

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
13 July 2006 (13.07.2006)

PCT

(10) International Publication Number  
WO 2006/073260 A2

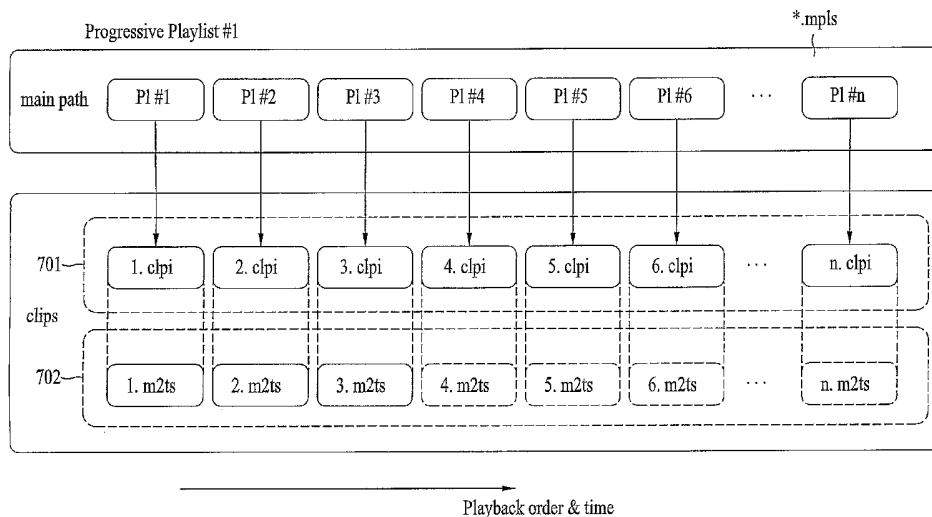
- (51) International Patent Classification:  
*G11B 20/10* (2006.01)     *G11B 27/00* (2006.01)
- (21) International Application Number:  
PCT/KR2006/000020
- (22) International Filing Date: 3 January 2006 (03.01.2006)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:  
60/641,780     7 January 2005 (07.01.2005)     US  
10-2005-0096450     13 October 2005 (13.10.2005)     KR
- (71) Applicant (for all designated States except US): **LG ELECTRONICS INC.** [KR/KR]; 20, Yoido-dong, Youngdungpo-gu, Seoul 150-721 (KR).
- (72) Inventors; and
- (75) Inventors/Applicants (for US only): **SEO, Kang Soo** [KR/KR]; 104-1504, Chowon APT., #898, Pyeongchon-dong, Dongan-gu, Anyang-si, Gyeonggi-do 431-070 (KR). **YOO, Jea Yong** [KR/KR]; 205-808, Ssangyong APT., Garak 2-dong, Songpa-gu, Seoul 138-747 (KR). **KIM, Byung Jin** [KR/KR]; 111-204, Chonggu APT., Hansolmaeul, #110, Jeongja-dong, Bundang-gu, Seongnam-si, Gyeonggi-do 463-914 (KR).

- (74) Agents: **BAHNG, Hae Cheol** et al.; 15th Floor, Yosam Building, 648-23 Yeoksam-dong, Kangnam-gu, Seoul 135-080 (KR).
- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.
- (84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Published:**  
— without international search report and to be republished upon receipt of that report

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR REPRODUCING DATA FROM RECORDING MEDIUM USING LOCAL STORAGE



(57) Abstract: A method and apparatus for reproducing data from a recording medium using a local storage is disclosed. A method for reproducing data from a recording medium using a local storage includes the steps of: a) downloading data associated with the recording medium, and storing the downloaded data in the local storage; b) generating a virtual package using a binding unit with missing data associated with the recording medium, such that the virtual package includes a PlayList capable of being played before all the data is completely; and c) playing the PlayList while the missing data is downloaded. Therefore, the method can effectively reproduce additional data, which is downloaded in the local storage in association with the recording medium, resulting in the creation of more convenient functions for a user.

WO 2006/073260 A2



---

*For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

**[DESCRIPTION]****METHOD AND APPARATUS FOR REPRODUCING DATA FROM RECORDING  
MEDIUM USING LOCAL STORAGE****5   Technical Field**

The present invention relates to a method and apparatus for reproducing data from a recording medium, and more particularly to a method and apparatus for reproducing data from a recording medium using a local storage contained in an optical recording/reproducing device.

**Background Art**

Generally, there has been widely used an optical disc acting as a recording medium capable of recording a large amount of data therein. Particularly, there has recently been developed a high-density optical recording medium capable of recording/storing high-quality video data and high-quality audio data for a long period of time, for example, a Blu-ray Disc (BD).

The BD based on the next-generation recording medium technique has been considered to be the next-generation optical recording solution capable of storing much more data than a conventional DVD. In recent times, many developers have conducted intensive research into the international standard technical specification associated with the BD along

with those of other digital devices.

In association with the above-mentioned situation, there has recently been developed an optical recording/reproducing device based on the BD international standard, but the BD  
5 international standard has not yet been completed, such that many limitations and problems occur in developing the optical recording/reproducing device.

Particularly, the above-mentioned optical recording/reproducing device must consider not only a basic  
10 function for recording/reproducing data of the BD, but also an additional function for enabling the optical recording/reproducing device to interact with peripheral digital devices. In other words, the optical recording/reproducing device must receive an external input  
15 signal, must display the received signal, and must reproduce desired data using the external input signal and the BD.

However, an apparatus for reproducing data from the recording medium to simultaneously reproduce the external input signal and data of the BD has not yet been established, such that  
20 many limitations and problems occur in developing a BD-based optical recording/reproducing device.

### **Disclosure of Invention**

Accordingly, the present invention is directed to a method  
25 and apparatus for reproducing data from a recording medium

using a local storage that substantially obviate one or more problems due to limitations and disadvantages of the related art.

An object of the present invention is to provide a method and  
5 apparatus for downloading data associated with a recording medium from an external part, storing the downloaded data in a local storage, and reproducing the downloaded data using the recording medium associated with the local storage.

Additional advantages, objects, and features of the invention  
10 will be set forth in part in the description which follows and in part will become apparent to those having ordinary skill in the art upon examination of the following or may be learned from practice of the invention. The objectives and other advantages of the invention may be realized and  
15 attained by the structure particularly pointed out in the written description and claims hereof as well as the appended drawings.

To achieve these objects and other advantages and in accordance with the purpose of the invention, as embodied and  
20 broadly described herein, a method for reproducing data from a recording medium using a local storage comprises the steps of: a) downloading data associated with the recording medium, and storing the downloaded data in the local storage; b) generating a virtual package using a binding unit with  
25 missing data associated with the recording medium, such that

the virtual package includes a PlayList capable of being played before all the data is completely downloaded, and c) playing the PlayList while the missing data is downloaded.

The missing data can be indicative of stream data.

5 The PlayList can reproduce data downloaded in the local storage.

The virtual package can be generated by a binding unit manifest file associated with the recording medium.

In another aspect of the present invention, there is provided  
10 a method for reproducing data from a recording medium using a local storage comprising the steps of: a) generating a virtual package for reproducing data in the recording medium and data in the local storage; b) downloading data associated with the recording medium, and updating the virtual package  
15 using a binding unit with missing data, such that the virtual packaged includes a PlayList capable of being played before all the data is completely downloaded, and c) playing the PlayList while the missing data is downloaded.

The PlayList can a main path and at least one sub path, and  
20 data referenced by the main path and data referenced by the sub path can be simultaneously downloaded.

The sub path can be configured with a single SubPlayItem, and downloaded part of a clip referenced by the SubPlayItem can be reproduced while the missing part of the clip is  
25 downloaded.

The sub path can be configured with a plurality of SubPlayItems, and downloaded clips referenced by the SubPlayItems can be reproduced while the missing clips can be downloaded. Also downloaded part of a clip referenced by one  
5 of the SubPlayItems can be reproduced while the missing part of the clip is downloaded.

The PlayList can be played by a BD-J module.

The virtual package can be updated by a request of BD-J applications

10 The method can further comprise the step of: in case that playback position by the PlayList reaches the missing data point during playback, stopping the playback.

The method can further comprise the step of: in case that an error occurs while the PlayList is played, the error is  
15 corrected by execution of BD-j applications.

The PlayList can be indicative of a Progressive PlayList. Data referenced by the Progressive PlayList can be indicated by a binding unit manifest file associated with the recording medium. The step b) for updating the virtual package can  
20 include the step of: confirming whether the binding unit manifest file contains Progressive PlayList information.

In another aspect of the present invention, there is provided an apparatus for reproducing data from a recording medium using a local storage comprising: a pickup unit for reading  
25 data of the recording medium; a local storage for downloading

data associated with the recording medium, and storing the downloaded data; and a controller for generating a virtual package using a binding unit with missing data associated with the recording medium, such that the virtual package  
5 includes a PlayList capable of being played before all the data is completely downloaded, the controller for playing the PlayList while the missing data is downloaded.

In case that the virtual package exists, the controller can update the virtual package to generate a new virtual package,  
10 such that the new virtual package includes the PlayList capable of being played before all the data is completely downloaded.

The missing data can be indicative of stream data.

The PlayList can reproduce data downloaded in the local  
15 storage.

The controller can generate the virtual package by a binding unit manifest file associated with the recording medium.

The PlayList can include a main path and at least one sub path. In this case, the controller can download data  
20 referenced by the main path and data referenced by the sub path simultaneously.

The controller can reproduce downloaded part of a clip referenced by the PlayList while the missing part of the clip is downloaded. The controller can reproduce downloaded clips  
25 by the PlayList while the missing clips are downloaded. Also,

the controller can reproduce downloaded part of a clip referenced by the PlayList while the missing part of the clip is downloaded.

The controller includes a BD-j module, and the BD-J module  
5 can play the PlayList.

The PlayList can be indicative of a Progressive PlayList. Data referenced by the Progressive PlayList can be indicated by a binding unit manifest file associated with the recording medium. In generating the virtual package, the controller can  
10 confirm whether the binding unit manifest file contains Progressive PlayList information.

It is to be understood that both the foregoing general description and the following detailed description of the present invention are exemplary and explanatory and are  
15 intended to provide further explanation of the invention as claimed.

#### **Brief Description of Drawings**

The accompanying drawings, which are included to provide a  
20 further understanding of the invention and are incorporated in and constitute a part of this application, illustrate embodiment(s) of the invention and together with the description serve to explain the principle of the invention.

In the drawings:

25 FIG. 1 is a conceptual diagram illustrating a method and

apparatus for reproducing data from a recording medium according to the present invention;

FIG. 2 is a conceptual diagram illustrating a file structure recorded in an optical disc acting as a recording medium and  
5 a method for reproducing a specific title using the file structure;

FIG. 3 is a structural diagram illustrating a data record structure of an optical disc acting as a recording medium according to the present invention;

10 FIG. 4A is a block diagram illustrating an optical recording/reproducing device according to the present invention;

FIG. 4B is a block diagram illustrating an apparatus for reproducing data using a local storage from among overall  
15 components contained in the optical recording/reproducing device according to the present invention;

FIG. 5 is a conceptual diagram illustrating a method for generating a virtual package according to the present invention;

20 FIG. 6 is a conceptual diagram illustrating a relationship for generating a virtual package according to the present invention;

FIG. 7 is a structural diagram illustrating a PlayList and an example for playing the PlayList according to the present  
25 invention;

FIG. 8 is a structural diagram illustrating an apparatus for reproducing data of a recording medium using a playback system according to the present invention;

FIG. 9 shows a PlayList including a sub path configured with a single SubPlayItem (SPI) and an example for playing the PlayList according to the present invention; and

FIG. 10 shows a PlayList including a sub path configured with a plurality of SubPlayItems (SPIs) and an example for reproducing the PlayList according to the present invention.

10

#### **Best Mode for Carrying Out the Invention**

Reference will now be made in detail to the preferred embodiments of the present invention, examples of which are illustrated in the accompanying drawings. Wherever possible, the same reference numbers will be used throughout the drawings to refer to the same or like parts.

A method and apparatus for reproducing data from a recording medium using a local storage will hereinafter be described with reference to the annexed drawings.

Prior to describing the present invention, it should be noted that most terms disclosed in the present invention correspond to general terms well known in the art, but some terms have been selected by the applicant as necessary and will hereinafter be disclosed in the following description of the present invention. Therefore, it is preferable that the

terms defined by the applicant be understood on the basis of their meanings in the present invention.

A recording medium for use in the present invention is indicative of all recordable mediums, for example, an optical disc, and a magnetic tape, etc., according to various recording schemes. It should be noted that the term "Disc" is considered to be the same as the recording medium according to the present invention.

For the convenience of description and better understanding of the present invention, the optical disc, such as a BD, will hereinafter be exemplarily used as the above-mentioned recording medium in the present invention. It should be noted that technical ideas of the present invention can be applied to other recording mediums without departing from the scope and spirit of the invention.

The term "local storage" is indicative of a storage unit contained in an optical recording/reproducing device 10 shown in FIG. 1. In more detail, the term "local storage" is indicative of a component capable of receiving necessary information or data from a user, and storing the received information or data. For example, a general local storage may be indicative of a Hard Disc Drive (HDD), but it should be noted that the term "local storage" of the present invention is not limited to the HDD, and is applicable to other examples as necessary.

Particularly, the term "local storage" is indicative of a storage unit for storing data associated with a recording medium such as a BD. The data associated with the recording medium is generally downloaded from an external source.

5 In association with the above-mentioned description, it is obvious to those skilled in the art that the local storage may directly read some permission data from the recording medium, and may generate system data (e.g., metadata) associated with record/reproduction operations of the  
10 recording medium, such that the system data may be stored in the local storage.

The term "binding unit" is indicative of a set of files stored in the local storage. Particularly, the binding unit is indicative of a set of information (i.e., information set)  
15 associated with a specific recording medium. In this case, the information set is combined with or is added to a file contained in the recording medium, such that it can simultaneously reproduce data of the recording medium and data of the local storage.

20 For the convenience of description, data recorded in the recording medium is referred to as "original data", data associated with the recording medium from among a plurality of data units stored in the local storage is referred to as "additional data".

25 The term "Title" is indicative of a reproduction unit

interfacing with a user, and each title is linked to a specific object file, such that a stream associated with a corresponding title recorded in a disc is reproduced according to a command or program stored in the object file.

5 Particularly, for the convenience of description and better understanding of the present invention, a first title in which movies and interactive information according to an MPEG2 compression scheme are recorded is called an HDMV title, and a second title in which movies and interactive  
10 information according to a Java program are recorded is called a BD-J title.

FIG. 1 is a conceptual diagram illustrating a method and apparatus for reproducing data according to the present invention. Unified usages of the optical  
15 recording/reproducing 10 and peripheral devices are shown in FIG. 1.

The optical recording/reproducing device 10 can record/reproduce data in/from various optical discs having different formats. If necessary, the optical  
20 recording/reproducing device 10 can record/reproduce specific data in/from only a specific optical disc such as a BD, or can reproduce the data from the optical disc without recording the data in the same. It should be noted that the present invention exemplarily uses a BD-player capable of  
25 reproducing data from the BD or a BD-recorder capable of

recording data in the BD in consideration of correlation between the BD and peripheral devices for the convenience of description. In association with the above-mentioned description, it is well known in the art that the optical recording/reproduction device 10 is also applicable to a drive embedded in a specific device such as a computer.

The optical recording/reproducing device 10 records or reproduces data in/from the optical disc 30, receives an external input signal, performs a signal process on the received signal, and transmits the signal processed result to the external display 20, such that a user can view the signal processed result on the display 20. In this case, there is no limitation in a receivable external signal. For example, representative external input signals may be determined to be a DTV-associated signal and an Internet-associated signal, etc. Specifically, the Internet is indicative of a communication network to which a user easily gains access, such that the user can download specific Internet data using the optical recording/reproducing device 10, and can use the downloaded data.

In association with the above-mentioned description, a person for providing content data used as an external source is generally referred to as a content provider (CP).

The term "Content" is indicative of content data of the title, and is equal to data received from an author of a recording

medium.

When original data is recorded in the optical disc 30 seated in the optical recording/reproducing device 10, and additional data associated with the original data is present  
5 in other storage places (e.g., Internet), the present invention aims to reproduce the additional data using the local storage.

For example, it is assumed that multiplexed AV (Audio/Video) streams are recorded as the original data recorded in the  
10 optical disc, and additional data for use in the Internet is an audio stream different from an audio stream (e.g., Korean) of the original data. In this case, some users may download specific audio stream (e.g., English) acting as additional data from the Internet, may desire to reproduce the  
15 downloaded audio stream along with the AV stream acting as original data, or may desire to reproduce only the additional data. In order to implement the above-mentioned desires of the users, correlation between the original data and the additional data must be established, and there is needed a  
20 systemized method for managing/reproducing the above-mentioned data according to a user request.

For the convenience of description, although a signal recorded in the disc is referred to as the original data, and other signals existing in the outside of the disc are  
25 referred to as the additional data, it should be noted that

the original data and the additional data classified according to their data acquisition methods are not limited to only specific data.

Generally, additional data may be indicative of audio data, presentation graphic (PG) data, interactive graphic (IG) data, or text subtitle, etc., but the additional data may also be indicative of a multiplexed AV stream including the above-mentioned data and video data. In other words, data associated with the original data simultaneously existing in the outside of the optical disc may act as additional data.

In order to satisfy the above-mentioned user requests, a predetermined file structure must be established between the original data and the additional data. Accordingly, a file structure and data record structure for use in the BD will hereinafter be described with reference to FIGS. 2~3.

FIG. 2 is a conceptual diagram illustrating a file structure for reproducing/managing the original data recorded in an optical disc, and a method for reproducing a specific title according to the file structure.

Referring to FIG. 2, the file structure according to the present invention includes one or more BDMV directories (BDMV) under a single root directory. The BDMV directory includes not only an index file "index" acting as a general file (i.e., an upper file) capable of guaranteeing user interactivity, but also the above-mentioned object. The file

structure includes a variety of directories for storing information of actual data recorded in a disc and other information associated with a method for reproducing the data, for example, a PlayList directory (PLAYLIST), a clip information directory (CLIPINF), a stream directory (STREAM), an auxiliary directory (AUXDATA), a BD-J directory (BDJO), a metadata directory (META), and a backup directory (BACKUP). The above-mentioned directories and a variety of files included in the directory will hereinafter be described.

10 The META directory includes a metadata file acting as "Data about Data". In other words, the META directory includes a search file and a metadata file for Disc Library and the like. The BDJO directory includes a BD-J object file for reproducing a BD-J title.

15 The AUXDATA directory includes an additional data file for reproducing data of the disc. For example, the AUXDATA directory includes a "Sound.bdmv" file for providing a user with sound data when an interactive graphic function is executed, and an "11111.otf" file for providing the user with font information when data of the disc is reproduced.

The STREAM directory includes a plurality of AV stream files recorded in a disc according to a specific format. Generally, individual streams are recorded using an MPEG-2 based transport packet, and the STREAM directory uses extension names of stream files (01000.m2ts and 02000.m2ts) as a

25

specific extension name "\*.m2ts". Particularly, if video/audio/graphic information from among the above-mentioned streams is multiplexed, the multiplexed information is called an AV stream, and a single title is composed of at least one AV stream file.

The CLIPNF directory is composed of a plurality of clip information files (01000.clpi and 02000.clpi) connected to the above-mentioned stream files on a one-to-one basis. Particularly, the clip information files (\*.clpi) record attribute information and timing information of the stream files (\*.m2ts) therein. The clip information files (\*.clpi) connected to the stream files (\*.m2ts) on a one-to-one basis are generically named a "Clip". In other words, this means that a single clip is indicative of data composed of a stream file (\*.m2ts) and a clip information file (\*.clpi).

The PLAYLIST directory includes a plurality of PlayList files (\*.mpls). Each PlayList file (\*.mpls) includes one or more PlayItems and one or more SubPlayItems. Each PlayItem and each SubPlayItem are adapted to designate a playing interval during which a specific clip is reproduced. The PlayItem (PI) and SubPlayItem (SPI) include information associated with a specific clip to be reproduced, i.e., information associated with a reproduction start time (IN-Time) and other information associated with a reproduction termination time (OUT-Time) of the specific clip.

In association with the above-mentioned description, a process for reproducing data using at least one PlayItem (PI) in the PlayList file is referred to as a main path, and a process for reproducing data using individual SubPlayItems (SPIs) is referred to as a sub path. The PlayList file must contain a single main path. The PlayList file may contain at least one sub path according to the presence or absence of the SubPlayItem (SPI) as necessary.

In conclusion, the PlayList file acts as a basic reproduction/management file unit contained in overall reproduction/management file structures for reproducing a desired clip by combination of one or more PlayItems.

The BACKUP directory stores a plurality of duplicate files, i.e., a duplicate file (also called "copied files") of the index file "index" storing information associated with disc reproduction, a duplicate file of the object file "MovieObject", duplicate files of all PlayList files (\*.mpls) contained in the PLAYLIST directory, and duplicate files of all clip information files (\*.clpi) contained in the CLPINF directory. If the above-mentioned files ("index", "MovieObject", "\*.mpls", and "\*.clpi") are damaged, a disc reproduction process is also fatally damaged, such that the BACKUP directory is designed to pre-store duplicate files of the above-mentioned files as backup files.

In association with the above-mentioned description, a method

for reproducing a specific title using the above-mentioned disc file structure is shown in FIG. 2.

If a user enters a title reproduction command in association with a title used as an index file (also called "index table"), reproduction of the title begins. A detailed description thereof will hereinafter be described.

The index file (index.bdmv) includes first playback information "First Playback" indicative of information associated with a first reproduction image when data of a corresponding disc is loaded, top menu information "Top Menu" for providing a menu image, and at least one title information "Title #1 ~Title #n".

If the optical disc 30 is loaded in the optical recording/reproducing device 10, title menu information associated with the index table is provided to the user via the display 20. If the user selects a specific title or a specific menu contained in a menu image, data reproduction begins according to a scenario pre-defined by a disc manufacturer. In other words, if the user enters a command for reproducing a specific title (e.g., title #1), a specific Playlist file is executed according to a command contained in the object file (MovieObject) of the reproduction/management file structure. Thereafter, one or more clips (e.g., Clip #1 ~ Clip #3) configuring the title #1 are reproduced by a specific PlayItem and/or SubPlayItem contained in the

PlayList file according to the PlayList file information.

FIG. 3 is a structural diagram illustrating a data record structure recorded in a recording medium according to the present invention. In more detail, FIG. 3 shows a .disc record format of information associated with the file structure.

As shown in FIG. 3, from the viewpoint of an inner area of the disc, the above-mentioned disc structure sequentially includes a file system information area serving as system information for managing overall files, a database area for recording a PlayList file and a clip information file to reproduce a recorded AV stream (\*.m2ts), and an AV stream area for recording a plurality of streams composed of audio data, video data, and graphic data, etc. Particularly, it should be noted that data recorded in the AV stream area may be determined to be original data as previously stated above.

The present invention provides a method and apparatus for simultaneously reproducing original data (e.g., a file structure shown in FIG. 2) recorded in the disc and additional data recorded in the local storage, and a variety of preferred embodiments according to the present invention will hereinafter be described.

FIG. 4A is a block diagram illustrating the optical recording/reproducing device 10 according to the present invention.

Referring to FIG. 4A, the optical recording/reproducing device 10 includes a pickup 11, a servo 14, a signal processor 13, and a microprocessor 16. The pickup 11 reads original data recorded in the optical disc and management information including reproduction/management file information. The servo 14 controls operations of the pickup 11. The signal processor 13 receives a reproduction signal from the pickup 11, restores the received reproduction signal to a desired signal value, or modulates a signal to be recorded into another signal recorded in the optical disc, such that it transmits the restored or modulated result. The microprocessor 16 controls overall operations of the above-mentioned components.

A controller 12 downloads additional data from outside of the optical disc upon receiving a command from a user, stores the downloaded additional data in the local storage 15, and configures a binding unit using binding unit manifest information recorded in a binding unit manifest file of the local storage. In addition, the controller 12 generates a virtual file structure (hereinafter referred to as a virtual package) using the binding unit manifest information recorded in the binding unit manifest files, such that it can reproduce data contained in the recording medium and data contained in the local storage. The controller reproduces the original data and/or the additional data using the

virtual package upon receiving a request from a user.

The virtual package generated by the controller 12 includes at least one virtual PlayList including a PlayItem capable of reproducing a plurality of clips.

5 In association with the above-mentioned description, the binding unit manifest file, the name mapping information, and the virtual package will hereinafter be described with reference to FIG. 5.

The AV decoder 17 finally decodes output data (i.e., original  
10 data and/or additional data) upon receiving a control signal from the controller 12, and provides the user with the decoded result.

The AV encoder 18 converts an input signal into a specific  
format signal (e.g., an MPEG2 transport stream) upon  
15 receiving a control signal from the controller 12, and transmits the converted result to the signal processor 13.

The new virtual package may be stored in the local storage 15  
such that it can be re-used in future. Also, the new virtual  
package may be temporarily stored in an additional dynamic  
20 memory, and may then be used.

FIG. 4B is a block diagram illustrating an apparatus for  
reproducing data of an optical disc using a local storage  
from among overall components contained in the optical  
recording/reproducing device 10 according to the present  
25 invention.

Information stored in the local storage 15 will hereinafter be described. The local storage 15 according to the present invention stores file information (Directory-File for disc\_ID #n dependent) including directories and files of individual disc identification (ID) information, and a plurality of additional clips downloaded from an external part.

The local storage 15 may include a binding unit manifest file for forming a binding unit of the local storage 15, and performing a binding operation associated with a disc package. Specifically, the local storage 15 may include a plurality of file information units (Directory-File for disc\_ID #n dependent) to cope with different discs. Therefore, the local storage 15 additionally requires a file system for managing the above-mentioned file information units. Specifically, the file system is also referred to as a local storage file system 41, and the local storage file system 41 is indicative of a system for managing all files stored in the local storage 15.

Therefore, if an optical disc (e.g., disc\_ID #1) of a specific disc ID (disc\_ID) is loaded in the optical recording/reproducing device 10, the controller 12 contained in the optical recording/reproducing device 10 recognizes ID information of the disc using the pickup 11 and the signal processor 13. If the local storage 15 stores file information, the controller reads binding unit file

information equal to the loaded disc ID information from among all file information units stored in the local storage 15, generates a virtual package by binding (or combining) the read information and the disc package, and reproduces both 5 the original data of the disc and the additional data of the local storage using the formed virtual package.

In association with the above-mentioned description, if no file information exists in the local storage 15, the optical recording/reproducing device downloads data from an external 10 input source to form a binding unit, reads the binding unit file information, and forms a virtual package by binding (or combining) the read information with the disc package.

A method for generating the virtual package according to the present invention will hereinafter be described with 15 reference to FIG. 5.

FIG. 5 is a conceptual diagram illustrating a method for generating the virtual package according to the present invention.

Referring to FIG. 5, if a specific disc is loaded in the 20 optical recording/reproducing device 10 on the condition that the information and contents shown in FIG. 4b are stored in the local storage 15, the optical recording/reproducing device 10 reads the file system information 41 contained in the local storage 15, and at the same time reads disc file 25 system information 42 including the file structure recorded

in the disc, resulting in the creation of the VFS. In more detail, the virtual file system (VFS) is indicative of a file system virtually constructed to manage both the file system contained in the local storage 15 and the other file system of the loaded disc.

The optical recording/reproducing device generates a new virtual package to simultaneously reproduce original data recorded in the disc and additional data recorded in the local storage using the above-mentioned virtual file system (VFS). For this purpose, the optical recording/reproducing device 10 reads file information (Directory-File for disc\_ID #1 dependent) associated with the disc (e.g., disc\_ID #1) from the file system, forms a binding unit according the read binding unit manifest information, reads the formed binding unit, and performs a binding operation for replacing the read file information with a disc package of the loaded disc (disc\_ID #1) according to either name mapping information or Progressive PlayList information recorded in the binding unit manifest file, or appending the read file information to the disc package according to either name mapping information or Progressive PlayList information recorded in the binding unit manifest file. A detailed description of the Progressive PlayList will be described with reference to the annexed drawings from FIG. 7.

25 In association with the above-mentioned description, the name

mapping information according to the present invention is indicative of information recorded in the binding unit manifest file contained in a directory and/or a file structure of a local storage, and is used when the binding  
5 unit is formed or when the binding operation is executed to form the virtual package.

"Files and Titles Relation List" information, "Progressive Playlist" information, and "Credential" information may be contained in the binding unit manifest file. The  
10 aforementioned "Files and Titles Relation List" information is adapted to delete/verify data in title units. The aforementioned "Credential" information is adapted to bind or combine files shared by different content providers with each other.

15 In association with the above-mentioned description, the virtual package formed by the aforementioned binding operation is indicative of a file structure for reproducing/managing an original clip 422 and an additional clip 412. The original clip 422 is composed of original data  
20 recorded in the disc. The additional clip 412 is composed of additional data recorded in the local storage.

FIG. 6 is a conceptual diagram illustrating a relationship for generating a virtual package according to the present invention.

25 A specific file structure (i.e., the disc package of FIG. 2)

is recorded in the loaded disc, and may be referred to as a disc package. A local storage file system 41 is contained in the local storage. The local storage file system 41 includes a binding unit combined with the loaded disc (e.g., disc\_ID 5 #1), and a binding unit manifest file.

In association with the above-mentioned description, all files referenced by the binding unit manifest file and the binding unit manifest file of the disc may act as Read-only files.

10 The binding unit manifest file includes name mapping information. The name mapping information includes a variety of information associated with the binding unit. For example, the name mapping information includes list information of files contained in the binding unit, location information 15 where the files of the binding unit are located in the virtual package when the binding unit is combined with the disc, and file name information in the virtual package, etc. Therefore, the virtual file system (VFS) performs a binding operation for combining the binding unit with the disc 20 package of the loaded disc using the above-mentioned binding unit manifest information, such that it forms a new virtual package.

For example, according to the disc package of the disc shown in FIG. 6, the BDMV directory acting as a lower directory of 25 the root directory includes an index file (index.bdmv), an

object file (MovieObject.bdmv), a PlayList file (00000.mpls),  
a clip information file (01000.clpi), a stream file  
(01000.m2ts), and an Auxiliary file (sound.bdmv). Under the  
aforementioned condition of the disc package, if the binding  
5 unit 61 associated with the loaded disc (e.g., the disc  
having information "org\_ID #1 and disc\_ID #1") includes a  
specific PlayList file (Apr2005.mpls) 611, and clips (i.e., a  
clip information file (Apr2005.clpi) 612 and a stream file  
(Apr2005.m2ts) 613) managed by the PlayList file  
10 (Apr2005.mpls) 611, and the disc-dependent-shared directory  
(Shared) includes an Auxiliary file (Japanese.otf) 614. A  
method for generating the virtual package 51 using the disc  
package and the binding unit will hereinafter be described.

The name mapping information according to the present  
15 invention is indicative of information associated with files  
contained in the binding unit. A PlayList file  
(Apr2005.mpls) 611 is changed (or modified) to a PlayList  
file (00000.mpls) 511 of the PLAYLIST directory in the  
virtual package 51, a clip information file (Apr2005.clpi)  
20 612 is changed to a clip information file (02000.clpi) 512 of  
the CLIPINF directory in the virtual package 51, a stream  
file (Apr2005.m2ts) 613 is changed to a stream file  
(02000.m2ts) 513 of the STREAM directory in the virtual  
package 51, and an auxiliary file (Japanese.otf) 614 is  
25 changed to an auxiliary file (11111.otf) 514 of the AUXDATA

directory in the virtual package 51. The name mapping information includes location- and file name- information of the aforementioned changed files.

Therefore, when the virtual file system (VFS) 40 performs the  
5 binding operation, if the binding operation is performed according to the name mapping information recorded within the binding unit manifest file in the local storage 15 and the virtual package is generated by the binding operation, the resultant file structure is as follows.

10 The virtual package 51 includes the BDMV directory acting as a lower directory of the root directory. The BDMV directory includes an index file (Index.bdmv), an object file (MovieObject.bdmv), a PLAYLIST directory, a CLIPINF  
15 directory, a STREAM directory, and an AUXDATA directory. The PLAYLIST directory includes a PlayList file (00000.mpls) 511 acquired when a PlayList file of the disc package 62 is replaced with a PlayList file of the binding unit. The CLIPINF directory includes a clip information files (02000.clpi) 512 of the binding unit which is appended to the  
20 clip information file (01000.clpi) of the disc package 62. The STREAM directory includes a stream file a stream file (02000.m2ts) 513 of the binding unit which is appended to the stream file (01000.m2ts) of the disc package 62. The AUXDATA directory includes an Auxiliary file (11111.otf) 514 of the  
25 binding unit which is appended to the Auxiliary file

(sound.bdmv) of the disc package 62. In this process, the virtual package 51 is generated.

In association with the above-mentioned description, the index file (index.bdmv) and the object file (MovieObject.bdmv), which act as upper files of the virtual package 51, may be updated in conventional index and object files (Index.bdmv and MovieObject.bdmv) contained in the disc on the basis of the newly-formed virtual PlayList file (00000.mpls) 511. Specifically, if a title is changed to another title by the PlayList file (00000.mpls) 511 contained in the virtual package 51 (e.g., if a new title is added, a conventional title is deleted, or a title reproduction scenario is changed to another scenario), the aforementioned index and object files may be updated on the basis of the newly-formed virtual package.

In association with the above-mentioned description, a method for downloading data to construct a virtual package, and a method for updating the virtual package using the downloaded data will hereinafter be described.

Firstly, the downloading application recognizes current binding unit structure information contained in the loaded disc-dependent directory and the disc-dependent-shared directory (Shared), and informs an external CP of the recognized binding unit structure information. A user selects content data downloaded for reproducing data of the

disc.

Thereafter, the CP provides all data (e.g., an index table and a PlayList, etc.) of database files appropriate for a disc on the basis of the report result received from the  
5 downloading application associated with the current binding unit structure information.

Finally, the downloading application of the optical recording/reproducing device downloads data of the selected files for the disc in the local storage, forms a virtual file  
10 system (VFS), and forms/update a virtual package using the virtual file system (VFS).

FIG. 7 is a structural diagram illustrating a PlayList and an example for reproducing the PlayList according to the present invention.

15 According to the present invention, PlayList reproduction is performed by downloading data reproduced by the PlayList.

In association with the above-mentioned description, a PlayList reproducible by the local storage can be classified into two types (i.e., a first PlayList and a second PlayList).  
20 The first PlayList can be played after all data reproduced by the PlayList has been completely downloaded. The second PlayList can be played although data is not completely downloaded, such that the remaining data can be downloaded simultaneously with reproducing pre-downloaded data. The  
25 present invention relates to the aforementioned second

PlayList. Specifically, the Progressive PlayList shown in FIG. 5 corresponds to the second PlayList. Hereinafter, the remaining data is named as missing data (missing stream file, missing clip and etc.). Namely, the missing data means 5 undownloaded data before data reproduction by a PlayList according to present invention starts.

In association with the above-mentioned description, the PlayList may be a Disc-dependent PlayList or a Disc-independent PlayList. If the second PlayList is equal to 10 the disc-dependent PlayList, a user can freely download/reproduce data referenced by the second PlayList at any time, or data referenced by the second PlayList can be reproduced by itself without being combined with a disc package. The present invention relates to the disc-dependent 15 PlayList. According to the present invention, data associated with a specific disc is downloaded from an external part, and the downloaded data is combined with the disc package as previously stated above.

A method for reproducing data according to the aforementioned 20 second PlayList will hereinafter be described with reference to FIG. 7. Specifically, Progressive PlayList is used as an example in FIG. 7. It is assumed that the virtual package exists before downloading data associated with a loaded disc from an external source in FIG. 7.

25 In association with the above-mentioned description, the

virtual package may be generated from a disc package contained in the disc after the disc is loaded in the optical recording/reproducing device. Otherwise, the virtual package may be generated by combining data of the disc package with  
5 data of the local storage. The virtual package may be updated by the data downloaded from the external source. A method for updating the virtual package after downloading some stream files will hereinafter be described in detail.

Referring to FIG. 7, a PlayList file (\*.mpls) of the  
10 Progressive PlayList #1, the clip information file 701 of a clip reproduced by the Progressive PlayList #1, and some (1.m2ts, 2.m2ts, and 3.m2ts) of the stream file 702 are downloaded before data reproduction starts.

If a update request for the virtual package occurs when all  
15 data associated with the Progressive PlayList #1 is not fully downloaded, the virtual package including the Progressive PlayList #1 is generated using the downloaded data (i.e., a PlayList file (\*.mpls), a clip information file 701, and stream files 1, 2, and 3 (1.m2ts, 2.m2ts, and 3.m2ts))  
20 according to the binding unit manifest file.

The aforementioned Progressive PlayList #1 is indicative of a PlayList capable of being played before all the data is downloaded. Therefore, after the virtual package is updated by the binding unit with missing stream files, the BD-J  
25 application can play the PlayList although all stream files

required to play the PlayList are not in place. The missing stream data (4.m2ts ~ n.m2ts) not downloaded before the virtual package is updated is downloaded while the virtual package is updated or while the Progressive PlayList #1 is  
5 played.

After the missing stream data (4.m2ts ~ n.m2ts) is stored at the location indicated by a binding unit manifest file, a BD-J application generates a request to reproduce a specified clip. After the player receives the above-mentioned request,  
10 it does not need to wait for a predetermined time to pass until a title is changed to another title. The player sets specified stream data to be read-only, and can immediately reproduce a corresponding clip according to the name mapping information contained in the binding unit manifest file.

15 In association with the above-mentioned description, the data reproduced by the Progressive PlayList #1 may be determined by Progressive PlayList information of the binding unit manifest file.

In association with the above-mentioned description, if data  
20 receiving a playback command is not equal to data specified by the Progressive PlayList information of the binding unit manifest file, or is determined to be pre-reproduced data, the aforementioned data receiving the playback command cannot be reproduced.

25 If a request for updating the virtual package occurs before

all data is not completely downloaded, it is confirmed whether the update of the virtual package is allowed by referring to the binding unit manifest file. In other words, it is confirmed whether the binding unit manifest file allows  
5 the PlayList (e.g., Progressive PlayList) of the present invention.

In association with the above-mentioned description, specific information indicative of the allowance or rejection of the Progressive PlayList may be included in the Progressive  
10 PlayList information. Namely, it can be confirmed by confirming whether the binding unit manifest file contains Progressive PlayList information. If the binding unit manifest file contains the Progressive PlayList information, the virtual package is updated so that the updated virtual  
15 package includes the Progressive PlayList. However, if the binding unit manifest file does not contain the Progressive PlayList information, the virtual package cannot be updated before the download of data is completed.

The update request of the virtual package may be executed by  
20 the BD-J application. If the missing data downloaded after the PlayList starts its reproduction is stored in a specific location indicated by the binding unit manifest file, the BD-J application may generate a command for reproducing the stored data, and may delete unnecessary data, which is not to  
25 be reproduced in the future, by referring to the binding unit

manifest file.

The Playlist file (\*.mpls) and the clip information file (\*.clpi) may be downloaded before the Progressive Playlist starts its reproduction. In other words, data, which is  
5 downloaded during the reproduction time of the Progressive Playlist whereas it is not downloaded before the reproduction begins, may be stream data.

The Progressive Playlist may reproduce downloaded stream files instead of stream files recorded in a recording medium.  
10 In other words, the Progressive Playlist should be treated as a new title that is bound to a disc. After streaming and playback operations are performed, the Playlist should be treated as a normal BD-J title.

FIG. 7 shows an example in which some of stream files are  
15 downloaded and then the virtual package is updated, but it should be noted that the present invention is not limited to the example of FIG. 7 and is applicable to other examples. For example, the virtual package may be updated on the condition that only the Playlist file (\*.mpls) and the clip  
20 information file (\*.clpi) are downloaded. In this case, stream data unused for updating the virtual package may be immediately downloaded simultaneously with updating the virtual package, or may be reproduced simultaneously with being downloaded upon receiving a playback command from a  
25 user.

In association with the above-mentioned description, the PlayList of the present invention may also be played by the following BD-J module.

The present invention downloads data associated with a recording medium from an external source, stores the downloaded data in a local storage, and generating a new virtual package using a binding unit with missing data before the data is completely downloaded in the local storage, such that the new virtual package includes a PlayList capable of being played while the missing data is downloaded. In this case, the PlayList is played by the generated virtual package, and the downloading/reproducing operations of the remaining data are the same as in FIG. 7.

In association with the above-mentioned description, if the missing data must be reproduced during the playback by the PlayList of the present invention, the present invention can stop the playback. In association with the above-mentioned description, a malfunction or error (e.g., undownloaded data needs to be reproduced) generated during the playback by the PlayList may be corrected by the BD-J application.

The BD-J application can play the Progressive PlayList even if the Progressive PlayList includes the missing clips. However, if the playback position reaches the missing stream point during the playback operation, the player allows the media player and the Playback Control Engine to enter a

stopped state. If the playback position is changed to the missing clip point by a jump operation of a user, the playback operation may stop at the jumped place and then the stop event may be transmitted to the BD-J application. If  
5 the BD-J application specifies a playback position with the missing stream, the player may allow the media player and the Playback Control Engine to enter a stopped state.

FIG. 8 is a block diagram illustrating an apparatus for reproducing data from a recording medium using a playback  
10 system according to the present invention.

In association with the above-mentioned description, the term "playback system" is indicative of a collective reproduction processor implemented by software and/or hardware contained in the optical recording/reproducing device. The playback  
15 system is indicative of a system which reproduces data of a recording medium loaded in the optical recording/reproducing device, and at the same time reproduces/manages data (e.g., data downloaded from an external part) stored in the local storage in association with the recording medium.

20 Specifically, the playback system 17 includes a user event manager 171, a module manager 172, an HDMV module 174, a BD-J module 175, a playback control engine 176, a presentation engine 177, and a virtual file system 40. A detailed description thereof will hereinafter be described.

25 The HDMV module 174 acting as an additional reproduction

process module capable of reproducing the HDMV title and the BD-J module 175 acting as an additional reproduction process module capable of reproducing the BD-J title are constructed independently of each other. The HDMV module 174 receives a  
5 command or program contained in the above-mentioned object file "Movie Object", and the BD-J module 175 receives a command or program contained in the above-mentioned object file "BD-J Object", such that the HDMV module 174 and the BD-J module 175 process the received command or program. The  
10 HDMV module 174 includes a command processor 174a, and the BD-J module 175 includes an application manager 175a and a Java VM 175b, such that the HDMV module 174 and the BD-J module 175 can receive/process the above-mentioned command.

The playback system includes the module manager 172, the  
15 playback control engine 176, and the presentation engine 177. The module manager 172 transmits a user command to the HDMV module 174 and the BD-J module 175, and controls operations of the HDMV module 174 and the BD-J module 175. The  
20 playback control engine 176 analyzes PlayList file content data recorded in a disc upon receiving a reproduction command from the HDMV module 174 and the BD-J module 175, and performs a reproduction function on the basis of the analyzed result. The presentation engine 177 decodes/displays a  
25 specific a specific stream reproduced/managed by the playback control engine 176.

Specifically, the playback control engine 176 is classified into a playback control function 176a for managing all the reproduction operations, and a player register 176b for storing player status register (PSR) information and general purpose register (GPR) information. The playback control function 176a may also be indicative of the playback control engine 176 as necessary.

In association with the above-mentioned description, the playback control engine 176 can access one of a plurality of smaller clips contained in the Progressive PlayList while another clip is downloaded.

The module manager 172, the HDMV module 174, the BD-J module 175, and the playback control engine 176 contained in the playback system can be processed by software. It is preferable that the above-mentioned components 172, 174, 175, and 176 are processed by software instead of hardware. The presentation engine 177, the decoder, and the planes are generally hardware based. Constituent components (e.g., 172, 174, 175, and 176) processed by software may be integrated in one unit, such that the integrated unit may be installed into the controller 12. Therefore, it should be noted that the above-mentioned components of the present invention be understood on the basis of their meanings, and are not limited to their implementation methods such as hardware or software implementation.

In association with the above-mentioned description, the characteristics of the playback system 17 are as follows.

Firstly, as stated above, the HDMV module 174 for the HDMV title and the BD-J module 175 for the BD-J title are  
5 configured independently of each other, and the two modules 174 and 175 cannot be executed at the same time. In more detail, the BD-J title cannot be reproduced when the HDMV title is reproduced, and the HDMV title cannot be reproduced when the BD-J title is reproduced.

10 Secondly, the playback system includes a BD-J applications for actually managing the local storage 15, such that the BD-J applications can manage a network function contained in the optical recording/reproducing device in the same manner as in additional data downloaded from an external part, and can  
15 form a virtual package by editing files stored in the local storage 15 or combining the files with the disc package. In other words, the BD-J applications form a virtual file system (VFS) 40 capable of managing a disc file system and a local storage file system as a single system, and form/manage the  
20 virtual package for reproducing original data and additional data using the virtual file system (VFS) 40. In association with the above-mentioned description, a method for generating the virtual package using the virtual file system (VFS) 40 has been previously stated above.

25 Thirdly, the HDMV title and the BD-J title receive user

commands based on different schemes, respectively. The HDMV title and the BD-J title use different methods for performing individual user commands. There is needed a predetermined unit capable of receiving the user command, and transmitting  
5 the received user command to either the HDMV module 174 or the BD-J module 175. In this case, the above-mentioned operations are performed by the user event manager 171a. For example, if the received command is determined to be a user command entered by a User Operation (UO), the user event  
10 manager 171a transmits the user command to the module manager 172 or the UO controller 171a. Otherwise, if the received command is determined to be a user command entered by a key event, the user event manager 171a transmits the received user command to the Java VM 175b contained in the BD-J module  
15 175.

Fourthly, one of the modules 174 and 175 manages the above-mentioned playback control engine 176 (also called a master). In more detail, when reproducing the HDMV title, the HDMV module 174 acts as a master of the playback control engine  
20 176. When reproducing the BD-J title, the BDJ module 175 acts as a master of the playback control engine 176.

In association with the above-mentioned description, the playback system may include a metadata manager 173. The metadata manager 173 provides a user with the Disc Library  
25 and Enhanced Search Metadata Application.

The PlayList according to the present invention may include at least one sub path along with a main path. If the PlayList includes the sub path, the main path and the sub path must be decoded/displayed at the same time.

5 In association with the above-mentioned description, in order to reproduce the main path and the sub path contained in the PlayList, data reproduced by the sub path is pre-downloaded, and other data reproduced by the main path is downloaded, such that a playback operation may begin. In this case, data  
10 referenced by the sub path must be pre-downloaded, such that a playback time is delayed a little.

To decode/display the main path and the sub path at the same time, data reproduced (referenced) by the main path of the PlayList and data reproduced (referenced) by the sub path of  
15 the PlayList are simultaneously downloaded and reproduced. A method for simultaneously downloading/reproducing data referenced by the main path of the PlayList and data referenced by the sub path of the PlayList will hereinafter be described with reference to FIGS. 9 and 10.

20 FIG. 9 shows a PlayList including a sub path configured with a single SubPlayItem (SPI) and an example for playing the PlayList according to the present invention. FIG. 10 shows a PlayList including a sub path configured with a plurality of SubPlayItems (SPIs) and an example for playing the PlayList  
25 according to the present invention.

The PlayList including a sub path configured with a single SubPlayItem (SPI) and its reproduction example are shown in FIG. 9.

The present invention may update the virtual package and may  
5 play the PlayList on the condition that only some part of the clip specified by the SubPlayItem is downloaded. The missing part of the clip may be downloaded while the virtual package is updated or while the PlayList is played, and a detailed description thereof will hereinafter be described with  
10 reference to FIG. 9.

The SubPlayItem (SPI) of the sub path of the Progressive PlayList #1 includes IN-Time information and Out-Time information of a clip to be reproduced. The clip specified by the SubPlayItem includes a single clip information file  
15 (11.clpi) 903 and a single stream file (11.m2ts) 904.

The update request of the virtual package may occur on the condition that the PlayList file (\*.mpls) of the Progressive PlayList #1, a clip information file 901 and some files (1.m2ts and 2.m2ts) of the stream files 902 reproduced by the  
20 main path, and a clip information file (11.clpi) 903 of the clips and some parts 904a of the stream file (11.m2ts) 904 reproduced by the sub path are downloaded. If the above-mentioned update request of the virtual package occurs, the virtual package is updated by the downloaded data. If the  
25 virtual package is updated, the aforementioned Progressive

PlayList #1 can be played.

The missing data data (3.m2ts ~ n.m2ts) 904b not downloaded before the virtual package is updated is downloaded while the Progressive PlayList #1 is played. The player can recognize  
5 how much data of the clip reproduced by the sub path is downloaded on the basis of information indicative of a downloaded length of the sub path data.

The PlayList (denoted by "Progressive PlayList #1" in FIG. 10) including a sub path configured with a plurality of  
10 SubPlayItems (SPIs) and its reproduction example are shown in FIG. 10.

If the sub path is configured with a plurality of SubPlayItems (SPIs), the clips referenced by the SPIs may be downloaded while the virtual package is updated or while the  
15 PlayList is reproduced in the same manner as in the aforementioned clips referenced by the PlayItem (PI) of the main path.

Referring to FIG. 10, similar to the main path configured with a plurality of PlayItems (PIs), the sub path is  
20 configured with a plurality of SubPlayItems (SPIs) (SPI #11 ~ SPI #m), and each SubPlayItem (SPI) designates a single clip. The PlayList file (\*.mpls) of the Progressive PlayList #1, the clip information file 1001 of the main path, the clip information file 1003 of the sub path, some files (1.m2ts and  
25 2.m2ts) of the stream file 1002 of the main path, and some

files (11.m2ts and 12.m2ts) of the stream file 1004 of the sub path are downloaded, and a request for updating the virtual package may occur.

The virtual package is updated before all the data reproduced by the Progressive PlayList #1 is downloaded, such that the virtual package includes the Progressive PlayList #1 according to the binding unit manifest file of the virtual package. If the virtual package is updated, the Progressive PlayList #1 may be played by the updated virtual package.

10 The missing data (3.m2ts ~ n.m2ts) not downloaded before the aforementioned reproduction operation begins may be downloaded while the virtual package is updated or while the Progressive PlayList #1 is reproduced, as previously stated above.

15 In association with the above-mentioned description, if the sub path is configured with a plurality of SubPlayItems (SPIs), the PlayList can be played on the condition that only some data of the clip is downloaded as in FIG. 9. The reproduction or playback methods shown in FIGS. 9 and 10 may  
20 be properly mixed according to the size of a downloaded clip. It is obvious to those skilled in the art that the aforementioned reproduction method of the PlayList may be applied to the main path as well.

Although the single sub path is exemplarily depicted in FIGS.  
25 9 and 10, at least one or two sub paths may exist in FIGS. 9

and 10. In this case, the simultaneous reproduction of the main path and the sub path may be performed as previously stated above, such that a detailed description thereof will be omitted for the convenience of description.

5 In association with the above-mentioned description, although FIGS. 9 and 10 exemplarily show a method for updating the virtual package after some stream files are downloaded, the present invention may update the virtual package on the condition that only the PlayList file (\*.mpls) and the clip  
10 information file (\*.clpi) are downloaded. In this case, stream data unused for updating the virtual package may be downloaded simultaneously with updating the virtual package, or may be downloaded/reproduced after a user enters a playback command, as previously stated in FIG. 7.

15 According to the preferred embodiment of the present invention, a new virtual package may be generated before all the data is completely downloaded, such that the new virtual package includes a PlayList capable of being played before all the data is completely downloaded, as previously stated  
20 in FIG. 7.

In association with the playback or reproduction operation of the present invention, the player must get to the next data to be reproduced before the playback or reproduction of current data is terminated, due to latency time of the  
25 decoder. In association with the present invention, the

aforementioned situation means that the data to be reproduced must be downloaded and stored at a specified location, before the player goes to a data storage location to reproduce the data. In this manner, it is preferable that data reproduced  
5 by the sub path is downloaded and stored at a specified location while other data is reproduced prior to the aforementioned movement of the player.

An apparatus for reproducing data from a recording medium using a local storage will hereinafter be described with  
10 reference to FIGS. 4a and 8. The apparatus for reproducing data from the recording medium using the local storage includes a pickup 11 for reading data of the recording medium; a local storage 15 for downloading data associated with the recording medium, and storing the downloaded data;  
15 and a controller 12 for generating a virtual package using a binding unit with missing data associated with the recording medium. Wherein the virtual package includes a PlayList capable of being played before all the data is completely downloaded. Also, the controller 12 can plays the PlayList  
20 while the missing data is downloaded.

In association with the above-mentioned description, in case that the virtual package already exists, the controller 12 updates the virtual package to generate new virtual package, such that the new virtual package includes the inventive  
25 PlayList.

In association with the above-mentioned description, the inventive PlayList can be called as Progressive PlayList. The controller 12 can generate the virtual package by a binding unit manifest file associated with the recording medium. The  
5 binding unit manifest file can contain Progressive PlayList information. In case that the binding unit manifest file does not contain Progressive PlayList information, the virtual package may not be generated before all the data is completely downloaded.

10 In association with the above-mentioned description, the missing data is indicative of stream data. The inventive PlayList reproduces data downloaded in the local storage. The PlayList can be configured with a main path and at least one sub path. In this case, the controller 12 can control to  
15 download simultaneously data referenced by the main path and data referenced by the sub path. The controller 12 can reproduce some clips or part of a clip while missing clip or missing part of the clip is downloaded.

The inventive PlayList may be reproduced/managed by the BD-J  
20 module 175, and the controller 12 may include the BD-J module. As apparent from the above description, a method and apparatus for reproducing data from a recording medium using a local storage according to the present invention can effectively reproduce additional data, which is downloaded in  
25 the local storage in association with the recording medium,

resulting in the creation of more convenient functions for a user.

### **Industrial Applicability**

5 It will be apparent to those skilled in the art that various modifications and variations can be made in the present invention without departing from the spirit or scope of the inventions. Thus, it is intended that the present invention covers the modifications and variations of this invention  
10 provided they come within the scope of the appended claims and their equivalents.

**[CLAIMS]**

1. A method for reproducing data from a recording medium using a local storage comprising the steps of:
  - 5 a) downloading data associated with the recording medium from an external source, and storing the downloaded data in the local storage;
  - b) generating a virtual package using a binding unit with missing data associated with the recording medium, such that  
10 the virtual package includes a PlayList capable of being played before all the data is completely downloaded; and
  - c) playing the PlayList while the missing data is downloaded.
2. The method according to claim 1, wherein the missing data  
15 is indicative of stream data.
3. The method according to claim 1, wherein the PlayList reproduces data downloaded in the local storage.
- 20 4. The method according to claim 1, wherein the virtual package is generated by a binding unit manifest file associated with the recording medium.
5. A method for reproducing data from a recording medium  
25 using a local storage comprising the steps of:

a) generating a virtual package for reproducing data in the recording medium and data in the local storage;

b) downloading data associated with the recording medium, and updating the virtual package using a binding unit with  
5 missing data, such that the virtual package includes a PlayList capable of being played before all the data is completely downloaded; and

c) playing the PlayList while the missing data is downloaded.

10 6. The method according to claim 5, wherein the PlayList includes a main path and at least one sub path, and data referenced by the main path and data referenced by the sub path are simultaneously downloaded.

15 7. The method according to claim 6, wherein the sub path is configured with a single SubPlayItem, and downloaded part of a clip referenced by the SubPlayItem is reproduced while the missing part of the clip is downloaded.

20 8. The method according to claim 6, wherein the sub path is configured with a plurality of SubPlayItems, and downloaded clips referenced by the SubPlayItems are reproduced while the missing clips are downloaded.

25 9. The method according to claim 6, wherein the sub path is

configured with a plurality of SubPlayItems, and downloaded part of a clip referenced by one of the SubPlayItems is reproduced while the missing part of the clip is downloaded.

5 10. The method according to claim 5, wherein the PlayList is played by a BD-J module.

11. The method according to claim 5, wherein the virtual package is updated by a request of BD-J applications.

10

12. The method according to claim 5, further comprising the step of:

in case that playback position by the PlayList reaches the missing data point during playback, stopping the playback.

15

13. The method according to claim 5, further comprising the step of:

in case that an error occurs while the PlayList is played, the error is corrected by execution of BD-J applications.

20

14. The method according to claim 5, wherein the PlayList is indicative of a Progressive PlayList.

15. The method according to claim 14, wherein data referenced  
25 by the Progressive PlayList is indicated by a binding unit

manifest file associated with the recording medium.

16. The method according to claim 15, wherein the step b) for updating the virtual package includes the step of:

5 confirming whether the binding unit manifest file contains Progressive PlayList information.

17. An apparatus for reproducing data from a recording medium using a local storage comprising:

10 a pickup unit for reading data of the recording medium;  
a local storage for downloading the data associated with the recording medium, and storing the downloaded data; and  
a controller for generating a virtual package using a binding unit with missing data associated with the recording medium,  
15 such that the virtual package includes a PlayList capable of being played before all the data is completely downloaded, the controller for playing the PlayList while the missing data is downloaded.

20 18. The apparatus according to claim 17, wherein in case that the virtual package exists, the controller updates the virtual package to generate a new virtual package, such that the new virtual package includes the PlayList capable of being played before all the data is completely downloaded.

19. The apparatus according to claim 17, wherein the missing data is indicative of stream data.

20. The apparatus according to claim 17, wherein the PlayList  
5 reproduces data downloaded in the local storage.

21. The apparatus according to claim 17, wherein the controller generates the virtual package by a binding unit manifest file associated with the recording medium.

10

22. The apparatus according to claim 17, wherein the PlayList includes a main path and at least one sub path, and controller simultaneously downloads data referenced by the main path and data referenced by the sub path.

15

23. The apparatus according to claim 22, wherein the sub path is configured with a single SubPlayItem, and the controller reproduces downloaded part of a clip referenced by the SubPlayItem while the missing part of the clip is downloaded.

20

24. The apparatus according to claim 22, wherein the sub path is configured with a plurality of SubPlayItems, and the controller reproduces downloaded clips referenced by the SubPlayItems while the missing clips are downloaded.

25

25. The apparatus according to claim 22, wherein the sub path is configured with a plurality of SubPlayItems, and the controller reproduces downloaded part of a clip referenced by one of the SubPlayItems while the missing part of the clip is  
5 downloaded.

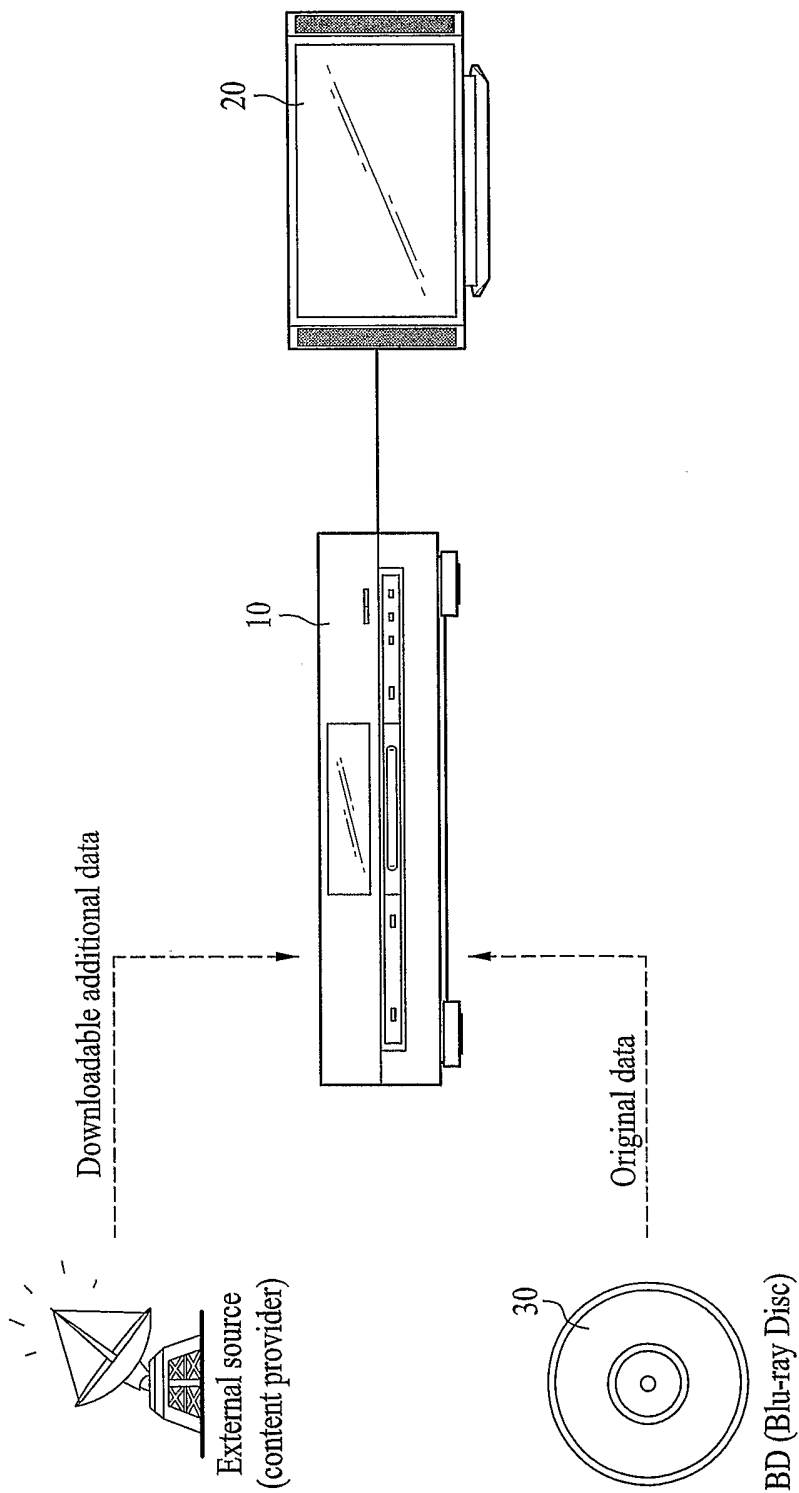
26. The apparatus according to claim 17, wherein the controller includes a BD-J module, and  
wherein the BD-J module plays the PlayList.  
10

27. The apparatus according to claim 17, wherein the PlayList is indicative of a Progressive PlayList.

28. The apparatus according to claim 23, wherein data  
15 referenced by the Progressive PlayList is indicated by a binding unit manifest file associated with the recording medium.

29. The apparatus according to claim 24, wherein the  
20 controller, in generating the virtual package, confirms whether the binding unit manifest file contains Progressive PlayList information.

FIG. 1



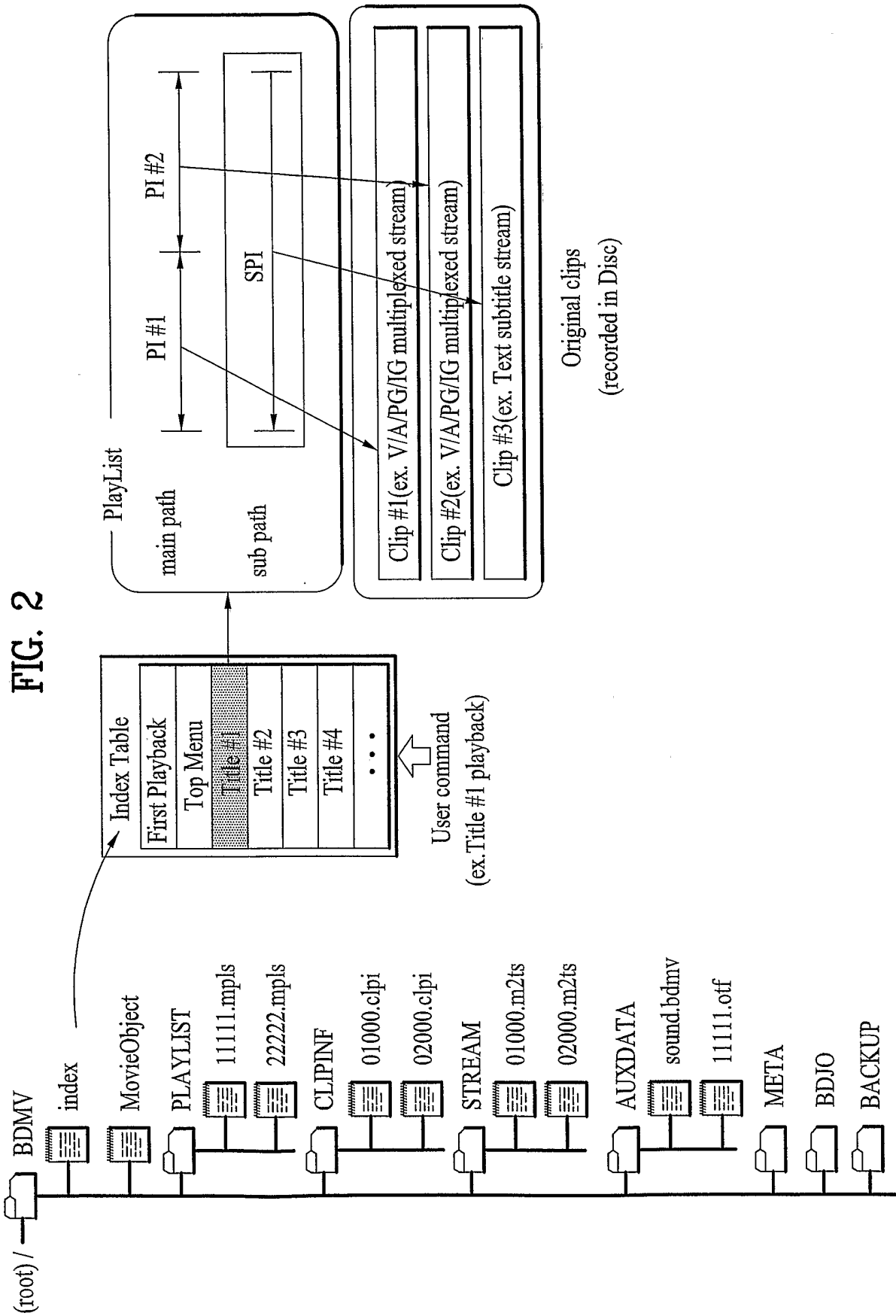


FIG. 2

FIG. 3

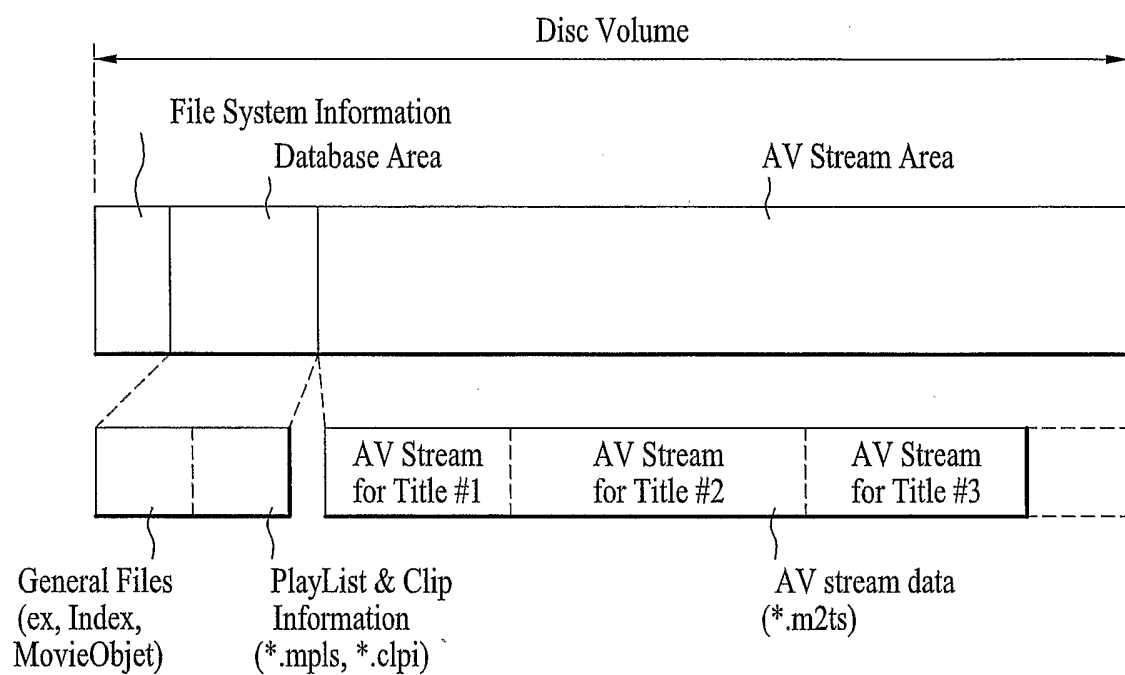


FIG. 4A

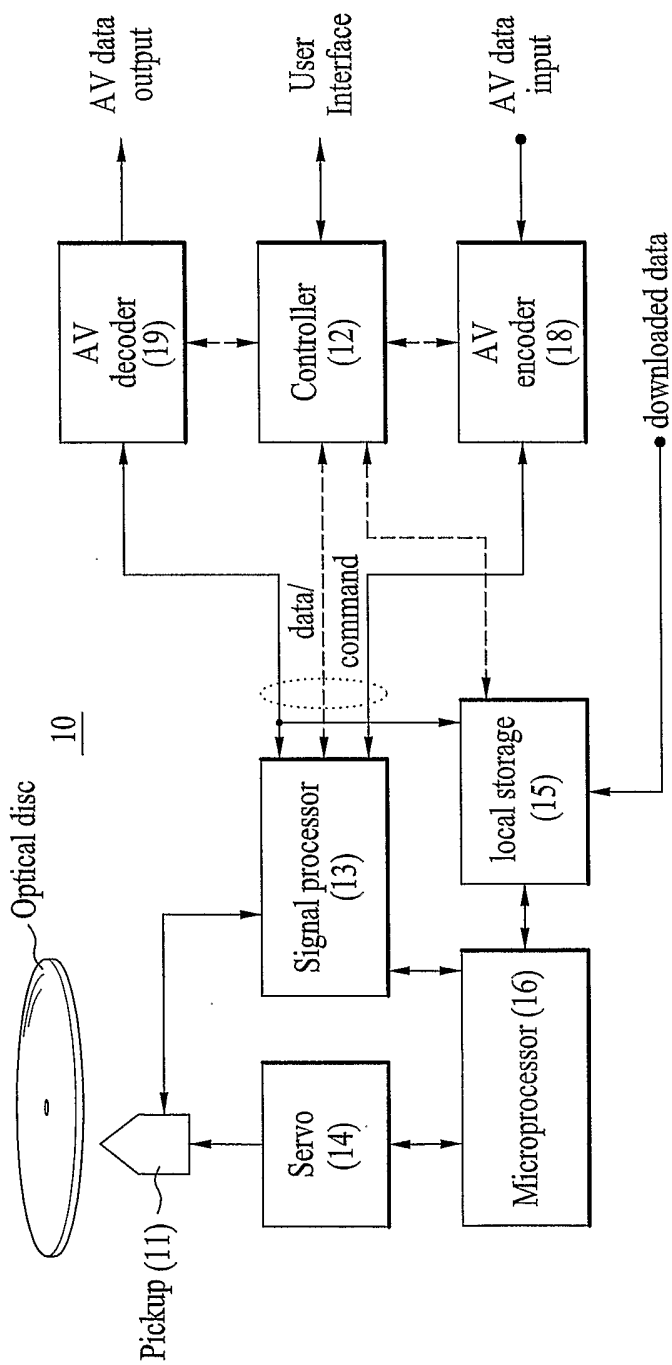


FIG. 4B

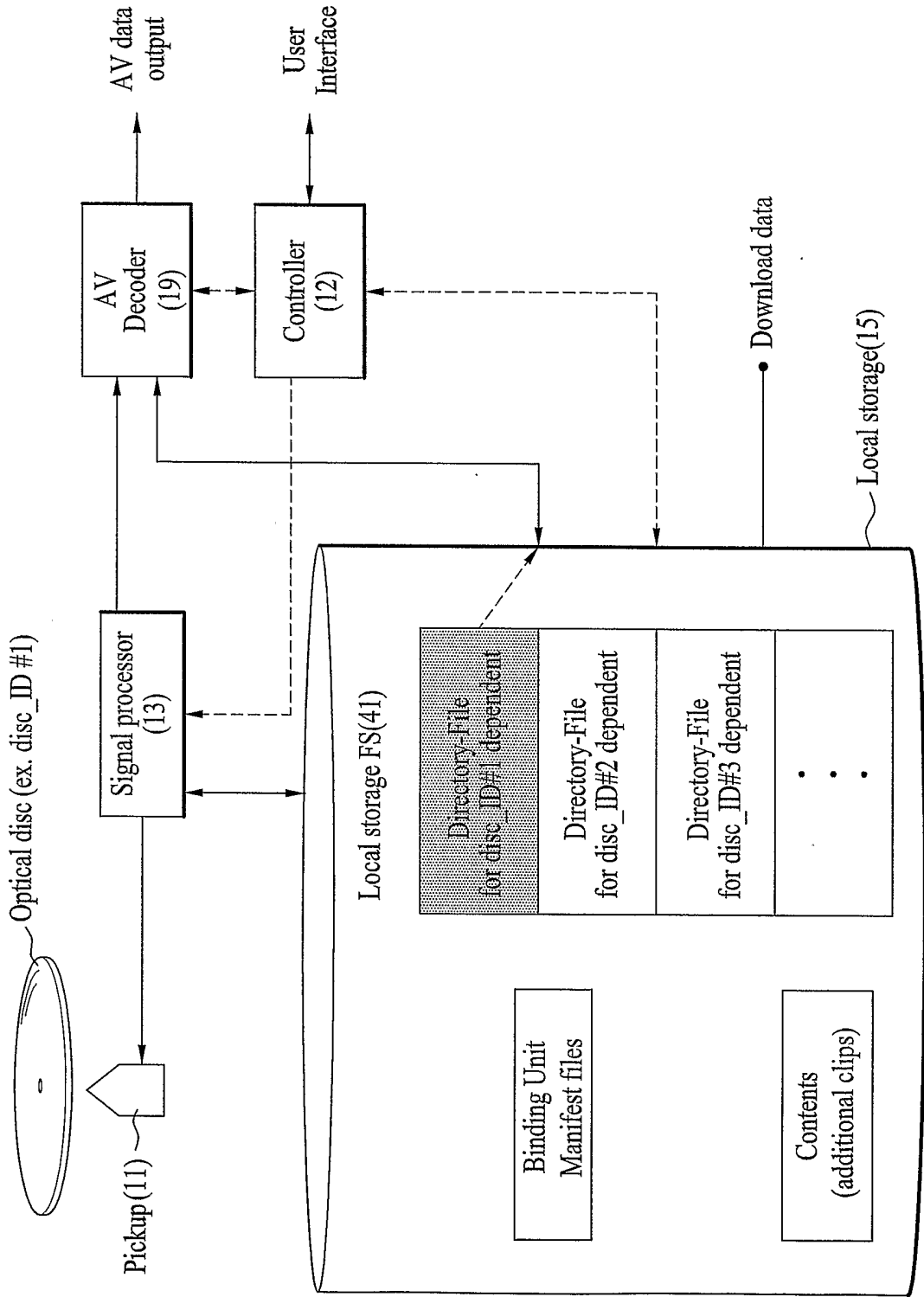


FIG. 5

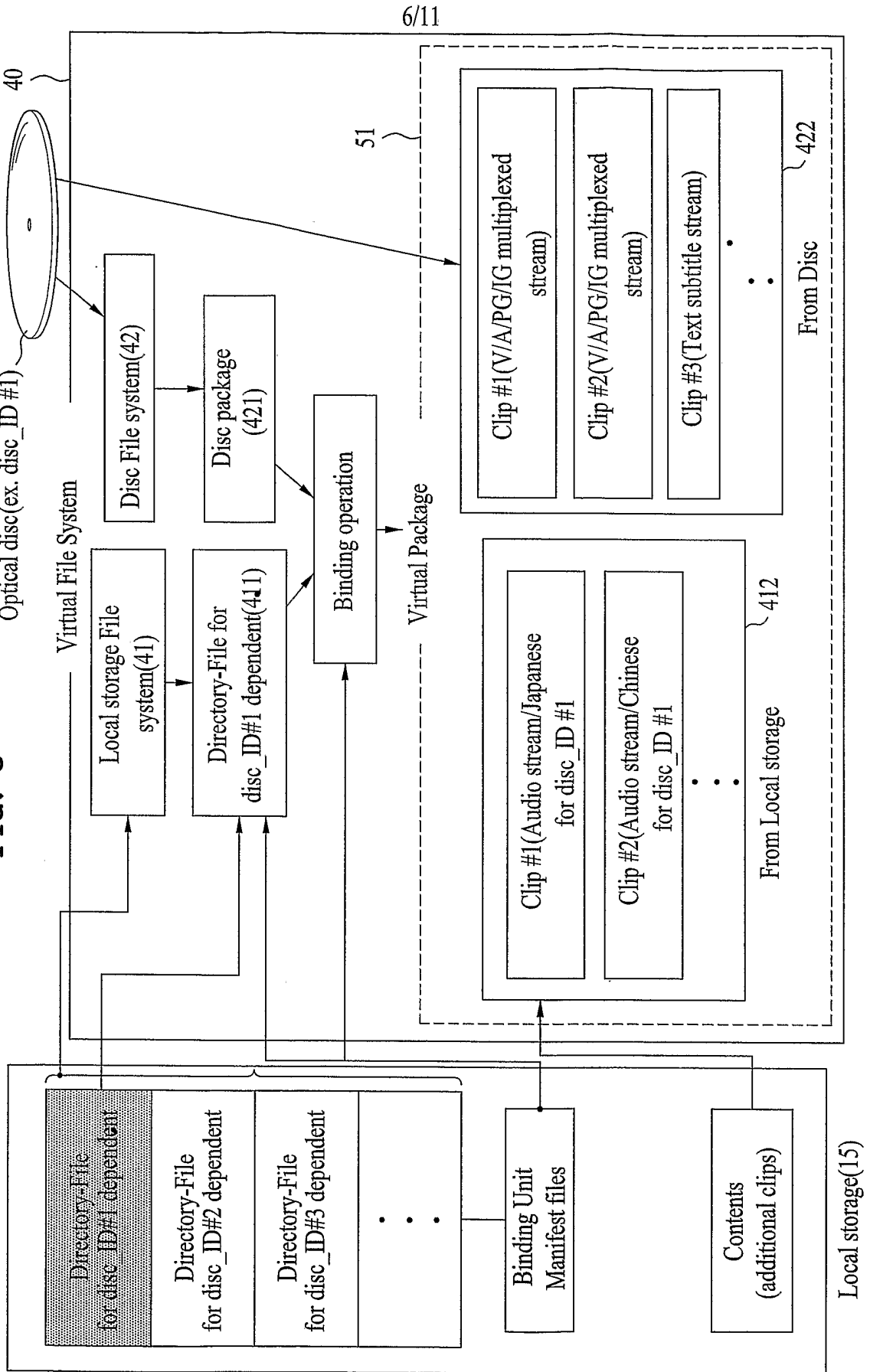


FIG. 6

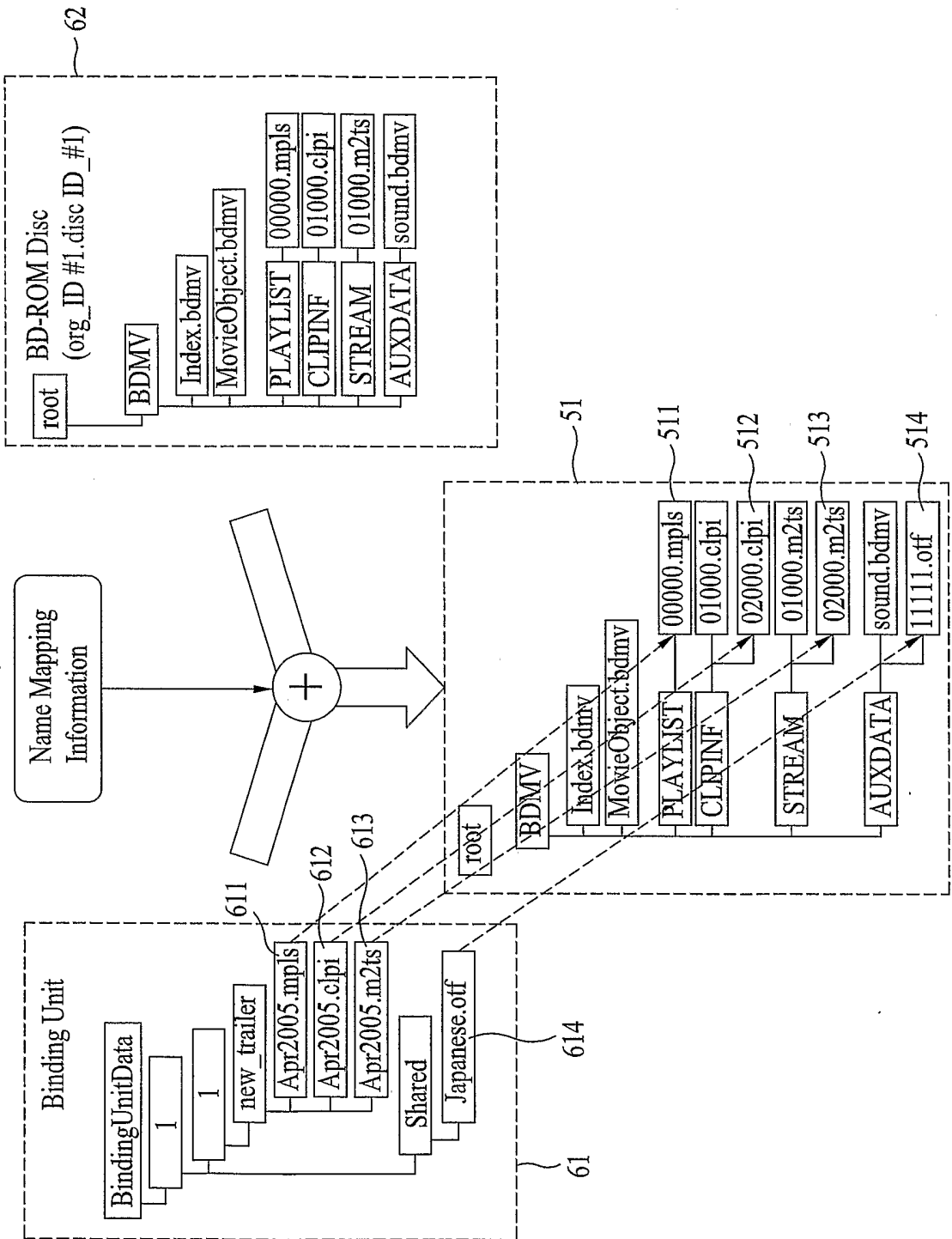


FIG. 7

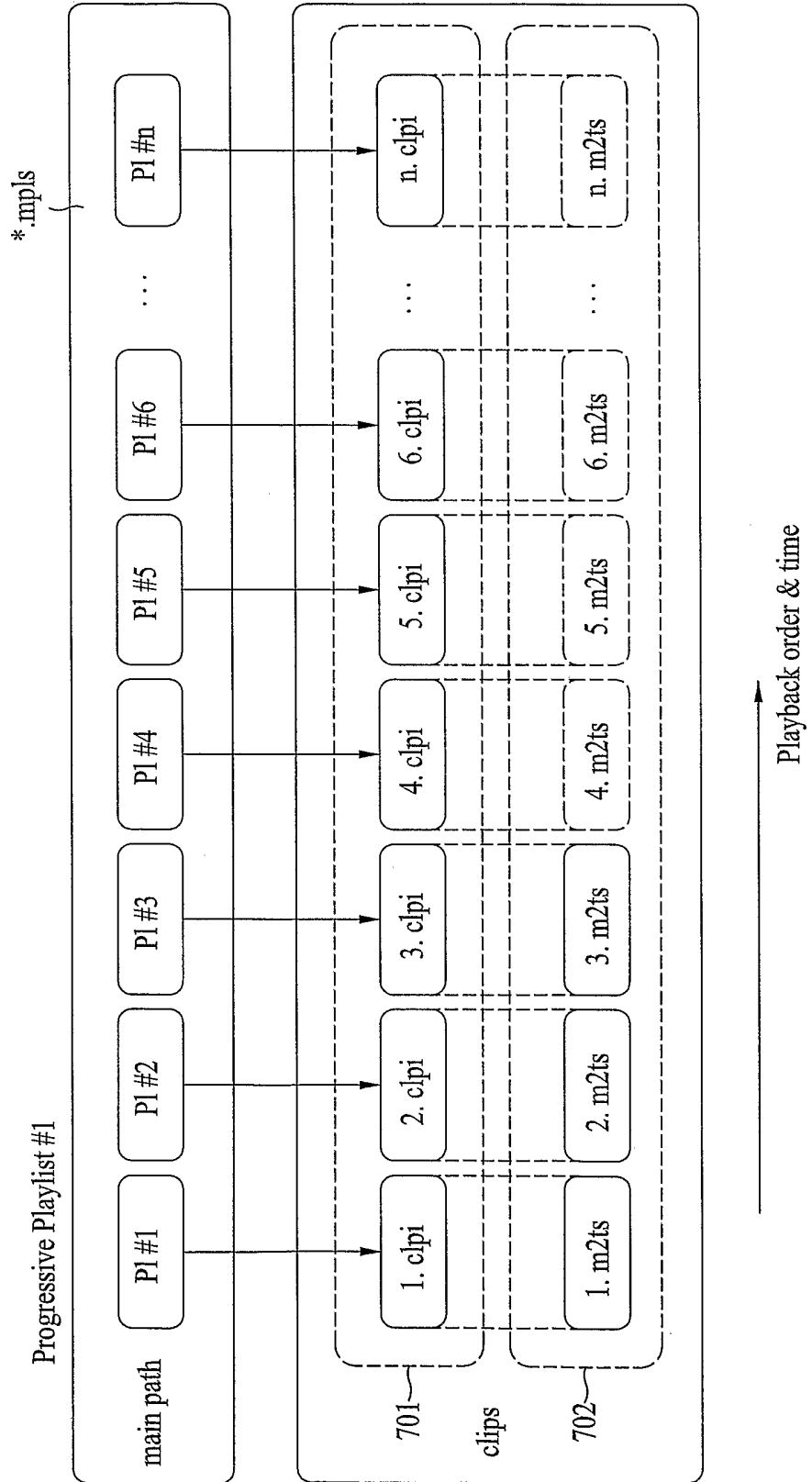


FIG. 8

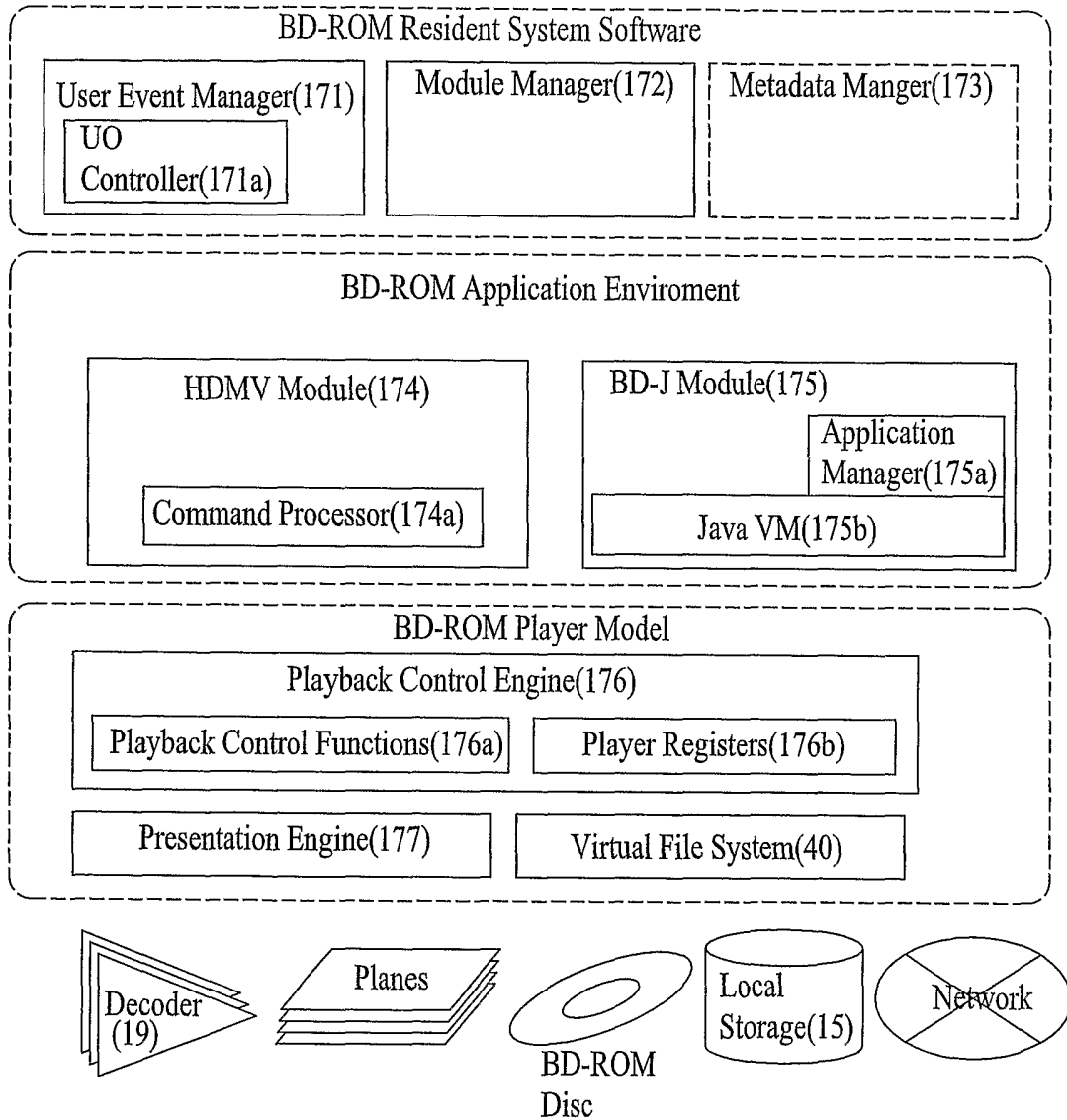


FIG. 9

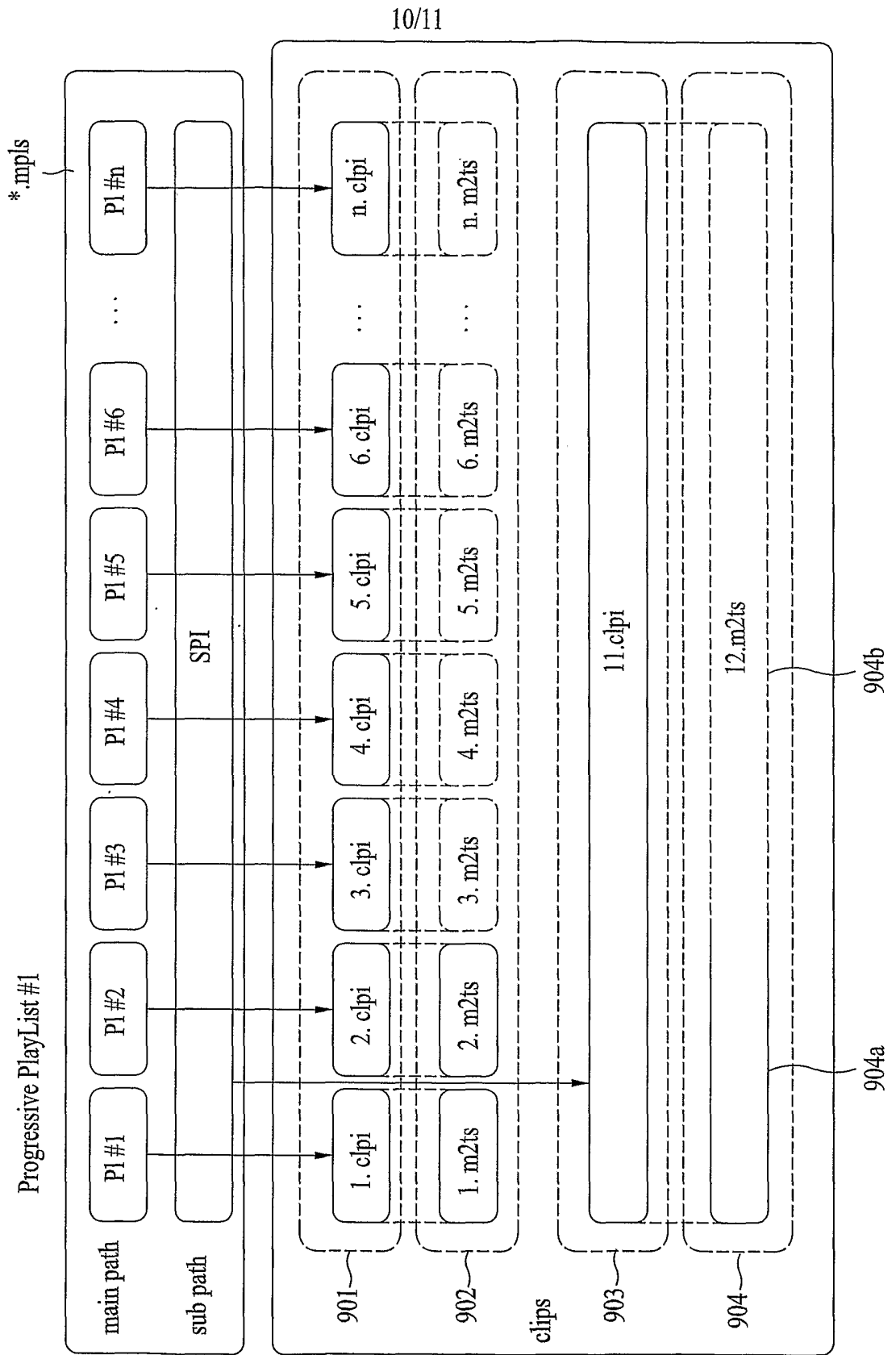


FIG. 10

