A DEVICE FOR SYNCHRONISING DIGITAL DATA STREAMS IN REAL-TIME

Technical Field

The present invention relates to a device and method for synchronising signals in real-time, and in particular, digital audio and video (A/V) signals generated by multiple encoders such as those used in video-on-demand (VOD) services.

Background of the Invention

Video-on-demand services typically involve the streaming of a stored group of data streams including a video stream, an audio stream(s), a script stream(s), and auxiliary text streams, across the Internet from a server to a client in response to the client's request.

Recently, with the emergence of wideband networks such as the digital subscriber loop (DSL), cable modems and the like, the usage of VOD services providing high quality audio and video data transmission has increased. As such, a perceived need has developed for the VOD framework to support multiple A/V stream simultaneous playback wherein multiple A/V data streams representing different aspects of a common recording session are transmitted from the VOD server to the client.

For instance, students may seek to view a pre-recorded university lecture. In order to facilitate this situation, two video streams may be generated from two separate cameras wherein a first camera captures a lecturer delivering a lesson, whilst a second camera captures lecture notes that are projected on to an overhead screen, and a microphone captures the audio component of the recording session. The captured video streams are each encoded with the captured audio stream into data encoding formats such as MPEG, Windows Media™, and Real Audio/Video™, which enables students to access multimedia educational resources on-line.

Another application of multiple A/V streams simultaneous playback arises where multiple cameras are located in a court room to record different aspects of a court session. For instance, it may be useful to concurrently capture the expressions and/or reactions of various court room participants such as lawyers, witnesses, jurors, defendants and the like. At a later time, a judge or any other interested party can review the captured audio-visual streams from the same court session by virtue of a VOD service capable of simultaneous playback of multiple audio-visual streams.

However, it is perceived that when multiple A/V streams are simultaneously played back using prior art arrangements, they have tended to suffer from synchronisation problems. One factor which contributes to this problem is that various encoders used in prior systems experience different latency characteristics due to unique initialisation periods, library load periods and the like. Consequently, pictures that have been captured by separate cameras at different times are incorrectly rendered on a client's screen at the same time. The same synchronisation problem affects the simultaneous playback of audio streams over the prior art. Clearly, this problem is undesirable to a user of a prior art VOD service as the user's perception of information relayed by the prior art is adversely affected.

Prior art inventions have attempted to rectify this problem by employing high precision system clocks to synchronise components in the VOD system with hardware clock synchronisers located in separate cameras and encoder servers in the system. However, this approach has failed to alleviate the problem because the recorded data streams must still be manually edited using professional video and audio editing software on a laborious frame-by-frame and sample-by-sample basis respectively. Moreover, this is an expensive and time-consuming exercise which tends to exacerbate the latency period prior to the A/V data streams being available for distribution via the prior art VOD network.

Summary of the Invention

The present invention seeks to provide a device for, and a method of, synchronising encoded audio-visual data streams in real-time.

In a first broad form, the present invention provides a device for synchronising a plurality of encoded data streams:

wherein each encoded data stream includes a plurality of video data elements having a video timing information, and, a plurality of audio data elements having an audio timing information;

the said device including:

a means of selecting a set of video data elements from each encoded data stream, wherein the video timing information of the video data elements in each set of selected video data elements satisfy a video timing criteria;

a means of selecting a set of audio data elements from each encoded data stream, wherein the audio timing information of the audio data elements in each set of selected audio data elements satisfy an audio timing criteria;

a means of modifying each set of selected video data elements, wherein a corresponding modified set of video data elements is output;

a means of modifying each set of selected audio data elements, wherein a corresponding modified set of audio data elements is output;

wherein, the modified sets of video data elements and their corresponding modified sets of audio data elements form a plurality of modified encoded data streams, whereby said plurality of modified encoded data streams are synchronised.

Preferably, the video timing information of the video data elements in each encoded data stream includes video presentation time stamps representing the sequential order in which each video data element is to be processed for display in the respective encoded data streams.

Preferably, the audio timing information of the audio data elements in each encoded data stream includes audio presentation time stamps representing the sequential order in which each audio data element is to be played in the respective encoded data streams.

Preferably, the video timing information of the video data elements in each encoded data stream includes video decoder time stamps representing the sequential order in which each

video data element is to be decoded in the respective encoded data streams.

Preferably, the audio timing information of audio data elements in each encoded data stream includes audio decoder time stamps representing the sequential order in which each audio data element is to be decoded in the respective encoded data streams.

Preferably, the video timing criteria includes determining whether the presentation time stamps of video data elements in each encoded data stream, are equal to or greater than the presentation time stamp of a reference video data element in the respective encoded data streams.

Preferably, the means of determining the reference video data element includes:

a means of calculating a stream editing time interval;

whereby, the reference video data element includes a video data element in each encoded data stream wherein the sum of the reference video data element's presentation time stamp and its presentation time stamp duration is equal to or greater than the stream editing time interval.

Preferably, the reference video data element of each encoded data stream includes an I-frame, said I-frame containing information to construct an image without reference to other video data elements in the respective encoded data streams.

Preferably, the video presentation time stamp duration includes a time difference between consecutive video presentation time stamps.

Preferably, the means of calculating a stream editing time interval for each encoded data stream includes:

- a means of detecting a local encoding start time for each of the encoded data streams, said local encoding start time being the actual time at which an encoder server commenced generating the encoded data stream;
- a means of registering the local encoding start time of each of the encoded

data streams wherein a local encoding start time register time is measured for each encoded data stream;

- a means of determining a synchronisation time, said synchronisation time being equal to or greater than all of the local encoding start time register times measured;

whereby, the stream editing time interval of each encoded data stream includes the difference between the synchronisation time, and, the local encoding start time of the respective encoded data streams.

Preferably, video data elements in each encoded data stream having a presentation time stamp less than the presentation time stamp of the reference video data element in the respective encoded data streams, are discarded.

Preferably, the audio timing criteria includes determining whether the presentation time stamps of audio data elements in each encoded data stream are greater than or equal to a presentation time stamp of a reference audio data element in the respective encoded data streams.

Preferably the present invention includes a means of determining a reference audio data element in each encoded data stream, said reference audio data element including an audio data element wherein the sum of the reference audio data element's presentation time stamp and its audio presentation time stamp duration is equal to or greater than the stream editing time interval of the respective encoded data streams.

Preferably, the audio presentation time stamp duration includes a time difference between consecutive audio presentation time stamps.

Preferably, audio data elements in each encoded data stream having a presentation time stamp less than the presentation time stamp of the reference audio data element in the respective encoded data stream, are discarded.

Preferably, the means of modifying the video data elements in each set of selected video data elements includes modifying the video timing information of the video data elements in each set of selected video data elements.

Preferably, the means of modifying the video timing information of video data elements in each set of selected video data elements includes:

- a means of comparing the audio presentation time stamp of the reference audio data element in each encoded data stream with the video presentation time stamp of the reference video data element in the encoded data stream, wherein the smaller of the reference audio data element presentation time stamp and the reference video data element presentation time stamp, is designated as a starting presentation time stamp in the respective encoded data stream;
- a means of setting the starting presentation time stamp of each encoded data stream to a starting time value;
- a means of replacing the presentation time stamps of the video data elements in each set of selected video data elements with the difference between the video presentation time stamps and the starting presentation time stamp.

Preferably, the starting time value includes a time value of 0 relative to a common reference clock.

Preferably, the means of modifying the video timing information in each set of selected video data elements includes a means of modifying the video decoder time stamp of each of the video data elements in each set of video data elements.

Preferably, the means of modifying the audio data elements in each set of selected audio data elements includes modifying the audio timing information of the audio data elements in each set of selected audio data elements.

Preferably, the means of modifying the audio timing information of the audio data elements in each set of selected audio data elements includes replacing the audio presentation time stamps of the audio data elements with the difference between the audio presentation time stamps and the value of the reference audio presentation time stamp of the respective encoded data streams.

Preferably, the means of modifying the audio timing information of the audio data elements in each set of selected audio data elements includes a means of modifying the audio decoder time stamps of the audio data elements in each set of selected audio data elements.

Preferably, an encoded data stream in the plurality of encoded data streams is generated by an encoder server, the encoder server including:

- a video data stream encoder for generating video data elements in the encoded data stream;
- an audio data stream encoder for generating audio data elements in the encoded data stream;
- an encoder multiplexer for multiplexing the generated video data elements and audio data elements;
- an encoder memory store;
- an encoder processor; and
- an encoder clock.

Also preferably, the encoder memory store includes a first encoder memory buffer for storing an encoded data stream.

Also preferably, the encoder memory store includes a second encoder memory buffer for storing a modified encoded data stream.

Also preferably, the encoder server includes a means of applying the video timing criteria to the video data elements of an encoded data stream.

Also preferably, the encoder server includes a means of modifying the video timing information of the video data elements in a set of selected video data elements.

Also preferably, the encoder server includes a means of applying the audio timing criteria to the audio data elements of an encoded data stream.

Also preferably, the encoder server includes a means of modifying the audio timing information of the audio data elements in a set of selected audio data elements.

Preferably, the present invention includes a system resource server, said system resource server including:

- a means of determining the synchronisation time;
- a means of registering the local encoding start times of each encoded data stream in the plurality of encoded data streams;
- a means of communicating the synchronisation time to the encoder server;
- a common reference time clock; and
- a means of synchronising the common reference time clock with the encoder server clock.

More preferably, the means of synchronising the common reference time clock with the encoder clock includes a means of implementing an SNTP protocol.

More preferably, the means of communicating the synchronisation time to the encoder server includes communicating the synchronisation time after a synchronisation time query is received from the encoder server.

Also preferably, the present invention includes a means of providing the encoder server with a recording schedule, said recording schedule including:

- an encoding start time for which the encoder server is to commence generating an encoded data stream;
- an encoding end time for which the encoder server is to cease generating the

encoded data stream.

More preferably, the recording schedule is provided by the system resource server.

Preferably, the present invention includes a data server for archiving the modified encoded data streams.

Preferably, the present invention includes a video server for storing the modified of encoded data streams.

Preferably, the video server includes a cache memory buffer for storing the modified sets of encoded data streams.

Preferably, the encoder clock is synchronised with the common time reference.

Preferably, the video timing information of the encoded data streams is generated by reference to the encoder clock.

Preferably, the audio timing information of audio data elements in each encoded data stream is generated by reference to an encoder clock.

Alternatively, in the first broad form of the present invention, the means of modifying the video data elements in each set of selected video data elements includes:

- a decoder server for decoding the video data elements in each set of selected video data elements, wherein decoded video data streams are output; and
- an encoder server for re-encoding the decoded video data streams;

whereby modified sets of video data elements are generated.

Also alternatively, in the first broad form of the present invention, the means of modifying the audio data elements in each set of selected audio data elements includes:

- a decoder server for decoding the audio data elements in each set of selected

audio data elements, wherein decoded audio data streams are output; and

- an encoder server for re-encoding the decoded audio data streams;

whereby the modified sets of audio data elements are generated.

In a second broad form, the present invention provides a device for playing and/or rendering a plurality of modified encoded data streams produced in accordance with the first broad form of the present invention.

Preferably, the second broad form of the present invention includes:

- a demultiplexer for demultiplexing the audio data elements and video data elements in each modified encoded data stream;
- an audio decoder for decoding the demultiplexed audio data elements in the modified encoded data streams;
- a video decoder for decoding the demultiplexed video data elements in the modified encoded data streams;
- an audio player for playing the decoded audio data elements;
- a video renderer for rendering the decoded video data elements.

More preferably, the second broad form of the present invention includes a means of synchronising the audio decoder, video decoder, demultiplexer, audio player and video renderer with the common reference time referred to in the first broad form of the present invention.

In a third broad form, the present invention provides a combination including the first broad form and the second broad form of the present invention.

In a fourth broad form, the present invention provides a method of synchronising a plurality of encoded data streams:

wherein each encoded data stream includes a plurality of video data elements having a video timing information, and, a plurality of audio data elements having an audio timing information;

said method including the steps of:

(i) selecting a set of video data elements from each encoded data stream, wherein the video timing information of the video data elements in each set of selected video data elements satisfy a video timing criteria;

(ii) thereafter, selecting a set of audio data elements from each encoded data stream, wherein the audio timing information of the audio data elements in each set of selected audio data elements satisfy an audio timing criteria;

(iii) thereafter, modifying each set of selected video data elements, wherein a corresponding modified set of video data elements is output;

(iv) thereafter, modifying each set of selected audio data elements, wherein a corresponding modified set of audio data elements is output;

wherein, the modified sets of video data elements and their corresponding modified sets of audio data elements form a plurality of modified encoded data streams, whereby said plurality of modified encoded data streams are synchronised.

In a fifth broad form, the present invention provides a method of playing back a plurality of modified encoded data streams produced in accordance with the third broad form of the present invention.

Preferably, the fifth broad form includes the steps of:

- demultiplexing the audio data elements and video data elements in each modified encoded data stream;
- decoding the demultiplexed audio data elements to provide an audio data stream;
- decoding the demultiplexed video data elements to provide a video data stream;
- playing the audio data stream using an audio player;
- rendering the video data stream using a video renderer.

Also preferably, the fifth broad form includes the step of synchronising the audio decoder, video decoder, demultiplexer, audio player and video renderer with a common reference

time used to generate the plurality encoded data streams.

In a sixth broad form, the present invention provides a combination including the fourth broad form and the fifth broad form of the present invention.

In a seventh broad form, the present invention provides a plurality of modified data streams produced in accordance with the first broad form of the present invention.

In an eighth broad form, the present invention provides a plurality of modified data streams produced in accordance with the fourth broad form of the present invention.

Brief Description of the Drawings

The present invention will become more fully understood from the following detailed description of a preferred but non-limiting embodiment thereof, described in connection with the accompanying drawings, wherein:

- Figure 1 shows a block diagram of a first embodiment of the present invention implemented using a video-on-demand framework.

- Figure 2 shows a time-sequence diagram of the first embodiment of the present invention including the timing sequences for synchronisation and playback of encoded data streams.

- Figure 3 depicts four unsynchronised encoded data streams to be processed by the first embodiment of the present invention..

- Figure 4 shows the structure of a data packet used by the SNTP protocol in a first embodiment of the present invention.

- Figure 5 shows an example of an XML-coded message containing

information relating to the local encoding start time of an encoder server used in the first embodiment of the present invention.

- Figure 6 shows an example of an XML-formatted message which is used to request a synchronisation-time, and, for replying to the request for a synchronisation-time in the first embodiment of the present invention.

- Figure 7 shows a block diagram of a recording session in accordance with the first embodiment of the present invention.

- Figure 8 shows a flow-chart of a first data stream synchronisation algorithm used in the first embodiment of the present invention.

- Figure 9 illustrates the implementation of the first data stream synchronisation algorithm used in the first embodiment of the present invention in which frames in the multiple encoded data streams are selectively discarded or modified in order to provide synchronisation between the data streams.

- Figure 10 shows a flow-chart of a second data stream synchronisation algorithm used in a second embodiment of the present invention.

- Figure 11 illustrates the implementation of a second embodiment of the present invention in which frames in the multiple encoded data streams are selectively discarded or re-encoded in order to provide synchronisation between the data streams.

- Figure 12 shows a block diagram illustrating a first embodiment of circuitry used to synchronously playback multiple modified encoded data streams.

Best Modes for Carrying Out the Invention

Figure 1 shows a block diagram of a first embodiment (1) of the present invention including first and second encoder servers (2,3) for encoding audio-video (A/V) data streams captured from different sources in a common recording session in real-time, a session resource server (SRS) (4) for coordinating the recording session, a data server (5) for archiving synchronised A/V data streams received from the first and second encoder servers (2,3) into groups of data streams in accordance with a grouping policy, a video server (6) for caching groups of synchronised A/V data streams received from the data server, a client server (7) which is capable of accessing cached groups of synchronised A/V data streams stored on the video server via an on-demand service, a decoder server (8) disposed at the client server (7) for decoding the encoded data streams, a sound card (36) and a video player (37) for playing and rendering decoded audio and video data respectively. It would be understood by a person skilled in the art that the implementation of the servers used in the present embodiment includes any suitable software and/or hardware, or combination thereof.

The first and second encoder servers (2,3), the data server (5), the video server (6), the SRS (4) and the client server (7) are interconnected via a local area network (LAN) (9). However, in alternative embodiments of the present invention, the servers may be interconnected using any other suitable communication network including a WAN, the Internet or the like.

The SRS (4) includes a reference clock (39) implemented using a cesium-based clock. The SRS (4) also includes a Simple Network Time Protocol (SNTP) server (10) which synchronises the system clocks (38) located in each of the first and second encoders (2,3), the data server (5), the video server (6), and the client server (7), with the reference clock in accordance with the SNTP protocol so as to ensure that the servers are operating in relation to a common time reference. Figure 4 shows an example of an SNTP packet used in the first embodiment to facilitate synchronisation between the reference clock of the

SRS, and, the system clocks of the encoder servers, data server, video server, and the client server.

The servers used in the first embodiment (1) each include a messaging interface (11) in order to facilitate inter-server communication. The messaging interfaces (11) each include a message parser for interpreting received messages. Messages are coded in accordance with the Extensive Markup Language (XML) standard however any other suitable coding language may be used in alternative embodiments.

In the first embodiment, the first and second encoder servers (2,3) include MPEG encoders (12) for encoding their respective input A/V signals in accordance with MPEG standard formats. The MPEG encoders (12) each include an audio encoder and a video encoder for encoding a captured audio data stream and a captured video data stream respectively. By way of example, the first and second encoder servers are fed an audio data stream which has been captured from a common audio source – for instance, a single microphone (13) mounted in a lecture theatre. Also by way of example, the first and second encoder servers are fed a first and a second video data stream respectively which have been captured using separate digital video cameras (14,14') mounted at different positions in the same lecture theatre.

The MPEG audio encoder (12) encodes the captured audio data stream into MPEG-formatted audio frames. The MPEG video encoder encodes the captured video data stream into MPEG-formatted video frames. The stream of MPEG-formatted audio frames are multiplexed with the stream MPEG-formatted video frames to form a stream of MPEG-formatted A/V data frames.

When encoding incoming captured audio and video data, the MPEG encoders (12) embed a presentation time stamp (PTS) into each frame that is generated – wherein, the PTS of a frame represents a time relative to the reference clock time, at which the encoded frame is to be played or rendered. Similarly, the MPEG encoders (12) embed a decoder time stamp (DTS) into each frame that is generated – wherein, the DTS of a frame represents a time,

relative to the common reference clock time, at which the encoded frame is to be decoded.

It would be understood by a person skilled in the art that in alternative embodiments of the present invention, that the first and second encoder servers (2,3) may encode the incoming A/V data streams into other suitable encoding formats such as RealAudio™, Windows Media™, or any combination of suitable encoding formats.

The SRS (4) includes a schedule server (15) which transmits a common recording schedule to the first and second encoder servers (2,3) via its messaging interface (11). The recording schedule includes information defining an encoding start time and an encoding end time relative to the common time reference, at which the first and second encoder servers (2,3) are to commence encoding the incoming captured data streams (18) and cease encoding the incoming captured data streams (18) respectively.

The SRS (4) further includes a database (16) for storing, amongst other things, information relating to the number of encoder servers (2,3), data servers (5) and video servers (6) connected to the LAN at any given time (9), the activity status of the servers, and performance characteristics of the servers. The SRS (4) periodically updates information stored in the database(16), for instance, by monitoring and/or polling the servers to determine their status. Alternatively, the information stored in the database (16) may be updated manually by a network administrator.

In accordance with the timing sequence shown in Fig. 2, a schedule manager program (17) in the first and second encoder servers (2,3) receives the recording schedule transmitted by the SRS (4) and sets timers in the respective encoder servers (2,3) to automatically commence encoding and cease encoding incoming captured data at the times designated in the received recording schedule.

More specifically, the schedule manager programs (17) detect whether the encoder server system clock (38) times correspond with the common encoding start time designated in the recording schedule. The encoder servers (2,3) commence encoding the incoming A/V data

when this condition is met. Similarly, the schedule manager programs (17) detect when the encoder server system clock (38) times correspond with the common end encoding time contained in the received recording schedule, and cause the encoder servers (2,3) to cease encoding incoming captured data streams (18) accordingly.

At the same time that the encoder servers (2,3) are encoding the incoming captured data streams (18), the encoded A/V streams generated by the encoder servers (2,3) are written in real-time to temporary editing files (19) stored on the respective encoder servers (2,3).

It would be understood by a person skilled in the art that the *actual time* at which each encoder server (2,3) begins encoding incoming captured A/V data streams (18) will differ in view of the unique latency characteristics, such as initialisation delays, load library delays and the like, associated with each of the encoder servers (2,3). Thus, each encoder servers (2,3) typically commences encoding at a different local encoding start time and this value is also stored in the temporary editing files (19) located on the respective encoder servers (2,3). By way of example, Fig. 3 shows four unsynchronised encoded A/V data streams, their respective local encoding start times (21), and, their respective latency periods (20) between the scheduled encoding start time (51) and the actual local encoding start times (21).

In accordance with the time sequence shown in Fig. 2, the encoder servers (2,3) each transmit a message to the SRS (4) stating their respective local encoding start times (21). An example format of such a message sent from the encoder servers to the SRS is shown in Fig. 5.

The SRS (4) registers the local encoding start times received from the encoder servers (2,3) in the SRS database (16). Also, the SRS (4) records the time at which each local encoding start time (21) is received by the SRS (4) – that is, a register time of each local encoding start time.

The SRS (4) transmits a message to the encoder servers (2,3) once all of the encoder

servers (2,3) in the network have registered their local encoding start times (21) with the SRS (4). The SRS (4) determines when all encoder servers (2,3) have registered their local encoding start times because it is aware of the number of encoder servers (2,3) connected to the LAN (9) at any given time by virtue of the information stored in the SRS database (16).

The SRS (4) further includes a stream synchronisation program (23) which calculates a common synchronisation time (24) which is to be subsequently used by the encoder servers (2,3) in synchronising encoded A/V data streams. In the first embodiment, the synchronisation time (24) is determined to be the latest register time of the local encoder start times registered at the SRS (4). Other suitable synchronisation times may be selected in alternative embodiments of the present invention which broadly adhere to the following criteria:

(i) all encoder servers (2,3) involved in any given recording session have been started before the determined synchronisation time (24); and

(ii) the local encoding start times (21) of all of the encoder servers (2,3) have been registered at the SRS (4) before the determined synchronisation time (24).

It would be appreciated by a person skilled in the art that, where the synchronisation time (24) is set too early, then synchronisation of the encoded A/V data streams will be unsuccessful given that not all encoder servers (2,3) may have yet commenced encoding the incoming captured data (18). Conversely, where the synchronisation time (24) has been set too late, then excessive amounts of data may subsequently be discarded, as will become more readily apparent in view of the following description.

In accordance with the timing sequence shown in Fig. 2, the encoder servers (2,3) transmit a message to the SRS (4) querying the synchronisation time (24) for the present recording session. Upon receiving the synchronisation time query from the encoder servers (2,3), the

SRS (4) responds by transmitting a message containing the synchronisation time (24) to the encoder servers (2,3). Figure 6 shows an example of a message format used by the encoder servers (2,3) to facilitate a synchronisation time query to the SRS (4). The message format shown in Fig. 6 is also suitable for use by the SRS (4) in the first embodiment to communicate a reply to the synchronisation time query to the encoder servers (2,3).

Each encoder server (2,3) calculates a stream editing time segment (25) in respect of the encoded A/V data stream generated. In the first embodiment of the present invention, the stream editing time segment (25) is taken to be the difference between the synchronisation time (24) determined by the SRS (4), and, the local encoding start time (21) of the encoder server in question. The stream editing time segment (25) will be different for each encoder server due to the different local encoding start times (21) of the encoder servers (2,3). Again, by way of example, Fig. 3 illustrates the different stream editing time segments (25) for a group of four encoded A/V data streams.

The stream editing time segments (25) are calculated using the stream synchronisation programs (23) which are stored and executed on the encoder servers (2,3). The stream synchronisation program (23) is written in C+, Java or any other suitable programming language.

The stream synchronisation programs (23) of the encoder servers (2,3) are also responsible for automatically editing the encoded A/V data streams stored in the temporary editing files (19) in real-time. In the first embodiment of the present invention, the A/V data in a temporary editing file (19) is edited in accordance with a first data stream editing algorithm. Figure 9 shows a flow-chart outlining the operation of the first data stream editing algorithm upon an encoded A/V data stream as implemented in the first embodiment.

Figure 8 shows, by way of example, an encoded A/V data stream which is to be synchronised with a plurality of other encoded A/V data streams in accordance with the

first embodiment of the present invention. The encoded video and audio frames of the encoded A/V data stream are stored in the temporary editing file of an encoder server. Figure 9 also shows a set of video frames (40) selected from the video frames of the encoded data stream, which satisfy a video timing criteria, and, a set of audio frames (41) selected from the encoded data stream, which satisfy an audio timing criteria. The operation of the first data stream editing algorithm upon the video data frames and audio data frames will now be described in greater detail as follows:

(i) Data in the temporary editing file is classified as being either video frames (26), or, audio frames (27);

(ii) A first video frame (28) is located in the temporary editing file wherein the sum of the first video frame's (28) PTS and its PTS duration (29) is greater than or equal to the stream editing time segment (25). The first video frame (28) is designated as VF(f) and its PTS is designated as PTS(v). The PTS duration is determined to be the time interval between consecutive video frame PTSs;

(iii) Video frames in the temporary editing file (19) having a PTS smaller than that of PTS(v) are discarded from the temporary editing file (19). Non-discarded video frames (26) form the set of selected video frames (40);

(iv) A next sequential video I-frame (30) following the first video frame is located in the temporary editing file (19), wherein, the I-frame (30) contains all the data needed to construct an image without reference to previous and/or succeeding video frames (26). The PTS of the I-frame (30) is equal to or larger than that of PTS(v). The I-frame (30) is designated as VF(0) and its PTS is designated as PTS(i);

(v) All video frames (26) in the temporary editing file (19) having a PTS that is smaller than PTS(i) are discarded from the temporary editing file (19). Non-

discarded audio frames form the selected set of audio data elements (41);

(vi) A first audio frame (31) is located in the temporary editing file (19), wherein the sum of the PTS of the first audio frame (31) and an audio frame duration (32) is greater than or equal to the stream editing time segment (25). The first audio frame (31) is designated as AF(o) and its PTS is designated as PTS(a). The audio frame's PTS duration (32) is determined to be the time interval between consecutive audio frame PTSs;

(vii) All audio frames in the temporary editing file (19) having a PTS value of less than that of PTS(a) are discarded from the temporary editing file (19);

(viii) PTS(v) and PTS(a) are compared, wherein the smaller of PTS(v) and PTS(a) is designated as PTS(0);

(ix) The PTS value of PTS(0) is set to a reference clock time value of 0;

(x) The PTS values of each selected audio (41) and video frame (40) is replaced by the difference between its current value and PTS(0);

The DTS of each selected video and audio frames (40, 41) in the temporary editing file (19) is replaced in a similar manner as outlined above with respect to PTS values.

In the first embodiment, as the encoded A/V data streams in the temporary editing files (19) are edited by the stream synchronisation programs (23), the resulting modified A/V data streams are written in real-time to streaming memory buffers (34) in their respective encoder servers (2,3). The streaming buffers (34) are implemented using random-access memory. The contents of the streaming buffers (34) are subsequently pushed to the data server (5) by network streaming programs (33) in each of the encoder servers (2,3).

In the first embodiment, the data server (5) includes a hard-disk drive upon which groups

of synchronised A/V data streams received from the first and second encoder servers (2,3), are archived. The data server (5) archives the groups of related synchronised A/V data streams in accordance with a software-implemented grouping policy. For instance, a grouping program written in C+, Java or the like, which is stored and executed on the data server (5) marks the frames of the related synchronised A/V data streams with a unique recording session number.

The grouping policy program also stores groups of synchronised A/V data streams within a common spatial region of the hard-disk drive to reduce the time taken to retrieve the groups of synchronised data streams from the data server (5). Thus, for example, when any given A/V data stream is copied from the data server (5) to the video server (6), all A/V data streams having the same group number will also be transferred.

Also in the first embodiment, the video server (6) includes a cache server (35) having a cache memory for storing groups of synchronised A/V data streams that have been copied from the data server (5). The cache memory allows a group of synchronised A/V data streams to be accessed at a speed which is suitable for an on-demand service upon request from a client server (7).

Groups of synchronised A/V data streams are periodically uploaded from the data server (5) upon request from the video server (6), and, are stored in the video server cache memory (35) using a combination of suitable caching strategies known to persons skilled in the art. A download server disposed on the data server (5) assists in facilitating the transfer of the groups of synchronised A/V data streams from the data server (5) to the video server (6).

The caching strategies are implemented by a caching program stored on the cache server (35) which is programmed to automatically download groups of synchronised A/V data streams from the data server (5), and, discard certain groups of synchronised A/V data streams. By way of example, groups of synchronised A/V data streams are retrieved and stored in the cache memory (35) according to any one of the following factors:

(i) the frequency of client server (7) requests for any given group of synchronised A/V data streams over a given period of time;

(ii) the relative spatial locality of a group of synchronised A/V data streams relative to another group of synchronised A/V data streams in the data server (5).

Conversely, the caching program identifies groups of synchronised A/V data streams that have not been requested by the client server (7) within a given time period, and discards them from the cache server (35) to allow other groups of synchronised A/V streams to be copied from the data server (5) into the video server (6).

After the end of a recording session and archiving has taken place, the client server (7) may access a group of synchronised A/V data streams by sending an access request to the video server (6). The client access request includes details of the recording session number of a particular group of synchronised data streams which the client wishes to review. The video server (6) will cache the requested group of synchronised A/V data streams if the requested group of synchronised A/V data streams are not already cached. The video server (6) will then transmit the cached group of streams simultaneously to the client server (7) for playback by the client server (7).

Figure 12 is a diagram showing the implementation of the client server (7) which implements the simultaneous playback of the synchronised A/V data streams transmitted from the video server (6). The client server (7) includes a bank of demultiplexers (42) for demultiplexing the incoming multiple A/V data streams into component encoded audio data frames and video data frames.

The client server (7) also includes audio decoders (43) for decoding the demultiplexed encoded audio data frames, video decoders (44) for decoding the demultiplexed encoded video data frames, a client server sound card (36) for playing back the decoded audio data frames, and a video renderer (37) for rendering the decoded video data frames.

The client server (7) is synchronised with the common reference time used to generated the encoded A/V data streams, and, this ensures that the simultaneous playback of the multiple A/V data streams is synchronised. The synchronisation of the client server (7) and the common reference time may be achieved using an SNTP protocol.

In a second embodiment of the present invention, a second data stream editing algorithm is used by the stream synchronisation programs (23) of the first and second encoding servers (2,3) to modify the data streams stored in the temporary editing files (19). Figure 10 shows a flow-chart outlining the operation of the second data stream editing algorithm upon an encoded A/V data stream as used in the second embodiment.

Figure 11 shows, by way of example, an encoded A/V data stream which is to be synchronised with other encoded A/V data streams in accordance with the second stream synchronisation algorithm. The video and audio data frames of the encoded A/V data stream are stored in the temporary editing file of an encoder server.

Figure 11 also shows a set of video frames (47) selected from the video frames of the encoded data stream which satisfy a video timing criteria, and, a set of audio frames (50) selected from the encoded data stream which satisfy an audio timing criteria. The operation of the second data stream editing algorithm upon the encoded A/V data stream will now be described in greater detail as follows:

(i) Classify data in the temporary editing file (19) as either video frames (26) or audio frames (27);

(ii) Locate a first video frame VF(0) (46), wherein the sum of the first video frame's PTS and its PTS duration, is equal to or greater than the value of the stream editing time segment (25). Designate the PTS of the first video frame (46) as PTS(v);

(iii) Discard all video frames having PTS values that are less than that of PTS(v) and mark VF(o) as the first video frame in the modified stream. Video frames which are not discarded form a set of selected video frames (47);

(iv) Locate a first audio frame (48) in the temporary editing file (19) wherein the sum of the first audio frame's PTS and its PTS duration is equal to or greater than the value of the stream editing time segment (25). Designate the first audio frame (48) as AF(o) and its PTS as PTS(a);

(v) Discard all the audio data frames in the temporary editing file (19) having a PTS value that is smaller than that of PTS(a). Audio data frames that are not discarded form a set of selected audio data frames (50);

(vi) Compare PTS(v) and PTS(a), wherein the smaller of PTS(v) and PTS(a) is designated as PTS(o) and its value is set to clock reference value 0;

(vii) Modify the encoded A/V data stream in the temporary editing file (19) by decoding the original stream from PTS(v) onwards for video frames and from PTS(a) onwards for audio frames, and, re-encoding the decoded data as a modified encoded data stream. As a result of the re-encoding of the selected video frames, VF(o) is re-encoded as an I-frame;

The term "I-frame" or "key frame" in the specification refers to a type of frame used in accordance with MPEG standards which contains all the data required to recreate a complete image without reference needing to be made to preceding or successive frames in a given data stream. It would be understood by a person skilled in the art that in alternative embodiments of the present invention that other types of suitable frames may be used.
It would be understood by a person skilled in the art that throughout the specification the term "data element", "frame", "packet" and the like to refer generally to unit of data.

Those skilled in the art will appreciate that the invention described herein is susceptible to

variations and modifications other than those specifically described without departing from the scope of the invention. All such variations and modification which become apparent to persons skilled in the art, should be considered to fall within the spirit and scope of the invention as broadly hereinbefore described. It is to be understood that the invention includes all such variations and modifications. The invention also includes all of the steps and features, referred or indicated in the specification, individually or collectively, and any and all combinations of any two or more of said steps or features.

THE CLAIMS DEFINING THE INVENTION ARE AS FOLLOWS:

1. A device for synchronizing a plurality of encoded data streams:

wherein each encoded data stream includes a plurality of video data elements having a video timing information, and, a plurality of audio data elements having an audio timing information;

the said device including:

a means of selecting a set of video data elements from each encoded data stream, wherein the video timing information of the video data elements in each set of selected video data elements satisfy a video timing criteria;

a means of selecting a set of audio data elements from each encoded data stream, wherein the audio timing information of the audio data elements in each set of selected audio data elements satisfy an audio timing criteria;

a means of modifying each set of selected video data elements, wherein a corresponding modified set of video data elements is output;

a means of modifying each set of selected audio data elements, wherein a corresponding modified set of audio data elements is output;

wherein, the modified sets of video data elements and their corresponding modified sets of audio data elements form a plurality of modified encoded data streams, whereby said plurality of modified encoded data streams are synchronised;

wherein said video timing criteria includes determining whether a presentation time stamps of video data elements in each encoded data stream, are equal to or greater than the presentation time stamp of a reference video data element in the respective encoded data streams; and

wherein said reference video data element is determined from a stream editing time interval, said reference video data element includes a video data element in each encoded data stream wherein the sum of the reference video data element's presentation time stamp and its presentation time stamp duration is equal to or greater than the stream editing time interval.

2. The device as claimed in claim 1, wherein the reference video data element of each encoded data stream includes an I-frame, said I-frame containing information to construct an image without reference to other video data elements in the respective encoded data streams.

3. The device as claimed in claim 1 or 2, wherein the video presentation time stamp duration includes a time difference between consecutive video presentation time stamps.

4. The device as claimed in any of claims 1 to 3, wherein the calculation of a stream editing time interval for each encoded data stream includes:

a means of detecting a local encoding start time for each of the encoded data streams, said local encoding start time being the actual time at which an encoder server commenced generating the encoded data stream;

a means of registering the local encoding start time of each of the encoded data streams wherein a local encoding start time register time is measured for each encoded data stream;

a means of determining a synchronization time, said synchronization time being equal to or greater than all of the local encoding start time register times measured;

whereby the stream editing time interval of each encoded data stream includes the difference between the synchronization time, and, the local encoding start time of the respective encoded data streams.

5. The device as claimed in any of claims 1 to 4, wherein said video data elements in each encoded data stream having a presentation time stamp less than the presentation time stamp of the reference video data element in the respective encoded data streams, are discarded.

6. The device as claimed in any of claims 1 to 5, wherein said audio timing criteria includes determining whether the presentation time stamps of audio data elements in each encoded data stream are greater than or equal to a presentation time stamp of a reference audio data element in the respective encoded data streams.

7. The device as claimed in any of claims 1 to 6, including a means of determining a reference audio data element in each encoded data stream, said reference audio data element including an audio data element wherein the sum of the reference audio data element's presentation time stamp and its audio presentation time stamp duration is equal to or greater than the stream editing time interval of the respective encoded data streams.

8. The device as claimed in claim 7, wherein the audio presentation time stamp duration includes a time difference between consecutive audio presentation time stamps.

9. The device as claimed in any of claims 7 or 8, wherein the audio data elements in each encoded data stream having a presentation time stamp less than the presentation time stamp of the reference audio data element in the respective encoded data stream, are discarded.

10. The device as claimed in any of claims 1 to 6, wherein the means of modifying the video data elements in each set of selected video data elements includes modifying a video timing information of the video data elements in each set of selected video data elements.

11. The device as claimed in claim 10, wherein the means of modifying the video timing information of video data elements in each set of selected video data elements includes:

a means of comparing the audio presentation time stamp of the reference audio data element in each encoded data stream with the video presentation time stamp of the reference video data element in the encoded data stream, wherein the smaller of the reference audio data element presentation time stamp and the reference video data element presentation time stamp, is designated as a starting presentation time stamp in the respective encoded data stream;

a means of setting the starting presentation time stamp of each encoded data stream to a starting time value;

a means of replacing the presentation time stamps of the video data elements in each set of selected video data elements with the difference between the video presentation time stamps and the starting presentation time stamp.

12. A device for synchronizing a plurality of encoded data streams using a system resource server, said system resource server including:

a means of determining a synchronisation time;

a means of registering a local encoding start time of each encoded data stream in the plurality of encoded data streams;

a means of communicating the synchronisation time to an encoder server;

a common reference time clock; and

a means of synchronising the common reference time clock with the encoder server clock.

13. A method of synchronising a plurality of encoded data streams:

wherein each encoded data stream includes a plurality of video data elements having a video timing information, and, a plurality of audio data elements having an audio timing information;

said method including the steps of:

(i) selecting a set of video data elements from each encoded data stream, wherein the video timing information of the video data elements in each set of selected video data elements satisfy a video timing criteria;

(ii) thereafter, selecting a set of audio data elements from each encoded data stream, wherein the audio timing information of the audio data elements in each set of selected audio data elements satisfy an audio timing criteria;

(iii) thereafter, modifying each set of selected video data elements, wherein a corresponding modified set of video data elements is output;

(iv) thereafter, modifying each set of selected audio data elements, wherein a corresponding modified set of audio data elements is output;

wherein, the modified sets of video data elements and their corresponding modified sets of audio data elements form a plurality of modified encoded data streams, whereby said plurality of modified encoded data streams are synchronised;

wherein the video timing criteria includes the step of determining whether the presentation time stamps of video data elements in each encoded data stream, are equal to or greater than the presentation time stamp of a reference video data element in the respective encoded data streams.

14. The method as claimed in claim 13 wherein the step of determining the reference video data element includes the further step of:

(i) calculating a stream editing time interval;

whereby, the reference video data element includes a video data element in each encoded data stream wherein the sum of the reference video data element's presentation time stamp and its presentation time stamp duration is equal to or greater than the stream editing time interval.

15. The method as claimed in claim 13 or 14, wherein the reference video data element of each encoded data stream includes an I-frame, said I-frame containing information to construct an image without reference to other video data elements in the respective encoded data streams.

16. The method as claimed in any of claims 13 to 15, wherein the video presentation time stamp duration includes a time difference between consecutive video presentation time stamps.

17. The method as claimed in any of claims 13 to 16 wherein the step of determining a stream editing time interval for each encoded data stream includes the further steps of:

(i) detecting a local encoding start time for each of the encoded data streams, said local encoding start time being the actual time at an encoder server commences generating the encoded data stream;

(ii) thereafter registering the local encoding start time of each of the encoded data streams wherein a local encoding start time register time is measured for each encoded data stream;

(iii) thereafter, calculating a synchronisation time, said synchronisation time being equal to or greater than all of the local encoding start time register times measured;

whereby, the stream editing time interval of each encoded data stream includes the difference between the synchronisation time, and, the local encoding start time of the respective encoded data streams.

18. The method as claimed in any of claims 13 to 17 including the step of discarding video data elements in each encoded data stream having a presentation time stamp less than the presentation time stamp of the reference video data element in the respective encoded data streams.

19. The method as claimed in any of claims 13 to 18 wherein the audio timing criteria includes the step of determining whether the presentation time stamps of audio data elements in each encoded data stream are greater than or equal to a presentation time stamp of a reference audio data element in the respective encoded data streams.

20. The method as claimed in claim 19 including the step of determining a reference audio data element in each encoded data stream, said reference audio data element including an audio data element wherein the sum of the reference audio data element's presentation time stamp and its audio presentation time stamp duration is equal to or greater than the stream editing time interval of the respective encoded data streams.

21. The method as claimed in claim 20 wherein the audio presentation time stamp duration includes a time difference between consecutive audio presentation time stamps.

22. The method as claimed in any of claims 19 to 21 including the step of discarding audio data elements in each encoded data stream having a presentation time stamp less than the presentation time stamp of the reference audio data element in the respective encoded data stream.

23. The method as claimed in any of claims 13 to 22 wherein the step of modifying the video data elements in each set of selected video data elements includes the further step of modifying the video timing information of the video data elements in each set of selected video data elements.

24. The method as claimed in claim 23 wherein the step of modifying the video timing information of video data elements in each set of selected video data elements includes the further steps of:

(i) comparing the audio presentation time stamp of the reference audio data element in each encoded data stream with the video presentation time stamp of the reference video data element in the encoded data stream, wherein the smaller of the reference audio data element presentation time stamp and the reference video data element presentation time stamp, is designated as a starting presentation time stamp in the respective encoded data stream;

(ii) setting the starting presentation time stamp of each encoded data stream to a starting time value;

(iii) replacing the presentation time stamps of the video data elements in each set of selected video data elements with the difference between the video presentation time stamps and the starting presentation time stamp.