THINMANAGER®
A ROCKWELL AUTOMATION TECHNOLOGY

White Paper

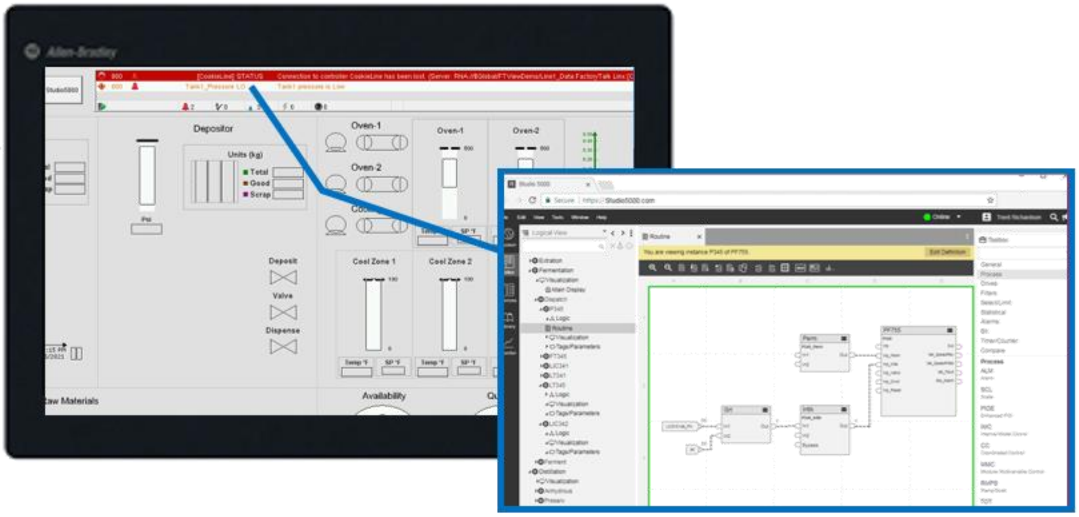
ThinManager® Logix PinPoint™

Contents

Introduction

ThinManager® Logix PinPoint™ is a value-add software application providing a visual interface of Studio 5000 Logix Designer with automated controller project navigation used for troubleshooting production operations from operator work terminals. It can be purchased as a recommended product with a new ThinManager installation or can be purchased independently to add to an existing ThinManager installation. It includes a license that allows you to run Studio 5000 Logix Designer in view-only mode.

The desired outcomes of ThinManager Logix PinPoint are to reduce downtime spent troubleshooting in Logix code with automated tag navigation and easily train new employees with little to no knowledge of existing PLC code. The automated tag navigation provided by ThinManager Logix PinPoint has proven to play a critical role in industries that perform many changeovers and spend time troubleshooting on the spot during production.



To successfully implement ThinManager Logix PinPoint into your visualization system, there are three essential process step definitions:

1. Identification of the trigger application
2. Configuration and deployment of ThinManager Logix PinPoint (PinPoint.exe file)
3. Customization of the delivery of Logix (by terminal, user, location, etc.)

This document provides step-by-step instructions to install and configure ThinManager Logix PinPoint software with your visualization system. It will also provide a set of best practices and examples for the application of ThinManager Logix PinPoint in your ThinManager configuration.

Identify Trigger Application

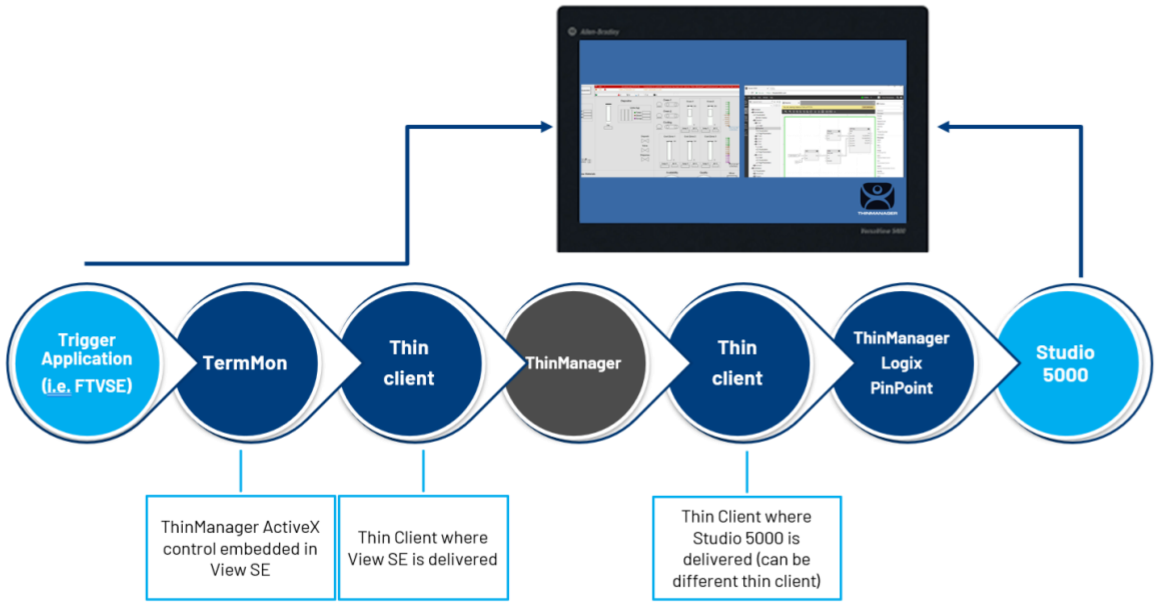
ThinManager Logix PinPoint is compatible with any application that supports a Windows ActiveX control, such as:

- FactoryTalk View SE
- AVEVA InTouch HMI
- Microsoft Excel

In this guide, we will focus on using FactoryTalk View SE as our client application. The principles that are applied to FactoryTalk View SE can also be applied to other client applications as well.

Configure and Deploy ThinManager Logix PinPoint

ThinManager Logix PinPoint™ is a C# application that connects to Logix Services to automate Logix through ThinManager-generated Events.


Trigger Application (i.e. FTVSE)
TermMon
Thin client
ThinManager
Thin client
ThinManager Logix PinPoint
Studio 5000
ThinManager ActiveX control embedded in View SE
Thin Client where View SE is delivered
Thin Client where Studio 5000 is delivered (can be different thin client)

Here are the steps that you need to configure ThinManager Logix PinPoint to work with ThinManager software:

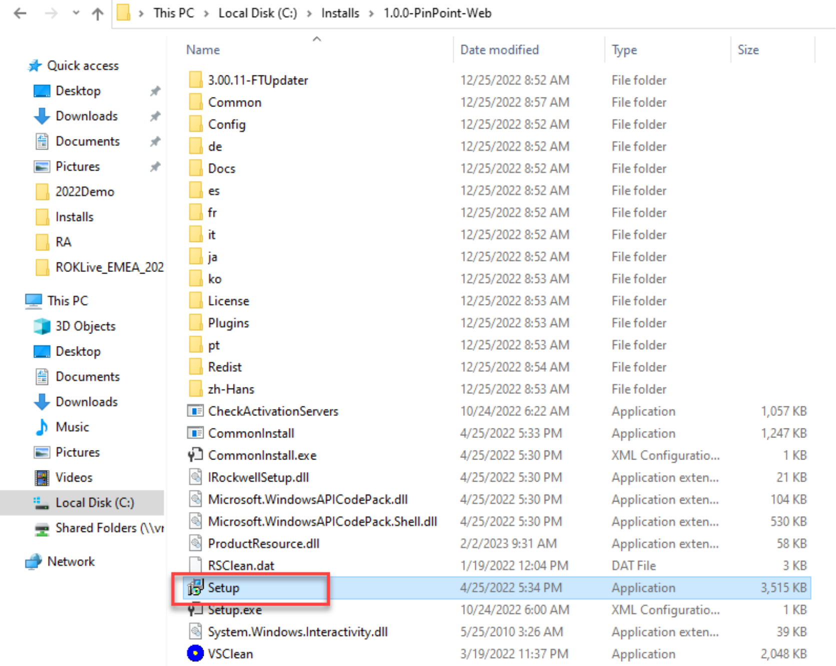
1. Install and license the ThinManager Logix PinPoint application on the machine that is running Studio 5000 Logix Designer.
2. Update TermMon ActiveX Control on Client Servers.
3. Add the TermMon ActiveX control and appropriate VBA code to the project.
4. (For ThinManager version 13.0 only) Add an Event to handle all ThinManager Logix PinPoint requests.
5. Create a new Display Client that launches the ThinManager Logix PinPoint executable. There are two types of apps you can deploy:
    a. ThinManager Logix PinPoint application with a UI (user interface)
    b. ThinManager Logix PinPoint Application with CLI (command-line interface)

Install ThinManager Logix PinPoint Software

ThinManager Logix PinPoint and Studio 5000 Logix Designer can be downloaded from the Rockwell Automation Product Compatibility and Download Center using your purchased serial number and product key. Run the installer for ThinManager Logix PinPoint on the same computer that will run the Studio 5000 Logix Designer software.

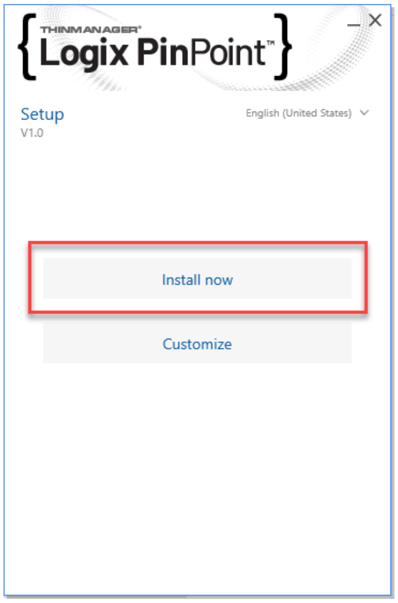
To install the software:

1. Extract the software and double-click on `Setup.exe` from the root folder.

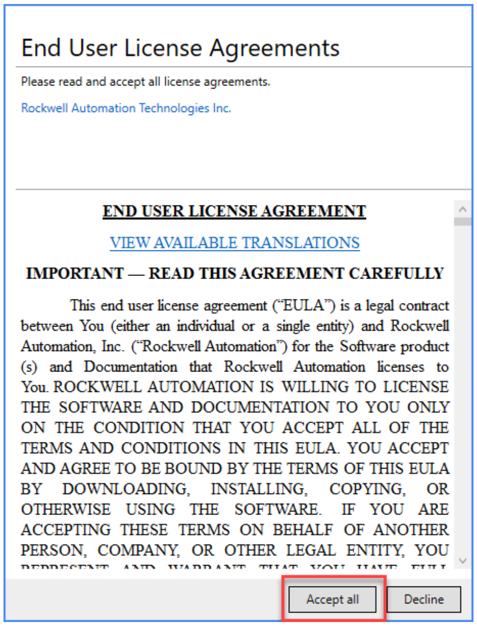


Name	Date modified	Type	Size
3.00.11-FTUpdater	12/25/2022 8:52 AM	File folder	
Common	12/25/2022 8:57 AM	File folder	
Config	12/25/2022 8:52 AM	File folder	
de	12/25/2022 8:52 AM	File folder	
Docs	12/25/2022 8:52 AM	File folder	
es	12/25/2022 8:52 AM	File folder	
fr	12/25/2022 8:52 AM	File folder	
it	12/25/2022 8:52 AM	File folder	
ja	12/25/2022 8:52 AM	File folder	
ko	12/25/2022 8:53 AM	File folder	
License	12/25/2022 8:53 AM	File folder	
Plugins	12/25/2022 8:53 AM	File folder	
pt	12/25/2022 8:53 AM	File folder	
Redist	12/25/2022 8:54 AM	File folder	
zh-Hans	12/25/2022 8:53 AM	File folder	
CheckActivationServers	10/24/2022 6:22 AM	Application	1,057 KB
CommonInstall	4/25/2022 5:33 PM	Application	1,247 KB
CommonInstall.exe	4/25/2022 5:30 PM	XML Configuratio...	1 KB
IRockwellSetup.dll	4/25/2022 5:30 PM	Application exten...	21 KB
Microsoft.WindowsAPICodePack.dll	4/25/2022 5:30 PM	Application exten...	104 KB
Microsoft.WindowsAPICodePack.Shell.dll	4/25/2022 5:30 PM	Application exten...	530 KB
ProductResource.dll	2/2/2023 9:31 AM	Application exten...	58 KB
RSClean.dat	1/19/2022 12:04 PM	DAT File	3 KB
Setup	4/25/2022 5:34 PM	Application	3,515 KB
Setup.exe	10/24/2022 6:00 AM	XML Configuratio...	1 KB
System.Windows.Interactivity.dll	5/25/2010 3:26 AM	Application exten...	39 KB
VSClean	3/19/2022 11:37 PM	Application	2,048 KB

2. Click the **Install Now** button.



3. Read through the End User License Agreement and click the **Accept all** button to accept the agreement and install the software.



End User License Agreements

Please read and accept all license agreements.

Rockwell Automation Technologies Inc.

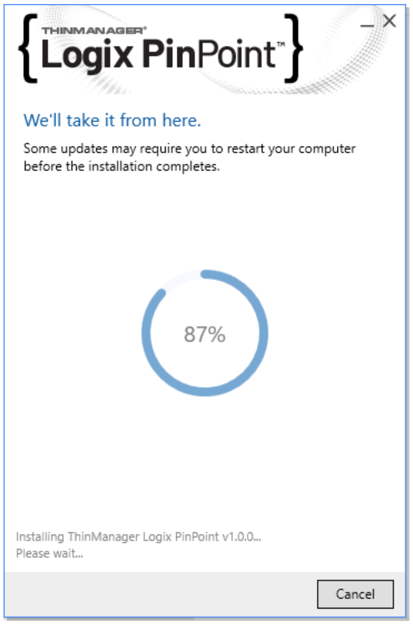
END USER LICENSE AGREEMENT

VIEW AVAILABLE TRANSLATIONS

IMPORTANT — READ THIS AGREEMENT CAREFULLY

This end user license agreement ("EULA") is a legal contract between You (either an individual or a single entity) and Rockwell Automation, Inc. ("Rockwell Automation") for the Software product (s) and Documentation that Rockwell Automation licenses to You. ROCKWELL AUTOMATION IS WILLING TO LICENSE THE SOFTWARE AND DOCUMENTATION TO YOU ONLY ON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS AND CONDITIONS IN THIS EULA. YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS EULA BY DOWNLOADING, INSTALLING, COPYING, OR OTHERWISE USING THE SOFTWARE. IF YOU ARE ACCEPTING THESE TERMS ON BEHALF OF ANOTHER PERSON, COMPANY, OR OTHER LEGAL ENTITY, YOU

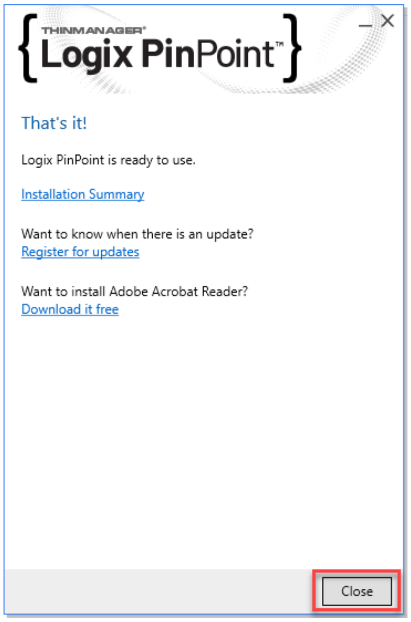
4. Allow the installation to proceed. When it is finished, you will see a "That's it" screen. Click the **Close** button to close the installer.



We'll take it from here.

Some updates may require you to restart your computer before the installation completes.

Installing ThinManager Logix PinPoint v1.0.0...
Please wait...



That's it!

Logix PinPoint is ready to use.

Installation Summary

Want to know when there is an update?
Register for updates

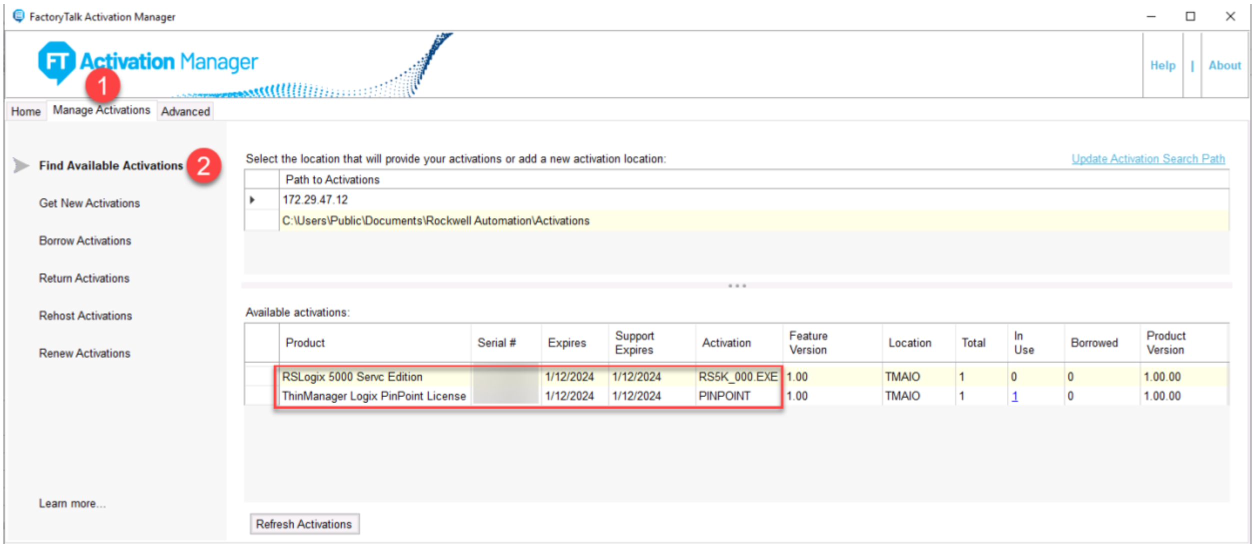
Want to install Adobe Acrobat Reader?
Download it free

The applications for ThinManager Logix PinPoint can be found in the `C:\Program Files (x86)\Rockwell Software\PinPoint` directory after running the installer.

Install the purchased ThinManager Logix PinPoint activations using the process found in technote QA7280, FactoryTalk Activation Manager: Activating software with Internet access (custhelp.com). For computers that do not have an internet connection, follow the process found in technote QA26927, FactoryTalk Activation Manager: offline activation (custhelp.com). After you have successfully downloaded the ThinManager Logix PinPoint license you should see two licenses included in FactoryTalk Activation Manager listed in **Manage Activations → Find Available Activations**:

Product	Activation
RSLogix 5000 Servc Edition	RS5K_000.EXE
ThinManager Logix PinPoint License	PINPOINT

For more information about the functionality limitations with the included Studio 5000 Service Edition license please see technote IN1803, About RSLogix 5000 and Studio 5000 Service Edition (custhelp.com).



Select the location that will provide your activations or add a new activation location:

Path to Activations
172.29.47.12
C:\Users\Public\Documents\Rockwell Automation\Activations

Available activations:

Product	Serial #	Expires	Support Expires	Activation	Feature Version	Location	Total	In Use	Borrowed	Product Version
RSLogix 5000 Servc Edition	[redacted]	1/12/2024	1/12/2024	RS5K_000.EXE	1.00	TMAIO	1	0	0	1.00.00
ThinManager Logix PinPoint License	[redacted]	1/12/2024	1/12/2024	PINPOINT	1.00	TMAIO	1	1	0	1.00.00

Register TermMon ActiveX Control

ThinManager Logix PinPoint automatically installs and registers a version of the TermMon ActiveX control that is needed to use the software. The TermMon.ocx file is found in the `C:\Program Files (x86)\Rockwell Software\PinPoint` directory. You may also download the latest version of the TermMon ActiveX Control from the website: https://downloads.thinmanager.com.

You may need to copy the file onto the computer hosting your client and register the ActiveX Control. Follow the process found in technote QA45262, Updating the TermMon ActiveX in FactoryTalk View SE (custhelp.com). These steps can be adapted for other HMI applications, as the ActiveX Control need only be copied to a directory where the HMI application has Windows OS permissions to access it. The REGSVR32 command will point to the appropriate ActiveX Control location on the computer's drive. For installations of FactoryTalk View SE, the TermMon.ocx file needs to replace the existing one that is present in the RSView Enterprise folder: `c:\Program Files (x86)\Rockwell Software\RSView Enterprise\TermMon.ocx`.

The TermMon ActiveX control should be updated on every machine that:

- Hosts the HMI client software
  - Example: FactoryTalk View SE Client
- Hosts the development environment for your HMI software
  - Example: FactoryTalk View Studio

Add VBA Code and Objects to FactoryTalk View SE Display

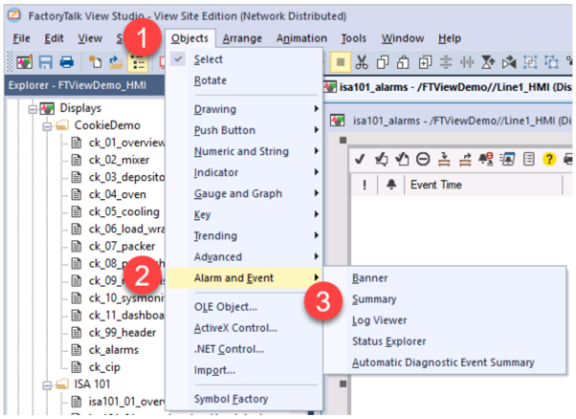
This document discusses using FactoryTalk Logix PinPoint with FactoryTalk View SE software, but you could easily adapt these steps to be used with any program that supports Microsoft VBA code.

Alarm Summary

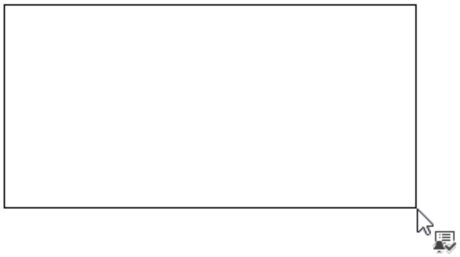
First, we will use the example of coding our FactoryTalk View SE software to open a rung of logic in Studio 5000 Logix Designer every time we highlight an alarm in the Alarm Summary object. On our

FactoryTalk View SE Display, we need to expose the correct VBA event to pass the tag information to the ThinManager Logix PinPoint application.

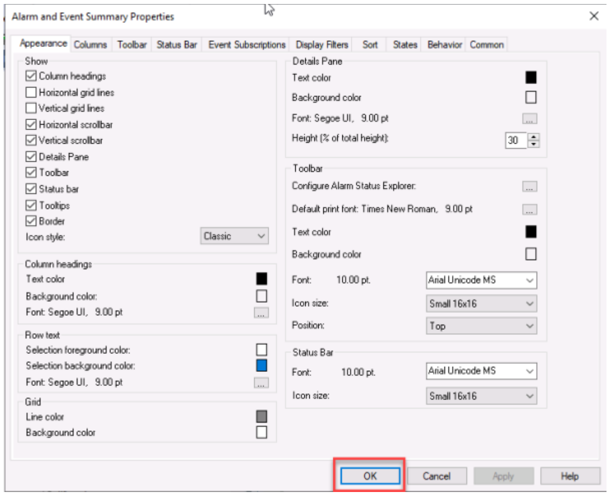
1. Open FactoryTalk View Studio.
2. Select your application and open it.
3. Open an existing Display you want to add the Alarm Summary object to (or create a new Display). You may also use an existing Alarm Summary Object and skip steps 4, 5, and 6.
4. Go to the **Objects** menu and navigate to **Alarm and Event → Summary**.



5. Drag your mouse cursor across the Display to draw a new ActiveX control on the screen.



6. When the Alarm and Event Summary Properties window pops up, click **OK** to keep the default settings.



Alarm and Event Summary Properties

Appearance | Columns | Toolbar | Status Bar | Event Subscriptions | Display Filters | Sort | States | Behavior | Common

Show
☑ Column headings
☐ Horizontal grid lines
☐ Vertical grid lines
☑ Horizontal scrollbar
☑ Vertical scrollbar
☑ Details Pane
☑ Toolbar
☑ Status bar
☑ Tooltips
☑ Border
Icon style: Classic

Column headings
Text color ■
Background color: □
Font: Segoe UI, 9.00 pt

Row text
Selection foreground color: □
Selection background color: 🟦
Font: Segoe UI, 9.00 pt

Grid
Line color ■
Background color □

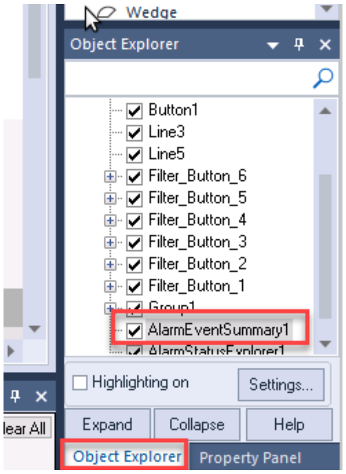
Details Pane
Text color ■
Background color □
Font: Segoe UI, 9.00 pt
Height (% of total height): 30

Toolbar
Configure Alarm Status Explorer:
Default print font: Times New Roman, 9.00 pt
Text color ■
Background color □
Font: 10.00 pt. Arial Unicode MS
Icon size: Small 16x16
Position: Top

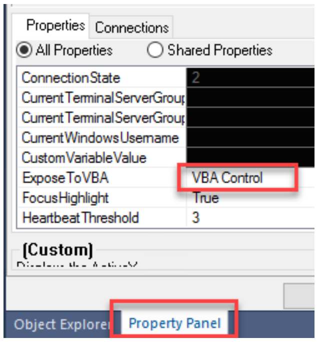
Status Bar
Font: 10.00 pt. Arial Unicode MS
Icon size: Small 16x16

OK | Cancel | Apply | Help

7. Click on the newly drawn Alarm Summary object. You may use the Object Explorer to highlight it instead of finding it on the Display.



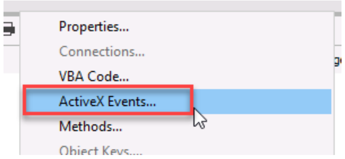
8. Click the **Property Panel** tab at the lower right-hand corner of the screen or open from the View → Property Panel menu at the top of the screen. Scroll down to the ExposeToVBA field and change the value to **VBA Control**.



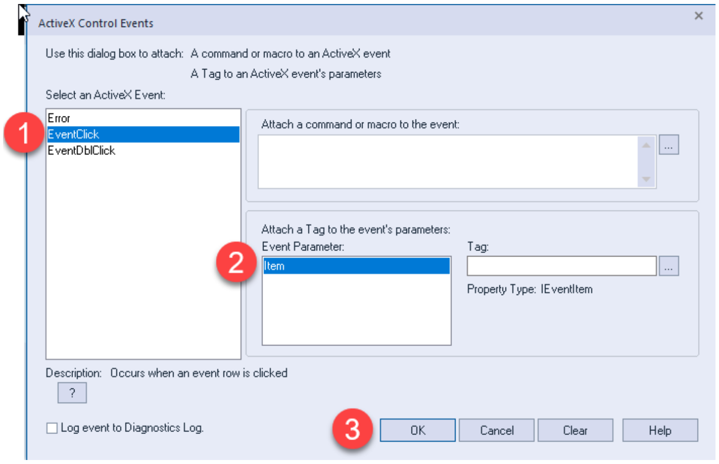
ConnectionState	2
CurrentTerminalServerGrouj	
CurrentTerminalServerGrouj	
CurrentWindowsUsername	
CustomVariableValue	
ExposeToVBA	VBA Control
FocusHighlight	True
HeartbeatThreshold	3

[Custom]

9. Right-click on the Alarm Summary object in the Display and select ActiveX Events.



10. Select **EventClick**, **Item**, and then click **OK**.


ActiveX Control Events

Use this dialog box to attach: A command or macro to an ActiveX event
A Tag to an ActiveX event's parameters

Select an ActiveX Event:

Error
EventClick
EventDblClick

Attach a command or macro to the event:

Attach a Tag to the event's parameters:

Event Parameter: Item

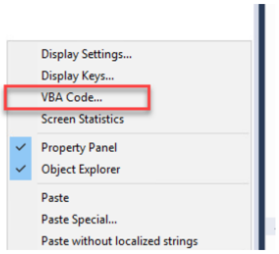
Tag:

Property Type: IEventItem

Description: Occurs when an event row is clicked

□ Log event to Diagnostics Log.

11. Right-click on the Display and select **VBA Code** to pop up the Microsoft Visual Basic editor.



12. Notice how there is now a new event in the VBA code, `Private Sub AlarmEventSummary1_EventClick(ByVal Item As FTAlarmEventSummary.IEventItem)` (the name may vary depending on the name of your Alarm Summary object).
13. Copy and paste this starter code into your new method between the VBA method declaration (Private Sub) and the end of the VBA method declaration (End Sub). Modify as necessary to make sure that the TermMon ActiveX control and the Alarm Summary object are the same names you use in your Display:

```
'Declare variables

Dim Tag As String
Dim Scope As String
Dim Path As String

'There may be a more elegant way to parse the result of
GetColumValue, but that is the key call before executing PinPoint

Dim eventsourceResult As String
Dim AlarmTagEnum() As String
eventsourceResult = Item.GetColumnValue("eventsource")
Dim eventsourceLen As Integer
eventsourcelen = Len(eventsourceResult)
Dim TagScope As String
TagScope = Right(eventsourceResult, eventsourcelen -
InStrRev(eventsourceResult, ":"))
Dim TagScopeLen As Integer
TagScopeLen = Len(TagScope)

'If the tag is coming from the HMI tag server, it's not a valid
tag so pop up a message box

If InStr(TagScope, ".") = 0 Then
MsgBox ("Pinpoint will not work on this tag!")
Exit Sub
End If
```

```
'Parse out the parameters from the IEventItem string that the
ThinManager Logix PinPoint event needs

Tag = Right(TagScope, TagScopeLen - InStr(TagScope, "."))
Scope = Left(TagScope, InStr(TagScope, ".") - 1)
Path = "Project=C:\Demo\2022Demo\CookieLine_v33_wTMEvents.ACD"
Dim TermMonResult As Integer

'Pass the parameters from above and trigger a new ThinManager
Logix PinPoint event

TermMonResult = TermMonControl1.PinPoint(3, Tag, Scope, Path)

'Uncomment the next line for diagnostics. This will pop up a
message box. A result of 0 means the PinPoint event was sent to
the server, anything else is a failure.

'MsgBox TermMonResult
```

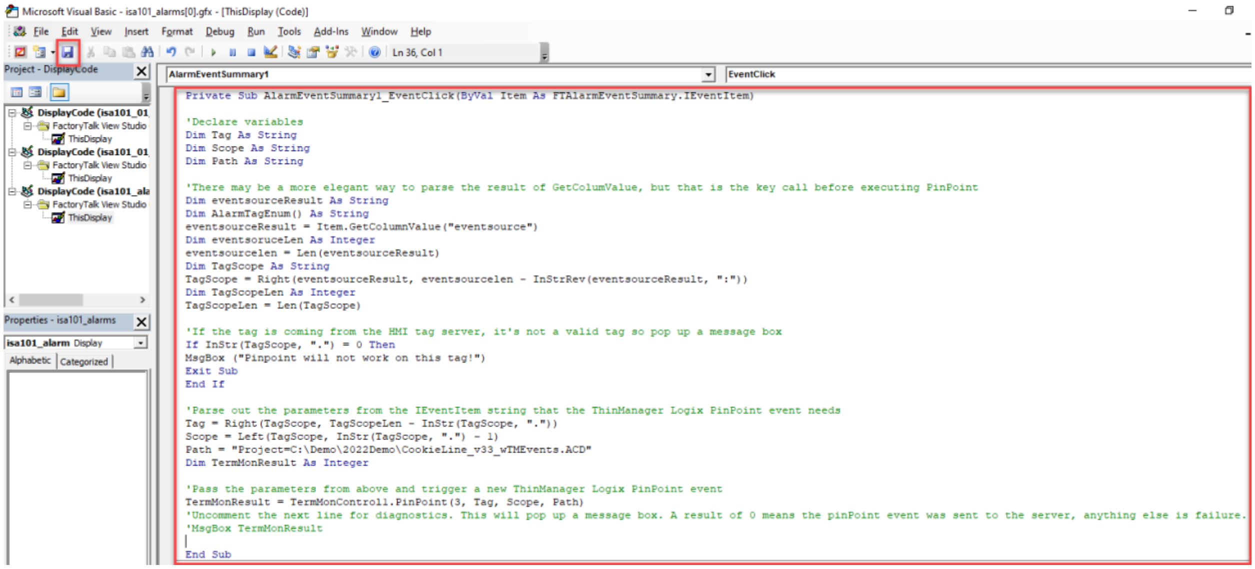
Note: For an explanation of the different arguments of the PinPoint VBA subroutine, see the *PinPoint Settings and Configuration* → *VBA* section of this document.

14. We need to make sure the TermMon ActiveX control is activated whenever the FactoryTalk View SE Display is called. Look for a VBA method definition of `Private Sub Display_AnimationStart()`. If one does not exist, copy in all the following code to activate the TermMon ActiveX control whenever the Display starts:

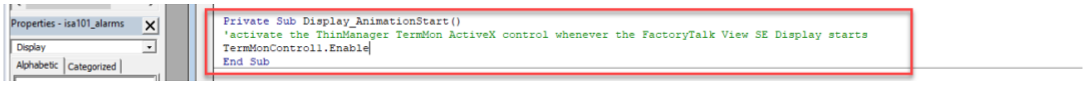
```
Private Sub Display_AnimationStart()
'activate the ThinManager TermMon ActiveX control
TermMonControl1.Enable
End Sub
```

If the `Display_AnimationStart()` method definition already exists, simply add the two middle lines of code to the existing method definition. You should not define the `Display_AnimationStart()` method twice. Note that the name of your TermMon ActiveX control may be different than what is in the code above.

15. The two VBA methods should look like the screenshot below. Click on the **Save** button to save the VBA code. Close the VBA editor.

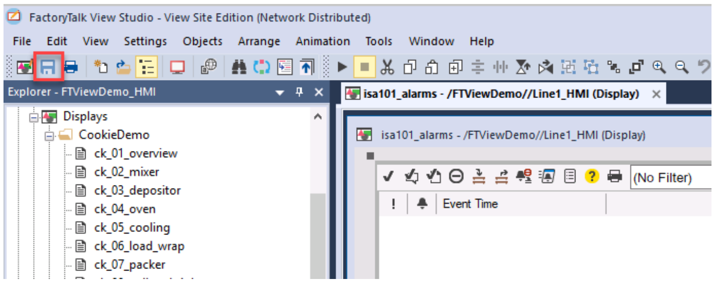


```
Private Sub AlarmEventSummary1_EventClick(ByVal Item As FTAlarmEventSummary.IEventItem)

'Declare variables
Dim Tag As String
Dim Scope As String
Dim Path As String

'There may be a more elegant way to parse the result of GetColumValue, but that is the key call before executing PinPoint
Dim eventsourceResult As String
Dim AlarmTagEnum() As String
eventsourceResult = Item.GetColumnValue("eventsource")
Dim eventsoruceLen As Integer
eventsourcelen = Len(eventsourceResult)
Dim TagScope As String
TagScope = Right(eventsourceResult, eventsourcelen - InStrRev(eventsourceResult, ":"))
Dim TagScopeLen As Integer
TagScopeLen = Len(TagScope)

'If the tag is coming from the HMI tag server, it's not a valid tag so pop up a message box
If InStr(TagScope, ".") = 0 Then
MsgBox ("Pinpoint will not work on this tag!")
Exit Sub
End If

'Parse out the parameters from the IEventItem string that the ThinManager Logix PinPoint event needs
Tag = Right(TagScope, TagScopeLen - InStr(TagScope, "."))
Scope = Left(TagScope, InStr(TagScope, ".") - 1)
Path = "Project=C:\Demo\2022Demo\CookieLine_v33_wTMEvents.ACD"
Dim TermMonResult As Integer

'Pass the parameters from above and trigger a new ThinManager Logix PinPoint event
TermMonResult = TermMonControl1.PinPoint(3, Tag, Scope, Path)
'Uncomment the next line for diagnostics. This will pop up a message box. A result of 0 means the pinPoint event was sent to the server, anything else is failure.
'MsgBox TermMonResult

End Sub
```



```
Private Sub Display_AnimationStart()
'activate the ThinManager TermMon ActiveX control whenever the FactoryTalk View SE Display starts
TermMonControl1.Enable
End Sub
```

16. Click the **Save** icon for the Display in FactoryTalk View Studio (if it is not greyed out). Relaunch the FactoryTalk View SE Client software on the thin client or browse to another Display and then back to this one to update the VBA for the Display.

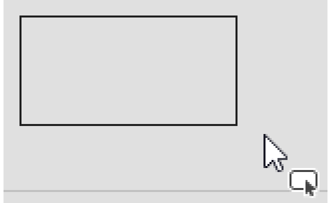

FactoryTalk View Studio - View Site Edition (Network Distributed)
File Edit View Settings Objects Arrange Animation Tools Window Help
Explorer - FTViewDemo_HMI
Displays
CookieDemo
ck_01_overview
ck_02_mixer
ck_03_depositor
ck_04_oven
ck_05_cooling
ck_06_load_wrap
ck_07_packer
isa101_alarms - /FTViewDemo//Line1_HMI (Display)
isa101_alarms - /FTViewDemo//Line1_HMI (Display)
(No Filter)
!
Event Time

Button

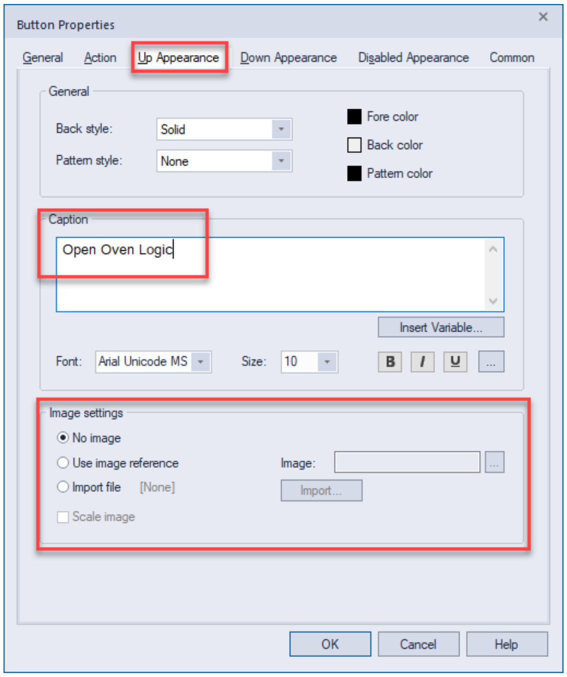
In this example, we will open a specific rung of code in ThinManager Logix PinPoint whenever a button is pressed in a FactoryTalk View SE Display. Be sure to follow steps 1 through 8 in the Alarm Summary section of this document so you have the TermMon ActiveX object added to the Display and interacting with the VBA code. This is to demonstrate that you could call up a rung of code based on a specific tag without using alarms in your HMI application.

To add a button to call up a rung of code in Studio 5000:

1. Open FactoryTalk View Studio.
2. Select your application and open it.
3. Open an existing Display you want to add the button object to (or create a new Display). You may also use an existing button and skip steps 4, 5, and 6.
4. Go to the **Objects** menu and navigate to **Push Button → Button**. Drag your mouse cursor across the Display to draw a new button on the screen.



5. When the Button Properties window pops up, click the **Up Appearance** tab. Set the text you want for the button by typing in text into the Caption field, or set an image using the Image settings section of the window.



Button Properties

General

Back style: Solid

Pattern style: None

■ Fore color

□ Back color

■ Pattern color

Caption

Open Oven Logic

Font: Arial Unicode MS

Size: 10

Image settings

● No image

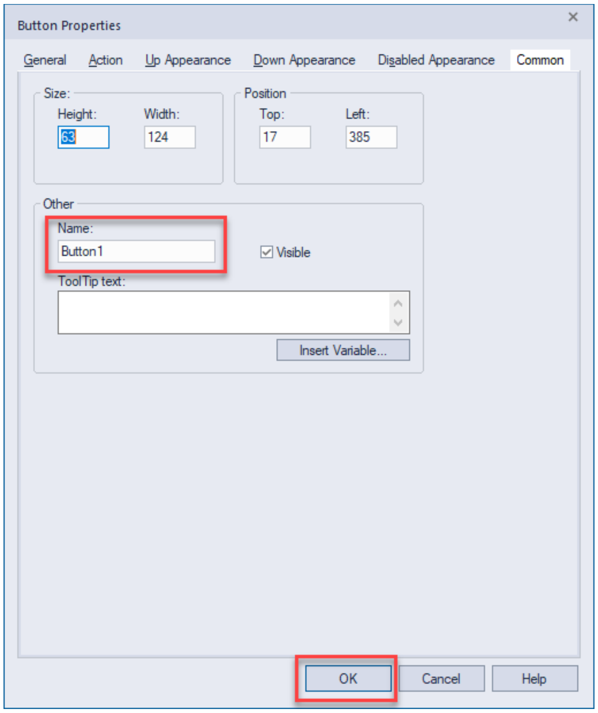
○ Use image reference

○ Import file [None]

□ Scale image

Image:

6. Click on the **Common** tab and take note of what the button name will be in the VBA code by looking at the Name field. Feel free to change it to whatever you desire. Click the **OK** button to close the window.

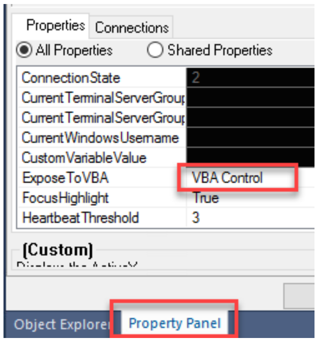


Size:
Height: 63
Width: 124

Position
Top: 17
Left: 385

Other
Name: Button1
☑ Visible
ToolTip text:

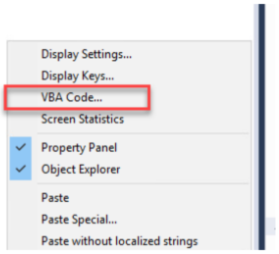
7. Click on the newly drawn button. Click the **Property Panel** tab at the lower right-hand corner of the screen or open from the View → Property Panel menu at the top of the screen. Scroll down to the ExposeToVBA field and change the value to **VBA Control**.



ConnectionState	[redacted]
CurrentTerminalServerGrou	[redacted]
CurrentTerminalServerGrou	[redacted]
CurrentWindowsUsername	[redacted]
CustomVariableValue	[redacted]
ExposeToVBA	VBA Control
FocusHighlight	True
HeartbeatThreshold	3

(Custom)

8. Right-click on the Display and select **VBA Code** to pop up the Microsoft Visual Basic editor.

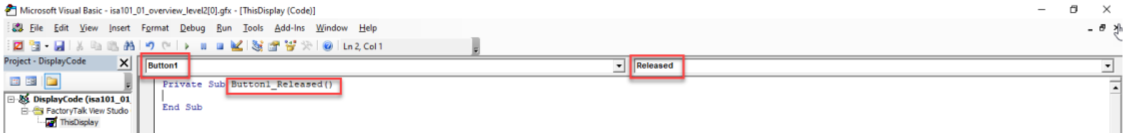


17. We need to make sure the TermMon ActiveX control is activated whenever the FactoryTalk View SE Display is called. Look for a VBA method definition of `Private Sub Display_AnimationStart()`. If one does not exist, copy in all the following code to activate the TermMon ActiveX control whenever the Display starts:

```
Private Sub Display_AnimationStart()
'activate the ThinManager TermMon ActiveX control
TermMonControl1.Enable
End Sub
```

If the `Display_AnimationStart()` method definition already exists, simply add the two middle lines of code to the existing method definition. You should not define the `Display_AnimationStart()` method twice. Note that the name of your TermMon ActiveX control may be different than what is in the code above.

9. Look for a subroutine with the button name in it. It should be formatted <ButtonName>_Released(). If you do not see one, changed the dropdowns at the top of the window to the button name and Released as shown in the screenshot below. This should automatically generate a VBA subroutine for you to use.

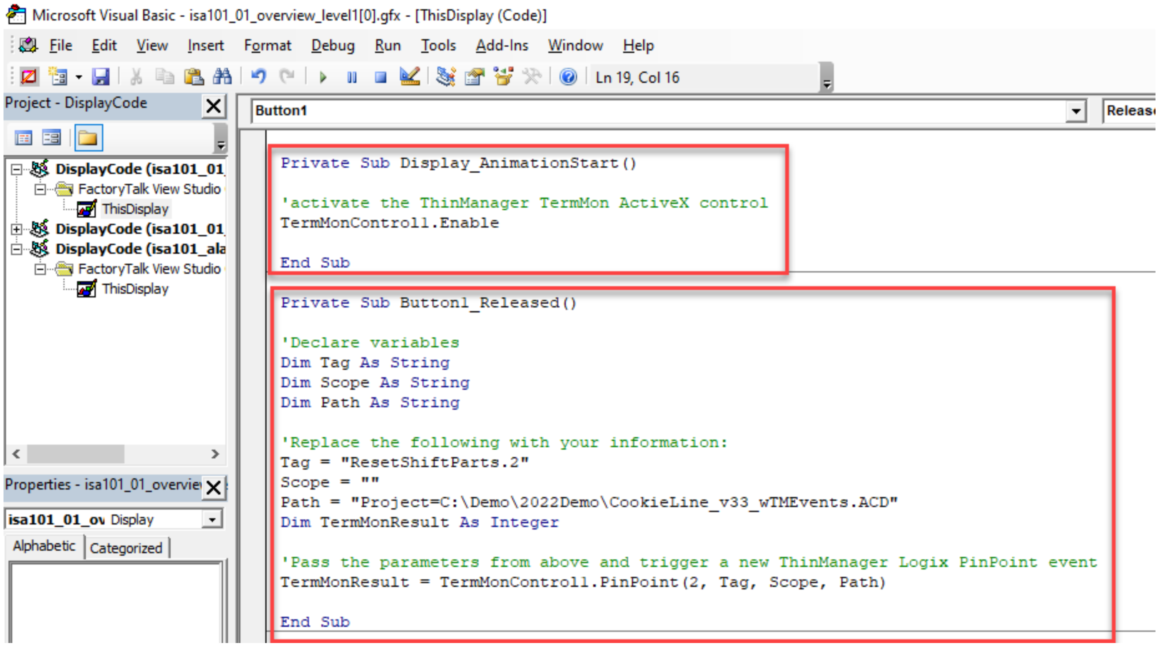

Microsoft Visual Basic - isa101_01_overview_level2[0].gfx - [ThisDisplay (Code)]
File Edit View Insert Format Debug Run Tools Add-Ins Window Help
Ln 2, Col 1
Project - DisplayCode
DisplayCode (isa101_01
FactoryTalk View Studio
ThisDisplay
Button1
Released

```
Private Sub Button1_Released()

End Sub
```

10. Copy and paste this starter code into your new method between the VBA method declaration (Private Sub) and the end of the VBA method declaration (End Sub). Be sure to specify the Tag, Scope, and Path of the tag you want to access in the controller by replacing the <>s in the code below:

```
'Declare variables
Dim Tag As String
Dim Scope As String
Dim Path As String

'Replace the following with your information:
Tag = <Tag name in the controller>
Scope = <Scope of the tag in the controller>
'Optional: leave Path blank with ""s
Path = <Project=File path to .acd file>
Dim TermMonResult As Integer

'Pass the arguments from above and trigger a new ThinManager
Logix PinPoint event
TermMonResult = TermMonControl1.PinPoint(2, Tag, Scope, Path)
```

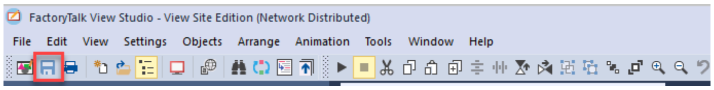
11. The two VBA methods should look like the screenshot below. Click on the **Save** button to save the VBA code. Close the VBA editor.



```
Private Sub Display_AnimationStart()

'activate the ThinManager TermMon ActiveX control
TermMonControl1.Enable

End Sub
```

```
Private Sub Button1_Released()

'Declare variables
Dim Tag As String
Dim Scope As String
Dim Path As String

'Replace the following with your information:
Tag = "ResetShiftParts.2"
Scope = ""
Path = "Project=C:\Demo\2022Demo\CookieLine_v33_wTMEvents.ACD"
Dim TermMonResult As Integer

'Pass the parameters from above and trigger a new ThinManager Logix PinPoint event
TermMonResult = TermMonControl1.PinPoint(2, Tag, Scope, Path)

End Sub
```

12. Click the **Save** icon for the Display in FactoryTalk View Studio (if it is not greyed out). Relaunch the FactoryTalk View SE Client software on the thin client or browse to another Display and then back to this one to update the VBA for the Display.

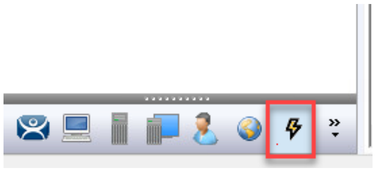

FactoryTalk View Studio - View Site Edition (Network Distributed)
File Edit View Settings Objects Arrange Animation Tools Window Help

Add Event to ThinManager Server

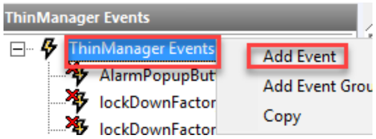
You need to add an Event to the ThinManager server to handle any ThinManager Logix PinPoint Events. Events are only supported with ThinManager software version 13.0 and newer. ThinManager software version 13.1 and newer has a built-in Event Type specifically for PinPoint

To add a PinPoint Event to a version 13.1 ThinManager server:

1. Open the ThinManager Server and click on the **Events** icon in the lower left-hand corner. You may need to expand the navigation bar if you do not see the Events icon.

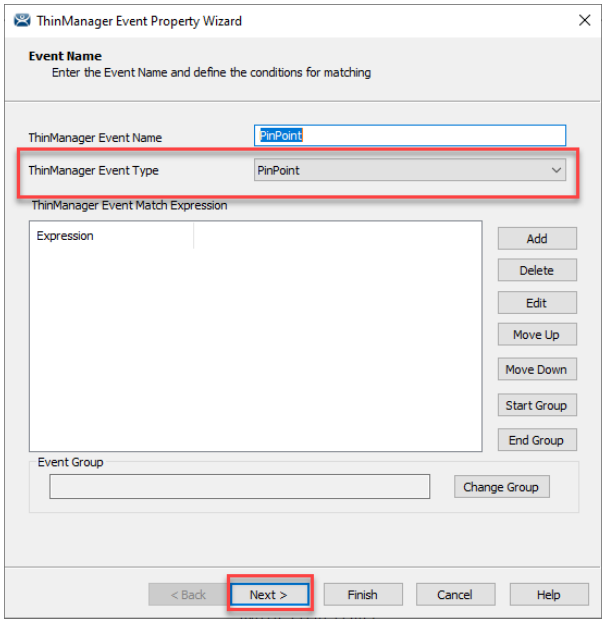


2. Right-click ThinManager Events and click **Add Event** to bring up the ThinManager Event Property Wizard.



ThinManager Events

3. Click the **Next** button on the Event Wizard to get to the Event Action page.
4. Change the ThinManager Event Type to **PinPoint**. Click the **Next** button.



ThinManager Event Property Wizard

Event Name

Enter the Event Name and define the conditions for matching

ThinManager Event Name: PinPoint

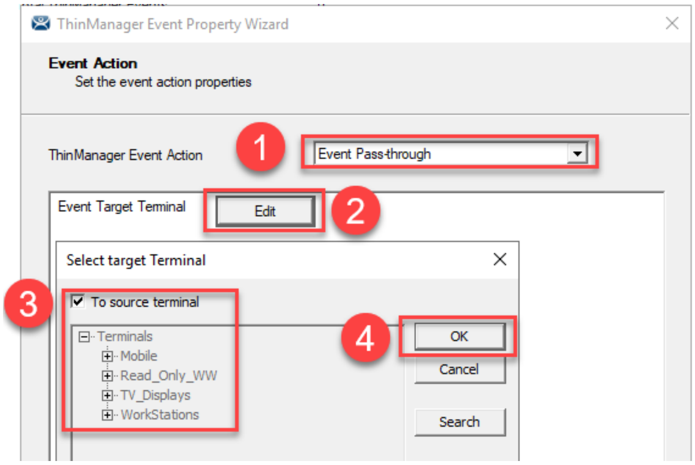
ThinManager Event Type: PinPoint

ThinManager Event Match Expression

Expression

Event Group

5. For ThinManager Event Action drop-down select **Event Pass-through**. Click the **Edit** button to change which terminal receives the event. Click the **OK** button when you are finished.
    a. For example, if you want the Event Action to be Add a Display Client, you will choose an Event Destination Terminal and a Display Client. In the Select target Terminal window, the **To source terminal** checkbox will direct any Event from any terminal back at itself. This would be desired if you want the Event to trigger on multiple terminals without having to create multiple Events.



ThinManager Event Property Wizard

Event Action

Set the event action properties

ThinManager Event Action: Event Pass-through

Event Target Terminal

Select target Terminal

☑ To source terminal

- Terminals
  - Mobile
  - Read_Only_WW
  - TV_Displays
  - WorkStations

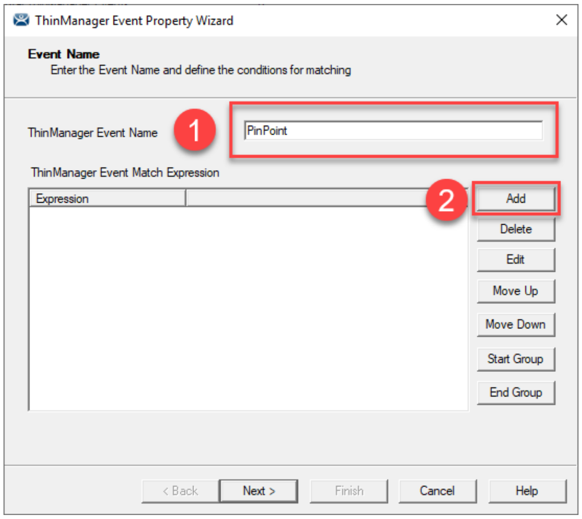
6. Click the **Finish** button to exit the ThinManager Event Property Wizard.

The Event is now active and listening for any incoming ThinManager Logix PinPoint Events to be sent from the source application.

Note: In the *Other Terminal* section in this document, we describe a method that allows you to add multiple PinPoint Events to a ThinManager server. This allows you to customize where the PinPoint Event is delivered. For example, you may wish to trigger the PinPoint Event to be delivered to other terminals depending on which terminal triggered the Event.

Adding an Event to a ThinManager version 13.0 of ThinManager server is like the steps for version 13.1, except that version 13.0 does not include the Event Type field. This field allows you to specify PinPoint as the Event Type, saving you from using a Match Expression to define it. Follow the steps to add an event for a version 13.1 ThinManager server. Here is the additional configuration you must perform with a version 13.0 ThinManager server Event:

1. In the ThinManager Event Name field type PinPoint, then click the **Add** button.



ThinManager Event Property Wizard

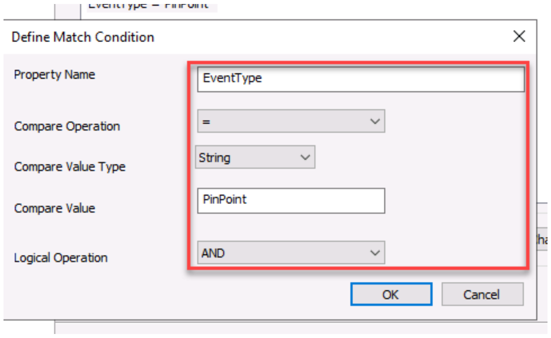
Event Name

Enter the Event Name and define the conditions for matching

ThinManager Event Name: PinPoint

ThinManager Event Match Expression

2. To define an Event Match Condition set the following fields and then click **OK**:
    a. Property Name: `EventType`
    b. Compare Operation: =
    c. Compare Value Type: `String`
    d. Compare Value: `PinPoint`
    e. Logical Operation: `AND`



Define Match Condition

Property Name: EventType

Compare Operation: =

Compare Value Type: String

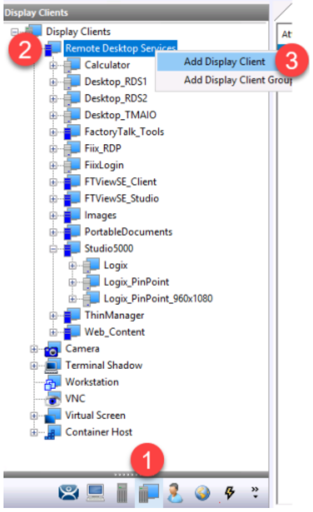
Compare Value: PinPoint

Logical Operation: AND

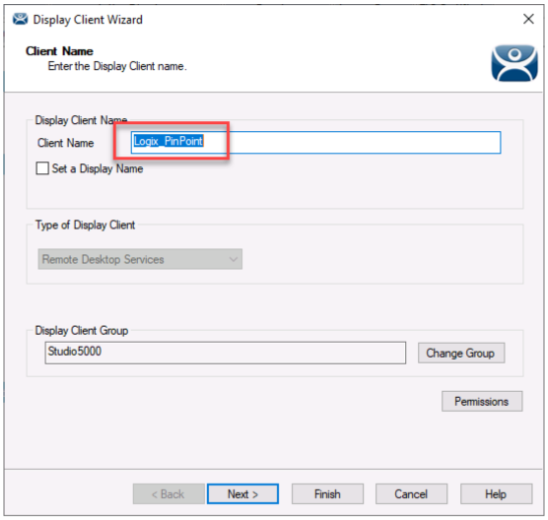
Create a ThinManager Logix PinPoint Display Client

You can launch the ThinManager Logix PinPoint application using the ThinManager AppLink feature. This is the easiest way to configure the process to create the Display Client is as follows:

1. Launch the ThinManager Admin Console.
2. Navigate to the Display Clients section of the console by clicking on the **Display Clients** icon at the bottom left of the screen. Right-click **Remote Desktop Services** and choose the **Add Display Client** option.



3. Enter any name you choose into the Client Name field.



Display Client Wizard

Client Name

Enter the Display Client name.

Display Client Name

Client Name: Logix_PinPoint

☐ Set a Display Name

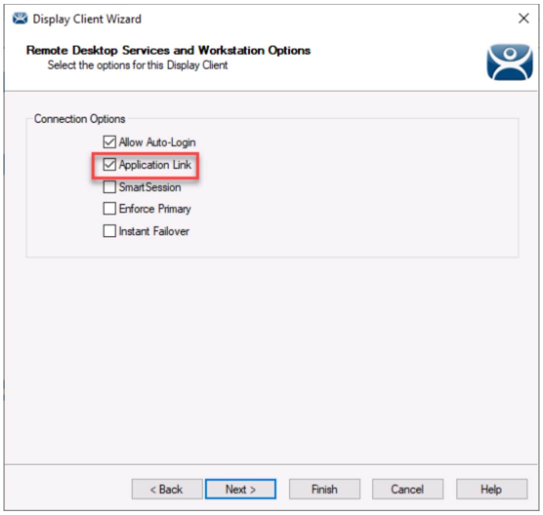
Type of Display Client

Remote Desktop Services

Display Client Group

Studio5000

4. Click the **Next** button until you get to the Remote Desktop Services and Workstation Options page of the Display Client Wizard. Put a check in the **Application Link** box.



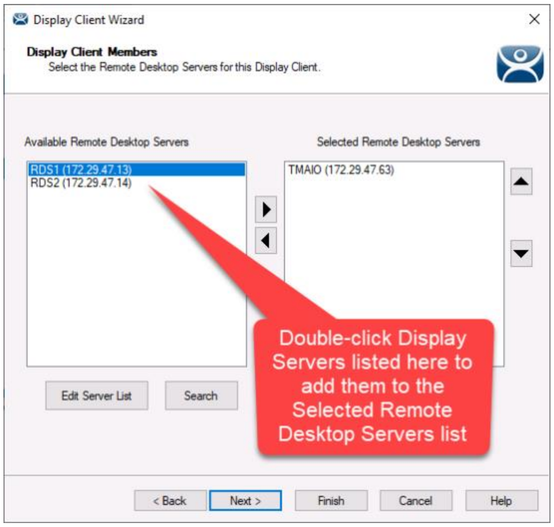
Display Client Wizard

Remote Desktop Services and Workstation Options
Select the options for this Display Client

Connection Options

☑ Allow Auto-Login
☑ Application Link
☐ SmartSession
☐ Enforce Primary
☐ Instant Failover

5. Click the **Next** button until you get to the Display Client Members page of the Display Client Wizard. **Double-click** on any servers in the Available Remote Desktop Server list to add them to the Selected Remote Desktop Server list. These should be servers that have the ThinManager Logix PinPoint and the Studio 5000 Logix Designer applications installed on them.



Display Client Wizard

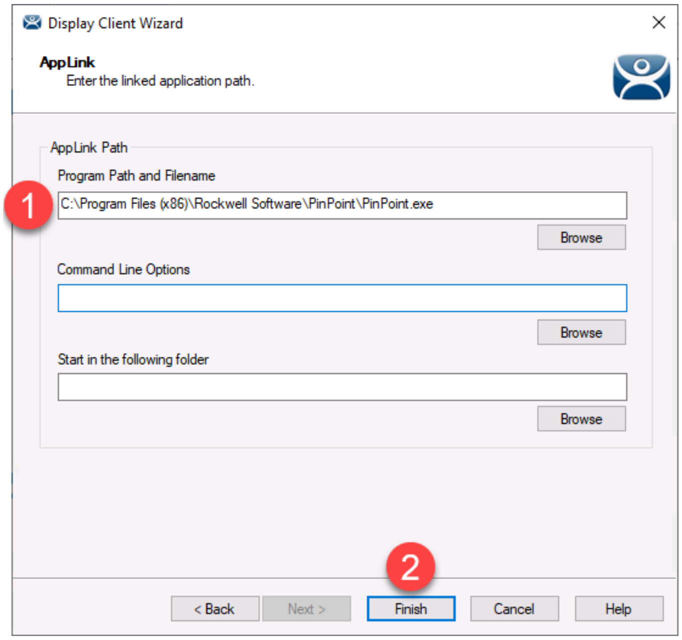
Display Client Members
Select the Remote Desktop Servers for this Display Client.

Available Remote Desktop Servers
RDS1 (172.29.47.13)
RDS2 (172.29.47.14)

Selected Remote Desktop Servers
TMAIO (172.29.47.63)

Double-click Display Servers listed here to add them to the Selected Remote Desktop Servers list

6. Click the **Next** button to get to the AppLink page of the Display Client Wizard. Enter the following value. Click the **Finish** button.
    a. Program Path and Filename field: `C:\Program Files (x86)\Rockwell Software\PinPoint\PinPoint.exe`



Display Client Wizard

AppLink
Enter the linked application path.

AppLink Path

Program Path and Filename: C:\Program Files (x86)\Rockwell Software\PinPoint\PinPoint.exe

Command Line Options:

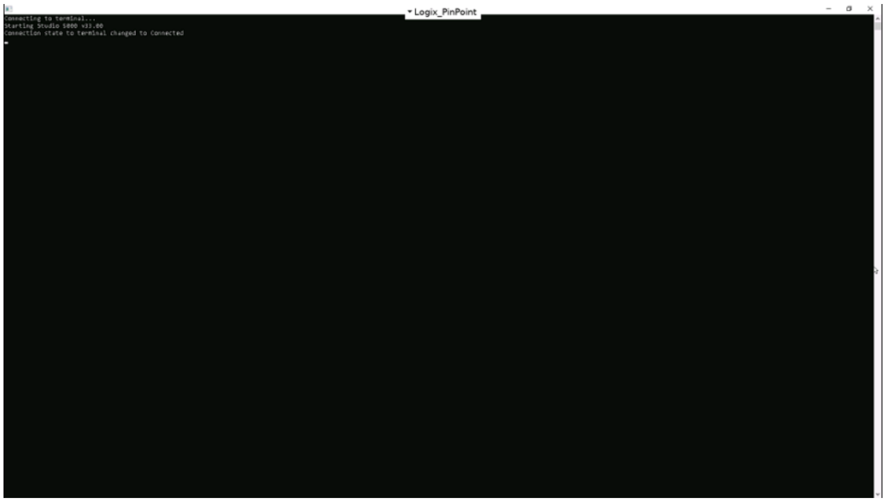
Start in the following folder:

Note: you can also add the initial Studio 5000 .acd file to launch in the **Command Line Options** field: `--InitialProject "<Path to your Studio 5000 .acd file>"`. This option is also configurable in the application settings explained later in the *PinPoint Settings and Configuration* section of this document.

7. After a Display Client is added to a terminal, this is how ThinManager Logix PinPoint will appear in the session. Note that it is maximized and will require that an operator at the terminal minimize it to see Studio 5000 Logix Designer. For a strategy to start this application minimized, please see the *Example With Batch File* section in this document.

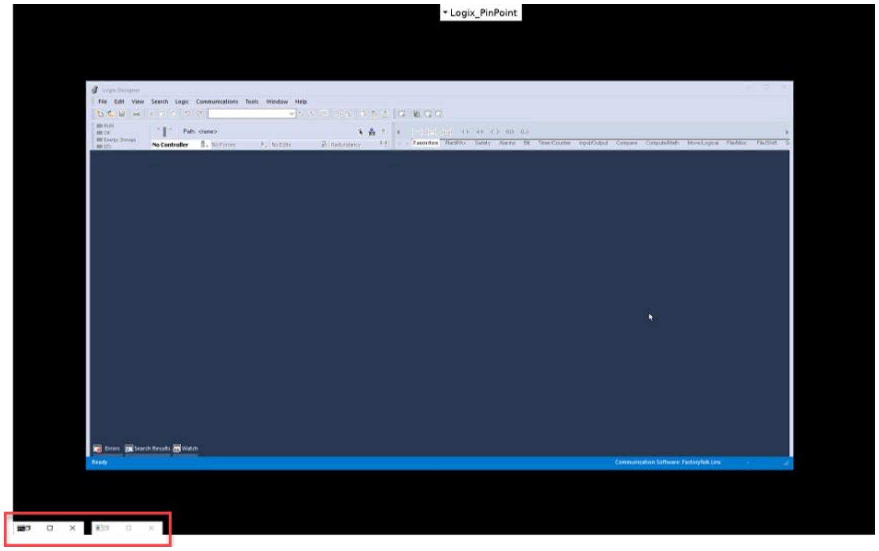
Create a Command-line ThinManager Logix PinPoint Display Client

There are two options for launching the ThinManager Logix PinPoint command-line executable. First, you can launch it directly using the ThinManager AppLink feature. This is the easiest way to configure the Display Client, but the downside is that the Studio 5000 Logix Designer window will open behind the command-line window as is shown below in the screenshot:



```
Connecting to terminal...
Starting Studio 5000 v33.00
Connection state to terminal changed to Connected
```

Thus, every time the Display Client is launched you will have to minimize the command-line window. A second option is to create a Display Client that launches a command prompt batch (.bat) file that launches minimized so that the Studio 5000 Logix Designer window is at the forefront of the Display Client session.



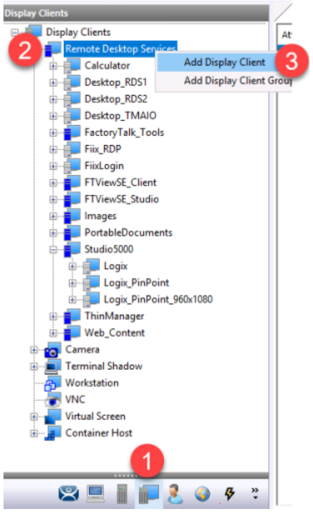
Example With Batch File

1. Create a batch file using Notepad or some other text editor. Enter the following syntax and save it as a .bat file.

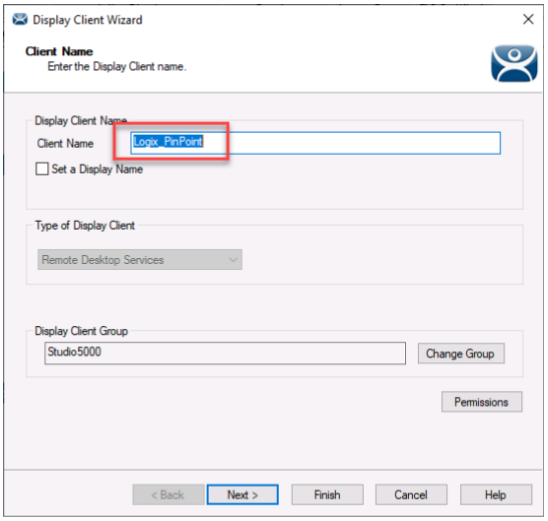
   ```
   @echo off
   title PinPoint launch script
   start /min "" /wait "C:\Program Files (x86)\Rockwell
   Software\PinPoint\PinPointCmd.exe" --InitialProject "<Path to
   your Studio 5000 .acd file>"
   shutdown /f /l
   ```

   Note: if you are using this Batch file for the ThinManager Logix PinPoint application with UI, change the executable name from `PinPointCmd.exe` to `PinPoint.exe`.

2. Launch the ThinManager Admin Console.
3. Navigate to the Display Clients section of the console by clicking on the **Display Clients** icon at the bottom left of the screen. Right-click **Remote Desktop Services** and choose **Add Display Client**.



4. Enter in any name you choose into the Client Name field.



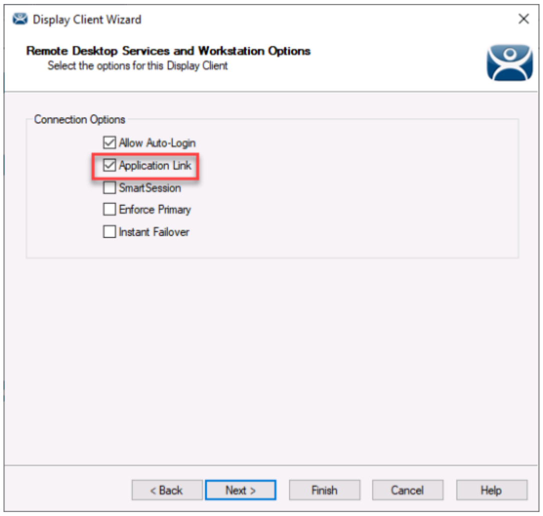
Client Name
Enter the Display Client name.

Display Client Name
Client Name: Logix_PinPoint
□ Set a Display Name

Type of Display Client
Remote Desktop Services

Display Client Group
Studio5000

5. Click the **Next** button until you get to the Remote Desktop Services and Workstation Options page of the Display Client Wizard. Put a check in the **Application Link** box.



Display Client Wizard

Remote Desktop Services and Workstation Options

Select the options for this Display Client

Connection Options

☑ Allow Auto-Login

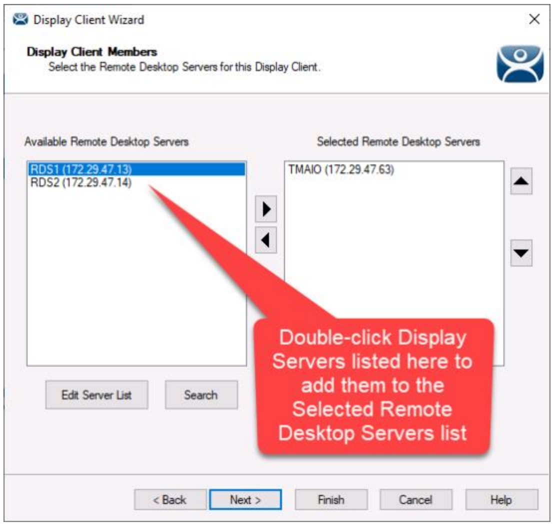
☑ Application Link

☐ SmartSession

☐ Enforce Primary

☐ Instant Failover

6. Click the **Next** button until you get to the Display Client Members page of the Display Client Wizard. **Double-click** on any servers in the Available Remote Desktop Server list to add them to the Selected Remote Desktop Server list. These should be servers that have the ThinManager Logix PinPoint and the Studio 5000 Logix Designer applications installed on them.



Display Client Wizard

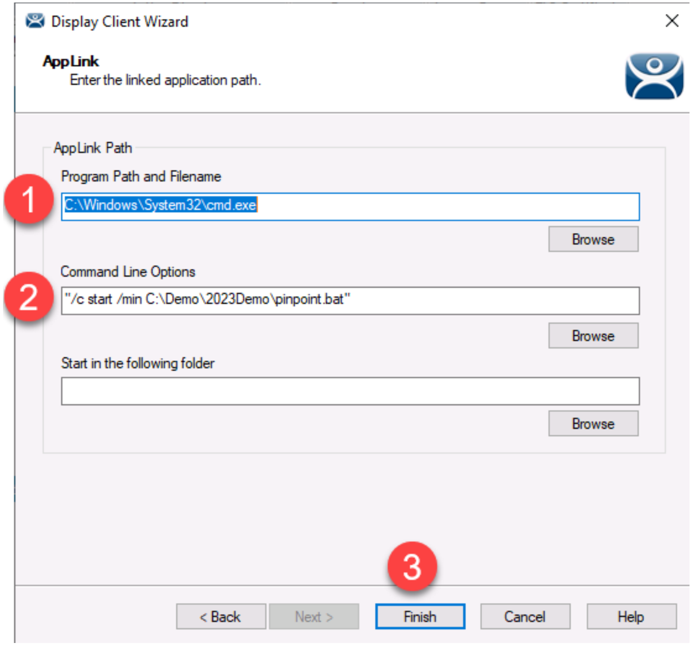
Display Client Members
Select the Remote Desktop Servers for this Display Client.

Available Remote Desktop Servers
RDS1 (172.29.47.13)
RDS2 (172.29.47.14)

Selected Remote Desktop Servers
TMAIO (172.29.47.63)

Double-click Display Servers listed here to add them to the Selected Remote Desktop Servers list

7. Click the **Next** button to get to the AppLink page of the Display Client Wizard. Enter in the following values. Click the **Finish** button.
    a. Program Path and Filename field: `C:\Windows\System32\cmd.exe`
    b. Command Line Options: `"/c start /min <path to your batch file>"`



Display Client Wizard

AppLink
Enter the linked application path.

AppLink Path

Program Path and Filename: C:\Windows\System32\cmd.exe

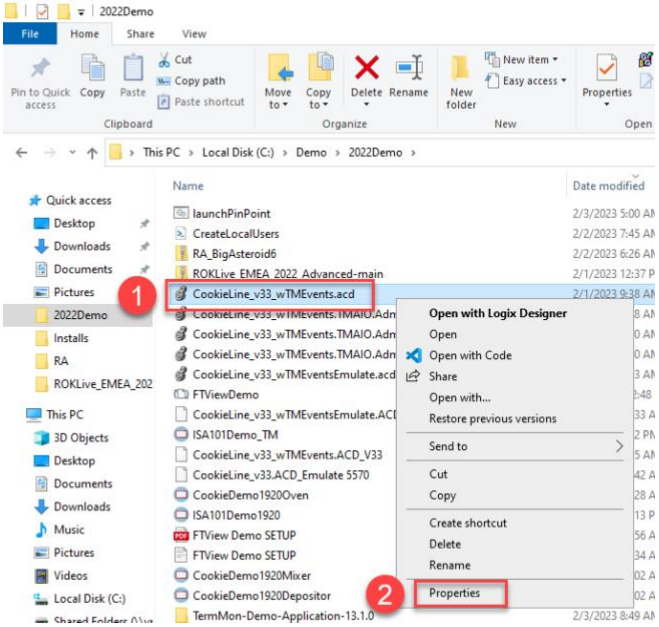
Command Line Options: "/c start /min C:\Demo\2023Demo\pinpoint.bat"

Start in the following folder:

File Permissions

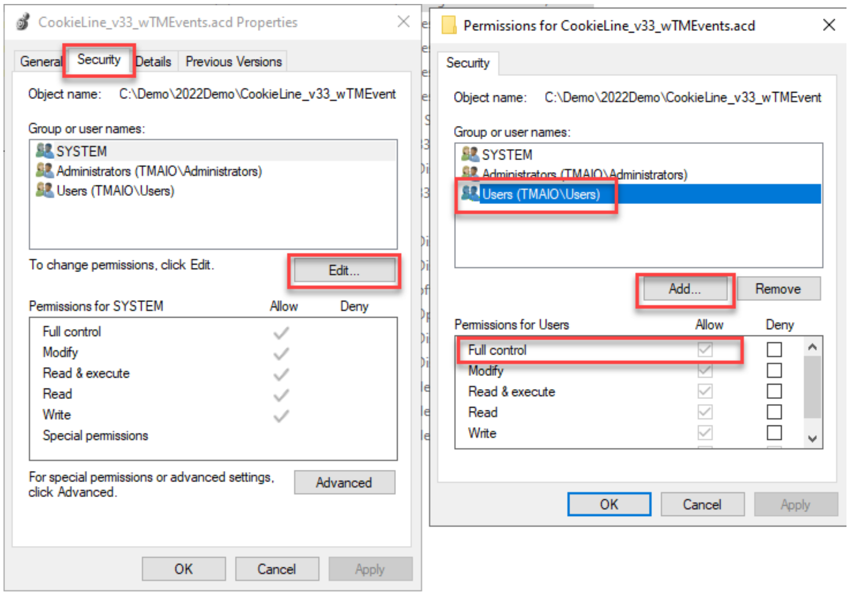
You need to grant all users permission to the .acd file for ThinManager Logix PinPoint to launch it. To do this:

1. Navigate to the .acd file in Windows Explorer. Right-click on the .acd file and select Properties.



Name	Date modified
launchPinPoint	2/3/2023 5:00 AM
CreateLocalUsers	2/2/2023 7:45 AM
RA_BigAsteroid6	2/2/2023 6:26 AM
ROKLive_EMEA_2022_Advanced-main	2/1/2023 12:37 P
CookieLine_v33_wTMEvents.acd	2/1/2023 9:38 AM
CookieLine_v33_wTMEvents.TMAIO.Adm	
CookieLine_v33_wTMEvents.TMAIO.Adm	
CookieLine_v33_wTMEvents.TMAIO.Adm	
CookieLine_v33_wTMEventsEmulate.acd	
FTViewDemo	
CookieLine_v33_wTMEventsEmulate.ACI	
ISA101Demo_TM	
CookieLine_v33_wTMEvents.ACD_V33	
CookieLine_v33.ACD_Emulate 5570	
CookieDemo1920Oven	
ISA101Demo1920	
FTView Demo SETUP	
FTView Demo SETUP	
CookieDemo1920Mixer	
CookieDemo1920Depositor	
TermMon-Demo-Application-13.1.0	2/3/2023 8:49 AM

2. Click on the **Security** tab and then the **Edit** button. Make sure that Users have full control of the file by putting a check in the Full Control check box. You may need to add Users to the list using the Add button. Click the **OK** button on both windows to close them when you are finished.



Object name: C:\Demo\2022Demo\CookieLine_v33_wTMEvent

Group or user names:
SYSTEM
Administrators (TMAIO\Administrators)
Users (TMAIO\Users)

To change permissions, click Edit.

Permissions for SYSTEM	Allow	Deny
Full control	✓	
Modify	✓	
Read & execute	✓	
Read	✓	
Write	✓	
Special permissions		

For special permissions or advanced settings, click Advanced.

Object name: C:\Demo\2022Demo\CookieLine_v33_wTMEvent

Group or user names:
SYSTEM
Administrators (TMAIO\Administrators)
Users (TMAIO\Users)

Permissions for Users	Allow	Deny
Full control	☑	□
Modify	☑	□
Read & execute	☑	□
Read	☑	□
Write	☑	□

Note: Only one instance of the .acd file can be launched per user session. If you want to launch multiple instances of the FactoryTalk Logix PinPoint application with the same .acd file, you must make unique copies of that .acd file to use for each instance of FactoryTalk Logix PinPoint.

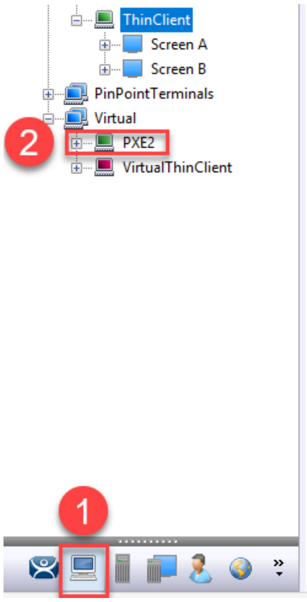
Event Customization

In this section of the document, we will outline how to deploy the PinPoint Display Client as well as customize how it is delivered to a terminal.

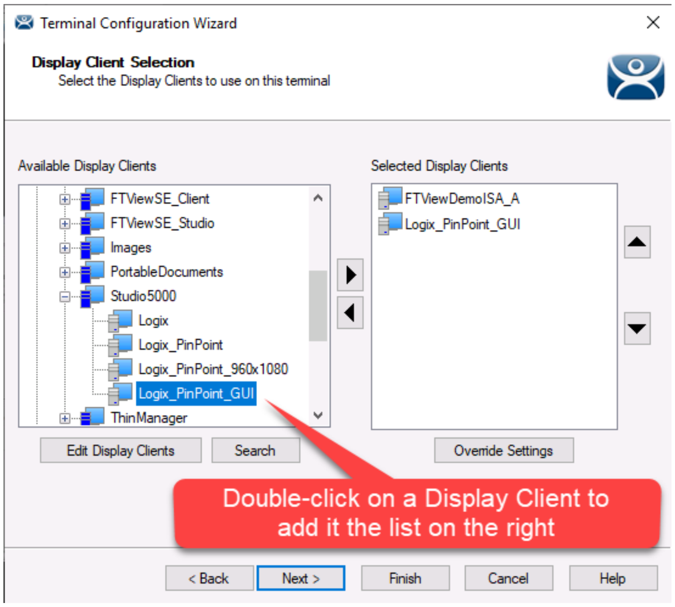
Add Display Client to the Terminal

Once you have created a ThinManager Logix PinPoint Display Client, it's time to add it to a terminal.

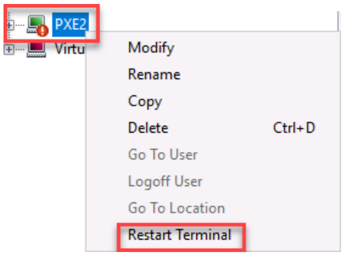
1. Open the ThinManager Administration Console.
2. Click on the **Terminals** icon at the bottom left and double-click on any terminal that you have already configured.



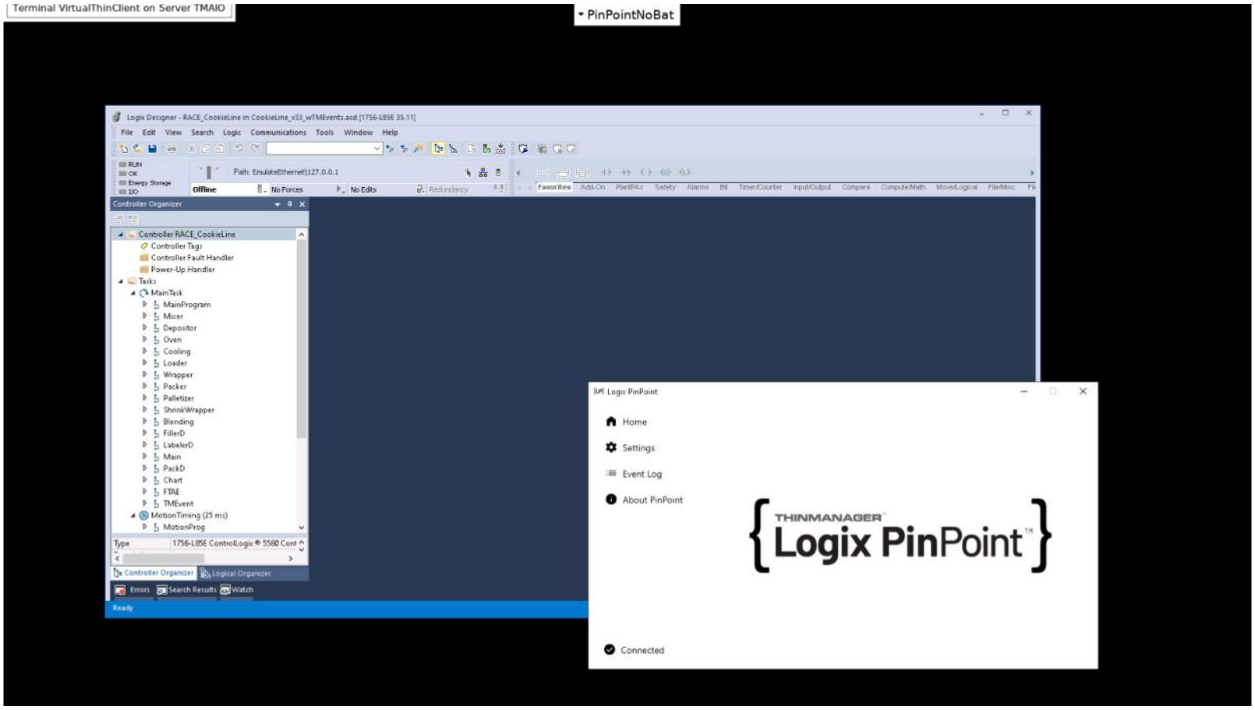
3. Click the **Next** button until you get to the Display Client Selection page of the Terminal Configuration Wizard. Browse the Available Display Clients list to find the ThinManager Logix PinPoint Display Client you created in the Create a ThinManager Logix PinPoint Display Client section. Double-click the Display Client to add it to the Selected Display Clients list.



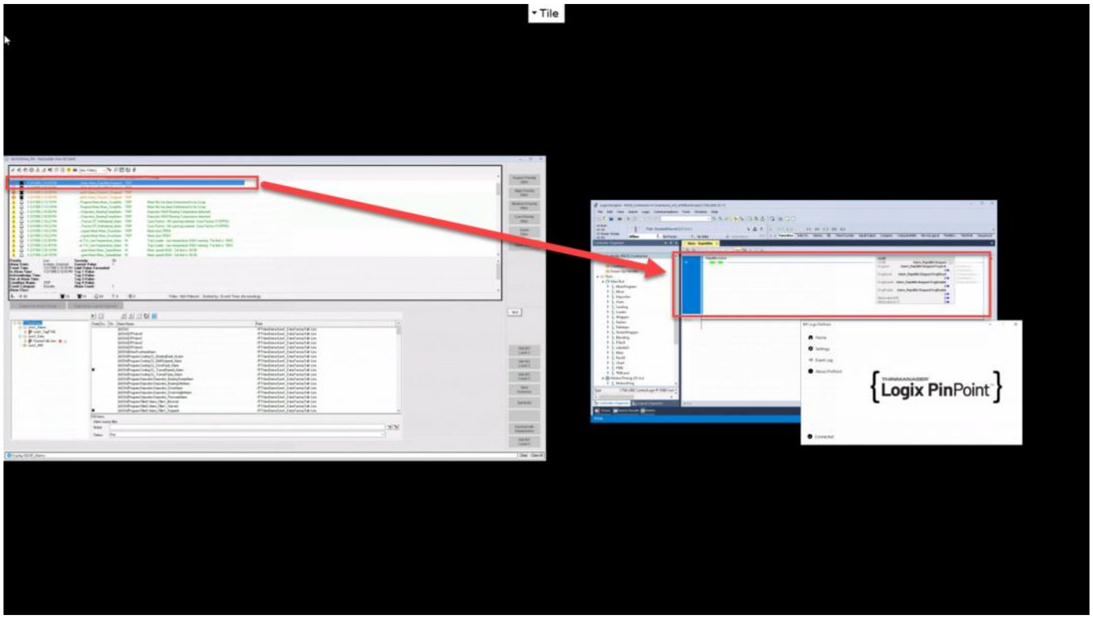
4. Adjust any other terminal settings needed. You can, for example, specify the user account needed to launch the remote Windows session by clicking the **Next** button until you get to the Log In Information page of the Terminal Configuration Wizard. Enter in and verify the Windows Log In Information accordingly.
5. Click the **Finish** button to close the Terminal Configuration Wizard whenever you are ready to test your new terminal settings.
6. Right-click on the terminal in the ThinManager Administration Console and select **Restart** to apply the changes to the terminal.



7. The terminal should now load ThinManager Logix PinPoint running with the specified Initial Project (if specified). Note that your screenshot may look different from the one below, depending on if you minimized the ThinManager Logix PinPoint Application using the .bat file method described in this document, and if you chose to use the GUI or command-line version of the ThinManager Logix PinPoint application. This screenshot is just an illustration of how there should be two windows opened after launching.



Test your setup to verify that it works. We'll use the FactoryTalk View SE HMI with Alarm Summary as an example. Tiling is configured on the terminal to show that when you click on an alarm on the left, you see the associated rung of code displayed on the right:



PinPoint Settings and Configuration

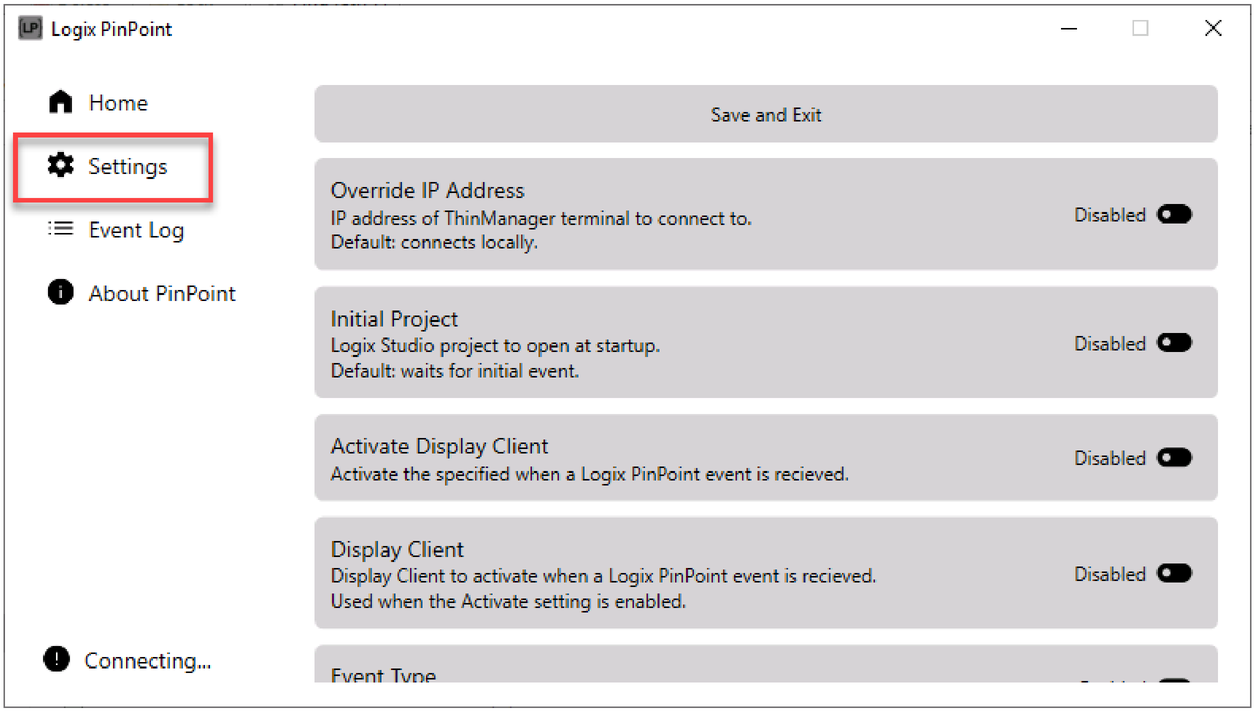
PinPoint settings are specified in a few different ways:

1. ThinManager Logix PinPoint UI (or associated JSON settings file)
2. Command-line options in the Display Client configuration
3. VBA code or ThinManager Event arguments

The order the settings and parameters is interpreted are listed the same as above, so if you pass an argument in VBA code, and that same setting was specified in the JSON file, the VBA code will take precedence.

ThinManager Logix PinPoint UI & JSON File

Click on the Settings button in the ThinManager Logix PinPoint application to get access to the configurable parameters for ThinManager Logix PinPoint.



Override IP Address
IP address of ThinManager terminal to connect to.
Default: connects locally. Disabled

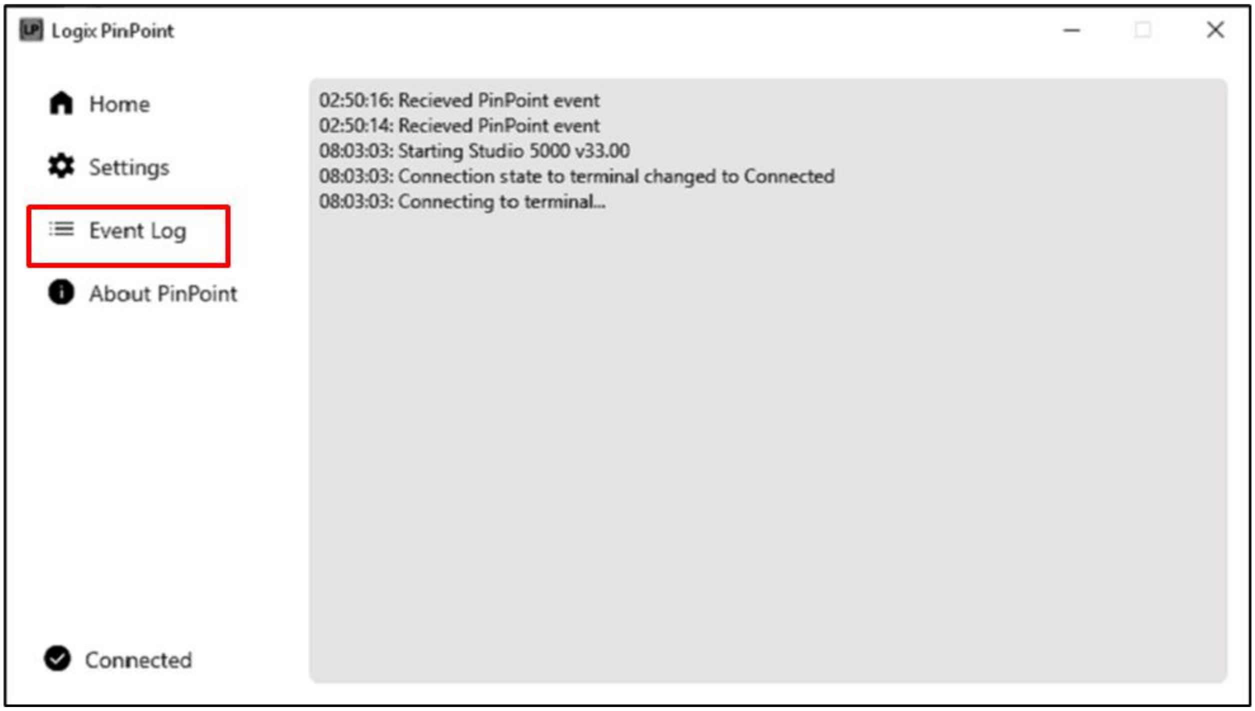
Initial Project
Logix Studio project to open at startup.
Default: waits for initial event. Disabled

Activate Display Client
Activate the specified when a Logix PinPoint event is recieved. Disabled

Display Client
Display Client to activate when a Logix PinPoint event is recieved.
Used when the Activate setting is enabled. Disabled

Event Type

The log for each user's instance of ThinManager Logix PinPoint can be found by clicking on the **Event Log** button in the ThinManager Logix PinPoint UI:



```
02:50:16: Recieved PinPoint event
02:50:14: Recieved PinPoint event
08:03:03: Starting Studio 5000 v33.00
08:03:03: Connection state to terminal changed to Connected
08:03:03: Connecting to terminal...
```

The settings and log file for PinPoint is found by navigating to the AppData folder for each user that runs the application: `C:\Users\[UserName]\AppData\Local\Rockwell Software\PinPoint\settings.json`

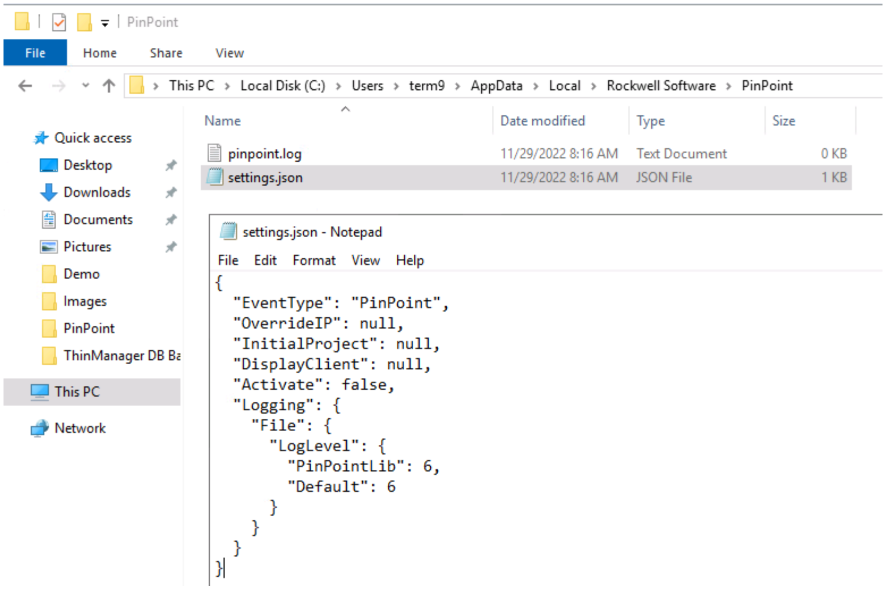
Here are the options that can be configured with ThinManager Logix PinPoint:

Parameter name	Description
`EventType`	Always set to PinPoint, do not change
`OverrideIP`	The IP address of the terminal that ThinManager Logix PinPoint needs to connect. The default is blank, which connects it to the terminal that the ThinManager Logix PinPoint application is running on.
`InitialProject`	The Initial ACD file to launch in Studio 5000 Logix Designer.
`DisplayClient`	The DisplayClient name to bring to the forefront or make active if the Activate parameter is set to `true`.
`Activate`	By default `no`. Setting it to `yes` will bring a specified Display Client to the foreground, making it the active Display Client on the terminal.

To change how logging behaves, you may edit the parameters in the “Logging” area of the JSON file using the parameters listed in the table below. These settings only affect what is logged to the text file in the user’s PinPoint folder:

Parameter name	Description
`File`	Not configurable, do not change. Used internally to configure the File logger.
`PinPointLib`	Logging from the PinPointLib library. A value of 1 through 6 is specified, with 1 being the most detailed logging, and 6 turning logging off.
`Default`	Logging from everything, including the Microsoft .NET libraries used by ThinManager Logix PinPoint. A value of 1 through 6 is specified, with 1 being the most detailed logging, and 6 turning logging off.

The UI version of the ThinManager Logix PinPoint application modifies this JSON file, so making changes in either is equivalent. Here is what a typical ThinManager Logix PinPoint JSON file looks like:



This PC › Local Disk (C:) › Users › term9 › AppData › Local › Rockwell Software › PinPoint

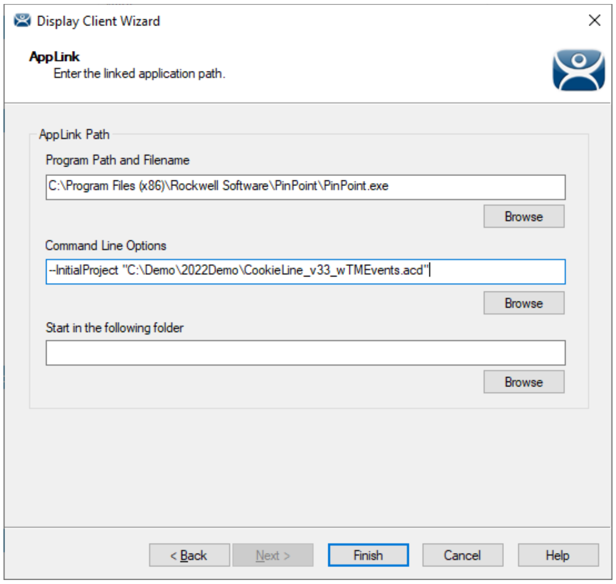
Name	Date modified	Type	Size
pinpoint.log	11/29/2022 8:16 AM	Text Document	0 KB
settings.json	11/29/2022 8:16 AM	JSON File	1 KB

```
{
  "EventType": "PinPoint",
  "OverrideIP": null,
  "InitialProject": null,
  "DisplayClient": null,
  "Activate": false,
  "Logging": {
    "File": {
      "LogLevel": {
        "PinPointLib": 6,
        "Default": 6
      }
    }
  }
}
```

Command-line Options

Command-line options are another way to configure the behavior of FactoryTalk Logix PinPoint. These are put into the Command Line Options field of your ThinManager Logix PinPoint Display Client, or they are passed as arguments in a .bat file. Below are two examples with screenshots, one with arguments passed through the Display Client configuration, the other through a .bat file that the Display Client launches.

Pass arguments through Display Client Configuration:



Display Client Wizard

AppLink
Enter the linked application path.

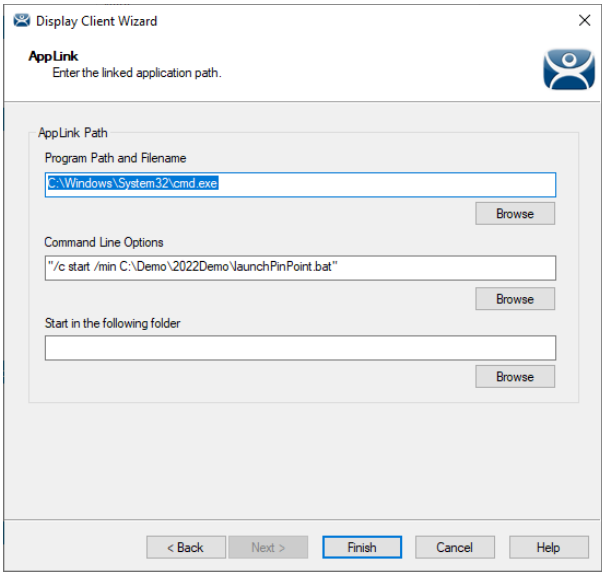
AppLink Path

Program Path and Filename: C:\Program Files (x86)\Rockwell Software\PinPoint\PinPoint.exe

Command Line Options: --InitialProject "C:\Demo\2022Demo\CookieLine_v33_wTMEvents.acd"

Start in the following folder:

Pass arguments through a .bat file. In the example below, the Display Client opens command prompt which then executes the .bat file minimized. The arguments for the PinPoint application are passed through the .bat file while it executes in command prompt.



Display Client Wizard

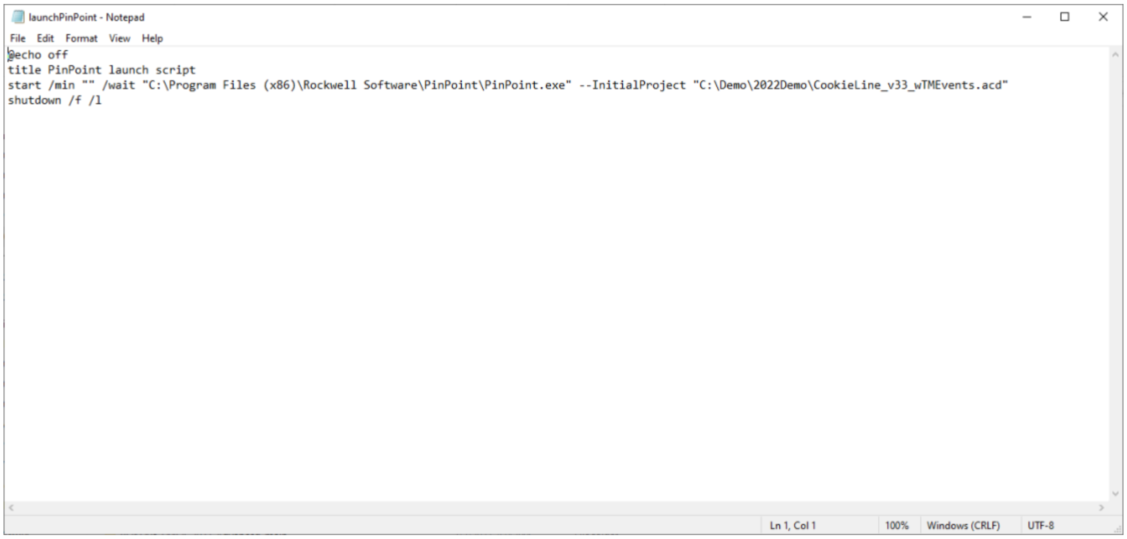
AppLink
Enter the linked application path.

AppLink Path

Program Path and Filename: C:\Windows\System32\cmd.exe

Command Line Options: "/c start /min C:\Demo\2022Demo\launchPinPoint.bat"

Start in the following folder:



```
@echo off
title PinPoint launch script
start /min "" /wait "C:\Program Files (x86)\Rockwell Software\PinPoint\PinPoint.exe" --InitialProject "C:\Demo\2022Demo\CookieLine_v33_wTMEvents.acd"
shutdown /f /l
```

VBA

Here is an explanation of the options that you can configure with the PinPoint method in the VBA code:

Function **PinPoint**(*NavType, TagName, ScopeName, EventArgs*) As Integer

NavType (enum) – Value ranging from 1 to 5 meaning (this parameter is used with version 34.01 of Studio 5000 software or higher, otherwise it is ignored):

1. Tag definition in the .acd file
2. First reference bit in the .acd file
3. Next reference bit in the .acd file
4. First destructive bit in the .acd file (Requires Logix Designer v35.00.00)
5. Next destructive bit in the .acd file (Requires Logix Designer v35.00.00)

TagName (String) – The name of the tag that you want to reference in the ACD file

ScopeName (String) – The scope of the tag you want to reference in the ACD file

EventArgs (String) – Arguments that you specify that should be passed to the Event on the ThinManager server

Here is a table of additional Event arguments that you can pass with the PinPoint VBA method using the EventArgs field:

Argument	**Description**
`Project`	The path to the ACD file you want to launch in Studio 5000 Logix Designer.
`Activate`	By default `no`. Setting to `yes` will bring a Display Client specified in the PinPoint settings configuration file to the foreground, making it the active Display Client on the terminal.

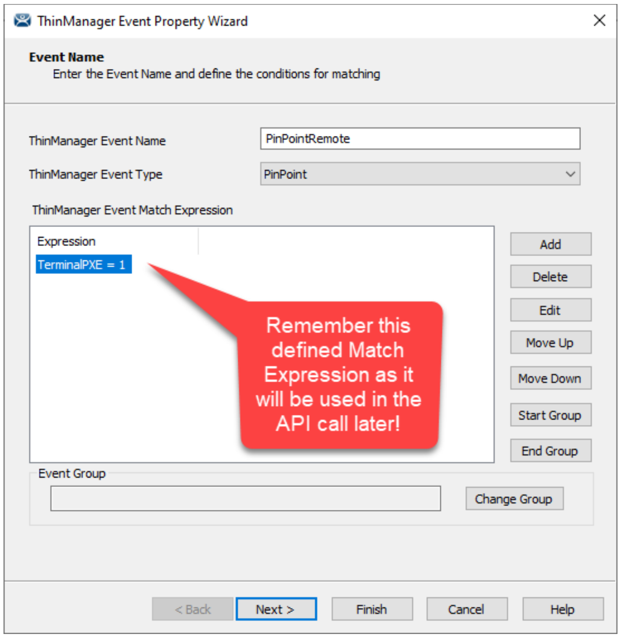
Note: If you are passing multiple arguments in the EventArgs field, each needs to be passed to the server as new lines of code (in most programming languages that would be `\n`). In VBA, to pass through a new line you concatenate the two strings with `& vbNewLine &`. Thus, the arguments passed to the Event are `Argumment1 & vbNewLine & Argument2`, where `Argument1` and `Argument2` are the strings that you are passing in quotes, such as `"Project=C:\Demo\2023Demo\CookieLine_v33_wTMEvents.acd"`. Multiple arguments are used in the *Other Terminal* section of this document.

Customized Event and Terminal Options

ThinManager Logix PinPoint is flexible. You can trigger ThinManager Logix PinPoint Events from the API. You can also use multiple monitors and/or mobile thin clients to trigger ThinManager Logix PinPoint Events on other terminals.

API

ThinManager version 13.0 and newer can send Events using the API. This includes ThinManager Logix PinPoint Events. The limitation of using the API is that it is not context aware. That means we need to specify which terminal the Event is sent to in the Event Configuration. This is done by calling a specific ThinManager Event for each terminal you want to execute the ThinManager Logix PinPoint Event on. Here is an example of an Event that was created on the ThinManager server for the API:



ThinManager Event Property Wizard

Event Name
Enter the Event Name and define the conditions for matching

ThinManager Event Name: PinPointRemote

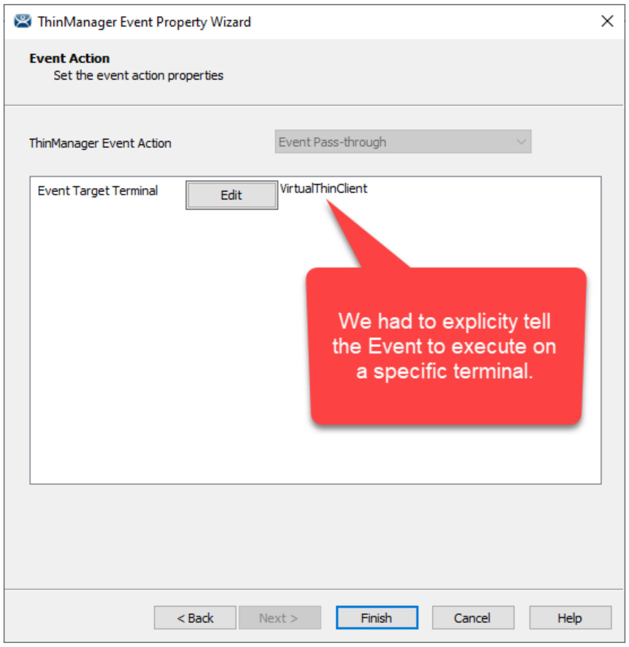
ThinManager Event Type: PinPoint

ThinManager Event Match Expression

Expression
TerminalPXE = 1

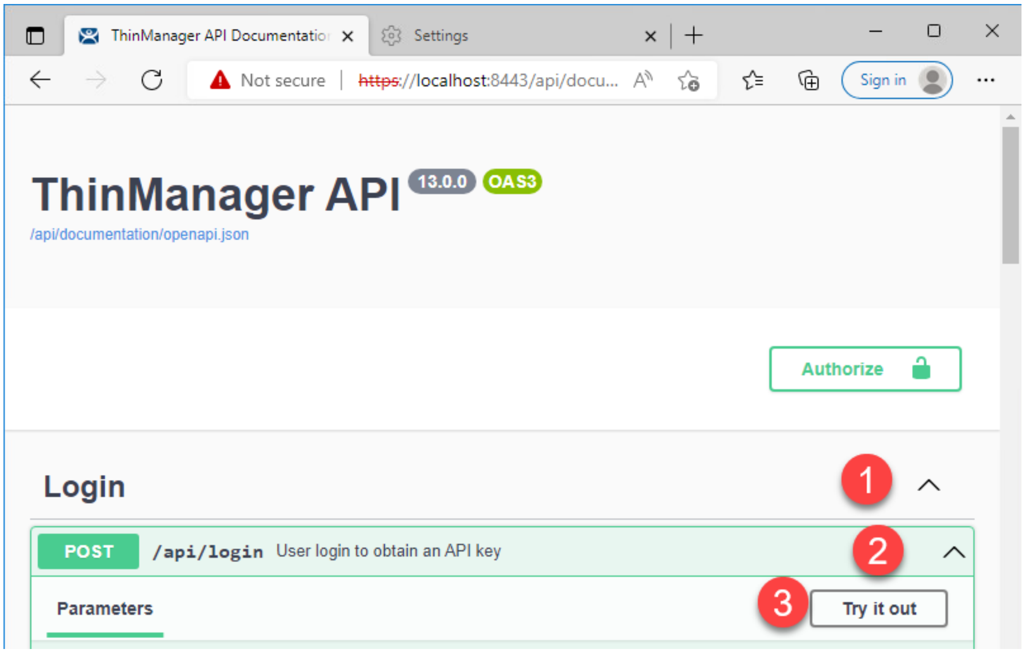
Event Group

Remember this defined Match Expression as it will be used in the API call later!


ThinManager Event Property Wizard
Event Action
Set the event action properties
ThinManager Event Action
Event Pass-through
Event Target Terminal
Edit
VirtualThinClient
We had to explicity tell the Event to execute on a specific terminal.
< Back
Next >
Finish
Cancel
Help

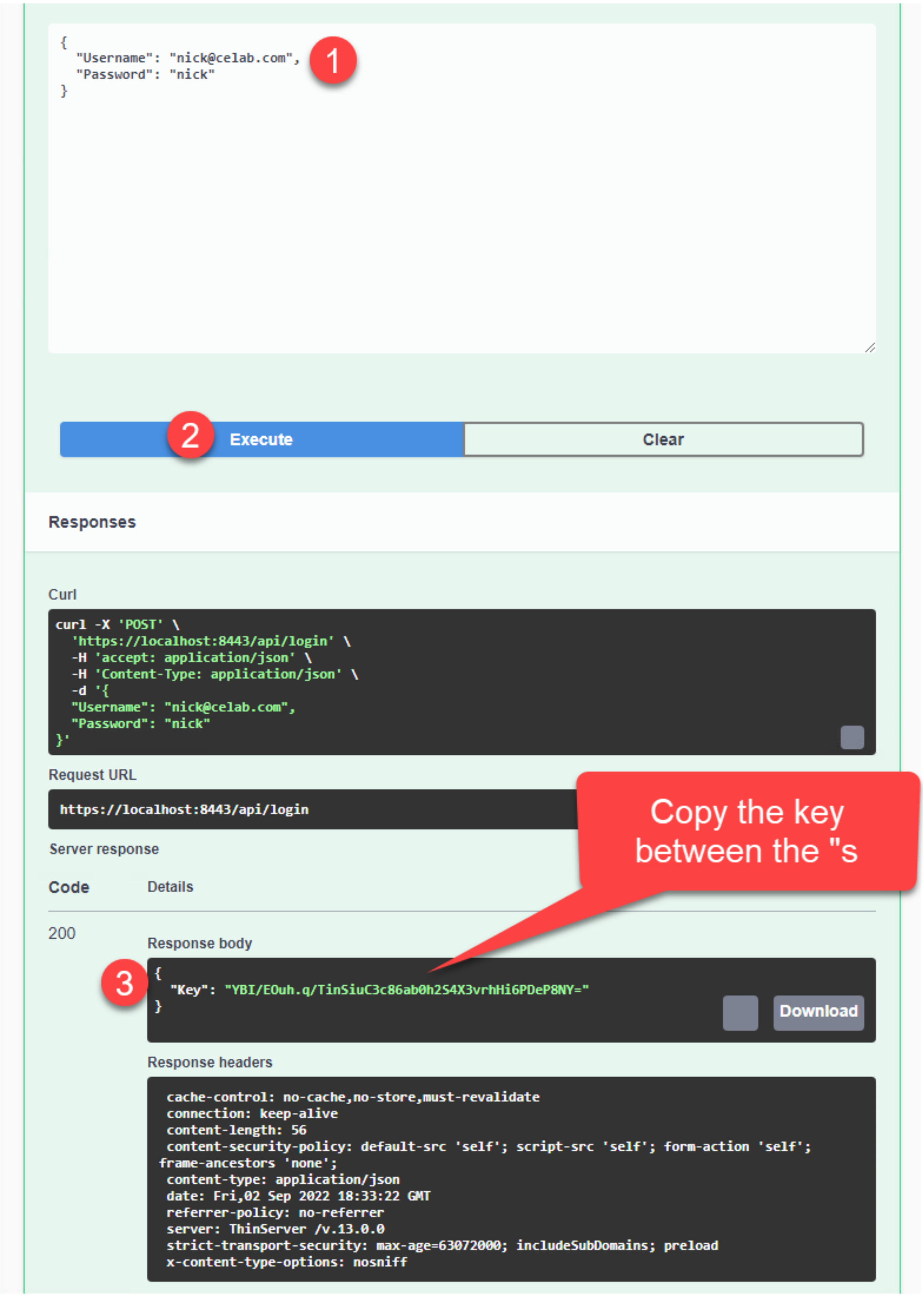
For this example, we will use the ThinManager API webpage to test triggering a ThinManager Logix PinPoint Event. We navigate to https://localhost:8443/api/documentation. Since this is running a Swagger web interface, we can test the code. To test the code, you need to authenticate with the ThinManager server using a certificate. You can do this by using the webpage to log in and generate a key for you:

1. Expand the Login section of the webpage by clicking on the **Login** bar and then on the **POST /api/login** bar to expand it. Click on the **Try it out** button.


ThinManager API Documentation
Settings
Not secure
https://localhost:8443/api/docu...
Sign in
ThinManager API
13.0.0
OAS3
/api/documentation/openapi.json
Authorize
Login
1
POST
/api/login
User login to obtain an API key
2
Parameters
3
Try it out

2. Replace the username and password with their respective values. Press the **Execute** button. Scroll down to the Server response section and copy and paste the key found in between the “”s of the text within the Response body box.

a. Note: You should use your UPN (User Principal Name) login account for Windows to log into the ThinManager server with the API. For example, domainname\user would be represented as user@domainname.com if you are using the UPN form. This user also needs to have permission to connect to and administer the ThinManager server as is set up in the ThinManager Security section of the ThinManager server.



```
{
  "Username": "nick@celab.com",
  "Password": "nick"
}
```

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:8443/api/login' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "Username": "nick@celab.com",
  "Password": "nick"
}'
```

Request URL

```
https://localhost:8443/api/login
```

Copy the key between the "s

Server response

Code Details

200

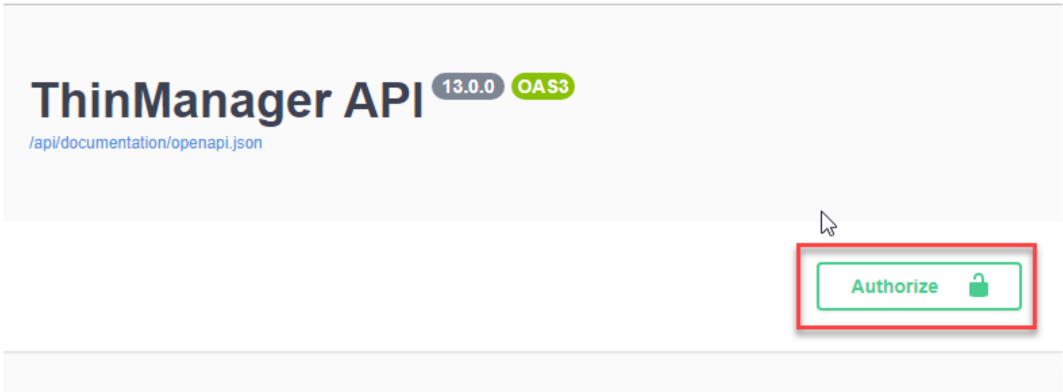
Response body

```
{
  "Key": "YBI/EOuh.q/TinSiuC3c86ab0h2S4X3vrhHi6PDeP8NY="
}
```

Response headers

```
 cache-control: no-cache,no-store,must-revalidate
 connection: keep-alive
 content-length: 56
 content-security-policy: default-src 'self'; script-src 'self'; form-action 'self';
frame-ancestors 'none';
 content-type: application/json
 date: Fri,02 Sep 2022 18:33:22 GMT
 referrer-policy: no-referrer
 server: ThinServer /v.13.0.0
 strict-transport-security: max-age=63072000; includeSubDomains; preload
 x-content-type-options: nosniff
```

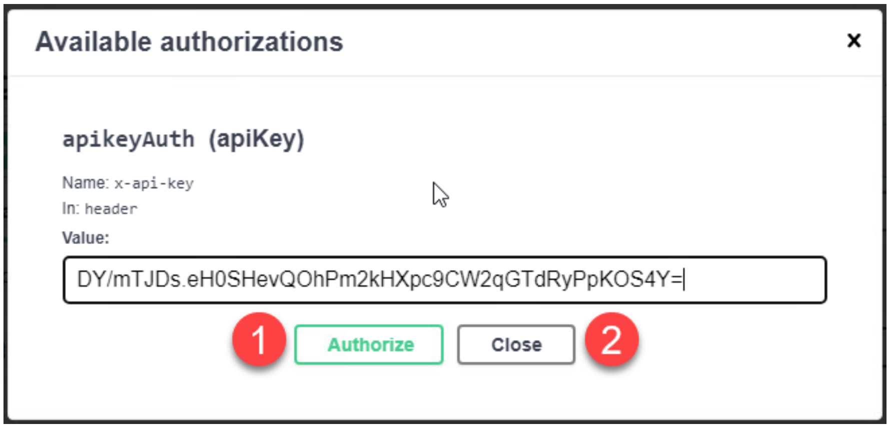
3. Scroll up to the top of the webpage and click on the **Authorize** button.



ThinManager API 13.0.0 OAS3

/api/documentation/openapi.json

4. Paste in the key copied previously and click the **Authorize** followed by the **Close** button.



Available authorizations

apikeyAuth (apiKey)

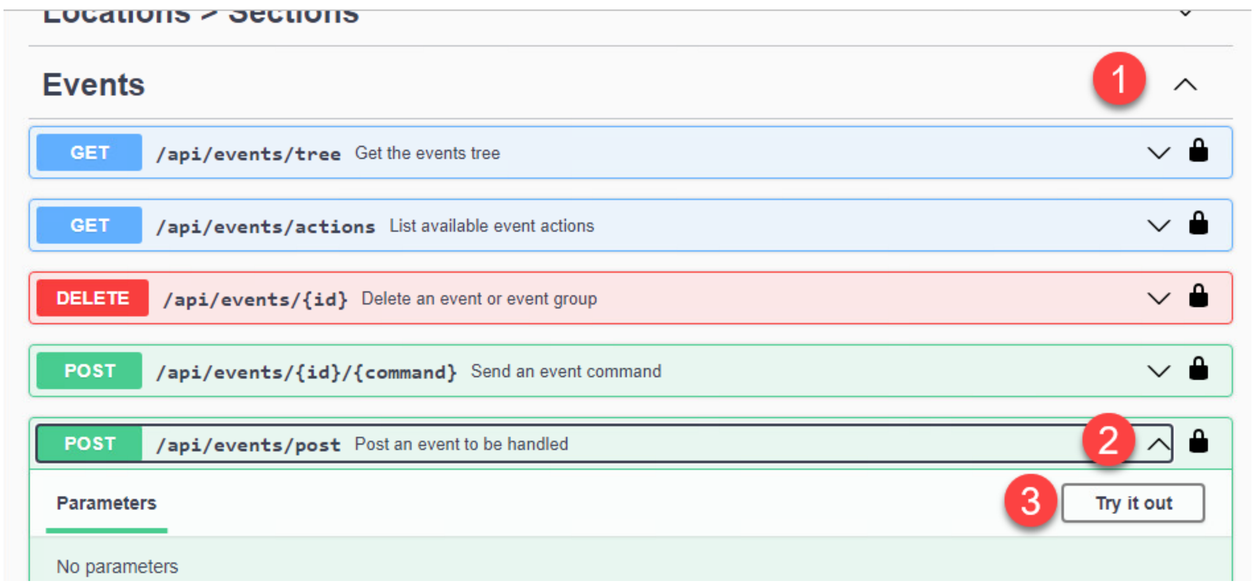
Name: x-api-key

In: header

Value:

DY/mTJDs.eH0SHevQOhPm2kHXpc9CW2qGTdRyPpKOS4Y=

Once the server is authorized, we can trigger an Event by going to the **Events -> POST /api/events/post** method, which will post an Event to be handled by the ThinManager server:



Locations > Sections

Events

GET /api/events/tree Get the events tree

GET /api/events/actions List available event actions

DELETE /api/events/{id} Delete an event or event group

POST /api/events/{id}/{command} Send an event command

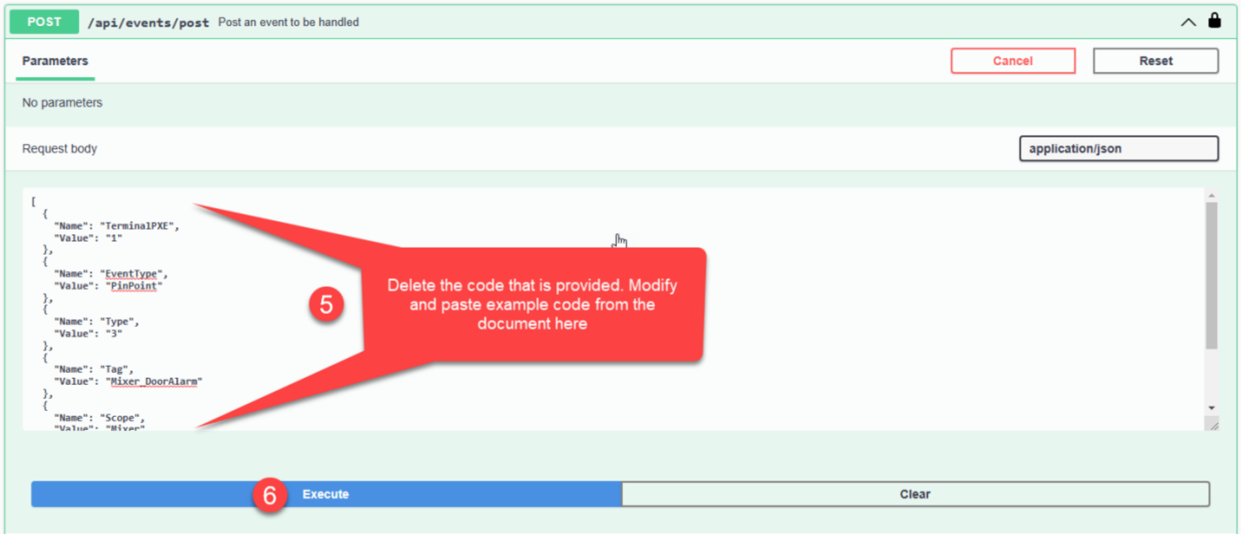
POST /api/events/post Post an event to be handled

Parameters

No parameters

Modify the following code, replacing expressions between the <>s. Copy and paste into the API website and click **Execute**:

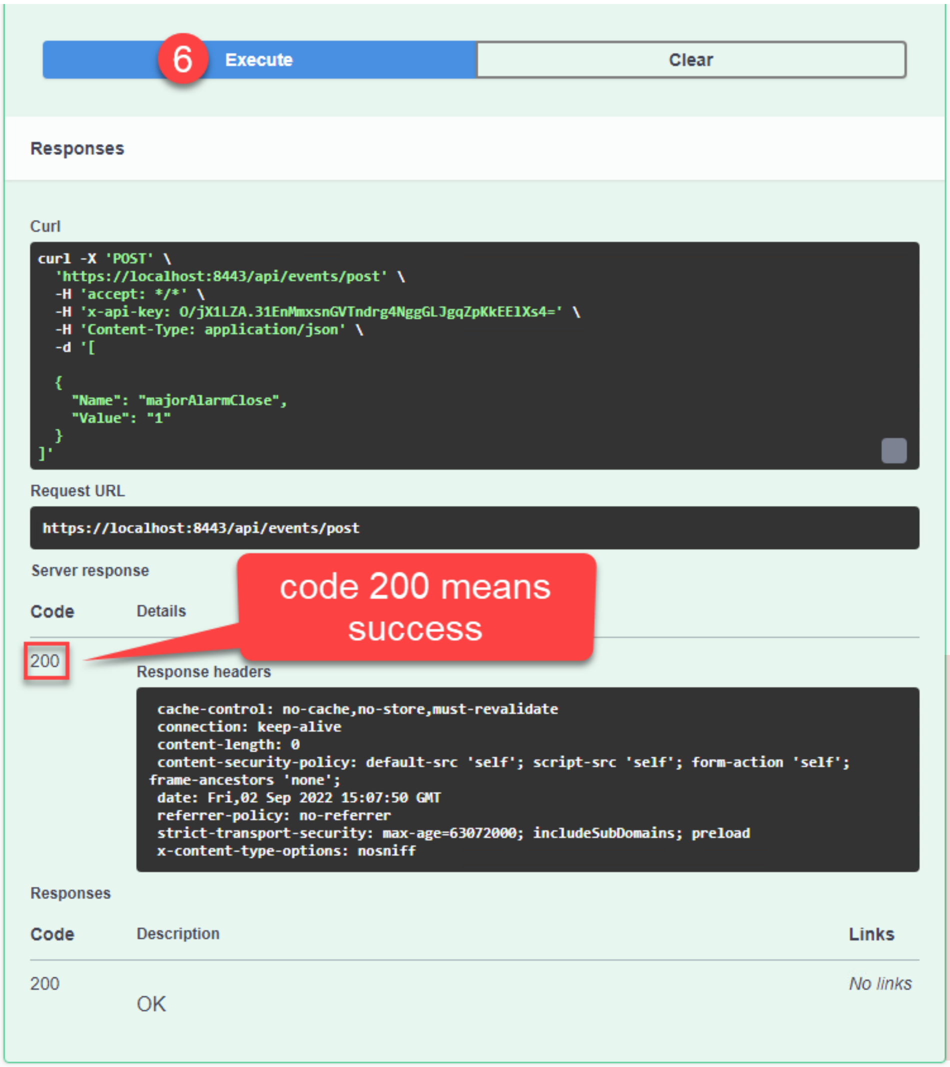
```
[
  {
    "Name": "<Match Expression Property Name>",
    "Value": "<Match Expression Compare Value>"
  },
  {
    "Name": "EventType",
    "Value": "PinPoint"
  },
  {
    "Name": "Type",
    "Value": "<Sepcify a value between 1 and 5 for navigation
type, see NavType (enum) in document>"
  },
  {
    "Name": "Tag",
    "Value": "<Sepcify the Tag name in the controller>"
  },
  {
    "Name": "Scope",
    "Value": "<Specify the Scope of the tag in the controller>"
  },
  {
    "Name": "Project",
    "Value": "<//Specify the .acd file path on your server, using
double back-slashes \\ instead of \ >"
  }
]
```



```
[
  {
    "Name": "TerminalPXE",
    "Value": "1"
  },
  {
    "Name": "EventType",
    "Value": "PinPoint"
  },
  {
    "Name": "Type",
    "Value": "3"
  },
  {
    "Name": "Tag",
    "Value": "Mixer_DoorAlarm"
  },
  {
    "Name": "Scope",
    "Value": "Mixer"
```

5 Delete the code that is provided. Modify and paste example code from the document here

6 Execute



Responses

Curl

```
curl -X 'POST' \
  'https://localhost:8443/api/events/post' \
  -H 'accept: */*' \
  -H 'x-api-key: O/jX1LZA.31EnMmxsnGVTndrg4NggGLJgqZpKkEE1Xs4=' \
  -H 'Content-Type: application/json' \
  -d '[

  {
    "Name": "majorAlarmClose",
    "Value": "1"
  }
]'
```

Request URL

```
https://localhost:8443/api/events/post
```

Server response

Code	Details
200	Response headers

```
 cache-control: no-cache,no-store,must-revalidate
 connection: keep-alive
 content-length: 0
 content-security-policy: default-src 'self'; script-src 'self'; form-action 'self';
frame-ancestors 'none';
 date: Fri,02 Sep 2022 15:07:50 GMT
 referrer-policy: no-referrer
 strict-transport-security: max-age=63072000; includeSubDomains; preload
 x-content-type-options: nosniff
```

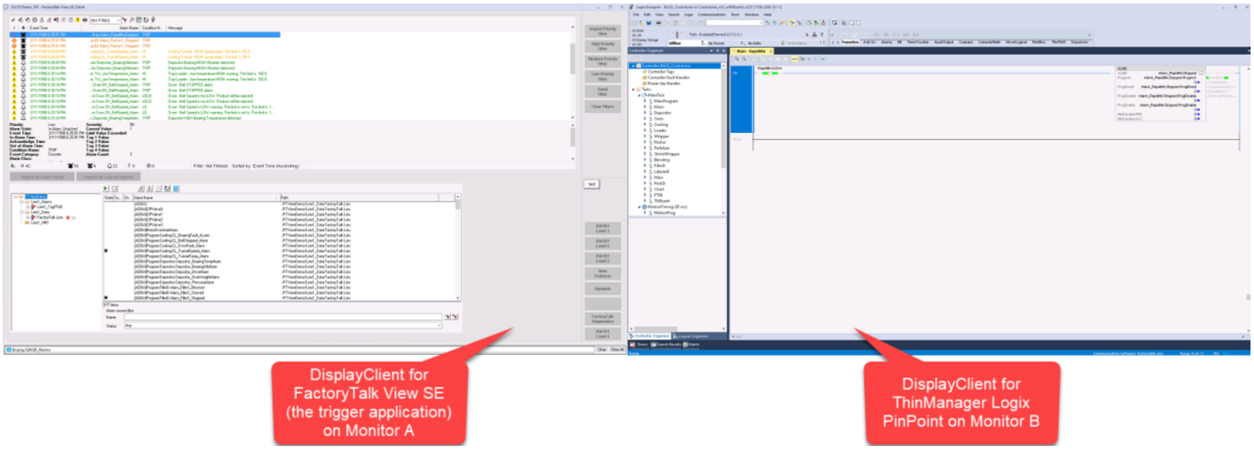
Responses

Code	Description	Links
200	OK	*No links*

A note that receiving code 200 means that the server was sent the Event and it was received, but it does not necessarily mean that an Event was triggered. The Event still must be valid by matching an expression the server is expecting. You should manually verify that the event was received by the ThinManager server by checking the Event Log tab for the Event you are trying to trigger.

Multimonitor

The triggering application and the ThinManager Logix PinPoint application are usually run as two separate AppLink Display Clients. That means that you can run the triggering application's Display Client on one monitor and the Display Client for ThinManager Logix PinPoint on another monitor. No special configuration is required on the ThinManager Logix PinPoint side or the Event side on the ThinManager server if you are passing the Event back to the source terminal. You simply configure the Display Clients to run on different monitors using the terminal's configuration.



DisplayClient for FactoryTalk View SE (the trigger application) on Monitor A

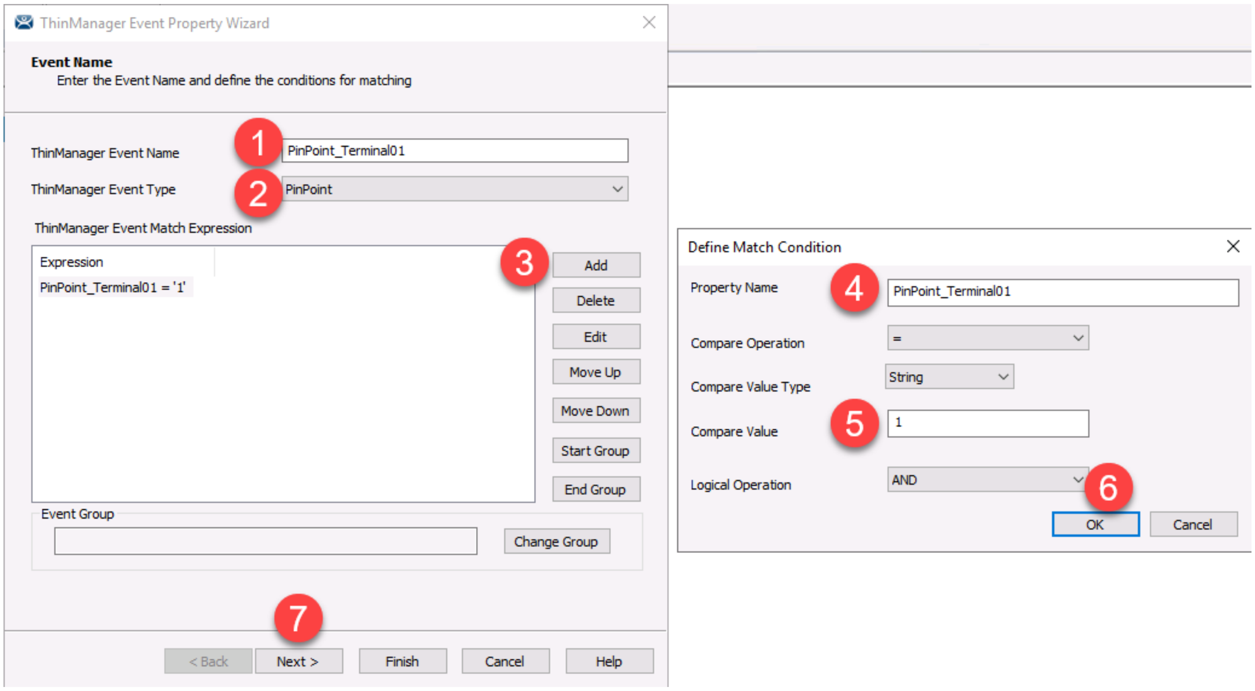
DisplayClient for ThinManager Logix PinPoint on Monitor B

Other Terminals and Multiple Events

You can trigger ThinManager Logix PinPoint to trigger on other ThinManager terminals. This can be done by creating multiple Events, one for each terminal you wish to execute the Event on. The catch is that you will be required to use a different VBA method to send these Events. The method arguments are arranged differently, and you will have one more Event property to pass as a result.

Add an event to your ThinManager using the same process as the Add Event to ThinManager Server section of this document used, but this time make sure to do the following:

- Specify a unique name for the event in the ThinManager Event Name field
- Set the ThinManager Event Type to PinPoint
- Create a unique ThinManager Event Match Expression using the Add button
- Set the Event Target Terminal to another terminal



ThinManager Event Property Wizard

Event Name
Enter the Event Name and define the conditions for matching

1 ThinManager Event Name: PinPoint_Terminal01

2 ThinManager Event Type: PinPoint

ThinManager Event Match Expression

Expression
PinPoint_Terminal01 = '1'

Define Match Condition

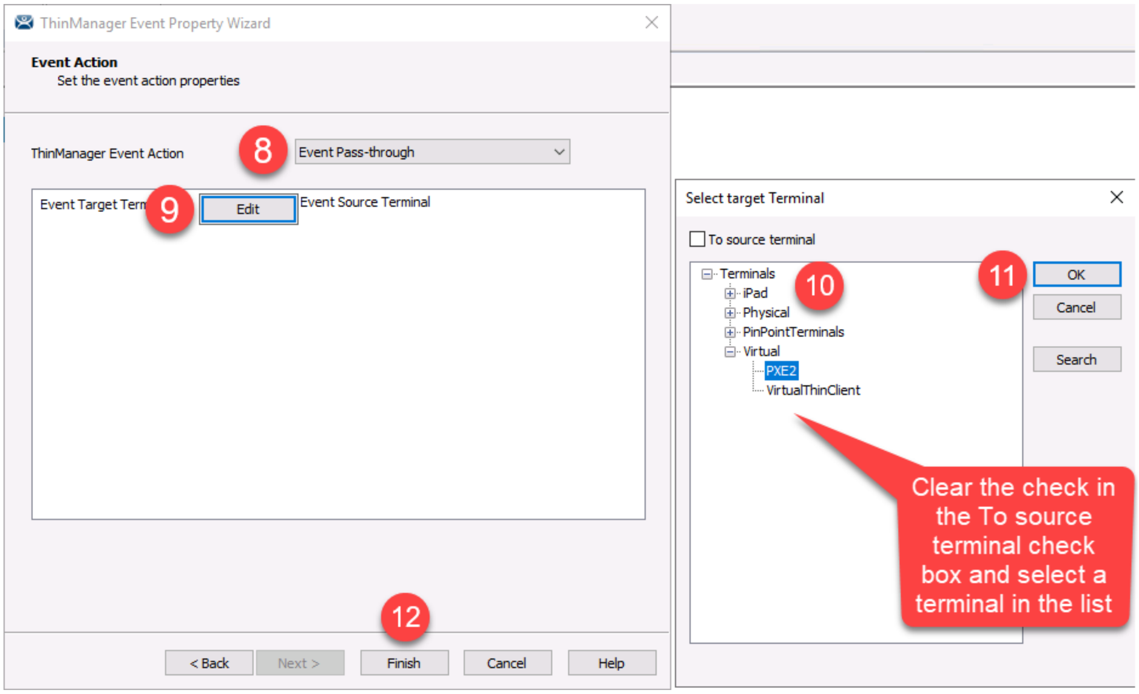
4 Property Name: PinPoint_Terminal01

Compare Operation: =

Compare Value Type: String

5 Compare Value: 1

Logical Operation: AND



ThinManager Event Action: Event Pass-through

Event Target Terminal: Event Source Terminal

□ To source terminal

Clear the check in the To source terminal check box and select a terminal in the list

Set the VBA code on the trigger application side to match the newly created Event:

```
'Send the following PinPoint Event to be processed by the
ThinManager server
MatchExpression = "PinPoint_Terminal01"
Tag = "<Tag name>"
Scope = "<Tag scope>"
Path = "Project=<File path to your .acd file>"
Dim TermMonResult As Integer

'Pass the arguments from above and trigger a new ThinManager
Logix PinPoint Event
TermMonResult = TermMonControl1.PinPoint(3, Tag, Scope,
MatchExpression & vbNewLine & Path)
```

Note: see the *Add VBA Code to a FactoryTalk View SE Display* section of this document for more details about the PinPoint method syntax in VBA code.

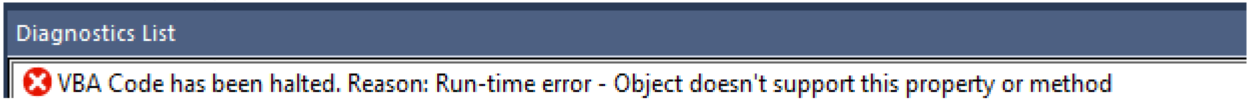
The resulting VBA code on one terminal will now trigger a PinPoint Event to be executed on a remote terminal. You may have multiple ThinManager Logix PinPoint Events if the Event Name and Event Match Expression is unique for each Event.

The Display Client for ThinManager Logix PinPoint can also be delivered to a mobile thin client. You can simply configure the Display Client to run on a different thin client within the ThinManager terminal tree.

Troubleshooting

This section lists common issues with ThinManager Logix PinPoint and how to solve them.

VBA Error: Object doesn't support this property or method



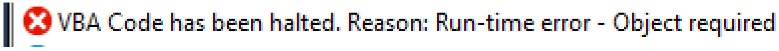
Diagnostics List

VBA Code has been halted. Reason: Run-time error - Object doesn't support this property or method

You need to regenerate VBE EXD files for FactoryTalk View SE. Follow the procedure in this technote: [TermMon ActiveX for ThinManager not Connecting to Thin Client (custhelp.com)]()

Note: Make sure to have the FactoryTalk View Studio Enterprise installer handy, as you may have to do a repair installation of FactoryTalk View Studio Enterprise after following the procedure above to regenerate necessary VBA support files.

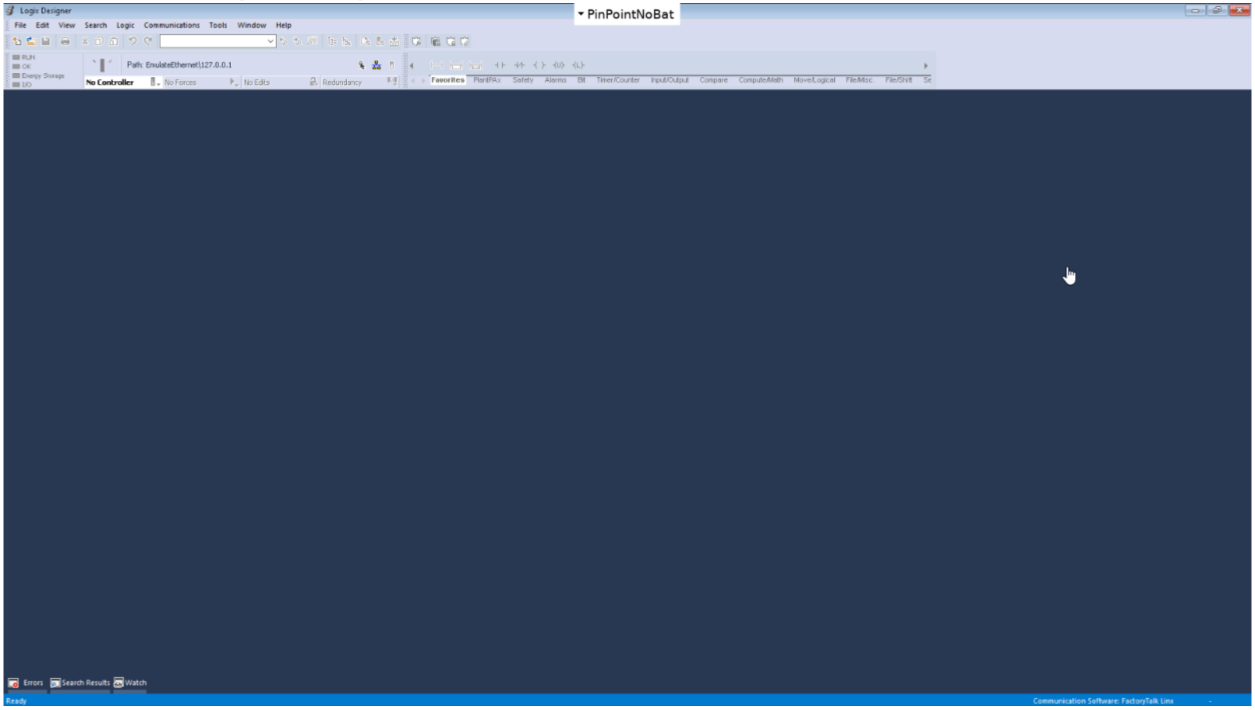
VBA Error: Object Required



VBA Code has been halted. Reason: Run-time error - Object required

Make sure that the TermMon ActiveX control is added to your Display and the Expose to VBA is used. See the *Add VBA Code and Objects to FactoryTalk View SE Display* section of this document.

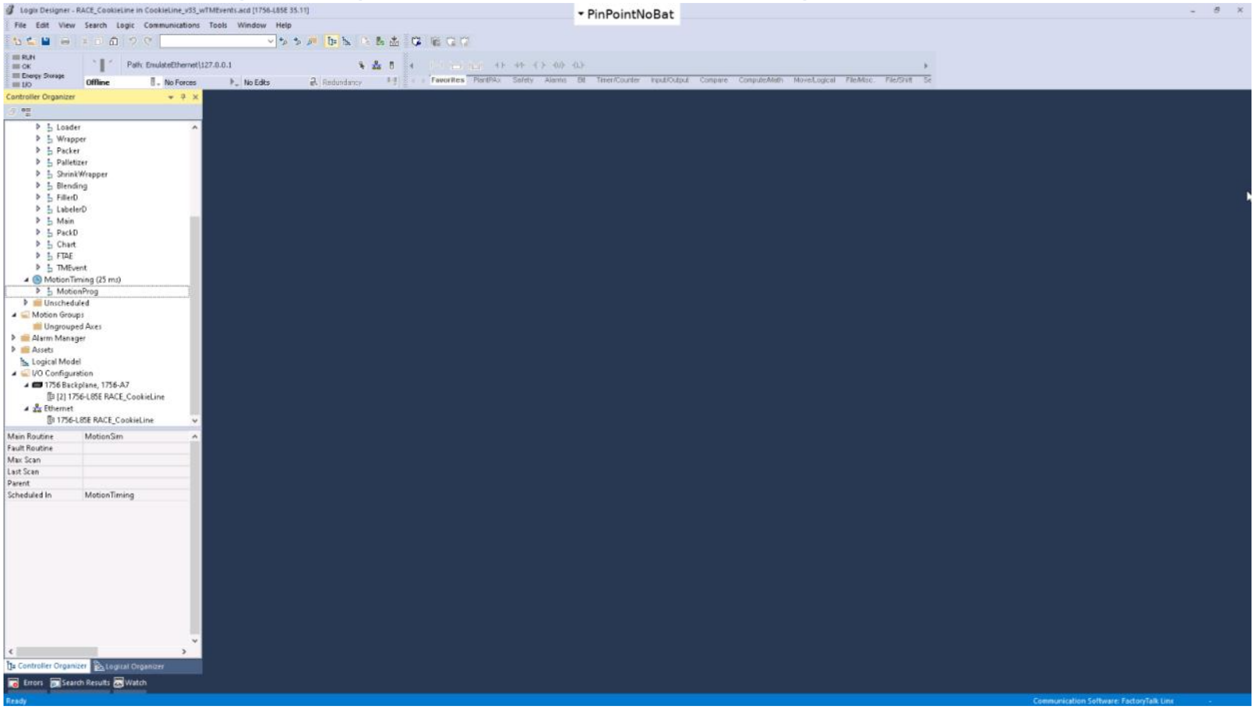
Studio 5000 Logix Designer Does Not Open .ACD file



Studio 5000 Logix Designer may not open the .acd file when it is launched but sits on an empty screen. They may happen after you have configured it to open a .acd file in the settings, through VBA, or a command-line parameter. If this is the case, check the following:

1. You have set the proper permissions to the file. See the *File Permissions* section of this document.
2. You have the correct path specified to the file in your settings, command-line parameter, or VBA code.

Studio 5000 Logix Designer Does Not Browse to Rung


Logix Designer - RACE_CookieLine in CookieLine_v33_wTMEvents.acd [1756-L85E 35.11]

Path: EmulateEthernet\127.0.0.1

Offline | No Forces | No Edits | Redundancy

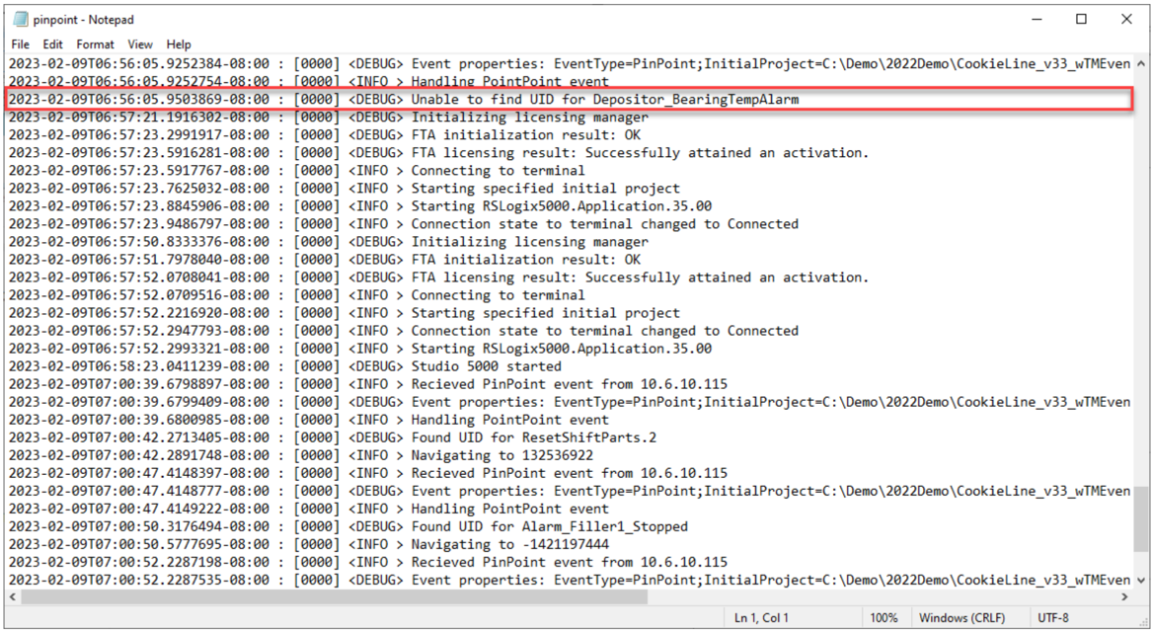
Main Routine	MotionSim
Fault Routine	
Max Scan	
Last Scan	
Parent	
Scheduled In	MotionTiming

Ready

Communication Software: FactoryTalk Linx

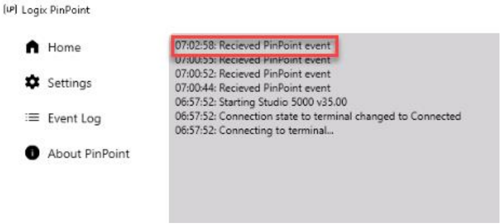
You think you have set up all the components of ThinManager Logix PinPoint, but you don't see anything happen when you trigger an Event with code or REST API.

1. Make sure that you don't have another copy of the .acd file open on the same machine, as only one .acd file can be used per session of Studio 5000 Logix Designer.
2. Make sure that you are browsing to a valid tag.
3. Check the error logs. Enable logging by setting the value for `PinPointLib` in the settings file to 1. Then relaunch PinPoint and check the generated log file. See the *ThinManager Logix PinPoint UI & JSON File* section for more details about the settings file and the log file. Here is an example of a log file with errors:



```
2023-02-09T06:56:05.9252384-08:00 : [0000] <DEBUG> Event properties: EventType=PinPoint;InitialProject=C:\Demo\2022Demo\CookieLine_v33_wTMEven
2023-02-09T06:56:05.9252754-08:00 : [0000] <INFO > Handling PointPoint event
2023-02-09T06:56:05.9503869-08:00 : [0000] <DEBUG> Unable to find UID for Depositor_BearingTempAlarm
2023-02-09T06:57:21.1916302-08:00 : [0000] <DEBUG> Initializing licensing manager
2023-02-09T06:57:23.2991917-08:00 : [0000] <DEBUG> FTA initialization result: OK
2023-02-09T06:57:23.5916281-08:00 : [0000] <DEBUG> FTA licensing result: Successfully attained an activation.
2023-02-09T06:57:23.5917767-08:00 : [0000] <INFO > Connecting to terminal
2023-02-09T06:57:23.7625032-08:00 : [0000] <INFO > Starting specified initial project
2023-02-09T06:57:23.8845906-08:00 : [0000] <INFO > Starting RSLogix5000.Application.35.00
2023-02-09T06:57:23.9486797-08:00 : [0000] <INFO > Connection state to terminal changed to Connected
2023-02-09T06:57:50.8333376-08:00 : [0000] <DEBUG> Initializing licensing manager
2023-02-09T06:57:51.7978040-08:00 : [0000] <DEBUG> FTA initialization result: OK
2023-02-09T06:57:52.0708041-08:00 : [0000] <DEBUG> FTA licensing result: Successfully attained an activation.
2023-02-09T06:57:52.0709516-08:00 : [0000] <INFO > Connecting to terminal
2023-02-09T06:57:52.2216920-08:00 : [0000] <INFO > Starting specified initial project
2023-02-09T06:57:52.2947793-08:00 : [0000] <INFO > Connection state to terminal changed to Connected
2023-02-09T06:57:52.2993321-08:00 : [0000] <INFO > Starting RSLogix5000.Application.35.00
2023-02-09T06:58:23.0411239-08:00 : [0000] <DEBUG> Studio 5000 started
2023-02-09T07:00:39.6798897-08:00 : [0000] <INFO > Recieved PinPoint event from 10.6.10.115
2023-02-09T07:00:39.6799409-08:00 : [0000] <DEBUG> Event properties: EventType=PinPoint;InitialProject=C:\Demo\2022Demo\CookieLine_v33_wTMEven
2023-02-09T07:00:39.6800985-08:00 : [0000] <INFO > Handling PointPoint event
2023-02-09T07:00:42.2713405-08:00 : [0000] <DEBUG> Found UID for ResetShiftParts.2
2023-02-09T07:00:42.2891748-08:00 : [0000] <INFO > Navigating to 132536922
2023-02-09T07:00:47.4148397-08:00 : [0000] <INFO > Recieved PinPoint event from 10.6.10.115
2023-02-09T07:00:47.4148777-08:00 : [0000] <DEBUG> Event properties: EventType=PinPoint;InitialProject=C:\Demo\2022Demo\CookieLine_v33_wTMEven
2023-02-09T07:00:47.4149222-08:00 : [0000] <INFO > Handling PointPoint event
2023-02-09T07:00:50.3176494-08:00 : [0000] <DEBUG> Found UID for Alarm_Filler1_Stopped
2023-02-09T07:00:50.5777695-08:00 : [0000] <INFO > Navigating to -1421197444
2023-02-09T07:00:52.2287198-08:00 : [0000] <INFO > Recieved PinPoint event from 10.6.10.115
2023-02-09T07:00:52.2287535-08:00 : [0000] <DEBUG> Event properties: EventType=PinPoint;InitialProject=C:\Demo\2022Demo\CookieLine_v33_wTMEven
```

4. You may be referencing the same alarm and tag. The tab will be switched, or a new tab will open in Studio 5000 Logix Designer if the tag you are referencing is in a different routine than the one that is currently selected and open.
5. Make sure that ThinManager Logix PinPoint is receiving an Event. You can check the log file for ThinManager Logix PinPoint or look at the Event Log portion of the ThinManager Logix PinPoint GUI.



```
07:02:58: Recieved PinPoint event
07:00:55: Recieved PinPoint event
07:00:52: Recieved PinPoint event
07:00:44: Recieved PinPoint event
06:57:52: Starting Studio 5000 v35.00
06:57:52: Connection state to terminal changed to Connected
06:57:52: Connecting to terminal...
```

If PinPoint is not receiving any Events, try restarting the ThinServer service on the ThinManager server. Also, check to make sure that your code is valid. Debug your code and make sure that the PinPoint method returns a result of 0, meaning success. For example, in the code below, the variable `TermMonResult` should return an integer value of 0.

```
Dim TermMonResult As Integer
TermMonResult = TermMonControl1.PinPoint(2, Tag, Scope, Path)
```

Studio 5000 Logix Designer Does Not Browse to Structured Text

Structured text navigation with ThinManager Logix PinPoint requires Logix Designer v33.01.00 or newer. Check that the version of Logix Designer used supports navigation to structured text.

Additional Resources

ThinManager + FactoryTalk ViewSE Deployment Guide	https://literature.rockwellautomation.com/idc/groups/literature/documents/at/tm-at001_-en-e.pdf
ThinManager Product Profile	https://literature.rockwellautomation.com/idc/groups/literature/documents/pp/tm-pp001_-en-p.pdf
ThinManager User Guide	https://literature.rockwellautomation.com/idc/groups/literature/documents/um/tm-um001_-en-p.pdf
Overview of Studio 5000 Service Edition	About RSLogix 5000 and Studio 5000 Service Edition (custhelp.com)

Get Support from Rockwell Automation

Technical Support Center	Support | Rockwell Automation
Local Technical Support Phone Numbers	Phone/Onsite Support (custhelp.com)
Product Compatibility and Download Center (PCDC)	Product Compatibility & Download Center from Rockwell Automation

Updated 07/14/2023

ThinManager – A Rockwell Automation Technology
1220 Old Alpharetta Rd - Suite 390
Alpharetta, GA 30005

1-877-239-4282
sales@thinmanager.com

For more information, please visit **thinmanager.com**