INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

Applicant:

PATENT/US2007/065391
PATENT/US2007/065387
Priority:
Publication:
Filing:
International:
International:
International:
World:

Language: English

Classification:

Publication Data:

Date: March 2006 (29.03.2006)
Number: 60/787,393

Date: 2 May 2006 (02.05.2006)
Number: 60/746,193

Date: 18 August 2006 (18.08.2006)
Number: 60/822,925

Date: 19 August 2006 (19.08.2006)
Number: 60/822,927

Date: 28 March 2007 (28.03.2007)
Number: PCT/US2007/065387

Date: 28 March 2007 (28.03.2007)
Number: PCT/US2007/065391

Inventors; and


Agent: YOUNG, Andrew, F.; Lackenbach Siegel, LLP, One Chase Road, Scarsdale, NY 10583 (US).


Designated States (unless otherwise indicated, for every kind of regional protection available): A IPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM)

Continued on next page

Abstract: The present invention provides a system, method, and apparatus for visual browsing, deep tagging, and synchronized commenting.
Declarations under Rule 4.17:
— as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(H))
— as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(Hi))
— of inventorship (Rule 4.17(iv))

Published:
— without international search report and to be republished upon receipt of that report
— the filing date of the international application is within two months from the date of expiration of the priority period
— with information concerning one or more priority claims considered void
A SYSTEM, METHOD, AND APPARATUS FOR VISUAL BROWSING,
DEEP TAGGING, AND SYNCHRONIZED COMMENTING

CROSS REFERENCE TO RELATED APPLICATIONS

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to a system, method, and apparatus for visual browsing, deep tagging, and synchronized commenting systems. More specifically, the present invention provides a variety of methods and tools including user interfaces, programming models, data models, algorithms, and others within a client server software and hardware architectural model for use with video and other time-based media.

2. Description of the Related Art

Consumers are shooting more and more personal video using camera phones, webcams, digital cameras, camcorders and other devices, but consumers are typically not skilled videographers nor are they able or willing to learn complex, traditional video editing and processing tools like Apple iMovie or Windows Movie Maker. Nor are most users willing to watch most video "VCR-style", that is in a steady steam of unedited, undirected, unlabeled video.

Thus consumers are being faced with a problem that will be exacerbated as both the number of videos shot and the length of those videos grows (supported by increased processing speeds, memory and bandwidth in end-user devices such as cell phones and digital cameras) while the usability of editing tools lags behind. The result will be more and longer video files whose usability will continue to be
limited by the inability to locate, access, label, discuss, and share granular sub-
segments of interest within the longer videos in an overall library of videos.

In the absence of editing tools of the videos, adding titles and comments to
the videos as a whole does not adequately address the difficulty. For example,
there may be only three 15-second segments of interest scattered throughout a 10
minute long, unedited video.

The challenge faced by viewers is to find those few short segments of
video, which are of interest to them at that time without being required to scan
through the many sections, which are not of interest.

The reciprocal challenge is for users to help each other find those
interesting segments of video. As evidenced by the broad popularity of chat
rooms, blogs etc. viewers want a forum in which they can express their views
about content to each other, that is, to make comments. Due to the time-based
nature of the video, entering and tracking comments and/or tags or labels on
subsegments in time of the video or other time-based media is a unique and
previously unsolved problem.

Additional challenges described in Applicant's incorporated references
apply equally well here, and including especially:

a. the fact that video and accompanying audio is a time dependent,
four dimensional object which needs to be viewed, manipulated and managed by
users on a two-dimensional screen when time is precious to the user who does not
wish to watch entire, unedited, amateur videos (discussed in detail below with
regard to the special complexities of digitally encoded video with synchronized
audio (DEVSA) data);
b. the wide diversity of capabilities of the user devices which users wish to use to watch such videos ranging from PCs to cell phones (as noted further below); and

c. the need for any proposed solution to be able to be structured for ready adaptation and re-encodation to the rapidly changing capabilities of the end-user devices and of the networks which support them.

Those with skill in the art should recognize the more generic terminology "time-based media" which encompasses not only video with synchronized audio but also audio alone plus also a range of animated graphical media forms ranging from sequences of still images to what is commonly called 'cartoons'. All of these forms are addressed herein. The terms, video, time-based media, and digitally encoded video with synchronized audio (DEVSA) are used as terms of convenience within this application with the intention to encompass all examples of time-based media.

A further detriment to the consumer is that video processing uses a lot of computer power and special hardware often not found on personal computers. Video processing also requires careful hardware and software configuration by the consumer. Consumers need ways to edit video without having to learn new skills, buy new software or hardware, become expert systems administrators or dedicate their computers to video processing for great lengths of time.

Consumers have been limited to editing and sharing video that they could actually get onto their computers, which requires the right kind of hardware to handle their own video, and also requires physical movement of media and encoding if they wish to use video shot by another person or which is taken from stock libraries.
When coupled with the special complexities of digitally encoded video with synchronized audio the requirements for special hardware, difficult processing and storage demands combine to reverse the common notion of using "free desktop MIPS and GBs" to relieve central servers. Unfortunately, for video review and editing the desktop is just is not enough for most users. The cell phone is certainly not enough, nor is the Personal Digital Assistant (PDA). There is, therefore, a need for an improved method and system for shared viewing and editing of time-based media.

Those with skill in the conventional arts will readily understand that the terms "video" and "time-based media" as used herein are terms of convenience and should be interpreted generally below to mean DEVSA including content in which the original content is graphical.

Currently available editing tools are typically too difficult and time consuming for consumers to use, largely deriving from their reliance on the same user interface metaphors and import-edit-render pattern of high-end commercial video editing packages like Avid. One form of editing is to reduce the length and/or to rearrange segments of longer form video from camcorders by deleting unwanted segments and by cut-and-paste techniques. Another form of editing is to combine shorter clips (such as those from devices such as cell phones) into longer, coherent streams. Editors can also edit - or make "mixes" – using video and/or audio produced by others if appropriate permission is granted.

This application addresses a unique consumer and data model and other systems that involve manipulation of time-based media. As introduced above, those of skill in the art reviewing this application will understand that the detailed discussion below addresses novel methods of receiving, managing, storing, manipulating, and delivering digitally encoded video with synchronized audio.
(Conveniently referred to as "digitally encoded video with synchronized audio (DEVSA)).

In order to understand the concepts provided by the present, and related inventions, those of skill in the art should understand that DEVSA data is fundamentally distinct from and much more complex than data of those types more commonly known to the public and the broad data processing community and which is conventionally processed by computers such as basic text, numbers, or even photographs, and as a result requires novel techniques and solutions to achieve commercially viable goals (as will be discussed more fully below).

Techniques (editing, revising, compaction, etc.) previously applied to these other forms of data types cannot be reasonably extended due to the complexity of the DEVSA data, and if commonly known forceful extensions are orchestrated they would

- Be ineffective in meeting users’ objectives and/or
- Be economically infeasible for non-professional users and/or
- Make the so-rendered DEVSA data effectively inoperable in a commercially realistic manner.

Therefore a person skilled in the art of text or photo processing cannot easily extend the techniques that person knows to DEVSA.

What is proposed for the present invention is a new system and method for managing, storing, manipulating, editing, operating with and delivering, etc. DEVSA data. As will be discussed herein the demonstrated state-of-the-art in DEVSA processing suffers from a variety of existing, fundamental challenges associated with known DEVSA data operations. The differences between DEVSA and other data types and the consequences thereof are discussed in the following paragraphs.
This application does not address new techniques for digitally encoding video and/or audio or for decoding DEVSA. There is substantive related art in this area that can provide a basic understanding of the same and those of skill in the electronic arts know these references. Those of skill in the art will understand however that more efficient encoding/decoding to save storage space and to reduce transmission costs only serves to greatly exacerbate the problems of operating on DEVSA and having to re-save revised DEVSA data at each step of an operation.

A distinguishing point about video and, by extension stored DEVSA, is to emphasize that video or stored DEVSA represents an object with four dimensions: X, Y, A-audio, and T-time, whereas photos can be said to have only two dimensions (X, Y) and can be thought of as a single object that has two spatial dimensions but no time dimension. The difficulty in dealing with mere two dimensional photo technology is therefore so fundamentally different as to have no bearing on the present discussion (even more lacking are text art solutions).

Another distinguishing point about stored DEVSA that illustrates its unique difficulty in editing operations is that it extends through time. For example, synchronized (time-based) comments are not easily addressed or edited by subsequent users.

Those with skill in the art should be aware of an obvious example of the challenges presented by this time dependence in that it is common for Internet users to post comments on Web sites about specific news items, text messages, photos or other objects which appear on Web sites. The techniques for doing so are well known to those with skill in the art and are commonly used today. The techniques are straightforward in that the comment is a fixed, single data object
and the object commented upon is a fixed, single data object. However the corollaries in the realm of time-based media are not well known and not supported within the current art.

As an illustrative example, consider the fact that a video may extend for five minutes and encompass 7 distinct scenes addressing 7 distinct subjects. If an individual wishes to comment upon scene 5/subject 5, that comment would make no sense if it were tied to the video as a whole. It must be tied only to scene 5 that happens to occur from 3 minutes 22 seconds until 4 minutes 2 seconds into the video.

Since the video is a time-based data object, the comment must also become a time-based data object and be linked within the time space of the specific video to the segment in question. Such time-based comments and such time-dependent linkages are not known or supported within the related arts but are supported within this model.

A stored DEVSA represents an object with four dimensions: X, Y, A, T: large numbers of pixels arranged in a fixed X-Y plane which vary smoothly with T (time) plus A (audio amplitude over time) which also varies smoothly in time in synchrony with the video. For convenience this is often described as a sequence of "frames" (such as 24 frames per second). This is however a fundamentally arbitrary choice (number of "frames" and use of "frame" language) and is a settable parameter at encoding time. In reality the time variance of the pixel's change with time is limited only by the speed of the semiconductors that sense the light.

Before going further it is also important for those of skill in the art to understand the scale of these DEVSA data elements that sets them apart from other text or photo data elements. As a first example, a 10-minute video at 24
"frames" per second would contain 14,400 frames. At 600x800 pixel resolution, 480,000 pixels, one approaches 7 billion pixel representations.

When one adds in the fact that each pixel needs 10-20 bits to describe it and the need to simultaneously describe the audio track, there is a clear and an impressive need for an invention that addresses both the complexity of the data and the fact that the DEVSA represents not a fixed, single object rather a continuous stream of varying objects spread over time whose characteristics can change multiple times within a single video. To date no viable solutions have been provided which are accessible to the typical consumer, other than very basic functions such as storing pre-encoded video files and manipulating those as fixed files.

While one might have imagined that photos and video offer similar technical challenges, the preceding discussion makes it clear again that the difficulties in dealing with mere two dimensional photos which are fixed in time are therefore so fundamentally different and less challenging as to have no bearing on the present discussion.

Some additional facts about DEVSA should be well understood by those of skill in the art; and these include:

a. Current decoding technology allows one to select any instant in time within a video and resolve a "snapshot" of that instant, in effect rendering a photo of that instant and to save that rendering in a separate file. As has been shown, for example in surveillance applications, this is a highly valuable adjunctive technology but it fails to address the present needs.

b. It is not possible to take a "snapshot" of audio, as a person perceives it. Those of skill in the electronic and audio-electronic arts
recognize that audio data is a one dimensional data type: (amplitude versus time). It is only as amplitude changes with time that it is perceivable by a person. Electronic equipment can measure that amplitude if desired for special reasons.

The present application, and those related family applications apply to this understanding of DEVSA when the actual video and audio is compressed (as an illustration only) by factors of a thousand or more but remains nonetheless very large files. Due the complex encoding and encodation techniques employed, those files cannot be disrupted or manipulated without a severe risk to the inherent stability of the underlying video and audio content.

The conventional manner in which users edit digitized data, whether numbers, text, graphics, photos, or DEVSA, is to display that data in viewable form, make desired changes to that viewable data directly and then re-save the now-changed data in digitized form.

The phrase above, "make desired changes to that viewable data", could also be stated as "make desired changes to the manner in which that data is viewed" because what a user "views" changes because the data changes, which is the normative modality. In contrast to this position, the proposed invention changes the viewing of the data without changing the data itself. The distinction is material and fundamental.

In conventional data changes, where storage cost is not an issue to the user, the user can choose to save both the original and the changed version. Some sophisticated commercial software for text and number manipulation can remember a limited number of user-changes and, if requested, display and, if further requested, may undo prior changes.
This latter approach is much less feasible for photos than for text or numbers due to the large size and the extensive encoding required of photo files. It is additionally far less feasible for DEVSA than for photos because the DEVSA files are much larger and because the DEVSA encoding is much more complex and processor intensive than that for photo encoding.

Hi a similar analysis, the processing and storage costs associated with saving multiple old versions of number or text documents is a small burden for a typical current user. However, processing and storing multiple old versions of photos is a substantial burden for typical consumer users today. Most often, consumer users store only single compressed versions of their photos. Ultimately, processing and storing multiple versions of DEVSA is simply not feasible for any but the most sophisticated users even assuming that they have use of suitable editing tools.

As will be discussed, this application proposes new methodologies and systems that address the tremendous conventional challenges of editing heavily encoded digitized media such as DEVSA.

Hi a parallel problem, known to those with skill in the conventional arts associated with heavily encoded digitized media such as DEVSA, is searching for content by various criteria within large collections of such DEVSA.

Simple examples of searching digitized data include searching through all of one's accumulated emails for the text word "Anthony". Means to accomplish such a search are conventionally known and straight-forward because text is not heavily encoded and is stored linearly. On the Internet, companies like Google and Yahoo and many others have developed and used a variety of methods to search out such text-based terms (for example "Washington's Monument").
Similarly, number-processing programs follow a related approach in finding instances of a desired number (for example the number "$1,234.56").

However, when the conventional arts approach digitally encoded graphics or, more challengingly, digitally encoded photos, and far more challengingly, DEVSA, managing the problem becomes increasingly difficult because the object of the search becomes less and less well-defined in terms, (1) a human can explain to a computer, and (2) a computer can understand and use algorithmically. Moreover, the data is ever more deeply encoded as one goes from graphics to photos to DEVSA.

Conventional efforts to employ image recognition techniques for photos and video, and speech recognition techniques for audio and video/audio, require that the digitized date be decoded back to viewable/audible form prior to application such techniques. As will be discussed later, repetitive encoding/decoding with edits introduces substantial risks for graphical, photographic, audio and video data.

As an illustrative example of the substantial challenges of searching, consider the superficially simple graphics search question: "Search the file XYZ graph which includes 75 figures and find all the elements which are "ovals"."

If the search is being done with the same software which created the original file, the search may be possible. However, if the all the user has are images of the figures, the challenges are substantial. To name a few:

1. The user and the computer first have to agree on what "oval" means. Consider the fact that circles are "ovals" with equal major and minor axes.
2. The user and computer have to agree if embedded figures such as pictures or drawings of a dog should be included in the search since the dog's eyes may be "oval".

3. The user and computer have to agree if "zeros" and/or "O's" are ovals or just text.

The point is that recognizing shapes gets tricky.

Turning to photos, unless there are metadata names or tags tied to the photo, which explain the content of the photo, determining the content of the photo in a manner susceptible to search is a largely unsolved problem outside of very specialized fields such as police ID photos. Distinguishing a photo of Mt. Hood from one of Mt. Washington by image recognition is extremely difficult.

This application proposes new methods, systems, and techniques to enable and enhance use, editing and searching of DEVSA files via use of novel types of metadata and novel types of user interactions with integrated systems and software. Specifically related to the distinction made above, this application addresses methods, systems and operational networks that provide the ability to change the manner in which users view digitized data, specifically DEVSA, without necessarily changing the underlying digitized data.

Those of skill in the art will recognize that there has been a tremendous commercial and research demand to cure the long-felt-problem of data loss where manipulating the underlying DEVSA data in situ.

Repetitive encoding and decoding cycles are very likely to introduce accumulating errors with resultant degradation to the quality of the video and audio. Therefore there is strong demand to retain copies of original files in addition to re-encoded files. Since, as stated previously, these are large files even
after efficient encoding, economic pressures make it very difficult to keep many copies of the same original videos.

Thus, the related art in video editing and manipulation favors light repetitive encoding which in turn uses lots of storage but requires keeping more and more copies of successive versions of the encoded data to avoid degradation thus requiring even more storage. As a consequence, those of skill in the art will recognize a need to overcome the particular challenges presented by the current solutions to manipulation of time-based media.

As an illustrative example only, those of skill in the art should recognize the below comparison between DEVSA and other somewhat related data types.

The most common data type on computers (originally) was or involved numbers. This problem was well solved in the 1950s on computers and as a material example of this success one can buy a nice calculator today for $9.95 at a local non-specialty store. As another example, both Lotus® and now Excel® software systems now solve most data display problems on the desktop as far as numbers are concerned.

Today the most common data type on computers is text. Text is a one-dimensional array of data: a sequence of characters. That is, the characters have an X component (no Y or other component). All that matters is their sequence. The way in which the characters are displayed is the choice of the user. It could be on an 8x10 inch page, on a scroll, on a ticker tape, in a circle or a spiral. The format, font type, font size, margins, etc. are all functions added after the fact easily because the text data type has only one dimension and places only one single logical demand on the programmer, that is, to keep the characters in the correct sequence.
More recently a somewhat more complex data type has become popular, photos or images. Photos have two dimensions: X and Y. A photo has a set of pixels arranged in a fixed X-Y plane and the relationship among those pixels does not change. Thus, those of skill in the art will recognize that the photo can be treated as a single object, fixed in time and manipulated accordingly.

While techniques have been developed to allow one to "edit" photos by cropping, brightening, changing tone, etc., those techniques require one to make a new data object, a new "photo" (a newly saved image), in order to store and/or retrieve this changed image. This changed image retains the same restrictions as the original: if one user wants to "edit" the image, the user needs to change the image and re-save it. It turns out that there is little "size", "space", or "time" penalty to that approach to photos because, compared to DEVSA, images are relatively small and fixed data objects.

In summary, DEVSA should be understood as a type of data with very different characteristics from data representing numbers, text, photos or other commonly found data types. Recognizing these differences and their impacts is fundamental to the proposed invention. As a consequence, an extension of ideas and techniques that have been applied to those other, substantially less complex data types have no corollary to those conceptions and solutions noted below. The present invention provides a new manner of (and a new solution for) dealing with DEVSA type data that both overcomes the detriments represented by such data noted above, and results in a substantial improvement demonstrated via the present system and method.

The present invention also recognizes the earlier-discussed need for a system to manage DEVSA data while providing extremely rapid response to user input without changing the underlying DEVSA data.
What is also needed is a new manner of dealing with DEVSA that overcomes the challenges inherent in such data and that enables immediate and timely response to DEVSA data, and especially that DEVSA data and time-based media in general that is amended-or-updated on a continual or rapidly changing basis.

What is not appreciated by the related art is the fundamental data problem involving DEVSA and current systems for manipulating the same in a consumer responsive manner.

What is also not appreciated by the related art is the need for providing a data model that accommodates (effectively) all present modern needs involving high speed and high volume video data manipulation.

Accordingly, there is a need for an improved system and data model for visual browsing, deep tagging, and synchronized commenting of time-based media.

SUMMARY OF THE INVENTION

The present invention proposes a response to the detriments noted above.

Another proposal of this invention is to provide extremely easy-to-use network-based tools for individuals, who may be professional experts or may be amateur consumers (both are referred to herein as users or editors), to upload their videos and accompanying audio and other data (hereinafter called videos) to the Internet, to "edit" their videos in multiple ways and to share those edited videos with others to the extent the editor chooses.

Another proposal of the present invention is to provide a variety of methods and tools including user interfaces, programming models, data models,
algorithms, etc. within a client/server software and hardware architectural model, often an Internet style model, which allow users to more effectively discover and preview and view videos and other time-based media in order to chose and locate sub-segments in time that are of particular interest to them; further to assist others in doing so as well and further to introduce comments to be shared with others on selected sections of the videos.

Another proposal of the invention includes an editing capability that includes, but is not limited to, functions such as abilities to add video titles, captions and labels for sub-segments in time of the video, lighting transitions and other visual effects as well as interpolation, smoothing, cropping and other video processing techniques, both under user-control and automatically.

Another proposal of the present invention is to provide a system for editing videos for private use of the originator or that may be shared with others in whole or in part according to permissions established by the originator, with different privacy settings applying to different time sub-segments of the video.

Another proposal of the present invention is to provide an editing system wherein if users or editors desire, multiple versions are easily created of a video targeted to specific sub-audiences based, for example, on the type of display device used by such sub-audience.

Another proposal of the present invention is to reduce the dependencies on the user's computer or other device, to avoid long user learning curves, and to reduce the need for the user to purchase new desktop software and hardware. To meet this alternative proposal, all video processing and storage must take place on powerful and reliable server computers accessible via the Internet or similar networks.
Another proposal of the present invention is to provide an editing system capable of coping with future advances in consumer or network-based electronics and readily permitting migration of certain software and hardware functions from central servers to consumer electronics including personal computers and digital video recorders or to network-based electronics such as transcoders at the edge of a wireless or cable video-on-demand network without substantive change to the solutions described herein.

Another proposal of the present invention is that videos and associated data linked with the video content may be made available to viewers across multiple types of electronic devices and who are linked via data networks of variable quality and speed, wherein, depending on the needs of that user and that device and the qualities of the network, the video may be delivered as a real-time stream or downloaded in encoded form to the device to be played back on the device at a later time.

Another proposal of the present invention is to accomplish all of these and other capabilities in a manner, which provides for efficient and cost-effective information systems design and management.

Another proposal of the present invention is to provide an improved video operation system with improved user interaction over the Internet.

Another proposal of the present invention is to provide an improved system and data model for shared viewing and editing of a time-based media that has been encoded in a standard and recognized manner and optionally may be encoded in more than one manner.

Another proposal of the present invention is to provide a system, data model, and architecture that enable comments synchronized with DEVSA as it extends through time.
What is additionally proposed for the present invention is a new way for managing, storing, manipulating, operating with and delivering, etc. DEVSA data stored in a recognized manner using playback decision tracking, that is tracking the decisions of users of the manner in which they wish the videos to be played back which may take the form of Playback Decision Lists (PDLs) which are time-dependent metadata co-linked to particular DEVSA data.

Another proposal of the present invention is to provide a data system and operational model that enables generation and tracking of multiple and independent (hierarchical) layers of time-dependent metadata that are stored in a manner linked with video data that affect the way the video is played back to a user at a specific time and place without changing the underlying stored DEVSA.

It is another proposal of the present invention to provide a system, method, and operational model that tracks via time-dependent metadata (via play back decision track or PDLs) individual user preferences on how to view video.

Another proposal of the present invention is to enable a system for deep tagging video data to identify a specific user, in a specific hierarchy, in a specific modality (soccer, kids, fun, location, family, etc) while enabling a sharable or defined group interaction.

Another proposal of the present invention is to enable a operative system that determines playback decision lists (PDLs) and enables their operation both in real-time on-line viewing of DEVSA data and also enables sending the PDL logic to an end-user device for execution on that local device, when the DEVSA is stored on or delivered to that end-user device, to minimize the total bit transfer at each viewing event thereby further minimizing response time and data transfer.

The present invention provides a system, method, and apparatus for visual browsing, deep tagging, and synchronized commenting regarding interactive
Operational modules are provided that allow users to more effectively discover and preview videos in order to choose and locate sub-segments in time that are of particular user interest, to deep tag or label segments as desired for future retrieval and to provide user comments viewable by others on selected sections of the video subject matter. In summary, there are three major alternative components involved in meeting these proposals:

Component 1: Provides efficient means to preview videos, select potentially interesting segments, and view only those that appear to be of most interest. This is referred to by the term "visual browsing".

Component 2: Provides efficient and effective means to label or "deep tag" those interesting segments or time intervals within the video or the video as a whole for future retrieval by the user and by others. This is referred to by the term "deep tagging".

Component 3: Provides efficient and effective means to enter comments synchronized (by time internal to the video) with those interesting segments or with the video as a whole for future retrieval by the user and by others. This is referred to by the term "synchronized commenting".

The principal proposal of Component 1, visual browsing, is to provide a convenient system for users: (a) to preview a lengthy video rapidly in a manner which is easy to learn, (b) to identify and select potentially interesting segments rapidly and easily using methods which are (i) consistent with users' experience with other methods of viewing information, the end-user devices and the Internet, (ii) consistent with the time-dependent nature of video, and; (iii) take advantage of internal characteristics of the video such as scene changes, image types such as
face close-ups (potentially identified by video object recognition techniques) and image characteristics such as blurry vs. non-blurry; (c) to play the selected segments and, if desired, choose to save, forward, email, share, etc. those segments; and (d) to easily proceed, backward or forward, to other segments of potential interest.

The principal proposal of Component 2, deep tagging, is to provide a convenient system for users: (a) to identify a specific time interval, a "segment", within a longer video as being of specific interest, (b) to "deep tag" this interval with an identifying name or phrase or icon or other identifier and (c) to have that segment retrievable by the user or by others by means of the "deep tag" treated as a searchable database entity; (i) in such a manner that the user retrieving the segment can view only the segment and the deep tag identified by the deep tag without having to view or search the entire video; (ii) wherein even if the video has not been edited and (iii) without changing the original DEVSA in any manner.

Another proposal of Component 2 is that multiple users may add individual deep tags that may overlap time segments of the same video without interference.

Yet another proposal of Component 2 is that an individual user may control which other users may observe and use his deep tags.

Other proposals of Component 2 are that deep tags may be placed in a searchable database wherein they can be searched by a variety of means typical of Internet search engines such as Google®; for example: by users who enter the deep tag, by category, by interest group, by time entered, by word or phrase, etc.

One principal proposal of Component 3 is to provide a convenient system for users: (a) to identify a specific time point or interval, a "segment", within a longer video as being of specific interest, (b) to enter written or spoken comments
associated with and synchronized with this segment or with an entire video and
(c) to have those comments retrievable as a searchable database entity and able to
be viewed or heard by the user or by others who access the video by a variety of means.

Taken together it should be recognized that Components 1, 2 and 3
constitute a cyclical process. That process may be exemplified as follows:

1. user 1 views a video first
   a. selects interesting segments using visual browsing tools
   b. deep tags those segments and adds synchronized comments
   c. shares those segments with users 2 - 6
   d. allows general users access to his deep tags and
      synchronized comments

2. users 2 - 6 employ user 1's deep tags and synchronized comments
to view the video but
   a. explore it further
   b. select more and/or different interesting segments
   c. add more deep tags and synchronized comments to their
      individually selected segments
   d. share it with a distinct set of friends and also
   e. make it available for general users.

3. general users view selected segments, deep tags and synchronized
   comments of users 1 - 6 and those of other general users and
   continue the process.
It is another object of the present invention that Components 1, 2 and 3 each separately enhances the cyclical process by adding an additional layer of interest and an additional search mechanism.

As will be discussed in more detail below, as this process continues, and selections of interesting segments accumulate, deep tags accumulate and synchronized comments accumulate all around this initial video. Thus, for example by the time a user 126 comes along there is great deal of information available about this and many other videos which makes user 126 able to zoom in to what is very likely to be of real interest to him in many different videos.

As a consequence, one of the overarching proposals of the present invention is to make all this information: video, deep tags, synchronized comments etc. available to all users (subject to tiered or other permissions) in order to facilitate user ability to find and enjoy the video segments they will like most along with the deep tags and synchronized comments made by others whose deep tags and synchronized comments are of interest to them. All of this is to be done without changing the underlying DEVSA as is explained herein and in Applicant's related applications noted and incorporated by reference.
BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 represents an illustrative flow diagram for an operational system and architectural model for one aspect of the present invention.

Fig. 2 represents an illustrative flow diagram of an interactive system and data model for shared viewing and editing of time-based media enabling a smooth interaction between a video media user and underlying stored DEVSA data.

Fig. 3 is an illustrative flow diagram for a web-based system for enabling and tracking editing of personal video content.

Fig. 4 is a screen image of the first page of a user's list of the user's uploaded video data.

Fig. 5 is a screen image of edit and data entry page allowing a user to "add" one or more videos to a list of videos to be edited as a group.

Fig. 6 is a screen image of an "edit" and "build" step using the present system.

Fig. 7 is a screen image of an edit display page noting three videos successively arranged in text-like formats with thumbnails roughly equally spaced in time throughout each video. The large image at upper left is a 'blow-up' of the current thumbnail.

Fig. 8 is a screen image of a partially edited page where selected frames with poor video have been "cut" by the user via 'mouse' movements.

Fig. 9 is a screen image of the original three videos where selected images of a "pool cage" have been "cut" during a video edit session. The user is now finished editing.

Fig. 10 is a screen image of the first pages of a user list of uploaded video data. The original videos have not been altered by the editing process.
Fig. 11 is a flow diagram of multi-user visual browsing, deep tagging and synchronized commenting.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Reference will now be made in detail to several embodiments of the invention that are illustrated in the accompanying drawings. Wherever possible, same or similar reference numerals are used in the drawings and the description to refer to the same or like parts or steps. The drawings are in simplified form and are not to precise scale. For purposes of convenience and clarity only, directional terms, such as top, bottom, up, down, over, above, and below may be used with respect to the drawings. These and similar directional terms should not be construed to limit the scope of the invention in any manner. The words "connect," "couple," and similar terms with their inflectional morphemes do not necessarily denote direct and immediate connections, but also include connections through mediate elements or devices.

Description of Invention: The present invention proposes a system including three major, enabingly-linked and alternatively engagable components, all driven from central servers systems.

1. A series of user interfaces;
2. An underlying programming model and algorithms; and
3. A data model.

In a preferred mode all actual video manipulation is done on the server, but local servers, consumer devices, or other effective computer systems may be engaged for operation. The "desktop" or other user interface device needs only to
operate Web browser software or the equivalent, a video & audio player which can meet the server's requirements and its own internal display and operating software and be linked to the servers via the Internet or another suitable data connection. As advances in consumer electronics permit, other implementations become feasible and are described in the last section. Those alternative implementations certain functions can migrate from the servers to end-user devices or to network-based devices without changing the basic design or intent of the invention.

10 **The User Interface**

An important component of a successful video editing system is a flexible user interface which:

1. is consistent with typical user experience, but not necessarily typical video editing user interfaces,

2. will not place undue burdens on the end-user's device, and

3. is truly linked to the actual DEVSA.

A major detriment to be overcome is that the DEVSA is a four dimensional entity which needs to be represented on a two dimensional visual display, a computer screen or the display of a handheld device such as a cell phone or an iPod®.

These proposals take the approach of creating an analog of a text document made up, not of a sequence of text characters, but of a sequence of "thumbnail" frame images at selected times throughout the video. For users who express the English language as a preference, these thumbnails are displayed from left to right in sequential rows flowing downward in much the way English text is
displayed in a book. (Other sequences will naturally be more appropriate for users whose written language progresses in a different manner.

A useful point is to have the thumbnails and the "flow" of the video follow a sequence similar to that of the user's written language; such as left-to-right, top-to-bottom, or right-to-left. A selected frame may be enlarged and shown above the rows for easier viewing by the user. Figure 7 shows an example.

As a further example, a 5 minute video might be initially displayed as 15 thumbnail images spaced about 20 seconds apart in time through the video. This user interface allows the user to quickly grasp the overall structure of the video. The choice of 15 images rather than some higher or lower number is initially set by the server administrator but when desired by the user can be largely controlled by the user as he/she is comfortable with the screen resolution and size of the thumbnail image.

By means of mouse (or equivalent) or keyboard commands, the user can "zoom in" on sub-sections of the video and thus expand to, for example, 15 thumbnails covering 1 minute of video so that the thumbnails are only separated by about 4 seconds. Whenever desired, the user can "zoom-in" or "zoom-out" to adjust the time scale to meet the user's current editing or viewing needs. One approach is the so-called "slider" wherein the user highlights a selected portion of the video timeline causing that portion to be expanded (zoomed-in) causing additional, more closely placed thumbnails of just that portion to be displayed. Additionally, other view modes can be provided, for example the ability to see the created virtual clip in frame (as described herein), clip (where each segment is shown as a single unit), or traditional video editing time based views.

Additional methods of displaying thumbnails over time can also be used to meet specific user needs. For example, thumbnails may also be generated
according to video characteristics such as scene transitions or changes in content (recognized via video object recognition).

The user interfaces allow drag and drop editing of different video clips with a level of ease similar to that of using a word processing application such as Microsoft Word®, but entirely within a web browser. The user can remove unwanted sections of video or insert sections from other videos in a manner analogous to the cut/copy-and-paste actions done in text documents.

A noted previously, these "drag, drop, copy, cut, paste" edit commands are stored within the data model as metadata, do not change the underlying DEVSA data, and are therefore in clear contrast with the related art.

The edit commands, deep tags and synchronized commentary can all be externally time-dependent at the user's option. As an elementary example, "If this is played between March 29 and March 31, Play Audio: "HAPPY BIRTHDAY". Ultimately, all PDL may be externally time dependent if desired.

Other user interface representations of video streams on a two dimensional screen are also possible and could also be used without disrupting the editing capabilities described herein. One example is to arrange the page of thumbnail images in time sequence as if they were a deck of cards or a book thus creating an apparent three-dimensional object where the depth into the "deck of cards" or the "book" is a measure of time. Graphical "tabs" could appear on the cards or book pages (as on large dictionaries) which would identify the time (or other information) at that depth into the deck or book. The user could then "cut the deck" or "open the book" at places of his choosing and proceed in much the same way as described above. These somewhat different representations would not change the basic nature of the claims herein. There can be value in combining multiple such representations to aid users with diverse perception preferences or
to deal with large quantities of information.

In the preceding it has been assumed that the "user" has the legal right to modify the display of the DEVSA, which may be arguably distinguished from a right to modify the DEVSA itself. There may be cases where there are users with more limited rights. The user interface will allow the individual who introduces the video and claims full edit rights, subject to legal review, to limit or not to limit the rights of others to various viewing permissions and so-called "editing" functions (these are "modifying the display" edits noted earlier. These permissions can be adjusted within various sub-segments of the video. It is expected that the addition of deep tags and commentary by others will not generally be restricted in light of the fact that the underlying DEVSA is not compromised by these edit commands as is explained more fully below.

Before going further, and in order to fully appreciate the major innovation described in this and the related applications, it is necessary to introduce a new enabling concept which is referred to as the Playback Decision List or hereafter "PDL." The PDL is a portion of metadata contained within a data model or operational system for manipulating related video data and for driving, for example, a flash player to play video data in a particular way without requiring a change in the underlying video data (DEVSA). This new concept of a PDL is best understood by considering its predecessor concepts that originated years ago in film production and are used today by expert film and video directors and editors.

The predecessor concept is an Edit Decision List or EDL. It is best described with reference to the production of motion pictures, hi such a production many scenes are filmed, often several times each, in a sequence that has no necessary relationship to the story line of the movie. Similarly,
background music, special effects, and other add-ons are produced and recorded or filmed independently. Each of those film and audio elements is carefully labeled and timed with master lists.

When these master lists are complete, the film's director and editor sit down, often for a period of months, and review each element while gradually writing down and creating and revising an EDL which is a very detailed list, second by second, of which film sequences will be spliced together in what sequence perhaps with audio added to make up the entire film. Additionally, each sequence may have internal edits required such as fade-in/out, zoom-in/out, brightens, raise audio level and so on. The end result is an EDL. Technicians use the EDL to, literally in the case of motion picture, cut and paste together the final product. Some clips are just cut and "left on the cutting room floor". Expert production of commercial video follows a very similar approach.

The fundamental point of an EDL is that one takes segments of film or video and audio and possibly other elements and links them together to create a new stream of film or video, audio, etc. The combining is done at the film or video level, often physically. The original elements very likely were cut, edited, cropped, faded in/out, or changed in some other manner and may no longer even exist in their original form.

This EDL technique has proven to be extremely effective in producing high quality film and video. It requires a substantial commitment of human effort, typically many staff hours per hour of final media and is immensely costly. It further requires that the media elements to be edited be kept in viewable/hearable form in order to be edited properly. Such an approach is economically impossible when dealing with large quantities of consumer-produced video. The PDL concept introduced herein provides a fundamentally
different way to obtain a similar end result. The final "quality" of the video will depend on the skill and talent of the editor nonetheless.

The PDL incorporates as metadata associated with the DEVSA all the edit commands, deep tags, commentary, permissions, etc. introduced by a user via a user interface (as will be discussed). It is critical to recognize that multiple users may introduce edit commands, deep tags, commentary, permissions, etc. all related to the same DEVSA without changing the underlying video data. The user interface and the structure of the PDL allow a single PDL to retrieve data from multiple DEVSA.

The result is that a user can define, for example, what is displayed as a series of clips from multiple original videos strung together into a "new" video without ever changing the original videos or creating a new DEVSA file. Since multiple users can create PDLs against the same DEVSA files, the same body of original videos can be displayed in many different ways without the need to create new DEVSA files. These "new" videos can be played from a single or from multiple DEVSA files to a variety of end-user devices through the use of software and/or hardware decoders that are commercially available. For performance or economic reasons, copies or transcodings of certain DEVSA files may be created or new DEVSA files may be rendered from an edited segment, to better serve specific end-user devices without changing the design or implementation of the invention in a significant manner.

Since multiple types of playback mechanisms are likely to be needed such as one for PCs, one for cell phones and so on, the programming model will create a "master PDL" from which algorithms can create multiple variations of the PDL suitable for each of the variety of playback mechanisms as needed. The PDL executes as a set of instructions to the video player.
As discussed earlier, in certain cases it is advantageous to download an entire encoded file in a form suitable to a specific device type rather than stream a display in real time. In the "download" case, the system will create the file using the PDL and the DEVSA, re-encode for saving it in the appropriate format, and then send that file to the end-user device where it is stored until the user chooses to play it. This "download" case is primarily a change in the mode of delivery rather a fundamentally distinct methodology.

The crucial innovation introduced by PDL is that it controls the way the DEVSA is played to any specific user at any specific time. It is a control list for the DEVSA player (flash player/video player). AU commands (edits, sequences, deep tags, comments, permissions, etc.) are executed at playback time while the underlying DEVSA does not change. This makes the PDL in stark contrast to an EDL which is a set of instructions to create a new DEVSA out of previously existing elements.

Having competed the overall supporting discussion, reference is made now to Fig. 1, an architectural review of a system model 100 for improving manipulation and operations of video and time-based DEVSA data. It should be understood, that the term "video" is sometimes used below as a term of convenience and should be interpreted to mean DEVSA, or more broadly time-based media.

By viewing the technological architecture of system model 100, those of skill in the art will recognize that an end-user 101 may employ a range of known user device types 102 (such as PCs, cell phones, PDAs, iPods et al.) to create and view DEVSA/video data.
Devices 102 include a plurality of user interfaces, operational controls, video management requirements, programming logic, local data storage for diverse DEVSA formats, all represented via capabilities 103.

Capabilities 103 enable a user of a device 102 to perform multiple interaction activities 104 relative to a data network 105. These activities 104 are dependent upon the capacities 103 of devices 102, as well as the type of data network 105 (wireless, dial, DSL, secure, non-secure, etc.).

Activities 104 including upload, display, interact, control, etc. of video, audio and other data via some form of data network 105 suited to the user device in a manner known to those of skill in the art. The user's device 102, depending on the capabilities and interactions with the other components of the overall architecture system 100, will provide 103 portions of the user interface, program logic and local data storage.

Other functions are performed within the system environment represented at 107 which typically will operate on servers at central locations while allowing for certain functionality to be distributed through data network 105 as technology allows and performance and economy suggest without changing the architecture and processes as described herein.

All interactions between system environment 107 and users 101 pass through a user interface layer 108 which provides functionality commonly found on Internet or cell phone host sites such as security, interaction with Web browsers, messaging etc. and analogous functions for other end-user devices.

As discussed, the present system 100 enables user 101 to perform many functions, including uploading video/DEVSA, audio and other information from his end-user device 102 via data network 105 into system environment 107 via a first datapath 106.
First data path 106 enables an upload of DEVSA/video via program logic upload process loop 110. Upload process loop 110 manages the uploading process which can take a range of forms.

For example, in uploading video/DEVSA from a cell phone, the upload process 110 can be via emailing a file via interactions 104 and data network 105. A second example, for video captured by a video camera, the video may be transferred from the camera to the user's PC (both user devices 102) and then uploaded from the PC to system environment 107 web site via the Internet in real time or as a background process or as a file transfer. Physical transmission of media is also possible.

During system operation, after a successful upload via uploading process loop 110, each video is associated with a particular user 101 and assigned a unique user and upload and video identifier, and passed via pathway HOA to an encode video process system 111 where it is encoded into one or more standard forms as determined by the system administrators or in response to a user request. The encoded video/DEVSA then passes via conduit HIA to storage in the DEVSA storage files 112. At this time, the uploaded, encoded and stored DEVSA data can be manipulated for additional and different display (as will be discussed), without underlying change. As will be more fully discussed below, the present data system 100 may display DEVSA in multiple ways employing a unique player decision list (PDL) for tracking edit commands as metadata without having to re-save, and re-revise, and otherwise modify the initially saved DEVSA.

Additionally, and as can be viewed from Fig. 1, during the upload (105-106-110), encodation (10A-111), and storage (111A-112) processes stages of system 100; a variety of "metadata" is created about the DEVSA including user ID, video ID, timing information, encoding information including the number and
types of encodings, access information, and many other types of metadata, all of which passes via communication paths 114 and 112A to the Metadata / PDL storage facility (ies) 113. There may be more than one metadata/PDL storage facility. As will be later discussed, the PDL drives the software controller for the video player on the user device via display control 116/play control 119 (as will be discussed).

Such metadata will be used repeatedly and in a variety of combinations with other information to manage and display the DEVSA combined with the metadata and other information to meet a range of user requirements. The present system also envisions a controlled capacity to re-encode a revised DEVSA video data set without departing from the scope and spirit of the present invention.

It is expected that many users and others including system administrators will upload (over time) many DEVSA to system environment 107 so that a large library of DEVSA (stored in storage 112) and associated metadata (stored in storage 113) will be created by the process described above.

Following the same data path 106 users can employ a variety of functions generally noted by interaction with video module 115. Several types of functionalities 115A are identified as examples within interact with video module 115; including editing, visual browsing, commenting, social browsing, etc. Some of these functions are described in related applications. These functions include the user-controlled design and production of permanent DEVSA media such as DVDs and associated printing and billing actions 117 via a direct data pathway 117A, as noted. It should be noted that there is a direct data path between the DEVSA files 112 and the functions in 117 (not shown in the Figure for reasons of readability.)
Many of the other functions 115A are targeted at online and interactive display of video and other information via data networks. The functions 115 interact with users via communication path 106; and it should be recognized that functions 115A use, create, and store metadata 113 via path 121.

User displays are generated by the functions 115/115A via path 122 to a display control 116, which merges additional metadata via path 121A, thumbnails (still images derived from videos) from 112 via paths 120.

Thumbnail images are created during encoding process 111 and optionally as real time process acting on the DEVSA without modifying the DEVSA triggered by one of the functions 115/115A (play, edit, comment, etc.).

Logically the thumbnails are part of the DEVSA, not part of the metadata, but they may be alternatively and adaptively stored as part of metadata in 113. An output of display control 116 passes via pathway 118 to play control 119 that merges the actual DEVSA from storage 112 via pathway 119A and sends the information to the data network 105 via pathway 109.

Since various end-user devices 102 have distinct requirements, multiple play control modules may easily be implemented in parallel to serve distinct device types. It is also envisioned, that distinct play control modules 119 may merge distinct DEVSA files of the same original video and audio with different encoding via 119A depending on the type of device being supported.

It is important to note that interactive functions 115/115A do not link directly to the DEVSA files stored at 112, only to the metadata/PDL files stored at 113. The display control function 116 links to the DEVSA files 112 only to retrieve still images. A major purpose of this architecture within system 100, is that the DEVSA, once encoded, is preferably not manipulated or changed - thereby avoiding the earlier noted concerns with repeated decoding, re-encoding
and re-saving. All interactive capabilities are applied at the time of play controll9 as a read-only process on the DEVSA and transmitted back to user 110 via pathway 109.

Those with skill in the art should recognize that PDLs and other metadata as discussed herein can apply not only to real time playback of videos and other time-based media but also to the non-real-time playback of such media such as might be employed in the creation of permanent media such as DVDs.

Referring now to Fig. 2, in a manner similar to that discussed with Fig. 1, here an electronic system, integrated user interface, programming module and data model 200 describes the likely flows of information and control among various components noted therein. Again, as noted earlier, the term "video" is sometimes used below as a term of convenience and should be interpreted by those of skill in the art to mean DEVSA.

Here, an end-user 201 may optionally employ a range of user device types 202 such as PCs, cell phones, iPods etc. which provide user 201 with the ability to perform multiple activities 204 including upload, display, interact, control, etc. of video, audio and other data via some form of a data network 205 suited to the particular user device 202.

User devices 202, depending on their capabilities and interactions with the other components of the overall architecture for proper functioning, will provide local 203 portions of the user interface, program logic and local data storage, etc., as will also be discussed.

Other functions are performed within the proposed system environment 207 which typically operates on one or more servers at central locations while allowing for certain functionality to be distributed through the data network as
technology allows and performance and economy suggest without changing the program or data models and processes as described herein.

As shown, interactions between system environment 207 and users 201 pass through a user interface layer 208 which provides functionality commonly found on Internet or cell phone host sites such as security, interaction with Web browsers, messaging etc. and analogous functions for other end-user devices.

As noted earlier, users 201 may perform many functions; including video, audio and other data uploading DEVSA from user device 202 via data network 205 into system environment 207 via data path 206.

An upload video module 210 provides program logic that manages the upload process which can take a range of forms. For video from a cell phone, the upload process may be via emailing a file via user interface 208 and data network 205. For video captured by a video camera, the video can be transferred from a camera to a user's PC and then uploaded from the PC to system environment 207 via the Internet in real time or as a background process or as a file transfer. Physical transmission of media is also possible.

During operation of system 200, and after successful upload, each video is associated with a particular user 201, assigned a unique identifier, and other identifiers, and passed via path 210A to an encode video process module 211 where it is encoded into one or more standard DEVSA forms as determined by a system administrators (not shown) or in response to a particular user's requests. The encoded video data then passes via pathway 211A to storage in DEVSA storage files 212.

Within DEVSA files in storage 212, multiple ways of encoding a particular video data stream are enabled; by way of example only, three distinct ways 212B, labeled D_A, D_B, D_C are represented. There is no significance to the
use of three as an example other than to illustrate that there are various forms of DEVSA encoding and to illustrate this diversity system 200 enables adaptation to any particular format desired by a user and/or specified by system administrators.

One or more of the multiple distinct methods of encoding may be chosen for a variety of reasons. Some examples are distinct encoding formats to support distinct kinds of end-user devices (e.g., cell phones vs. PCs), encoding to enhance performance for higher and lower speed data transmission, encoding to support larger or smaller display devices. Other rationales known for differing encodation forms are possible, and again would not affect the processes or system and model 200 described herein. A critical point is that the three DEVSA files 212B labeled $D_A$, $D_B$, $D_C$ are encodings of the same video and synchronized audio using differing encodation structures. As a result, it is possible to store multiple forms of the same DEVSA file in differing formats each with a single encodation process via encodation video 211.

Consequent to the upload, encode, store processes a plurality of metadata 213A is created about that particular DEVSA data stream being uploaded and encoded; including user ID, video ID, timing information, encoding information, including the number and types of encodings, access information etc. which passes by paths 214 and 212A respectively to the Metadata / PDL (playback decision list) storage facilities 213. Such metadata will be used repeatedly and in a variety of combinations with other information to manage and display the DEVSA combined with the metadata and other information to meet a range of user requirements.

Thus, as with the earlier embodiment shown in Fig. 1, those of skill in the art will recognize that the present invention enables a single encodation (or more
if desired) but many metadata details about how the encoded DEVSA media is to be displayed, managed, parsed, and otherwise processed.

It is expected that many users and others including system administrators (not shown) will upload many videos to system environment 207 so that a large library of DEVSA and associated metadata will be created by the process described above.

Following the same data path 206, users 201 may employ a variety of program logic functions 215 which use, create, store, search, and interact with the metadata in a variety of ways a few of which are listed as examples including share metadata 215A, view metadata 215B, search metadata 215C, show video 215D etc. These data interactions utilize data path 221 to the Metadata / PDL databases 213. A major functional portion of the metadata is Playback Decision Lists (PDLs) that are described in detail in other, parallel submissions, each incorporated fully by reference herein. PDLs, along with other metadata, control how the DEVSA is played back to users and may be employed in various settings.

As was shown in Fig. 2 many of the other functions in program logic box 215 are targeted at online and interactive display of video and other information via data networks. As was also shown in Fig. 1, but not indicated here, similar combinations of metadata and DEVSA can be used to create permanent media.

Thus, those of skill in the art will recognize that the present disclosure also enables a business method for operating a user interface 208.

It is the wide variety of metadata, including PDLs, created and then stored which controls the playback of video, not a manipulation of the underlying and encoded DEVSA data.
In general the metadata will not be dependent on the type of end-user device utilized for video upload or display although such dependence is not excluded from the present disclosure.

The metadata does not need to incorporate knowledge of the encoded DEVSA data other than its identifiers, its length in clock time, its particular encodings, knowledge of who is allowed to see it, edit it, comment on it, etc. No knowledge of the actual images or sounds contained within the DEVSA is required to be included in the metadata for these processes to work. While this point is of particular novelty, this enabling system 200 is more fully illustrative.

Such knowledge of the actual images or sounds contained within the DEVSA while not necessary for the operation of the current system enables enhanced functionalities. Those with skill in the art will recognize that such additional knowledge is readily obtained by means of techniques including voice recognition, image and face recognition as well as similar technologies. The new results of those technologies can provide additional knowledge that can then be integrated with the range of metadata discussed previously to provide enhanced information to users within the context of the present invention. The fact that this new form of information was derived from the contents of the time-based media does not imply that the varied edit, playback and other media manipulation techniques discussed previously required any decoding and re-encoding of the DEVSA. Such knowledge of the internal contents of the time-based media can be obtained by decoding with no need to re-encode the original video so the basic premises are not compromised.

User displays are generated by functions 215 via path 222 to display control 216 which merges additional metadata via path 221A, thumbnails (still images derived from videos) from DEVSA storage 212 via pathway 220. (Note
that the thumbnail images are not part of the metadata but are derived directly from the DEVSA during the encoding process 211 and/or as a real time process acting on the DEVSA without modifying the DEVSA triggered by one of the functions 215 or by some other process. Logically the thumbnails are part of the DEVSA, not part of the metadata stored at 213, but alternative physical storage arrangements are envisioned herein without departing from the scope and spirit of the present invention.

An output of display control 216 passes via pathways 218 to play controller 219, which merges the actual DEVSA from storage 212 via data path 219A and sends the information to the data network via 209. Since various end-user devices have distinct requirements, multiple play control modules may be implemented in parallel to serve distinct device types and enhance overall response to user requests for services.

Depending on the specific end-user device to receive the DEVSA, the data network it is to traverse and other potential decision factors such as the availability of remote storage, at playback time distinct play control modules will utilize distinct DEVSA such as files $D_A$, $D_B$, or $D_c$ via 219A.

The metadata transmitted from display control 216 via 218 to the play control 219 includes instructions to play control 219 regarding how it should actually play the stored DEVSA data and which encoding to use.

The following is a sample of a PDL - playback decision list - and a tracking of user decisions in metadata on how to display the DEVSA data. Note that two distinct videos (for example) are included here to be played as if they were one. A simple example of typical instructions might be:
Instruction (exemplary):

- Play video 174569, encoding b, time 23 to 47 seconds after start:
  - Fade in for first 2 seconds – personal decision made for tracking as metadata on PDL.
  - Increase contrast throughout – personal decision made for PDL.
  - Fade out last 2 seconds - personal decision made for PDL.

- Play video 174569, encoding b, time 96 to 144 seconds after start
  - Fade in for first 2 seconds - personal decision made for PDL.
  - Increase brightness throughout - personal decision made for PDL.
  - Fade out last 2 seconds - personal decision made for PDL.

- Play video 174573 (a different video), encoding b, time 45 to 74 seconds after start
  - Fade in for first 2 seconds - personal decision for PDL.
  - Enhance color AND reduce brightness throughout, personal decision for PDL.
  - Fade out last 2 seconds - personal decision for PDL.

The playback decision list (PDLs) instructions are those selected using the program logic functions 215 by users who are typically, but not always, the originator of the video. Note that the videos may have been played "as one" and then have had applied changes (PDLs in metadata) to the visual video impression and unwanted video pieces eliminated. Nonetheless the encoded DEVSA has not been changed or overwritten, thereby minimizing risk of corruption, the expense of re-encoding has been avoided and a quick review and co-sharing of the same video and audio among video editors has been enabled.
Much other data may be displayed to the user along with the DEVSA including metadata such as the name of the originator, the name of the video, the groups the user belongs to, the various categories the originator and others believe the video might fall into, comments made on the video as a whole or on just parts of the video, deep tags or labels on the video or parts of the video.

It is important to note that the interactive functions 215 for reviewing and using DEVSA data, do not link to the DEVSA files, only to the metadata files, it is the metadata files that back link to the DEVSA data. Thus, display control function 216 links to DEVSA files at 212 only to retrieve still images. A major purpose of this data architecture and data system 200 imagines that the DEVSA, once encoded via encodation module 211, is not manipulated or changed and hence speed and video quality are increased, computing and storage costs are reduced. AU interactive capabilities are applied at the time of play control that is a read-only process on the DEVSA.

Those of skill in the art should recognize that in optional modes of the above invention each operative user may share their metadata with others, create new metadata, or re-use previously stored metadata for a particular encoded video.

Referring now to Fig. 3 an operative and editing system 300 comprises at least three major, linked components, including (a) central servers 307 which drive the overall process along a plurality of user interfaces 301 (one is shown), (b) an underlying programming model 315 housing and operatively controlling operative algorithms, and (c) a data model encompassing 312 and 313 for manipulating and controlling DEVSA and associated metadata.

Those of skill in the art should understand that all actual video manipulation is done on the server. Thus this concept depicted here envisions that
a "desktop" or other user interface device need only to operate Web browser software and its own internal video player and display and operating software and be linked to servers 307 via the Internet or another suitable data network connection 305. Those of skill in the art should understand that the PDL produces a set of instructions for the components of the central system environment, any distributed portions thereof and end-user device video player and display. The PDL is generated on the server while the final execution of the instructions generally takes place on the end-user device.

As a consequence, the present discussion results in "edit-type commands" becoming a subset of the metadata described earlier.

Those of skill in the art should understand that while much of the discussion in this application is focused on video. The capabilities described herein apply equally to audio. They would also apply to many forms of graphic material, and certainly all graphic material which has been encoded in video format. Other than time-dependent functions (that is time internal to the DEVSA), they apply equally to photographic images and to text.

During operation, a user (not shown) interfaces with user interface layer 308 and system environment 307 via data network 305. A plurality of web screen shots 301 is represented as illustrated examples of the process of video image editing that is shown in greater detail with Figs. 4 through 10.

During personal editing of content, a user (not shown) interacts with user interface layer 308 and transmits commands through data network 305 along pathway 306.

As shown a user has uploaded multiple, separate videos vid 1, vid 2, vid 3 using processes 310, 310', 310". Then via parallel processes 310 the three videos are encoded in process 311. In this example we show each video being encoded
in two distinct formats \(D_{\text{Id1A}}B_{\text{V}i<\pi B}\) based either on system administration rules or on user requests. Via path 311A two encoded versions of each of the three videos is stored in 312 labeled respectively \(D_{\text{Id1A}}D_{VkIB}\) and so on where those videos of a specific user are retained and identified by user at grouping 312B.

It should be similarly understood, that the initial uploading steps 310 for each of the videos generate related metadata and PDLs 313 transferred to a respective storage module 313, where each user's initial metadata is individually identified in respective user groupings 313A.

Those of skill in the art will understand that multiple upload and encode steps allow users to display, review, and edit multiple videos simultaneously. Additionally, it should be readily recognized that each successive edit or change by an individual is separately tracked for each respective video for each user. When editing multiple videos like this - or just one video - the user is creating a new PDL which is a new logical object which is remembered and tracked by the system.

As will be understood, videos may be viewed, edited, and updated in parallel with synchronized comments, deep tagging and identifying.

The present system enables social browsing of others’ multiple videos with synchronized commenting for a particular single video or series of individual videos.

A display control 316 receives data via paths 312A and thumbnails via path 320 for initially driving play controller 319 via pathway 318.

As is also obvious from Fig. 3, an edit program model 315 (discussed in more detail below) receives user input via pathway 306 and metadata and PDLs via pathway 321.
The edit program model 315 includes a controlling communication path 322 to display control 316. As shown, the edit program model 315 consists of sets of interactive programs and algorithms for connecting the users' requests through the aforementioned user interfaces 308 to a non-linear editing system on server 307 which in turn is linked to the overall data model (312 and 313 etc.) noted earlier in-part through PDLs and other metadata.

Since multiple types of playback mechanisms are likely to be needed such as one for PCs, one for cell phones and so on, the edit program model 315 will create a "master PDL" from which algorithms can adaptively create multiple variations of the PDL suitable for each of the variety of playback mechanisms as needed. Here, the PDL is executed by the edit program model and algorithms 315 that will also interface with the user interface layer 308 to obtain any needed information and, in turn, with the data model (See Fig. 2) which will store and manage such information.

The edit program model 315 retrieves information from the data model as needed and interfaces with the user interface layer 308 to display information to multiple users. Those of skill in the arts of electronic programming should also recognize that the edit program model 315 will also control the mode of delivery, streaming or download, of the selected videos to the end-user; as well as perform a variety of administrative and management tasks such as managing permissions, measuring usage (dependency controls, etc.), balancing loads, providing user assistance services, etc. in a manner similar to functions currently found on many Web servers.

As noted earlier the data model generally in Figs. 1 and 2, manages the DEVSA and its associated metadata including PDLs. As discussed previously, changes to the metadata including the PDLs do not require and in general will not
result in a change to the DEVSA. However for performance or economic reasons
the server administrator may determine to make multiple copies of the DEVSA
and to make some of the copies in a different format optimized for playback to
different end-user device types. The data model noted earlier and incorporated
here assures that links between the metadata associated with a given DEVSA file
are not damaged by the creation of these multiple files. It is not necessary that
separate copies of the metadata be made for each copy of the DEVSA; only the
linkages must be maintained.

One PDL can reference and act upon multiple DEVSA. Multiple PDLs
can reference and act upon a given DEVSA file. Therefore the data model takes
special care to maintain the metadata to DEVSA file linkages.

Referring now to Figs. 4-10, an alternative discussion of images 301 is
discussed in order to demonstrate how the process can appear to the user in one
element of how a user can "edit" DEVSA by changing manner in which it is
viewed without changing the actual DEVSA as it is stored. Li Fig. 4, a user has
uploaded via upload modules 310A a series of videos that are individually
characterized with a thumbnail image, initial deep tagging and metadata. The first
page is shown.

hi Fig. 5, options ask whether to add a video or action to a user's PDL (as
distinguished from a user's EDL), and a user may simply click on a "add"
indicator to do so. Multiple copies of the same video may be entered as well
without limit.

hi Fig. 6, a user has added and edited three videos of his or her choosing to
the PDL and has indicated a "build" instruction to combine all selected videos for
later manipulation.

hi Fig. 7, an edit display page is provided and a user can see all three
selected videos in successively arranged text-like formats with thumbnails via 320 equally spaced in time (roughly) throughout each video. Here 2 lines for the first 2 videos and 3 lines for the third video just based on length. Here at the beginning and end of each video there is a vertical bar signifying the same and a user may "grab" these bars using a mouse or similar device and move left-right within the limits of the videos. A thin bar (shown in Fig. 7 about 20% into the first thumbnail of the first video) also enables and shows where an image playback is at the present time and where the large image at the top is taken from. If the user clicks on PLAY above, the video will play through all three videos without a stop until the end thus joining the three short videos into one, all without changing the DEVSA data.

In Fig. 8, a user removes certain early frames in the second two videos to correct lighting and also adjusted lighting and contrast by using metadata tools. A series of sub-images maybe viewed by grouping them and pressing "Play."

In Fig. 9 the user has continued to edit his three videos into one continuous video showing his backyard, no bad lighting scenes, no boat, no "pool cage". It is less than half the length of the original three, plays continuously and has no bad artifacts. The three selected videos will now play as one video in the form shown in Figure 9. The user may now give this edited "video" a new name, deep tags, comments, etc. It is important to note that no new DEVSA has been created, what the user perceives as a new "video" is the original DEVSA controlled by new PDLs, and other metadata created during the edit session described in the foregoing. The user is now finished editing in this example.

In Fig. 10, a user has returned to the initial user video page where all changes have been made via a set of PDLs and tracked by storage module 313 for
ready playing in due course, all without modifying the underlying DEVSA video. His original DEVSA are just as they were in Fig. 4.

The present invention provides a highly flexible user interface and such tools are very important for successful video editing systems. The invention is also consistent with typical user experience with Internet-like interactions, but not necessarily typical video editing user interfaces. The invention will not place undue burdens on the end-user's device, and the invention truly links actual DEVSA with PDL.

Referring now to Fig. 11 an operative system 1100 for visual browsing, deep tagging, and synchronized comments comprises at least three major, linked components, all driven from central servers 1107 including (a) a plurality of user interfaces represented as user interface layer 1108 that is linked to a variety of end-user devices 1102 used by end-users 1101 (one is shown) via a plurality of data networks 1105 (one is shown), (b) an underlying programming model including the programming module 1115 operatively housing and controlling operative algorithms and programming, and (c) a data model or system encompassing operative modules 1112 and 1113 for manipulating and controlling stored, digitally encoded time-based media such as video and audio, DEVSA, and associated metadata.

Those of skill in the art should understand that, in the present embodiment, all actual video manipulation is done on the server. Thus, this concept depicted here envisions that a "desktop" or other user interface device need (at a minimum) only to operate Web browser software and its own internal video player and display and operating software linked to servers 1107 via the Internet or another suitable data network connection 1105. As an alternative embodiment those of skill in the art will recognize that the present system may be adapted to desktop
operations under special circumstances where Internet access is not available or desirable.

Thus, the operational and software architecture of Fig. 11 has a form very similar to that described in earlier Figs. 1, 2, and 3. The primary details described herein are beyond those described in the related applications listed above as cross-references occur within modules 1115 and 1113 and their interactions. The roles, actions, and capabilities of upload video 1110, encode video 1111, display control 1160, play control 1119 and DEVSA storage module 1112 are similar to those described in the discussion of the previous Figures.

Those of skill in the art should again understand that the PDL produces a set of instructions for the end-user device video player and display software and hardware. In the present embodiment, the PDL is generated on the server while the final execution of the instructions generally (but not always) takes place on the end-user devices 1102.

As a consequence, the present discussion results in "edit-type commands" including visual browsing elements, informational tags and synchronized comments becoming a subset of the metadata described earlier.

Those of skill in the art should further understand that while much of the discussion in this application is focused on video, the capabilities described herein apply equally to audio data. The capabilities would additionally apply to many forms of graphic material, and certainly all graphic material that has been encoded in video format. Other than time-dependent functions, these capabilities apply equally to photographic images and to text.

During common operation, a user 1101 interfaces with user interface layer 1108 and system environment 1107 via data network 1105 and pathway 1106. In a practical sense, a plurality of screen displays would be observed by the user.
1101 as user 1101 interacts with the functions operably retained within visual browsing 1115A, deep tagging 1115B and/or synchronized comments 1115C within programming module 1115.

During operation, as user 1101 interacts with the functionalities, features, and algorithms contained in programming module 1115, programming module 1115 interacts with metadata/PDL data storage 1113 both uploading information of user inputs and downloading information about the media and about other users' activities and information. The programming module 1115 also interacts with display control 1116 in the manner discussed previously to repeatedly create new displays of media in response to user inputs and according to algorithms and functionalities that respond to metadata (both new and previously stored). Each user's activities are tracked, analyzed and stored in metadata/PDL storage module 1113 as metadata and linked to the appropriate videos, the internal time within those videos, the user's group affiliations, and such other data as may be needed to carry out the functions described herein. Specifically, metadata/PDL data storage module 1113 will store the deep tags and synchronized comments created by each user 1101 and link those tags and comments to specific time intervals internal to the specified video or other time-based media.

Since multiple types of playback mechanisms are likely to be needed such as one for PCs, one for cell phones and so on; programming module 1115 will preferably create a "master PDL" from which algorithms, functionalities, and features can adaptively create multiple variations of the PDL suitable for each of the variety of playback mechanisms as needed. Here, as shown, the PDL is executed by programming module 1115 and will also operatively interface with user interface 1108 to obtain any needed information and, in turn, with the data model (See Fig. 2) which will store and manage such information.
During preferred operation, programming model 1115 retrieves information from the data model as needed and interfaces with user interface 1108 to display information to multiple users 1101. Those of skill in the arts of electronic programming should also recognize that programming model 1115 will optionally also control the mode of delivery, streaming or download or create fixed media such as DVD, of the selected videos to the end-user; as well as perform a variety of administrative and management tasks such as managing permissions, measuring usage (via known analysis modes including heat maps, dependency controls, etc.), balancing loads, providing user assistance services, etc. in a manner similar to functions currently found on many Web servers.

As noted earlier, the concept and overall design of the PDL along with the programming model and the data model when coupled with a suitable user interface extends smoothly to virtually any data type (text, photos, graphics, etc.) and is not limited to video or audio or other time-based media. In a more general form the invention described herein can be applied to any data type. (Hi the following the terms "web site" and "desktop" are used as terms of convenience to reflect current day experience. The "desktop" might well be a cell phone. Those of skill in the art should recognize that virtually any variety of client/server arrangement would act in the same manner when implemented following the new methods, models, tools, et al. introduced herein.)

1. A web site stores multiple data files in one or more fixed formats.
2. Each user of that web site can create a set of metadata about those data files which controls the way those files are displayed to that user and to others.
3. The website (not the desktop) uses the metadata to control how the data is displayed to a viewer without changing the original data file or the metadata.

4. The viewer selects or allows the server to select for him which set of metadata is to be used to display data to him.

5. The data can be streamed or downloaded to the user or used to create permanent media such as DVDs.

6. No special software resides on the desktop.

7. The desktop does have standard software such as web browsers, video players, etc. which will execute instructions sent from the server.

The present invention also considers specific extensions to the editing and viewing models discussed herein. A user who is editing a file can choose to create multiple virtual versions targeted at multiple sub-audiences. These multiple versions would represent distinct metadata but would not change the underlying DEVSA. Two examples will illuminate this capability.

a. An editor may choose to permit members of his club to see all of a video while allow public users to see only a defined subset of the video.

b. An editor may determine that some scenes in a video would not be suitable for cell phone users because they require better screen resolution than is now available on cell phones. The editor could then create a "cell phone users" version of the video to be viewable to cell phone users plus a "desktop users" version to be shown to those with higher resolution displays.

Those of skill in the art of designing computer systems for video media will recognize some of the substantive advantages the present invention has over the related art. These include the following:
(a) Since edit commands as employed herein including "cuts", "fades", etc. do not change the underlying DEVSA, at some later time the editor can decide that what had been cut is in fact of value and retrieve it and reuse it. Related art relies on the originator of the video to save original copies, i.e., the related art, edited content is saved "as edited" thus causing the original information to be lost unless the user specifically saves it a special file. An alternative approach found in the related art would be for a system to save encoded versions of all original and all edited files. While this would accomplish the goal of preserving information, it would result in extremely high data storage expense, very high encoding loads, and major data management challenges in tracking version changes.

(b) The same edit information can be tied to multiple copies of the DEVSA, which can be encoded in the same or in diverse forms. The edit information can be changed without requiring changing the encoded DEVSA. I.e., related art, most edit commands affect the actual video, audio and other data itself so that each copy of the DEVSA incorporates the edit commands. Hence in the related art any changes to the edit commands requires that all DEVSA files be re-encoded which represents a major expense, administrative overhead and potentially significant time delay.

As another benefit of the present invention, since edit commands, deep tags, titles, etc. are stored separately from the underlying DEVSA, it is possible to re-encode the video, audio and other information and/or to introduce different encoding and/or decoding technologies without loss of the edit information created by editors. I.e., related art, most edit commands affect the actual video,
audio and other data itself. Thus each edited file would require individual re-
encoding if it were desired to introduce a new or different encoding technology.

A major challenge for any large data center operation is creating and
maintaining back-up copies of all data files. This can be a fairly elaborate and
extensive process. Each time a new data file is created or an existing file is
modified, a new back-up copy needs to be created and maintained. Since the
invention herein allows many new and changed edits to a DEVSA without
changing the DEVSA and since the DEVSA is a large and complex data file, the
complexity and cost such a data back-up process is substantially reduced, hi the
related art, a much larger number of DEVSA files would be created resulting in
increased complexity and cost of operations.

A second major challenge for large Web site operations is the need to
operate data centers at multiple locations in order to improve both performance
and reliability. In order to operate data centers at multiple locations data must be
replicated and synchronized across those locations as well if all the advantages are
to be gained. hi an advantage similar to that found in data back-up, the invention
herein allows many new and changed edits to a DEVSA without changing the
DEVSA and since the DEVSA is a large and complex data file, the complexity
and cost of such a data replication and synchronization process is substantially
reduced. In the related art, a much larger number of DEVSA files would be
created resulting in increased complexity and cost of replication and
synchronization of DEVSA files.

Those of skill in the art will recognize that the present invention enables at
least the following commercial uses: 1. The invention is useful in a web-based
personal video sharing system in which users can edit their own or other users'
videos into new videos for sharing via the web site or publishing to blogs or to
other websites; 2. The system could be used with commercial content by consumers to make "mixes" of movies or music videos; and 3. Video journalists could quickly edit a report from the field based on video they uploaded as well as stock footage from online libraries to produce a broadcast copy without damaging any of the original source materials.

In view of the above disclosure and with the entire disclosure as a supporting structure, the focus of the present invention consists of three major, linked components, all driven from central servers: (1) A series of user interfaces; (2) An underlying programming models and algorithms; and (3) a data model.

It is envisioned, that in an initial implementation all actual data manipulation and management is done on the servers. The "desktop" or other user interface device needs only to operate Web browser or similar software, a suitable video and audio player, and its own internal display and operating software and be operatively linked to the servers via the Internet or another suitable data connection. As advances in consumer electronics permit, other implementations become feasible. In those alternative implementations certain functions can migrate from the servers to end-user devices or to network-based devices without changing the basic design or intent of the invention.

The resulting visual browsing segment selection, deep tags, synchronized comments, etc. become a subset of the metadata described in the data model application incorporated herein by reference. The programming model and data model used herein follow the same model employed in the video editing and data model application noted above and incorporated herein by reference.

Much of the discussion in this application is focused on video because it is a well-known example of time-based media. The capabilities described herein apply equally to audio. They would also apply to many forms of graphic material,
certainly all graphic material which has been encoded in video format including
animation such as cartoons. Other than time-dependent functions, they apply
equally to photographic images and to text.

An important component of a successful video browsing, deep tagging and
synchronized commenting system is a flexible user interface which: (1) is
consistent with typical user experience but not necessarily typical video editing
user interfaces, (2) will not place undue burdens on the end-user's device, and (3)
is truly linked to the actual DEVSA.

A major challenge to be overcome is that the DEVSA is a four
dimensional entity which needs to be represented on a two dimensional display, a
computer screen or the display of a handheld device such as a cell phone or an
iPod®.

A. Component 1: Visual Browsing for Previewing and Viewing the
Video:

This discussion of visual browsing takes the approach of creating an
analog of a text document made up, not of a sequence of text characters, but of a
sequence of informational sequence indicators which allow the user to perceive
the progression through time of one or more DEVSAs on a two-dimensional
display. It must be noted that although the display is two-dimensional, the time-
progression is one-dimensional. The "thumbnail" frame images at selected times
throughout the video used as examples in the accompanying Figures 4 - 10 are a
valuable example but many other examples will come readily to mind to those of
skill in the art. Examples include icons representing scene changes detected by
discontinuities in the video stream, images reconstructed by image recognition or
icons generated by voice or other sound recognition. Sound bites might work well for audio. Deep tags or synchronized comments as discussed herein may very as informational sequence indicators in many circumstances. The fundamental point remains of taking the one-dimensional time progression and presenting it as a one dimensional sequence of informationaly useful indicators to enable a user to easily and quickly find those portions of a DEVSA of interest and value to him or her at that time. Such indicators can be of any data type which can be stored as metadata or created/or by appropriate process at or near display time. In the following discussion, "thumbnails" will continue to be used as examples with no intent to limit the extension to other types of informational sequence indicators.

For users who express the English language as a preference, these thumbnails are displayed from left to right and, if the display allows and the user chooses, in sequential rows flowing downward in much the way English text is displayed in a book. (Other sequences will naturally be more appropriate for users whose written language progresses in a different manner.

The point is to have the thumbnails follow a sequence similar to that of the user's written language or some other pattern with which the user is comfortable. Images flowing right to left, bottom to top for users who are more comfortable with such an arrangement is/are a minor adjustment.). A selected frame may be enlarged and shown above the rows for easier viewing by the user as was shown in Fig. 7 for example.

As an example, a 5-minute video might be initially displayed as 15 thumbnail images spaced about 20 seconds apart in time through the video. This user interface allows the user to quickly grasp the overall structure of the video. The choice of 15 images rather than some higher or lower number is initially set
by the server administrator but when desired by the user can be largely controlled
by the user as he/she is comfortable with his/her current user device's screen
resolution and size of the thumbnail image.

By use of a mouse (or equivalent) or keyboard commands, the user can
"zoom in" on sub-sections of the video and thus expand to, for example, 15
thumbnails covering 1 minute of video so that the thumbnails are only separated
by about 4 seconds.

Whenever desired, the user can "zoom-in" or "zoom-out" to adjust the
time scale to meet the user's current editing or viewing needs. One approach is
the so-called "slider" wherein the user highlights a selected portion of the video
timeline causing that portion to be expanded (zoomed-in) causing additional,
more closely placed thumbnails of just that portion to be displayed.

Additionally, other view modes can be provided, for example the ability to
see the created virtual segment in frame (as described herein), clip (where each
segment is shown as a single unit), or traditional video editing time based views.

Additional methods known to those of skill in the art of displaying
thumbnails over time can also be used to meet specific user needs. For example
thumbnails may also be generated according to video characteristics such as scene
transitions or changes in content (recognized, e.g., via video or audio object
recognition).

1. Video Timeline Thumbnail Preview

A key component to the visual browsing user interface is providing the
user with the ability to jump to desired segments of video while viewing the
video.
Similar to the thumbnail previews outlined above, the user interface displays representative thumbnails of segments along a timeline, allowing the user to review and navigate to different segments of the video in a targeted manner, rather than randomly selecting points on a timeline in hope of finding segments of interest. The timeline preview also allows the user to simultaneously view the view in progress, while searching ahead/behind for other sections of interest.

Those of skill in the operational arts will recognize that this invention includes substantive advantages over the previous state of art. These include but are not limited to:

1. user controlled zoom in/out to control spacing of thumbnail images
2. automatic selection of thumbnail position based on internals of video such as scene changes and object recognition
3. creation of new thumbnails by clicking on timelines.

B. Component 2: Deep Tagging the Video and Displaying Deep Tags:

Both creators and viewers of content have the ability to attach "deep tags" - personal labels or guides - to segments of video, that is specific time intervals within a video, as distinguished from "tags" on the entirety of a video. The term "deep tag" is meant to indicate specifically that the informational "tag" applies only to a specific time interval within a video or, more generally, a DEVSA, not to the video or DEVSA as a whole. (Naturally the system would allow users to 'tag' entire videos but such tags would not fall under the current discussion.) Users will be able to view previously entered segment deep tags along a video timeline and enter new deep tags if desired. When entering a deep tag, the user can highlight a thumbnail or a range of thumbnails or a portion of the timeline in
order to tie his deep tag to that thumbnail or range of thumbnails. When previewing a video, the system will display all deep-tagged segments, allowing the user to jump directly to that video segment. Additionally, while viewing a video, deep tags associated with different segments will be displayed within the informational sequence indicators also allowing the user to browse / navigate to different segments of the video.

It is important to note that deep tags which are entered become part of the Playback Decision List (PDL) described in detail above. Even more importantly, by establishing a deep tag on a heretofore untagged segment or on an over/underlapping segment a user has effectively edited the video for viewers using that deep tag because, if so requested, the PDL will now playback the video according to the new deep tag rather than other deep tags which may have been created. This capability may be referred to as "virtual editing" since the effect on at least some future viewers (those who follow this user's deep tags) is as if the video has been edited but no change has been made to the underlying DEVSA nor to any other user's deep tags or virtual edits.

The following example explains this virtual edit capability further.

a. User Smith uploads a video named "roller".

b. User Jones views "roller" and deep tags 5 segments as "exciting".

c. User Jones tells his friends to view "roller" via "deep tags by Jones which say 'exciting'".

d. Via the PDL mechanism, the friends will see just those segments which Jones deep tagged as 'exciting'.

e. Thus Jones effectively has edited "roller" into a sequence of highlights of his own choosing.
f. Jones activities regarding "roller" have no effect at all on any activities which may have been or will be taken by Smith or Williams or any other party with respect to "roller". Each user can perform his or her own "virtual edit" via this mechanism.

g. The DEVSA associated with "roller" is not affected, only metadata concerning "roller".

h. As will be clear from the discussions accompanying Figs. 4 - 10 and from applicant's related application concerning editing time-based media which is incorporated herein by reference, that fact that "roller" may have been created from multiple DEVSA using such an editing process does not limit the effectiveness of the above described processes nor does it require any changes to the component DEVSAs which were utilized to create "roller".

1. Deep Tagged Segments as Descriptive Content:

Deep tagged segments also provide additional information about the content for the viewer. For example, while watching a video of a person's trip to Europe, the creator or subsequent viewers of the content may choose to deep tag different segments of the video with location descriptors such as "England" and "France." As the viewer is watching the video, these deep tags will be displayed to the viewer, providing further context for segments of the video regardless of whether the viewer chooses to jump ahead / behind within the video. Such additional deep descriptive content provides greater content and context for the user while choosing whether the video is of interest to him or her.

   Users may choose which deep tags they wish to see by standard database access means such as "only the creator's deep tags", "only my interest group's
deep tags", "only my deep tags", etc. Users may also choose deep tags across multiple videos. For example: "Show all segments of all videos with deep tags matching 'beach+Naples'."

A user may also control which other users may see the deep tags entered by that user via a set of permission controls.

Creators and managers may block others from adding deep tags to videos he created via a set of permission controls.

All deep tags entered are received by the programming module from the user interface and stored in the data model as metadata linked to the DEVSA at a specific time point within the DEVSA without any change in the DEVSA itself. Deep tags are naturally hierarchical and much value can be gained from that hierarchy such as /sports/MA/Brockton/soccer/tigers/mcpherson/goals. Similarly the programming module retrieves deep tags from the data model and delivers them to the user interface for display when the appropriate user interface calls are entered.

As used herein, deep tags are searchable data and are identified by the user who entered the deep tag and his profile unless the user denies permission to make that information available to others.

The deep tagging ability described here is unique in that any user, not just the creator, can make new deep tags on any segment of the video and, because of the virtual edit capability, each user can create his own "highlight" version of any video (or "edited" videos) with no change to the video(s) and with no change to any other user's highlights. In all previous references, a new DEVSA would be required in order to accommodate each user's new "highlights".

The ability to introduce deep tags, that is tags tied to specific time intervals within DEVSA, whether unedited, edited by traditional means, or edited
by the novel means described herein and in related applications, actually introduces and implements a new information type and concept unknown in the current or prior art. Essentially deep tags provide an easily searchable database entity, typically, but not necessarily, text, and link into an ordered path in the four dimensional space of a large number of heavily encoded digital videos with synchronized audio.

C. **Component 3: Synchronized Commenting on the Video and Displaying Comments**

It is common on the Internet, in chat rooms and the like, for users to share their comments with others. Typically, however, those comments are about a well-defined, fixed object, an event, a person, a photo, a piece of text or some thing else which, at least for some interval, is fixed in time. Comments of this type are referred to herein as Fixed Comments.

Video however extends in time so many comments make little sense when applied to a video as whole but must be tied in synchrony to some particular segment or even a specific short interval of time within the video. Thus, comments on video or other time-based media (more generally DEVSA) have fundamentally distinct properties from the usual comments about objects fixed in time and should be understood to be a new type of information. For purposes of clarification, comments synchronized in time with video will be called "synchronized comments".

Synchronized comments should not be confused with comments on live, ongoing events. The latter might occur for instance in a chat room during a baseball game where users will make comments on the game during the game or
at least when the game is broadcast when they are all watching the game at the same time. Comments of this type are referred to herein as Live Comments.

The synchronized comments addressed herein are made by users at distinct times because they are not watching the videos at the same time. Thus, the comments are synched to the time internal to the videos themselves not to any absolute time frame. Subsequent viewers see previous comments synchronized to the time internal to the video independent of when the comment was entered. As an option, the calendar time at which a "synchronized comment" was made may be noted and stored for the sake of reference. Additionally, calendar time can be utilized as a parameter to control how and when the synchronized comment is to be displayed or otherwise utilized.

At the commenter's option, synchronized comments can be tied to a chosen thumbnail or to the time line of the video so that the synchronized comment appears as the video is played at the time of that thumbnail or at the point in the timeline when it was entered.

User comments can also be tied to an entire video. In such a case they are not synchronized comments but rather Fixed Comments. Synchronized comments can be written or oral, that is they can be in any form transmissible and storable via then current data networks and servers.

Synchronized comments differ from deep tags in that synchronized comments do not directly identify segments of video but rather add information that is available to the viewer.

Synchronized comments are searchable data and are identified by the user who entered the synchronized comment and his profile unless the user denies permission to make subsets of that information available to others.
The synchronized commenting ability described here is unique in that any user, not just the creator, can make new synchronized comments on any segment of the video and, because of the virtual edit capability, which applies to synchronized comments in the same manner as it does deep tags, each user can create his own "highlight" version of any video with no change to the video and with no change to any other user's highlights. In all previous references, a new DEVSA would be required in order to accommodate each user's new "highlights". The synchronized comment capability actually introduces and implements a new information type and concept unknown in the current or prior art.

The present invention enables substantive uses, and these include:

(A) Application in multiple implementation structures to perform functions such as those described in the above paragraphs: Implemented as a web site employing a user interface, programming module and data model such as described above and in related patent applications.

(B) Application implemented with functionality primarily on end-user devices with digital video recording capabilities (examples are digital video recorders or personal computers) wherein DEVSA arriving at the end-user device could be linked to PDLs before it arrives with time-progress indicators, deep tags, synchronized comments, etc. regarding its content and the user could use the invention to control playback of the DEVSA in the manner described previously. The user also could add time-progress indicators deep tags and synchronized comments or Fixed Comments and have those additions to the metadata sent via data networks to other users in a manner similar to that done on the Internet.

As illustrative examples, implementation (B) would provide system for a cable TV company to download a pay-per-view movie to a DVR, and:
1. To employ PDLs and user specific permissions to allow different displays of the movie for different users such as an X-rated version for adults and a G-rated version for others.

2. To employ synchronized comments incorporating a variety of closed caption language translations as the user requests: Ukrainian, Japanese, English, etc.

3. To employ deep tags to provide expert commentary on parts of the movie.

4. To provide time sequence indicators to assist viewers in visual browsing of the movie.

5. To employ a multitude of forms of metadata as discussed herein to permit users to choose alternative playing modes of the movie such as is possible with certain DVDs including alternative endings, differing sound tracks, etc.

Implementation (B) would further permit users to generate such PDLs, synchronized comments and deep tags to accomplish the above. For instance, parents could employ PDLs and user-specific permissions to edit movies themselves prior to allowing their children to watch them.

(C) A mixed implementation wherein DEVSA is delivered to end-user devices via distinct networks or the same networks as time-progress indicators, deep tagging and synchronized comment and Fixed Comment information. (E.g., DEVSA is delivered via cable TV, satellite or direct broadcast while time-progress indicators, deep tagging and synchronized comment and Fixed Comment information is delivered and sent via the Internet. Due to the special capabilities of this invention, especially the logical separation of the metadata from the DEVSA, a unique identification of the DEVSA plus a well-defined time indicator
within the DEVSA is adequate to allow the performance of the functions
described herein.) This implementation "C" has the advantage of more easy
integration of traditional broadband video distribution technologies such as cable
TV, satellite TV, and direct broadcast with the information sharing capabilities of
the Internet as enabled by the current invention.

As illustrative examples, implementation (C) would provide mechanisms
for general Internet users to provide PDLs, synchronized comments and deep tags
to accomplish the same ends as those described for implementation (B), including
examples wherein:

1. A Finnish Film Society (for example) could provide via a
web site linked to the DVR, English translations for Finnish films which
would be displayed as synchronized comments as in example number (B)
2 above. These translations could be text or audio delivered via the
Internet to the DVR or alternatively to another user device.

2. A professional film expert could offer commentary on films
as the film progresses in the form of deep tags provided via a web site
linked to the DVR or alternatively to another user device.

3. A chat group's comments on the film could be displayed
synchronized with the progress of the film via a web site linked to the
DVR or alternatively to another user device.

In all examples (herein and elsewhere), since the DVR is linked to the
Internet, if the user pauses, fast forwards, etc., the DVR would provide
information to any linked Internet sites about the current time position of the
video thus keeping metadata and video synchronized.

(D) A mixed implementation as in "C" above with the addition that the
end-user devices such as digital video recorders make available individual usage
data such as view, fast forward, etc. as a function of time within each DEVSA and such usage data is made available to the programming module and data model as an additional form of metadata for processing, analysis, and storage and display via the user interface. A simple example of how such information might be used would be: If more than 80% of the last 1000 viewers fast-forwarded through this 45 second interval, it is probably boring and I should skip it.

As illustrative examples, implementation (D) would provide a system for users watching a football game or any other video being or having been recorded on a DVR to have the same kinds of capabilities illustrated with respect to (B) and (C) above, but in addition gain useful information from the actions of others who have watched the video and, in turn, to provide such information to subsequent watchers, including:

1. While watching a pre-recorded or partially pre-recorded football game many viewers will fast forward through time outs, commercials, lengthy commentaries, half-time, etc. Similarly, many viewers will repeat or slow-play interesting or exciting plays. Via capturing those multiple user actions through the Internet, analyzing that data and then distributing that analyzed data to subsequent viewers, at the user's choice, the fast forwarding could be done automatically using PDLs.

2. While watching the same football game viewers could press "thumbs-up" or "thumbs-down" type buttons, which are a form of deep tag, to signify interesting and non-interesting sequences. Via capturing those multiple user actions through the Internet, analyzing that data and then distributing that analyzed data to subsequent viewers, at the user's choice, only sequences with a high percentage of thumbs-up would
be shown thus enabling the user to watch "highlights" as selected by his predecessor viewers.

3. While watching the same football game viewers could enter text or iconic synchronized comments which would then be shared in a similar manner.

4. While watching the same football game viewers could enter Instant Messaging messages directed to specific friends which would appear as synchronized comments to those specific friends who watched the game later.

hi all examples, since the DVR is linked to the Internet, if the user pauses, fast forwards, etc., the DVR would provide information to any linked Internet sites about the current time position of the video thus keeping metadata and video synchronized.

Usage data could pass via one or more data networks, direct from said end-user device or via another of the user's devices such as a PC linked to the Internet and hence to the server wherein operates the programming module, etc. To the degree permitted by the DVR or similar device the programming module could provide signals to control both playback and user interface displays generated by the DVR. The fundamental point is to make use of both the DEVSA storage and data gathering capabilities of many individual end-user devices such as DVRs and, if available, their externally controlled playback and user interface capabilities, while making full use of the multiple user, statistical, centralized analysis and data management capabilities of the programming module and data model as described above.

A specific advantage to implementation D, and to a lesser extent implementation C, is that a DVR user who might be the 10,000th viewer of a
broadcast program has the advantage of all the experiences of the previous 9,999 viewers with regard to what parts of the show are interesting, exciting, boring, or whatever plus their time-progress indicators, deep tags and synchronized comments on what was going on.

In the claims, means- or step-plus-function clauses are intended to cover the structures described or suggested herein as performing the recited function and not only structural equivalents but also equivalent structures. Thus, for example, although a nail, a screw, and a bolt may not be structural equivalents in that a nail relies on friction between a wooden part and a cylindrical surface, a screw's helical surface positively engages the wooden part, and a bolt's head and nut compress opposite sides of a wooden part, in the environment of fastening wooden parts, a nail, a screw, and a bolt may be readily understood by those skilled in the art as equivalent structures.

Having described at least one of the preferred embodiments of the present invention with reference to the accompanying drawings, it is to be understood that the invention is not limited to those precise embodiments, and that various changes, modifications, and adaptations may be effected therein by one skilled in the art without departing from the scope or spirit of the invention as defined in the appended claims.
WHAT IS CLAIMED IS:

1. An electronic system, for at least one of a synchronized commenting and a deep tagging of at least a first time-based media by a plurality of users including at least a first user, said electronic system, comprising:
   - at least one user computerized electronic memory device enabling a manipulation of said time-based media;
   - user interface means for transferring said at least first time-based media from said computerized electronic memory device to means for encoding and for storing said at least first time-based media in at least a first initial encoded state in an electronic system environment;
   - metadata system means for creating, storing, and managing at least a first layer of time-dependent metadata in a manner associated with said at least first initial encoded state of said encoded time-based media without modifying said at least first initial encoded state of said encoded time-based media, and in a manner associated with each respective said user;
   - time sequence means in said metadata system means for generating a sequence of time informational indicators enabling each said user to perceive a useful progression through time of said at least first time-based media;
   - electronic interaction system means for enabling said plurality of users to interact respectively with said time sequence means and said metadata system means for creating, storing, and managing said at least first layer of metadata according to a plurality of stored respective playback decision lists of ones of said plurality of users; and
   - said electronic interaction system means including means for enabling a plurality of display control modes and a plurality of play modes of said encoded
time-based media according to said respective playback decision lists of ones of
said plurality of users.

2. An electronic system, according to claim 1, wherein:
said time informational indicators include at least one indicator selected
from a group consisting of:
visual icons, visual icons representing scene changes detected by
discontinuities in said time-based media, visual images reconstructed by
an image reconstruction system, tags generated by one of a sound or a
visual image in a recognition system, and thumbnail images, thereby said
at least one indicator enables a convenient visual browsing by said
plurality of users.

3. An electronic system, according to claim 1, wherein:
said electronic interaction system means for enabling said plurality of
users to interact respectively with said time sequence means and said metadata
system means, further comprises:
means for deep tagging time-based metadata and for attaching user
personalized deep tags to selected interval segments of said encoded time-based
media,
said user personalized deep tags including at least one tag type
selected from a group comprising:
user identification, user hierarchy, user-defined use modalities,
user descriptive comments reviewable by other users, user
instructions to jump to a particular selected sequence in a visual
browsing enabled mode, user-personalized sequence indicator
identifiers, electronic instructions to change a visual display
instruction of a selected sequence, and a system-searchable deep
tag available to other users.

4. An electronic system, according to claim 1, wherein:
said electronic interaction system means for enabling a plurality of users
to interact respectively with said time sequence means and said metadata system
means, further comprises:
means for enabling a plurality of user interactions, said user
interactions including at least one user interaction from a group
comprising:
editing, virtual browsing, tagging, deep tagging, commenting,
synchronized commenting, social browsing, granting of
permissions, and creation of a permanent media form linked to
respective said user modifications; and
said electronic interaction system for enabling enables a selective storage
of each respective users' interaction in respective user playback decision lists.

5. An operational system, for providing a web-based system enhancing a use
for at least one of a plurality of users of time-based media, comprising:
means for receiving via a user interface system a user-transferred time-
based media in an electronic operational environment including an electronic
memory device and a user interface subsystem;
means for encoding said uploaded time-based media and for storing said
encoded time-based media in an initial state;
a metadata creation system for establishing metadata associated with said
uploaded time-based media;
means for providing a system of sequenced time informational indicators enabling said user to at least one of a visually and an audibly perceive a progression through time of said encoded time-based media;

an electronic interaction system enabling said at least one user to modify said established metadata associated with said encoded time-based media in at least a first stored playback decision list via a communication path including said user interface system, whereby said stored playback decision list of said at least one user modifies said established metadata without modifying said encoded time-based media in said initial state;

said electronic interaction system including a display control system and a play control system enabling said at least one of said plurality of users to display and play said encoded time-based media in a modified manner according to said at least one playback decision list without modifying said encoded time-based media; and

said electronic interaction system enabling others of said plurality of users to modify said metadata in respective user-linked playback decision lists and for storing each respective user playback decision list separately.

6. An operational system, according to claim 5, wherein:
said user modifications include at least one user modification from a group comprising:
editing, virtual browsing, tagging, deep tagging, commenting, synchronized commenting, social browsing, granting of permissions, and creation of a permanent media form linked to respective said user modifications; and

said electronic interaction system enabling storage of each respective user's modifications in respective user playback decision lists.
7. An operational system, according to claim 6, wherein:

said user modifications include at least said synchronized commenting;

and

said electronic interaction system includes means for enabling each respective user to view respective user's synchronized comment modifications and separately store additional synchronized comments, whereby said electronic interaction system enables at least one of an enhanced multiple user synchronized commenting and a multiple level deep tagging as an enhanced system performance.

8. An operational system, according to claim 7, wherein:

said permanent media form includes at least one of said time-based media in said first encoded standard and established metadata associated with said individually established user playback decision list, whereby said operational system enables convenient transfer of each said stored time-based media.

9. An operational system, according to claim 8, further comprising:

means for capturing and storing metadata and playback decision list data relating to said time-based media enabled for tracking respective user's editing of said initial state encoded time-based media;

programming module means for controlling at least one of a display control, a play control, and a multi-level user editing of said time-based media;

and

user transfer means for enabling multiple user playback interactions with said encoded time-based media and selected ones of said user's playback decision lists, whereby said electronic interaction system includes means for storing said at least one of said user modifications as a synchronized comment for at least an initial portion of said initial user-transferred time-based media, whereby said
operational system enables comments related to at least a part or all of said transferred time-based media.

10. An operational system, according to claim 8, wherein:

said means for capturing and storing includes means for directly identifying user-defined portions of time-based metadata, whereby said means for directly identifying includes one of means for adding comments and means for adding user information, whereby said system enables enhanced utility of said initially uploaded time-based media.

11. A method for operating a media system providing a web-based system enhancing use of time-based media by at least one of a plurality of users, comprising the steps of:

providing means for receiving via a user interface system a user-transferred time-based media in an electronic operational environment including an electronic memory device and a user interface subsystem;

providing means for encoding said uploaded time-based media and for storing said encoded time-based media in an initial state;

providing a metadata establishing system for establishing and managing separate time informational sequence indicators in metadata associated with said transferred time-based media;

providing an electronic interaction system enabling said at least a first user of a plurality of users to modify said established metadata associated with said encoded time-based media in at least a first stored playback decision list via a communication path including said user interface system, whereby said first user stored playback decision list modifies said established metadata without modifying said encoded time-based media in said initial state;
providing said electronic interaction system including a display control system and a play control system enabling at least one of said plurality of users to display and play said established time-based media in a modified manner according to said at least one playback decision list without modifying said encoded time-based media;

said step of providing said electronic interaction system enabling others of said plurality of users to modify said established metadata in respective user-linked playback decision lists and for storing each respective user playback decision list separately.

12. A method for operating a media system according to claim 11, wherein:

said user modifications include at least one of a group of user modifications comprising:

editing, virtual browsing, tagging, deep tagging, commenting, synchronized commenting, social browsing, granting of permissions, and creation of a permanent media form linked to respective said user modifications; and

said electronic interaction system enabling storage of each respective user's modifications in respective user playback decision lists.

13. A method of operating, according to claim 12, wherein:

said user modifications include at least one of synchronized commenting and deep tagging.

14. A method of operating according to claim 13, wherein:

said user modification is said synchronized commenting; and

said step of providing said electronic interaction system includes a step of providing means for enabling each said respective user to view other respective users' synchronized comment modifications and separately store synchronized
additional comments, whereby said electronic interaction system enablesenhanced multiple user synchronized commenting as an enhanced system
performance.

15. A method of operating, according to claim 12, wherein:

said permanent media form includes at least one of said time-based media
in said first encoded standard and established metadata associated with said
encoded time-based media modified according to said at least one stored user
playback decision list, whereby said operational system enables transfer of each
said stored time-based media.

16. A system for providing enhanced time-based media editing, comprising:

a computer system receiving at least a first of a plurality of user transfers
of said time-based media in an operational environment through a user interface
system;

means for encoding said at least first of said user transfers of said time-
based media in an initial state separate from subsequent user transfers;

computer memory means for storing said encoded first time-based media
in said initial state separate from said subsequent user transfers;

a metadata creation system for initially establishing metadata associated
with respective user transfers of time-based media

said computer memory means storing said established metadata associated
with said time-based media separate from said encoded time-based media in said
initial state;

means for individually modifying said established metadata as an
individual playback decision list and for individually storing said playback
decision list separately from said respective initial state time-based media and
said respective initial metadata, thereby enabling an individual modification of
respective said playback decision lists without a modification of said initial state encoded time-based media;

means for enabling at least one of a visual browsing, a deep tagging, and a synchronized commenting regarding electronic time-based media content,

at least a first user interface means;

at least a first underlying programming module for enabling interacting with said at least a first user; and

an interactive data model constructing and tracking a user modification and review of each user action relative to said at least one of a visual browsing, a deep tagging, and a synchronized commenting within respective user playback decision lists.
FIG. 2
FIG. 6
FIG. 11

User Interface Layer 1108

Motionbox System Environment

End User 1101

User Devices 1102

Partial User Interface, Program Logic and Local Storage of DEVSA 1103

Upload / Display / Interact / Control 1104

Data Network 1105

DEVSA & PDL appropriate to end user device 1106

User Interface Layer 1108

Encode Video 1111

1110A

Upload Video 1110

1114

Metadata / PDLs 1113

1121

Programming Module

Visual Browsing 1115A

Tagging 1115B

Synchronized Commenting 1115C

1115

1112

DEVSA 1112

1119A

Play Control 1119

Retrieved Thumbnails 1120

Display Control 1116

1118