White Paper
Blu-ray Disc™
Read-Only Format

2.B Audio Visual Application Format Specifications for BD-ROM Version 2.5

July 2011

COPYRIGHT

TRADEMARK

DISCLAIMER

The information contained herein is believed to be accurate as of the date of publication. However, none of the Blu-ray Disc Association, its Members, or its License Office will be liable for any damages, including indirect or consequential, from use of the White Paper or reliance on the accuracy of this document.

LICENSING

License is required from the Blu-ray Disc Association for the application of the System Description Blu-ray Disc™ Format in both disc and equipment products.

NOTICE

For any further explanation of the contents of this document, or in case of any perceived inconsistency or ambiguity of interpretation, please consult with:

Blu-ray Disc Association

License Office
10 Universal City Plaza, T-100,
Universal City, CA 91608 U.S.A.

Fax.: +1-818-763-9027
Web Site: http://www.blu-raydisc.info
E-mail: license@bdamail.com

Table of Contents:

1 Introduction

This Whitepaper is indented to provide an overview of the Blu-ray Disc Prerecorded application format (BD-ROM) which is being designed through periodic consultation between executives of the technical divisions of Hollywood studios and the Blu-ray Disc application format experts.

1.1 Background

1.1.1 Aims of BD-ROM

BD-ROM is being designed not only for pre-packaged High Definition (HD) movie content but also as a key component of a consumer HD platform. As shown in Figure 1-1 below, the Blu-ray Disc (BD) platform is being designed to provide access to HD content throughout the home via HD digital broadcast recording and HD playback functions.

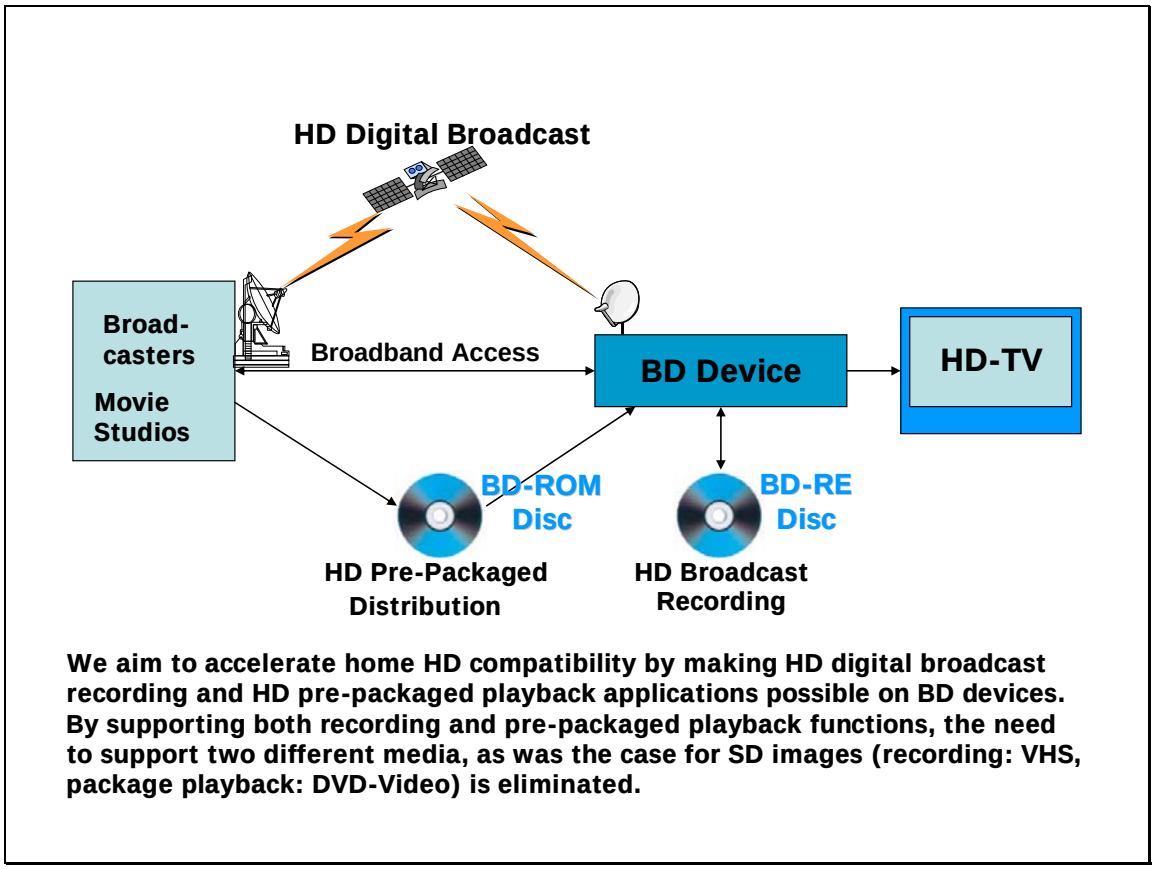


Figure 1-1 – HD Content Distribution as envisioned by the BD Application Standard

When DVD-Video was developed, Standard Definition (SD) image recording devices were already present in the market (VHS). The user therefore had to contend with two different forms of media for SD: VHS for recording and DVD-Video for package media playback. For HD however, we are at the initial rollout stage for HD broadcast receivers, HD recording functions via package media, and HD pre-packaged content distribution. By combining all of these functions into one platform, BD provides a one-stop-shop to fulfill the desires of users who want to use all of these functions. Simply put, the user will be HD compatible by using BD.

BD-ROM players started to appear in the market in 2006. Recently 3D movies have become very popular in the theatres and it was decided to add 3D playback capabilities to the BD-ROM format. From the end of 2009 a new version of the BD-ROM format specification is available that includes Stereoscopic 3D.

1.1.2 Understanding Basic Issues (Required Specifications)

Understanding 1: Market Trends (DVD is still going as strong as ever)

1. When comparing VHS with DVD, there are clear differences: (1) improvement in maneuverability by going from tape to optical disc, (2) improvements in image quality via MPEG-2, (3) going from linear movie viewing to viewing multimedia titles including special features.
2. Popularity of DVD-Video business. US retail sales for 2004: 37.125 million DVD players sold, 1,518.3 million titles sold (Source: DVD Entertainment Group).
3. It will therefore be extremely important to design BD-ROM so that there is clear integration/continuity with DVD-Video, but also with sufficient differentiation.

Understanding 2: Consumer Perspective

1. Consumers expect high image quality with large-screen displays. Plasmas, LCDs, rear-projection televisions, digital projectors and other large-screen, HD capable display devices are now becoming popular. DVD image resolution (SD) is insufficient on these types of large-screen, HD capable displays.
2. Special Features for Movies are Important:
   US consumers are especially drawn to special features (production shots, outtakes, scenes cut from the movie release, etc.) and cite these special features as one of the major reasons for purchasing DVD-Video titles.
3. Need for new ways to view Movies:
   Consumers may not pay attention to BD unless we can provide new, revolutionary ways to view movies and movie related content in addition to providing a HD experience e.g. through services provided via broadband.
4. User Interface Integration is important:
   We need to keep in mind that users are familiar with (and are comfortable with) the DVD-Video User Interface but that they also desire a richer interaction experience.

Understanding 3: Studio Perspective

1. The DVD-Video business is going extremely well, but competition between titles has also increased dramatically.
   - There is a need to include many special features and games to clearly distinguish titles.
   - Increased Profit by releasing the DVD-Video title quickly after the theatrical release (In the US, DVD-Video titles are released about 5 months after the theatrical release of the movie).
   - Production schedules are becoming shorter and title programs are becoming more complex – the burden on the production is getting heavier. Production needs to become more efficient.
2. BD-ROM Production:
   - The DVD-Video production area is shouldering a large burden. Assuming that current DVD-Video producers will also take on BD-ROM production, a smooth integration of the production process is essential.
   - We would like to make BD-ROM easier to manufacture than DVD-Video (and capable of doing more, easily).
   - Desire for New Type of Titles & Services:
     We would like to promote the wide use of BD by providing new services and functions through BD-ROM titles. This will be another means of re-selling catalogue titles (i.e., continue to ride the wave of DVD sales).
3. Desire to Implement New Business Models:
   - The introduction of DVD-Video allowed the industry to move from the "rental" driven VHS business model to a "sales" driven DVD business model with a higher profit margin.
   - Studios are looking forward to gaining a new source of income through the introduction of new package media. To prepare for this eventuality, they are looking to new means of distribution including those that use broadband technology.

1.1.3 Evolution of Package Media for Movies

BD-ROM design is conducted based on the probable evolution of package media as envisioned in Figure 1-2 below, with due consideration to the issues (usage requests) outlined previously.

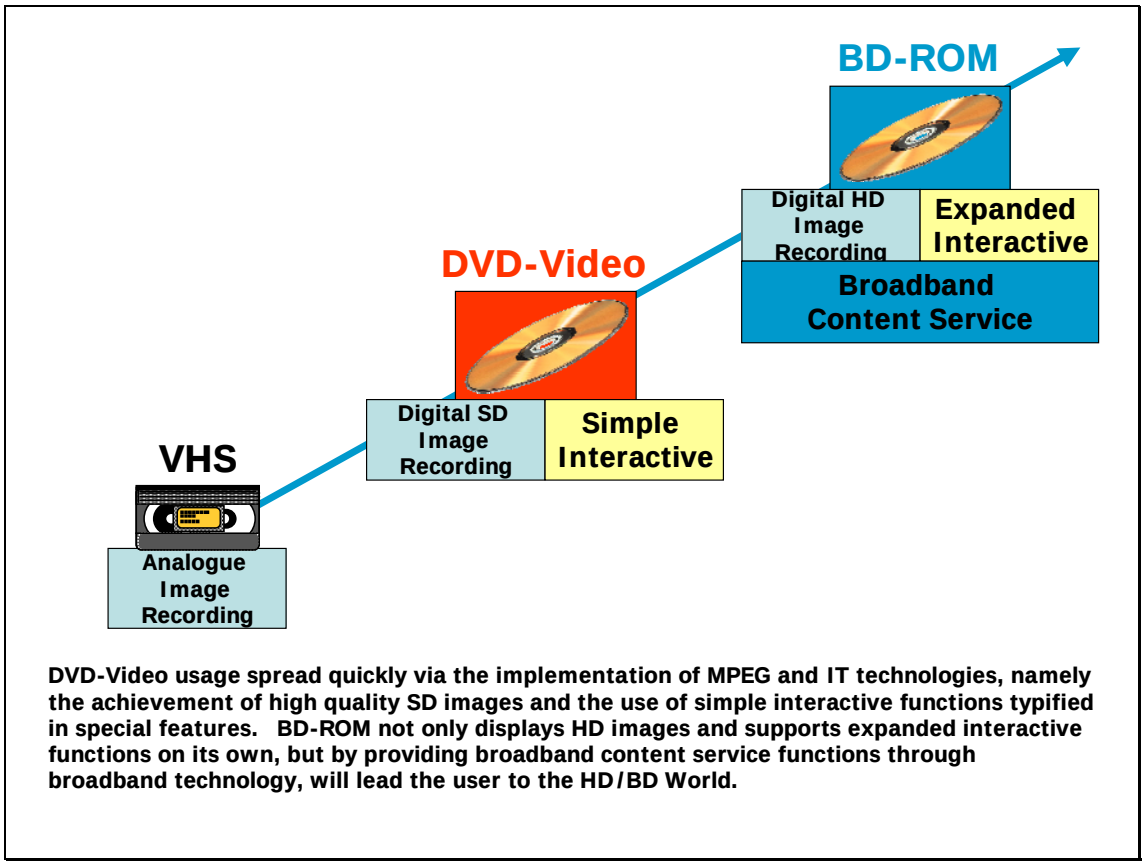


Figure 1-2 –Evolution of Movie Package Media

The use of DVD-Video spread quickly because of high quality SD image technology and simple interactive functions implemented through MPEG and IT technologies.

BD-ROM is being designed with an architecture for a "BD World" that clearly distinguishes itself from DVD by (1) realizing High Definition images, (2) creating an "expanded" interactive environment, and (3) creating a platform for broadband content service functions that will expand the viewer's way of enjoying content.

1.2 Overview

For the purposes of providing the reader with an understanding of BD-ROM features, this Whitepaper categorizes related features into "modes" of which two are defined ("HDMV" and "BD-J"). The categorization used in this Whitepaper does not represent the actual structure of BD-ROM nor does this Whitepaper provide a description of the complete set of features supported by BD-ROM.

Introduction to HDMV mode

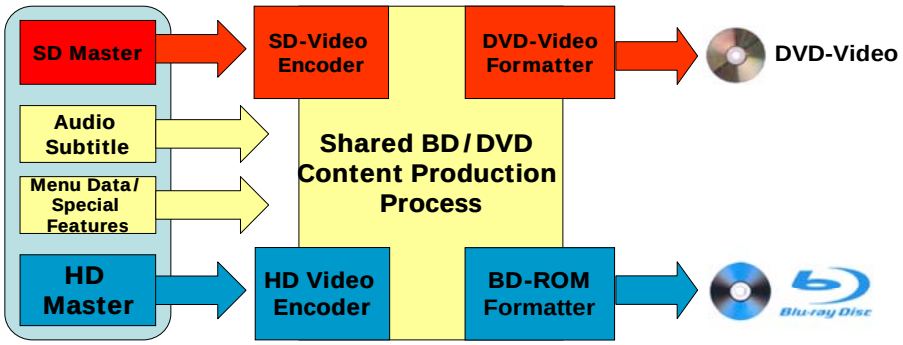
BD-ROM provides an easy-to-author framework for creation of High Definition movie experiences known as "HDMV" (High Definition Movie) mode. HDMV has been designed from the ground up to support a feature set that supersedes DVD-Video while emphasizing production continuity with existing media formats. HDMV supports all of the well known DVD-Video features such as seamless multi-angle and multi story, Language Credits (dynamic selection of a credits sequence depending on the users Language choice), Director's cuts, Trilogy collections etc.

Here are some of the key features offered by HDMV:

- Industry Standard High Definition Video and Surround Sound Audio:
  - MPEG-2, MPEG-4 AVC, SMPTE VC-1 and MPEG-4 MVC video formats.

  - LPCM as well as Dolby®[1] Digital, Dolby® Digital Plus, Dolby® Lossless, DTS digital surround®[2], DTS-HD®, DRA and DRA Extension audio formats.
- Picture-in-picture and Audio Mixing:
  - "Picture-in-picture" – Secondary video can be overlaid on the Primary video.
  - "Audio Mixing" - Secondary audio can be mixed with the Primary audio.
- Independent High Definition Graphic planes:
  - Two independent High Definition graphic planes and one High Definition video plane simplifies the process of Authoring both Menu and Subtitle graphics compared to DVD-Video.
- Improved Menu features:
  - "Multi-page Menus" - Menu presentations can be changed with no interruption to AV playback.
  - "Pop-up Menus" – Menus can be shown or removed from display based on User request.
  - Full color High Definition animated Buttons and animated Menu transition effects.
  - "Button-sounds" – sounds can be presented when Menu Buttons are selected or activated.
- Improved Subtitle features:
  - High Definition "Bitmap Subtitles" supporting full color images with frame-accurate animation effects up to full video frame rate.
  - High Definition "Text Subtitles" utilizing vector-based fonts, encoded text data and multiple style definitions. Text Subtitles do not affect the bandwidth available for a BD-ROM Title's AV stream.
- Additional Features:
  - "Browsable Slideshow" - still images may be presented and changed without interruption to audio playback.

HDMV has been developed with a focus on ease-of-authoring, ease-of-verification and reduced content production cost. This has been achieved in three ways: 1) ground-up modern design of flexible data structure definitions; 2) integration into the current content production and material preparation process used for DVD-Video titles; 3) enabling the possibility of simultaneous production of DVD-Video and BD-ROM titles as shown in Figure 1-3 below.



BD-ROM is designed to implement new functions that differentiate it from existing media and to ensure compatibility with BD-RE. Additionally, the design has an emphasis on BD-ROM content production. From the perspective of content production/raw material preparation, title production is made that much more difficult if the design demands a completely different process in creating an HD movie mode-led title. (Impediments would include higher costs for new BD-ROM content production tools, new specialized equipment, new tool implementation, operator training, etc.) Therefore, as shown above, the HD movie mode has been designed with emphasis on compatibility with DVD in the content production process that will allow simultaneous DVD and BD-ROM title production.

Figure 1-3 – Example of HD Movie mode content production process

[1] Dolby and the double-D symbol are registered trademarks of Dolby Laboratories.
[2] DTS, DTS digital surround and DTS-HD are registered trademarks of Digital Theater Systems, Inc.

If the BD-ROM standard is based solely on the assumption of a content production and material preparation process that is entirely different from DVD-Video, then the cost of Title production becomes prohibitively more expensive (costs for new development of BD-ROM content production tools, exclusive material necessary for production, new tool implementation, operator training, etc.), thereby potentially hindering the spread of BD-ROM.

Details of the HDMV platform are given in Section 4 along with more detailed examples of HDMV applications.

Introduction to BD-J mode

BD-ROM also provides a fully programmable application environment with network connectivity thereby enabling the Content Provider to create highly interactive, updateable BD-ROM titles. This mode is based on the Java™[3] platform and is known as "BD-J". Content Providers are able to include interactive Java applications on a BD-ROM disc in various ways (one application for the entire disc, one application per Title, etc.).

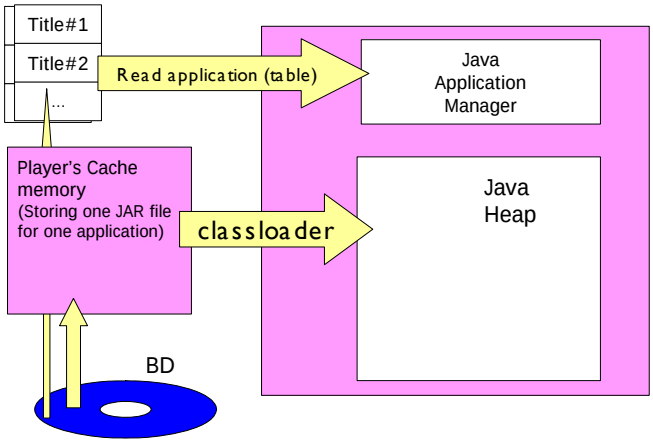


Figure 1-4 – Overview of Java application tables in BD-ROM

Possible BD-J applications include:

- A BD-ROM Title that supports downloading trailers for a sequel from a Content Provider's website and playback under application control.
- A BD-ROM disc with a set of games, each game associated with a Title in the disc's table of content. The main Menu of the disc allows downloading subsequent games from a Content Provider's website under certain conditions, like solving a puzzle for example.
- A BD-ROM Title is distributed supporting only a small number of languages. Later support for more languages (i.e. subtitle and or audio streams) can be downloaded by the BD-J application on the disc.

Java is a platform independent programming environment deployed in a wide verity of environments: Server based applications can be supported through the Java 2 platform Enterprise Edition (J2EE), while Desktop based applications can be supported through the Java 2 platform Standard Edition (J2SE), and Consumer Electronics based applications (for devices like cell-phones and interactive digital receivers) can be supported through the Java 2 platform Micro Edition (J2ME). Java provides an open and flexible programming environment for BD-ROM.

BD-J provides a Java UI & graphics framework along with support for Local Storage and Internet connectivity features thereby creating a complete and future proof solution. A BD-ROM disc can contain

[3] Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. For more information please refer to: http://www.sun.com/suntrademarks/

a mix of titles based on HDMV and BD-J. Details of the BD-J platform are given in Section 5 along with more detailed examples of BD-J applications.

2 Database information

2.1 BD-ROM data structure

Figure2-1 below provides a simplified overview of the BD-ROM data structure. BD-ROM has four layers for managing AV stream files as follows: ***Index table***, ***Movie Object/BD-J Object***, ***PlayList*** and ***Clip***.

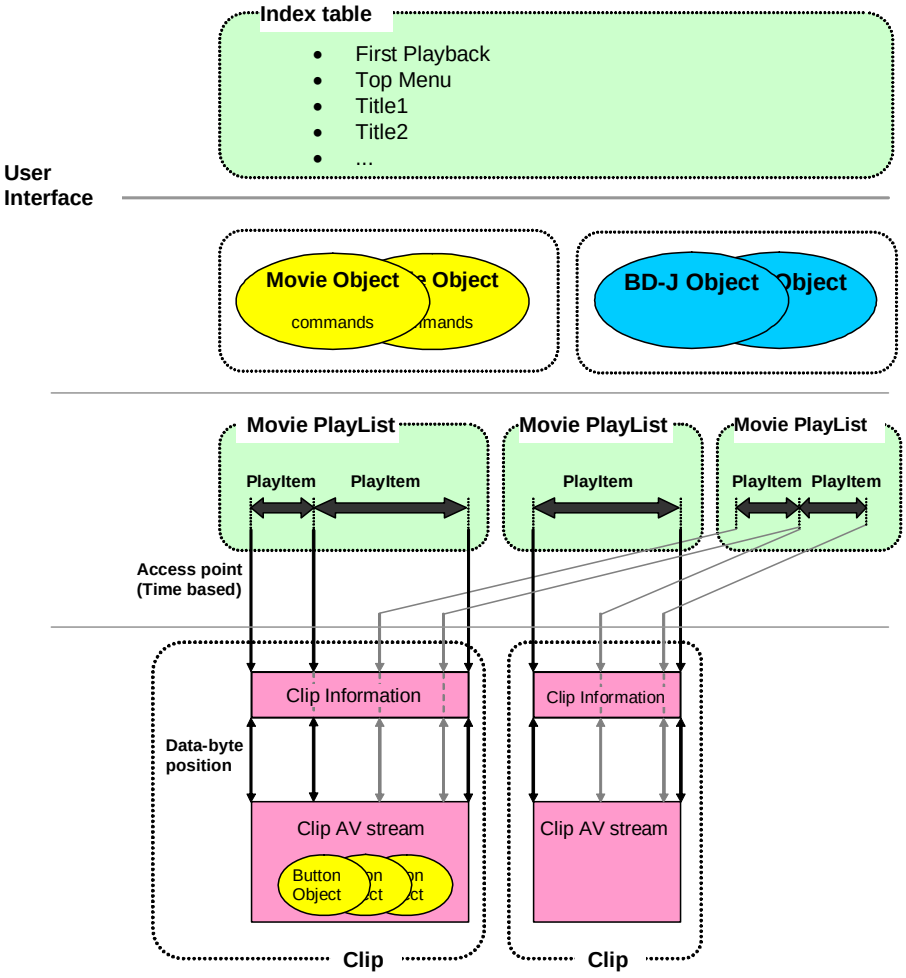


Figure 2-1 – Simplified structure of BD-ROM

2.1.1 Index table

The "Index Table" is a top-level table that defines the Titles and the Top Menu of a BD-ROM disc. This table contains entry points for all of the Titles and the Top Menu. The Player references this table whenever a Title or Menu is to be executed e.g. whenever the Title Search or Menu Call operation is called, the player refers to this table to determine the corresponding Movie Object/BD-J Object that is to be executed.

The Index Table also has an entry to a Movie Object/BD-J Object designated for "First Playback" – this can be used by Content Providers to perform automatic playback. When the disc is loaded, the player refers to the "First Playback" entry to determine the corresponding Movie Object or BD-J Object that shall be executed.

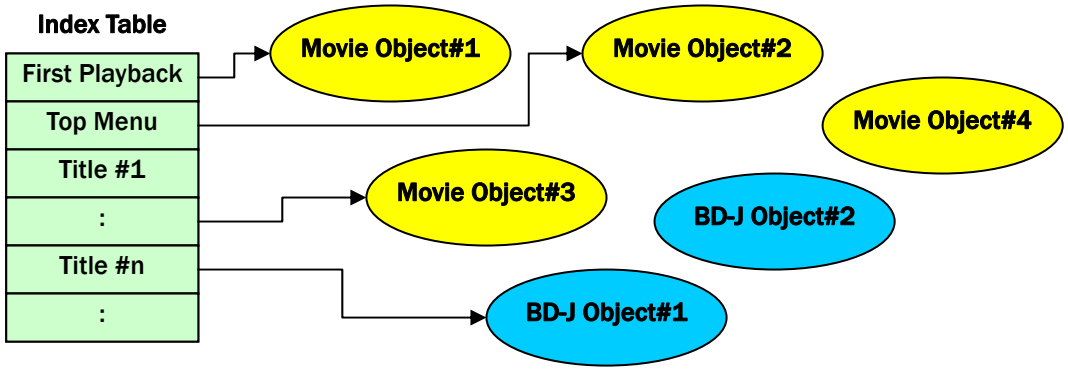


Figure 2-2 – An illustration of Index Table

2.1.2 Movie object

A "Movie Object" consists of an executable navigation command program (HDMV program). This enables dynamic scenario description. Movie Objects exist in the layer above PlayLists.

Navigation commands in a Movie Object can launch PlayList playback or another Movie Object. This enables the Content Provider to define a set of Movie Objects for managing playback of PlayLists in accordance with a user's interaction and preferences.

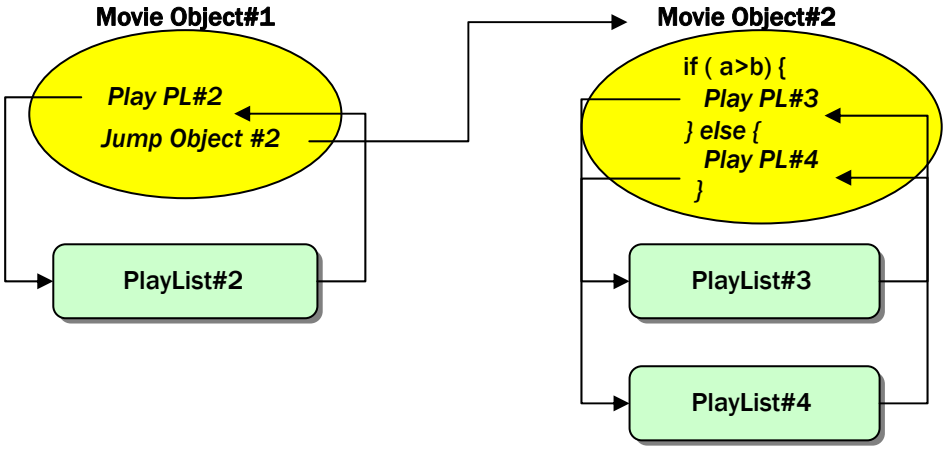


Figure 2-3 – An illustration of Movie Object and PlayList

2.1.3 Java Xlet

When a Title associated with a BD-J Object in the Index Table on disc is selected, the corresponding application is automatically launched and its lifecycle is bound to the Title. A BD-J application is a Java Xlet which is controlled by the BD-ROM player's Application Manager through its Xlet interface. The Xlet interface has four states as follows: loaded, paused, active and destroyed. Once a BD-J application is destroyed, any resources allocated to it, such as memory and AV control, shall be released.

2.1.4 Movie PlayList

A "Movie PlayList" is a collection of playing intervals in the Clips. One such playing interval is called a *PlayItem* and consists of an IN-point and an OUT-point, each of which refers to positions on a time axis of the Clip. Therefore a PlayList is a collection of PlayItems. Here the IN-point means a start point of a playing interval, and the OUT-point means an end point of the playing interval.

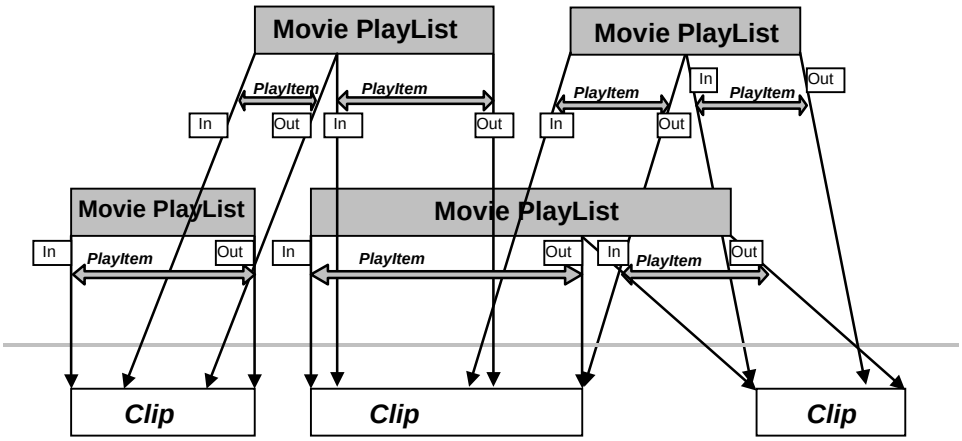


Figure 2-4 – An illustration of Movie PlayList

2.1.5 Clip

An AV stream file together with its associated database attributes is considered to be one object. The AV stream file is called a "*Clip AV stream file"*, and the associated database attribute file is called a "*Clip Information file"*.

An object consisting of a Clip AV stream file and its corresponding Clip information file is called a ***Clip***.

(1) Clip AV stream file

A Clip AV stream file stores an MPEG-2 Transport Stream (ISO/IEC 13818-1) contained in a structure compliant with the BD-ROM AV specification. This structure is called the "*BDAV MPEG-2 Transport Stream"*, an overview of which is provided in Section 3.1.

(2) Clip Information file

The Clip Information file stores the time stamps of the access point into the corresponding AV stream file. The Player reads the Clip Information to find out the position where it should begin to read the data from the AV stream file.

There is a one-to-one relationship between a Clip AV stream file and a Clip Information file.

2.2 Mechanism to realize "Browsable Slideshow", "Pop-up Menu" and "Text subtitle" applications

BD-ROM provides a framework to realize "Browsable Slideshow", "Pop-up Menu" and "Text subtitle" applications by providing a mechanism called "sub-path".

The sub-path is a structure that enables the author to easily add a separate (out-of-mux) stream to the main Clip on the main-path.

Figure 2-5 shows the relation between the main-path and the sub-path in case of Text subtitle application. As shown in the figure, a SubPlayItem is logically added to the PlayItem in the PlayList to associate a separate Clip (SubPath) to the PlayItem on the Main path. The text subtitle presentation path using the SubPlayItem is synchronized with the main path using PlayItems in the PlayList.

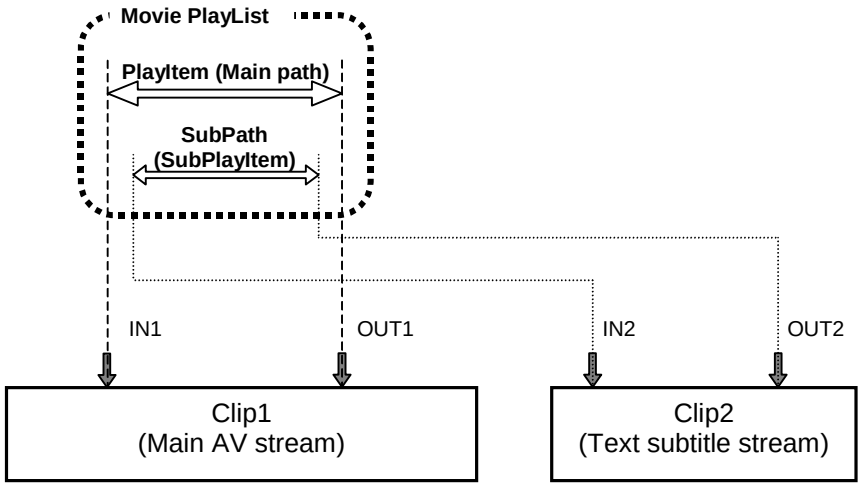
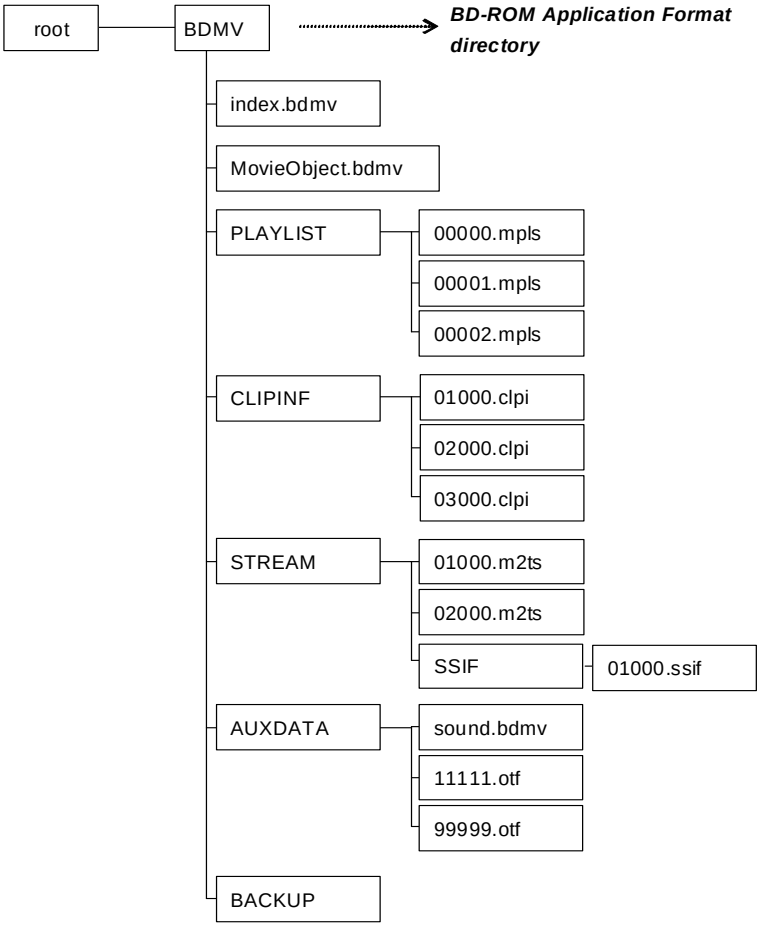


Figure 2-5 - An example of Main path and Sub path structure with Text subtitle presentation path

2.3 Directory and File structure

All BD-ROM application files are stored under a “BDMV” directory.



1) “BDMV” directory:
The BDMV directory contains the PLAYLIST, CLIPINF, STREAM, AUXDATA and BACKUP directories.

2) “PLAYLIST” directory:
The PLAYLIST directory contains the Database files for Movie PlayLists.

3) “CLIPINF” directory
The CLIPINF directory contains the Database files for Clips.

4) “STREAM” directory
The STREAM directory contains AV stream files.

5) “AUXDATA” directory
The AUXDATA directory contains Sound data files and Font files.

6) “BACKUP” directory
The BACKUP directory contains copies of the "index.bdmv” file, the “MovieObject.bdmv” file, all the files in the PLAYLIST directory and all files in the CLIPINF directory.

7) “SSIF” directory
Stereoscopic Interleaved files shall be placed under this directory.

8) "index.bdmv" file
The "index.bdmv" file stores information describing the contents of the BDMV directory. There is only one index.bdmv file under the BDMV directory and its filename is fixed to "index.bdmv".

9) "MovieObject.bdmv" file
The "MovieObject.bdmv" file stores information for one or more Movie Objects. There is only one MovieObject.bdmv under the BDMV directory and its filename is fixed to " MovieObject.bdmv ".

10) "xxxxx.mpls" file
The "xxxxx.mpls" files store information corresponding to Movie PlayLists. One file is created for each Movie PlayList. The filenames of these files are in the form "xxxxx.mpls", where "xxxxx" is a 5-digit number corresponding to the Movie PlayList.

11) "zzzzz.clpi" file
The "zzzzz.clpi" files store Clip information associated with a Clip AV stream file. The filenames of these files are in the form "zzzzz.clpi", where "zzzzz" is a 5-digit number corresponding to the Clip.

12) "zzzzz.m2ts" file
The "zzzzz.m2ts" files contains a BDAV MPEG-2 transport stream. The names of these files are in the form "zzzzz.m2ts", where "zzzzz" is a 5-digit number corresponding to the Clip. The same 5-digit number "zzzzz" is used for an AV stream file and its associated Clip information file.

13) "sound.bdmv" file
The "sound.bdmv" file stores data relating to one or more sounds associated with HDMV Interactive Graphic streams applications. This file may or may not exist under the AUXDATA directory. If it exists, there shall be only one sound.bdmv file and its filename is fixed to "sound.bdmv ".

14) "aaaaa.otf" file
The "aaaaa.otf" file stores the font information associated with Text subtitle applications. The names of these files are in the form "aaaaa.otf", where "aaaaa" is a 5-digit number corresponding to the Font.

15) "zzzzz.ssif" file
The "zzzzz.ssif" file is a Stereoscopic Interleaved file that is composed from two BDAV MPEG-2 transport streams. One of the BDAV MPEG-2 transport streams includes an MPEG-4 MVC Base view video stream for left eye or right eye, and the other includes an associated MPEG-4 MVC Dependent view video stream for right eye or left eye respectively. This file is used only when 3D video is played back.
The 5-digit number "zzzzz" is the same as the number used for the AV stream file "zzzzz.m2ts" that includes the MPEG-4 MVC Base view video stream.

3 MPEG2 Transport stream for BD-ROM

The Blu-ray Disc Read-Only Format (BD-ROM) use a common format for stream multiplexing – this format is based on the MPEG-2 Transport Stream industry standard (ISO/IEC 13818-1).

3.1 BDAV MPEG-2 Transport Stream

A MPEG-2 Transport Stream is stored in a Clip AV stream file in a structure known as the *"BDAV MPEG-2 Transport Stream"*. A BDAV MPEG-2 Transport Stream conforms to the data structure illustrated in Figure 3-1. The BDAV MPEG-2 Transport Stream is constructed from one or more "Aligned units", where:

1) The size of an Aligned unit is 6144 bytes (2048*3 bytes).

2) An Aligned unit starts from the first byte of the source packets.

3) The length of a source packet is 192 bytes. One source packet consists of one TP_extra_header structure and one MPEG2 transport packet structure. The length of the TP_extra_header structure is 4 bytes and the length of the transport packet structure is 188 bytes

4) One Aligned unit consists of 32 source packets.

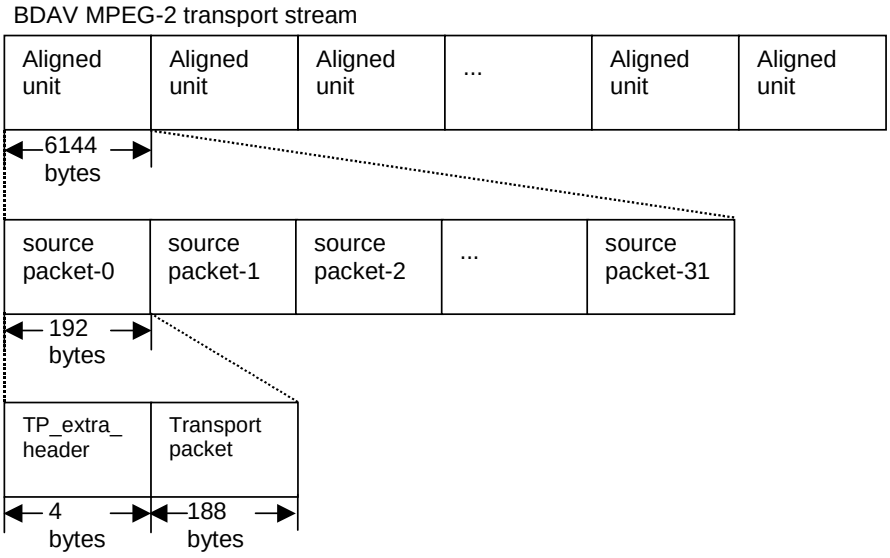


Figure 3-1 – Structure of BDAV MPEG-2 transport stream

Aligned units are recorded in three consecutive logical sectors on the BD-ROM disc. The size of one logical sector is 2048 bytes. The maximum multiplex rate of the BDAV MPEG-2 Transport Stream is 48Mbps.

The decoder model of the BDAV MPEG-2 Transport Stream is described in the "White Paper: BD RE - Logical and Audio Visual Application Format Specifications" available at: "http://www.blu-raydisc.com/".

3.2 Elementary streams in BDAV MPEG-2 Transport Stream

Video, audio, graphics and text elementary streams are coded in the PES packet payload of the BDAV MPEG-2 Transport Stream. The coding method for each of these elementary streams is specified in Figure 3-2 below.

Name of elementary stream	Coding method of elementary stream
Primary video stream	MPEG-2 video stream
	MPEG-4 AVC video stream
	SMPTE VC-1 video stream
	MPEG-4 MVC video stream
Secondary video stream	MPEG-2 video stream
	MPEG-4 AVC video stream
	SMPTE VC-1 video stream
Primary audio stream	Linear PCM audio stream
	Dolby Digital audio stream
	Dolby Digital Plus audio stream
	Dolby Lossless audio stream
	DTS digital surround audio stream
	DTS-HD audio stream
	DRA audio stream
	DRA Extension audio stream
Secondary audio stream	Dolby Digital Plus audio stream
	DTS-HD LBR audio stream
Graphics stream	Presentation graphics stream
	Interactive graphics stream
Text subtitle stream	HDMV Text subtitle stream

Figure 3-2 - Elementary streams in the BDAV MPEG2 Transport Stream

3.3 Video streams for 2D

3.3.1 Primary video stream

Primary video streams shall be MPEG-2 video format (ISO/IEC 13818-2) compliant, MPEG-4 AVC video format (ISO/IEC 14496-10) compliant or SMPTE VC-1 video format compliant.
The video formats shown in Figure 3-3 can be used for BD-ROM video streams. Not all BD-ROM Players support 50Hz video formats.

Video	CODECs	MPEG-2: MP@HL and MP@ML
		MPEG-4 AVC: HP@4.1/4.0 and MP@4.1/4.0/3.2/3.1/3.0
		SMPTE VC-1: AP@L3 and AP@L2
	Max. bitrate	40Mbps
	HD	1920x1080x59.94-i, 50-i (16:9) 1920x1080x24-p, 23.976-p (16:9) 1440x1080x59.94-i, 50-i (16:9) 1440x1080x24-p, 23.976-p (16:9) 1280x720x59.94-p, 50-p (16:9) 1280x720x24-p, 23.976-p (16:9)
	SD	720x480x59.94-i (4:3/16:9) 720x576x50-i (4:3/16:9)

Figure 3-3 - Specification of BD-ROM Primary video streams

3.3.2 Secondary video stream

Secondary video is used for the Picture-in-Picture application as explained in section 4.1.7. Secondary video streams can use the same format shown in the above Figure 3-3 and the following additional formats.

- 720x480x24-p, 23.976-p (4:3/16:9)
- 720x576x25-p (4:3/16:9)

Not all BD-ROM Players support HD video formats and 50Hz video formats of the Secondary video.

3.4 Audio streams

3.4.1 Primary audio stream

The BD-ROM specification defines eight types of Primary audio stream formats with various configurations and settings, as shown in Figure 3-4 below. Some formats, configurations and settings are optional for BD-ROM Players.

	CODEC	LPCM	Dolby Digital	Dolby Digital Plus	Dolby Lossless	DTS digital surround	DTS-HD	DRA	DRA Extension
Primary Audio	Max. bitrate	27.648 [Mbps]	640 [kbps]	4.736 [Mbps]	18.64 [Mbps]	1.524 [Mbps]	24.5 [Mbps]	1.5 [Mbps]	3.0 [Mbps]
	Max.ch	8(48kHz, 96kHz), 6(192kHz)	5.1	7.1	8(48kHz, 96kHz), 6(192kHz)	5.1	8(48kHz, 96kHz), 6(192kHz)	5.1	7.1
	bits/ sample	16, 20, 24	16 – 24	16 – 24	16 - 24	16, 20, 24	16 - 24	16	16
	Sampling frequency	48kHz, 96kHz, 192kHz	48kHz	48kHz	48kHz, 96kHz, 192kHz	48kHz	48kHz, 96kHz, 192kHz	48kHz	48kHz, 96kHz

Figure 3-4 – Specification of BD-ROM Primary audio streams

3.4.2 Secondary audio stream

BD-ROM supports two types of Secondary audio stream formats as shown in Figure 3-5 below. Secondary audio is used to add additional audio presentation to Primary audio stream. Thus, audio output of the BD-ROM player are mixed audio signal of Primary audio stream and Secondary audio stream. Application image is explained in section 4.1.7.

	CODEC	Dolby Digital Plus	DTS-HD LBR
Secondary Audio	Max. bitrate	256 [kbps]	256 [kbps]
	Max.ch	5.1	5.1
	bits/ sample	16	16
	Sampling frequency	48kHz	48kHz

Figure 3-5 – Specification of BD-ROM Secondary audio streams

3.5 Presentation Graphics and Interactive Graphics streams

BD-ROM provides two types of graphics streams as shown in Figure 3-6 below. The Presentation Graphics stream (available in HDMV and BD-J) is intended for Subtitles and Animated Graphics, and the Interactive Graphics (available only in HDMV) is intended for Menu Graphics.

Graphics	Plane size	1920x1080/1280x720/720x480/720x576	
	Color	8bit Index lookup table(24 bit color + 8 bit alpha)	
	Compression	Run Length Encoding	
	Presentation planes	2 planes	
	Presentation Plane name	Presentation Graphics	Interactive Graphics
	Main usage	Subtitles	Menus
	Animation Effects	Fade In/Out, Color changing, Wipe In/Out, Scrolling	

Figure 3-6 – Specification of BD-ROM Graphics streams

3.6 Text subtitle streams

BD-ROM also supports Text subtitle streams. A Text subtitle is defined by a series of character codes plus font and style information. Text subtitles are available in addition to the Bitmap based Subtitles provided by BD-ROM Presentation Graphics streams.

Text Subtitle	Plane size	1920x1080/1280x720/720x480/720x576
	Character encoding	Unicode2.0 (UTF-8 and UTF-16BE), Shift-JIS, KSC 5601-1987 (including KSC 5653), GB18030-2000, GB2312, BIG5
	Presentation Plane name	Presentation Graphics
	Color	8bit Index lookup table(24 bit color + 8 bit alpha)
	Effect	Fade In/Out, Color changing
	Presentation style	Text position/flow/alignment, font style/size/color
	User change-able style	Text position, font size, line space

Figure 3-7 – Specification of BD-ROM Text subtitle streams

4 HDMV mode

The BD-ROM HDMV platform provides a flexible, simple framework for creation of interactive High Definition and Standard Definition movie experience applications. This section will provide an overview of some of the key features provided in HDMV.

4.1 Core functions

4.1.1 Out-of-Mux stream Framework

HDMV provides a framework for individual stream handling. An Out-of-Mux stream is an additional stream which is decoded while the main MPEG stream is decoding. The Out-of-Mux framework provides support for new applications such as Pop-Up Menus, Browsable Slideshow with background music, Button click sound and Text subtitle display.

Decoder model

The HDMV decoder model is equipped with two read buffers, two preloading buffers and two switches. The second read buffer enables the supply of an Out-of-Mux audio stream to the decoder even while the main MPEG stream is being decoded. The preloading buffers cache Text subtitles, Interactive Graphics and sounds effects (which are presented at Button selection or activation). The preloading buffer stores data before movie playback begins and supplies data for presentation even while the main MPEG stream is being decoded.

This switch between the ECC decoder and buffers selects the appropriate buffer to receive demodulated packet data from any one of read buffers or preloading buffers. Before starting the main movie presentation, effect sounds data (if it exists), text subtitle data (if it exists) and Interactive Graphics (if preloaded Interactive Graphics exist) are preloaded and sent to each buffer respectively through the switch. The main MPEG stream is sent to the primary read buffer (RB1) and the Out-of-Mux stream is sent to the secondary read buffer (RB2) by the switch.

The audio decoder also has a switch to select a read buffer for source audio data. In the case of a Browsable Slideshow with background music, the switch selects the secondary read buffer (RB2) to store an Out-of-Mux audio stream and continue supplying the audio stream to the decoder. In all other cases the switch selects the primary read buffer (RB1).

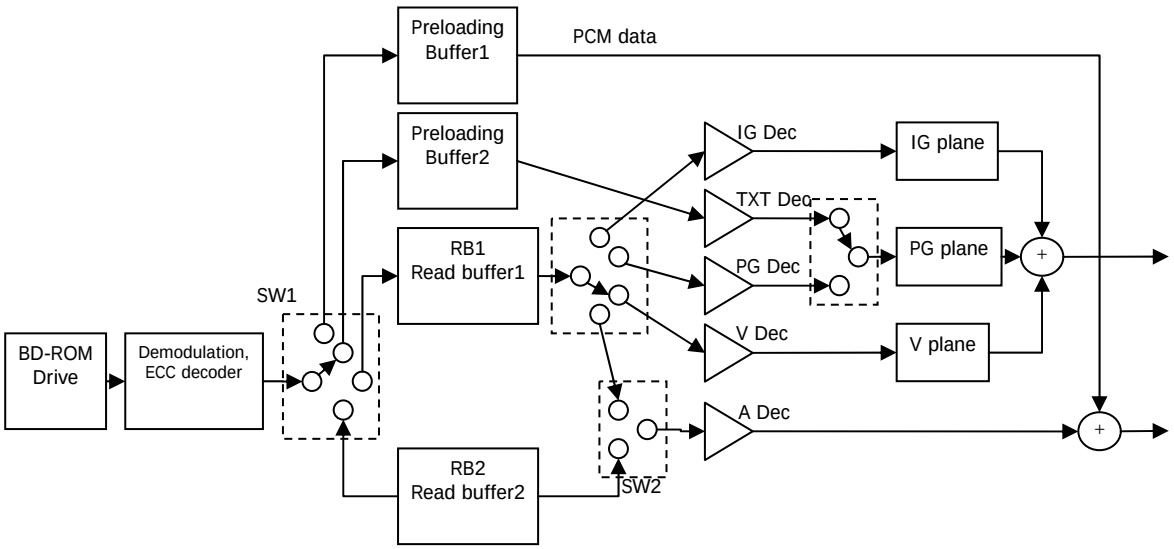


Figure 4-1 – Decoder model

4.1.2 Graphics Framework

HDMV provides two graphics frameworks for compositing graphics on video: the Interactive Graphics system and the Presentation Graphics system.

A BD-ROM Interactive graphics stream contains information required to provide a series of interactive displays, which appear and disappear with frame accuracy, that are supplemental to an associated HDMV presentation. It is envisaged that Interactive graphics streams will typically be used to provide both the display and associated commands of graphical interactive displays during a HDMV presentation.

A BD-ROM Presentation graphics stream, available in both HDMV and BD-J modes, contains information required to provide non-interactive images that are supplemental to an associated BD-ROM presentation. The images described in the stream are designed for graphic overlay, with frame accuracy, on the associated video image. It is envisaged that BD-ROM Presentation graphics streams will typically be used to provide subtitle services and/or other animated graphics during a HDMV or BD-J presentation.

1) Graphics planes

As shown in Figure 4-2, HDMV defines two independent full HD resolution (1920x1080) graphics planes for graphics which are composited on the video plane. One graphics plane is assigned for subtitling applications (Presentation Graphics or Text Subtitles) and the other plane is assigned to interactive applications (HDMV or BD-J mode interactivity graphics).

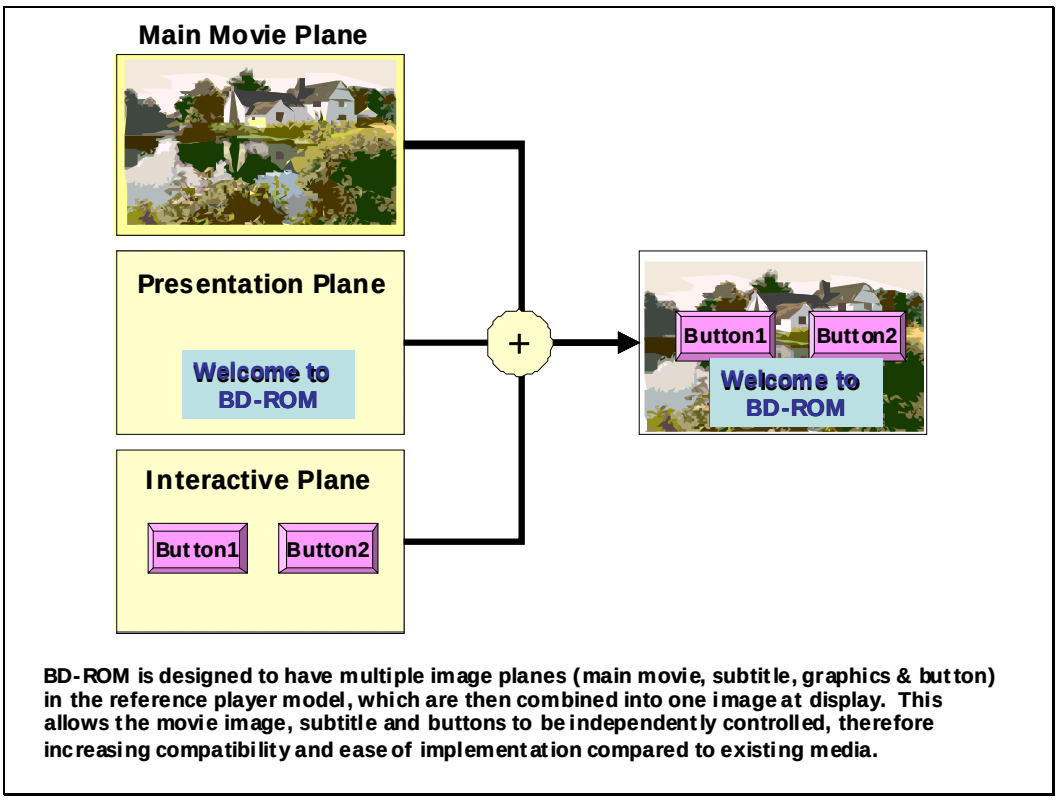


Figure 4-2 – Graphics planes

Each graphics plane has 8-bits per pixel, with each pixel value referring to an index entry in a Palette for translation to YCrCb color and 8-bit (256level) alpha. This color capability offers an enhanced visual experience and allows compelling content to be displayed using the HDMV Interactive Graphics system.

2) Graphics model

The HDMV graphics systems define a flexible decoding and composition system for providing graphics displays whereby graphic images may be reused, with different effects applied, in one or more graphics displays.

A HDMV graphics stream consists of one or more “Segments” – “Segments” are the basic syntactical element of HDMV graphics streams. There are three types of Segments - Graphics Object Segment, Composition Segment and Palette Segment:

- Composition Segment – defines the appearance of a graphics display.
- Graphics Object Segment – Bitmap image data compressed with an RLE compression schema.
- Palette Segment – color and transparency data (up to 256 entries) for translation of 8bit index color to full color when compositing on the video plane.

Segments are processed by the BD-ROM HDMV graphics decoder as shown in Figure 4-3 below.

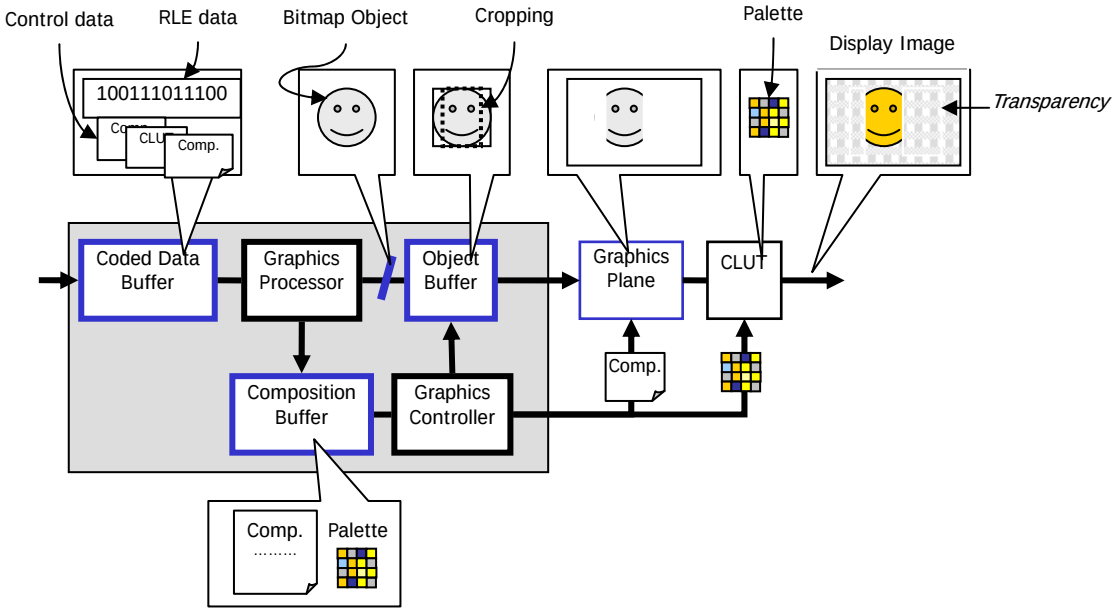


Figure 4-3 – Illustration of BD-ROM HDMV Graphics decoding

A Segment first arrives at the Coded Data Buffer. The Graphics Processor extracts the Segment at the time defined by a system time-stamp associated with the Segment. When Composition and Palette Segments arrive at the Graphics processor, they are decoded to the Composition Buffer.

When Graphics Object Segments arrive at the Graphics Processor, the Graphics Processor decodes the Graphics Object to an uncompressed 8-bit graphics object which is then stored in the Object Buffer. Once a Graphics Object has been decoded, it is available for use by one or more graphics displays as described in Composition Segment.

The Graphics Controller is responsible for compositing graphics images on to the graphics plane in accordance with the description in the Composition Segment. The composited image on the graphics plane is transformed to full color and transparency by the CLUT module and then overlaid on the video image. The decoder implements a Pipelined decoding model such that Graphics Displays may be assembled in the Graphics Plane while, at the same time, new Graphics data is decoded into the Object Buffer.

3) Graphics animations

Support for graphics effects is part of the graphics tool set for Content Providers to create rich BD-ROM Graphics Displays. Supported effects include scrolls, wipes, cuts, fades (transparency changes) and color changes. All of these effects may be utilized in both Interactive (e.g. to be used for Menu page transitions) and Presentation Graphics streams (e.g. to be used for advanced Subtitles or Karaoke).

Composition Segments indicate the Graphics Objects to be used for a graphics display and may define a cropping transform to be applied when compositing the Graphics Object. Composition Segments also indicate the Palette to be used for the graphics display. Effects are realized by providing multiple Compositions Segments which change cropping areas of Graphics Objects (e.g. to provide wipes, scrolls and cuts) as illustrated in Figure 4-4 and/or reference different Palettes (e.g. to provide fades or color changes).

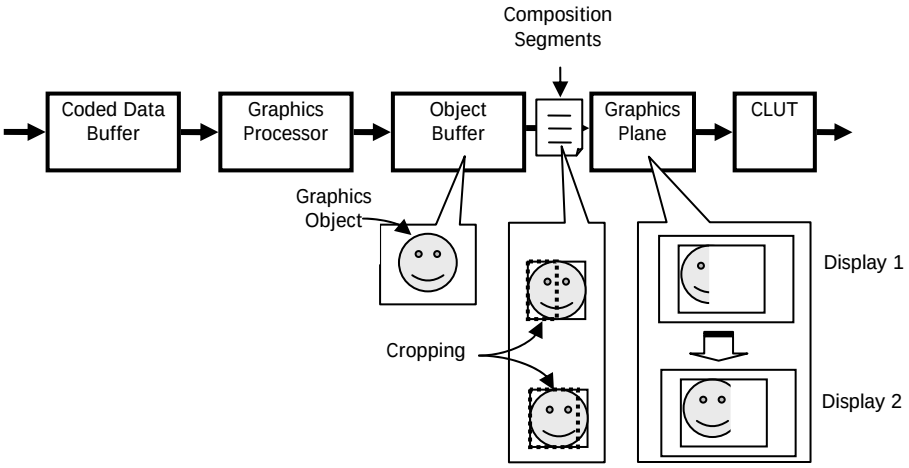


Figure 4-4 – Illustration of wipe effect

HDMV Interactive Graphics are further extended to support animated sequences of graphics for Buttons. The Normal, Selected and Activated states of a Button may be animated with a sequence of different images. With 8-bit index color and transparency support along with support for frame rates up to the underlying video frame rate, the creative possibilities are greatly expanded over existing formats.

4.1.3 Text Subtitle Framework

HDMV provides support for Text based subtitles. This framework enables the Content Provider to create Subtitling applications using character code sequences (text) plus style information. The Text subtitle stream is stored as an Out-Of-Mux stream (as described in Section 4.1.1) which does not impinge on the bandwidth of the main AV stream thereby enabling the Content Provider to supply several Subtitle streams without affecting the quality of the associated audio and video presentation. The BD-ROM player can present either a Text Subtitle stream or a Presentation Graphics Subtitle stream – it will not present both at the same time.

A Text subtitle stream consists of a sequence of "Text Dialogs", each of which consists of a text to be presented. Text Dialogs contain up to a maximum of two Text Regions, each of which may be presented with an individual presentation Style. All text in a Text Dialog shall be presented during the same presentation time slot which is frame-accurately synchronized with the main AV stream.

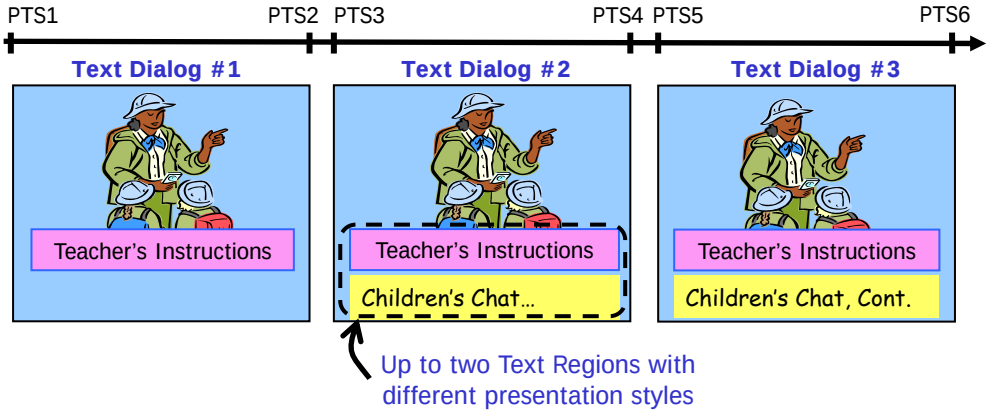


Figure 4-5 – Illustration of Text Dialogs and Text Regions

Text subtle streams are rendered to an 8-bit index image and a Palette is applied for translation to YCrCb color and 8-bit (256level) alpha prior to display.

The Text subtitle framework supports various presentation styles as shown in Figure 4-6:

- A "Text Region" is defined by position and size within the Graphic plane. Each Text Region can have a unique background color.
- A "Text box" is defined by position and size within a Text Region. Text boxes are used for the placement of text for display.
- Text boxes define the style of text for display as follows:
  - Text arrangement using text flow (e.g. left-to-top progression), text alignment (e.g. left alignment) and line space style attributes.
  - Font styles using font type (e.g. Arial, Courier), font style (e.g. bold, italic), font size and font color can be set for each Text Region.
- In addition to the Text box style, variations of font styles are allowed for character by character (known as "in-line" styles).

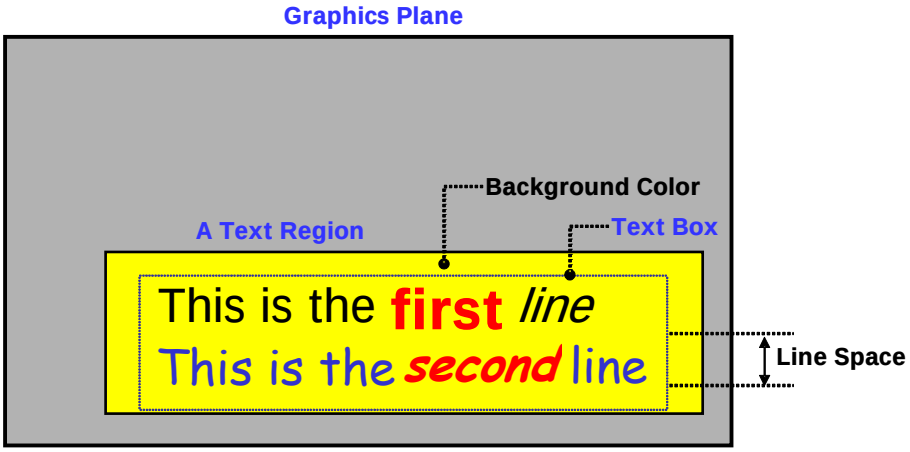


Figure 4-6 – Illustration of Text Region and Text Box

Figure 4-7 shows an example of some temporal effects which can be performed using the HDMV Text subtitle format. Fading of text is realized by simple Palette changes. Seamless presentation between individual Text subtitle displays is guaranteed.

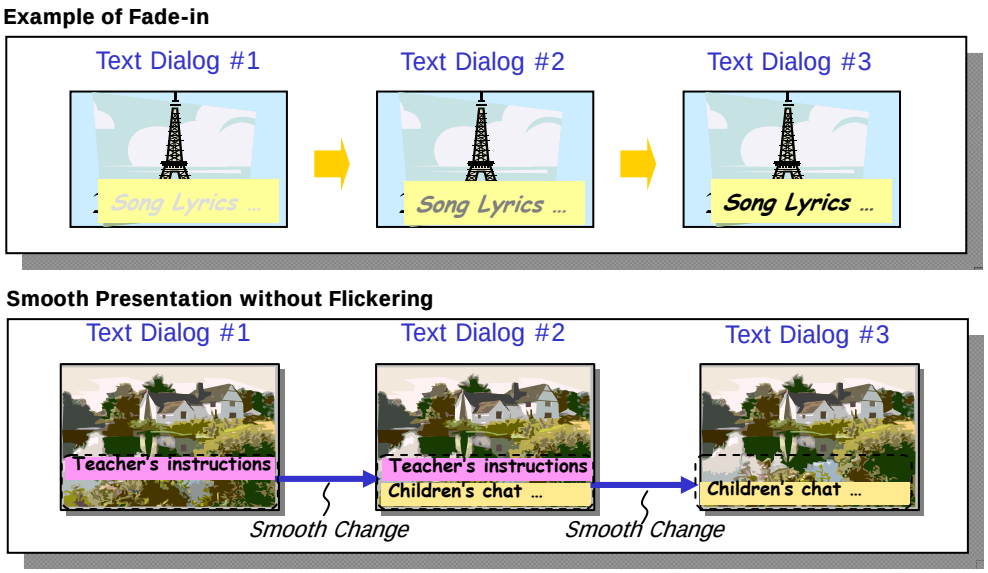


Figure 4-7 – Example of Text subtitle effects

4.1.4 Interactivity Framework

1) Pop-Up Menus

HDMV Interactive Graphics support a "Pop-Up" Menu Interface: once playback of video has begun, HDMV graphical interactive content may be activated during the playback of video by pressing a 'Pop-Up' Button on the remote. In this case, video playback can continue while the HDMV Interactive graphics are on the screen or video playback may be paused – this is determined by the Content Provider using navigation commands.

Menus that support a "Pop-Up" Menu Interface are always pre-loaded. As shown in Figure 4-8, several pages of HDMV Interactive Graphics data can be pre-loaded before playback starts. This Interactive Graphics data is kept in memory during playback of the AV stream and is not displayed until requested by the user.

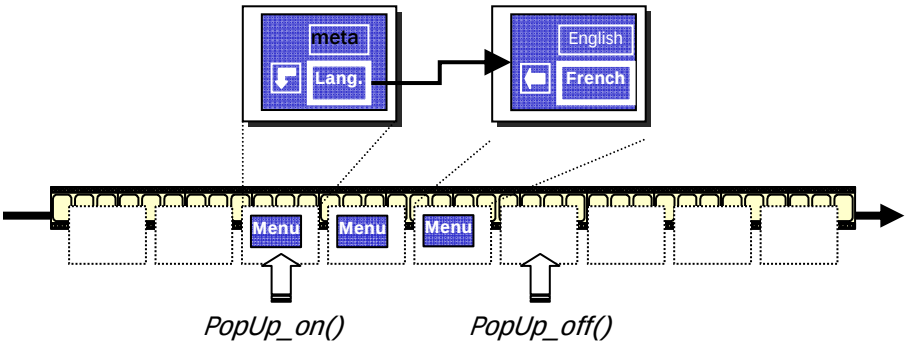


Figure 4-8 – Illustration of Pop-Up Menus

2) Always-On Menus

HDMV Interactive Graphics support an "Always-On" Menu Interface; Interactive Graphics content that cannot be removed from the screen by user request is called "Always-On". This is one of the methods provided by HDMV to present interactivity to the user and is similar to that provided by DVD-Video. For example, a Menu implemented with the Always-On interface may be presented to the user when the disc is inserted into the player.

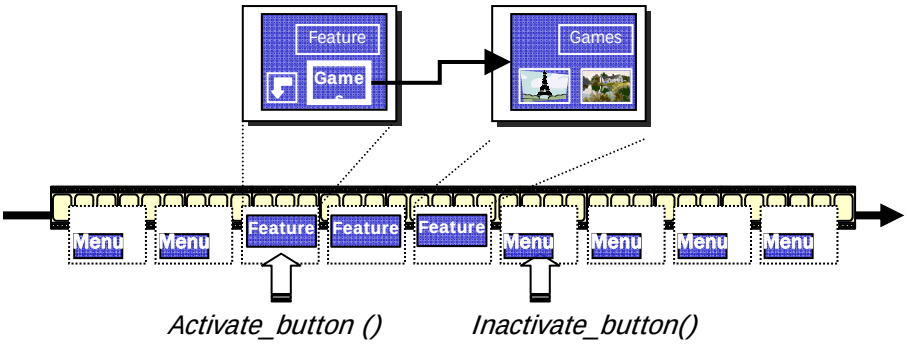


Figure 4-9 – Illustration of Always-On Menus

Menus that support an "Always-On" Menu Interface may be pre-loaded or multiplexed with video. If the HDMV Interactive Graphics stream is multiplexed with video, PTS/DTS timestamps can frame accurately determine when the Always-On Menu shall appear and disappear.

3) Multi-page Menus

The HDMV Interactive Graphics framework provides a scheme for Menu Page definition, thereby allowing a large amount of data to be presented in an organized manner with special commands available for inter-page navigation. When a Button is activated, a corresponding navigation command is executed which causes the display to change to a specified page. This action is performed with no visible interruption to the screen allowing a seamless user experience.

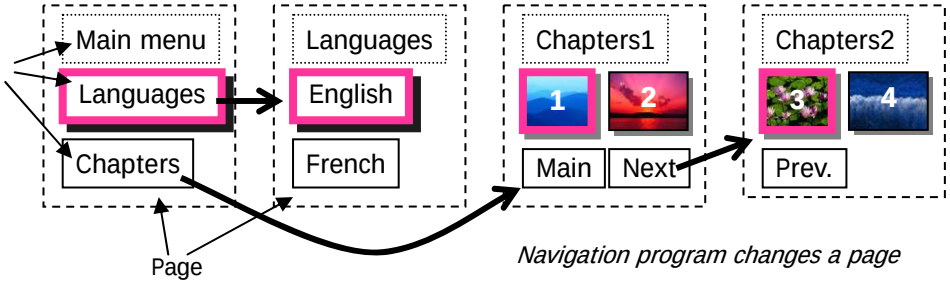


Figure 4-10 – Example of Multi-page Menu

4) Button enabling and disabling

The HDMV Interactive Graphics framework also provides a scheme for dynamic graphics display. On a single page, this enables the Content Provider to determine dynamically which Buttons are visible and invisible at any point in time. This scheme could be used, for example, to provide Buttons that present a set of options and when one of those Buttons is selected, additional Buttons appear. When a Button is enabled it becomes visible and can be navigated to. This action is performed with no visible interruption to the screen allowing a seamless user experience. The author may choose to either keep the earlier Buttons accessible or disable them which would clear them from the display.

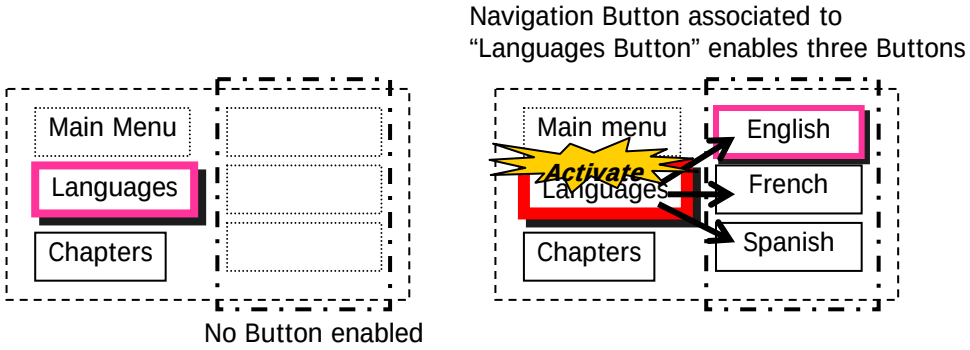


Figure 4-11 – Example of Button enabling and disabling

4.1.5 Command Framework

HDMV provides a simple programming platform to enable the Content Provider to author interactive movie contents, such as those seen in the DVD-Video market. This platform provides a scheme to manage the behavior of Menus, Browsable Slideshow pages and so on.

There are two types of Objects which contain navigation commands – the Movie Object and the Button Object. A Movie Object is executed when the Title associated with the Movie Object begins playback. Movie Object navigation commands are used to manage PlayList playback. While a PlayList is under playback, the state of a Movie Object is maintained and resumes after PlayList playback is terminated. A Button Object is an alternative programming method that is available while the PlayList is under playback and a Button Object is executed by user activation or system timer.

4.1.6 Programming commands and Registers

HDMV navigation commands have three operation groups: playback operation group, compare operation group, and arithmetical and bitwise operation group. The playback operation group manages PlayList playback, execution of Movie Objects, execution of Titles and control of the Graphics display

(Button enabling and disabling). The comparison operation group provides comparison functions between parameters and/or given values and provides a Boolean result.

The player has two types of Registers: General Purpose Registers and Player Status Registers. General Purpose Registers provide the Content Provider with 4096 4-bytes unsigned registers. Player Status Registers represent the Player's playback status, configuration and preferences.

4.1.7 Picture-in-Picture framework

Picture-in-Picture is the presentation of Secondary video stream overlaid on a Primary video stream. Picture-in-Picture application is supported both in HDMV mode and BD-J mode, but only with 2D video. Figure 4-16 shows an example of Picture-in-Picture application. In the example, a movie which displays a director is encoded as a Secondary video stream and it is overlaid on a Primary video stream as main movie. Secondary audio is also available in order to mix the director's commentary to a Primary audio stream in this example.

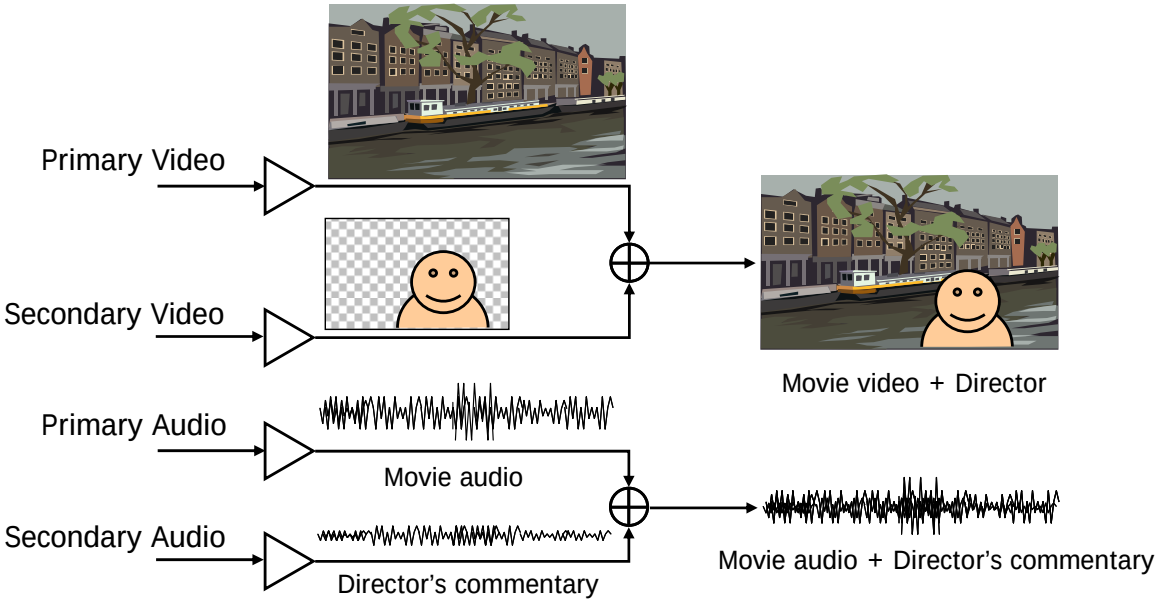


Figure 4-16 – Example of Picture-in-Picture

1) Synchronous type of Picture-in-Picture

Synchronous type of Picture-in-Picture provides a Secondary video stream which synchronizes with an associated Primary video stream. Typical application using this type of Picture-in-Picture is director's commentary with main movie.

2) Asynchronous type of Picture-in-Picture

Asynchronous type of Picture-in-Picture provides a Secondary video presentation that can start from any time on the timeline of the corresponding Primary video. Typical application using this type of Picture-in-Picture is a video pop-up menu.

4.2 Application Examples

4.2.1 Interactive Menus

The HDMV Interactive Graphics framework is used to provide interactive Menu displays. For instance, changing the display image of selected Buttons and changing the graphics display of the page (Buttons appearing and disappearing) with Button activation. This framework enables the author to provide flexible Menu navigation while the movie is presented.

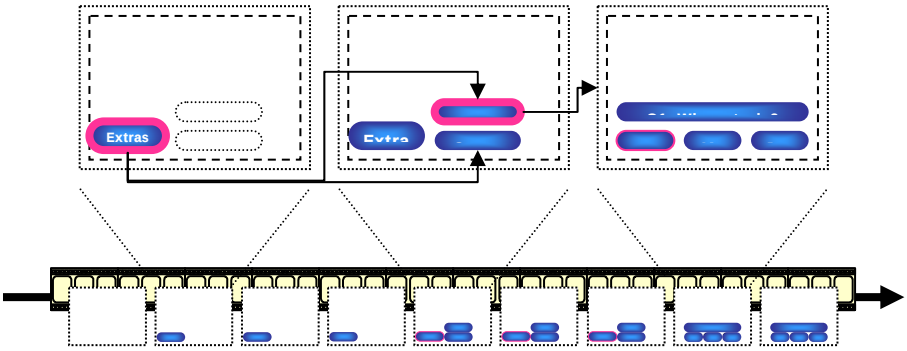


Figure 4-12 – Example of Multi-page Menu with dynamic Button display

4.2.1.1 Browsable Slideshow with background music

The HDMV decoder enables simultaneous decoding of still picture data and audio data. Since the decoding process of each is independent, the decoder enables the user to freely change (skip next or skip back) the still picture while not interrupting the audio presentation. This means that background music can continuously be presented during a Browsable Slideshow application.

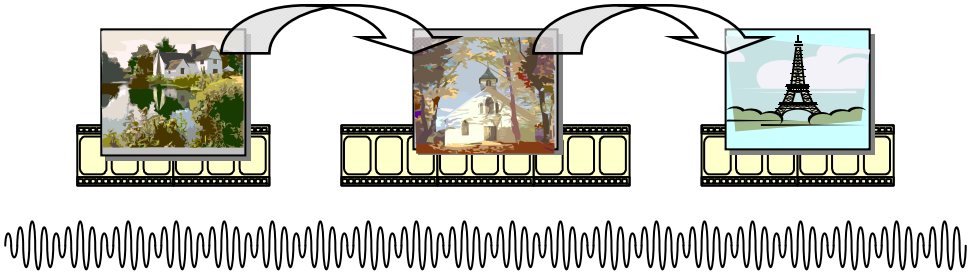


Figure 4-13 – Example of Browsable Slideshow with background music

4.2.1.2 Button sounds

The concept of Button Sounds is available in HDMV. Both the Select and Activate actions may be associated with short duration sounds which are mixed with the underlying audio.

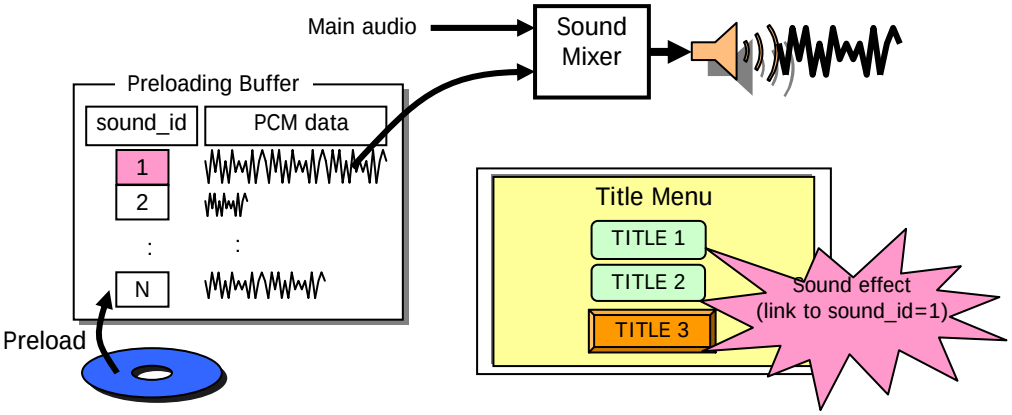


Figure 4-14 – Example of Button sounds

4.2.2 User Changeable Subtitle styles

In addition to the basic style control made available in the Text subtitle format, the Content Provider can also define a set of various presentation positions and font sizes for the display region as user selectable styles. In this case, the user can change the presentation style by selecting one style from the set of user selectable styles as shown in Figure 4-15 below.

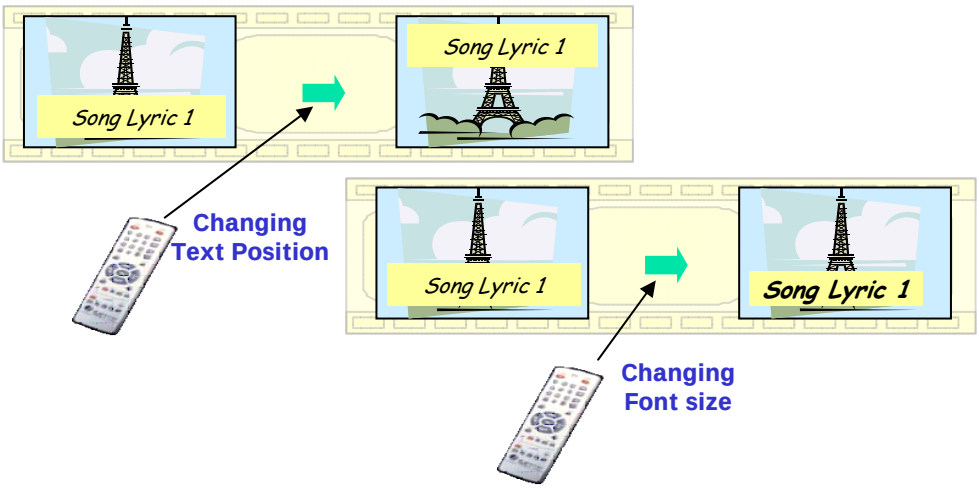


Figure 4-15 – User change to presentation styles

5 BD-J mode

This section will cover the main features of the BD-J mode platform which is shown in Figure 5-1 below.

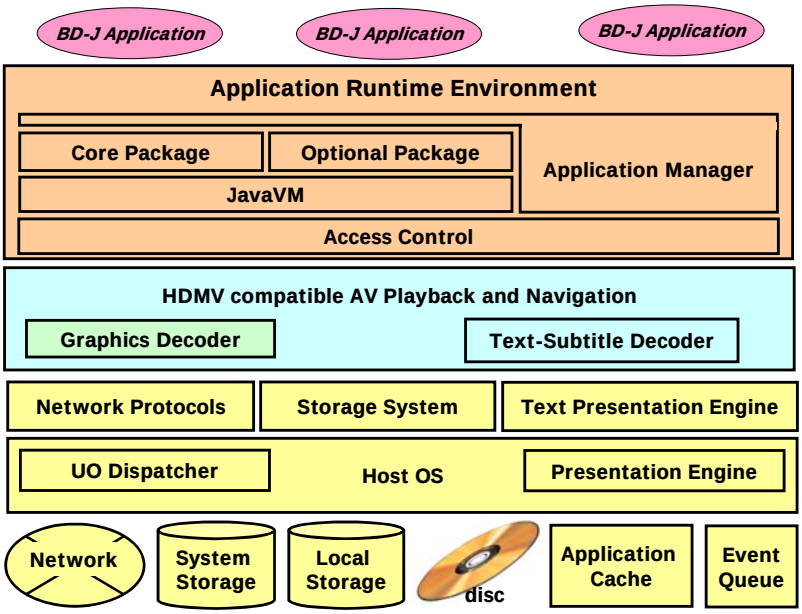


Figure 5-1 – Overall BD-J system model

BD-J is based on the Java 2 Micro-Edition (J2ME) Personal Basis Profile (PBP) – a Java profile that was developed for consumer electronics devices.

5.1 Core functions

5.1.1 Application Execution / Management

A key concept of BD-J is the BD-J Object. A BD-J Object is a Java Xlet that is registered in the Application Management Table (AMT). Each Title on a disc or even the disc itself can have an associated AMT.

At least one application in the AMT must be signaled as "autostart". This application will be started when the corresponding Title is selected and from thereon the BD-J platform is used by the BD-J application. This could include selecting another Title and launching other applications signaled in the AMT or downloading from the Internet.

5.1.2 GUI framework and User Interface

BD-J includes a GUI framework that is suitable for a CE environment. A BD-J application's GUI can be operated with a remote control with a required set of keys and an optional pointing device. The set of required keys includes at least the keys needed to support the User Operations in HDMV applications.

The GUI framework in BD-J is similar to the framework defined in HAVi[4] UI and is not a desktop GUI framework like Swing or AWT. The GUI framework will be based on the core of AWT, but the widget set includes mechanisms for remote control navigation and easy customization of look and feel.

5.1.3 Device model & functions like HAVi

BD-J includes a HAVi-like device model that maps to the BD-ROM system resources. One of the devices supported in the model is the Screen device that is build up of a Background Device, a Video Device and a Graphics Device. The configuration of the Screen and its constituent devices is under control of the BD-J application, as shown in Figure 5-2 below.

[4] HAVi is owned by the HAVi Organization. For more information, please refer to: http://www.havi.org/

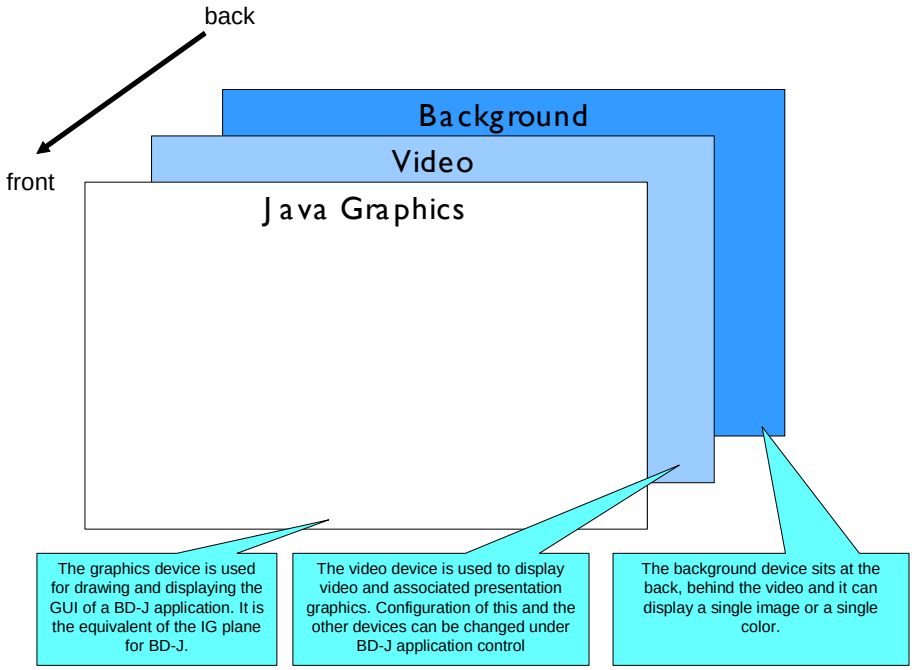


Figure 5-2 – BD-J system device model

Supported resolutions of the devices are compatible with the supported video format resolutions. The Graphics Device uses a 32-bit color RGB color model.

BD-J graphics can use alpha for overlay with video. Additionally, the video can be scaled behind the BD-J graphics and the video background device can display a single image.

5.1.4 AV Playback and Navigation and Subtitle/Audio Language Control

BD-J includes a media framework similar to JMF for the playback of media content related to the BD-ROM disc. It is assumed that the BD-ROM disc will be the prime source for media files, but it will not be the only one; other sources could be the studio's web server and local storage.

The unit of playback in BD-J is the Playlist, just as in HDMV. All features of HDMV, except Interactive Graphics, can be used by a BD-J Application. HDMV Interactive Graphics is replaced by BD-J graphics. Supported features include video, audio, Presentation Graphics, Text Subtitle component selection, media-time and playback-rate (trick-mode) control.

The BD-J Video Device is a combination of the HDMV Video and Presentation Graphics planes. Both Video and Presentation graphics will play back in the Video Device.

5.1.5 Other (static) content format functions (Graphics, Text, Audio Clips)

BD-J includes standard Java libraries for decoding and displaying images in JFIF (JPEG), PNG and other image formats. These images can be displayed on the Java graphics plane using standard Java graphics functions. An image can also be rendered in the background plane using a BD-J specific package.

Text can be rendered using standard Java text functions. These text-rendering functions are extended with a more advanced text layout manager that integrates with the BD-J UI framework. The text is rendered using a vector-based font either coming from the disc, the player (default font) or downloaded from the network.

Button sounds from HDMV can also be used by the Java UI framework. Sound files can be loaded and rendered as a reaction to the user pressing a key, or as a reaction on a marked event related to the movie - or as a reaction to any event generated by a BD-J Application.

5.1.6 Access control, security scheme, application authentication scheme

The BD-J environment uses the Java 2 security model to authenticate signed applications and to grant them permissions that go beyond the core functions (the BD-J defined sandbox).

The authentication scheme of BD-J applications is based on signing the JAR files that contain the applications. The relation between the authentication of BD-J applications coming from the disc and the BD Copy Protection System is out of scope for this paper, but an efficient workable scenario will be part of the BD-ROM Full Feature specification. The BD-J classloader will only load authenticated applications when the disc is in the player.

Authenticated applications can use a (signed) permission request file to acquire permissions that go beyond the BD-J sandbox. Permissions can be acquired for:

- Reading and writing to local and system storage
- Using the network connection (to connect to defined servers)
- Access of the file system on the BD-ROM disc
- Title selection of other titles on the BD-ROM disc
- Control of other running BD-J applications

5.1.7 Internet Connectivity & Download of New Contents/Applications

BD-J contains the Java network package. Java applications can use this package to connect to servers on the Internet. The physical connection might differ between implementations e.g. Ethernet, telephone line, etc. At the network level, TCP/IP is supported and the HTTP protocol may be used. Moreover, the Java package for secure connections is included (JSSE) as part of the BD-J platform. Before a BD-J application can use the network connection, it must be authenticated and have suitable permission to use the network.

The web sites to which the application will go are under full control of the Content Provider. This control is guaranteed in two ways:

- Only (disc) authenticated BD-J applications are allowed to run when the disc is played. The application controls the use of the network connection.
- In addition, permissions defined on the disc can restrict the use of the (TCP/IP) network connection to certain sites.

5.1.8 System/Local Storage

BD-J will include support for storage. Two flavors of storage are included – mandatory System Storage and optional Local Storage. All storage is accessed using methods from the Java IO package.

System storage is storage that will be present in all BD-J players. The required minimum size of this system storage will permit storage of application data like settings, high-scores etc. It will not be big enough to store downloaded AV material. For this purpose, optional local storage is available. Typically system storage will be implemented using Flash memory and the optional local storage will be implemented on a HDD.

Since storage is a shared resource between all discs played on the player, Java access control is part of BD-J. BD-J applications can only access a disc specific part of the storage space and cannot access the part belonging to other discs.

5.1.9 Binding scheme for on-the-disc and off-the-disc content

A binding scheme between media content (AV files, subtitles, Java applications files, database files) on the disc and content (related to the disc) stored on local storage is defined. This scheme enables a seamless user experience to be provided when playing back media data, regardless of the origin of the data.

5.2 Application Examples

BD-J allows many possible application types. In this section we will cover a few typical examples in more detail.

5.2.1 AV playback control

One of the main features of BD-J is playback of A/V material. A disc bound BD-J application can be created which is started when the disc is put into the player. This application could present a Menu on the screen, e.g. while playing an introduction of the movie in a scaled-down manner in a corner of the screen which allows language selection, chapter selection, and display of background information that might be retrieved from disc or from the Internet. Once the user selects playback of a Title, the disc application becomes invisible but allows the user to use trick modes with a simple on-screen GUI on top of the video (as long as the application on the disc allows this). The user also has the option of going back to the full-screen Menu of the disc application using one of the remote control keys.

BD-J features used in this example include: media control (including video scaling, playback speed, language component selection), GUI framework, and Internet connectivity.

5.2.2 Subtitle Updates

The BD-J application described above can be further extended to allow the user to obtain subtitles in a language that is not supported on the disc. The Content Provider can publish new or updated subtitle files on a website dedicated to the disc Title. The BD-J application on the disc can include the retrieval of this subtitle file and storage (in encrypted form) in the player's local storage. After storing the subtitle file, the application can select the new subtitles for a Title.

Only BD compliant players and trusted and authenticated applications will be able to do this and only from trusted and authenticated websites. The trust scheme will make use of the Java 2 security scheme and be tied to the CPS of the disc.

Additional BD-J features used in this example: downloading data into local storage, combined playback of subtitles from local storage with video from disc, merged file-system view, Java 2 security model.

5.2.3 Download new Movie trailer

When the Content Provider that published the disc is launching a sequel to the Title, they may also choose to publish a trailer for the sequel on their website, specifically for holders of the current title. A BD-J application, present on the disc, can connect to this website and see if there is new content available. The BD-J application can inform the user that a trailer for the new sequel movie is available e.g. by showing a number of (JPEG) images in the Main Menu. After the user has selected to view the trailer, the BD-J application downloads this trailer, while at the same time showing some background information on the actors in this sequel. When the download of the trailer to local storage is completed, the application plays it back, showing at the bottom of the screen the movie theatres where this movie can be seen.

Additional BD-J features used in this example: downloading A/V material to local storage, playback of A/V material from local storage, display of (JPEG) images from local storage, retrieval and usage of user information (for the display of localized information).

5.2.4 Play games on the disc and also online game

BD-J is not only a good solution for flexible media-playback from disc and from the Internet, it can also be used for games. A disc can contain, besides the movie Title, a Title that contains a set of games. The Java application associated with the Title displays the Menu of available games. The set of games can be a combination of games coming from the disc and games downloaded in JAR files from the Content Provider's website. Games can retrieve high scores from the Internet and achieving a new high-score can result in the user's alias appearing in the updated game results. Game applications can make use of the Java graphics and UI input features of the Java programming environment.

Additional BD-J features used in this example: multiple application support, Java graphics, user input (keys, optional pointing device).

5.2.5 Advanced Applications

With the features described above it is possible to create new Advanced Applications, for example:

- An online shopping application that may allow the end-user to buy Title related merchandising.
- Chat applications that may allow on-line discussion with other purchasers of the same Title.

5.2.6 Application Illustrations

Figure 5-3 below, further illustrates potential BD-J application types. This illustration includes an application that allows a movie director to give comments on the movie, to control playback of the disc and to point to certain items in the video. Note that this does not have to be a live commentary, but can be scripted at the server side.

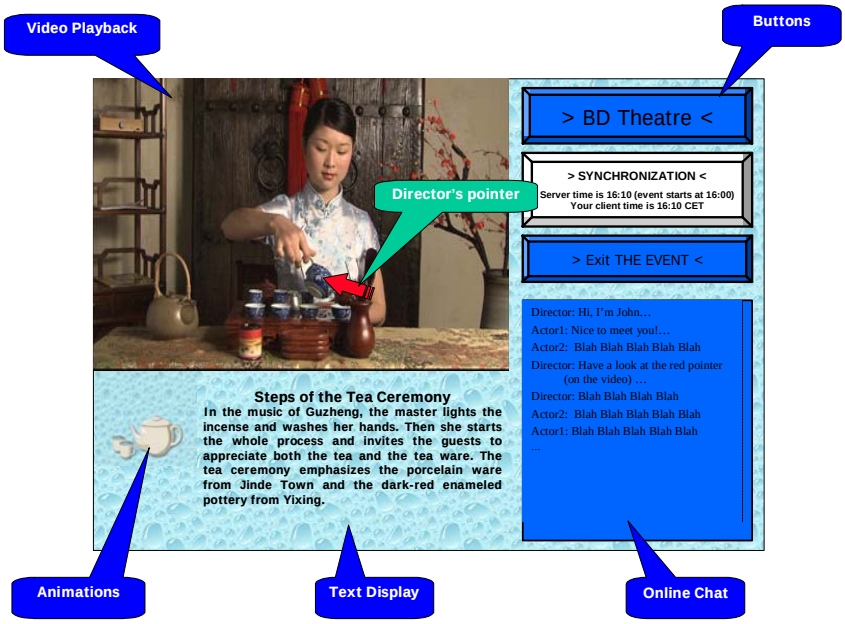


Figure 5-3 – Example of BD-J application

The four pictures below illustrate the use of multiple concurrent applications. One typical example of this is a main application that controls media playback and a second application that displays some information transparently on top of the video. The main game Menu that allows launching various games is another example.

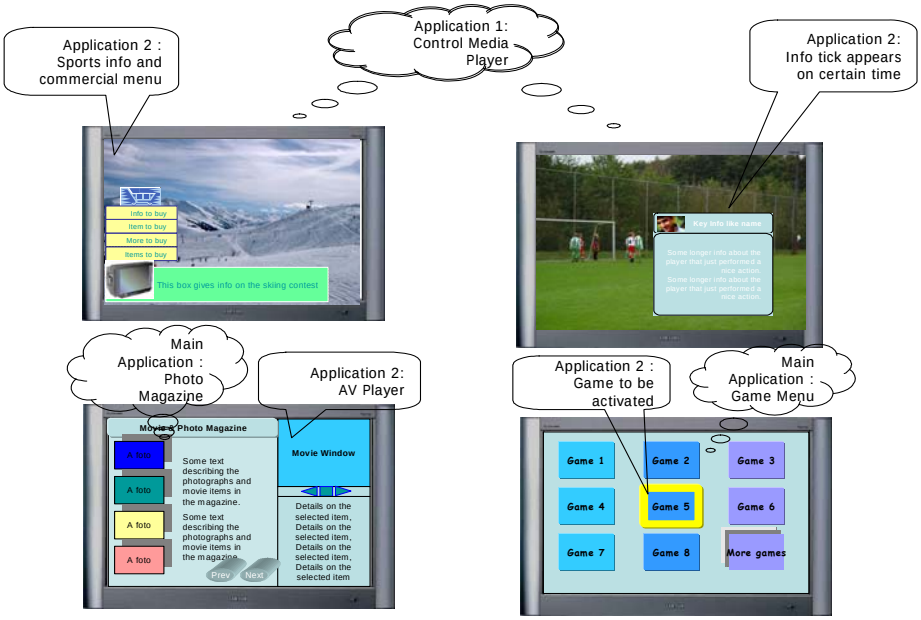


Figure 5-4 – Example of multiple concurrent BD-J applications

6 Overview of Stereoscopic 3D

Full-HD 2D movie experience has been realized through Blu-ray Disc application since 2006. Lately 3D is getting more and more popular in theatrical experience. To make 3D viewing a pleasant experience in homes and to launch 3D movie market in 2010 from users/studios requirement, feasible solutions have been sought after to overcome different issues. These mainly concern picture quality, video coding efficiency, interference between 3D video, 3D subtitle and 3D menu, and backward compatibility with existing 2D players for packaged media application.

3D players introduced the concept of output mode. The output mode is either 2D Output Mode or Stereoscopic Output Mode. 3D players in 2D Output Mode support all functions of Profile2 (See section 7) supported in the current 2D players so that 3D players can playback 2D discs. 3D players in Stereoscopic Output Mode support enhanced 3D technologies in order to playback 3D discs.

Blu-ray 3D standard includes the following key technologies to overcome the concerns above.

- Stereoscopic 3D provides Full HD picture to each eye to realize same picture quality as 2D.
- MPEG-4 MVC video codec is introduced to achieve very high coding efficiency. This technology makes it possible to store over two hours of Full HD 3D content on a 50 GB Blu-ray disc.
- Depth control technology is introduced to avoid interference between 3D video, 3D subtitle and 3D menu.

6.1 Backward compatibility with 2D BD-ROM players

BD-ROM specification for 2D was released in 2006, and many BD-ROM products (Players, BD-ROM movie titles etc) have been released and many consumers are enjoying BD-ROM. For smooth introduction of Blu-ray 3D, backward compatibility with current 2D players is one of the important points from consumer and studio perspective.

Based on the background above, Blu-ray 3D specification is carefully designed to keep backward compatibility. As a result, the backward compatibility shown in Figure 6-1 is realized if the BD-ROM 3D disc is authored this way. Content author can make 2D/3D compatible disc for both 2D and 3D player.

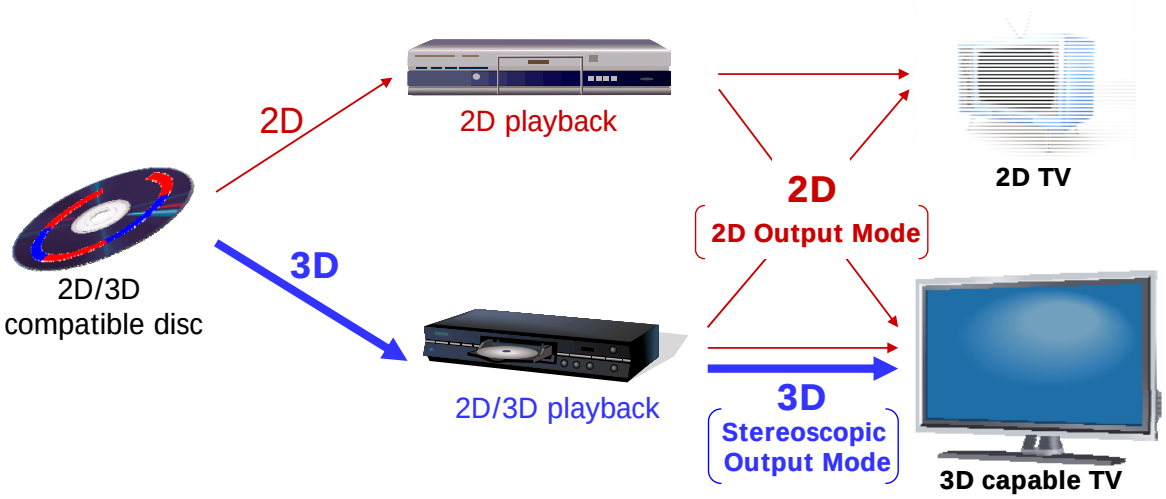


Figure 6-1 – Backward compatibility with 2D/3D compatible disc, BD-ROM Player and TV

6.2 Elementary streams in BDAV MPEG-2 Transport Stream for Blu-ray 3D

Video, audio, graphics and text elementary streams are coded in the PES packet payload of the BDAV MPEG-2 Transport Stream. The coding method for each of these elementary streams is specified in Figure 6-2 below.

Name of elementary stream	Coding method of elementary stream
Video stream	MPEG-4 MVC video stream
Audio stream	Same as 2D (See Figure 3-2)
Graphics stream	One plane plus offset Presentation graphics stream
	Stereoscopic Presentation graphics stream
	One plane plus offset Interactive graphics stream
	Stereoscopic Interactive graphics stream

Figure 6-2 – Elementary streams in the BDAV MPEG2 Transport Stream for Blu-ray 3D

6.2.1 Video stream

Video streams for Blu-ray 3D shall be MPEG-4 MVC video format (Annex H of ITU-T H.264 | ISO/IEC 14496-10) compliant.

MPEG-4 MVC video stream used on BD-ROM disc is composed of two parts. One is MPEG-4 MVC Base view video stream and the other is MPEG-4 MVC Dependent view video stream. The MPEG-4 MVC Base view video stream is compatible with MPEG-4 AVC video stream so that 2D players can decode MPEG-4 MVC Base view video stream for 2D playback. Content creator can choose that either the left or right eye stream is the MPEG-4 MVC Base view video stream to be used by a 2D player as the source for the backward compatible 2D stream.

The video formats shown in Figure 6-3 can be used for Blu-ray 3D. Not all 3D players support 50Hz video format.

Video	CODEC	MPEG-4 MVC
	Max. bitrate	MPEG-4 MVC Base view: 40Mbps MPEG-4 MVC Dependent view: 40Mbps MPEG-4 MVC Base view + MPEG-4 MVC Dependent view : 60Mbps
	3D Video format	1920x1080x23.976-p x 2 1280x720x59.94-p x 2 1280x720x50-p x 2 (Note)
	2D Compatibility	MPEG-4 MVC Base view is 2D player compatible

Note: if a 50Hz TV system is used for both HDTV and SDTV, as it is the case for Europe.

Figure 6-3 - Specification of BD-ROM Video streams for Blu-ray 3D

6.2.2 Coding efficiency of MPEG-4 MVC codec

Inter-view prediction is exploited to achieve high efficiency in video compression in addition to intra-view prediction. (See Figure 6-4) MPEG-4 MVC encoding technology is able to achieve about 50% bit rate reduction based on subjective picture quality for the MPEG-4 MVC Dependent view as compared to the simulcast case (i.e., separate encoding of the left-eye picture and right-eye picture).

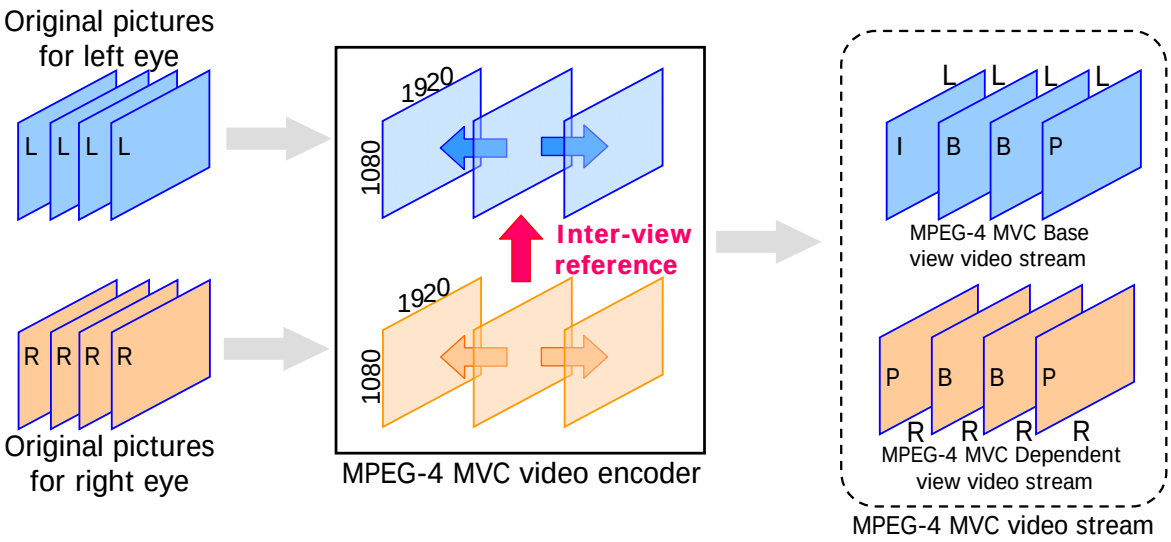


Figure 6-4 – 3D video compression by MPEG-4 MVC

6.2.3 BDAV MPEG-2 Transport Streams for Blu-ray 3D

In order to keep backward compatibility with 2D BD-ROM players and realize full HD picture quality, two BDAV MPEG-2 Transport Streams are used for Blu-ray 3D (Figure 6-5). One is 2D compatible Transport Stream (main TS) and the other is 3D extended Transport Stream (sub TS).

The main TS includes MPEG-4 MVC Base view video stream and other elementary stream(s) necessary for 2D playback. The main TS is fully compatible with 2D players. The sub TS includes MPEG-4 MVC Dependent view video stream and other elementary stream(s) necessary for 3D playback. Both main TS and sub TS are decoded simultaneously for 3D playback.

The sum of maximum multiplex rate of main TS and sub TS is 64Mbps while the maximum multiplex rate of each TS (main TS and sub TS) is 48Mbps in order to realize high quality full HD picture to each eye.

In addition to above, both a MPEG-4 MVC Base view stream and the corresponding MPEG-4 MVC Dependent view stream can be multiplexed in one TS (Note: This TS is less than or equal to 48Mbps TS rate for backward compatibility).

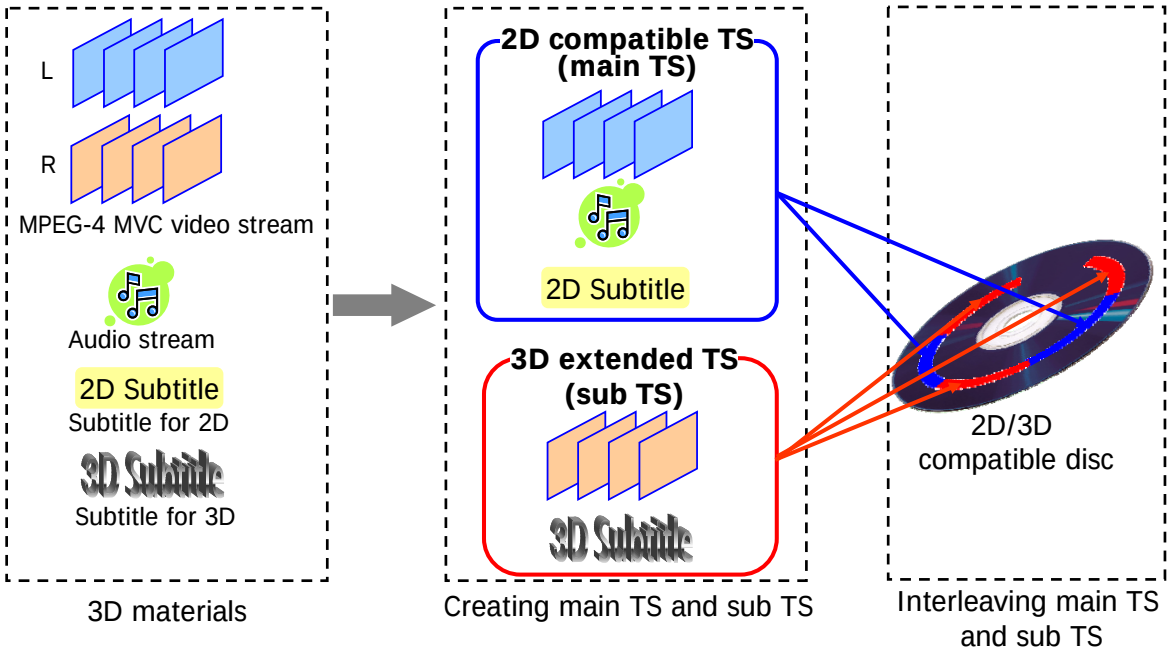


Figure 6-5 – Two MPEG-2 Transport Streams for 2D/3D compatible disc

6.2.4 Interleaving of 2D compatible TS (main TS) and 3D extended TS (sub TS)

Two TSs, main TS and sub TS, described in section 6.2.3 are divided into blocks. Typically each block contains about a few seconds of AV data, and blocks of the main TS and blocks of sub TS are interleaved on a disc (Figure 6-6).

When 2D/3D compatible disc is played back in 2D player, only main TS is read from the disc jumping sub TS blocks and the main TS is played back as 2D (2D playback path in Figure 6-6). Jump distance(=size of each block of sub TS) is carefully designed to satisfy BD-ROM drive performance of existing 2D players.

When 2D/3D compatible disc is played back in 3D player with Stereoscopic Output Mode, both main TS and sub TS are read from the disc continuously and these streams are played back as 3D (3D playback path in Figure 6-6). In order to read blocks of main TS and blocks of sub TS continuously, Stereoscopic Interleaved file (zzzzz.ssif) is introduced. 3D BD-ROM player reads the Stereoscopic Interleaved file in which blocks from main TS and blocks from sub TS are interleaved.

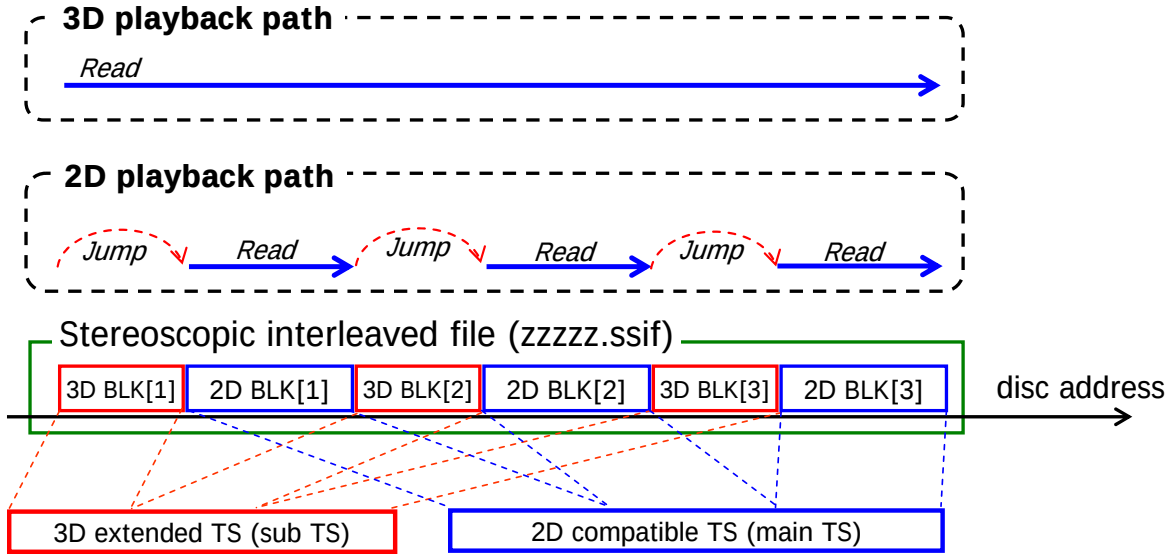


Figure 6-6 – Data allocation of 2D compatible TS and 3D extended TS

6.3 3D graphics with 3D video

Blu-ray 3D has the following two presentation types for the various 3D graphics formats with 3D video.

- One plane plus offset presentation type
  - This presentation type is prepared for ease-of-authoring and reduced content production cost. Object(s) for menu or subtitle can be shared for 2D and 3D in this presentation type.
  - Consumers can see flat objects at the same depth from the screen.
- Stereoscopic presentation type
  - This presentation type is prepared for more sophisticated 3D presentation. Independent objects are drawn to two planes, one of which is for left eye and the other of which is for right eye.
  - Consumers can see 3D objects each of which has different depth.

Content authors can choose which presentation type they want to use for each graphics layer, e.g. One plane plus offset presentation type for PG, Stereoscopic presentation type for IG or Stereoscopic presentation type for BD-J graphics and Background.

Note: Not all 3D players support BD-J graphics, Background and Graphics in Stereoscopic Output Mode.

6.3.1 One plane plus offset presentation type

One plane plus offset presentation type can be used for Presentation graphics (PG) stream, Interactive graphics (IG) stream, Text subtitle stream and BD-J graphics. Decoder model for the one plane plus offset Presentation graphics (PG) stream and one plane plus offset Interactive graphics (IG) stream is composed of the following two parts (Figure 6-7).

- Graphics decoder explained in section 4.1.2.
- Offset function connected to the graphics plane

Structure of one plane plus offset Presentation and Interactive graphics stream

Elementary stream structure of one plane plus offset Presentation/Interactive graphics stream is same as Presentation/Interactive graphics stream for 2D respectively, and the difference is only in the decoder model.

Decoder model for one plane plus offset Presentation/Interactive graphics stream

Decoder model for one plane plus offset graphics has an offset function in addition to decoder function for graphics stream for 2D. Thus Presentation/Interactive graphics stream for 2D can be reused as one plane plus offset Presentation/Interactive graphics stream for 3D.

Drawing model for one plane plus offset BD-J Graphics and Background

Drawing model for one plane plus offset BD-J Graphics and Background is same as for 2D so that same graphics assets and/or BD-J programs can be reused for 3D.

Note: Background is the lowest graphics layer so that offset value is not applied.

Graphics plane is shifted by offset sequence (described in 6.3.3) in the offset function before the graphics plane is overlaid onto 3D video. For example, when subtitle on the PG plane is expected to be presented between screen and user, right-shifted PG plane is overlaid onto the left video, and left-shifted PG plane is overlaid onto the right video.

Depth of graphics plane is controlled by the offset sequence frame by frame basis to keep a certain distance from the 3D video. From 3D content production point of view, this depth adjustment by appropriate offset sequence is important to avoid interference between 3D video, menu and subtitles.

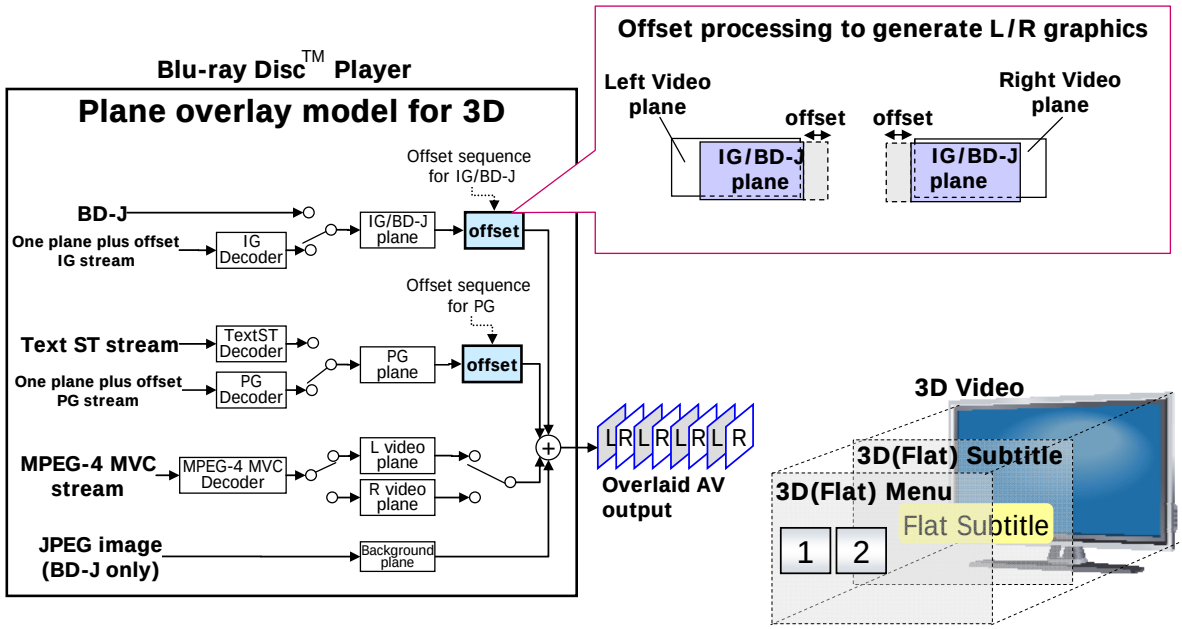


Figure 6-7 – Plane overlay model for one plane plus offset presentation type

6.3.2 Stereoscopic presentation type

Stereoscopic presentation type can be used for Stereoscopic Presentation/Interactive graphics stream, Stereoscopic Text ST stream, BD-J graphics and background image(BD-J only).

Structure of Stereoscopic Presentation and Interactive graphics stream

Stereoscopic graphics stream is composed of independent 2D graphics streams called "Left-eye Graphics stream" and "Right-eye Graphics stream". Structure of the Left-eye Graphics stream and Right-eye Graphics stream is same as graphics stream for 2D described in section 4.1.2.

Decoder model for Stereoscopic Presentation and Interactive graphics stream

Decoder of Stereoscopic Graphics stream is composed of Left-eye Graphics decoder and Right-eye Graphics decoder. Decoder model of Left-eye Graphics decoder and Right-eye Graphics decoder is same as decoder model of graphics decoder for 2D described in section 4.1.2.

Drawing model for Stereoscopic BD-J Graphics and Background

Drawing model for Stereoscopic BD-J Graphics and Background is extended to draw left eye image and right eye image synchronously in order to avoid image inconsistencies between left eye and right eye.

Graphics planes for left-eye and right-eye are shifted by offset sequence (described in 6.3.3) in the offset function before the graphics plane is overlaid onto 3D video to keep certain distance from 3D video.

In this presentation type, all objects can be 3D and depth of one object can be different from another object.

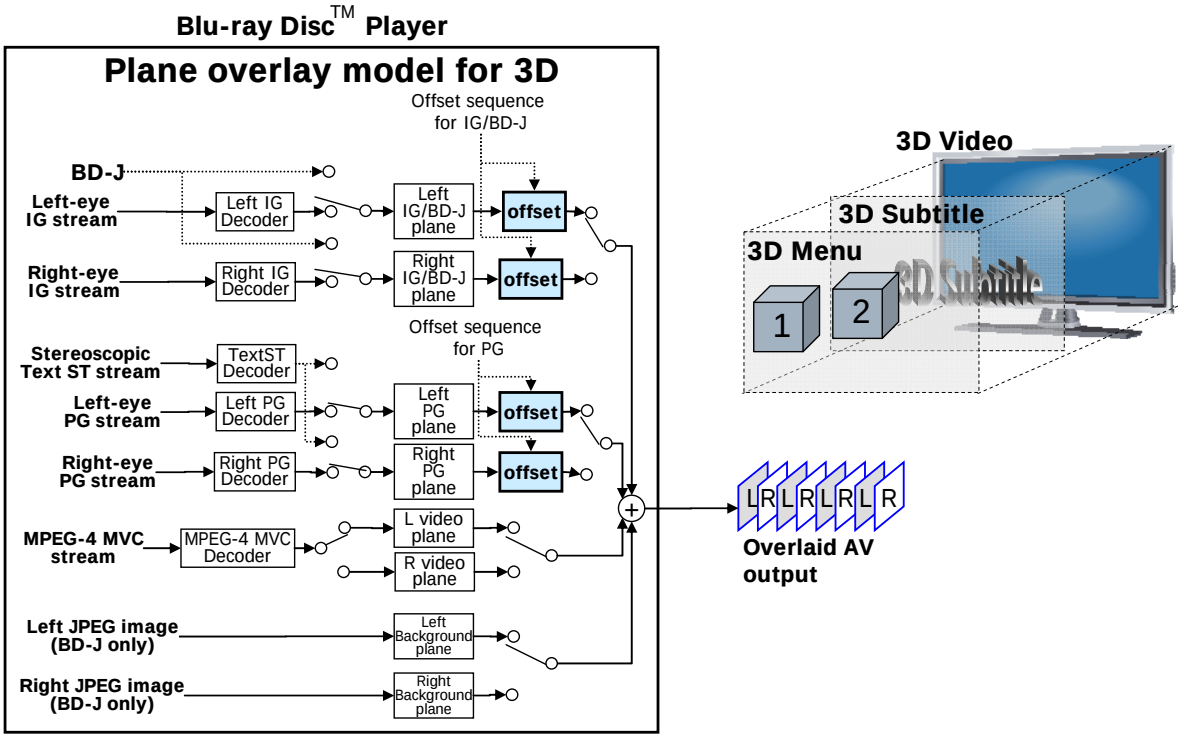


Figure 6-8 – Plane overlay model for Stereoscopic presentation type

6.3.3 Offset metadata for menu and subtitles

If depth position of 3D menus and subtitles overlaid on 3D video are not properly controlled, they sometimes appear to be behind the video, which can cause interference between 3D video, menu and subtitles. To solve this problem, offset metadata is introduced, by which depth position of 3D menu and subtitles can be adjusted to the depth for every video frame.

Offset metadata is a list of the offset sequences for Presentation Graphics plane, Interactive Graphics plane and BD-J Graphics plane. This offset metadata is used to set offset value for the planes when the PG/Text ST and/or IG/BD-J is presented during 3D playback.

The offset metadata is transported in a SEI message of the first picture of each GOP of MPEG-4 MVC Dependent view video stream(Figure 6-9). Offset metadata includes plural offset sequences, and each graphics type (PG/TextST, IG and BD-J) is associated with one of the offset sequences by offset sequence id in a PlayList (PG/TextST or IG) or application setting (BD-J).

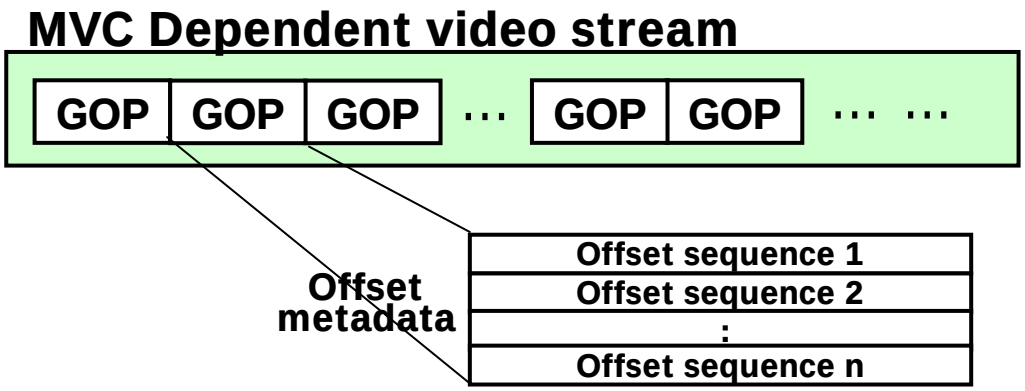


Figure 6-9 – Offset metadata in MPEG-4 MVC Dependent view video stream

6.4 Stereoscopic 3D capability in BD-J mode

Stereoscopic 3D features described above (such as MPEG-4 MVC video playback, one plane plus offset presentation type, stereoscopic presentation type and offset metadata) are utilized by BD-J applications via BD-J APIs extended for Stereoscopic 3D.

For example, BD-J application can control the presentation type of BD-J graphics, Video and Background independently via BD-J APIs extended for Stereoscopic 3D.

Note: Not all 3D players support Stereoscopic 3D capability in BD-J mode.

7 Player profile

Since its release in 2006, new features have been added to the BD-ROM application standard. In order to distinguish supported features of BD-ROM player, BD-ROM introduces four kinds of Profiles which indicate what features are supported in a BD-ROM player (Figure 7-1). For example, BD-J network access was added for Profile2. Figure 7-1 shows a relation between each Profile and supported features. User can easily know what features are supported by designated logo indication on the BD-ROM title and player.

Note: Profile 3 and 4 are not shown in the Figure 7-1.

- Profile 3 is prepared for audio only BD-ROM players.
- Profile 4 is reserved on purpose.

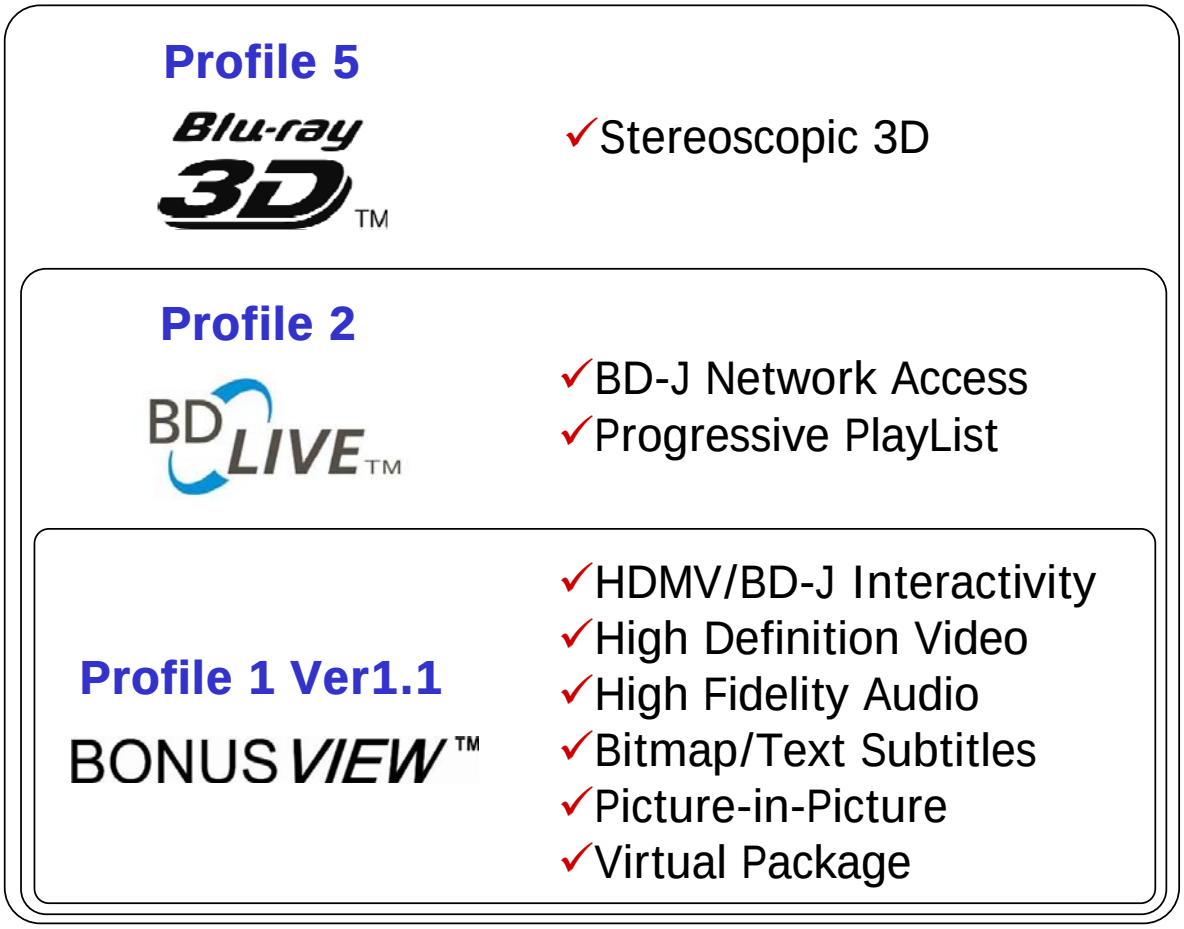


Figure 7-1 – Profiles and supported features

Annex A MPEG-4 MVC video streams

This section contains the coding constraints on MPEG-4 MVC video streams as defined in the BD-ROM specification. This information is intended to be used for the development of other 3D applications, not limited to Blu-ray Disc, with generally the same constraints to improve interoperability.

HDMV MPEG-4 MVC video stream shall comply with Annex H of ITU-T H.264 | ISO/IEC 14496-10.
HDMV MPEG-4 MVC video stream is composed of Base view video and Dependent view video.
HDMV MPEG-4 MVC Base view video streams shall conform to constraints on HDMV MPEG-4 AVC video streams (See BD-ROM spec). Additional constraints on MPEG-4 MVC Base view video stream and constraints on HDMV MPEG-4 MVC Dependent view video streams are specified in this Section.

A.1 General constraints

HDMV MPEG-4 MVC video streams shall conform to the following constraints.

- Definitions
  - HDMV MPEG-4 MVC video stream: video stream which comply with Annex H of ITU-T H.264 | ISO/IEC 14496-10 with additional constraints defined in this section
  - HDMV MPEG-4 MVC Base view video stream: Base view video stream which complies with Annex H of ITU-T H.264 | ISO/IEC 14496-10 with additional constraints defined in BD-ROM spec and this section.
  - HDMV MPEG-4 MVC Dependent view video stream: non-base view video sub-bitstream which complies with Annex H of ITU-T H.264 | ISO/IEC 14496-10 with additional constraints defined in this section.
  - View component: A coded representation of a view in a single access unit.
  - Base view component: A coded representation of Base view in a single access unit.
  - Dependent view component: A coded representation of Dependent view in a single access unit.
  - Corresponding view component: Opposite (Base/Dependent) view component with same PTS.
  - Dependent unit: A set of NAL units that are consecutive in decoding order and contain exactly one non-Base view component. A dependent unit starts from a MVC Dependency representation delimiter NAL unit.
- The following conditions shall not change in an HDMV MPEG-4 MVC video stream carried within the transport packets with the same PID value in an AV stream file.
  - profile_idc
  - level_idc
  - pic_width_in_mbs_minus1
  - pic_height_in_map_units_minus1
  - aspect_ratio_idc
  - Frame-rate = time_scale / num_units_in_tick / 2
  - CpbSize[cpb_cnt_minus1], which is derived from cpb_size_scale and cpb_size_value_minus1, if hrd_parameters() is present. If hrd_parameters() is not present, CPB size shall not change.
  - BitRate[cpb_cnt_minus1], which is derived from bit_rate_scale and bit_rate_value_minus1, if hrd_parameters() is present. If hrd_parameters() is not present, maximum input bit-rate to CPB shall not change.
  - frame_mbs_only_flag
  - entropy_coding_mode_flag
  - view_id - nal_unit_header_mvc_extension()
  - number_of_offset_sequences
- The following conditions of Corresponding Base view video streams and Dependent view video streams shall be the same.
  - level_idc
  - pic_width_in_mbs_minus1
  - pic_height_in_map_units_minus1
  - aspect_ratio_idc
  - Frame-rate = time_scale / num_units_in_tick / 2
  - frame_mbs_only_flag
  - entropy_coding_mode_flag

- Allowed combinations of parameters for MPEG-4 MVC video streams are defined in Table A- 1.

Table A- 1– Allowed combinations of parameters for MPEG-4 MVC video formats

horizontal size of frame	vertical size of frame	pic_width_ in_mbs_ minus1	pic_height_ in_map_ units_ minus1	frame_ mbs_ only_flag	frame-rate	progressive/ interlace
1920	1080	119	67	1	23.976 (24000/1001)	progressive
1280	720	79	44	1	59.94 (60000/1001)	progressive
1280	720	79	44	1	50	progressive

A.2 Additional constraints on MPEG-4 MVC Base view video stream

A.2.1 General Constraints

- The following fields shall have the following pre-determined values.
  - Sequence Parameter Set
    - profile_idc shall be set to 100.
- pic_parameter_set_id in MPEG-4 MVC Base view video stream shall refer to PPS only in the MPEG-4 MVC Base view video stream.
- The pic_parameter_set_id used by PPS in MPEG-4 MVC Base view component shall not be used by PPSs in the corresponding MPEG-4 MVC Dependent view component.
- seq_parameter_set_id in MPEG-4 MVC Base view video stream shall refer to Sequence Parameter Set only in the MPEG-4 MVC Base view video stream.
- The seq_parameter_set_id used by SPS in MPEG-4 MVC Base view video stream shall not be used by Subset SPS in the MPEG-4 MVC Dependent view video stream.

A.2.2 Prohibited NAL unit

Following NAL units shall not be present.

- Prefix NAL unit
  - The fields of nal_unit_header_mvc_extension() for MPEG-4 MVC Base view video stream shall be assumed as the following values.
    - non_idr_flag is set according to nal_unit_type of corresponding Base view component. If nal_unit_type of Base view component is set to 5, non_idr_flag is set to 0, otherwise non_idr_flag is set to 1.
    - priority_id is set to 0
    - view_id is set to 0
    - temporal_id is set to the same value as Corresponding view component of MPEG-4 MVC Dependent view video stream.
    - anchor_pic_flag is set to the same value as Corresponding view component of MPEG-4 MVC Dependent view video stream.
    - inter_view_flag is set to 1.
- Coded slice extension NAL unit
- Subset Sequence Parameter set NAL unit

A.2.3 Prohibited SEI message

Following SEI message shall not be present.

- Non-required view component SEI message
- View dependency change SEI message
- MVC scalable nesting SEI message

A.3 Additional constraints on MPEG-4 MVC Dependent view video stream

A.3.1 General Constraints

- The following fields shall have the following pre-determined values.
  - Subset Sequence Parameter Set(Subset SPS)
    - num_level_values_signalled_minus1 shall be set to 0 - seq_parameter_set_mvc_extension()
    - vui_mvc_num_ops_minus1 shall be set to 0 - if mvc_vui_parameters_extension() exist
    - vui_mvc_low_delay_hrd_flag shall be set 0 - if exist
    - vui_mvc_pic_struct_present_flag shall have same value as pic_struct_present_flag in VUI parameters of SPS for corresponding Base view component.
  - Sequence Parameter Set
    - profile_idc shall be set to 128.
  - Coded slice extension NAL unit
    - svc_extension_flag shall be set to 0
    - view_id of nal_unit_header_mvc_extension shall not be set to 0.
- SPS encoded in Subset SPS, shall have the same values as SPS for corresponding MPEG-4 MVC Base view video stream except for following conditions
  - seq_parameter_set_id
  - profile_idc
- level_idc encoded in Sequence parameter set MVC extension, seq_parameter_set_mvc_extension(), shall have the same value as level_idc encoded in SPS in Subset SPS.
- MVC VUI parameters, mvc_vui_parameters_extension(), encoded in Subset SPS shall have the same values as VUI parameters in SPS for corresponding MPEG-4 MVC Base view video stream except for the following conditions:
  - hrd_parameters() – if exist
- VUI parameters, vui_parameters(), encoded in SPS in Subset SPS for MPEG-4 MVC Dependent view video stream shall have the same values as VUI parameters in SPS for corresponding MPEG-4 MVC Base view video stream except for the following conditions:
  - hrd_parameters() – if exist
  - max_dec_frame_buffering - if exist
  - num_reorder_frames - if exist
  - max_bytes_per_pic_denom - if exist
- The sum of NumBytesInNALunit for the Dependent unit shall have the same constraint as MPEG-4 AVC High Profile video stream.
- SliceRate shall have the same constraint as MPEG-4 AVC High Profile video stream.
- In case of level 4 and 4.1, the size of CPB for MPEG-4 MVC Dependent view video shall be less than or equal to 30000 [1000bits].
- Minimum compression ratio (MinCR) for MPEG-4 MVC Dependent view component of level 4.1 shall be MinCR=4 for both movie stream and still picture.
- pic_parameter_set_id in MPEG-4 MVC Dependent view video stream shall refer to PPS only in the MPEG-4 MVC Dependent view video stream.
- The pic_parameter_set_id used by MPEG-4 MVC Dependent view component shall not be used by PPSs in corresponding MPEG-4 MVC Base view component.
- seq_parameter_set_id in MPEG-4 MVC Dependent view video stream shall refer to Subset Sequence Parameter Set only in the MPEG-4 MVC Dependent view video stream.
- The seq_parameter_set_id used by Subset SPS in MPEG-4 MVC Dependent view video stream shall not be used by SPS in corresponding MPEG-4 MVC Base view video stream
- If decoded reference picture marking syntax is coded in a picture in MPEG-4 MVC Base view video stream, the same decoded reference picture marking syntax shall be coded in the corresponding picture in a MPEG-4 MVC Dependent view video stream, i.e. DPB management is done in the same way between the base view and the dependent view.
- If decoded reference picture marking syntax is repeated using a Decoded reference picture marking repetition SEI in a view component in a MPEG-4 MVC Base view video stream, the same syntax shall be repeated in the Corresponding view component in a MPEG-4 MVC Dependent view video stream by using a Decoded reference picture marking repetition SEI.

- When Buffering Period SEI and/or Picture Timing SEI are encoded in MPEG-4 MVC Base view video stream, same SEIs shall be encoded in MVC scalable nesting SEI with same values for MPEG-4 MVC Dependent view video stream except for seq_parameter_set_id.
- HRD conformance
  - MPEG-4 MVC Dependent view video stream shall conform to Type 2 (NAL level) HRD conformance.
  - It shall comply with output timing conformance.
  - It shall be guaranteed that the HRD conformance can be verified by using PTS and DTS coded in system layer, i.e. it is not mandatory to carry Buffering period SEI and Picture timing SEI for the verification of the conformance.
  - The HRD parameters, hrd_parameters(), if exist, in VUI parameters, vui_parameters(), in SPS encoded in Subset SPS for MPEG-4 MVC Dependent view video stream shall conform to HRD conformance for MPEG-4 MVC Dependent view component independently.
  - The HRD parameters, hrd_parameters(), if exist, in MVC VUI parameters, mvc_vui_parameters_extension(), shall conform to HRD conformance for both MPEG-4 MVC Base view component and MPEG-4 MVC Dependent view component as MVC access unit.

A.3.2 Prohibited NAL unit

Following NAL units shall not be present.

- Sequence parameter set extension NAL unit
- Coded slice of the auxiliary coded picture without partitioning NAL unit
- Coded slice extension with slice_type = 7

A.3.3 Prohibited SEI message

Following SEI message shall not be present.

- Any SEI message NOT contained in MVC scalable nesting SEI message
- Following SEI messages contained in MVC scalable nesting SEI message
  - Stereo video information SEI message
  - Pan-scan rectangle SEI message
  - User data unregistered SEI message that contains GOP structure map
  - User data unregistered SEI message that contains Closed Caption
  - Non-required view component SEI message
  - View dependency change SEI message

A.3.4 GOP Structure

The constraints for MPEG-4 MVC Dependent view video GOP structure shall be the same as MPEG-4 AVC video streams as shown in BD-ROM spec except for the following constraints. The term “picture” used in BD-ROM spec indicates each view component for MPEG-4 MVC video streams.

- Data structure
  - The first dependent unit in a GOP in decoding order is an Anchor view component.
  - One Subset SPS (Subset Sequence Parameter Set) shall be provided in the first dependent unit of every GOP. This Subset SPS is referenced by all PPSs in a GOP and no other Subset SPS shall appear in a GOP, i.e. subsequent access units in a GOP shall have no Subset SPS. An Anchor view component which has a Subset SPS indicates the start of a GOP (in coding order).
  - A PPS (Picture Parameter Set) shall be stored in the first dependent unit or in the dependent unit that refers to this PPS in a GOP with the following restrictions.
    - Maximum number of PPSs that can be stored in a dependent unit is defined as follows.
    - There shall be at least one and at most 30 PPSs in the first dependent unit in a GOP.
    - There shall be one or zero PPSs in each dependent unit, except for the first dependent unit in a GOP.
    - There shall be no set of PPSs which share same pic_parameter_set_id within the first dependent unit of GOP.
  - Corresponding view components of Access Point P picture shall be Access Point P picture.
- GOP Structure for Base view video stream and Dependent view video stream shall be the same. Following conditions between Base view video stream and Dependent video stream shall be the same.
  - Open/Closed GOP structure.

- number of view components
- In case Dependent view component is component for B picture, the corresponding view components of Base view video shall also be view component for B picture.
- In case Dependent view component is component for non-reference B picture, the corresponding view components of Base view video shall also be view component for non-reference B picture.
- nal_ref_idc of a NAL unit with slice data for Base view component and nal_ref_idc of a NAL unit with slice data for corresponding Dependent view component shall have the same value.
- display order(POC)
- decoding delay (See BD-ROM spec)

Closed GOP:
The first view component in decoding order is an Anchor view component associated with an IDR base view component. Because an Anchor view component associated with an IDR base view component prohibits view component referencing over GOP boundary, all view component in a closed-GOP can be correctly decoded even when random access to this GOP is executed.

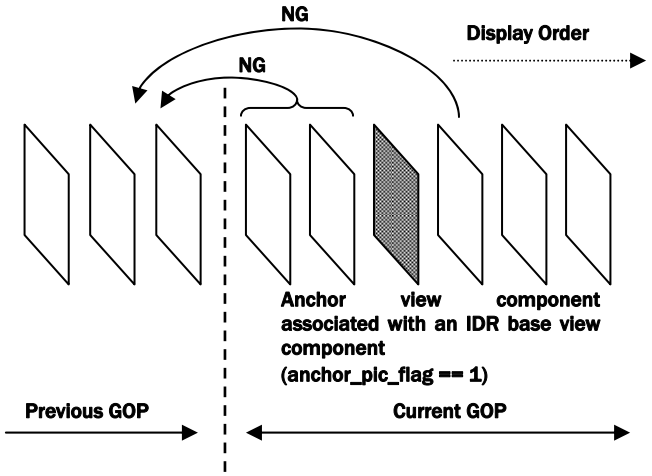


Figure A- 1 – Example of Closed GOP for MVC Dependent view video stream

Open GOP:
The first view component in decoding order is an Anchor view component associated with a non-IDR I base view component. Because an Anchor view component associated with a non-IDR I base view component does not prohibit view component referencing over GOP boundary, view components prior to the I view component in display order cannot be correctly decoded when random access to this GOP is executed.

In order to correctly decode view components subsequent to the first Anchor view component associated with a non-IDR I base view component in display order, the following conditions shall be satisfied:

- Pictures prior to the first Anchor view component associated with a non-IDR I base view component in display order may use reference to past, future and Corresponding view component. It is assumed that these view components are not displayed in case of random access to an Open GOP.
- Pictures subsequent to the first Anchor view component associated with a non-IDR I base view component in display order may use past and future references, but these view components shall not use past reference to view components prior to the first Anchor view components in display order.

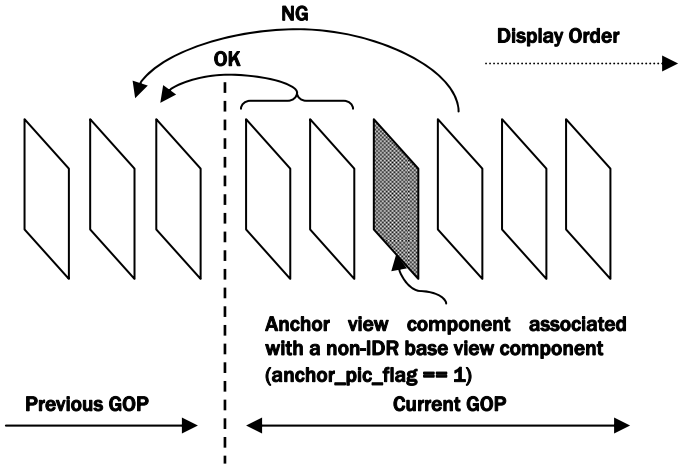


Figure A- 2 – Example of Open GOP for MVC Dependent view video stream

A.3.5 NAL units in Dependent Unit

A dependent unit in HDMV MPEG-4 MVC Dependent video streams shall be composed of multiple of segments, each of which is carried by following NAL (Network Abstraction Layer) units. An access unit (AU) in HDMV MPEG-4 MVC Base view video streams shall be composed as defined in BD-ROM spec.

- First dependent unit in a GOP of MPEG-4 MVC Dependent view video stream shall have following NAL units in listed order(Note1).
  - MVC Dependency representation delimiter NAL unit (nal_unit_type= 0x18)
  - A Subset SPS NAL unit
  - PPS NAL unit(s)
  - SEI NAL units(s) – if exist
  - Following NAL unit is repeated by number of slices
    - Coded slice extension NAL unit(s)
      - Coded slice of Anchor picture
  - A Filler data NAL unit(Note1) – if exist
  - An End of sequence NAL unit(Note2) – if exist
  - An End of stream NAL unit(Note3) – if exist
- Subsequent dependent units in a GOP of MPEG-4 MVC Dependent view video stream shall have following NAL units in listed order(Note1).
  - MVC Dependency representation delimiter NAL unit (nal_unit_type= 0x18)
  - PPS NAL unit(s) – if exist
  - SEI NAL units(s) – if exist
  - Following NAL unit is repeated by number of slices
    - Coded slice extension NAL unit(s)
      - Coded slice of an Anchor picture or a non-Anchor picture
  - A Filler data NAL unit(Note1) – if exist
  - An End of sequence NAL unit(Note3) – if exist
  - An End of stream NAL unit(Note3) – if exist
- SEI messages in a dependent unit shall be stored in the following order.
  - MVC scalable nesting SEI message that contains Buffering period SEI message – if exist
  - MVC scalable nesting SEI message that contains User data unregistered SEI message with Graphics Plane Offset metadata (defined in Section A.3.6 for details) – in case of the first dependent unit in each GOP
  - Other MVC scalable nesting SEI messages – if exist

Note1: Filler data NAL unit can be placed in any position unless it precedes the first slice NAL unit.
Note2: When an End of sequence NAL unit exists in MPEG-4 MVC Base view component, an End of sequence NAL unit shall exist in MPEG-4 MVC Dependent view component in a same access unit.
Note3: When an End of stream NAL unit exists in MPEG-4 MVC Base view component, an End of stream NAL unit shall exist in MPEG-4 MVC Dependent view component in a same access unit.

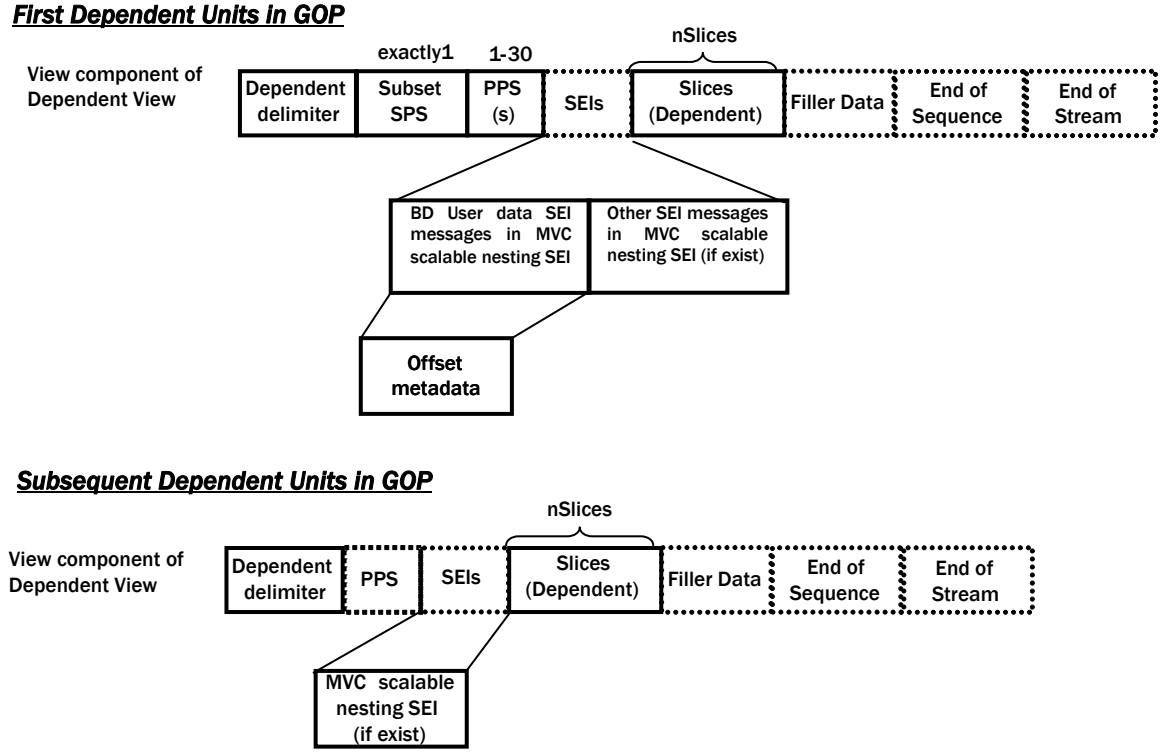


Figure A- 3 – NAL units in access unit for Dependent view video streams

A.3.6 Offset metadata

Offset metadata is a list of the offset sequence for Presentation Graphics plane, Interactive Graphics plane and BD-J Graphics plane. This offset metadata is used to set offset value for the planes when the PG/Text ST and/or IG/BD-J is presented during Stereoscopic Output Mode.

The *offset_metadata()* shall be stored in the user data container (defined in BD-ROM spec) which is contained in MVC scalable nesting SEI message of the first view component in decoding order of each GOP of MPEG-4 MVC Dependent view video stream.

NAL unit which contains a MVC scalable nesting SEI message for user data container of offset_metadata() shall not contain the other data.

Syntax of the user data container is defined in BD-ROM spec.

A.3.6.1 offset_metadata -Syntax

Syntax	No. of bits	Mnemonic
offset_metadata() {		
marker_bit	1	bslbf
reserved_for_future_use	3	bslbf
frame_rate	4	uimsbf
reserved_for_future_use	5	uimsbf
PTS[32..30]	3	uimsbf
marker_bit	1	bslbf
PTS[29..15]	15	uimsbf
marker_bit	1	bslbf
PTS[14..0]	15	uimsbf
marker_bit	1	bslbf
reserved_for_future_use	1	
number_of_offset_sequences	6	uimsbf
number_of_displayed_frames_in_GOP	8	uimsbf
marker_bit	1	bslbf
reserved_for_future_use	15	bslbf

for(offset_sequence_id = 0;
offset_sequence_id < number_of_offset_sequences;
offset_sequence_id++) {		

offset_sequence() {		
for(i=0; i < number_of_displayed_frames_in_GOP; i++){		
//presentation order		
Plane_offset_direction_flag*[i]*	1	bslbf
Plane_offset_value*[i]*	7	uimsbf
}		
}		
}		
}		

A.3.6.2 offset_metadata -Semantics

marker_bit: marker_bit shall be set to '1'.

frame_rate: This field indicates frame-rate of the video stream which includes this offset_metadata(). The value shall be set to a value according to the frame_rate defined in BD-ROM spec.

PTS: This field indicates a PTS value of the first video frame in presentation order of this GOP in 90kHz clock.

number_of_offset_sequences: This field indicates number of offset_sequence(). The maximum value of *number_of_offset_sequences* is 32. The ***offset_sequence_id*** values are defined by the order described

in the for-loop of *offset_sequence_id*, starting from zero. Thus, *offset_sequence_id* value starts at 0x00 and run through to 0x1F.

number_of_displayed_frames_in_GOP: This field indicates the number of frames to be displayed in a GOP in which this offset_metada() is contained.

Plane_offset_direction_flag*[i]*: This flag indicates the direction of horizontally shifting the associated Graphics Plane when the Graphics Plane is overlaid on the i-th video frame in presentation order of this GOP. See BD-ROM spec for the meaning of this flag of the Plane offset.

Plane_offset_value*[i]*: This value indicates amount of the pixels for horizontally shifting the associated Graphics Plane when the Graphics Plane is overlaid on the i-th video frame in presentation order of this GOP. See BD-ROM spec for the meaning of the value of the Plane offset.

If the Plane_offset_value is zero, the associated Plane_offset_direction_flag shall be set to 1_b.

A.3.7 Still picture

Still picture for MPEG-4 MVC Base view video and Dependent view video shall be compliant with constraints in BD-ROM spec. Note that term "picture" used in BD-ROM spec indicates each view component for MPEG-4 MVC video streams..

Additional constraints for Dependent view still picture are as follows:

- Dependent unit of MVC Dependent view video for AVC still picture shall have following NAL units in listed order.
  - MVC Dependency representation delimiter (nal_unit_type = 0x18)
  - Subset SPS NAL unit
  - PPS NAL unit
  - SEI NAL units(s) - if exist
  - Coded slice extension of an Anchor picture NAL unit(s)
  - A Filler data NAL unit(Note1) – if exist
  - An End of sequence NAL unit
  - An End of stream NAL unit(Note2) – if exist

Note1: Filler data NAL unit can be placed in any position unless it precedes the first slice NAL unit.
Note2: When an End of stream NAL unit exists in MPEG-4 MVC Base view component, an End of stream NAL unit shall exist in MPEG-4 MVC Dependent view component in a same access unit.

A.4 Other constraints

Following constraints are applied for all HDMV MPEG-4 MVC video streams.

- Maximum video bit-rate (MaxBR)
  - The Maximum video bit-rate (MaxBR) for MPEG-4 MVC Dependent view video stream : 40000 [1000 bits/s]
  - Total Maximum video bit-rate of MPEG-4 MVC Base view video stream and MPEG-4 MVC Dependent view video stream: 60000 [1000 bits/s]
  - In case of 1920x1080 video format with frame-rate of 23.976, *frame_mbs_only_flag*=1 and *entropy_coding_mode_flag*=1, when sum of maximum video bit-rate of MPEG-4 MVC Base view video stream and MPEG-4 MVC Dependent view video stream is greater than 48000 [1000 bits/s], both MPEG-4 MVC Base view components and MPEG-4 MVC Dependent view components shall have following conditions;
    - deblocking_filter_control_present_flag shall be set to 1
    - disable_deblocking_filter_idc shall be set to 1
- Maximum number of total frames for MPEG-4 MVC video stream which are stored in DPB shall be 6 in case of frame_mbs_only_flag is set to 1. Maximum number of frames of MPEG-4 MVC Dependent view video in DPB shall be 4.
- Each view component shall be encoded as multi-slice view component with 6 or more slices per view component. It is recommended that number of macroblock rows in every slice in the view component are as equal as possible.

- In case of 1920x1080 video formats with *frame_mbs_only_flag*=1, it is recommended that the number of macroblock rows in each slice be 10, 11, 12 or 13 macroblock rows (e.g. 11/12/11/12/11/11 configuration) or some similar configuration.
- In case of 1280x720 video formats, it is recommended that the number of macroblock rows in each slice be 6, 7, 8 or 9 macroblock rows (e.g. 7/8/7/8/7/8 configuration) or some similar configuration.

- If corresponding Base view is composed of GOP with length greater than 1 second, the maximum bit-rate of an MPEG-4 MVC Dependent view video shall be less than or equal to $15*10^6$ bits/second.
- If *frame_mbs_only_flag* is set to 1 for MPEG-4 MVC Base view video stream and MPEG-4 MVC Dependent view video stream, the number of video frames displayed in a GOP of the HDMV MPEG-4 MVC Base view video stream and MPEG-4 MVC Dependent view video stream shall be less than or equal to the maximum number that varies depending on the *maximum bit-rate* of the video stream as defined in Table A- 2 and Table A- 3.

Here, the terminology "*maximum bit-rate"* means the maximum of the input rate to EB for each view of MPEG-4 MVC video stream. It shall be the largest of the maximum value among sequences that consist of a Clip AV stream.

Table A- 2 – Maximum number of video frames displayed in a GOP for MPEG-4 MVC Base view video stream

Video format	frame_rate_ value [Hz]	frame_ mbs_ only_flag	Rx1 values in BD-ROM spec [bits/second]	Rbx1 values in BD-ROM spec [bits/second]	Maximum number of frames displayed in a GOP
1920x1080 video formats	23.976	1	$1.2*40*10^6$ for level 4.1, $1.2*24*10^6$ for level 4	$40*10^6$ for level 4.1, $24*10^6$ for level 4	24
1280x720 video formats	59.94	1	$1.2*40*10^6$ for level 4.1, $1.2*24*10^6$ for level 4	$40*10^6$ for level 4.1, $24*10^6$ for level 4	60
1280x720 video formats	50	1	$1.2*40*10^6$ for level 4.1, $1.2*24*10^6$ for level 4	$40*10^6$ for level 4.1, $24*10^6$ for level 4	50
1920x1080 video formats	23.976	1	$1.2*15*10^6$	$15*10^6$	48
1280x720 video formats	59.94	1	$1.2*15*10^6$	$15*10^6$	120
1280x720 video formats	50	1	$1.2*15*10^6$	$15*10^6$	100

Table A- 3 – Maximum number of video frames displayed in a GOP for MPEG-4 MVC Dependent view video stream

Video format	frame_rate_ value [Hz]	frame_ mbs_ only_flag	Rx10 values in BD-ROM spec [bits/second]	Rbx10 values in BD-ROM spec [bits/second]	Maximum number of frames displayed in a GOP
1920x1080 video formats	23.976	1	$1.2*40*10^6$ for level 4.1, $1.2*24*10^6$ for level 4	$40*10^6$ for level 4.1, $24*10^6$ for level 4	24
1280x720 video formats	59.94	1	$1.2*40*10^6$ for level 4.1, $1.2*24*10^6$ for level 4	$40*10^6$ for level 4.1, $24*10^6$ for level 4	60
1280x720 video formats	50	1	$1.2*40*10^6$ for level 4.1, $1.2*24*10^6$ for level 4	$40*10^6$ for level 4.1, $24*10^6$ for level 4	50
1920x1080 video formats	23.976	1	$1.2*15*10^6$	$15*10^6$	48
1280x720 video formats	59.94	1	$1.2*15*10^6$	$15*10^6$	120
1280x720 video formats	50	1	$1.2*15*10^6$	$15*10^6$	100

A.5 PES packet of the HDMV MPEG-4 MVC video stream

This Section describes coding constraints on the PES packet of the HDMV MPEG-4 MVC video stream.

- The first byte of a PES packet payload shall be the first byte of view component for Base view video stream.
- The first byte of a PES packet payload shall be the first byte of dependent unit for Dependent view video stream.
- If the coded frame is frame structure, one PES packet shall contain only one MVC view component. If the coded frame is a pair of field structure, one PES packet shall contain the pair of MVC view components for field pictures.
- PTS value of PES packet for MVC view-component shall indicate the same time as PTS value of PES packet for corresponding MVC view-component which composes one coded picture.
- Constraints on the semantics of the fields in PES packet are as follows:
  stream_id: This field shall be set to $1110\ 0000_b$
  (ISO/IEC 13818-10 MPEG-4 AVC video stream).
  PTS_DTS_flag: PTS_DTS_flag shall be equal to 11_b if the PES packet payload contains an I picture, a P picture or a B picture which PTS and DTS differs. PTS_DTS_flag shall be set equal to 10_b if PES packet payload contains a B picture which PTS and DTS are the same.
- If a PES packet contains a dependent unit that starts with a Subset Sequence parameter set NAL unit (GOP start), both the first byte of the PES packet and the first byte of the MVC Dependency representation delimiter NALunit shall be multiplexed in the same transport packet.

A.6 MPEG-2 TS multiplexing of the MPEG-4 MVC video stream

- When transport packets carry a byte stream NAL unit containing offset_metadata(), the transport packets shall not contain any data other than the byte stream NAL unit.
- The *transport_priority* field of transport packets which carry a byte stream NAL unit containing offset_metadata() shall be set to 1_b.
- The *transport_priority* field of transport packets which don't carry a byte stream NAL unit containing offset_metadata() shall be set to 0_b

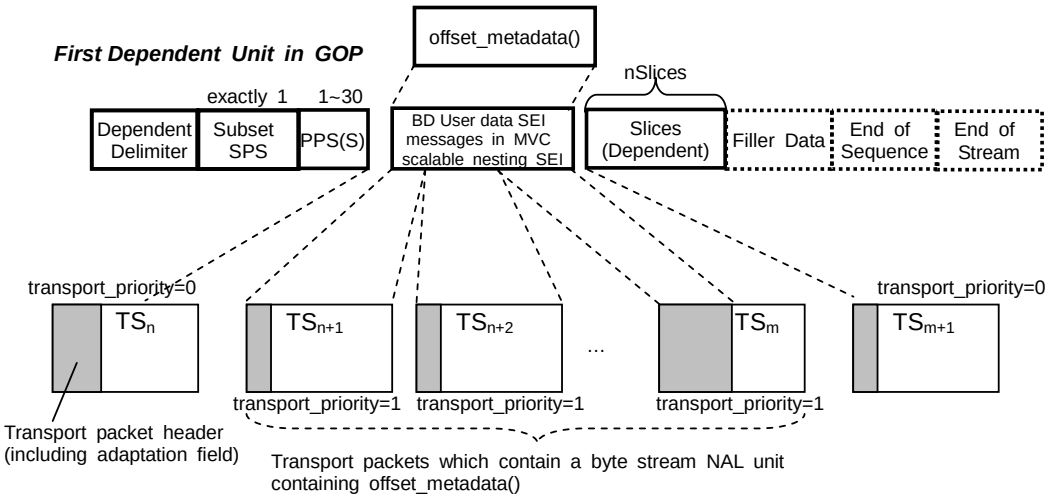


Figure A- 4 – MPEG-2 TS multiplexing of MVC Dependent view stream which includes offset metadata

- If application_type is set to 1, 2 or 3 and Clip AV stream file contains both MPEG-4 MVC Base view video stream and MPEG-4 MVC Dependent view video stream, the following conditions shall be applied;
  - ☐ The first transport packet of PES packet header with MPEG-4 MVC Base view component for the first Access Unit in a GOP (GOP-1) shall precede the first transport packet of PES packet header with corresponding MPEG-4 MVC Dependent view component for the first Dependent Unit in the GOP (GOP-1).
  - ☐ The last transport packet of PES packet with MPEG-4 MVC Dependent view component for the last Dependent Unit in the GOP (GOP-1) shall precede the first transport packet of PES packet header with MPEG-4 MVC Base view component for the first Access Unit of the following GOP (GOP-2).

  Note that PID for MPEG-4 MVC Base view video stream and MPEG-4 MVC Dependent view video streams are different. PID among MPEG-4 MVC Dependent view video streams are also different.

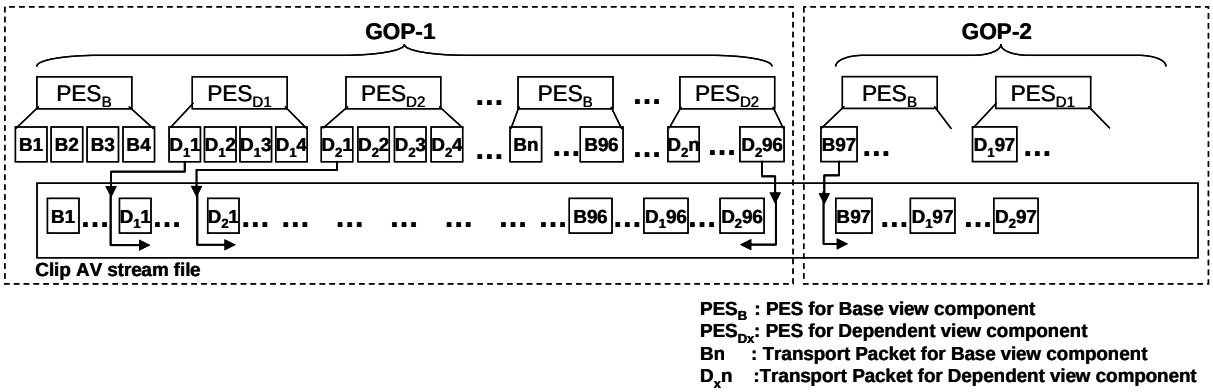


Figure A- 5 – MPEG-2 TS multiplexing of MPEG-4 MVC video stream within same Clip AV stream file