LiquidDVD™ – A DVD-Video Authoring System

Afzal Hossain and Nasima Parveen
Nanova Corporation
4962 El Camino Real, Suite 101, Los Altos, CA 94022, U.S.A.

Abstract

LiquidDVD™ is a software system for DVD-Video and DVD-ROM title development for commercial market. This paper describes LiquidDVD system architecture and issues we faced during its design and development. DVD format supports video playback sequences with multiple audio and sub-picture tracks. Playback sequences may contain multiple camera angles and can have different parental control ratings at different sections. DVD format also allows menu driven access to different points in a playback sequence with a good magnitude of flexibility. Several searching mechanisms are also provided in the format. These features are implemented through a set of data structures described in the DVD-Video Book [1]. A DVD-Video title can be developed by creating the low level data structures and then tying them up to create the desired playback sequences. Knowledge of DVD-Video data structures is needed to create a title this way. Most available commercial authoring systems use this approach. On the other end, the user can look at the playback sequence from the presentation point of view. The user describes the intended presentation, an authoring system then determines an appropriate way to implement the sequence using DVD-Video data structures. A minimal knowledge of DVD data structure is required to author a title following this approach. LiquidDVD uses this approach. The format that the user use to describe a video playback sequence is known as Movie Description Format (MDF). This paper also describes MDF and how decisions are made to create the internal DVD data structures.

Key words: DVD, MPEG, AC-3, compression, authoring, MDF.

Introduction

The term DVD Authoring is used to mean all pre processing and formatting of elementary assets such as MPEG-2 [8, 17] video streams, AC-3 [12, 17] audio streams, JPEG or bit-mapped still images and sub-titles into DVD-Video data structures. Figure 1 illustrates the authoring process. The pyramid at the left in Figure 1 represents the hierarchy of the DVD data structure. The lowest level is a completed disc from a mastering and replication facility. This disc contains DVD-ROM physical structure [2]. At this level, data structure contains Logical Blocks, ECC Blocks, Sides and Layers. UDF-Bridge File System data structures is developed on the physical layer. UDF-Bridge is a combination of file systems that supports ISO 9660 File System [4,14] and UDF File System [5,6,15] simultaneously. UDF-Bridge allows a disc to be read by older computer file system based on ISO 9660 and encourages newer implementations to use UDF File systems. DVD-Video data structures are stored in UDF-Bridge File System format. A full-length feature film can be laid out on a DVD disc using DVD-Video data structures.

The right side of Figure 1 illustrates the authoring process. A feature film content and the way it would be viewed, is the input to the process. The first phase is the pre-processing and formatting of all input data into DVD-Video data structures. All DVD-Video data structures that make up the desired content are written out in several binary files. These files are then used to create an UDF-Bridge File System Structure. Physical sectors to be assigned to all DVD-Video files are dictated by the DVD-Video specification. All files containing DVD-Video data structures are laid out on consecutive sectors. These sectors form a DVD-Video zone on a disc. There may be other files on the same disc. Sectors allocated to these files form the DVD-Other zone. Completed UDF-Bridge File Structure can be laid out on physical sectors on a DVD disc for mastering and replication.

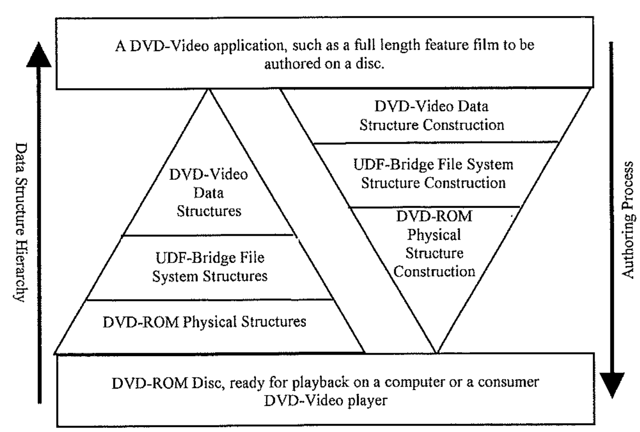


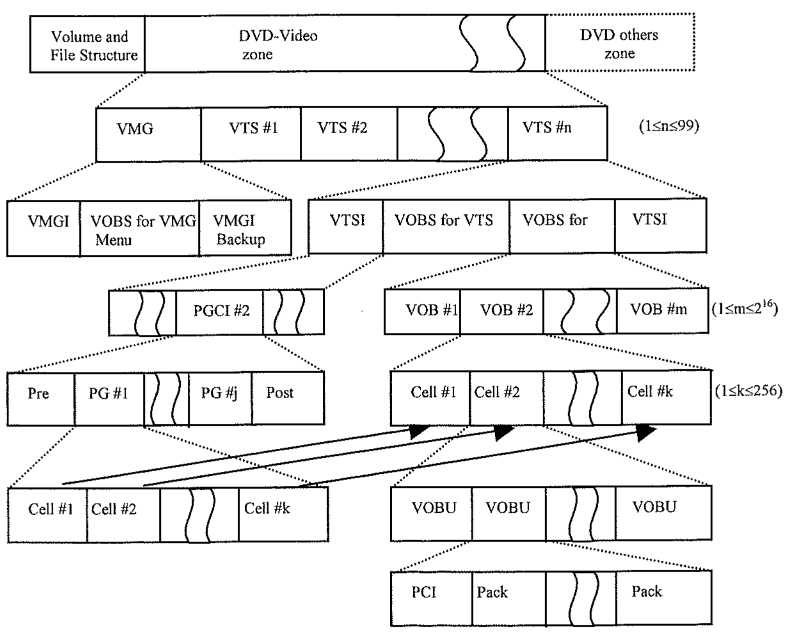
Figure 1: DVD-Video data structure hierarchy and authoring steps

The specification for the DVD-ROM physical format is given in [2]. Figure 2 illustrates the DVD-Video data structure. At the highest level, a DVD-Video disc's UDF-Bridge file structure has a DVD-Video zone and an optional DVD-Other zone. DVD-Video zone starts with a section called Video Manager or VMG which is followed by one or more Video Title Sets or VTSs. Data in the DVD-Video zone is divided into two groups - Presentation data and Navigation data.

Presentation Data is the data to be played back for video, audio, and sub-picture. The lowest level data structure of DVD disc is logical block. Each logical block, also known as pack, is 2048-bytes long. Presentation data and part of navigation data are divided into an integer number of packs. A number of presentation data packs and one navigation pack are multiplexed together to create a data structure called Video Object Unit (VOBU). An integer number of VOBU forms a Cell. During playback of a title, cells provide random access points in a title. One or more cells are logically combined to form Video Object (VOB) of a title. VOB conforms to Program stream described in [7]. The Video Object Set (VOBS) is a collection of VOBs.

Navigation Data controls the playback of presentation data. The main components of navigation data structure are, Video Manager Information (VMGI),Video Title Set Information (VTSI), Program Chain Information (PGCI), Presentation Control Information (PCI) and Data Search Information (DSI).

A DVD disc may contain upto 99 different titles. Each title is organized as Video Title Set (VTS) and has its own navigation menu allowing user to select different part of the title. The root menu which is used to branch to all the titles on the disc resides in the Video Manager (VMG).
Video Manager (VMG) includes Video Manager Information (VMGI), VOBS for Menu and backup for VMGI. VMGI includes attributes for the Menu and the titles, Title Search Pointers and Program Chain Information (PGCI) for Menu. Each VTS contains Video Title Set Information (VTSI), VOBS for title, VOBS for menu and backup for VTSI. VTSI contains attributes for title and menu, Part of Title Search pointers, PGCI for menu and PGCI for title. Program Chain (PGC) is the logical unit to present a part of or the entire title or a menu. A PGC is composed of one Program Chain Information (PGCI) and one or more VOBs. Each PGCI contains Navigation commands and one or more Programs. A program contains the order of cell playback to control the presentation of the VOB(s) of the title or the menu. PGCI describes the information on total presentation time of cells, order, playback time and position of cells constituting the PGC in a VOB. PGCI also includes the information on still time of PGC, cell and VOBU. Navigation Commands control the transfer of presentation among PGCs and may be executed at the beginning or at the end of a PGC or after the presentation of a cell.
Presentation Control Information (PCI) determines the condition of playback of Presentation data. PCI describes information on presentation start and end time of Video Object Unit (VOBU), angle information of a cell, highlight



as well as selection and activation information of sub-picture display. Data Search Information (DSI) is the Navigation data to search Video Object Unit (VOBU). DSI includes the information on Fast Forward/Fast Backward play of VOBU, seamless playback and synchronization information of audio and sub-picture data.

Navigation Commands are used to control the interactive features, such as, selecting a title, jumping to a particular cell, selecting language for audio, sub-titles, highlight and activation color of sub-picture etc. of a disc. Each command consists of a single instruction or a combination of two or three instructions. Content providers realize arbitrary branching structure of a title, language selection capabilities and other interactive features by using navigation commands and pre-defined number of general and system parameters.

Because of the quantity and diversification of the content that DVD disc can hold, DVD authoring can be a quite complex process. The extensive features and functionality of DVD-Video means that the detailed authoring is necessary to get the content correctly organized prior to mastering the disc. From user's input, LiquidDVD internally constructs the DVD-Video data structure and thus hiding most of the intricacies of the DVD-Video data structure from the user. While authoring a DVD-Video disc, user provides the contents (video, audio, sub-titles etc.), describes number of titles, number of random access points in each title, number of audio and sub-pictures and all the interactive features of the targeted disc in MDF file. LiquidDVD takes the information and presentation data and creates the DVD-Video data structures as described above to be recorded on the disc. The system also creates UDF-Bridge File Structure and provides means for verification of the file structure and DVD-Video structure.

System Architecture

LiquidDVD system is based on the Movie Description Format MDF. A graphical user interface (GUI) works as a front-end tool for the user to capture and create a movie description file using the MDF format. The user describes all the intended playback features of the targeted disc in this file. The user then invokes the formatter engine to perform various tasks including the final disc image creation. The system supports a good number of features. A job control file is used to control the engine operation to support these features. Major components of the encoding engine are as follows-

a) Stream Analyzer
b) Sub-picture Encoder
c) Still picture Encoder
d) Presentation Control Information Generator
e) Program Stream Multiplexer
f) Navigation Control Generator
g) UDF-Bridge Formatter

In addition, a DVD playback application is also a part of the system to assist verification of the created data structures. Figure 3 describes the system architecture.

Graphical User Interface

There are two major functions of the graphical user interface front-end. It lets the user create a description file of all possible playback sequences in a DVD-Video disc. The user can invoke the encoding engine for various features. This function is achieved by creating different job control files that the encoding engine understands. Major components of the Graphical User Interface are – Media Catalog Manager, MDF Editor, Menu and Button Editor, and MDF Writer.

The Media Catalog Manager is used to create a catalog of all assets used in the title development. Typical assets types are MPEG elementary video stream, AC-3 audio stream, LPCM audio stream, BMP or JPEG files for menu still pictures and sub-titles. Assets must be cataloged before using them in any playback sequence. Media Catalog manager can also check if asset properties conform to DVD specification.

MDF editor is used for creating movie description files. User uses a point-and-click method to describe the intended playback sequence. Number of audio and sub-picture streams, number of angles, parental management information and access points for each independent playback sequence (title) are entered in a database. Name and location of media assets as cataloged is also entered in a database. DVD menu for interactive playback is also described using Menu Editor. With all necessary information, the MDF Writer creates an MDF file for the encoding engine.

DVD-Video encoding engine

The encoding engine maps the playback behavior of a DVD disc described in MDF to DVD-Video data structures. All the presentation data must be in proper format and the mapping information must be available to encode the content to DVD-Video data structures. Video data is MPEG-1 [11, 17] or MPEG-2 encoded digital video, audio data can be linear PCM (Pulse Coded Modulation), AC-3 or MPEG audio [9]. Video and audio encoding is done separately and encoded video and audio bit streams are used directly as input to the encoding engine. Menus and sub-titles could be created as BMP, AVI or JPEG files and need to be encoded before the bit stream multiplexing. The first step of the encoding process is to analyze presentation data for DVD-Video 1.0

compliance. Stream analyzer procedure performs the bit stream compliance checking and also creates the information necessary during bit stream multiplexing stage. It checks the bit rate of encoded contents to be recorded on the disc against available disc space and total bit rate of the disc. It creates the mapping information, calculates the total number of Video Object Unit (VOBU) of each cell. Still picture encoder is a single frame encoder and encodes the still pictures for Menus as MPEG intra frames (I-frame). Sub-picture encoder is a variable run-length encoder and is used to encode sub-title sub-pictures. The input to this encoder is AVI file and output is run-length encoded sub-picture unit (SPU), a DVD-Video data structure. Presentation control information generator generates PCI and DSI navigation data structures and creates the navigation packs (2048-bytes long). Stream multiplexer combines encoded video, 1-8 audio, 1-32 sub-picture streams and presentation control navigation data together to a single data stream compliant with MPEG-2 Program Stream [7] and creates Video Object (VOB) files of the title. This process converts the presentation data to 2048-bytes packs, the lowest level data structure of DVD-Video. To guarantee the synchronous playback of the presentation data, correct value of System Clock Reference (SCR), Presentation Time Stamp (PTS) and Decode Time Stamp (DTS) values of the access units of the presentation data are needed and this process computes and encodes these values in the presentation data packs. It then multiplexes the presentation data packs and navigation packs and creates the VOBU data structure. Stream multiplexer also models the behavior of decoder buffers of the presentation data and manages them from overflow and underflow during playback of the presentation data. This stage computes the total presentation time of each cell of a title, address table of all VOBUs of a VOB for Navigation data VMGI, VTSI, PGCI etc. Control data generator creates VMGI, VTSI, and PGCI Navigation data structures of the disc. This process encodes the attributes of the title,

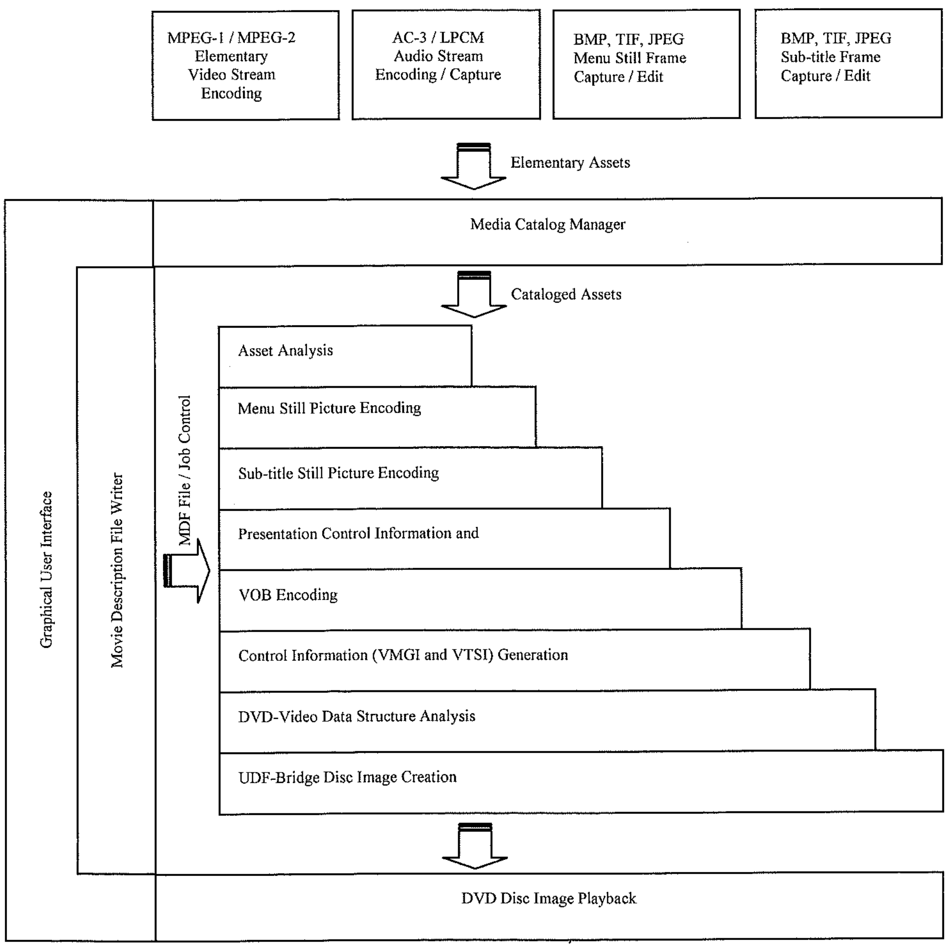


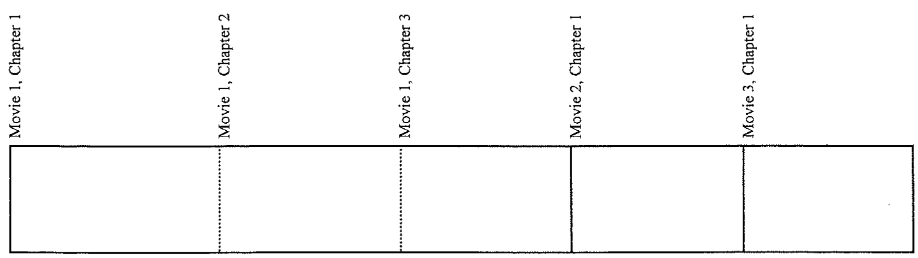
Figure 3: LiquidDVD™ System Architecture

parental and regional control and copy protection information. It creates the presentation structure PGCs and their links and First Play PGC of the disc. First play PGC is played as soon as a disc is inserted in a DVD player. This process also encodes all the interactive features of the disc and creates information and information backup files for VMGI and VTSI. Finally, Disc Image Creator combines the Video Object (VOBs) files and the control files, creates the UDF-Bridge file structure. The resulting file contains the image of the final disc to be mastered and replicated. The final disc image can also be played back by DVD disc image playback application. The user can analyze and check the correctness of the data structure created by stream multiplexer and control data generator by VOB analyzer function of the encoding engine.

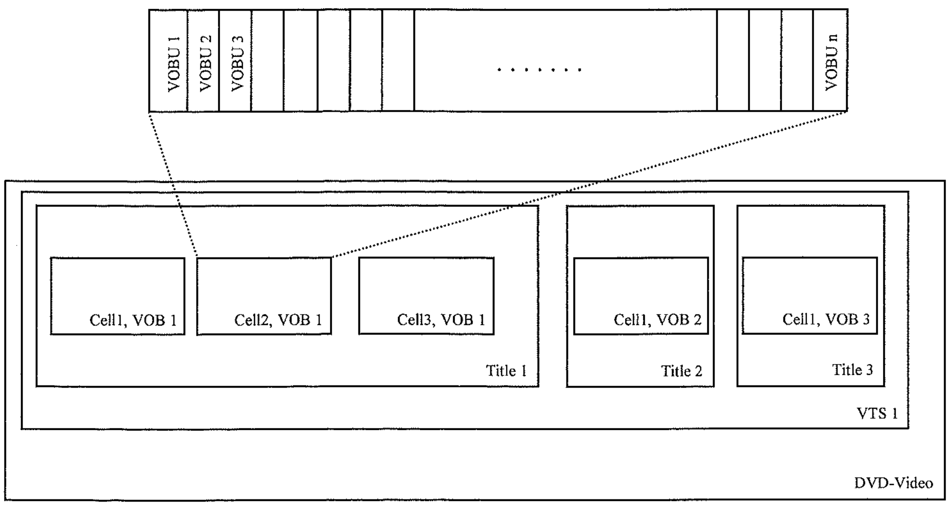
Movie Description Format

The content and the playback behavior are the most important items to an end user. There are quiet a number of parameters that affect the playback behavior of a disc. Few of them are – the number of independent video sequences, number of random access points and access methods, menus and the nature of interactivity, number of audio tracks, number of sub-title tracks, playback lengths and types, number of camera angles, parental control etc. Complete perception of a user about a disc can be used to define a disc in a sufficient manner. Based on this concept, MDF gives a formatted way of capturing the playback behavior of a disc. Because all playback features need to be supported using DVD-Video format, all MDF constructs can be mapped to one or more DVD-Video data structures. Figure 4 shows an example of mapping of movie description in MDF to the corresponding DVD-Video data structures. There are three major sections in an MDF file-

a) Sequence Description Section
b) Interactivity Description Section
c) Miscellaneous Disc Information Section.



(a) A sequence of three independent movies. Movie 1 has three chapters. Movie 2 and Movie 3 has one chapter each. This sequence can be described in an MDF file.



(b) A DVD-Video mapping of the movie sequence described in (a).

Figure 4: Movie description in MDF and corresponding DVD-Video structure.

The sequence description section describes all playback sequences in a disc. Number of independent movies, number of chapters and their lengths in each movie, number of audio and subtitle tracks, video, audio and subtitle attributes etc. are described in this section.

All menu structures are defined in the interactivity description section. DVD-Video format supports several menu structures. Complete behavior of each of Title Menu, Root Menu, Part of Title Menu, Angle Menu, Language Menu, Sub Title Menu are defined in this section. Specifications for the button positions, highlight and activation colors and button commands are described in this section.

A DVD disc includes attributes for Copy Generation Management Systems (CGMS), playback region information, analog copy protection information etc. Attributes of this category are included in the Miscellaneous Disc Information Section.

Implementation Issues

There are several ways a disc can be authored by a user. The user can build low-level data structures such as individual VOBs, PGCs etc. and define links between each PGCs to obtain desired playback behavior. Or, the user can deal with objects at a higher level, such as chapters of a movie or different multimedia clips and define how those are to be played back. It is also possible to author a disc with both capabilities. DVD data structure is complex in nature, and it takes significant amount of time to author a disc. People using commercial available software that follow the first approach have reported that it takes anywhere from several days to a month to develop a practical commercial title. Therefore, it made sense to shield the user from having to deal with data structures at the lowest level. This reduces control over implementation, so an optimum level of control over the authoring process also had to be found. We therefore decided to deal with higher level data structures at the same time have full navigation programming capability. This represents one of the key features in MDF.

When we decided to deal with the objects at a higher level, we also had to decide on an algorithm to map those objects and the playback behavior to DVD-Video structures. DVD-Video specification defines each structure very clearly, but it does not specify how to implement a particular playback feature using those structures. There are many different ways to implement the same playback features. Decisions, such as, the number of VTS, number of titles in each VTS are indirectly controlled by the user through MDF. The software dynamically does lower level mapping. We had to pay special attention to PGC mapping. One of the design goals was to make the PGCs transparent to the users. However, PGCs require pre and post navigation commands. Another goal - complete user level control of navigation commands requires that PGCs be visible to the users in MDF. We achieved both goals by mapping high-level data objects to PGCs, which are transparent to user, and remain as an underlying assumption in the MDF file. Users have control over the navigation commands but they see them as navigation between high level structures such as chapters or scenes in a movie or multimedia clip.

Description of menus in MDF became a daunting task. There are numerous ways to implement interactivity in a DVD-Video disc. The design goal is to give the user complete flexibility to design interactive menus. Because MDF deals only with high level data structures, it becomes necessary that all menus also deal with high-level data structures. All button commands in any Menu are defined in terms of high-level data objects. The user specifies presence or absence of a specific menu in the Root Menu or the Title Menu through MDF.

We wanted that the training time for new user to learn LiquidDVD system be less than two hours and the productivity of an experienced user be maximal. These goals required that the total number of concepts or constructs in MDF be minimal. We wanted least number of constructs in MDF without giving up the flexibility in authoring. We had to go over the list of features in DVD-Video format to determine the minimal set of controls in LiquidDVD for the user. One example is sub-title color palettes for each PGC can be different. Each palette can contain 16 colors of which only four can be used by a sub-title. Therefore, having color palette definition for each PGC separately appeared practically unnecessary. We let the palette definition at only one place. To address the need of demanding projects, we made individual palette definition optional. There are many such examples where a detailed description is made optional to keep the MDF file size small. Optional features can be turned on by the user but not necessary.

Each Video Object (VOB) is MPEG Program stream [7] which is constructed by multiplexing elementary streams (audio, video, sub-picture) and navigation data. Synchronization of playback of elementary streams and prevention of elementary stream buffers overflow and underflow during playback is the major issue of VOB creation. This issue has become more complicated due to support of multiple audio (1-8) and sub-picture streams (1-32) by DVD Video and physical placement of VOB on the disc. We developed an innovative algorithm to solve this issue.

Another issue has come from the restriction imposed by DVD-Video on the presentation duration of Video Object Unit (VOBU). The presentation period of VOBU is 0.4-1.0 second. When creating a VOBU, it is necessary to divide the presentation data such that the above limit is not exceeded. Instead of keeping track of presentation time during VOBU creation, LiquidDVD creates VOBU boundaries during elementary stream analysis step and uses this information to divide the presentation data during VOBU creation.

Synchronization of the display sequence of sub-pictures during playback as well as the synchronization of sub-picture display with video display has been an important issue. Sub-pictures are displayed during valid sub-picture display period. Certain restrictions imposed by DVD-Video on the placement of sub-picture packs in the multiplexed data stream have made this issue more complicated. We have used MDF to solve this problem. MDF defines the start and the duration of each sub-picture. LiquidDVD uses this information during VOBU creation to synchronize the display of sub-picture with the display of video.

To do fast forward or backward, start address of VOBUs at the interval of every 0.5-second presentation time is required. Presentation duration of all VOBU in a cell may not be same and actually varies from 0.4 second to 1.0 second. And this information needs to be updated after the end of every VOBU for all VOBUs including the most recently created VOBU of a cell. LiquidDVD solves this problem by saving the address of all VOBUs of a particular cell and at the end of cell creation, it calculates the address of all VOBUs relative to the current VOBU.

Presentation start and end time of VOB are included in DSI navigation data and are same for all navigation packs of a VOB. This is an issue because the presentation end time of VOB is available at the end of VOB creation. LiquidDVD has solved this problem by calculating the total presentation time of a title before starting to create VOB and uses it in all VOBUs.

Conclusion

For DVD to become a successful media for consumer application, millions more DVD titles must be developed with rapid pace. We have presented a solution to address the throughput issue in this process. We have developed several practical DVD titles using LiquidDVD system. Initial experience confirms throughput improvement. For a new user, most of the training time is used to understand DVD navigation commands. Very little time is spent to learn and use LiquidDVD system. Therefore, after the first title, the user takes only few hours to develop additional titles.

The system is especially good for full-length feature film conversion. This application received special attention during the design of the system. High level objects such as chapters of a movie, trailers etc. can be described in MDF easily. Labor extensive tasks such as adding multiple sub-title and language tracks are also very easy with the system. LiquidDVD is a step toward ubiquitous DVD.

About Authors

Afzal Hossain is a system architect at Nanova Corporation. He has developed the architecture of the LiquidDVD DVD Authoring System. Before joining Nanova Corporation, Afzal has worked at Intel Corporation on its Merced and Pentium Processor design projects. He has developed the micro-architecture of the Merced Processor, has developed incremental layout verification software for the design of the Pentium Processor. Afzal has designed the architecture of an Algorithmic Memory Test Pattern Generator (APG) that became a leadership product of Schlumberger Technologies for high speed SDRAM testing. Afzal has a BS degree in Electrical and Electronics Engineering from Bangladesh University of Engineering and Technology, Bangladesh and an MS degree in Computer Engineering from Syracuse University, Syracuse, New York. Currently, he is pursuing a Ph.D. degree at Syracuse University. His research interest includes digital audio and video compression, storage and transmission, digital systems architecture, and VLSI engineering.

Nasima Parveen is also a system architect at Nanova Corporation and has developed the architecture of LiquidDVD DVD Authoring System. Before joining Nanova Corporation, Nasima has worked on UltraSPARC-I and UltraSPARC-II Processors design projects at Sun Microsystems. She has designed the Floating Point and Graphics unit of the UltraSPARC processors. At C-Cube Microsystems, she has designed its MPEG-2 Encoder chip. Nasima has a BS degree in Electrical and Electronics Engineering from Bangladesh University of Engineering and Technology, Bangladesh and an MS degree in Computer Engineering from Syracuse University, Syracuse, New York. Her research interest includes digital audio and video compression technology, digital multimedia systems architecture and design, computer graphics and VLSI engineering.

References

[1] "DVD Specification for Read-Only Disc. Part 3 VIDEO SPECIFICATION", Version 1.0, DVD Forum, Toshiba, Tokyo, 1996.

[2] "DVD Specification for Read-Only Disc. Part 1 PHYSICAL SPECIFICATION", Version 1.0, DVD Forum, Toshiba, Tokyo, 1996.

[3] "DVD Specification for Read-Only Disc. Part 2 FILE SYSTEM SPECIFICATION", Version 1.0, DVD Forum, Toshiba, Tokyo, 1996.

[4] Standard ECMA-119, "Volume and File Structure of CDROM for Information Interchange", 2nd Edition, ECMA, Geneva, 1987.

[5] Universal Disc Format™ Specification, Version 1.02, Optical Storage Technology Association, Santa Barbara, 1996.

[6] Standard ECMA-167, "Volume and File Structure for Write-Once and Rewritable Media using Non-Sequential Recording for Information Interchange", 2nd Edition, ECMA, Geneva, 1994.

[7] Standard ISO/IEC 13818-1, "Generic coding of moving pictures and associated information: Systems", ISO/IEC, Geneva, 1996.

[8] Standard ISO/IEC 13818-2, "Generic coding of moving pictures and associated information: Video", ISO/IEC, Geneva, 1996.

[9] Standard ISO/IEC 13818-3, "Generic coding of moving pictures and associated information: Audio", ISO/IEC, Geneva, 1995

[10] Standard ISO/IEC 11172-1, "Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbit/s: Systems", ISO/IEC, Geneva, 1993.

[11] Standard ISO/IEC 11172-1, "Coding of moving pictures and associated audio for digital storage media at up to 1.5 Mbit/s: Video", ISO/IEC, Geneva, 1993.

[12] "Digital Audio Compression (AC-3) Standard", Advanced Television Systems Committee, Washington, 1994.

[13] Standard ISO/IEC 646, "7-bit code character set for information interchange", ISO/IEC, Geneva, 1991.

[14] Standard ISO 9660, "Volume and file structure of CD-ROM for information interchange", ISO, Geneva, 1988.

[15] Standard ISO/IEC 13346, "Volume and file structure of write-once and rewritable media using non-sequential recording for information interchange", ISO/IEC, Geneva, 1995.

[16] The Unicode Standard – Worldwide Character Encoding, Version 1.1, Unicode, Inc.

[17] V. Bhaskaran, K. Konstantinides, "Image and Video Compression Standards, Algorithms and Architectures", Kluwer Academic, Boston, 1995.

[18] W.B. Pennebaker, J.L. Mitchell, "JPEG Still Image Data Compression Standard", Van Norstrand Reinhold, New York, 1993.