

XML Receiver Driver

01/10/2024 1:43 pm EST

Disclaimer

Since the Bold XML driver connects to a single receiver, no encryption is used. If deemed necessary, an encryption technology may be employed in the future.

Table of changes

Table 1 describes changes to the specification from the previous version.

Table 1 – Specification changes

Change Description
The Bold XML driver operates in both Serial and TCP mode.
The Bold XML driver can function as either the TCP host or client.
All communications (inbound and outbound) occur on the same port (serial or TCP). The driver does not split inbound and outbound messages between ports.
The Bold XML driver will not send an Error message for malformed messages.
The Bold XML driver will only NAK a malformed signal if in serial mode. Malformed TCP messages are forwarded as an invalid signal.
The four byte binary integer message preamble was dropped in favor of a beginning of message (BOM) and ending of message (EOM) character.
All XML tags are not case sensitive.
The DNIS attribute of the Packet tag was changed to SDNIS.
The MinsAgo tag was changed to MinAgo.
The DateStamp tag was changed to Date.
The EventID tag was changed to Tag.
Several attributes were changed to tags.
A Nak tag was added for the Response Message.

The Security tag was removed. Security information (User ID and Password) will be softly configured in Manitou and sent as parameters.
The DevId attribute of the Reverse tag was dropped and should be softly configured in Manitou as a parameter.
The ViewId attribute of the Reverse tag was dropped and should be softly configured in Manitou as a parameter.
Added new attributes to the Response tag for internal Manitou use.
Added new elements to provide address for GPS signal, plus Precision and unit.
Packet tag ID attribute length increased to 128.
Zone tag length increased to 12.
Corrected Date tag to say "customer date and time" instead of "UTC date and time".

Parameter Setup

Table 2 describes any parameters available for the driver and details their function.

Table 2 – Driver Parameters

Value	Description
HB	Indicates whether the driver should expect a heartbeat. If not, then the driver will issue a “receiver down” message if no signals are received after a certain period. • Default – 1 (True).
MIN	Number of minutes with no signal activity before the driver will send a “receiver down” message. This only applies if not using a heartbeat. • Default – 60
DEBUG	When true (DEBUG=1), the FEP will log data received from the receiver to a message file located on the root of the C drive. A maximum of 2 MB will be written per Manitou FEP Receiver before older messages are overwritten. • Default – 0 (False).

Communications

The Manitou XML Receiver Driver is capable of communicating through a serial connection or over an IP network.

XML Tags - All XML tags are case insensitive.

Beginning of Message - All messages begin with the ASCII Start of Text character, 0x02. Furthermore, all messages expect an XML version tag as the first tag of each message:

e.g., 0x02<?xml version="1.0"?>...

Message Content - All messages must be formatted in proper XML form. Additional white space (i.e., carriage return, line feed, spaces, and tab) for formatting is acceptable, but not required. XML examples in this document contain white space formatting for readability.

End of Message - Messages must include the closing tag for the outermost container (i.e., </Packet> for alarm messages, </Reverse> for reverse commands, and </Response> for reverse command responses, etc.). The ASCII End of Text character, 0x03, completes all messages.

e.g., ...</Packet>0x03

Acknowledgments - The driver will acknowledge (ACK) messages with an XML string once successfully processed.

0x02<?xml version="1.0"?><Ack/>0x03

Negative Acknowledgements - The driver will only send a negative acknowledgment (NAK) when using serial communications. A maximum of two NAKs will be sent for a given message. If a malformed message is presented a third time, the driver will ACK the message and forward it as an invalid message.

Since TCP/IP guarantees receipt of what was sent, the driver will not NAK malformed messages received over this protocol. Rather, it will ACK and forward them into the Manitou system as an invalid message.

The NAK string is:

0x02<?xml version="1.0"?><Nak/>0x03

Heartbeat - If the heartbeat is enabled, the heartbeat message is expected from the receiver at a regular interval in absence of an actual signal message or reverse command response. If no messages are received for a period of 65 seconds, the driver will send a “receiver down” alarm to the Manitou system. The heartbeat signal is:

0x02<?xml version="1.0"?><Heartbeat/>0x03

If the heartbeat is not enabled, the driver will send a “receiver down” message if no signal activity is received for a configurable number of minutes (60 by default).

Protocols Supported

The XML receiver driver supports several message protocols. Message protocols are reported as the EvType attribute of the Signal Tag. The supported protocols are:

- “SIA” – Any standard SIA codes, such as BA, BR, FA, FR, etc.
- “CID” – Any standard Contact ID codes, such as E101, R101, E103, etc.
- “SYS” – Manitou standard system codes. Current valid system codes are listed in Table 3 (subject to modification).
- <Other> – Any Transmitter Protocol Type configured in Manitou which correlates to Event Map entries. Transmitter Protocol Types are limited to 8 characters.

Table 3 – Manitou Standard Codes

Code	Description	Code	Description
*1	GSM Link Fail	*L	Low Battery
*2	GSM No Response	*LC	Late-To-Close
*3	Land Line Link Fail	*LG	Late-To-Checkin
*4	Land Line No Response	*LO	Late-To-Open
*5	GSM Resp OK	*LS	Late-To-Start
*6	Land Line Resp OK	*LT	Late-To-Test
*7	GSM Remote Link Fail	*M	Message
*8	GSM Remote Resp OK	*MD	Missing Dual Signal
*A	Activation	*N	Cancel
*A1	Unknown Card	*NB	No Battery
*A2	Unassigned Card	*O	Open
*A3	Unauthorized Access	*OE	Early Open
*AX	Unexpected Area	*OF	Off-line
*B	Bypass	*OL	Late Open
*BA	Burglary Alarm	*ON	On-line
*C	Close	*OR	Re-Open

*CA	Call Attempts	*OU	Unscheduled Open
*CE	Early Close	*OX	Unexpected Open
*CI	Caller ID	*P	Force Arm
*CL	Late Close	*PA	Panic Alarm
*CU	Unscheduled Close	*Q	Unauthorized
*CX	Unexpected Close	*R	Restore
*D	Door Access	*RB	Bell Restore
*DF	Device Test Fail	*RO	Restore Overdue
*DO	Device Test Okay	*RX	Unexpected Restore
*DT	Device Fault	*S	Supervisory
*E	Test	*SE	Service End
*EM	Equipment Message	*SS	Service Start
*ER	Error	*ST	Status
*F	AC Loss	*T	Trouble
*FA	Fire Alarm	*TA	Trap Account
*FO	Foreign Account	*TB	Bell Trouble
*FR	Fire Restore	*TP	Tamper
*FT	Fire Trouble	*TT	Two-trip Ignore
*FX	Unexpected Fire Test	*U	Unbypass
*G	Battery OK	*V	Alarm with Audio
*H	Duress	*W	Runaway Warning
*I	User Number	*X	Auxiliary
*ID	User ID	*Y	System Restore
*J	Trouble Restore	*Z	System Alarm
*K	AC Restore		

Message Structure

Packet (Alarm) Message (Inbound)

All Alarm messages are contained within the Packet tag, and will contain one to many Signal, Telemetry or GPS tags. Table 4 contains a list of acceptable XML tags with their respective attributes.

Table 4 – Alarm Message XML Tags

Tag	Parent	Attribute	Required	Length	Description
RawData	None		No	Variable	Non-decoded tag to include the raw data string (i.e., original message coded to XML for delivery to the FEP). Example: <?xml version="1.0"?><RawData>[#9090 NMA01] </RawData><Packet ID="9090"><Signal EvType="SIA" Event="MA"><Zone>01</Zone></Signal></Packet>
Packet	None		Yes		Outer most tag, defining an alarm message.
		ID	Yes	1-128	Account Number (up to 32 characters) or Transmitter ID. In either case, the length cannot exceed the length of the field in the version of Manitou to which the signal is being sent.
		SDNIS	No	1-20	Pseudo-DNIS. Provides a way to logically group customers on the same receiver.
		Line	No	1-5	Virtual line number (decimal). Valid range 1-65534.
		CallerId	No	0-20	Caller ID digits. Mutually exclusive with “Emergency” attribute.
		Emergency	No	0-20	Caller ID digits for an emergency 911 lookup (i.e., elevator alarm). Mutually exclusive with “CallerId” attribute. This message should be stand alone, containing no Signal, Telemetry or GPS tags.

Signal	Packet		No		Single signal block. A Packet may have multiple Signal, Telemetry or GPS blocks, but at least one must be present.
		EvType	Yes	8	Eight character mnemonic signifying the Event Type. See Protocols Supported for a detailed description of possible values.
		Event	Yes	1-12	Event Code.
Telemetry	Packet		No		Single telemetry block. A Packet may have multiple Signal, Telemetry or GPS blocks, but at least one must be present.
		EvType	Yes	8	Eight character mnemonic signifying the Event Type. See Protocols Supported for a detailed description of possible values.
		Event	Yes	1-12	Event Code.
GPS	Packet		No		Single Global Positioning System (GPS) block. A Packet may have multiple Signal, Telemetry or GPS blocks, but at least one must be present.
		EvType	Yes	8	Eight character mnemonic signifying the Event Type. See Protocols Supported for a detailed description of possible values.
		Event	Yes	1-12	Event Code.
ListenIn	Packet		No		Listen-In notification. Note: CallerId and SDNIS or Line attributes are required in the Packet parent.
Area	Signal		No	0-4	Area value.
AreaInfo	Signal		No	0-35	Area text description.
Zone	Signal or GPS		No	0-12	Zone value. Note that the length cannot be greater than 5 if the version of Manitou to which the signal is being sent supports only 5 characters.
Sensor	Signal or GPS		No	0-12	Sensor value.
User	Signal		No	0-12	User Number.

UserInfo	Signal		No	0-50	User text description.
PointID	Signal		No	0-40	Point ID information.
MinAgo	Signal or Telemetry		No		Number of minutes ago the event occurred (exclusive to Date). Note: Only include if the event occurred in the past.
Date	Signal or Telemetry		No		Customer date and time the signal occurred, in MM/DD/YYYY HH:MM:SS format (exclusive to MinAgo). Hours are in 24 hour format. Note: Only include if the event occurred in the past.
Tag	Signal or Telemetry		No		Provides a mechanism to tie back into an event on the receiver (i.e., unique identifier for a video file).
Unit	Telemetry		No	0-25	Units which applies to this telemetry data (i.e., degrees celcius).
Value	Telemetry		No	0-14	Actual data value.
Latitude1	GPS		Yes		Latitude value.
Longitude1	GPS		Yes		Longitude value.
Direction2	Latitude or Longitude		No	N, S, E, W	Direction (N/S for Latitude, E/W for Longitude).
Degrees2	Latitude or Longitude		No	0-180	Number of degrees (0-90 for Latitude, 0-180 for Longitude).
Minutes2	Latitude or Longitude		No	0-59	Number of minutes.
Seconds2	Latitude or Longitude		No	0-59	Number of seconds.
Tenths2	Latitude or Longitude		No	0-9	Number of tenths.
Speed	GPS		No		Current speed.
Heading	GPS		No	0-360	Heading degrees.

Power	GPS		No	0-100	Battery power level (percent).
UnitOfMeasure	GPS		No		0 = Meters, 1 = Yards
Precision	GPS		No	0-40	Using the <UnitOfMeasure> how precise is the reading.
Comment	GPS		No	0-40	Comment/Point ID Text.
URL	Signal or GPS		No	0-2048	URL Address (currently limited to one per Signal or GPS). Example: <URL> https://www.???.com/ProprietaryMapSolution.aspx?MapID=1234&MapMode=1 </URL>;
URLInfo	Signal or GPS		No	0-50	URL Address text description (currently limited to one per Signal or GPS). Note: Only valid if URL given as well.
Address	GPS		No	0-50	Nearest Address Line corresponding to the GPS location.
City	GPS		No	0-50	City for Address corresponding to the GPS location.
Region	GPS		No	0-50	Region for Address corresponding to the GPS location.
Postcode	GPS		No	0-16	Postcode for Address corresponding to the GPS location.
Country	GPS		No	0-20	Country for Address corresponding to the GPS location.
CrossStreet	GPS		No	0-50	Nearest Cross Street to the GPS location.

¹Latitude and Longitude may be provided in one of three formats:

1. As a floating point decimal (i.e., 68.4719).
2. As a string value (i.e., "N68.28.43,9" – punctuation is required as noted).
3. Broken out into its individual components: direction, degrees, minutes, seconds, and tenths.

²When providing latitude/longitude as individual components, all five components must be present.

Example: Three signals received as part of a single packet: activation, restore, and an open.

1

```
0x02<?xml version="1.0"?><Packet ID="527" Line="4"><Signal EvType="SYS" Event="*A"><Area>1</Area><Zone>1</Zone><Date>05/06/2005 15:30:23</Date></Signal><Signal EvType="SYS" Event="*R"><Area>1</Area><Zone>1</Zone><Date>05/06/2005 15:31:00</Date></Signal><Signal EvType="SYS" Event="*OP"><Area>1</Area><User>2</User><UserInfo>Rod Coles</UserInfo><Date>05/06/2005 15:31:00</Date></Signal></Packet>0x03
```

Example: Telemetry message received for freeze trouble providing temperature data.

1

```
0x02<?xml version="1.0"?><Packet ID="527"><Telemetry EvType="SIA" Event="ZT"><Unit>Degrees Celcius</Unit><Sensor>12</Sensor><Value>20</Value></Telemetry></Packet>0x03
```

Example: GPS message received.

1

```
0x02<?xml version="1.0"?><Packet ID="527"><GPS EvType="SYS" Event="*M"><Latitude><Direction>N</Direction><Degrees>68</Degrees><Minutes>28</Minutes><Seconds>43</Seconds><Tenths>9</Tenths></Latitude><Longitude><Direction>W</Direction><Degrees>14</Degrees><Minutes>27</Minutes><Seconds>32</Seconds><Tenths>7</Tenths></Longitude><Speed>59 mph</Speed><Heading>192 deg</Heading><Power>78%</Power></GPS></Packet>0x03
```

Or

1

```
0x02<?xml version="1.0"?><Packet ID="527"><GPS EvType="SYS" Event="*M"><Latitude>N68.28.43.9</Latitude><Longitude>W14.27.32.7</Longitude><Speed>59 mph</Speed><Heading>192 deg</Heading><Power>78%</Power></GPS></Packet>0x03
```

Example: Sending Address information with GPS

1

```
0x02<?xml version="1.0"?><Packet ID="527"><GPS EvType="SYS" Event="*M"><Latitude>N68.28.43.9</Latitude><Longitude>W14.27.32.7</Longitude><Speed>59 mph</Speed><Heading>192 deg</Heading><Power>78%</Power><Address>15380 Black Forest Road</Address><City>Colorado Springs</City><Region>CO</Region><Postcode>80908</Postcode><Country>US</Country><CrossStreet>Shoup Road & Black Forest Road</CrossStreet></GPS></Packet>0x03
```

Example: Emergency 911 caller ID message received.

1

```
0x02<?xml version="1.0"?><Packet ID="7195932829" Emergency="7195932829" SDNIS="9891"></Packet>0x03
```

Example: Listen-In message received.

1

```
0x02<?xml version="1.0"?><Packet ID="7195932829" CallerId="7195932829" SDNIS="9891"><ListenIn/></Packet>0x03
```

Audio and Video (Binary) Data

The Bold XML driver will accept audio and video messages (binary data). The Binary tag may only be contained by the <Packet> tag. Due to the nature of binary data, encoding the data is required. The Bold XML driver utilizes the Base 64 encoding for binary data, using the character set 'A' – 'Z', 'a' – 'z', '0' – '9', '+', and '/'. The padding character is '='.

The sender may break the audio/video data into multiple blocks (Data tags) if desired. Manitou will concatenate all Binary Data blocks together within each <Binary> tag to form a single, complete audio or video unit. Please note, only one <Binary> block is supported per <Packet>.

Table 5 - Binary XML Tag

Tag	Parent	Attribute	Required	Length	Description
Binary	Packet		No		Binary data for audio or video.
		Type	Yes	1 to 4 numeric digits in the range of 1 - 9999	Indicates the type of audio/video file, where: - General: 1 = Text 2 = HTML 3 = XML 4 = URL - Video: 1000 = Ozvision OZC 1001 = Redcare NGR 1002 = RSI MPG 1003 = RSI JPG 1004 = MPEG 1005 = AVI 1006 = MOV, MP4 1007 = WMV 1008 = JPG 1009 = March R5 1010 = Honeywell Rapid-Eye Multimedia 1011 = Bosch Videojet MPG 1012 = Geuterbrueck GVI

					◦ 1013 = ExacqVision AVI ◦ 1014 = Axis ASF ◦ 1015 = Domonial ◦ 1016 = FLV • Audio: ◦ 5000 = WAV ◦ 5001 = MP3 ◦ 5002 = WMA • Other: ◦ 9999 = other audio/video type
		Ext	Yes, if type = 9999; otherwise, no	1-6	File extension (i.e., "M4A") for other (un-integrated into Manitou) audio/video types. This extension is provided to the operating system to determine an appropriate application to play the clip.
		DataType	Yes, if type = 9999; otherwise, no	1	Identifies the binary data as Audio or Video when the "other" type (9999) is used: • V = Video • A = Audio
		Length	Yes	1-10	Total length, in bytes, of the un-encoded binary data contained within this <Binary> tag.
Data	Binary		Yes (at least one)		Binary audio/video data encoded in Base64 using character set 'A'-'Z', 'a'-'z', '0'-'9', '+' and '/'. The padding character is '='.
		Length	Yes	1-10	Length, in bytes, of the Base64 encoded binary data contained within this <Data> tag. Note: this number excludes the line feed and carriage return characters.

Each <Binary> and <Data> tag must include a Length attribute. For the binary tag, the length represents the total number un-encoded of binary bytes (the original file size). Regarding data tags, the length is the number of encoded base 64 bytes contained by the tag.

The Length attributes for <Binary> and <Data> tags must be accurate. A known issue exists in the 1.4.9-2 Bold XML driver where the driver may become “trapped” for an extended period of time (hours or more). To illustrate, examine the following message containing an MP3 file. The MP3 file contains a total of 5,159,592 bytes as defined in the <Binary> tag which is encoded into 6,879,456 base 64 bytes:

1

```
<?xml version="1.0"?><Packet ID="527"><Signal EvType="SIA" Event="PA"><Zone>1</Zone></Signal><Binary Type="5001" Ext="mp3" DataType="A" Length="5159592"><Data Length="6879456">SUQzAwAAAAAhdIRJVDIAAAATAAAATGlmZSBMZnZlE9yZGluYXJ5VFIFUgAAAAUAAAyMDA0UFJJVgAAACcAAAFdNL01IZGhhQ2xhc3NQcmItYXJ5SUQAvH1g0SPj4kuGoUikKihEHIBSSVYAAAApAABXTS9NZWRpYUNsYXNzU2Vjb25kYXJ5SUQAAAAAAAAAAAAAAAAAAAAAFBSSVYAAAAfAABXTS9XTUNvbnRlbnRjRAAEDrx7w5QCRqFDnm/5fjoUFJJVgAAAlAAAFdNL1VuaXF1ZUZpbGVJZGVudGlma...</Data></Binary></Packet>
```

To achieve enhanced performance, the Bold XML driver must perform a special buffered data read. This is accomplished by specifying the number of base 64 bytes to read at a time. As an example, if the driver is using a 100 kb buffer, it will read 100 kb at a time until the total length of base 64 bytes are read. If, during a read, only 2 kb are available (i.e., the receiver had a failure and aborted the send), the driver will continue to read until the complete 100 kb buffer is filled. If this situation occurs, the driver is essentially neutralized for this period of time without any notifications being raised.

This “trap” issue is addressed in the driver beginning with version 1.5.0 such that it will continue a read until the total number of bytes instructed are read (as in 1.4.9-2) or until a three second or longer period with no bytes read occurs. Please note, it is still possible for the 1.5.0+ driver to become “trapped” if a trickle of bytes is coming in at less than a three second interval (i.e., a two second heartbeat is produced by the receiver).

Additional sample:

1

```
<?xml version="1.0"?><Packet ID="527"><Signal EvType="SIA" Event="MA"><Zone>1</Zone></Signal><Binary Type="5001" Ext="mp3" DataType="A" Length="10881858"><Data Length="14509144">SUQzAwAAAAAhdIRJVDIAAAIAAAAVGhIIFNIYVRZRVIAAAAFAAAAMjAwNFBSSVYAAAAAnAABXTS9NZWRpYUNsYXNzUHJpbWVyeUUEALx9YNEj4+JLhqFlpCooRB5QUklWAAAAKQAAV00vTWVkaWFDZGFzc1NIY29uZGFyeUUEAAAAAAAAAAAAAAAAAAAAABQUklWAAAAHwAAV00vV01Db250ZW50SUQA07EGL/NgsE6gtvLbTjxo1BSSVYAAACKAABXTS9VbmldWVGaWxISWRlbnRpbmllcgBBAE0ARwB...</Data></Binary></Packet>
```

Reverse Channel Message (Outbound)

The Bold XML receiver driver also allows for commands to be sent back to the receiver. The actual command values and parameters are not pre-defined - rather they can be softly defined in Manitou. User ID and password validation are also sent as parameters. Table 6 displays a list of supported tags.

Table 6 – Reverse Channel Message XML Tags

Tag	Parent	Attribute	Required	Length	Description
Reverse	None		Yes		Outermost tag, defining a reverse command.
		Reverseld	Yes		Unique integer identifier within the Bold XML driver so responses from the receiver can be matched up with their respective request.
		Name	Yes	1-35	Reverse command name or mnemonic (defined softly in Manitou).
Parameter1	Reverse		No		First parameter value required for the specified command (defined softly in Manitou).
Parameter2	Reverse		No		Second parameter value required for the specified command (defined softly in Manitou).
ParameterN	Reverse		No		Nth parameter value required for the specified command (defined softly in Manitou).

Example: Possible format for the device located on bus 1, port 16 to record for 120 seconds. Parameters 1 and 2 provide the user ID and password, respectively.

1

```
0x02<?xml version="1.0"?><Reverse Reverseld="507" Name="RECD"><Parameter1>Bold</Parameter1><Parameter2>Pass</Parameter2><Parameter3>1</Parameter3><Parameter4>16</Parameter4><Parameter5>120</Parameter5><Parameter6>Seconds</Parameter6></Reverse>0x03
```

Response Message (Inbound)

Response messages are replies from the receiver to requests made via reverse channel messages, and will contain one Response tag. Table 7 displays a list of supported tags.

Table 7 – Response Message XML Tags

Tag	Parent	Attribute	Required	Length	Description
Response	None		Yes		Outermost tag, defining a reverse command response.
		Reverseld	Yes		Unique Integer identifier within the Bold XML driver so it can match responses with their respective request (echoed back).

		Final	No		Indicates if the response is the final message (the request is complete). Valid values are: • Y = Completed • N = Still processing If omitted, then Final is assumed as “N” (exception, Nak [see below]).
		Status	No		Reports the overall outcome of the request. Valid values are: • 0 = Success • 1 = Failed If successful, requested data, if any, should be included in the HTML/Link/Text/XML tag(s). If failed, error messages, if any, should be included in the HTML/Link/Text/XML tag(s).
		Line	No	1-5	Virtual line number (decimal). Valid range 1-65534.
		Type	No		Retransmission response type.
		RetransId	No		Retransmission unique Id.

The following tags are mutually exclusive. One, and only one, should be present in the response message.

Ack	Response		No	Indicates the receiver has accepted the request for processing.
Nak	Response		No	Indicates the receiver is refusing to perform the request (i.e., the request was malformed, invalid parameter(s), missing parameter(s), etc.).
	Nak	ErrNo	No	Error number.
HTML	Response		No	Response data formatted in HTML (can include links).
Link	Response		No	URL link.
Text	Response		No	Response data formatted as ASCII text.
XML	Response		No	Response data formatted in XML.

Example: Reverse request acknowledgment, further messages to follow.

1

```
0x02<?xml version="1.0"?><Response Reverseld="1029" Final="N"><Ack/></Response>0x03
```

NOTE: Optional text is not supported in with the Ack message. If the request is completed upon the Ack but optional text is desired, send two response messages: an Ack (Final="N") and an HTML/Link/Text/XML (Final="Y").

Example: reverse request acknowledgment, no further messages to follow.

1

```
0x02<?xml version="1.0"?><Response Reverseld="1029" Final="Y"><Ack/></Response>0x03
```

Example: possible response to a pan/tilt/zoom request (multiple formats illustrated).

1

```
0x02<?xml version="1.0"?><Response Reverseld="1029" Final="Y" Status="0"><HTML><HEAD><TITLE>Request Res  
ponse</TITLE></HEAD><BODY>PTZ command completed successfully.</BODY></HTML></Response>0x03
```

Or

1

```
0x02<?xml version="1.0"?><Response Reverseld="1029" Final="Y" Status="0"><Link>HTTP://myvideo.com/camera1.p  
hp</Link></Response>0x03
```

Or

1

```
0x02<?xml version="1.0"?><Response Reverseld="1029" Final="Y" Status="0"><Text>PTZ command completed succe  
ssfully.</Text></Response>0x03
```

Or

1

```
0x02<?xml version="1.0"?><Response Reverseld="1029" Final="Y" Status="0"><XML><Result>PTZ command completed successfully.</Result></XML></Response>0x03
```

Example: Failed response for a pan/tilt/zoom request (single format illustrated).

1

```
0x02<?xml version="1.0"?><Response Reverseld="1029" Final="Y" Status="1"><Text>PTZ command failed. The device is unavailable on the network.</Text></Response>0x03
```

Example: Negative acknowledgment received for reverse command.

1

```
0x02<?xml version="1.0"?><Response Reverseld="211"><Nak ErrNo="26">Address parameter out of range</Nak></Response>0x03
```

Or

```
0x02<?xml version="1.0"?><Response Reverseld="211"><Nak ErrNo="26"/></Response>0x03
```