

Camera functions

Purpose

The purpose of this script library is to add functions to get errors and statistics from connected cameras.

Prerequisites

Import the script library, *scExtensions.zip*, under script libraries in Ethis Admin.

Import the script library, *scCameras.zip*, under script libraries in Ethis Admin.

Requires Ethis 14 or later.

License requirements

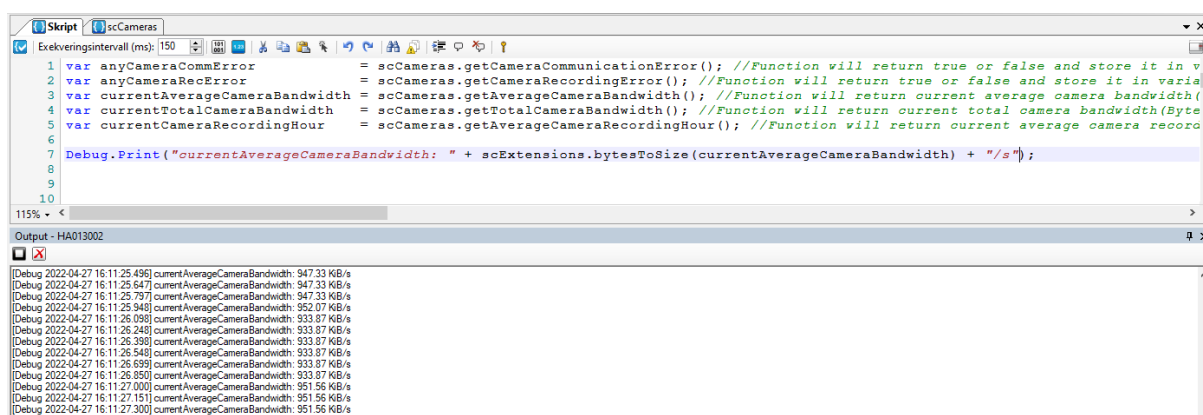
The solution requires the license function “Script libraries”. Check the Kentima license model on Kentimas website for further information.

Script library with explanation

This script library provides two methods for accessing data. Either manually, by calling specified functions in the library or event based, which will send events to the main script when specified status changes. For example if a bandwidth value goes above or below a certain setpoint.

Method 1, manually calling functions

In your main script you can add one or all of the following lines of code, found at the bottom of the script library *scCameras*, "Main script, Method 1".



```
1 var anyCameraCommError = scCameras.getCameraCommunicationError(); //Function will return true or false and store it in v
2 var anyCameraRecError = scCameras.getCameraRecordingError(); //Function will return true or false and store it in varia
3 var currentAverageCameraBandwidth = scCameras.getAverageCameraBandwidth(); //Function will return current average camera bandwidth(
4 var currentTotalCameraBandwidth = scCameras.getTotalCameraBandwidth(); //Function will return current total camera bandwidth(Byte
5 var currentCameraRecordingHour = scCameras.getAverageCameraRecordingHour(); //Function will return current average camera recora
6
7 Debug.Print("currentAverageCameraBandwidth: " + scExtensions.bytesToSize(currentAverageCameraBandwidth) + "/s");
8
9
10
```

Output - HA013002

```
[Debug 2022-04-27 16:11:25 496] currentAverageCameraBandwidth: 947.33 KB/s
[Debug 2022-04-27 16:11:25 647] currentAverageCameraBandwidth: 947.33 KB/s
[Debug 2022-04-27 16:11:25 797] currentAverageCameraBandwidth: 947.33 KB/s
[Debug 2022-04-27 16:11:25 948] currentAverageCameraBandwidth: 952.07 KB/s
[Debug 2022-04-27 16:11:26 098] currentAverageCameraBandwidth: 933.87 KB/s
[Debug 2022-04-27 16:11:26 248] currentAverageCameraBandwidth: 933.87 KB/s
[Debug 2022-04-27 16:11:26 398] currentAverageCameraBandwidth: 933.87 KB/s
[Debug 2022-04-27 16:11:26 548] currentAverageCameraBandwidth: 933.87 KB/s
[Debug 2022-04-27 16:11:26 699] currentAverageCameraBandwidth: 933.87 KB/s
[Debug 2022-04-27 16:11:26 850] currentAverageCameraBandwidth: 933.87 KB/s
[Debug 2022-04-27 16:11:27 000] currentAverageCameraBandwidth: 951.56 KB/s
[Debug 2022-04-27 16:11:27 151] currentAverageCameraBandwidth: 951.56 KB/s
[Debug 2022-04-27 16:11:27 300] currentAverageCameraBandwidth: 951.56 KB/s
```

And as you can see in the image above, you can use the function *Debug.Print()*, if you want to print any of the function results to the *Debug Output Panel*.

Method 2, Event driven method

The second method, which is a bit more advanced, can be used to create instances of “Check” functions. Which will return events to the main script(or other script libraries), based on the inputs to the functions. For example, you can monitor average camera bandwidth and get an event in your script if the value passes a certain threshold(limit). This means you don’t have to “poll” for new information from your main script.

The available “Check” functions(with corresponding inputs) are as follows.

scCameras.CommunicationErrorCheck(interval(Number), name(String))

scCameras.RecordingErrorCheck(interval(Number), name(String))

scCameras.AverageCameraBandwidthCheck(limit(Number), interval(Number), name(String))

scCameras.TotalCameraBandwidthCheck(limit(Number), interval(Number), name(String))

scCameras.AverageCameraRecordingHourCheck(limit(Number), interval(Number), name(String))

Explanation of inputs

Interval: How often script library function checks for a status change, in milliseconds. Default value is 10000, which equals 10 seconds.

Name: A name for the instance, can be used for example to trigger a specific alarm when the event arrives in the main script. Default value is ‘<NONAME>’.

Limit: Limit, in number of Bytes, which will trigger an event.

Default values

AverageCameraBandwidth: 5 MiB/s

TotalCameraBandwidth: 100 MiB/s

AverageCameraRecordingHour: 100 MiB

You can find example code at the bottom of the script library *scCameras*, “Main script, Method 2”.

Five instances of one of each of the following objects will be created. Events will be caught and printed to *Debug Output Panel*. Other examples how this can be used is to trigger an alarm, show values in Ethis Client or send to other system using OPC.

scCameras.CommunicationErrorCheck(), Will return an event if any or all cameras go offline.

scCameras.RecordingErrorCheck(), Will return an event if any or all cameras have an active recording error.

scCameras.AverageCameraBandwidthCheck(), Will return an event if average camera bandwidth go above/below a specified limit(Bytes/s).

scCameras.TotalCameraBandwidthCheck(), Will return an event if total camera bandwidth(based on average camera bandwidth) go over or under a specified limit(Bytes/s).

scCameras.AverageCameraRecordingHourCheck(), Will return an event if average camera recording for last hour goes above/below a specified limit(Bytes).

```
11 try
12 {
13     var init;
14     if(init == undefined)
15     {
16         init = true;
17         //var ce1 = new scCameras.CommunicationErrorCheck(10000, "ce1"); //interval, name
18         var ce1 = new scCameras.CommunicationErrorCheck(); //interval, name
19         ce1.getEventTarget().onDataAvailable = function(event)
20         {
21             try
22             {
23                 if(event.isCommError) // This will happen if there is any camera which currently has a communication error
24                 {
25                     Debug.Print("Global communication error is Active, meaning one or more cameras are offline.");
26                 }
27                 else if(!event.isCommError) // This will happen if there previously was an active communication error, but the error
28                 {
29                     Debug.Print("Global communication error is Inactive, meaning all cameras are online.");
30                 }
31             }
32             catch(e)
33             {
34                 Debug.Print("ce1.getEventTarget().onDataAvailable(): " + e);
35             }
36         }
37         //var rel = new scCameras.RecordingErrorCheck(10000, "rel"); //interval, name
38         var rel = new scCameras.RecordingErrorCheck(); //interval, name
39         rel.getEventTarget().onDataAvailable = function(event)
40         {
41             try
42             {
43                 if(event.isRecError) // This will happen if there is any camera which currently has a recording error
44                 {
45                     Debug.Print("Global communication error is Active, meaning one or more cameras are offline.");
46                 }
47             }
48             catch(e)
49             {
50                 Debug.Print("rel.getEventTarget().onDataAvailable(): " + e);
51             }
52         }
53     }
54 }
```

Output - HA013002

```
[Debug 2022-04-27 16:06:12.006] Created new instance of object 'scCameras.CommunicationErrorCheck()' name: <NONAME>, interval: 10 seconds
[Debug 2022-04-27 16:06:12.006] Created new instance of object 'scCameras.RecordingErrorCheck()' name: <NONAME>, interval: 10 seconds
[Debug 2022-04-27 16:06:12.007] Created new instance of object 'scCameras.AverageCameraBandwidthCheck()' name: <NONAME>, interval: 10 seconds, limit: 5.00 MB/s
[Debug 2022-04-27 16:06:12.007] Created new instance of object 'scCameras.TotalCameraBandwidthCheck()' name: <NONAME>, interval: 10 seconds, limit: 100.00 MB/s
[Debug 2022-04-27 16:06:12.008] Created new instance of object 'scCameras.AverageCameraRecordingHourCheck()' name: <NONAME>, interval: 10 seconds, limit: 100.00 MB
[Debug 2022-04-27 16:06:13.045] Average camera bandwidth is below limit of: 5.00 MB/s, current average bandwidth: 967.50 KB/s
[Debug 2022-04-27 16:06:13.047] Total camera bandwidth is below limit of: 100.00 MB/s, current total bandwidth: 12.28 MB/s
[Debug 2022-04-27 16:06:13.048] Average camera recording per hour, last week, is above limit of: 100.00 MB, current average recording: 270.84 MB
```