CHIPKIN

LonWorks FieldServers: Overview, Configuration, and Application

Article Highlights

This article serves as a comprehensive guide to LonWorks technology, offering detailed insights into various critical aspects. It begins with an extensive LonWorks overview, covering essential topics like network variable types, LonWorks addressing, network managers, and data transfer mechanisms. Furthermore, it delves into FieldServer configuration essentials, providing a thorough understanding of basic configuration files, commissioning processes, and structured variable types.

The discussion extends to network management applications, showcasing examples of configuration files for device identification, mapping, and domain/subnet control. Additionally, special case applications are explored, including working with user network variable types (UNVTs), LonMark standards, and function blocks. Whether you're a novice or an experienced user, this article offers valuable insights into understanding LonWorks technology for building automation and control systems.

Contents

1 LonWorks Overview

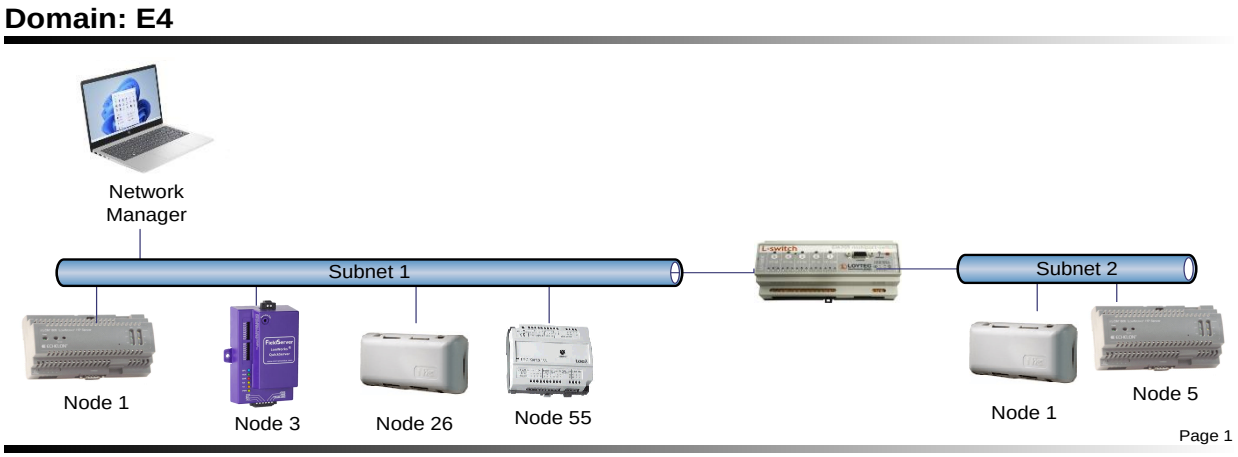
LonWorks is a communication protocol used in building automation and control networks. It enables devices from different manufacturers to communicate and work together within a network, facilitating centralized control and monitoring of building systems such as lighting, HVAC, and security.

1.1 Network Variable Types

In LonWorks networks, various types of network variables play crucial roles in facilitating communication and data exchange. These include:

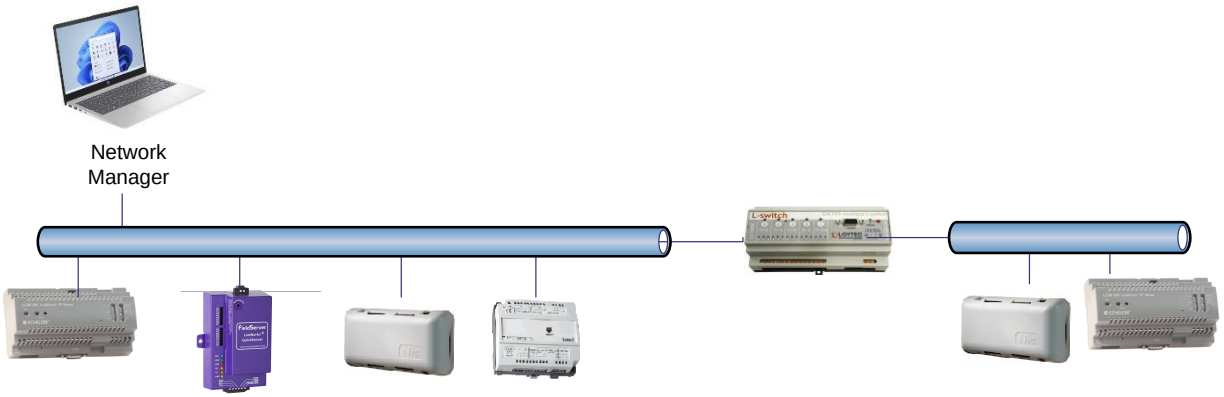
- SNVT – Standard Network Variable Type
- UNVT – User Network Variable Type
- SCPT – Standard Configuration Parameter Type
- UCPT – User Configuration Parameter Type

1.2 LonWorks addressing



- DSN – Domain, Subnet, Node.
- Typically, the Network Manager assigns the domain address for a network, following which the required subnets for a domain are determined by routers needed in the system. Node addresses on each subnet are then allocated by the Network Manager.
- The LNS report utility can print out all DSN addresses for a Domain.

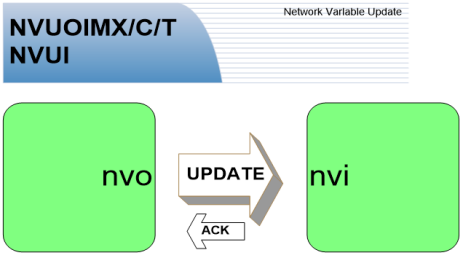
1.3 Network Managers



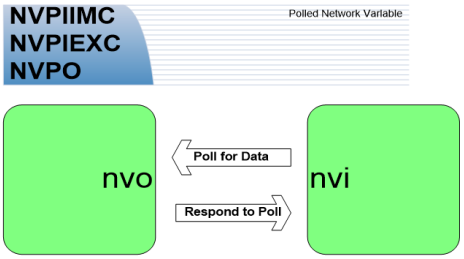
- LonMaker, LonWatcher, and others…
- Traditional LonWorks setup requires a network manager.
- Network manager "binds" data points together and can then be removed.
- Network Managers allocate addressing on the network.
- A LonWorks network can be commissioned without a Network Manager, but this requires more advanced techniques.

1.4 Data Transfer – Update vs Polled

- Update variables send data to a remote device using an event driven mechanism.
- Event driven mechanisms include Continuous update, Update on change, and Throttling mode.

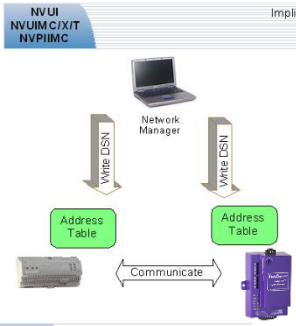


- Polled variables wait to be queried for data, and then respond when queried.
- Typically, polled variables operate on a continuous update methodology.

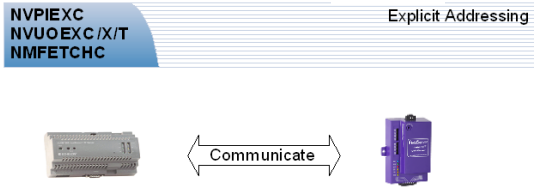


1.5 Data Transfer – Implicit vs Explicit

- With implicit addressing, the network manager assigns addresses for communication and ensures (via address tables in the devices) that communication connections are known.



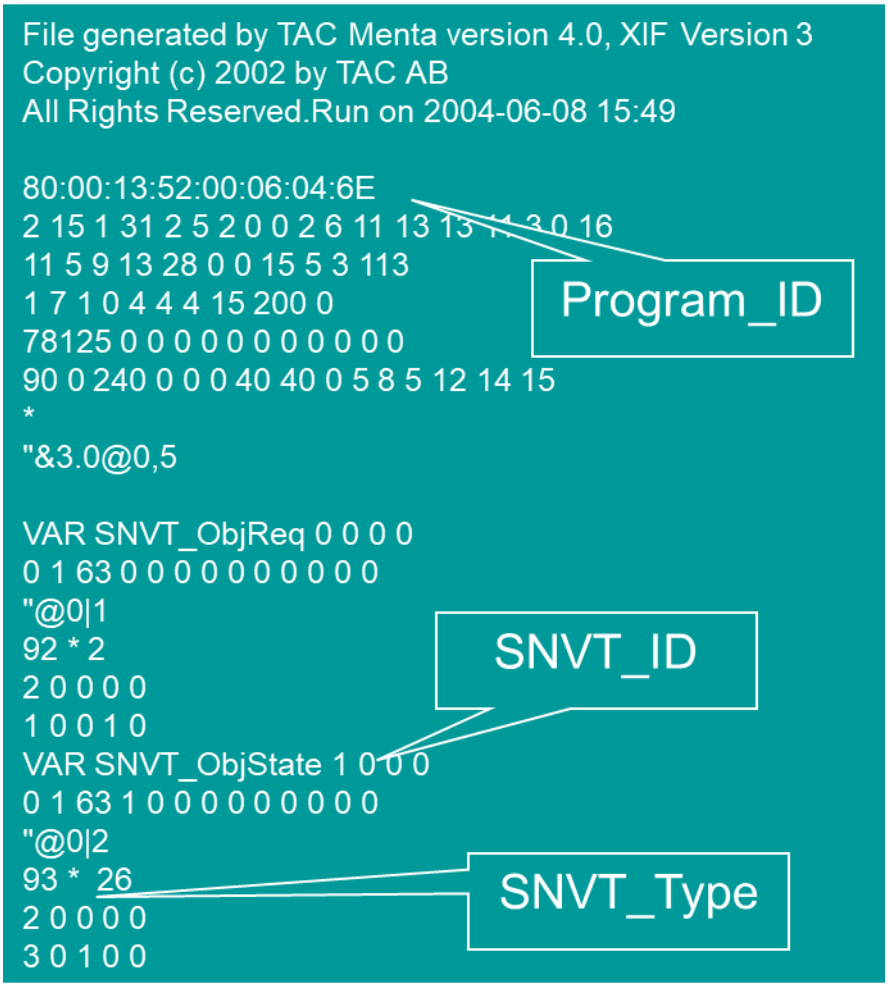
- With explicit addressing, the device knows the address of the point in the remote device and communicates directly without the assistance of the Network Manager.



1.6 XIF files

- XIF (external interface) files contain information about the variables on a device, and all the necessary variable properties.

- XIF files go a long way to providing all the information needed for FieldServer configuration, but do not contain DSN information.



1.7 Neuron ID's vs Program ID's

- A Neuron ID Uniquely defines the Neuron chip on the network. No two Neuron chips in the world have the same ID.
- A Program ID defines an application on a hardware platform, and it is possible for two hardware platforms to have the same Program ID because they are running the same application.
- If an application changes, then its Program ID must change too. This means when Lon variables are modified in the FieldServer configuration, the Program ID must change.
- If the Program ID changes for a device, then that device needs to be removed from a network, and then recommissioned.
- The Lesson to be learned: Plan your variable list carefully before commencing with binding.

2 FieldServer Configuration overview

2.1 Basic Configuration File (General Parameters)

Define Title and Data Array:

//==		
// FieldServer Information		
Bridge		
Title		
EC001 FieldServer v1.00a		
//==		
// Data Arrays		
Data_Arrays		
Data_Array_Name	Data_Format	Data_Array_Length
DA_Temps	FLOAT	10

2.2 Basic Configuration File (Connection Information)

Define Connection and Node:

//==========	
// LonWorks Connection	
Connections	
Adapter	Protocol
Lonworks	Lonworks
//==========	
// LonWorks Node	
Nodes	
Node_Name	Protocol
EC001	Lonworks

2.3 Basic Configuration File (Mapping)

Define SNVT's (Map them to the Data Array):

//==========						
// LonWorks Map Descriptors						
Map_Descriptors						
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Lon_Function	Function	Node_Name	SNVT_Type
nviSpaceTmp	DA_Temps	0	NVUI	Server	EC001	SNVT_temp_p
nvoSpaceTmp	DA_Temps	1	NVUOIMX	wrbx	EC001	SNVT_temp_p

2.4 XIF for Basic Configuration File

fserver.xif:

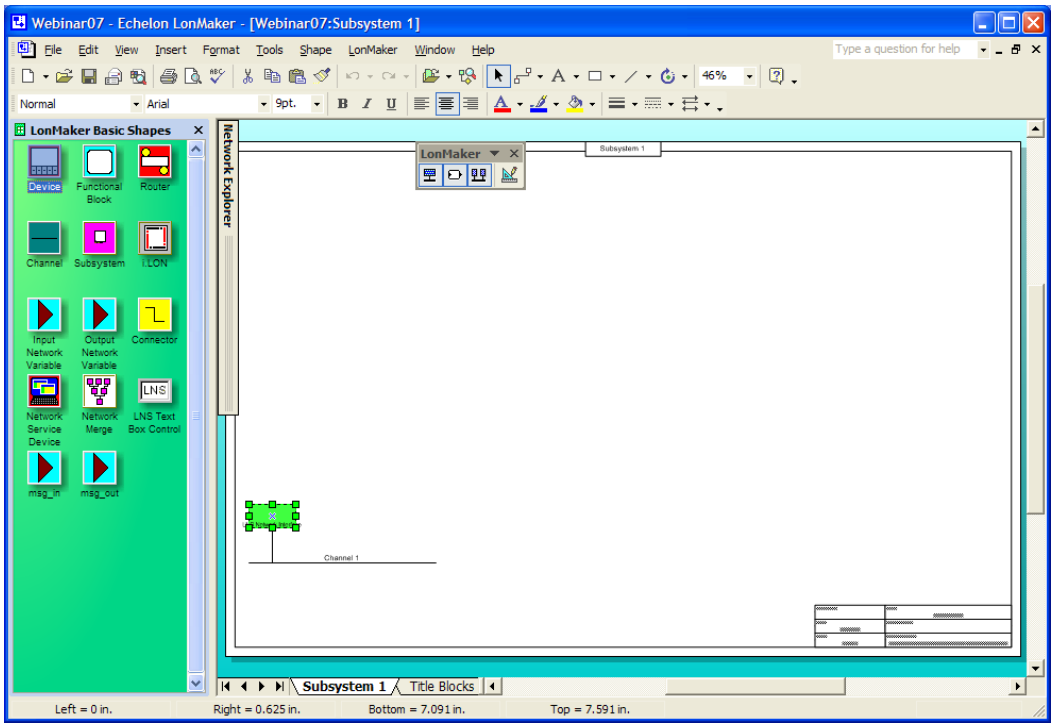
```
File: fserver-2.xif generated by LonDriver Revision 1.20(g), XIF Version 4.0
Copyright (c) 2000-2004 by FieldServer Technologies
All Rights Reserved.  Run on Tue Jul 26 07:58:26 2005
80:00:95:47:1E:02:04:36
2 15 1 2 0 14 11 3 3 12 14 11 11 11 11 3 0 16 63 0 1 11 2
0 5 7 13 28 0 0 15 5 3 109 63
1 7 1 0 4 4 4 15 200 0
78125 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 5 8 5 12 14 15
*
"EC001 FieldServer v1.00a
VAR nviSpaceTmp 0 0 0 0
0 1 63 0 0 0 0 0 0 0 0 0
*
105 * 1
2 0 0 1 0
VAR nvoSpaceTmp 1 0 0 0
0 1 63 1 0 0 0 0 0 0 0 0
*
105 * 1
2 0 0 1 0
```

2.5 Commissioning EC001 into LonMaker

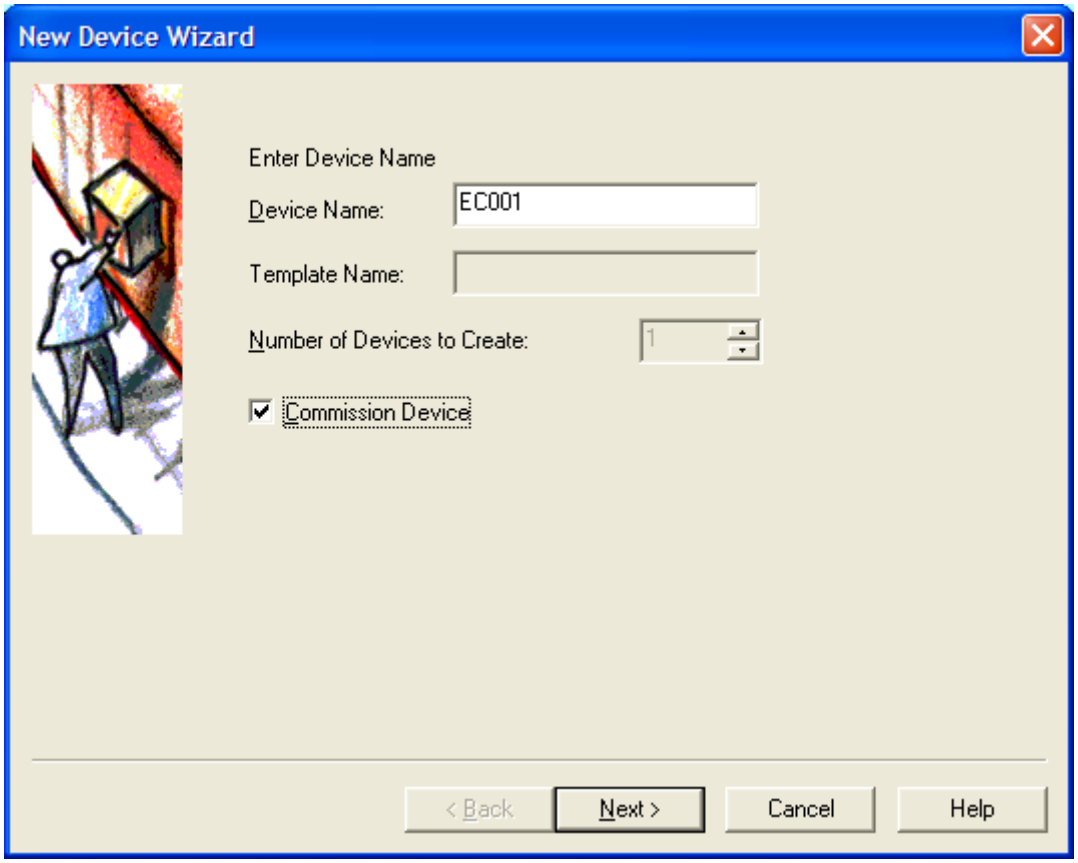
Step 1: Load config into FieldServer and reboot FieldServer

Step 2: Open a project in LonMaker

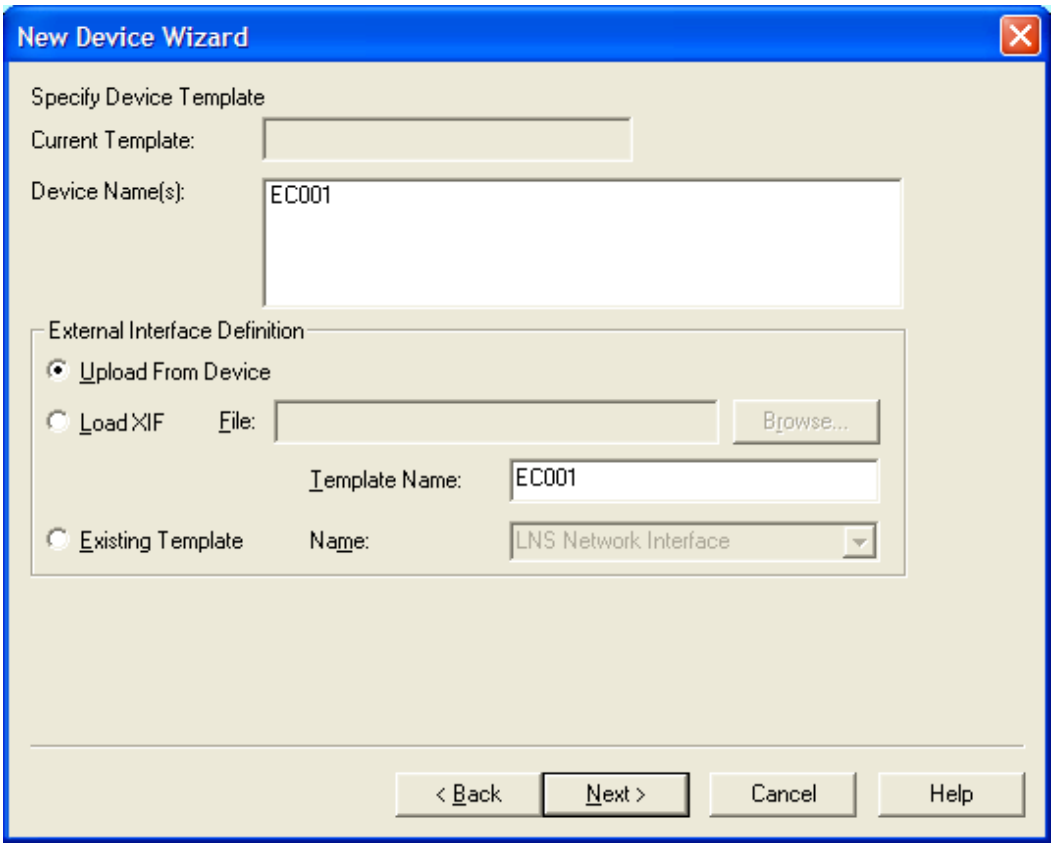
Step 3: Drag device into Visio drawing



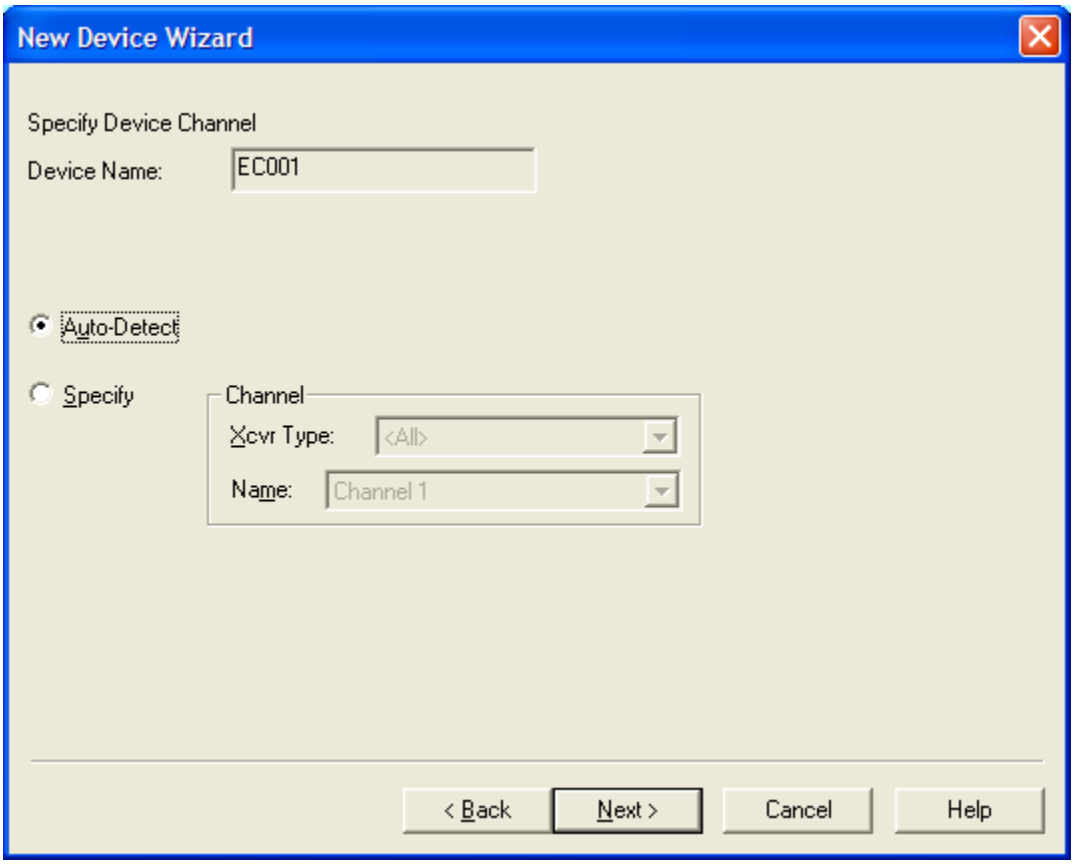
Step 4: Give the device a name & select the “commission device” option



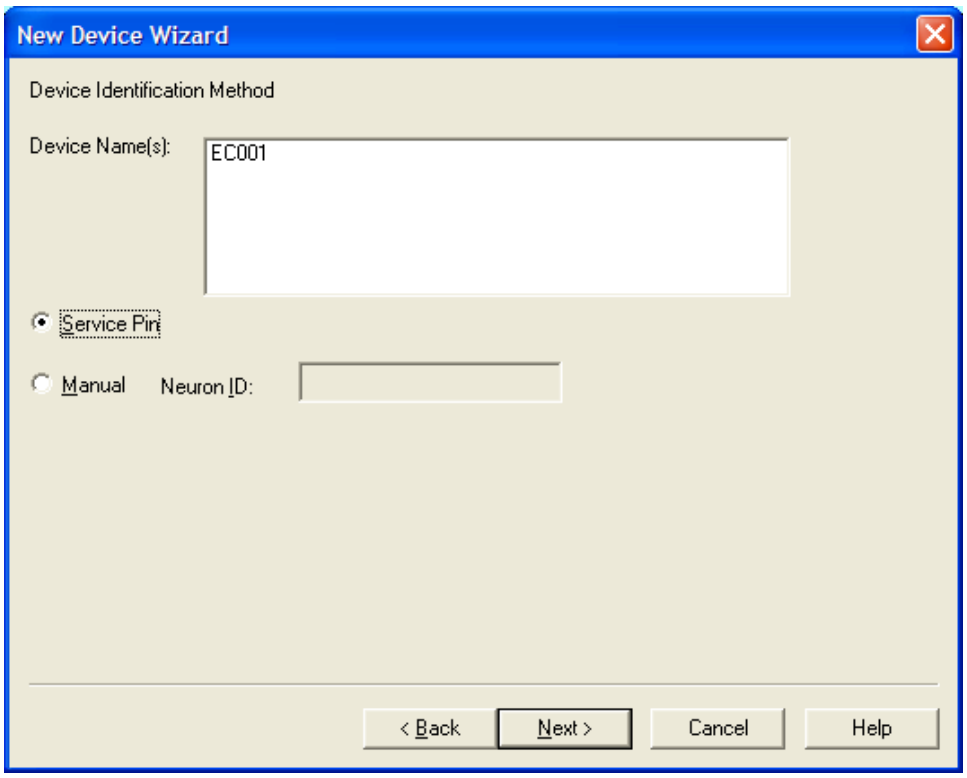
Step 5: For XIF definition, select “Upload from Device”



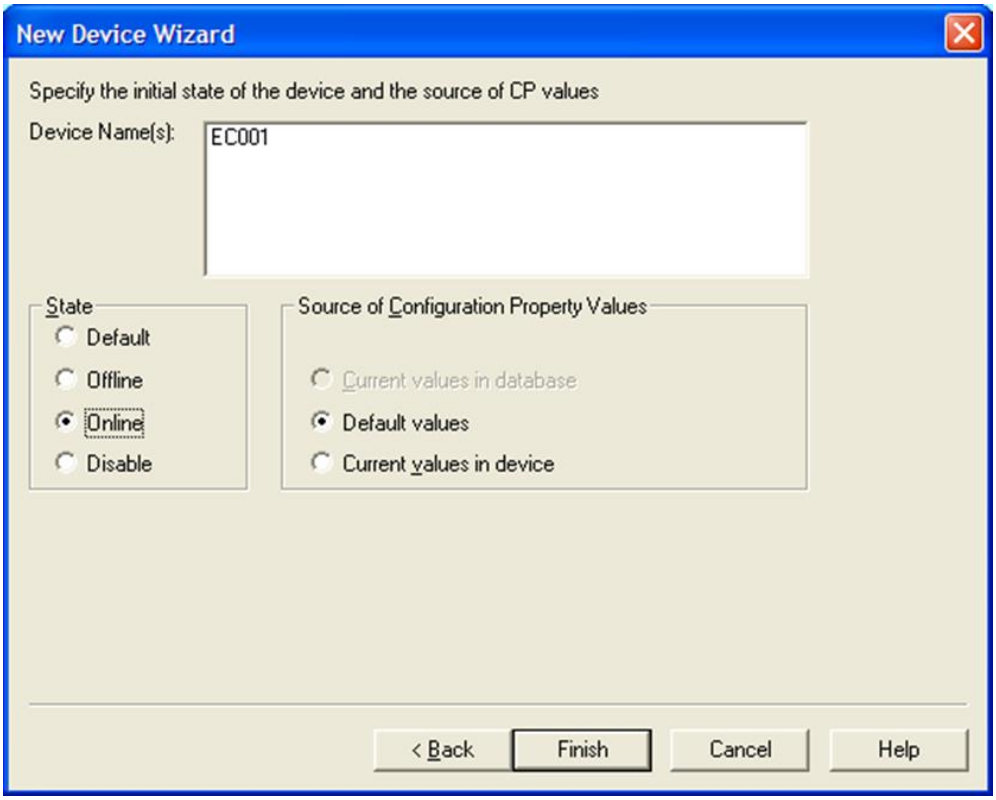
Step 6: For Device Channel, select “Auto Detect”



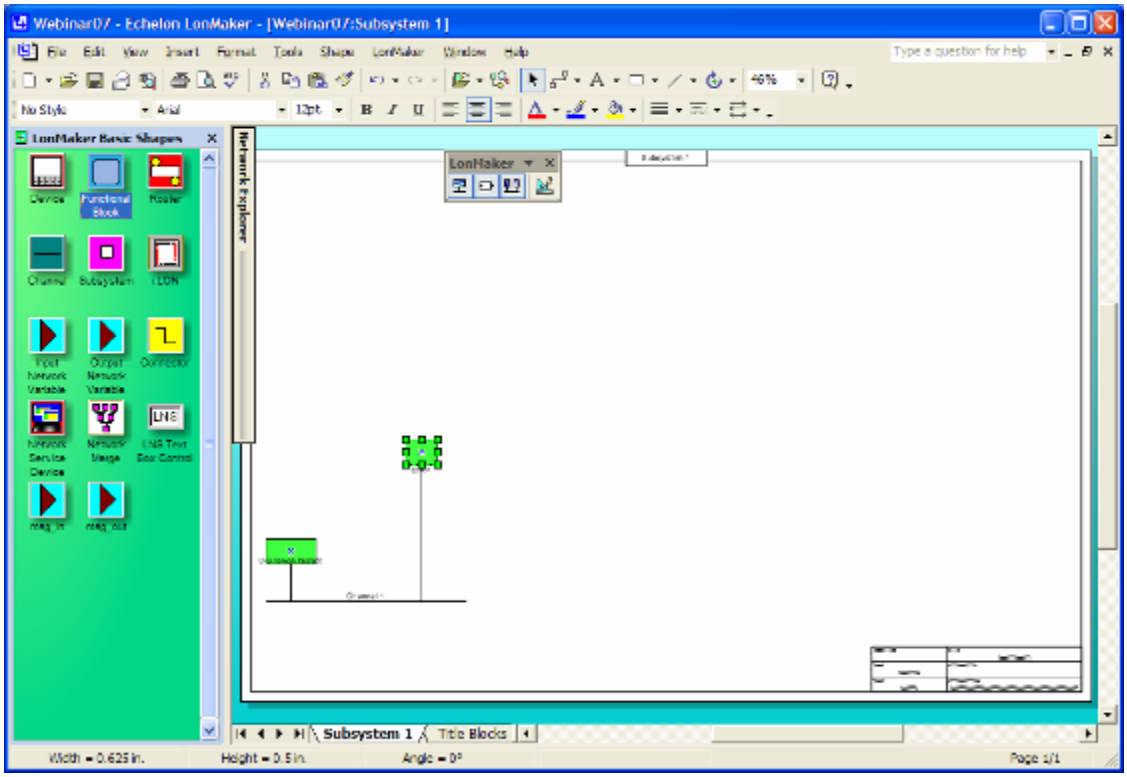
Step 7: Leave “Device Properties” as is
Step 8: For Device Identification Method, select “Service Pin”



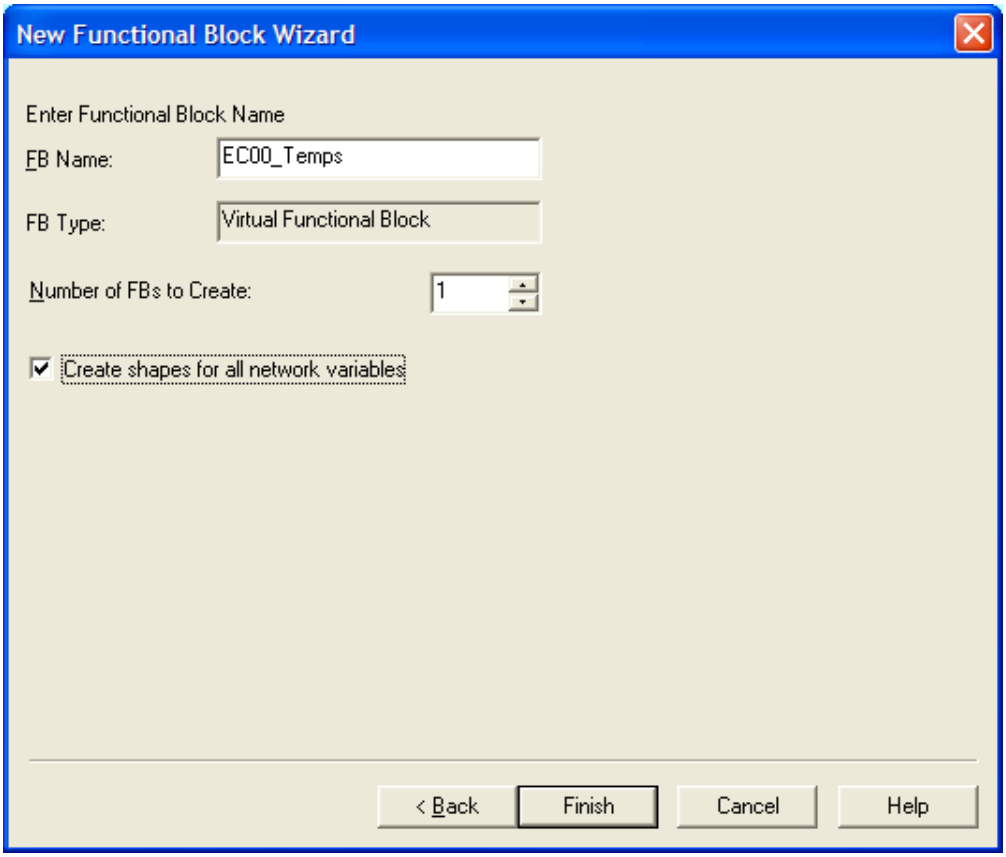
Step 9: Do not “Load Application Image” (Leave unchecked)
Step 10: For State, select “Online” and leave the rest as is:



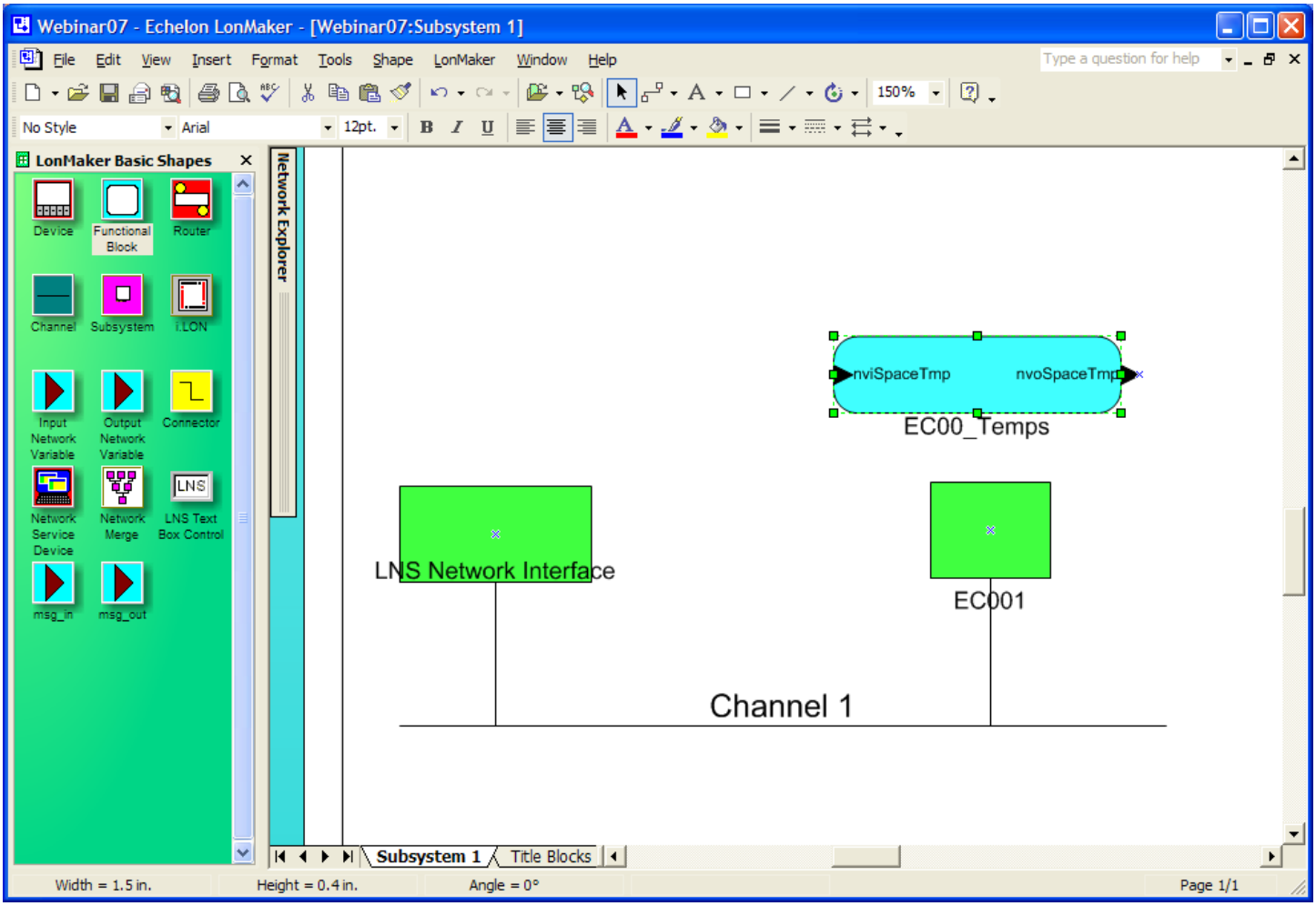
Step11: Press the Service Pin on the FieldServer, wait for EC001 to commission.
Step 12: Drag a Function Block into Visio



Step13: Leave the first menu as is with “Virtual Function Block” selected. Click on “Next”.
Step 14: Give the FB a Name, and Select “Create shapes for all network variables”



The end result: You are now ready to bind to other devices using the “Connector” Tool.



2.6 Structured Variable Types

Allow for 2 points in the Data Array

//Structured SNVT							
Map_Descriptors							
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Function	Node_N...	Lon_Function	SNVT_Index	SNVT_Type
nvoRhEnable	DA_AO_01	1	rdbc	Lon_01	NMFETCHC	9	SNVT_switch
nvoUnOccRh	DA_AO_01	3	rdbc	Lon_01	NMFETCHC	10	SNVT_lev_percent

- Structured Variable Types pack more than one variable into the variable type
- SNVT_switch, for example, has two variables associated with it: Value (0-100) and state (0/1).
- The FieldServer can extract all variables from a structured type. Some are easy to work with, others are not.
- Note: while extracting ASCII characters out of a structured SNVT is possible, it does not make much sense in the “Gateway” model due to limitations of other protocols.

2.7 Selective transfer of Structured Variables

//==========							
// Temperature Mapping in Deg_F							
Map_Descriptors							
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Lon_Function	Function	Node_ Name	SNVT_Type	SNVT_Option
nviAlarm_1	DA_Binary_State	0	NVUI	Server	EC001	SNVT_switch	State
nvoAck_1	DA_Binary_State	1	NVUOIMX	wrbx	EC001	SNVT_switch	State

Load only the value needed

- When working with a Structured Variable, FieldServer allows you to fetch just the portion of the structured variable needed by declaring the SNVT_Option field.

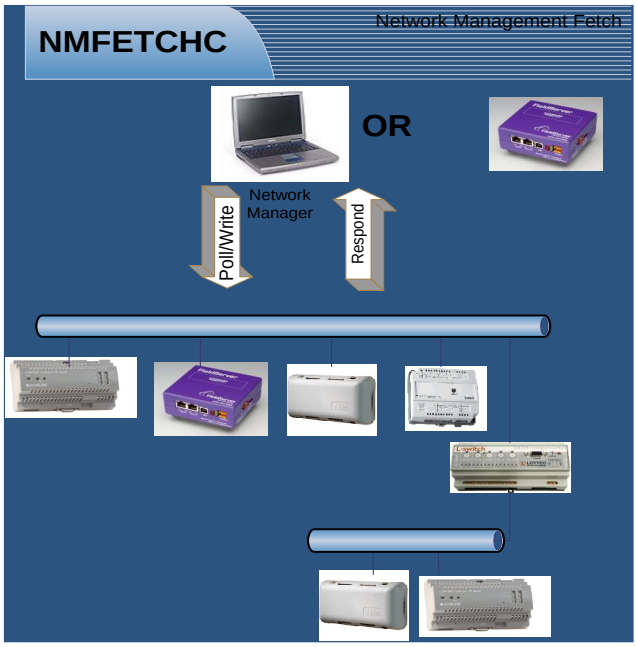
2.8 SNVT and UNVT Scaling

//==========							
// Temperature Mapping in Deg_F							
Map_Descriptors							
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Lon_Function	Function	Node_ Name	SNVT_Type	SNVT_Units
nviSpaceTmp	DA_Temps	0	NVUI	Server	EC001	SNVT_temp_p	Deg_F
nvoSpaceTmp	DA_Temps	1	NVUOIMX	wrbx	EC001	SNVT_temp_p	Deg_F

Scale Celsius to Fahrenheit

- All SNVT's include required scaling in their definition.
- The FieldServer automatically scales the raw data according to the SNVT sca ling rules. Value in Data Array is the scaled value.
- Note that LonWorks is inherently METRIC. Thus, non-metric applications still require extra parameters in the configuration (SNVT_Units field)
- When working with UNVT's, raw data is received in a byte Data Array. Required scaling and byte concatenation needs to be done in the configuration.

3 Network Management Application



3.1 Data Transfer Mechanism

- This is a special function intended for use by host systems that need to monitor the status of variables on a network (e.g: Network Managers)
- This function uses explicit polling to obtain the needed variables.
- It is possible to write data using this function too
- This function has some very useful applications when integrating the FieldServer into larger networks

3.2 Example Configuration File (Identify Devices)

```
//=============================================
//  Common Information
Bridge
Title
NMManager

//=============================================
//
//  Remote Nodes
Nodes
Node_name       Subnet_ID       Node_ID       Adapter       Protocol
Lon_01          1               1             Lonworks      Lonworks
```

Subnet on which the remote device resides

Node Address of the remote device

3.3 Example Configuration (Forcing Domain and Subnet)

```
//=============================================
//  Common Information
Bridge
Title                   System_Address
":D48:S01:NMManager"    11
```

Force FieldServer address to Domain 48 (hex), Subnet 1.

Node Address of the FieldServer

//==========				
//				
// Remote Nodes				
Nodes				
Node_name	Subnet_ID	Node_ID	Adapter	Protocol
Lon_01	1	1	LonWorks	LonWorks

Subnet on which the remote device resides

Node Address of the remote device

3.4 Working with UNVT's

Once you have the data in the data array, you need to know what it means

Need to know Byte Length of UNVT

//UNVT								
Map_Descriptors								
Map_Descriptor_Name	Data_Array_Name	Dat…	Lon_Function	Function	Node_Name	SNVT_index	SNVT_Type	…VT_byte_length
D1nvoBoilerEnable	DA_BI_01	3	NMFETCHC	rdbc	Lon_1	46	SNVT_switch	
D1nvoInUse	DA_AI_01	5	NMFETCHC	rdbc	Lon_1	22	UNVT	1
D1nvoManValue	DA_AI_01	6	NMFETCHC	rdbc	Lon_1	23	UNVT	4
D1nvoModBoilrShare	DA_AI_01	10	NMFETCHC	rdbc	Lon_1	30	UNVT	6

- By definition, these are USER defined, so a road map of the UNVT structure is needed before one can do anything useful with it.
- A road map must contain byte length of UNVT, as well as the breakdown of what each byte represents.

3.5 LonMark

- LonMark is the controlling body for network standards in the LonWorks environment.
- LonMark's objective is promoting interoperability and standardized interfaces.
- LonMark certification for non-LonWorks devices is possible through FieldServer.

LONMARK International

LonMark configurations require special parameters.

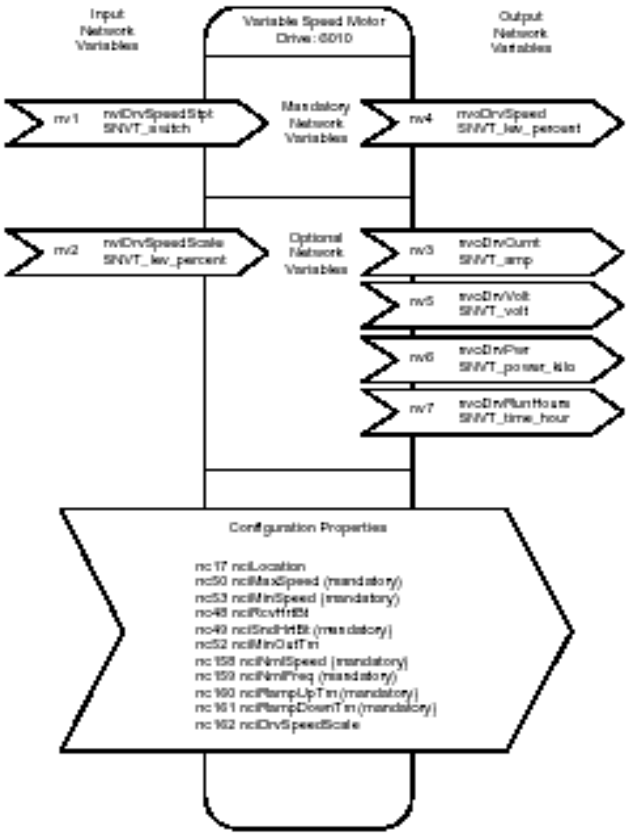
//==========	
//	
// Common Information	
Bridge	
Title	System_Station_Address
&3.2@13110EMCP2;FS	1

Map_Descriptors					
Map_Descriptor_Name	….	Node_Name	SNVT_Type	SNVT_Option	Nv_Selfdoc_text
nviStartCmd		Lon_Srv_1	SNVT_switch	state	"@0|01;StartCmd"
nviFaultResetCmd		Lon_Srv_1	SNVT_switch	state	"@0|02;FaultResetCmd"

Due to the complexity of LonMark configurations, enlisting the assistance of FieldServer Technologies is advised.

3.6 Function Blocks

- Function Blocks help separate data within a device for more convenient binding.
- Allocation of Lon variables to function blocks is possible via configuration in the FieldServer.



3.7 Function Blocks (Config Example)

//==========	
// Common Information	
//	
Bridge	
Title	System_Address
&3.2@0Node,520[25]Out,521[25]In;FS	11

This title declares 25 FB's called "Out", and 25 FB's called "In". LonMark profiles are 520 and 521.

//							
// FB Map							
Map_Descriptors							
Map_Descriptor_Name	Data_Array_Name	Data_Array_Offset	Lon_Function	Funct...	Node_Name	SNVT_Type	Nv_Selfdoc_text
nvoAnalog_01	Analog_Out	0	NVUOIMC	WRBC	Lon_Srv_1	SNVT_lev_percent	@1|01;-
nvoAmp1_01	Out_01	0	NVUOIMC	WRBC	Lon_Srv_1	SNVT_amp	@1#01;-
nvoAmp2_01	Out_01	1	NVUOIMC	WRBC	Lon_Srv_1	SNVT_amp	@1#01;-

First Map Descriptor Declares the FB. The next 2 declare the variables in the FB

The Self Documentation text is used to declare a variables association with a Function Block.