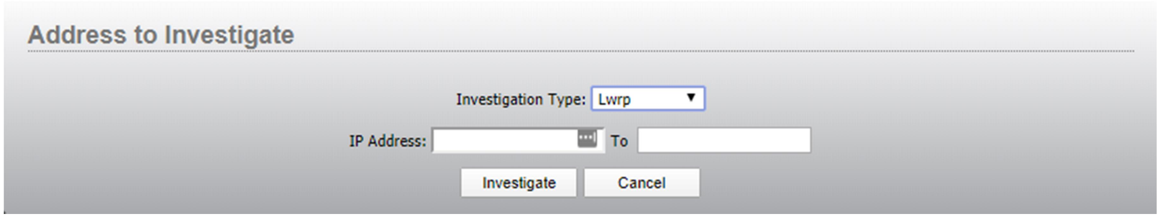
Version 1.7.0.01 Changes

RestApi Device

Please note: This is very much a beta feature. Please report any problems using this feature so that we can address them as soon as possible.

This version adds a Rest API device that can be added into the system in order to send rest api commands to a rest api enabled device. In order to add a rest api device, click on the devices link and click the plus icon to add a new device. This will display the Address to Investigate dialog.



Address to Investigate

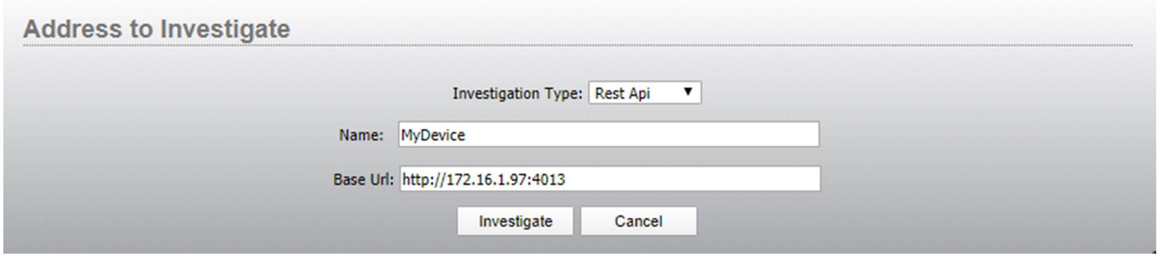
Investigation Type: Lwrp

IP Address: To

From the Investigation Type drop down, select Rest API. The dialog will change to show a name and a Url field.

Note: name field only available as of 1.7.0.02. In 1.7.0.01 a name is derived from the url.

Enter a name for the device and the http address to send rest api commands to. Alternate ports may be used by using a colon and the port number:



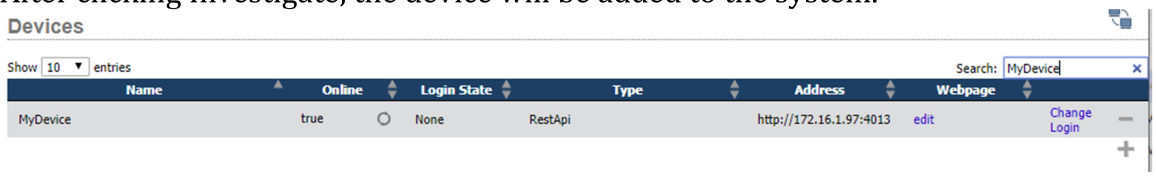
Address to Investigate

Investigation Type: Rest Api

Name: MyDevice

Base Url: http://172.16.1.97:4013

After clicking Investigate, the device will be added to the system.



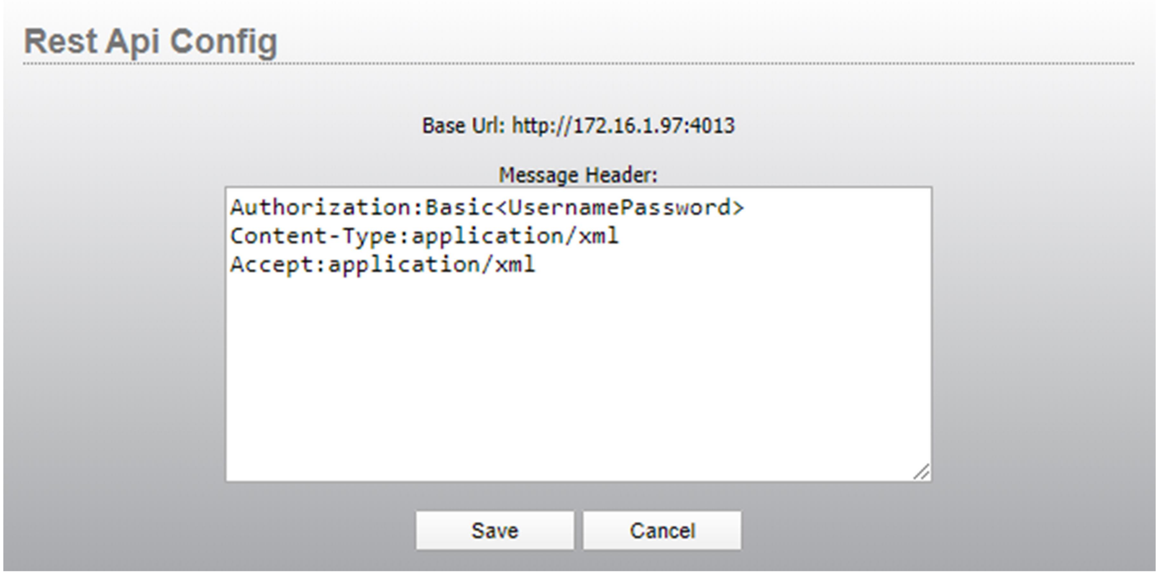
Devices

Show 10 entries

Search: MyDevice

Name	Online	Login State	Type	Address	Webpage	
MyDevice	true ○	None	RestApi	http://172.16.1.97:4013	edit	Change Login

You will notice that for rest api devices, instead of a web page link you will see an edit link. The edit link will bring up a dialog for customizing the message header used by each rest api command:



Rest Api Config

Base Url: http://172.16.1.97:4013

Message Header:

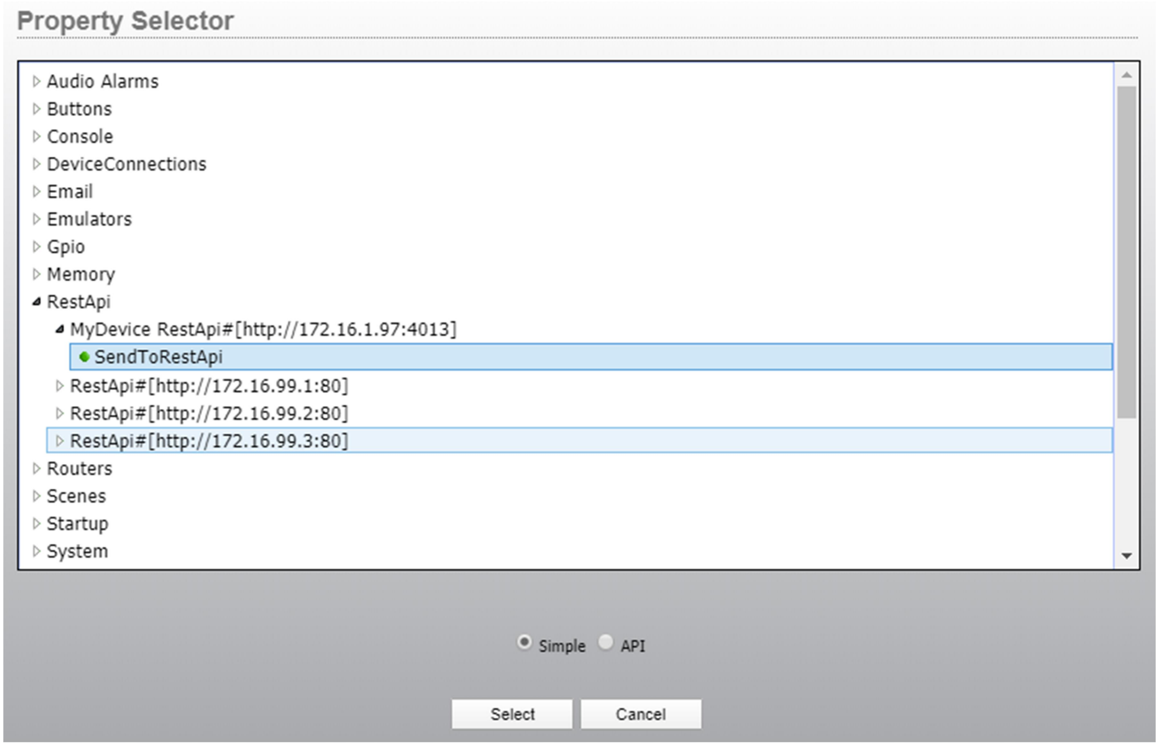
```
Authorization:Basic<UsernamePassword>
Content-Type:application/xml
Accept:application/xml
```

The <UsernamePassword> chunk will be replaced by a basic authentication conversion of the user name and password. If no user name and password is defined, the Authorization may be removed. Currently this device only supports basic authentication. You can use this dialog to add custom header messages if they are needed by the rest api device.

The ChangeLogin link will allow you to assign a username and password to be used by rest api commands in the Authorization portion of the header.

Sending a command to the Rest API device

Once the device has been added to the system, it will be available in logic flows.



Property Selector

- Audio Alarms
- Buttons
- Console
- DeviceConnections
- Email
- Emulators
- Gpio
- Memory
- RestApi
  - MyDevice RestApi#[http://172.16.1.97:4013]
    - SendToRestApi
  - RestApi#[http://172.16.99.1:80]
  - RestApi#[http://172.16.99.2:80]
  - RestApi#[http://172.16.99.3:80]
- Routers
- Scenes
- Startup
- System

● Simple ○ API

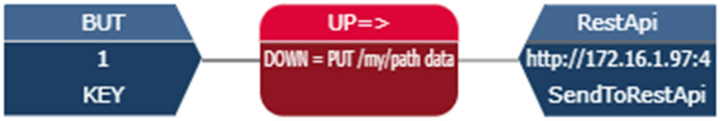
The SendToRestApi property needs to be structured in a specific way. The property value to send needs to have an operator (usually GET or PUT), a path, and optionally some data. Note that the path portion should be only the data after the device path. So for example:

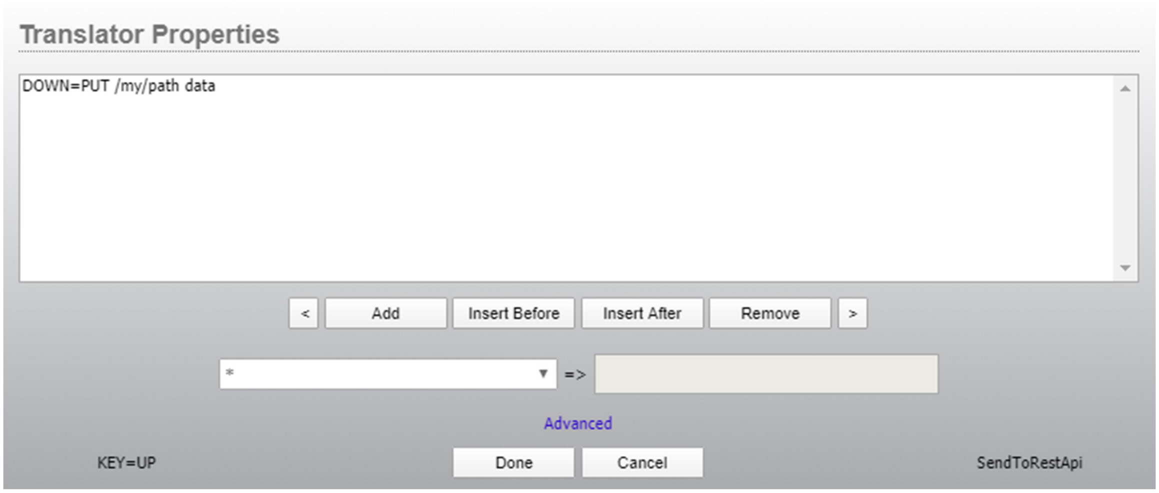
PUT /my/path/url hello

When the device sends data it will prepend the device path when it sends to the device so that the http command becomes:
PUT http://172.16.1.97:4013/my/path/url hello

The data portion is optional and depends on the specification of the api you are trying to communicate with.

For example, a logic flow that sends to a rest api device when a button is pushed might look like:





DOWN=PUT /my/path data

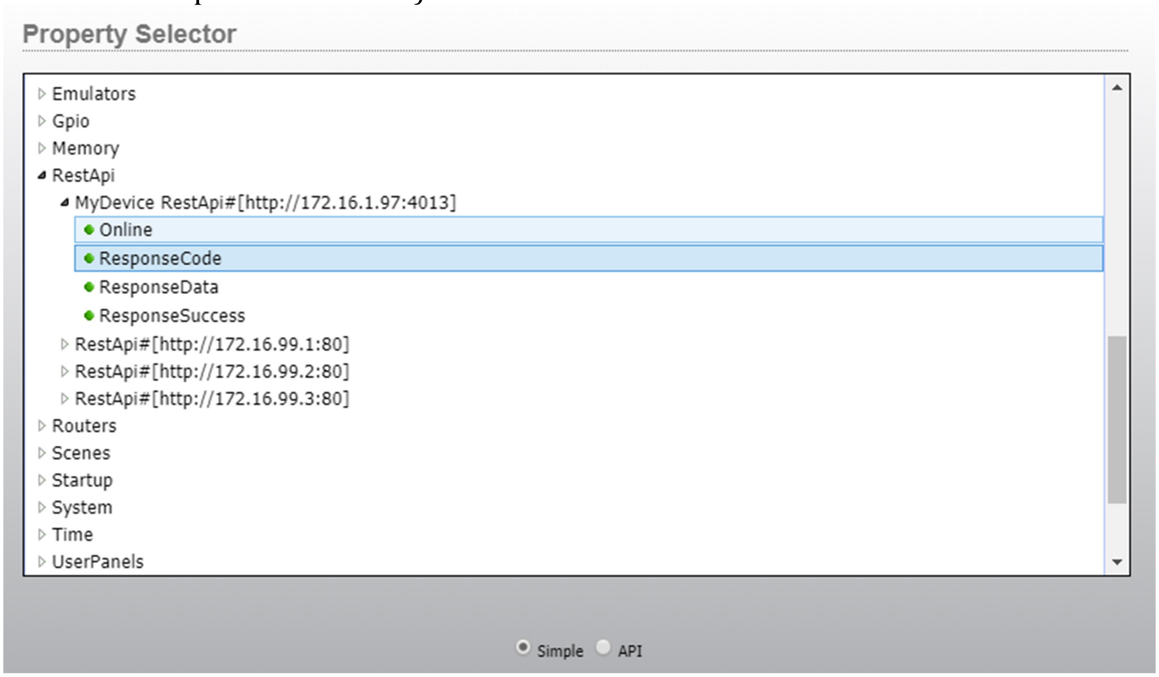
KEY=UP

SendToRestApi

Sap property memory slots and string builder memory slots can also be useful to build rest api commands according to the device's specification and then pass the command through to the device whenever the memory slot changes.

Receiving data from the Rest API device

Please review the rest api technical details for more information on the pitfalls of using rest api to capture information from a device. In this case you would use the example above to send a request to the device probably using the GET operator. The information that is returned can be found in three read only properties (only seen if a start point is selected):



Response code returns the http response code that results from the last message sent. For example, it might return 200 if the request succeeded. Or it might return

404 if the path was not found. For more information on http response codes, see: https://developer.mozilla.org/en-US/docs/Web/HTTP/Status.
The response data property will contain the actual data returned from the command. In many cases you may need a regex to filter the data for what you want to match in the logic flow.

The Response Success property will return true or false depending on whether the last rest api command was successful or not.

Rest API technical details

While rest APIs are currently very popular and therefore, we are adding them to PathfinderCore PRO out of necessity, we believe them to be generally ill suited for many broadcast applications. To provide an example we recently did some work with a product whose control protocol was exclusively Rest API. One of the first things our customer wanted to do was have PathfinderCore Pro do an action whenever a user pressed a button. The problem is that in order to detect that button push we would have needed to poll at 2.5 times the minimum amount of time the button might be held down. For computers this is not difficult until you start considering 25 buttons to a device and deployments of 600 to 700 devices. Now you have a situation where you are spending massive amounts of compute cycles on messages which basically say - nothing has changed. The point of that story is to say that applications that need low latency responses to state changes do not scale well with non-event driven protocols like Rest API. The consumer of such protocols needs to be cognizant of the cpu load ramifications for what they are doing. Currently if the user wants to poll, they would need to use an interval timer and a logic flow. We might add a way to pre program parameters that should be polled which would simplify that, but we opted not to do that in this incarnation because of the potential abuse possibilities described above. If all you care about is triggering a change and perhaps getting a response back that the change was made, then rest apis work well. But if real time state change information is required then care needs to be taken and other options if available are advised. This article on Rest versus web sockets gives a great analogy using a general and his army of the difference between REST and a socket based architecture.

https://www.pubnub.com/blog/websockets-vs-rest-api-understanding-the-difference/

Which is to say use this feature with care.

TimeStamp Memory Slots

CalculateTime property

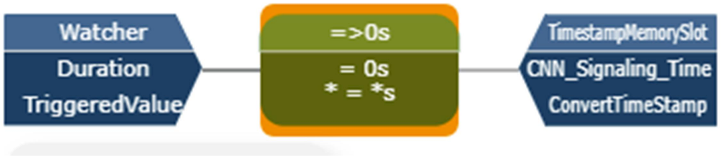
This version adds a calculate time write only property to time stamp memory slots. This property can be used to convert values from one time format to the pattern specified in the time stamp memory slot. For example, you could assign a number with either m, h, or s to the end and it would convert the value to the requested format in minutes hours or seconds. If the time stamp memory slot pattern was:

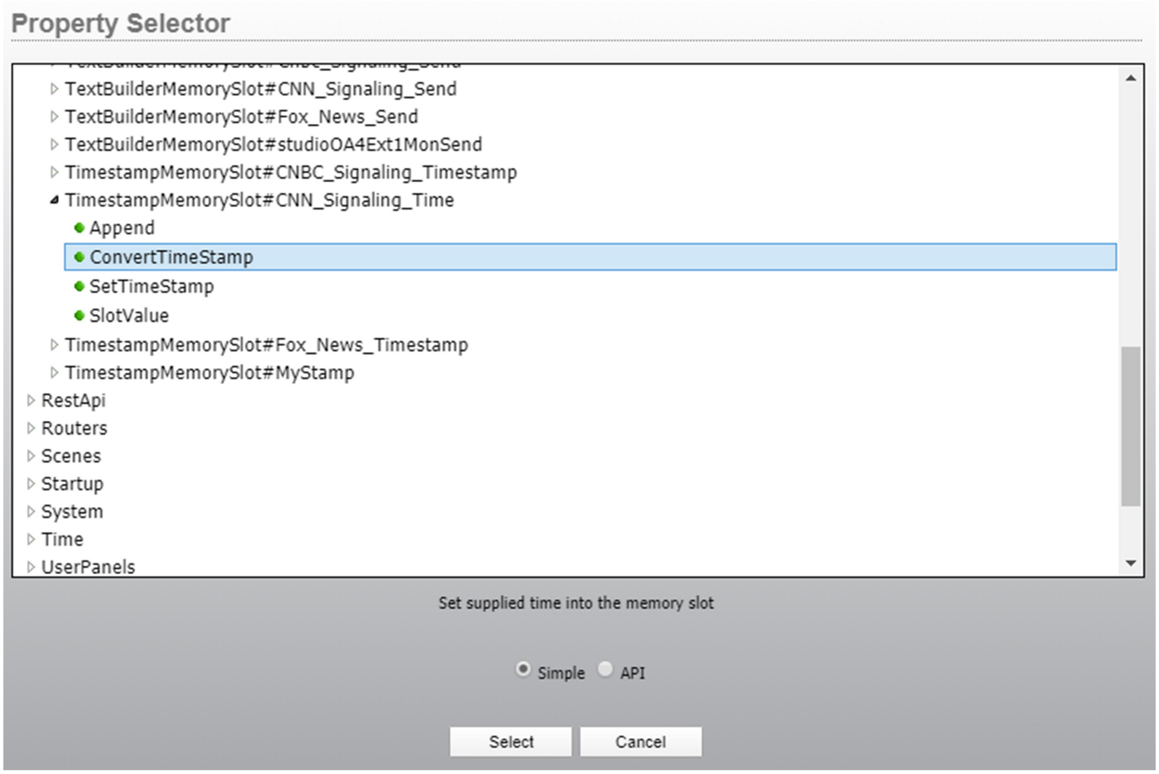
hh:mm:ss

and you passed in

100s

The value of the memory slot would become: 00:01:40.



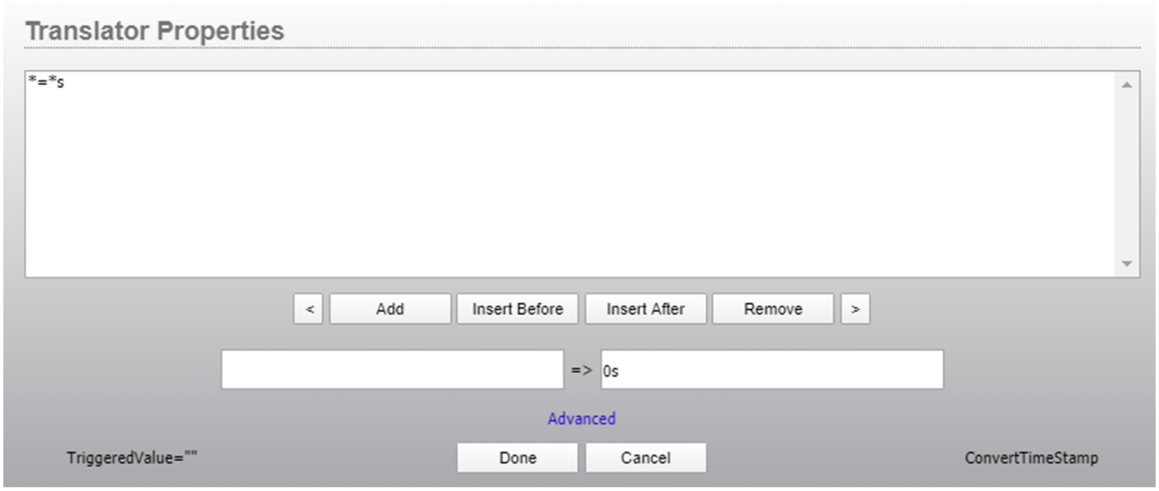


Property Selector

- TextBuilderMemorySlot#CNBC_Signaling_Send
- TextBuilderMemorySlot#CNN_Signaling_Send
- TextBuilderMemorySlot#Fox_News_Send
- TextBuilderMemorySlot#studioOA4Ext1MonSend
- TimestampMemorySlot#CNBC_Signaling_Timestamp
- TimestampMemorySlot#CNN_Signaling_Time
  - Append
  - ConvertTimeStamp
  - SetTimeStamp
  - SlotValue
- TimestampMemorySlot#Fox_News_Timestamp
- TimestampMemorySlot#MyStamp
- RestApi
- Routers
- Scenes
- Startup
- System
- Time
- UserPanels

Set supplied time into the memory slot

● Simple ○ API



```
*=*s
```

=> 0s

TriggeredValue=""

ConvertTimeStamp

Passing other time values than just numbers will try to interpret the value as time, convert it, and then express it in the correct pattern.

Version 1.7.0.02 Changes

RestApi Device

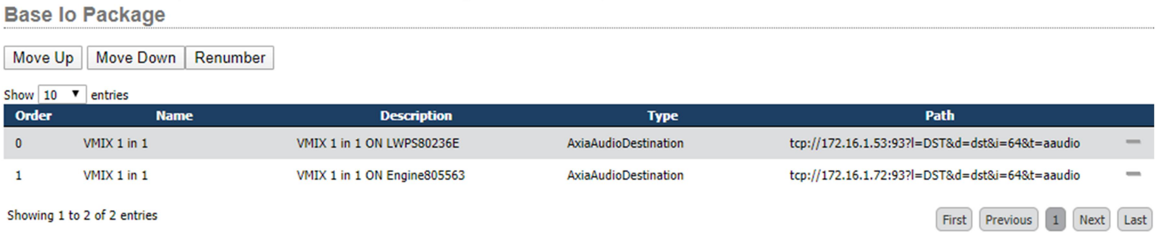
Added a name field to the investigation dialog in order to apply a more user-friendly name. This is described in more detail above in version 1.7.0.01 notes.

Virtual Router

Enabled property for Virtual Source and Destination base Ios

Each virtual source and destination's base points in a virtual router now also include an enable property which could be true or false. This can be used to disable the base point from being considered during route changes and route state analysis. This property is only available via the API tree in logic flows as it is considered an advanced property. Here is an example of where this feature might be useful. Let's say we are mixing audio for an airchain using a vmixer. We might create a virtual router with the virtual mixer destinations and the possible studio sources. However, what if for redundancy sake we wished to duplicate the settings on a vimixer on a second engine thereby allowing us to feed the air chain off of either engine and take one or the other out of service. To accomplish this, we could create a virtual router with destination base points using vmixer channels from both engines and where the source points are the same source base point duplicated.

Destination (note the different engine paths):

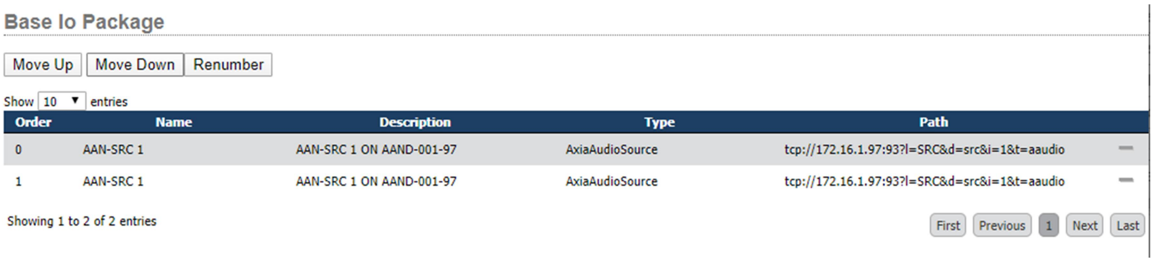


Base Io Package

Order	Name	Description	Type	Path
0	VMIX 1 in 1	VMIX 1 in 1 ON LWPS80236E	AxiaAudioDestination	tcp://172.16.1.53:93?l=DST&d=dst&i=64&t=aaudio
1	VMIX 1 in 1	VMIX 1 in 1 ON Engine805563	AxiaAudioDestination	tcp://172.16.1.72:93?l=DST&d=dst&i=64&t=aaudio

Showing 1 to 2 of 2 entries

Source:



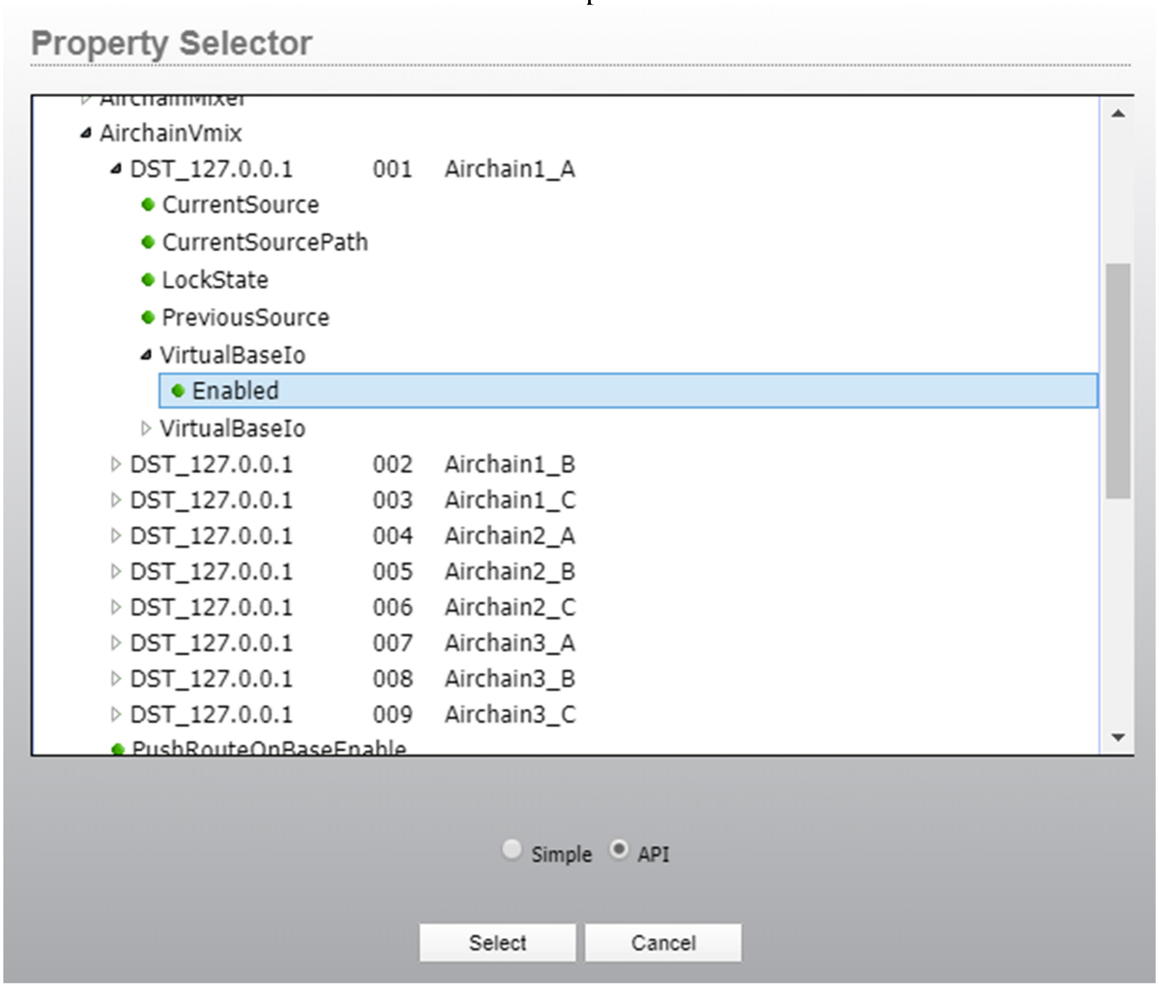
Base Io Package

Order	Name	Description	Type	Path
0	AAN-SRC 1	AAN-SRC 1 ON AAND-001-97	AxiaAudioSource	tcp://172.16.1.97:93?l=SRC&d=src&i=1&t=aaudio
1	AAN-SRC 1	AAN-SRC 1 ON AAND-001-97	AxiaAudioSource	tcp://172.16.1.97:93?l=SRC&d=src&i=1&t=aaudio

Showing 1 to 2 of 2 entries

If this virtual source was routed to the virtual destination, the same base source would get routed to the vmix destination on both engines. And the route would only show active if it was successfully made on both engines. However, if we wanted to take engine number 2 offline for maintenance, our route changes would still be sent to both engines, but the state would not be displayed since the offline engine did not show the successful route.

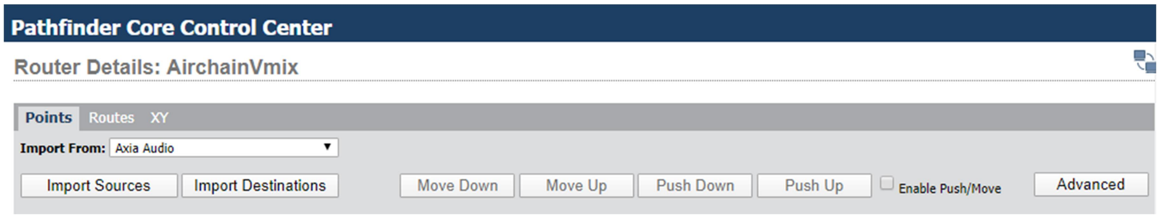
Using the enable property of the base points, we could create a scene or a logic flow that disabled the destination base points for the second engine while it is in maintenance. The virtual router would continue to work as if those base points did not exist in the destination. Then we could execute a second scene when the engine comes back online to re-enable those base points.


Property Selector
AirchainMixer
AirchainVmix
DST_127.0.0.1 001 Airchain1_A
CurrentSource
CurrentSourcePath
LockState
PreviousSource
VirtualBaseIo
Enabled
VirtualBaseIo
DST_127.0.0.1 002 Airchain1_B
DST_127.0.0.1 003 Airchain1_C
DST_127.0.0.1 004 Airchain2_A
DST_127.0.0.1 005 Airchain2_B
DST_127.0.0.1 006 Airchain2_C
DST_127.0.0.1 007 Airchain3_A
DST_127.0.0.1 008 Airchain3_B
DST_127.0.0.1 009 Airchain3_C
PushRouteOnBaseEnable
Select
Cancel

○ Simple ● API

Advanced: Push Route on Base Io Enable

This feature works with the base io enable/disable feature above. When viewing a virtual router on the points tab, there is now an advanced button on the far right hand side:

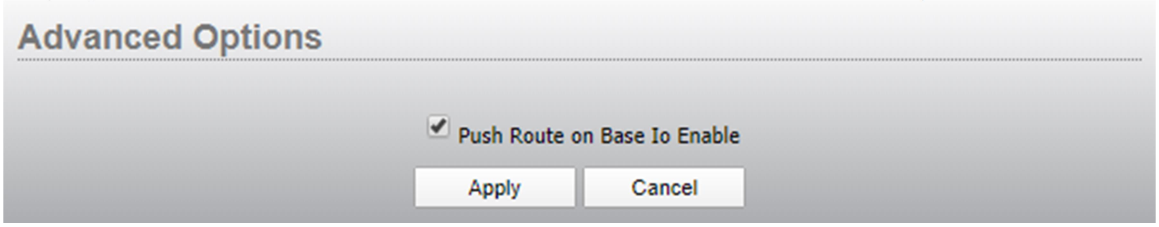

Pathfinder Core Control Center

Router Details: AirchainVmix

Import From: Axia Audio

□ Enable Push/Move

Clicking the advanced button will open a dialog for advanced options. Currently the only option is a checkbox for Push Route on Base Io Enable which by default is off.


Advanced Options

☑ Push Route on Base Io Enable

This option defines what happens when a virtual source or destination's base point was set to enabled=false and changes to enabled=true. If the option is checked, then the existing route state will be re-applied in order to update the state of the base point that was just re-enabled with the current source. If this option is not enabled, no change will be made when the base point is enabled meaning that the route state may change to something like Other since the newly enabled base point does not fit the current source to destination pairing.

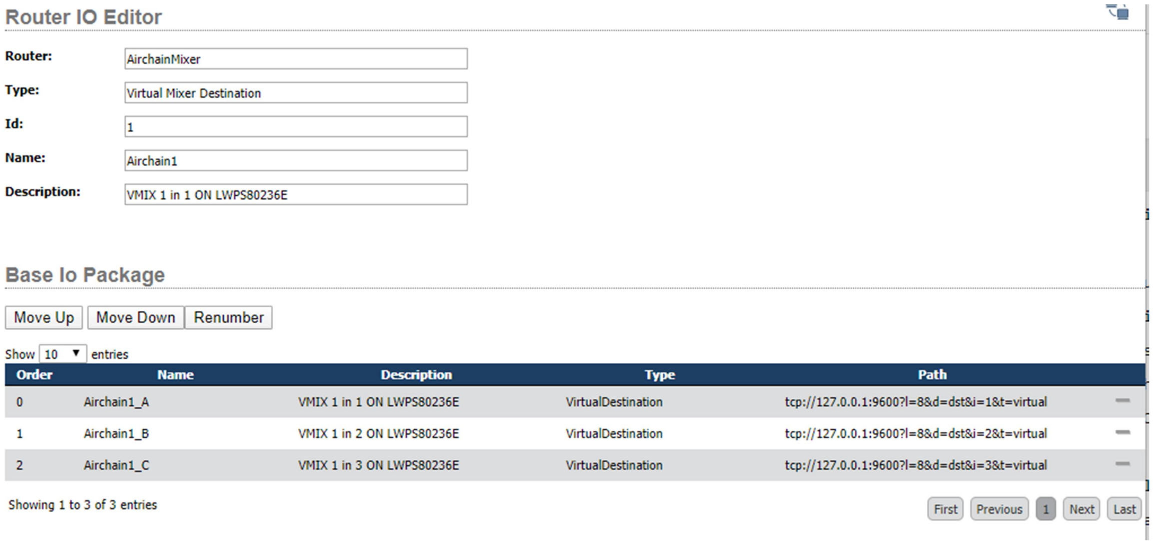
Virtual Mixing Router

Please note: This is very much a beta feature. Please report any problems using this feature so that we can address them as soon as possible.

This version adds a new type of virtual router called a virtual mixing router. To create one, browse to the routers page and click the plus icon to add the router. From the Router Type, select Virtual Mixing Router, provide a name and description, and click Add.

The virtual mixing router is created and populated in the same way as a virtual router but it behaves differently. A virtual mixing router only allows a single base point for each source and assumes that the base points of a destination mix together. Therefore, you can apply multiple sources to the same destination up to the number of base points you have defined for the destination. Let's see if we can make this clearer through an example. Let's say you have 10 studios and 4 airchains. However, some of these studios are news rooms which may need to be mixed in with the main studio. In this case you can use a vmixer on an engine to mix these together. And we could use the Virtual Mixing Router to dynamically route studios into the vmixer. To do this we would create the virtual mixing router like this:

Destination:



Router IO Editor

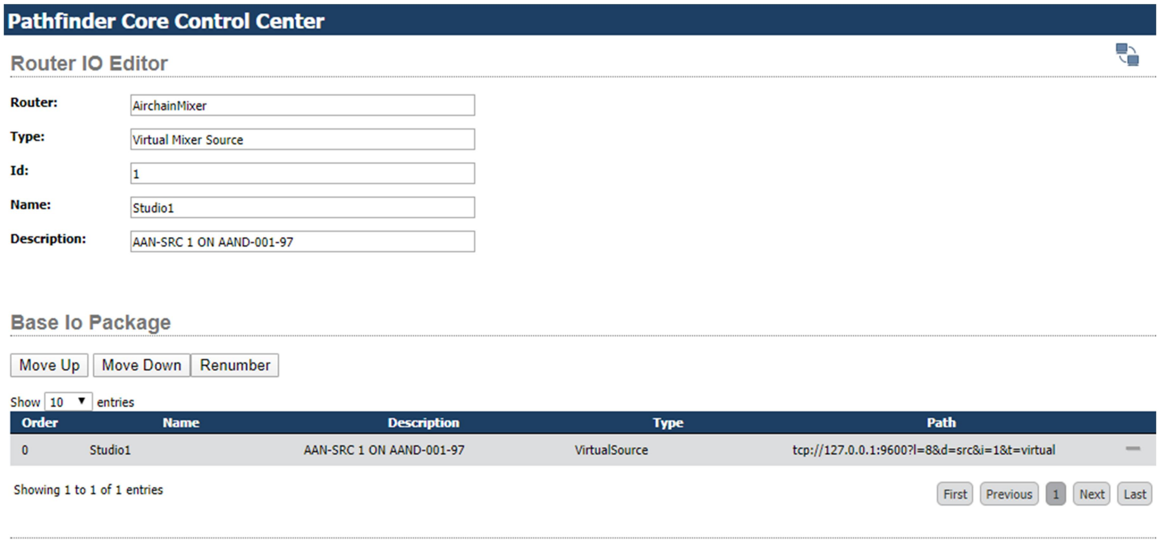
Router:	AirchainMixer
Type:	Virtual Mixer Destination
Id:	1
Name:	Airchain1
Description:	VMIX 1 in 1 ON LWPS80236E

Base Io Package

Order	Name	Description	Type	Path
0	Airchain1_A	VMIX 1 in 1 ON LWPS80236E	VirtualDestination	tcp://127.0.0.1:9600?l=8&d=dst&i=1&t=virtual
1	Airchain1_B	VMIX 1 in 2 ON LWPS80236E	VirtualDestination	tcp://127.0.0.1:9600?l=8&d=dst&i=2&t=virtual
2	Airchain1_C	VMIX 1 in 3 ON LWPS80236E	VirtualDestination	tcp://127.0.0.1:9600?l=8&d=dst&i=3&t=virtual

Showing 1 to 3 of 3 entries

Source:



Pathfinder Core Control Center

Router IO Editor

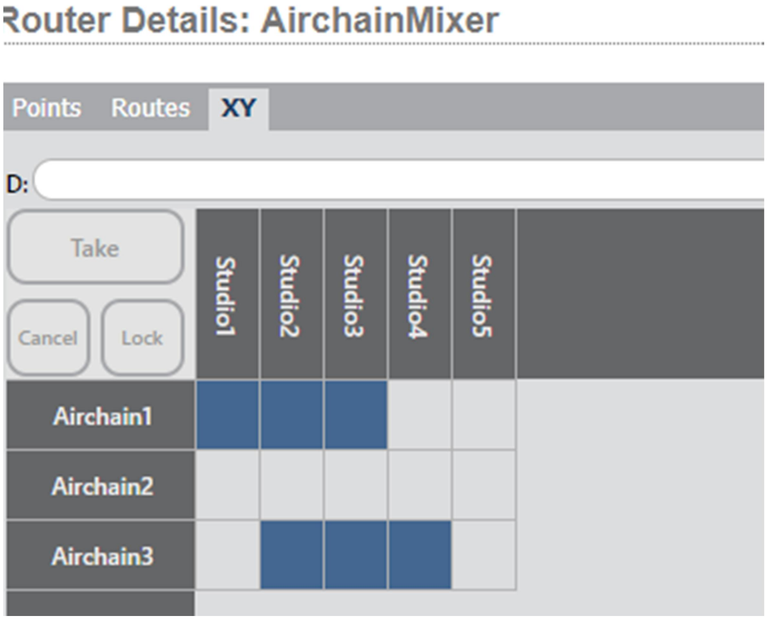
Router:	AirchainMixer
Type:	Virtual Mixer Source
Id:	1
Name:	Studio1
Description:	AAN-SRC 1 ON AAND-001-97

Base Io Package

Order	Name	Description	Type	Path
0	Studio1	AAN-SRC 1 ON AAND-001-97	VirtualSource	tcp://127.0.0.1:9600?l=8&d=src&i=1&t=virtual

Showing 1 to 1 of 1 entries

If we added additional sources and destinations where the destinations were comprised of multiple mixed (vmix) destinations and the source was a single source, we end up with a router that can look like this:



	Studio1	Studio2	Studio3	Studio4	Studio5
Airchain1	■	■	■		
Airchain2					
Airchain3		■	■	■	

You will notice that the destination AirChain1 actually has three sources assigned to it and since under the hood the destination's base points are in the same vmixer, these are mixed together. It is important to note that PathfinderCore PRO does not actually do any audio mixing. No audio passes through PathfinderCore PRO. Rather you are telling PathfinderCore PRO that these destinations are all a part of a mixer.

Routing a source to a destination in a Virtual Mixing Router is accomplished differently than in a normal router. In a normal router we assign a source to a destination using the CurrentSource or CurrentSourcePath property. With a Virtual Mixing Router we use a different set of properties called AddSource, RemoveSource, and SwapSource.

When you apply a source number to a virtual mixing destination's AddSource Property, the system will review the destination's base points. If the source is already assigned to one of the destination's base points then no change is made. If it is not assigned and all of the base points already have a source assigned to them, then no change is made. If it is not assigned and once of the destination's base points is cleared, then the source will be applied to that base point. Therefore, you can route as many sources to the destination as the destination has base points and they will be dynamically used.

Conversely, when a source number is applied to a destination's RemoveSource property, the system will check to see if that source is applied to any of the destination's base points, and if so it will clear that base point.

The SwapSource property checks to see if a source is applied to one of the destination's base points and if not, it attempts the same actions as AddSource. If it is applied, then it executes a remove source. This is useful for toggle buttons that add and remove sources from a destination.

While this router's sources can only have a single base point. It is possible to marry points together by putting the base points in a virtual router and then targeting the virtual router as the base points of a virtual mixing router.

Version 1.7.1.04 Changes

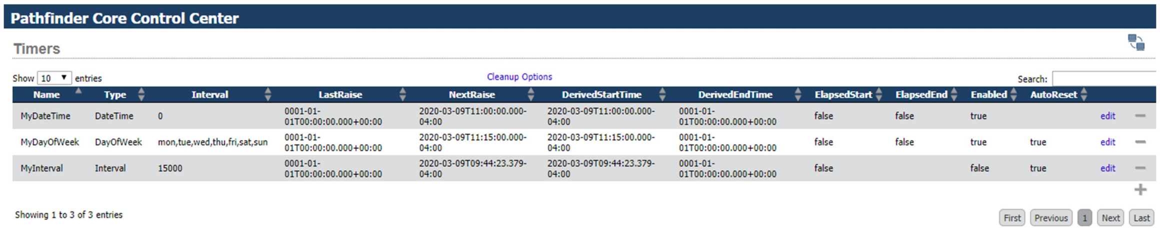
Timers

Note: The changes in this version of software are both ongoing and to be considered beta. It is recommended that you take a backup before moving to this version of software and monitor existing timer events closely after the initial update to make sure they are all working correctly. Report any issues to support if you encounter any.

This version adds several additional properties to DateTime and DayOfWeek timers. These timers can now represent a time range rather than just a specific time. As such there is now a start and end time property. Additionally, it is now possible to specify a property or route change to take place at the start and end times of the event without requiring a logic flow.

Timers List Changes

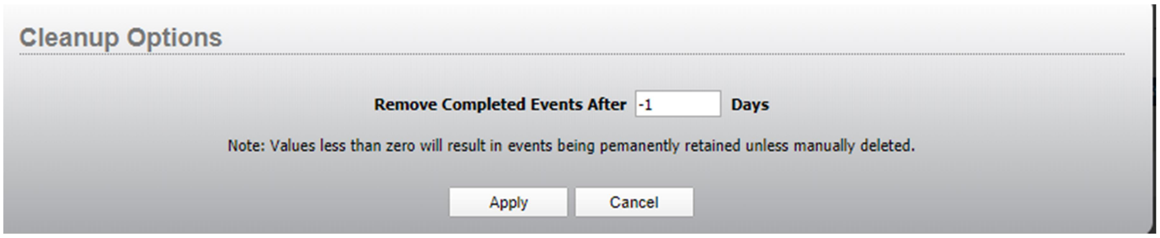
The timers list now contains a number of additional fields including DerivedStartTime, DerivedEndTime, and ElapsedEnd. It also has a Cleanup Options link which can be used to determine when expired timers will get deleted from the system as described below.



Name	Type	Interval	LastRaise	NextRaise	DerivedStartTime	DerivedEndTime	ElapsedStart	ElapsedEnd	Enabled	AutoReset		
MyDateTime	DateTime	0	0001-01-01T00:00:00.000+00:00	2020-03-09T11:00:00.000-04:00	2020-03-09T11:00:00.000-04:00	0001-01-01T00:00:00.000+00:00	false	false	true		edit	−
MyDayOfWeek	DayOfWeek	mon,tue,wed,thu,fri,sat,sun	0001-01-01T00:00:00.000+00:00	2020-03-09T11:15:00.000-04:00	2020-03-09T11:15:00.000-04:00	0001-01-01T00:00:00.000+00:00	false	false	true	true	edit	−
MyInterval	Interval	15000	0001-01-01T00:00:00.000+00:00	2020-03-09T09:44:23.379-04:00	2020-03-09T09:44:23.379-04:00	0001-01-01T00:00:00.000+00:00	false		false	true	edit	−

Since DateTime and DayOfWeek timers now have both a start and an end time, these are expressed in the list by the DerivedStartTime and DerivedEndTime columns. In the case of DateTime timers, these will carry the start and end times of the events. In the case of DayOfWeek timers these times will move to the next start and end time once a specific day's iteration of the event completes. The NextRaiseTime may be the next start time or the next end time depending on whether the event is still upcoming or is in the range between start and finish. The elapsed column is now titled ElapsedStart and represents whether the start of the event has elapsed whereas the ElapsedEnd represents whether the end of the event has elapsed.

Additionally, Core Pro timers now have a cleanup option accessible by clicking on the "Cleanup Option" link at the top of the timers list:



Cleanup Options

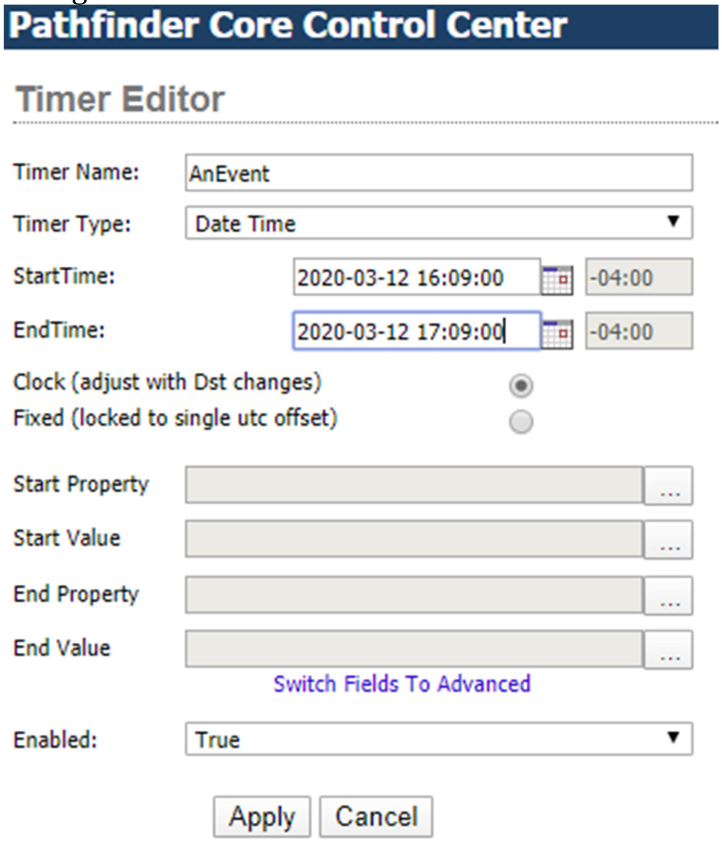
Remove Completed Events After -1 Days

Note: Values less than zero will result in events being pemanently retained unless manually deleted.

By default, the value will be -1 which means DateTime timers which have completed their execution will not be deleted from the system. Setting a number of days into this dialog and applying that number will cause single execution datetime events to be deleted from the system after the requested number of days has passed after the timer has completed executing. A value of 0 will cause the timer to be deleted sometime within the hour or two after the timer has completed execution.

DateTime Timer Changes

When creating or editing a date time, additional properties will be present in the configuration editor:



Pathfinder Core Control Center

Timer Editor

Timer Name: AnEvent

Timer Type: Date Time

StartTime: 2020-03-12 16:09:00 -04:00

EndTime: 2020-03-12 17:09:00 -04:00

Clock (adjust with Dst changes) ●

Fixed (locked to single utc offset) ○

Start Property

Start Value

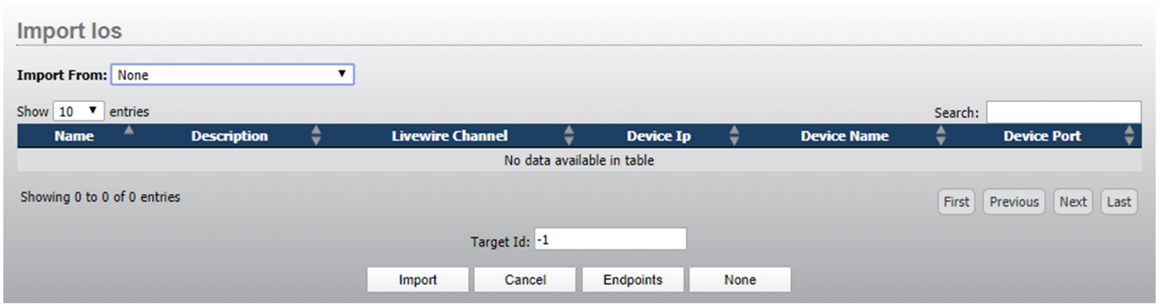
End Property

End Value

Enabled: True

There is now both a start and an end time so that a time range may be expressed. If the end time is not defined or is less than the start time, it will be ignored and will not execute. Within logic flows in addition to the elapsed time which will be set true when the start time executes, there is also an ElapsedEnd property which will be tripped to true when the end time of the time range is reached.

Additionally, 4 fields have been added for setting a route change or property change that should take place at the beginning and end of the event time range. If left blank, the event Elapsed, and ElapsedEnd properties can still be used in logic flows like previous versions of the software. However, you can add a change directly to the timer by clicking on the ellipsis button and selecting the correct parameters for what should happen at the start and/or end of the time range. The dialog that is presented closely follows the dialog for scene item selection. For example, clicking the Start Property ellipsis button will present the following dialog:



Import Ios

Import From: None

Show 10 entries

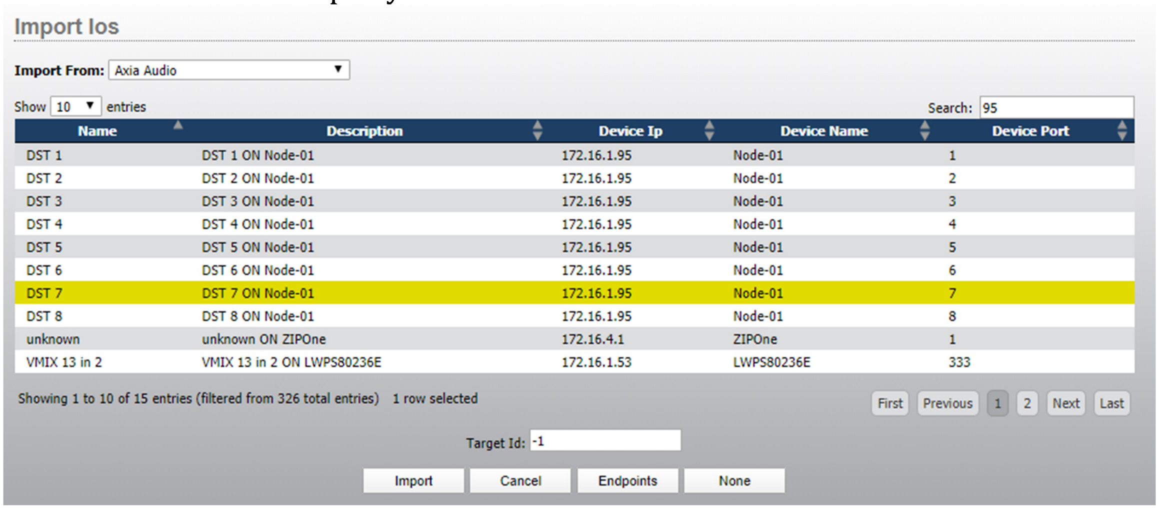
Search:

Name	Description	Livewire Channel	Device Ip	Device Name	Device Port
No data available in table					

Showing 0 to 0 of 0 entries

Target Id: -1

Selecting a router from the Import From list will present the router's destinations. Selecting a destination and clicking import will then make that destination be selected in the Start Property field.



Import Ios

Import From: Axia Audio

Show 10 entries

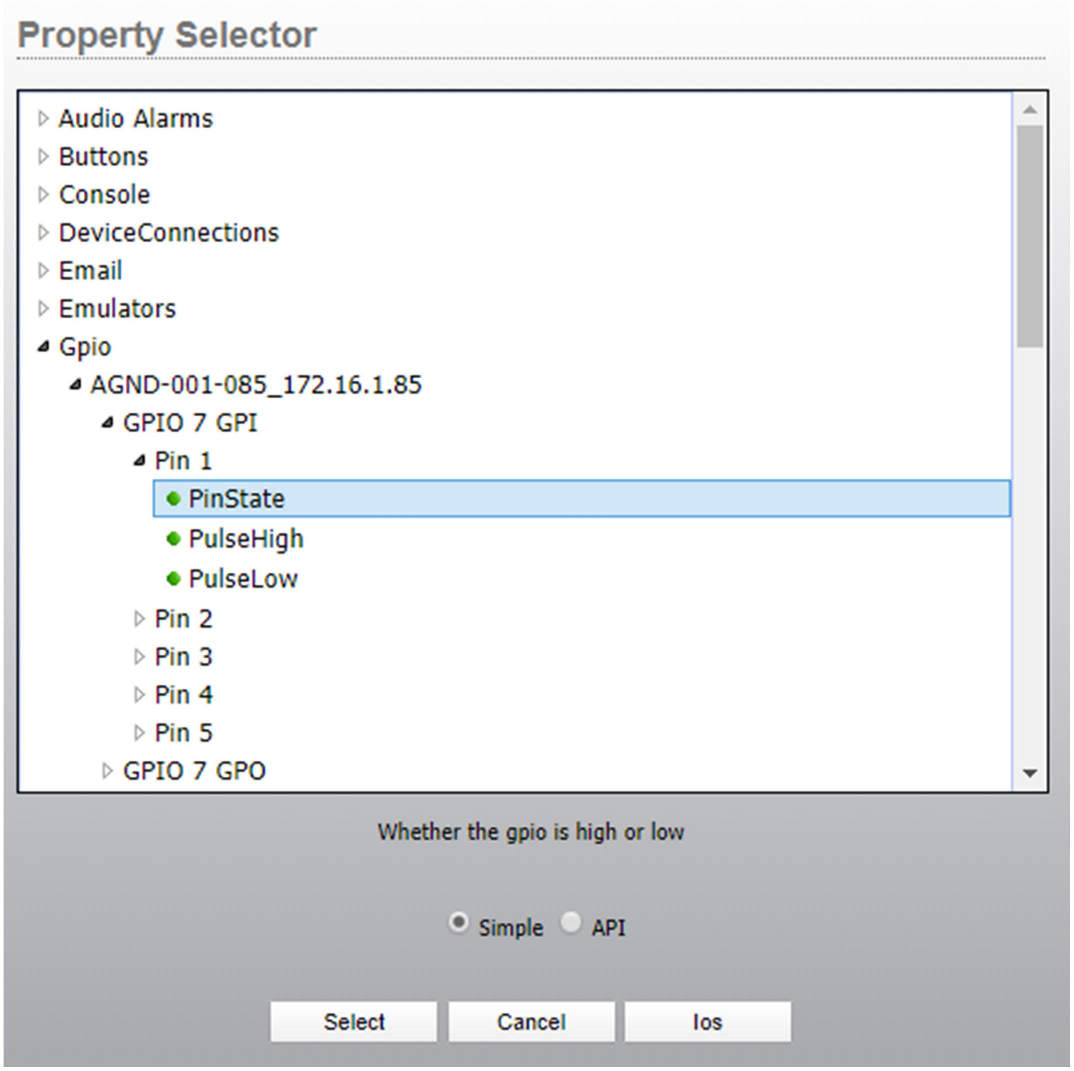
Search: 95

Name	Description	Device Ip	Device Name	Device Port
DST 1	DST 1 ON Node-01	172.16.1.95	Node-01	1
DST 2	DST 2 ON Node-01	172.16.1.95	Node-01	2
DST 3	DST 3 ON Node-01	172.16.1.95	Node-01	3
DST 4	DST 4 ON Node-01	172.16.1.95	Node-01	4
DST 5	DST 5 ON Node-01	172.16.1.95	Node-01	5
DST 6	DST 6 ON Node-01	172.16.1.95	Node-01	6
DST 7	DST 7 ON Node-01	172.16.1.95	Node-01	7
DST 8	DST 8 ON Node-01	172.16.1.95	Node-01	8
unknown	unknown ON ZIPOne	172.16.4.1	ZIPOne	1
VMIX 13 in 2	VMIX 13 in 2 ON LWPS80236E	172.16.1.53	LWPS80236E	333

Showing 1 to 10 of 15 entries (filtered from 326 total entries) 1 row selected

Target Id: -1

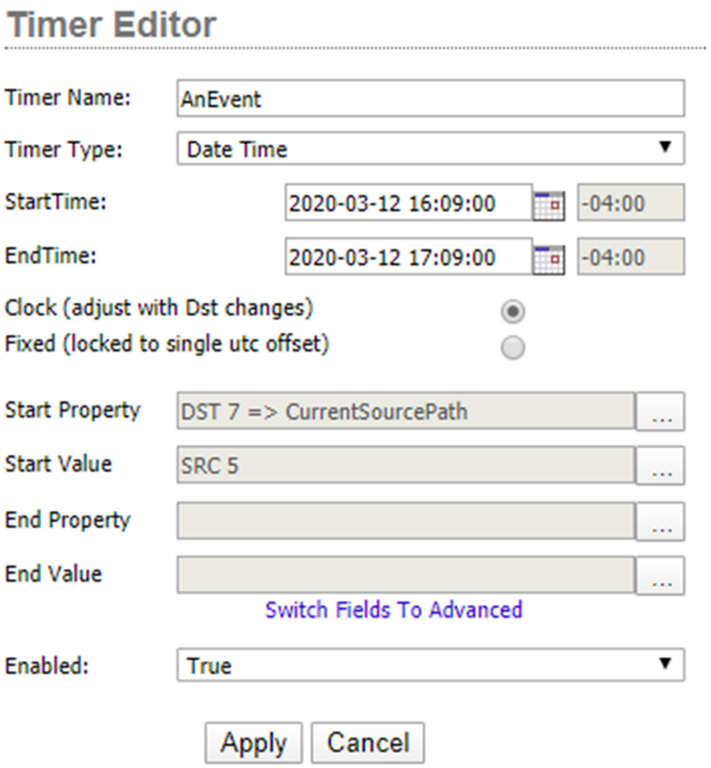
Selecting the endpoint button instead will bring up the standard logic flow end point dialog where any property in the system may be selected as the change point.



Whether the gpio is high or low

● Simple ○ API

The Ios button will return you to the io selection list. Selecting either a destination or an endpoint will fill in the start property field.



Timer Editor

Timer Name: AnEvent

Timer Type: Date Time

StartTime: 2020-03-12 16:09:00 -04:00

EndTime: 2020-03-12 17:09:00 -04:00

Clock (adjust with Dst changes) ●

Fixed (locked to single utc offset) ○

Start Property: DST 7 => CurrentSourcePath

Start Value: SRC 5

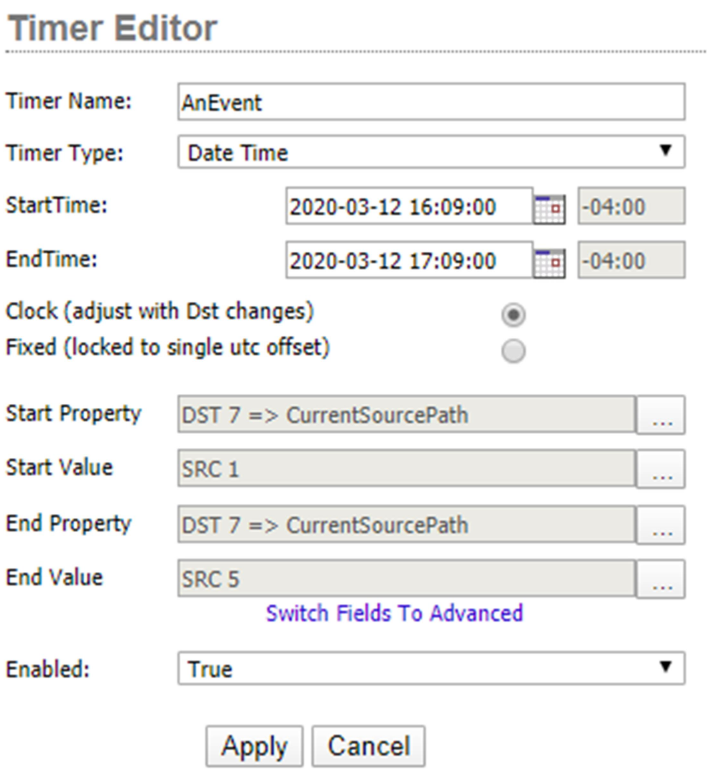
End Property:

End Value:

Enabled: True

Ordinarily the Start Value field will also get filled in with the current value of the Destination or property selected. Using the ellipsis button next to the start value field will allow you to select a different source or property value and the dialog will present different options accordingly.

This process can be repeated for the end property. For example, if you want to route a studio to an air chain at 4:09 PM and then back to an automation system at 5:09 PM on a specific day, the DateTime event might look like:



Timer Editor

Timer Name: AnEvent

Timer Type: Date Time

StartTime: 2020-03-12 16:09:00 -04:00

EndTime: 2020-03-12 17:09:00 -04:00

Clock (adjust with Dst changes) ●

Fixed (locked to single utc offset) ○

Start Property: DST 7 => CurrentSourcePath

Start Value: SRC 1

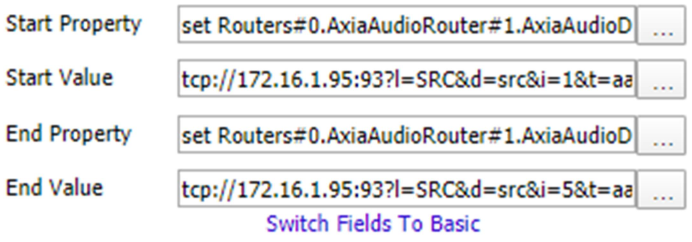
End Property: DST 7 => CurrentSourcePath

End Value: SRC 5

Enabled: True

Once stored the actions will be executed at the start and end times of the time range.

The “Switch Fields To Advanced” link will change the values presented in the property and value fields to their actual values rather than the friendly display and make them editable for advanced users:



Start Property: set Routers#0.AxiaAudioRouter#1.AxiaAudioD

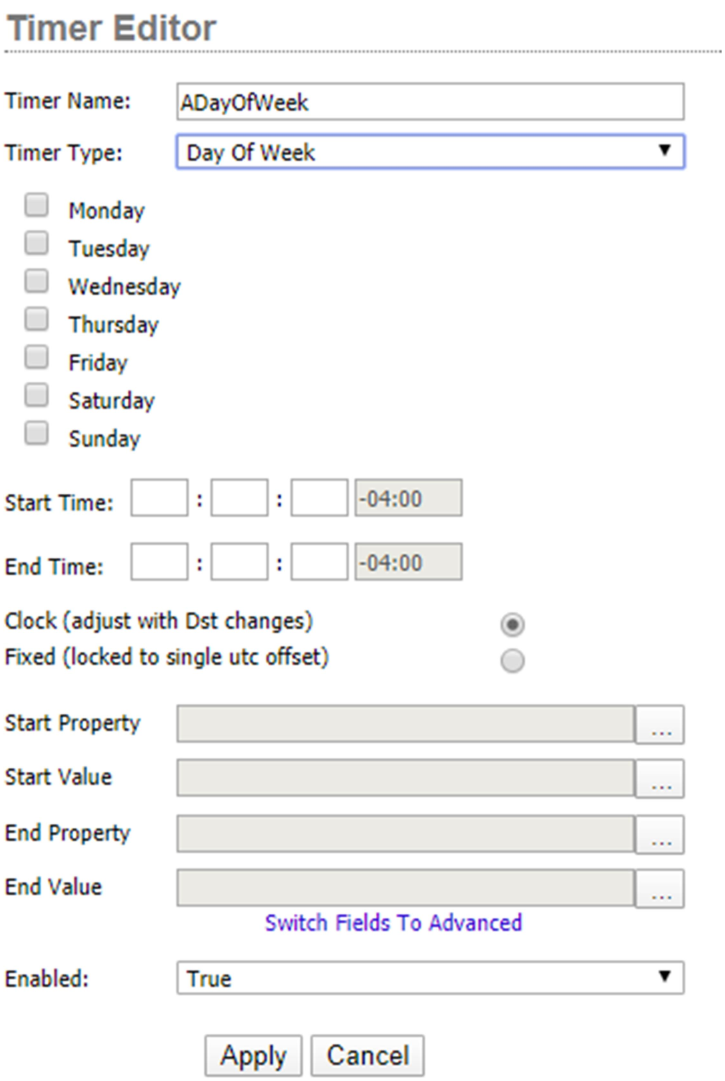
Start Value: tcp://172.16.1.95:93?l=SRC&d=src&i=1&t=aa

End Property: set Routers#0.AxiaAudioRouter#1.AxiaAudioD

End Value: tcp://172.16.1.95:93?l=SRC&d=src&i=5&t=aa

DayOfWeek Timer Changes

When creating a recurring timer (usually called a Day Of Week Timer), the changes to the editor are almost identical to what was described above for the one time date/time timer. An EndTime set of fields have been added and the same start and end property and value fields have been added to the DayOfWeek timer editor:



Timer Editor

Timer Name: ADayOfWeek

Timer Type: Day Of Week

□ Monday
□ Tuesday
□ Wednesday
□ Thursday
□ Friday
□ Saturday
□ Sunday

Start Time: : : -04:00

End Time: : : -04:00

Clock (adjust with Dst changes) ●
Fixed (locked to single utc offset) ○

Start Property

Start Value

End Property

End Value

Enabled: True

With a recurring timer, on the weekday that is selected for the timer to execute, the start time and end time values will be used to fill the derived start and end time properties for the day. At the correct start time on the day, the Elapsed property will be set to true and any configured start property change executed. And at the end time, the ElapasedEnd property will be set true, and the end property change will be executed if one is configured. The Elapsed and ElapsedEnd values will be reset to their false state sometime in the 2 hours leading up to midnight and/or early on the next day.

It is important to note that Elapsed and ElpasedEnd properties can also be used in logic flows for both DateTime and DayOfWeek events regardless of whether a change is configured in the event itself. However, changes configured in the event directly do not count against the logic licensing of the system.

UTC Offset Changes

Previous versions of the software would use the browser's UTC offset when creating a new clock based DateTime or DayOfWeek timer. This also caused odd display issues when editing events from a Core PRO in one timezone while using a browser in another. This version will always display the offset and specific time as it is set in Core PRO. Additionally, new events will be created using the offset defined in Core PRO by its current time zone rather than the browser's offset.

Version 1.7.4.06 Changes

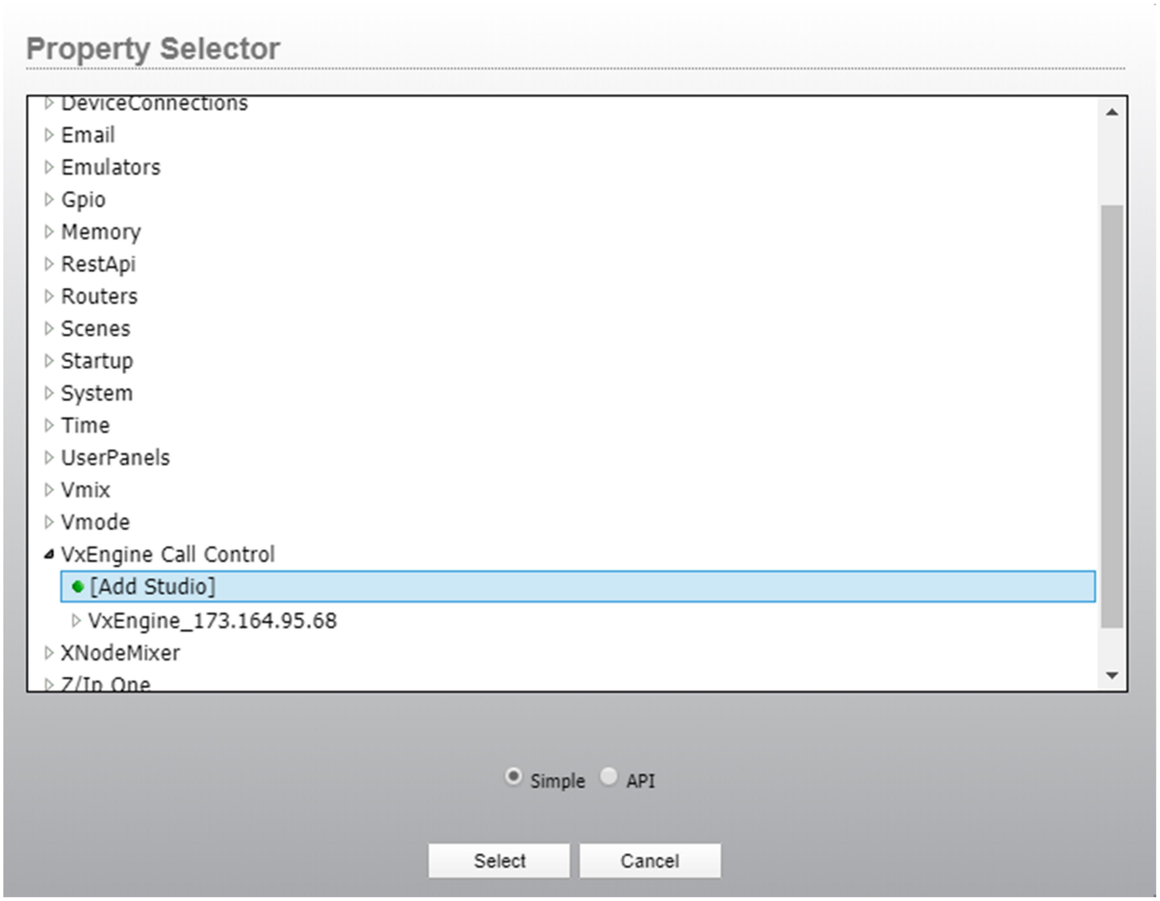
VX Control Protocol

Please note: This is very much a beta feature. Please report any problems using this feature so that we can address them as soon as possible.

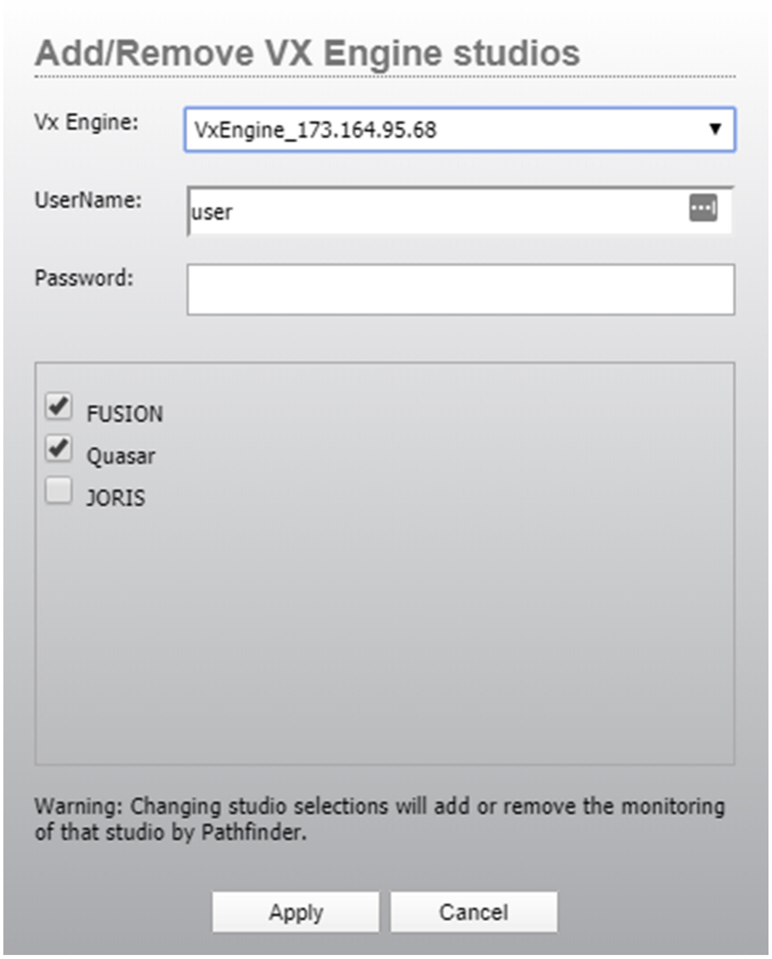
Version 1.7.4.06 adds the VX Control protocol into Pathfinder Core PRO. This protocol allows Pathfinder Core PRO logic flows and user panel controls to monitor and change Vx calls in a studio. Control points such as lines, studios, call states, comments, and call actions are now available to be used by Pathfinder Core PRO via this protocol.

It is important to understand that when you connect to a VX engine on the control port, you have to select which studio to interact with. As a result, if you want to monitor multiple VX engine studios in Pathfinder Core PRO at the same time, a separate TCP connection is required for each studio. Rather than enabling this by default when many customers may not need this functionality, we decided to allow the administrator to enable the studios they wish PathfinderCore PRO to interact with and leave the others disconnected. This conserves resources.

To begin using the VX Control protocol, go to the logic flows tab and select a logic flow view. Add a flow and double click on the end point. Within the tree you will see a new branch called VxEngine Call Control.



Clicking on the [Add Studio] branch will bring up a dialog where you can select which studios you want Pathfinder Core PRO to use and monitor.



Add/Remove VX Engine studios

Vx Engine: VxEngine_173.164.95.68

UserName: user

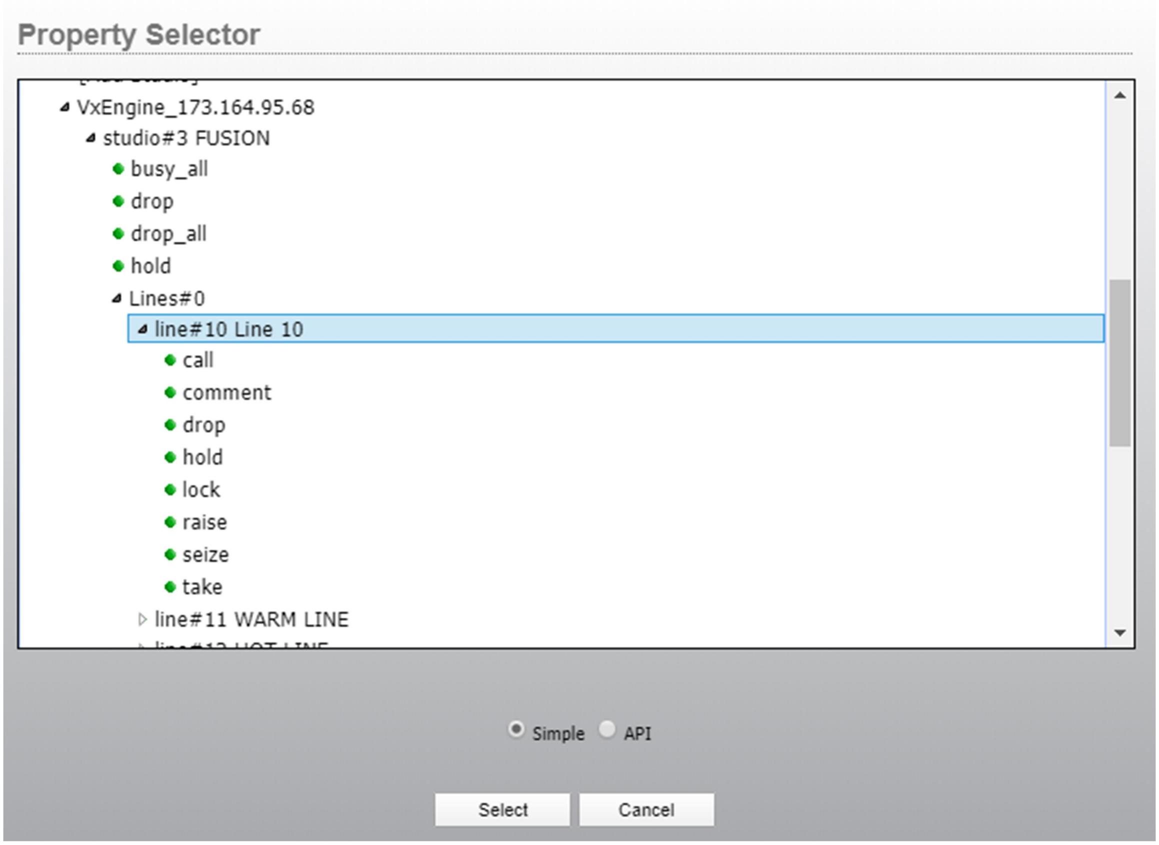
Password:

☑ FUSION
☑ Quasar
☐ JORIS

Warning: Changing studio selections will add or remove the monitoring of that studio by Pathfinder.

Select the VxEngine whose studios you wish to enable from the drop down and then enable the check boxes for the studios you wish to have available. You need to click apply for each engine whose studios you enable or disable. If you have changed the Lwcp password in the VxEngine, you need to supply the correct password in this dialog as well. If you change the password after adding the studios into PathfinderCore PRO, come back to this dialog, deselect the studios and apply and then reselect them using the new password and apply again. This is not a dialog that is specific to this instance of logic flow editing. You are adding or removing that studio support from the system and so it is generally a onetime change and does require some cpu resources while adding or removing the studio. We decided to place this dialog within logic flow because that will be where customers will most frequently be interacting with the VX Control protocol data points.

Once added, the studio and property items will then be available in logic flows under the specified engine:



Property Selector

- [Add Studio]
- VxEngine_173.164.95.68
  - studio#3 FUSION
    - busy_all
    - drop
    - drop_all
    - hold
    - Lines#0
      - line#10 Line 10
        - call
        - comment
        - drop
        - hold
        - lock
        - raise
        - seize
        - take
      - line#11 WARM LINE
      - line#12 HOT LINE

● Simple ○ API

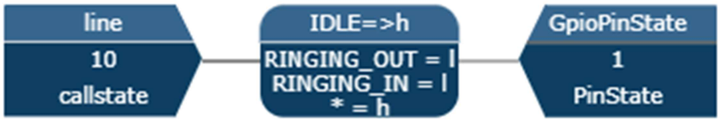
Select Cancel

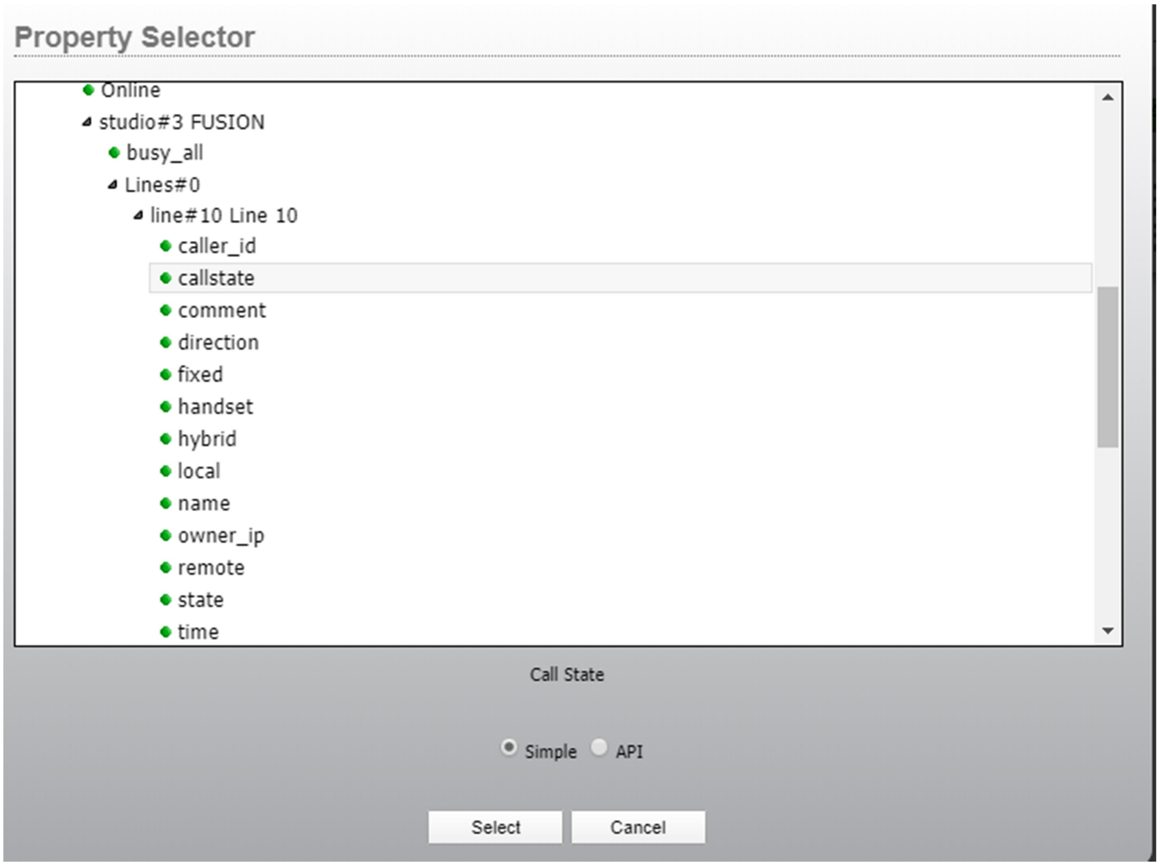
Warning: Currently if you add new studios to an engine, you need to collapse and re-expand the engine branch to see the new studio. Also if you uncheck (remove) a studio, you may need to refresh the logic flows web page before it is removed from the property tree. Removing (unchecking) a studio removes it from Pathfinder's control and monitoring entirely which may affect any other flows that use that studio.

Gpio Ring Example

As an example as to how this might be used, you could cause a gpio to light an indicator whenever a given line rings. To accomplish this, you would create a flow where the start point is the line's callstate property and the endpoint is a gpio.

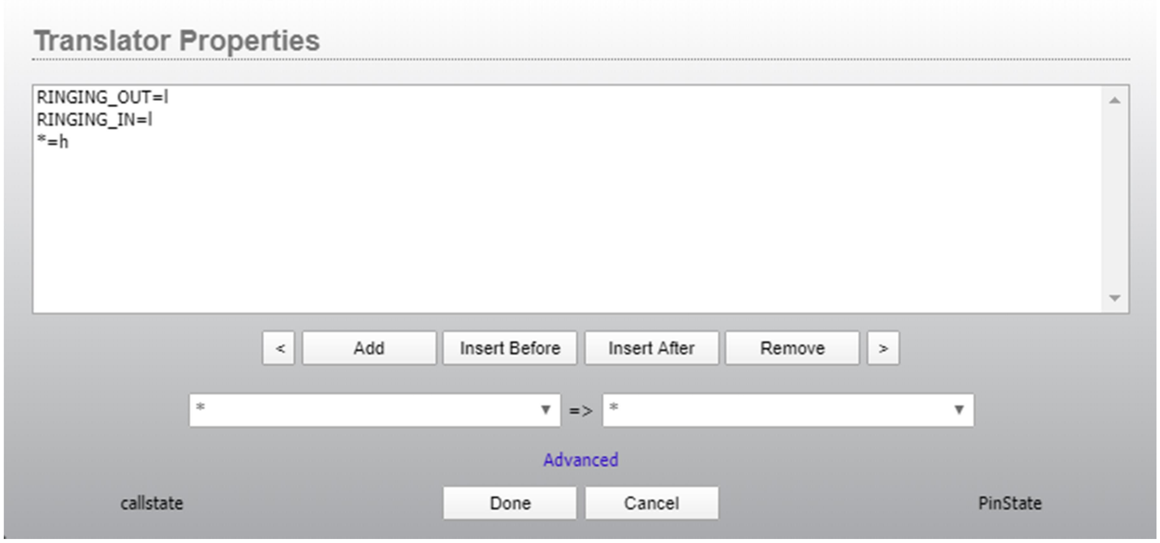
The flow would look like:





● Simple ○ API

For the translation you would create something like:



```
RINGING_OUT=l
RINGING_IN=l
*=h
```

* => *

callstate

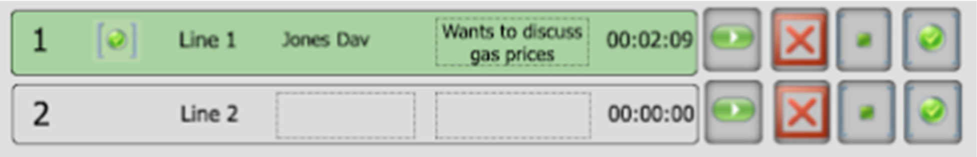
PinState

Additional flows could be added to trip other gpio pins for different lines.

It is important to note that the properties will differ significantly depending on whether you are selecting a start or end point. There are numerous properties that are “read only” or “write only” and so will only display in either the start or end point editing. Selecting a property in the logic flows tree will also display a brief description as to what the property does.

Panel Example

You can also use these properties in user panels. So for example, you could use a label object and bind the label's background color to the line call state and the textContent to the caller_id field. A button could then be used to answer and drop the calls. One of our support engineers crafted the following using the html 5 user panel designer during our initial testing.



1 Line 1 Jones Dav Wants to discuss gas prices 00:02:09

2 Line 2 00:00:00

This uses a large label where the call state is bound to the background color, and smaller transparent labels on top of that are bound to the caller_id, comment, and time properties. An image was also overlaid that swaps between multiple images to represent call state. Finally, additional buttons were added for the additional call actions such as seize, drop, hold, etc. Using these items you can develop a user interface to your liking that works with the vx phone system. This also allows you to create a user interface that mixes console fader control, call control, and other user interface items to achieve your specific needs.

Available Properties

Below find a list of the commonly used properties. Additional properties may be available that are not included in this list, so review the logic flow tree and property descriptions within the tree. Items in this table that are marked as true only in the syntax are write only actions where setting the property to true is a momentary state that triggers the action.

Object	Property	ReadWrite	Description	Syntax
Line				
	call	W	Call out using a specific hybrid.	<hybridnumber>_<numbertocall>
	caller_id	R	Caller Id	Text
	callstate	R	Call State	List of states
	comment	RW	Comment for the Line	Text
	direction	R	Direction for the Call	NONE INCOMING OUTGOING
	drop	W	Drop a line	True

	fixed	R	Is line fixed or selectable	True/False
	handset	R	Is call on the handset	True/False
	hold	W	Hold a line	ON_HOLD_READY ON_HOLD
	hybrid	R	Take the next call to the specified Hybrid	List of hybrid ids by name
	local	R	Local Phone Number	Text
	lock	W	Lock a line	True/False
	name	R	Name of the Line	Text
	owner_ip	R	IP address of the current line owner	Text
	raise	W	Raise a line in priority	True
	remote	R	Remote Phone Number	Text
	seize	W	Seize a line to reserve it for this client	True
	state	R	Line State	List of states
	take	W	Take the next call to the specified hybrid	List of hybrid ids by name
	time	R	Line Time	Time
Studio				
	busy_all	RW	Sets and reads studio busy_all state	True/False
	drop	W	Drop a specific hybrid	List of hybrid ids by name
	drop_all	W	Drop all calls	True
	hold	W	Hold the call that is currently on a specific hybrid	List of hybrid ids by name
	message	RW	Instant Message	Text

	mode	W	Set the studio mode for backwards compatibility	TALENT PRODUCER
	mute	RW	Studio Mute	True/False
	show_id	RW	Set or read Show Id	List of show ids by name
	take	W	Take a call	True
	next	R	Position of line that should be answered next	Number

Version 1.7.5.07 Changes

PathfinderCore PRO Scheduling Module

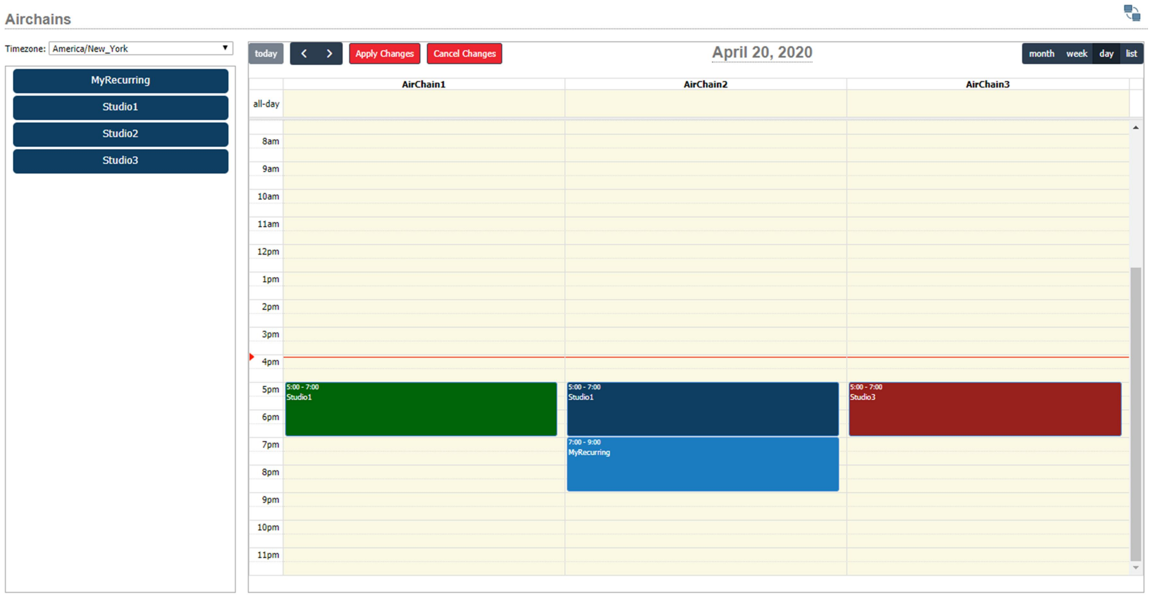
Please note: This is very much a beta feature.   Please report any problems using this feature so that we can address them as soon as possible. Also note that this feature requires an additional license as described below.

Scheduling License and Beta Testing

This feature requires a new scheduling license. During the beta period, prior to full sales being available of this license, beta testers may request a 90 day demo license for the scheduling module. Events created using this demo license will continue to function properly after the demo expires but without the ability to view them in the calendar. After demo expiration, the events will still be viewable and editable via the normal timers editor dialog. Because these timer events use a different object type, events created with the scheduling demo feature may not work if you downgrade the software to a previous version though. In addition to the demo license there is the ability with no license to open a demo calendar that has non-functioning data so that the concepts may be evaluated.

Calendar

This version of PathfinderCore PRO adds an optional calendar based scheduling module into PathfinderCore PRO. While scheduling of time based events has always been possible, the scheduling module simplifies this process with a drag and drop calendar user interface and the ability to dynamically combine properties and values during that process.


Airchains

Timezone: America/New_York

April 20, 2020

AirChain1: 5:00 - 7:00 Studio1

AirChain2: 5:00 - 7:00 Studio1

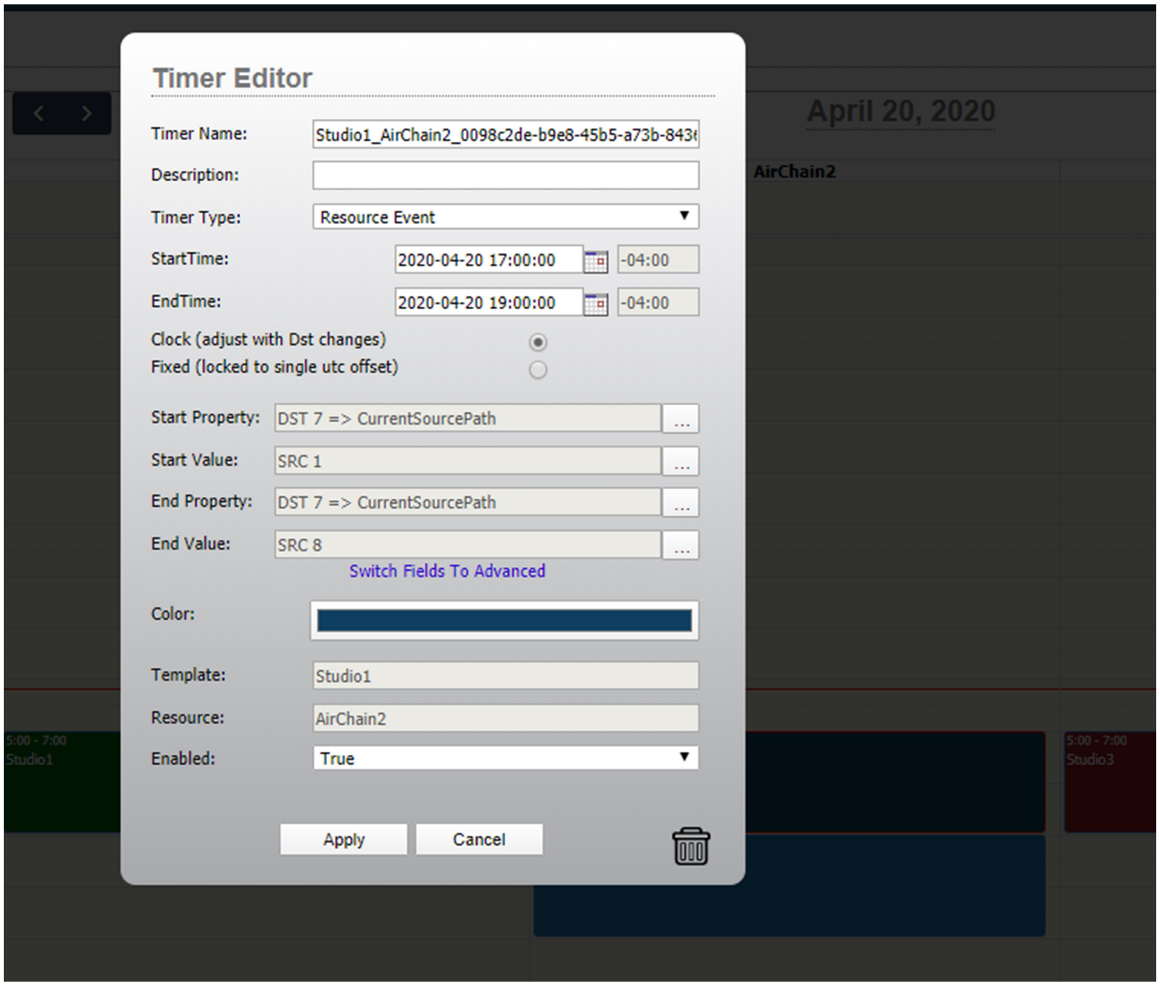
AirChain2: 7:00 - 9:00 MyRecurring

AirChain3: 5:00 - 7:00 Studio3

The scheduling module is comprised of 4 objects:

- Resources: A container for a set of events which also may include properties and values that interact with templates that are dropped onto the resource.
  - AirChains1, 2, and 3 are resources in the picture above.
- Templates: An event template with duration and start and end actions.
  - Templates are displayed in the list of rectangles on the left hand side of the screen in the picture above and can be drug onto the calendar.
- Calendars: A calendar based visualization of resources and their associated events along with drag and drop visualizations of templates.
  - This image above is a calendar with three resources, 4 currently displayed ResourceEvents, and 4 templates.
- ResourceEvents: An event with a start and end time and action to perform at the start and end. The action data may be comprised of information from both the resource and template used to create the event.
  - The events are displayed at the time values in the calendar.

The user work flow involves selecting an event template and dragging it into the correct resource in the calendar, then adjusting the time duration or placement by dragging the event or event bottom edge. That is all the user has to do to create an event, though details of the event may be further manipulated if the user has the rights to do so by clicking on the event as shown below:



Timer Editor

Timer Name: Studio1_AirChain2_0098c2de-b9e8-45b5-a73b-843

Description:

Timer Type: Resource Event

StartTime: 2020-04-20 17:00:00 -04:00

EndTime: 2020-04-20 19:00:00 -04:00

Clock (adjust with Dst changes) ●

Fixed (locked to single utc offset) ○

Start Property: DST 7 => CurrentSourcePath

Start Value: SRC 1

End Property: DST 7 => CurrentSourcePath

End Value: SRC 8

Switch Fields To Advanced

Color:

Template: Studio1

Resource: AirChain2

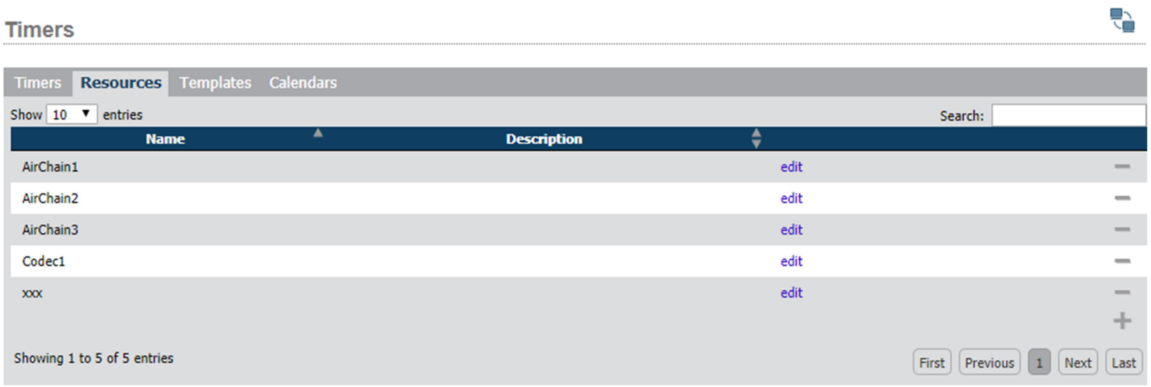
Enabled: True

In the calendar shown earlier, each event makes a route change of the studio source defined in the template to the AirChain destination which is defined by the resource. So when we drop a studio onto the resource, the destination from the resource is combined with the source from the event template in a new event with the correct time parameters that we define by dragging the duration. This allows the scheduling of studio to airchain routing to be a simple drag and drop procedure.

It is also important to note that no changes take effect until the Apply button is selected. The apply and cancel buttons will flash when there are pending changes that have not been committed.

Creating a Resource

Let’s dig a little deeper into how resources, templates, calendars, and resource events are created and interact. As an Administrator to create a calendar such as the one we see above, we would first need to define the resources and templates. To do this, click on the Timers navigation bar link and then click on the Resources tab.

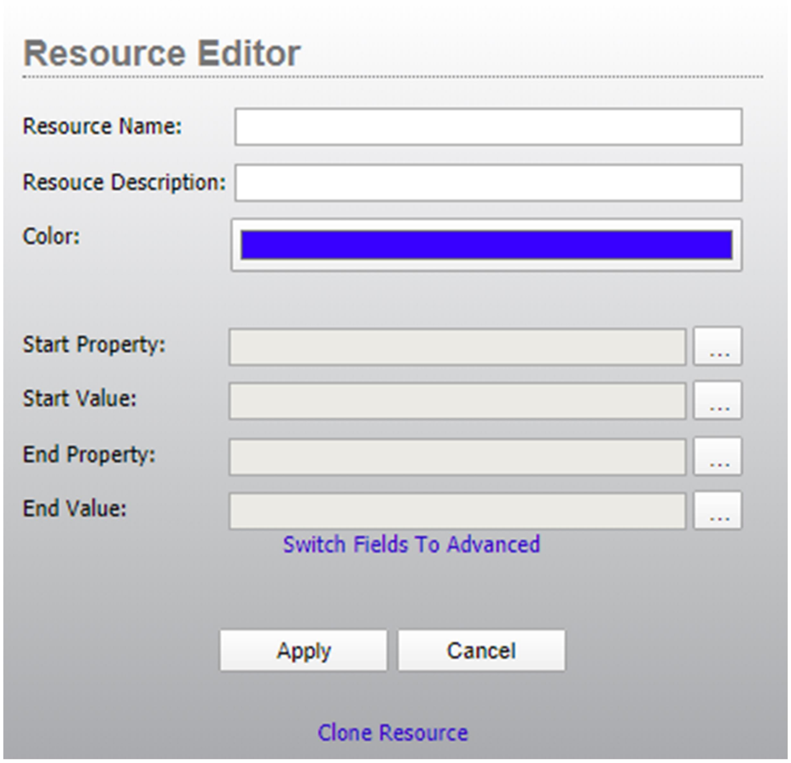


Show 10 entries Search:

Name	Description		
AirChain1		edit	−
AirChain2		edit	−
AirChain3		edit	−
Codec1		edit	−
xxx		edit	−

Showing 1 to 5 of 5 entries

Click on the plus icon to add a new resource:



Resource Editor

Resource Name:

Resouce Description:

Color:

Start Property:

Start Value:

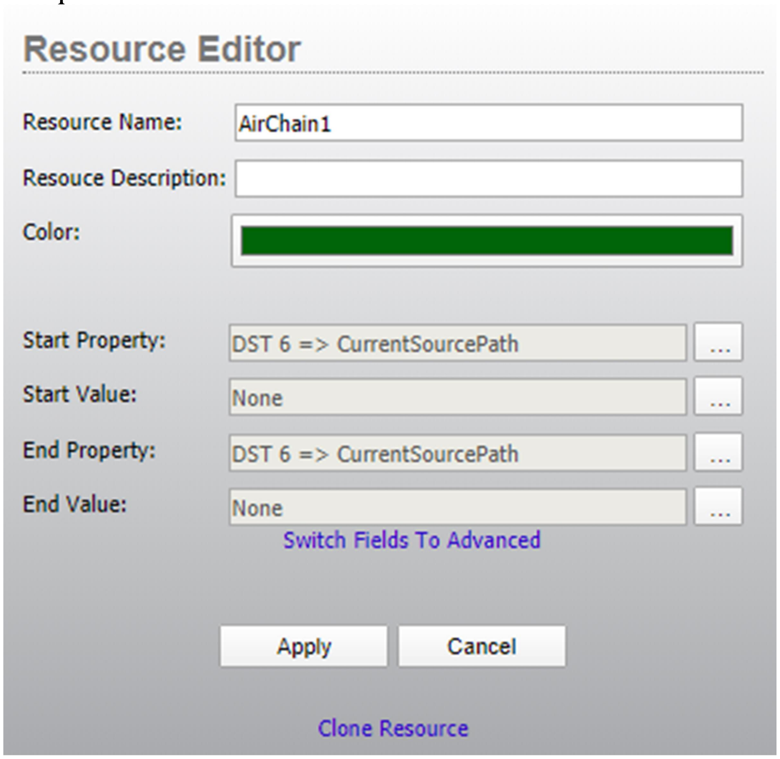
End Property:

End Value:

Each resource can have a name and description, a color, and a start property and value and end property and value. The start property and value will be what happens at a newly created event start, and the end property and value will be what happens at a newly created event end. In each case where the start and end property fields are used (resources, templates, and events), if an end action is not defined, nothing will happen at the end of the event time range.

It is important to understand that the values defined here are designed to be overridden by the template when a new template is dropped on the resource unless the template says to take data from the resource. We will see more of this when we get to templates below. So for example, we could leave all of the Start and End fields blank and define what happens only in the template. In the Studio and AirChain example above, we clicked on the ellipsis button and defined destinations and blank sources in these fields. The dialogs for property and value selections are the same as scene item selections and the timer changes in 1.7.1.04 above. In fact 1.7.1.04 is largely a preparatory step in the software development to getting scheduling done.

Click on the ellipsis buttons for each field and select the air chain destinations. The sources can be set to none or anything you like as we will override them in templates below.



Resource Editor

Resource Name: AirChain1

Resouce Description:

Color: 🟩

Start Property: DST 6 => CurrentSourcePath

Start Value: None

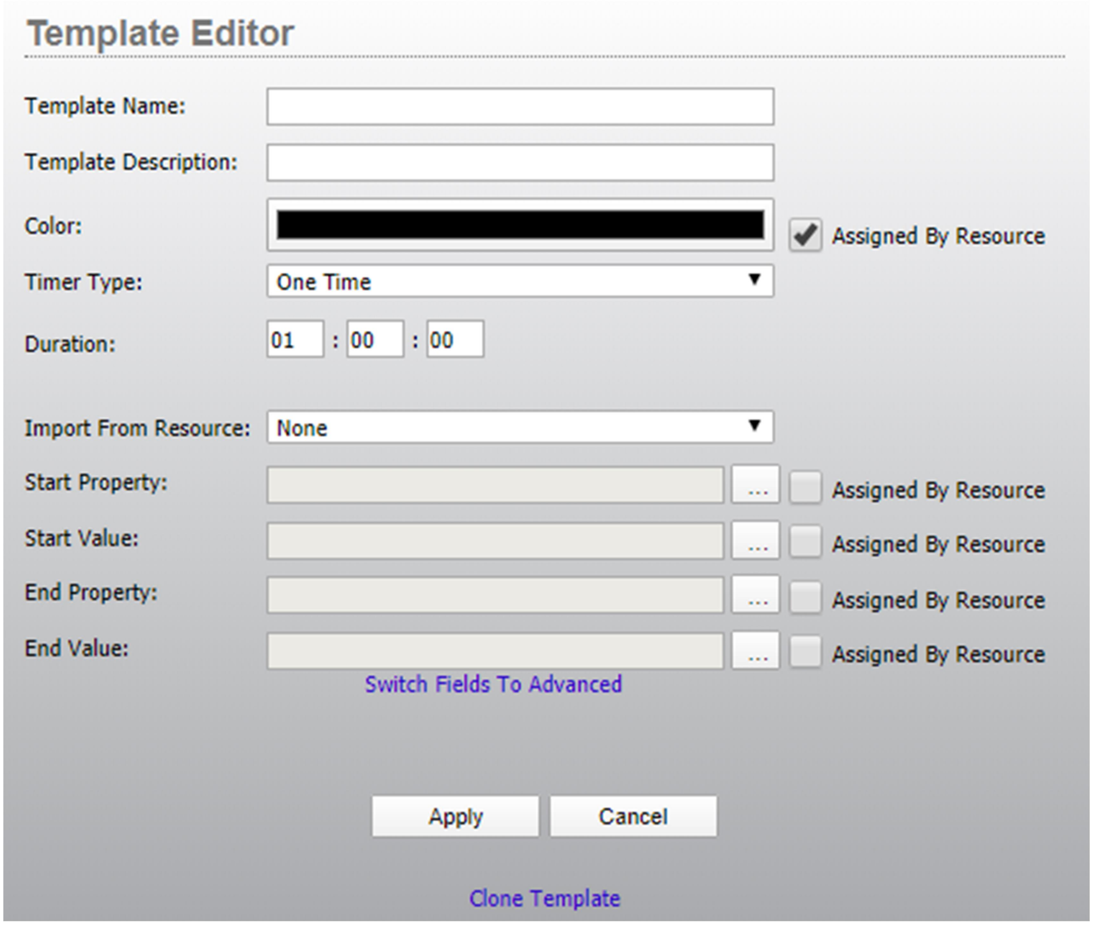
End Property: DST 6 => CurrentSourcePath

End Value: None

A color can also be picked. The clone resource link allows for quick duplication where you need to create several resources with only slight differences between them. Each resource does require a unique name.

Creating a Template

After creating several resources with different destinations for your Airchains, click on the templates tab and then use the plus icon to define a new template.



Template Editor

Template Name:

Template Description:

Color: ■ ☑ Assigned By Resource

Timer Type: One Time ▼

Duration: 01 : 00 : 00

Import From Resource: None ▼

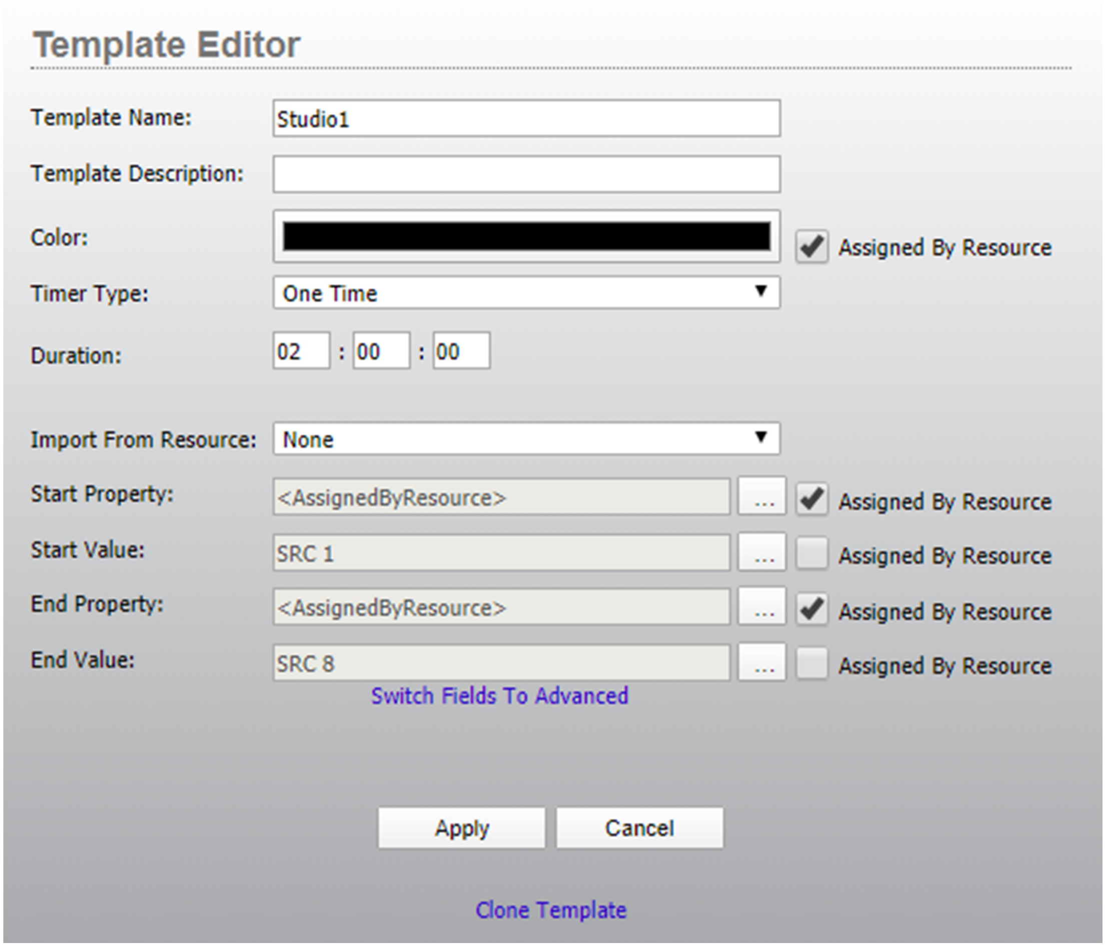
Start Property: ... □ Assigned By Resource

Start Value: ... □ Assigned By Resource

End Property: ... □ Assigned By Resource

End Value: ... □ Assigned By Resource

You will notice a checkbox next to several fields called Assigned by Resource. Enabling this checkbox indicates that the field will be obtained from the equivalent field in the resource when the template is dropped onto the resource. So for example, use the import from resource drop down to import the settings for the properties and values from one of the Airchain resources you created earlier as it is easier than manually selecting them again. Then provide a name and enable the checkboxes on the start and end properties. The duration is the default duration that will be used when dropping the template in the calendar but then can be adjusted by dragging the event. The result will look like:



Template Editor

Template Name: Studio1

Template Description:

Color: ☑ Assigned By Resource

Timer Type: One Time ▼

Duration: 02 : 00 : 00

Import From Resource: None ▼

Start Property: <AssignedByResource> ... ☑ Assigned By Resource

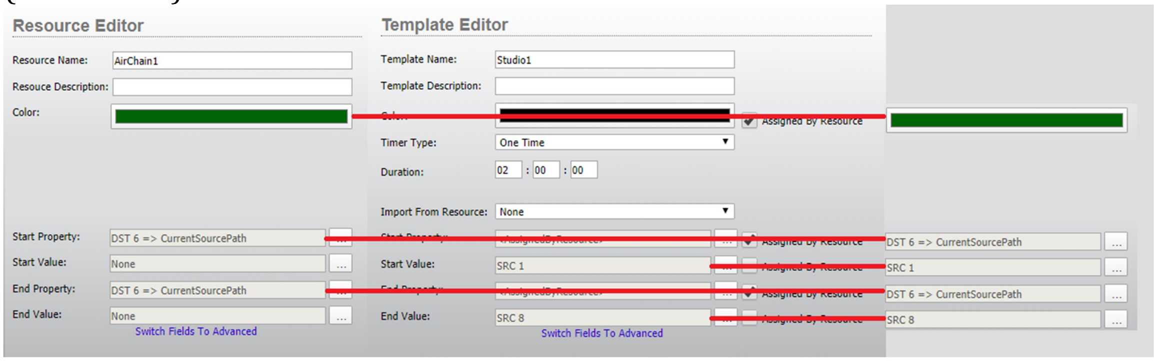
Start Value: SRC 1 ... □ Assigned By Resource

End Property: <AssignedByResource> ... ☑ Assigned By Resource

End Value: SRC 8 ... □ Assigned By Resource

Template to Resource Relationship

What we are saying is that when this template is dropped onto the resource, the template will provide the source values and the resource will provide the property (destination) and color values.



Resource Editor

Resource Name: AirChain1

Resouce Description:

Color:

Start Property: DST 6 => CurrentSourcePath ...

Start Value: None ...

End Property: DST 6 => CurrentSourcePath ...

End Value: None ...

Template Editor

Template Name: Studio1

Template Description:

Color: ☑ Assigned By Resource

Timer Type: One Time ▼

Duration: 02 : 00 : 00

Import From Resource: None ▼

Start Property: <AssignedByResource> ... ☑ Assigned By Resource

Start Value: SRC 1 ... □ Assigned By Resource

End Property: <AssignedByResource> ... ☑ Assigned By Resource

End Value: SRC 8 ... □ Assigned By Resource

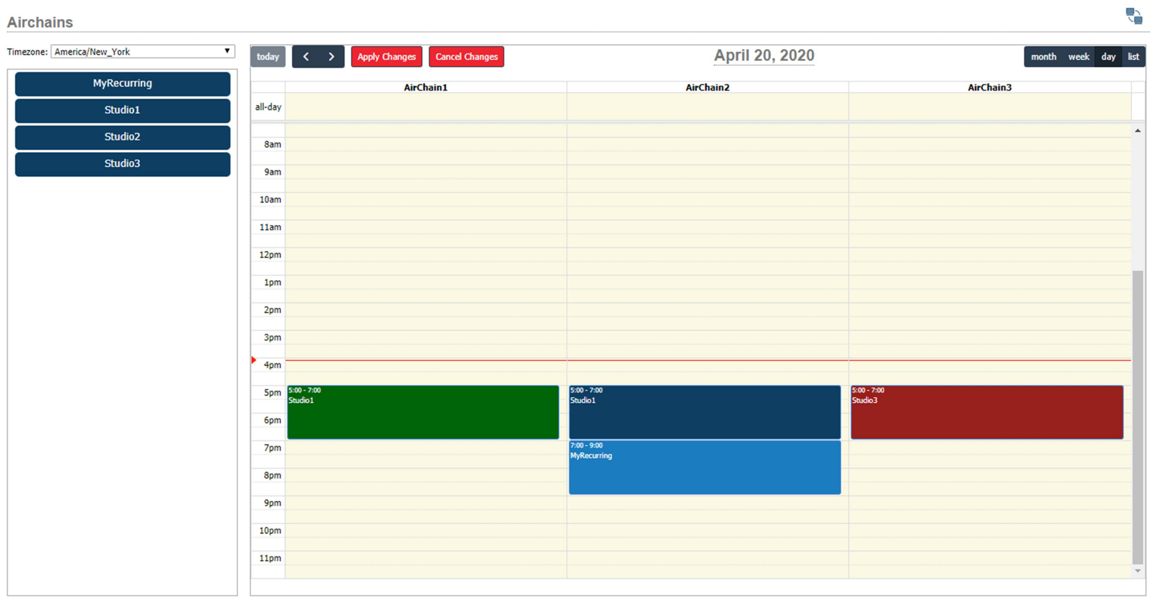
DST 6 => CurrentSourcePath ...

SRC 1 ...

DST 6 => CurrentSourcePath ...

SRC 8 ...

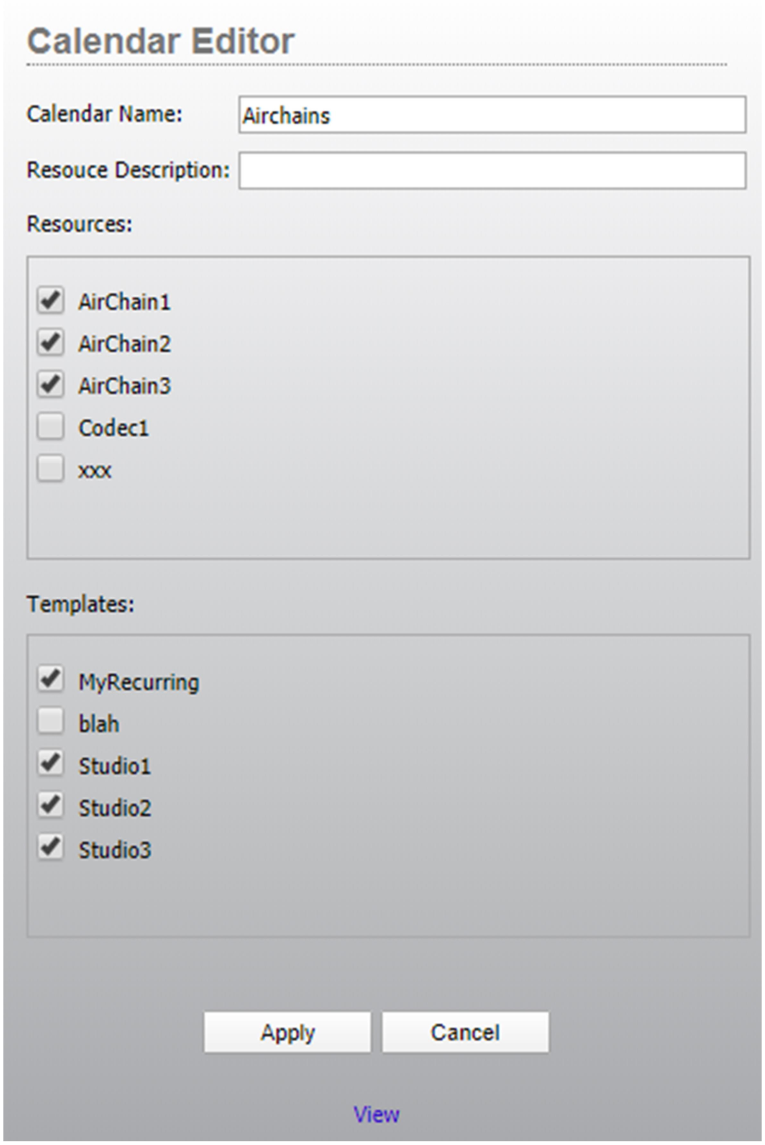
It is important to understand that if we had not turned on any of the "Assigned by Resource" checkboxes, all of the parameters from the template would be used. Why is it useful to split the parameters up in this way? If we look at our original calendar again we will see its usefulness.



Here if you drag studio 1 onto Airchain 1, then the event will have the airchain destination and the studio source. If you decide you made a mistake, you can drag the event to Airchain 2, and the routing destination for the event will update to reflect the change.

Creating a Calendar

Going back to our Administrative configuration again, we still need to actually create the calendar. To do that, click on the calendars tab and click the Plus button. The calendars dialog allows us to create a new named calendar and configure what resources and templates are allowed to be used by the calendar.



Calendar Editor

Calendar Name: Airchains

Resouce Description:

Resources:

☑ AirChain1
☑ AirChain2
☑ AirChain3
□ Codec1
□ xxx

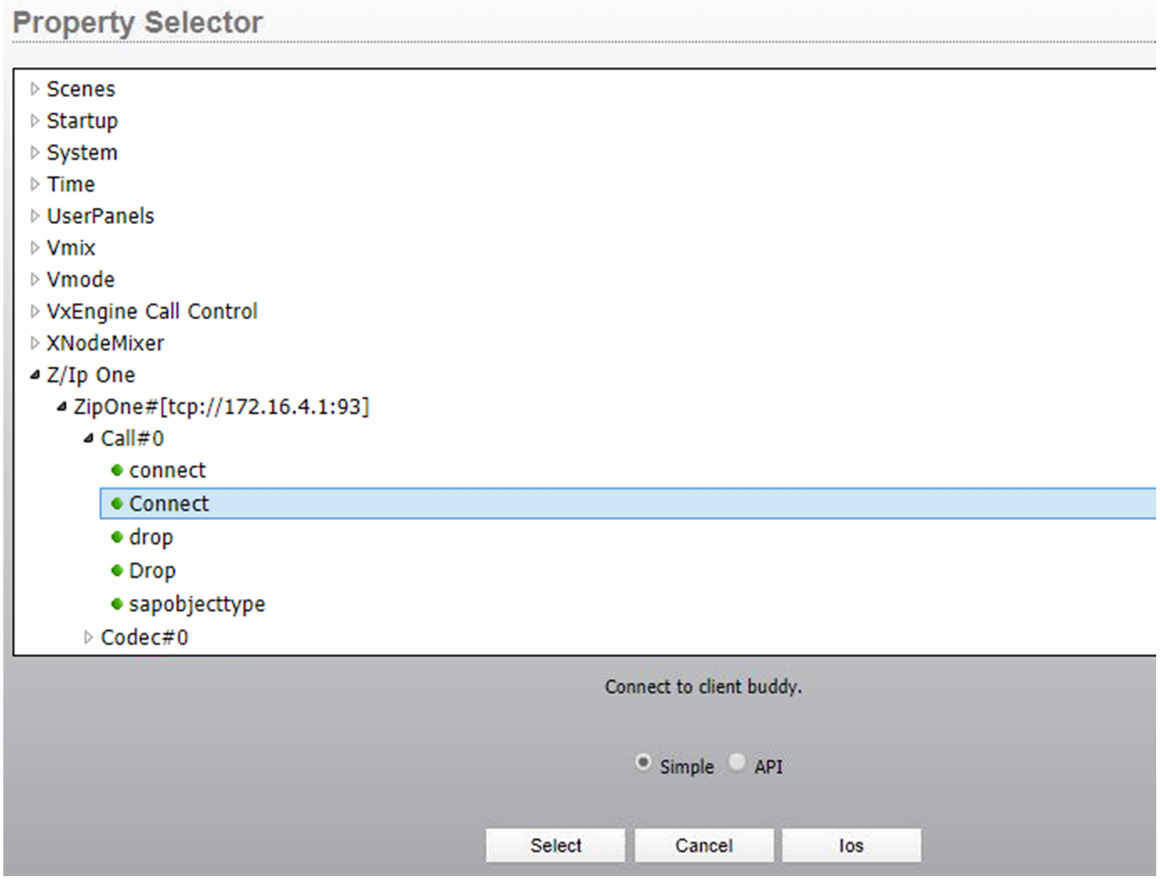
Templates:

☑ MyRecurring
□ blah
☑ Studio1
☑ Studio2
☑ Studio3

Supply a name and description for the calendar and then select the resources and templates that are available for use. It is important to note that the same resources and templates can be used on multiple calendars. Think of a calendar as a view of resource based events and the templates as the available events that may be applied. After applying the new calendar settings, it becomes available in the calendar list and the view button will allow a user to start working with it.

Zip One Dialer Example

While all of the examples so far have been to make route changes, they do not have to be. Any property in the system can be used in a resource or template and therefore scheduled. So for example we could change the resource and template to use ZipOne dialing parameters. When creating the resource after clicking the ellipsis button, use the Endpoints button to obtain the normal logic flows endpoint selector.



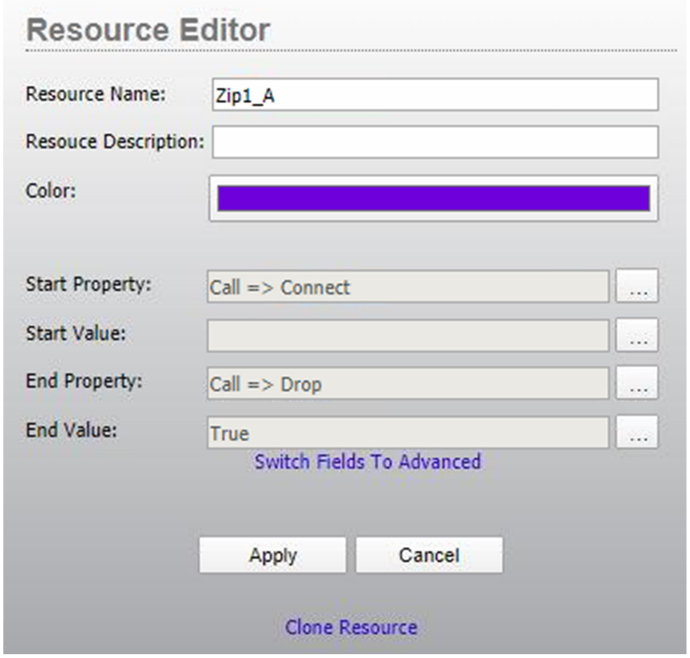
Property Selector

- Scenes
- Startup
- System
- Time
- UserPanels
- Vmix
- Vmode
- VxEngine Call Control
- XNodeMixer
- Z/Ip One
  - ZipOne#[tcp://172.16.4.1:93]
    - Call#0
      - connect
      - Connect
      - drop
      - Drop
      - sapobjecttype
    - Codec#0

Connect to client buddy.

● Simple ○ API

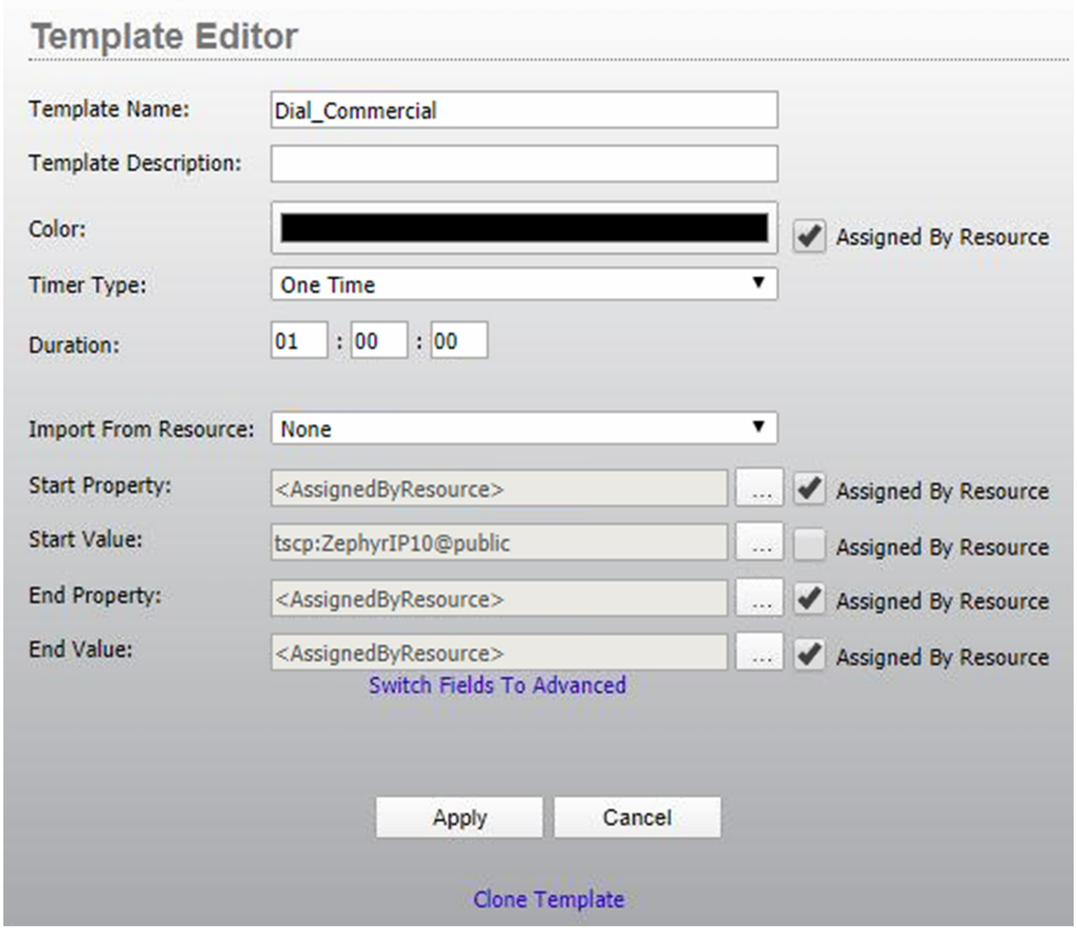
Therefore if we create our resource to look like:



Resource Editor

Resource Name:	Zip1_A
Resouce Description:	
Color:	🟪
Start Property:	Call => Connect
Start Value:	
End Property:	Call => Drop
End Value:	True

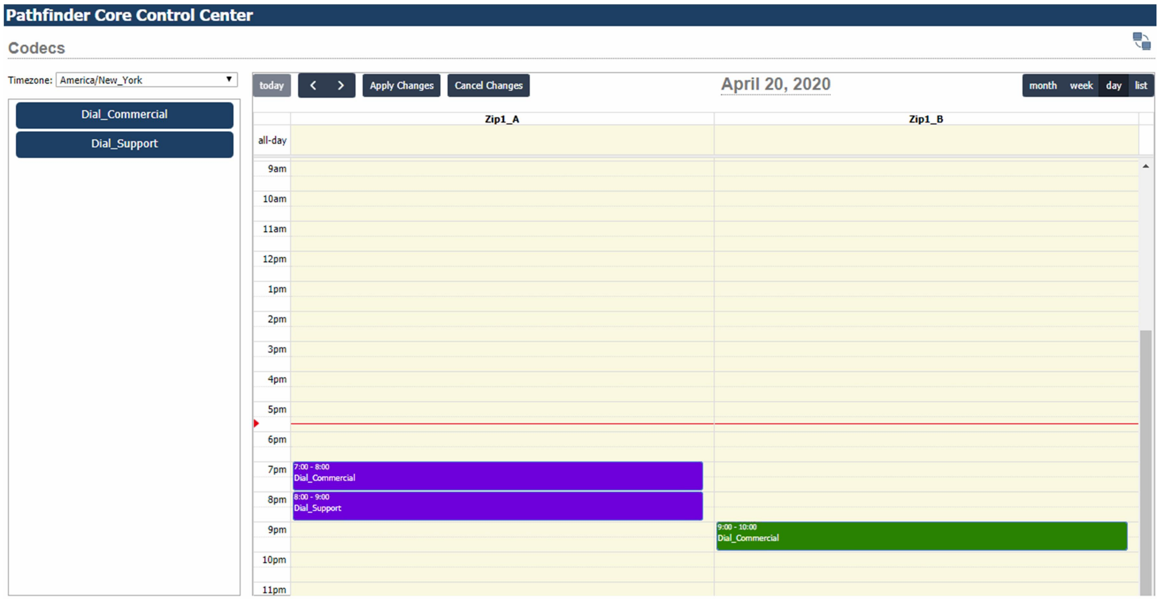
And our template to look like



Template Editor

Template Name:	Dial_Commercial	
Template Description:		
Color:	■	☑ Assigned By Resource
Timer Type:	One Time ▼	
Duration:	01 : 00 : 00	
Import From Resource:	None ▼	
Start Property:	<AssignedByResource> ...	☑ Assigned By Resource
Start Value:	tscp:ZephyrIP10@public ...	□ Assigned By Resource
End Property:	<AssignedByResource> ...	☑ Assigned By Resource
End Value:	<AssignedByResource> ...	☑ Assigned By Resource

In this instance the resource defines the Zip one Connect property as well as the Zip one drop property and value. And the template only supplies the dial location. Using this methodology we could create a calendar for scheduling Zip one dialing such as:



Timezone: America/New_York

7:00 - 8:00 Dial_Commercial

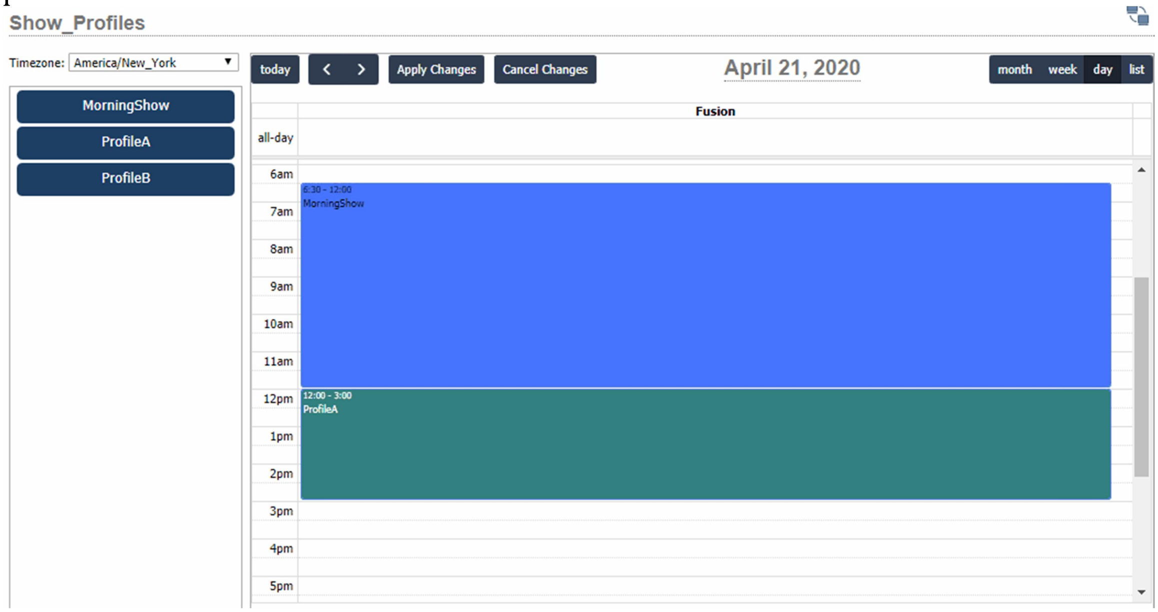
8:00 - 9:00 Dial_Support

9:00 - 10:00 Dial_Commercial

In this example our event templates are dial locations and our resources are the Zip One to use for the dialing.

Fusion Show Profiles

As one last example, we could create a calendar to schedule the loading of show profiles on a console:



Timezone: America/New_York

6:30 - 12:00 MorningShow

12:00 - 3:00 ProfileA

In this example, the templates are show profiles and the only resource on the calendar is the fusion console on which they are being scheduled.

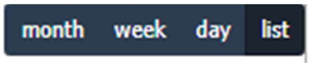
The point we are trying to make with these examples is that anything Pathfinder Core PRO knows about can be used for scheduling and multiple calendars may be created in order to schedule different types of things.

Timezone Selector

The calendar has a time zone selector in the top left corner. By default when the calendar is displayed, it queries PathfinderCore PRO for its current time zone and sets the calendar time zone to that time zone. Changing the time zone changes how the events are viewed. Adding events while in a different time zone than the Pathfinder Core PRO time zone will cause the event times to be converted to the Pathfinder Core PRO time zone when the apply button sends the changes to the system.

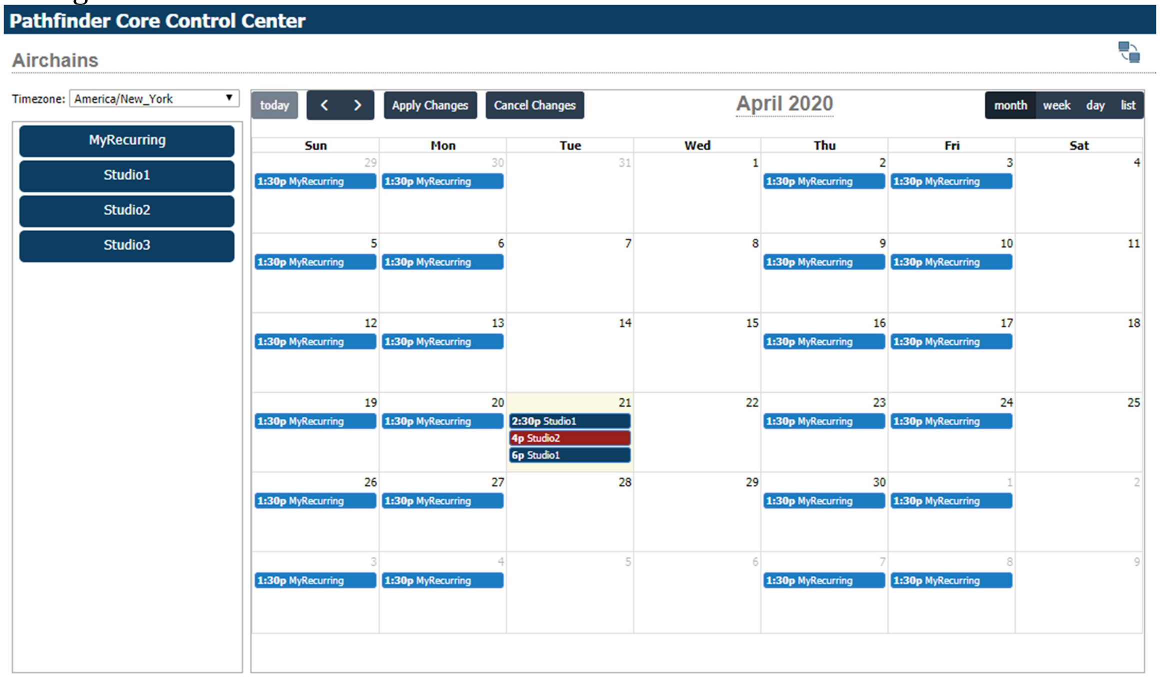
Calendar Controls and Views

Several controls exist across the top of the calendar. On the right hand corner, there are several buttons which change the view.



Clicking these buttons will change the calendar view to one of the following:

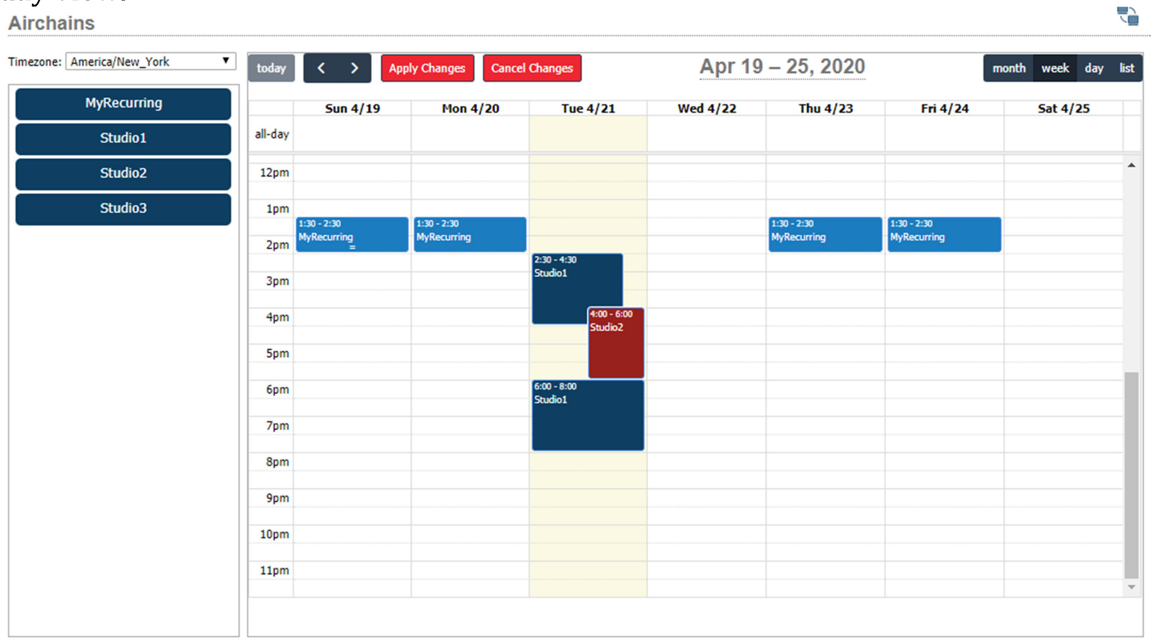
Month: Dropping templates onto the month view is prohibited as there is not enough resolution for time selection.



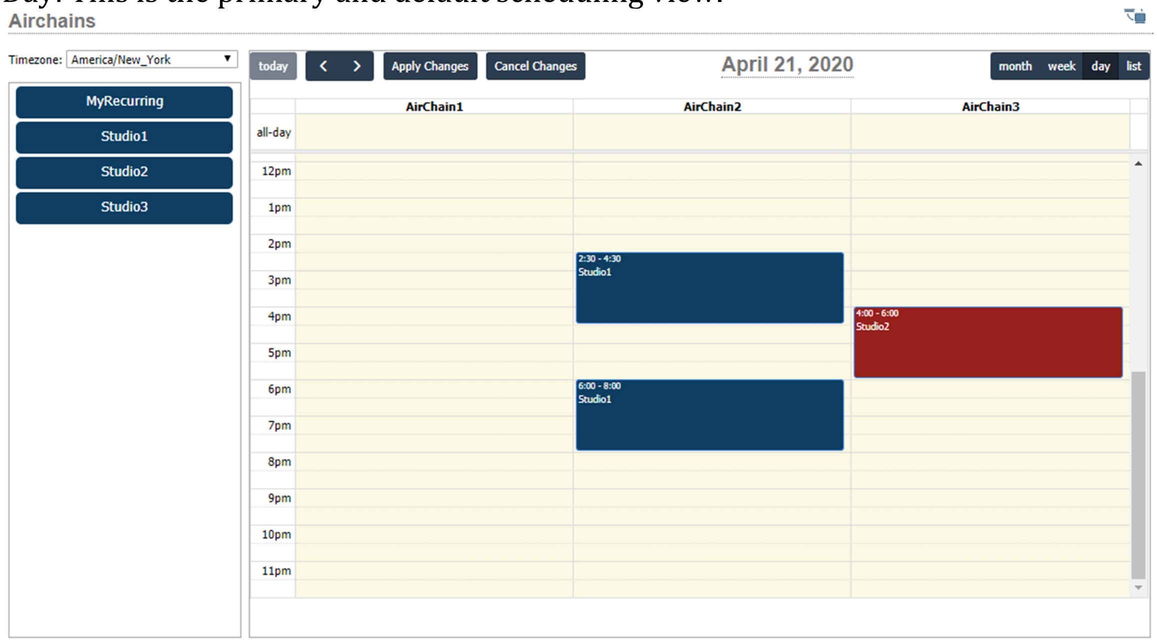
Timezone: America/New_York

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29 1:30p MyRecurring	30 1:30p MyRecurring	31	1	2 1:30p MyRecurring	3 1:30p MyRecurring	4
5 1:30p MyRecurring	6 1:30p MyRecurring	7	8	9 1:30p MyRecurring	10 1:30p MyRecurring	11
12 1:30p MyRecurring	13 1:30p MyRecurring	14	15	16 1:30p MyRecurring	17 1:30p MyRecurring	18
19 1:30p MyRecurring	20 1:30p MyRecurring	21 2:30p Studio1 4p Studio2 6p Studio1	22	23 1:30p MyRecurring	24 1:30p MyRecurring	25
26 1:30p MyRecurring	27 1:30p MyRecurring	28	29	30 1:30p MyRecurring	1 1:30p MyRecurring	2
3 1:30p MyRecurring	4 1:30p MyRecurring	5	6	7 1:30p MyRecurring	8 1:30p MyRecurring	9

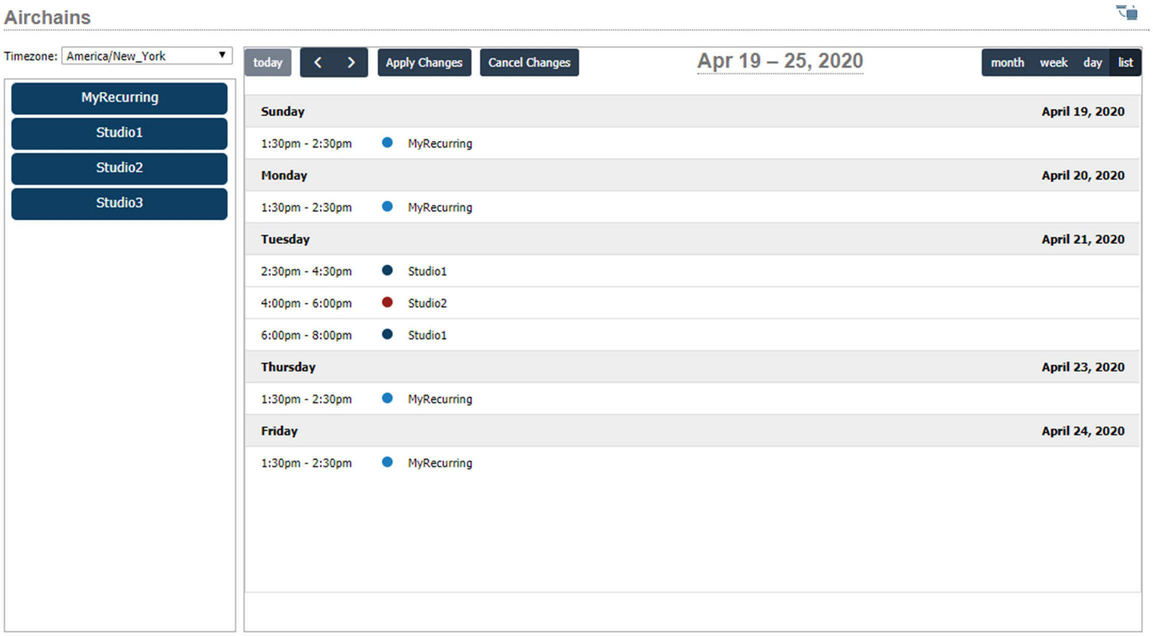
Week: Dropping a template onto a week view will always assign the template to the first resource assigned to the calendar. This can then be adjusted if necessary in the day view.


Timezone: America/New_York

Day: This is the primary and default scheduling view.


Timezone: America/New_York

List: This view is for informational purposes only and cannot be a drag and drop target.

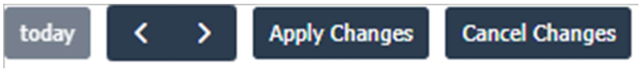


Timezone: America/New_York

Apr 19 – 25, 2020

Sunday		**April 19, 2020**
1:30pm - 2:30pm	● MyRecurring	
Monday		**April 20, 2020**
1:30pm - 2:30pm	● MyRecurring	
Tuesday		**April 21, 2020**
2:30pm - 4:30pm	● Studio1	
4:00pm - 6:00pm	● Studio2	
6:00pm - 8:00pm	● Studio1	
Thursday		**April 23, 2020**
1:30pm - 2:30pm	● MyRecurring	
Friday		**April 24, 2020**
1:30pm - 2:30pm	● MyRecurring	

On the left hand side is a today button that will bring the view back to the current day as well as left right arrows for navigating to the next day/week/month depending on the view.



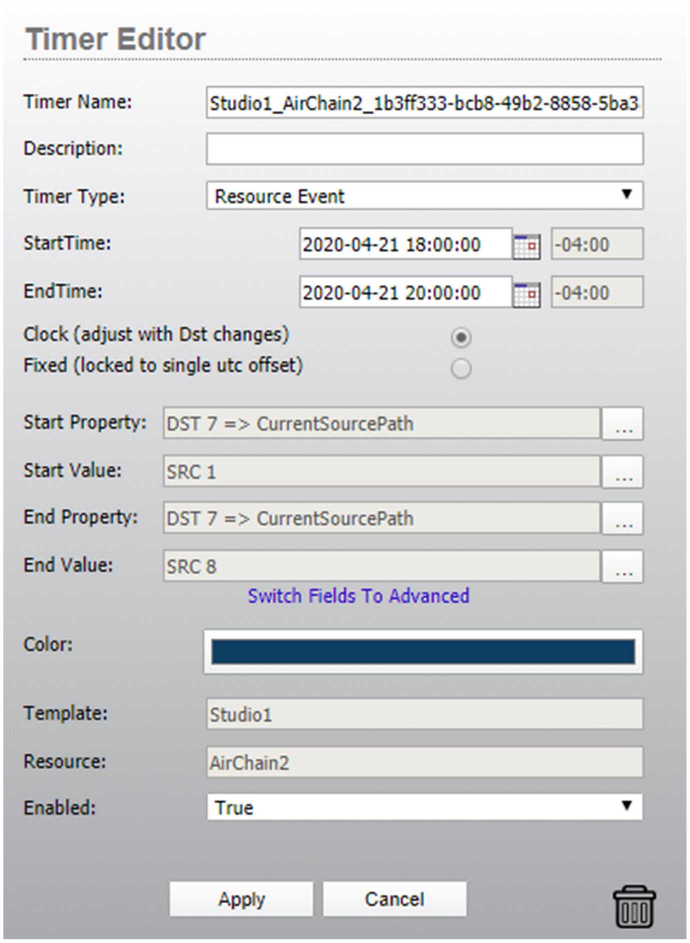
The apply and cancel buttons apply or cancel any pending changes to the Pathfinder Core PRO timer engine. These buttons will flash if there are pending changes. Events which have not been applied will not exist or be acted upon by PathfinderCore PRO.

Colors

You can define colors in the templates, resources, and in the events themselves and there is some internal logic as to which color is used when an event gets created. If the template has the Assigned by Resource checkbox enabled for the color then the resource color will be used. This is the default because when viewing the calendar using a week or month view where the resource columns are not visible, events associated with a specific resource will all have the same color making it easy to tell which events belong to which resource. However you may prefer to have the color relate to the template that created the event, in which case deselecting the Assigned by Resource checkbox in the template will cause newly created events to use the template color. Finally, once an event is created it can be edited by clicking on the event and a color specific to that event instance can be selected.

Event Editing

After an event is created, if the settings generated by the resource/template combination are not what you want, you can click on the event and edit it to do whatever you wish.



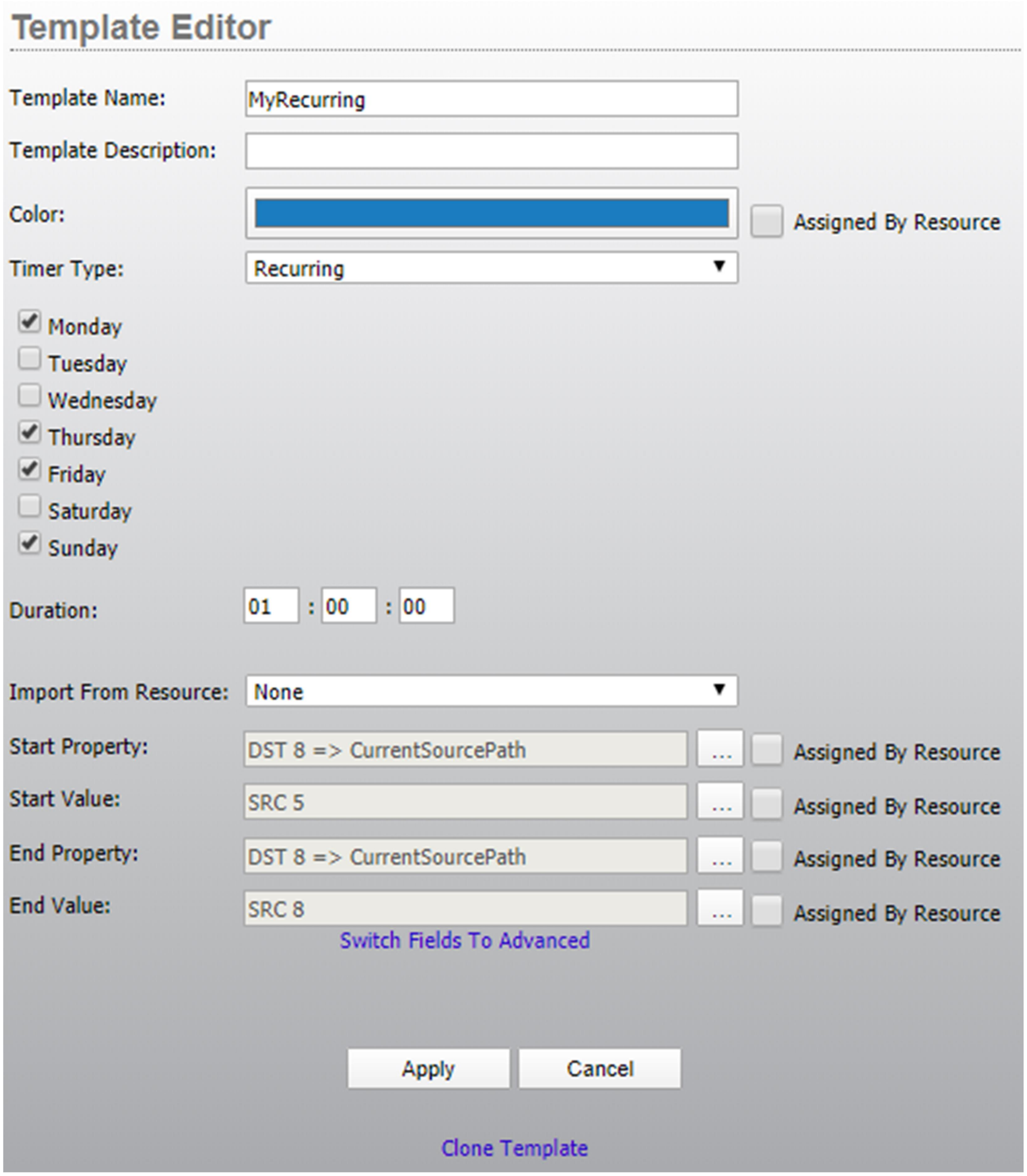
Timer Editor

Timer Name:	Studio1_AirChain2_1b3ff333-bcb8-49b2-8858-5ba3
Description:	
Timer Type:	Resource Event ▼
StartTime:	2020-04-21 18:00:00 -04:00
EndTime:	2020-04-21 20:00:00 -04:00
Clock (adjust with Dst changes)	●
Fixed (locked to single utc offset)	○
Start Property:	DST 7 => CurrentSourcePath ...
Start Value:	SRC 1 ...
End Property:	DST 7 => CurrentSourcePath ...
End Value:	SRC 8 ...
Color:	
Template:	Studio1
Resource:	AirChain2
Enabled:	True ▼

This is basically the same timer editing dialog that is discussed in the manual and in version 1.7.1.04 documentation above.   The only differences are the presence of the color, template, and resource fields and different options in the timer type field. Calendar based timers must be of type Resource Event or Recurring Resource Event. These two types equate to the DateTime event type and the DayOfWeek type but are designed to work with the calendar. You can change the name of the event from the default one that is generated, but the name must be unique within the timers in the system.

Recurring Events

When creating an event template or editing an event, it is possible to create it as a recurring event rather than a one-time event by selecting the Recurring timer type.



Template Editor

Template Name: MyRecurring

Template Description:

Color: 🟦 □ Assigned By Resource

Timer Type: Recurring ▼

☑ Monday
□ Tuesday
□ Wednesday
☑ Thursday
☑ Friday
□ Saturday
☑ Sunday

Duration: 01 : 00 : 00

Import From Resource: None ▼

Start Property: DST 8 => CurrentSourcePath ... □ Assigned By Resource

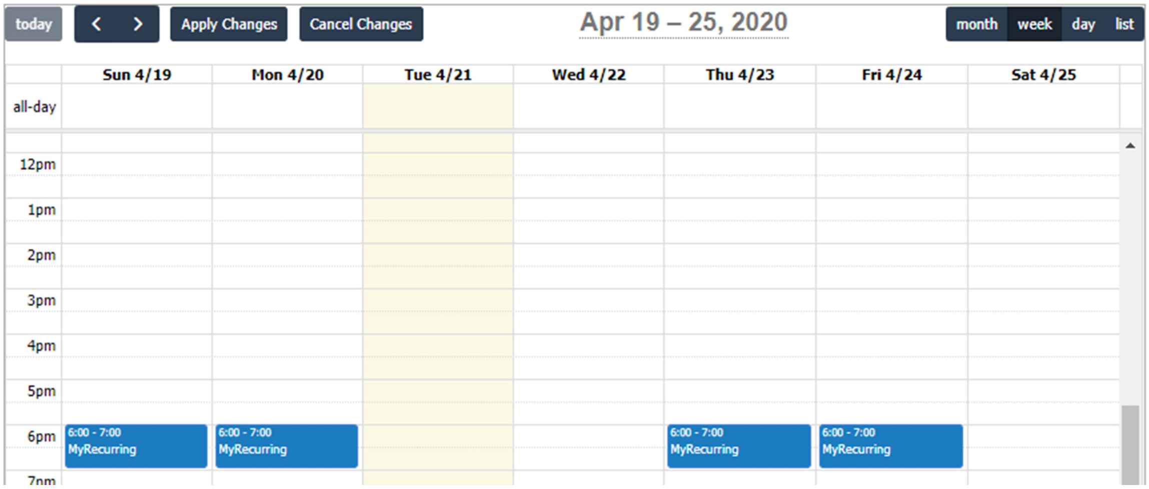
Start Value: SRC 5 ... □ Assigned By Resource

End Property: DST 8 => CurrentSourcePath ... □ Assigned By Resource

End Value: SRC 8 ... □ Assigned By Resource

This will add fields for selecting the days of the week on which the event should occur.

Dragging a recurring template onto the calendar will cause it to appear at the selected timeslot on all days of the week for which the template is configured.



One thing that can cause confusion is if you drag a recurring event to a day view on a day of the week that the recurring event is not scheduled to take place. This will appear to have no effect, but if you use the week view to zoom out you will see that the event has been added and applied to the correct days.

It is also important to note than changing the length or any of the properties of a recurring event will make that change in the base recurring event and will therefore affect all days the event takes place. For example, in the week view above, if you grab the duration handle on the bottom of one of these events and make the event longer, all of them will change because they are all just views of a single recurring event.

Deleting an Event

When editing an event, you should also notice the trashcan icon in the bottom right corner.

To delete an event that has been added to a calendar resource, click on the event and then click on the trash can icon. The system will ask you if you are sure you want to delete the event. It is important to note that deleting a recurring event will delete all instances of that recurring event. There is currently no way to delete a single day of a recurring event. Instead you would need to disable that day in the event and then re-enable it afterwards if you want it to happen on that day in the future.

Future

Work on the scheduling module is still ongoing so beta testing and feedback is welcome. Some of the to-do items we are already considering include:

- Time granularity adjustment field allowing you to set the granularity of the time grid to something other than 30 minute increments.

- Ability to display datetime and dayofweek non-resource based events in a Default resource object.
- Additional user security features for fine tuning access rights to scheduling functionality.

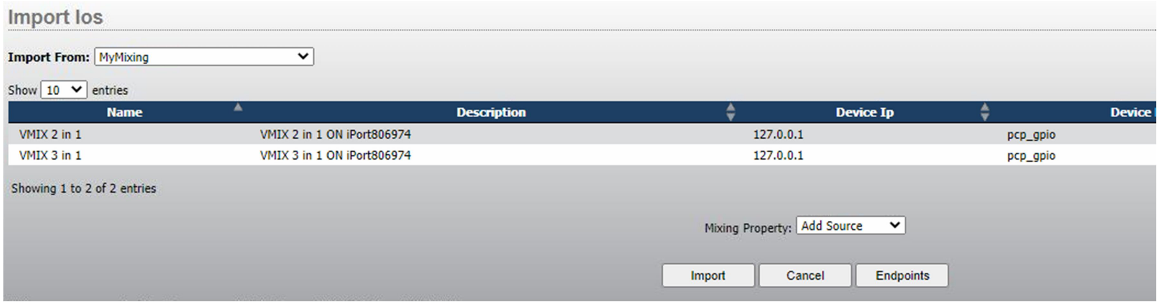
Version 1.7.6.08 Changes

Build Process

This version is a product of redesigning how PathfinderCore PRO code, images, and update packages are built to bring that process more in line with other Telos Alliance products and continuous integration policies. Please report any bugs you encounter.

Scenes, Timers, and Virtual Mixing Routers

Virtual mixing routers are inherently different in how they function from other routers. For details on Virtual Mixing Routers, see their section in this document under Version 1.7.0.02. With Virtual mixing destinations you do not change the source that is routed to the destination. Instead you add sources that will be mixed together by the mixing destination up to the supported quantity of the destination. As a result it uses different routing properties than a normal router. For example, AddSource adds a source to the mixing destination, and RemoveSource removes a source from the destination freeing that slot up for another source. Therefore when importing a destination to a scene or timer event, you need to be able to define whether the change will add or remove a source. To support this, a new drop down will appear in the import destination list when creating a scene item or defining a start or end property in a timer.



Import Ios

Import From: MyMixing

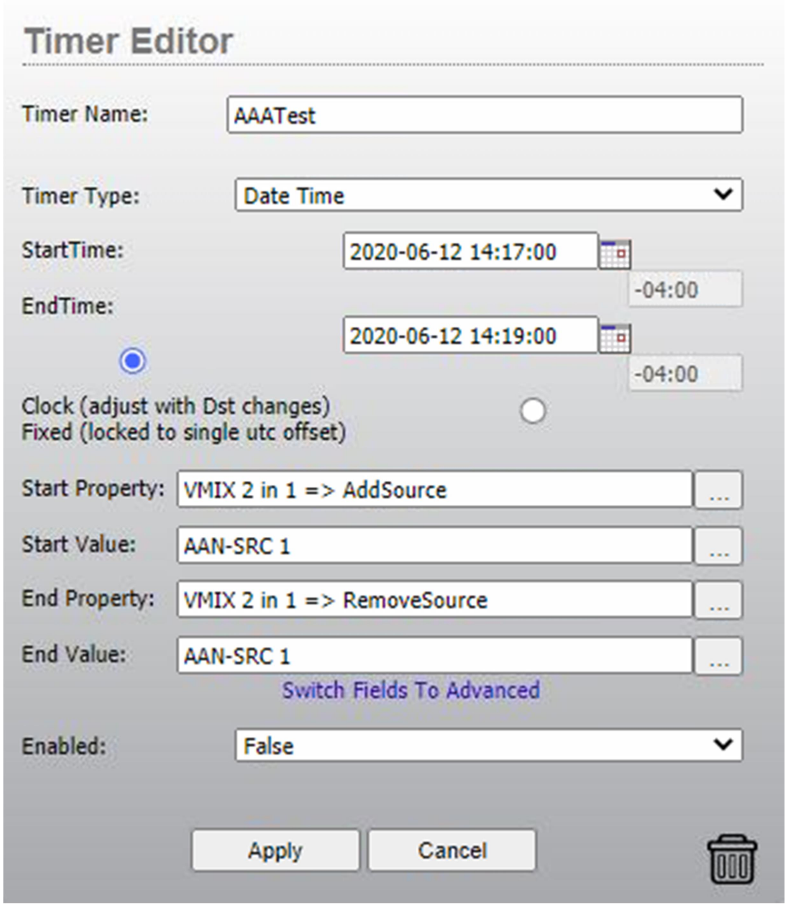
Show 10 entries

Name	Description	Device Ip	Device
VMIX 2 in 1	VMIX 2 in 1 ON iPort806974	127.0.0.1	pcp_gpio
VMIX 3 in 1	VMIX 3 in 1 ON iPort806974	127.0.0.1	pcp_gpio

Showing 1 to 2 of 2 entries

Mixing Property: Add Source

The Mixing property drop down allows you to select whether the operation will add or remove a selected source to/from the destination. For example:



Timer Editor

Timer Name: AAATest

Timer Type: Date Time

StartTime: 2020-06-12 14:17:00 -04:00

EndTime: 2020-06-12 14:19:00 -04:00

● Clock (adjust with Dst changes)

○ Fixed (locked to single utc offset)

Start Property: VMIX 2 in 1 => AddSource

Start Value: AAN-SRC 1

End Property: VMIX 2 in 1 => RemoveSource

End Value: AAN-SRC 1

Enabled: False

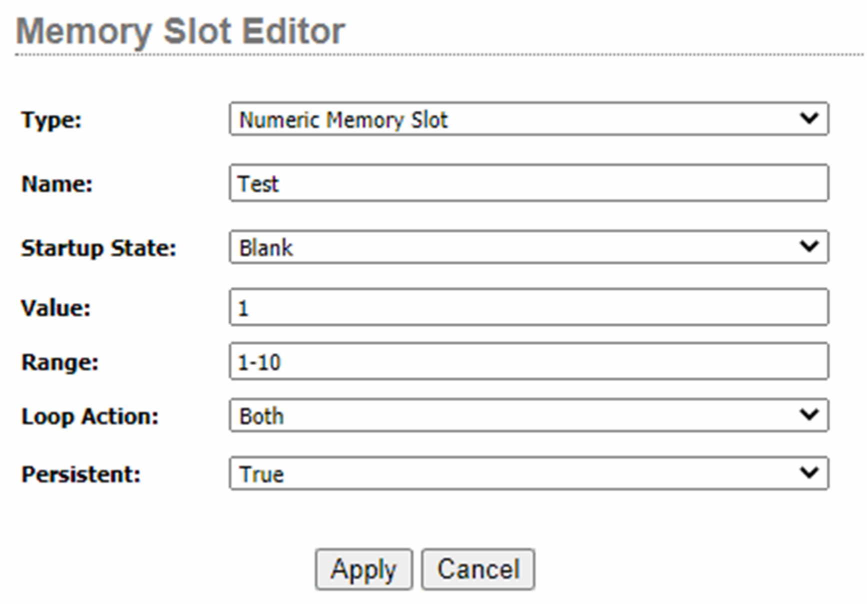
In this example the timer will add Source 1 to the destination at the start of the event time range, and it will remove it at the end.

Scenes and Virtual Mixing Routers

The AddSource and RemoveSource properties of a Virtual Mixing Router destination are write only properties. You are applying a source to the property to change the underlying routing state, but the property itself will not have a value. Whenever you use write only properties in a scene, the state cannot be determined and therefore the IsActive property will never become true on such scenes. In a future version we may add some special code to handle the situation of virtual mixing routers such that the scene will detect if after the source is applied whether it exists on the destination and adjust the value of the scene line accordingly. But for now scenes will not become IsActive using these properties.

Numeric Memory Slots – Range and LoopAction

Numeric Memory slots now have an optional range and loop action property.


Memory Slot Editor

Type: Numeric Memory Slot

Name: Test

Startup State: Blank

Value: 1

Range: 1-10

Loop Action: Both

Persistent: True

If the range field is left blank, then the numeric memory slot operates the way it always has. If you apply a range by filling the range field with a low and high value separated by a hyphen, then increment and decrement will not exceed the specified range. The LoopAction property only works in conjunction with a range and does nothing if no range is specified. The loop action field defines what happens when you try to increment or decrement beyond the specified range as follows:

- None = Incrementing beyond the max or decrementing below the min will do nothing and no change will be applied to the Value.
- Forward = Incrementing beyond the max will cause the value to loop back to the min, but decrementing below the min will make no changes.
- Backward = Decrementing below the min will cause the value to loop to the max, but incrementing beyond the max will make no changes.
- Both = Incrementing beyond the max will cause the value to loop to the min value, and decrementing below the min value will cause it to loop to the max value.

These properties make it easy to assert a valid range to a numeric memory slot as well as to implement things like paging functions where paging beyond the maximum number of pages will take you back to page 1.

Quasar 4 fader panel QSCONN Lwrp object

This version adds support for the QSCONN object in Quasar fader panels. It also creates a new object specifically for such panels. If you already have fader panels in your system, you may need to remove then and re-add them in order to get the correct object type. If they are appearing in your devices list as Quasar_4-Fader as opposed to QuasarFader4, you might need to remove and add the panel to PathfinderCore PRO again.

The QSCONN object is accessible via the API tree in logic flows under the Devices\<QuasarFaderPanelDevice>\LwrpInterpreter\LwrpRoot\QSCONN path. It is not a normally used object and so does not appear in the simple tree. It has two properties:

- Master: (RW): Holds the IP address of the master module to which this fader bank is attached.
- Status: (RO): Whether the Fader panel is currently connected or disconnected from the master panel.

The master property may be changed with a new ip address to change what master module the fader panel is connected to. This functionality should be used with care and discussed with support to be sure you understand the ramifications, but it can be used to make a fader module address a different master module dynamically.

Additionally in the current version of PathfinderCore PRO code and Quasar code, this Fader Module does not push changes to Core PRO. As a result status and master changes made in the web page of the device may not be reflected in Core PRO until the next poll time. Additionally changes made by one Core PRO server of this object may not be detected by a second server until its next poll time. This limitation may change in future software revisions.

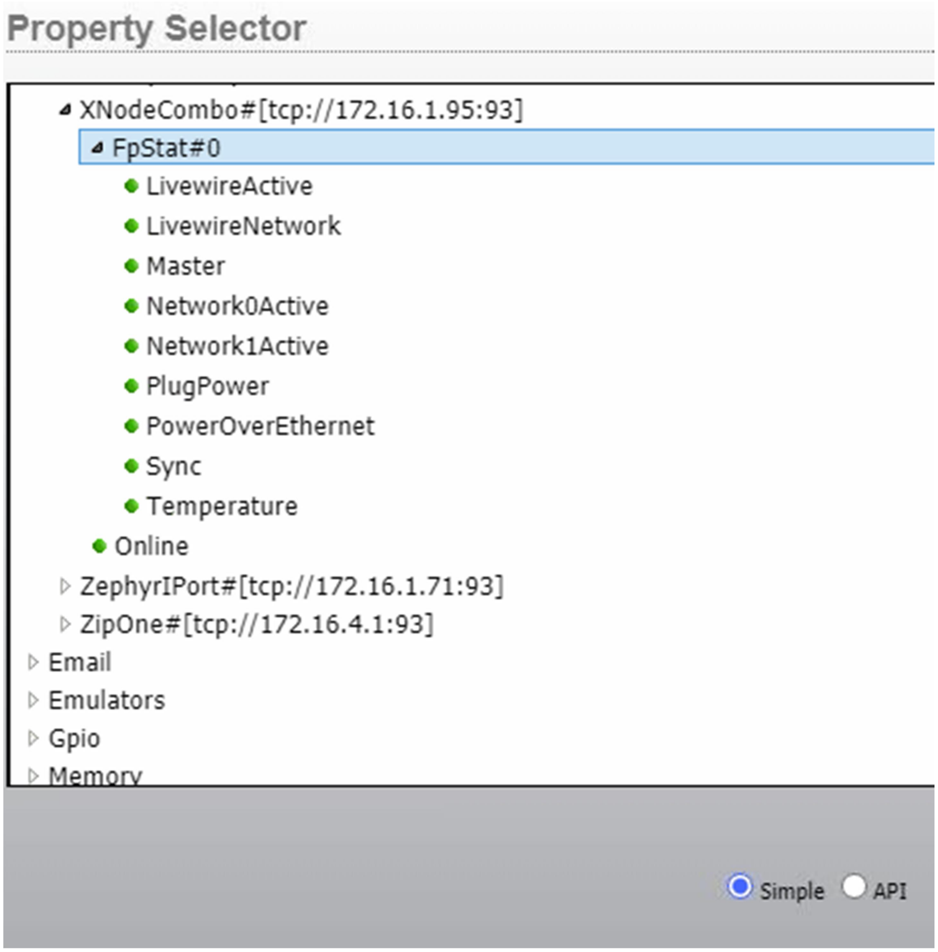
XNodes and FPSTAT Lwrp object

This version adds support for the FpStat object in Xnodes. The Fpstat object carries a number of read only properties related to temperature, power supply status, NIC link, etc. It can be found in the Simple tree under the DeviceConnections branch, and in the API tree under the device's lwrp path. The properties available are:

- Master: Whether this node is clock master
- Sync: Status information regarding the clock sync state
- LivewireActive: Whether the device is receiving livewire advertisements
- Temperature: Temperature in Celcius of the main board
- Plug Power: Whether the power supply has power from the IEC plug port.
- PowerOverEthernet: Whether the power supply is receiving power from the Ethernet port.
- Network0Active: Whether ethernet port 0 is active
- Network1Active: Whether ethernet port 1 is active
- LivewireNetwork: Which Nic (0 or 1) is the livewire network

It is important to note that these settings should be used in conjunction with the online property as opposed to replacing it. For example if all power is lost from the box, it is likely we will not actually see the change in the Plug Power or POE properties because we will have lost connection to the object that hosts those properties before we could read the change. So we must rely on the online state to know we are no longer connected. On the other hand if you have both plug and POE power applied and one of them is lost, we will still show as online but we can get an alert that one of the supplies is down. Similarly, if you lose connection on the NIC that you are using to monitor this object, then the NIC states in Core PRO for this object will not change. We have to rely on the Online state for that. On the other hand if the NICs in the XNode are set up to be redundant and one fails, we can now detect and alarm this condition.

In the current version of Xnode software, these parameters require polling to obtain state and do not automatically push changes. By default PathfinderCore PRO will poll these properties every 15 seconds. There is an advanced option in the configuration advanced options to change this called FpStatePollRate. You may need to view the default settings to obtain and override its syntax. This is a global setting and should be changed with caution and understanding of the consequences. In networks with large numbers of xnodes, decreasing this poll rate can affect the amount of communication that PathfinderCore PRO needs to process.



Property Selector

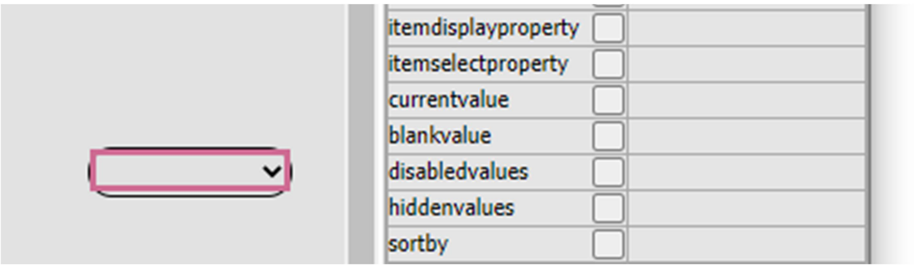
```
  ◢ XNodeCombo#[tcp://172.16.1.95:93]
    ◢ FpStat#0
        ● LivewireActive
        ● LivewireNetwork
        ● Master
        ● Network0Active
        ● Network1Active
        ● PlugPower
        ● PowerOverEthernet
        ● Sync
        ● Temperature
      ● Online
  ▷ ZephyrIPort#[tcp://172.16.1.71:93]
  ▷ ZipOne#[tcp://172.16.4.1:93]
▷ Email
▷ Emulators
▷ Gpio
▷ Memory
```

◉ Simple ○ API

Version 1.7.8.11 Changes

User Panels List Selector

This version adds several additional properties to the list selector component.



itemdisplayproperty	□	
itemselectproperty	□	
currentvalue	□	
blankvalue	□	
disabledvalues	□	
hiddenvalues	□	
sortby	□	

blankvalue

Each item in the list has a display value obtained from the display property and an actual value that gets used when the item is selected obtained from the itemselectproperty. If the current actual value is blank, there may not be an item in the list which matches a blank value. This allows you to define a value to use in the list selection if the current value is blank. For example, when using a virtual destination list, the none route source may return a currentsource value of blank rather than zero. The id value for the none source is zero. By entering 0 in this field, the 0 value will be used when selecting the current item in the list whenever a blank value is returned from the system. Without this setting, the list might display the last known selection rather than none.

disabledvalues

This field can accept a comma delineated list of values which if they exist in the list will display in the drop down list but be disabled and unselectable. Use the value from the itemselect property in this list not the display value. For example if the drop down list was showing the virtual sources of a virtual router, using an entry of:

-3,-1

Would make the Other and Previous sources disabled.

hiddenvalues

This field works exactly the same as disabledvalues except it will hide the entries presented in the comma delineated list.

sortby

This option allows you to sort the items in the drop-down list by the original inbound order, alphabetically by value, or alphabetically by the display name.

--option-background-color

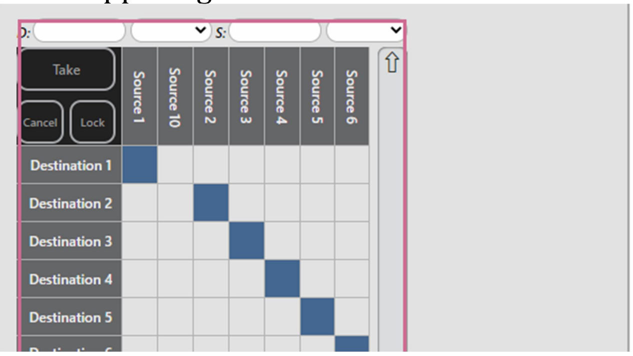
This allows you to define the background color of the drop-down menu that pops up when you try to select a new item.

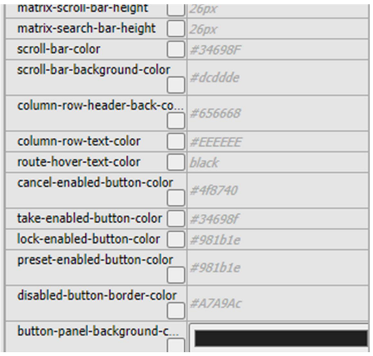
--option-color

This allows you to define the font color of the items in the drop-down menu that pops up when you try to select a new item.

User Panels Router XY Matrix

This version adds a property called button-panel-background-color which allows you to define the background color of the space around the Take and clear buttons in the upper right corner of the matrix:





matrix-scroll-bar-height	□	26px
matrix-search-bar-height	□	26px
scroll-bar-color	□	#34698F
scroll-bar-background-color	□	#dcddde
column-row-header-back-co...	□	#656668
column-row-text-color	□	#EEEEEE
route-hover-text-color	□	black
cancel-enabled-button-color	□	#4f8740
take-enabled-button-color	□	#34698f
lock-enabled-button-color	□	#981b1e
preset-enabled-button-color	□	#981b1e
disabled-button-border-color	□	#A7A9Ac
button-panel-background-c...	□	■

Version 1.7.9.12 Changes

User Panels All Components

Exposed the box-sizing css style for all components which defines whether the border is a part of the overall component dimensions or outside of them. See https://www.w3schools.com/css/css3_box-sizing.asp for details regarding this style. Note that it may not have effect on the complex custom components.

User Panels Label Component

Exposed the text-align and line-height css styles for labels. See https://www.w3schools.com/cssref/pr_text_text-align.ASP and https://www.w3schools.com/cssref/pr_dim_line-height.asp for these properties.

The text-align property will allow you to align the text in the label component to the left, center, or right of the component. The line-height style may be used to center the content vertically.

HP202 ConsoleIp

This version adds an HP202 object with a read/write ConsoleIp property to the HP202 device. This property allows you to read and change which console an HP202 device is connected to. This feature still requires manufacturer testing and verification as we do not have any HP202 devices to test with.

Generic Emulator Tcp Client

This version adds a PingMessage and PingInterval property to Generic Emulators. These properties are primarily for use when the emulator is configured to use the TcpClient connection method. Leaving the PingMessage blank will mean this feature is not used.

When a remote device closes its tcp connection normally, there is a handshake that takes place that tells us the connection has been closed. The emulator will then try to start reestablishing the connection. However, if the remote device loses power or the connection dies in a way that does not allow the handshake to take place, then the DeviceEmulator will not know the connection is dead until either a failed operating system keepalive packet failure (which might take up to an hour or two) or until we attempt to send a message to the remote device. The ping property and ping interval allow you to define a message that gets sent every so often to make sure the connection is still working. These messages should be something that makes sense to the application on the remote end. It also allows us to test not only that the connection is still intact but that the application at the remote end is still responding. Note that it still may take longer than the interval time to detect a failure (often 15 to 60 seconds). Therefore, it does not make sense to set the interval too low.

Version 1.7.13.15 Changes

Overview

Please note that this version is a major update and also represents a feature freeze in advance of a new 1.8 release version. Primarily, this version adds a new dark theme for user panels as well as a number of new components. Please report any issues or bugs you may run into to support. Bugs will still be updated in the 1.7 branch until we make the full release to 1.8. But any new features will go into a 1.9 beta version.

Routers

This version adds consistency to the None source in Virtual and SapProperty Routers. Specifically the CurrentSource property will report 0 instead of an empty string. And the CurrentSourcePath property will report ://. It is important to note that in rare situations this may be a breaking change. If you have a logic flow that uses the None source as a start point, you may need to review to make sure it still matches correctly and does not need to be updated after moving to this software version.

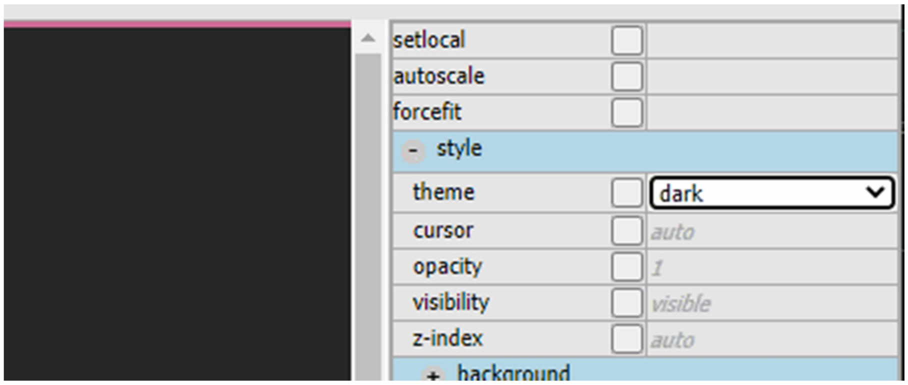
Additionally this version adds the ability to use the None source as a base point in a virtual router's source. This can be useful when you have multiple base points in your source and destination and want to specify how to clear only some of them. This should not be used to completely clear a virtual destination as the normal Virtual None source does that already.

Please note that these changes did require touching a number of places in the routing code so please report any bugs or issues you may encounter.

User Panels

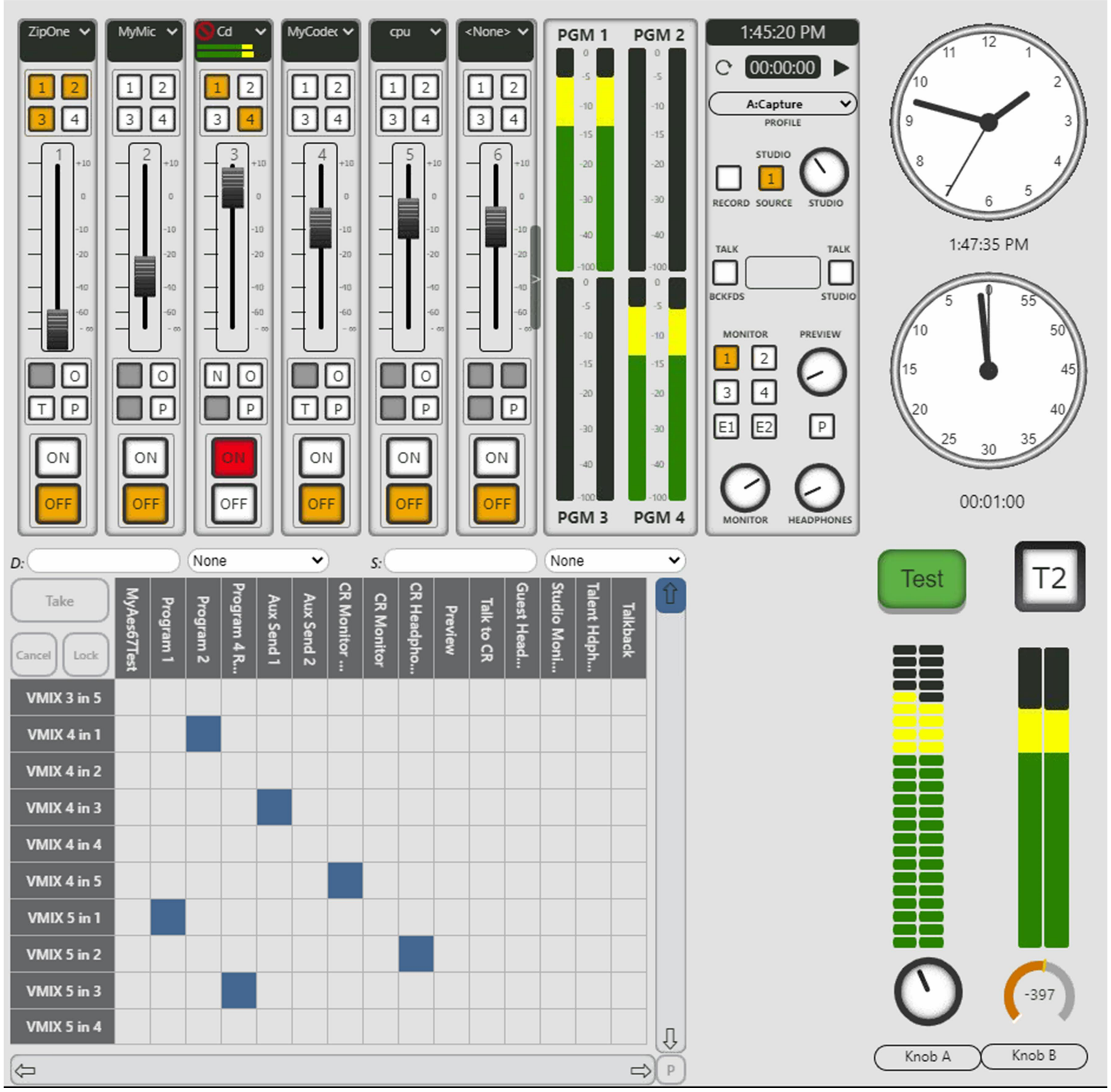
Dark Theme

This version of PathfinderCore PRO introduces a new theme to user panels. This theme may be selected by clicking within the main panel background of a new or existing panel and selecting the theme property from within the style section. There are only two options currently: default and dark.

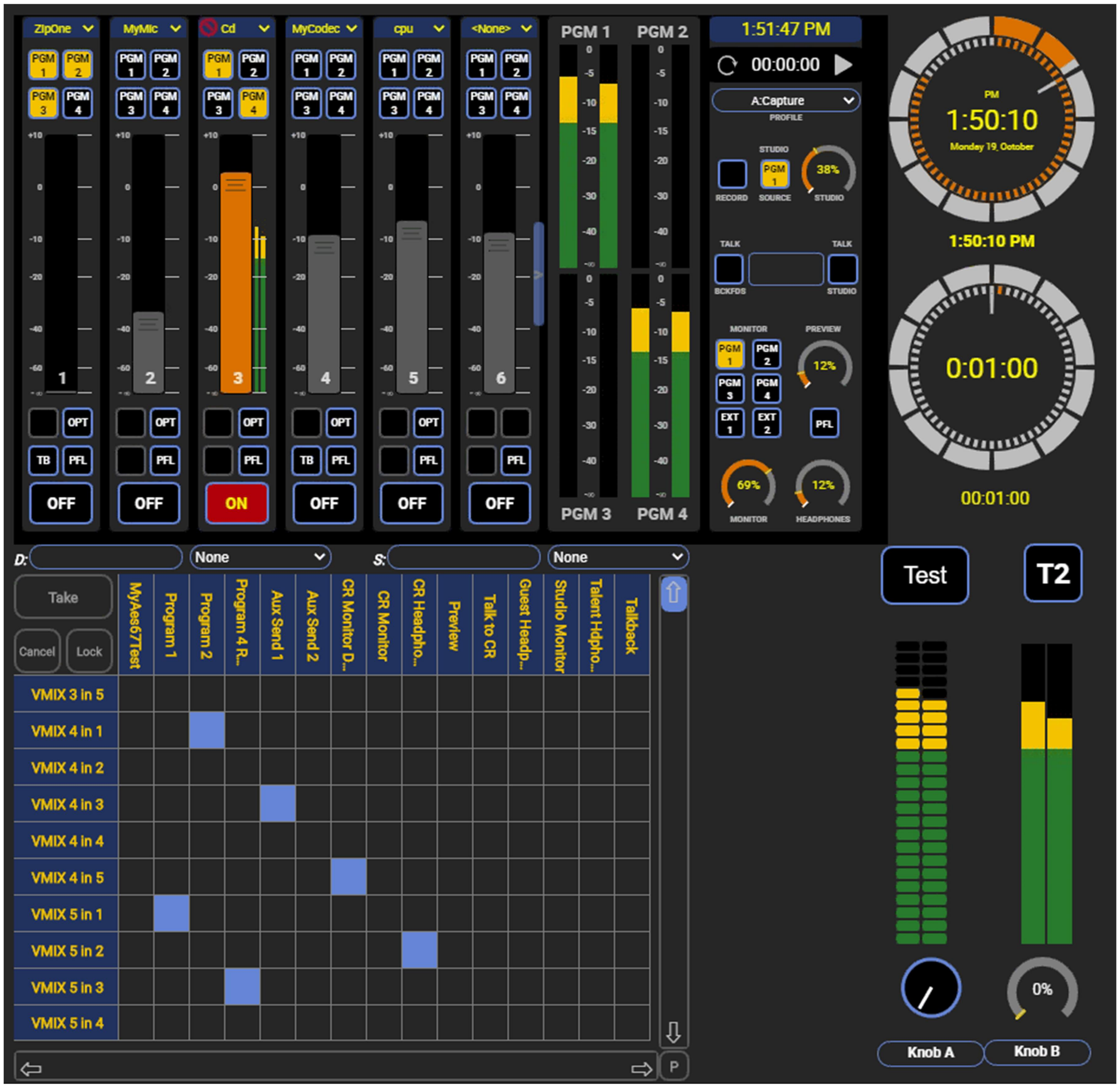


setlocal	□	
autoscale	□	
forcefit	□	
− style		
theme	□	dark
cursor	□	*auto*
opacity	□	*1*
visibility	□	*visible*
z-index	□	*auto*
+ background		

The default theme is the same theme that has been available previously. Please be aware that some theme colors and styles are still being adjusted and so may change as the beta testing continues. To give an example of the difference between the themes see the screenshots below:



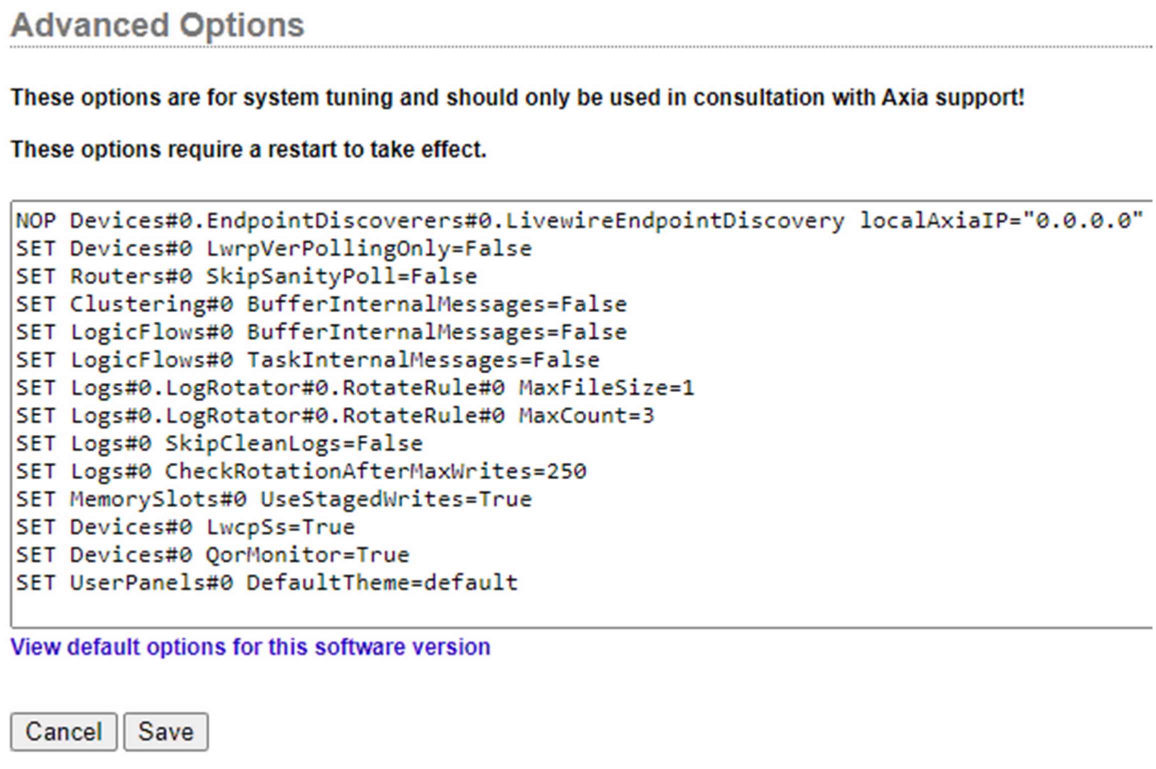
	MyAes67Test	Program 1	Program 2	Program 4 R...	Aux Send 1	Aux Send 2	CR Monitor ...	CR Monitor	CR Headpho...	Preview	Talk to CR	Guest Head...	Studio Moni...	Talent Hdph...	Talkback
VMIX 3 in 5															
VMIX 4 in 1			■												
VMIX 4 in 2															
VMIX 4 in 3					■										
VMIX 4 in 4															
VMIX 4 in 5							■								
VMIX 5 in 1		■													
VMIX 5 in 2									■						
VMIX 5 in 3				■											
VMIX 5 in 4															



	MyAes67Test	Program 1	Program 2	Program 4 R...	Aux Send 1	Aux Send 2	CR Monitor D...	CR Monitor	CR Headpho...	Preview	Talk to CR	Guest Headp...	Studio Monitor	Talent Hdpho...	Talkback
VMIX 3 in 5															
VMIX 4 in 1			■												
VMIX 4 in 2															
VMIX 4 in 3					■										
VMIX 4 in 4															
VMIX 4 in 5							■								
VMIX 5 in 1		■													
VMIX 5 in 2									■						
VMIX 5 in 3				■											
VMIX 5 in 4															

It is important to note that when changing between themes, it often requires a save of the panel in the new theme before the style will be correctly displayed. Additionally, existing panels may require some adjustment when changing to the new theme. In particular, styles which you have overridden with you own settings will retain your choices which may or may not be appropriate to the new style.

If you decide you prefer one theme to the other, you can set an Advanced option to make it the default theme for new panels.



Advanced Options

These options are for system tuning and should only be used in consultation with Axia support!

These options require a restart to take effect.

```
NOP Devices#0.EndpointDiscoverers#0.LivewireEndpointDiscovery localAxiaIP="0.0.0.0"
SET Devices#0 LwrpVerPollingOnly=False
SET Routers#0 SkipSanityPoll=False
SET Clustering#0 BufferInternalMessages=False
SET LogicFlows#0 BufferInternalMessages=False
SET LogicFlows#0 TaskInternalMessages=False
SET Logs#0.LogRotator#0.RotateRule#0 MaxFileSize=1
SET Logs#0.LogRotator#0.RotateRule#0 MaxCount=3
SET Logs#0 SkipCleanLogs=False
SET Logs#0 CheckRotationAfterMaxWrites=250
SET MemorySlots#0 UseStagedWrites=True
SET Devices#0 LwcpSs=True
SET Devices#0 QorMonitor=True
SET UserPanels#0 DefaultTheme=default
```

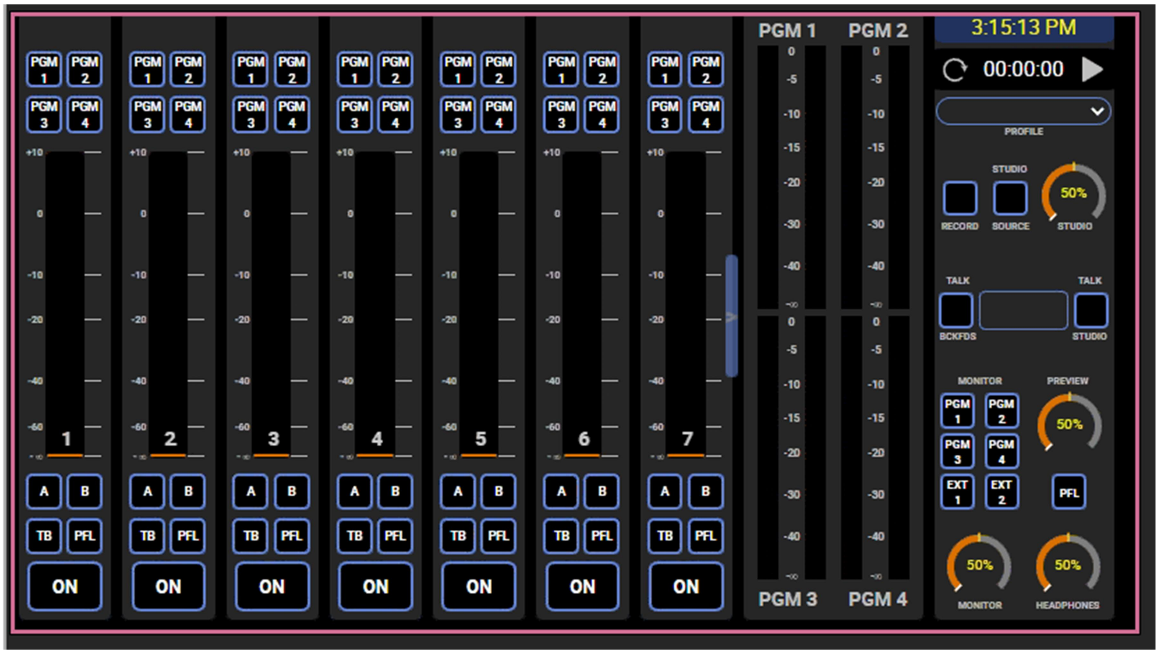
View default options for this software version

The only two options that can be used for the DefaultTheme advanced setting are default and dark. Setting this option to any other setting may prevent you from creating new panels until it is fixed. A reboot is required before any advanced setting takes effect.

It is also important to note that the theme just presents you with the default styling of the components. All properties for customizing that style to your own liking and design are still available in both themes.

Console Component

There are a variety of new or changed components that will be mentioned throughout the rest of this document that are designed to allow you to layout and design your own console control. However this new component will accomplish all of this with a single component as long as you are happy with the styles and layout this component presents.

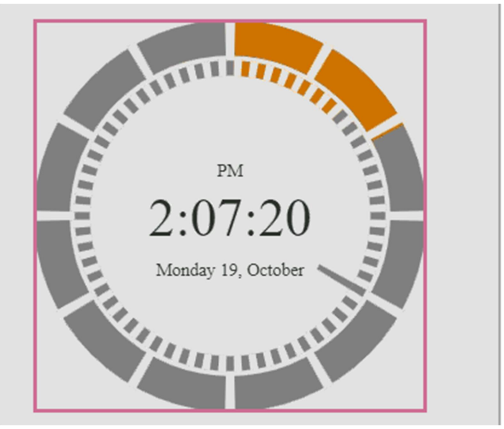


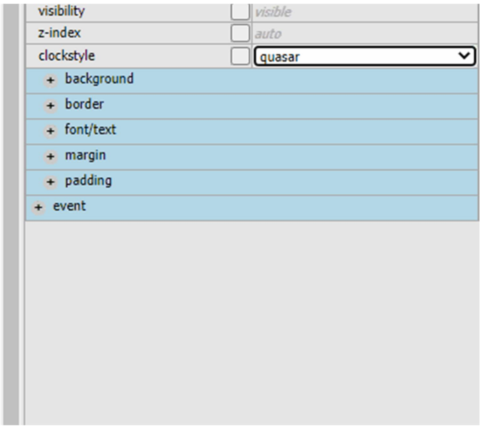
After dropping this component onto a panel, use the control property to select from a list of consoles. When the panel with the component executes, it will automatically display and layout the controls according to the type of console it has been attached to. Increasing or decreasing the height of the component will adjust the height and size of the components within the console. Changing the width of the component will present or hide paging buttons for the fader list depending on how many faders the console has. The fader count is not correctly displayed in the designer and is fixed to 8. The fader list is only populated with the actual fader count when the console is initialized in a running panel.

Important Note: The Fusion and Powerstation require version 3.2.1.28 of the fusion software in order to properly use the monitoring section with these consoles.

Analog Clock and Analog Countdown styles

The screenshots of the dark theme display a new style to clocks. While the traditional style is the default in the default theme, and the new quasar style clock is the default in the dark theme, this is an option within the clock component. Under the style section in the properties of the analog clock and analog countdown clock is a property called clockstyle. The available options are classic and quasar. So this style may also be selected in the default theme:




visibility □ visible
z-index □ auto
clockstyle □ quasar
+ background
+ border
+ font/text
+ margin
+ padding
+ event

The color palette in the default theme is designed for the default theme. In addition, a variety of properties have been added for use when the clock is in the quasar style in either theme so that you can adjust the colors to whatever you desire.

- Under background
    - quasar-clock-hours-color: changes the color of the background outside circle.
    - quasar-clock-elapsed-hours-color: changes the color of the elapsed hours in the outside circle.
    - quasar-clock-minutes-color: changes the color of the background inside circle.
    - quasar-clock-elapsed-minutes-color: changes the color of the elapsed minutes in the inside circle.
    - quasar-clock-marker-color: changes the color of the gaps between the mark points in the hours and minutes circle.
    - quasar-clock-second-hand-color: changes the color of the second hand.
- Under font/text
    - quasar-clock-ampm-color: changes the color of the am/pm text in the clock.
    - quasar-clock-time-color: changes the color of the time text in the clock.
    - quasar-clock-date-color: changes the color of the date text in the clock.

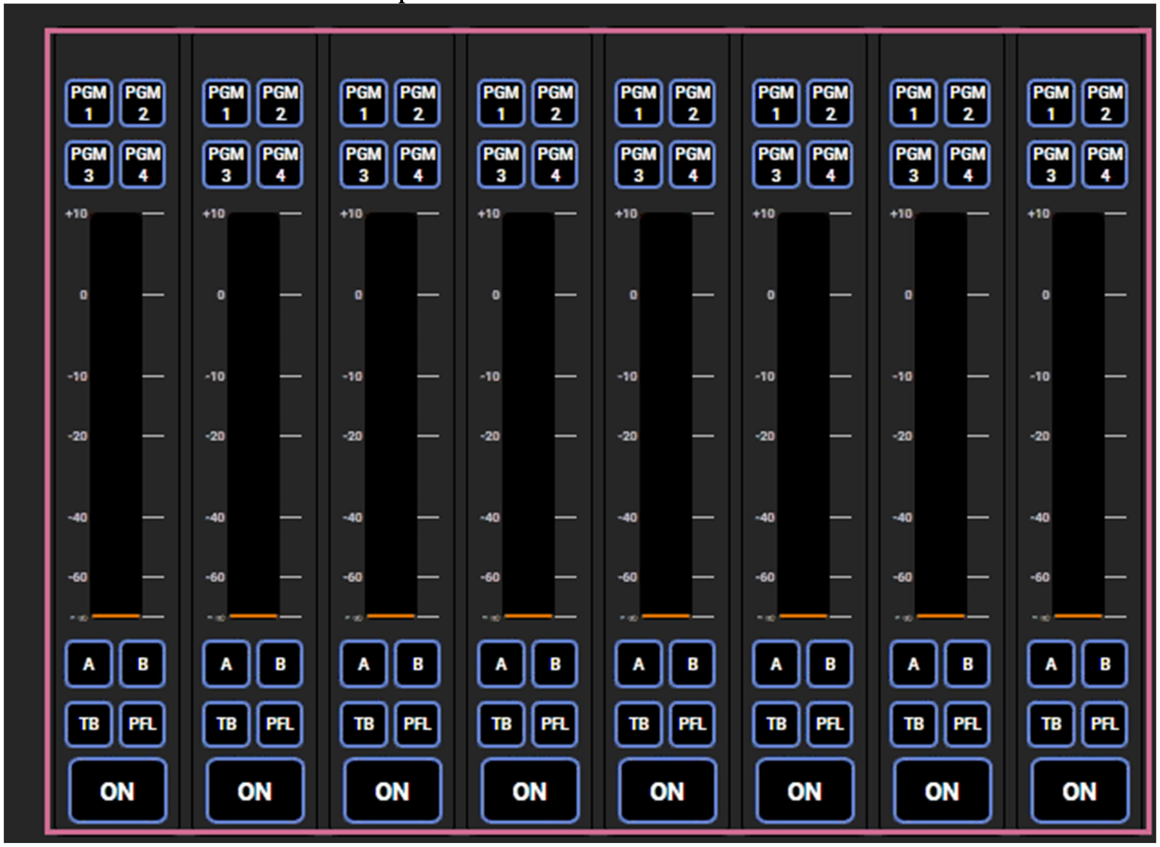
Console Fader Changes

The console fader has had some adjustments to its intelligence such that if you use the control property to select the fader to map to, it will figure out the correct audio io to use for metering. This uses new features in the metering service where metering for Qor/iQx consoles will use the Lwcpss protocol for metering under the hood. Additionally, when mapped to a Qor/iQx console there will be an additional option button. This will raise the menu-selected and menu-unselected event. The menu-enabled property may be disabled to prevent this button from being displayed or used. And the menu-indicator property may be used to light or unlight the button. While this button can be used for any purpose you like, its intent is to be used with the new eq/dynamics section as described later in this document.

There is also now a meter-location option which moves the confidence meter from the top of the fader to a vertical meter to the right hand side of the fader.

Console Fader Dock

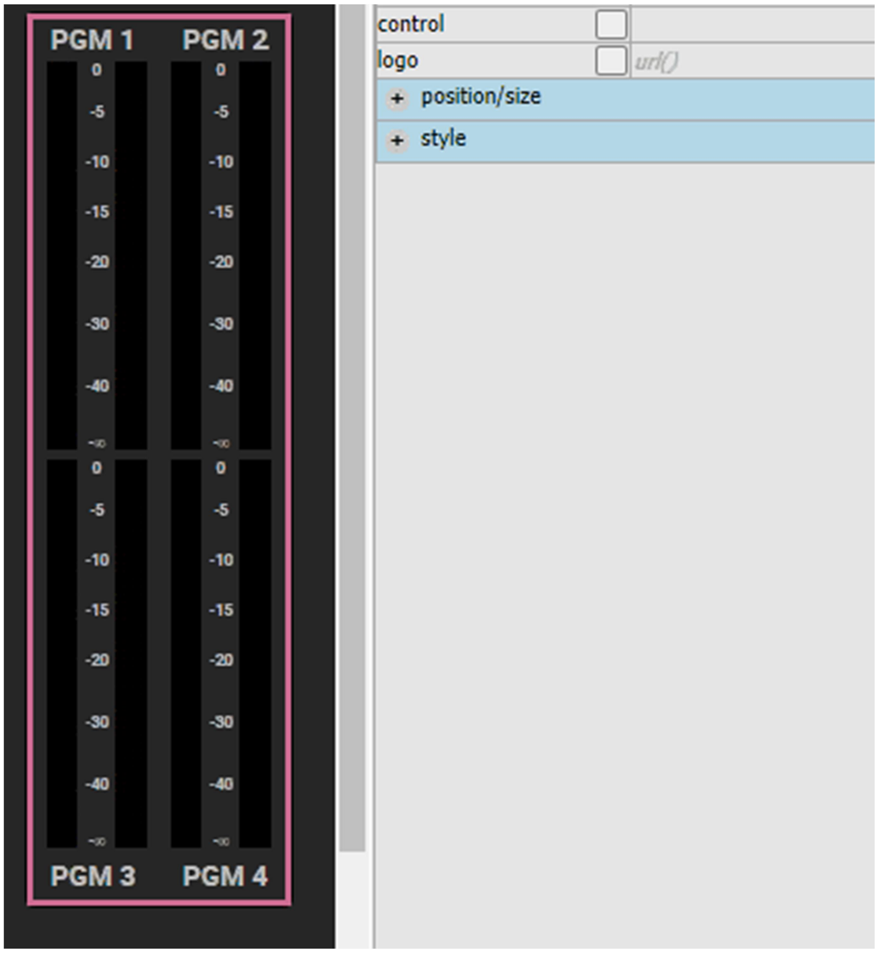
This version adds a new component called Console Fader Dock.



This component will present a list of consoles in the control property in the designer. When executing the panel connected to a console, the dock will auto-fill with all of the faders the console has. It will present page left and right buttons automatically if there are more faders than can be displayed in width designed in the console component.

Console Meters

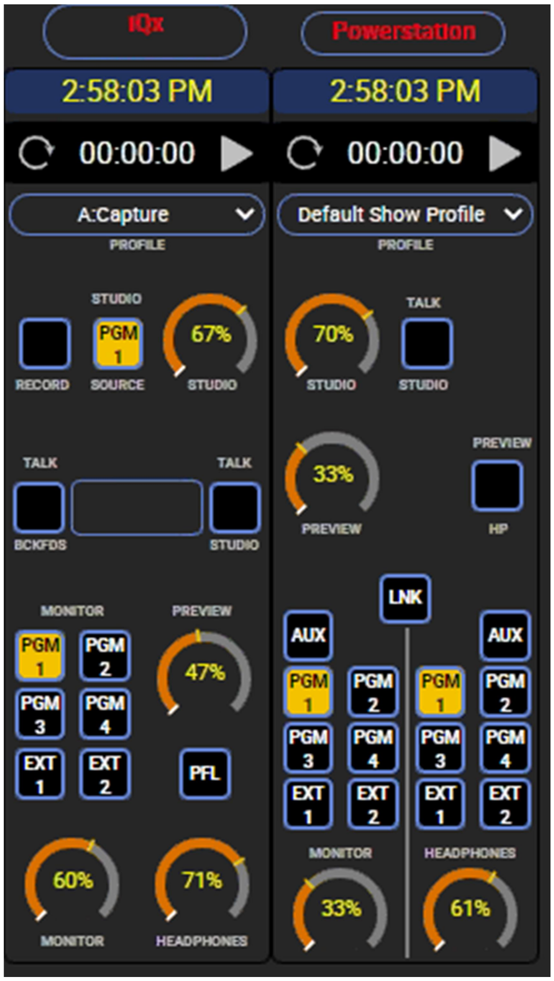
This version adds a new component called Console Meters which will present the program buss 1 through 4 meters.



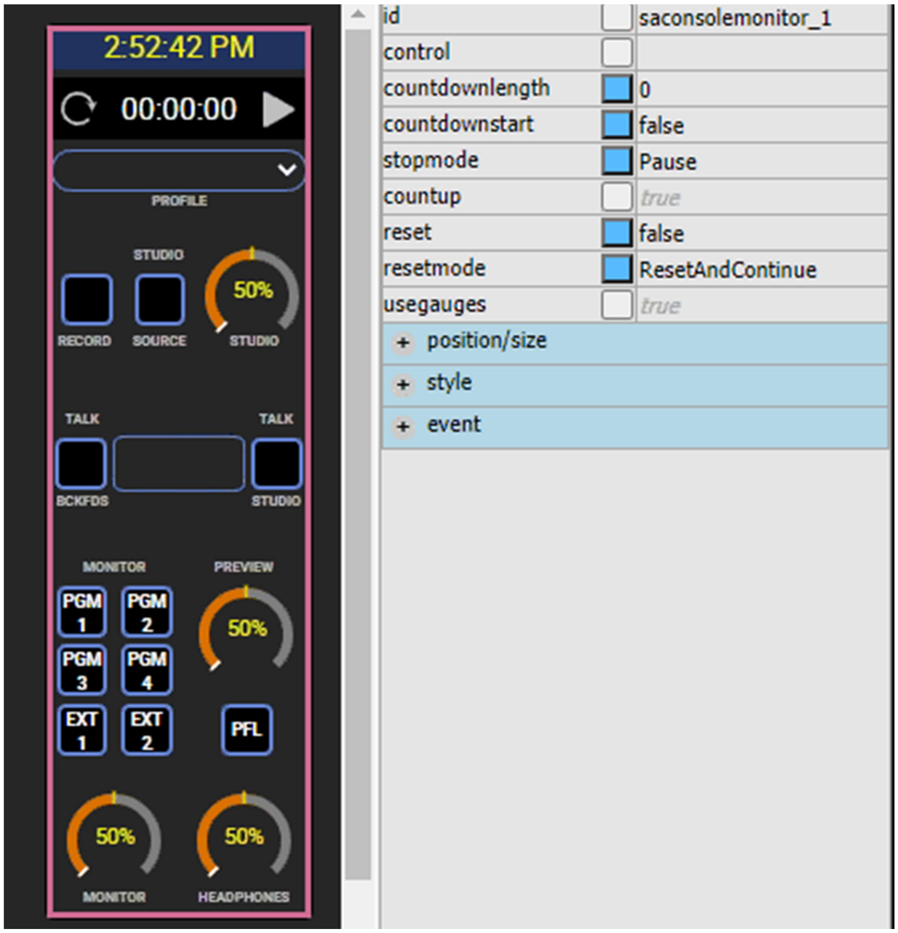
It also will present a list of consoles through the control property in the designer that will allow you to map the control to a specific console. Like the updated console fader, it will automatically figure out the correct metering paths (lwrp/lwcpss) from the console it is attached to it. In addition this module has a logo property which will allow you to upload a logo image which will sit between the top and bottom meters. The size ratio of the logo must be designed in order to display correctly. The height should be approximately one fifth of the width. The example svg we have used had dimensions of 6774x1452 and is then scaled up and down accordingly. This may take some experimentation to get correct.

Console Monitor Section

The Qor/iQx monitor section component has been removed from the control palette but will still work in existing panels. Instead a new intelligent monitor section component exists. Like the console fader, this component will alter its layout and display depending on the type of console it is pointed at. Under the hood it still uses the Qor/iQx monitor section control and displays it when the console is pointed at one of those types of consoles. When pointed at a Quasar or Fusion or Powerstation the components will present different options relative to that consoles capabilities:



Again use the control property to select from a list of consoles in the system.

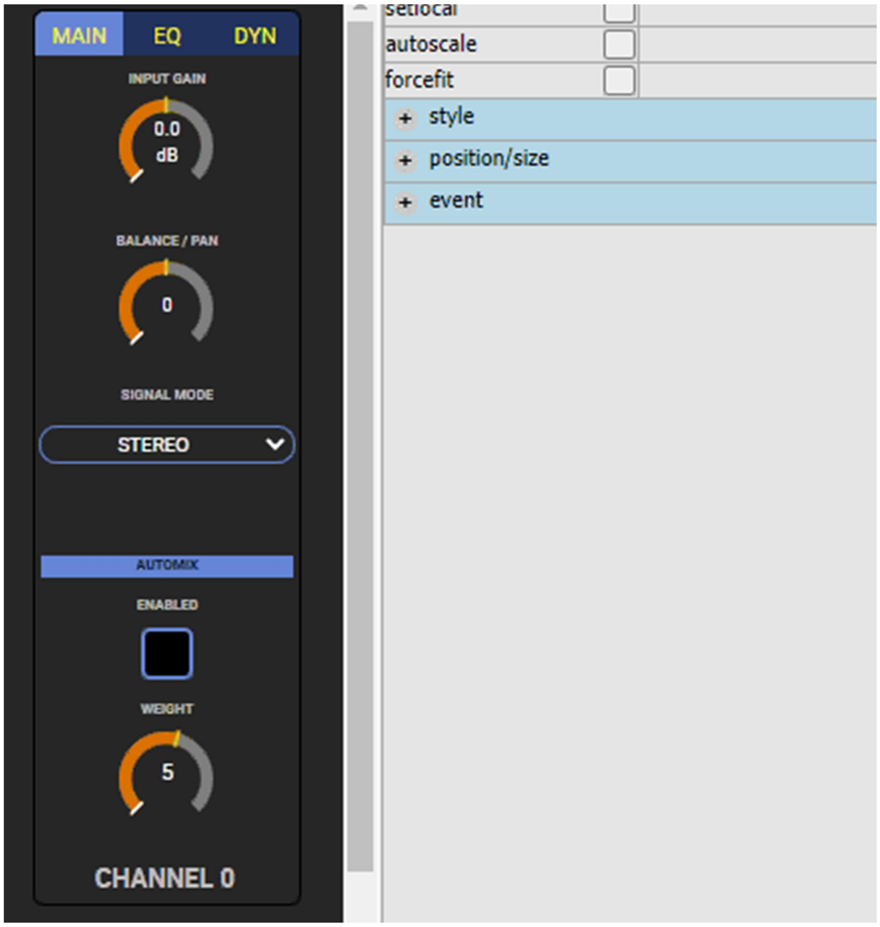


id	☐	saconsolemonitor_1
control	☐	
countdownlength	■	0
countdownstart	■	false
stopmode	■	Pause
countup	☐	*true*
reset	■	false
resetmode	■	ResetAndContinue
usegauges	☐	*true*
+ position/size		
+ style		
+ event		

Important Note: The Fusion and Powerstation require version 3.2.1.28 of the Fusion software in order to properly use the monitoring section with these consoles.

iQ Fader Processing

The iQ fader processing is a new component that access the eq and dynamics section of Qor and iQx faders. In order to use it, select the console from the control property and then assign a number 1 through 24 to the fader-number property to use it with a specific fader. This assignment can be automated using the menu button in the console fader object. Turn the binding of the fader-number property on. Then within each fader in the panel, use the menu assign and menu unassign event to assign the correct fader number to the component. A fader number of 0 can be used to make the component not have any effect. It is a good idea to hook the menu indicator to the eq/dynamics section fader-assign property as well so the menu button will only light if the current fader is selected in the component.



INPUT GAIN 0.0 dB

BALANCE / PAN 0

SIGNAL MODE STEREO

ENABLED □

WEIGHT 5

CHANNEL 0

setlocal □

autoscale □

forcefit □