Notes on Sentinel Protocol
November 26, 2002
Robert D. Sexton
Sr. Software Engineer
Pelco

The comments and questions contained herein refer to draft version 2b of the Sentinel Protocol.

Query Command: This command can interrogate devices based on UUID and/or NetID. Is it correct to assume that a device should not respond to a Query with both UUID and NetID specified if either one does not match?

Yes, this is correct.

If the control PC sends a *Query* message with both *UUID* and *NetID* supplied as parameters, a camera device should respond with a *Query Result* message if – and only if – it matches these parameters with its own *UUID* **and** *NetID* properties

Further, if a device does not have a NetID assigned and receives a Query referring to its UUID, how should it respond, if at all?

This is a good point.

My current thinking is that if a camera device has no *NetID* (i.e. it has not processed an *Address Offer* message), it isn't part of the Sentinel network. The purpose of the *Query* message is to discover devices that are currently part of the network. So, in the case of a device that does not have a *NetID*, it should not respond to the said *Query* message.

(On power-up, I suggest that a camera device clears its *NetID* property before sending *Address Discover* messages)

Bye Bye Command: Is this message to be sent if the device does not have a NetID? That is, if it has not connected to the system?

The purpose of the *Bye Bye* message is to advertise that fact that a camera device is about to leave the Sentinel network. If a device does not have a *NetID*, it is deemed not to be part of the network. In this situation, therefore, a camera device should not send a *Bye Bye* message

Description: Is it correct to assume that the values for the properties (except for the UUID) are decimal numbers?

Yes, these are decimal numbers.

I've made them a special case because of the customer's involvement in specifying what these will be. And since customers think better decimally rather than hexadecimally, I've reflected this in the specification. However, hexadecimal encoding of the description could be the way to go in future

Configuration: As noted in the document, specifications are needed for the camera sensor properties. Is the aspect ration of the camera assumed for this document? A value for the pixel (sel) width of the camera is included, but not the height. Given the potential for different aspect ratios in the future, should not provision be made for pixel height?

No, the aspect ratio is not assumed.

It's just that the intrinsic calibration algorithms that may be used don't seem to make use of a pixel height (i.e. *Ncy*) constant.

However, I think you're right. Some provision for pixel height should be made. So, I will revise the specification to include this

Diagnostics: A bit flag array is specified as one of the properties. No bits are defined. Are these open to manufacturer definition? There is no mask bit specified to access the flags. Is this expected to be part of the next revision?

I've purposely **not** defined these flags. In the specification, I've suggested that these could be used to indicate things like communications errors, internal failures, and so on. So, I'll leave it up to you – and others – to suggest what the flags could be used for. Then, I'll then revise the specification

Reset Diagnostics: No specification for this command is included.

Telemetry

Goto Preset: This command specifies preset and speed. Spectra 3 has a fixed preset acquisition speed. This speed is selected for quickest acquisition of preset position. since the speed cannot be controlled externally, a Spectra 3 must ignore the speed specification. It must also be noted that a number of presets in Spectra 3 have special purpose designations. For example, preset 34 moves the camera to the pan home position (absolute 0º.)

This is fine.

Due to the generic nature of the protocol specified, there will always be some mapping necessary. Each manufacturer will need to map the protocol according to what their camera device can/can't do.

So, in the case of a *Goto Preset* message, simply ignore the specified speed and use the 'fixed acquisition speed' instead

Presets: The range of presets possible in a Sentinel command is larger than that of a Spectra 3. Should a unit respond with a General Error result code in the event it receives an out of range preset request?

The Sentinel preset range should be treated as a logical range that gets mapped to a physical range by the manufacturer.

Say, for example, you have a pool of 32 user/system-definable physical presets to play with.

If necessary, 8 of these would need to be reserved for things like auto-pan left limit, auto-pan right limit, auto-park, and so on. In the specification, these represent logical presets F8, F9, FA, FB..FF respectively. If a camera device then receives a *Set Preset* message for – say – preset F8 (auto-pan left limit), this preset would be identified (by a camera device) as a reserved preset and be mapped to an appropriate physical preset.

This then leaves 24 user/system-definable physical presets. In order for the control PC to know what presets it can use, it should be possible for the control PC to retrieve a preset range from a camera. This could either be done by:

- Extending the *Configuration* properties
- Introducing the concept of *Capabilities/Dynamics* properties

Once the preset range is known, the control PC would be careful to set and go to those presets only. These logical presets would have a one-to-one mapping with the physical presets, therefore

Consequently, the requirement to generate a *Telemetry Result* message indicating a general error can be avoided

Pan/Tilt Speed: Spectra 3 is capable of 64 speeds for pan and tilt operation and pan speed has a "Turbo" speed option. How shall the small range of speeds in Sentinel commands be mapped into Spectra 3 speeds? Also, should the values used to specify speeds be spread over the possible range (00 to FF rather than 00 to 04) in order to more effectively accommodate future enhancements to available speeds? This would make the change transparent and not require a future software revision. Then the Sentinel system may use as few speeds as desired while having all possible speeds available. A Spectra 3 (and for that matter, any similar PTZ camera) would simply map the speed request into its own range, assuming that FF is maximum and 00 is stop.

Well, I've integrated quite a number of different types of PTZ domes. And, I can honestly tell you that there has never been a requirement to offer the user anything other than slow, medium, fast, and very fast speeds.

Therefore, I recommend that you should consider sensibly mapping the logical Sentinel speed range across the range of physical speeds. So, for the Spectra 3, the following values (in hexadecimal) could be considered:

Stop (00) == 00
Slow (01) == 10
Medium (02) == 20
Fast (03) == 30
Very Fast (04) == 3F (or FF for 'turbo' speed)

Auto-pan: The command for auto-pan specifies a speed. Spectra 3 has an adjustable scan speed, but it is not an "on the fly" setting. Is it necessary to specify a speed for each command? Can there be a value that means, "scan at the current speed setting"?

In this case, simply ignore the auto-pan speed specified and always 'scan at the current speed setting'.

Alternatively, the specification could be revised such that a speed == FF represents 'scan at the current speed setting'. I'll have a think about this

Pan Tilt Zoom Focus: For pan and tilt, the commands are similar in operation (save for the range of values as discussed above) to the current method in Spectra 3. Zoom speed is not an "on the fly" setting, however. A menu command is available to select a speed for zoom operations but it is not dynamically variable. This is as for the auto-pan command. Thus, a speed value that indicates, "use current setting" would be very effective. The range of zoom speeds is very limited on Spectra 3 and this would make it more effective. The alternative is a "Set Dynamics" command to set properties such as this.

In this case, simply ignore the zoom speed specified and 'use the current setting'.

The specification could be revised such that a speed == FF represents 'use the current setting'. Again, I'll have a think about this

Extended

The extended commands specify motion in terms of microsteps. It is recommended that this term not be used, as it confuses the issue with actual motor control.

With regards to terminology, I accept your point about *microsteps*. I'll rename this to something more generic to avoid any confusion

Spectra 3 accepts position commands specified as actual angles in 0.01º increments. Thus, a pan position can be specified with an integer value of 0 – 35999 to specify 0º – 359.99º. If the proposed range in the Sentinel document were mapped to this range, it is possible that small positional inaccuracies would creep into the operation. This also keeps pan and tilt positions in the same units.

Okay, I'm fine with this.

Let's assume the pan position to be a 16-bit integer value in the range 0000..8C9F hexadecimal (i.e. 0º..359.99º). The tilt position will be a 16-bit integer value in the range 0..2327 hexadecimal (i.e. 0º..89.99º).

I reckon that this is generic enough for other types of PTZ camera device to follow.

Zoom position is also described in microsteps. Again, it is suggested that this term not be used to prevent confusion with motor control requirements. Spectra 3 is now capable of controlling the zoom position in hundredths of a unit. That is, a value of 500 results in a 5X zoom setting. Alternatively, a scaled value may be used to specify settings within the available range. It is recommended that the absolute magnification setting be accommodated as part of the Sentinel protocol.

I would like to keep the logical zoom range defined as 0000..03FF hexadecimal (i.e. 0..1023). In this case, a logical zoom value – specified in a *Pan Tilt Zoom Focus* message – should be sensibly mapped to an equivalent physical zoom value.

I'm not too hot on the ins and outs of zoom (and focus). So, can you give me some more details about 'absolute magnification', and how it would be used within the specification?

Focus position is not externally available from Spectra 3. Focus settings are retained as part of presets, however. Thus, the focus setting is guaranteed for preset operation but cannot be specifically controlled otherwise.

Okay, I'm fine with this at the moment. I guess you will need to ignore focus set/get operations

Perhaps the inclusion of *Get/Set Capabilities/Dynamics* messages would enable the control PC to determine whether focus could be controlled?