



A Sierra Monitor Company

Driver Manual
(Supplement to the FieldServer Instruction Manual)

FS-8704-02 BACnet Ethernet

APPLICABILITY & EFFECTIVITY

Effective for all systems manufactured after May 2009

Driver Version:	1.17
Document Revision:	1

TABLE OF CONTENTS

1	BACnet Ethernet Description	4
2	Driver Scope of Supply	4
2.1	Supplied by FieldServer Technologies for this driver	4
2.2	Provided by the Supplier of 3 rd Party Equipment.....	4
2.2.1	<i>Hardware</i>	4
3	Hardware Connections	5
4	Configuring the FieldServer as a BACnet Ethernet Client	6
4.1	Data Arrays/Descriptors.....	6
4.2	Client Side Connection Descriptions	7
4.3	Client Side Node Descriptors.....	7
4.4	Client Side Map Descriptors	8
4.4.1	<i>FieldServer Related Map Descriptor Parameters</i>	8
4.4.2	<i>Driver Related Map Descriptor Parameters</i>	8
4.4.3	<i>Timing Parameters</i>	9
4.4.4	<i>Map Descriptor Examples</i>	10
5	Configuring the FieldServer as a BACnet Ethernet Server	11
5.1	Driver Specific FieldServer Parameters	11
5.2	Data Arrays.....	11
5.3	Server Side Connection Descriptors.....	12
5.4	Server Side Node Descriptors.....	12
5.5	Server Side Map Descriptors	13
5.5.1	<i>FieldServer Specific Map Descriptor Parameters</i>	13
5.5.2	<i>Driver Specific Map Descriptor Parameters</i>	14
5.5.3	<i>Map Descriptor Example</i>	15
Appendix A.	Useful Features	16
Appendix A.1.	BACnet object names	16
Appendix A.2.	Network number	16
Appendix A.3.	Accessing Data from BACnet Properties comprising Arrays of Values.....	17
Appendix A.4.	FieldServer implementation of BACnet priority Arrays.....	17
Appendix A.4.1.	<i>Accessing Priority Array information</i>	18
Appendix A.5.	BACnet State Text Preload	19
Appendix A.5.1.	<i>Method 1 – Using an Offset/User Table:</i>	19
Appendix A.5.2.	<i>Method 2 – Using a Single Data Array:</i>	20
Appendix A.5.3.	<i>Using Intrinsic Reporting for a Multistate Value</i>	21
Appendix A.6.	COV and Intrinsic Reporting	22
Appendix A.6.1.	<i>Notes on COV configuration:</i>	22
Appendix A.6.2.	<i>Notes on Intrinsic Reporting configuration:</i>	22
Appendix A.6.3.	<i>Map Descriptor Example - COV</i>	23
Appendix A.6.4.	<i>Map Descriptor Example – Intrinsic Reporting</i>	23
Appendix A.6.5.	<i>Map Descriptor Example - Enable or Disable COV</i>	23
Appendix B.	Troubleshooting	24
Appendix B.1.	Debugging a BACnet connection	24

Appendix C. Vendor Information 25
Appendix C.1. McQuay25
Appendix C.2. Trane.....25
Appendix C.3. Liebert25

Appendix D. Reference 26
Appendix D.1. Object_Type Legal Values – Abbreviation Descriptions.....26
Appendix D.2. Property Legal Values.....26
Appendix D.3. Units28
Appendix D.4. BACnet Specific Statistics31

1 BACNET ETHERNET DESCRIPTION

The BACnet Ethernet driver allows the FieldServer to transfer data to and from devices over Ethernet using BACnet Ethernet protocol. The FieldServer can emulate either a Server or Client.

All information in a BACnet system is represented in terms of objects. The Object_Identifier is a 32-bit code that identifies the type of Object (also identified by the Object_Type Property) and its "Instance" number, which together uniquely identify the Object within its BACnet device. Theoretically, a BACnet device could have over four million Objects of a particular type. The Object_Name is a text string, which has a unique capability. BACnet devices may broadcast queries for devices that contain Objects with a specific Object_Name. This can greatly simplify project setup.

BACnet requires one Device Object to be present in every BACnet device. The Device Object makes information about the device and its capabilities available to other devices on the networks. Before one BACnet device starts control-related communications with another, it needs to obtain some of the information presented by the other device's Device Object. Unlike other Objects, the Device Object's Instance number must be unique across the entire BACnet internetwork because it is used to uniquely identify the BACnet devices. It may be used to conveniently identify the BACnet device from other devices during installation.

Standard object types are used to hold real time data and other information. Each Object Type is referenced by a number, for example 0 represents an Analog Input.

Each Object consists of a number of prescribed properties, the main property being the Present_Value. Objects are monitored and controlled through their properties. The Analog Input Object is representative of the Objects involved directly with control elements and many of its Properties reflect this.

The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer.

2 DRIVER SCOPE OF SUPPLY

2.1 Supplied by FieldServer Technologies for this driver

FieldServer Technologies PART #	Description
FS-8915-10	UTP cable (7 foot) for Ethernet connection ¹

2.2 Provided by the Supplier of 3rd Party Equipment

2.2.1 Hardware

Part #	Description
	Ethernet 10/100 BaseT hub ²
Trane Building Control Unit (BCU)	Building control unit (BCU) with internal Ethernet adapter.

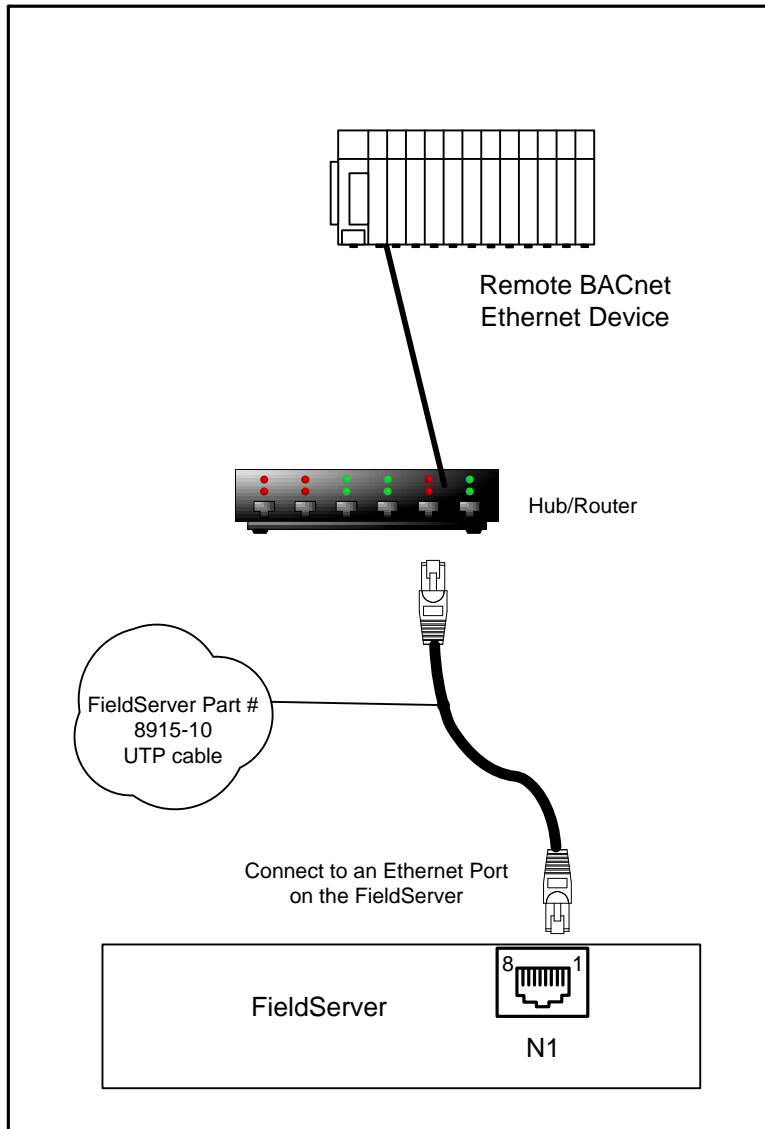
¹ This cable is necessary for connection to the driver. It is shipped with the FieldServer and not separately with the driver.

² Not all FieldServer models support 100BaseT. Consult the appropriate instruction manual for details of the Ethernet speed supported by specific hardware.

3 HARDWARE CONNECTIONS

It is possible to connect a BACnet/IP device using the N1 or the N2³ network ports. These ports need to be configured for BACnet/IP in the configuration file.

Configure the Controller according to manufacturer's instructions



³ Not all ports shown are necessarily available on the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

4 CONFIGURING THE FIELDSEVER AS A BACNET ETHERNET CLIENT

For a detailed discussion on FieldServer configuration, please refer to the FieldServer Configuration Manual. The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer (See “.csv” sample files provided with the FieldServer).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with a BACnet Ethernet Server.

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for BACnet Ethernet communications, the driver independent FieldServer buffers need to be declared in the “Data Arrays” section, the destination device addresses need to be declared in the “Client Side Nodes” section, and the data required from the Servers needs to be mapped in the “Client Side Map Descriptors” section. Details on how to do this can be found below.

4.1 Data Arrays/Descriptors

Note that in the tables, * indicates an optional parameter, with the bold legal value being the default.

Section Title		
Data_Arrays	Column Title	Legal Values
	Data_Array_Name	Provide name for Data Array Up to 15 alphanumeric characters
	Data_Array_Format	Provide data format. Each Data Array can only take on one format. Float, Bit, Uint16, Uint32, Sint16, Sint32, Packed_Bit, Byte, Packed_Byte, Swapped_Byte, AI, AV, AO, BI, BV, BO
	Data_Array_Length	Number of Data Objects. Must be larger than the data storage area required by the Map Descriptors for data placed in this array. 1-10,000

Example

```

// Data Arrays

Data_Arrays
Data_Array_Name , Data_Array_Format , Data_Array_Length
DA_AI_01 , Float , 200
DA_AO_01 , Float , 200
DA_DI_01 , Bit , 200
DA_DO_01 , Bit , 200
```

4.2 Client Side Connection Descriptions

Section Title		
Connection		
Column Title	Function	Legal Values
Adapter	Adapter Name	N1, N2 ⁴
Protocol	Specify protocol used	BACnet_Eth
Poll_Delay*	Time between internal polls	0-32000s; 0.25s

Example

```
// Client Side Connections

Connections
Adapter      , Protocol      , Poll_Delay
N1           , Bacnet_Eth    , 0500s
```

4.3 Client Side Node Descriptors

Section Title		
Nodes		
Column Title	Function	Legal Values
Node_Name	Provide name for Node	Up to 32 alphanumeric characters
Node_ID	BACnet device identifier of physical Server Node	1 - 16777215
Protocol	Specify protocol used	BACnet_Eth
Adapter	Specify port Adapter used	N1, N2 ⁴

Example

```
// Client Side Nodes

Nodes
Node_Name   , Node_ID   , Protocol   , Adapter
BCU_01     , 1         , Bacnet_Eth , N1
```

⁴ Consult the appropriate Instruction manual for details of the ports available on specific hardware.

4.4 Client Side Map Descriptors

4.4.1 FieldServer Related Map Descriptor Parameters

Section Title		
Map Descriptors		
Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor. This is used for Object_Name Property. Refer to Appendix A.1.	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer	One of the Data Array names from "Data Array" section above
Data_Array_Offset	Starting location in Data Array	0 to ("Data_Array_Length" -1)
Function	Function of Client Map Descriptor	Rdbc, Wrbc, Wrbcx, Ars

4.4.2 Driver Related Map Descriptor Parameters

Section Title		
Map Descriptors		
Column Title	Function	Legal Values
Node_Name	Name of remote Server Node.	One of the Node_Names specified in "Client Side Node Descriptors" above
Object_Type (Alias = Data_Type)	Type of object. Refer to Appendix D.1 for more information.	AI, AO, AV, BI, BO, BV, MI, MO, MV, NC
Object_Instance (Alias = Address)	Instance of the object on the device.	0, 1, 2, 3, ...4194303
Property	The BACnet property to be read.	Refer to Appendix D.2
Data_Array_Low_Scale*	Scaling zero in Data Array	-32767 to 32767, 0
Data_Array_High_Scale*	Scaling max in Data Array	-32767 to 32767, 100
Node_Low_Scale*	Scaling zero in Connected Node	-32767 to 32767, 0
Node_High_Scale*	Scaling max in Connected Node	-32767 to 32767, 100
Length*	Used to create an array of sequential Object_Instances on an Object_Type.	1 to max point count of the FieldServer, 1
Array_Index*	When referencing Multistate properties, allows the user to specify the index of the property to be read. If 1 is specified, the first one will be read, if 2 is specified, the second will be read, etc. If 0 is specified, the driver will return the total number (count) of array items linked to the property. If the parameter is not specified, a list of all items will be returned. Refer to Appendix A.3	0 to max number of array items in the BACnet Property, -

4.4.3 Timing Parameters

Section Title		
Map Descriptors		
Column Title	Function	Legal Values
Scan_Interval*	Rate at which data is polled	0-32000s, 2s

4.4.4 Map Descriptor Examples

```
// Client Side Map Descriptors
```

```
Map_Descriptors
```

Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Function	Node_Name	Data_Type	Object_ID	Property	Scan_Interval
CMD_AI_01	DA_AI_01	0	Rdbc	BCU_01	AI	1	Present_Value	20.000s
CMD_AI_02	DA_AI_01	1	Rdbc	BCU_01	AI	2	Present_Value	20.000s
CMD_AI_03	DA_AI_01	2	Rdbc	BCU_01	AI	3	Present_Value	20.000s
CMD_AO_01	DA_AO_01	0	Rdbc	BCU_01	AO	1	Present_Value	30.000s
CMD_AO_02	DA_AO_01	1	Rdbc	BCU_01	AO	2	Present_Value	30.000s
CMD_AO_03	DA_AO_01	2	Rdbc	BCU_01	AO	3	Present_Value	30.000s

```
Map_Descriptors
```

Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Function	Node_Name	Data_Type	Object_ID	Property	Scan_Interval
CMD_DI_01	DA_DI_01	0	Rdbc	BCU_01	BI	1	Present_Value	15.000s
CMD_DI_02	DA_DI_01	1	Rdbc	BCU_01	BI	2	Present_Value	15.000s
CMD_DI_03	DA_DI_01	2	Rdbc	BCU_01	BI	3	Present_Value	15.000s
CMD_DO_01	DA_DO_01	0	Rdbc	BCU_01	BO	1	Present_Value	30.000s
CMD_DO_02	DA_DO_01	1	Rdbc	BCU_01	BO	2	Present_Value	30.000s
CMD_DO_03	DA_DO_01	2	Rdbc	BCU_01	BO	3	Present_Value	30.000s

5 CONFIGURING THE FIELDSEVER AS A BACNET ETHERNET SERVER

For a detailed discussion on FieldServer configuration, please refer to the FieldServer Configuration Manual. The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer (See “.csv” files provided with the FieldServer).

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with a BACnet Ethernet Client

The configuration file tells the FieldServer about its interfaces, and the routing of data required. In order to enable the FieldServer for BACnet Ethernet communications, the driver independent FieldServer buffers need to be declared in the “Data Arrays” section, the FieldServer virtual node(s) needs to be declared in the “Server Side Nodes” section, and the data to be provided to the Clients needs to be mapped in the “Server Side Map Descriptors” section. Details on how to do this can be found below.

Note that in the tables, * indicates an optional parameter, with the bold legal value being the default.

5.1 Driver Specific FieldServer Parameters

Section Title		
Bridge		
Column Title	Function	Legal Values
Title	FieldServer name	Text
Network_Number*	Speify a unique network number if there are multiple virtual Server Nodes. Refer to Appendix A.2	1 – 65535, 5

Example

```
// FieldServer Driver specific parameters

Bridge
Title
BACnet Server
```

5.2 Data Arrays

Section Title		
Data_Arrays		
Column Title	Function	Legal Values
Data_Array_Name	Provide name for Data Array	Up to 15 alphanumeric characters
Data_Format	Provides Data format	Float, Bit, Uint16, Uint32, Sint16, Sint32, Packed_Bit, Byte, Packed_Byte, Swapped_Byte, AI, AV, AO, BI, BV, BO
Data_Array_Length	Number of Data Objects	1-10,000

Example

```
// Data Arrays

Data_Arrays
Data_Array_Name , Data_Format , Data_Array_Length
DA_AI_01 , Float , 200
DA_AO_01 , Float , 200
DA_DI_01 , Bit , 200
DA_DO_01 , Bit , 200
```

5.3 Server Side Connection Descriptors

Section Title		
Connections		
Column Title	Function	Legal Values
Adapter	Adapter Name	N1, N2 ⁵
Protocol	Specify protocol used	BACnet_Eth

Example

```
// Server Side Connections

Connections
Adapter , Protocol 6
N1 , Bacnet_Eth
```

5.4 Server Side Node Descriptors

Section Title		
Nodes		
Column Title	Function	Legal Values
Node_Name	Provide name for node	Up to 32 alphanumeric characters
Node_ID	BACnet station address of physical Server node	1 - 16777215
Protocol	Specify protocol used	BACnet_Eth
Node_Option*	Enable or disable COV for this Node. Refer to Appendix A.6 for further information.	COV_Enable , COV_Disable, -

⁵ Not all ports shown are necessarily supported by the hardware. Consult the appropriate Instruction manual for details of the ports available on specific hardware.

⁶ Note that no connection information is necessary on Server side.

Example

```
// Server Side Nodes

Nodes
Node_Name      , Node_ID  , Protocol      7
Virtual_BCU_11 , 1       , Bacnet_Eth
```

5.5 Server Side Map Descriptors

5.5.1 FieldServer Specific Map Descriptor Parameters

Section Title		
Map Descriptors		
Column Title	Function	Legal Values
Map_Descriptor_Name	Name of this Map Descriptor. This is used for Object_Name Property. Refer to Appendix A.1.	Up to 32 alphanumeric characters
Data_Array_Name	Name of Data Array where data is to be stored in the FieldServer	One of the Data Array names from "Data Arrays" section above
Data_Array_Offset	Starting location in Data Array	0 to ("Data_Array_Length" -1)
Function	Function of Server Map Descriptor	Server

⁷ Note that adapter is not declared under Server Side Nodes.

5.5.2 Driver Specific Map Descriptor Parameters

Section Title		
Map Descriptors		
Column Title	Function	Legal Values
Node_Name	Name of Node to fetch data from	One of the Node Names specified in "Server Side Node Descriptors" above
Object_Type (Alias = Data_Type)	Data type in Controller	AI, AO, AV, BI, BO, BV, MI, MO, MV, NC. Refer to Appendix D.1.
Object_Instance (Alias = Address)	Instance of the Object on the Device.	0, 1, 2, 3, ...4194303
Units*	The object units	Refer to Appendix D.3, No-Units
Data_Array_Low_Scale*	Scaling zero in Data Array	-32767 to 32767, 0
Data_Array_High_Scale*	Scaling max in Data Array	-32767 to 32767, 100
Node_Low_Scale*	Scaling zero in Connected Node	-32767 to 32767, 0
Node_High_Scale*	Scaling max in Connected Node	-32767 to 32767, 100
Active_Text ⁸ *	Specify the Active Text property of the Object	Any text string of length up to 40 characters, Active
Inactive_Text ⁸ *	Specify the Inactive Text property of the Object	Any text string of length up to 40 characters, Inactive
Relinquish_Default ⁸	Specify the value to be returned as Present_Value on startup or when control is relinquished. Must be specified for outputs. Refer to Appendix D.2	Any Float value
Length*	Used to create an array of sequential Object_Instances on an Object_Type.	1 to max point count of the FieldServer, 1
Notification_Class*	Specify the Mapdescriptor_Name of the Notification_Class Object that manages Intrinsic Reporting for this Map Descriptor. Refer to Appendix A.6.	One of the configured Mapdescriptor_Names of type NC (Notification Class). Objects are not available to any NC if left out.
Ack_Required*	For a Notification_Class Object, specify whether EventNotifications require a user Acknowledgement. Refer to Appendix A.6.	Yes, No
COV_Increment*	For a Server Map Descriptor of type AO or AI, initialize the COV_Increment property. Refer to Appendix A.6.	Any Float value, 0
Input_alarm_State*	For binary points the user can specify which of the states (0 or 1) to regard as an alarm state.	0, 1
Description*	Specify the object's description property. Eg Room Temp – Refer to Appendix A.6.3	Any text string of length up to 40 characters, If not configured, defaults to Object_Name.

⁸ See the BACnet DFS to determine if a particular object supports this property

5.5.3 Map Descriptor Example

```
// Server Side Map Descriptors

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Object_Type , Object_Instance , Units
SMD_AI_01 , DA_AI_01 , 0 , Passive , Virtual_BCU_11 , AI , 1 , Degrees-Fahrenheit
SMD_AI_02 , DA_AI_01 , 1 , Passive , Virtual_BCU_11 , AI , 2 , Degrees-Fahrenheit
SMD_AI_03 , DA_AI_01 , 2 , Passive , Virtual_BCU_11 , AI , 3 , Degrees-Fahrenheit
SMD_AO_01 , DA_AO_01 , 0 , Passive , Virtual_BCU_11 , AO , 1 , percent-relative-humidity
SMD_AO_02 , DA_AO_01 , 1 , Passive , Virtual_BCU_11 , AO , 2 , percent-relative-humidity
SMD_AO_03 , DA_AO_01 , 2 , Passive , Virtual_BCU_11 , AO , 3 , percent-relative-humidity

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Object_Type , Object_Instance
SMD_DI_01 , DA_DI_01 , 0 , Passive , Virtual_BCU_11 , BI , 1
SMD_DI_02 , DA_DI_01 , 1 , Passive , Virtual_BCU_11 , BI , 2
SMD_DI_03 , DA_DI_01 , 2 , Passive , Virtual_BCU_11 , BI , 3
SMD_DO_01 , DA_DO_01 , 0 , Passive , Virtual_BCU_11 , BO , 1
SMD_DO_02 , DA_DO_01 , 1 , Passive , Virtual_BCU_11 , BO , 2
SMD_DO_03 , DA_DO_01 , 2 , Passive , Virtual_BCU_11 , BO , 3
```

Appendix A. Useful Features

Appendix A.1. BACnet object names

When an external BACnet Client builds a list of Object Names, the BACnet Server Map Descriptor names determine the BACnet Object Name. If the Map Descriptor length is greater than 1, the Object Name will be suffixed with the index into the Map Descriptor. For example, if the Map Descriptor name is SMD_AI_01 and the length 3, then the Object Names will be SMD_AI_01[0], SMD_AI_01[1] and SMD_AI_01[2].

Appendix A.2. Network number

If multiple BACnet Nodes are specified on the Server side, the FieldServer automatically defaults to virtual operation. In effect, the FieldServer simulates a software router. BACnet identifies this FieldServer using a combination of it's network number and IP/MAC address. The default Network number of a FieldServer is 5. If there is more than one FieldServer on a BACnet intranet with multiple nodes, the default network number of the additional FieldServers will need to be changed and a unique number allocated to each.

A unique network number will need to be assigned if *both* of the following conditions are true:

- The FieldServer has multiple BACnet Server nodes.
- There is more than one FieldServer on a network which includes multiple BACnet nodes.

To override the FieldServer's default network number 5 include the following in the configuration file:

Section Title		
Bridge		
Column Title	Function	Legal Values
Title	FieldServer name	Text
Network_number*	Specify a unique network number if there are multiple virtual Server nodes.	1-65534, 5

Example

```
// FieldServer Driver specific parameters

Bridge
Title          , Network_Number
BACnet Server  , 6
```

Note: While it is theoretically possible to have up to 65535 virtual nodes, it is recommended that a maximum of 35 is configured. If an application requires the configuration of more than 35 virtual nodes please consult FST.

Appendix A.3. Accessing Data from BACnet Properties comprising Arrays of Values

Some BACnet Object properties (e.g. Priority_Array) are arrays of values (the Priority_Array property is an array of 16 values). In order to read a specific array entry, the Array_Index must be specified in the Map Descriptor. Array_Index is a Client Side Map Descriptor function

The following example shows a configuration that will read the Priority_Array value at Array_Index 7, belonging to Analog Output 1.

Map_Descriptors								
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Function	Node_Name	Data_Type	Object_ID	Property	Array_Index
CMD_AI_01	DA_AI_01	0	Rdbc	BCU_01	AO	1	Priority_Array	7

Details of the relevant BACnet properties and their associated arrays can be found in the BACnet Protocol Spec.

Appendix A.4. FieldServer implementation of BACnet priority Arrays

When BACnet Output objects are written to the Server side of the FieldServer, an associated write priority is given to each write value. When the FieldServer receives the write value, it stores it to the Map Descriptor Priority Array Table at the specified priority. The Priority Array Table is then scanned and the value with the highest priority is stored to the Data Array location specified by the Map Descriptor.

When a Write “Relinquished” command is received, the value is removed from the Priority Array Table and the next highest value from the Priority Array Table is stored to the Data Array.

If all values have been “Relinquished” from the Priority Array Table, then the Map Descriptors “Relinquish Default” value will be stored to the Data Array.

Appendix A.4.1. Accessing Priority Array information

The Priority Array table and its "In_Use" (or Not Relinquished) state are stored internally to every Map Descriptor, and cannot be accessed directly. The information can be accessed indirectly by specifying the following Data Arrays which will maintain an exact copy of the Priority Array Table for the Map Descriptor.

Section Title		
Map_Descriptors		
Column Title	Function	Legal Values
DA_Pri_Array	Name of Data Array where the Priority Array Table will be stored. Location 0 is the Relinquish Default value and locations 1 to 16 the different entries of the Priority Array Table.	Up to 16 alphanumeric characters
DA_Pri_Array_Offset*	Starting location in Data Array.	1-65535, 0
DA_Pri_In_Use	Name of Data Array that indicates if a particular Priority Value is in use. Location 0 indicates whether the Relinquish Default has been set and locations 1 to 16 indicate whether the index is in use (1), or Relinquished (0).	Up to 16 alphanumeric characters
DA_Pri_In_Use_Offset*	Starting location in Data Array.	1-65535, 0

```
// Analog Output Map_Descriptor for testing Priority Arrays

Map_Descriptors
Map_Descriptor_Name ,Data_Type ,Object_ID ,Function ,Data_Array_Name ,Data_Array_Index ,Node_Name ,Length ,Relinquish_default ,DA_Pri_Array ,DA_Pri_Array_Offset ,DA_Pri_In_Use ,DA_Pri_In_Use_Offset
CMD_AOP_1 ,AO ,1 ,Passive ,DA_OUT ,0 ,N1 11 ,1 ,40.56 ,DA_Pri_Array_1 ,0 ,DA_Pri_in_use_1 ,0
```

Appendix A.5. BACnet State Text Preload

BACnet Multistate Objects have a State_Text property. This property is defined as an array of character strings representing descriptions of all possible states of the Present_Value. The number of descriptions matches the number of states defined in the Number_Of_States property. The Present_Value, interpreted as an integer, serves as an index into the array.

When Multistate Objects are configured on a BACnet server it is necessary to define the State_Text property. This section illustrates how to define the State_Text character strings and how to associate these definitions with Multistate Server Map Descriptors.

The maximum permitted length of any State_Text string is 50 characters.

Appendix A.5.1. Method 1 – Using an Offset/User Table:

```
//set up a look up table

Offset_Table
Offset_Table_Name      , Table_String      , Table_Index_Value
FIRE_ALARM_TEXT       , SYSTEM READY  , 1
FIRE_ALARM_TEXT       , ALARM         , 2
FIRE_ALARM_TEXT       , MAINTENANCE   , 3
FIRE_ALARM_TEXT       , OFF-LINE      , 4
FIRE_ALARM_TEXT       , IN SERVICE    , 5
FIRE_ALARM_TEXT       , OTHER         , 6
```

Must be a positive integer.

```
Data_Arrays
Data_Array_Name      , Data_Format      , Data_Array_Length
DA_MI_01             , UINT16           , 100
```

```
Map_Descriptors
Map_Descriptor_Name  , Data_Array_Name  , Data_Array_Offset , Function , Node_Name , Data_Type , Object_ID , Length , State_Text_Array
CMD_MI_01           , DA_MI_01         , 0                 , Passive , N1 11    , MI       , 1         , 1      , Fire_Alm_Text
```

Appendix A.5.2. Method 2 – Using a Single Data Array:

Data_Arrays		
Data_Array_Name	Data_Format	Data_Array_Length
DA_MI_01	UINT16	100
DA_STATE_TXT	BYTE	200

Preloads			
Data_Array_Name	Preload_Data_Value	Preload_Data_Format	Preload_Data_Index
DA_STATE_TXT	MyState1 MyState2 MyState3 MyState4 MyState5 MyState6	String	0

Map_Descriptors							
Map_Descriptor_Name	Data_Type	Object_ID	Function	Data_Array_Name	Node_Name	Length	State_Text_Array
CMD_MI_01	MI	1	Passive	DA_MI_01	N1 11	1	Da_State_Txt

Appendix A.5.3. Using Intrinsic Reporting for a Multistate Value

To use Intrinsic Reporting for a Multistate value, it is necessary to classify each of the states as either Normal, Alarm or Trouble. This is done by adding another column to the Offset_Table, called Table_User_Value. Each state is then classified by inserting one of the following values in its row:

0 = normal

1 = alarm

2 = fault

```
//set up a look up table
```

Offset_Table			
Offset_Table_Name	, Table_String	, Table_Index_Value	, Table_User_Value
FIRE_ALARM_TEXT	, SYSTEM READY	, 1	, 0
FIRE_ALARM_TEXT	, ALARM	, 2	, 1
FIRE_ALARM_TEXT	, MAINTENANCE	, 3	, 2
FIRE_ALARM_TEXT	, OFF-LINE	, 4	, 2
FIRE_ALARM_TEXT	, IN SERVICE	, 5	, 0
FIRE_ALARM_TEXT	, OTHER	, 6	, 2

Note: The state value (Table_Index_Value) is an enumerated value greater than zero. Zero is not a valid value, but since many client side configurations will wake up with values of zero, the BACnet server will treat a value of zero as a normal value (i.e. not as an alarm or fault value)

Appendix A.6. COV and Intrinsic Reporting

The COV and Intrinsic Reporting services are two distinct ways in which point values can be reported to a client workstation as they change, i.e. in an event-driven opposed to a polling method. This can increase performance dramatically compared to polling method alone. It also reduces network traffic significantly.

The services are suited to different purposes:

- COV is suited to value updates. On analog points the sensitivity can be set using the COV_Increment property. Only changes larger than the COV_Increment value will be reported.
- Intrinsic Reporting is used for alarming. It is implemented via Notification_Class objects, which can receive subscriptions from client workstations that add themselves to the RecipientList property of a Notification_Class object. Notifications are done using ConfirmedEventNotification or UnconfirmedEventNotification. Intrinsic Reporting also allows for alarms to be acknowledged and for all subscribed client workstations to be notified of alarm acknowledgements.

Appendix A.6.1. Notes on COV configuration:

- No special configuration entries are needed to enable COVs. The service is enabled by default for all protocols except MSTP. Most client workstations will automatically subscribe to all points once they discover that the FieldServer supports COV services. This only applies to BACnet Objects - Property subscribes are not supported by the FieldServer.
- The Node_Option parameter can be configured to enable or disable COV. An example configuration is presented in Appendix A.6.5
- For analog Server Map Descriptors the user may optionally configure a COV_Increment value to adjust the reporting threshold. If it is not set the COV_Increment defaults to zero.
- Change of Value (COV) Notifications are generated for all data objects for which a remote client has issued a SubscribeCOV-Request. The SubscribeCOV-Request regulates whether Notifications are Confirmed or Unconfirmed. The remote client may also write the COV_Increment property in order to control the deadband for changes in analog values. The COV_Increment property can be initialized via the configuration file by setting the COV_Increment Map Descriptor Property. The value set by the configuration is an initial value that is loaded on startup. It would be replaced by any new value written by the Client
- COV is not available on the ProtoCessor

Appendix A.6.2. Notes on Intrinsic Reporting configuration:

- Intrinsic Reporting is managed by Notification Class objects. At least one Notification Class object must be configured for Intrinsic Reporting to work.
- Each Data_Object that is to be monitored by Intrinsic Reporting must be linked to a Notification Class object via the Notification_Class Map Descriptor Property. For analog points alarm limits must be set up, and for binary points, the Input_Alarm_State must be set up.
- The Notification_Class object contains properties that allow a client workstation to modify rules governing event reporting, such as event_type, days of week, start and end times etc.
- **The RecipientLists are non-volatile, and subscriptions must be renewed on system restart.**

Appendix A.6.3. Map Descriptor Example - COV

```
// Notification Class Object

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Data_Type , Object_ID , Ack_Required
SMD_NC_01 , DA_NC_01 , 0 , Passive , Virtual_Dev_11 , NC , 01 , -
```

```
Map_Descriptors
Map_Descriptor_Name, Description , Data_Array_Name, Data_Array_Offset, Function, Node_Name , Data_Type, Object_ID, Relinquish_Default, State_Text_Array, Notification_Class, High_Alarm, Low_Alarm, Input_Alarm_State, Confirmed, COV_Increment
SMD_AI_01 , Room Temp, DA_AI_01 , 0 , Passive , Virtual_Dev_11, AI , 01 , - , - , SMD_NC_01 , 100 , 10 , - , Yes , 1.0
```

Appendix A.6.4. Map Descriptor Example – Intrinsic Reporting

```
// Notification class Objects

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Object_Type , Object_Instance , Ack_Required
SMC_NC_01 , DA_NC_01 , 0 , Passive , Virtual_BCU_11 , NC , 1 , Yes
SMC_NC_02 , DA_NC_01 , 1 , Passive , Virtual_BCU_11 , NC , 2 , No

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Object_Type , Object_Instance , Notification_Class
SMD_AI_01 , DA_AI_01 , 0 , Passive , Virtual_BCU_11 , AI , 1 , SMC_NC_01
SMD_AO_01 , DA_AO_01 , 0 , Passive , Virtual_BCU_11 , AO , 1 , SMC_NC_02

Map_Descriptors
Map_Descriptor_Name , Data_Array_Name , Data_Array_Offset , Function , Node_Name , Object_Type , Object_Instance , Notification_Class
SMD_DI_01 , DA_DI_01 , 0 , Passive , Virtual_BCU_11 , BI , 1 , SMC_NC_01
SMD_DO_01 , DA_DO_01 , 0 , Passive , Virtual_BCU_11 , BO , 1 , SMC_NC_02
```

Appendix A.6.5. Map Descriptor Example - Enable or Disable COV

```
Nodes
Node_Name , Node_ID , Protocol , Node_Option
Virtual_Dev_11 , 11 , Bacnet_Eth , COV_Disable
```

Appendix B. Troubleshooting

Appendix B.1. Debugging a BACnet connection

- If duplicate Object_Instances are configured in the FieldServer, the second call of the Instance will overwrite the first one. This may cause a BACnet Object to be “lost.”
- If “Virtual_BCU_...” is not being indicated as the device description for the FieldServer on the BACnet SCADA system, then the FieldServer is not communicating with the SCADA system. If the Present_Value’s name is being indicated, but the Present_Value shows question marks, then it is likely that the Client side of the FieldServer is not communicating.
- Some of the BACnet IP features result in the creation of files (priarray.ini; desc.ini; alarms.ini) on the FieldServer. Sometimes updates of firmware can result in these files becoming outdated. Deleting these files will restore configuration defaults and may assist with configuration errors.
- Extra memory is required to store Map Descriptors that have the active/inactive text parameters specified. If the defaults are appropriate, do not specify these parameters. This will save memory and allow more Map Descriptors to be created

Appendix C. Vendor Information

Appendix C.1. McQuay

McQuay Units are shipped with a default Device instance of the last 6 digits of the McQuay Serial number.

Appendix C.2. Trane

When new points are added to the FieldServer it is important to restart the Summit Workstation or BCU, otherwise these new points may not be seen by the FieldServer.

Disconnect the FieldServer from the BACnet network when transferring images to the BCU.

Appendix C.3. Liebert

Polling BACnet addresses that are not configured for Liebert systems may cause the connection to fail in older versions of Liebert. Please contact your Liebert supplier for more information.

Appendix D. Reference

Appendix D.1. Object_Type Legal Values – Abbreviation Descriptions

- AI ANALOG_INPUT
- AO ANALOG_OUTPUT
- AV ANALOG_VALUE
- BI BINARY_INPUT
- BO BINARY_OUTPUT
- BV BINARY_VALUE
- MI MULTI_STATE_INPUT
- MO MULTI_STATE_OUTPUT
- MV MULTI_STATE_VALUE
- NC NOTIFICATION_CLASS_OBJECT

Appendix D.2. Property Legal Values

Legal Value	Description
Object_Identifier	This property is a numeric code that is used to identify the object. It is unique within the BACnet Device that maintains it.
Object_List	Relevant to Device Object Type. This property is a BACnetARRAY of Object_Identifier, one Object_Identifier for each object within the device that is accessible through BACnet services. An Object_Identifier is composed of Object Type and Object Instance and must be unique within a BACnet Device, e.g. Object Type = Analog Input, Object Instance = 3
Present_Value	This property contains the present value of the Input / Output / Value
Object_Name	Character string providing the name of a BACnet object. The set of characters used in the Object_Name is restricted to printable characters. The Object_Name is determined by the Map_Descriptor_Name.
Description	Character string describing a BACnet object. This can be defined by the user to give additional detail about the Object.
Out_of_service	The Out_Of_Service property, of type BOOLEAN, is an indication whether (TRUE) or not (FALSE) the physical input that the object represents is not in service. This means that the Present_Value property is decoupled from the physical input and will not track changes to the physical input when the value of Out_Of_Service is TRUE. In addition, the Reliability property and the corresponding state of the FAULT flag of the Status_Flags property shall be decoupled from the physical input when Out_Of_Service is TRUE. While the Out_Of_Service property is TRUE, the Present_Value and Reliability properties may be changed to any value as a means of simulating specific fixed conditions or for testing purposes. Other functions that depend on the state of the Present_Value or Reliability properties shall respond to changes made to these properties while Out_Of_Service is TRUE, as if those changes had occurred in the physical input.
Event_State	The Event_State property, of type BACnetEventState, is included in order to provide a way to determine if this object has an active event state associated with it. If the object supports intrinsic reporting, then the Event_State property shall indicate the event state of the object. If the object does not support intrinsic reporting, then the value of this property shall be NORMAL. Other values: FAULT, OFF-NORMAL, HIGH-LIMIT, LOW-LIMIT, LIFE-SAFETY-ALARM

Legal Value	Description
Units	This property contains the units associated with the Present_Value property.
Reliability	The Reliability property, of type BACnetReliability, provides an indication of whether the Present_Value or the operation of the physical input in question is "reliable" as far as the BACnet Device or operator can determine and, if not, why. The following values are supported: NO_FAULT_DETECTED, UNRELIABLE_OTHER
Priority_Array	This property relates to Output and Value Object Types and is a read only array that contains prioritized commands or NULLs in the order of decreasing priority. The highest priority (lowest array index) with a non-NULL value is the active command.
State_Text	Relevant to Multistate Object Types: This property is a BACnetARRAY of character strings representing descriptions of all possible states of the Present_Value. The number of descriptions matches the number of states defined in the Number_Of_States property. The Present_Value, interpreted as an integer, serves as an index into the array.
Number_Of_States	Relevant to Multistate Object Types: this property sets the total number of states for which descriptions will be returned as defined under the State_Text property. The number of states will be determined automatically by the largest state number used when configuring the Offset Table (Refer to Appendix A.5).
Max_Master	Relevant to BACnet MS/TP Device Object Type: The Max_Master property, of type Unsigned, shall be present if the device is a master node on an MS/TP network. The value of Max_Master specifies the highest possible address for master nodes and shall be less than or equal to 127. If the Max_Master property is not writeable via BACnet services, its value shall be 127.
Max_Info_Frames	Relevant to BACnet MS/TP Device Object Type: The Max_Info_Frames property, of type Unsigned, shall be present if the device is a node on an MS/TP network. The value of Max_Info_Frames specifies the maximum number of information frames the node may send before it must pass the token. If Max_Info_Frames is not writable or otherwise user configurable, its value shall be 1.
Active_Text	Relevant to Binary Object Types: This property, of type CharacterString, characterizes the intended effect of the ACTIVE state of the Present_Value property from the human operator's viewpoint. The content of this string is a local matter, but it is intended to represent a human-readable description of the ACTIVE state. For example, if the physical input is a switch contact, then the Active_Text property might be assigned a value such as "Fan 1 On".
Inactive_Text	This property, of type CharacterString, characterizes the intended effect of the INACTIVE state of the Present_Value property from the human operator's viewpoint. The content of this string is a local matter, but it is intended to represent a human-readable description of the INACTIVE state. For example, if the physical input is connected to a switch contact, then the Inactive_Text property might be assigned a value such as "Fan 1 Off".
Description	A character string giving more information about the Object associated with the Present_Value property.
Firmware_revision	The firmware revision of the application.
Relinquish_Default	This property is the default value to be used for the Present_Value property when all command priority values in the Priority_Array property have a NULL value.

Appendix D.3. Units

Unit	Variation 1	Variation 2	Variation 3
Amperes	Amps	A	
Bars			
BTUs			
BTUs-per-hour			
btus-per-pound			
btus-per-pound-dry-air			
centimeters			
centimeters-of-mercury			
centimeters-of-water			
cubic-feet			
cubic-feet-per-minute			
cubic-feet-per-seconds			
cubic-meters			
cubic-meters-per-hour			
cubic-meters-per-seconds			
Currency1			
Currency2			
Currency3			
Currency4			
Currency5			
Currency6			
Currency7			
Currency8			
Currency9			
Currency10			
cycles-per-hour			
cycles-per-minute			
days			
degrees-angular			
Degrees-Celsius	Deg-C	Deg_C	
degrees-Celsius-per-hour			
degrees-Celsius-per-minute			
Degrees-days-Celsius			
Degrees-days-Fahrenheit			
Degrees-Fahrenheit	Deg-F	Deg_F	
degrees-Fahrenheit-per-hour			
degrees-Fahrenheit-per-minute			
Degrees-Kelvin	Deg-K	Deg_K	
degrees-phase			
delta-degrees-Fahrenheit			
delta-degrees-Kelvin			
feet			
feet-per-minute			
feet-per-second			
foot-candles			
grams-water-per-kg-dry-air			
hectopascals			
Hertz	Hz		

Unit	Variation 1	Variation 2	Variation 3
Horsepower	HP		
hours			
imperial-gallons			
imperial-gallons-per-min			
inches			
inches-of-mercury			
inches-of-water			
Joules			
joules-per-degree-kelvin			
joules-per-kilogram-degree-kelvin			
joules-per-kilogram-dry-air			
Kilograms	Kg		
kilograms-per-hour			
kilograms-per-minute			
kilograms-per-second			
Kilohertz	KHz		
kilohms			
Kilojoules			
kilojoules-per-kilogram			
kilometers-per-hour			
Kilopascals	KPa		
kilovolt-amperes	kilovolt-amps	KVA	
kilovolt-amperes-reactive	KVAR		
kilovolts			
kilowatt-hour-per-square-foot			
kilowatt-hour-per-square-meter			
kilowatt-hours	KWh		
kilowatts	KW		
liters			
liters-per-hour			
liters-per-minute			
liters-per-second			
lumens			
luxes			
Megahertz	MHz		
megajoules			
megajoules-per-square-foot			
megajoules-per-square-meter			
megavolt-amperes	megavolt-amps		
Megavolt-amperes-reactive	MVAR		
megavolts			
megawatts	MW		
megohms			
meters			
meters-per-second			
miles-per-hour			
milliamperes	milliamps		
millibars			
millimeters			
millimeters-of-mercury			

Unit	Variation 1	Variation 2	Variation 3
Millivolts			
Milliwatts			
minutes			
months			
No-Units	No Units	No_Units	None
ohms			
parts-per-billion			
parts-per-million			
Pascals			
Percent			
percent-obscuration-per-foot			
percent-obscuration-per-meter			
percent-per-second			
percent-relative-humidity	% RH; %RH	Percent RH;	PercentRH
Per-hour			
per-minute			
per-second			
pounds-force-per-square-inch	PSI	pounds-force-per-sq-inch	
Pounds-mass			
pounds-mass-per-hour			
pounds-mass-per-minute			
pounds-mass-per-second			
Power-Factor	PF		
psi-per-degrees-fahrenheit			
radians			
revolutions-per-minute			
seconds	Secs	S	
square-centimeters			
square-feet			
square-inches			
square-meters			
Therms			
ton-hours			
Tons			
Tons-refrigeration			
US-gallons	Gallons		
us-gallons-per-minute	GPM		
Volt-Amperes	Volt-Amps	VA	
volt-amperes-reactive	VAR		
Volts	Voltage		
watt-hours	Wh		
Watts	W		
watts-per-square-foot			
watts-per-square-meter			
watts-per-square-meter-degrees-kelvin			
weeks			
years			

Appendix D.4. BACnet Specific Statistics

Stat	Description	Resolution
Link Control	A “who-is” link control message was send or received.	It is normal to receive a few link control messages. If the number is higher than the transmit/receive messages, however, there may be a problem with lost communications..
Unsupported Properties	A request for an unsupported property was received	This is not an error. BACnet clients often poll all properties of a particular object to determine which properties are supported.
Segmentation Not Supported	Data was requested but the response would have exceeded the maximum size of the APDU and could not be sent using an un-segmented message.	This is not an error - the BACnet client will use a different method to read data from the FieldServer.
Sequence Error	Invoke ID of a reply did not match the Invoke ID of the poll.	You should not see this message. It normally indicates a configuration error.
Write Access Denied	A write to an object was denied.	This typically happens when trying to write to an Input Object that is not Out-Of-Service. It is not possible to write to Input Objects.
Exception Errors	A BACnet Service was denied because it is not supported	This may be a problem on the Client system. Consult the PIC statement to determine what services are supported.