

Numera Libris Installation Guide

09/13/2024 10:33 am EDT

Introduction

Numera Libris is a GPS mobile device that provides alarm messages and plots the data on a map in the "Libris" portal. It does not contain any client information. However, when integrated with Manitou using the Bold UniversalConnector, it works in tandem to provide a great deal of information for the operator/agent responding to the alarm.

For more information on the Numera Libris product, please visit <http://numera.com/libris/>.

Requirements

This guide assumes that the MediaGateway and PBX modules have already been installed and configured. Please see the following guides for further information:

- [MediaGateway Installation Guide](#)
- [PBX Installation Guide](#)

Options

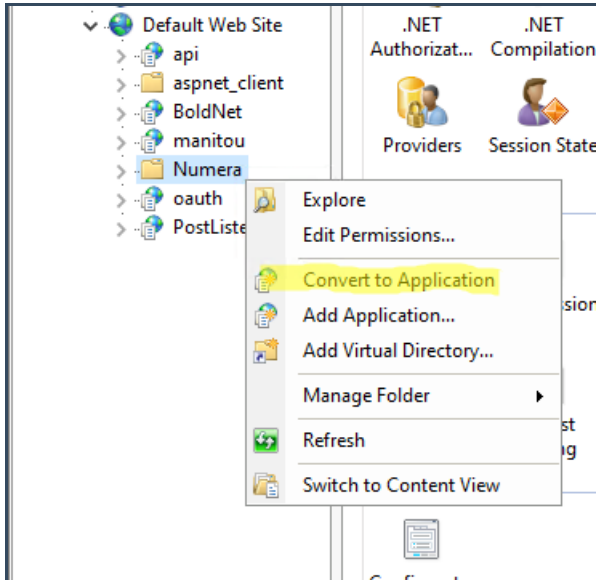
- TwoWayPLUS (specifically InstantConnect)
- BoldTrak GPS (to see latitude/longitude coordinates in Manitou – including historical data)
- Bold XML Receiver configured for the GPS MediaGateway.
- Manitou Transmitter Type for GPS system with Raw Programming enabled

Installation

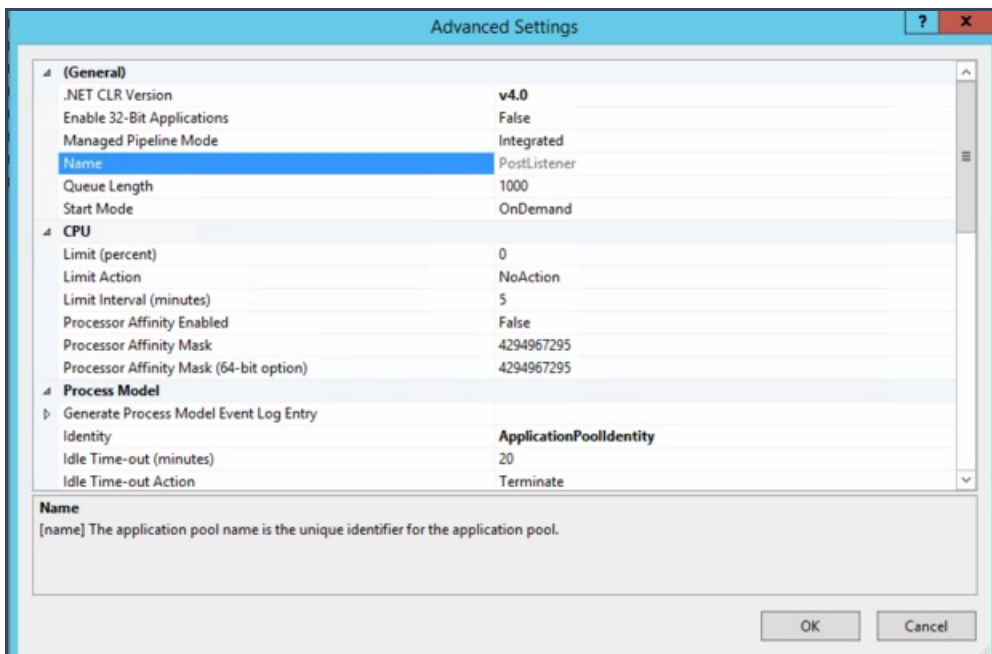
PutListener Setup

- Download and extract the "NumeraPutListener.zip" at the bottom of this guide. Rename the "PostListener" folder to "Numera" and copy it to the customer's IIS. Place it in the C:\inetpub\wwwroot\ directory.
 - For cloud customers, the PostListener folder will need to be setup on one of the hosted BoldNet servers and placed in the customer's specific wwwroot subdirectory (ie. C:\inetpub\wwwroot\
- In the Numera folder, make the following changes to the "POSTlistener.cshtml" file:
 - On the line that starts with "const string BASE_FILE_PATH", change the target folder to "C:\\PostListen\\NUMERA\\"; const string BASE_FILE_PATH = "C:\\PostListen\\NUMERA\\";
 - For cloud customers you will need to include the customer's PostListen subdirectory (ie. "C:\\PostListen\\<customer_name>\\NUMERA\\")
 - Towards the bottom of the file set the target log folder so that it looks like this: using (System.IO.StreamWriter file = File.AppendText(string.Format("c:\\PostListen\\NUMERA\\Log\\HTTPLListener_Error_Log.txt")))

- Cloud customers should look like: using (System.IO.StreamWriter file = File.AppendText(string.Format("c:\\PostListen\\<customer_name>\\NUMERA\\Log\\HTTPListener_Error_Log.txt")))
- Create the C:\\PostListen\\Numera\\ and C:\\PostListen\\Numera\\Log\\ folders on the web server.
- In IIS, underneath the "Default Web Site", right click the "Numera" folder and select "Convert to Application"

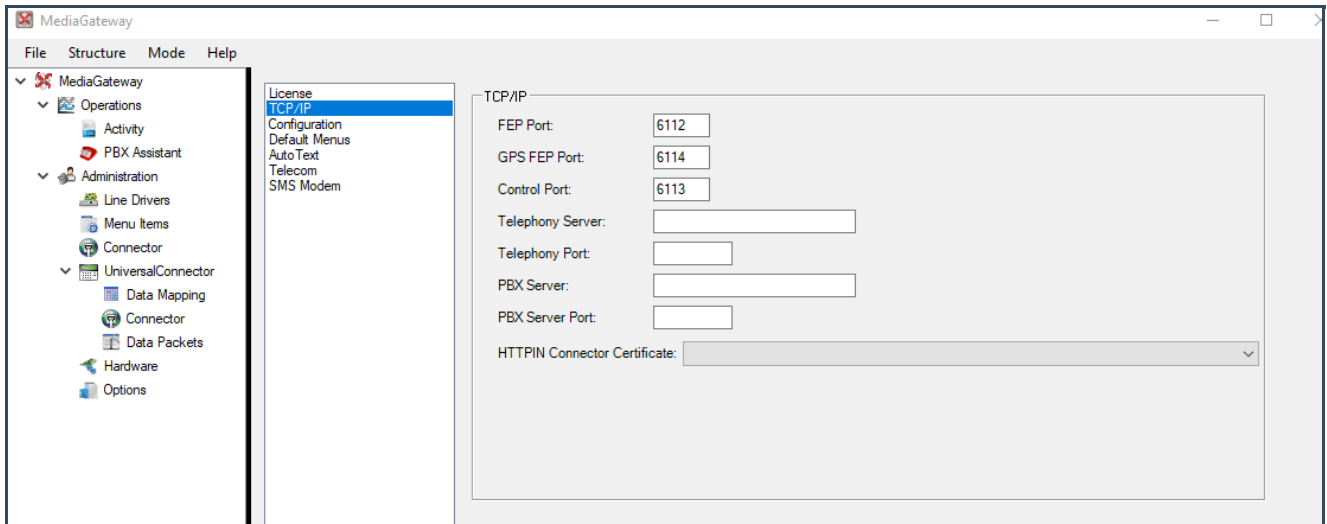


- Set the app pool for the PutListener to be the following:



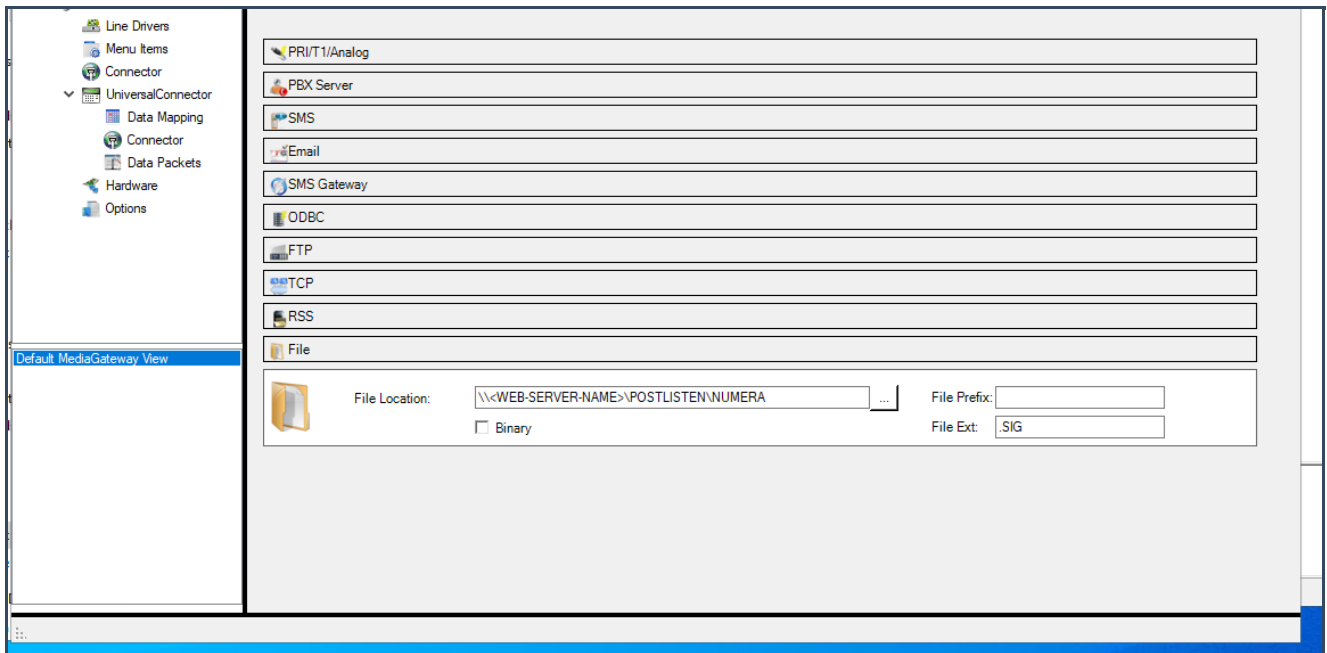
MediaGateway Configuration

- Confirm the GPS FEP Port is set to 6114 in the MediaGateway Console > Options > TCP/IP page.



File Connector

- Within the MediaGateway Console, navigate to the Connector screen.
- Select the row labeled "File" and click "Add".
- Name the new Connector 'Numera Libris'. Make sure Connector Type is set to "File" and click "OK".
- Set the File Location to "\\<WEB-SERVER-NAME>\Postlisten\Numera".
 - Cloud customers will use "\\<WebServerName>\PostListen\<CustomerFolder>\Numera"
- Make sure the File Ext. is set to .SIG



- Click Update at the top, and then click File > Save.

Data Map

- Click File > Data Mapping > Import and import the Data Map included in this guide.
- Click File > Save.

Line Driver

- Select the new "Numera Libris" Connector in the Line Driver drop down column. Set the Line Function to UniversalConnector. Right click the Properties field and set Menu = UCSEND, FIELDSET = LIBRIS-JSON, and set the FEP, RECEIVER, Line fields depending on the customer configuration.
- Click the red circle in the Driver column and start the driver.
- Click File > Save,

Receiver Line Prefix

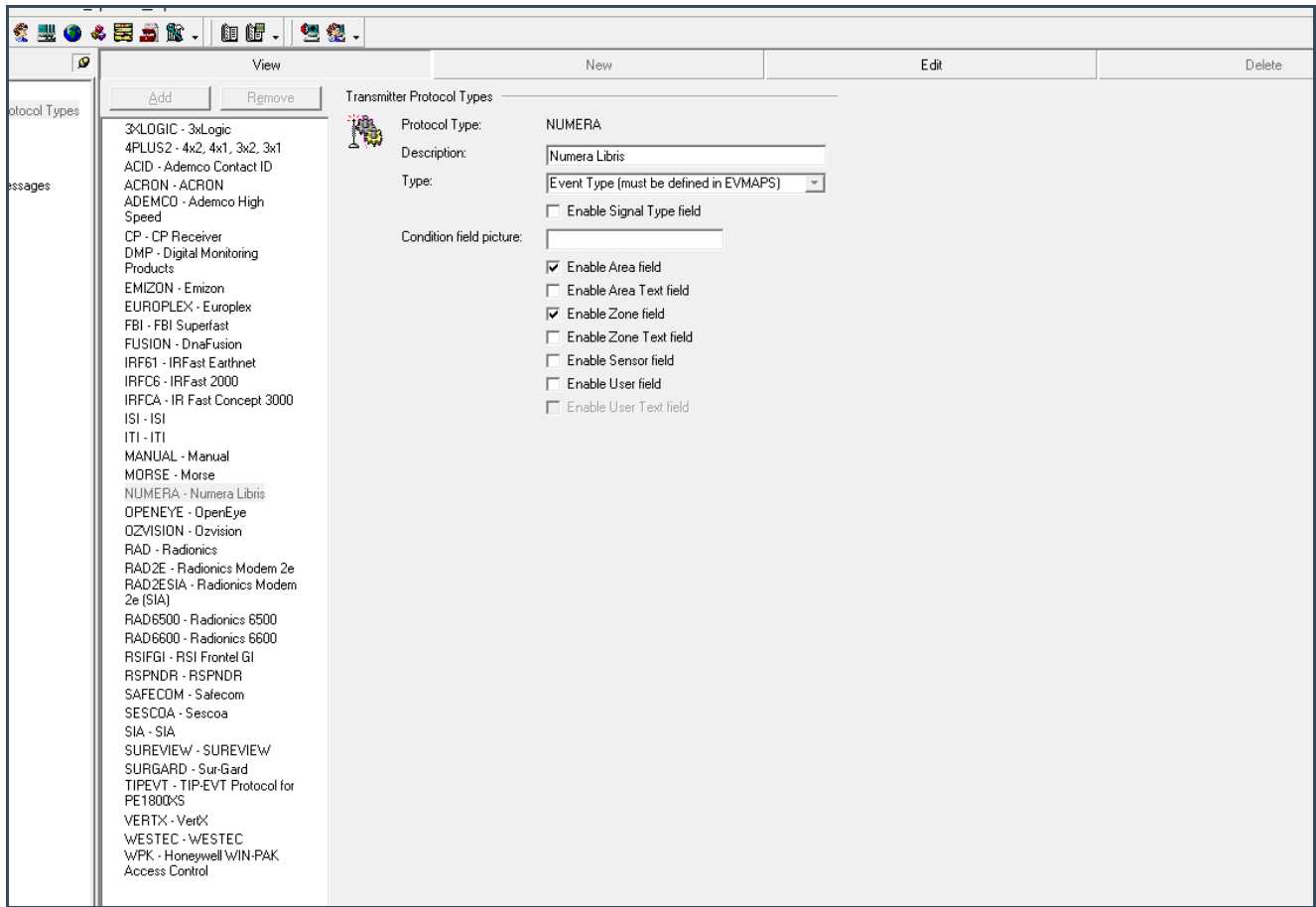
- Within Supervisor Workstation, navigate to Maintenance > Setup > Receivers
- Select Edit at the top, then select "Rec Line Prefixes" and click Add.
- Create a new Receiver Line Prefix and Description and click OK.
- Click Save at the top of the screen.

MediaGateway GPS Receiver Driver

- If there is no existing GPS Driver, create one using the Bold XML Receiver Type and the port set to 6114.
- Add the newly created Numera RLP to the GPS Receiver Driver.

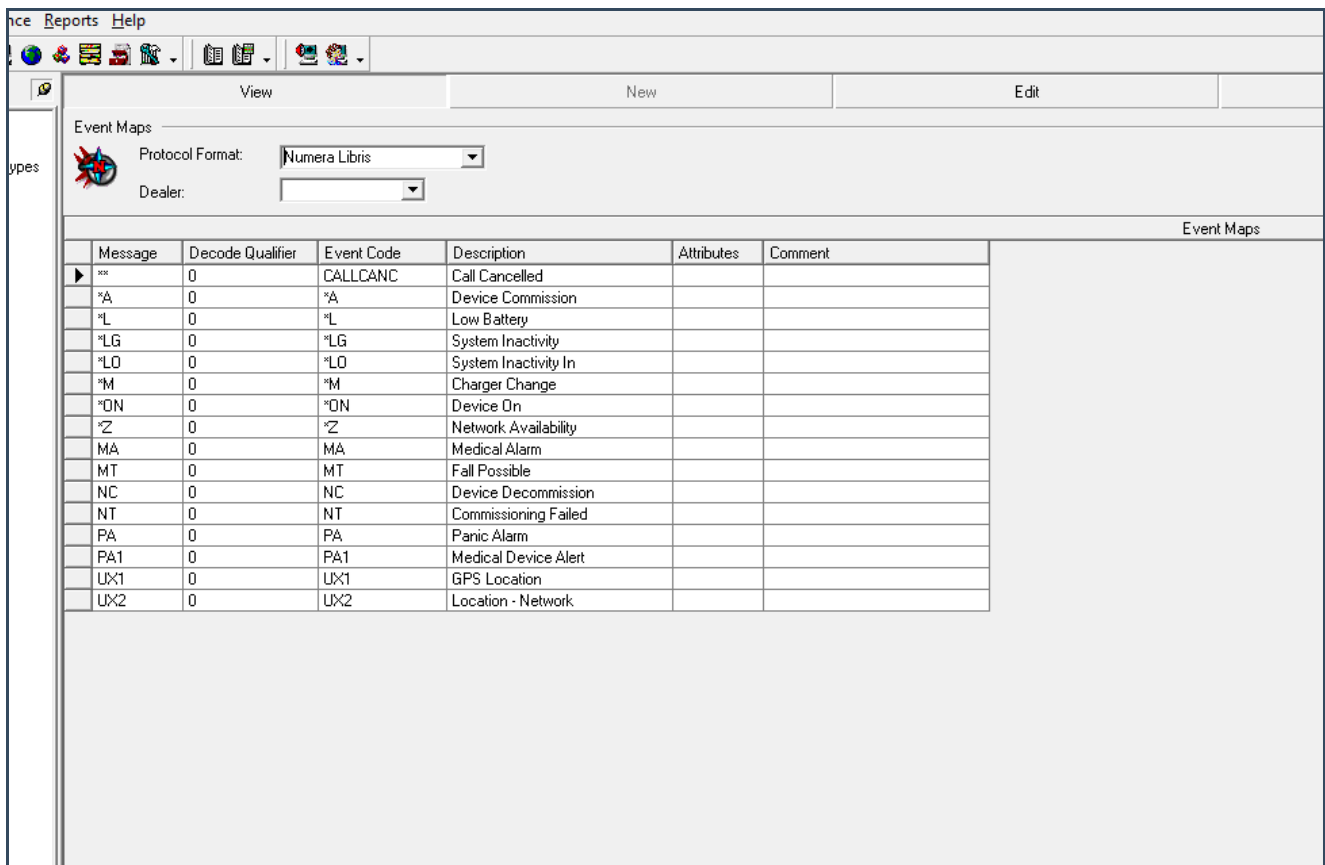
Transmitter Protocol Type

- In Supervisor Workstation, navigate to Maintenance > Setup > Transmitter Protocol Types.
- Select Edit at the top of the screen and click Add to create a new Numera Protocol Type:



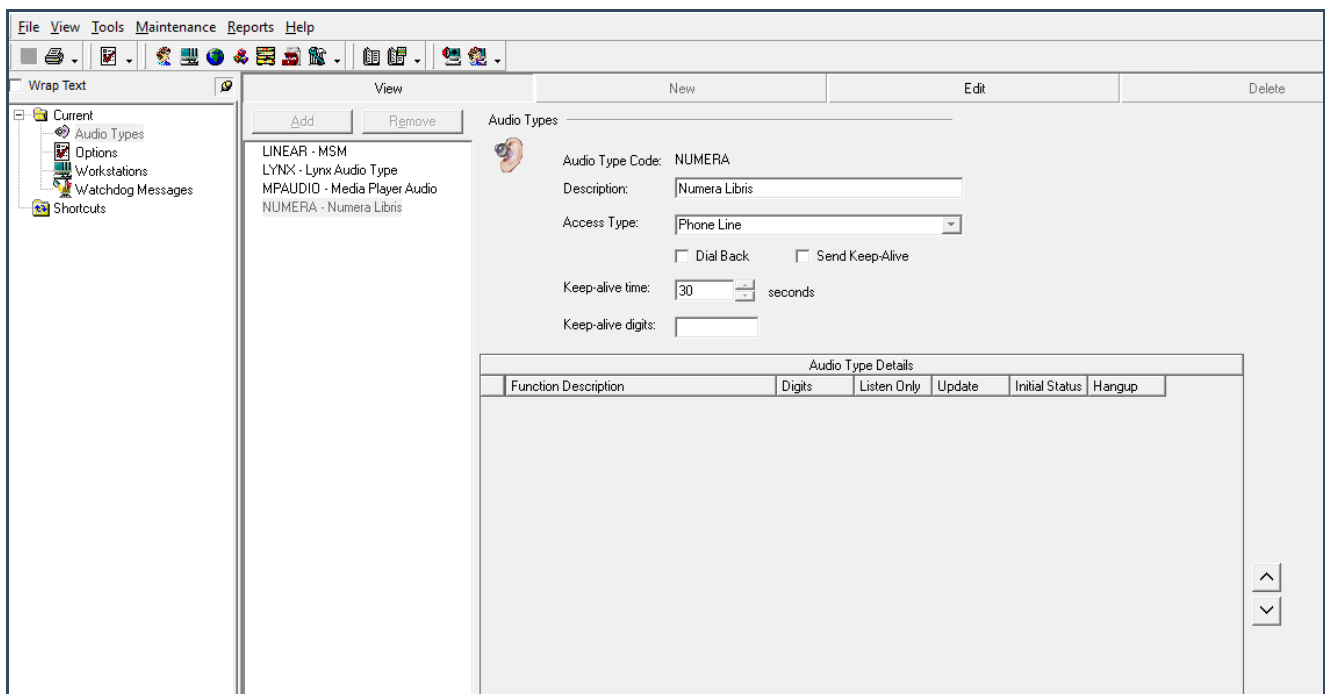
Event Map

- Navigate to Maintenance > Events > Event Maps.
- With the Protocol Format dropdown, select the newly created Numera Libris protocol.
- Select Edit at the top of the screen and add the following event mappings:



Audio Type

- Navigate to Maintenance > Setup > Audio Types.
- Select Edit at the top of the screen and click Add to add the following Audio Type:



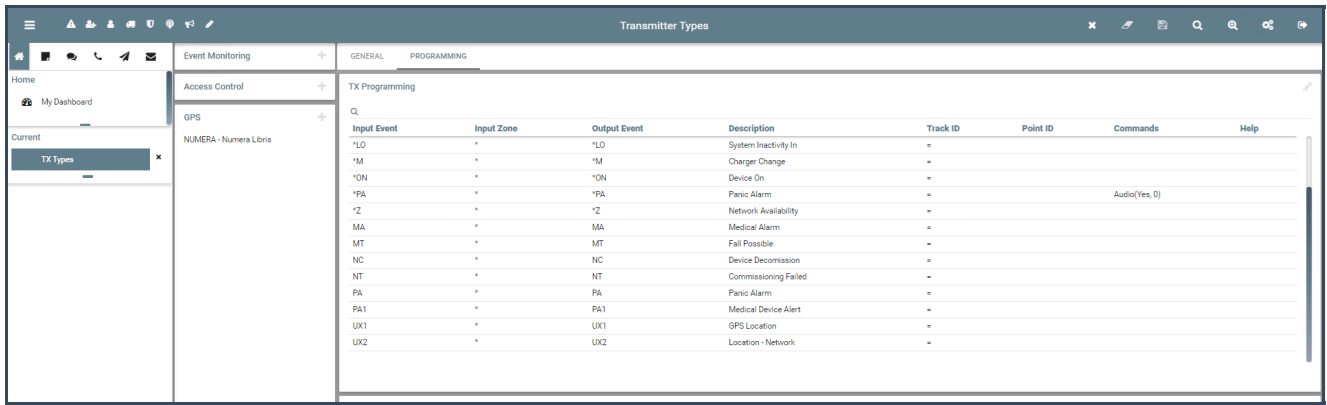
Transmitter Type

- In the Manitou Web Client, click on the hamburger menu at the top left and navigate to Maintenance > TX Types.
- Click the "+" next to GPS towards the bottom of the screen and create a new "Numera" Transmitter Type.
- Configure the new transmitter type with the following:

The screenshot shows the 'Transmitter Types' configuration page. The left sidebar contains navigation options: Home, My Dashboard, Current, TX Types (selected), NUMERATEST - Numera Tes..., numera, and Customer Search. The main content area has tabs for 'GENERAL' and 'PROGRAMMING'. Under 'GENERAL', the transmitter is named 'NUMERA - Numera Libris (GPS)' with a 'Protocol Type' of 'Numera Libris' and a 'Reverse Command Protocol'. The 'TX ID Input Mask' section has three groups, each set to 'Decimal'. The 'Audio Type' is 'Media Player Audio', with 'Audio Capable' checked and 'Create Call Session (No Listen-In)' and 'Drop Listen-In if no alarm' unchecked. The 'Video Type' has 'Video Capable' unchecked. The 'Options' section has 'Raw Event Programming' checked, and 'Monitored Transmission Path', 'Generate Late to Test only when Closed', and 'Locatable Device' unchecked.

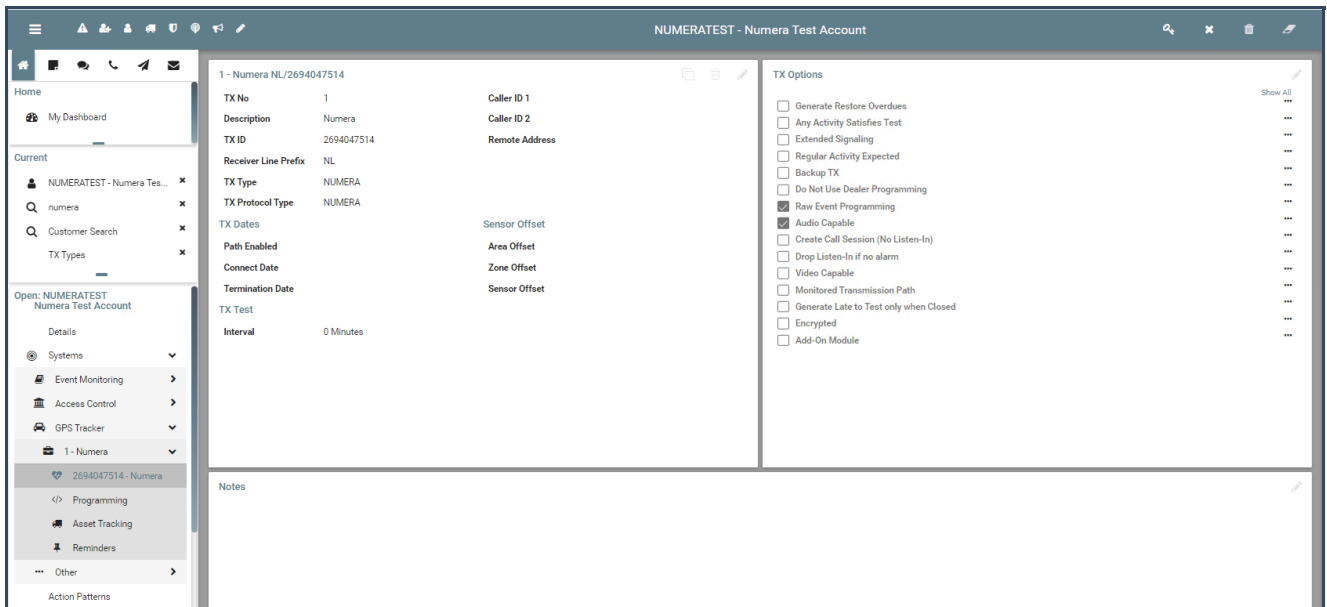
The screenshot shows the 'TX Programming' table for the 'NUMERA - Numera Libris' transmitter type. The table has columns for 'Input Event', 'Input Zone', 'Output Event', 'Description', 'Track ID', 'Point ID', 'Commands', and 'Help'. The table contains the following data:

Input Event	Input Zone	Output Event	Description	Track ID	Point ID	Commands	Help
**	*	CALLCANC	Call Cancelled	=			
*A	*	*A	Device Commission	=			
*L	*	*L	Low Battery	=			
*LG	*	*LG	System Inactivity	=			
*LO	*	*LO	System Inactivity In	=			
*M	*	*M	Charger Change	=			
*ON	*	*ON	Device On	=			
*PA	*	*PA	Panic Alarm	=		Audio(Yes, 0)	
*Z	*	*Z	Network Availability	=			
MA	*	MA	Medical Alarm	=			
MT	*	MT	Fail Possible	=			
NC	*	NC	Device Decommission	=			
NT	*	NT	Commissioning Failed	=			
PA	*	PA	Panic Alarm	=			



Test Account Transmitter Setup

- Ensure the Customer GPS transmitter has the RLP set to the new one we created, and TX Type / TX Protocol Type are set to Numera. Lastly ensure both Raw Event Programming and Audio Capable enabled.



Numera will need the following information:

- The URL endpoint for the monitoring center's IIS server – typically similar to <http://monitoringcenter.com/POSTlistener/POSTlistener.cshhtml>
- The monitoring center's phone number/hunt group for the inbound calls (regardless of whether they are using TwoWayPLUS)

Putlistener Whitelisting

- When testing with Numera, we will need to whitelist the signal source IP's/hostnames. Any blocked IP's/hostnames are logged at c:\PostListen\NUMERA\Log\HTTPListener_Error_Log.txt.

```
*HTTPListener_Error_Log.txt - Notepad
File Edit Format View Help
Request blocked from [172.30.190.70]
Request blocked from [131.140.250.156]
Request blocked from [118.29.113.79]
Request blocked from [manitoutest.com]
```

- Within the postlistener.cshtml file, add a line for each source IP/hostname to whitelist as shown below:

```
1 @functions {
2     public string LogRequest()
3     {
4         List<Tuple<string, string>> hostMapping = new List<Tuple<string, string>>();
5
6         const string BASE_FILE_PATH = "C:\\Nortek\\Numera\\";
7
8
9         hostMapping.Add(new Tuple<string, string>("118.29.113.79", "Numera-Libris"));
10        hostMapping.Add(new Tuple<string, string>("172.30.190.70", "Numera-Libris"));
11        hostMapping.Add(new Tuple<string, string>("131.140.250.156", "Numera-Libris"));
12        hostMapping.Add(new Tuple<string, string>("manitoutest.com", "Numera-Libris"));
13
14
15        string hostName = "";
16        try
17        {
18            hostName = Dns.GetHostEntry(Request.UserHostAddress).HostName;
19        }
```

Options

If Instant Connect lines were purchased with the integration, continue following the steps below.

Prerequisites:

- A phone number that will be tied to the Instant Connect lines needs to be added to the customer PBX.
 - For cloud customers this will be done by the SaaS team. The customer can provide their own number they wish to use, or we can reach out to OneTel to obtain a new number.
- The extensions that the phone number maps to will also need to be added to the customer PBX. Customer will need to provide these extensions.
 - For cloud customers, this is also done by the SaaS team.

Instant Connect Installation

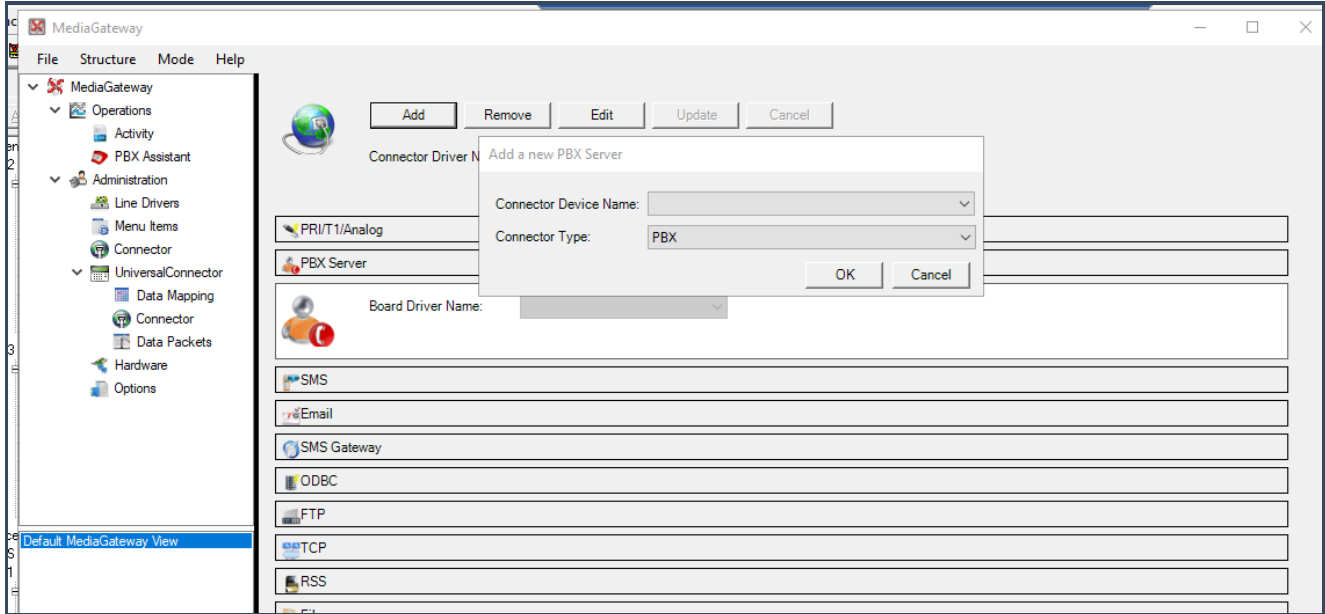
Adding the extensions to the switch config file.

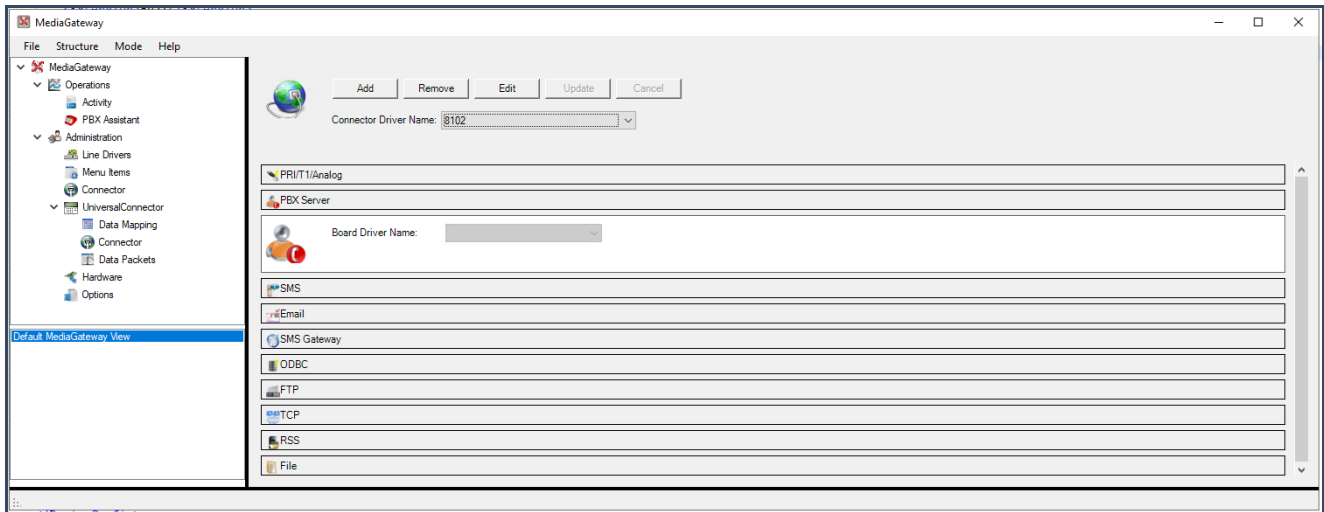
- Add the extensions to either the Avayaswitch.config or the Aeonixswitch.config (depending whether the customer's PBX is Avaya or Aeonix) located at C:\ProgramData\Bold Technologies\PBX Server\. Below is an example of a customer using extensions 8102 and 8103.
 - For cloud customers, it will be the Avayaswitch.config.

```
</DeviceConfig>
<DeviceConfig>
  <Extension>8102</Extension>
  <ExtensionPassword>1234</ExtensionPassword>
  <Dependency>Independent</Dependency>
  <ExtensionDescription>Instant Connect</ExtensionDescription>
  <Share>true</Share>
  <Instance>2</Instance>
  <RegisterTerminal>true</RegisterTerminal>
</DeviceConfig>
<DeviceConfig>
  <Extension>8103</Extension>
  <ExtensionPassword>1234</ExtensionPassword>
  <Dependency>Independent</Dependency>
  <ExtensionDescription>Instant Connect</ExtensionDescription>
  <Share>true</Share>
  <Instance>2</Instance>
  <RegisterTerminal>true</RegisterTerminal>
</DeviceConfig>
</Devices>
</SwitchConfig>
```

Creating the Line Drivers in the MediaGateway

- Navigate to the Mediagateway console > Connector page.
- Select the PBX Server row and click Add.
- In the drop-down, select one of the extensions we just added to the config file. Repeat this step to add all new extensions.
- Once finished, click Update and then File > Save.





- Navigate to the Menu Items page.
- Right click Menu Configurations and import the "MS_Two Way Instant Connect.xml" included in this guide.
- Navigate to the Line Drivers page.
- Add a line driver for each new extension using the image below as a reference.

8102	<input checked="" type="checkbox"/>		TwoWayPLUS	MENU=ELEV,LINEID=1,FEP=1,2,3,4,5,6,RECEIVER=18,19,20,21,LINE=12,TYPE=1,USEDNIS=1	
8103	<input checked="" type="checkbox"/>		Instant Connect	MENU=ELEV,FEP=1,2,3,4,5,6,RECEIVER=18,19,20,21,LINE=12,TYPE=1,USEDNIS=1,LINEID=2	

- The Instant Connect feature is now ready for testing. Please note that the phone call that gets sent to Manitou will arrive a few seconds after the Numera GPS event. To ensure that the operator handling the first Numera GPS event is also prompted to accept the phone call, the phone call event code (*PA) will need to be given a higher priority than any of the GPS alarm event codes. You can configure this in Supervisor Workstation > Events > Event Codes and adjusting the Priority level for the *PA event.

Additional Information

Numera Libris / Manitou Event Code Translations Using HTTP POST/PUT

Reported Condition	Manitou Interpretation (Description)	Event Code
Normal	Panic	PA
MedicalDevice	Medical Device Alert (Libris+ only)	PA1
Fall	Medical Alarm	MA
Classify	Fall Possible	MT

GPSLoc	Location Update - GPS	UX1
NetworkLoc	Location Update - Network	UX2
BatteryLow	Battery Low	*L
ON	Device Powered On	*ON
Activation	Device Commissioned (Device was activated)	*A
Deactivation	Device Decommissioned (Deactivated)	NC
FailActivation	Device Commissioning Failed	NT
NetworkChange	Network Availability Change	*Z
LowActivityAlarm	System Inactivity	*LG
LongChargeAlarm	System Inactivity (In charger)	*LO
ChargeChange	Charger Change (On/Off charging cradle)	*M

*Note – Not all events from Numera Libris are transmitted to the Bold UniversalConnector through the API. Additionally, many of these events are defaulted to “Alarm = No.” Please ensure that any events you desire to have presented to an operator are modified in the Transmitter Type Event Programming.

Nortek/Numera Libris API Raw Events & Descriptions

The following is an excerpt from the Nortek API document:

Event Type	Displays As	Description
------------	-------------	-------------

LocationFix	Location update	A Location fix (satellite or network) was obtained
Normal	Call	User-initiated phone call from a button press
ON	Powered on	The device turned on or started after a reboot
Classify	Classify	Event created when analyzing activity for a fall
Note	Notes updated	A Patient note was updated
Status	Status	Status log entry
Debug	Debug	Debug log entry
ChargeChange	Charge status	Indicates the state of charging
NetworkChange	Network	Network availability change
Fall	Fall Assistance Call	Fall detected and call initiated
Logfile	Unknown	A request to download logs
BatteryLow	Low Battery Alert	Device has crossed the critical threshold of charge
Activation	Device Activation	Device was activated
FailActivation	Device Fail Activation	The activation failed
Deactivation	Device Deactivated	Device de-activated in UI or through API
MagCalibration	Magnetometer Calibration	Device was re-calibrated based on device's location
LowActivityAlarm	Low Activity Alarm	User has not had significant movement for the duration configured

LongChargeAlarm	Long Charge Alarm	Device left in charger too long
MedicalDevice	Medical Device Alert	Generated as part of Libris+ events

[DM_LIBRIS-JSON.xml](#)

[MS Two Way Instant Connect.xml](#)

[numeraPutListener.zip](#)

[NumeraEventPut.json](#)