

Milestone Supervision with AppVision

29/08/2018 Rev 5

PRINCIPLE

The supervision of a Milestone video system enables the following features:

- Viewing cameras live images,
- Viewing cameras in playback,
- Support video multi stream
- Send PTZ command to cameras,
- Send preset command to cameras,
- Create/delete/update preset,
- Acquisition of video events (signal loss, detection),
- Acquisition of video analytic events,
- Acquisition of alarm events,
- Export video AVI or native format,
- Reconnect to another redundant server,
- Acquisition of PTZ position
- Command PTZ position in absolute

The AppVision interface has been validated with the Milestone SDK systems compatible with SDK MIP2018R3.

SETUP

Download Milestone components from our web site / member space:

MIPSDK_2018R3_Redist_Installer_x86.msi or MIPSDK_2018R3_Redist_Installer_x64.msi
AppPlayerMilestone.x.y.zip
AppDriverMilestone.x.y.zip
AppWizardMilestone.x.y.zip

On each AppVision client and server:

- Install Milestone redistributable
- Copy files of SDK Milestone Bin directory (*C:\Program Files\Milestone\MIPSDK\Bin*) in AppVision Bin directory (*C:\Program Files\AppVision 4.0\Bin*)
- Unzip AppPlayerMilestone, AppDriverMilestone et AppWizardMilestone in AppVision Bin directory
- [Run *imgViewerRegister.bat* in the Bin directory]

MILESTONE CONFIGURATION

In the case of the multi stream management by camera, it is necessary to order the streams of the cameras: stream high resolution first and low resolution last.

This will allow automatic switching between the high and low resolution streams depending on the size of the image.

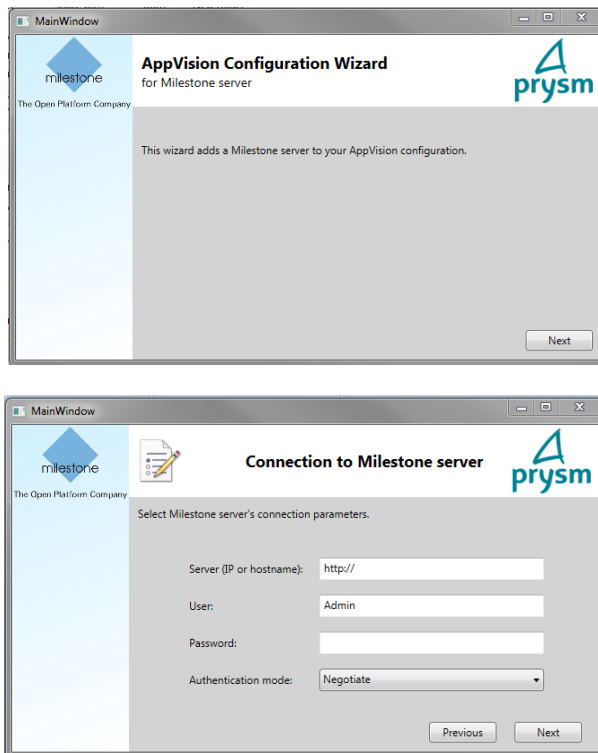
Milestone Supervision with AppVision

29/08/2018 Rev 5

APPVISION CONFIGURATION

To add a Milestone in the supervisor system configuration,

- Run Milestone wizard:
C:\Program Files\AppVision 4.0\Bin\AppWizardMilestone.exe



- Enter connection settings,
- Select cameras,

Automatically the wizard adds variables in the supervisor configuration, see the next section for details of variables.

PROTOCOL

A protocol allows to control multiple systems Milestone.

To activate Milestone protocol, add a protocol in AppVision configurator and select the Manager:
AppDriverMilestone.exe

Milestone Supervision with AppVision

29/08/2018 Rev 5

VARIABLES TREE

Milestone system is modeled by a set of variables organized in a tree:

{Milestone1}

	Parameters : see details below
.Comm	: server communication Type: Logical Value: 0/1/2/3 -> unknown/connected/disconnected/default
.Alarm	: alarm acquisition variable, see details below
.Cam001	: camera node Parameters: see details below
.Video	: video stream Type: Camera
.Comm	: camera communication Type: Logical Value: 0/1/2/3 -> unknown/connected/disconnected/default
.Recording	: camera recording status Type: Logical Value: 0/1 -> not recording / recording
.Event	: camera event Type: Enumeration Value: 0/1 -> end detection / start detection
.Analytic	: camera analytic event Type: Text See below
.CmdRecording	: camera recording command Type: Logical Value: 0/1 -> not recording / recording
.CmdPreset	: preset command Type: Text Value : preset name
.Position	: acquire camera position PTZ Type : Text Value : {pan};{tilt};{zoom}
.CmdPosition	: command camera position PTZ Type : Text Value : {pan};{tilt};{zoom}

Milestone Supervision with AppVision

29/08/2018 Rev 5

MILESTONE VARIABLE NODE

Source: protocol link,

for example: \$P.Milestone

Address: {username}:{password}@{ip_server_milestone}

for example: Admin:123@192.168.0.100

Parameters:

CREATECONTROLPLAYER=AppPlayerMilestone.dll

AUTHENTICATION={Negotiate/Basic/Windows}

CAMERA VARIABLE NODE

Parameters:

Name	Description
Shortcut	Parameter <i>Shortcut</i> is visible in the camera properties of Milestone Management Application, it is the property Shortcut N°
FQID	Parameter FQID is not necessary if the <i>Shortcut</i> parameter is set, this parameter is not visible in Milestone Management Application, it is automatically provided by the wizard AppVision Milestone
EnableDigitalZoom	true/false : enable digital zoom (false by default)
EnableVisibleHeader	true/false : header visible (false by default)
EnableVisibleCameraName	true/false : camera name visible (true by default)
EnableVisibleTimeStamp	true/false : indique playback time stamp visible (true by default)
MaintainImageAspectRatio	true/false : maintain image aspect ratio (true by default)
UsingMulticast	true/false : using multicast (false by default)
MOTIONDETECTIONACTIVE	true/false : if motion detection is monitored
TEMPO_POSITION	Temporization for position reading in millisecond (1000 by default)
POSITION_DEGRE	true/false : if true pan, tilt in ° from -180 to 180, if false from -1 to 1 (false by default)
ZOOM_MIN	Min value of zoom (0 by default)
ZOOM_MAX	Max value of zoom (0 by default)

Milestone Supervision with AppVision

29/08/2018 Rev 5

ANALYTIC EVENT VARIABLE

Name: {Camera_variable_name}.Analytic

Type: Text

Variable state contains the analytic message description.

Variable info contains the analytic detail: rule, objects detected.

This variable has state parameters:

- Source : analytic source
- Rule : analytic rule
- Object1 : 1st object with {object_name}={object_value}
- Object2 : ...

Sample of script server to generate an alarm on analytic event:

```
public class ScriptServerAnalytic : Prysm.AppVision.Script.ScriptServerBase
{
    public override void OnVariableStateChanged(VariableState vs)
    {
        switch( vs.Name )
        {
            case "ML.Camera1.Analytic":

                // test if your analytic system provide a Speed object
                for( int i=1;;i++)
                {
                    var objecti = vs.GetParameter("Object"+i);
                    if( string.IsNullOrEmpty(objecti) ) // last object?
                        break;
                    if( objecti.StartsWith("Speed=") )
                    {
                        double speed;
                        if( double.TryParse( objecti.Substring(6), out speed ) )
                            $V.ML.C_AXISM3044VNetworkCamera192168086Camera1.AlarmSpeed = (speed>100);
                        break;
                    }
                }
                // note: you can also test: vs.GetParameter("Rule")

                break;
            }
        }
    }
}
```

Milestone Supervision with AppVision

29/08/2018 Rev 5

ALARM VARIABLE

Name: {Milestone server variable nam}.Alarm

Type: Numeric or Enumeration

Variable state contains alarm state produced by Milestone system.

This variable has several state parameters:

- Source: source name
- Message: message
- ItemName: item name
- StateName: state name
- Priority: alarm priority
- RuleType: rule type
- RuleName: rule name

There is only one alarm variable to acquire all the alarms, so you have to dispatcher these alarms with a script server on other alarm variables according to the naming rules in the Milestone system.

For example if RuleName contains alarm id name:

```
public class ScriptServerAlarm : Prysm.AppVision.Script.ScriptServerBase
{
    public override void OnVariableStateChanged(VariableState st)
    {
        if( st.Name == "ML.Alarm" )
            switch( st.GetParameter("RuleName") )
            {
                case "GC Z001":
                    $V.GC.Z001 = st.Value.ToInt();
                    break;
                case "GC Z002":
                    $V.GC.Z002 = st.Value.ToInt();
                    break;
            }
    }
}
```

Milestone Supervision with AppVision

29/08/2018 Rev 5

CONFIGURATION OF REDUNDANT MILESTONE SERVER

The supervisor can support a redundant Milestone server.

In the case of a redundant Milestone system, there are 2 Milestone servers:

- A main Milestone server
- A secondary Milestone server

During a failed video stream call on the primary server, the stream is requested on the secondary server.

The secondary Milestone server address must be declared with the ADDRESS2 parameter of the Milestone server node variable with the same syntax as the primary server address, for example:

ADDRESS2 = Admin:[123@192.168.0.101](#)

ATTENTION: the ID of the same camera in the main Milestone server and the secondary Milestone server (FQID) is not the same, it is possible to declare the ID of the camera in the secondary Milestone server with the parameter FQID2 of the variable camera node.