



**Method and Apparatus for Recording and Playback  
of Computer KVM Session Video Image Data**

## Introduction

Video image recording is widely used in a great number of practical applications. In many cases, the changes occur only in small or scattered areas, and the video images either change slowly; remain unchanged for a period of time, or leave large portions of the image unchanged. One prominent example is the recording and playback of a computer KVM session in which the video or graphics output of a computer is streamed to a user's monitor. Another example is found in video surveillance recording, where a video camera constantly records images of a selected area. This whitepaper discusses a video recording method that is suitable for recording computer video sessions and has been implemented in ATEN's Control Center Video Session Recorder (CCVSR) solution.

ATEN's CCVSR is an innovative software solution that automatically makes a video recording of the operations administrators perform when using their KVM over IP switches. The solution comprises a server running CCVSR software (the CCVSR server), and at least one KVM over IP switch. The KVM over IP switch acts as a video capture device that collects, processes, and transmits video data from a source computer to a storage device – i.e., the CCVSR server. The CCVSR server stores the video data and can produce any portion of the stored video images for playback, according to user request.

The CCVSR method implements an innovative video data recording scheme for recording computer video sessions. In traditional video recording sessions, the data is composed of a stream of individual image frames that have been captured and saved. With the CCVSR method, however, each image frame is divided into multiple blocks. Once all the blocks of each initial frame have been saved, only the blocks within that image frame that have changed are saved. The KVM over IP switch captures and processes the video image frames, but for each frame, only transmits the blocks that have changed to the CCVSR server. The CCVSR server stores the changed blocks in a data file, together with block indexes and locations, and maintains a virtual frame table to keep track of the data block locations for each video image frame. It also maintains a snapshot file containing snapshots with timestamps of the virtual frame table. The data file and the snapshot file are used to reproduce the video images for playback of the session records. The apparatus and the recording method are detailed in the following sections.

## Apparatus

Each computer normally has a console consisting of a monitor and user input devices such as a keyboard and mouse (referred to as a KVM), that allows users to operate the system. When a KVM over IP switch is connected to several computers, it supports the remote consoles (virtual KVMs), running on the client PCs over a TCP/IP network. Once an administrator from a client PC logs in to the KVM over IP switch and accesses one of the connected computers, a computer KVM session is established. The video or graphics output of the accessed computer is streamed to the client PC's monitor, and the keyboard and mouse input from the client's PC is transmitted to the accessed computer.

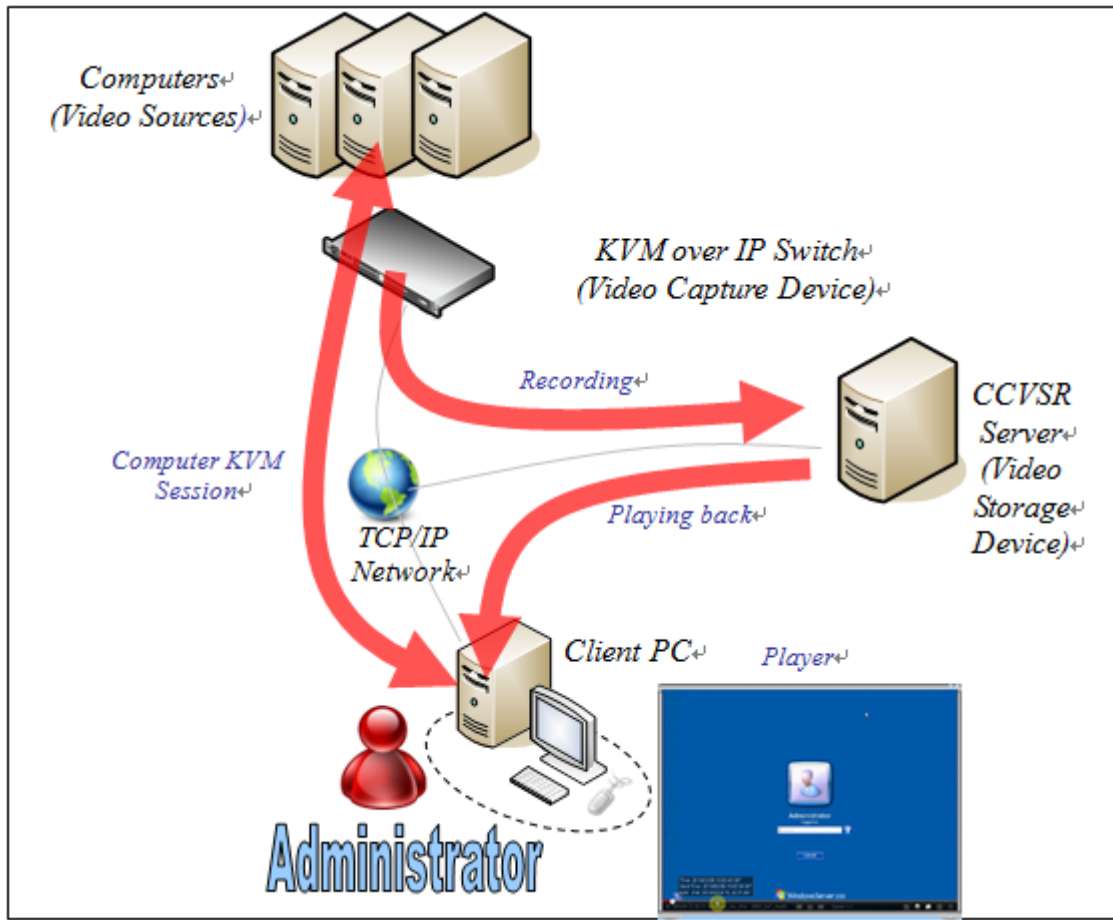


Figure 1 Apparatus

If a CCVSR server is added and configured properly, an additional session, between the KVM over IP switch and the CCVSR server, is automatically established to record both the video and non-video data of the computer KVM session. When the administrator logs out of the accessed computer and the computer KVM session ends, the CCVSR recording session automatically terminates, as well.

Records of the KVM session, including video and non-video data, are stored in the CCVSR server. Authorized users can remotely access the CCVSR server, search the records, and select a record for playback. The selected video record and a player program are downloaded from the CCVSR server to the authorized users' PCs, where they can play back the selected video and key strokes. The downloaded player program and the video record can be saved separately on the local PC (with password protection) for later playback.

The apparatus is illustrated in Figure 1. Although the CCVSR server keeps video image data and non-video information (keyboard data, mouse data, username, port number, access, bus ID, login and logout events, etc.), the method currently under discussion focuses on the recording of video data.

## Video Capture and Processing Method

Video data recording can be divided into two steps. The first is video capture; the second is video storage. If the video capture occurs at the source computer, the computer has to have a software capture program installed to perform the capture. The capture software is usually called an agent, and the capture method is termed agent-based recording. In contrast, the method we present can be termed agentless recording, since the video capture is performed by the KVM over IP switch, and there is no need for a software agent on the video source computer

The KVM over IP switch conducts two sessions for each recorded video session. One session connects to the client PC; the other session connects to the CCVSR server. The video image data that is transmitted to the CCVSR server may be the same as that being transmitted to the client PC, or it may be different. With the CCVSR server method the latter case is used, and different video image data is transmitted to the CCVSR. In order to minimize network bandwidth during transmission and to reduce the amount of disk space needed to store the video data, an advanced video processing procedure is implemented concurrent with the video capture, as illustrated in Figure 2.

To begin, each KVM over IP switch has a number of analog to digital converters (ADCs) that convert analog video signals to digital video data. Each ADC can connect to a computer that serves as the video source. Video image frames are sampled according to a selected frame rate, and processed. The procedure differs from more usual methods in that instead of compressing and transmitting entire image frames, each video frame is divided into an array of blocks, and only those blocks that have changed are compressed and transmitted. Each block within the frame is designated by a block index. The simplest example is shown in Figure 3, which depicts a frame divided into 4 blocks (2 x 2), with the blocks counted sequentially from left to right and top to bottom. The block index is an integer, starting from one, indicating the position of each block within a frame. (An alternate method of keeping track of a block's location is to use a pair of integers that represent the row and column numbers of its position.)

At the beginning of each recorded session, one complete video frame -- the initial frame -- consisting of a number of initial blocks is processed and transmitted. Thereafter, as subsequent frames are processed only the changed blocks in the changed frames are transmitted. Specifically, the KVM over IP switch compares each block of the current frame to its corresponding block in the previous frame to detect if there have been any changes, and then compresses the changed blocks. For each video frame that contains changes (changed frames), the KVM over IP switch transmits a timestamp followed by the changed blocks, with each changed block containing a block index indicating its position.

The timestamp is a number that can represent either an absolute or relative time value. The absolute time value is precise to the second; the relative time value indicates the number of fractions of a second that have elapsed since the last absolute time value. For a changed frame, if more than one second has elapsed since the time of the last changed frame, an absolute timestamp is used. If the elapsed time is less than one second since the time of the last changed frame that had an absolute value, a relative timestamp is used. In one implementation, the timestamp is a 4-byte number, the relative timestamp is an integer value between 0 and 999 representing the number of milliseconds since the last absolute time value; the absolute timestamp is an integer value representing the number of seconds since a predefined absolute time point, such as 00:00:00 on Jan. 1st, 2000. Thus,



an absolute timestamp and a relative timestamp may be distinguished by whether or not its value is greater than 999.

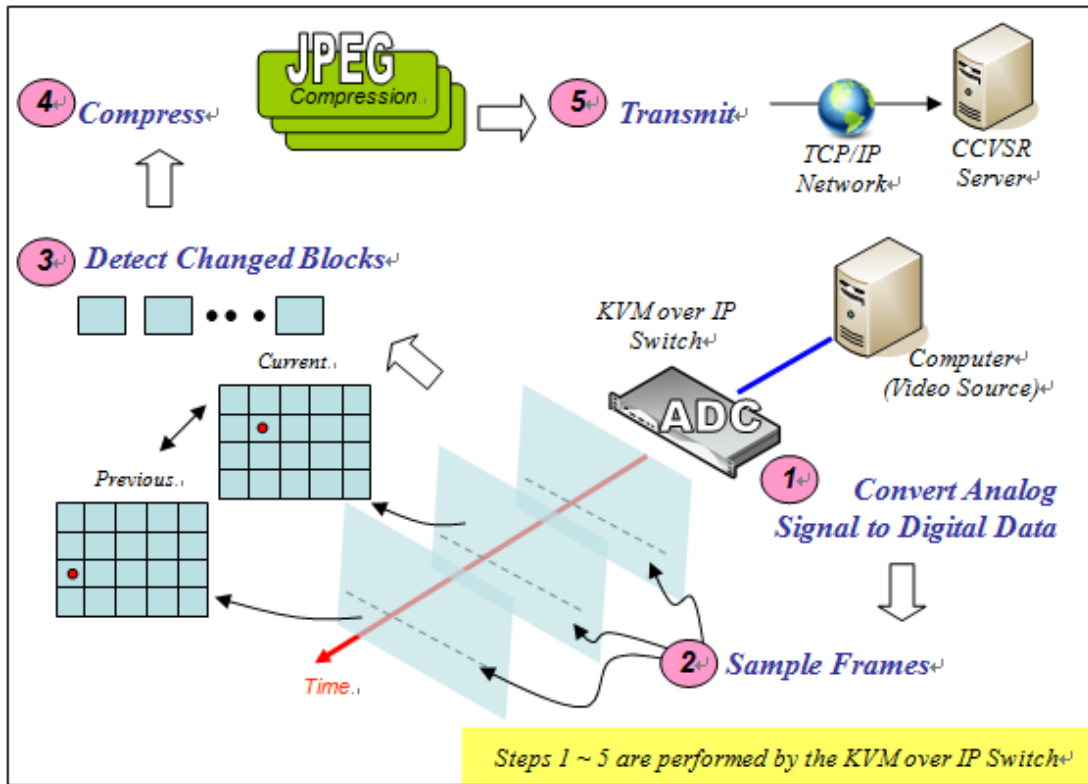


Figure 2 Video Capture Process

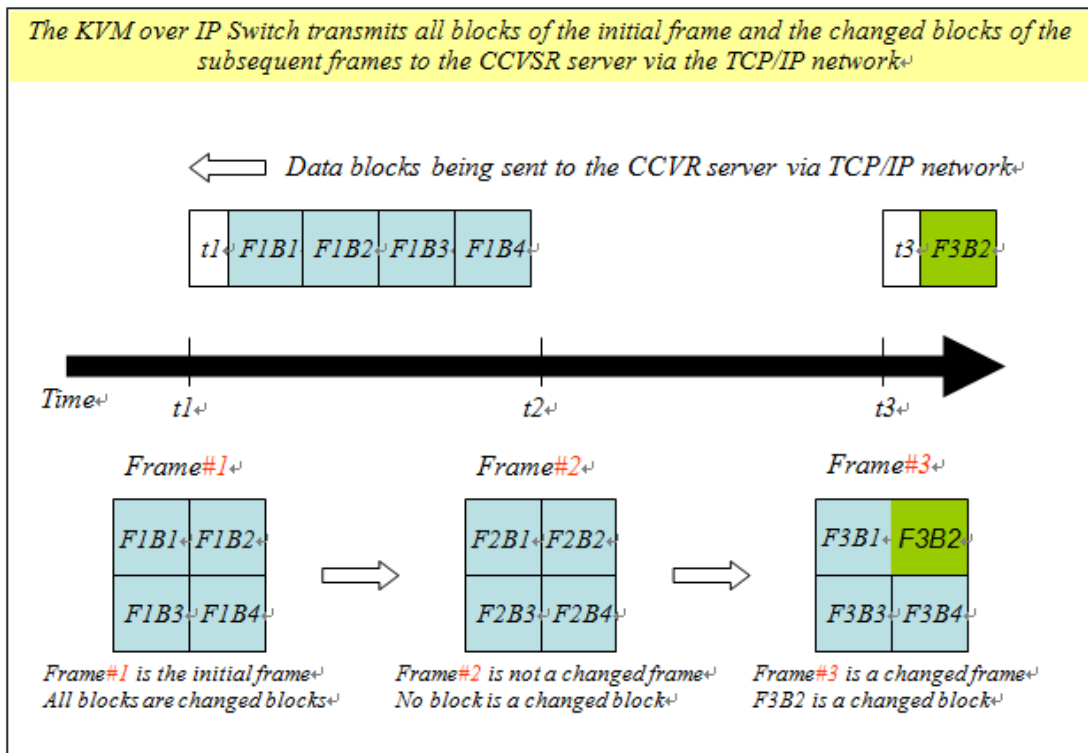


Figure 3 KVM over IP Switch Video Data Processing Method

An alternative to the timestamp implementation discussed above is to start each recording session with an initial timestamp indicating an absolute time for the start, with subsequent timestamps inserted at a fixed amount of time relative to the last transmitted frame.

## Video Storage Method

Captured video data is transmitted via the KVM over IP switch to the CCVSR server. The CCVSR server maintains a data file, a virtual frame table, and a snapshot file for each video session it records and stores, as illustrated in Figure 4.

The data file stores blocks of image data sequentially as they are received from the KVM over IP switch, and a block index that is stored with the block. Thus, at the beginning of the data file, all of the blocks that comprise the initial frame are stored. Thereafter, as the timestamps and the changed blocks of each of the changed frames are received from the KVM over IP switch, they are sequentially stored in the data file. During video recording, new data is continuously appended to the data file and no data gets overwritten.

For each block of a video frame (as represented by its block index), a virtual frame table stores the data location within the data file of where the block, or the most recent occurrence of that block, is stored. A data location within the data file may be represented by an offset from the beginning of the data file. The location of the beginning of a block is typically used as the data location of that block. Initially, the virtual frame table stores the data locations of the blocks contained in the initial frame. Thereafter, the virtual frame table is updated each time a changed block is recorded in the data file, and the new data location replaces or overwrites the original data location for that block in the virtual frame table.

The virtual frame table may be thought of as a virtual frame which is divided into virtual blocks in the same way as an actual frame. The difference is that each virtual block of a virtual frame is a reference to a data location in the data file, rather than the data itself. The virtual frame table is constantly updated and always stores the most recent location for each block that makes up the frame. During video recording, the virtual frame table is kept in the CCVSR server's memory.

A snapshot file stores copies, or takes a snapshot, of the virtual frame table at various intervals. A snapshot is a time point record – it represents a copy of the virtual frame table at a specific point in time, and is stored with a timestamp that indicates the absolute time of the time point (i.e., date, hour, minute and second), or a relative time (i.e., relative to the time of the initial frame, or relative to the time of the last snapshot). The time point snapshots are continuously appended to the snapshot file and no data gets overwritten. The history is preserved in the snapshot file at various intervals.

The virtual frame table may be discarded from the CCVSR's memory after the recording session ends, since it no longer serves any purpose. The data and snapshot files are saved, however, since they constitute a record of the video session, and the video images of a recorded session can be played back by using these two files. (Alternatively, these two files can be merged into a single file.) After the recording session ends, the CCVSR server adds a record of the start and end times in the non-video information associated with the port that the recording session occurred on. They can be used by a search and filter function to provide a list of playback selections for the user to choose from.

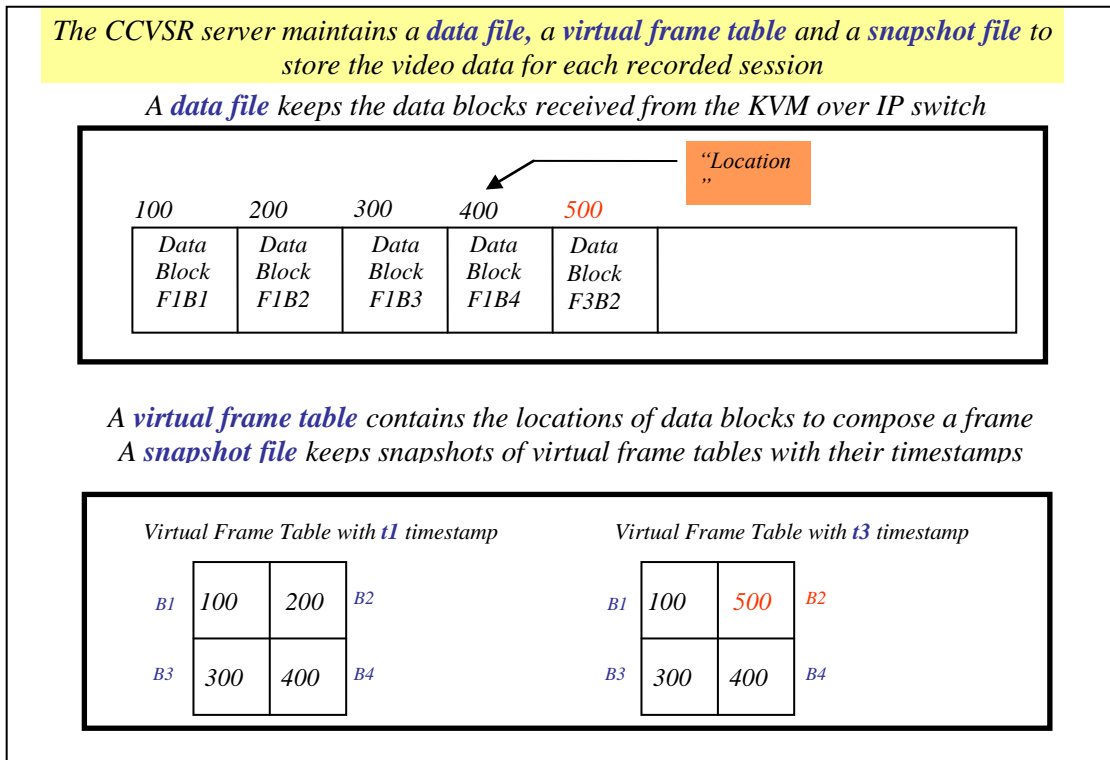


Figure 4 CCVSR Server Video Storage Method

## Video Reproduction and Playback

The procedure for reproducing the image frames from the stored video data uses software referred to as a player, and may be performed by the CCVSR server or by a client PC. The CCVSR server provides an interface that enables users to search the stored session records. Once the user selects a session record for playback, the CCVSR server begins to reproduce it on the player. If the player is being run on the client PC, the playback data is transmitted from the CCVSR server to the client PC.

Each selected session record has a snapshot file and a data file associated with it. The player reads the virtual frame table contained in the snapshot file to ascertain the data locations of all blocks making up the frame. Based on the virtual frame table information, it reads the appropriate data blocks from the data file. The player then processes each data block, which includes decompressing the data and assembling the decompressed blocks to form a video image frame. The player repeats these procedures to reconstruct the subsequent video frame, but it only reads the frames and blocks that have changed. In other words, once a frame has been reconstructed, it can be continuously updated by replacing only the changed blocks. As a result, recorded video images are successively played back from the start time point.

The snapshot and data files are inherently governed by the CCVSR server, and can not be externally controlled by the users. If the users would like to play back a session without operating under a CCVSR server session, the player can export the session record being played back to a separate password protected video file and player program. Users can copy the exported video file and player program to any client PC for subsequent playback.

## Conclusion

This whitepaper discusses the video data recording method implemented in ATEN's CCVSR solution. Each video image frame is divided into multiple blocks. The KVM over IP switch processes the video frames and only transmits the changed blocks for each frame to the CCVSR server. The CCVSR server stores the changed blocks (with block indexes), in a data file. The CCVSR server also maintains a virtual frame table which stores the data location for each block of the frame within the data file. The virtual frame table is continuously updated as changed blocks get stored in the data file. The CCVSR server further stores a snapshot file containing snapshots, along with their timestamps, of the virtual frame table. The data and snapshot files, which constitute the recorded video data that is available for playback, offer a video format that achieves high compression for certain types of video images; in particular, slow-changing video images or video images that remain unchanged for large portions of the image frames.

## About ATEN

ATEN is a global manufacturer and innovator of connectivity products spanning KVM (keyboard/video/mouse), data center management and professional A/V solutions. The company's comprehensive product line serves a diverse base of enterprise, government, SME and SOHO customers. Specifically, ATEN offers a full line of KVM switches, remote management products, LCD KVMs and consoles, power management appliances, data connectivity, extenders and audio video hardware.