FieldServer Protocol Driver Sheet
XML over HTTP Driver

FS-8704-15 **Version: 1.15 / Rev. 4.C**

1 DESCRIPTION

The XML Driver is built on HTTP web technology (Port 80) and it uses pages formatted in XML syntax to respond with or decoded and store. Both a client and a server are supported.

The Server side is an XML formatted response of the internal Data Array structure contained within the FieldServer, requested from a Remote Client device to the FieldServer URL.

The Client uses a HTTP GET request to a specified URL to request XML data. The driver has the ability to decode the XML response and store different Elements uniquely identified by some attribute within the Element. The data of the matching Element is stored in the FieldServer Data Arrays.

1.1 Connection Facts

FieldServer Mode	Comments
Server	This mode is always enabled within the XML driver and is requested by "http://<ip address>/data_arrays.xml" where <ip address> corresponds to the Fieldserver address.
Client	Supports multiple client connections to different URLs, with associated decoding map descriptors linked to an active GET URL request.

2 FORMAL DRIVER TYPE

Ethernet

Client or Server

3 COMPATIBILITY

FieldServer Model	Compatible with Driver
FS-B35 Series	Yes
ProtoNode/ProtoAir	Yes
QuickServer FS-QS-10xx	No
QuickServer FS-QS-12xx	Yes
QuickServer FS-QS-20xx	No
QuickServer FS-QS-22xx	Yes

4 CONNECTION INFORMATION

Connection Type: Ethernet
Ethernet Speeds Supported: 10Base-T, 100Base-T[1]
Port: 80

5 COMMUNICATION FUNCTIONS SUPPORTED

NOTE: The driver does not support HTTP 1.0 encoding, only HTTP 1.1 encoding is supported.

The XML driver supports both GET and POST as a Client and Server.

A remote client device can use a HTTP GET request to retrieve the Data stored in the FieldServer Data Arrays formatted in a XML page, and POST will be used to modify a specified Data Array Element.

When the FieldServer is used to Retrieve data from a remote device, a READ operation will perform a GET to a specified URL, and a WRITE operation will perform a POST to a specified URL with the contents of the Write_Cmd parameter being used.

5.1 Data Types Supported

FieldServer Data Type	Description (or Device Data Type)
FLOAT, INT, UINT, BYTE, BIT	The default data type is Float.

5.2 Server Options Supported

The driver currently supports GET and POST response to retrieve and update data stored within the FieldServer Data Arrays.

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

5.3 Server HTTP Get Operation

The Following URL is supported for retrieiving data “http://<ip address>/data_arrays.xml”. The response is formatted with the following XML Schema (XSD):

```
<xs:schema elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
 <xs:element name="DATA_ARRYS">
   <xs:attribute name="FST_XML_VERSION" type="xs:string" />
   <xs:attribute name="MAX_INDEX" type="xs:string" />
   <xs:attribute name="BRIDGE_TITLE" type="xs:string" />
   <xs:complexType>
     <xs:sequence>
       <xs:element name="DATA_ARAY">
         <xs:attribute name="NAME" type="xs:string" />
         <xs:attribute name="FORMAT" type="xs:string" />
         <xs:attribute name="LENGTH" type="xs:string" />
         <xs:attribute name="INDEX" type="xs:string" />
         <xs:complexType>
           <xs:element name="Data" type="xs:string">
             <xs:attribute name="OFFSET" type="xs:string" />
             <xs:attribute name="DATA_AGE" type="xs:string" />
             <xs:attribute name="STATUS" type="xs:string" />
           </xs:element>
         </xs:complexType>
       </xs:element>
   </xs:complexType>
 </xs:element>
</xs:schema>
```

An example of a typical request and response is included below:

```
http://192.168.1.5/data_arrays.xml
```

Response:

```
<DATA_ARRAYS FST_XML_VERSION="1.00" MAX_INDEX="5" BRIDGE_TITLE="Lonworks Server">
        <DATA_ARRAY NAME="DA_Group" FORMAT="Byte" LENGTH="99" INDEX="1">
                <DATA OFFSET="0" DATA_AGE="3:00:39:36.736s" STATUS="0">0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
        0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
        0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</DATA>
        </DATA_ARRAY>
        <DATA_ARRAY NAME="DA_P16_1Sta" FORMAT="Byte" LENGTH="2" INDEX="2">
                <DATA OFFSET="0" DATA_AGE="34.705s" STATUS="0">0 0</DATA>
        </DATA_ARRAY>
        <DATA_ARRAY NAME="DA_Pnl16_1" FORMAT="Byte" LENGTH="8" INDEX="3">
                <DATA OFFSET="0" DATA_AGE="3:00:39:36.646s" STATUS="0">0 0 0 1 0 0 0 0</DATA>
        </DATA_ARRAY>
        <DATA_ARRAY NAME="DA_P16_2Sta" FORMAT="Byte" LENGTH="2" INDEX="4">
                <DATA OFFSET="0" DATA_AGE="34.050s" STATUS="0">0 0</DATA>
        </DATA_ARRAY>
        <DATA_ARRAY NAME="DA_Pnl16_2" FORMAT="Byte" LENGTH="8" INDEX="5">
                <DATA OFFSET="0" DATA_AGE="3:00:39:36.548s" STATUS="0">1 0 0 0 0 0 0 0</DATA>
        </DATA_ARRAY>
</DATA_ARRAYS>
```

5.4 Server HTTP POST Operation

Data Array updates are done via POST commands, and the URL is “http://<ip address>/post.cgi”. The payload of the post should be in the following format:

```
DATA_ARRAY_NAME.OFFSET=VALUE
```

Keyword	Description
DATA_ARRAY_NAME	FieldServer Data Array Name.
OFFSET	Offset into the Data Array.
VALUE	Numerical value of the data array.

5.5 Client Read (GET) Operations Supported

The FieldServer supports Active Read command that Reads an XML URL at a regular interval.

The following parameters are configured for this purpose:

Parameter	Description
XML-URL	Page to load on the Target Web Server.
Linked_Map_Descriptor	Active read command associated with that specific Map descriptor.
Element	XML Element needing to be stored. Below see an example: ***ELEMENT.CHIL_ELEMENT.GRAND_CHILD_ELEMENT.ATTRIBUTE*** NOTE: That if the last parameter is an Element, the Content of the Element will be stored.
Search_Value	A Unique attribute value that can be used to identify the ELEMENT required.

5.6 Client Write (POST) Operations Supported

The FieldServer supports POST commands via the WRITE operation within the FieldServer. The Post command relies on two parameters:

Parameter	Description
XML-URL	Page to load on the Target Web Server.
Write_Command	The Payload of the POST http packet.

MSA Safety
1991 Tarob Court, Milpitas, California 95305 USA
O. +1 408 262-6611 TF: +1 800 72 SIERRA (74377) E. SMC-insidesales@msasafety.com
www.sierramonitor.com