# AV-use File Systems for Multiple High-definition Era

Nobuaki Kohinata
Damien Le Moal
Mika Mizutani

*OVERVIEW: Accompanying the spread of AV equipment fitted with large-capacity HDDs and high-speed network interfaces, a new style of enjoying content—in which all recorded content can be enjoyed freely anywhere in the home—will become mainstream in the near future. In the file system for handling this new viewing/listening style, processing must be performed at high efficiency while assuring the access rates for writing to the HDD storing content and for reading data from the HDD. Aiming to create a middleware solution to meet the above-mentioned requirements, Hitachi has developed, and is presently commercializing, an AV-use file system that enables simultaneous access to multiple "high definition" content (i.e. HDTV programs). Focusing on developing middleware for improving the added-value of HDDs, we are continuing to intensify and push forward our research and development on fundamental technologies for supporting people's "new digital lives."*

## INTRODUCTION

ACCOMPANYING the popularization of AV (audio-visual) equipment fitted with network I/Fs (interfaces) and large-capacity HDDs (hard disk drives), and the launch of terrestrial digital broadcasting, it is considered that, from now onwards, the way that users view and listen to content will continue to change. For example, the viewing/listening pattern is such that HDTV (high-definition TV) content that has already been recorded can be freely enjoyed in the home while programs on all channels are being recorded. Making this kind of viewing a reality necessitates a scheme that allows multiple read/writing processing operations on an HDD simultaneously at high efficiency while assuring the rate for each HDD access (see Fig. 1).

In facing these challenges, Hitachi is proposing

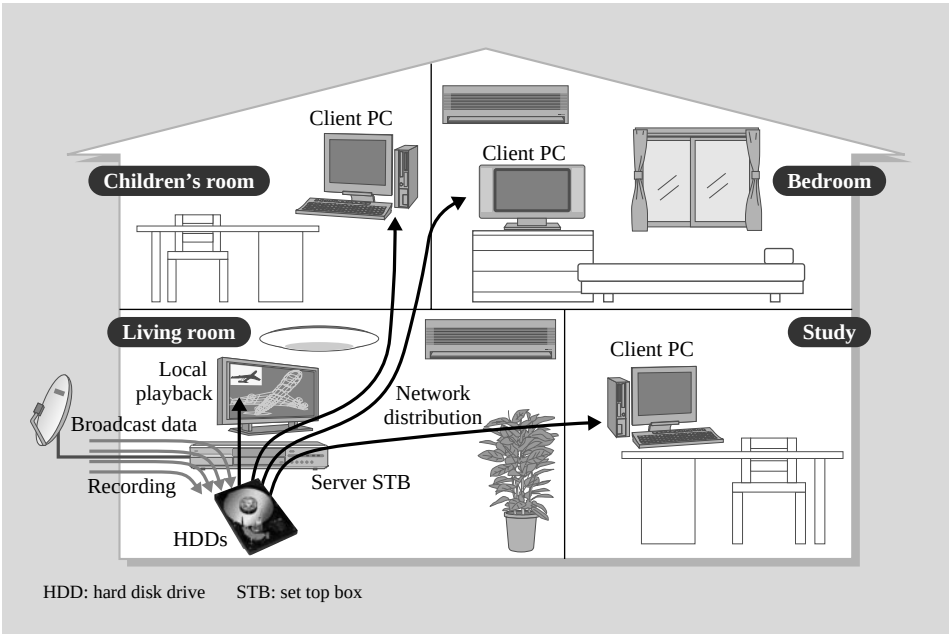


*Fig. 1—Server STB for Multi-rooms.*
*With the start of terrestrial digital broadcasting and the popularization of home networks, it has become necessary that HDDs of server STBs must handle multiple reading and writing simultaneously. Improving the performance of server STBs without raising the parts cost requires using a file system that accounts for the characteristics of high-definition content and HDDs.*

middleware solutions for improving the added value of HDDs.

In the rest of this article, a high-performance AV-use file system—which enables simultaneous processing of multiple content data while keeping down increases in parts cost—is presented, an example of its application is described, and the effectiveness of its application is explained.

## NEEDS OF FILE SYSTEMS

The term "file system" can be taken variously according to the intended system. In this article, it relates to either a data-management method that records data on "local HDDs" fitted in integrated devices like AV equipment and STBs (set-top boxes) and then reads or rewrites the recorded data or a sequential concept or software program relevant to this process.

Such a file system is provided as one function of a ordinary OS (operating system), and presently there is a broad range of file systems used for various OSs (see Table 1).

As regards the capability of file systems fitted in PCs and servers, although such systems are slow by nature compared to volatile memory allowing high-speed access, they must meet basic needs such as reliably recording large amounts of data (like that stored on HDDs) on non-volatile recording media and managing that data(1).

Moreover, various integrated devices other than PCs and servers now use HDDs, so the needs of file systems have become diversified. For example, even when a system cannot shutdown normally (such as during accidental power failure), ease of operation—like assuring the uniformity of management information of data (metadata) and checking and fixing that data promptly—is required.

From now onwards, as a result of the popularization of AV equipment fitted with high-speed network I/Fs and large-capacity HDDs of the several-hundred-gigabyte class, it is considered that the popular way of viewing content will change to one in which once all programs are recorded, the user can enjoy the content of their choice at their preferred time and place. In response to this change in viewing style, efficiently managing simultaneous read/write of multiple content at a constant access rate to the HDD while keeping down the cost of AV equipment necessitates a file system aimed at specific applications (see Table 2).

TABLE 1. File Systems for Various OSs
*Other than those listed below, file systems using recording media such as optical disks and flash memories are available.*

| OS | File system | Development basis |
|---|---|---|
| MS-DOS*1 | FAT16 | Microsoft |
| MS-Windows*1 | FAT32 | Microsoft |
| | NTFS | Microsoft |
| Mac OS*2 | HFS+ | Apple |
| Linux*3 | Ext2FS | Remy Card |
| | Ext3FS (2) | Stephen Tweedie |
| | ReiserFS (3) | Namesys |
| | JFS (4) | IBM |
| | XFS (5) | SGI |

OS: operating system
FAT: file allocation table
NTFS: NT file system
HFS+: hierarchical file system plus
Ext2FS: second extended file system
Ext3FS: third extended file system
JFS: journaled file system

*1 MS-DOS and Windows are registered trademarks of Microsoft Corporation in the U.S. and other countries.
*2 Mac OS is a trademark of Apple Computer, Inc. registered in the U.S. and other countries.
*3 Linux is a trademark of Linus Torvalds.

TABLE 2. Data Sizes of Representative Video Content
*In the case of 120-min-worth of HDD content, disk capacity of 18 Gbytes is consumed.*

| Type of content | Average bit rate | Movie time | Data size |
|---|---|---|---|
| SD | 8 Mbit/s | 120 min | 7.2 Gbyte |
| HD | 20 Mbit/s | 120 min | 18 Gbyte |

SD: standard definition
HD: high definition

## DEVELOPMENT OF A FILE SYSTEM FOR AV USE

### Implementation Strategy

In responding to the new needs concerning file systems, Hitachi has developed, and is currently commercializing, a file system for AV use. With Linux (which is widely used as an OS for AV equipment and STBs) as a target, this file system for AV use is placed in the same layer as Ext3FS (third extended file system) etc. under a VFS (virtual file system) (see Fig. 2). As a result of this configuration, for example in the case of a single HDD unit, the volume of additional information related to program guides and recorded content utilizes the volume of the Ext3FS and the main body of content utilizes the volume of the AV-use file system. Moreover, regardless of form factor (namely, 2.5-inch or 3.5-inch HDDs), disk capacity, or manufacturer, the developed AV-use file system can be applied to a variety of HDDs.

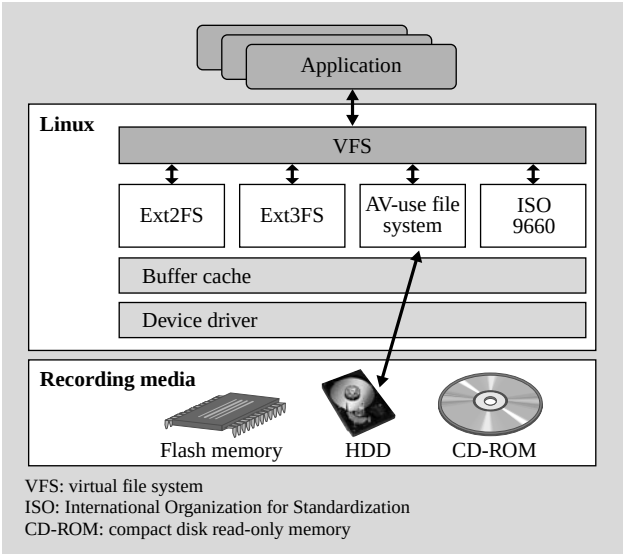


*Fig. 2—Position of AV-use File System.*
*By means of invoking a "system call" from an application connected to the file system, the AV-use file system can be used at exactly the same time as another file system.*

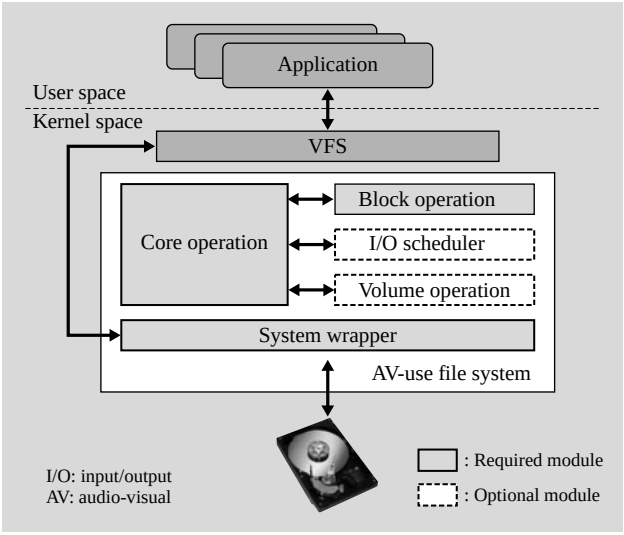


*Fig. 3—Modular Structure of AV-use File System.*
*OS-dependent parts are absorbed by the system wrapper. By changing the modules, OSs other than Linux can be easily transplanted.*

## Development Target

(1) Performance

Attaining high day-to-day access performance to content requires that content is recorded in a way that does not finely divide it up (as in "fragmentation"). This is because in the case of performance of an HDD, fragmentation of files causes fragmentation of I/O (input/output) to the HDD, which in turn adds to the seek operation of the disk head. Moreover, it is also necessary to avoid recording image content disproportionately on the inner diameter (with slower access speed than the outer diameter of the disk) only.

(2) Operability

On abnormal shutdown (namely, accidental power shut-off), the high-speed file system is checked and repaired, and to ensure that the user is not aware of the operation time of that sequence, it is necessary to support "journaling" (ordered data mode) for metadata.

For example, as for Ext2FS (second extended file system: with no journaling function), on restart after a system crash, the file system is checked and repaired by means of "fsck" commands. However, as the capacity of HDDs has increased, that checking/repair time has also increased and, accordingly, a great deal of time is needed until the system is restarted. By utilizing support by journaling for metadata, however, it is possible to quickly restore the system and restart operation.

(3) Scalability

To improve performance and operability in response to the user's utilization environment, a variety of setups must be possible. For example, a file-management function definable by freely adding metadata, appropriate value of data block size for workloads (data type: text, video, etc.), etc. can be changed.

## Features of AV-use File System

The developed AV-use file system—based on the development goals outlined in this article—has the main features listed as follows:

(1) Maximum file-system size (partitioned): 4,100 petabytes (with data block size: 4 Mbyte; metadata block size: 4 kbyte)

(2) Compact metadata structure makes possible high-speed access to metadata and reduction of disk consumption.

(3) Data block size: possible to choice from 4 kbyte to 4 Gbyte

(4) Data arrangement considering realtime characteristic

(5) Metadata-journaling support

(6) Architecture with easy-to-expand functionality

## Modular Structure

As a result of flexible extendibility of single features, various functions can be "plugged" into the core operation module (see Fig. 3).

(1) Core operation

With the file system as a modular configuration,

one module performs the main processing (core operation), providing journaling and various management functions (for metadata block, disk, volume, i-nodes, files, I/O, etc.).

(2) System wrapper

OS-dependent components are taken in, and all I/O processing issued to the HDD is supported.

(3) Volume operation

Plug-in modules that can assign additional information to the metadata structure are supported. For example, right of access and ordering of priority can be defined according to content copyright.

(4) Block operation

Plug-in modules that provide fragmentation algorithms for data blocks are supported.

(5) I/O scheduler

For example, the I/O scheduler uses access priority defined according to volume operation in order to determine the order of priority of file I/Os.

## APPLICATION EXAMPLE AND EFFECTIVENESS OF AV-USE FILE SYSTEM

### Multi-room System

This system can store several kinds of HDTV content on an HDD formatted by the AV-use file system, and it enables simultaneous viewing of the content in several locations via a network as well as local playback. This means that, as well as local playback, the demands for distribution on three networks and double-speed playback at local distribution addresses can be handled.

### Performance Results

The maximum number of HDTV contents that can be read out simultaneously from the HDDs (applied in the AV-use file system) at each client in the multi-room system was measured visually (see Fig. 4). Our new AV-use file system can stably read out a maximum of 10 units of HDTV content, regardless of the location on the disk, which represents a more-than-twofold increase in performance compared with Ext2FS.

## CONCLUSIONS

This article described a new AV-use file system for taking account of the characteristics of high-definition image content (like HDTV programs) and for efficiently utilizing an HDD. With this file system, multiple HDTV content can be handled simultaneously.

From now into the future, focusing on development of middleware for improving the added-value of HDDs, Hitachi will continue to push forward with research and development on fundamental technologies for supporting people's "new digital life".

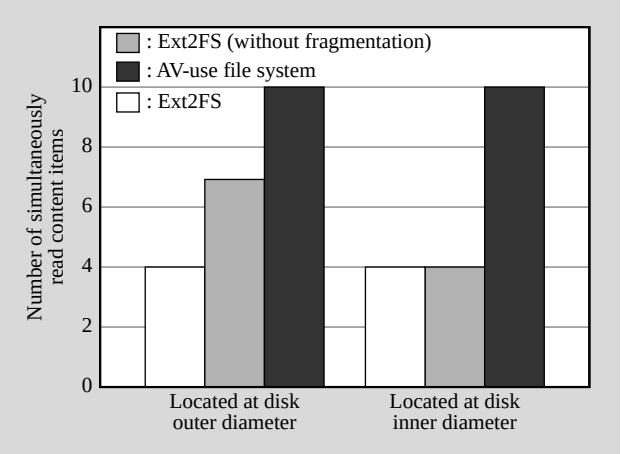


*Fig. 4—Performance of AV-use File System (Reading). 25-Mbit/s hard-disk content was located at either the outer or the inner diameter of an HDD and measured.*

## REFERENCES

(1) Dominic Giampaolo, "Practical File System Design" (1999).
(2) Stephen Tweedie, "Journaling the Linux ext2fs Filesystem," LinuxExpo '98 (1998).
(3) ReiserFS web site, http://www.namesys.com
(4) JFS web site, http://jfs.sourceforge.net
(5) Project XFS web site, http://oss.sgi.com/projects/xfs/

## ABOUT THE AUTHORS

Nobuaki Kohinata

*Joined Hitachi, Ltd. in 1996, and now works at the 6th Department of the Systems Development Laboratory. He is currently engaged in research and development of network and HDD middleware for consumer electronics devices.*

Demien Le Moal

*Joined Hitachi, Ltd. in 2000, and now works at the 3rd Department of the Systems Development Laboratory. He is currently engaged in the research and development of file systems and middleware for AV streaming.*

Mika Mizutani

*Joined Hitachi, Ltd. in 1987, and now works at the 6th Department of the Systems Development Laboratory. She is currently engaged in the research and development of networks and HDD middleware for consumer electronics devices. Ms. Mizutani is a member of The Information Processing Society of Japan (IPSJ).*