

Digital Video Management Software (DVMS)

1. General Features

- 1.1. The DVMS software shall be designed and implemented as a client-server application suite which utilizes the Microsoft DirectShow architecture for video manipulation, including video capture, video compression and storage.
- 1.2. The DVMS software shall be designed using a camera abstraction layer. Video shall be captured in such a way as to provide for the seamless support of multiple, disparate video source technologies making the underlying video capture technology transparent to users of the DVMS and also allowing for the seamless addition of new capture technologies as they become available.
- 1.3. The DVMS shall support all of the following video source technologies in a single, unified software architecture:
 - 1.3.1. Direct capture of analog (NTSC or PAL) video using digital video capture devices that are internalized within a video management server;
 - 1.3.2. Indirect capture of analog (NTSC or PAL) video over an IP network through the use of analog-to-IP network video encoders;
 - 1.3.3. Direct capture of digital video from IP-based network video devices, from industry-leading vendors, such as Axis Communications, Sony, and IQInVision.
- 1.4. The DVMS shall provide for direct support of the following encoding and transport methods for IP-based video sources in such a way that the use of camera manufacturer-supplied COM application software interfaces such as ActiveX controls are not required.
- 1.5. Acquisition of multipart/x-mixed-replace JPEG encoded (MJPEG) video streams using HTTP for video transport and stream control;
 - 1.5.1. Acquisition of ISO-compliant MPEG-4 Part 2 (ISO/IEC 14496-2) encoded video streams using RTP (IETF RFC 3550) for video transport and RTSP (IETF RFC 2326) for stream control.
- 1.6. The DVMS shall provide enterprise-wide "single-seat administration" of all DVMS software components via an IP network, as follows:
 - 1.6.1. Support for push-based, secure distribution of application software updates for all DVMS server software components, where the update process occurs in parallel for all selected servers;
 - 1.6.2. Support for pull-based updates of video client application software via a centralized data store;
 - 1.6.3. Support for remote configuration of all DVMS server software components;
 - 1.6.4. Support for remote monitoring of all DVMS server software components;
 - 1.6.5. Support for remote configuration of the centralized data store.

2. Server Component

2.1. The server component shall be designed in such a way that:

2.1.1. The server component shall execute as a Windows “system service”, so that a Windows user is not required to be logged into the operating system in order to maintain full DVMS functionality.

2.1.2. The server component shall maintain full DVMS functionality regardless of the user rights of a locally logged-on Windows user;

2.1.3. A locally logged on Windows user with less than administrative rights shall not have the ability to stop, start or otherwise control the running state of the server component;

2.1.4. A Windows user with administrative rights shall be able to control access to Windows applications, application settings, operating system settings and other functions without compromising DVMS functionality.

2.2. The server component shall encode video utilizing one or more of the following video compression algorithms, configurable on a per-camera basis:

2.2.1. Microsoft MPEG-4 video codec. Implemented in such a manner that the resulting video files shall be able to be reviewed on any PC with the Microsoft Windows operating system, and a DirectShow-compatible media player such as the Microsoft Windows Media Player version 6.4 or higher.

2.2.2. Standard Motion-JPEG (MJPEG) video codec. Implemented in such a manner that the resulting video files shall be able to be reviewed on any PC with the Microsoft Windows operating system, Microsoft DirectX v9.0 or higher, and a DirectShow-compatible media player such as the Microsoft Windows Media Player version 6.4 or higher.

2.2.3. ISO-standard MPEG-4 video codec. Implemented in such a manner that the resulting video files shall be able to be reviewed on any PC with an ISO-compliant MPEG-4 video decompressor and a multimedia player application.

2.2.4. All vendors utilizing MPEG-4 video compression and decompression algorithms shall provide proof of patent licensing from the MPEG LA licensing authority.

2.3. The server component shall store video events as a user-accessible file within the NTFS file system, without requiring the user or administrator to extract the video event from an image database or other proprietary storage database for purposes of archive or review.

2.4. The server component's video recording architecture shall not record to a database structure so as to avoid the need to repair the database or loss of recorded video in the event of server power loss, unexpected application exit or drive failure.

2.5. The server component shall have the ability to associate multiple storage volumes to a camera device. The server shall have the ability to track recordings between volumes without user intervention. The server shall be compatible with any 3rd party backup software for recorded video archiving.

2.6. The server component shall be designed with a multi-threaded architecture to take advantage of multi processor computers or computers with multi core processors.

2.7. The server shall have the ability to record the original format from an IP camera source or transcode video from IP camera sources

2.7.1. The transcoding capability shall be able to encode video from Motion JPEG to MPEG4

2.8. The transcoding shall be able to resize video which is transmitted to client viewing applications. The client video application shall communicate the smallest required resolution to display video in a tiled view. The server component shall dynamically scale the video to the lowest size for display on the client to save bandwidth without affecting the recorded video quality.

2.9. The server component shall allow for third-party integration through the implementation of an application programming interface (API.) The API shall grant internal or third-party developers the ability to add the following video functionality to their applications, without the need for the DVMS client software to be installed or otherwise invoked:

- Display live camera views;
- Perform video archive search and retrieval functions;
- Control pan-tilt-zoom cameras;
- Add / Modify / Delete user accounts;
- Initiate recording of external alarm events; and
- Modify a subset of the server configuration.

2.10. The server component shall provide the ability to control one (1) or more pan-tilt-zoom cameras through the simultaneous use of one (1) or more of the following camera control protocols:

- American Dynamics ASCII;
- Axis (IP)
- Canon VCC-4;
- Kalatel;
- Pelco ASCII;
- Pelco D;
- Pelco P;
- Panasonic;
- Panasonic (IP);
- Philips;
- Samsung;
- Sensormatic;
- Sony VISCA;
- Sony VISCA (IP);
- Toshiba (IP);
- Ultrak (Diamond);
- Vicon.

3. Server Administration Console Component

3.1. The DVMS shall provide an “administration console” software component that is capable of attaching to one (1) or more DVMS systems, and reporting the following events for each DVMS that is being monitored:

- Installed DVMS software version;
- Total amount of system memory;
- Total amount of available system memory;
- Total CPU utilization;
- Total DVMS uptime;
- Video Source status;
- Current recording status;
- Volume where video events are currently being stored.
- Storage volume status;
- Total number of days of video that are currently available on the selected storage volume;
- Amount of storage that is currently being utilized;
- Amount of storage that remains before triggering the FIFO-mode video event deletion process.
- Provide a list of events that have occurred on the selected server since the initial connection of the administration console;
- Listing of currently connected clients, to include the following details:
 - Connection number;
 - Client (source) IP address;
 - Verbose description of the client;
 - The username utilized by the selected client.

3.2. The administration console shall automatically and without user intervention provide a process whereby critical system events shall be brought to the attention of the user in a visual manner.

3.3. The administration console shall provide the ability to remotely perform, in parallel, a push-mode, unattended software update to one (1) or more servers over an IP network. While updates are being performed, update events will be received and logged by the administration console.

3.4. The administration console shall remain connected to each defined server the length of the user session. If a connection is broken, the administration console will visually notify the user that the server has become unreachable.

3.5. The administration console shall provide for notification to one (1) or more administrator-configured e-mail addresses, if any of the following events occur, and if the originating server has been configured to provide such events:

- Connection Lost;
- Camera Sync Lost;
- Volume Offline;
- Hardware Key Missing.

3.6. The e-mail server configuration shall be of the following type and offer the following functionality:

- Utilize SMTP protocol over TCP port 25 to connect to an administrator-defined e-mail server;

Utilize SMTP Authentication if so configured by an administrator.

3.7. The administration console shall provide the administrator the ability to control the running state of the core video server software on the selected DVMS. Such abilities shall include:

Starting the video server system service;
Stopping the video server system service.

3.8. The administration console shall provide the administrator the ability to retrieve a filterable list of historical events from the log files maintained on each DVMS.

3.9. The administration console shall be capable of saving the current session with all user or administrator-configured settings to a file.

3.10. The administration console shall be able to restore a user session from the settings located within a saved session file.

4. Administration Service Component

4.1. The administration service software component shall be designed and implemented as a Windows system service and shall allow for out-of-band management of the core DVMS video server software, while providing the following functionality:

- 4.1.1. Ability to start and stop the core video server software on the DVMS on which it is installed;
- 4.1.2. Ability to perform software updates of the DVMS software;
- 4.1.3. Send informational event data to the connected administration console client.

5. Configuration Server Component

5.1. The configuration server software component shall be designed and implemented as a Windows system service while providing the following functionality:

- 5.1.1. Create and maintain an encrypted database of user accounts and passwords.
- 5.1.2. Associate an unlimited number of viewing layouts for each user account.
- 5.1.3. Allow properly configured client applications to connect to and authenticate against the internal user database.
- 5.1.4. Allow authenticated clients to retrieve the portion of the configuration specific to the authenticated user.
- 5.1.5. Allow authenticated clients to retrieve the product version number of the client software stored in the application directory in which the configuration server software component resides.
- 5.1.6. Allow authenticated administrators to add, modify and delete user and view data stored in the configuration server software component database.

6. System Configuration Client Component

6.1. The server and configuration server components shall be able to be configured either locally or remotely over an IP network through the use of a system configuration client component. The design of the system configuration client shall be such that whether the system being configured is local or remote it is transparent to the administrator.

6.2. The system configuration client shall allow the administrator to create an unlimited number of user accounts for each server.

6.3. The administrator shall be allowed to configure the following user-specific settings:

- Define whether the user account is allowed to configure the server;
- Define whether the user account is granted the ability to control server functionality through the use of an application programming interface (API);
- Define whether user account status events shall be entered into a log file;
- Control access to cameras on a per-camera basis, which shall include the following:
 - Grant / Deny access for live view;
 - Grant / Deny access for playback;
 - Grant / Deny access for the following PTZ-specific functionality:
 - The ability to show a stored preset, on a per-preset basis;
 - The ability to set a stored preset, on a per-preset basis;
 - The ability to control the PTZ through the use of a virtual joystick controller or a joystick controller that is attached to the users workstation.

6.4. The server configuration client shall grant to the administrator the ability to configure the following camera settings, on a per-camera basis:

- Specify a name for the camera input;
- Control whether or not the camera input is enabled;
- Enable the text insertion of the PC date and time into the captured video;
- Enable the text insertion of the camera name into the captured video;
- Define whether the camera connected to an analog camera input is color or black and white;

6.5. Control the following analog input camera image adjustments, with real-time previewing of any changes made by the administrator. The changes shall affect the hardware video decoder and shall therefore be global in nature:

- Contrast;
- Brightness;
- Hue;
- Saturation;
- Sharpness;
- Gamma;

6.6. The system configuration client shall allow the administrator to independently select the frame-per-second (FPS) recording rates for each of the available recording modes. The FPS selections shall be in one (1) FPS increments, from one (1) frame every sixty (60) seconds (1/60 FPS) to a maximum of thirty (30) frames per second (30 FPS.)

- 6.7. The system configuration client shall allow the administrator to configure video motion detection on a per-camera basis, and shall be the same for both analog and IP video types.
 - 6.7.1. VMD configuration shall be designed in such a way that the administrator will utilize a tool to add regions-of-interest to a still image of the selected camera input;
 - 6.7.2. The administrator shall be able to configure a sensitivity setting, where the lower the numerical value, the less sensitive to motion VMD becomes, and where the higher the value, the more sensitive to motion VMD becomes.
 - 6.7.3. VMD shall analyze the regions for objects in motion during the periods of time where VMD has been enabled.

- 6.8. The system configuration client shall allow the administrator to configure the automatic operation of an attached PTZ camera in the following manner:
 - 6.8.1. For each VMD region that the administrator configures on a camera, the administrator shall be able to define one (1) or more associated PTZ camera preset locations and a corresponding priority level. When VMD determines that motion is occurring in such a region, the DVMS will issue commands to the defined PTZ cameras that will cause the cameras to show the configured preset locations, given the following:
 - 6.8.1.1. If no other VMD region alarms are active, then the defined PTZ cameras will respond immediately;
 - 6.8.1.2. If a VMD region is currently in an alarmed state, the DVMS will compare the priority level of the new alarm with any currently active VMD region alarms. If the priority level of the new alarm is:
 - 6.8.1.3. Higher than the currently active alarm, the new alarm will take precedence and the PTZ camera will move to the defined preset location.
 - 6.8.1.4. Lower than the currently active alarm, the currently active alarm will maintain precedence and the PTZ camera will maintain its position.
 - 6.8.1.5. Equal to the currently active alarm, the DVMS will determine whether to automatically alternate between preset locations based upon the following criteria:
 - 6.8.1.6. If target cycling is enabled, the DVMS will cause the PTZ camera to alternate between preset locations as configured by priority level.
 - 6.8.1.7. If target cycling is disabled, the DVMS will cause the PTZ camera to show the preset location of the new alarm.

- 6.9. The system configuration client shall allow the administrator to configure pre- and post- alarm settings on a per-camera basis, where:
 - 6.9.1. Pre-alarm recording shall be configurable from between zero (0) and one-hundred and twenty (120) seconds, in increments of five (5) seconds.
 - 6.9.2. Post-alarm recording shall be configurable from between zero (0) and one-hundred and twenty (120) seconds, in increments of one (1) second.
 - 6.9.3. The FPS of the pre-alarm recording shall be at the same FPS as the triggering alarm event (VMD or External.)
 - 6.9.4. The pre-alarm, alarm, and post-alarm events shall be displayed to the user as a single, contiguous event.

- 6.10. The system configuration client shall allow the administrator to configure external alarm input and output settings in the following manner:

- 6.10.1. Each input shall be capable of being defined to trigger recording on one (1) or more camera inputs in the DVMS.
- 6.10.2. Each output shall be independently capable of generating an alarm state for one (1) or more of the following functions on a per-camera basis:

- VMD alarm active;
- Software-generated alarm active;
- Loss of video signal active.

- 6.11. The system configuration client shall include a recording schedule planner which will allow the administrator to create multiple recording schedules and allow for per-camera configuration of the recording plan within each schedule.

- 6.11.1. The recording planner shall allow the administrator to create the recording plan from among the following recording modes:

- Scheduled, continuous recording;
- Recording due to an external alarm;
- Recording due to VMD;
- Pre-alarm recording.

- 6.11.2. The recording schedule planner shall allow the administrator to create a recording schedule for a particular date; whether it is a weekday; whether it is a weekend; or a schedule to be executed every day of the week.

- 6.11.3. The server shall be designed in such a way that any additions, modifications or deletions that an administrator makes to a recording schedule will take effect within one (1) minute after the administrator has completed modifications of the schedule.

- 6.12. The system configuration client shall include a method whereby the administrator can define unlimited disk storage volumes for video recordings. The volume configuration shall include the ability to:

- 6.12.1. Define an unlimited number of storage volumes;
- 6.12.2. Define which camera inputs are allowed to record to the currently selected volume;
- 6.12.3. Specify either a local pathname (ex: C:\Video) or a universal naming convention (UNC) pathname (ex: \\DVR\Video) as the target for each defined storage volume.
- 6.12.4. Where a UNC pathname is utilized, the DVMS shall be capable of recording to the path without requiring a user be logged-on locally.

- 6.13. The DVMS will utilize the following user credentials when writing to a UNC path:

- 6.1.1.If the PC is configured for the 'workgroup' security model, the user credentials shall be the local SYSTEM user account.
- 6.1.2.If the PC is joined to a domain, the user credentials shall be the domain computer account.

7. Video Client Component

7.1. The video client component shall be designed in such a way as to allow an unlimited number of users to utilize the client, where:

7.1.1. A system configuration component shall allow the administrator to define an unlimited number of customized live viewing layouts for each defined user;

7.1.2. A local or centralized data store ("Configuration Server") shall contain the viewing layouts for each defined user;

7.1.3. Each position in a camera layout can be defined from any camera input on any server to which the user has been granted live viewing access;

7.1.4. Each position in a camera layout may be configured in such a way as to minimize network traffic through the ability to select the FPS of the video stream which is delivered to the client for live viewing;

7.1.5. Each position in a camera layout may be configured in such a way as to minimize network traffic through the ability to control the compression level of the video stream which is delivered to the client for live viewing.

7.2. The video client component shall be designed in such a way as to allow the use of customized maps for the authorized users of the client where;

7.2.1. A system configuration component shall allow the administrator to import any .jpg or .bmp image;

7.2.2. Each image may be configured to have a unique map name;

7.2.3. Each image shall be available in a preview window that shall display the image, provide image dimensions and the image (map) name;

7.2.4. The administrator shall have the ability to 'drag and drop' map icons (displayed as world icons) onto any image (map) that shall serve as a link to the represented map;

7.2.5. The administrator shall have the ability to 'drag and drop' video inputs (displayed as camera icons) on any server to which the user has been granted live viewing access, onto the image (map);

7.2.6. Each video input shall have a view cone that may be configured in such a way as to reflect the field of view of the video input;

7.2.7. Each video input shall display a thumbnail when the mouse hovers over the input;

7.2.8. Each view cone shall display the recording status and connection status of the camera when viewed in the video client;

7.2.9. The user shall have access to live view, search and quick review for each video input placed on the image (map);

7.3. The video client component shall connect to the centralized data store and authenticate the user connection. Once authenticated, the client component shall automatically retrieve the authenticated users customized viewing layouts from the centralized data store.

7.4. The video client component shall perform a version check when accessing the centralized data store. If a newer version of the client component exists at the centralized data store, then the user shall be prompted to update the client component in such a way as to not require the intervention of an administrator in order to accomplish the update.

7.5. The video client component shall be able to be configured so that any user account may have the following abilities:

7.5.1. Supervisor – a user account with the supervisor flag shall be allowed to configure viewing layouts for all user accounts in the local or centralized data stores.

7.5.2. Auto Full-screen – a user account with the “full-screen” flag shall, upon video client login, automatically display the default viewing layout in such a way so that the graphical user interface (GUI) is hidden and the live views are stretched to fill the user’s screen.

7.5.3. Auto-sequence – a user account with the “auto-sequence” flag set shall, upon video client login, automatically begin sequencing through viewing layouts according to how the viewing layout has been configured by the administrator.

7.6. The video client component shall be able to be configured to automatically login a user account without requiring user intervention.

7.7. The video client component shall offer the user the following functionality for viewing real-time video streams:

7.7.1. The user shall be able to initiate a mode whereby the video client component will cycle through the live viewing layouts created by the administrator, and where the inclusion of a layout in the cycle and the amount of time to dwell on a layout shall be able to be configured by the administrator in the system configuration component;

7.7.2. The user shall be able to change the viewing resolution of a particular camera view on a layout to 640 x 480 by performing a double-click of the mouse on the camera view, and where a double-click on the 640 x 480 camera view shall return the user to the original view layout;

7.7.3. The user shall be able to control a “full-screen” mode, where the live viewing layout fills the entire screen without displaying window borders, title bars, or other user interface objects.

7.8. The video client component shall offer the user the following functionality for reviewing archived event video:

7.8.1. The user shall be able to retrieve a list of events for which video has been recorded through the use of a search function. The search function shall allow the user to filter the recorded event video according to the following criteria:

- The server on which a camera input resides;
- The camera input, displayed to the user by the administrator-defined camera name;
- A date and time for the earliest point at which event video should be retrieved;
- A date and time for the latest point at which event video should be retrieved;
- Whether the event video is of the scheduled recording type;
- Whether the event video is due to an external alarm;
- Whether the event video is due to VMD.

7.8.2. The user shall be able to perform a “smart search” function which utilizes video motion detection on recorded video events, regardless of the mode of the original event. This VMD search function shall return a list of events for which motion has occurred in the user-specified areas of interest and shall encompass all of the filtering criteria of the standard search function.

7.9. The video client component shall offer the user the ability to, during playback of event video, take a single, still image of a video frame and shall:

7.9.1. Allow the user to zoom in and zoom out to a particular point of interest by utilizing the mouse;

7.9.2. Allow the user the ability to smooth the image to achieve greater detail;

7.9.3. Allow the user to save the resulting modified image in the Windows Bitmap (BMP) or Joint Photographers Experts Group (JPEG) file format.

7.9.4. Allow the user to print the image along with a report title and a note field, and where any alterations made to the original image are notated on the printout

7.10. The video client component shall offer the user the ability to export video events from the server in the standard Microsoft AVI file format, with the video compressed utilizing the same algorithm that it was initially compressed with: the Microsoft MPEG-4, ISO MPEG-4 or MJPEG video codecs. Additionally, the video export function shall:

7.10.1. Allow the user to save the video event as a file to a location on a local disk drive, or a network location to which the user has access;

7.10.2. Allow the user to directly create a CD-recordable or DVD-recordable disc which contains the video event file without requiring the user to perform intermediate steps outside of the client component;

7.10.3. Allow the user to add the video event to a queue, which shall allow the user to build a list of video events and that can be used at any point to create a CD-recordable or DVD-recordable disc.

7.11. The video client component shall offer the user the following functionality for controlling pan-tilt-zoom cameras to which the administrator has granted the user access:

7.11.1. Unconstrained direct control of the pan, tilt and zoom axes of a PTZ camera through the use of a virtual joystick, or through the optional use of a standard Windows-compatible external joystick.

The ability to show the preset locations stored in the PTZ camera;
The ability to store the preset locations in the PTZ camera;
The ability to open and close the iris.

7.12. The video client component shall offer the user the ability to display up to four independent video windows for the purpose of playback of time-synchronized video events. Each video window shall be selectable from any camera on any server to which the user has been granted access.

7.13. The video client component shall allow the user to enable or disable live, real-time monitoring of audio for any camera which has been configured by the administrator with an audio source.

7.14. The video client shall be designed in such a way as to allow for multiple, simultaneous instances of the application software to be loaded into system memory, where:

Each instance shall have its own, unique configuration;
Each instance shall operate independently of the others;

- 7.15. The size and position of the application window shall be stored so as to allow for data persistence, in such a way that when the user launches an instance of the video client component, the application window will reappear at same size and location as when the user last exited that particular application instance.

8. Web Client Component

- 8.1. The Web Client component shall include as a separate application a video client that is viewable using Internet Explorer 6.0 or greater.
- 8.2. The Web Client component shall include a royalty free, unlimited use license, ActiveX control.
- 8.3. The Web Client component shall support the live viewing of 1 to 16 video inputs, where the users has been granted access.
- 8.4. The server component of the web client shall be developed and integrated as part of the digital video management system, and shall not require the use of a 3rd party web server (such as Microsoft IIS)
- 8.5. The view layouts of the web client shall be populated with cameras using a drag and drop method of assigning cameras to views
- 8.6. The web client shall allow for control of Pan Tilt and Zoom cameras using both control buttons and keyboard shortcuts
- 8.7. The Web Client component shall offer playback from the video server.

9. Alarm Client Component

9.1. The Alarm Client component shall be designed in such a way that live video streams are automatically displayed on the users screen, when the following conditions are met:

The alarm client has been configured to monitor the alarm status of the particular camera;
The alarm event occurs during a time period for which the alarm client has been configured to monitor.

9.2. When a monitored alarm event occurs, as defined in 8.1, the event shall be added to a list of the most recent alarms, allowing for rapid recall of event video by the user.

9.3. When the user selects an event from the event listing, a new window shall appear containing both the current live video stream from the alarmed camera adjacent to the archived video footage associated with that particular alarm event.

9.4. The alarm client shall be designed in such a way as to allow for multiple, simultaneous instances of the application software to be loaded into system memory, where:

Each instance shall have its own, unique configuration;
Each instance shall operate independently of the others;

9.5. The size and position of the application window shall be stored so as to allow for data persistence, in such a way that when the user launches an instance of the alarm client component, the application window will reappear at same size and location as when the user last exited that particular application instance.