client's time data.

2. The data transmission system as claimed in claim 1, wherein said decoding and reproducing means decodes in response to said client's time data and said data requesting means comprises a subtractor for obtaining a difference between said time data and said reference clock signal when said server's time data is received, request timing determining means for compensating a relation between said client's time data and the received server's time data in accordance with said difference and determining said request timing in accordance with said client's time data and the compensated relation, and request transmission means responsive to said request timing determining means for generating and transmitting said request.

3. A data transmission system comprising:

   at least one server apparatus having a memory storing coded moving picture image data and coded audio data and transmitting means for incontinuously transmitting a video burst including said coded moving picture image data and an audio burst including coded audio data read from said memory to a network system respectively; and

   at least client apparatus including receiving means for receiving said video and audio bursts directing to said client apparatus and decoding means for decoding said coded moving picture image data in said video burst and said coded audio data in said audio burst to be reproduced continuously in time base,

   wherein said server apparatus further comprises video time data generation means for generating video time data indicative of transmission time of the next video burst to provide continuously reproducing said moving picture image data and said audio time data generation means for generating audio time data indicative of transmission time of the next audio burst, and data attaching means for attaching said video time data to said coded moving picture image data in said video burst and for attaching said audio time data to said coded audio data in said audio burst, and said client apparatus further comprises extraction means for extracting said video time data from said video burst from said receiving means and said audio time data from said audio burst from said receiving means, client's time data generation means for generating a client reference time data, and first data requesting means for transmitting a first request for transmitting the next video burst in accordance with the extracted first server's time data and time indicated by said client's reference time data and second data requesting means for transmitting a second request for transmitting said the next audio burst in accordance with the extracted audio time data and present time indicated by said client's reference time data.

4. The data transmission system as claimed in claim 3, wherein said decoding and reproducing means decodes said data in response to said reference clock signal and said first data requesting means comprises a first subtractor for obtaining a first difference between said first server's time data and said reference clock signal when said first server's time data is received, first compensating means for compensating a first relation between said reference clock signal and the received first server's time data in accordance with said first difference, and first requesting timing determining means for determining said first request timing in accordance with said time data and the compensated first relation and said second data requesting means comprises a second subtractor for obtaining a second difference between said second time data and said reference clock signal when said second time data is received, second compensating means for compensating a second relation between said reference clock signal and the received second time data in accordance with said second difference, and second requesting timing determining means for determining said second request timing in accordance with said second time data and the compensated second relation.

5. A data transmission system comprising:

at least one server apparatus having a memory storing data of at least one of coded moving picture image data and coded audio data and transmitting means for intermittently transmitting a burst including said data read from said memory to a network; and

at least client apparatus including receiving means for receiving the transmitted burst directing to said client and decoding means for decoding the received data in the received burst to be reproduced continuously in time base,

wherein said server apparatus further comprises time interval data generation means for generating time interval data indicative of time interval of successive coding to provide continuously reproducing said moving picture image data and said audio data and data attaching means for attaching said time interval data to said data in said burst and said client apparatus further comprises extraction means for extracting said time interval data from the received data, a clock circuit means for generating a reference clock signal, and data requesting means for periodically transmitting a request for transmitting the next burst at a timing determined in accordance with said time interval data and time indicated by said reference clock signal.

6. A data transmission system comprising:

at least one server apparatus having a memory storing data of at least one of coded moving picture image data and coded audio data and transmitting means for intermittently transmitting a burst including said data read from said memory to a network in response to data request; and

at least client apparatus including data request transmission means for transmitting said data request, receiving means for receiving the transmitted data in said burst directing to said client apparatus and decoding means for decoding said data from said receiving means to be reproduced continuously in time base,

wherein said client apparatus further comprises network buffer means for temporally storing the received data and supplying the temporally stored data to said decoding means, buffer observing means for detecting an empty area of said network buffer means regarding the stored data, and request transmitting means transmitting said data request to said server when said empty area is higher than a reference and said server apparatus further comprises request receiving means for receiving said data request from said client and operating said transmitting means to transmit the next burst in response to said data request.

7. The data transmission system as claimed in claim 6, wherein said request transmitting means transmits said data request including empty area size data indicative of a size of the detected empty area and said transmission means controls an amount of data of said coded data to be transmitted in accordance with said capacity of said empty area.

8. The data transmission system as claimed in claim 6, wherein said client apparatus further comprises request number data generating means responsive to said buffer memory observing means for generating request number data indicative of successively increasing number when said empty area is higher than said reference, said request transmitting means further transmits said request number data with said transmission request, and said server apparatus transmits the received request number data with said data requested by said transmission request to said client apparatus.

9. The data transmission system as claimed in claim 6, wherein said buffer observing means generates a test signal and transmitting said test signal to said server when said empty area is higher than said reference, said server apparatus further comprises test means for receiving said test signal and immediately transmitting a response signal at least once, said client apparatus further comprises data buffer control means for controlling a capacitance of said data buffer means, response signal receiving means for receiving said response signal and measuring means for measuring an interval between when said test signal is transmitted and when said response signal is received, said data buffer observing means further generates amount control data in accordance with said interval and transmits said amount control data with said next burst transmission request, said request receiving means in said server apparatus controls transmitting said burst including said coded data at an amount determined in accordance with said amount control data, and said data buffer control means controls said capacitance of said data buffer means in accordance with said interval.

10. The data transmission system as claimed in claim 9, wherein said client apparatus further comprises statistically processing means for statistically processing data of said interval and supplies the processed data of said interval to said data buffer observing means and said data buffer means.

11. The data transmission system as claimed in claim 9, wherein said server apparatus further comprises test signal generation means for generating a test signal and transmitting said test signal to said server in response to said data request, calculation means, and compensation means, said client apparatus further comprises test means for receiving said test signal and immediately transmitting a response signal including data transmitting amount data and capacity data of said data buffer means, said server apparatus further comprises response signal receiving means for receiving said response signal and measuring means for measuring an interval between when said test signal is transmitted and when said response signal is received, said calculation means calculates a data rate from the said interval and said capacity data and said compensation means compensates the data amount data from said response signal receiving means in accordance with said data rate from said calculation means.

12. The data transmission system as claimed in claim 11, wherein said client apparatus further comprises statistically processing means for statistically processing data of said interval data and supplies the processed data of said interval to said calculation means.

13. The data transmission system as claimed in claim 6, wherein said data transmission system comprises first and second server apparatus, said client apparatus further comprises separating means for separating the received data into said coded moving picture image data and said coded audio data at the same time, said buffer means comprises first and second buffer circuits for temporally storing the separated coded moving picture image data and the separated coded audio data respectively, first and second data buffer observing means for observing first and second empty area of said first and second buffer circuits, and request transmitting means includes first and second request transmitting circuits for respectively transmitting first and second next burst transmission requests to said first and second server apparatus in response to said first and second data buffer observing means independently of each other.

14. The data transmission system as claimed in claim 6, wherein said transmitting means transmits said nest burst including said data of a data amount less than a predetermined value.

15. The data transmission system as claimed in claim 7, wherein said client apparatus further comprising number data generation means responsive to said data buffer observing means for generating request number data indicative of successively increasing number when said empty area is higher than said reference and, said request transmitting means further transmitting said request number data with said transmission request and said server apparatus transmits the received request number data in said next burst.

16. The data transmission system as claimed in claim 7, wherein said data transmission system comprises first and second server apparatus, said client apparatus further comprises separating means for separating the received data into said coded moving picture image data and said coded audio data at the same time, said buffer means comprises first and second buffer circuits for temporally storing the separated coded moving picture image data and the separated coded audio data respectively, first and second data buffer observing means for observing first and second empty area of said first and second buffer circuits, and request transmitting means includes first and second request transmitting circuits for respectively transmitting first and second next burst transmission requests to said first and second server apparatus in response to said first and second data buffer observing means independently of each other.

17. A reproduction apparatus to be connected to a network to receive burst data including server's time data indicative of time of next transmitting said burst data for continuously reproducing said moving picture image data and said audio data and at least one of coded moving picture image data and coded audio data, comprising:

receiving means for receiving the said burst data directing to the same from said network;
extraction means for extracting said server's time data from the received burst data;
reference clock means for generating a reference clock signal;
data requesting means for transmitting a request for transmitting the next burst data at a request timing determined in accordance with said server's time data from said extraction means and time indicated by said reference clock signal;
decoding and reproducing means for decoding and reproducing the received data continuously in time base in response to said reference clock signal.

18. A reproduction apparatus to be connected to a network to receive burst data at least one of coded moving picture image data and coded audio data, the top of burst data transmitted in response to data request from the same including interval data indicative of periodically transmitting said burst data comprising:

receiving means for receiving the said burst data directing to the same from said network;
extraction means for extracting said interval data from said top of burst data from said receiving means;
reference clock means for generating a reference clock signal;
data requesting means for transmitting a request for transmitting the next burst data at a request timing determined in accordance with said interval data from said extraction means and time indicated by said reference clock signal; and
decoding and reproducing means for decoding and reproducing the received data continuously in time base in response to said reference clock signal.

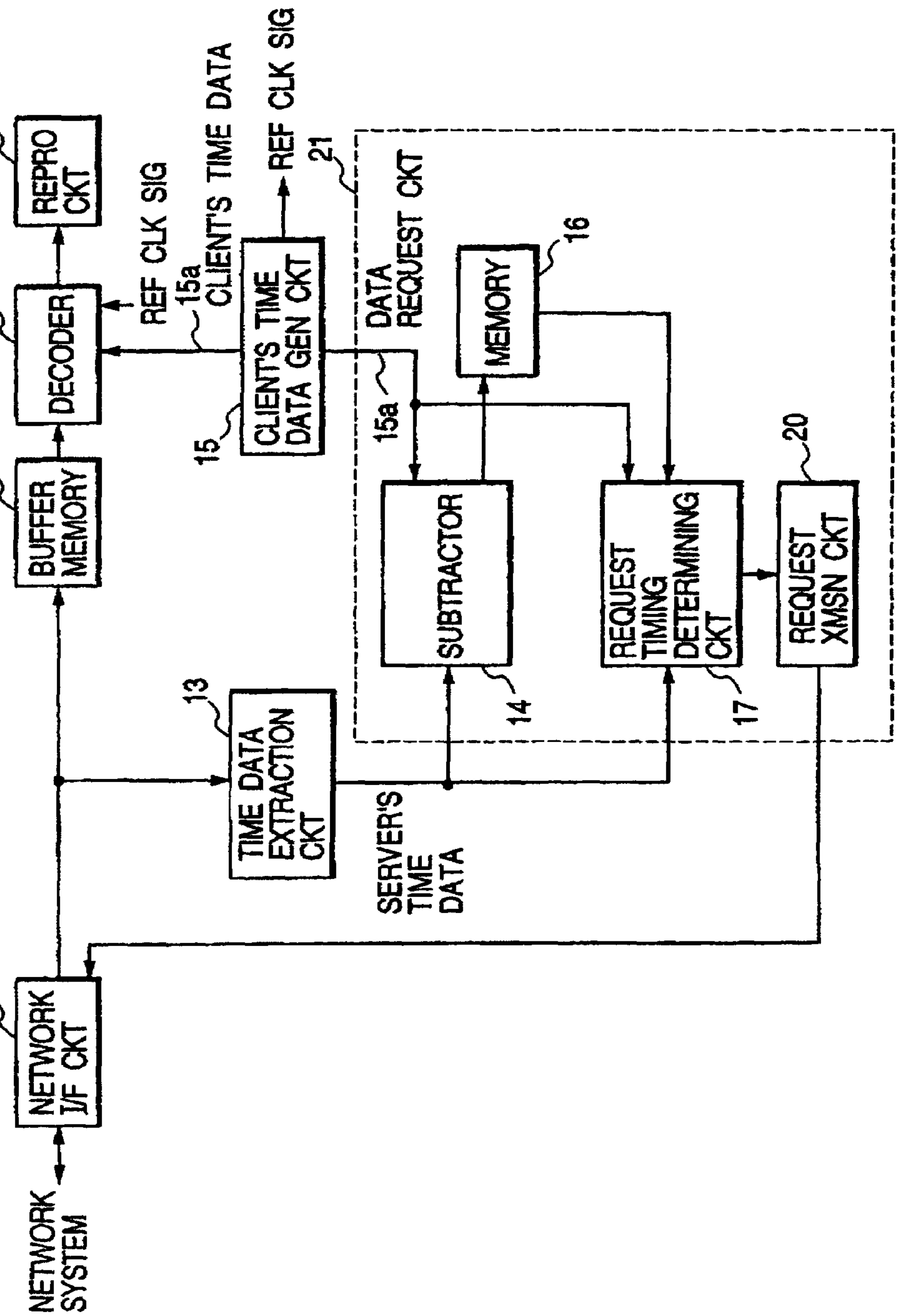
19. A reproduction apparatus to be connected to a network to receive burst data at least one of coded moving picture image data and coded audio data, comprising:

receiving means for receiving the said burst data directing to the same from said network;
reference clock means for generating a reference clock signal;
data buffer means for temporally storing the received burst data and supplying the temporally stored data;
decoding and reproducing means for decoding and reproducing the received burst data from said data buffer continuously in time base in response to said reference clock signal;
request transmission means for observing an empty area of the stored data and transmitting a next burst transmission request to said server when said empty area is higher than a reference.

20. The reproduction apparatus as claimed in claim 19, further comprising generating data demanding amount data corresponding to said reference, said request transmission means further transmitting said data demanding amount data with said next burst transmission request.

21. The reproduction apparatus as claimed in claim 19, further comprising request number data generating means for generating request number data indicative of successively an increasing number and said request transmitting means further transmitting said request number data with said transmission request

FIG. 1


Ca
NETWORK SYSTEM
NETWORK I/F CKT
11
BUFFER MEMORY
12
TIME DATA EXTRACTION CKT
13
SERVER'S TIME DATA
DECODER
18
REPRO CKT
19
REF CLK SIG
15a
CLIENT'S TIME DATA
CLIENT'S TIME DATA GEN CKT
15
REF CLK SIG
21
DATA REQUEST CKT
15a
MEMORY
16
SUBTRACTOR
14
REQUEST TIMING DETERMINING CKT
17
REQUEST XMSN CKT
20

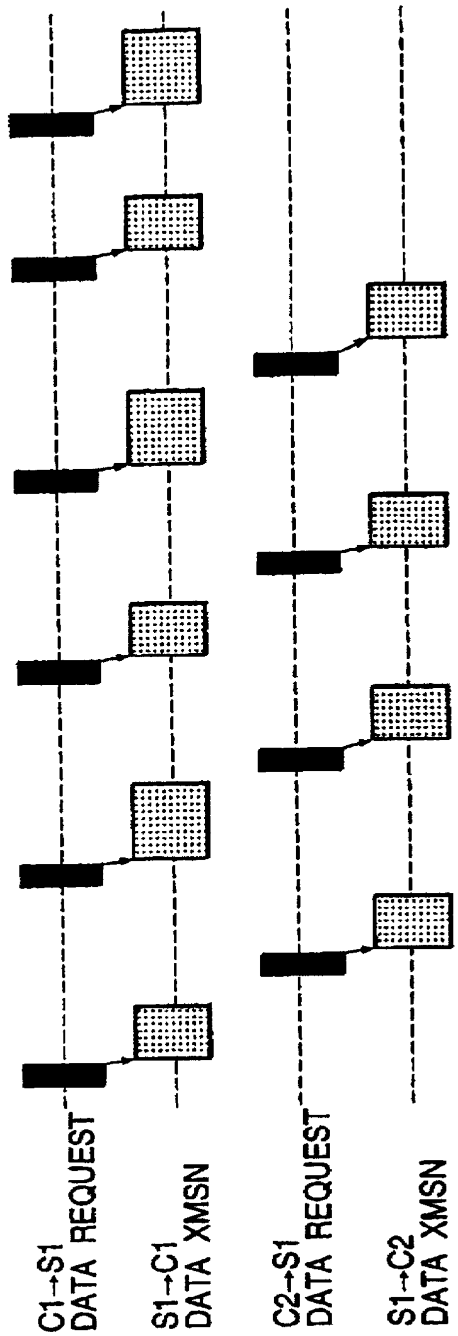
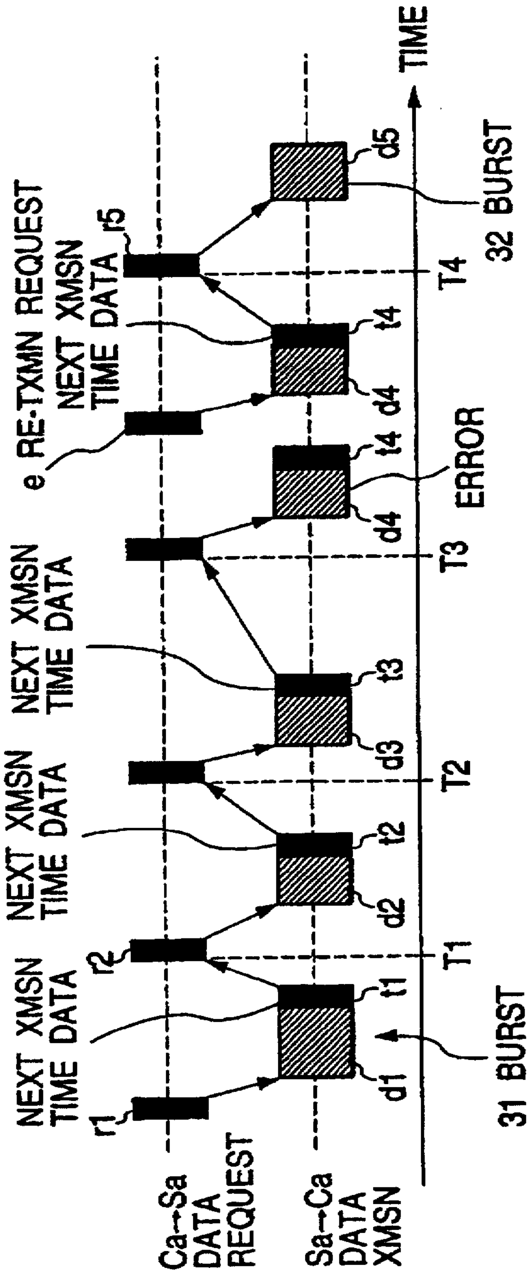


FIG. 2A
FIG. 2B
FIG. 2C
FIG. 2D

IPR2022-01227
EXHIBIT 1003 - PAGE 00747

FIG. 3A

FIG. 3B



IPR2022-01227
EXHIBIT 1003 - PAGE 00748

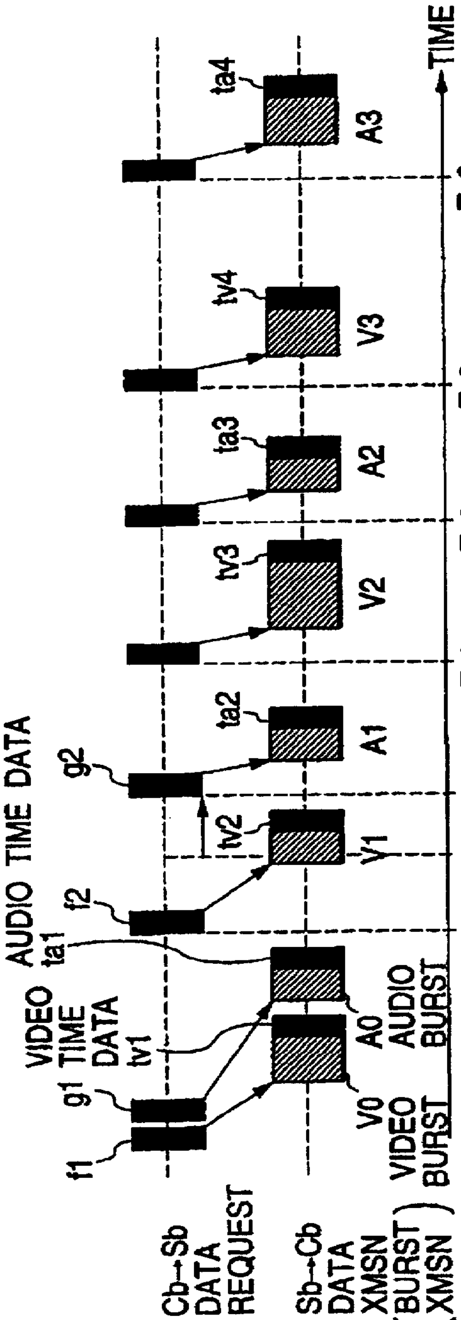
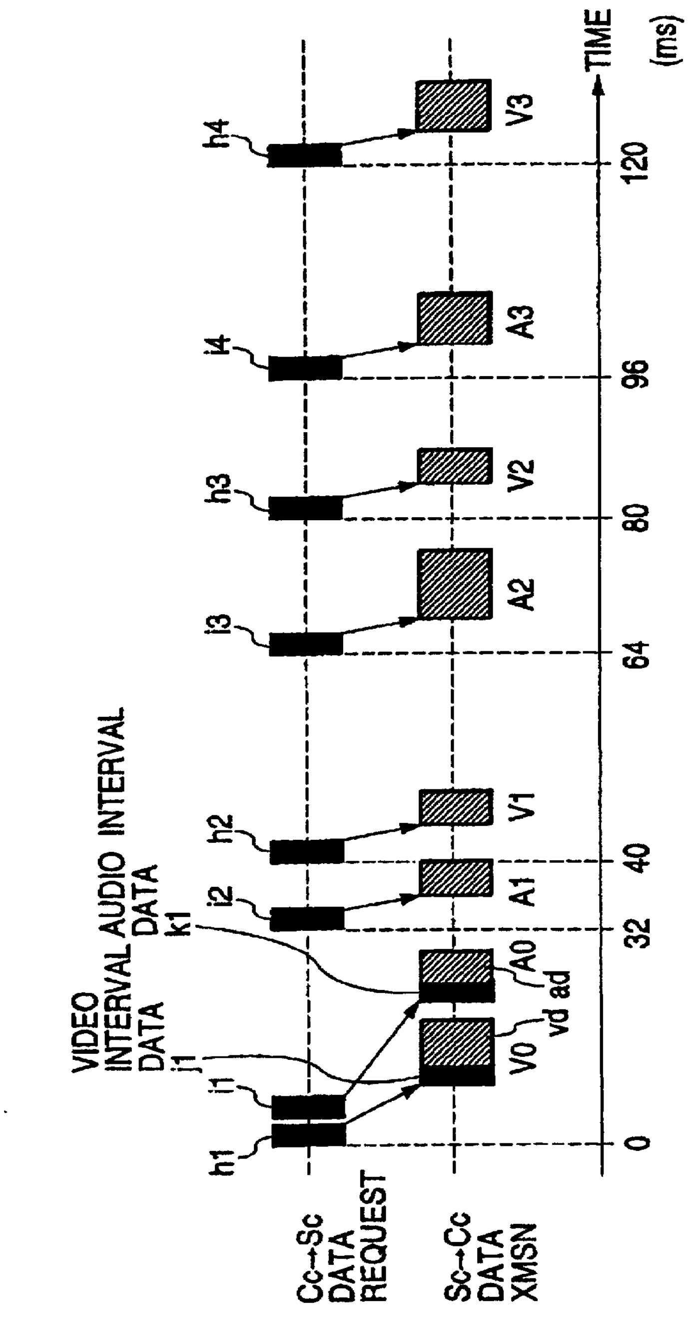


FIG. 4A

FIG. 4B

IPR2022-01227
EXHIBIT 1003 - PAGE 00749



VIDEO INTERVAL DATA j1
AUDIO INTERVAL DATA k1
Cc→Sc DATA REQUEST
Sc→Cc DATA XMSN
h1
i1
i2
h2
i3
h3
i4
h4
V0
vd
A0
ad
A1
V1
A2
V2
A3
V3
0
32
40
64
80
96
120
TIME
(ms)

FIG. 5A

FIG. 5B

FIG. 6

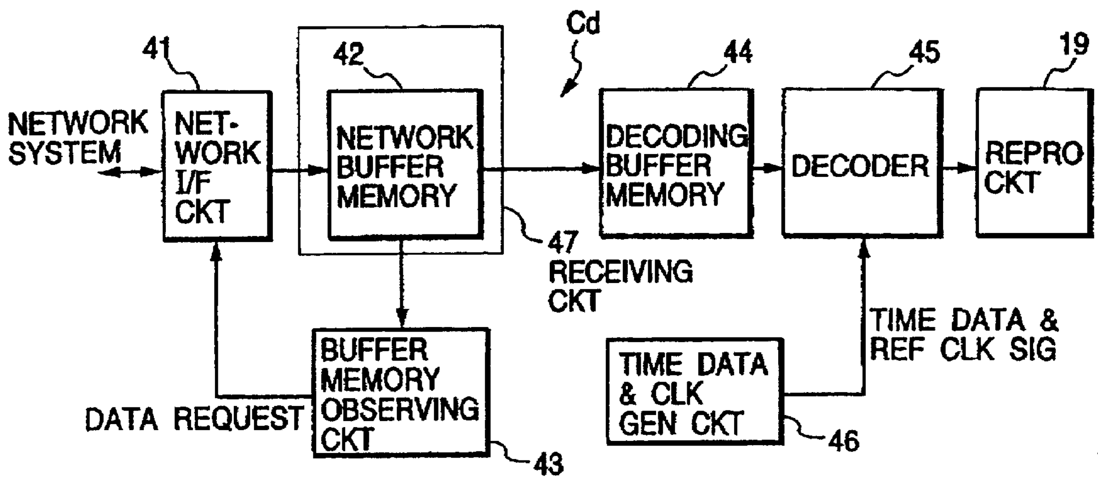
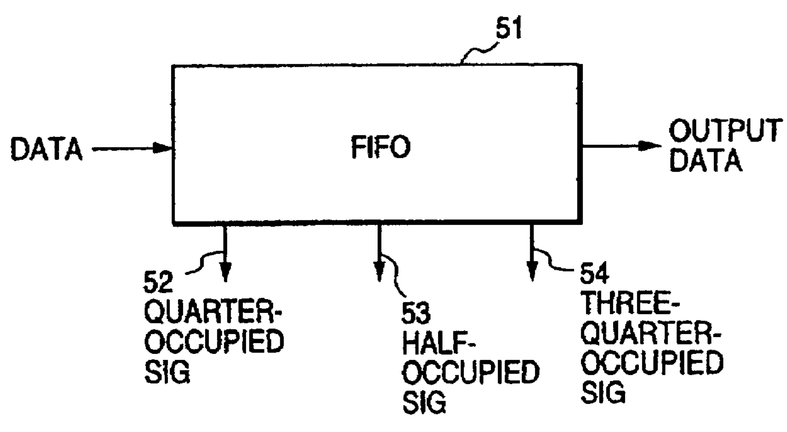

Cd
41
42
44
45
19
NETWORK SYSTEM
NET-WORK I/F CKT
NETWORK BUFFER MEMORY
DECODING BUFFER MEMORY
DECODER
REPRO CKT
47 RECEIVING CKT
TIME DATA & REF CLK SIG
BUFFER MEMORY OBSERVING CKT
DATA REQUEST
TIME DATA & CLK GEN CKT
46
43

FIG. 7


51
DATA
FIFO
OUTPUT DATA
52 QUARTER-OCCUPIED SIG
53 HALF-OCCUPIED SIG
54 THREE-QUARTER-OCCUPIED SIG

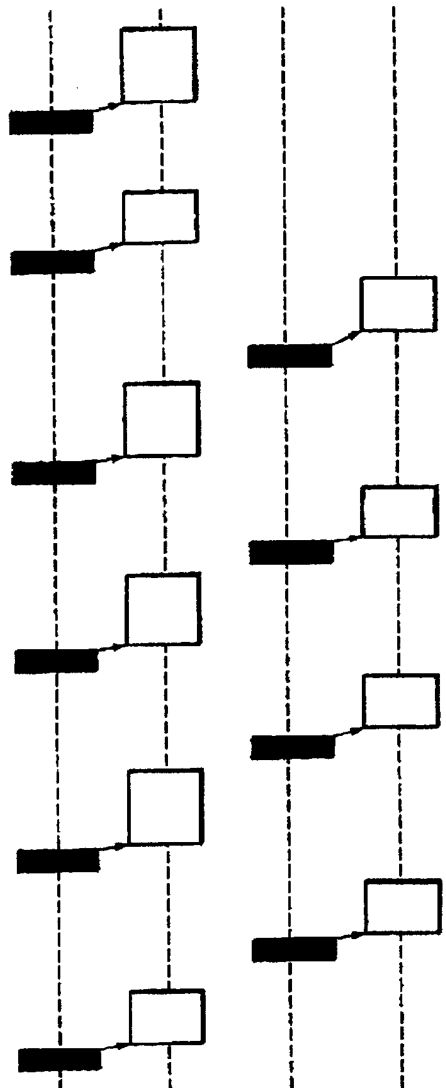


FIG. 8A Cd1→Sd1 DATA REQUEST

FIG. 8B Sd1→Cd1 DATA XMSN

FIG. 8C Cd2→Sd1 DATA REQUEST

FIG. 8D Sd1→Cd2 DATA XMSN

IPR2022-01227
EXHIBIT 1003 - PAGE 00752

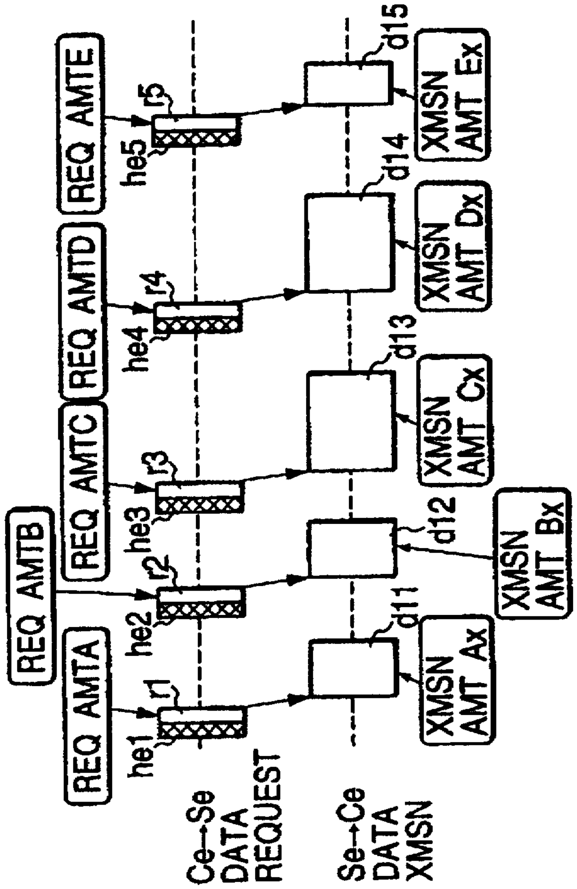

Ce→Se
DATA
REQUEST
Se→Ce
DATA
XMSN
REQ AMTA
REQ AMTB
REQ AMTC
REQ AMTD
REQ AMTE
he1
he2
he3
he4
he5
r1
r2
r3
r4
r5
d11
d12
d13
d14
d15
XMSN AMT Ax
XMSN AMT Bx
XMSN AMT Cx
XMSN AMT Dx
XMSN AMT Ex

FIG. 9A

FIG. 9B

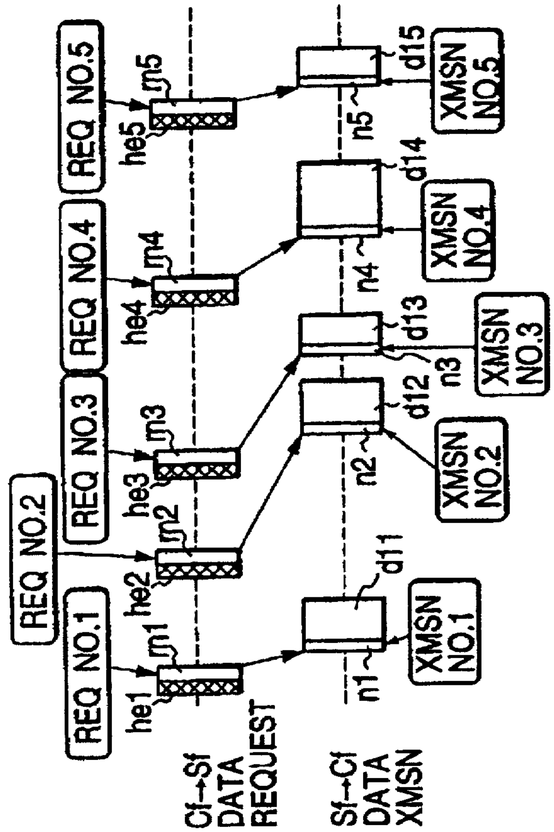


FIG. 10A

FIG. 10B

IPR2022-01227
EXHIBIT 1003 - PAGE 00754

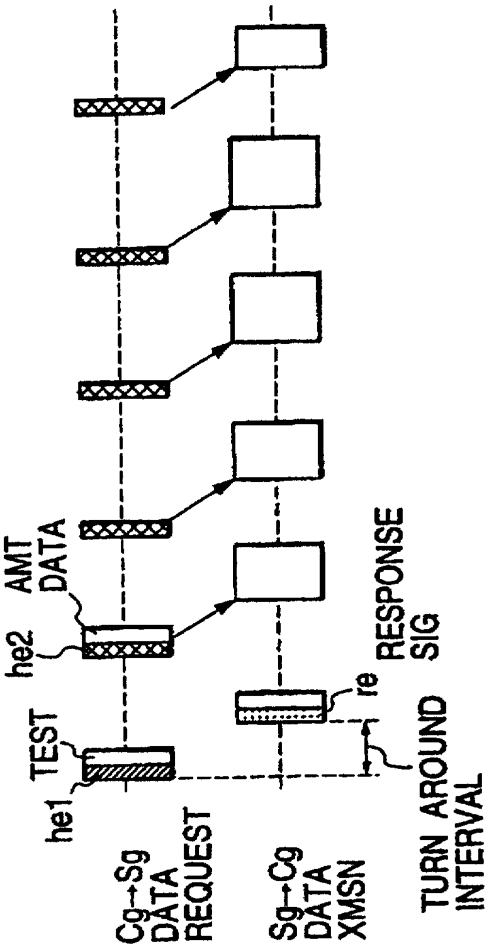


FIG. 11A

FIG. 11B

IPR2022-01227
EXHIBIT 1003 - PAGE 00755

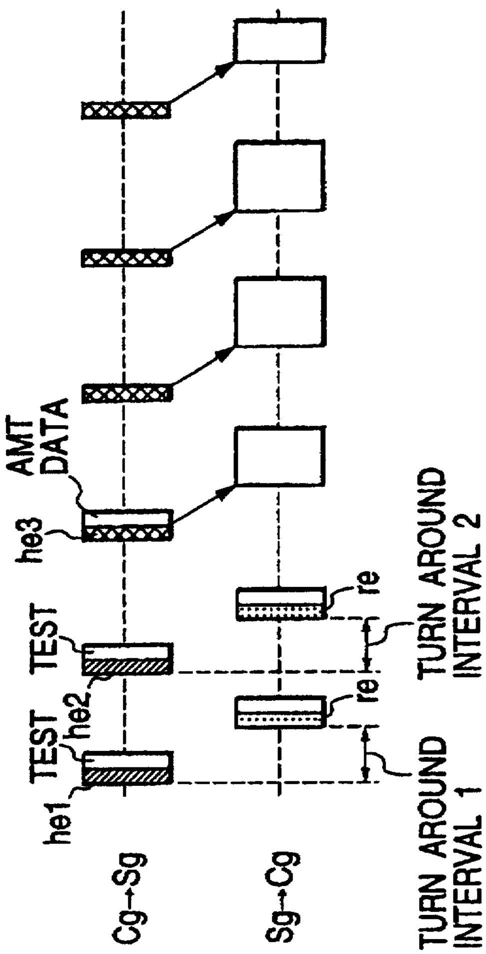
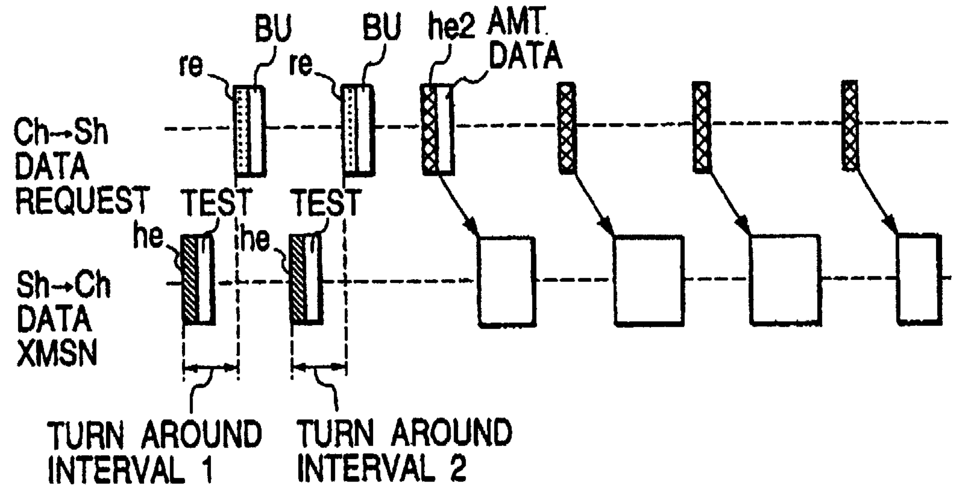


FIG. 12A

FIG. 12B

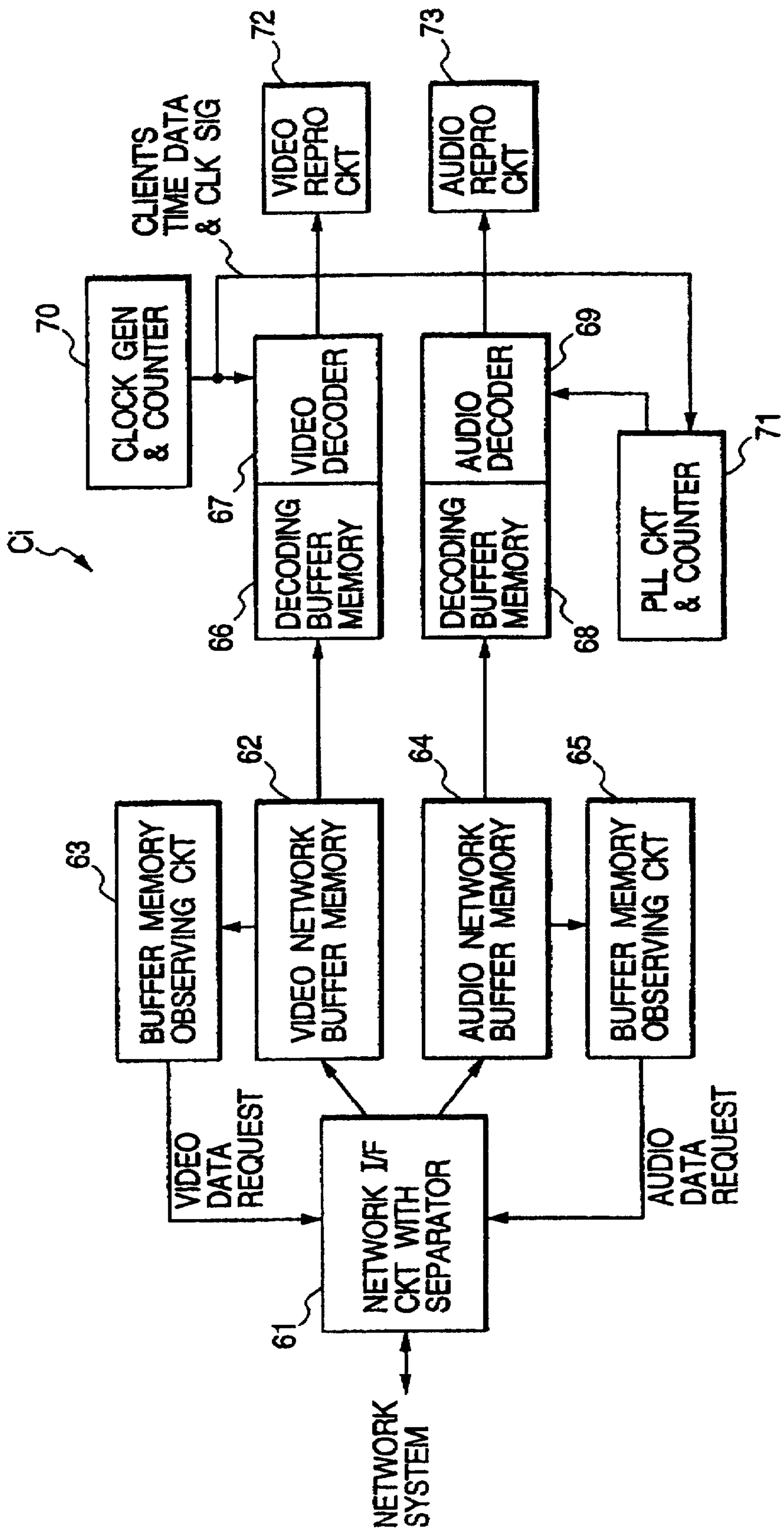
FIG. 13A

FIG. 13B



IPR2022-01227
EXHIBIT 1003 - PAGE 00757

FIG. 14


Ci
NETWORK SYSTEM
61
NETWORK I/F CKT WITH SEPARATOR
VIDEO DATA REQUEST
AUDIO DATA REQUEST
63
BUFFER MEMORY OBSERVING CKT
62
VIDEO NETWORK BUFFER MEMORY
64
AUDIO NETWORK BUFFER MEMORY
65
BUFFER MEMORY OBSERVING CKT
66
DECODING BUFFER MEMORY
67
VIDEO DECODER
68
DECODING BUFFER MEMORY
69
AUDIO DECODER
70
CLOCK GEN & COUNTER
71
PLL CKT & COUNTER
CLIENTS TIME DATA & CLK SIG
72
VIDEO REPRO CKT
73
AUDIO REPRO CKT

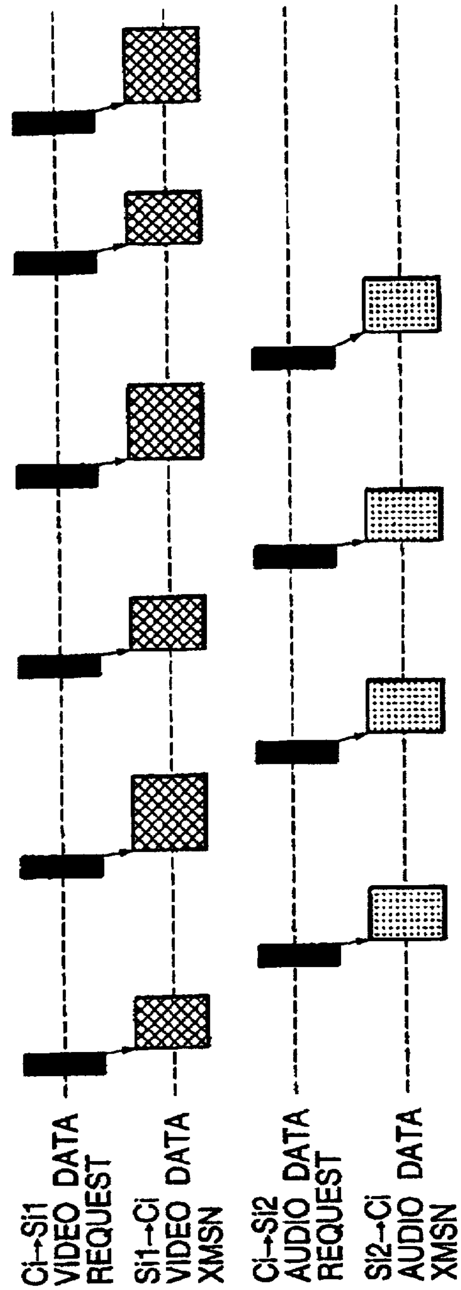


FIG. 15A

FIG. 15B

FIG. 15C

FIG. 15D

IPR2022-01227
EXHIBIT 1003 - PAGE 00759

FIG. 16A

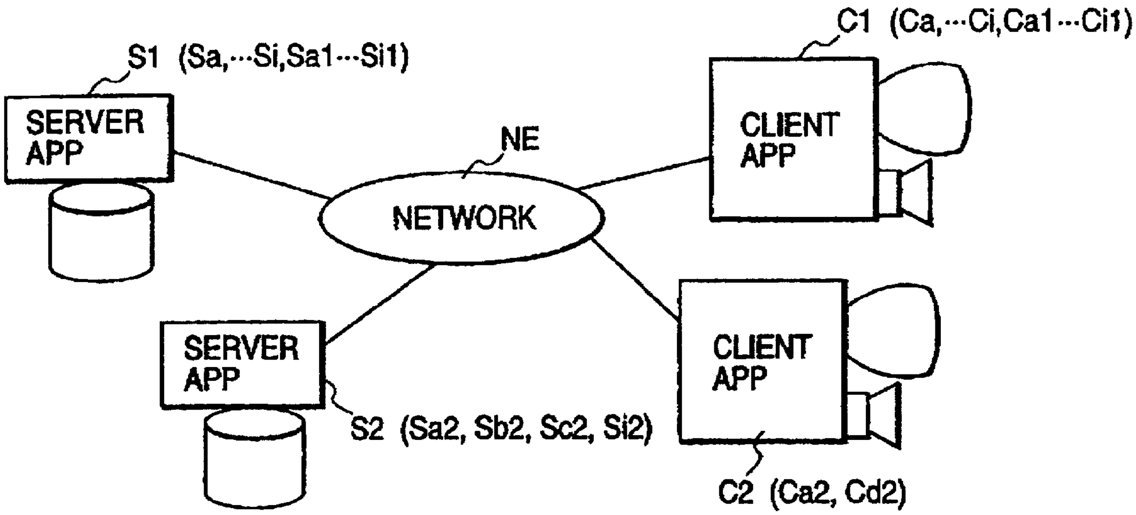


FIG. 16B

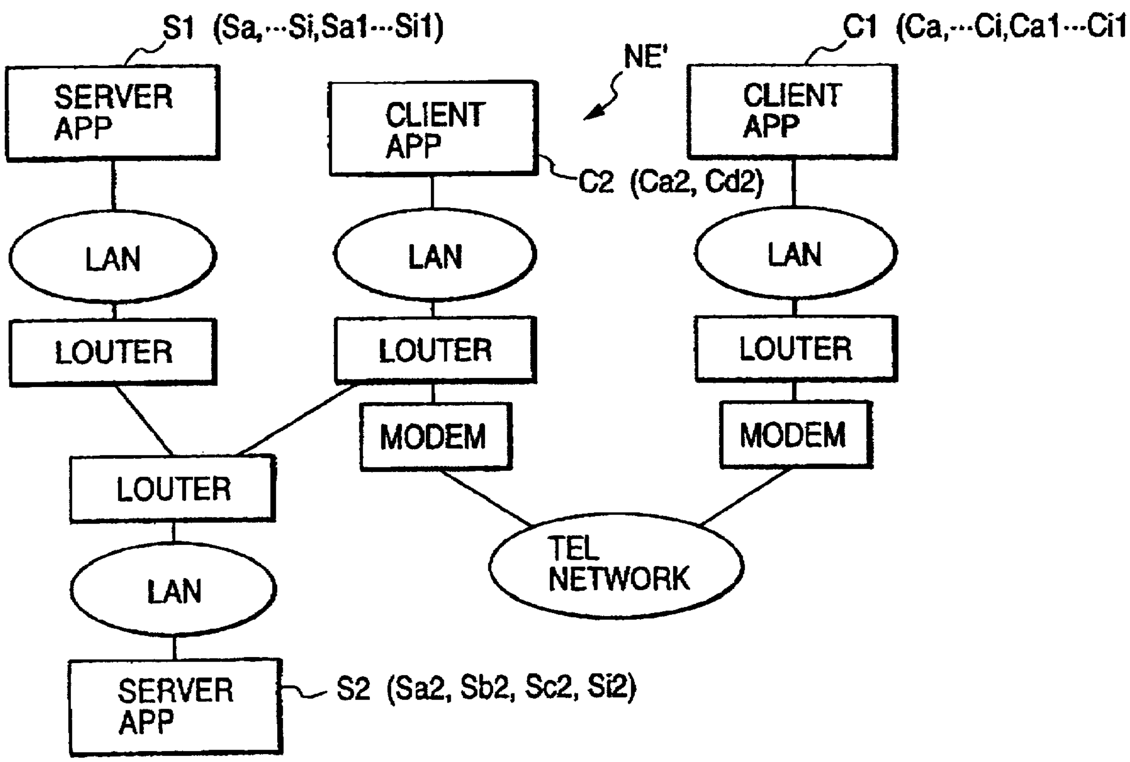


FIG. 17

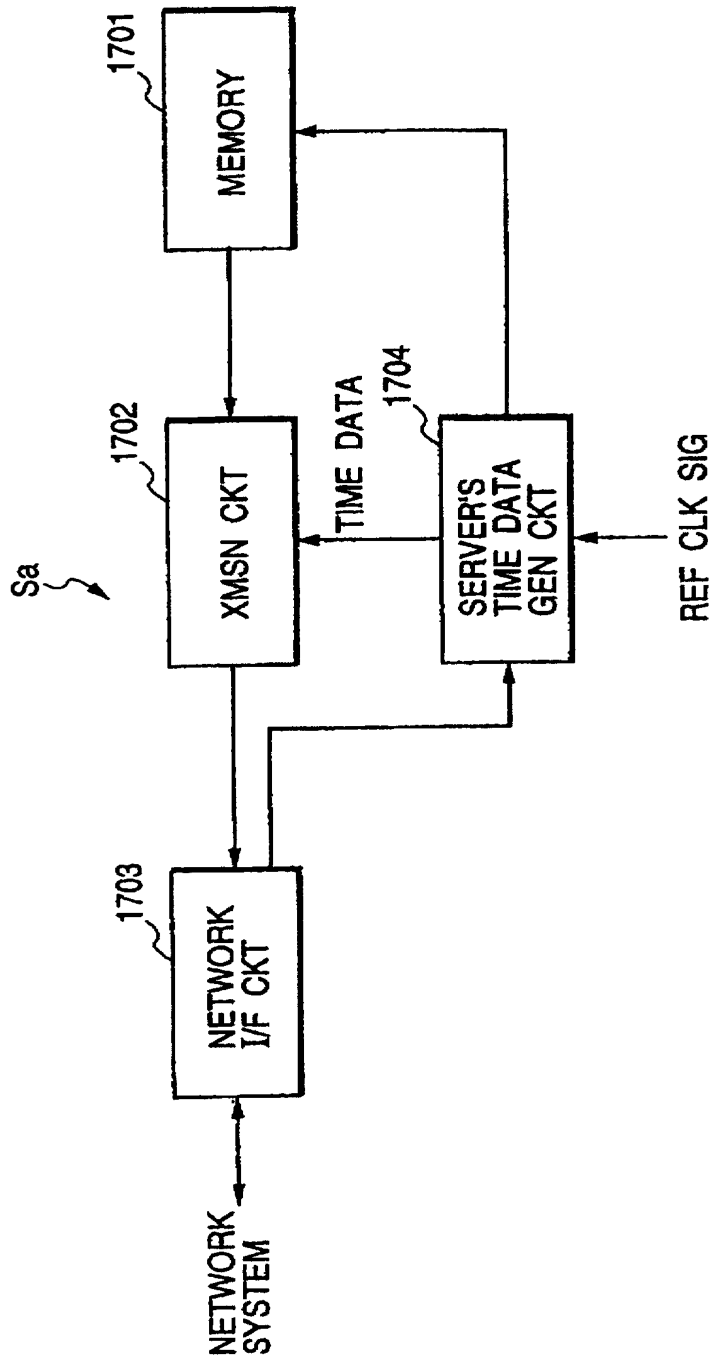

MEMORY
1701
XMSN CKT
1702
TIME DATA
1704
SERVER'S TIME DATA GEN CKT
REF CLK SIG
Sa
NETWORK I/F CKT
1703
NETWORK SYSTEM

FIG. 18

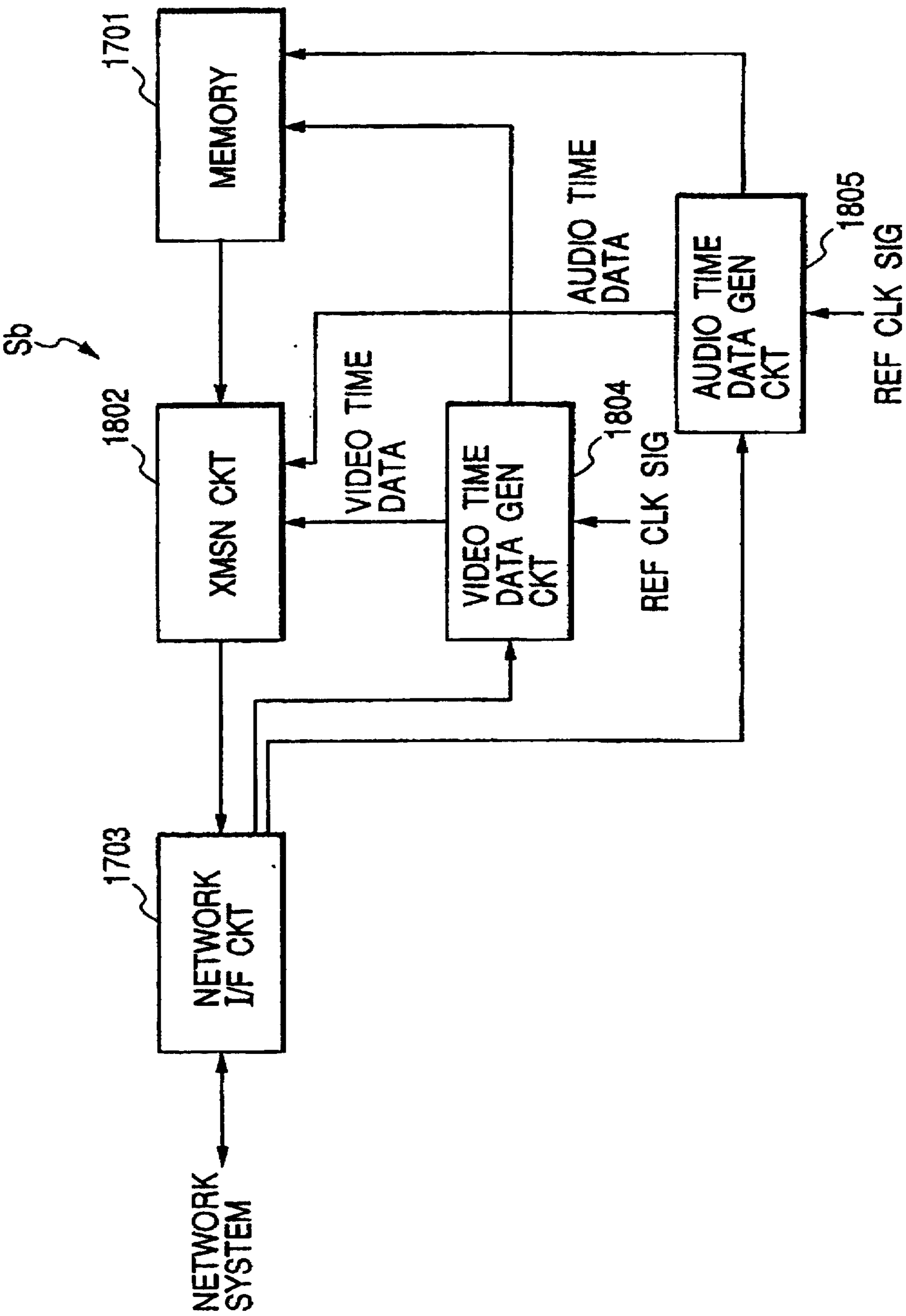

Sb
1701
MEMORY
1802
XMSN CKT
1703
NETWORK
I/F CKT
NETWORK
SYSTEM
VIDEO TIME
DATA
AUDIO TIME
DATA
1804
VIDEO TIME
DATA GEN
CKT
REF CLK SIG
1805
AUDIO TIME
DATA GEN
CKT
REF CLK SIG

FIG. 19

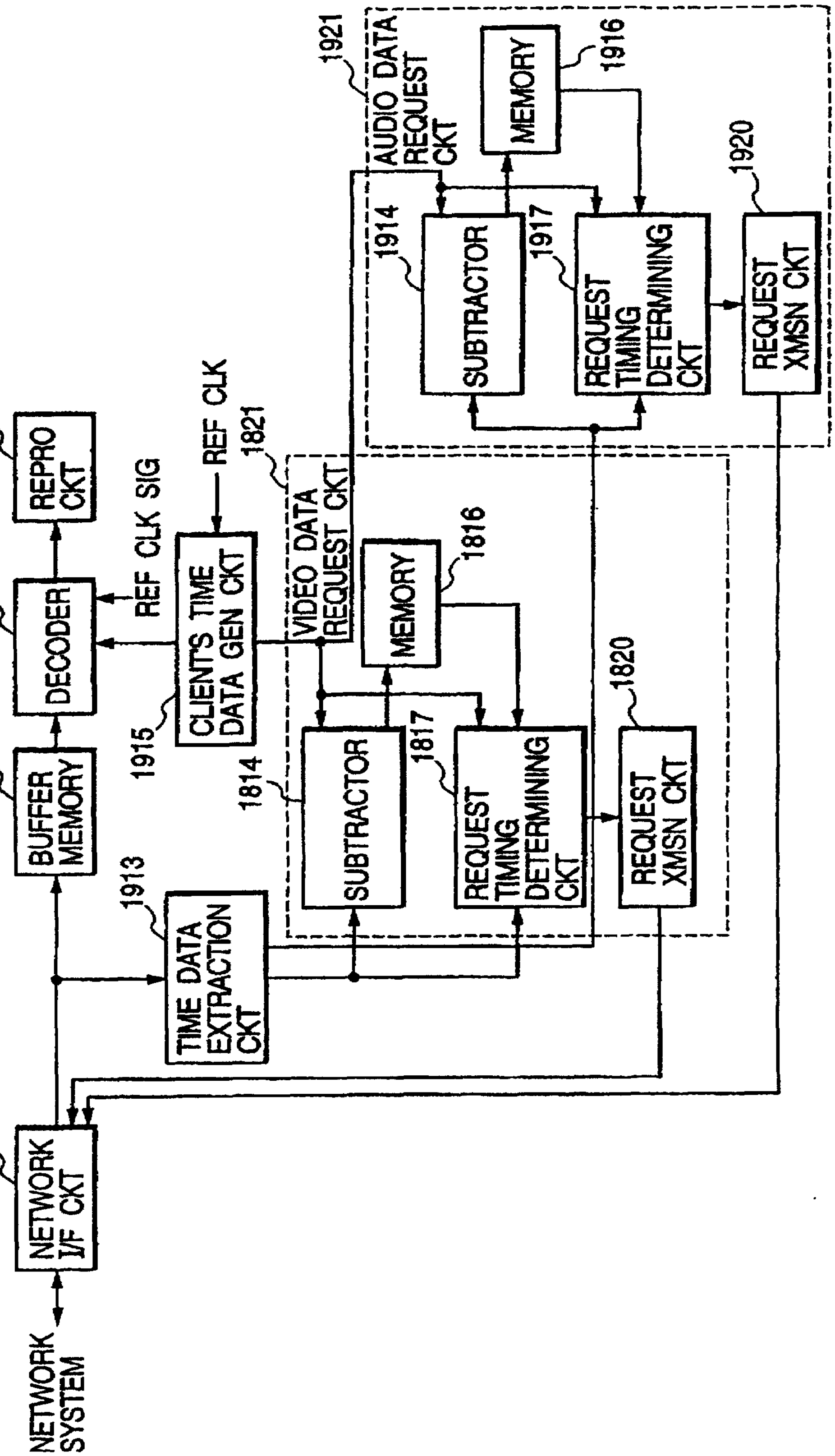

Cb
NETWORK SYSTEM
11
NETWORK I/F CKT
12
BUFFER MEMORY
1918
DECODER
19
REPRO CKT
1913
TIME DATA EXTRACTION CKT
1915
CLIENT'S TIME DATA GEN CKT
REF CLK SIG
REF CLK
1821
VIDEO DATA REQUEST CKT
1814
SUBTRACTOR
1816
MEMORY
1817
REQUEST TIMING DETERMINING CKT
1820
REQUEST XMSN CKT
1921
AUDIO DATA REQUEST CKT
1914
SUBTRACTOR
1916
MEMORY
1917
REQUEST TIMING DETERMINING CKT
1920
REQUEST XMSN CKT

FIG. 20

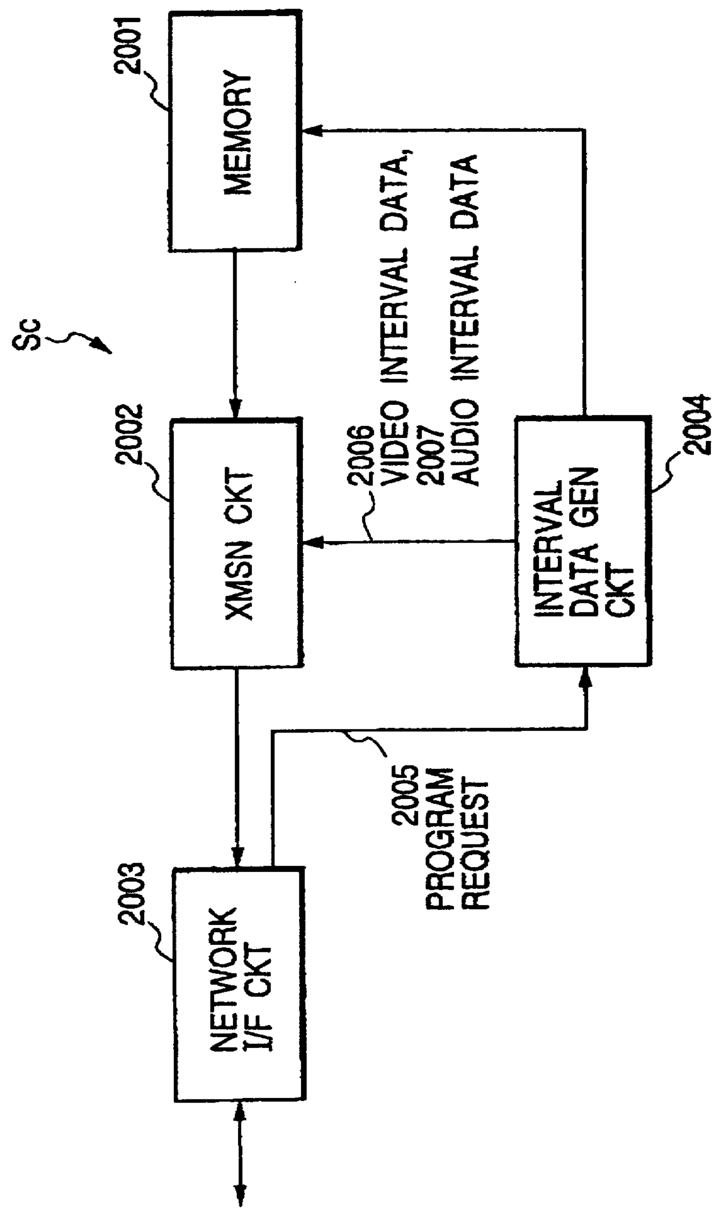

Sc
2001
MEMORY
2002
XMSN CKT
2003
NETWORK I/F CKT
2004
INTERVAL DATA GEN CKT
2005 PROGRAM REQUEST
2006 VIDEO INTERVAL DATA,
2007 AUDIO INTERVAL DATA

FIG. 21

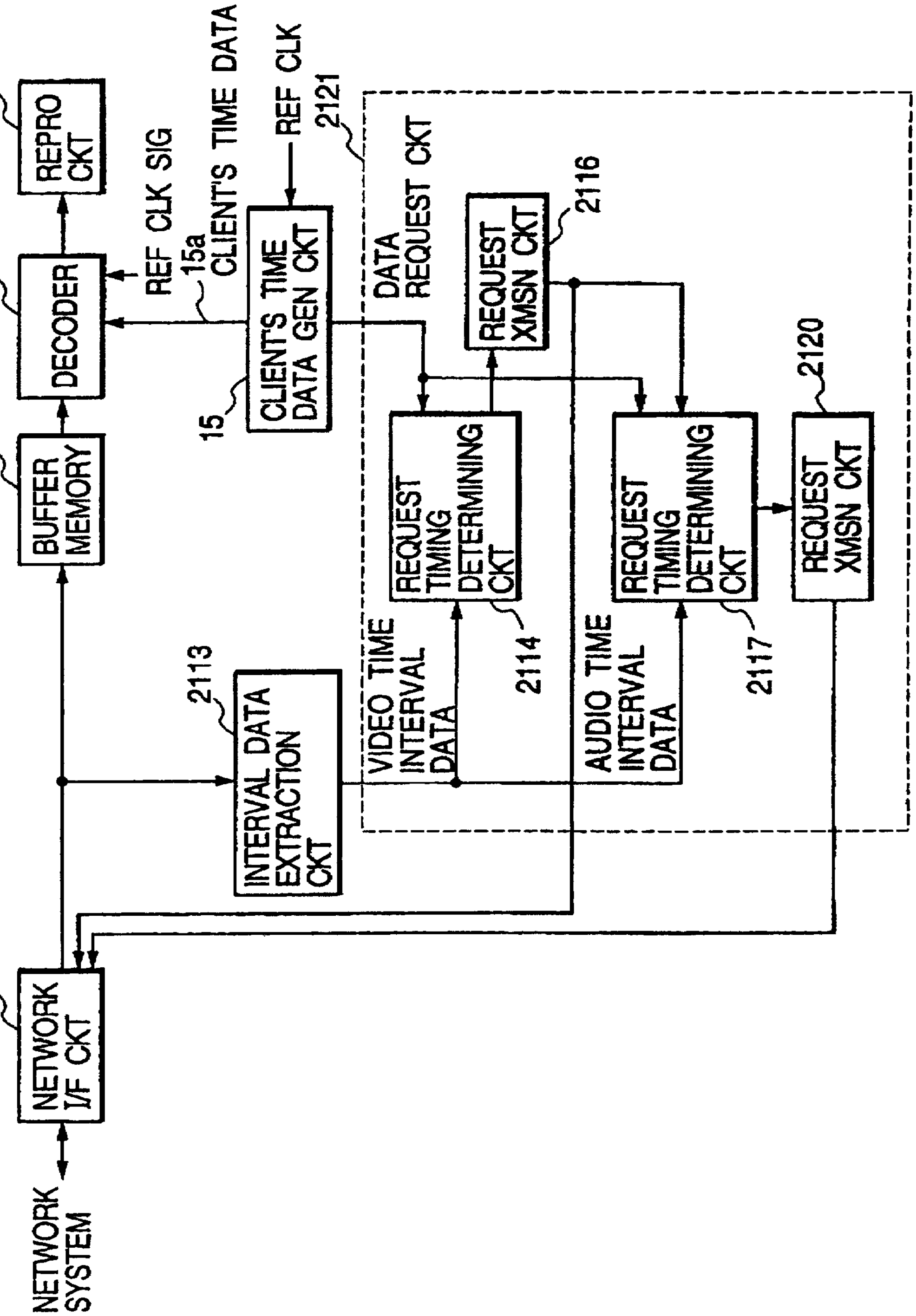

Cc
NETWORK SYSTEM
NETWORK I/F CKT
11
BUFFER MEMORY
12
DECODER
18
REPRO CKT
19
REF CLK SIG
15a
CLIENT'S TIME DATA
CLIENT'S TIME DATA GEN CKT
15
REF CLK
2121
DATA REQUEST CKT
REQUEST XMSN CKT
2116
REQUEST TIMING DETERMINING CKT
2114
REQUEST TIMING DETERMINING CKT
2117
REQUEST XMSN CKT
2120
INTERVAL DATA EXTRACTION CKT
2113
VIDEO TIME INTERVAL DATA
AUDIO TIME INTERVAL DATA

FIG. 22

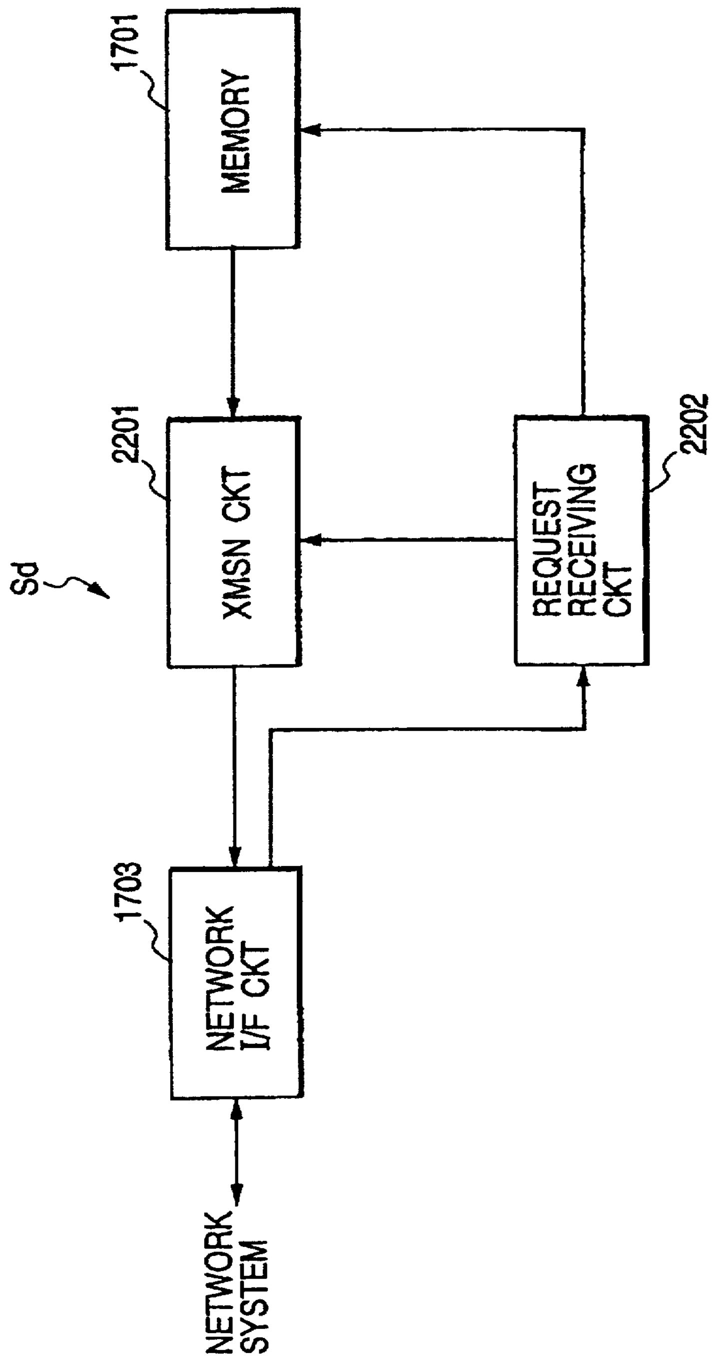

Sd
MEMORY
1701
XMSN CKT
2201
REQUEST RECEIVING CKT
2202
NETWORK I/F CKT
1703
NETWORK SYSTEM

FIG. 23

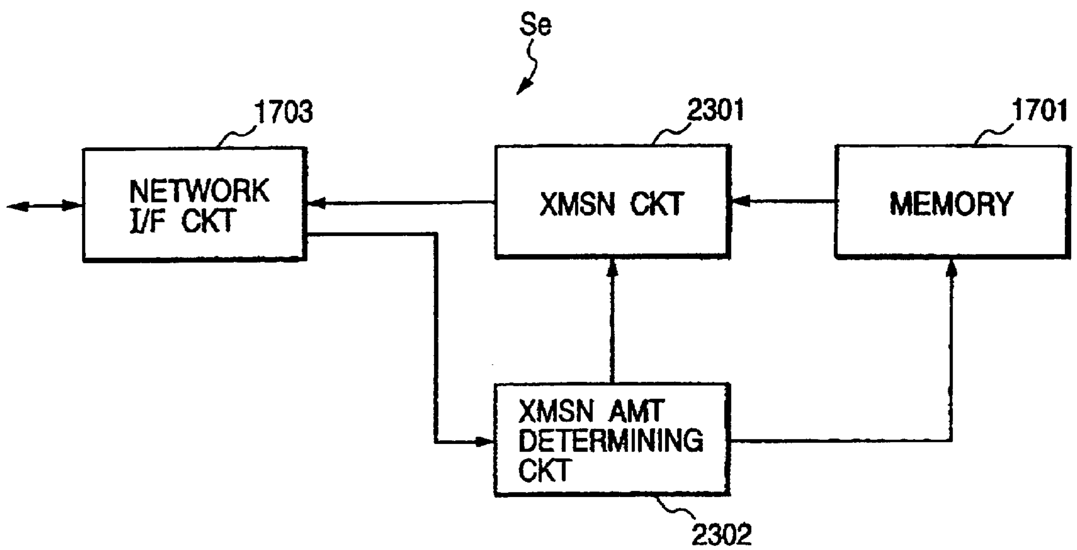

Se
1703
NETWORK I/F CKT
2301
XMSN CKT
1701
MEMORY
XMSN AMT DETERMINING CKT
2302

FIG. 24

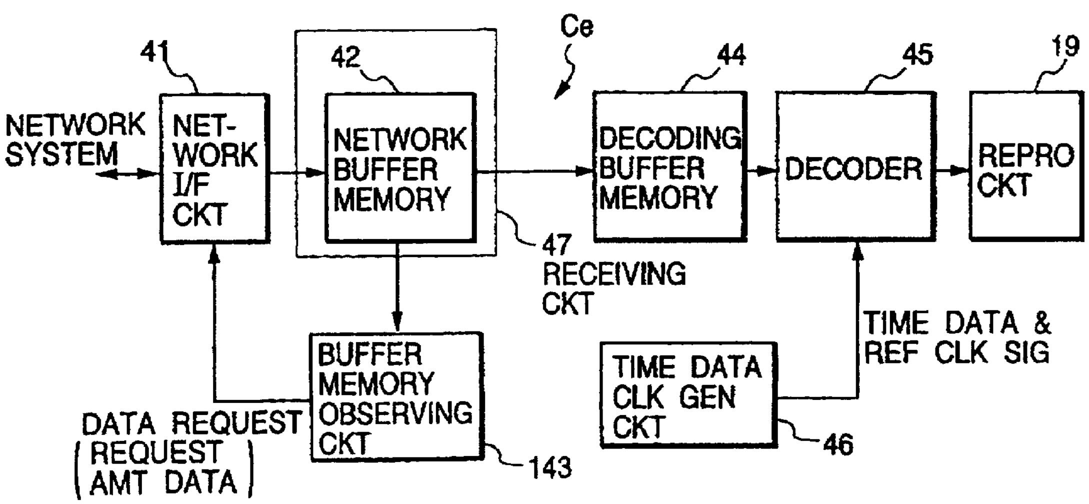

Ce
41
NETWORK SYSTEM
NET-WORK I/F CKT
42
NETWORK BUFFER MEMORY
44
DECODING BUFFER MEMORY
45
DECODER
19
REPRO CKT
47
RECEIVING CKT
BUFFER MEMORY OBSERVING CKT
143
DATA REQUEST (REQUEST AMT DATA)
TIME DATA CLK GEN CKT
46
TIME DATA & REF CLK SIG

FIG. 25

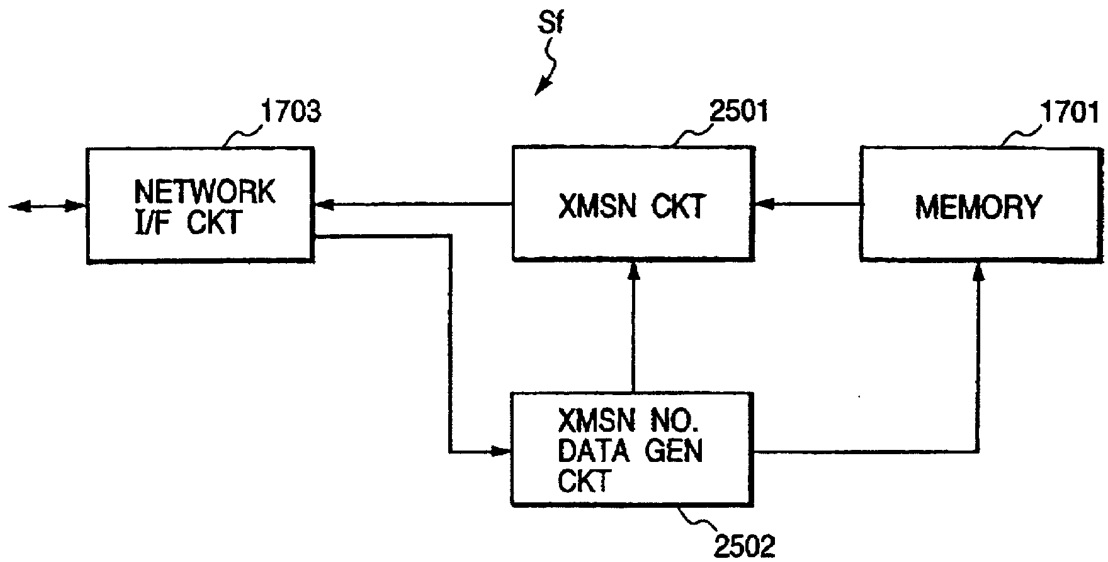

Sf
1703
2501
1701
NETWORK I/F CKT
XMSN CKT
MEMORY
XMSN NO. DATA GEN CKT
2502

FIG. 26

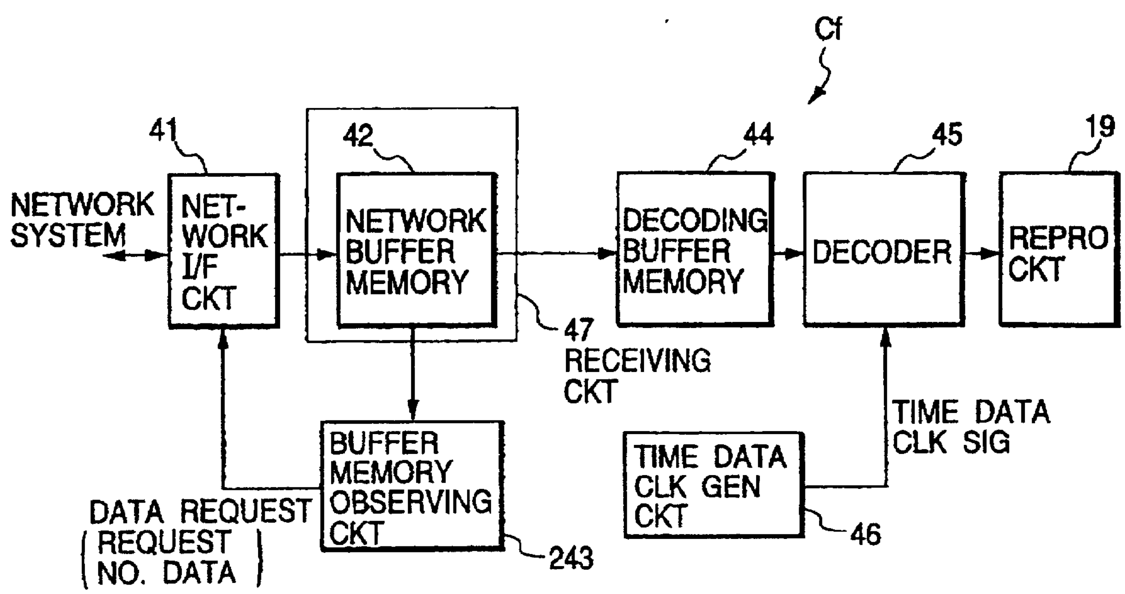

Cf
41
42
44
45
19
NETWORK SYSTEM
NET-WORK I/F CKT
NETWORK BUFFER MEMORY
DECODING BUFFER MEMORY
DECODER
REPRO CKT
47 RECEIVING CKT
BUFFER MEMORY OBSERVING CKT
243
DATA REQUEST (REQUEST NO. DATA)
TIME DATA CLK GEN CKT
46
TIME DATA CLK SIG

FIG. 27

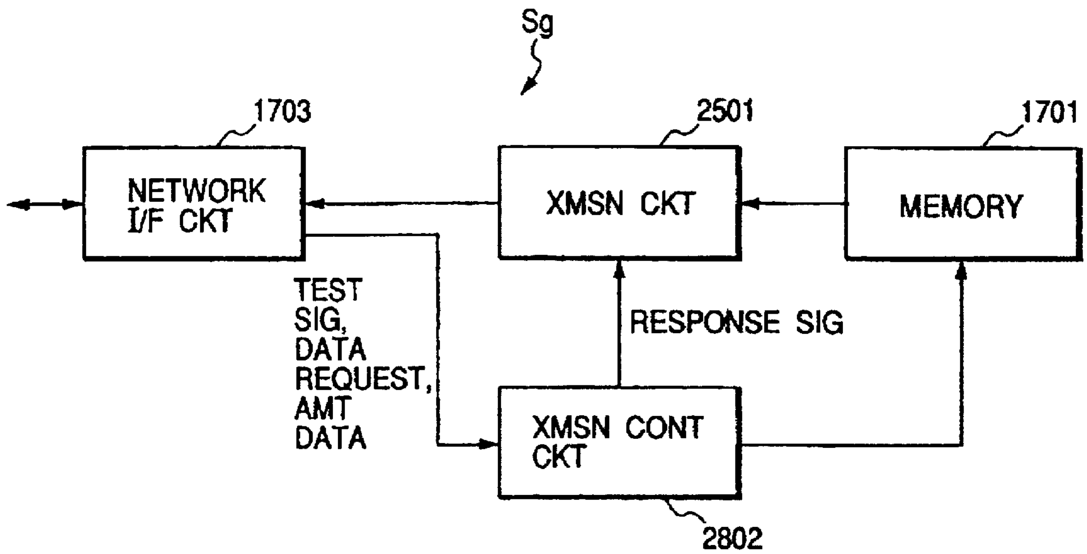

Sg
1703
2501
1701
NETWORK I/F CKT
XMSN CKT
MEMORY
TEST SIG, DATA REQUEST, AMT DATA
RESPONSE SIG
XMSN CONT CKT
2802

FIG. 28

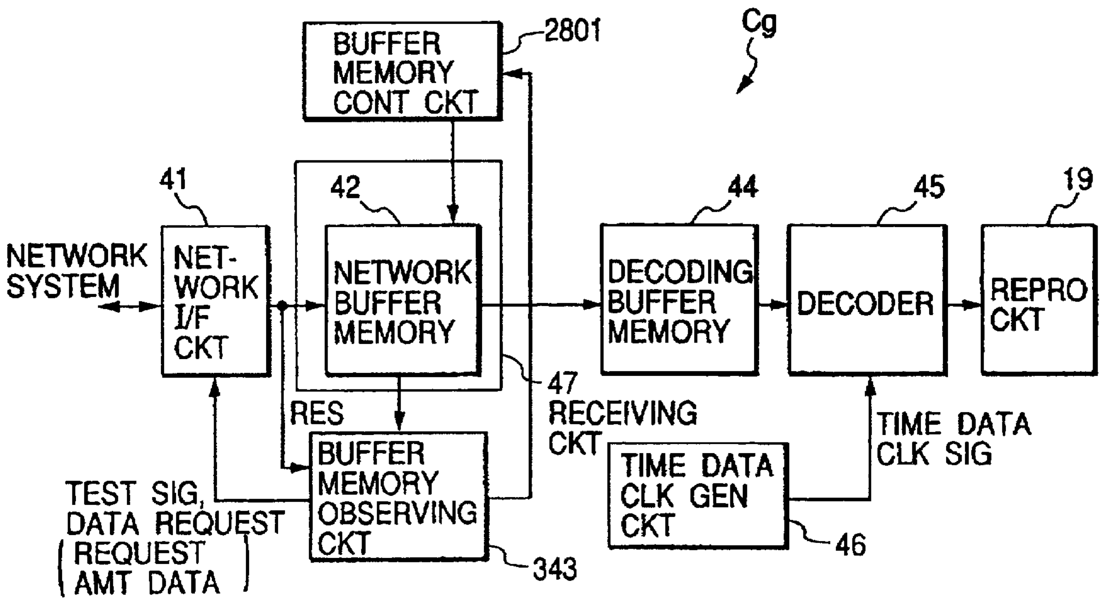

BUFFER MEMORY CONT CKT
2801
Cg
41
42
44
45
19
NETWORK SYSTEM
NET-WORK I/F CKT
NETWORK BUFFER MEMORY
DECODING BUFFER MEMORY
DECODER
REPRO CKT
47 RECEIVING CKT
RES
BUFFER MEMORY OBSERVING CKT
TIME DATA CLK SIG
TIME DATA CLK GEN CKT
TEST SIG, DATA REQUEST (REQUEST AMT DATA)
343
46

FIG. 29

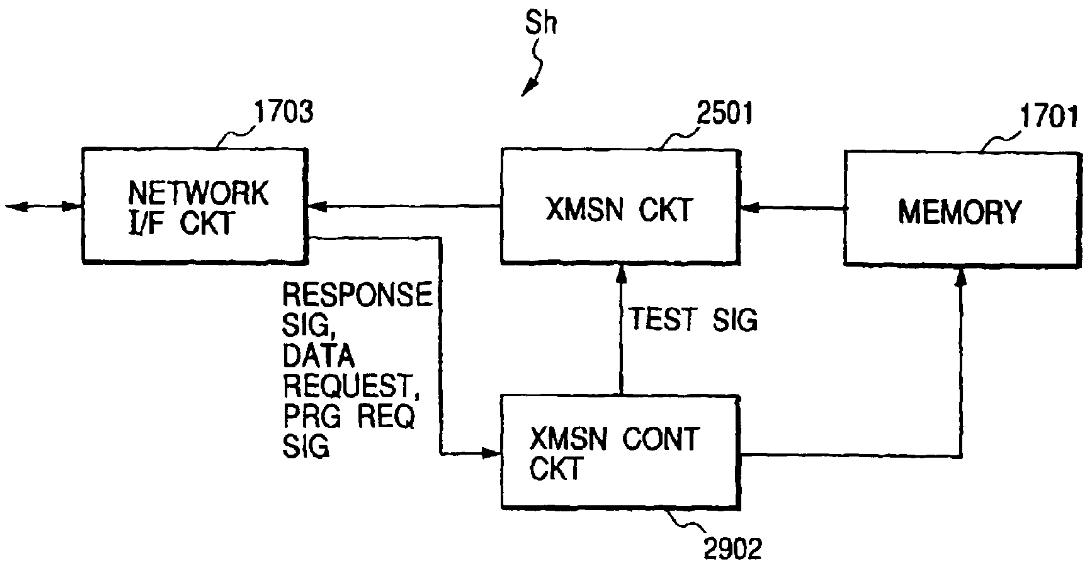

Sh
1703
NETWORK I/F CKT
2501
XMSN CKT
1701
MEMORY
RESPONSE SIG, DATA REQUEST, PRG REQ SIG
TEST SIG
XMSN CONT CKT
2902

FIG. 30

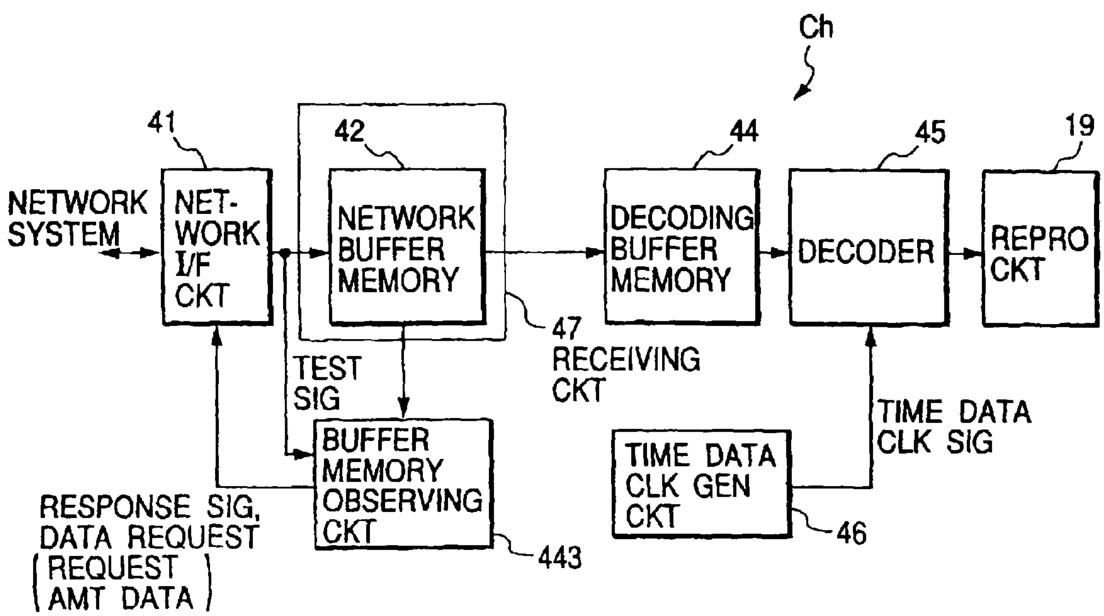
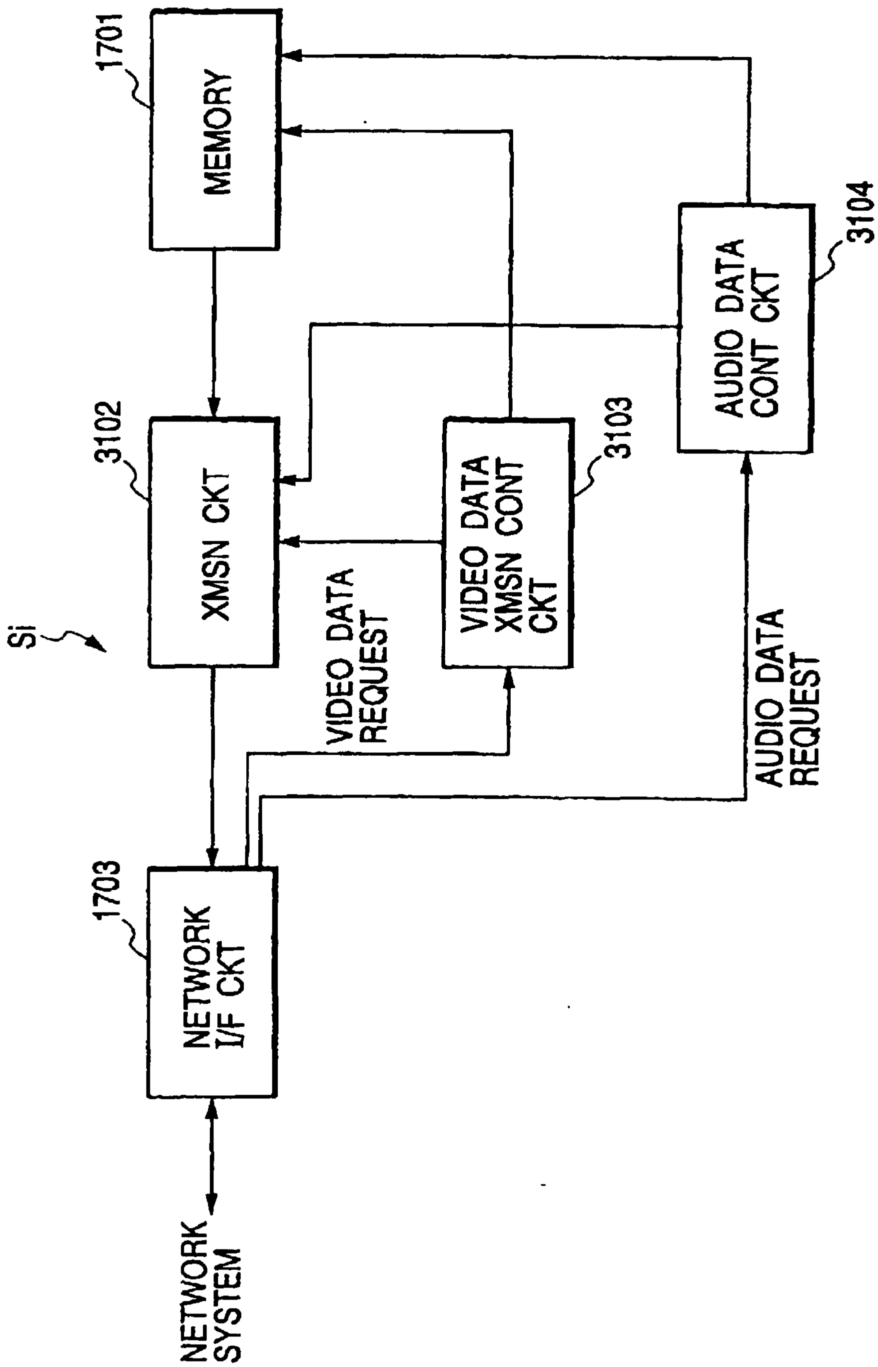

Ch
41
42
44
45
19
NETWORK SYSTEM
NET-WORK I/F CKT
NETWORK BUFFER MEMORY
DECODING BUFFER MEMORY
DECODER
REPRO CKT
47 RECEIVING CKT
TEST SIG
BUFFER MEMORY OBSERVING CKT
443
RESPONSE SIG, DATA REQUEST (REQUEST AMT DATA)
TIME DATA CLK GEN CKT
46
TIME DATA CLK SIG

FIG. 31



Si
1701
MEMORY
3102
XMSN CKT
3103
VIDEO DATA XMSN CONT CKT
3104
AUDIO DATA CONT CKT
1703
NETWORK I/F CKT
VIDEO DATA REQUEST
AUDIO DATA REQUEST
NETWORK SYSTEM

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 895 420 A3

(12) EUROPEAN PATENT APPLICATION

(88) Date of publication A3:
30.10.2002 Bulletin 2002/44

(43) Date of publication A2:
03.02.1999 Bulletin 1999/05

(21) Application number: **98306129.2**

(22) Date of filing: **31.07.1998**

(51) Int Cl.7: **H04N 7/173**, H04N 7/24

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **01.08.1997 JP 20776697**
05.09.1997 JP 24128097

(71) Applicant: **Victor Company of Japan, Ltd.**
Yokohama 221-0022 (JP)

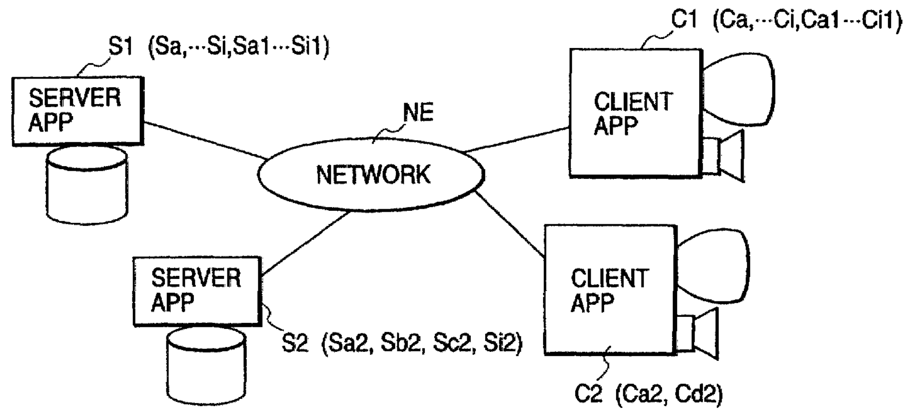
(72) Inventors:
- **Yamada, Yasuhiro**
  Singapore 128797 (JP)
- **Tsutsumi, Masahito**
  Yokosuka-shi, Kanagawa-ken (JP)

(74) Representative:
Dempster, Benjamin John Naftel et al
Withers & Rogers,
Goldings House,
2 Hays Lane
London SE1 2HW (GB)

(54) A data transmission system and reproducing apparatus

(57) A data transmission system including: at least one server apparatus for supplying video (moving picture image) data and audio data and at leas client apparatus coupled through a network receiving the video data and audio data is disclosed. The server apparatus transmits a burst including the video data and audio data. The client apparatus receives and reproduce the video and audio data and server's time data indicative of a timing of requesting the next burst. The next burst transmission request is transmitted according to the clock of the client and the server's time data. The video data and audio data may be transmitted independently. The video data and audio data may be transmitted periodically. The next burst transmitting request may be transmitted when an empty data is less than a reference. Data demanding amount data may be transmitted from the client apparatus to said server apparatus. Data transmission number data may be transmitted in the next burst transmission signal. A test signal is transmitted between the client and the server to determined the data demanding amount data and a capacity of the buffer memory and corresponding reproducing apparatus is also disclosed.

FIG. 16A



EP 0 895 420 A3

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 98 30 6129

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
A	EP 0 402 954 A (HARRIS CORP) 19 December 1990 (1990-12-19) * column 3, line 10 - line 15 * * column 14, line 13 - line 26 * ---	1-21	H04N7/173 H04N7/24
A	US 5 477 542 A (IWAMI NAOKO ET AL) 19 December 1995 (1995-12-19) section "Summary of the invention" * column 4, line 20 - column 10, line 45 * ---	1-21	
A	US 5 528 284 A (IWAMI NAOKO ET AL) 18 June 1996 (1996-06-18) * column 3, line 8 - line 23 * ---	1-21	
X	EP 0 660 613 A (SONY CORP) 28 June 1995 (1995-06-28) * column 13, line 1 - column 15, line 24; figure 17 * ---	6,13,14, 19	
A	US 5 565 924 A (REIBMAN AMY R ET AL) 15 October 1996 (1996-10-15) * column 1, line 9 - column 2, line 4 * * figure 3 * ---	6-16, 19-21	TECHNICAL FIELDS SEARCHED (Int.Cl.6) H04N
A	"METHOD TO DELIVER SCALABLE VIDEO ACROSS A DISTRIBUTED COMPUTER SYSTEM" IBM TECHNICAL DISCLOSURE BULLETIN, IBM CORP. NEW YORK, US, vol. 5, no. 37, May 1994 (1994-05), pages 251-255, XP000453152 ISSN: 0018-8689 * page 252, paragraph 5 * -----	6-16, 19-21	

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	21 August 2002	La, V

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

European Patent Office

Application Number
EP 98 30 6129

CLAIMS INCURRING FEES

The present European patent application comprised at the time of filing more than ten claims.

☐ Only part of the claims have been paid within the prescribed time limit. The present European search report has been drawn up for the first ten claims and for those claims for which claims fees have been paid, namely claim(s):

☐ No claims fees have been paid within the prescribed time limit. The present European search report has been drawn up for the first ten claims.

LACK OF UNITY OF INVENTION

The Search Division considers that the present European patent application does not comply with the requirements of unity of invention and relates to several inventions or groups of inventions, namely:

see sheet B

☒ All further search fees have been paid within the fixed time limit. The present European search report has been drawn up for all claims.

☐ As all searchable claims could be searched without effort justifying an additional fee, the Search Division did not invite payment of any additional fee.

☐ Only part of the further search fees have been paid within the fixed time limit. The present European search report has been drawn up for those parts of the European patent application which relate to the inventions in respect of which search fees have been paid, namely claims:

☐ None of the further search fees have been paid within the fixed time limit. The present European search report has been drawn up for those parts of the European patent application which relate to the invention first mentioned in the claims, namely claims:

European Patent Office

LACK OF UNITY OF INVENTION SHEET B

Application Number
EP 98 30 6129

The Search Division considers that the present European patent application does not comply with the requirements of unity of invention and relates to several inventions or groups of inventions, namely:

1. Claims: 1-5,17,18

A data transmission system comprising at least one server apparatus for incontinuously transmitting coded moving picture image data and coded audio data in every burst to a network system, and at least one client apparatus for receiving the transmitted data in said burst and for decoding the received data to be continuously reproduced in time base, wherein said server apparatus generates server's time data indicative of time of the next burst to provide continuously reproducing said moving picture image data and said audio data and attaches said server's time data to said data in said burst and said client apparatus extracts said server's time data from the received data, generates client's time data indicative of present time of said client apparatus, and transmits a request for the next burst at a request timing determined in accordance with extracted server's time data and present time indicated by said client's time data.

2. Claims: 6-16,19-21

A data transmission system comprising at least one server apparatus for intermittently transmitting a burst including coded moving picture image data and coded audio data to a network in response to data request, and at least one client apparatus for transmitting said data request, for receiving the transmitted data in said burst and for decoding the received data to be reproduced continuously in time base, wherein said client apparatus temporally stores the received data in a network buffer before decoding them, detects an empty area of said network buffer regarding the stored data, and transmits said data request to said server when said empty area is higher than a reference, and said server apparatus transmits the next burst in response to said data request.

IPR2022-01227
EXHIBIT 1003 - PAGE 00775

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 98 30 6129

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

21-08-2002

Patent document cited in search report		Publication date	Patent family member(s)			Publication date
EP 0402954	A	19-12-1990	US	5128776	A	07-07-1992
			US	5426513	A	20-06-1995
			AT	143558	T	15-10-1996
			DE	69028660	D1	31-10-1996
			DE	69028660	T2	03-04-1997
			EP	0402954	A2	19-12-1990
US 5477542	A	19-12-1995	JP	6284148	A	07-10-1994
US 5528284	A	18-06-1996	JP	6237451	A	23-08-1994
EP 0660613	A	28-06-1995	EP	0660613	A1	28-06-1995
			US	5644506	A	01-07-1997
			CN	1113405	A	13-12-1995
			WO	9502947	A1	26-01-1995
US 5565924	A	15-10-1996	US	5543853	A	06-08-1996
			CA	2167001	A1	20-07-1996
			EP	0723345	A2	24-07-1996
			JP	3168243	B2	21-05-2001
			JP	8251580	A	27-09-1996

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 984 584 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
08.03.2000 Bulletin 2000/10

(51) Int Cl.[7]: **H04L 12/18**, H04L 29/06

(21) Application number: 99306950.9

(22) Date of filing: 01.09.1999

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: 04.09.1998 US 148244
18.03.1999 US 272673

(71) Applicant: America Online, Inc.
Dulles, VA 20166 (US)

(72) Inventors:
- Lippert, Patrick
Mission Viejo, California 92691 (US)
- Enete, Noel
Seal Beach, California 90740 (US)
- Watson, Richard
Austin, Texas 78751 (US)
- Pai, Ashok
Irvine, California 92612 (US)
- Lippke, David
Herndon, Virginia 20170 (US)

(74) Representative: Charig, Raymond Julian
Eric Potter Clarkson,
Park View House,
58 The Ropewalk
Nottingham NG1 5DD (GB)

(54) Internet multimedia broadcast system

(57) A method and system for playback of live and pre-recorded multimedia data in real-time over a large scale communication network, such as the world-wide web. The invention allows a user connected to a specialized network to play back a multimedia broadcast. The invention includes a process that allows multimedia content to be created and scheduled as a playlist. The playlist data is compressed and transmitted to a host system as part of a capture protocol. The captured playlist data may then be "broadcast" to users. A user may choose a channel selection from a playlist, and cause the selected item to be downloaded and played back by means of a playback tool. The host system includes a highly efficient distribution system that allows a large number of users accessing the host through Terminal Information Handlers to rapidly access one or more channels of multimedia data. The system architecture provides a "fan out" mechanism that includes a master broadcast process that accepts a multi-stream data flow and then distributes the multi-stream data flow to essentially every Terminal Information Handler accessible to the host. The load of providing data streams is thus spread among a large number of Terminal Information Handlers, reducing access latency and providing support for hundreds of thousands of users over a large scale communication network.

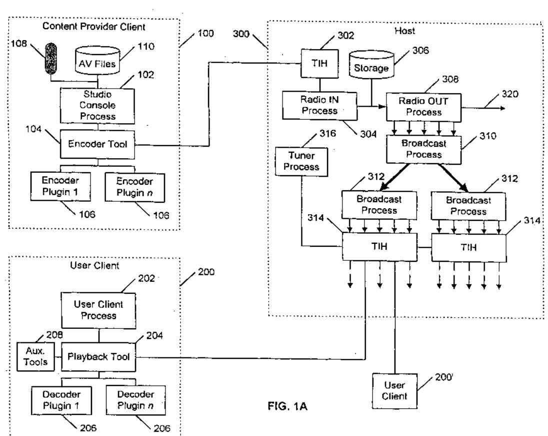


FIG. 1A

EP 0 984 584 A1

Description

TECHNICAL FIELD

[0001] This invention relates to a method and system for playback of live and pre-recorded multimedia data in real-time over a large scale communication network, such as the world-wide web.

BACKGROUND

[0002] The world wide web, or Internet, is a large scale communication network that has become increasingly popular with users for accessing data from a huge variety of sources. Currently, a large percentage of users access the Internet through modems, typically having a data rate between about 33.6 Kbps and 14.4 Kbps. For data such as text or simple graphics, such speeds are adequate to enable a "web browsing" experience that most users find enjoyable. These data rates are also acceptable for transmittal and playback of multimedia data *(e.g.,* audio, video, still images, text, hyperlinks, *etc.)* in real-time if the data is highly compressed. Accordingly, a number of schemes have been implemented for allowing users to access and download compressed multimedia data streams for local decompression and playback.

[0003] One such approach for real-time multimedia data playback utilizes the Internet multicast protocol. Multicasting includes transmitting a communication from one site to a group of selected receivers (multicasting differs from broadcasting in that a broadcast is sent to everyone who has the equipment or connection to receive it). Under this scheme, a user initiates an action *(e.g.,* selection of a uniform resource locator - URL - address) that requests that a multimedia data stream be transmitted from a server to the user's client system, for decompression and playback by a software process executing on the client. The request causes the router to which the user is coupled to access special multicast Internet addresses. The requested data is then supplied to the user (or "subscriber") by means of a multicasting backbone (MBone), which is a network of special Intemet sites that supports Internet Protocol multicasting for a limited number of users. MBone provides a faster technology than the Internet for transmitting real-time audio and video programs.

[0004] A disadvantage of Internet multicasting is that a multicast event (*e.g.*, a live radio show) can only effectively support several hundreds to a few thousands of users with uninterrupted data streams. Due to bandwidth constraints and the lack of guaranteed quality of service for the Internet, adding more users may cause objectionable pauses in content (*e.g.*, pauses in multimedia playback). Thus, for some events, Internet multicasting cannot meet user demand.

[0005] Accordingly, there is a need for an architecture that can provide for playback of live and pre-recorded multimedia data in real-time over a large scale communication network. such as the world-wide web, for a large number of users, typically in the hundreds of thousands of users. The present invention provides a method and system for such an architecture.

SUMMARY

[0006] The preferred embodiment of the invention allows a user connected to a specialized network having a "Multimedia Broadcast" feature to play back a multimedia broadcast. One aspect of the invention includes a Studio Console Process executing on a content provider client system that allows multimedia content to be created and scheduled as a "playlist". The data comprising a playlist may be pre-recorded or live. The playlist data is compressed under the control of the Studio Console Process and transmitted to a host system as part of a "capture" protocol controlled by a "radio IN" process. The captured playlist data may then be "broadcast" to users who "tune in" to a "live multimedia show", or stored for later broadcast. (In this context, "broadcast" simply means transmittal of a data stream to a user who selects content to be played back on the user's client system).

[0007] A user may choose a "channel" selection from a playlist, and cause the selected item to be downloaded and played back by means of a "playback tool". The data stream corresponding to the selected item optionally may include embedded non-audio content. such as hypertext links to other forms, URLs to web pages, static images, video images, *etc*.

[0008] The host system includes a highly efficient distribution system that allows a large number of users accessing the host through Terminal Information Handlers to rapidly access one or more channels of audio data augmented by non-audio multimedia data. In particular, the inventive system architecture provides a "fan out" mechanism that includes a master broadcast process that accepts a multi-stream data flow from a "radio OUT" process and then distributes the multi-stream data flow to essentially every Terminal Information Handler accessible to the host. The load of providing data streams is thus spread among a large number of Terminal Information Handlers, reducing access latency to typically less than about 3 seconds, and providing support for hundreds of thousands of users over a large scale communication network such as the world-wide web.

[0009] In particular, in one aspect the invention includes a system for playback of live and pre-recorded multimedia data in real-time over a large scale communication network, including a computer system having a plurality of terminal information handlers for managing general information flow to and from a plurality of users; an output process for assembling multiple multimedia data streams for distribution; at least one broadcast process, in communication with the output process, for distributing the assembled multiple multimedia data streams to each of the terminal information handlers; and a selector process, in communication with the terminal information handlers, for receiving a channel request from a user through an terminal information handler associated with the user, mapping the channel request to a corresponding one of the multiple multimedia data streams, and enabling transmission of the corresponding one multimedia data stream to the user through the associated terminal information handler. In another aspect, the invention includes a corresponding method.

[0010] The details of one or more embodiments of the invention are set forth in the accompanying drawings and the description below. Other features, objects, and advantages of the invention will be apparent from the description and drawings, and from the claims.

DESCRIPTION OF DRAWINGS

[0011] FIG. 1A is block diagram of the preferred embodiment of the invention.

[0012] FIG. 1B is block diagram of a portion of an alternative embodiment of the invention.

[0013] FIG. 2 is a "screen shot" of one embodiment of a graphical user interface (GUI) for defining a data stream within a studio console process in accordance with the invention.

[0014] FIG. 3 is a "screen shot" of one embodiment of a GUI for defining a playlist schedule within a studio console process in accordance with the invention.

[0015] FIG. 4 is a "screen shot" of one embodiment of a GUI for displaying playback information and allowing selection of a playback channel within a user client process in accordance with the invention.

[0016] FIG. 5A is a block diagram showing the client-host architecture for a capture session.

[0017] FIG. 5B is a data flow diagram showing the data flow for establishing a capture session.

[0018] FIG. 5C is a data flow diagram showing the data flow for recording during a capture session.

[0019] FIG. 5D is a data flow diagram showing the data flow for ending a capture session.

[0020] FIG. 6A is a data flow diagram showing the data flow architecture for playback of a multimedia data stream.

[0021] FIG. 6B is a data flow diagram showing the data flow for starting a playback session.

[0022] FIG. 6C is a data flow diagram showing the data flow for switching a playback channel.

[0023] FIG. 6D is a data flow diagram showing the data flow for ending a playback session.

[0024] Like reference numbers and designations in the various drawings indicate like elements.

DETAILED DESCRIPTION

Overview

[0025] The preferred embodiment of the invention allows a user connected to a specialized network having a "Radio" feature *(e.g.,* the America Online network) to play back an audio broadcast that may be augmented with non-audio multimedia data, such as video and still images, URLs, text fields, *etc.* One aspect of the invention includes a Studio Console Process executing on a content provider client system that allows such content to be created and scheduled.

[0026] For example, the Studio Console Process enables a content provider to schedule a week's worth of prerecorded content that changes daily. The content provider uses a form in the Studio Console Process to assemble a "playlist" for each day. If the content provider then needs to conduct a live event on the Radio system, the content provider uses the Studio Console Process to override any existing playlist. When the live event is completed, the playlist resumes as originally scheduled.

[0027] The data comprising a playlist may be pre-recorded or live. The playlist data is compressed under the control of the Studio Console Process and transmitted to a host system as part of a "capture" process. The captured playlist data may then be "broadcast" to users who "tune in" to a "live multimedia show", or stored for later broadcast. (In this context, "broadcast" simply means transmittal of a data stream to a user who selects content to be played back on the user's client system).

[0028] A user may initiate a form that displays available playlists for one or more "channels" of a host system, choose a selection from a playlist, and cause the selected item to be downloaded and played back by means of a "playback tool". The data stream corresponding to the selected item optionally may include embedded non-audio content, such as hypertext links to other forms, URLs to web pages, static images, or video images.

[0029] The host system includes a highly efficient distribution system that allows a large number of users to rapidly access one or more channels of audio data augmented by non-audio data. The latency of access - the time between a request for playback of a particular channel and commencement of playback on a user client - is typically less than about 3 seconds, compared to tens of seconds for many prior art systems.

System Architecture - Content Provider Client

[0030] FIG. 1A is block diagram of the preferred embodiment of the invention. Content to be broadcast is typically developed by a content provider utilizing a content provider client 100. A Studio Console Process 102 executes on the content provider client 100. The Studio Console Process 102 allows a content provider to define ("author") a play list data stream to be uploaded to a host 300 during a "capture" session for "broadcast" to users selecting the channel assigned to the playlist. (Details of a capture session are described below.) The playlist data stream includes audio content and may include images or other data which may be inserted into the data stream as the data is being compressed. In the preferred embodiment, the Studio Console Process 102 communicates with a separate encoder tool 104 that responds to and carries out the directions of the content provider input by means of the Studio Console Process 102. In particular, the encoder tool 104 selects one of a plurality of encoder "plugins" 106 to perform necessary data compression of the multimedia content of the data stream. The encoder tool 104 also packages the compressed multimedia content into packets for transmission to the host 300. In the preferred embodiment, audio data is given priority over non-audio data to reduce the likelihood of any objectionable interruption in audio playback. In general, other types of data may be appended to audio packets as long as the total transmission packet (*e.g.*, an Internet protocol packet) does not exceed a target bit rate *(e.g.,* 10 Kbps).

[0031] Audio content may come from "live" sources to the content provider client 100, such as an input to a computer sound card from a microphone 108 or "line-in" source (*e.g.,* an analog or digital audio playback device such as an audio tape or CD player), or a direct input to the computer from a direct digital connection *(e.g.,* CDROM player). Alternatively, audio content may come from pre-recorded ("canned") sound files, such as WAV and ART files, located on storage media 110 accessible to the content provider client 100.

[0032] Similarly, images (still or video) may come from "live" sources to the content provider client 100, such as video cameras. Still images can be from standard file formats, including ART, BMP, GIF or JPG (JPEG) formats. In the preferred embodiment, all such formats are re-compressed into a single format (ART) for the broadcast data stream.

[0033] One aspect of the invention is that a content provider may change an encoder plugin 106 "on-the-fly" to accommodate different content inputs as such inputs change. For example, an encoder plugin 106 particularly well-suited for speech may be used for a time (for example, while a conductor is explaining a music piece about to be played), and then an encoder plugin 106 particularly well-suited for music may be switched in when the source material changes (for example, when the music piece is being played).

[0034] FIG. 2 is a "screen shot" of one embodiment of a graphical user interface (GUI) for defining a data stream within the Studio Console Process 102. The preferred content definition GUI for the Studio Console Process 102 includes controls that allow the content provider to do at least the following in "authoring" a playlist entry:

- Select an audio source 150 ("live" or "canned") for a data stream.
- Define an audio target 152 for the data stream (*e.g.*, a file name for storage on the content provider client 100 or on the host 300 for later broadcast, or a channel name for a live broadcast).
- Select an encoder 154 for compressing the audio source data stream. The list of available encoders allows selection of varying degrees of compression, depending on the desired final sound quality and the nature of the audio source material (*e.g.*, speech generally can be compressed more than music). Examples of such encoders include those available from Voxware, Inc. of Princeton, NJ. For example, the Voxware VR12 Speech Codec requires a bit rate of only 1.5 Kbps for speech, while a music codec may require as much as 10 Kbps.
- Define a "stream name" 156, which is a name for the data stream that is delivered to a user's playback tool whenever playback is started for a particular virtual channel.
- Define a "stream description" 158 that provides a detailed description of the data stream that can be delivered to a user's playback tool on demand. For example, the description may contain rich-text - HTML formatted text that may contain font and color information tags.
- Define "caption text" 160, which is simple text that can appear within the data stream. In the preferred embodiment, when a playback tool receives a complete caption string. the caption string is forwarded to an auxiliary tool for display in a text field on a form, if the field exists. The caption text may be used for closed captioning or for other "headline" updates, and is preferably in "rich text format" (RTF). The caption text may also included hyperlinks.
- Select image data 162 that can be delivered appended to and interleaved with the audio data. The image data is inserted in the data stream wherever available space is found after the audio packets (a target bit rate for the data stream is set at authoring time) to expedite transfer to the client. When data is received at the playback tool, it is forwarded to an auxiliary tool for rendering. The original image data may be in any of a variety of formats, such as the well-known ART, BMP, GIF or JPG (JPEG) image formats, but is preferably transformed into a single multimedia ART format for broadcast. A number of images may be linked to form a "slide show". To save space in the data stream itself, a global ID reference to an image or video data may be delivered instead of an image itself. The global ID is passed by the user's playback tool to an aux-

iliary tool for rendering. This approach is useful if the referenced data resides on the client's system *(i.e.,* the data was downloaded previously). Video data can also be delivered appended to and interleaved with the audio data. As with image data, video data is inserted in the data stream wherever available space is found after the audio packets. However, a significant amount of bandwidth is required for video data, even for highly compressed video images, and thus should be limited for low-bandwidth users. When video data is received at the playback tool, it is forwarded to an auxiliary tool for rendering.

- Select or define URLs 164 that can be delivered appended to and interleaved with the audio data. When URLs are received at the playback tool, they forwarded to an auxiliary tool which attaches them to a button or image on a form (if one exists) displayed to the user. When the button or image is selected by the user, the attached URL is activated in conventional fashion.

[0035] The Studio Console Process 102 also may be used to schedule playlist content on a channel for an extended period of time. FIG. 3 is a "screen shot" of one embodiment of a GUI for defining a playlist schedule within the Studio Console Process 102. The preferred playlist definition GUI for the Studio Console Process 102 includes controls that allow the content provider to do at least the following in scheduling a playlist:

- Define a schedule 170 of start and stop times for each of a plurality of audio content files.
- Select a channel 172 on which multimedia content will be made available to users.
- Define a release date 174 for the playlist.
- Check the overall start and stop times 176 and file characteristics 178 for the playlist content.

System Architecture - User Client

[0036] FIG. 1A also shows a user client 200. In the preferred embodiment, a User Client Process 202 communicates through a playback tool 204 across a network *(e.g.,* the Internet) with the host 300 to access playlists and select a channel for playback. The User Client Process 202 responds to and carries out the directions of a user by means of the playback tool 204. In particular, the playback tool 204 selects one of a plurality of decoder plugins 206 to perform necessary data decompression of the multimedia content of a received data stream. The playback tool 204 also parses multimedia packets received from the host 300. In particular, the playback tool 204 processes all audio data internally, but forwards all non-audio data to one or more auxiliary tools 208, each of which can manage and render such data.

[0037] In the preferred embodiment, while a data stream may contain all types of data, each particular calling form displayed by the User Client Process 202 notifies the auxiliary tools 208 of the data that pertain to the form. That way, different forms can have different presentations for the same data stream. For example, one small form may only have start and stop buttons for the audio portion of the stream with no image box. The auxiliary tools 208 would not try to render any accompanying image data. Another, larger form may have an image box that is updated by an auxiliary tool 208 whenever a new image appears in the data stream.

[0038] In the preferred embodiment. the user client process 202 provides the ability for a user to play back a particular playlist selection until a different channel is selected or the playback session is ended. Multimedia content may be delivered in a one-time transmission or may be continuously broadcast in a looping manner. In either event, the user is able to join an active broadcast at any point during its transmission.

[0039] FIG. 4 is a "screen shot" of one embodiment of a GUI for displaying playback information and allowing selection of a playback channel within the User Client Process 202. (Details of a playback session are described below.) The preferred playback GUI includes controls that allow the user to do at least the following:

- Display a playlist 400 of stream names for selectable data streams.
- Display a stream description 404 for a selected stream name.
- Display accompanying caption text 406.
- Display accompanying graphics 408.
- Display active link buttons 410 for accompanying URLs.
- Include other desired or convenient controls, such a stop button 412. For example, if the user presses the stop button 412, the playback tool 204 should blank out the stream name, stream description, link button, and caption text. The playback tool 204 could also display a default name, such as "no station selected". The playback tool 204 may also cause a blank image sent to the appropriate auxiliary tool 208 if the user presses the stop button 412 while an image is being rendered. A standard graphic can be displayed instead.

System Architecture - Host Broadcast System Architecture

[0040] Referring to FIG. 1A, a playlist of multimedia content generated by a content provider client 100 is transferred in conventional fashion over a network *(e.g.,* the Internet) to a host 300. The host 300 generally includes one or more "front-end" Terminal Information Handlers (TIH) 302 that manage general information flow to and from the host 300 for multiple users. For example, on the America Online network, each TIH 302 can manage about 63 concurrent users. A radio IN process 304 executing on the host 300 receives data from a content provider client 100 that is transmitted by means of "ri" (radio input) messages. The radio IN process 304 generally stores multimedia content as files in a storage device 306 that is accessible to the host 300 for later broadcast or for archiving of the multimedia content. However. the radio IN process 304 may also be used to directly transfer the received multimedia content to a radio OUT process 308 for broadcast to users requesting playback.

[0041] In accordance with a primary aspect of the invention, the radio OUT process 308 provides multiple data streams of multimedia content for access by users. In particular. the radio OUT process 308 may access multiple data streams from one or more storage devices 306, or accept "live" feeds of multimedia content from the radio IN process 304 *(e.g.,* a live interview or live reporting from the scene of a news event). Thus, the radio OUT process 308 gathers and assembles the data packets representing such multimedia content data streams into a "broader" multi-stream data flow.

[0042] In order to provide a highly efficient distribution system that allows a large number of users to rapidly access one or more channels of multimedia data, the inventive system architecture provides a "fan out" mechanism that includes a master broadcast process 312. The broadcast process 312 accepts the multi-stream data flow from the radio OUT process 310 and then distributes the multi-stream data flow (shown as thick arrows in FIG. 1A) along with instances of itself 312 to essentially every terminal information handler 314 accessible to the host 300. (Although only two instances of the broadcast process 312 and two TIHs 314 are shown, any number may be selected). Thus, the load of providing data streams is spread among a large number of Terminal Information Handlers 314.

[0043] For example, in the America Online network, one instance of each broadcast process would be distributed within an internal network to a "pod", each comprising a large number of individual servers. Coupled to each pod in a ring are multiple TIHs 314. Executing on each pod is an instance of the broadcast process 312. which circulates the multi-stream data flow among all TIHs within the pod. In this manner, any one channel of multimedia data stream within the multi-stream data flow is available for transmittal to a user requesting playback with very little delay. As noted above, in one embodiment of the invention, the access latency is typically less than about 3 seconds.

[0044] FIG. 1B is block diagram of a portion of an alternative embodiment of the invention. In this embodiment, the Terminal Information Handlers are configured hierarchically. A multi-stream data flow from a broadcast process 350 is transmitted to a top-level terminal front end processor (TFEP) 352 of conventional design, which controls multiple TIHs 354. The TFEP 352 then re-transmits the multi-stream data flow to dependent TIHs 354, reducing data traffic within the overall system compared to the ring architecture referenced above. Multiple user clients 356 may then access selected channels of the multi-stream data flow through the TIHs 354.

[0045] Once the multi-stream data flow content is available in the TIHs 314, any channel of the multi-stream data flow is available for transmittal to a user requesting playback. To allow a user to select a particular channel, a tuner process 318 executing within the host 300 is coupled to each TIH 314. The tuner process 318 intercepts channel requests from users and commands the TIH 314 with which the user is in communication to deliver a particular multimedia data stream to that user from the multi-stream data flow. Thereafter, packets from that data stream are transmitted to the requesting user (see below for further discussion of playback sessions).

[0046] An additional function of the tuner process 318 is that it can "map" a channel name to a channel number. For internal efficiencies, each channel of the multi-stream data flow is given a channel number. However, it may be desirable to give one or more channel names to each channel number. Thus, channel "1" may be assigned the channel name of "The AOL Radio Hour" on one playback list form displayable to users for a particular time slot, but be assigned the channel name of "The Motley Fool" on the same or another playback list form for a different time slot. Accordingly, in the preferred embodiment, the tuner process 318 maintains a map of channel names to channel numbers. The tuner process 318 then maps an incoming channel name from a user's channel request to the corresponding channel number. That channel number is then used to select the corresponding data stream on the TIH 314 to transmit to the user.

[0047] The inventive architecture may also be used in conjunction with conventional Internet multicast systems by providing a connection 320 from the radio OUT process 308 to a multicast server system, in known fashion. This capability may be useful when a channel is expected to have a relatively small audience which would not tax the "fan out" characteristics of the multicast system. However. for large "fan out" needs, the inventive architecture provides the advantages noted above.

Capture Session

[0048] FIG. 5A is a block diagram showing the client-host architecture for a capture session. A content provider is provided with the Studio Console Process 102, encoder tool 104, and encoder plugins 106 described above, which may be considered to be a single "capture tool" 502 executing on a content provider client 100. The capture tool 502 communicates with the host 300 over a network through a Send function 504, which takes care of the details of data transmission in known fashion. The host 300 communicates with the capture tool 502 through a Terminal Information Handler (TIH) 302, as described above. The capture process requires only one host process, the radio IN process 304, whose function is to receive data from a content provider that is transmitted by means of "ri" (radio input) messages. The radio IN process 304 stores multimedia content as files in a storage device 306 that are accessible to the host 300 for later broadcast. There are three phases of a capture session: Starting the capture session, Recording, and Ending the capture session.

[0049] For a live input *(e.g.,* a microphone capture), initiating a capture session will start recording, compressing the multimedia data using the selected encoder plugin 106, packetizing the compressed multimedia data as a data stream, and uploading the data stream to the host 300. Since the data is compressed and transmitted while the "live" data (*e.g.*, voice from a microphone) is being recorded, the author has no ability to edit and review the content. This option will most likely be used for unrehearsed content. For pre-recorded input (*e.g.*, a WAV file), initiating a capture session will start compressing the multimedia data using the selected encoder plugin 106, packetizing the compressed multimedia data as a data stream, and uploading the data stream to the host 300. Because the input data is file based, the author has the ability to edit and review content off-line before submitting it to the host 300.

[0050] FIG. 5B is a data flow diagram showing the data flow for establishing a capture session (the representation of the Send function 504 and TIH 302 from FIG. 5 have been omitted for clarity). To establish a capture session, the user invokes a "connect" function 506 in a form 508 of the capture tool 502, which uploads a CLIENT_CONNECT message 510 to the radio IN process 304 within the host 300. The radio IN process 304 returns the same message 512 to acknowledge that a connection has been established.

[0051] FIG. 5C is a data flow diagram showing the data flow for recording during a capture session. After the host 300 has been informed that the content provider client 100 will be sending data, the capture tool 502 initiates transfer of multimedia content with an "rc" (radio control) START_STREAM message 514. The radio IN process 304 returns the same message 516 to acknowledge the start of data transmission and opens 515 a file to store the data stream. The initialization message is then followed by "ri" (radio input) DATA messages 518-l to 518-*n* from the capture tool 502. The DATA messages contain the audio data and any supplemental data appended to the audio data. At the end of the data stream, the capture tool 502 sends an "rc" (radio control) END_STREAM message 520. The radio IN process 304 returns the same message 522 to acknowledge the end of data transmission and closes 523 the data file used to store the data stream.

[0052] FIG. 5D is a data flow diagram showing the data flow for ending a capture session. The capture session is stopped by closing the open form 508 of the capture tool 502 or by pressing the appropriate cancel control. When this happens, the capture tool 502 sends an "rc" (radio control) CLIENT_DISCONNECT message 524 to the radio IN process 304. The radio IN process 304 returns the same message 526 to acknowledge that the connection has been terminated. When this occurs, the capture tool 502 is shutdown by the content provider client 100.

Playback Session

[0053] FIG. 6A is a data flow diagram showing the data flow architecture for playback of a multimedia data stream. A user is provided with a user client process 202, playback tool 204, and decoder plugins 206 as described above, which may be considered to be a single "playback tool" executing on the user client 200. The playback tool includes a playback form 602 to accept user selection of a channel. The channel selected is communicated to the processes described above with respect to FIG. 1A, which may be considered to be a single "broadcast system" 604 from the user's perspective.

[0054] The broadcast system 604 manages the delivery of radio control messages ("rc") and radio broadcast ("rb") data to the user client 200 through a Terminal Information Handler, as described above. There are four phases of playback sessions: Start Playback, Playback, Switching Channels, and Ending Playback.

[0055] FIG. 6B is a data flow diagram showing the data flow for starting a playback session. To establish a playback session, the user uses the playback tool to download a selection form from the host 300. In particular, the broadcast system 604 will have initialized the playback form 602 with the available channels. A channel is selected from the playback form 602, and the playback tool notifies the broadcast system 604 of the selected channel through a CHANNEL REQUEST command message 606.

[0056] Once the broadcast system 604 has been informed that the user client 200 has selected a particular radio channel for playback *(i.e.,* the tuner process 316 receives an "rw" token with a channel name), the broadcast system 604 initiates the broadcast with a "rc" (radio control) START_CMD message 608. This initialization message is immediately followed by "rb" (radio broadcast) data messages 610. The data messages 612 preferably alternate header information and multimedia data. In the preferred embodiment, an identical header token is repeated with every data transmission so that users can join in the broadcast at any particular moment.

IPR2022-01227
EXHIBIT 1003 - PAGE 00783

The final "rb" data packet 614 transmitted contains an "end-of-file" (EOF) identifier. In the preferred embodiment, if the playback content loops, the first data header will be transmitted again, and the broadcast will restart. Otherwise, a special "rc" STOP_PLAY_0 message (not shown) is sent to shut down that particular channel.

[0057] FIG. 6C is a data flow diagram showing the data flow for switching a playback channel. The basic process is similar to the process described for FIG. 6B. However. when a program is playing, the user can request a different program using the playback form 602. The playback tool notifies the broadcast system 604 of the new selected channel through a channel request message 606'. The broadcast system 604 then initiates the new broadcast with a "rc" (radio control) START_CMD message 608'. This initialization message is immediately followed by "rb" (radio broadcast) data messages 610' for the new channel. Whenever a new channel is started, the stream name and stream description is sent to the appropriate auxiliary tool 208 for display if provided by the playback form 602.

[0058] FIG. 6D is a data flow diagram showing the data flow for ending a playback session. The playback session is stopped by closing the open playback form 602 or by pressing the appropriate cancel control. When this happens, a TERMINATE SESSION command message 616 is sent to the broadcast system 604 to force a STOP_PLAY_0 618 message to be sent back to the user client 200. When this occurs, the playback tool is shutdown.

Computer Implementation

[0059] The invention may be implemented in hardware or software, or a combination of both. However, preferably, the invention is implemented by means of a computer program executing on one or more programmable systems each comprising at least one processor. a data storage system (including volatile and non-volatile memory and/or storage elements). at least one input device, and at least one output device. Program code is applied to input data to perform the functions described herein and generate output information. The output information is applied to one or more output devices, in known fashion.

[0060] Each such program may be implemented in any desired computer language (including machine, assembly, high level procedural, or object oriented programming languages) to communicate with a computer system. In any case, the language may be a compiled or interpreted language.

[0061] Each such computer program is preferably stored on a storage media or device *(e.g.,* ROM or magnetic media) readable by a general or special purpose programmable computer, for configuring and operating the computer when the storage media or device is read by the computer to perform the procedures described herein. The inventive system may also be considered to be implemented as a computer-readable storage medium, configured with a computer program, where the storage medium so configured causes a computer to operate in a specific and predefined manner to perform the functions described herein.

[0062] A number of embodiments of the present invention have been described. Nevertheless, it will be understood that various modifications may be made without departing from the spirit and scope of the invention. For example, while specific controls have been shown in FIGS. 2-4, other controls may be used to provide similar functionality. Further, while FIGS. 5A-5D and 6A-6D show preferred messaging and data flow protocols, other messaging and data flow protocols may be used to provide similar functionality. Accordingly, other embodiments are within the scope of the following claims.

Claims

1. A system for playback of live and pre-recorded multimedia data in real-time over a large scale communication network, including:

(a) a computer system having a plurality of terminal information handlers for managing general information flow to and from a plurality of users;

(b) an output process for assembling multiple multimedia data streams for distribution;

(c) at least one broadcast process, in communication with the output process, for distributing the assembled multiple multimedia data streams to each of the terminal information handlers; and

(d) a selector process, in communication with the terminal information handlers, for receiving a channel request from a user through an terminal information handler associated with the user, mapping the channel request to a corresponding one of the multiple multimedia data streams, and enabling transmission of the corresponding one multimedia data stream to the user through the associated terminal information handler.

2. The system of claim 1, further including an input process for receiving multimedia data streams for distribution.

3. The system of claim 1, further including at least one storage device for storing multimedia data streams for later transmission.

4. A method for playback of live and pre-recorded multimedia data in real-time over a large scale communication network, including the steps of:

(a) providing a plurality of terminal information handlers for managing general information flow to and from a plurality of users;

(b) assembling multiple multimedia data streams for distribution;

(c) distributing the assembled multiple multimedia data streams to each of the terminal information handlers; and

(d) receiving a channel request from a user through an terminal information handler associated with the user, mapping the channel request to a corresponding one of the multiple multimedia data streams, and enabling transmission of the corresponding one multimedia data stream to the user through the associated terminal information handler.

5. The method of claim 4, further including the step of receiving multimedia data streams for distribution.

6. The method of claim 4, further including the step of storing multimedia data streams for later transmission.

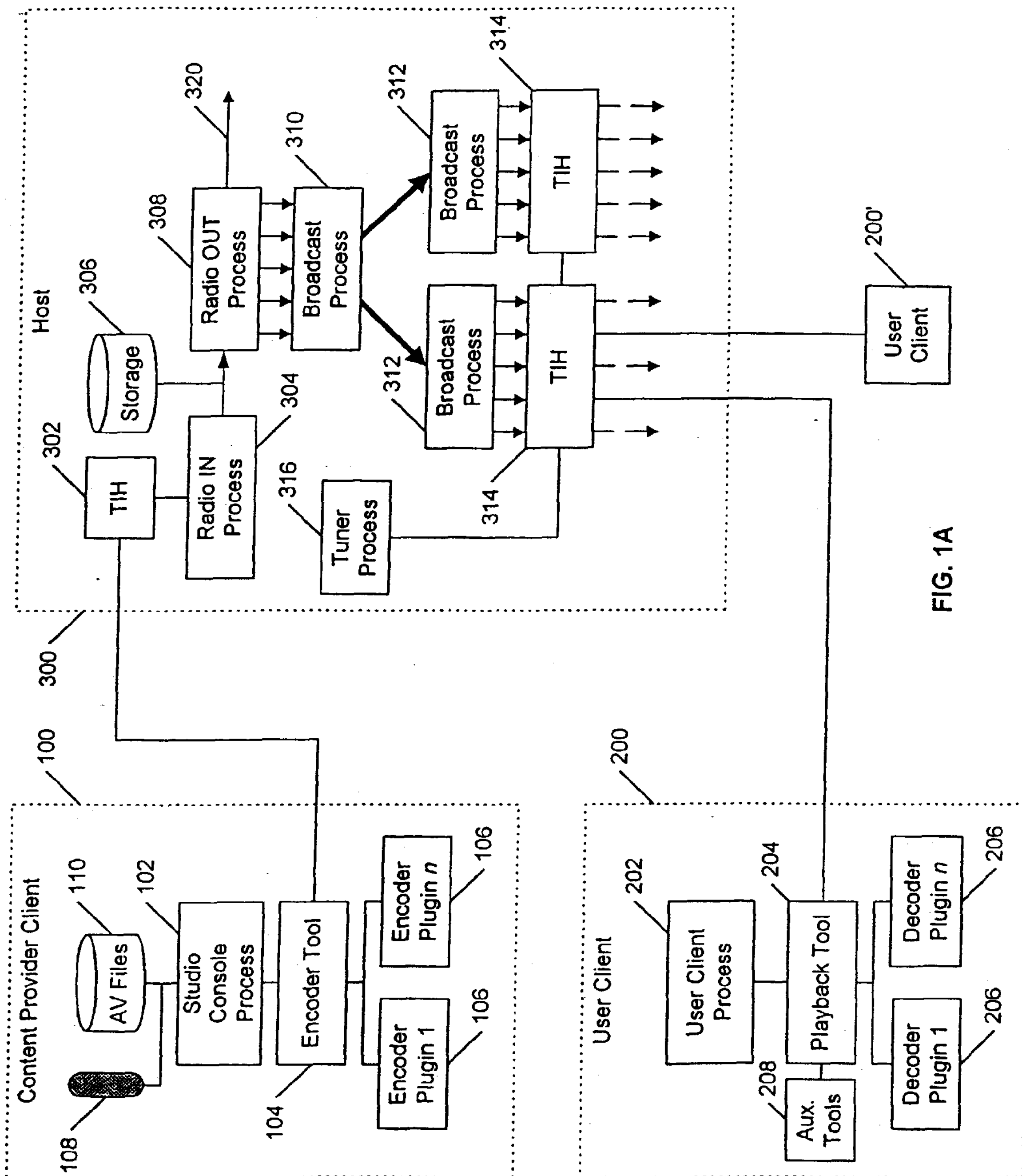

Host
300
TIH
302
Radio IN Process
304
Storage
306
Radio OUT Process
308
320
Broadcast Process
310
Broadcast Process
312
Broadcast Process
312
TIH
314
TIH
314
Tuner Process
316
User Client
200'
Content Provider Client
100
AV Files
110
108
Studio Console Process
102
Encoder Tool
104
Encoder Plugin 1
106
Encoder Plugin n
106
User Client
200
User Client Process
202
Playback Tool
204
Aux. Tools
208
Decoder Plugin 1
206
Decoder Plugin n
206

FIG. 1A

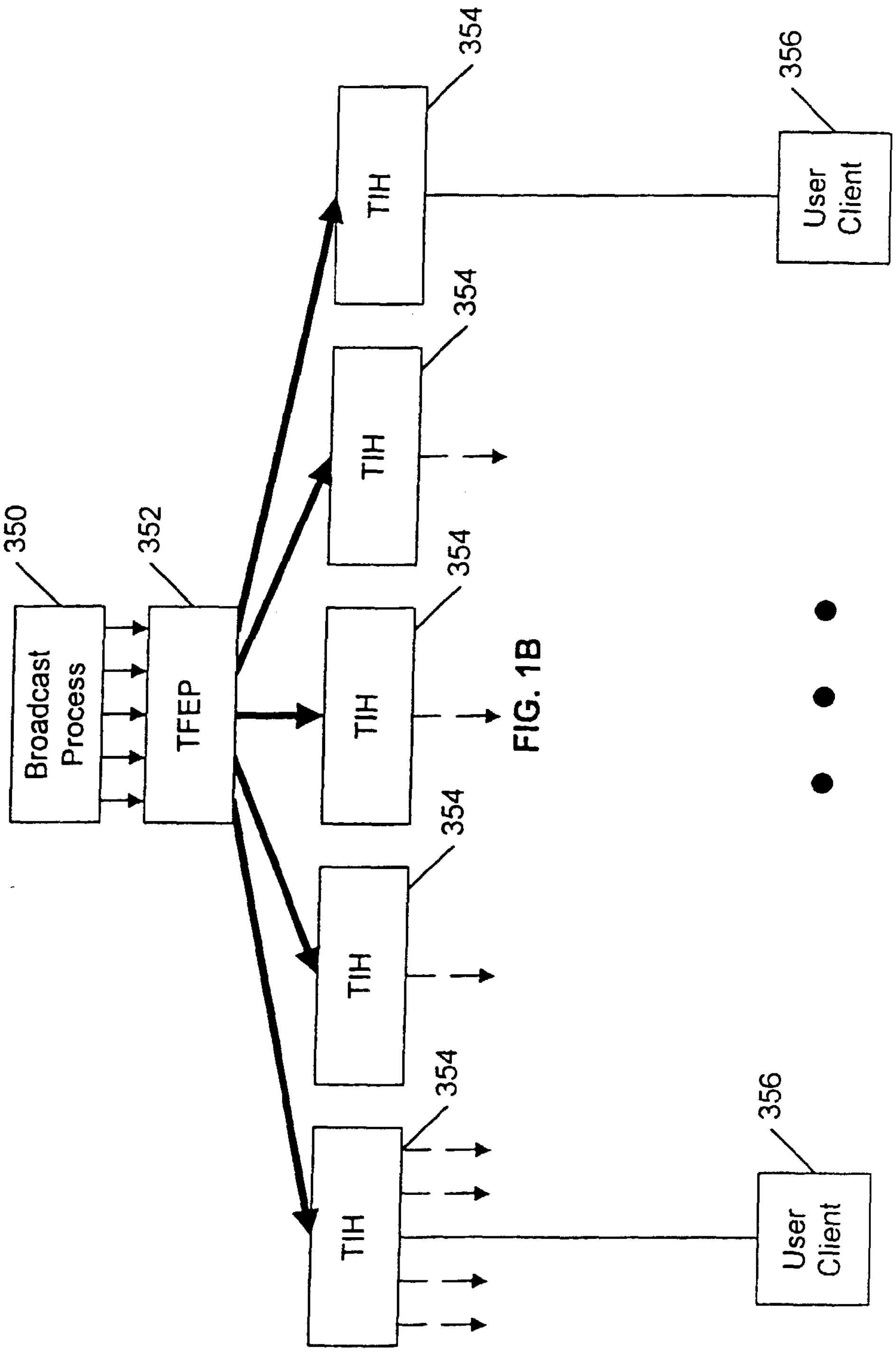
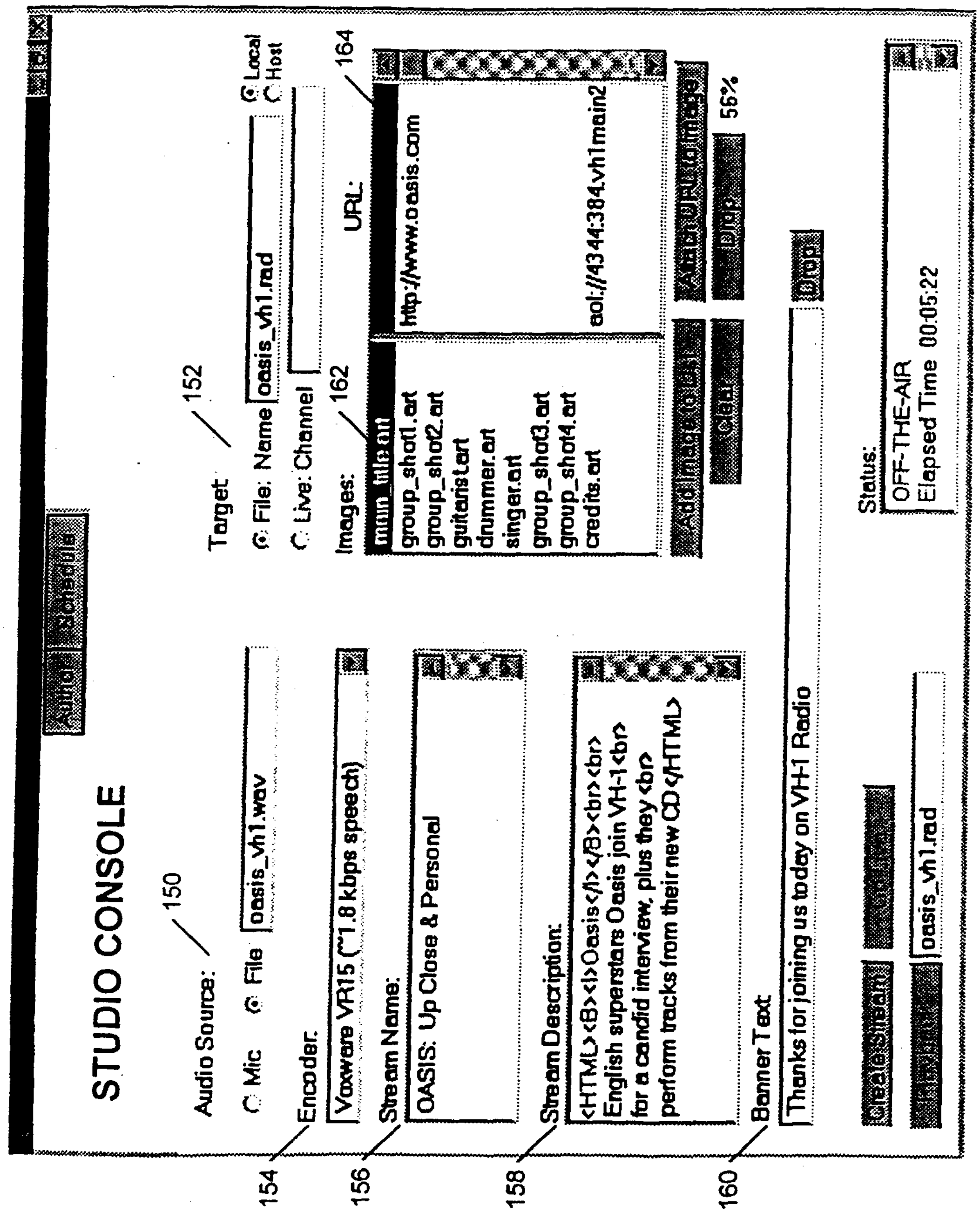

350
Broadcast Process
352
TFEP
354
TIH
354
TIH
354
TIH
354
TIH
354
TIH
356
User Client
356
User Client

FIG. 1B


STUDIO CONSOLE
Admin
Schedule
Audio Source:
Mic
File
oasis_vh1.wav
Encoder:
Voxware VR15 (~1.8 kbps speech)
Stream Name:
OASIS: Up Close & Personal
Stream Description:
<HTML><B><I>Oasis</I></B>

English superstars Oasis join VH-1
for a candid interview, plus they
perform tracks from their new CD</HTML>
Banner Text:
Thanks for joining us today on VH-1 Radio
Create Stream
oasis_vh1.rad
Target:
File: Name
oasis_vh1.rad
Local
Host
Live: Channel
Images:
URL:
main_title.art
group_shot1.art
group_shot2.art
guitarist.art
drummer.art
singer.art
group_shot3.art
group_shot4.art
credits.art
http://www.oasis.com
aol://4344:384.vh1main2
Add Image to List
Attach URL to Image
Clear
Drop
56%
Drop
Status:
OFF-THE-AIR
Elapsed Time 00:05:22
150
152
154
156
158
160
162
164

FIG. 2

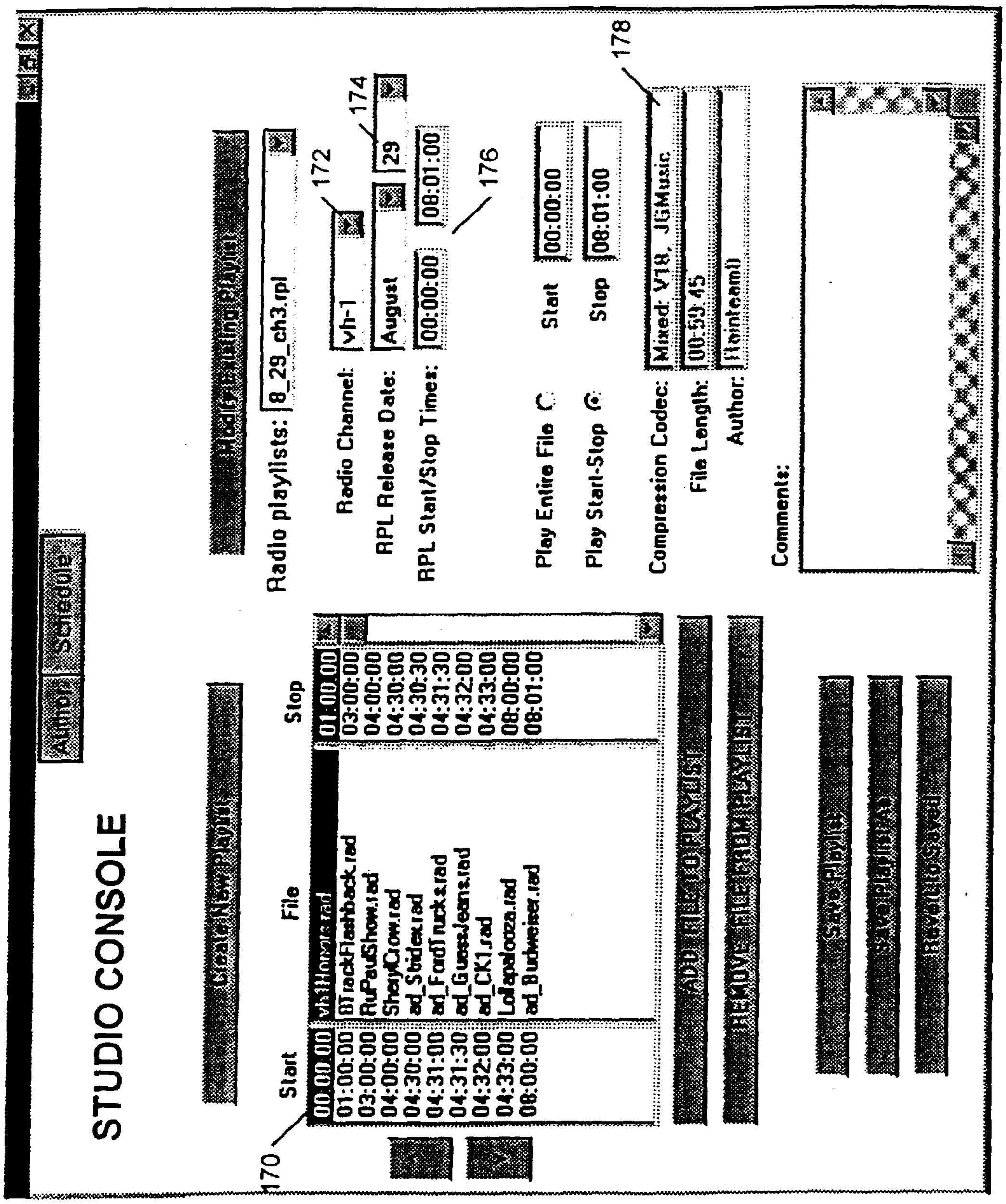


FIG. 3

IPR2022-01227
EXHIBIT 1003 - PAGE 00789

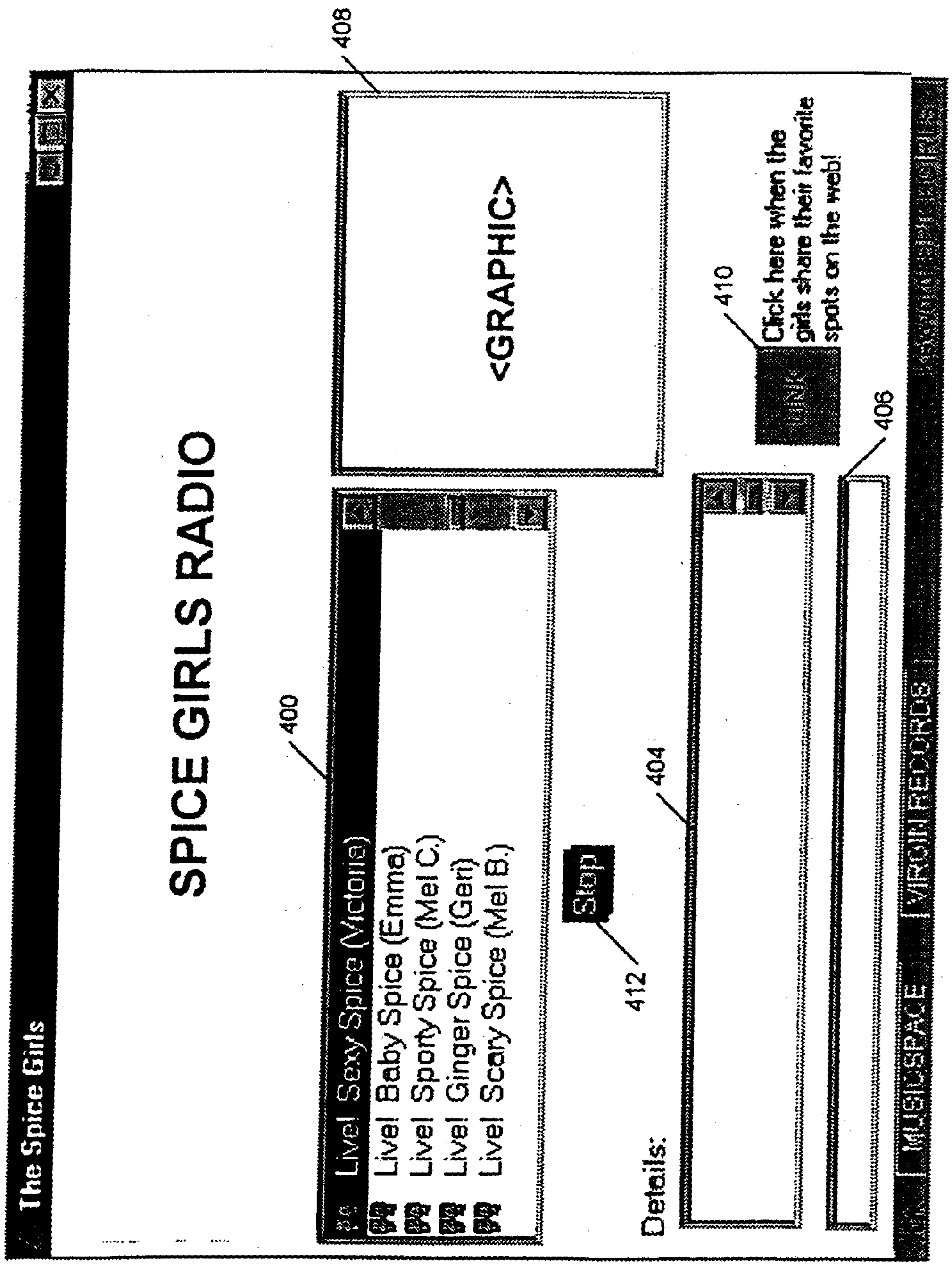

The Spice Girls
SPICE GIRLS RADIO
400
Live! Sexy Spice (Victoria)
Live! Baby Spice (Emma)
Live! Sporty Spice (Mel C.)
Live! Ginger Spice (Geri)
Live! Scary Spice (Mel B.)
Stop
412
Details:
404
406
408
<GRAPHIC>
410
LINK
Click here when the girls share their favorite spots on the web!
MUSIC SPACE
VIRGIN RECORDS

FIG. 4

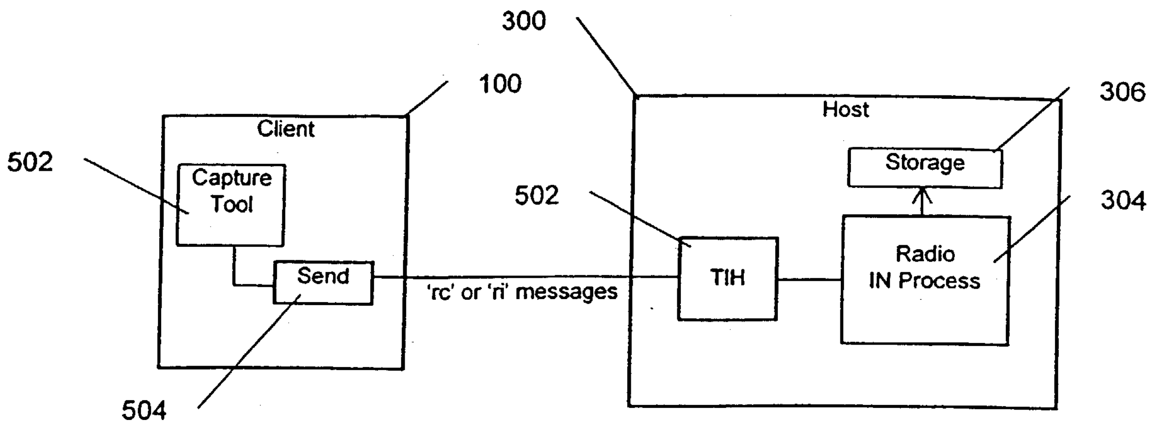

300
100
306
Host
Client
502
Storage
Capture
Tool
502
304
Radio
IN Process
TIH
Send
'rc' or 'ri' messages
504

FIG. 5A

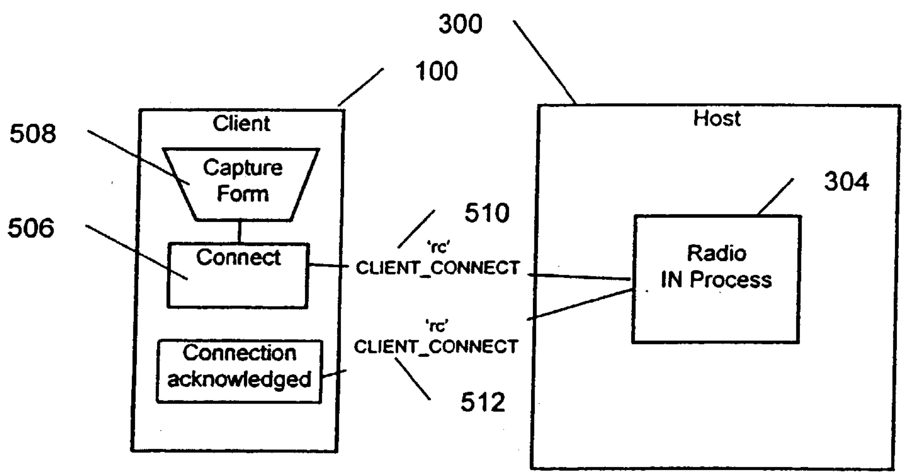

300
100
Host
Client
508
Capture
Form
304
510
506
Connect
'rc'
CLIENT_CONNECT
Radio
IN Process
'rc'
CLIENT_CONNECT
Connection
acknowledged
512

FIG. 5B

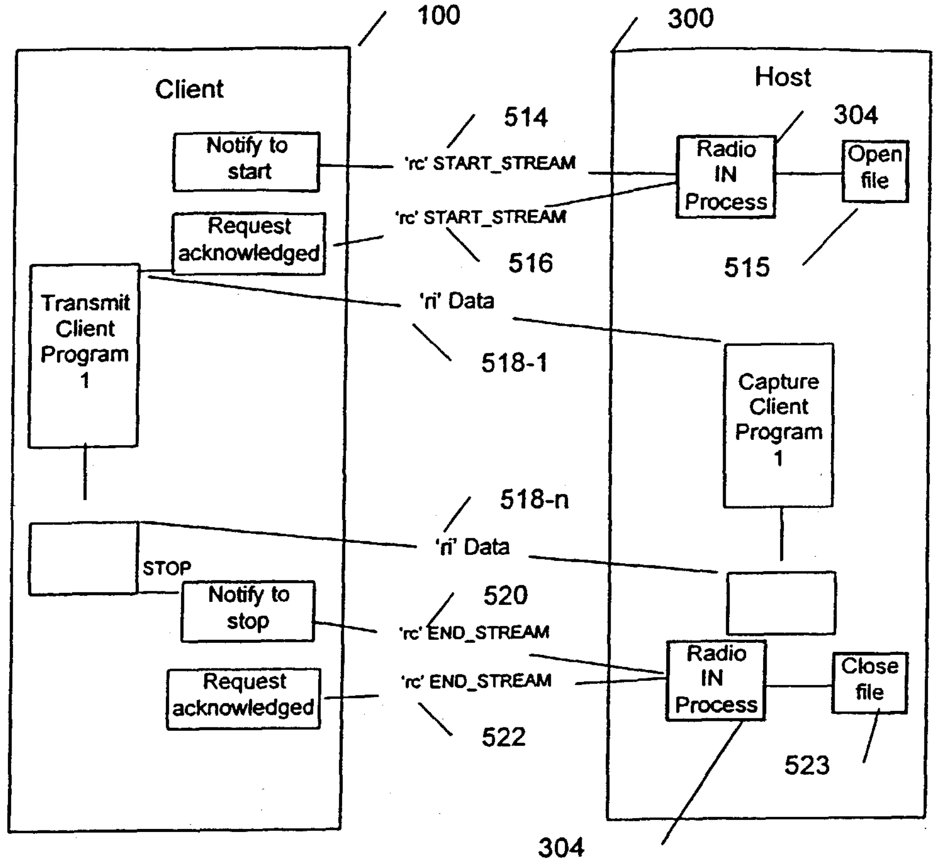

100
300
Client
Host
514
304
Notify to start
'rc' START_STREAM
Radio IN Process
Open file
Request acknowledged
'rc' START_STREAM
516
515
Transmit Client Program 1
'ri' Data
518-1
Capture Client Program 1
518-n
'ri' Data
STOP
Notify to stop
520
'rc' END_STREAM
Radio IN Process
Close file
Request acknowledged
'rc' END_STREAM
522
523
304

FIG. 5C

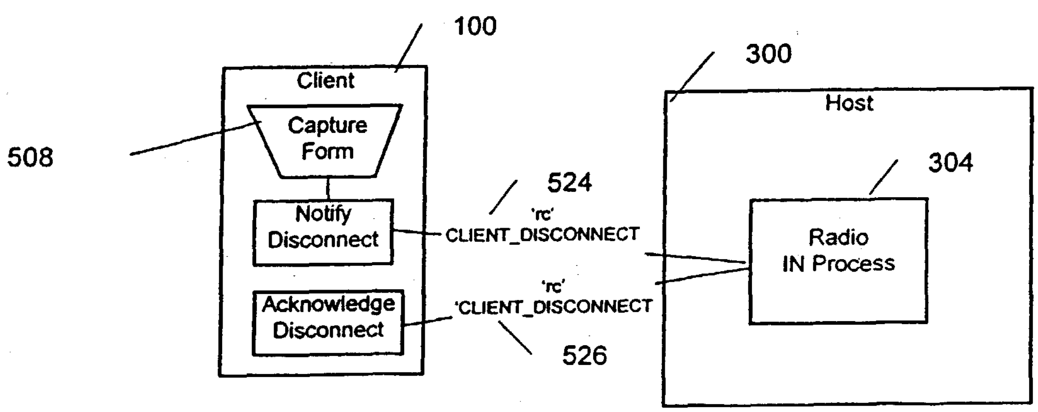

100
300
Client
Host
508
Capture Form
304
524
Notify Disconnect
'rc' CLIENT_DISCONNECT
Radio IN Process
Acknowledge Disconnect
'rc' 'CLIENT_DISCONNECT
526

FIG. 5D

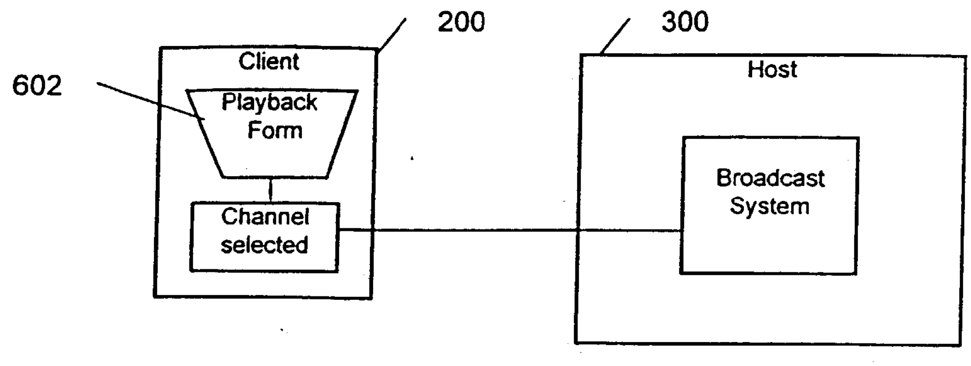

200
300
602
Client
Playback Form
Channel selected
Host
Broadcast System

FIG. 6A

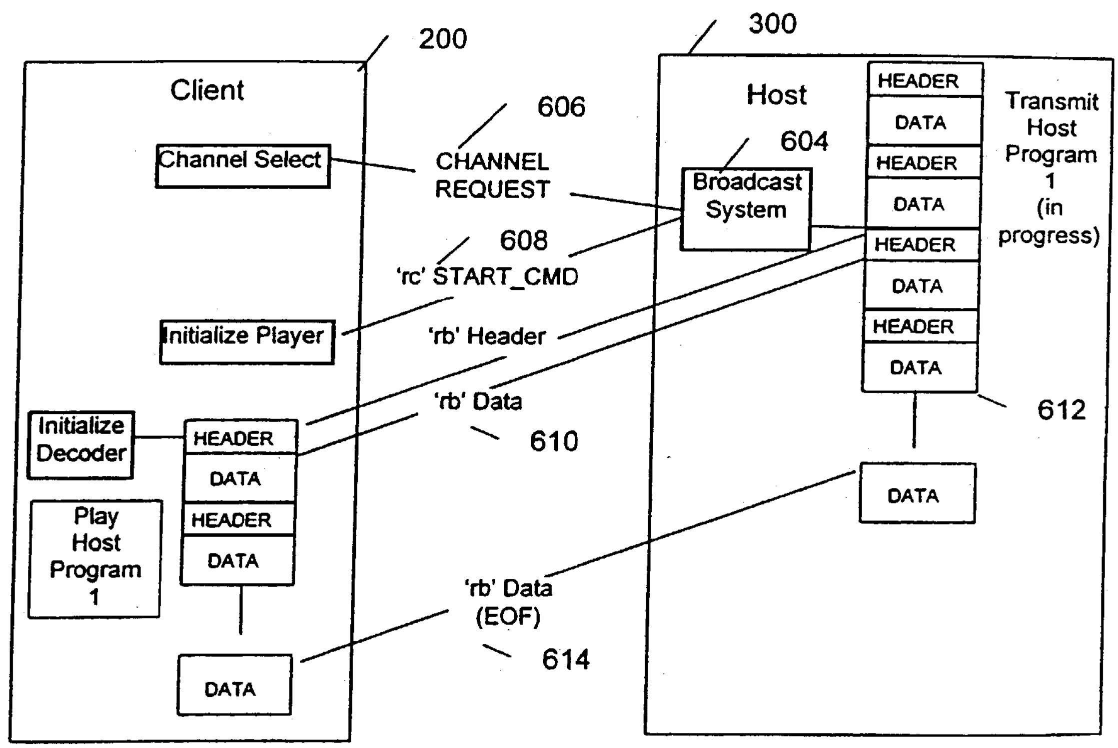

200
300
Client
Channel Select
606
CHANNEL REQUEST
Host
604
Broadcast System
HEADER
DATA
HEADER
DATA
HEADER
DATA
HEADER
DATA
Transmit Host Program 1 (in progress)
608
'rc' START_CMD
Initialize Player
'rb' Header
612
Initialize Decoder
HEADER
DATA
HEADER
DATA
'rb' Data
610
Play Host Program 1
DATA
'rb' Data (EOF)
614
DATA

FIG. 6B

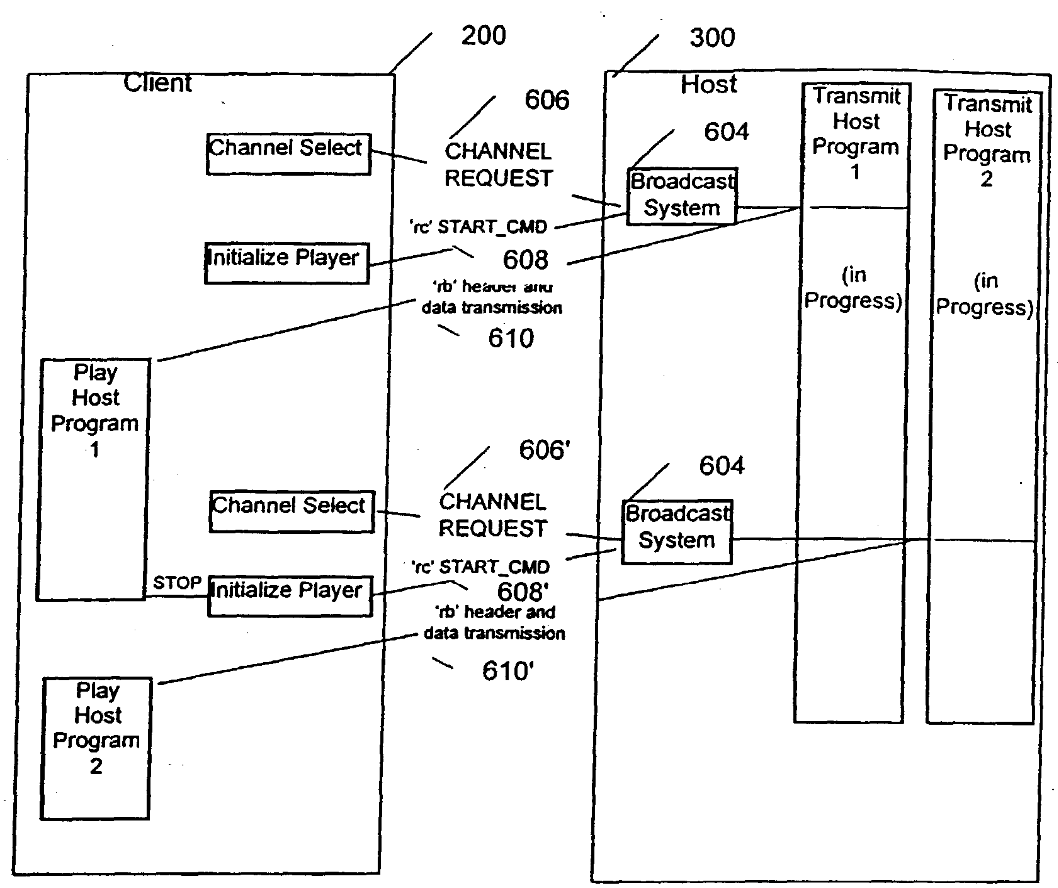

200
300
Client
Host
Transmit Host Program 1
Transmit Host Program 2
606
604
Channel Select
CHANNEL REQUEST
Broadcast System
'rc' START_CMD
Initialize Player
608
'rb' header and data transmission
610
(in Progress)
(in Progress)
Play Host Program 1
606'
604
Channel Select
CHANNEL REQUEST
Broadcast System
'rc' START_CMD
STOP
Initialize Player
608'
'rb' header and data transmission
610'
Play Host Program 2

FIG. 6C

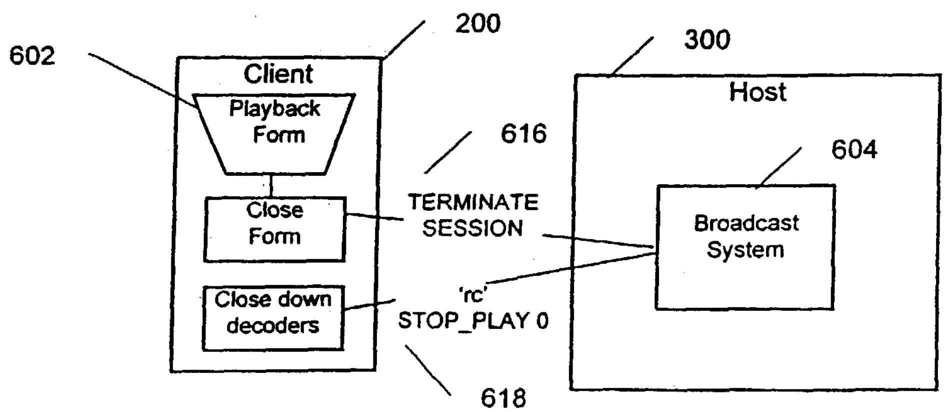

602
200
300
Client
Playback Form
Host
616
604
Close Form
TERMINATE SESSION
Broadcast System
Close down decoders
'rc' STOP_PLAY 0
618

FIG. 6D

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 99 30 6950

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	US 5 778 187 A (BUTTERWORTH JAMES F ET AL) 7 July 1998 (1998-07-07) * abstract * * column 1, line 5 - line 15 * * column 3, line 4 - column 4, line 38 * * column 6, line 6 - line 21 * * column 17, line 4 - line 31 * * figure 18 * ---	1-6	H04L12/18 H04L29/06
X	US 5 557 724 A (KEMBEL JOHN ET AL) 17 September 1996 (1996-09-17) * abstract * * column 4, line 2 - line 60 * * column 8, line 1 - line 21 * * column 8, line 62 - column 9, line 6 * * column 10, line 53 - line 60 * * figures 1,3,15-18 * -----	1-6	
			TECHNICAL FIELDS SEARCHED (Int.Cl.7)
			H04L

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	17 December 1999	Poggio, F

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 99 30 6950

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

17-12-1999

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5778187 A	07-07-1998	AU 3002097 A	26-11-1997
		WO 9742582 A	13-11-1997
		US 5983005 A	09-11-1999
US 5557724 A	17-09-1996	NONE	

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 113 642 A2

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
04.07.2001 Bulletin 2001/27

(51) Int Cl.7: **H04L 29/06**

(21) Application number: **00311131.7**

(22) Date of filing: **13.12.2000**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE TR
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **16.12.1999 US 461808**

(71) Applicant: **ACTV, INC.**
New York, NY 10020 (US)

(72) Inventors:
- **Ullman, Craig D.**
Brooklyn, New York 11201 (US)
- **Abato, Michael R.**
White Plains, New York 10605 (US)
- **Harrington, Jeffrey M.**
Brooklyn, New York 11215 (US)
- **Duda, Carl R.**
Santa Barbara, California 93103 (US)

(74) Representative: **Needle, Jacqueline**
W.H. BECK, GREENER & CO
7 Stone Buildings
Lincoln's Inn
London WC2A 3SZ (GB)

(54) **Enhanced video programming system and method using a local host for network communication**

(57) A local host is used to facilitate network communication between a user machine (202) and a remote server (224) by way of a network (222) such as the Internet. A web browser (204) is included in the user machine (202) and is provided with hidden frames or layers in which pages and/or applets are stored. The hidden frames or layers also include a plug-in (212) programmed to function as a web server at the user machine. The plug-in (212) issues a command which causes the web browser (204) to establish a connection with the remote server (224). This initiates the applet (208) and also causes the plug-in (212) to open a TCP/IP socket (220) to open and maintain communication between the user machine (202) and the remote server (224). The applet (208) may then communicate with the remote server (224) by way of the plug-in (212).

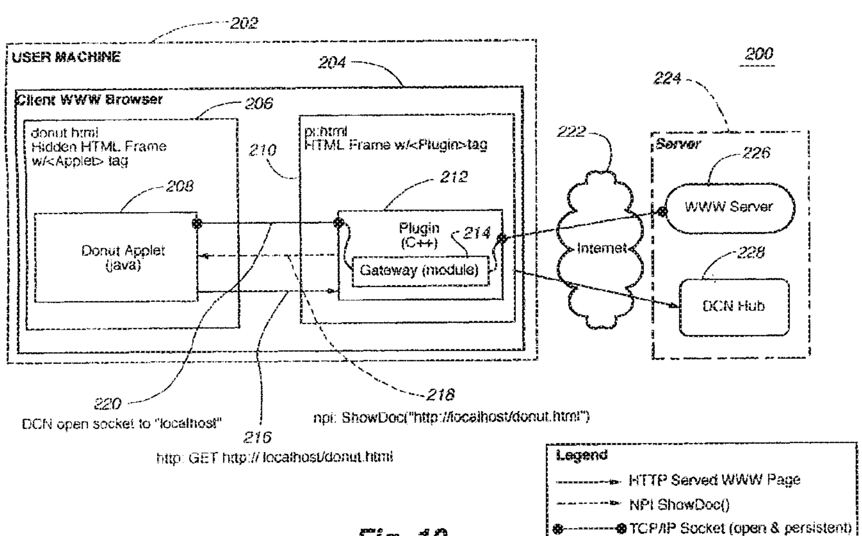


Fig. 10

EP 1 113 642 A2

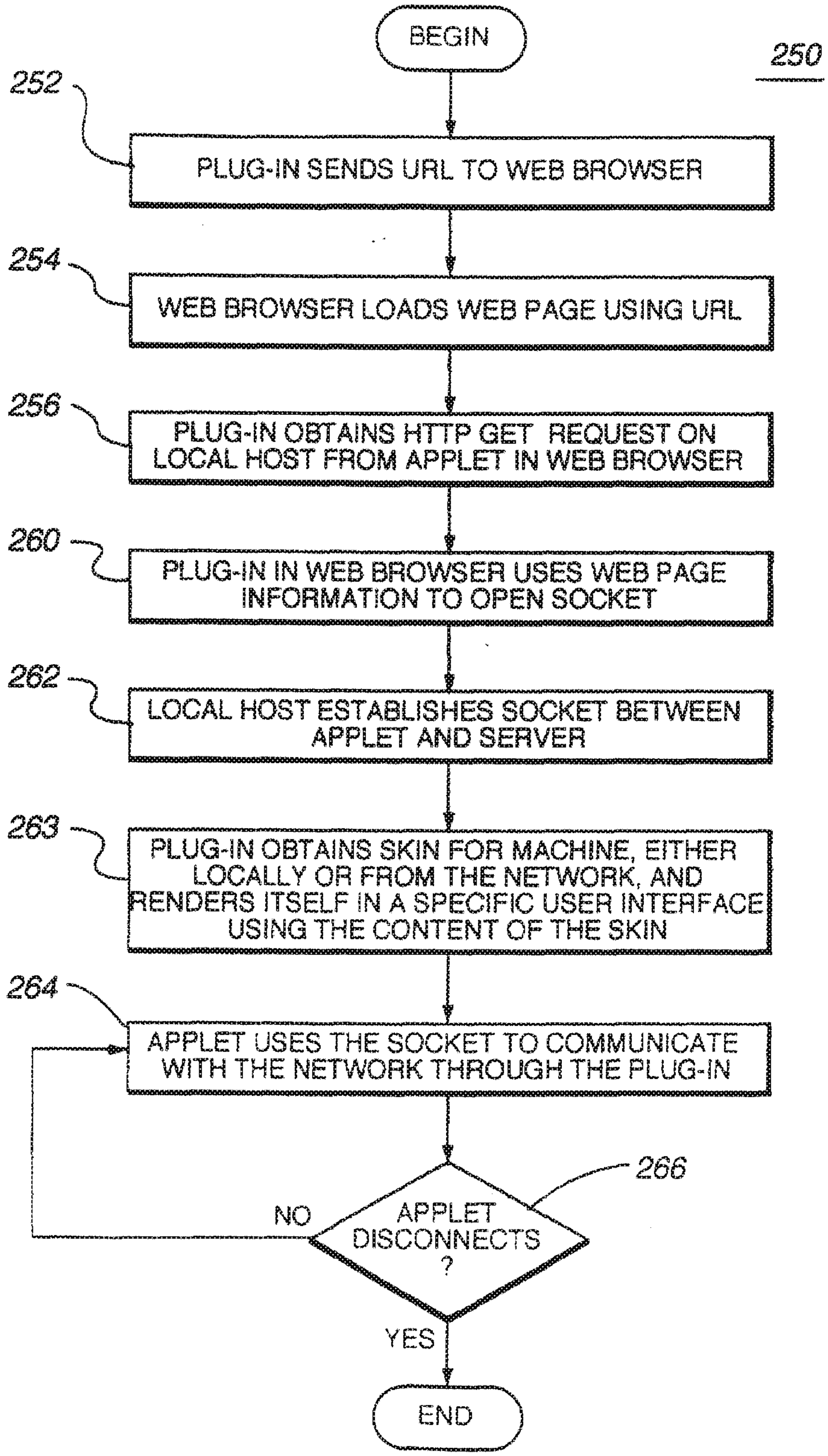


Fig. 11

IPR2022-01227
EXHIBIT 1003 - PAGE 00798

Description

[0001] The present invention relates to a method and apparatus for network communication by distributing network addresses to user machines.

[0002] Computers have the capability to provide massive amounts of educational and entertainment information by way of the Internet. Currently, on-line systems offer a variety of different services to users, including news feeds, electronic databases (either searchable by user directly on the on-line system, or downloadable to the user's own computer), private message services, electronic newsletters, real time games for play by several users at the same time, and job placement services, to name a few. However, currently most on-line communications occur merely through text. This is in contrast to the audio/visual presentation of the alternative electronic medium, television. However, it is expected that as multi-media's incessant growth continues, audio/visual programs will proliferate and text will become less and less dominant in the on-line environment.

[0003] Even though these programs will be introduced, the Internet will remain essentially user unfriendly due to its very massiveness, organization, and randomness. Simply stated, there is no order or direction in the Internet. Specific pieces of information can be hard to find, and it is even harder to put that piece of information into a meaningful context.

[0004] Television, on the other hand, has been criticized for being a passive medium. Whilst interactive television systems have increased the level of user interaction, and thus, provided greater learning and entertainment opportunities, vast information resources such as databases are inaccessible from such a medium.

[0005] The present invention seeks to close the gap between video programming and the vast information resources of the Internet.

[0006] According to a first aspect of the present invention, there is provided a method for providing a machine with network communication with a server, comprising:

receiving a request for a network connection;
generating a reference to a local connection in response to the request;
detecting communication at the local connection; and
establishing a connection with the local connection in response to the detecting and for facilitating network communication with the machine.

[0007] In embodiments, a local web server function is provided at the local connection.

[0008] Preferably, the receiving step comprises receiving networked content, and/or a request for a web page from a web browser, and/or a request from an applet.

[0009] The present invention also extends to apparatus for providing a machine with network communication with a server, comprising:

receiving means for receiving a request for a network connection;
generating means for generating a reference to a local connection in response to the request; and
detecting means for detecting communication at the local connection; and
connection means for establishing a connection with the local connection in response to the detecting and for facilitating network communication with the machine.

[0010] In embodiments, the receiving means may be arranged to receive networked content, and/or a request for a web page from a web browser, and/or a request from an applet.

[0011] In an embodiment, the apparatus further comprises detecting means for detecting communication at the local connection, which detecting means may be arranged to receive information for use in opening a socket with the local connection.

[0012] According to a further aspect of the invention, there is provided a method for distributing network addresses to user machines for use in obtaining content associated with the addresses, comprising:

receiving a plurality of network addresses identifying network locations of particular content;
receiving a time value of a time parameter associated with each of the network addresses; and
transmitting each of the network addresses to a user based upon each of the corresponding time values.

[0013] The present invention also extends to apparatus for distributing network addresses to user machines for use in obtaining content associated with the addresses, comprising:

receiving means for receiving a plurality of network addresses identifying network locations of particular content;
means for receiving a time value of a time parameter associated with each of the network addresses; and
transmission means for transmitting each of the network addresses to the user machines based upon each of the corresponding time values.

[0014] According to a still further aspect of the present invention, there is provided a method for distributing dynamic network addresses to user machines for use in obtaining content associated with the addresses, comprising:

receiving a network address containing a variable, the network address identifying varying network lo-

IPR2022-01227
EXHIBIT 1003 - PAGE 00799

cations of particular content based upon the variable;
receiving an associated description for the network address;
resolving the variable in the network address based upon information related to an intended recipient of the network address; and
transmitting the network address along with the associated description to a user machine corresponding with the intended recipient.

[0015] The present invention also extends to apparatus for distributing dynamic network addresses to user machines for use in obtaining content associated with the address, comprising:

receiving means for receiving a network address containing a variable, the network address identifying varying network locations of particular content based upon the variable, and an associated description for the network address;
resolution means for resolving the variable in the network address based upon information related to an intended recipient of the network address; and
transmission means for transmitting the network address along with the associated description to a user machine corresponding with the intended recipient.

[0016] Embodiments of the present invention will hereinafter be described, by way of example, with reference to the accompanying drawings, in which;

Figure 1 is a diagram showing the receipt and decoding of video signals at a subscriber location using a method of the invention;
Figure 2 is a diagram showing an alternative embodiment to achieve the integration of Internet information with video content;
Figure 3 is a flow diagram of the basic software of the invention;
Figure 4 is a diagram showing an embodiment in which URLs are directly transmitted to a user;
Figure 5 shows an embodiment of a system comprising a digital cable box;
Figure 6 shows an embodiment of a system including a digital T.V.;
Figure 7 shows an example of a user interface;
Figure 8 shows an example of a display providing a user interface;
Figure 9 is a diagram showing an embodiment of a system having distributed communication servers;
Figure 10 shows an example of a logical structure for a local host;
Figure 11 is a flow chart of a method for implementing a local host of network communication;
Figure 12 illustrates a user interface displaying a playlist to an author of the playlist content;
Figure 13 shows a user interface which permits an author to edit the content of a playlist;
Figure 14 illustrates a user interface displaying playlist items pushed to a user;
Figure 15 is a diagram of a data structure for playlist entities;
Figure 16 is a flow chart of a method for implementing a playlist; and
Figure 17 is a flow chart of a method for processing dynamic URLs.

[0017] Figure 1 illustrates an embodiment of a computer based system for receiving a video program along with embedded uniform resource locators (URLs) which direct a user's computer 16 to address locations, or web sites, on the Internet 20 to retrieve related web pages. The web pages correspond to the video presentation. The particular video programming can be delivered in analog, digital or digitally compressed formats (e.g. MPEG2) via any transmission means, including satellite, cable, wire, television broadcast or sent via the web.

[0018] The video programming is preferably created at a centralized location, for example, as content creation 4 indicated in Figure 1, for distribution to subscribers. Program creation may be accomplished by any appropriate means. After a video program is created, uniform resource locators (URLs) are embedded. In one embodiment, the URLs are embedded into the vertical blanking interval of the video programming by a URL encoder 8, as shown in Figure 1. In this embodiment, the URLs are encoded onto eight fields of line 21 of the VBI. Line 21 is the line associated with close captioning, among other things. However, the URLs may additionally and/or alternatively be embedded in other fields of the VBI, in the horizontal portion of the video, as part of the audio channel, in any subcarrier to the video, or if the video is digital, in one of the data fields.

[0019] Although Figure 1 shows the video with the URLs broadcast over the same transmission line, the URLs may be sent down independently of the video program on a data channel. In this embodiment, the URLs may be forwarded to the remote sites either prior to initiation or during the program. Preferably, the URLs have associated time stamps which indicate to the subscriber stations when, during the video program, to display the particular web pages addressed by the URLs. Alternatively, the user can select when to call the particular web pages for display with the video program.

[0020] The particular information in line 21 is not part of the visual part of the program, and thus, is not perceptible to the human eye, thereby making it ideal to send data information to the users. Whilst the bandwidth capacity of line 21 is limited, as a system as described transmits only the URLs, and not full web pages, there is more than enough capacity. Furthermore, no additional hardware is necessary at the computer 16 to receive the video and retrieve the web pages.

[0021] Once the video program is created, it may be transmitted to user sites over any transmission means, including broadcast, cable, satellite, or Internet, and may reside on video servers. Furthermore, the video program, with or without embedded URLs, may be encoded onto storage means such as a video tape, for example of VHS or Beta format, or an optical disc such as CD or DVD, or any other medium.

[0022] Preferably, each receiver station comprises any Intel x86 machine (preferably a 486 processor, pentium processor, etc), an Apple Computer, UNIX or any other type of standard computer workstation. The local computer 16 is preferably connected to either a cable and/or broadcast television or to a local VCR or other video source. At each subscriber site, the local personal computer 16 preferably receives the cable transmission by cable connection on the back of the personal computer 16. The video/audio program may be processed for display on the computer screen using a PC card capable of displaying video signals on a computer monitor in an appropriate TV format such as PAL or NTSC. One example of a PC card is a WinTV card. In addition to the cable connection, there is the Internet 20 connection created concurrently with the cable connection.

[0023] The Internet 20 connection may be via high-speed line, RF, conventional modem or by way of two-way cable carrying the video programming. The local PC 16 has Internet access via, for example, an ASCII software mechanism. In an embodiment, at each subscriber site, an associated local URL decoder 12 extracts the URLs, preferably embedded in the vertical blanking interval, with the use of a suitable VBI decoder device. The URL decoder 12 may be either a stand-alone unit or a card which is implemented into the personal computer 16.

[0024] In the embodiment shown in Figure 2, the uniform resource locators (URLs) are encoded into the video as described above. Again, the URLs are preferably encoded onto eight fields of line 21 of the VBI, but may also be sent independently of the video. In this embodiment, a URL decoder 24 is located at the server site rather than at the subscriber location. When the decoder 24 receives the video program signal, it strips out the URL codes on line 21 of the VBI and delivers these codes independently to an Internet server 28. The URL code is then subsequently delivered over the Internet 20 to the user PC 16. Simultaneously, the video is broadcast over conventional broadcast or cable transmission means 36 to the user's personal computer 16.

[0025] The alternative shown in Figure 4, does not use the VBI. In this embodiment, the system runs an online service over the Internet 20. This service is in the form of an Internet web site 62 which provides a user-interface to a database 78 and to one or more associated data servers 90. The service provides member accounts to TV broadcasters 66 who sign up to use the illustrated system in conjunction with their broadcasts. Each member broadcaster will enter the service at their computer 70 through web browser software 74 using their member account by entering various identification and password information. Once within their account, the member will be provided with a graphical user interface for pre-scheduling URLs for transmission to users 118 over a direct Internet connection 94 at particular times of day. The same user interface, or a variation of it, can be used by broadcasters for live transmission 82 of URLs to users at the same time as a broadcast 86.

[0026] One example of this interface might be a scheduling calendar (daily, weekly, monthly, yearly) in which the broadcaster 66 may allocate time periods which coincide with their broadcasts 86, and during which they will send out URLs to their users to link to web pages. For each time period (for example, a particular hour long period during the day) determined by the broadcaster 66 to be a broadcast period (a period during which they want to transmit URLs that correspond to a television show being broadcast from their TV broadcast facility 110 to the external TV 114 of the user 118 at that time), the broadcaster 66 may then enter a series of URLs into an associated file ("Link File") for transmission over the Internet 20 at that time. This Link File may have a user interface such as a spreadsheet, table, or list, or it may be simply a tab-delimited or paragraph-delimited text-file. As an example, each of the records in the Link File consists of a data structure which may contain information such as:

[0027] (<timecode>,<URL>,<label or title>, <additional information>,<additional information>,...)

[0028] The above data structure is just one example. The records in the Link File preferably specify the time, Internet address (i.e. URL), label (such as an associated name), and some optional additional information, for each web page the broadcaster 66 desires to launch during a show.

[0029] When a broadcaster 66 modifies their calendar and/or the Link File associated with any given time period(s) in their calendar, this information is saved into the database 78 which is attached to the site 62. Each broadcaster 66 may maintain multiple calendars in the database 78 if they broadcast in different time zones, for example.

[0030] The database 78 provides the Link File records for upcoming time periods to a server 90, which may be one server or a distributed network of server programs on multiple computers across the network, to be utilized for scaling to large national or global audiences. The server 90 provides the Link File records, including the URLs, to the user's personal computer 16, which is connected via a network. Examples of possible networks include the public Internet 94, a direct private network, or even a wireless network.

[0031] One feature of the embodiment illustrated in Figure 4 is that one or more broadcasters 66 may utilize the same schedule in the database 78 for their own broadcasts 86 or during the same broadcast. For example, a network broadcaster may develop a master schedule and various affiliate broadcasters may subscribe to that schedule or copy it (in the database) and add or delete specific URLs in the schedule for their local audiences or unique programming. This scheme enables affiliates to insert URLs for local advertisers or local subjects into a sequence of more general URLs provided by their network broadcaster 66. In other words, the affiliate can add links that ride on the network feed and then redistribute it to their local audiences.

[0032] The system of Figure 4 also enables personalization in the form of unique series of URLs specific to each user's unique profile, which are directly sent over the Internet 20 to each user's specific client software 106. This can be achieved from the broadcaster 66 to each individual user 118, or to particular collections of users. To accomplish personalization, the service may send a different stream of URLs to each user's client software program 106. The stream of URLs sent depends upon a user profile stored in the database 78 or the client software program 106, a user profile which is built on demand or over time for each user 118 based on criteria such as the location of the user, choices the user makes while using a client software program 106, choices the broadcaster 66 makes during a broadcast 86, or automatic choices made by an algorithm (such as a filter) residing on the service 62. Personalization enables each user to receive URLs which are uniquely relevant to their interests, demographics, history, or behaviour in the system.

[0033] Once the URLs have reached the personal computer 16, the operation of all of the systems shown in Figures 1, 2 and 4 is similar.

[0034] In one embodiment, a JAVA enabled browser 98 as well as specialized software 106 are installed on the computer 16. The JAVA enabled browser 98 allows the computer 16 to retrieve the web pages 102 and is presently the preferred software, as it is platform independent, and thus, enables efficient and flexible transfer of programs, images, etc., over the Internet 20. The specialized interface software 106 (hereinafter, "client software") acts as an interface between the video programming and the Internet functions. The client software 106 retrieves URLs from the video program (embodiment of Figure 1) or directly from the Internet connection (embodiments of Figures 2 and 4), interprets these URLs and directs the JAVA enabled browser 98 to retrieve the particular relevant web pages 102. The client software 106 also synchronizes web pages to the video content for display on the user's computer 16, as shown in Figures 3 and 4 and explained in more detail below.

[0035] As explained above, the URLs may be encoded and embedded into the video signal by inserting them into the vertical blanking interval (VBI).

[0036] Alternatively, the URLs may be entered by member TV broadcasters 66 along with specified times for transmitting the URLs to the user. At the appropriate times, the URLs are sent directly over the Internet to the user's PC 16 via the client software 106 over a direct point-to-point or multicasting connection.

[0037] The system may have the capability to detect identical URLs sent directly after one another and to cause the browser not to fetch URLs in these particular cases. As shown in Figure 3, once the URL code is received at the computer, the client software 106 first interprets the URL and determines in step 42 whether the particular URL has been received previously. If it has already been received, the next received URL is interpreted for determination of prior receipt. If the particular URL has not been detected before, the software checks for misspelling in step 46 and any other errors, and if errors exist, corrects these particular errors. Once again, it is determined whether the URL has been previously detected. If it has, the next URL is accessed in step 38. If the URL has not been detected, the specific URL is added to the URL list in step 54. The specific URL is then sent to the web browser, preferably a JAVA enabled browser 98. Upon receipt of the URL, the browser 98, in step 58, will access the web site address 122 (Figure 4) indicated by the URL and retrieve the cited web page(s) 102 via the Internet.

[0038] Viewers can view the integrated presentation in the following manner. As mentioned above, the video signal is processed and displayed on a video window on the PC screen using a WinTV card, for example. The corresponding audio is forwarded to the audio card and sent to the PC speakers.

[0039] The retrieved web pages 102, referenced by the URL, are optionally time stamped to be displayed on the computer screen when predetermined related video content is displayed in the video window, thus enhancing the video presentation by providing in-depth information related to the video content thereto. Another section on the screen is also preferably used to represent an operational control panel. This control panel provides a list of the URLs which have been broadcast and correspondingly received by the computer 16. This control panel is updated to add a URL code each time a new URL code is received by the PC 16. This list gives the subscriber the flexibility to go back and retrieve particularly informative or interesting web pages that have already been displayed earlier in the program, or alternatively, to print them out for future reference. Furthermore, the list may include URLs referring to web pages not displayed with the broadcast program, but which provide further information on a certain topic of interest to the viewer.

[0040] In an example, a viewer may begin watching a musical video featuring a band. As the video is received by the PC 16, URLs are either being received with the video signal or are being received directly via the Internet 20 or another data channel, and are interpreted by the client software 106. Upon direction and command, the JAVA enabled browser 98 retrieves particular web pages 102 from Internet 20 web sites identified in the URLs. These web pages 102 are then displayed on the video screen at particular times. So, for example, whilst the viewer is watching the music video, biographical information on the band may also be displayed adjacent to the video window. Web pages 102 may also include an upcoming concert schedule, and/or audio clips of the band's music may be downloaded from the Internet 20.

[0041] As another example, a user may be watching a program relating to financial news. Whilst the narrator is shown discussing high tech stocks, web pages corresponding to detailed financial performance information on high tech stocks, environment and characteristics may be displayed with the video on the computer screen. If the personalization features are included, web pages associated with a particular user's stock may be fetched and displayed on the computer screen with the video program. When the program narrator switches to a discussion on the weekly performance of the Dow Jones, web pages presenting related financial performance information may be simultaneously displayed.

[0042] A user may view the interactive program using a television set 114 or other display monitor in conjunction with the display screen of the personal computer 16. In this case, the relevant web pages are shown on the personal computer 16 whilst the video program is displayed on the television monitor 114. In this alternative, a cable set top box receives the television program from the multi-channel cable. The personal computer 16 also receives the video program from the multi-channel cable and extracts the URLs, embedded in the vertical blanking interval of the video signal or directly transmitted 94 over the Internet 20. The client software 106 extracts the URLs and retrieves the particular web pages as described above. The web pages are then synchronized with the particular video frames and presented to the user. It is understood that a hyperlink may exist on the web site that will allow the user to automatically load the client software and call up the specific television channel referenced in the web site. For example, someone browsing the Internet 20 may come upon a major television network's web site. It is possible then to scroll to an interesting story and then to click on an hyperlink to turn on the software which tunes the TV window to the network.

[0043] Instead of receiving the video program from a transmission means, the video program may be addressed directly from the user site if the video program, with or without embedded URLs, has been stored on appropriate means. The storage means may be a videotape in any format, such as VHS or Beta, or an optical disc in any format, such as DVD or CD-ROM. In this case, the user PC 16 and/or television 114 are connected to a video tape player, a disc drive, or other appropriate device.

[0044] Figures 5 and 6 show two alternative examples of systems which may be employed. As shown in Figure 5, a user may view an interactive program using a television set 18 or other display monitor in conjunction with a digital cable box 140. In this case, the digital cable box 140 performs the functions of the personal computer 16 shown in Figures 1, 2 and 4, and the client software is stored in memory in the digital cable box 140. In one embodiment, the digital cable box 140 includes two tuners, thus allowing both the web page and the video program to be simultaneously viewed on the same screen. If video and web stream, however, are carried on one channel, then only one tuner is necessary.

[0045] The client software retrieves URLs from the received video program, directly from the Internet connection 20 or via a separate data channel, interprets these URLs and directs the web enabled browser to retrieve the particular relevant web pages, and synchronizes the retrieved web pages to the video content for display on the television 18. The relevant web pages are preferably shown in one frame of the television 18 while the video program is displayed in another frame. Alternatively, the web page can replace the video program on the display.

[0046] In this embodiment, the digital cable set top box 140 receives the television program from the multichannel cable. The URLs can be encoded into the digital program channel using MPEG1, MPEG2, MPEG4, MPEG7 or any other compression video scheme. Alternatively, the URLs can be transmitted to the digital cable boxes 140 from an Internet server 148. The digital cable box 140 decodes the URLs from the digital video signal or directly transmitted over the Internet 20. The client software decodes the URLs and retrieves the particular web pages as described above. Preferably, the web pages are synchronized with the particular video frames and presented to the user.

[0047] As with all the embodiments described above, instead of receiving the video program from a transmission means, the video program may be addressed directly from a local video source 144 if the video program, with or without embedded URLs, is stored on a storage means such as a video tape or optical disc. In this embodiment, the digital cable box 140 is connected to a VCR, disc drive or other appropriate device.

[0048] Figure 6 illustrates an embodiment where a digital TV 152 is the remote reception unit and performs the functions of the personal computer, shown in Figures 1, 2 and 4, and the digital cable box 140 shown in Figure 5. A processor means and memory are incorporated in the digital TV 152, and the client software and web browser software are implemented in memory in the digital TV 152. All of the functions described above with reference to the other embodiments are performed in a similar manner by the digital TV 152 embodiment.

[0049] Although the digital cable box/TV 140, 18 and digital TV 152, shown in Figures 5 and 6, are incorporated into the embodiment of Figure 1, in substitution for the PC 16, they may also be substituted for the PC 16 shown in Figures 2 and 4.

[0050] A user may view the video and web content on one screen (in two windows), or with the video on one display screen and the web content on a separate display monitor. Alternatively, a user may access the video or web content separately. Thus, a user may branch from video to web content and vice versa.

[0051] The systems described herein are well-suited to the education environment. Thus, students and teachers may access one or more web servers. Software components including instructor and student user software, authoring software and database assessment software are provided. An instructor may, for example, use content creation software on a personal computer to easily integrate into the curriculum current information published on the web through an interface 156 shown in Figure 7. The instructor creates a playlist (i.e. linkfile) 160, the playlist 160 comprising a list of web pages, text notes and questions. The web sites and questions are set out in a predetermined order and can be assigned times. Preferably, the URLs identifying the web site and time stamps are sent automatically to the desktop of each student in the virtual community, either during a playback of a pre-recorded program or during a live event.

[0052] At each of the student workstations, the program is directed by the playlist 160. In other words, the playlist 160 provides the structure for the program. At predetermined times as indicated by the playlist 160, the browser will fetch and display a web page in a frame on the computer screen. Because program events can be set up in this manner at predetermined times, the entire program and playlist can be prerecorded and stored in a web database for later access by students.

[0053] It will be appreciated that the students and the instructor may be located anywhere, as long as they are all connected to the web. Because a server controls the program, the instructor output comes from the server and the student workstations are automatically updated by the web server.

[0054] This educational embodiment integrates web content and other media with collaborative groupware functionality to create an interactive environment for students and teachers. The student may receive a traditional video lesson through a frame in his or her web browser, or from a television. Separate frames may be simultaneously provided as shown in Figure 8, which shows the browser displaying: web pages 176 automatically delivered to each student's desktop with information or exercises that complement the video presentation; a chat dialogue frame 168 for conversing with the instructor and/or other students online; and an interactive playlist 164 of web pages and questions comprising the lesson.

[0055] In the student interface of Figure 8, each student may perform a virtual experiment, for example, during a physics lesson to learn about gravity. In addition, the students may converse with one another and with the instructor using the chat dialogue frame 168. They may also send web pages to one another and provide answers to questions from the teacher via the chat dialogue frame 168 of the student interface 176. With the chat feature, students may break into subgroups for collaborative learning. Whenever a student in the group sends a message, the message is sent to the Internet server 20 and every other student in the subgroup receives and views the message in their chat dialogue frame 168.

[0056] The instructor, however, may retain control over the chat feature. For example, the instructor may terminate the chat feature or web push to terminate unruly on-line conversations or the sending of web pages by students.

[0057] The systems described herein are more powerful than conventional distance learning systems as they allow the instructor to freely and conveniently exercise almost any type of testing strategy. The instructor may test students using a combination of the chat dialogue feature and web pages. For example, multiple choice questions and short answer questions can appear in the chat window 168. Essay questions, requiring longer answers, become web pages. As mentioned above, students can perform virtual experiments online. Once the instructor's personal computer receives student answers, student scoring may be presented to the instructor in any format including tables, charts, diagrams, bar graphs, etc. The instructor, thus, may analyze the results and has the capability of providing realtime feedback to the students.

[0058] Students may also receive individualized feedback via branched interactive audio, video and/or graphics responses. For example, the workstation may branch to a particular audio response, preferably prerecorded in the instructor's own voice, based on the student response to a multiple-choice question. A plurality of potential audio responses may be made available at the student's workstation, for example, by a method as described in US patent No. 5,537,141. Additionally and/or alternatively, personalized video, audio and graphics segments may be delivered and displayed to the student based on a student answer or personal profile, for example, in a manner as described in US patent No. 5,724,091.

[0059] Responses to student answers may be more substantive using a memory feature comprising an algorithm which selects an interactive response to the user based not only on the student's current answer selection, but also on the student's previous responses. The algorithm, preferably stored in memory at each student's workstation and under processor control, selects an output interactive response based on student responses. In an example, a student who gets three or more answers in sequence right receives a more difficult question. However, a student who fails to correctly answer one or more of the three questions receives an easier question.

[0060] The system illustrated in Figure 9 is capable of servicing large numbers of users, for example, several schools. As shown, communications servers 180 distribute and route message across a LAN, WAN and the Internet. At the heart of the system is a group database server 184, and this is surrounded by several communication servers 180 which each serve an area 192.

Each communication server 180 is surrounded by squares representing user stations 188. The communication servers 180 are organized in node relationships with one another.

[0061] Each node is responsible for serving an area 192. An area 192 is defined as a virtual location serviced by a single communication server 180 (or "com server"). An area 192 may be a single school, an office, or may consist of several actual physical locations. The defining characteristic of an area 192 is that messages sent from one member of an area 192 to another need not be routed outside of the servicing com server 180.

[0062] An area member is analogous to the frequently used term "user". For example, a "user" may be a student in an educational environment.

[0063] The distributed communication system shown in Figure 9 permits the dynamic addition of communication servers 180 within a group with little or no administrative tasks as well as the addition of groups within an overall communications network. A communication server group consists of several defined virtual areas 192 (preferably, consisting of no more than 250 members each), each area 192 serviced by a single com server 180. This system allows members of one area 192, or group, to easily communicate with members of another area 192 or group without any configuration changes.

[0064] In the past, service of very large numbers of users has required large expensive servers and networks. Furthermore, as the user base increased, performance suffered and the hardware had to be upgraded to service the demand.

[0065] The distributed communication system allows the same, relatively inexpensive, machines to serve an ever-increasing user base. This is accomplished by routing messages from one server to another when necessary following substantially the same core pattern as IP routing and DNS lookups. If a message is for a member not belonging to the current area 192 or group, the message is routed through the distributed communication system until its destination, or someone who knows the destination and can deliver the message, is found. The destination may be cached so subsequent messages for that member or group may be more efficiently delivered.

[0066] Referring to Figure 9, if a message is posted by member "A" and is intended only for the members of group 1, the message never leaves the area 1 com server. However, if the message is intended for members of area 1 and for members of area 2, the area 1 com server forwards the message to the group database server 184. The message is broadcast to the members of area 1 and tagged in the group database server 184 as belonging to area 2. The message is then routed to area 2 and broadcast to area 2 members. With this technique, any member may potentially send a message to any other member. If the area com server 180 does not recognize the destination, the message is forwarded up the line. Each com server 180 does not need to know about any other server 180. Messages are routed until they delivered. If undeliverable, the original sender is notified.

[0067] New areas 192 can be added on the fly. When a new com server 180 is added to the network, it registers itself with the database application. Henceforth, any message destined for the new area 192 may be routed properly without altering the other area servers 180.

[0068] This method and system works for global messages or for user to user messages. Furthermore, new groups may also be dynamically added. Once added, each new group database server 184 registers itself with the existing database servers 184. This distribution of load permits nearly unlimited expansion with existing software and hardware. Each server manages a finite number of members, cumulatively serving a growing community.

[0069] Users need not be informed as to the particular com server 180 they should connect to. Members are directed to a single URL. The selection of the server for user connection is determined by load balancing software. In this manner, the network may appear to be a global network of servers or simply a local classroom.

[0070] The architecture described, which uses database servers as routing gateways, enables the system to serve with minimum administration and configuration and with lower end, cost-effective hardware.

[0071] Figure 10 illustrates a system 200 arranged to utilise an entity referred to as a local host for facilitating network communication between a user machine 202 and a server 224 through a network 222. The network 222 may be any suitable network, for example, such as the Internet. A local host connection may provide increased functionality in network communication by overcoming certain limitations which exist, for example, within applets in the JAVA programming language. Applets are small applications, typically written in the JAVA programming language. In order to meet required security criteria, the JAVA programming language does not permit certain functions to be undertaken by use of applets. For example, applets typically cannot read to, or write from, a hard disk on a user's machine. The security criteria help to ensure that a user does not unknowingly download an applet as part of a web page, for example, where the applet has the capability to adversely affect the user's system processing or stored data. However, there are situations where it may be advantageous to eliminate or reduce the security restrictions applied to applets in a particular controlled environment. In those situations, the local host may enable the functions of applets to be increased.

[0072] In system 200 illustrated, the user machine 202 has a local host or connection with the server 224 through Internet 222. The user machine 202 includes a web browser 204, which may be implemented by any type of web browser for Internet communication. The web browser 204 includes a HyperText Markup Language (HTML) page, referred to as a donut.html page, within a hidden HTML frame 206. Instead of frames, it may use layers such as those used with dynamic HTML (DHTML) pages. The page 206 may be stored within a hidden frame or within a layer in memory on the user machine 202. Hidden frame or layer 206 includes a donut applet 208 having functions for communicating with the server 224, and the applet 208 may include, for example, applications requiring communication with a server.

[0073] The term "donut" is used only as a label and, in this example, refers to a particular file storing user-profile information for use in transmitting content to a user. The user-profile information may include a wide variety of information relating to a user such as, for example, one or more of the following: age of the user, sex of the user, marital status of the user, prior activities of the user, income range of the user, number of people in the user's household, occupation of the user, industry of the user, length of residence, interests of the user, and other information, such as demographic and activity-based information, about the user.

[0074] Use of a "donut" storing user profile information is described in US application No. 09/409305 filed 29th September 1999.

[0075] The web browser 204 also includes a pi.html page within an HTML frame or layer 210 stored in memory of the user machine 202. The pi.htlm page in frame or layer 210 is loaded from the server 244. The frame or layer 210 also includes a plug-in 212 arranged to implement a gateway module 214 for communication between the applet 208 and the Internet 222. A "plug-in" is an auxiliary program which works with a software package or other program to enhance its capability. The plug-in 212 implements the gateway module 214 in order to provide for a local host that effectively functions as a web server. In particular, upon being loaded, pi.htlm page 210 initiates the plug-in 212 and the gateway module 214, in the C++ programming language, for example, to open a connection 220 with the applet 208.

[0076] Plug-in 212 communicates through the Internet 222 with the server 224. The server 224 may include a web server 226, which may be any web server able to provide for Internet and/or other network communication. The server 224 may also include a distributed community network (DCN) hub 228, for example, as described in US patent application No. 09/396693 filed on 15th September 1999.

[0077] In operation, the plug-in 212 provides a ShowDoc command 218; for example, npi: ShowDoc(http://localhost/donut.html"). The term "NPI" (or "npi") refers to the "Netscape Plugin Interface" which is an application program interface used by plug-ins in web browsers such as the Netscape Navigator program and the Microsoft Internet Explorer program. The "ShowDoc" command is a command in the NPI and the JAVA programming language instructing a web browser to show in a frame of the web browser the content from the URL provided in the command. Therefore, the ShowDoc command is used to push content to the web browser.

[0078] The ShowDoc command 218 initiates an applet tag located in the page 206. Upon receiving the ShowDoc command 218, the web browser 204 responds by sending a GET request 216 in order to establish a connection with the server 224, which in turn initiates the applet 208. The GET request 216 includes a URL for a particular page, for example, "http:GEThttp://localhost/donut.html." Plug-in 212 receives the GET request 216 and uses it to open a Transmission Control Protocol/Internet Protocol (TCP/IP) socket 220 to plug-in 212, which then functions as a local host using the gateway module 214 to permit open and persistent communication with the server 224. A socket is a known way to provide Internet communication, and it specifies an Internet Protocol address of a station and a port number. The applet 208 subsequently communicates with the server 224 through the module 214, which appears to applet 208 as a web server.

[0079] Figure 11 is a flow chart showing a method 250 for implementing a local host within the system 200. In order to implement a local host, the plug-in 212 locally functions as a web server, and the host which the JAVA code in the applet 208 is allowed to access will be the same host as that which originally served the HTML page containing the applet 208. The plug-in 212 requires functionality to respond to only one HTTP request which it generates itself by means of the ShowDoc command 218 for displaying a URL into a hidden frame or layer. That URL refers to the gateway module 214 on a port known only to the plug-in 212.

[0080] In the method 250, the plug-in 212 sends a URL to the web browser 204 using the ShowDoc command 218, such as npi:
ShowDoc((http://localhost/donut/html") (step 252). The web browser 204 loads a web page or other networked content using the URL in order to generate the GET request 216 (step 254), and the plug-in 212 receives the GET request 216 from the applet 208 in web browser 204 (step 256). The GET request includes a URL such as "http: GET http://localhost/donut.html" for indicating an address for the local host.

[0081] Plug-in 212 in the web browser 204 uses the web page information or other networked content in order to open the TCP/IP socket 220 to a local host including the gateway module 214 in plug-in 210 (step 260). In establishing the local host, the gateway module 214 provides for socket 220 between the applet 208 and the server 224 (step 262). In particular, plug-in 212 acts as a proxy, forwarding requests from the applet 208 to the remote server 224. Plug-in 212 may optionally obtain an entity referred to as a "skin", explained below, to render itself in a specific user interface for the machine 202 (step 263). The applet 208 then uses the socket 220 to communicate with the server 224 through plug-in 212 and remote server 244. Plug-in 212 maintains the socket 220 and detects whether the applet 208 disconnects the connection (step 266). If the connection is disconnected, the plug-in 212 removes the socket 220.

[0082] The plug-in 212 preferably has the ability to use multiple skins, which are an integrated set of graphics, colors, and other multimedia content, and their particular arrangement to form a specific, branded user interface. Plug-in 212 and a DCN; identified above, have the ability to distribute and receive an arbitrary object. One such object may be the package of graphics and other content that comprise a skin. Before a program begins, when the user logs on to the network, the system checks to see if the particular skin is resident on the user's machine. If the skin is not resident, the plug-in 212 requests the skin object, and the DCN automatically distributes the skin to the plug-in 212. Once the skin is resident, the plug-in 212 renders itself using those graphics, colors, and other content.

[0083] In addition to requesting and distributing a skin when the user logs on, a programmer may also dynamically change a user's skin during the program. The logic for loading the new skin during the program is the same as the logic for loading a skin at the start of a program. This skin change may also be invoked by a playlist request or a donut event based on user-profile information as described in the application identified above.

[0084] The following provides one example of the use of a local host. During an event that combines automatically delivered web content through a web browser plug-in with a video stream, a question appears in the user's web browser containing an interactive question. This may be written, for example, in Macromedia Flash, a common web authoring tool. The question reads: Have you brought a new car this year? Yes or No. If the user clicks on "Yes", the JavaScript program in the web page containing the question may execute a post with the format of http://localhost?userclick=Yes. The local host is therefore the "pipe" that connects the web browser to the plug-in. The plug-in receives this information through an open socket and acts on it to send the user another web page with content applicable to persons who have recently bought a new car.

[0085] A playlist permits one user, referred to as an author, to push a list of URLs or network addresses to other users. A playlist is a structured collection of URLs that are sent to a user's machine to either be displayed at a specified time ("pushed") or presented in a tree structure to be pulled down later when the user wants to find more detail. The playlist may reference a network address, such as a URL, a description for each URL, a time value, a frameset, a frame or layer, a prefetch time (if appropriate), a pull within a push context, and potentially a variable for a dynamic URL. The playlist permits the author to effectively push various types of content to users, such as advertisements, interactive games, sports scores, narrative content, or any type of multimedia information. The narrative content may be used, for example, to provide textual descriptions accompanying video presentations.

[0086] The author enters a URL and an associated description along with a time value. Based upon the time value, the author's machine transmits the URL, along with the description and web page or other networked content, to the user. The frame (or layer) and frameset indicates where to display the web page or other networked content on the user's machine. The prefetch time may be used to assemble a page corresponding to the URL in a hidden frame or layer. The dynamic URL permits the URL to be customised based upon other information such as a user's geographic location, as explained below. In addition to web pages from the Internet, the playlist may use the URL to retrieve web or other content from a variety of sources such as, for example, from the user's hard disk drive or other storage medium. In addition to web pages, the playlist may be used to retrieve other content such as, for example, an audio stream, a video file, or other multimedia information

[0087] The pull within the push context provides other ways to retrieve content using the playlist, as illustrated by the following example. During a news program, the television broadcast shows a clip of a news conference. A playlist item, a web page or other networked content designed to appear in a particular frame in a particular frameset, is pushed (automatically delivered) to all the users on-line. In addition, the news organization broadcasting the conference has several other web pages which the user may be interested in as background material. This other networked content is sent as a "pull within a push" playlist item. The additional content does not automatically appear in the user's browser. Instead, the user receives an indication that other networked content can be accessed. If the user requires to access the other content available, the user accesses the content in an appropriate manner, for example, by using a mouse to click on a graphic, or by way of other user inputs. Thus, the pull within the push context permits content to be pushed to a user, and through that content the user may pull additional content to the user machine.

[0088] Additionally and/or alternatively to URLs, the playlist may include Uniform Resource Identifiers (URIs). A URI is a compact string of characters for identifying an abstract or physical resource. More specifically, URIs provide a simple and extensible means for identifying a resource, and a URI may be further classified as a locator, a name, or both. The specification of URI syntax and semantics is derived from concepts introduced by the World Wide Web global information initiative. URIs include, for example, URLs and Uniform Resource Names (URNs). A URL is a subset of a URI which identifies resources via a representation of their primary access mechanism, such as their network "location", rather than identifying the resource by name or other attribute of that resource. The term URN refers to a subset of URI which is required to remain globally unique and persistent even when the resource ceases to exist or becomes unavailable.

[0089] Figure 12 is a diagram of a user interface 300 displaying a playlist to an author of the playlist content. In this example, the playlist content is shown in a section 302 displayed within a screen 300. The content includes, for each URL, a textual description 308, a type of content 310, a URL 312, and a time value 314. The textual description 308 permits the author to provide a potentially more descriptive identifier for the content associated with a particular URL. The type of content 310 identifies for each URL the type of associated content or how it is to be displayed, and examples include push content (PSH), a ticker display (TIC), and an advertisement (ADV). The section 302 includes a playlist menu item 304 for a user to select in order to view playlist section 302 listing the playlist. It also includes an edit item 306 for a user to select in order to create or edit the playlist content.

[0090] Upon selecting item 306, the author's machine displays user interface 320 shown in Figure 13. The user interface 320 includes a section 322 for the author to enter a textual description for a particular URL. A section 324 permits the author to enter the URL. A section 326 provides a pull-down menu permitting the author to identify a particular frame or layer on the user's machine in which the content associated with the URL in section 324 will be displayed. A section 328 provides a pull-down menu permitting the author to select a frame set identifying how many frames or layers will be displayed on the user's machine. As shown, one option is a triple frame set in which three frames are displayed. Other examples include a single, double, or quadruple frame set for displaying, respectively, one, two or four frames.

[0091] A time section 330 permits the author to identify a time value for the author's machine to transmit to a user's machine the URL in section 324, the description in section 322, and optionally an associated web page or other networked content. In this example, the author may enter values for hours, minutes, and seconds. The author's machine uses the entered time value as a timer, and it pushes the associated URL to the user's machine upon time-out of the entered time value. Other time parameters are possible for determining when to push the URL to a user's machine. For example, the URL may be pushed at a specified particular time and day and/or at a predetermined amount of time after the occurrence of a specified event.

[0092] In edit section 320, the author may select an add section 332 to add the entered information to the playlist and select a cancel section 334 to cancel the entered information. The author typically selects the sections by use of a cursor-control device to "click on" the section and/or by entering a command using a keyboard. It will be appreciated that any suitable user input interface may be provided.

[0093] Figure 14 is a diagram of a user interface 340 displaying a playlist of items pushed to a user. User interface 340 includes a section 342 for displaying playlist items pushed to the user. Usually, only the descriptions are displayed to the user, whilst associated URLs, web pages and/or other networked content, may be stored on the user's machine and linked with the descriptions. User interface 340 also includes frames or layers in which to display content associated with the URLs pushed to the user, and these may include, for example, one or more frames 356 for displaying video content, one or more frames 350 for displaying advertisements, one or more frames 352 for displaying games, and one or more frames 354 for displaying a ticker display, for example, of sports scores.

[0094] Upon selecting a description among the playlist items in section 342, the user may retrieve and view associated content. The description may thus operate as a hypertext link. For example, by selecting description 344, the ticker display in section 354 may be displayed. By selecting description 346, the game in frame 352 may be displayed. By selecting description 348, the advertisement in frame 350 is displayed. The user may select other descriptions from amongst the playlist items in section 342 in order to view the associated content or web pages or other networked content. The frame or layer in which the content is displayed depends upon the frame or layer information identified by the author in section 326 of edit section 320.

[0095] Figure 15 illustrates an example of a data structure 360 for playlist entities. The playlist entity provides a name and identifier to the collection of URLs. Data structure 360 includes multiple entities or other data structure elements for storing the playlist entities and associated data. In this example, these entities include a Playlist entity 362, a PlaylistItem entity 364, a frame or layer entity 366, a ResFolder entity 368, and a ResItem entity 370.

[0096] The actual URL data is stored in a resData field 371 of ResItem entity 370. ResFolder entity 368 is used by the author of the playlist to organize data in ResItem entity 370 into a folder hierarchy for organizational convenience and typically has no effect on the end user experience. A defFrameSetID field 372, a defFrame field 373, and a description field 374 are similarly provided as conveniences for the author.

[0097] PlaylistItem entity 364, also referred to as "item", represents a specified resItem in entity 370. It is identified by a resItemID field 376 in a playlist identified by playlistID field 376. The item is to be shown after a time specified by a timeOffset field 377. In particular, the item is pushed to the user's machine at that time unless a pull only feature is set as identified in a pullOnly field 378, in which case the item is a pull within a push context. The item is to be shown in the frame or layer specified by a frame field 379 and a frameSetID field 380. A description field 381 specifies the description seen by the user in the displayed playlist. Default values for frame field 379, frameSetID field 380, and description field 381 are derived from the corresponding item in ResItem entity 370. A prefetch field 382 indicates the pre-loading of the URL in a staging area. A staging area for prefetch involves the assembly of a page in a hidden

IPR2022-01227
EXHIBIT 1003 - PAGE 00808

frame for subsequent display of a completely assembled web page. One example of this is described in US patent application No. 09/397298 filed on 15th September 1999.

[0098] A parentID field 383 of PlaylistItem entity 364 specifies a parent item in the hierarchy of items. In a flat playlist, parent ID is null. When in a hierarchy, timeOffset field 377 is relative to a time offset for a parent item. The relationship is recursive, making it easy to push a playlist at any time and to reuse URL resources and their relationships in the authoring tool.

[0099] Frame entity 366 specifies a specific frame or layer within a frameset into which an item is to be pushed or pulled. It is referred to by both ResItem entity 370 and PlaylistItem entity 364. These entities may exist in multiple databases or even by multiple vendors and are manipulated by authoring tools, for example, such as those available in the C++, JAVA, and Tango programming languages. These entities may be exchanged and manually edited in Extensible Markup Language (XML) and transmitted over a distributed community network. An example of a distributed community network is described in US patent application No. 09/396693 filed 15th September 1999.

[0100] Figure 16 is a flow chart of a method 400 for implementing a playlist. Method 400 may be implemented using software modules for execution by a corresponding machine. In method 400, the system receives playlist information from the author in edit section 320 (step 402). This information may include a URL, a description, a frame or layer identification, and a time value. The system uses the playlist information to generate and display a playlist as shown in section 302 (step 404). The system continually monitors the playlist to determine if it contains an item (step 406). If it contains an item, the system determines if prefetch was invoked (step 408) and, if so, it pushes the playlist item with a web page or other networked content to the user's machine along with a prefetch time (step 410). The user's machine may use a web page staging area to display the playlist item (step 412). That is, the web page may be assembled in a hidden frame or layer and displayed when assembled. If prefetch was not invoked, the system determines if the offset time as entered by the author has expired (step 414). Once the offset time expires, the system pushes the playlist item with a web page or other networked content to the user's machine and displays the item as shown in section 342 (step 416). Particular content is also pushed to the user's machine based upon the user's selection of playlist items (step 413).

[0101] The system determines if the author has closed the playlist (step 418). If the playlist remains open, the system determines if the author enters more playlist information in edit section 320. If more playlist information is to be entered, the system returns to step 402 to receive the playlist information and then generate and display a new playlist at step 404. If the user does not enter more playlist information, the system returns to step 406 to process any additional items in the playlist.

[0102] A dynamic URLs feature permits a system to dynamically configure a URL, URI, or other content identified based upon resolution of a variable. Thus, the system may customize URL content for particular uses. The dynamic URL is important as it enables content to be localised and enables the transmission of various types of content, for example, related to the playlist. For example, a national television broadcast might include a commercial for a particular vehicle. The advertiser may wish to supplement that advertisement with information local to each viewer. A dynamic URLs feature enables viewers to be sent web pages identifying local dealers.

[0103] In one specific example, a user Bob lives in Los Angeles and there is a specific automobile dealer based a few miles away from Bob on Ventura Blvd. The playlist author may have a playlist entry for an automobile advertisement which includes a dynamic URL. The server software may determine which one of many URLs that references all particular dealers in America should be sent to Bob, or Bob's user profile (donut) in his client may parse the dynamic URL, determine the specific web address of the Ventura automobile dealer web page, and push that network address into Bob's browser.

[0104] The dynamic URLs may be stored and distributed in a playlist as described above and executed as part of steps 410 and 416. A dynamic URL may be identified by the presence of a variable with particular characters. The dynamic URL may contain the variable brackets "{" and "}" as follows, for example, PHS://http//content.qa.actv.com/{variable}.

[0105] Figure 17 is a flow chart of a method 430 for processing dynamic URLs, URIs, or other network addresses. Method 430 may be implemented using software modules for execution by a corresponding machine. In method 430, the system retrieves a playlist item including a URL (step 431) and determines if the retrieved URL is a dynamic URL (step 432). It may determine if the URL is dynamic by parsing the URL to detect the presence of a variable in brackets or other distinguishing characters. If it is not a dynamic URL, the system returns the playlist item with no processing to the URL (step 446).

[0106] If it is a dynamic URL, the system resolves the variable in the URL (step 434). The system determines if the variable references or invokes a particular process (step 436). If a process is not involved, the system obtains a definition for the variable (step 444). If the variable does invoke a process, the system retrieves and executes the process in order to generate a definition for the variable (step 438). The system may resolve the variable based upon an identification of an intended recipient of the URL in order customize content for the user. For example, it may obtain user-profile information, and select a variable definition based upon the user-profile information, and/or it may use a geographic location of the user to resolve the variable. The user's location may be determined, for example, from the user-profile information. Using the variable definition, the system assembles a completed URL step (440) and returns the playlist item with the assembled URL in steps 410 and 416 of playlist method 400 (step 442).

[0107] As indicated above, user profile information may be collected from the activities of a user by way of a user machine and/or may comprise demographic information about the user. The user profile information may be stored local to the user, for example, by the user machine, and may be used to resolve the variable of the dynamic URL.

Claims

1. A method for providing network communication between a machine and a server, the method comprising the steps of:

    receiving a request for a network connection;
    generating a reference to a local connection in response to the request;
    detecting communication at the local connection; and
    establishing a connection with the local connection in response to the detecting and for facilitating network communication with the machine.

2. A method as claimed in Claim 1, wherein the establishing step comprises providing a local web server function at the local connection.

3. A method as claimed in Claim 1 or Claim 2, wherein the establishing step comprises providing a socket for the communication.

4. A method as claimed in any preceding claim, further comprising downloading networked content from a network for use in establishing the connection.

5. A method as claimed in any preceding claim, wherein the receiving step comprises receiving networked content.

6. A method as claimed in any preceding claim, wherein the receiving step comprises receiving a request for a web page from a web browser.

7. A method as claimed in any preceding claim, wherein the receiving step comprises receiving the request from an applet.

8. A method as claimed in any preceding claim, wherein the generating step comprises providing an address of the local connection.

9. A method as claimed in any preceding claim, wherein the detecting comprises receiving information for use in opening a socket with the local connection.

10. A method as claimed in any preceding claim, further comprising detecting a disconnection from the local connection.

11. A method as claimed in any preceding claim, further comprising obtaining particular content for use in rendering a specific user interface for the network communication.

12. Apparatus for providing network communication between a machine and a server, comprising:

    receiving means for receiving a request for a network connection;
    generating means for generating a reference to a local connection in response to the request;
    detecting means for detecting communication at the local connection; and connection means for establishing a connection with the local connection in response to the detecting and for facilitating network communication with the machine.

13. Apparatus as claimed in Claim 12, wherein the connection means is arranged to provide a local web server function at the local connection.

14. Apparatus as claimed in Claim 12 or Claim 13, wherein said connection means provides a socket for the communication.

15. Apparatus as claimed in any of Claims 12 to 14, further comprising means for downloading networked content from a network for use in establishing the connection.

16. Apparatus as claimed in any of Claims 12 to 15, wherein said receiving means is arranged to receive networked content.

17. Apparatus as claimed in any of Claims 12 to 16, wherein said receiving means is arranged to receive a request for a web page from a web browser.

18. Apparatus as claimed in any of Claims 12 to 17, wherein said receiving means is arranged to receive the request from an applet.

19. Apparatus as claimed in any of Claims 12 to 18, wherein said generating means comprises means for providing an address of the local connection.

20. Apparatus as claimed in any of Claims 12 to 19, wherein said detecting means comprises means for receiving information for use in opening a socket with the local connection.

21. Apparatus as claimed in any of Claims 12 to 20, further comprising means for detecting a disconnection from the local connection.

22. Apparatus as claimed in any of Claims 12 to 21, further comprising means for obtaining particular content for use in rendering a specific user interface for the network communication.

23. A method for distributing network addresses to user machines for use in obtaining content associated with the addresses, the method comprising the steps of:

receiving a plurality of network addresses identifying network locations of particular content;
receiving a time value of a time parameter associated with each of the network addresses; and
transmitting each of the network addresses to a user based upon each of the corresponding time values.

24. A method as claimed in Claim 23, further comprising receiving an associated description for each network address, and transmitting the associated description along with each network address

25. A method as claimed in Claim 24, wherein the step of receiving the associated description comprises receiving a textual description for each of the network addresses.

26. A method as claimed in any of Claims 23 to 25, wherein the step of receiving the network addresses comprises receiving a plurality of uniform resource locators.

27. A method as claimed in any of Claims 23 to 26, wherein the step of receiving the network addresses comprises receiving a plurality of uniform resource identifiers.

28. A method as claimed in any of Claims 23 to 27, wherein the step of receiving the time value comprises receiving a time-out value.

29. A method as claimed in any of Claims 23 to 28, further comprising the steps of receiving for each of the network addresses an indication of a frame or o layer in which to display corresponding descriptions on user machines.

30. A method as claimed in any of Claims 23 to 29, further comprising the step of receiving for each of the network addresses an indication of a type of a content identified by the network address.

31. A method as claimed in any of Claims 23 to 30, wherein the transmitting step comprises transmitting to user machines networked content corresponding to each of the network addresses.

32. A method as claimed in any of Claims 23 to 31, wherein descriptions are associated with each network address, and further comprising permitting a user to specify the network addresses, associated descriptions, and the time values.

33. A method as claimed in Claim 32, wherein the permitting step comprises 5 displaying a user interface for receiving from the user the network addresses, the associated descriptions, and the time values.

34. A method as claimed in any of Claims 23 to 33, further comprising receiving an associated description for each network address, and wherein:

the step of receiving the associated description comprises receiving an indication of the particular content; and
the transmitting step comprises transmitting the indication of the particular content to user machines for permitting users to select the indication in order to retrieve the particular content.

35. A method as claimed in any of Claims 23 to 34, further comprising receiving an associated description for each network address, and wherein:

the step of receiving the associated description comprises receiving a hypertext link; and
the transmitting step comprises transmitting the hypertext link to user machines.

36. A method as claimed in any of Claims 23 to 35, wherein the step of receiving the network addresses comprises receiving an identification of advertising content.

37. A method as claimed in any of Claims 23 to 36, wherein the step of receiving network addresses comprises receiving an identification of sports content, narrative content, and/or interactive game content.

38. Apparatus for distributing network addresses to user machines for use in obtaining content associated with the addresses, comprising:

receiving means for receiving a plurality of net-

IPR2022-01227
EXHIBIT 1003 - PAGE 00811

work addresses identifying network locations of particular content;
time receiving means for receiving a time value of a time parameter associated with each of the network addresses; and
transmission means for transmitting each of the network addresses to user machines based upon each of the corresponding time values.

39. Apparatus as claimed in Claim 38, wherein said receiving means is arranged to receive an associated description for each network address, and said transmission means is arranged to transmit the associated descriptions along with the network addresses.

40. Apparatus as claimed in Claim 39, wherein said receiving means is arranged to receive an indication of a frame or layer in which to display the corresponding descriptions on the user machines.

41. Apparatus as claimed in any of Claims 38 to 40, wherein said receiving means is arranged to receive a plurality of uniform resource locators.

42. Apparatus as claimed in any of Claims 38 to 41, wherein said receiving means is arranged to receive a plurality of uniform resource identifiers.

43. Apparatus as claimed in any of Claims 38 to 42, wherein said receiving means is arranged to receive a textual description for each of the network addresses.

44. Apparatus as claimed in any of Claims 38 to 43, wherein said time receiving means is arranged to receive a time-out value.

45. Apparatus as claimed in any of Claims 38 to 44, wherein said receiving means is arranged to receive an indication of a type of a content identified by the network address.

46. Apparatus as claimed in any of Claims 38 to 45, wherein said transmission means is arranged to transmit to the user machines networked content corresponding to each of the network addresses.

47. Apparatus as claimed in any of Claims 38 to 46, wherein descriptions are associated with each network address, and further comprising means for permitting a user to specify the network addresses, the associated descriptions, and the time values.

48. Apparatus as claimed in Claim 47, wherein said permitting means is arranged to display a user interface for receiving from the user the network addresses, the associated descriptions, and the time values.

49. Apparatus as claimed in any of Claims 38 to 48, wherein:

said receiving means is arranged to receive an indication of the particular content; and
said transmission means is arranged to transmit the indication of the particular content to user machines to enable users to retrieve the particular content by selecting the indication.

50. Apparatus as claimed in any of Claims 38 to 49, wherein:

said receiving means is arranged to receive a hypertext link; and
said transmission means is arranged to transmit the hypertext link to user machines.

51. Apparatus as claimed in any of Claims 38 to 50, wherein said receiving means is arranged to receive an identification of advertising content.

52. Apparatus as claimed in any of Claims 38 to 51, wherein said receiving means is arranged to receive an identification of sports content, narrative content, and/or interactive game content.

53. A method for distributing dynamic network addresses to user machines for use in obtaining content associated with the addresses, comprising the steps of:

receiving a network address containing a variable, the network address identifying varying network locations of particular content based upon the variable;
receiving an associated description for the network address;
resolving the variable in the network address based upon information related to a user machine intended to receive the network address; and
transmitting the network address along with the associated description to said user machine.

54. A method as claimed in Claim 53, further comprising the steps of receiving a time value of a time parameter associated with the network address, and transmitting the network address to the user machine based upon said time value.

55. A method as claimed in Claim 54, wherein the step of receiving the time value comprises receiving a time-out value.

56. A method as claimed in any of Claims 53 to 55,

IPR2022-01227
EXHIBIT 1003 - PAGE 00812

wherein the resolving step comprises executing a process associated with the variable.

57. A method as claimed in any of Claims 53 to 56, wherein the resolving step comprises obtaining information based upon the variable.

58. A method as claimed in any of Claims 53 to 57, wherein the resolving step comprises assembling the network address using the information.

59. A method as claimed in any of Claims 53 to 58, wherein the step of receiving the network address comprises receiving a uniform resource locator.

60. A method as claimed in any of Claims 53 to 59, wherein the step of receiving the network address comprises receiving a uniform resource identifier.

61. A method as claimed in any of Claims 53 to 60, wherein the step of receiving the description comprises receiving a textual description for the network address.

62. A method as claimed in any of Claims 53 to 61, further comprising receiving for the network address an indication of a frame or layer in which to display the corresponding description on the user machine.

63. A method as claimed in any of Claims 53 to 62, further comprising receiving for the network address an indication of a type of a content identified by the network address.

64. A method as claimed in any of Claims 53 to 63, wherein the transmitting step comprises transmitting to the user machine networked content corresponding to the network address.

65. A method as claimed in any of Claims 53 to 64, wherein the step of receiving the network address comprises receiving an identification of advertising content.

66. A method as claimed in any of Claims 53 to 65, wherein the step of receiving the network address comprises receiving an identification of sports content, narrative content, and/or interactive game content.

67. A method as claimed in any of Claims 53 to 66, wherein the resolving step comprises resolving the variable based upon user-profile information related to a user.

68. A method as claimed in Claim 67, wherein the resolving step comprises resolving the variable based upon demographic information related to the user.

69. A method as claimed in Claim 67 or Claim 68, wherein the resolving step comprises resolving the variable based upon activity-based information related to the user.

70. A method as claimed in any of Claims 67 to 69, wherein the resolving step comprises resolving the variable based upon at least one of the following: the age of a user, the sex of a user, the marital status of a user, prior activities of a user, the income range of a user, the number of people in a user's household, the occupation of a user, the industry of a user, the length of residence of a user, and/or the interests of a user.

71. A method as claimed in any of Claims 53 to 70, wherein the resolving step comprises resolving the variable based upon identification of a geographic location of a user.

72. Apparatus for distributing dynamic network addresses to user machines for use in obtaining content associated with the address, comprising:

receiving means for receiving a network address containing a variable, the network address identifying varying network locations of particular content based upon the variable, and for receiving an associated description for the network address;
resolution means module for resolving the variable in the network address based upon information related to a user machine intended to receive the network address; and
transmission means for transmitting the network address along with the associated description to said user machine.

73. Apparatus as claimed in Claim 72, further comprising time receiving means for receiving a time value of a time parameter associated with the network address, and wherein said transmission means is arranged to transmit the network address to said user machine based upon the corresponding time value.

74. Apparatus as claimed in Claim 73, wherein said time receiving means is arranged to receive a timeout value.

75. Apparatus as claimed in any of Claims 72 to 74, wherein the resolution means is arranged to execute a process associated with the variable.

76. Apparatus as claimed in any of Claims 72 to 75, wherein the resolution means is arranged to obtain information based upon the variable.

77. Apparatus as claimed in any of Claims 72 to 76,

IPR2022-01227
EXHIBIT 1003 - PAGE 00813

wherein the resolution means is arranged to assemble the network address using the information.

78. Apparatus as claimed in any of Claims 72 to 77, wherein said receiving means is arranged to receive a uniform resource locator.

79. Apparatus as claimed in any of Claims 72 to 78, wherein said receiving means is arranged to receive a uniform resource locator.

80. Apparatus as claimed in any of Claims 72 to 79, wherein said receiving means is arranged to receive a textual description for the network address.

81. Apparatus as claimed in any of Claims 72 to 80, wherein said receiving means is arranged to receive for a network address an indication of a frame or layer in which to display the corresponding description on the user machine.

82. Apparatus as claimed in any of Claims 72 to 81, wherein said receiving means is arranged to receive an indication of a type of a content identified by the network address.

83. Apparatus as claimed in any of Claims 72 to 82, wherein the transmission means is arranged to transmit to the user machine networked content corresponding to the network address.

84. Apparatus as claimed in any of Claims 72 to 83, wherein said receiving means is arranged to receive an indication of advertising content.

85. Apparatus as claimed in any of Claims 72 to 84, wherein said receiving means is arranged to receive an identification of sports content, narrative content, and/or interactive game content.

86. Apparatus as claimed in any of Claims 72 to 85, wherein the resolution means is arranged to resolve the variable based upon user-profile information related to a user.

87. Apparatus as claimed in Claim 86, wherein the resolution means is arranged to resolve the variable based upon demographic information related to the user.

88. Apparatus as claimed in Claim 86 or Claim 87, wherein the resolution means is arranged to resolve the variable based upon activity-based information related to the user.

89. Apparatus as claimed in any of Claims 86 to 88, wherein the resolution means is arranged to resolve the variable based upon at least one of the following: the age of a user, the sex of a user, the marital status of a user, the prior activities of a user, the income range of a user, the number of people in a user's household, the occupation of a user, the industry of a user, the length of residence of a user, and/or the interests of a user.

90. Apparatus as claimed in any of Claims 72 to 89, wherein the resolution means is arranged to resolve the variable based upon identification of a geographic location of the user.

91. A method of receiving information over a network, the information related to a program, comprising:

activating a computer program associated with a user machine, the computer program operating as local server to provide communication between the user machine and the network; and

receiving the information at the user machine from the network via the computer program.

92. A method of providing information related to a program over a network, comprising:

selecting information related to the program; and

transmitting the information from the network to a user machine, the user machine associated with a computer program, the computer operating as local server to provide communication between the user machine and the network.

93. A method of receiving information over a network, the information related to a television program, comprising:

activating a computer program associated with a user machine, the computer program operating as local server to provide communication between the user machine and the network; and

receiving the information at the user machine from the network via the computer program.

94. A method of providing information related to a television program over a network, comprising:

selecting information related to the television program; and

transmitting the information from the network to a user machine, the user machine associated with a computer program, the computer program operating as local server to provide communication between the user machine and the network.

95. A method as claimed in any of Claims 91 to 94, further comprising the step of reading data from a memory device coupled to the user machine.

96. A method as claimed in any of Claims 91 to 95, further comprising the step of writing data from a memory device coupled to the user machine.

97. A method as claimed in any of Claims 91 to 96, wherein the network is the Internet.

98. A method as claimed in Claim 97, wherein the information related to the program further comprises a uniform resource locator.

99. A method as claimed in any of Claims 91 to 98, wherein the program is provided from broadcast television, storage media such as VHS tape, Beta tape, CD-ROM, or DVD, Internet and/or a video server.

100. A method as claimed in any of Claims 91 to 99, further comprising the step of transmitting a response from the user machine via the computer program to the network.

101. A method as claimed in any of Claims 91 to 100, wherein the computer program is a plug-in in a web browser.

102. A method as claimed in Claim 101, wherein the plug-in instantiates gateway means for communication with the network.

103. A method as claimed in Claim 102, wherein, for communication purposes, the plug-in appears to the user machine and the network to be a web server.

104. A method of receiving information related to a television program over the Internet, comprising:

activating a computer program associated with a user machine, the computer program operating as local server to provide communication between the user machine and the network; and
receiving the information at the user machine from the network via the computer program, the information including a uniform resource locator from a playlist containing at least one uniform resource locator related to the television program.

105. A method of providing information related to a television program over a network, comprising:

selecting information including a uniform resource locator from a playlist containing at least one uniform resource locator related to the telveion program; and
transmitting the information from the network to a user machine, the user machine associated with a computer program, the computer program operating as local server to provide communication between the user machine and the network.

106. A method as claimed in Claim 104 or Claim 105, further comprising the step of reading data from a memory device coupled to the user machine.

107. A method as claimed in any of Claims 104 to 106, further comprising the step of writing data from a memory device coupled to the user machine.

108. A method as claimed in any of Claims 104 to 107, wherein the uniform resource locator is directed to a web site providing additional information related to the content of the television program.

109. A method as claimed in any of Claims 104 to 107, wherein the uniform resource locator is directed to a web site providing additional information related to advertising associated with the television program.

110. A method as claimed in any of Claims 104 to 109, wherein the television program is provided from broadcast television, storage media such as VHS tape, Beta tape, CD-ROM, or DVD, Internet and/or a video server.

111. A method as claimed in any of Claims 104 to 110, further comprising the step of transmitting a response from the user machine via the computer program to the network.

112. A method as claimed in any of Claims 104 to 111, wherein the computer program is a plug-in in a web browser.

113. A method as claimed in Claim 112, wherein the plug-in instantiates gateway means for communication with the network.

114. A method as claimed in Claim 112 or Claim 113, wherein, for communication purposes, the plug-in appears to the user machine and the network to be a web server.

115. A communications apparatus for providing network communications, comprising:
instantiating gateway means within the communications apparatus; and opening a connection to a network with the gateway means; and emulat-

ing a web server for communications with the network.

116. A communications apparatus as claimed in Claim 115, wherein the gateway means utilizes information transmitted over a network and received by a browser, the browser being supported by the communications apparatus.

117. A communications apparatus as claimed in Claim 115 or Claim 116, wherein the gateway means is instantiated in response to a command from an applet.

118. A method for connecting to a network server, comprising:

loading an htlm page on a user machine with a web browser in communication with the network server;
instantiating a plug-in module with the html page;
instantiating a gateway module with the plug-in module; and
opening a TCP/IP socket connection to the network server in response to a request from an applet; and
emulating a web server for communications a web server in the user machine.

119. A computer readable medium containing instructions for controlling a computer system in a user machine to implement a local host for communicating between the user machine and a network server, by:

instantiating a gateway module within the user machine; and
opening a connection to the network server with the gateway module; and emulating a web server for communications with the network server.

120. A computer readable medium containing instructions for controlling a computer system in a user machine to implement a local host for communicating between the user machine and a network server, by:

loading an htlm page on a user machine with a web browser in communication with the network server;
instantiating a plug-in module with the html page;
instantiating a gateway module with the plug-in module; and
opening a TCP/IP socket connection to the network server in response to a request from an applet.

121. An apparatus arranged to receive information over a network, the information related to a program, comprising:

a user machine; and
a computer program associated with the user machine, the computer program operating as local server to receive the information from the network and provide the information to the user machine.

122. An apparatus arranged to provide information related to a program over a network comprising:
an information selector, the information selector containing and transmitting information related to the program to a computer program associated with a user machine, the computer program operating as local server to receive the information from the network and provide the information to the user machine.

123. An apparatus capable of receiving information over a network, the information related to a television program, comprising:

a user machine; and
a computer program associated with the user machine, the computer program operating as local server to receive the information from the network and provide the information to the user machine.

124. An apparatus capable of providing information related to a television program over a network, comprising:
an information selector, the information selector containing and transmitting information related to the program to a computer program associated with a user machine, the computer program operating as a local server to receive the information from the network and provide the information to the user machine.

125. An apparatus for receiving information over a network, the information related to a program comprising:

means for activating a computer program associated with a user machine, the computer program operating as local server to provide communication between the user machine and the network; and
means for receiving the information at the user machine from the network via the computer program.

126. An apparatus for providing information related to a program over a network, comprising:

means for selecting information related to the program; and
means for transmitting the information from the network to a user machine, the user machine associated with a computer program, the computer program operating as local server to provide communication between the user machine and the network.

127. An apparatus for receiving information over a network, the information related to a television program, comprising:

means for activating a computer program associated with a user machine, the computer program operating as local server to provide communication between the user machine and the network; and
means for receiving the information at the user machine from the network via the computer program.

128. An apparatus for providing information related to a television program over a network, comprising:

means for selecting information related to the television program; and means for transmitting the information from the network to a user machine, the user machine associated with a computer program, the computer program operating as local server to provide communication between the user machine and the network.

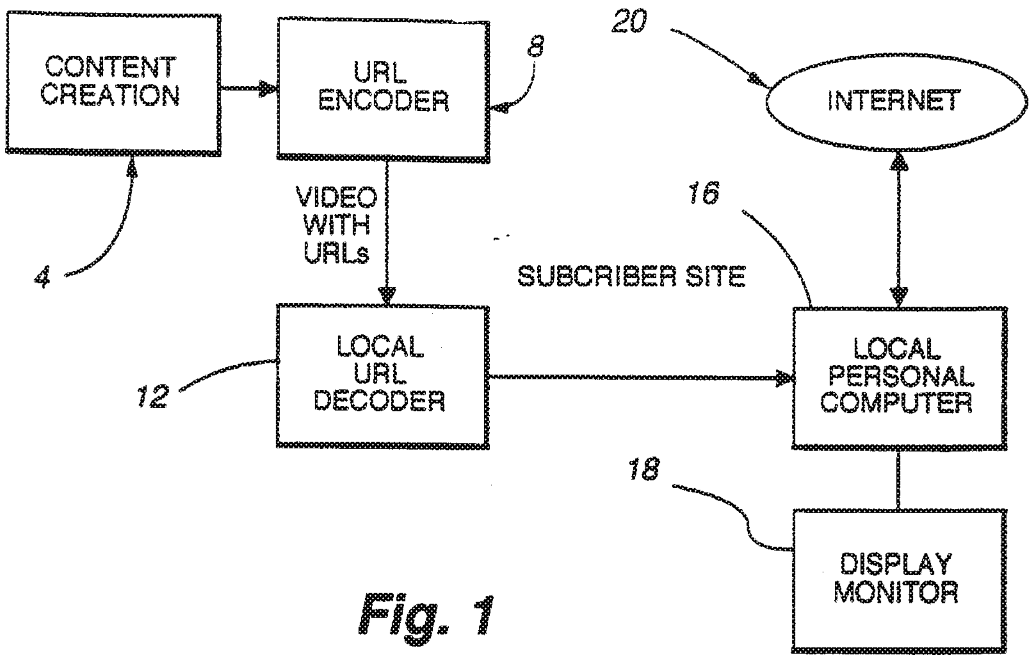

CONTENT CREATION
URL ENCODER
8
20
INTERNET
4
VIDEO WITH URLs
16
SUBCRIBER SITE
LOCAL URL DECODER
12
LOCAL PERSONAL COMPUTER
18
DISPLAY MONITOR

Fig. 1

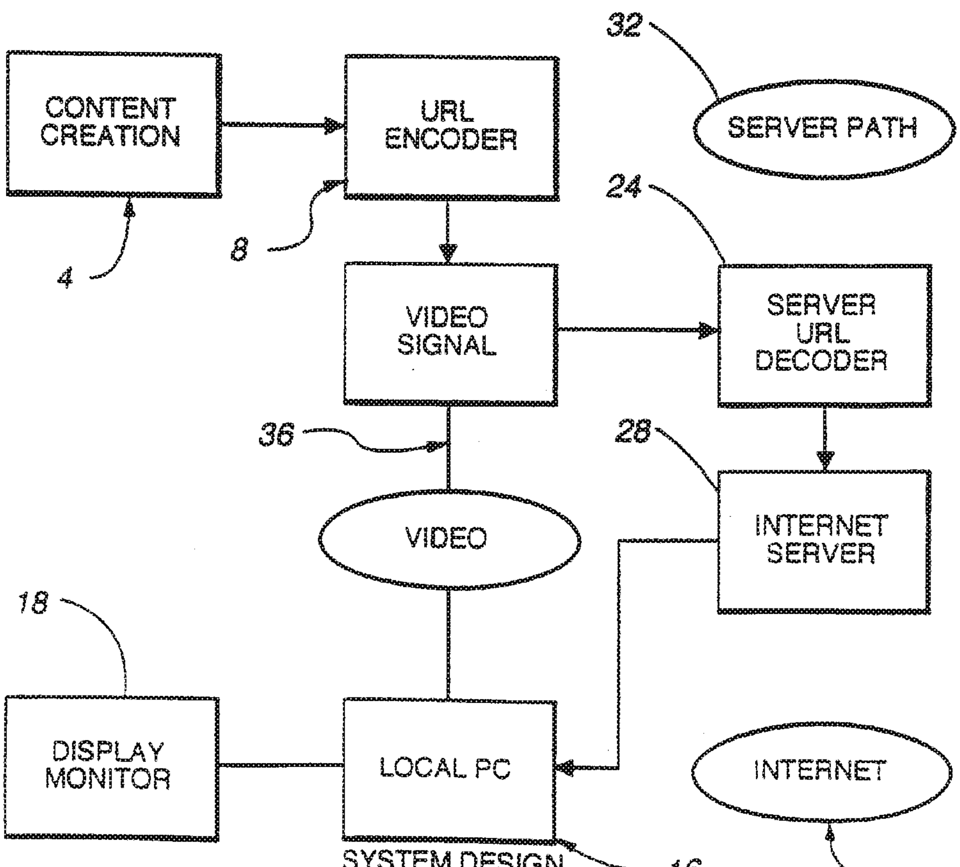
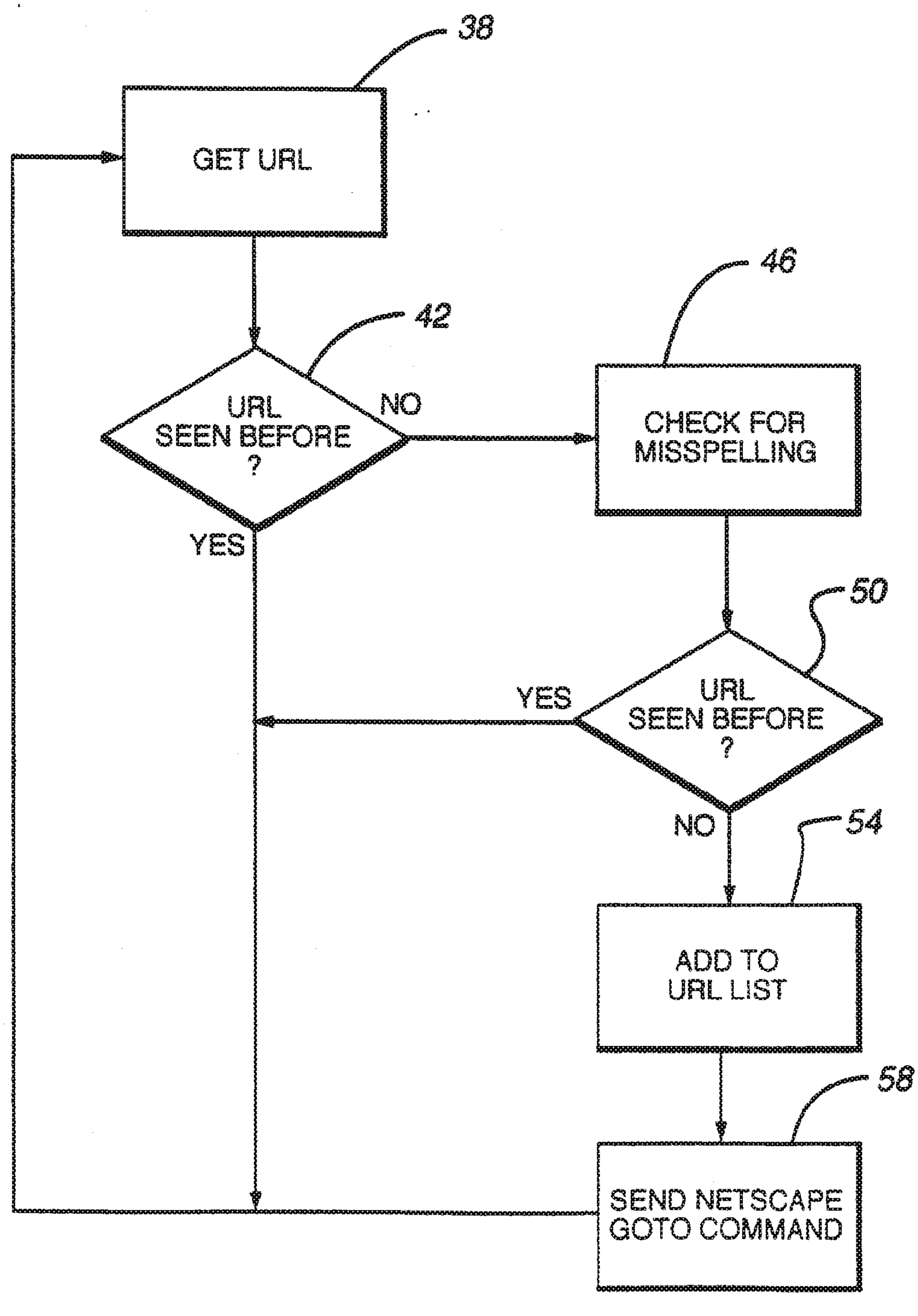

32
CONTENT CREATION
URL ENCODER
SERVER PATH
4
8
24
VIDEO SIGNAL
SERVER URL DECODER
36
28
VIDEO
INTERNET SERVER
18
DISPLAY MONITOR
LOCAL PC
INTERNET
SYSTEM DESIGN
16
20

Fig. 2



SOFTWARE DESIGN

Fig. 3

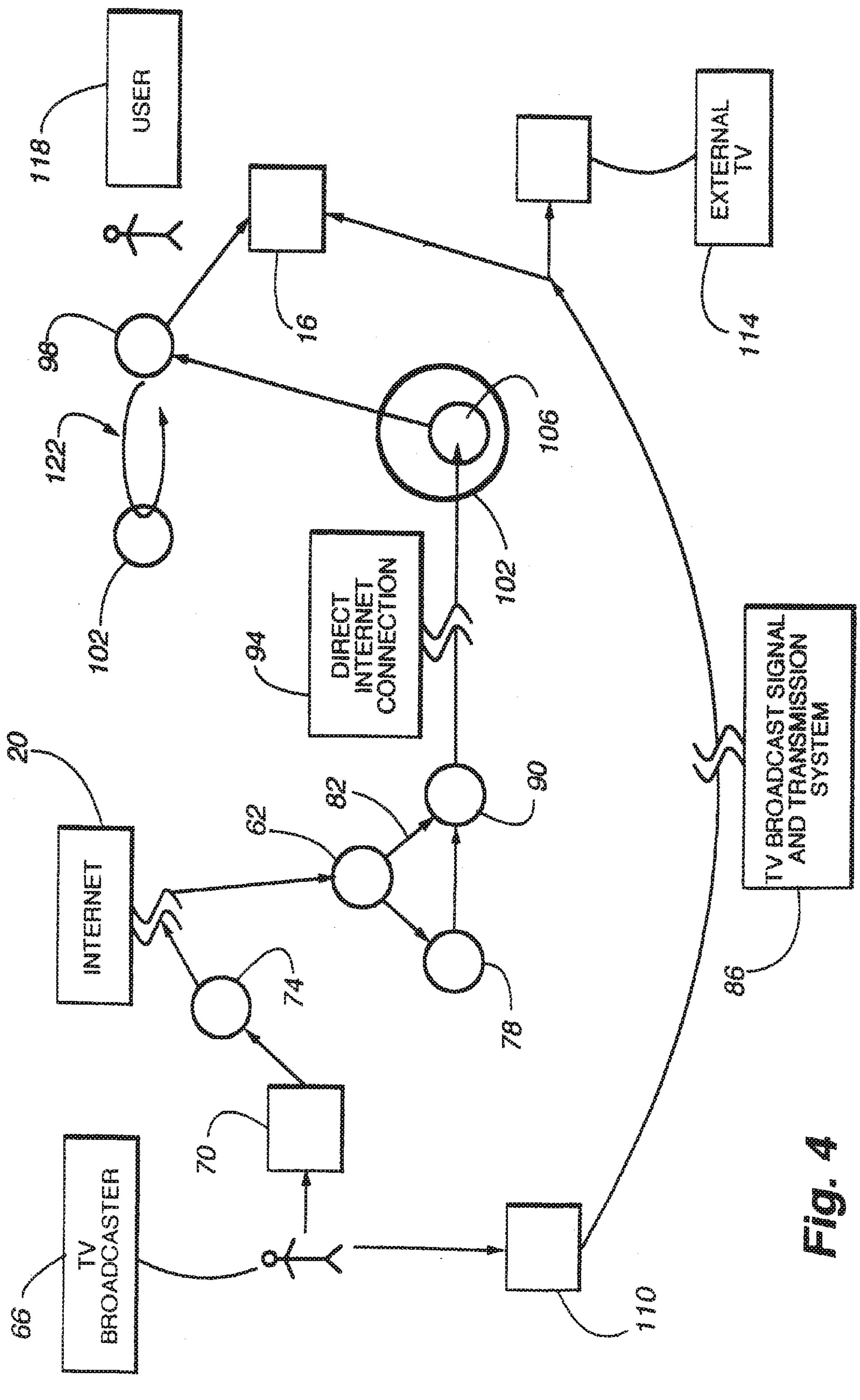

TV BROADCASTER
66
70
110
INTERNET
20
74
62
82
78
90
DIRECT INTERNET CONNECTION
94
TV BROADCAST SIGNAL AND TRANSMISSION SYSTEM
86
102
102
106
122
98
16
118
USER
EXTERNAL TV
114

Fig. 4

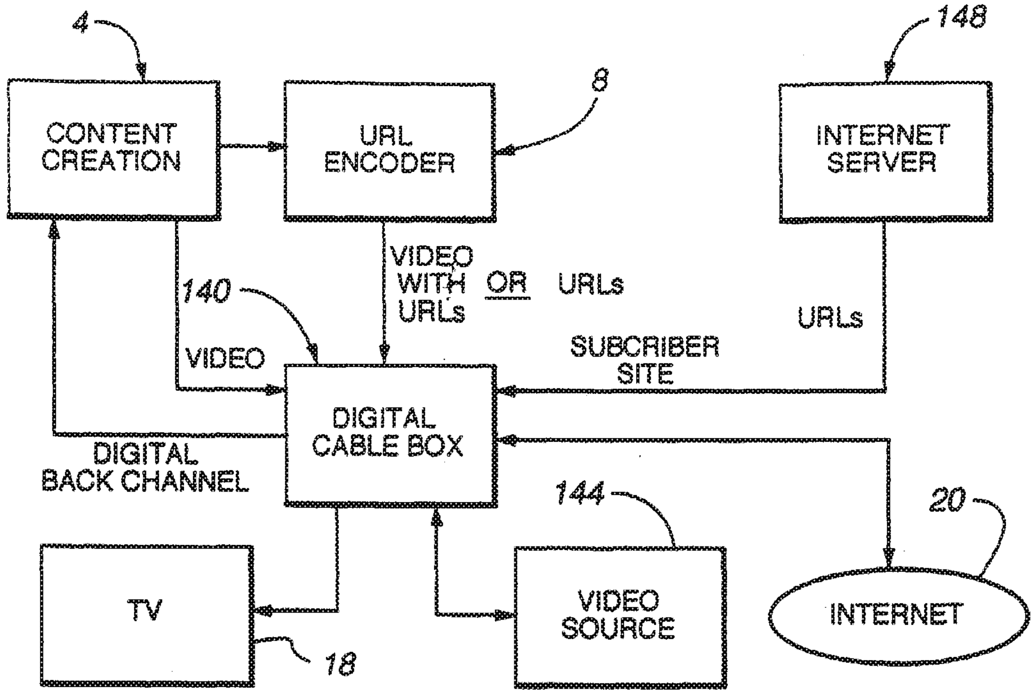

4
148
8
CONTENT CREATION
URL ENCODER
INTERNET SERVER
VIDEO WITH URLs
OR
URLs
140
URLs
VIDEO
SUBCRIBER SITE
DIGITAL CABLE BOX
DIGITAL BACK CHANNEL
144
20
TV
VIDEO SOURCE
INTERNET
18

Fig. 5

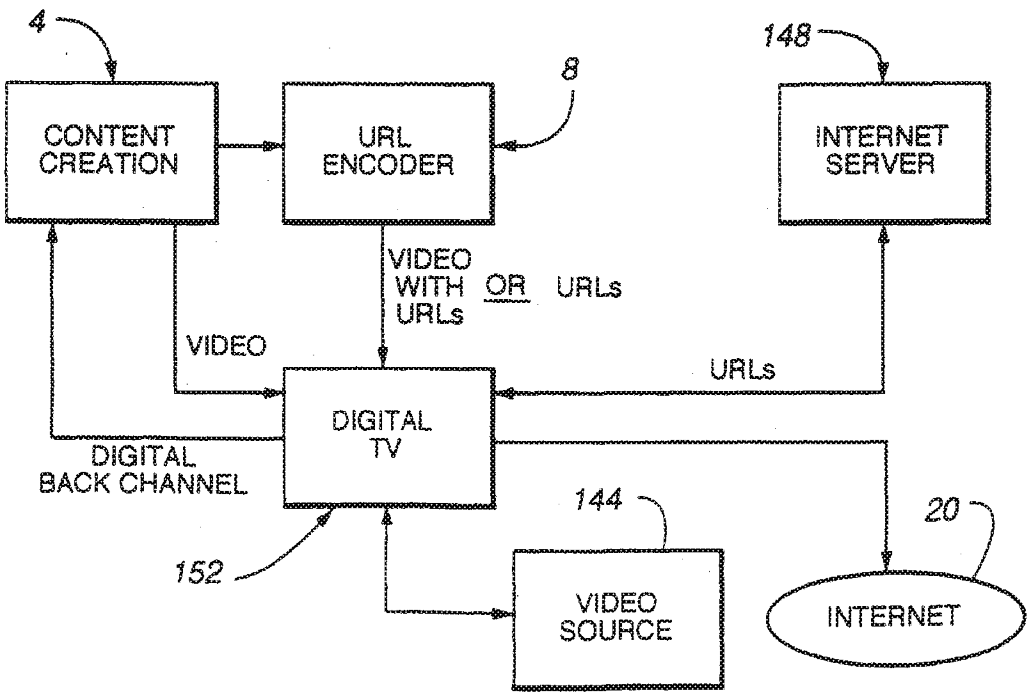

4
8
148
CONTENT CREATION
URL ENCODER
INTERNET SERVER
VIDEO WITH URLs
OR
URLs
VIDEO
URLs
DIGITAL TV
DIGITAL BACK CHANNEL
144
20
152
VIDEO SOURCE
INTERNET

Fig. 6

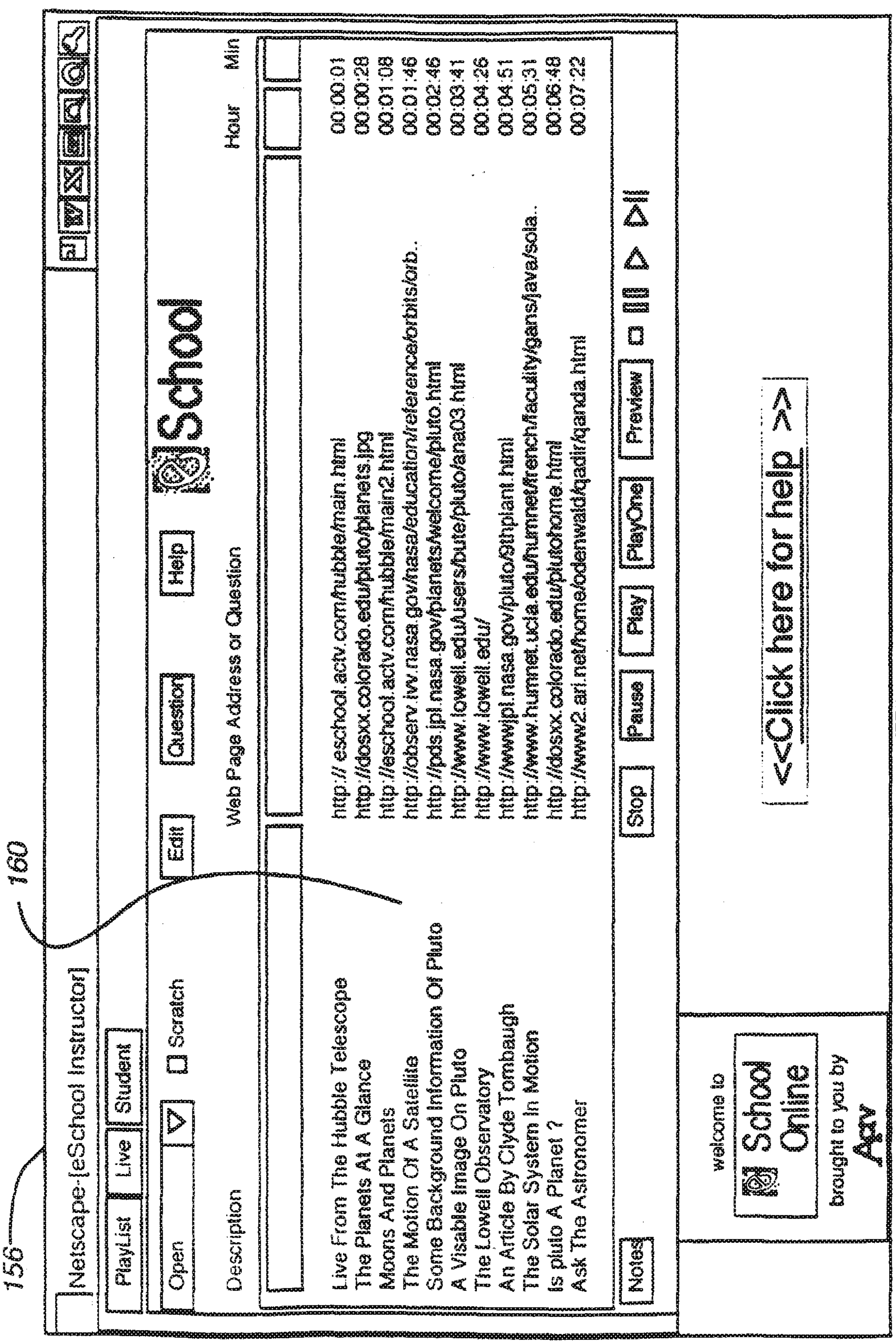

156
160
Netscape-[eSchool Instructor]
PlayList
Live
Student
Open
Scratch
Edit
Question
Help
School
Description
Web Page Address or Question
Hour
Min
Live From The Hubble Telescope
The Planets At A Glance
Moons And Planets
The Motion Of A Satellite
Some Background Information Of Pluto
A Visable Image On Pluto
The Lowell Observatory
An Article By Clyde Tombaugh
The Solar System In Motion
Is pluto A Planet ?
Ask The Astronomer
Notes
Stop
Pause
Play
PlayOne
Preview
welcome to
School Online
brought to you by
<<Click here for help >>

Fig. 7

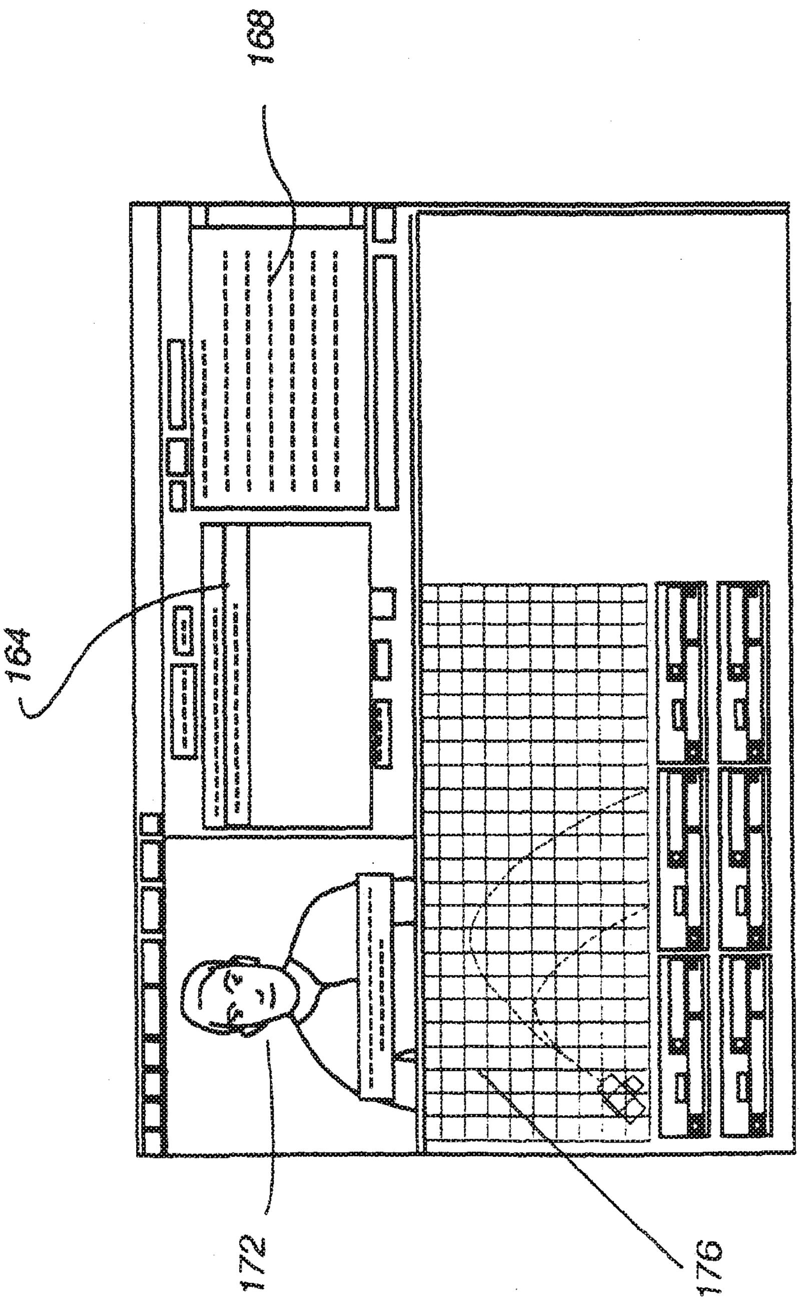


Fig. 8

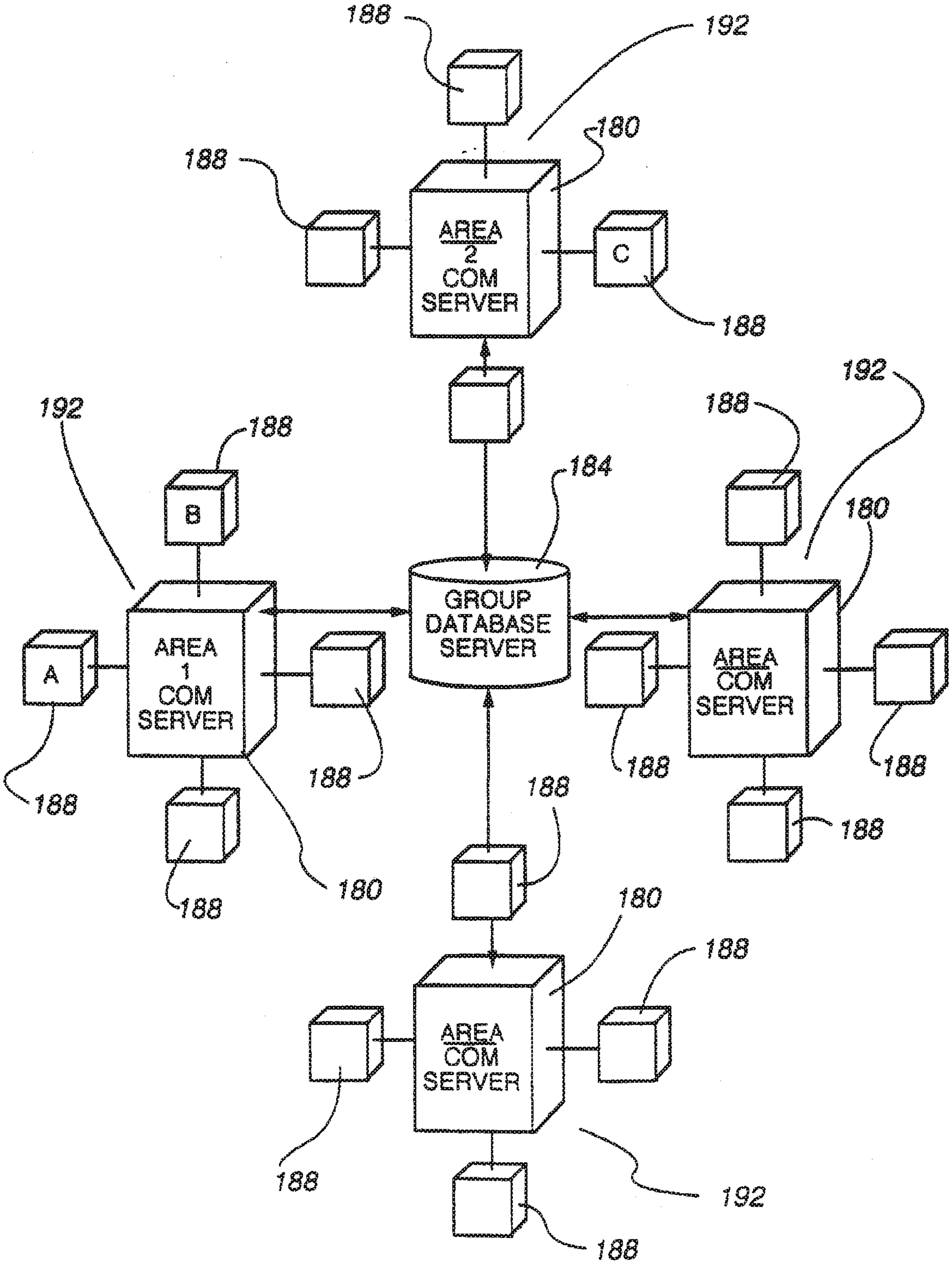

188
192
180
188
AREA
2
COM
SERVER
C
188
192
188
188
192
180
184
B
GROUP
DATABASE
SERVER
A
AREA
1
COM
SERVER
AREA
COM
SERVER
188
188
188
188
188
188
188
180
180
188
AREA
COM
SERVER
188
192
188

Fig. 9

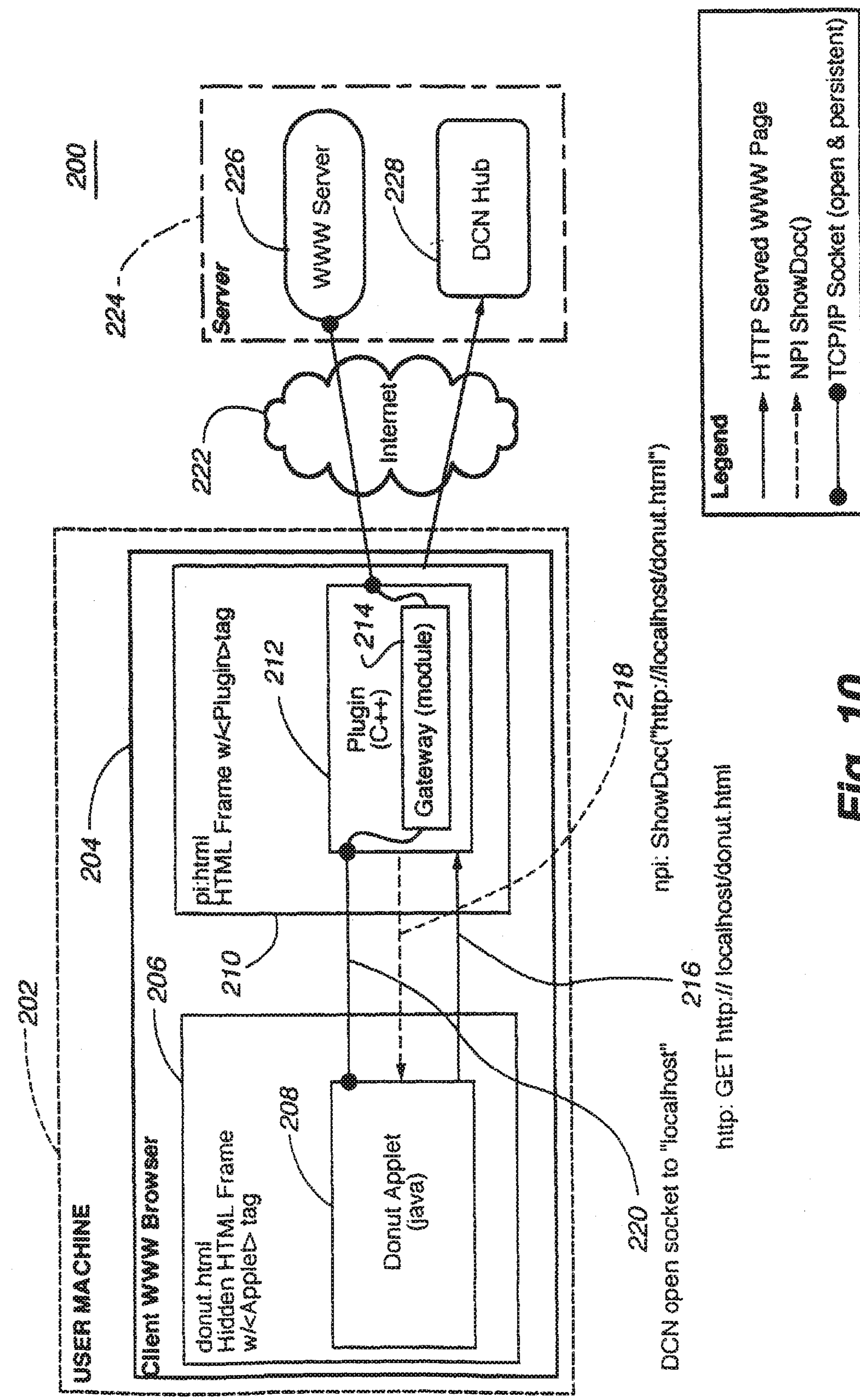

200
202
USER MACHINE
Client WWW Browser
204
206
donut.html
Hidden HTML Frame
w/<Applet> tag
208
Donut Applet
(java)
210
pi.html
HTML Frame w/<Plugin>tag
212
Plugin
(C++)
214
Gateway (module)
216
218
220
DCN open socket to "localhost"
http: GET http:// localhost/donut.html
npi: ShowDoc("http://localhost/donut.html")
222
Internet
224
Server
226
WWW Server
228
DCN Hub
Legend
HTTP Served WWW Page
NPI ShowDoc()
TCP/IP Socket (open & persistent)

Fig. 10

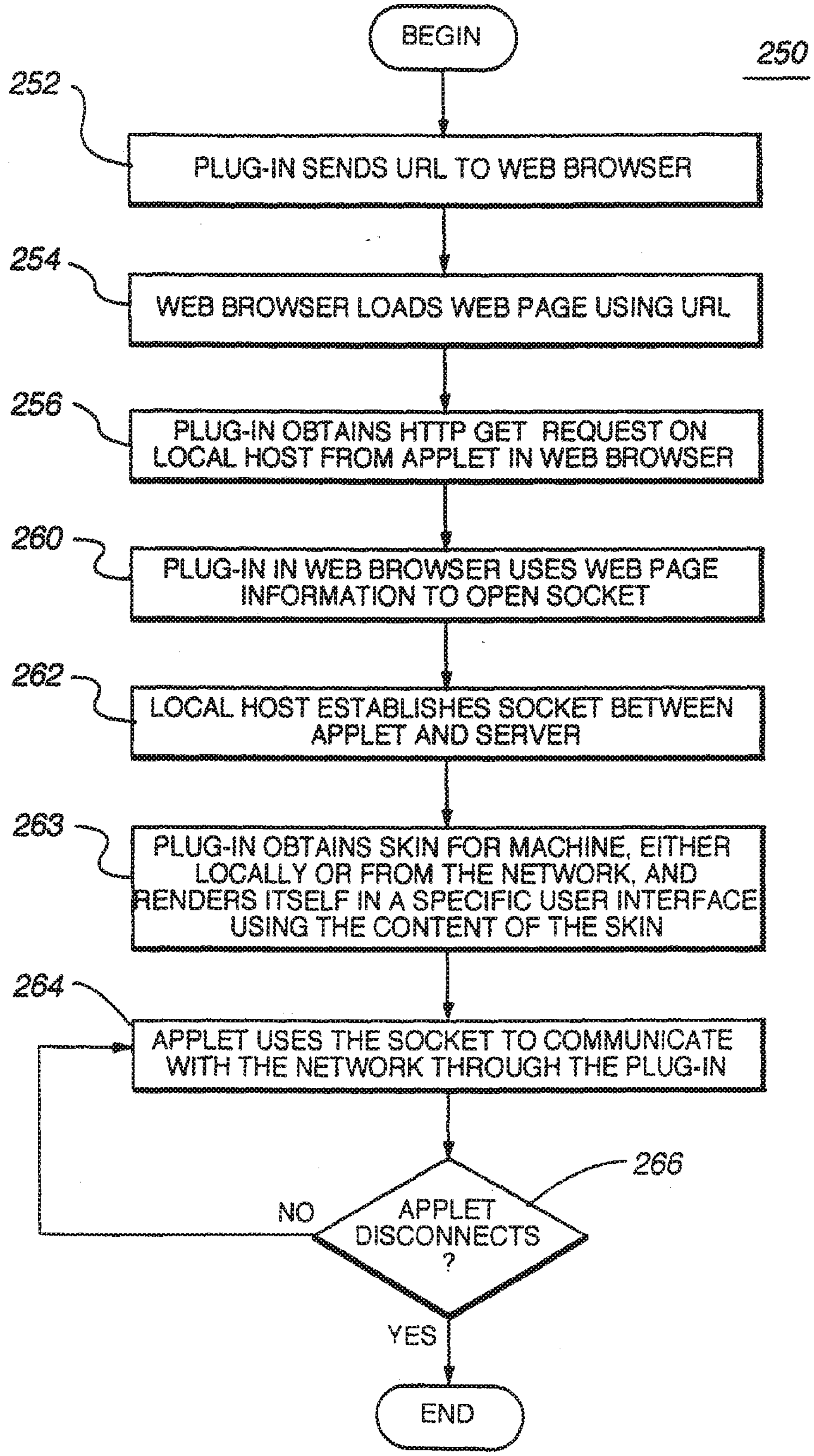

BEGIN
250
252
PLUG-IN SENDS URL TO WEB BROWSER
254
WEB BROWSER LOADS WEB PAGE USING URL
256
PLUG-IN OBTAINS HTTP GET REQUEST ON LOCAL HOST FROM APPLET IN WEB BROWSER
260
PLUG-IN IN WEB BROWSER USES WEB PAGE INFORMATION TO OPEN SOCKET
262
LOCAL HOST ESTABLISHES SOCKET BETWEEN APPLET AND SERVER
263
PLUG-IN OBTAINS SKIN FOR MACHINE, EITHER LOCALLY OR FROM THE NETWORK, AND RENDERS ITSELF IN A SPECIFIC USER INTERFACE USING THE CONTENT OF THE SKIN
264
APPLET USES THE SOCKET TO COMMUNICATE WITH THE NETWORK THROUGH THE PLUG-IN
266
APPLET DISCONNECTS ?
NO
YES
END

Fig. 11

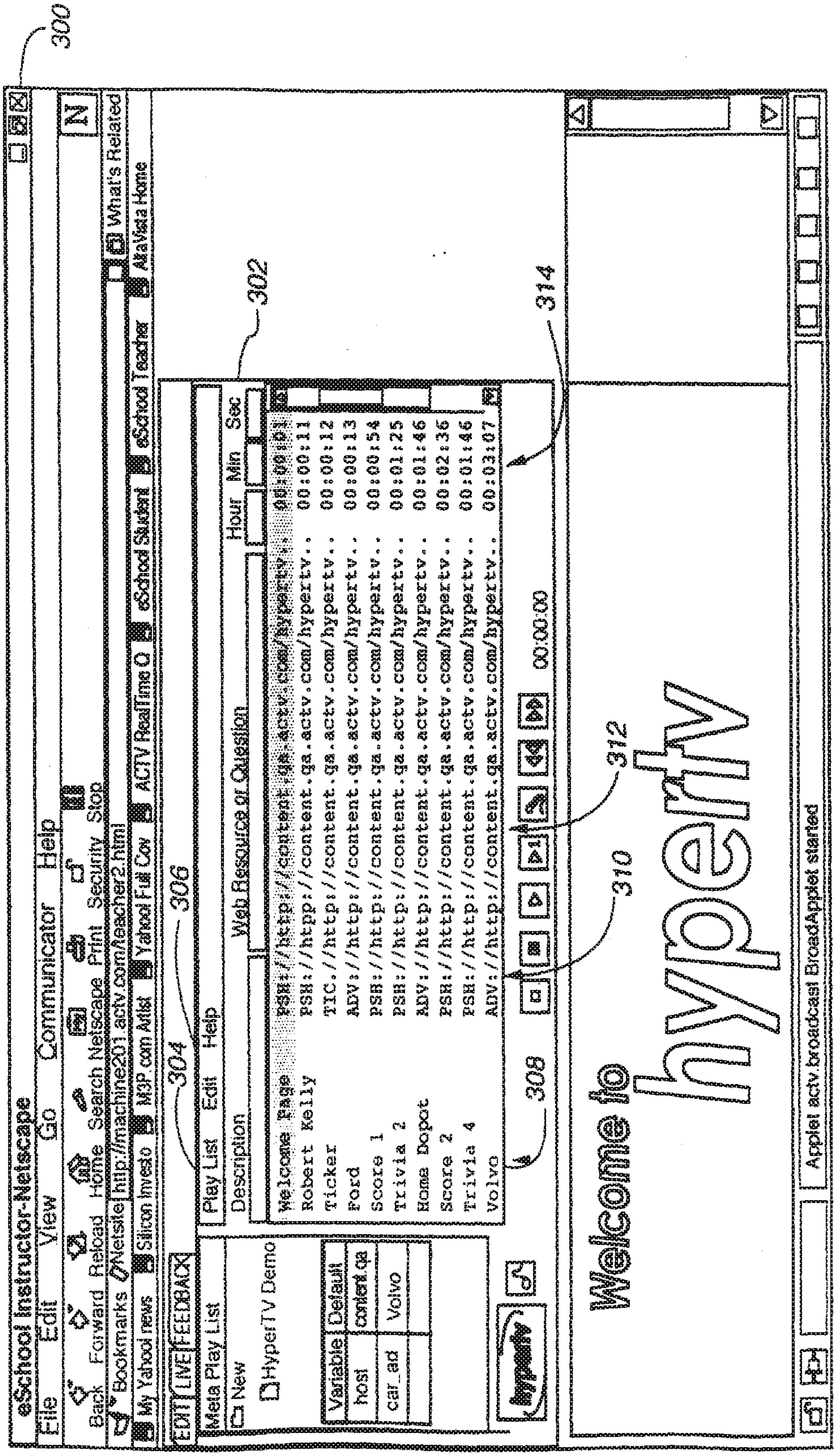

300
eSchool Instructor-Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Bookmarks Netsite http://machine201.actv.com/teacher2.html
What's Related
My Yahoo! news Silicon Investo M3P.com Artist Yahoo! Full Cov ACTV RealTime Q eSchool Student eSchool Teacher AltaVista Home
EDIT LIVE FEEDBACK
Meta Play List
New
HyperTV Demo
Variable Default
host content.qa
car_ad Volvo
302
304
306
Play List Edit Help
Description
Web Resource or Question
Hour Min Sec
Welcome Page PSH://http://content.qa.actv.com/hypertv.. 00:00:01
Robert Kelly PSH://http://content.qa.actv.com/hypertv.. 00:00:11
Ticker TIC.//http://content.qa.actv.com/hypertv.. 00:00:12
Ford ADV://http://content.qa.actv.com/hypertv.. 00:00:13
Score 1 PSH://http://content.qa.actv.com/hypertv.. 00:00:54
Trivia 2 PSH://http://content.qa.actv.com/hypertv.. 00:01:25
Home Depot ADV://http://content.qa.actv.com/hypertv.. 00:01:46
Score 2 PSH://http://content.qa.actv.com/hypertv.. 00:02:36
Trivia 4 PSH://http://content.qa.actv.com/hypertv.. 00:01:46
Volvo ADV://http://content.qa.actv.com/hypertv.. 00:03:07
308
310
312
314
00:00:00
Welcome to
hypertv
Applet actv.broadcast.BroadApplet started

Fig. 12

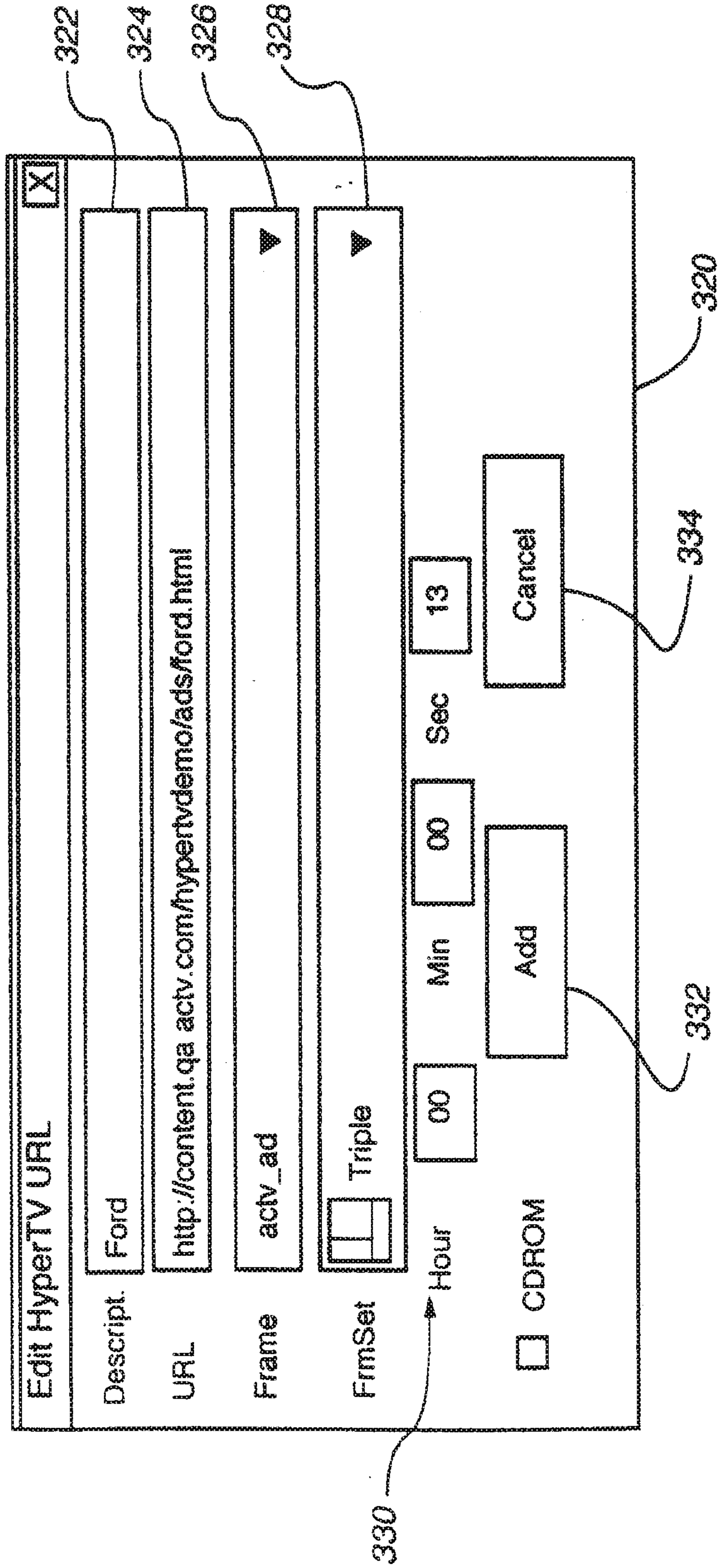

Edit HyperTV URL
X
Descript.
Ford
URL
http://content.qa actv.com/hypertvdemo/ads/ford.html
Frame
actv_ad
FrmSet
Triple
Hour
00
Min
00
Sec
13
CDROM
Add
Cancel
322
324
326
328
320
330
332
334

Fig. 13

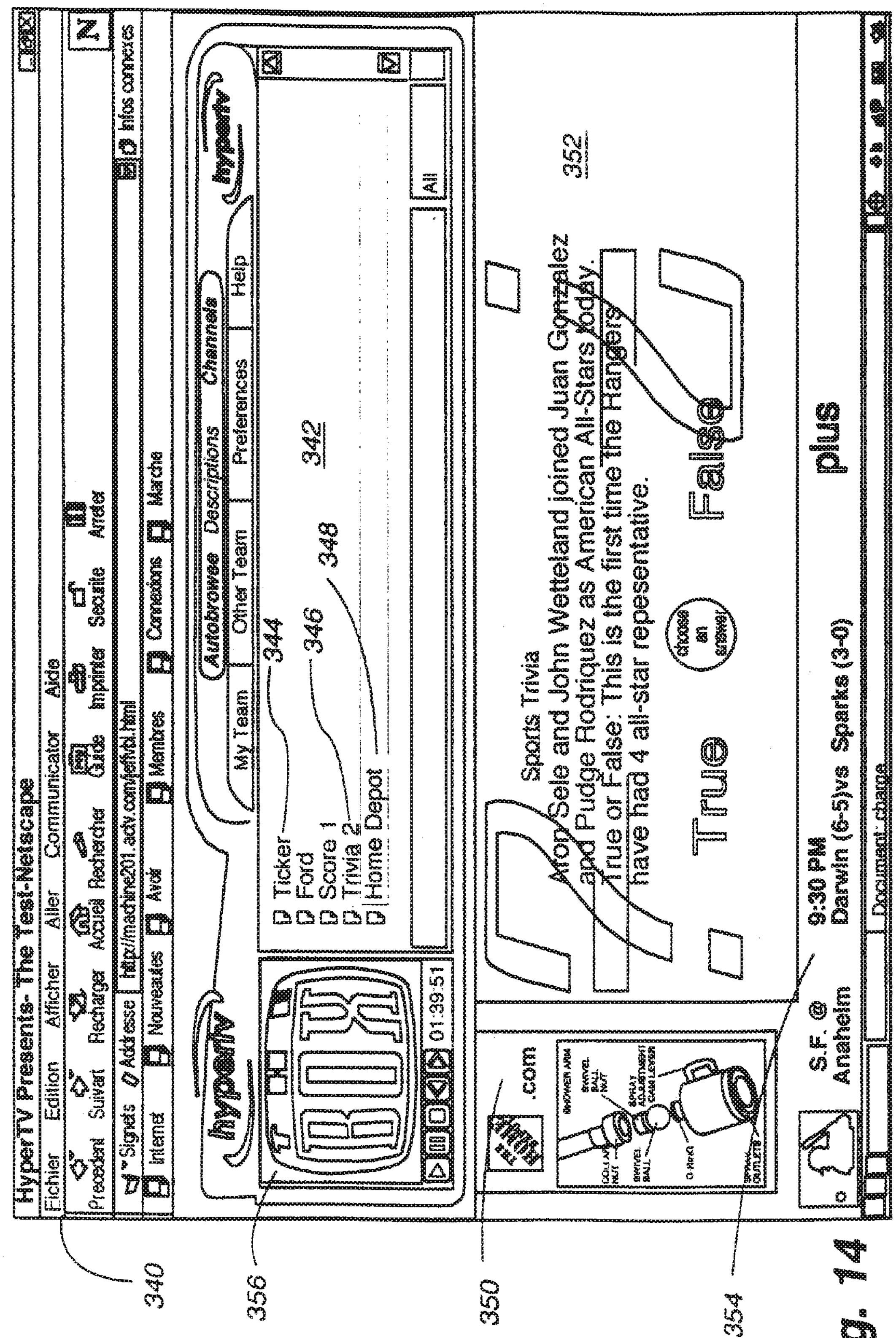

HyperTV Presents- The Test-Netscape
Fichier Edition Afficher Aller Communicator Aide
Precedent Suivant Recharger Accueil Rechercher Guide Imprimer Securite Arreter
Signets Addresse http://machine201.actv.com/jeffvbl.html
Infos connexes
Internet Nouveautes Avoir Membres Connexions Marche
hypertv
01:39:51
Ticker
Ford
Score 1
Trivia 2
Home Depot
My Team
Other Team
Preferences
Help
Autobrowse
Descriptions
Channels
All
.com
Sports Trivia
Aron Sele and John Wetteland joined Juan Gonzalez and Pudge Rodriquez as American All-Stars today.
True or False: This is the first time the Rangers have had 4 all-star representative.
True
choose an answer
False
plus
S.F. @ Anaheim
9:30 PM
Darwin (6-5)vs Sparks (3-0)
Document: charge
340
356
350
354
344
346
348
342
352

Fig. 14

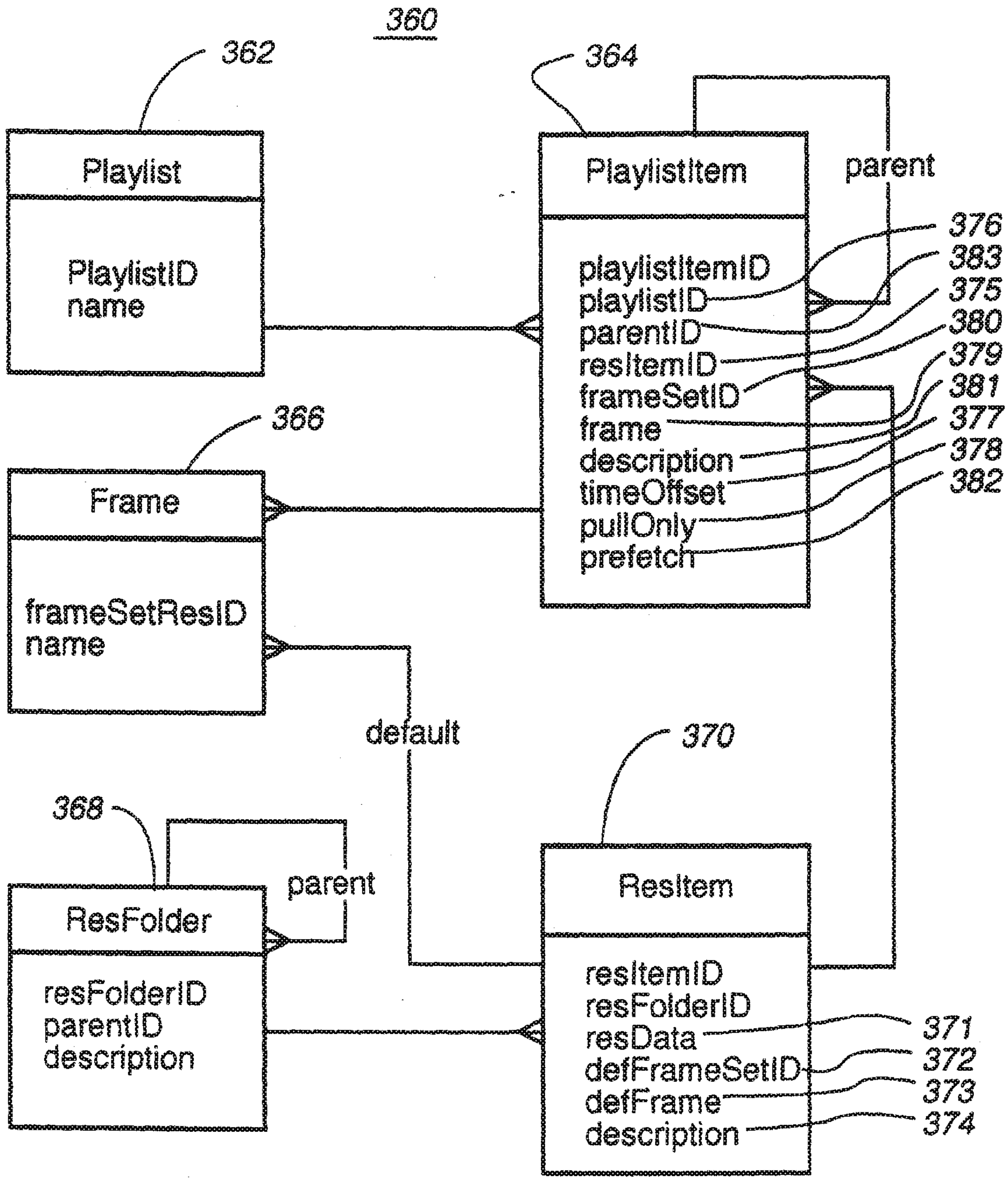


Fig. 15

IPR2022-01227
EXHIBIT 1003 - PAGE 00830

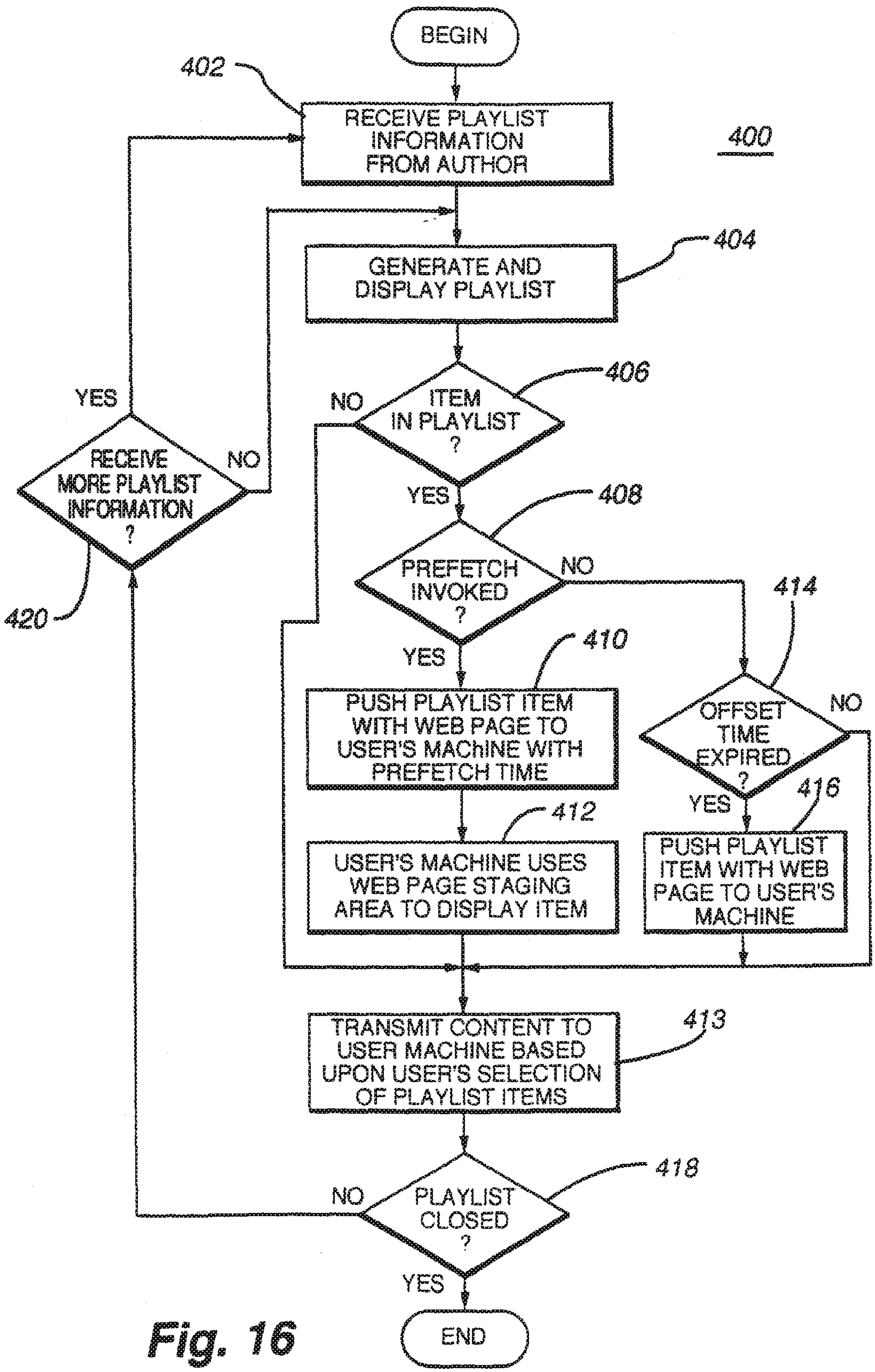

BEGIN
402
RECEIVE PLAYLIST INFORMATION FROM AUTHOR
400
GENERATE AND DISPLAY PLAYLIST
404
ITEM IN PLAYLIST ?
406
NO
YES
RECEIVE MORE PLAYLIST INFORMATION ?
420
YES
NO
PREFETCH INVOKED ?
408
NO
YES
414
PUSH PLAYLIST ITEM WITH WEB PAGE TO USER'S MAChINE WITH PREFETCH TIME
410
OFFSET TIME EXPIRED ?
NO
YES
416
USER'S MACHINE USES WEB PAGE STAGING AREA TO DISPLAY ITEM
412
PUSH PLAYLIST ITEM WITH WEB PAGE TO USER'S MACHINE
TRANSMIT CONTENT TO USER MACHINE BASED UPON USER'S SELECTION OF PLAYLIST ITEMS
413
PLAYLIST CLOSED ?
418
NO
YES
END

Fig. 16

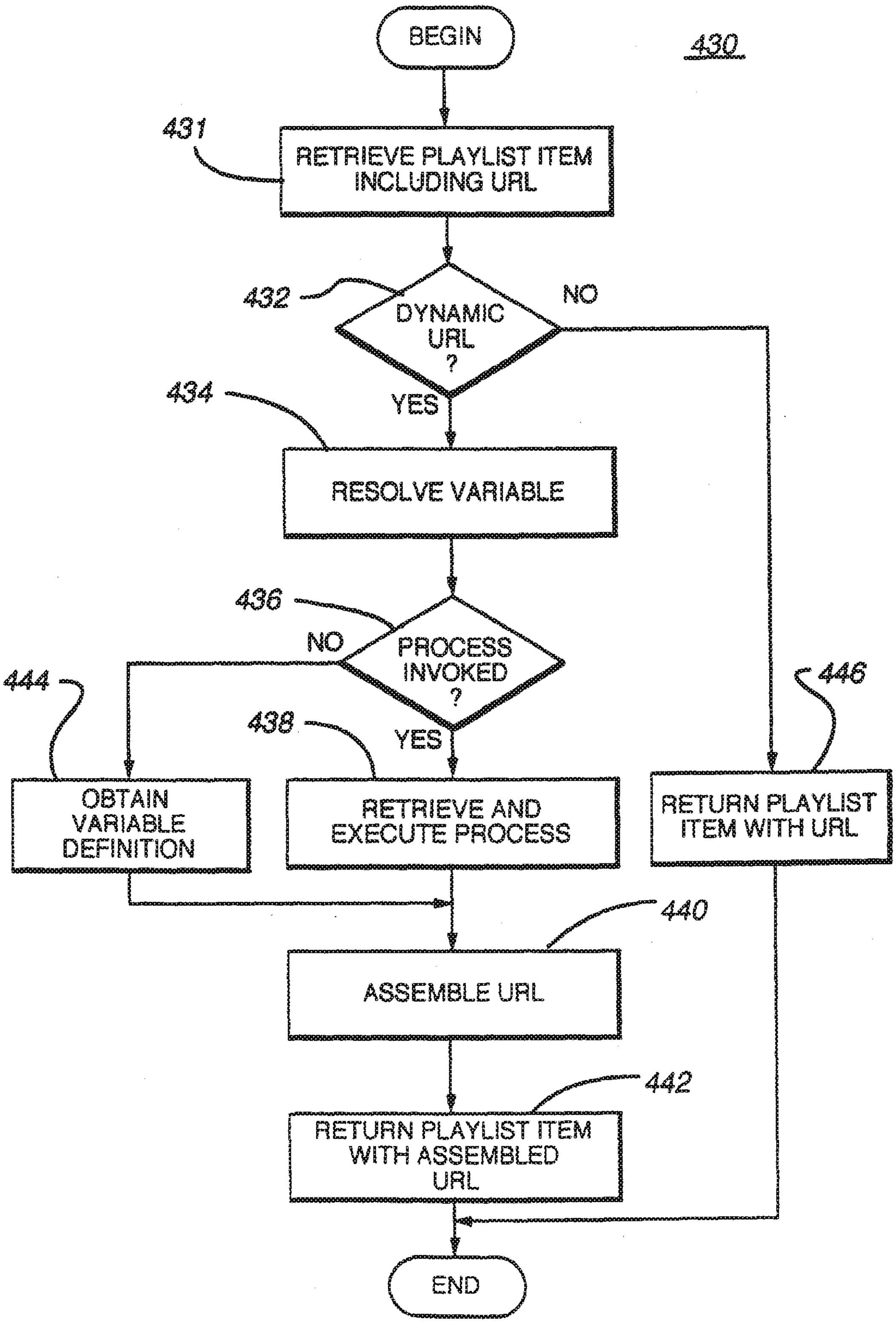

430
BEGIN
431
RETRIEVE PLAYLIST ITEM INCLUDING URL
432
DYNAMIC URL ?
NO
YES
434
RESOLVE VARIABLE
436
PROCESS INVOKED ?
NO
YES
444
438
446
OBTAIN VARIABLE DEFINITION
RETRIEVE AND EXECUTE PROCESS
RETURN PLAYLIST ITEM WITH URL
440
ASSEMBLE URL
442
RETURN PLAYLIST ITEM WITH ASSEMBLED URL
END

Fig. 17

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 395 005 A1

(12) # EUROPEAN PATENT APPLICATION

(43) Date of publication:
03.03.2004 Bulletin 2004/10

(51) Int Cl.7: **H04L 12/56**, H04Q 7/38

(21) Application number: **03255325.7**

(22) Date of filing: **27.08.2003**

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LI LU MC NL PT RO SE SI SK TR
Designated Extension States:
AL LT LV MK

(30) Priority: **30.08.2002 GB 0220239**

(71) Applicant: **Hutchison Whampoa Three G IP (Bahamas) Limited**
Nassau, New Providence (BS)

(72) Inventor: **Danneel, Jerome**
Berkshire, SL6 8RG (GB)

(74) Representative: **Butcher, Ian James et al**
A.A. Thornton & Co.
235 High Holborn
London WC1V 7LE (GB)

(54) **Improved method for streaming in a mobile telecommunications network**

(57) A method for implementing Fast Start packet-switched streaming in a mobile telecommunications network such as a UMTS network over real-time bearers that will improve the user's experience of the service as well as minimising the waste of resources. The present idea specifies the requirements on, in the case of a UMTS network, the UTRAN to make this method work efficiently.

EP 1 395 005 A1

Description

[0001] The present invention relates to the implementation of streaming in a mobile telecommunications network and in particular a Universal Mobile Telecommunications System (UMTS) network which enables the UMTS Terrestrial Radio Access Network (UTRAN) to enhance the experience of streaming users while saving radio network resources.

Background

[0002] Currently over the internet, users can access multimedia files, such as video clips or audio clips, in two ways. The first one is download and the second one is streaming.

When downloading a file, the user first makes a copy of the whole file on his/her terminal and then plays it from the terminal. Hence, there is a certain delay during which the user has to wait before they can view or listen to the file. The larger the multimedia file, the longer the delay.

On the other hand, when streaming a multimedia file, the file is played directly as it arrives from the internet onto the user's terminal. Hence, the delay before starting to play does not depend on the size of the file and the delay could be virtually zero. Another advantage of streaming is that the user does not need to store the whole file on his/her terminal, hence requiring much less memory.

[0003] However, streaming a file requires that there is no interruption in the flow of data coming from the internet, otherwise the multimedia file being played to the user would be interrupted. Because the internet is an unreliable network, such a temporary lack of data is very likely. As a matter of fact, a multimedia file streamed over the internet is sent in individual packets. Although the streaming server (the equipment sending the multimedia file to the user) tries to feed the streaming player (the application running on the user's terminal and showing the data to the user) with a continuous flow of packets, each packet is transmitted individually over the internet on a best-effort basis. The travelling time across the network is then different for each packet and some packets might even be lost due to congestion in the internet.

[0004] Therefore, the only way to guarantee that the streaming player always has data to present to the user is to make sure that the data arrives in the streaming player a few seconds before it is to be played. Hence, all streaming players perpetually buffer a few seconds of multimedia files to accommodate delayed or lost packets. This buffer is of a more or less constant length and basically fills with data coming from the internet and empties when data has been shown to the user. However, this buffering means that the user still has to wait a certain amount of time before the file can start to be played. The longer the buffer, the lower the probability that data is missed from the played file, but also the longer the user has to wait.

[0005] One technique used over the internet to reduce this delay while streaming is called Fast Start. Basically, the streaming server, instead of always sending the data at the rate the file is to be shown to the user, sends the first few seconds of the file at a much higher throughput. The delay before showing the file can be significantly reduced if a high throughput can be achieved on the first few seconds of the file. For instance, if the player needs to buffer 10 seconds of the file to ensure low loss of data, the initial delay could be only 5 seconds if the server can send some initial data at a throughput twice the rate at which the file is shown to the user. Fast Start streaming over the internet requires the player and the server to support this functionality and to agree to use it.

[0006] In a UMTS packet-switched network, both download and streaming delivery techniques will be supported to provide the users with multimedia files. However, streaming over a UMTS network will be able to use real-time bearers. Such bearers do not work on a best-effort basis like the internet, but can guarantee the bit rate and the delay for the transmission of the packets over the network. As a result, the delivery of the packets to the player is much more reliable than over the internet, and temporary congestion in the network should not affect the streaming data flow.

[0007] Nevertheless, because the radio link remains non-reliable by nature (due to changing radio conditions) and because the quality of the radio link (in terms of errors, delays and packet losses) has a very high cost in terms of resources, it is foreseen that some buffering will still be required, even if in smaller amount than for streaming over the internet. Since buffering is required, then Fast Start streaming techniques could also be used to reduce the delay during which the user must wait during the initial buffering. However, fast start streaming over a UMTS network and specifically over a real-time bearer poses a problem.

[0008] Indeed, the efficient use of a real-time UMTS bearer assumes that the terminal is "Quality of Service (QoS) aware", i.e. it can determine the precise characteristics of the bearer needed to carry the service and then request those characteristics from the network. In the case of streaming, the player will determine the bandwidth needed by mean of signalling from the server (the bandwidth parameter in the SDP file describing the multimedia file) and then request a real-time bearer service with the appropriate bit rate from the UMTS network. The bearer service provided by the UMTS core network to the terminal is called Packet Data Protocol (PDP) context. Such a real-time bearer would have the following attributes:

- Traffic class = streaming
- Guaranteed bit rate = derived from the bandwidth parameter

The UMTS core network then requests in turn a bearer service from the UTRAN for this terminal. The bearer service provided by the UTRAN to the UMTS core network for a particular terminal is called Radio Access Bearer (RAB). Its attributes are the same as that of the PDP context above.
Finally, a radio link (radio bearer) is established by the UTRAN with the terminal. This bearer will have characteristics based on the QoS parameters of the RAB. Basically, the radio bearer will allow data to be sent to the user at the guaranteed bit rate but no more, in an attempt to optimise the resources.
Because, the guaranteed bit rate basically equals the rate at which the data are to be played, this means that fast start streaming, even if used, would not reduce the delay, since the throughput of the link is limited in the UMTS network by the guaranteed bit rate.

[0009] 3GPP specifications also allow the terminal to specify a maximum bit rate higher than the guaranteed bit rate. However, the use and interpretation of this maximum bit rate is not very clear in the UTRAN, and might lead to significant waste of resources if set too high and not used properly. Indeed, the 3GPP documents simply recommend the UTRAN to use the maximum bit rate to do code reservation. In the case of streaming, and especially that of fast start streaming, such interpretation might lead to the UTRAN reserving downlink code resource, Node B channel elements and transport resources for a very high bit rate for the lifetime of the bearer. However, these will be used only for the first few seconds, the rest of the lifetime of the bearer being used at the guaranteed bit rate only, With the higher level resources being unecessarily reserved.

Summary of the Invention

[0010] The present invention provides a method of controlling a radio link to a mobile terminal in a mobile communications system, in which:

the terminal transmits a request for a link to be established including a request for a first bit rate and a request for a second bit rate higher than said first bit rate; a link is established with said terminal allocating at least said first bit rate, if sufficient resources are available in the network; and in which a bit rate higher than said first bit rate but not exceeding said second bit rate is allocated to said link for an initial time period, if sufficient resources are available in the network.

[0011] The present inventiom further provides a mobile communications network comprising:

means arranged to receive a request from a mobile terminal for the establishment of a radio link between the mobile communications network and the mobile terminal, the request defines a first bit rate and a second bit rate higher than said first bit rate;
means arranged to establish a radio link at at least said first bit rate but no higher than said second bit rate depending on the resources available in the network; and
means arranged to control the bit rate of the radio link and to reduce the bit rate allocated to the link to a rate no lower than said first bit rate after an initial time period.

[0012] Thus, a method for implementing fast start packet-switched streaming in a mobile telecommunications network such as a UMTS network over real-time bearers that will improve the user's experience of the service as well as minimising the waste of resources is proposed. The present idea also specifies the requirements on the terminal as well as on, in the case of a UMTS network, the UTRAN to make this method work efficiently.

[0013] The present invention will now be discussed in more detail with reference to a UMTS mobile telecommunications network, however it will be appreciated that the described method could be applied to any mobile communications network to achieve the stated aims.

[0014] It will be appreciated that in the specific example of streaming in a UMTS network, the "first bit rate" is equivalent to the guaranteed bit rate and the "second bit rate" is equivalent to the maximum bit rate requested when fast start streaming is required.

[0015] The method first relies on a terminal's ability to request a UMTS bearer with different QoS parameters when fast start is to be used than when fast start is not to be used. The difference will mainly reside in the value of the requested maximum bit rate.

- If fast start is not to be used, then the maximum bit rate will generally be the same as the guaranteed bit rate.
- If fast start is to be used, then the maximum bit rate shall be higher than the guaranteed bit rate.

The exact calculation of the maximum bit rate should be performed by the terminal and be based on its capabilities in terms of radio bearers, buffering and streaming player.

[0016] Secondly, the method also relies on the way that the UTRAN will interpret these parameters and a specific UTRAN behaviour based on the following 3 steps.

[0017] Step 1 corresponds to the call admission process. If the requested maximum and guaranteed bit rates are different, the UTRAN shall try to establish a radio bearer for the maximum bit rate. In the case where this maximum bit rate cannot be established because of load reason (admission control failed), then the UTRAN shall try to establish the highest bit rate possible between the maximum and the guaranteed. Only if the guaranteed bit rate cannot be established should the bearer request be rejected.

[0018] Step 2 starts when a bearer higher than the guaranteed bit rate has been established, and when fast start streaming commences. The UTRAN shall monitor the radio bearer in order to detect the end of the fast start phase. This could be achieved in either of the following two ways:

- The UTRAN could start a timer (set by the operator and dependent on the bit rate) at the setup of the bearer, and declare the end of the fast start phase upon the expiry of the timer.
- The UTRAN could monitor the throughput of the radio bearer, and declare the end of the fast start phase when a certain criteria is met, such as the actual throughput is lower than X kbps for more than Y seconds.

The first method above would probably be easier to implement and optimise than the second one, but the capacity gain of the second might be bigger.

[0019] Step 3 occurs when the UTRAN has detected the end of the fast start phase. At this point the UTRAN shall reconfigure the whole radio bearer to the guaranteed bit rate. This means that all resources (code, channel elements, transport, etc.) will be scaled down to allow only the guaranteed bit rate to be transmitted.

[0020] In the foregoing the present invention has been described in relation to streaming and in particular streaming bearers. The dynamic control of the streaming bearer bit rate allows the implementation of Fast Start streaming on a UMTS system while conserving resources once the Fast Start period has ended. However it will be appreciated that the present invention is more generally concerned with the dynamic control of the streaming bearer bit rate from the UTRAN side whatever the actual data being transmitted represents. For example, any data type which warrants a real time bearer due to its delay sensitive nature and which requires buffering could have applied to it the dynamic control of the bit rate of the bearer which is the subject of the present invention.

[0021] In the foregoing the present invention has been described in relation to the beginning of the life of a bearer. However it will be appreciated that dynamic control of the bit rate allocated to a bearer can be undertaken during the life time of the bearer. In the context of streaming, the streaming player on the mobile terminal may be arranged to keep the streaming server updated with the status of its buffers thereby enabling the streaming player to adjust its data flow according to the requirements of the steaming player's buffers. However, as discussed above, due to the fixed nature of the bearers in terms of bit rate, the streaming server could supply data at a higher rate but this would not translate to the streaming player due to the UTRAN limitations on the bearer bit rate. Additionally the UTRAN has no information on the status of the streaming player buffers. However, the UTRAN does monitor data being sent to it from the core network, and it is therefore proposed that the present invention be applied to the UTRAN so that if the streaming server attempts to send data at a higher rate the UTRAN responds by resetting the bearer to the maximum bit rate, as requested by the mobile terminal at the establishment of the bearer. The streaming player buffers may be emptied if there is interference on the radio link which causes a drop in data rates or if the user utilises a fast forward or rewind option on the streaming player.

Benefits of the present idea

[0022] The present idea proposes a method of implementing fast start streaming in a UMTS packet-switched network.

[0023] It enables a reduction in the time that the user has to wait before the multimedia file can be shown to him.

[0024] However, if radio resources are not sufficient to allow fast start streaming, then the service that the user would get from the network is the same as in the case where fast start was not used.

[0025] Also the present method enables the UTRAN to make an efficient use of all its resources, since the whole radio bearer is reconfigured.

[0026] Finally, the present method is fully compliant with the 3GPP Release 99 and later standards and does not involve any new signalling between the different equipments. As a matter of fact, this method mainly specifies the UTRAN behaviour for a particular type of RAB request for which the 3GPP specifications are quite vague since such behaviour is assumed to be implementation dependent.

Claims

1. A method of controlling a radio link to a mobile terminal in a mobile communications system, in which:

   the terminal transmits a request for a link to be established including a request for a first bit rate and a request for a second bit rate higher than said first bit rate; a link is established with said terminal allocating at least said first bit rate, if sufficient resources are available in the network; and in which a bit rate higher than said first bit rate but not exceeding said second bit rate is allocated to said link for an initial time period, if sufficient resources are available in the network.

2. A method as claimed in claim 1 in which after said initial time period the radio link is allocated a higher bit rate not exceeding said second bit for a further initial time period, if sufficient resources are availa-

ble in the network.

3. A method as claimed in claim 1 or 2 in which the or each initial time period is a predetermined time period.

4. A method as claimed in claim 3 in which the predetermined time period is dependent on the bit rate allocated for the or each initial time period.

5. A method as claimed in claim 3 or 4 in which the allocated bit rate is reduced to said first bit rate after the or each initial time period.

6. A method as claimed in claim 1 or 2 in which the actual data rate used by said link during the or each initial time period is monitored, and when the actual data rate falls to or below the first bit rate the first bit rate is allocated to the link.

7. A method as claimed in any preceding claim in which the link is established at the highest bit rate that resources will allow while not exceeding the second bit rate.

8. A method as claimed in any preceding claim in which the bit rates to be requested are calculated by the terminal.

9. A method as claimed in claim 8 in which said calculation of the second bit rate is based on buffering and streaming player capabilities of the terminal and on capabilities of radio bearers.

10. A method as claimed in any preceding claim in which the said link is established on a real-time bearer.

11. A method as claimed in any preceding claim in which the link is used for streaming of a data file to the terminal.

12. A method as claimed in claim 11 in which the streaming is packet-switched streaming.

13. A method as claimed in claim 11 or 12 in which the data file is a multimedia data file.

14. A method as claimed in any of claims 11 to 13 in which during the first mentioned initial time period Fast Start streaming is conducted.

15. A method as claimed in claim 14 in which, the first bit rate is the bit rate required for streaming and the second bit rate is a higher bit rate used for Fast Start streaming.

16. A method as claimed in any preceding claim in which the mobile telecommunications system is a UMTS.

17. A method as claimed in claim 16 in which the link is established by the UTRAN.

18. A mobile communications network comprising:

    means arranged to receive a request from a mobile terminal for the establishment of a radio link between the mobile communications network and the mobile terminal, the request defines a first bit rate and a second bit rate higher than said first bit rate;
    means arranged to establish a radio link at at least said first bit rate but no higher than said second bit rate depending on the resources available in the network; and
    means arranged to control the bit rate of the radio link and to reduce the bit rate allocated to the link to a rate no lower than said first bit rate after an initial time period.

19. A network as claimed in claim 18 wherein said means arranged to control the bit rate is further arranged to allocate a higher bit rate not exceeding said second bit rate to said radio link, after said initial time period, for a further initial time period, if sufficient resources are available in the network.

20. A network as claimed in claims 18 or19 in which the or each initial time period is a predetermined time period.

21. A network as claimed in claim 20 in which the predetermined time period is dependent on the bit rate allocated for the or each initial time period.

22. A network as claimed in claim 20 or 21 in which the allocated bit rate is reduced to said first bit rate after the or each initial time period.

23. A network as claimed in claim 18 further comprising a data throughput monitoring means which is arranged to monitor the actual data rate used by said link during the or each initial time period, and when the actual data rate falls to or below the first bit rate the first bit rate is allocated to the link.

24. A network as claimed in any of claims 18 to 23 in which the link is established at the highest bit rate that resources will allow while not exceeding the second bit rate.

25. A network as claimed in any of claims 18 to 24 in which the bit rates to be requested are calculated by the terminal.

26. A network as claimed in claim 25 in which said calculation of the second bit rate is based on buffering and streaming player capabilities of the terminal and on capabilities of radio bearers.

27. A network as claimed in any of claims 18 to 26 in which the said link is established on a real-time bearer.

28. A network as claimed in any of claims 18 to 27 in which the link is used for streaming of a data file to the terminal.

29. A network as claimed in claim 28 in which the streaming is packet-switched streaming.

30. A network as claimed in claim 28 or 29 in which the data file is a multimedia data file.

31. A network as claimed in any of claims 28 to 30 in which during the or each initial time period Fast Start streaming is conducted.

32. A network as claimed in claim 31 in which the first bit rate is the bit rate required for streaming and the second bit rate is a higher bit rate used for Fast Start streaming.

33. A network as claimed in any of claims 18 to 32, being a UMTS.

34. A UMTS network as claimed in claim 33 in which the UTRAN comprises the means arranged to establish.

35. A UMTS network as claimed in claim 34 in which the UTRAN comprises the means arranged to control.

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 03 25 5325

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.7)
X	US 6 072 787 A (JOKINEN HARRI ET AL) 6 June 2000 (2000-06-06) * column 4, line 3-19 * * column 5, line 44-60 * * column 6, line 5-9 * * column 6, line 17-21 * * column 6, line 36-43 * * column 6, line 60-67 * * column 8, line 17-39 * ---	1,2,6-8, 10-13, 16-19, 23-25, 27-30, 33-35	H04L12/56 H04Q7/38
X	WO 02 06986 A (SHAREWAVE INC) 24 January 2002 (2002-01-24) * figure 25 * * abstract * * page 38, line 13-28 * * page 49, line 6-10 * * page 50, line 4-11 * -----	1,2,6-8, 10-13, 18,19, 23-25, 27-30	

TECHNICAL FIELDS SEARCHED (Int.Cl.7)
H04L
H04Q

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	17 December 2003	Chimet, D

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 03 25 5325

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

17-12-2003

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 6072787 A	06-06-2000	FI 96558 B	29-03-1996
		AU 696802 B2	17-09-1998
		AU 3524095 A	19-04-1996
		CA 2200307 A1	04-04-1996
		EP 0783811 A2	16-07-1997
		FI 971270 A	26-03-1997
		WO 9610305 A2	04-04-1996
		JP 10509566 T	14-09-1998
		US 6240079 B1	29-05-2001
WO 0206986 A	24-01-2002	AU 6319401 A	30-01-2002
		EP 1302048 A2	16-04-2003
		WO 0206986 A2	24-01-2002

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 418 756 A2

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
12.05.2004 Bulletin 2004/20

(51) Int Cl.7: **H04N 7/167**, H04N 7/16

(21) Application number: **03104017.3**

(22) Date of filing: **29.10.2003**

(84) Designated Contracting States:
AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LI LU MC NL PT RO SE SI SK TR
Designated Extension States:
AL LT LV MK

(30) Priority: **29.10.2002 US 284049**

(71) Applicant: **General Instruments, Motorola Inc**
Horsham PA 19044 (US)

(72) Inventors:
- **Chen, Annie On-yee**
92014, Del Mar (US)
- **Tang, Lawrence W**
92128, San Diego (US)
- **Murphy, Patrick**
92123, San Diego (US)
- **Okimoto, John I**
92128, San Diego (US)
- **Cochran, Keith R.**
San Diego 92108 (US)
- **Hutchings, George T**
18901, Doylestown (US)

(74) Representative: **McCormack, Derek James et al**
Motorola
European Intellectual Property Operations
Midpoint
Alencon Link
Basingstoke Hampshire RG21 7PL (GB)

(54) **Method and system for encrypting material for distribution**

(57) Streaming content is encrypted by segmenting the content into a plurality of crypto periods, and by encrypting the content for each of a plurality of crypto periods with a different cryptographic key. The crypto periods may be based on either (i) fixed time intervals, (ii) a fixed number of packets, (iii) a fixed marker count, or (iv) a pseudo random number of packets. Methods are provided for determining how to record the key changing criteria, and how to convey this information to video on demand (VOD) servers.

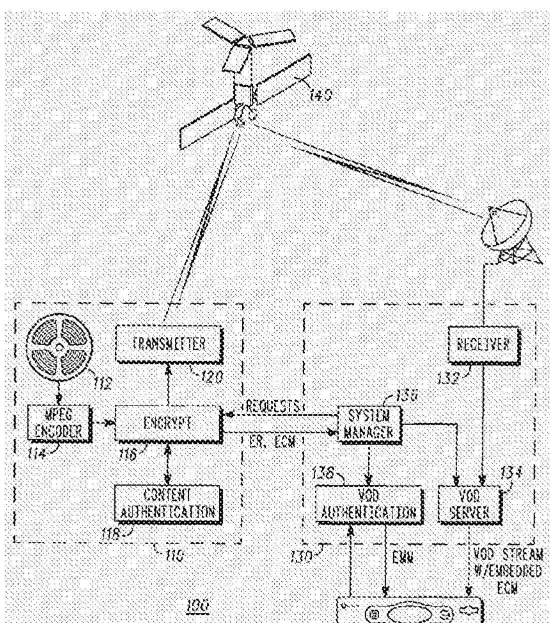


FIG. 1

EP 1 418 756 A2

Description

TECHNICAL FIELD OF THE INVENTION

[0001] The invention relates to a method and a system for encrypting material such as video material for distribution. In particular, it relates conditional access and copy protection techniques, and more particularly to such techniques for interactive, on-demand digital program content such as video-on-demand (VOD) programming distributed via cable and satellite networks.

BACKGROUND

[0002] Recent advances in cable and satellite distribution of subscription and "on-demand" audio, video and other content to subscribers have given rise to a growing number of digital set-top boxes (sometimes referred to as Digital Consumer Terminals or "DCTs") for decoding and delivering digitally broadcast programming. These set-top boxes often include additional circuitry to make them compatible with older analog encoding schemes for audio/video distribution. As the market for digital multimedia content of this type grows and matures, there is a corresponding growth of demand for new, more advanced features.

[0003] Video-on-demand (hereinafter VOD) and audio-on-demand are examples of features made practical by broadband digital broadcasting via cable and satellite. Unlike earlier services where subscribers were granted access to scheduled encrypted broadcasts (e.g., movie channels, special events programming, pay per view purchases, etc.), these on-demand services permit a subscriber to request a desired video, audio or other program at any time and to begin viewing the content at any point therein. Upon receiving the request for programming (and, presumably, authorization to bill the subscriber's account), the service provider then transmits the requested program to the subscriber's set-top box for viewing/listening. The program material is typically "streamed" to the subscriber in MPEG format for immediate viewing/listening, but can also be stored or buffered in the set-top box (typically on a hard-disk drive or "HDD") for subsequent viewing/listening.

[0004] The Motion Picture Association of America (hereinafter MPAA) is a trade association of the American film industry, whose members include the industry's largest content providers (i.e., movie producers, studios). The MPAA requires protection of VOD content from piracy. Without adequate security to protect their content, its member content providers will not release their content (e.g., movies) for VOD distribution. Without up-to-date, high-quality content, the VOD market would become non-viable.

[0005] Access control methods, which may include encryption, are continually evolving to keep pace with the challenges of video-on-demand (VOD) and other consumer-driven interactive services. With VOD, head-end-based sessions are necessarily becoming more personalized. In this scenario, video streams are individually encrypted and have their own set of unique keys.

[0006] One key area of concern, especially for direct content providers and movie companies, is VOD copy protection. The method by which content is produced and delivered to consumers is constantly changing. Under the newest scenarios, content delivery can occur over data backbones, satellite networks and the Internet, increasing the potential for hackers to get digitally perfect copies of the VOD content. As the VOD industry develops and adapts to the piracy threat by providing more sophisticated encryption schemes, piracy becomes more difficult, but the potential gain to the video "pirate" for achieving successful encryption breaches (successful content copying) remains a considerable attraction to hackers.

[0007] Assuming that physical security and network security measures are adequate at the movie company, the VOD encoding company and at the MSO (Multiple System Operator) or satellite operator's facilities, the primary points of VOD vulnerability to piracy occur when VOD content is transmitted over widely accessible communication networks such as a satellite channel, the Internet or a cable system. Such transmissions can occur between the movie company and the VOD encoder, between the VOD encoder and the MSO or satellite operator, and between the MSO or satellite operator and the VOD customer. Because of the ease with which such transmissions can be intercepted, these are the points where the risk of piracy is the greatest.

SUMMARY OF THE INVENTION

[0008] According to the invention, techniques are provided to pre-encrypt VOD material with a changing cryptographic key and to convey this information to VOD servers so that the VOD servers can send out the corresponding ECMs (Entitlement Control Messages) when the encrypted content is delivered to a consumer's digital set top.

[0009] Further according to the invention, multiple encryption keys are added when pre-encrypting VOD material. More specifically, methods are provided for determining when to change encryption keys; how to record the key changing criteria, and how to convey this information to the VOD servers.

[0010] Further according to the invention, streaming content is encrypted by segmenting the content into a plurality of crypto periods, and encrypting the content for each of a plurality of crypto periods with a different cryptographic key. The crypto periods may be established as follows:

1) <u>Fixed crypto period:</u> Define a crypto time interval and change the key each time the crypto time-interval passes.

2) Fixed number of packets: Determine a number of content packets "n" corresponding to a suitable time interval and change the cryptographic key every "n" packets.

3) Fixed "marker" count: Using a suitable MPEG-II field type as a "marker", such as an I-frame header, change the cryptographic key every time "n" markers have passed in the stream, where "n" is selected to produce a suitable crypto period. The MPEG-II I-frame header is one example of a suitable "marker." Alternatively, any other suitable, recurring MPEG-II encoding element could be used as a stream "marker" to delimit segments of the MPEG-II stream.

4) Random crypto period: Change the crypto-period randomly within upper and lower constraints on the crypto period, using a pseudo-random algorithm. Calculate a number of packets for each crypto period and change the key after that number of packets. Generate an index file indicating at which packet numbers the encryption key should be changed.

[0011] The invention is particularly useful for generating rapidly changing encryption keys, and for methods of communicating how and when to change the keys in the context of, for example, the MediaCipher-II conditional access (CA) system available from the Broadband Communications Sector of Motorola, Inc., Horsham, Pennsylvania, USA. Motorola's MediaCipher-II system is capable of changing keys at rates (crypto periods) which are measured in fractions of a second, rather than several seconds.

GLOSSARY

[0012] Unless otherwise noted, or as may be evident from the context of their usage, any terms, abbreviations, acronyms or scientific symbols and notations used herein are to be given their ordinary meaning in the technical discipline to which the invention most nearly pertains. The following glossary of terms is intended to lend clarity and consistency to the various descriptions contained herein, as well as in prior art documents:

CA Conditional Access. A means by which access to content is granted only if certain prerequisite conditions are met (e.g., payment of a subscription fee, time-dependent license, etc.)

CAS Conditional Access System. A means of allowing system users to access only those services that are authorized to them, comprises a combination of authentication and encryption to prevent unauthorized reception

CP Crypto Period. A period covering a portion of an encrypted stream during which a specific encryption key is valid.

ECM Entitlement Control Message. Entitlement Control Messages are private conditional access information which specify control words and possibly other, typically stream-specific, scrambling and and/or control parameters.

EMM Entitlement Management Message. Conditional access messages used to convey entitlements or keys or other parameters to users, or to invalidate or delete entitlements or keys. For example, an EMM can be used in combination with an ECM to determine an encryption key. Without the EMM, the key cannot be derived. The following categories of EMM are possible:

EMM-G: EMM for the whole audience
EMM-S: Shared EMM between the elements of a group.
EMM-U: EMM for a single client.

ER Encryption Record. Contains information about how specific program content is encrypted, and rules for decoding.

ERS Encryption Renewal System. A system by which a conditional access license is renewed.

Internet The Internet (upper case "I") is the vast collection of inter-connected networks that all use the TCP/IP protocols. The Internet now connects many independent networks into a vast global internet. Any time two or more networks are connected together, this results in an internet (lower case "i"; as in international or inter-state).

MPAA Motion Picture Association of America

MPEG Moving Pictures Experts Group

MPEG-II MPEG-2 is the standard for digital television (officially designated as ISO/IEC 13818, in 9 parts).

MSO Multiple System Operator. A company that owns multiple cable systems.

PCR Program Clock Reference. PCR information is embedded into MPEG-II streams to accurately synchronize a program clock on the receiving system to the MPEG-II stream.

IPR2022-01227
EXHIBIT 1003 - PAGE 00843

VOD Video-On-Demand. The service of providing content through subscriber selection off a large menu of options, available to a viewer at any time.

[0013] Embodiments of the present invention will now be described by way of example with reference to the accompanying drawings, in which:

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] **Figure 1** is a block diagram of a system for delivering pre-encrypted video content, in accordance with the invention.

[0015] **Figure 2A** is a diagram showing a changing-key encryption scheme for pre-encrypted content using a fixed crypto period, in accordance with the invention.

[0016] **Figure 2B** is a diagram showing a changing-key encryption scheme for pre-encrypted content using a crypto period based on a fixed number of packets, in accordance with the invention.

[0017] **Figure 2C** is a diagram showing a changing-key encryption scheme for pre-encrypted content using a crypto period delimited by a fixed number of MPEG-II I-frames, in accordance with the invention.

[0018] **Figure 2D** is a diagram showing a changing-key encryption scheme for pre-encrypted content using a "random" crypto period, in accordance with the invention.

DETAILED DESCRIPTION OF EMBODIMENTS OF THE INVENTION

[0019] The invention relates to conditional access and copy protection techniques and more particularly to such techniques for interactive, on-demand digital program content such as video-on-demand (VOD) programming distributed via cable and satellite networks.

[0020] In order to protect against interception and copying of digital program content, a pre-encryption procedure is employed whereby server-based VOD content is stored in an encrypted form, then delivered directly to viewers without further encryption processing. The VOD content is encrypted at the point where it is encoded, and is distributed to content resellers (e.g., MSO's, satellite operators, etc.) in encrypted form. Content encoders generally do not distribute directly to end-users (viewers). Typically, encryption is accomplished separately and uniquely for each reseller.

[0021] **Figure 1** is a block schematic diagram of a system 100 for delivery of pre-encrypted program content, within which an embodiment of the present invention can be incorporated. The system 100 is suitably a conditional access system (CAS) which is a system for granting conditional access to certain digital content (movies, etc.), the "conditions" being licensing conditions (fee paid, access granted starting on date xx/xx/xx at xx:xx until yy/yy/yy at yy:yy, etc.). It is noted that although the entire system 100 is not typically included in one CAS, it could be.

[0022] At a content encoder's location 110, master content 112 (e.g., movies and other program content) is encoded into digital form via a suitable (e.g., MPEG-II) encoder 114. This content is then encrypted in an encryption system 116, to be "encrypted content." A content authorization system 118 is used to, e.g., manage, renew and verify valid licensing for the encrypted content. This can permit, for example, encryption by the encryption system 116 only if valid licensing exists for any particular destination. At a minimum, system 118 will control whether encryption can occur, independently of content destination. The encryption system 116 can generate a "personalized" encryption for each destination content reseller (e.g., MSO). Such a feature is not, however, required. Instead, the same encryption process could be used for a plurality of different MSOs. The encrypted content is transmitted via a transmitter (XMIT) 120 over a suitable transmission medium 140 to a receiver 132 at a reseller's location 130. The transmission medium is shown as being a satellite, but it can be the Internet, a cable network, or any other suitable delivery mechanism.

[0023] The receiver 132 receives the encrypted content and stores it in a VOD server 134 from which it can be re-transmitted to end-users. A system manager 136 (e.g., computer system that controls operation of a reseller's various transmission and communications resources) communicates with the encryption system 116 to make requests for program content, and to receive encryption records (ER) defining how the requested program content is encrypted/encoded and to receive entitlement control messages (ECMs) associated with the encryption of the program content. Typically, the encryption system 116 and the system manager 136 are parts of an ECM Renewal System (ERS) by which authorizations to distribute/decode program content are managed and renewed. It should be appreciated, however, that the ECM renewal can be separate from the other functions included in encryption system 116. As an example, a centralized ERS can be provided. It is also noted that the System Manager 136 would typically be provided by the VOD vendor, although it may be provided by others.

[0024] At the reseller's (e.g., MSO's) location, a user authorization system 138 ("VOD Auth.") receives requests from end users for program content, and verifies that appropriate authorizations are in place for the end user to view the requested content. If the appropriate authorizations are in place, then the user authorization system 138 instructs the VOD server 134 to deliver the requested (encrypted) content to the user's VOD playback device 150 (e.g., set-top box) and generates an Entitlement Management Message (EMM) for the requested content for delivery to the VOD playback device 150, along with the requested content. In an alternate embodiment, the EMM is sent well in advance, e.g., from the CAS.

[0025] An ECM contains encryption information specific to the program content which, in combination with a valid EMM, can be used to derive a decryption key for decrypting the content. ECMs are typically embedded within the program content, and due to the encryption mechanisms employed cannot be used to derive valid encryption keys absent a valid EMM for the content. EMMs may also include conditional access information, such as information about when, how many times, and under what conditions the content may be viewed/played.

[0026] Those skilled in the art will appreciate that when the inventive concepts are used with pre-encrypted content, ECM authorizations will change over time. Thus, ECM data embedded in the content will need to be updated with "renewed" ECMs, or ECMs with authorizations based on subscriber specific rights (for example to copy one or more times). With multiple key changes in the content, the server (which "plays out" the content with the ECMs) must know when to switch ECM sets from one crypto period to the next. Several methods to accomplish this synchronization are disclosed herein. It should also be appreciated that the decoder will decrypt (if it has the proper ECMs) by looking at the transport scrambling control bits in the MPEG packet headers.

[0027] A technique that can be used to improve the security of encrypted streaming content such as VOD content is to change the cryptographic keys (encryption keys) at a plurality of points within the content. In order to make it more difficult for "pirates" to steal these keys, it is desirable to use as many different cryptographic keys as possible to encrypt one item of content. However, this creates a number of new issues:

1) Determining the number of sets of cryptographic keys that should be employed to encrypt one item of content, and determining an upper limit on how frequently keys can be changed.

2) Determining how and where, within the program content, to effect the cryptographic key changes, and how to encode those key changes.

3) Determining how to communicate the cryptographic key sets to VOD servers.

4) Determining how to synchronize cryptographic key changes with the corresponding ECMs when the content is streamed to the consumer at time of purchase..

5) Determining how to handle the ECM renewal process.

[0028] The inventive technique addresses these issues by defining a cryptographic key change methodology that permits rapid key changes with straightforward, simple key change synchronization at the time of decryption. This is accomplished, in part, by taking advantage of the MPEG-II data stream structure.

[0029] Present encryption schemes employ a simple, conventional two-key encryption technique to encrypt VOD content. Both keys taken together are essentially a single "cryptographic key set" used to encrypt the entire content. For example, symmetric (i.e., private) keys can be used for encryption. In an alternate implementation, one of the keys can comprise a "public key", and be delivered with the content. The other key is required in combination with the public key to decrypt the content, and is delivered as part of a successful authorization or licensing process. Neither key is useful absent the other key. Although a public key implementation is possible, a private key approach is currently the preferred implementation.

[0030] A problem with encrypting the VOD content with a single set of keys is that an aggressive "attack" using exhaustive cryptographic "cracking" techniques (e.g., a "brute force" approach) could discover a set of keys that will decode the content. Once broken, the content can be reproduced "in the clear" (i.e., unencrypted), thereby completely thwarting the security offered by the encryption scheme. As is well known in the art, key size is a factor in minimizing the likelihood of a successful brute force attack.

[0031] For highest security and greatest protection against cryptographic "cracking" attacks by "pirates", it is highly desirable to increase the number of separate cryptographic keys used by changing the keys at numerous points during the encryption process. The greater the number of "crypto periods" (separately encrypted segments of the content), the more difficult it becomes to "crack" the encryption scheme. If, for example, cryptographic keys were to be changed every 0.5 seconds within a VOD stream (i.e., a crypto period of 0.5 seconds), then the would-be "pirate" would be forced to crack the encryption scheme for each and every 0.5 seconds of content. Each successful breach of encryption security would only produce 0.5 seconds of "clear" (unencrypted) content. For a 90 minute movie, this would require 10,800 separate successful breaches of the encryption scheme. Given the time and effort required to accomplish each breach, this presents a formidable barrier to piracy.

[0032] The inventive technique maintains all cryptographic keys separate from the encoded/encrypted content. A set of ECMs (Entitlement Control Messages) conveying information about a set of keys is multiplexed into the VOD stream by the VOD server when delivering the VOD content to an end user's VOD playback device (e.g. set-top box). A separate EMM (Entitlement Management Message) from an authorization system is delivered to the VOD playback device. The EMM contains the remaining information required to decode/decrypt the VOD content.

[0033] There are two points in the streaming VOD de-

IPR2022-01227
EXHIBIT 1003 - PAGE 00845

livery process that dictate the practical upper limit on how frequently keys can be changed within VOD content: the VOD server and the set-top box. Since the content can be encoded "off-line", in a non real-time fashion, there is virtually no practical limit to how frequently cryptographic keys can be changed on the encoding/encryption side of the process. However, the VOD server and/or the set-top box may operate in real-time. VOD server limitations on how frequently ECMs can be multiplexed into the VOD stream to the set-top box set a first upper limit on key change frequency. The rate at which the set-top box can switch encryption keys as a part of its decryption process sets a second upper limit. The maximum rate at which cryptographic keys can be changed is determined by the smaller of these two upper limits.

[0034] The inventive technique for implementing rapid cryptographic key changes uses any of four different basic schemes (techniques, methodologies) for breaking up (subdividing, segmenting, sectioning) the content to be encoded/encrypted into a plurality of "crypto periods", covered by different cryptographic keys. These are:

1) Fixed crypto period: Define a crypto time interval and change the key each time the crypto time-interval passes.

2) Fixed number of packets: Determine a number of content packets "n" corresponding to a suitable time interval and change the cryptographic key every "n" packets.

3) Fixed "marker" count: Using a suitable MPEG-II field type as a "marker", such as an I-frame header, change the cryptographic key every time "n" markers have passed in the stream, where "n" is selected to produce a suitable crypto period. The MPEG-II I-frame header is one example of a suitable "marker." Alternatively, any other suitable, recurring MPEG-II encoding element could be used as a stream "marker" to delimit segments of the MPEG-II stream.

4) Random crypto period: Change the crypto-period randomly within upper and lower constraints on crypto period, using a pseudo-random algorithm. Calculate a number of packets for each crypto period and change the key after that number of packets. Generate an index file indicating at which packet numbers the encryption key should be changed. It is noted that instead of using a packet count to define each crypto period, a time interval could be used.

[0035] For the sake of ensuring clarity of the terminology used herein, to "encode" does not necessarily mean to "encrypt." All encryption is encoding, of a sort. The conversion to MPEG is an encoding process. The process of securing with cryptographic keys is encryption. Both encoding and encryption are performed on the program content. The data stream which is segmented, and for which keys are changing, is essentially the encrypted (e.g., by 116) data stream, which has previously been encoded (e.g., by 114).

[0036] **Figures 2A-2D** illustrate these four schemes for breaking up the content to be encoded/encrypted into a plurality "crypto periods."

Fixed crypto period

[0037] Under this encoding/encryption scheme, the encoding system picks (selects) a suitable time interval (crypto period) consistent with the known performance limitations of elements of the VOD delivery and playback infrastructure. Assuming MPEG-II encoding, the encoding/encryption system can use the PCR (Program Clock Reference) headers embedded in an MPEG-II stream of an item of program content to determine the exact amount of program time that has passed at any point in the stream. An initial cryptographic key is generated and encryption of the stream begins with the initial key. When analysis of the PCR information in the MPEG-II stream indicates that the crypto period has passed, a new key is generated and encryption resumes at the next MPEG-II "packet" using the new key. This new key is used until the PCR information once again indicates that the crypto period has passed since the key was changed, and the process repeats until the end of the stream, generating a new encryption key for each subsequent segment of the stream equivalent to a crypto period of program time. Each encryption key is saved for encoding into a set of ECMs for the encoded/encrypted program content. Alternatively, ECMs may be generated and saved as content is encrypted. The ECM set is provided to the VOD reseller (e.g., MSO) upon completion of licensing/authorization of rights to the program content. An encryption record (ER) is also generated, describing the scheme by which the program content was encoded/encrypted and the number of associated ECMs. In another approach, the ERS can take the original ECM set and "retrofit" (i.e., modify) it for each VOD reseller's conditional access system. The ER conveys information which permits such an implementation.

[0038] **Figure 2A** illustrates this fixed crypto period encoding scheme 200a, showing an encoded/encrypted MPEG-II stream 202, divided into a plurality of segments 204. Each segment corresponds to a series of MPEG-II packets covered by a crypto period defined by a fixed time interval ΔT. Each segment 204 is encrypted according to a different encryption key, used to generate an ECM 206 associated with each separately encrypted crypto period. The ECMs 206 are maintained separately from the encrypted MPEG-II stream 202.

Fixed number of packets

[0039] In a manner similar to that of the fixed crypto period scheme, this scheme initially determines a suitable crypto period. However, unlike the fixed crypto period scheme, the "fixed number of packets" scheme then examines the encoding of the MPEG-II stream for an item of program content to determine a suitable number "n" of MPEG-II packets which correspond to a crypto period. A sufficient extra number of packets in "n" is allowed to account for any variability inherent to MPEG-II encoding and to ensure that no MPEG-II segment of "n" packets will exceed the VOD distribution/playback system's key processing capabilities. It should be appreciated that the streaming content being encrypted comprises a sequence of packets. An initial key is chosen, and encryption of the MPEG-II stream begins, changing the key after each "n" MPEG-II packets in the stream. The number of packets per time interval can vary dramatically.

[0040] As in the fixed crypto period scheme, the encryption key for each "n" packets is saved for encoding into a set of ECMs (ECM1, ECM2 ...) for the encoded/encrypted program content. Alternatively, as noted above, the ECMs can be generated (e.g., in real time) and saved as content is encrypted. The ECM set is provided to the VOD reseller (e.g., MSO) upon completion of licensing/authorization of rights to the program content, and an encryption record (ER) is also generated, describing the scheme by which the program content was encoded/encrypted and the number of associated ECMs. As previously indicated, the ERS could take the original ECM set and modify it for each VOD reseller's CAS.

[0041] **Figure 2B** illustrates an encoding scheme 200b that uses a fixed number of MPEG-II packets per crypto interval. An encoded/encrypted MPEG-II stream 202 is divided into segments of "n" MPEG-II packets each, where "n" represents the number of MPEG-II packets that correspond to a suitable period of program time to be used as a crypto period. Each segment 204 is encrypted according to a different encryption key, used to generate an ECM 206 associated with each separately encrypted crypto period. The ECMs 206 are maintained separately from the encrypted MPEG-II stream 202.

Fixed "marker" count

[0042] In this scheme, using a suitable, recurring element of MPEG-II encoding as a stream segment delimiter (marker), a number "n" of stream segments is determined that correspond to a suitable crypto period. The stream is then encrypted in "crypto segments" defined by "n" markers. The markers can optionally be transmitted "in the clear", (i.e., unencrypted) to facilitate decoding/decryption. Each crypto segment is encrypted using a different cryptographic key.

[0043] In a manner like that of the "fixed number of packets" scheme, the encryption key for each "n" packets is saved (or generated and then saved) for encoding into a set of ECMs for the encoded/encrypted program content. The ECM set is provided to the VOD reseller (e.g., MSO) upon completion of licensing/authorization of rights to the program content, and an encryption record (ER) is also generated, describing the scheme by which the program content was encoded/encrypted and the number of associated ECMs.

[0044] **Figure 2C** illustrates an encoding scheme 200c that uses MPEG-II "markers" 208 to delimit crypto intervals in the MPEG-II stream. An encoded/encrypted MPEG-II stream 202 is divided into a plurality of segments 204 delimited by a number "n" of "markers" 208. Each segment corresponds to a series of MPEG-II packets delimited by "n" markers 208 ("n" = 2 in the figure). A marker 208 can be any suitable recurring element of MPEG-II encoding, such as an I-frame header (shown in the Figure, as "I"). The number "n" is chosen such that each segment corresponds roughly to a suitable amount of program time for a crypto period. The markers 208 are transmitted "in the clear" to facilitate their identification during the decoding/decryption process. Each segment 204 is encrypted according to a different encryption key, used to generate an ECM 206 associated with each separately encrypted crypto period. The ECMs 206 are maintained separately from the encrypted MPEG-II stream 202.

"Random" crypto period

[0045] Using this scheme, suitable upper and lower limits are selected for crypto periods within the performance limits of the VOD distribution/playback system. The MPEG-II stream encoding scheme for the program content to be encrypted is then examined to determine a minimum number "x" and a maximum number "y" of MPEG-II packets corresponding to the selected upper and lower limits on crypto periods. Using a pseudo-random algorithm, an initial packet count "n" between "x" and "y" (inclusive) is determined. An initial key is chosen and encryption of the stream begins. When "n" stream packets have been encrypted, the pseudo-random algorithm is used to select a new value for "n." A new key is generated, and encryption resumes until the new number "n" of packets has been encrypted. The process repeats, generating a new packet count and a new key for each encrypted segment (crypto interval) of the stream. As each segment is encrypted, an index number is stored indicating the packet at which the segment begins, and the encryption key is saved (or ECMs are generated and then saved). An index file of all of the index numbers is generated and saved so that the VOD server knows when to insert ECMs. Decryption depends on packet marking and the ECMs. The index numbers can be packet numbers within the MPEG-II stream, packet counts or byte counts associated with each separately encrypted segment, or any other index number suitable for use by a multiplexing engine in determining which MPEG-II packets are associated with each separately encrypted segment.

[0046] As with the other schemes, the encryption key for each encrypted segment is saved for encoding into a set of ECMs for the encoded/encrypted program content. Alternatively, as in the other schemes discussed above, ECMs may be generated and saved as content is encrypted. The ECM set is provided to the VOD reseller (e.g., MSO) upon completion of licensing/authorization of rights to the program content, and an encryption record (ER) is also generated, describing the scheme by which the program content was encoded/encrypted and the number of associated ECMs. The ERS could alternatively take the original ECM set and "retrofit" (i.e., modify) it for each VOD reseller's conditional access system. As previously indicated, the ER conveys information which permits such an implementation.

[0047] **Figure 2D** illustrates a random crypto interval coding scheme 200d. An encoded/encrypted MPEG-II stream 202 is divided into a plurality of randomly-sized segments 204a, 204b, 204c, 204d, 204e, etc. Each segment corresponds to a number of MPEG-II packets (specific to that segment) derived using a pseudo-random sequence generator 210. A new packet count is determined for each segment, and is used in an encryption control mechanism 212, which generates a new key and index number ("A", "B", "C", "D", "E", "F", etc.) for each segment. Each segment 204'x' is encrypted according to the new encryption key, which is in turn used to generate an ECM 206a, 206b, 206c, 206d, 206e, 206f... associated with each separately encrypted segment 204a, 204b, 204c, 204d, 204e, 204f... The ECMs 206'x' are maintained separately from the encrypted MPEG-II stream 202. The index numbers ("A", "B", "C", "D", "E", "F", etc.) are stored in an index file 214, which is used to facilitate decoding by identifying which packets are associated with each ECM.

[0048] Although shown and described hereinabove in terms of MPEG-II encoding, it should be understood by those of ordinary skill in the art that the inventive technique is readily adaptable to other forms of encoding by making the necessary adaptations, substituting features of alternate video timing/synchronization and encoding elements as appropriate. Moreover, the various techniques described above can be combined. It is fully within the spirit and scope of the invention to do so, and the description presented hereinabove is intended to be illustrative, rather than limiting.

[0049] Information about the encoding of the program content (encoded/encrypted as described hereinabove) is conveyed to an ERS (ECM Renewal System) in a few separate pieces. First, the encoded/encrypted program content is maintained separately from any information about its encoding and encryption. Second, all of the encryption keys used to encrypt the program content (or the ECMs) are stored in the sequential order that they were used to encrypt the program content. Third, an encryption record defining the encoding/encryption method and other relevant encoding/encryption parameters is generated, stored and associated with the program content. In the case of random crypto period encoding, the associated index file (214) is also stored. Other implementations are also possible.

[0050] When a VOD "reseller", (e.g., MSO, satellite TV operator) successfully executes a licensing agreement for an item of program content via an ERS system, the ERS system transmits the encoded/encrypted program content to the reseller via any suitable (e.g., broadband) distribution means (e.g., satellite, cable or Internet link) for storage on a VOD server at the reseller's location. A set of ECMs (one-to-one correspondence with encryption keys, in sequential order, or a known permutation thereof) is generated and transmitted to the reseller, and the encryption record (ER) (or an encoded equivalent thereof) defining essential portions of the encryption/encoding technique used to encode/encrypt the program content is transmitted to the reseller. In the case of random crypto period encoding, the index file (214) is also transmitted, either separately or embedded into one or more of the other items (e.g., ECMs, ER) which are transmitted to the reseller.

[0051] It should be appreciated that content does not have to go through the ERS, although it can. The ER could be sent by the VOD reseller (e.g., MSO) to the ERS to identify the content. The ERS can then generate the ECMs and send them back.

[0052] For example, for the fixed methods of encoding (fixed period, fixed packet count, or fixed marker count), a field can be added to the ER defining the number of ECMs associated with the encryption/encoding process. For fixed marker count encoding, an ER field or fields defining the marker type can also be added where different types of markers might be employed in encoding/encrypting different program content. For (pseudo) random crypto period encoding, an ER field indicating the number of ECMs and an ER field, or set of ER fields containing the entire index file (or encoded equivalent thereof), can be added.

[0053] Referring again briefly to **Figure 1**, the VOD server 134 receives and stores the program content. The VOD server 134 also receives the set of ECMs and ER associated with the program content from the system manager 136. When an end-user requests and receives authorization to receive a VOD transmission of the program content (e.g., via the end-user's set-top box 150 and the user authorization system 138), the VOD server multiplexes the ECMs and the encoded/encrypted program content as defined by the ER into a stream and transmits it to the end user's VOD playback device (e.g., set-top box 150). The user authorization system generates an EMM containing license authorizations and additional information (key, encoding parameters, etc.) necessary to decode/decrypt the program content and transmits it to the user's VOD playback device 150. The VOD playback device 150 then has all of the elements necessary to decode, decrypt and display the program content, and proceeds to do so.

[0054] Alternatively, the ECM Renewal System (ERS) can generate a "template" of one set of ECMs that can be used to decrypt the program content. In this case, the VOD server would be instructed how to take this template and make corresponding ECMs for each crypto period. The method used to determine when to move from one encoded/encrypted segment to the next would be substantially the same as described hereinabove.

[0055] For (pseudo) random crypto period encoding, the VOD server retains the index file from the encryption record, and uses it to determine at what point in the stream to insert (multiplex in) the next ECM according to the index numbers stored therein.

[0056] Although the invention has been described in connection with various specific embodiments, those skilled in the art will appreciate that numerous adaptations and modifications may be made thereto without departing from the spirit and scope of the invention as set forth in the claims.

Claims

1. A method for pre-encrypting material with a cryptographic key comprising:

   encoding the material into digital form, encrypting the resulting encoded material, and transmitting the encrypted material over a transmission medium; and
   in the process of encrypting the encoded material, segmenting the encoded material into a plurality of segments, and using different cryptographic keys for different segments.

2. The method of claim 1, further comprising:

   controlling the encrypting of the material by selecting permission of encryption only if valid licensing exists.

3. The method of claim 1 or claim 2 wherein:

   the transmission medium is at least one of a satellite, the Internet, an intranet, and cable network.

4. A method of communication of encrypted information which includes the method of claim 1, claim 2 or claim 3 and further includes:

   at a receiver, receiving the encrypted material and storing it in a server from which it can be re-transmitted to end-users; and
   at the receiver, controlling operation of a reseller's various transmission and communications resources.

5. The method of claim 4, further comprising:

   generating a personalized encryption for each of a plurality of destination content resellers.

6. The method of claim 4 or claim 5, wherein:

   at least one of the resellers is a Multiple System Operator (MSO).

7. The method of any one preceding claim, wherein:

   the material comprises at least one of movies, data, audio, and other program content.

8. The method of any one preceding claim, wherein:

   the encoding is according to the MPEG-II standard.

9. The method of any one preceding claim, further comprising:

   generating an Entitlement Control Message (ECM) comprising encryption information specific to the material which, in combination with a valid Entitlement Management Message (EMM), can be used to derive an encryption key for decoding the encrypted material; and
   multiplexing the ECM within the encrypted material.

10. The method of claim 1, wherein,
   the material comprises video on demand (VOD) content.

11. A method according to claim 1 for encrypting streaming content, the method comprising:

   changing cryptographic keys at a plurality of points within the content.

12. The method of claim 11, further comprising:

   determining a number of sets of cryptographic keys that should be employed to encrypt one item of content, and determining an upper limit of how frequently keys can be changed.

13. The method of claim 11 or claim 12, further comprising:

   determining how and where, within the content, to effect the cryptographic key changes.

14. The method of claim 11, claim 12 or claim 13 and further comprising:

IPR2022-01227
EXHIBIT 1003 - PAGE 00849

determining how to communicate the cryptographic key sets to VOD servers.

15. The method of any one of claims 11 to 14, further comprising:

determining how to synchronize cryptographic key changes with corresponding entitlement control messages (ECMs) at a time when content is streamed to a consumer.

16. The method of any one of claims 11 to 15, further comprising:

determining how to handle an ECM renewal process.

17. The method of any one of claims 11 to 16, further comprising:

defining a cryptographic key change methodology that permits rapid key changes with straightforward, simple key change synchronization at the time of decryption.

18. The method of any one of claims 11 to 17, further comprising:

maintaining all cryptographic keys separate from the encrypted content.

19. The method of claims 11 to 18, further comprising:

multiplexing a set of Entitlement Control Messages (ECMs) conveying information about a set of keys into the stream when delivering the encrypted content.

20. The method of any one of claims 11 to 19, further comprising:

delivering a separate Entitlement Management Message (EMM) from an authorization system.

21. A method according to any one preceding claim including encrypting streaming content, comprising:

segmenting the content into a plurality of crypto periods; and
encrypting the content for each crypto period with a different cryptographic key.

22. The method of claim 21, wherein:

the crypto period comprises a fixed time interval; and
further comprising:

changing the cryptographic key each time the time interval passes.

23. The method of claim 22, further comprising:

generating an Entitlement Control Message (ECM) for each separately encrypted crypto period; and
maintaining the ECMs separately from the encrypted streaming content.

24. The method of claim 21, further comprising:

using Program Clock Reference (PCR) headers embedded in an MPEG-II stream of an item of program content to determine the amount of time that has passed at any point in the stream.

25. The method of claim 24, further comprising:

beginning the encryption by generating an initial cryptographic key, and when analysis of the PCR information in the MPEG-II stream indicates that the crypto period has passed, generating a new key and resuming encryption at the next MPEG-II packet using the new key;
using the new key until the PCR information once again indicates that the crypto period has passed since the key was changed; and
repeating the process until the end of the stream

26. The method of claim 21, wherein:

the streaming content is embodied in a sequence of packets; and
the crypto period comprises a fixed number of content packets.

27. The method of claim 26, further comprising:

allowing a sufficient number of packets to account for any variability inherent to the encoding procedure and to ensure that no segment of "n" packets will exceed a distribution/playback system's key processing capabilities.

28. The method of claim 26, further comprising:

choosing an initial key and beginning encryption of the streaming content; and
changing the key after each fixed number of packets in the streaming content.

29. The method of claim 26, further comprising:

saving information about the encryption key for each of a plurality of content packets into a set

IPR2022-01227
EXHIBIT 1003 - PAGE 00850