TECHNICAL CERTIFICATION PROGRAM

LAB EXERCISES GUIDE

ALL WORK WILL BE COMPLETED THROUGH THE REMOTE DESKTOP
AS YOU GO THROUGH EACH LAB, READ THE INSTRUCTIONS CAREFULLY.
MAKE SURE YOU ARE IN THE CORRECT STATION (JACE vs. PC)
BEFORE STARTING EACH LAB.
DO NOT SKIP ANY STEPS.

TRIDIUM

Confidentiality Notice

Trademark Notice

Copyright and Patent Notice

Credits

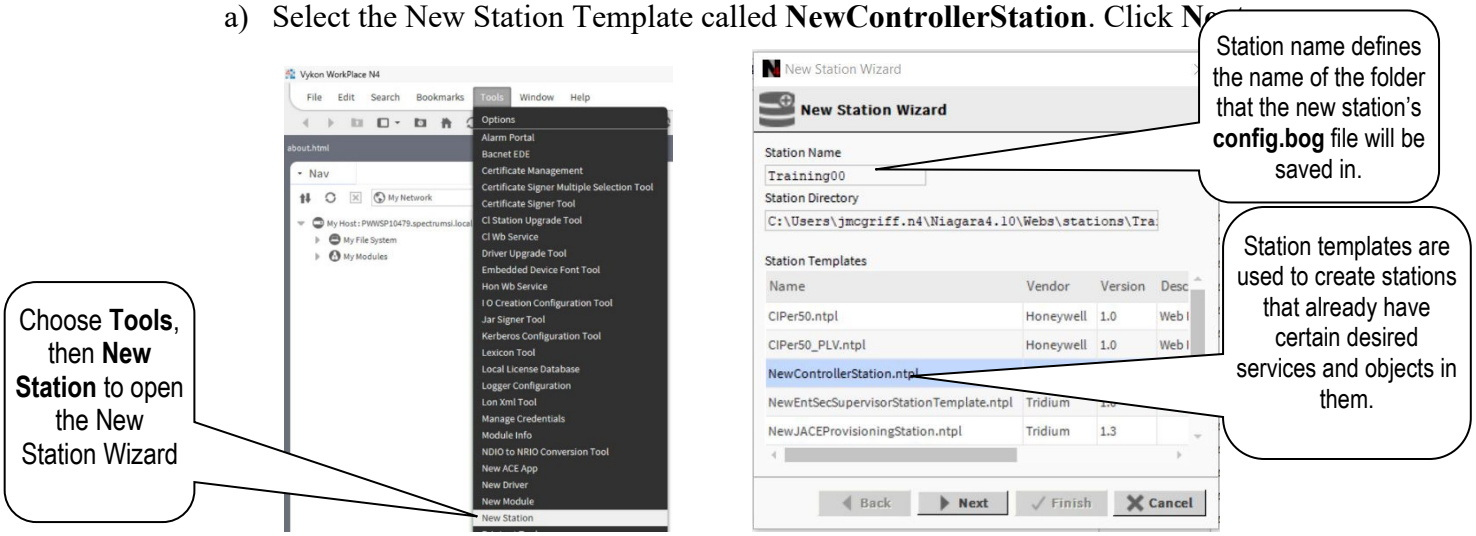
Special thanks to Charles Lunsford whose eye for detail and commitment to clear and easily understood instructions has greatly contributed to each page.

Table of Contents

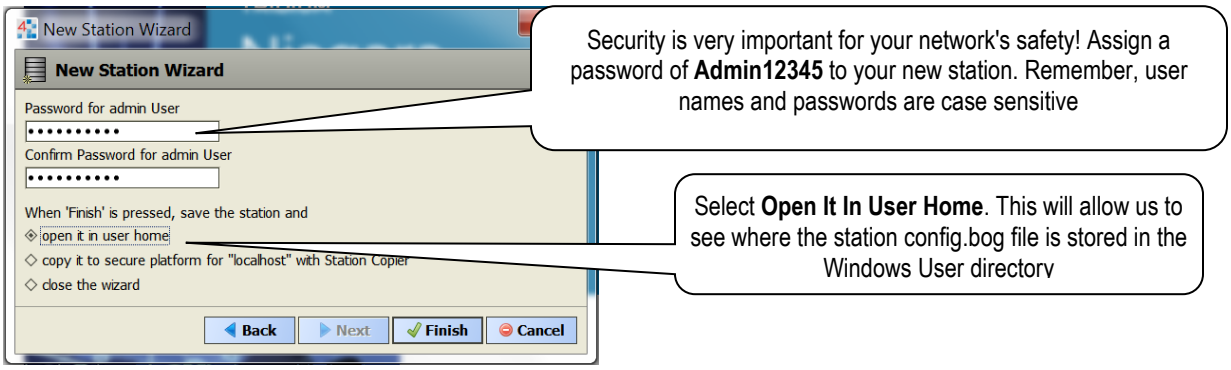
LAB #1: Creating a Station	15 min
Description:	In this lab you will create a new Controller station and run that station on the Local Host "***REMOTE DESKTOP PC"*** so that you can begin to design a custom Niagara solution.
Learning Objectives: Upon completion of this lab exercise, you will be able to:	1. Create a new station using the wizard. 2. Assign a Strong Password. 3. Understand the differences between user home and platform daemon home locations. 4. Transfer stations between user home and daemon home. 5. Start a station using the Application Director tool. 6. Open a secure Station connection. 7. Practice saving the Station and backing up the Platform.

1. Check the time zone and date / time settings on your class PC. Adjust the settings as needed to represent the location of the class you are attending.
2. On the desktop double click the **Install Platform Daemon** icon. Click yes to allow the app to run.
3. Open the **N4 Workbench** by double-clicking the **N4** 4.10 **Workbench** shortcut on your desktop:

4. In the Workbench, under the **Tools** menu, select **New Station**. Name your station **TrainingXX_Original**, where the XX represents your workstation number.
   a) Select the New Station Template called **NewControllerStation**. Click **Ne**



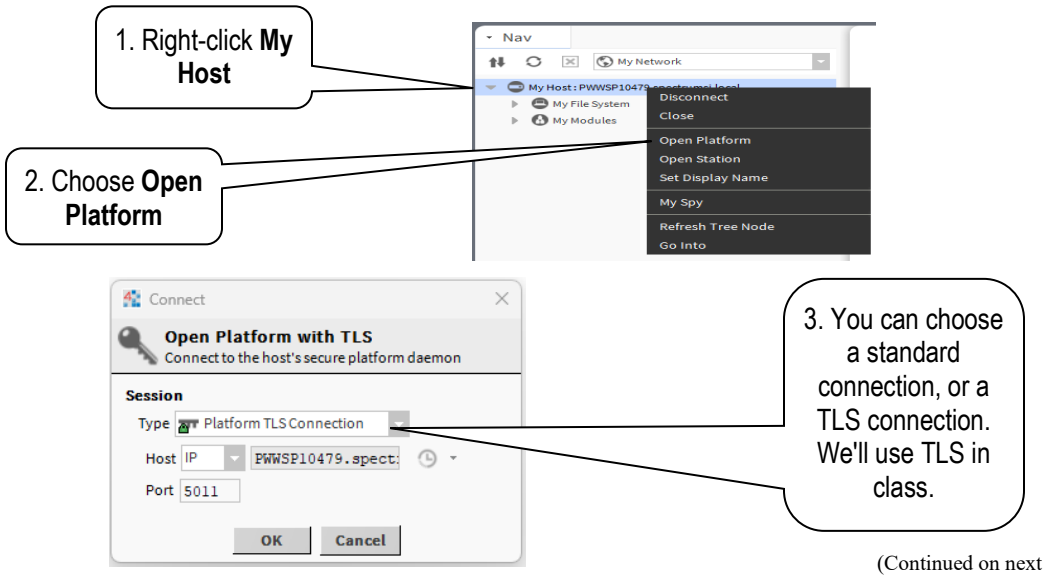
5. Security is VERY important in your station! Assign a password of **Admin12345** for the **admin** user. We want a strong password out in the field but a simple one for in class. The **Niagara 4** default is for a minimum of 10 characters, at least one of which must be a number, and one each of upper- and lower-case letters. Leave all port settings at the defaults.
   a) Select **Open it in User Home** when the **Finish** button is pressed.



Note: The station files are automatically placed into the **User Home** directory, which is unique to each Windows user defined on the PC. This is the **User** directory in Windows File Explorer for the user signed on to Windows when the station is created. Putting the station files here keeps station files more secure. But to start the station on the PC, the station must be copied to the **Platform Daemon User Home** directory (C:\ProgramData\Niagara4.10\vykon\stations) using the Station Copier tool. When the station runs on the computer, it is from here that it starts, and it is to here that it is saved.

6. When the station is created, note the display in the **Nav SideBar**. The station can be edited from this view in a not-running, or "off-line" mode (remember to click the **Save Bog** button to save changes to a station that is edited when it is not-running).
7. Collapse the **UserHome** view in the Nav SideBar.
8. Connect to the platform running on the PC
   a) Using the Workbench, right-click **My Host** in the Nav SideBar. **Open Platform**.
   b) Note the **Type** setting is **Platform TLS Connection.** Click **OK** to move to proceed.

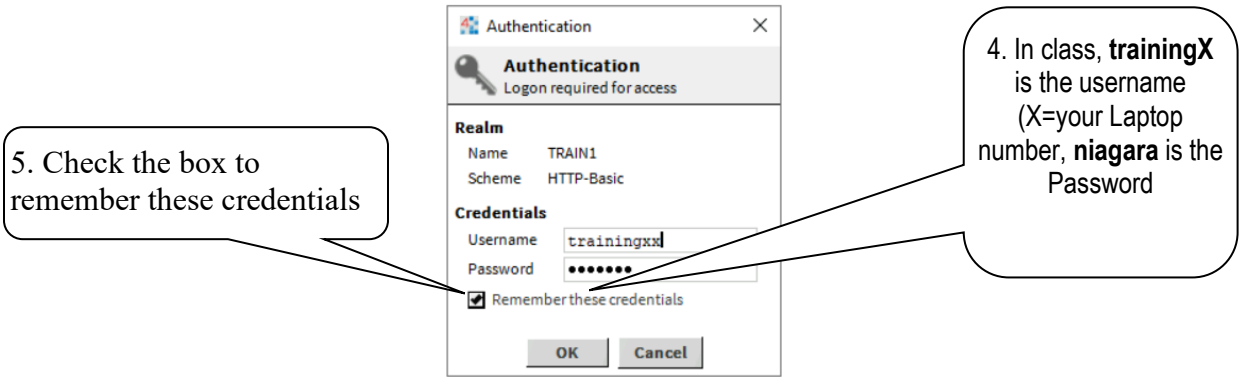
   Note: TLS stands for Transport Layer Security, which is one implementation of SSL, or Secure Socket Layer. Using TLS will encrypt the communication between the Workbench and the platform. It requires security certificates to be used which can be an issue to deal with, but it makes communication more secure.



(Continued on next page)

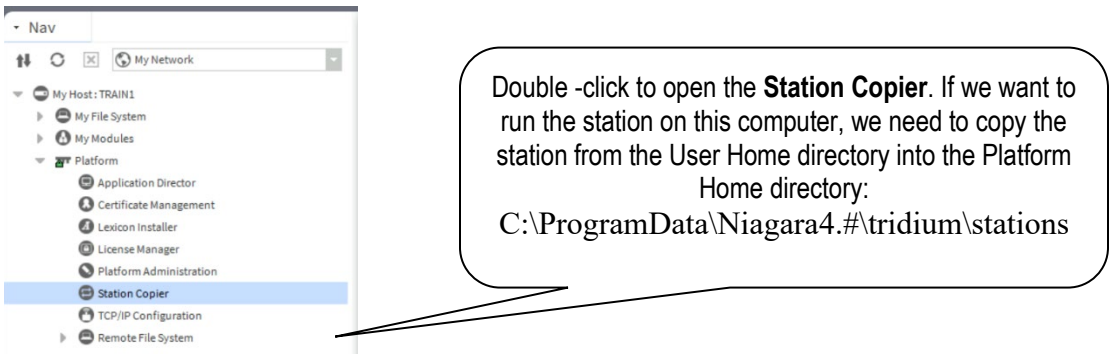
c) You may see a popup that asks you to evaluate a security certificate. Choose **Accept**.

d) The PC platform username in class is **trainingX**; the password is **niagara**. Check the box to **Remember these Credentials** if it is not already checked. (Remember that these credentials represent the Windows username and password). Click **OK** to make the platform connection.



9. Once the connection to the platform is made, the platform tool set appears.

10. In the Nav SideBar or the View Pane, double-click the **Station Copier** tool.

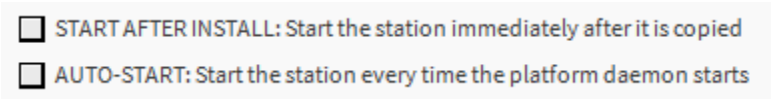


11. In the station copier tool, the new station you created is listed in the left column. Stations in this directory reside in the protected **User Home** directory. Think of this like as storage area for all your stations. We must copy the station from the left to the right column in this case. Note: **Platform Daemon User Home** is considered a remote host, and its running station is accessible via a Java virtual machine.

12. Select the new station in the left column. Click the **copy** button. 

a) As the station copy begins, you will be offered the opportunity to rename the station. Rename your station trainingxx "xx is your station number". Click the **Next** button.

b) You will see two check boxes in the next window.

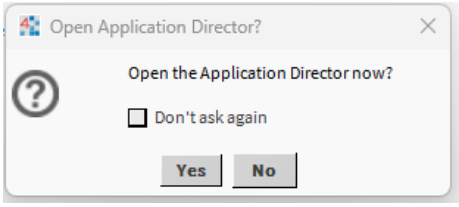


Deselect these two check boxes. Click the **Next** button.

c) The next window verifies that you are copying from local computer (User Home) to localhost (Daemon Home) and click the **Finish** button to start the transfer.

d) Once the station copy is complete, close the **Station Copier** pop-up by clicking the **Close** button.

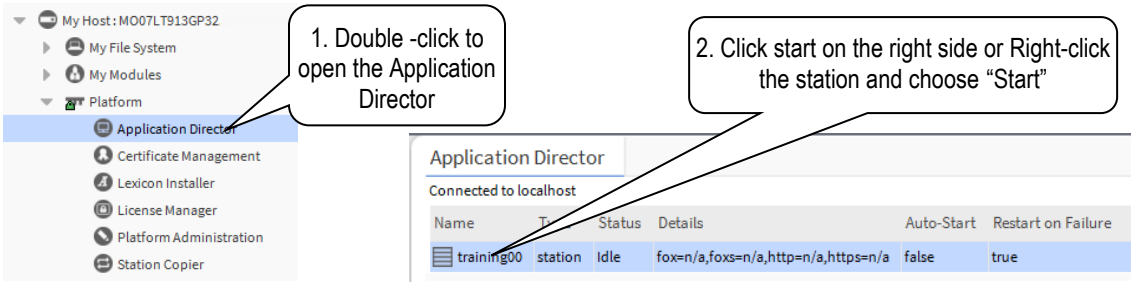
e) Select no in the next dialog box.



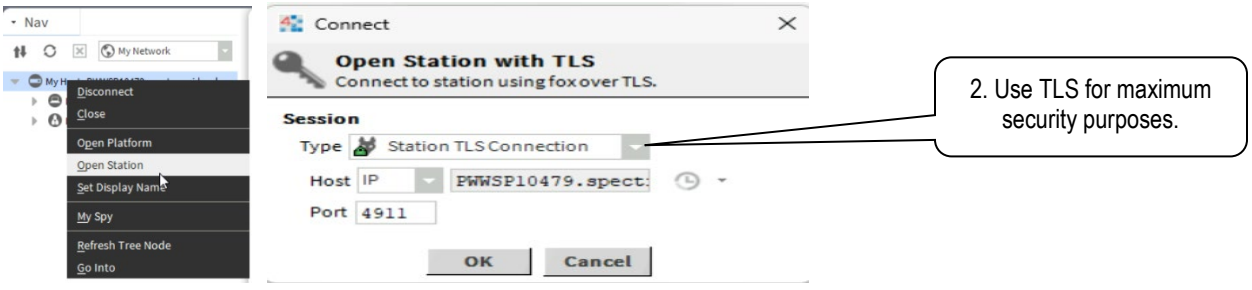
13. Next, we will start the station.

a) In the Nav Tree on the left-hand side expand Platform. Double-click on Application Director to open it..

b) The Application Director should now be open in the View Pane on the right. Select the trainingXX station then click the Start button on the right-hand side. (Alternatively, you may also right-click on the station and select Start from the pop-up menu.)

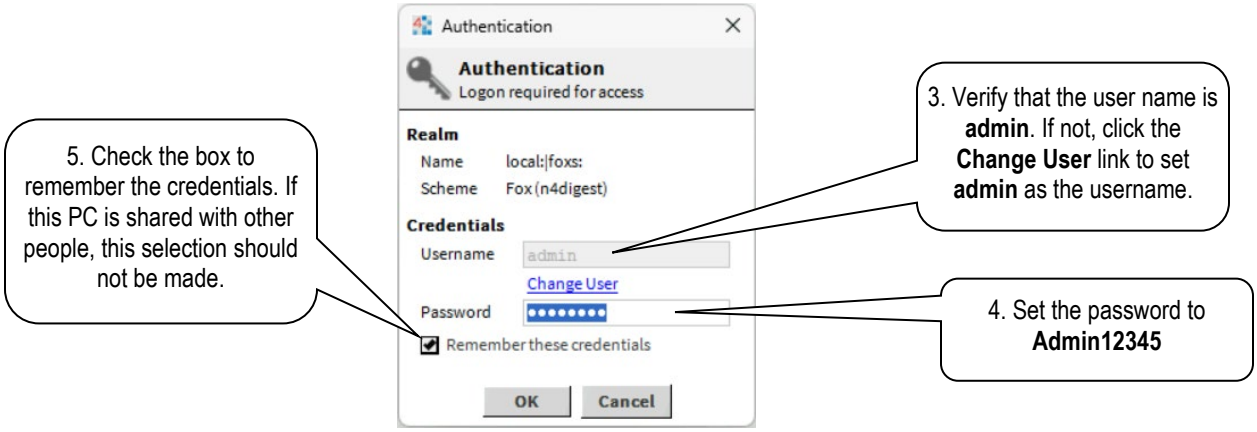


14. Once the station is running, right-click **My Host** in the Nav SideBar and choose **Open Station**. Set the **Type** setting to **Station TLS Connection**.



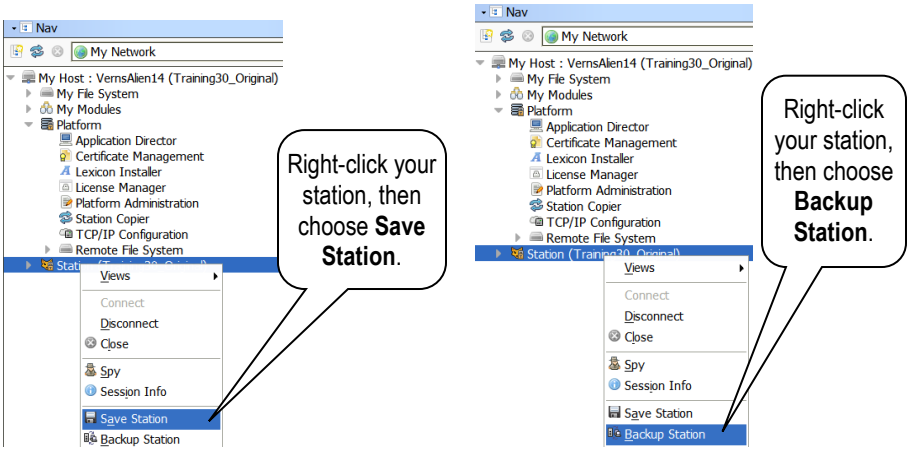
1. Right-click **My Host** and choose **Open Station**

Note: You may encounter a challenge to a server security certificate when opening a station connection this first time. Click **OK** to accept the Tridium self-signed certificate as necessary. The significance of this action will be discussed later in the course.



15. Use the station username **admin** and the password **Admin12345**.

16. Once you have established a connection to the station, **save** your station by right-clicking the station and choosing **Save Station**. [The station is running in RAM (Random Access Memory) and the **Save** action forces the running station to write its database back to the config.bog file (this is the static station database file) in the PC's persistent file space. *In the process a backup of the previous version of the config.bog file is often also created.*]



17. **Backup** your station by right clicking the station and choosing **Backup Station**. [*A Backup of a platform or a station is saved as a single distribution file on your engineering PC.*

18. (End of lab, stop here!)

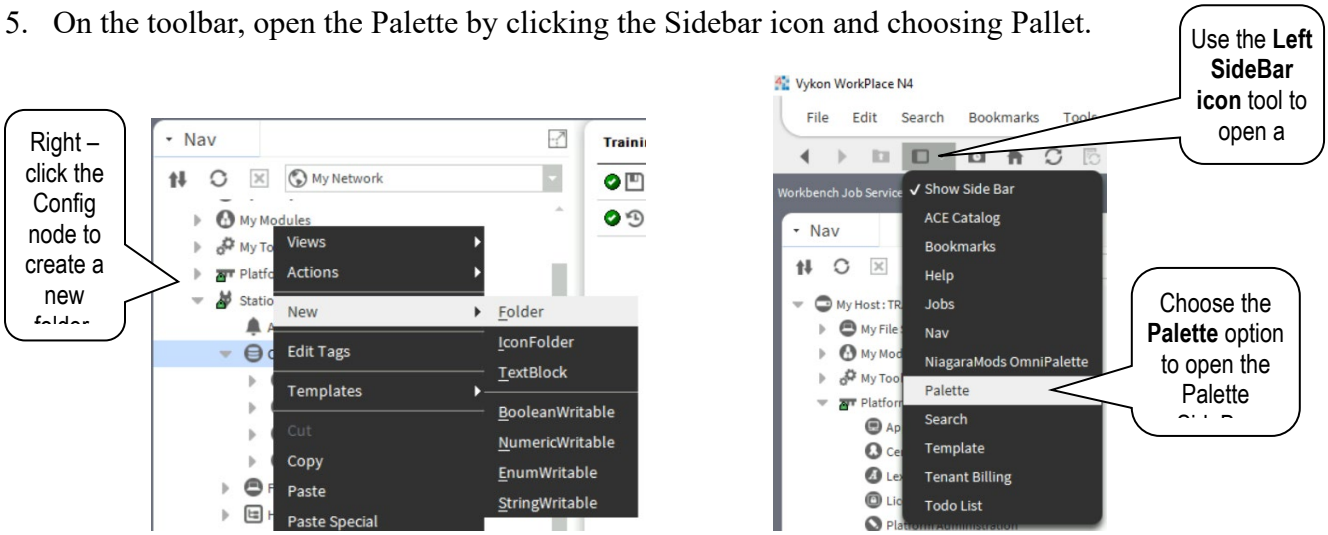
Best Practice: After you have accomplished any significant amount of work on a Niagara solution (especially on a customer's system) you should always get **both** a station copy and a backup copy.

This page is intentionally left blank!

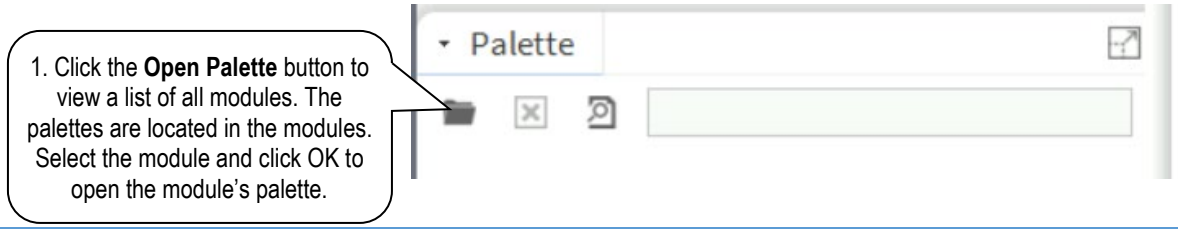
LAB #2: Hot Water Pump Control, Part 1	25 min
Description:	In this lab you will begin to create control logic using the Niagara workbench.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Use the components in the Niagara Common Object Model as well as components found in various module palettes. 2. Open the various views of Niagara objects and set up basic configuration properties. 3. Modify the presentation of an object's standard values. 4. Set an object's default value using the Action menu. 5. Override a Niagara object using the Action menu.

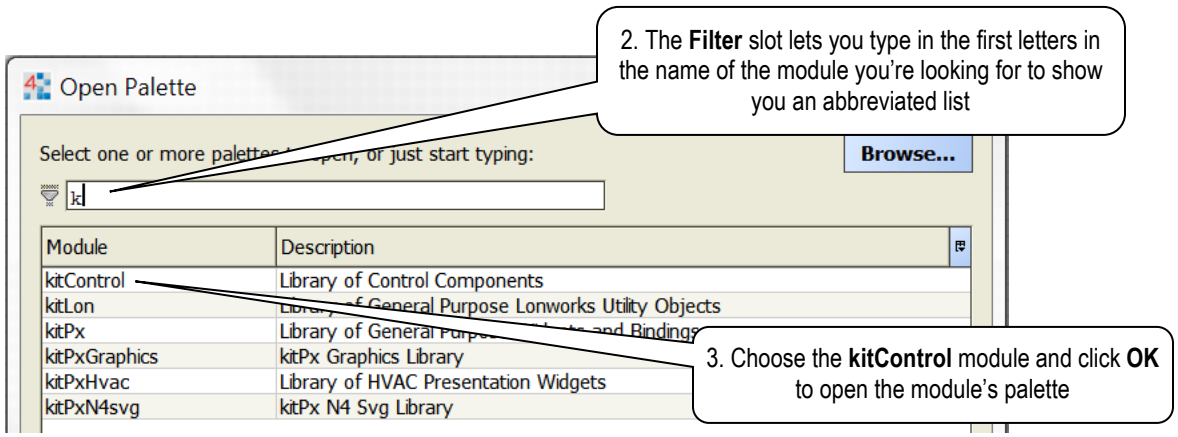
For this lab procedure, use the trainingXX station running on your PC. Italic text in brackets represents menu selections you can use to complete the procedure.

1. Now that the station is running on the PC, expand it in the Nav Tree, then right-click on the Config object and choose New, then Folder from the popup menu. Name the new folder Control_Logic.
2. In the Nav Tree, expand Config and locate the new folder. Right-click the Control_Logic folder, choose New, then Folder. Name the new folder "HeatingSystem"
3. Expand Control_Logic, then Right-click on the HeatingSystem folder, choose New, then Folder. Name the new folder "PumpControl"
4. Expand the HeatingSystem folder, then double-click the PumpControl folder to open the wire sheet.
5. On the toolbar, open the Palette by clicking the Sidebar icon and choosing Pallet.



6. Open the **kitControl** palette [*click the **Open Palette** button on the Palette SideBar*].

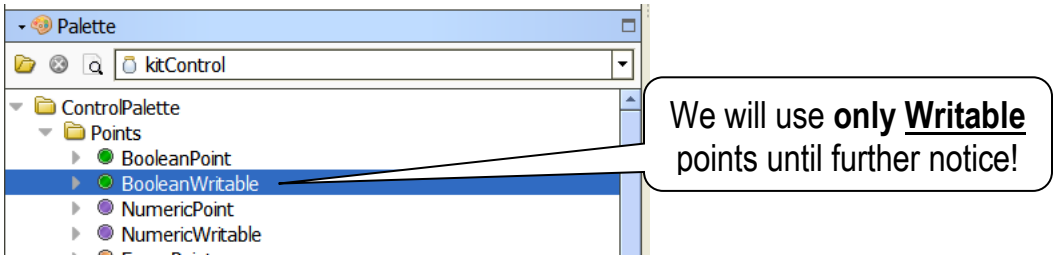




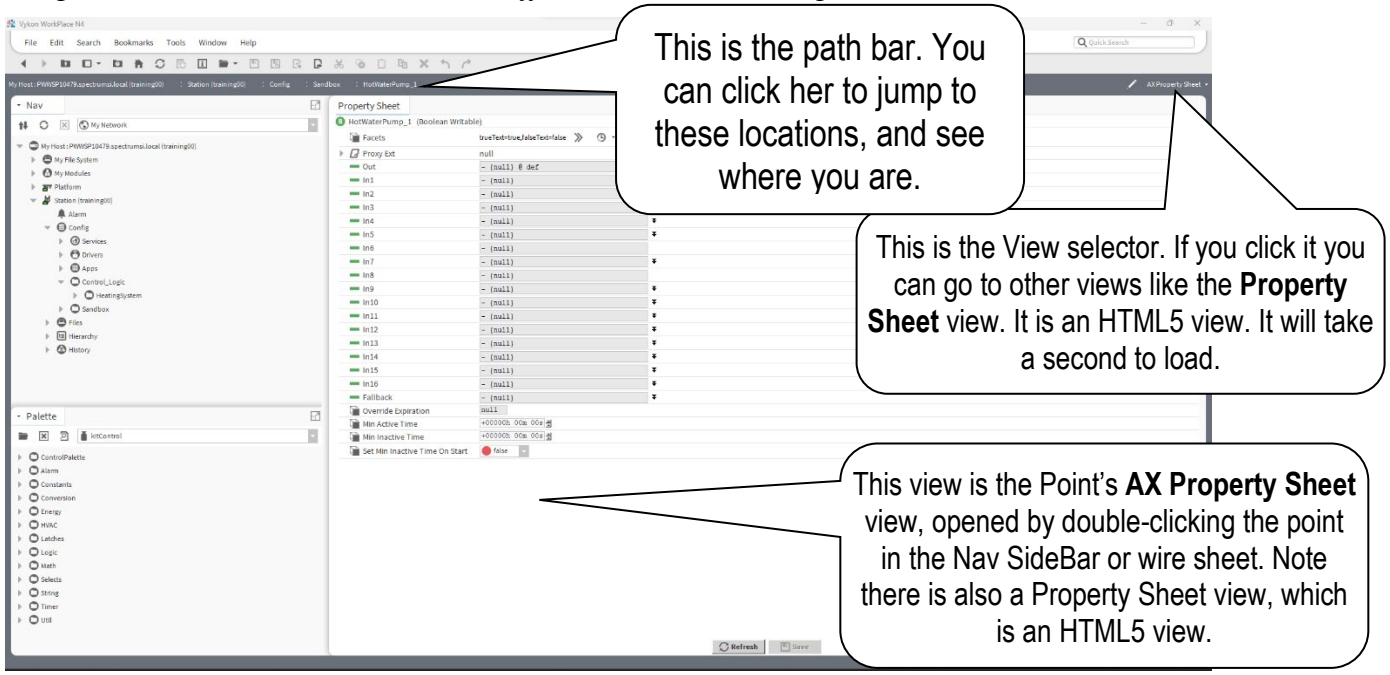
7. Expand the view of the **ControlPalette** folder.

   a) In the ControlPalette's **Points** folder, select a **BooleanWritable** point.

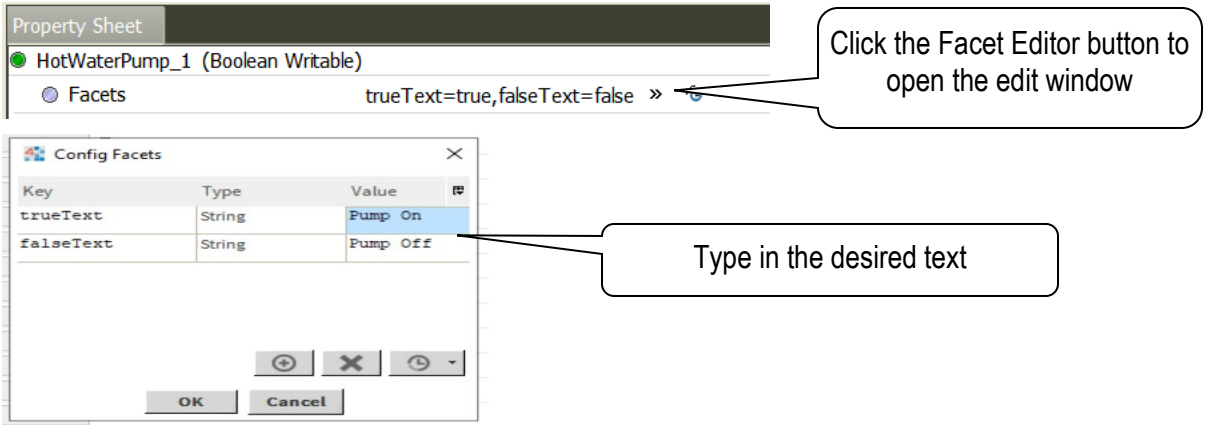
   b) Drag the point to the wire sheet and drop it. As you add the point to the wire sheet, rename the point to **HotWaterPump_1**.



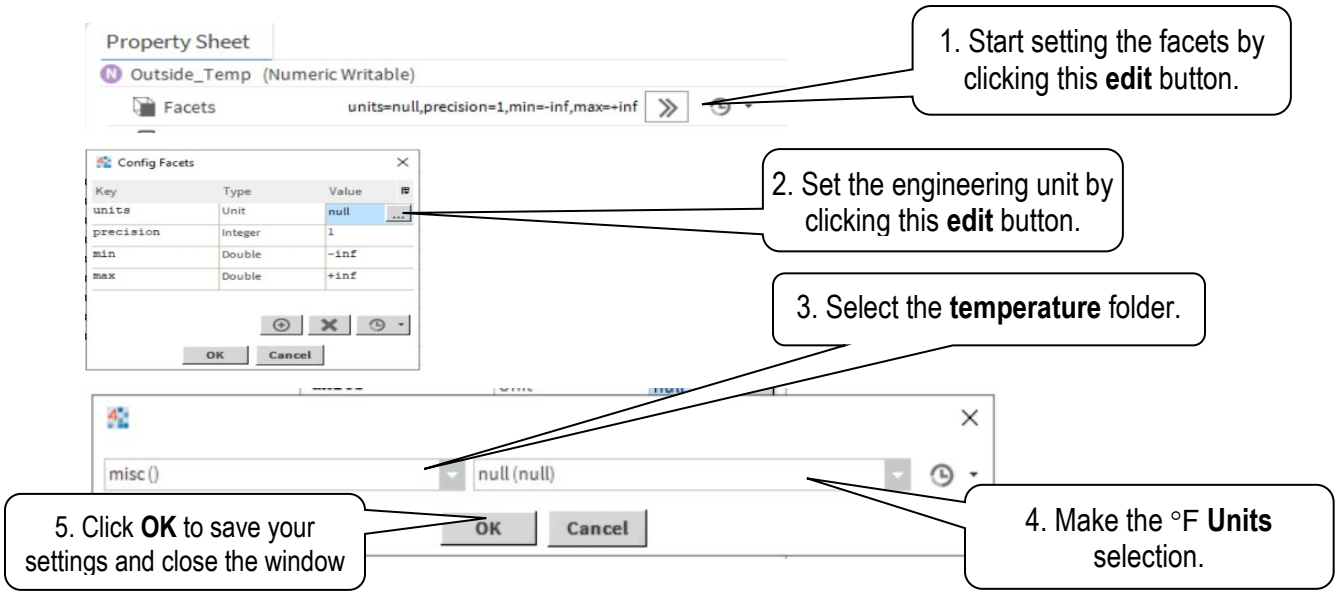
8. Double-click the point to open the point's **AX property sheet**. *If you pull down the **View** menu in the upper right corner of the Workbench, you'll see there is also* a ***Property Sheet*** *option. This view uses HTML 5 and offers some additional options.*



9. In the point's **AX Property Sheet**, click the » to set the point's facets. Set the true Text to **Pump_On** and for **false** text **to Pump_Off**. Click the OK button at the bottom of the dialog box, then click the **Save** button to Save the changes to the point.

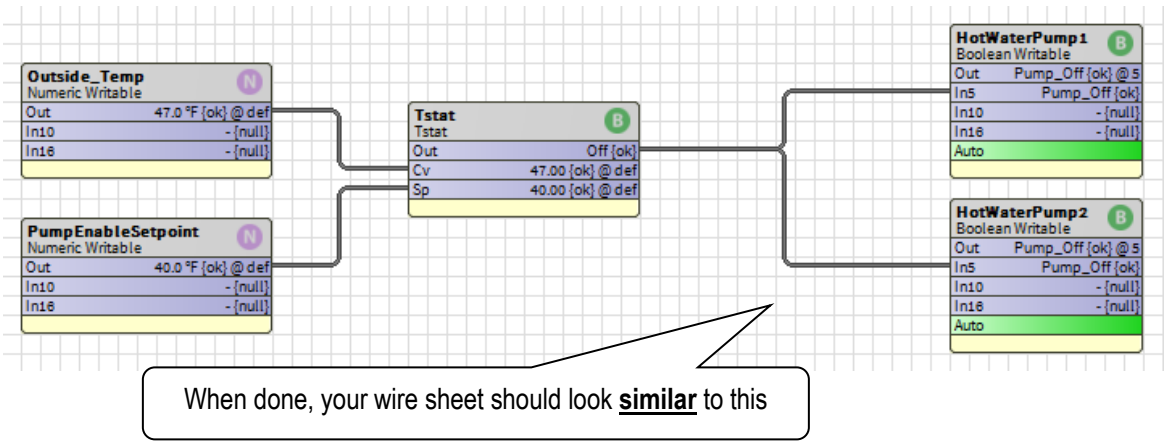


10. Return to the wire sheet of the **PumpControl** folder [*click the **PumpControl** folder in the gray path bar above the Nav SideBar or double-click the **PumpControl** folder in the Nav SideBar*].
11. Right-Click on the HotWaterPump_1 point and select Pin Slots at the bottom of the popup menu. Click to place a pin next to the In5, and Auto slots.
12. Add a **NumericWritable** point to the wire sheet. Name the point **Outside_Temp** as you add it to the wire sheet.
13. Open the AX Property Sheet of the point and set the point facets (engineering units) to °F



14. Right-click the name of the point in the AX Property sheet (note point name at the top of the property sheet window). Choose **Actions**, then **Set**. Set the point's **Fallback (default)** value to 40°F (*Verify that the **Fallback** (default) value updates*).
15. Switch back to the PumpControl wire sheet, right-click on HotWaterPump_1 and choose Actions>Set from the right click popup menu. Change the value to Pump_Off then Click the OK button to set the point's default value.

16. In the **kitControl** palette in the Palette SideBar, locate the **HVAC** folder.
17. From the **HVAC** folder in the **kitControl** palette, drag a **Tstat** object out to the wire sheet of the **PumpControl** folder.
18. Open the AX Property sheet of the **Tstat** (On your wire sheet!)object (double-click it) and adjust these settings:
    a) Set **Facets: trueText= On / falseText= Off**
    b) Set **Null** on **In Inactive** to **False**. (True sets the out to null when inactive)
    c) Set **Diff** (Differential) to **4.** (Click the ↧ icon and uncheck null, then enter a value of 4)
    d) Set **Action** to **Reverse** (A rising value at the input will cause the output to turn off)
    e) Set **Null** on **In Control** to **False**. (True nulls the out in the Diff "Dead band")
19. Return to the wire sheet of the **PumpControl** folder. Right-click the **Tstat** object and choose **Pin Slots**. Pin open the **Cv** (Control Variable) and **Sp** (Setpoint) inputs and the Diff (Differential)
20. Right-click the **Outside_Temp** point and choose **Duplicate**. Name the new point **PumpEnableSetpoint**.
21. Connect the **out** slot of the **Outside_Temp** numeric writable point to the **Cv** input on the **Tstat** object.
22. Connect the **out** slot of the **PumpEnableSetpoint** point to the **Sp** input on the **Tstat** object.
23. Connect the **out** slot of the **Tstat** object to the **In5** slot of the **HotWaterPump_1** point.
24. Right-click the **HotWaterPump_1** point and choose **Copy**.
25. Right-click the wire sheet and choose **Paste Special.**
    a) Set the **Number of Copies** setting to 1
    b) Select the **Keep All Links** check box
    c) Select the **Keep All Relations** check box
    d) Click **OK** to create another point called **HotWaterPump_2**. Note the link between the Tstat output and the new BooleanWritable points.
26. **Modify the Outside_Temp** value by setting the value above and below the Tstat **PumpEnableSetpoint** and monitor the effect on the Hot Water Pumps. Notice the impact of the differential on the status of the pumps. (*Suggestion: Right-Click the point, choose Actions and then Set to change the values*)



(End of lab, save station and stop here!)

LAB #3: Hot Water Pump Control, Part 2	15 min
Description:	In this lab you will add freeze protection to the Hot Water Pump Control system to prevent normally scheduled operation when Outside Air Temperature drops below a threshold.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create a weekly schedule. 2. Configure the default value and weekly event outputs. 3. Link the Schedule into your control logic as the primary means of control based on the time of day and the day of the week. 4. Implement a higher priority control for freeze protection.

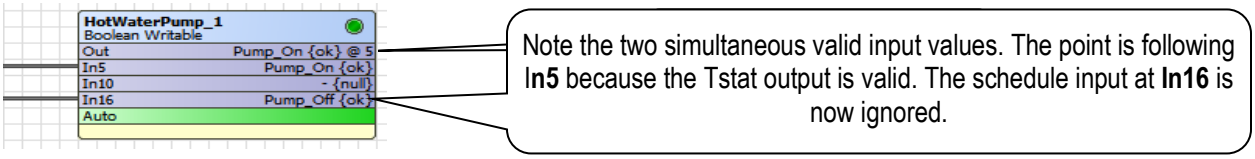
Configuring the Pumps to Follow a Schedule with Freeze Protection

1. Open the wire sheet of the **PumpControl** folder.
2. Open the **Schedule** palette (*click Open Palette, type an "S" into the filter slot. Select the Schedule module and click OK*)
3. In the Schedule palette, select a **BooleanSchedule** and add it to the wire sheet. Name the schedule **PumpSchedule**.
4. Double-click the new schedule to open the **AX Scheduler** view.
5. Select the **Properties** tab at the bottom of the Scheduler view.
6. Edit the schedules facets to use the following:
   a) True text = **Pumps_Enabled**
   b) False text = **Pumps_Off**
7. Set the default value of the schedule by selecting **Pumps_Off** from the pull-down menu.
8. Return to the wire sheet of the PumpControl folder.
9. Connect the output of the **PumpSchedule** to **input 16** on **HotWaterPump_1**.
10. Connect the output of the **PumpSchedule** to **input 16** on **HotWaterPump_2**.

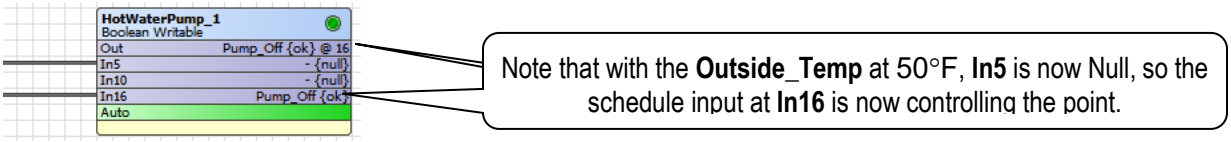
    Note: *We now have two input connections to the hot water pumps. Remember that the point will follow the input with the highest priority when more than one connection is made to a point. In this case, our pumps will not follow the schedule at all, because the input from the Tstat at input 5 is always a valid value. We need to change that!*
11. Open the property sheet of the **Tstat** block.
12. Change the setting called **Null on Inactive** to **True**. *We do not want to shut the pumps off with the **Tstat** block, just bring them on for freeze protection.*
13. Save your change and go back to the folder's wire sheet.

(*Continued on next page*)

14. Modify the **Outside_Temp** point again to a value of 30°F. Note the status of the pumps and the priority slot the point is following.



15. Modify the **Outside_Temp** point again to a value of 50°F. Note the status of the pumps and the priority slot the point is following.

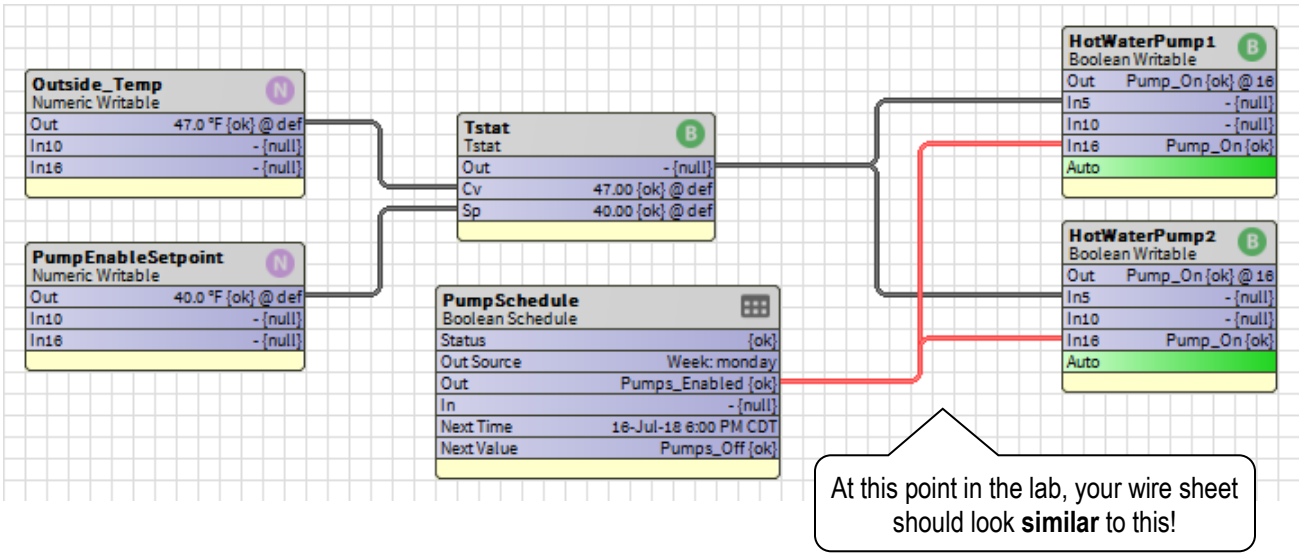


16. Open the **AX Scheduler** view of the schedule.

17. Click the **Weekly Schedule** tab at the bottom of the **AX Scheduler** view. Set the schedule for Monday to have the pumps running from 6:00 am to 6:00 pm.

18. Right click on Monday and choose **Apply M-F** to copy Monday's schedule to the remaining 4 days of the work week. Leave Saturday and Sunday set for the default output of **Pumps_Off**.

19. Return to the wire sheet view of the PumpControl folder.

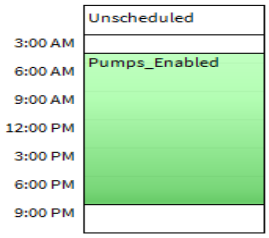


20. From the **Schedule** palette, drag a **Calendar Schedule** to the wire sheet of the folder. Call the new schedule **HolidayCalendar**.

21. Double-click the new **HolidayCalendar** to open the **AX Calendar Scheduler** view.

    a) Click the **Add** button. Enter the name **Christmas**
    b) Note the default **Type** setting is **Date**, this is what we want
    c) Remaining settings: Any Weekday, 25, Dec, Any Year
    d) Click **OK** to save the special event

22. Add **Thanksgiving** as a special event. Make these settings:
    a) Click the **Add** button and name the event
    b) for the **Type** setting, choose **Week and Day**
    c) Select **Thursday**, **Week 4**, **November** for the remaining settings.
23. Save your changes and return to the wire sheet of the **PumpControl** folder.
24. Open the Scheduler view of the **PumpSchedule**. Click the **Special Events** tab.
25. Click the **Add** button to add a special event. Call the event **Holidays**.
    a) Set the **Type** setting to **Reference**
    b) In the **Calendars** slot that appears, select the **HolidayCalendar** schedule (click on it!), then click the **OK** button.
    c) On the right side of the view, note that the schedule has no events defined by default. Right Click on the event output pane and select All Day Event. Set this event to output **Pumps_Off** by selecting from the drop-down menu.
26. Add another event called **Inventory** that will have the pumps running from 5:00 am to 9:00 pm on every Tuesday in January of only next year.
    a) Use the **Custom** Type setting when creating the event
    b) Choose: **Any Day**, **Jan**, **Tue**, **Any Week** as the event settings. Specify the next calendar year. Click **OK**.
    c) Set the 5:00 am - 9:00 pm times in the schedule on the right side of the view.
    d) Allow the remainder of the day to remain Unscheduled. This way, the Inventory special event merely deviates from the normal weekly schedule for the defined period of time. Outside this defined period, normally scheduled events are unaffected.



    e) Save your changes.
27. Click the Summery tab on the bottom of the page. In the calendar view, locate and select a Tuesday in January of next year. Verify that the schedule has the pumps running from 5:00 am to 9:00 pm on this day.

    Note that the order that the events appear in the Special Events view, top to bottom, reflects the priority of the event. The higher in the list, the higher the priority. If you have conflicting special events, this is important!

(*End of lab, save station and stop here*)

This page is intentionally left blank!

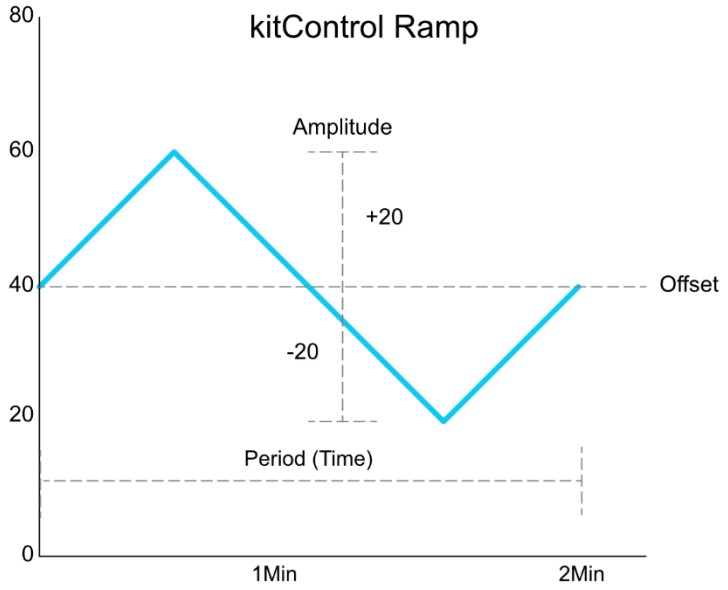
LAB #4: Hot Water Pump Control, Part 3	15 min
Description:	In this lab you will learn how to use a Ramp object to simulate fluctuating numerical values.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create and Configure a Ramp object. a. Amplitude b. Offset c. Period d. Update Interval 2. Link the Ramp into the Outside Air Temp object to simulate a fluctuating OATemp.

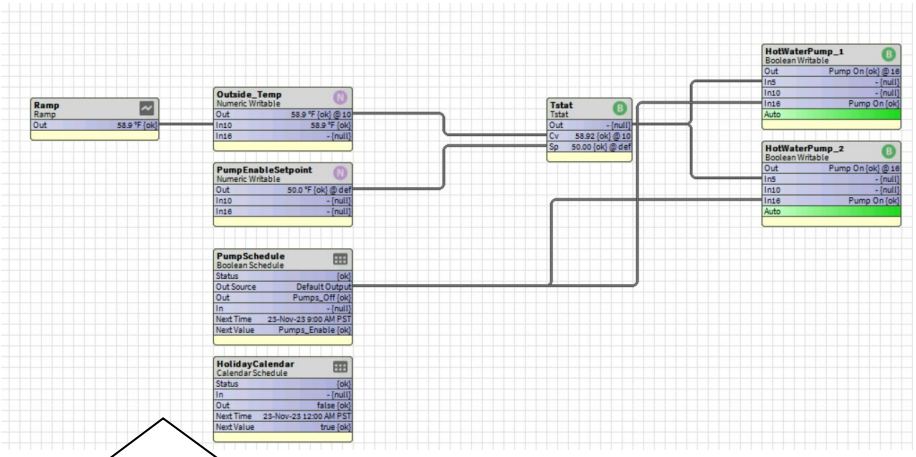
The technique practiced in this lab exercise is commonly used for testing and debugging of control system logic using simulated system variables. Make sure that your controls actually function as planned before installing the system in your customer's building!

1. Open the wire sheet of the **PumpControl** folder.
2. From the **kitControl** palette's **Util** folder, add a **Ramp** block to the wire sheet.
3. Open the property sheet of the new **Ramp** object. Make these settings:
   a) Set the Facets for and engineering unit of °F Set the **Amplitude** value to **20**
   b) Set the period setting to 2 minutes
   c) Set the Amplitude value to 20
   d) Set Offset to 40
   e) **Save** your changes

> **Note:** the settings for the ramp object represent a fluctuating temp in a range that you set. Here is the offset formula.
>
> **a) [Low temp + high temp / 2 = offset] or 20°F+60°F / 2= 40°F**
>
> **Offset** is the ***middle*** point between the high and low temperatures of your desired range. To calculate the needed offset, take the low number you know you want at the bottom of the range and add it to the high number you want at the top of the range. Divide the sum by 2.
>
> b) **[High Temp - Offset = Amplitude]**
>
> **Amplitude** is how far from the ***Offset*** you want to fluctuate *(*in both **High** and **Low** directions). To calculate the amplitude, take the number you want at the top of the range, subtract the offset you calculated first. The difference is the amplitude setting.

4. Return to the wire sheet of the **PumpControl** folder. Connect the output of the **Ramp** block to **In10** on the **Outside_Temp** point.
5. Observe the Pump points as they shift between control at priority levels 5 and 16 (following input slot 5 and input slot 16) as the simulated outside temperature rises and falls.



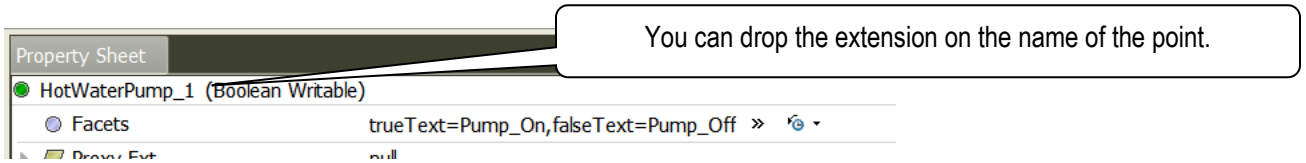


When the lab is complete, your wire sheet should look **similar** to this!

(End of lab, save station and stop here!)

LAB #5: Hot Water Pump Control, Part 4	20 min
Description:	In this lab you continue to build the control logic for a Hot Water Pump Control system.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create and Configure a LeadLagRuntime object. 2. Extend the functionality of a Hot Water Pump object with a Totalizer extension. 3. Open the Composite Editor and create new slots that are linked to lower-level slots.

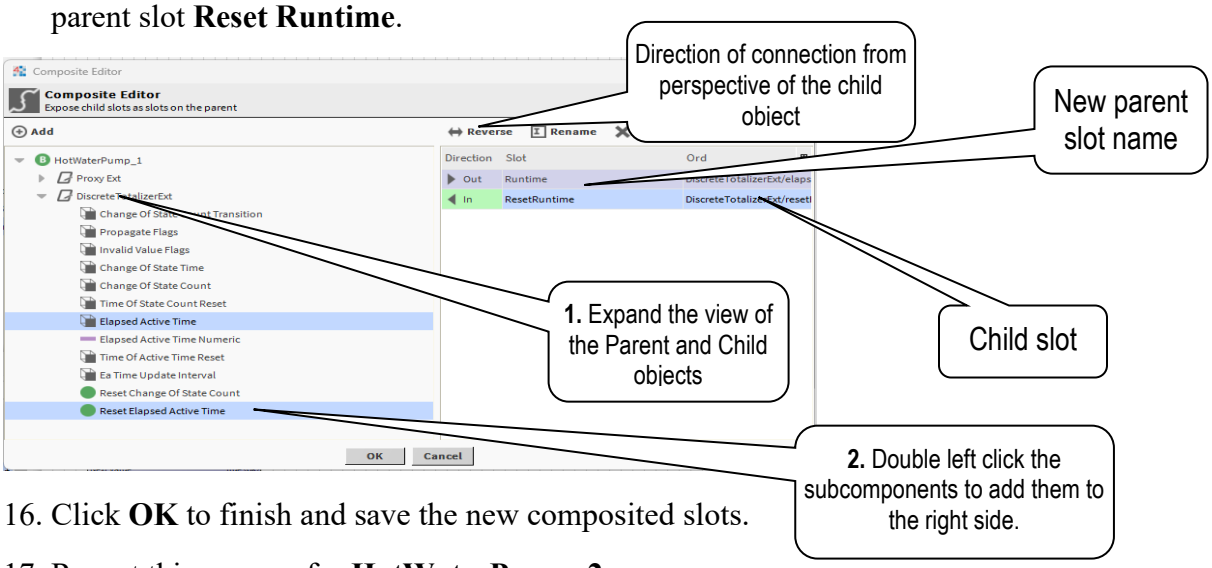
1. Open the wire sheet of the **PumpControl** folder.
2. Break the link between the schedule and the Hot Water Pumps.
3. From the **HVAC** folder of the **kitControl** palette, add a **LeadLagRuntime** block to the wire sheet. Rename the block to **PumpLL**.
4. Open the Property Sheet of the **PumpLL** block. Set **Max Runtime** = **5 minutes** (this will cause the lead pump to change every 5 minutes for class purposes)
5. Return to the wire sheet of the **PumpControl** folder.
6. Connect:
   a) the output of the **Pump schedule** to the **in** slot on the **PumpLL** block
   b) the **OutA** output of the **PumpLL** block to **in16** on **HotWaterPump_1**.
   c) the **OutB** output of the **PumpLL** block to **in16** on **HotWaterPump_2**.
7. Open the property sheet of the **HotWaterPump_1** point.
8. In the **kitControl** palette, expand the **ControlPalette** folder, then the **Extensions** folder. Locate and select the **DiscreteTotalizerExt** block.
9. Add this extension to the **HotWaterPump_1** point by dragging the extension to the unused grey space at the bottom of the property sheet, or by dropping it on the name of the point in the property sheet.



10. Open the extension's property sheet and set the **Change of State Count Transition** setting to **"To Active"**.
11. In the Nav SideBar, expand the view of **HotWater Pump_1**. Drag the totalizer extension from **HotWater Pump_1** and drop it on **HotWater Pump_2**. This will copy the extension to pump 2.
12. Return to the wire sheet of the **PumpControl** folder.
13. Right-click the **HotWater Pump_1**. Choose **Composite** from the menu.

14. In the **Composite Editor**, composite the **Elapsed Active Time**, naming the new parent slot **Runtime**.

15. In the **Composite Editor**, composite the **Reset Elapsed Active Time** slot, naming the new parent slot **Reset Runtime**.



16. Click **OK** to finish and save the new composited slots.

17. Repeat this process for **HotWaterPump_2**.

18. In the wire sheet of PumpControl do the following: on **HotWaterPump_1** to the **Runtime A** slot on the **PumpLL** block.

    a) Connect the output side of the new Runtime on HotWaterPump_1 to Runtime A slot on the PumpLL block.

    b) Connect the output side of the new Runtime on HotWaterPump_2 to Runtime B slot on the PumpLL block.

19. From the kitControl palette's **Logic** folder, add an **OR** block to the wire sheet. Call the OR block **PumpStatusOR**.

20. Connect the output of the **OR** block to the **Feedback** input on the **PumpLL** block. An Alarm may occur immediately on the Lead Lag Runtime object due to lack of Feedback.

21. Open the Property Sheet of the **PumpLL** block. Set **Feedback** to **True.**

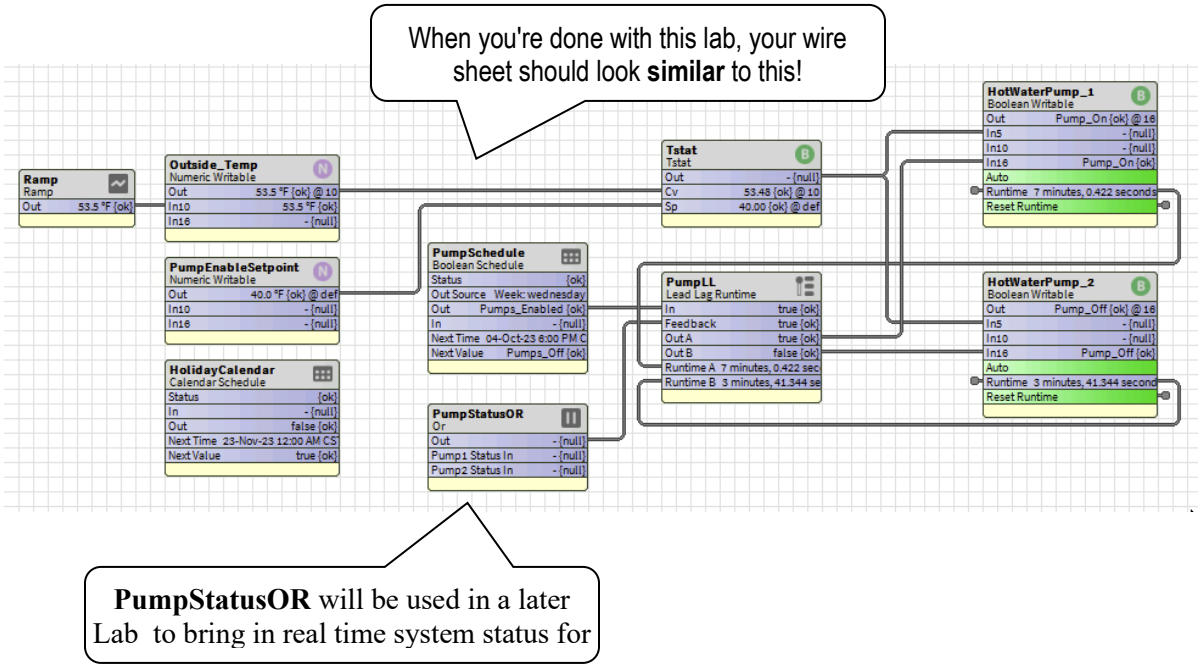
22. If necessary, Right-click the **PumpLL** object, choose **Actions**, then **Clear Alarm State**.

23. Right-click the PumpStatusOR block, choose **Views**, then **AX Slot Sheet**.

24. In the slot sheet, locate and double-click the **inA** slot. In the **Display Name Editor** that opens, type in **Pump 1 Status In**. Click **OK** to save your changes.

25. Repeat these basic steps to rename the **inB** slot to **Pump 2 Status In**.

Note: The **PumpStatusOR** block displays a red status color but will be used later to link to actual system data points for the purpose of bringing in real time system status for feedback.



(End of lab, save station and stop here!)

Note: *If you have time, feel free to continue with the Optional Lab on the next page.*

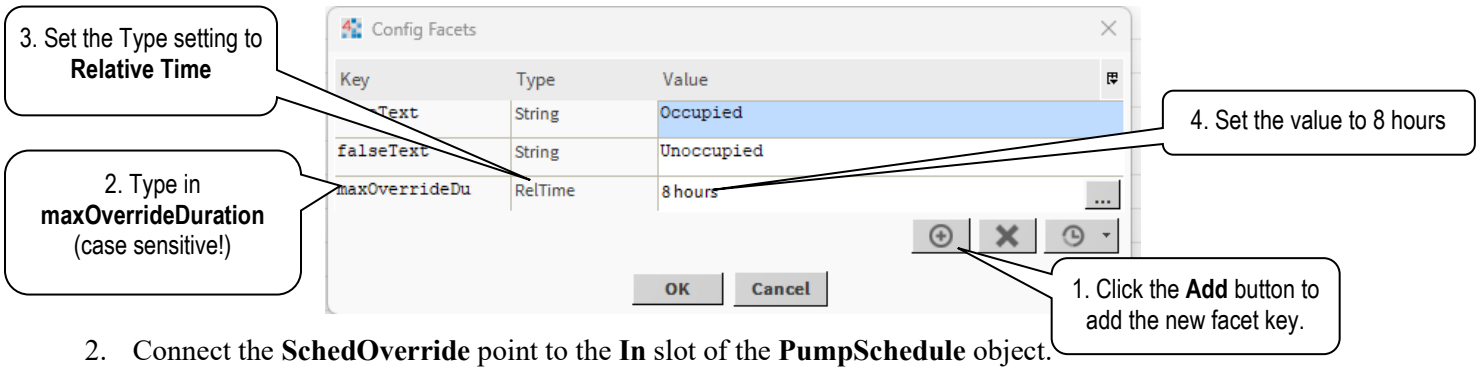
OPTIONAL LAB: Schedule Override	
Description:	In this lab you have an opportunity to expand your knowledge of Schedule-based control.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Add advanced Facet keys. 2. Enforce a Maximum allowable Override Duration: a. for an individual object, or b. for an entire station's Config database. 3. Customize the Schedule Override function by compositing commandable actions.

1. Create a **BooleanWritable** (named **SchedOverride**) on the wire sheet of the **PumpControl** folder.

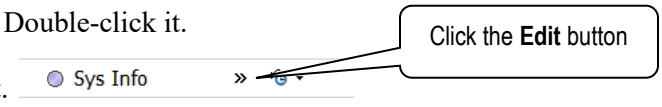
   a) On the **AX Slot Sheet** view of this point, Right-click the **Set** slot and choose **Config Flags** to make the Set command **Hidden**.

   b) On the property sheet view, set facets to true text: **Occupied** / false text: **Unoccupied**.

   c) Add an additional facet key named **maxOverrideDuration** (see illustration) and set the **Type** to **Relative Time**. Set the **Value** to **8 hours.**



2. Connect the **SchedOverride** point to the **In** slot of the **PumpSchedule** object.
3. Right-click the **SchedOverride** point and choose **Cut**.
4. Open the wire sheet view of the **PumpSchedule**. Right-click the wire sheet and **Paste**. Return to the wire sheet view of the PumpControl folder.
5. Open the **Composite** editor for the **PumpSchedule**.
6. From the **SchedOverride** point, composite the **Active** slot (rename to **Override_to_Occ_Mode**), the **Inactive** slot (rename to **Override_to_Unocc_Mode**), and the **Auto** slot (rename to **Cancel_Override**). **Note:** When you exit the Composite Editor, you will probably get an error message. Close the message and click **Refresh** on the button bar to see the composited slots.
7. In the Nav SideBar, locate the **Config** object. Double-click it.
8. Click the **Edit** button next to the Sys Info slot.

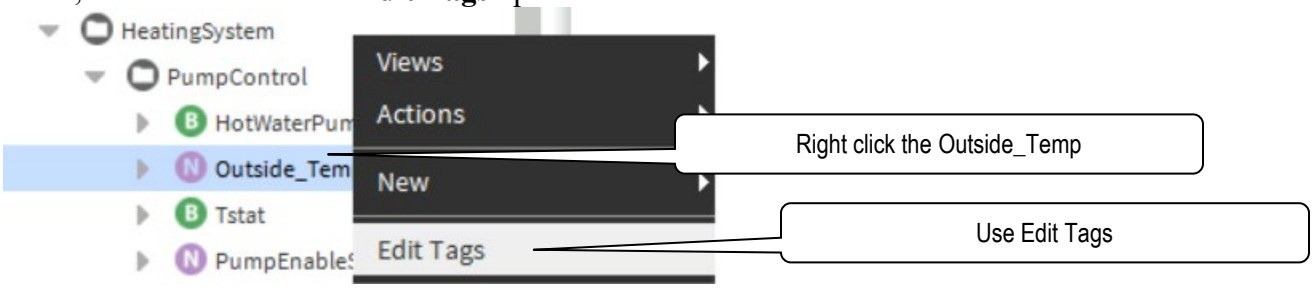


9. Add the same facet key as step 1-d. Save your changes.

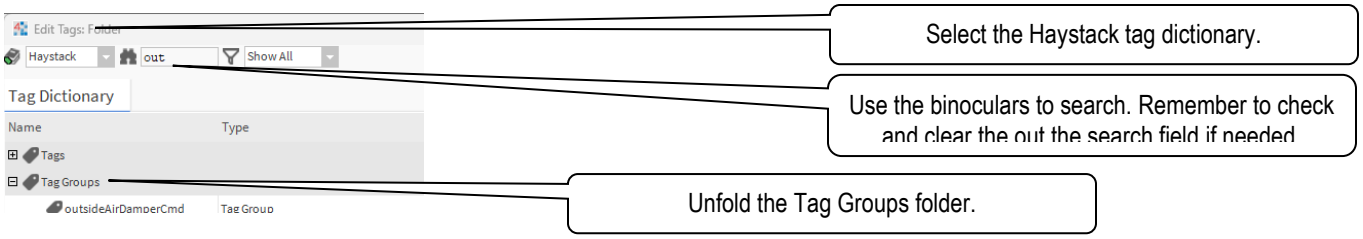
(End of lab, save station and stop here!)

LAB #6: Tagging Objects	15 min
Description:	In this lab you will explore Tag Dictionaries and apply semantic tags to Heating System components. In the lab exercises that follow you will be adding tags to components as you create them.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Apply individual Tags to Niagara components. 2. Distinguish between Direct tags and Implied tags. 3. Apply multiple Tags to a component with a Tag Group. 4. List tagged components with a Search Query.

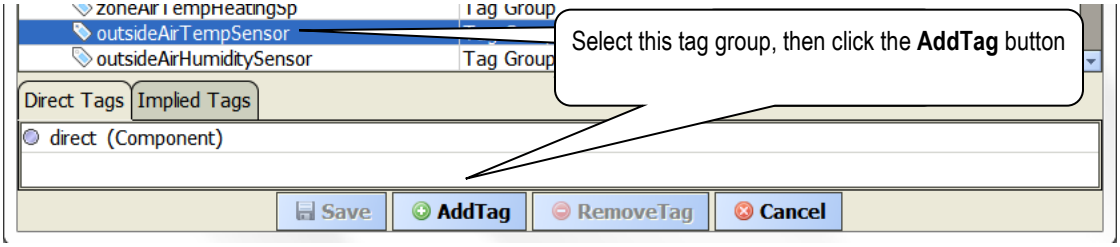
1. In the Nav SideBar, expand the station's Config-->Services.
2. Locate and double-click the **TagDictionaryService**.
3. Note the **Niagara** tag dictionary that is already in the **Tag Dictionary Manager** view. Note the setting called **Namespace**.
4. In the Palette SideBar, open the **Haystack** palette.
   a) There are two folders with dictionaries in them. From the Include smart relations (recommended) folder, add the **Haystack** tag dictionary to the TagDictionaryService by dragging it from the palette and dropping it onto the empty space in the **Tag Dictionary Manager** view. "Note that there is a **Namespace** column. Namespace is noneditable for this Dictionary type."
5. In the Nav SideBar, locate and expand the Control_Logic-->HeatingSystem-->PumpControl folders.
6. In the tree, locate the **Outside_Temp** point. Right-click the **Outside_Temp** point. In the menu, locate and select the **Edit Tags** option.



7. Select the **Haystack** tag dictionary, then expand the **Tag Groups** folder.

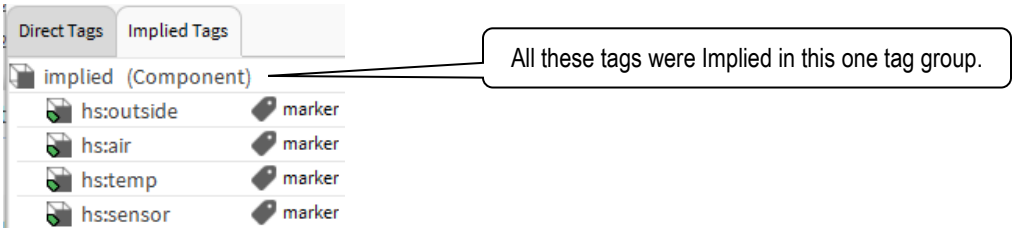


8. In the **Tag Groups** folder, locate the tag group called **outsideAirTempSensor**. Select the tag group (single left click on it), then click the **Add Tag** button (Or double left click it).



9. Click the **Save** button to save your changes. This will close the Edit Tags dialog box.

10. Open the Edit Tags dialog once again. In the **Implied Tags** window, note the tags that were added by adding this tag group. A search for any of these tags will return a result including this point.



11. Close the Edit Tags dialog box when ready to proceed.

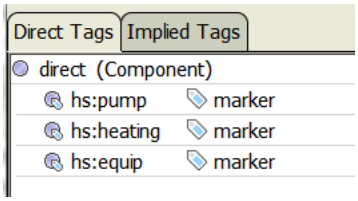
12. In the Nav SideBar, locate **HotWaterPump_1**. Right-click the point and choose **Edit Tags**.

13. Under the **Tags** folder in the **Haystack Tag Dictionary**, locate the tag called **pump**.

    a) Add the tag **pump** to the point

    b) Add the tag **heating** to the point

    c) Add the tag **equip** to the point



14. Assign the same set of tags to the **HotWaterPump_2** point.

15. To the **PumpEnableSetpoint** object in the **PumpControl** folder, add the tags **sp**, and **heating**.

16. Right-click the **PumpSchedule** object in the **PumpControl** folder, choose **Edit Tags**.

17. Click the **Implied Tags** tab to view the implied tags. Note that the marker tag **n:schedule** has already been applied to this object. Click **Cancel** to close the window.

18. In the upper right corner of the Workbench, locate the slot called **Quick Search**.

    a) In this slot, type **hs:sp** and press the **Enter** key. Note the search results.

    b) Modify the search string, typing in **hs:pump**. Note the search results.

    c) Modify the search string, typing in **hs:pump or hs:sp**. Note the search results.

    d) Modify the search string, typing in **hs:pump and hs:sp**. Note that this search yields no results.

(*End of lab, save station and stop here!*)

LAB #7: Create a Custom Tag Dictionary	15 min
Description:	In this lab you learn how to create and manage a Custom Tag Dictionary.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create a Custom Tag Dictionary. 2. Define the Tags for the new Tag Dictionary. 3. Create Tag Groups within the new Tag Dictionary. 4. Add Tags to an object using a custom Tag Group. 5. Perform a search using the new custom Tags.

1. In the **Services** container in the station, locate and double-click the **TagDictionaryService**. This opens the **Tag Dictionary Manager** view.
2. Open the **tagDictionary** palette in the Palette SideBar.
3. From the palette, drag and drop a **SmartTagDictionary** to the **Tag Dictionary Manager** view.
   a) Name the new dictionary **CustomDictionary**
   b) Select it and click the edit button. Set the **Namespace** to "**c**" (You will see this error until you do this!)

4. Locate the new tag dictionary in the Nav SideBar and double-click it to open the **Property Sheet** view (note: in Niagara 4, the default view is the AX Property Sheet).
5. In the Palette SideBar, expand the **Tags** folder in the tagDictionary palette.
6. From the **Tags** folder in the tagDictionary palette, create the following custom tags:

Drag From Palette/Folder/Type		
Tag Type:	**Name:**	**Default Value:**
Marker	heating	
Marker	hvac	
Marker	centralPlant	
String	note	heatingControl (expand the note and set the Default Value in the dropdown)
Marker	proxiedPoint	
Marker	graphic	
Marker	visible	
Marker	fcu	
Marker	boilerPlant	

7. In the Palette Sidebar, locate and select the **TagGroup** object. Drag a **TagGroup** from the palette and drop it onto the **Tag Group Definitions** object in the property sheet. Name the new tag group **HeatingSys**.
8. Expand the new **HeatingSys** tag group in the property sheet.
    a) From the **TagDefinitions** object on the CustomDictionary Ax Property sheet, drag down and drop these tags into your new Tag Group Tag List: **heating**, **hvac**, **centralPlant** and **note.**
    b) Verify that the default text value of the **note** tag is **heatingControl**.
9. In the Nav SideBar, locate the **Tstat** object in the **PumpControl** folder. Right-click the **Tstat** and choose **Edit Tags**.
10. Select the new **CustomDictionary**.
11. Unfold the **Tag Groups** object and locate the **HeatingSys** tag group. Select this tag group, then click the **Add** tag button. (Click Save!) Note: Don't see the tag list? Make sure the filter in the tag view is cleared!
12. Right-click the **Tstat** again and choose **Edit Tags**. Notice the four tags in the group display in the **Implied Tags** view. Close edit box.
13. In the Nav SideBar, expand the **Services** container.
14. Locate and double-click the **SearchService**. This will open the **Search** view.
15. In the search definition slot, type in **c:heating** and click the **Search** button.

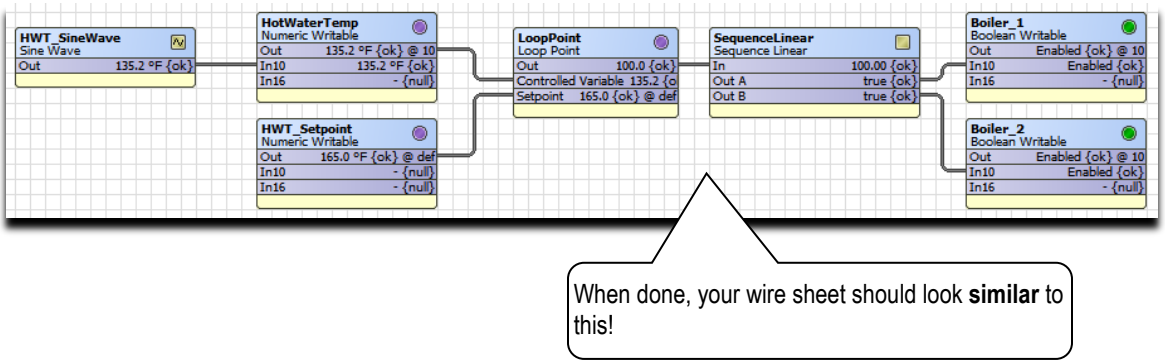
16. Change the search string to **c:centralPlant** and run a search. You should see the same result of the **Tstat** object since we assigned this tag to just that object. Click the **Tstat** object name and note the target of the hyperlink.
17. Run another search using the tag **hs:heating or c:heating** (the namespace **hs** points to the Haystack tag dictionary). Note the results.

(***End of lab, SAVE YOUR STATION and stop here!***)

LAB #8: Boiler Control Loop	20 min
Description:	In this lab you will configure a closed loop control scheme that controls hot water temperature.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Establish a Closed Loop control scheme using the Niagara LoopPoint object. 2. Configure the SequenceLinear object to provide staging control of multiple loads in a linear sequence. 3. Configure a SineWave object to simulate fluctuating numeric values.

1. In the **HeatingSystem** folder, create a new folder called **BoilerControl**. From the Custom tag dictionary, add the **graphic** and **boilerPlant** tags, and from the Haystack tag dictionary add the **steamPlant** tag to this object. (Remember to right click
2. Now on the wire sheet of the BoilerControl folder create a **BooleanWritable** point called **Boiler_1**. Set facets for this point to **Enabled** (true text) and **Off** (false text). Set **Off** as the Fallback (default value).
    a) From the CustomDictionary assign the tag group **HeatingSys** to this point.
    b) Assign the **boiler** tag to the point from the Haystack dictionary.
3. **Copy** and **Paste Special** to duplicate the Boiler_1 point with all links and relations intact. Name the new point **Boiler_2**. Verify that the HeatingSys tag group is still assigned to this new point.
4. In the BoilerControl folder, create a **NumericWritable** point called **HotWaterTemp**. Set the facets to °F and the default value to 165°F **degrees**.
    a) Modify the new point's facets by setting the **Min** value to 50°F and the **Max** value to 200°F
    b) From the CustomDictionary, assign the tag group **HeatingSys** to this point
    c) From the Haystack dictionary, assign the **temp** tag to the point
5. Copy & Paste Special (with links/relations) **HotWaterTemp**. Name the point **HWT_Setpoint**.
    a) Assign the **sp** tag to the point from the Haystack dictionary.
6. From the **kitControl** palette's HVAC folder, add a **LoopPoint** object (a.k.a. PID) to the folder's wire sheet. Make these settings in this new object:
    a) Set the Loop Action setting to Reverse (Rising input Þ Falling output)
    b) Set the Proportional Constant to 5 Set the **Proportional Constant** to 5
    c) **Set the Bias setting to 50**
    d) **Ramp Time to 1 minute (in a real job, this setting could be set at 1 hour or even longer)**

7. Return to the wire sheet of the folder. Right-click the **LoopPoint** object. Choose **Pin Slots** from the menu.
    a) Pin open the slots labeled **Controlled Variable** and **Setpoint**
    b) Click the "OK" button to return to the folder's wire sheet.
8. Connect:
    a) The output of the **HotWaterTemp** point to the **ControlledVariable** slot on the **LoopPoint** object.
    b) The output of the **HWT_Setpoint** point to the **Setpoint** slot on the **LoopPoint** object.
9. From kitControl's **HVAC** folder, add a **SequenceLinear** object to the folder's wire sheet.
    a) Connect the **Out** slot of the **LoopPoint** object to the **In** slot on the **Sequence Linear** object.
10. Set the **SequenceLinear** object's setting called **Number of Outputs** to **2**. Return to the folder's wire sheet.
11. Connect **Out A** of the **SequenceLinear** object to **in10** on the **Boiler_1** point. Connect **Out B** of the **SequenceLinear** object to **in10** on the **Boiler_2** point.
12. From kitControl's **Util** folder, add a **SineWave** object to the folder's wire sheet. Call the new object **HWT_SineWave**.
13. We want the hot water temp simulation to move between 55°C [135°F] and 85°C [185°F] over a period of 2 minutes. To make this work we need to make these settings in the new **HWT_SineWave** object.
    a) set the **Amplitude** to 25
    b) set the **Offset** to 160°F
    c) set the **Period** (*NOT THE INTERVAL*!) setting to 2 minutes
    d) set the **Facets** (engineering units) to °F
14. Return to the wire sheet of the folder. Connect the output of the **HWT_SineWave** to the **in10** slot on the **HotWaterTemp** object.



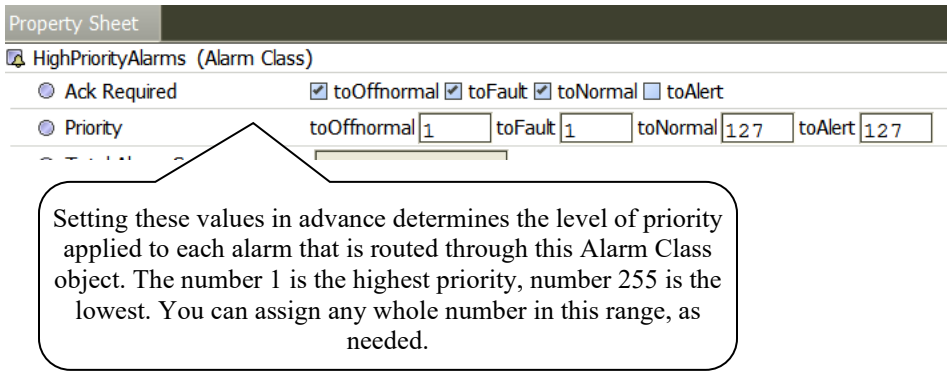
(End of lab, SAVE YOUR STATION and stop here!)

LAB #9: Alarm Notification	25 min
Description:	In this lab you will configure the Alarm service and extend the functionality of system components to allow for automatic alarm notification.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Configure the Alarm Service. 2. Configure Alarm Classes. 3. Add Alarm Recipients. 4. Add and configure Alarm Extensions. 5. Monitor and respond to alarm notifications. 6. Perform alarm database maintenance.

Part 1, Configure the Alarm Service

Procedure 1, Alarm Classes

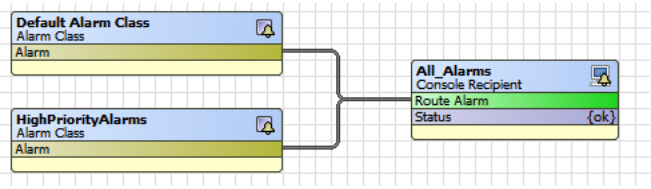
1. Double-click the **AlarmService** node in the **Services** component of your Station. This will open the wire sheet view of the service.
2. Open the **alarm** palette in the Palette SideBar.
3. From the palette, add a new **Alarm Class** object to the Wire Sheet of the AlarmService. Name the new Alarm Class **High_Priority_Alarms**.
   a) Open the **AX Property Sheet** of the **High_Priority_Alarms** class object. Set the **Priority** setting for this Alarm Class object to these settings:



Procedure 2, Alarm Recipients

1. From the **Alarm** palette's **Recipients** folder, add a **Console Recipient** to the **AlarmService** wire sheet. Name the console recipient **All_Alarms**.
2. Link from the **Alarm** slot of the **Default Alarm Class** to the **Route Alarm** slot of the **All_Alarms** console recipient.
3. Link the **High_Priority_Alarms** alarm class object to the **All_Alarms** console recipient.

The AlarmService wire sheet should resemble the illustration below.



Part 2, Alarm Extensions

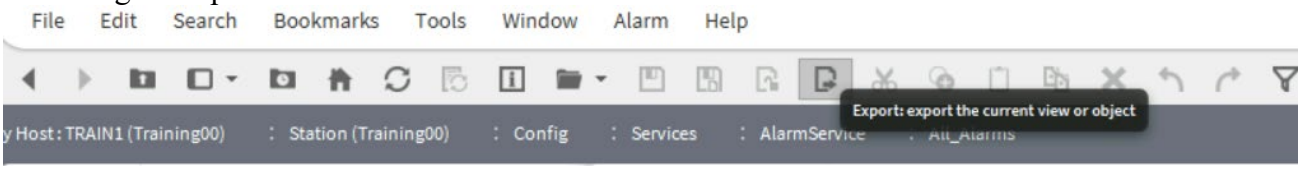
1. In the **BoilerControl** folder open the AX Property sheet of the **HotWaterTemp** point.
2. In the **Alarm** palette, expand the **Extensions** folder.
3. From the **Extensions** folder in the palette, add an **OutOfRangeAlarmExt** to the AX Property sheet of the **HotWaterTemp** pointExpand the OutOfRangeAlarmExt.
   a) Note the Source Name is set to %parent.displayName% by default.
   b) Set **To Normal Text:** Temperature is back to normal
   c) In the **Offnormal Algorithm** of the alarm extension
      i. set the **high limit** to 178°F
      ii. set the **lowlimit to** 142°F
      iii. set the **deadband to 4**
      iv. **High limit text**: Temperature is above limits.
      v. **Low limit text**: Temperature is below limits.
      vi. Check the **Low Limit Enable** and **High Limit Enable.**
   d) Set the **Alarm Class** setting to **High_Priority_Alarms.**
   e) **Save** your changes.
4. In the **Heating System Pump Control** container, locate the **Outside_Temp** point and open its AX Property sheet view.
5. From the **Extensions** folder in the Alarm palette, add an **OutOfRangeAlarmExt** to the property sheet of the **Outside_Temp** object's property sheet.
   a) Set the **Source Name** to **%parent.parent.parent.displayName%_%parent.displayName%**
   b) In the **Offnormal Algorithm** of the alarm extension
      i. Set the **low limit to** 32°F.
      ii. Check **lowlimitEnable** checkbox.
      iii. **Low limit message text**: Temperature is below limits.
   c) Set the **Alarm Class** setting to **High_Priority_Alarms.**
   d) **Save** your changes (an alarm may generate right away, check **Out** slot status)

Part 3, View Alarms

1. Open the wire sheet of the **AlarmService** component in your station.

2. Double-click on the **All_Alarms** console recipient to verify receipt of alarms. Note the **Source Name** column. Acknowledge the Outside_Temp alarm.

3. Click the **Export** icon on the Tool Bar and **export** the console **to pdf**. **Close the file** after reviewing the report.



4. Right-click the **AlarmService** component. Choose **View**, then **Alarm Ext Manager**. Note the **toOffnormalEnabled** setting for an individual alarm extension.

5. Right-click the **AlarmService** component, choose **Views**, then **Alarm Db View**.

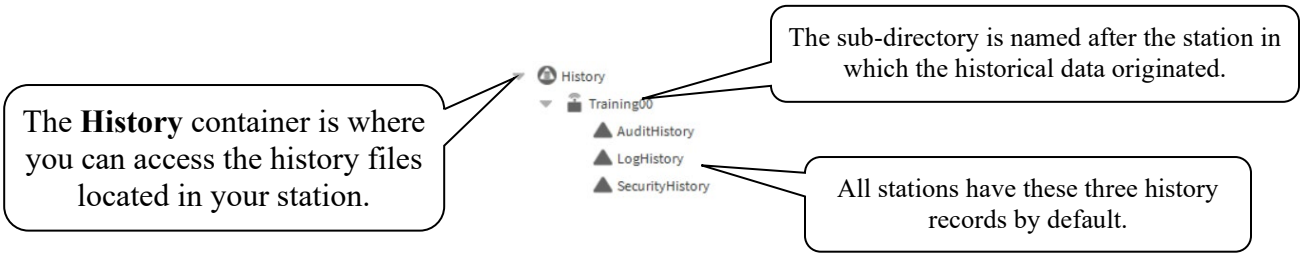
   a) Double-click an alarm in the database to see details of the alarm.

(*End of lab, **Backup the station** and stop here!*)

This page is intentionally left blank!

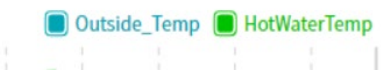
LAB #10: History / Trend Data Collection	20 min
Description:	In this lab you will configure the History services and extend the functionality of system components to allow for automatic trend data collection.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Configure History Services. 2. Add History Extensions. 3. Enable / Disable Trend Data Collection. 4. History Views. 5. Chart both Historical Trend Data and Real Time Data

1. In your station, locate the **History** container. Expand the view of this container, then expand the view of the container inside (named after your station). Note the history files inside this folder.



2. Open the **SecurityHistory** file and review the information in the record. Note the date and time of the last **Login** to the station, and the identity of the operator who most recently signed in.
3. Open the **History Palette** in the Palette SideBar. Expand the view of the **Extensions** folder in the palette and note the extensions inside.
4. Open the property sheet of the **HotWaterTemp** point in the **BoilerControl** folder.
5. From the History palette's Extension folder, drag-and-drop to add a **NumericCov** extension to the HotWaterTemp point's property sheet. When prompted name the extension to **HWT_History**. Once added, expand the **HWT_History** node to show the extensions properties. Make these settings in the extension:
   a) Set the **Enabled** to **True**
   b) Under the **History Config** set the **Capacity** setting to 600.
   c) Set **Tolerance** to **5**
   d) Save your changes.
6. Return to the History container in the Nav SideBar. Note the name of the new history file that was added when the extension was enabled.
7. Open the property sheet of the **Boiler_1** point. Add a **BooleanCov** extension to the point by dragging the extension from the palette and dropping it on the name of the point in the point's property sheet. Enter **Boiler1_History** as the History Name. (It may be necessary to right-click the History container and select Refresh Tree Node from the popup menu)
8. **Enable** the new extension.

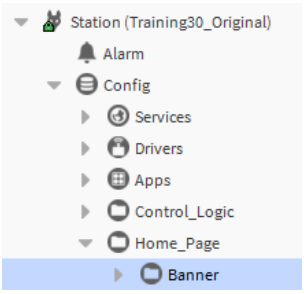
9. Locate the **Boiler_1's** extension in the **Nav SideBar**. Copy the extension to **Boiler_2** by dragging it from the **Boiler _1** point to the **Boiler _2** point and dropping it on **Boiler _2**. Enter **Boiler2_History** as the History Name.
10. Copy the same extension to **HotWaterPump_1** and **HotWaterPump_2** in the **PumpControl** folder. Name all new history extensions appropriately.
11. Double-click the **HistoryService** item in the **Services** container of your station. Note that all history extensions in the station are displayed.
12. **Enable** all histories that are currently disabled (select all the extensions that are currently disabled, right-click, choose **enable collection**).
13. Enable all histories that are currently disabled (Hold the CTRL key and select all the extensions that are currently disabled, right-click and choose Enable Collection.
14. Verify that the histories set up in the lab exist in the **History** folder for your station.
15. Open the **HotWaterTemp** history chart (double-click the history file). Modify the **Time Range** setting. Observe the difference in the chart that is built.
16. Double-click this history file HotWaterTemp to open a history chart. Modify the Time Range setting. Observe the difference in the chart that is built. Drag the Boiler_1 History onto this view and see how the two histories are combined onto one chart. Click the button next to each history and observe the change.



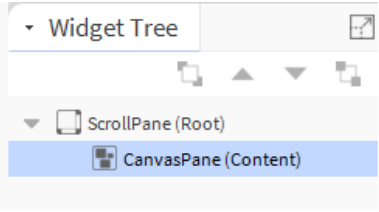
17. Double click the Boiler_1 history to open and review the Boiler_1 chart. From the View menu in the upper right-hand corner of the View pane select and choose History Table from the popup menu. Review the table entries. Right-click the **History** container in your station. Choose **Views**, then **Chart**.
18. Drag onto the chart the **HotWaterTemp** history and the **Boiler_2** point. Locate the **Export Wizard** button on the chart (next to the time filter) and click the button.
    a) Leave the **Base Ord** field at the default path
    b) In the Select Destination drop-down select **Station File**
    c) **File Name** leave as default.
    d) **View on Export**, check the box which sets the setting to **True.**
    e) Click **OK** to create the chart. Doesn't look different? It's not supposed to!
    f) In the Nav SideBar, expand the **Files** container. Note the new folder called **Chart**. Open this folder and you'll see the new chart stored there.

(***End of lab, save station, then backup the station, and stop here!***)

LAB #11: Creating a PX Include	15 min
Description:	In this lab you will create a simple PX file that will be used throughout this course to embed some standard visual information at the top of every other PX View.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create a New PX View. 2. Launch the Make Widget Wizard. 3. Create a simple PX file that can be included into other PX Views. 4. Create graphical widgets for real-time display of individual components and data points. 5. Save all changes to a PX View.

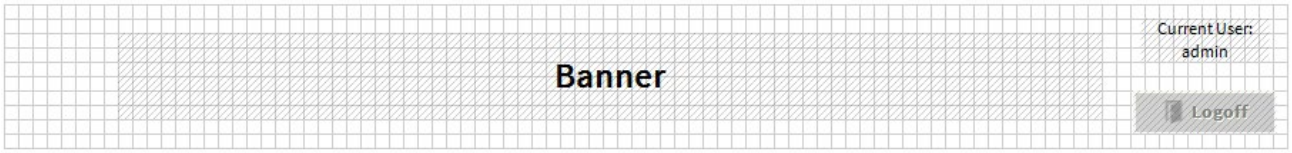


1. In the Nav SideBar, right-click **Config**, select **New**, then **Folder**. Create a new folder called **Home_Page**.
2. Right-click the new **Home_Page** folder, select **New**, then **Folder**. Create a new folder called **Banner** (this folder is simply an object on **which** to place a PX view).
3. Right-click the **Banner** folder. Choose **Views**, then **New View**. In the new view wizard that opens name the view **BannerPX**. Click **OK** to create the view. Note that the view opens in the **Edit** mode.
4. Review the **Locator Bar** and note that the new view is associated with the **Banner** folder in the station.



5. On the right side of the view, you'll see SideBars that focus on the PX view you're working on. Locate the **Widget Tree** SideBar.
6. Expand the **Scroll Pane** to reveal the **Canvas Pane**. Right-click the canvas pane and choose **Cut**.
7. After cutting the canvas pane, right-click the **Scroll Pane** and choose **Delete**.
8. You'll now see an object called **Root** in the SideBar. Right-click the root and choose **Paste**.
9. Double-click the middle of the Canvas Pane (the grid pattern) to open its **Properties**.
10. Set the **View Size** setting to 900 width by 100 height. Click **OK**. Resize the left and right SideBars, and save the view. Ensure the PX Editor is in Edit mode.
11. Open the **kitPX** palette in the Palette SideBar. Locate the **LogOff** button. Drag the button onto the view and position it as you see fit. Upper Right corner is recommended!
12. Drag the Banner folder from the Nav Tree and onto the Canvas Pane so that the **Make Widget Wizard** launches. Choose **Bound Label** from the list of widget sources.
    a) In the **Format Text** slot, type in **Current User:\n%user**()% (that's a set of parentheses, not a 0!)
    b) Click **OK** to create the widget, then position the text where you'd like it to display on the view.

13. Drag the **Banner** folder onto the view. In the wizard, select **Bound Label**. Enter the b-format string **%displayName%**.
    a) Click the **Font** slot to open its text properties. Using the **Font** slot, change the font size and style to **Bold** and **24 pt**.
    b) Using the **Foreground** setting, change the font color (foreground) to a color you like.
    c) Resize the label: large enough to accommodate the name of a typical folder / object.
    d) Click **OK** to create the bound label widget, then double-click the widget to open its properties.
    e) In the section labeled **Bound Label Binding**, locate the **ORD** slot. Click in this slot to open the **ORD editor**.
        i. Edit the text here until there is only **slot:** (This "relativizes" this Bound Label and allows the Banner to display the name of any Px you use it with)
    f) Click **OK**, then **OK** to save your changes.
    g) Click the **Save** button on the Tool Bar to save your changes. (When you save, it's possible that the text disappears from the view)
14. Double-click the canvas pane to open its properties. Click the background slot and select a background color. Feel free to experiment with several different colors.
    a) You can also click the drop-down arrow on the right side of this slot and choose Gradient. Experiment with different color combinations! This is your design.



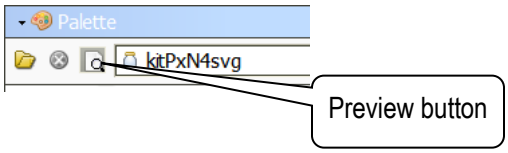
15. Click the **View / Edit Mode** button to save your changes and view your **BannerPX** view.

Note: This Banner will be a standard feature positioned at the top of all graphical views.

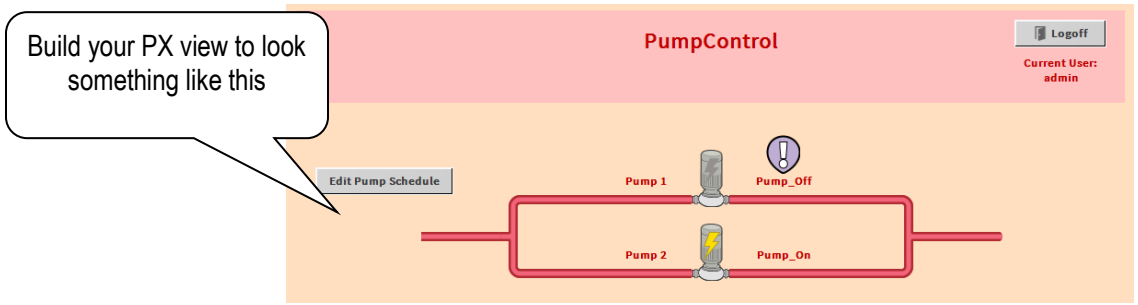
(*End of lab, SAVE YOUR STATION and **stop here**!*)

LAB #12: Creating a Pump Control PX View	25 min
Description:	In this lab you will continue to learn new tips and techniques for creating graphical views of a Niagara station using the PX Editor.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Include the Banner.px file into a new PumpControl.px view to display standard information in a predetermined position on the screen. 2. Create widgets using the collection found in the kitPxN4svg palette. 3. Create a hyperlink that allows the User to access a Weekly Schedule.

1. Right-click the **PumpControl** folder in the Nav SideBar. Choose **Views**, then **New View**.
2. In the new view wizard that opens, name the new view **PumpControlPX**. Click the **OK** button to create the view.
3. Right-click the **PumpControl** folder again, and select **Edit Tags**. From the **Custom** tag dictionary, add the **graphic** tag.
4. Double-click the view to open the **Properties** of the **Canvas Pane**. Set the **View Size** to 1100 x 900.
5. In the Nav SideBar, expand the **Files** container in the station, then expand the **PX** folder.
6. From the **PX** folder, drag and drop the **BannerPX** file onto the new **PX** view. Position the banner at the top of your **PumpControlPX** view. Toggle to the **View** mode to review what this looks like. Edit as you see fit.
7. In the Nav SideBar, locate the **HotWaterPump_1** point. Drag the point onto the PX view and drop it (make sure the view is in Edit mode). This will open the **Make Widget Wizard**.
8. In the wizard, select the **From Palette** option, then click the **Open Palette** button in the wizard. Choose the **kitPxN4svg** palette.
9. In the palette, expand the **Pumps** folder. Select the **PumpsInLine** selection and click **OK**.
10. Repeat this process to add **HotWaterPump_2** to the view.
11. In the Palette SideBar, open the palette of the **kitPxN4svg** module.
12. Unfold the **Piping** folder in the palette, continue drilling down into the **LargePiping** folder and the **Hws** folder. Add and arrange piping to the view to make it look similar to the view at the end of the lab.
    a) Hint! You can click the **Preview** button in the Palette SideBar to preview the piping.



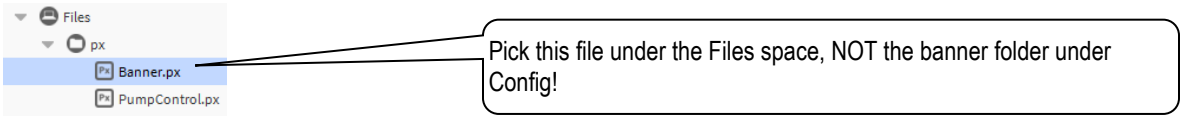
13. Open the **Edit** view (click the pencil on the tool bar) of this Px. In an empty spot on the Px view, right-click, choose **New**, then **Label**. There is no need to name the label.
    a) Double-click the new label to open its **Properties** view. Enter the text **Pump 1**.
    b) Edit the size and color as you see fit
    c) Repeat to add a label for pump 2
14. Drag **HotWaterPump_1** onto the **PX** editor. Choose **Bound Label** in the wizard. In the **Format Text** slot, type **%out.value%**. Click **OK**, then position the bound label on the view.
    a) Repeat to add a label for **HotWaterPump_2**
15. Open the properties of the canvas pane and set whatever background color you'd like to have. Remember you can set a gradient too if you can't decide on a single color, and remember that the canvas should work with the Banner background.
    a) Drag the **PumpSchedule** from the Nav SideBar onto the PX view edit window.
    b) In the Make Widget Wizard, choose **FromPalette**. Choose the **kitPX** palette
    c) Select the **HyperlinkButton** widget option, check the box labeled **Hyperlink** in the lower left corner of the wizard
    d) In the **Properties** displayed on the right side of the wizard, add text to the Text field that reflects the purpose of the button.
    e) Click **OK** to create the button.
    f) Switch to the **View** mode and test the button
16. Add a pump status indicator to the view
    a) Drag **HotWaterPump_1** onto the **PX** editor. Choose the **From Palette** source.
    b) In the **kitPxN4svg** palette, expand the **StatusIcons** folder and select the **Regular** option. Note: This in unseen unless the object is in Alarm / Override status, it will not show during running status.
    c) Click **OK** to create the widget.
    d) Repeat these steps adding the status of pump 2 to the PX view.
    e) In the **View** mode of the PX view, right-click **HotWaterPump_1**. Choose **Actions** and invoke a manual override on pump 1 (doesn't matter on or off). Note the status indicator becomes visible when the status of the point is in any status state other that **OK**. Icon would be red for alarm, purple for override, etc.



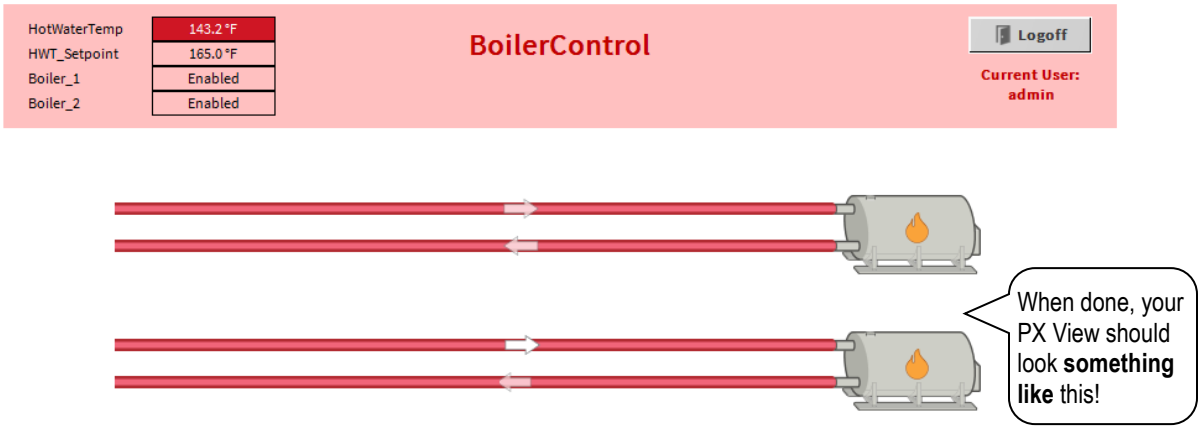
(*End of lab, SAVE YOUR STATION and **stop here**!*)

LAB #13: Boiler Control PX View	15 min
Description:	In this lab you will continue to learn new tips and techniques for creating graphical views of a Niagara station using the PX Editor.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create widgets using the collection found in the kitPxN4svg palette. 2. Create multiple widgets simultaneously. 3. Modify the visual properties of the widgets you create.

1. Right-click the **BoilerControl** folder. Choose **Views**, then **New View**. Create a new PX view called **BoilerControlPX**.
2. From the **Custom** tag dictionary, add the **graphic** tag to this object.
3. Edit the canvas pane to a size of 1100 x 900.
4. From the Nav SideBar, from the **BoilerControl** folder, drag out both **Boiler_1** and **Boiler_2** onto the PX editor simultaneously. Hint: Hold down the CNTRL button on your PC.
5. In the make widget wizard that opens, choose **From Palette**.
    a) open the **KitPxN4svg** palette
    b) Unfold the **Boilers** folder and choose the **BoilerRightRound** widget.
6. Add the **BannerPx** view to the top of the **BoilerControlPx** view. *Note: NOT the banner folder under config, you want the **BannerPx.px file** in the **px** folder of the **Files** space in your station!*



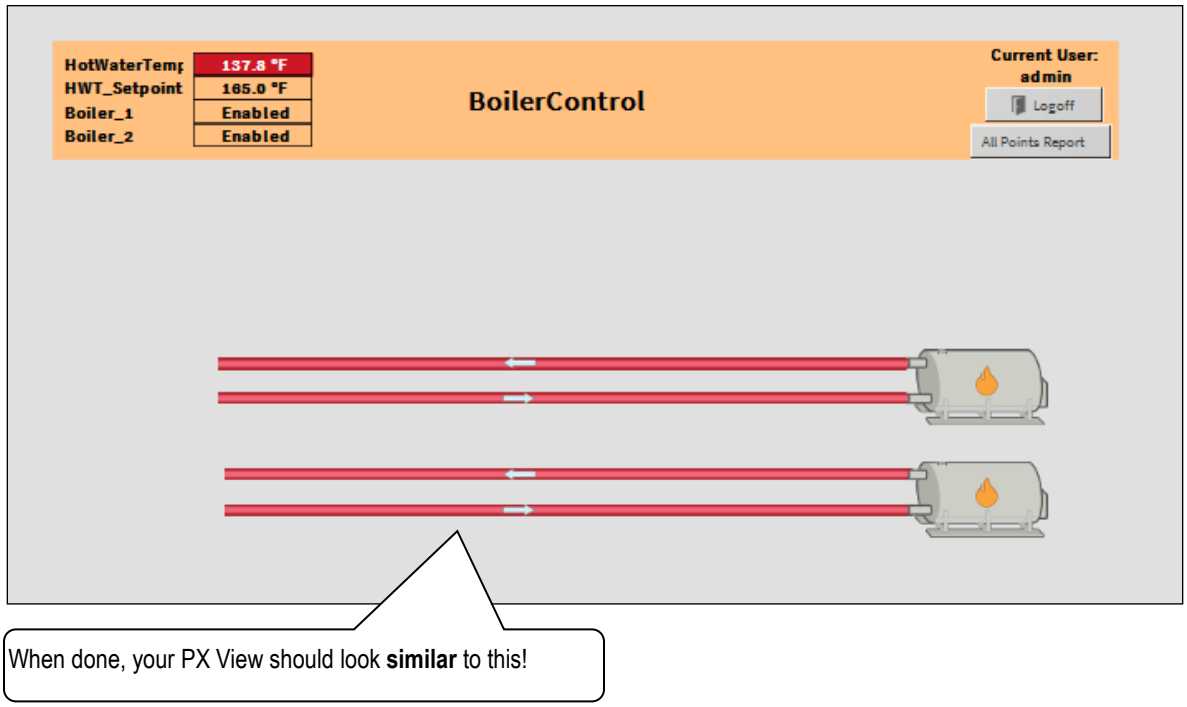
7. Add piping and Directional arrows from the **KitPxN4svg** palette to the view as you see fit to make the view look more finished.
8. Select these points in the Nav SideBar in this order: **HotWaterTemp**, **HWT_Setpoint**, **Boiler_1**, **Boiler_2**.
    a) Drag all four points onto the view at the same time.
    b) In the Make Widget Wizard, select **Bound Label** from the widget source list.
    c) Check ☑ **Format Text** and enter the b-format string **%out.value%**
    d) Check ☑ **Make Display Name Label**
    e) Check ☑ **Status** and select **Color** as the status option.
    f) Check ☑ **Border**
    g) Click **OK** to create these bound labels at the same time.
    h) Move the objects around as you see fit to make it look good!



(*If you have time, continue to the Optional Lab on the next page!*)

OPTIONAL LAB: Hyperlink to a Search Result	
Description:	In this lab you will continue to learn new tips and techniques for creating graphical views of a Niagara station using the PX Editor.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Practice standard Windows keystrokes, such as CNTL-C (copy) and CNTL-V (paste). 2. Practice the CNTL-L keystroke – a special Niagara keystroke that will yield the ORD for the current view. 3. Create a hyperlink to display the search results. 4. Practice invoking Actions on an object from a PX View.

1. Perform a search using the search parameters **n:point**.
2. Review the report that is built from the tag. While looking at the report, hold down the **Control** key on your PC keyboard and press the **L** key on the pc keyboard. You will see an ORD popup that represents the search parameters you define.
3. Hold the **Control** key again on your keyboard, and press the **C** key (this is a Copy command). After copying the ORD, click the **Cancel** button.
4. Under the Files then px open the **Edit** view of your PxInclude file (default view of the Banner folder).
5. From the Nav SideBar under config, drag the **Banner** folder onto the PX editor.
    a) In the Wizard that opens, choose **From Palette**. Choose the **kitPx** palette
    b) Choose the **Hyperlink Button** widget. Check the **Hyperlink** check box at the bottom of the wizard.
    c) In the Text field (right side of wizard view), type in All Points Report
    d) Click **OK** to create the button. Position the buttons on the right side to fit on the px view.
6. Double-click the new hyperlink button to open its properties.
7. Click the hyperlink slot to open its ORD editor
8. Hold the **Control** key on the keyboard while you press the **V** key on the keyboard (paste command). Click **OK** to save the change.
9. Toggle to the **View** mode of the PX view, click the button and observe the results.
10. Go to the Boiler or Pump Control Px view and try the new button.



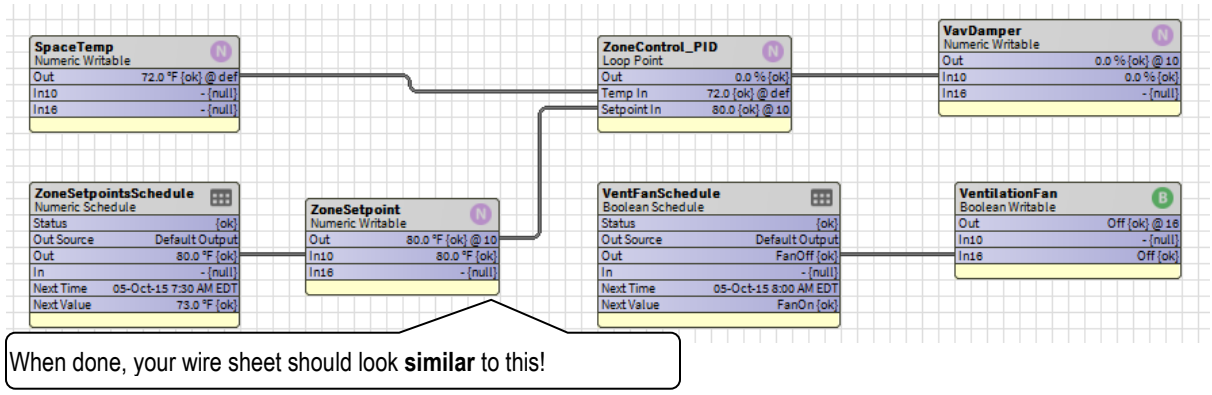
(*End of lab, SAVE YOUR STATION and **stop here**!*)

LAB #14: Zone Temp Control Loop	20 min
Description:	In this lab you will configure another closed loop control scheme for Zone Temperature control.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Configure control logic. 2. View and Understand Niagara Slot Details. 3. Set up schedule-based (numeric) setpoint control. 4. Set up schedule-based (boolean) control of a 1-speed fan.

1. In the **Control_Logic** folder, add a subfolder called **ZoneTempControl**. Open the wire sheet view.
2. Open the wire sheet and add a new **NumericWritable** point. Name it **SpaceTemp**.
   a) Edit the point facets to use °F (engineering units) to 1 decimal place of precision.
   b) Use the **Set** action to set a **default value** (fallback value) of 72°F.
   c) From the **Haystack** tag dictionary, add the **ZoneAirTempSensor** tag group to the point.
   d) From Haystack, add the tag **vavZone**.
3. Add to the wire sheet of the folder a **LoopPoint** object from the **kitControl** HVAC folder. Edit the **LoopPoint** object:
   a) Name the **LoopPoint** object **ZoneControl_PID**
   b) Pin open the **Controlled Variable** and **Setpoint** slots
   c) Set the **Proportional Constant** to 25
   d) Set the **Bias** setting to 50
   e) Set the **Action** setting to **Direct**
   f) Facets: Assign the engineering unit **%** (in the misc. drop down)
4. Return to the wire sheet of the **ZoneTempControl** folder. Right-click the **ZoneControl_PID** block. Choose **Views**, then **AX Slot Sheet**.
5. In the slot sheet, locate **the Controlled Variable** slot. Double-click this slot to open the **Display Name** editor view. Change the name of the slot to **Temp In**.
6. Using the **View Selector** in the upper right corner of the Workbench, switch to the **Property Sheet** view of the ZoneControl_PID object. This is the HTML5 property sheet view of the PID.
7. Locate the **Slot Details** 

button in the upper right corner of the **Property Sheet**. Click this button and note that it changes the view to display the slots in the Property Sheet view.
8. Locate the **Setpoint** slot and note that it displays in blue as it is a hyperlink. Click the **Setpoint** slot to open the **Display Name Editor**. Change the display name to **Setpoint In**.
9. Return to the wire sheet of the ZoneTempControl folder. Add a new **NumericWritable** point called **VavDamper**. Assign Facets: engineering unit = **%**.
10. Connect the **Out** slot of the ZoneControl_PID object to **In10** slot of the VavDamper object.
11. Connect the **Out** slot of the SpaceTemp point to the **Temp In** slot of the Loop Point.

12. Add to the wire sheet a **Numeric Schedule** (from the Schedules palette). Edit the schedule:
    a) Name the schedule **ZoneSetpointSchedule.**
    b) Set the facets to °F.
    c) Set the **Min** and **Max** values to 55°F and 85°F.
    d) Output a value of 80°F as the Default Value.
    e) Output a value of 73°F from 7:30 am to 5:00 pm Monday through Friday.
    f) Define a **Special Event** called **Holidays** that links to the **HolidayCalendar** schedule you created earlier in the class. Define the output as the unoccupied setpoint for all holidays.
13. Return to the wire sheet of the **ZoneTempControl** folder. Create a new **NumericWritable** point called **ZoneSetpoint.** (Will be used to override the setpoint defined in the schedule.)
    a) Assign a default value to this point of 73°F**.**
    b) Use the Recent History menu to assign to this point the same facets that you assigned to the schedule.
    c) From the **Haystack** tag dictionary, add the tag group **zoneAirTempEffectiveSp**
14. Connect the output of the schedule to **In10** on the **ZoneSetpoint** point.
15. Connect the output of the **ZoneSetpoint** to the **Setpoint In** slot on the **ZoneControl_PID** block.
16. Add a **BooleanWritable** point to the wire sheet. Call the new point **VentilationFan**.
    a) Set its facets to read **On** for the True state and **Off** for the False state.
    b) Assign the **Haystack** tags **equip** and **fan.**
17. Add a **BooleanSchedule** to the wire sheet. Call the schedule **VentFanSchedule**.
    a) Configure the schedule to output the state **FanOn** from 8:00 am until 5:00 pm Monday - Friday, and output the state **FanOff** at all other times (set up your facets accordingly).
    b) Configure a special event in this schedule that references the **HolidayCalendar** schedule we created earlier in class. Define the output as the unoccupied state for all holidays.
18. Connect the output of this new schedule to the **In16** slot on the **VentilationFan** point.



(*End of lab, SAVE YOUR STATION and **stop here**!*)

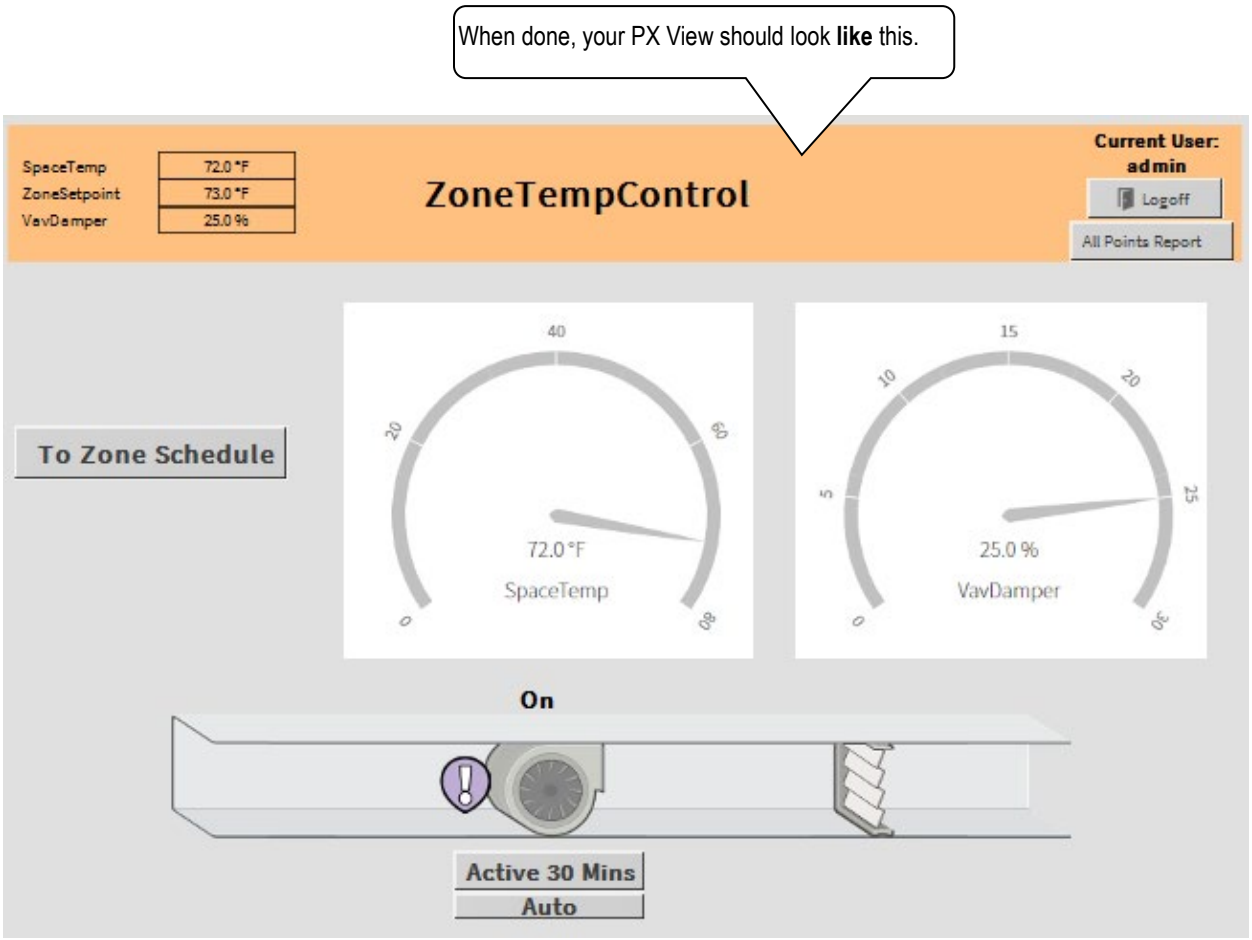
LAB #15: Zone Temp Control PX View	20 min
Description:	In this lab you will continue to learn new tips and techniques for creating graphical views of a Niagara station using the PX Editor.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Explore the contents of the webChart Palette. 2. Configure the CircularGauge widget 3. Configure Action properties associated with graphical representations of commandable data points. 4. Configure Action Widgets.

1. Right-click the ZoneTempControl folder. Choose Views, the New View. Call the new view **ZoneControlPx**. Resize, set colors and add the Banner.px like instructed in previous labs.
2. From the **Custom** tag dictionary, add the **graphic** tag to this object.
3. Add to the view a hyperlink to the schedule that controls the zone setpoint. (Use the from Palette kitPx)
4. Select the points in the SideBar in this order, **SpaceTemp**, **ZoneSetpoint**, **VavDamper**. Add them to the px view as bound labels, selecting **%out.value%, Make a Display Name Label**, **border**, **status** (color). Click **OK** to create the widgets.
5. Drag the **Space Temp** point again and add it to the px view. Choose **From Palette**.
6. Choose the **webChart** palette, then select the **CircularGauge** option. Click **OK** to add the widget. Resize as necessary.
7. Add the **VavDamper** point to the view using the same circular gauge option.
8. Add these widgets from the kitPxN4svg palette's Ductwork folder: **DuctHorzLong**, **DuctHorzEndLeft**, **DuctHorzEndRight**. Position as needed on the view.
9. Drag the **VentilationFan** onto the wire sheet. Choose **From Palette** in the wizard.
10. Select the **kitPxN4svg** palette. Link this point to the **FanHorzRight** widget in the **Fans / FansHorz** folder.
11. Drag the **VavDamper** point out again. Link it to the **DamperHorzParallel** widget from the **Dampers** folder of **kitPxN4svg** palette. Position on the ductwork.
12. Drag the **Ventilation Fan** point out again. In the wizard choose **Bound Label**. Set the text to read **Fan is %out.value%**.
13. Drag the **VentilationFan** point out again. This time choose **Actions** in the wizard.
    a) In the **Actions** menu that appears, choose **Active** and **Auto** at the same time. (Hold CTRL click one then the other.)
    b) Click **OK** to create the two new buttons.
14. We want to configure the button so that clicking the button causes the fan to run for 30 minutes, then it goes back to following the schedule command. To do this:
    a) Once the buttons are created, double-click the **Active** button.
    b) In the text slot, replace the word Active with this string: **Run Fan for 30 Minutes**
    c) Locate the slot called **actionArg**. Click in this slot to open the editor.

d) In the Action Arg Editor, deselect the **Prompt User** check box. Set the duration to **30 minutes**. Click **OK** to save your settings.

15. Drag the **VentilationFan** onto the view again. In the wizard choose **From Palette**. Use the **kitPxN4svg** palette.

a) Choose the **Regular** widget from the **StatusIcons** folder. Click **OK** to add the indicator.



(*End of lab, SAVE YOUR STATION and **stop here**!*)

LAB #16: BACnet Integration	20 min
Description:	In this lab you will learn how to configure a BACnet network using the IP Port, and will add a connection to a BACnet simulator that will be used in this class to simulate plant equipment.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Verify that a Niagara platform is BACnet capable. 2. Create and Configure a BACnet Network. 3. Discover active ports and Add BACnet Devices.

BACnet Remote Input / Output Integration

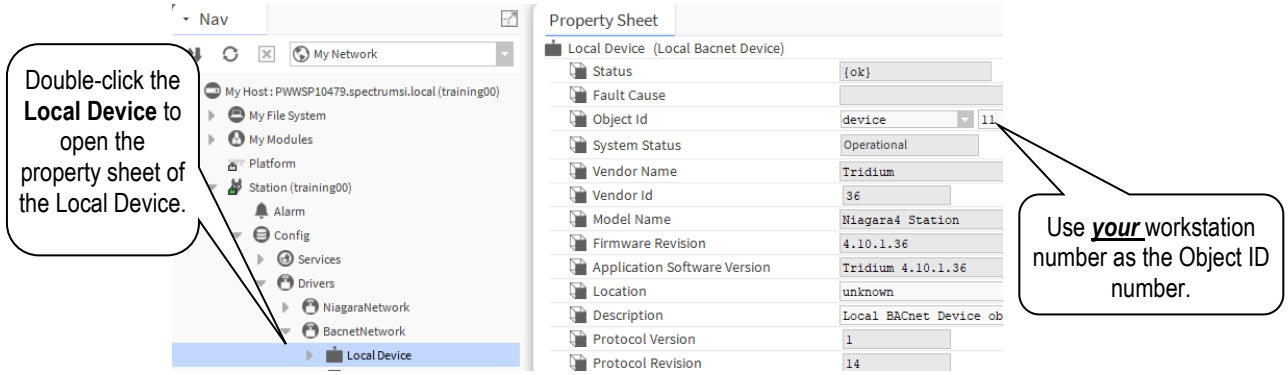
Part 1, Add a BACnet Network to the station

1. In the Palette SideBar, open the **Bacnet palette.**
2. Locate and select the **BacnetNetwork** object.
3. Expand the station **Config** component in the Nav SideBar and locate the **Drivers** container. (*The Drivers container should already contain the Niagara Network)*
4. Drag a **BacnetNetwork** out of the Bacnet palette, and drop it onto the **Drivers** component. Leave the name as is. (*Note: The Bacnet Network could also be added to your station by opening the Driver Manager window, then selecting the **New** button at the bottom of the window*)
5. Expand the Bacnet Network in the Nav Tree.

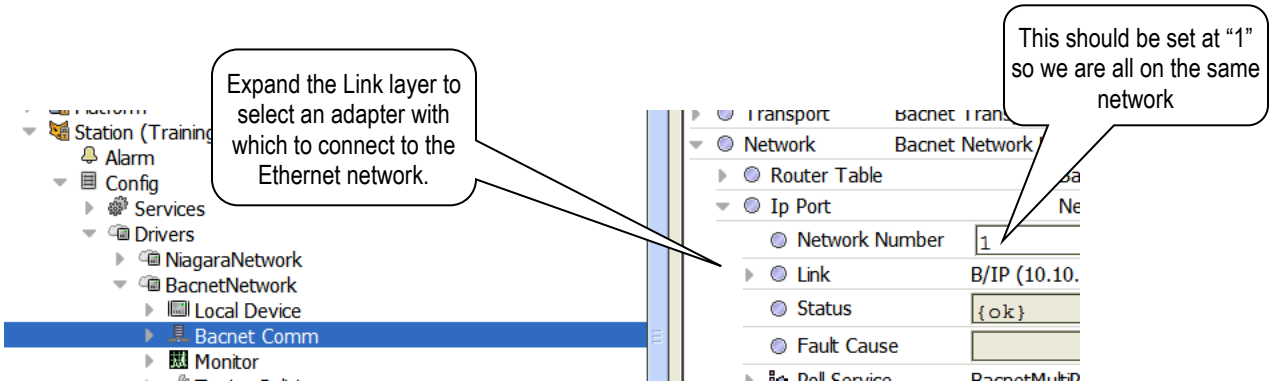
Part 2, Configure the BACnet Network for BACnet over IP

Note: Perform these steps in sequence. **SAVE after each configuration change.**

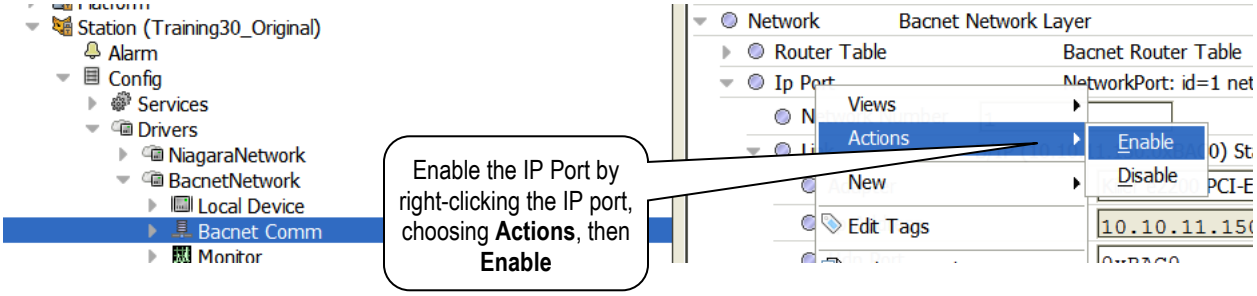
1. Configure the Bacnet Local Device to use your Workstation number as the Object ID number. To do this;
   a) In Nav SideBar, expand the view of the **Bacnet** network, then double-click **Local Device**.
   b) Set the **Object ID** number to your workstation number, then save your changes.



2. In the Nav SideBar, double-click the **Bacnet Comm** object, then expand the **Network** layer. Expand the **Ip Port** property. Verify the network is set to number 1. *Save any changes.*



3. Expand the **Link** layer. Verify that an appropriate **Network Adapter** has been selected. Your PC's network interface adapter should have been selected automatically. If not, select the adapter (connecting the PC to the Ethernet network) from the menu and save your change.
4. Verify that the correct IP Address is displayed. Collapse the Link layer.
5. Enable the IP Port by right-clicking the IP Port, then choose **Actions**, and **Enable**. Verify that the IP Port is Enabled! (*Port status should report OK*)



Note: You can also simply reconfigure the Enabled property, setting it to True.

Part 3, Discover and Learn the BACnet Devices

Note: In this exercise you will add one Bacnet Device (**BACnet_RIO_25)** to your station database.

1. Double-Click the **BacnetNetwork** to open the **AX Bacnet Device Manager**.
2. Click the "**Discover**" button below to discover all Bacnet devices on the network. Verify that the 1 is checked in the Networks of the Configure Device Discovery dialog box. Click the OK Button.
3. Find the newly discovered BACnet_RIO_25 controller in the Discovered list. Click on and drag the controller from the Discovered list to the Database list. *(You could have also selected the controller and then clicked the Add button.)* Click OK on the resulting dialog box to complete the addition. Verify that the device appears in the Database.

(End of lab, save station and stop here)

LAB #17: Create Proxy Points from a BACnet Controller	20 min
Description:	In this lab you will expand your integration skills by adding and configuring proxy points from your field controller. These points are for display of the real time status of equipment that is directly controlled by a smart device.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Open the AX Bacnet Point Manager and Discover potential proxy points. 2. Create a Bacnet point folder. 3. Add BACnet proxy points within a BACnet Point Folder with the desired point types and facets. 4. Apply Tags to the proxy points with the Point Manger in Tag Mode.

1. Expand the view of the **BACnet_RIO_25** device in the Nav SideBar.
2. Double-click the **Points** folder in the BACnet controller.
3. In the AX **BACnet Point Manager** view, create a new **BACnet Point Folder** by clicking the **New Folder** button at the bottom of the BACnet Point Manager view (or drag a new **BacnetPointFolder** out from inside of the **Points** folder in the **Bacnet Palette**). Name the folder **HeatingPump_IO**.
4. Double-Click the **HeatingPump_IO** BACnet Points Folder just created to view its **BACnet Point Manager** view.
5. Click the **Discover** button to discover all available points in the controller.
6. In the Discovered window, locate the points that include **your workstation number**. These points will include your workstation number in their name. It is important that you add only the points that include your workstation number.
7. Add points to your BACnet point folder in accordance with the following chart. **Be careful to select a point type** that is consistent with the intended use of the point in an operator's interface (FOLLOW THE CHART!!). **Make sure all points are enabled**.

Note: XX represents your workstation number!

Pay attention to the point type!

Network Point Name	Name the new proxy point: New Point Name	Point Type		Haystack tags.
StationLXX.AI01	**OATemp**	**Numeric Point**	°F	outsideAirTemp Sensor
StationLXX.BI01	**HWP1_Status**	**Boolean Point**	True text = On False text = Off	heating, pump
StationLXX.BI02	**HWP2_Status**	**Boolean Point**	True text = On False text = Off	heating, pump
StationLXX.BO01	**HWP1_Enable**	**Boolean Writable**	True text = On False text = Off	heating, pump
StationLXX.BO02	**HWP2_Enable**	**Boolean Writable**	True text = On False text = Off	heating, pump

8. Assign a default value (fallback) of **Off** to the pump enable points.
9. In the **BACnet Point Manager**, add another **BACnet Point Folder** called **BoilerControl_IO**.
10. To this new point folder, add points according to the following table. **Set the point type correctly** when creating the points as you cannot edit this choice after the point is created!

Network Point Name	Name the new proxy point: New Point Name	Point Type	Facets	Apply these tags from the Haystack tag dictionary
StationLXX.AI02	**BoilerSupplyTemp**	Numeric Point	**°C** [°F]	boiler, sensor, heating, temp
StationLXX.AI04	**BoilerReturnTemp**	Numeric Point	**°C** [°F]	boiler, sensor heating, temp
StationLXX.BO03	**Boiler1_Enable**	Boolean Writable	**True text = On False text = Off**	steamPlant, boiler, heating
StationLXX.BO04	**Boiler2_Enable**	Boolean Writable	**True text = On False text = Off**	steamPlant, boiler, heating

11. In the **BACnet Point Manager**, add another **BACnet Point Folder** called **ZoneControl_IO**.
12. To this new point folder, add points according to the following table. Remember to set the point type correctly when creating the points as you cannot edit this choice after the point is created!

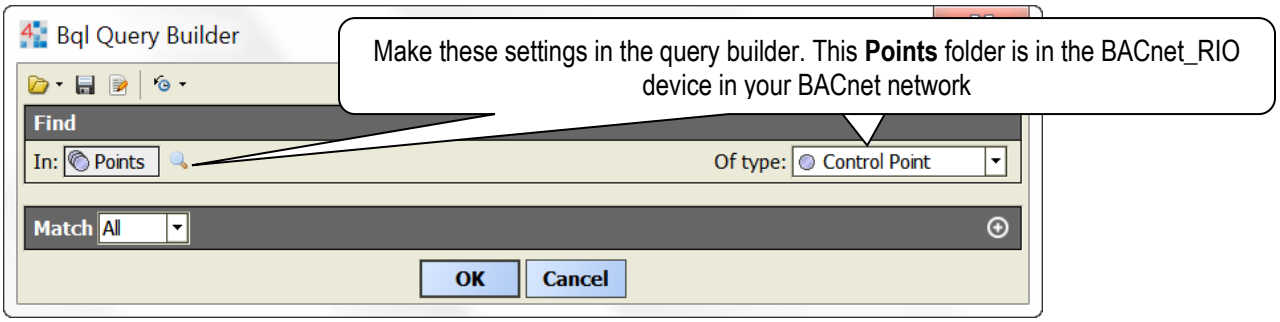
Network Point Name	Name the new proxy point: New Point Name	Point Type	Facets	Apply these tags from the Haystack tag dictionary
StationLXX.AO01	**ZoneDmpr**	Numeric Writable	**%** to 1 decimal (look in Misc folder for %)	hvac, vavZone , equip
StationLXX.AI03	**ZoneTemp**	Numeric Point	**°C** [°F]	hvac, equip From Haystack's tag group, add **zoneAirTempSensor**
StationLXX.BO05	**ZoneFan**	Boolean Writable	**True text = On False text = Off**	hvac, equip,

(End of lab, SAVE YOUR STATION and stop here!)

LAB #18: Tagging Objects using the Batch Editor	10 min
Description:	In this lab you will learn how to apply semantic tags to multiple Niagara objects with the use of the Program Service.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Back Up your Niagara station for disaster recovery. 2. Open the Batch Editor and Construct a simple Query. 3. Add a Tag to a group of Objects in batch fashion.

CAUTION: The Program Service provides a batch editor. This batch editor is powerful but unforgiving and there is no undo option. You are encouraged to Backup your station before using the Program Service in case the results from using the batch editor cause you to lose some data. This will allow you to restore the station, using the Distribution File Installer, in the event of an error.

1. Unfold the **Services** container in the Nav SideBar.
2. Double-click the **Program Service**. Note the default view is the **Batch Editor.**
3. Click the **Find Objects** button.
4. In the **BQL Query Builder**, restrict the scope of the query to only those objects that are in the **Points** container of the BACnet_RIO_25 device in the BACnet network.



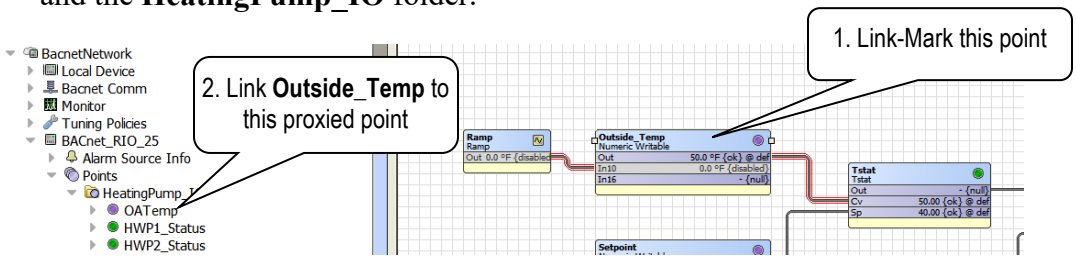
5. Click the **OK** button to execute the search.
6. Review the list of objects that were found in the search. **Note: Any editing you do will be applied to ALL the points / objects that are in this window, NOT just selected items!**
7. At this time you will need to select any objects which you DO NOT want to apply the tag (one at a time or all at once). Right-click the unwanted points and choose **Clear Selected Item.** In this case we do not need to clear any items. The tags will identify these objects as proxied points.
8. Once you have only the points you want in the **Object** window, click the **Add Tag** button.
9. Select the **proxiedPoint** Marker tag from the **CustomDictionary**.
   a) Click the **Add Tag** button at the bottom of the dialog box.
10. **Save** your changes, then review the results window that displays.
11. In the Nav SideBar, select any point in any BACnet point folder. Right-click the point and choose **Edit Tags**. Note the addition of the **proxiedPoint** tag to the object.

(*End of lab, SAVE YOUR STATION and **stop here**!*)

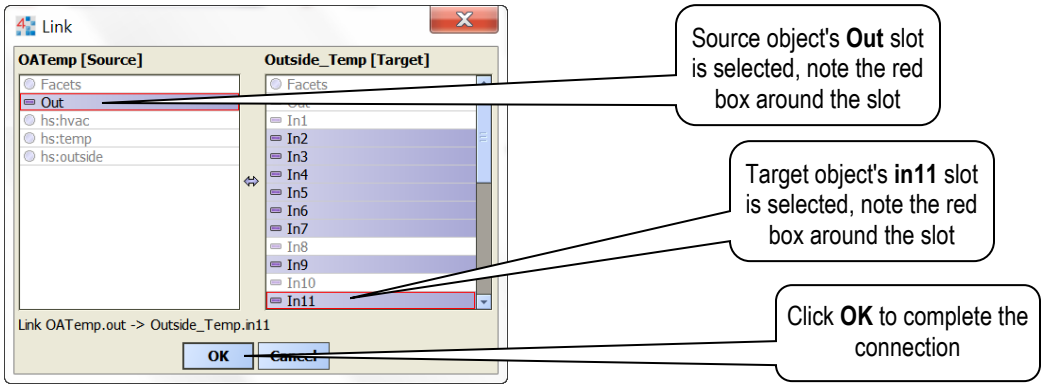
This page is intentionally left blank!

LAB #19: Linking Proxied Points to Control Logic	15 min
Description:	In this lab you will practice the technique of linking objects together using the Link Editor. This process, called Link Mark, allows you to link two objects even if they exist in different locations within the station architecture.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Initiate a Link Mark. 2. Explain the difference between the Source object and the Target object during the link process. 3. Link To a Target object from the Source Object. 4. Link From a Source object to the Target object. 5. Inspect a Link Knob to understand the nature of the link.

1. Open the wire sheet of the **PumpControl** folder. Right-click the **Outside_Temp** point and choose **Link Mark**.
2. In the Nav SideBar, expand the Bacnet network, the **BACnet_RIO_25** device, the **Points** folder, and the **HeatingPump_IO** folder.



3. Inside the **HeatingPump_IO** folder, locate the **OATemp** point. (Remember that we have already marked an object. Niagara is now waiting for us to specify how to complete the link.) Right-click the point and choose **Link To "Outside_Temp"**. The Link Editor opens.
   a) select **out** as the source slot in the **OATemp** point.
   b) select the **in11** slot as the target slot in the **Outside_Temp** point.



4. After making the connection, note the **Out** slot of the **Outside_Temp** point on the wire sheet. The **Out** slot value is still following the value output by the **Ramp** object (at slot in10).
   a) Open the property sheet of the **Ramp** object. Set the **Enabled** setting to **false**.

b) Return to the wire sheet of the **PumpControl** folder and examine the point's out value. It should now be following the input at **in11**.

5. With the PumpControl wire sheet still in view, use Link Mark to link to the objects that are accessible on the Nav Tree view of the **HeatingPump_IO** folder.

Link Mark this point / object in the PumpControl wire sheet view:	Right-click this point in the HeatingPump_IO container (Nav SideBar)	Slots to use:
HotWaterPump_1 (source)	**HWP1_Enable** (target)	**Out** to **In10**
HotWaterPump_2 (source)	**HWP2_Enable** (target)	**Out** to **In10**
PumpStatusOR (target)	**HWP1_Status** (source)	**Out** to **In A** Pump_1_Status_In
PumpStatusOR (target)	**HWP2_Status** (source)	**Out** to **In B** Pump_2_Status_In

6. Open the wire sheet of the **BoilerControl** folder in the station. Use Link Mark to link to the objects that are accessible on the Nav Tree view of the **BoilerControl_IO** folder.

Link Mark this point in the BoilerControl wire sheet:	Right-click this point in the BoilerControl_IO container (Nav SideBar)	Slots to use:
HotWaterTemp (target)	**BoilerSupplyTemp** (source)	**Out** to **In11**(Remember to disable the SineWave)
Boiler_1 (source)	**Boiler1_Enable** (target)	**Out** to **In10**
Boiler_2 (source)	**Boiler2_Enable** (target)	**Out** to **In10**

7. Open the wire sheet of the **ZoneTempControl** folder in the station. Use Link Mark to link to the objects that are accessible on the Nav Tree view of the **ZoneControl_IO** folder.

Link Mark this point in the ZoneTempControl wire sheet:	Right-click this point in the ZoneControl_IO container (Nav SideBar)	Slots to use:
SpaceTemp (target)	**ZoneTemp** (source)	**Out** to **In10**
VavDamper (source)	**ZoneDmpr** (target)	**Out** to **In10**
VentilationFan (source)	**ZoneFan** (target)	**Out** to **In10**

(*End of lab, SAVE YOUR STATION and **stop here**!*)

LAB #20: Commissioning the JACE	20 min
Description:	In this lab you will transfer the station that you have created using the Workbench to the JACE controller that is mounted on your remote Simulator Panel. From this point forward you will be working in a JACE that is connected to all of the smart device networks.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Open a Platform Connection to the JACE. 2. Configure the Commissioning Wizard. 3. Perform a Commissioning of the JACE. 4. Establish conventional Usernames, Passwords, and System Passphrases. 5. Verify that the Station is running on the JACE. 6. Open a Station Connection to the running station. 7. Verify and Tune platform Resources.

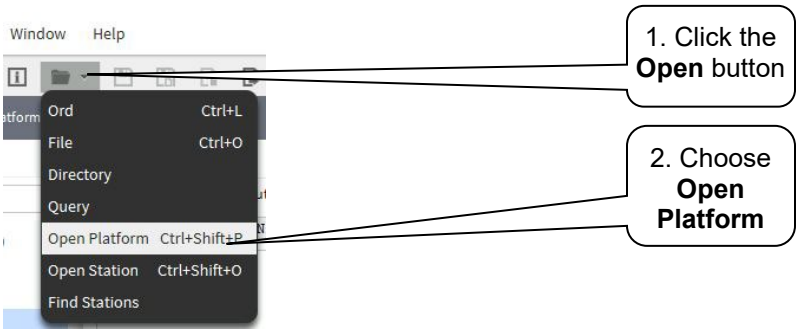
Do these steps EXACTLY as defined! DON'T start over or stop once the process starts!

Ask For Help

1. Save your station, then **close the station** that is running on your PC by right-clicking the station icon in the Nav SideBar and choosing **Close**. The station is still running!
2. Open a connection to your remote PC's platform, then open the **Application Director**. **Stop the station that is running on your remote PC.**
3. Open the remote PC platform's **Station Copier** tool (we need to copy the station from Platform Home to User Home).
4. In the right column (Platform Daemon User Home) select your station. Click the **Copy** button. As the station begins to copy, rename it to **TrainingXX_HalfWay**. Copy it to the left column (User Home). Close the Station Copier view when the copy completes.

> Context: At this stage in the exercise, assume that you open a package and remove a brand new, factory default, JACE controller. Before placing the new controller into stock (for new sales or ready spares) you want to power up the JACE and verify it is functional.

5. Initiate (Open) a secure Platform connection to the JACE:
   a) Click the **Open** button, choose **Open Platform**.



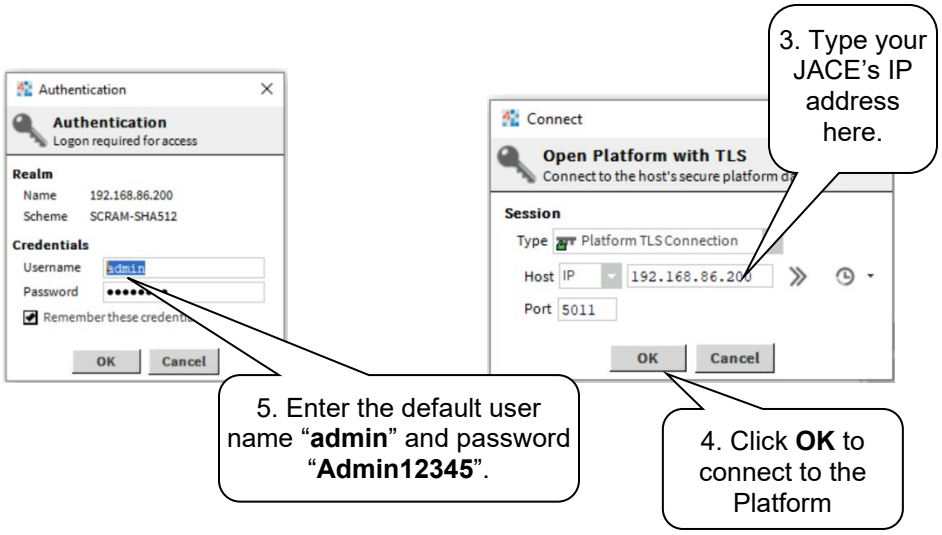
b) Type in the JACE's IP address. Note: The last two digits of the JACE's IP address are your Workstation number. Example, if the IP Address of your JACE is 192.168.86.2xx, then "xx" is your workstation number.

c) If prompted to verify a security certificate, click **OK** to **accept** the certificate.

d) Username = **admin**. Classroom

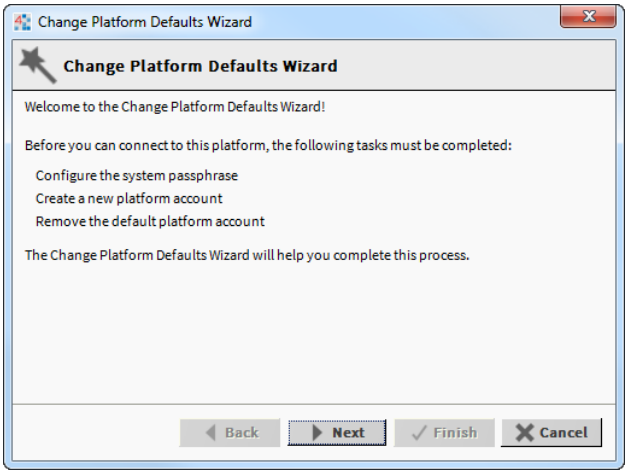
e) Default Password = **Admin12345**.

f) Click **OK** to establish the connection.

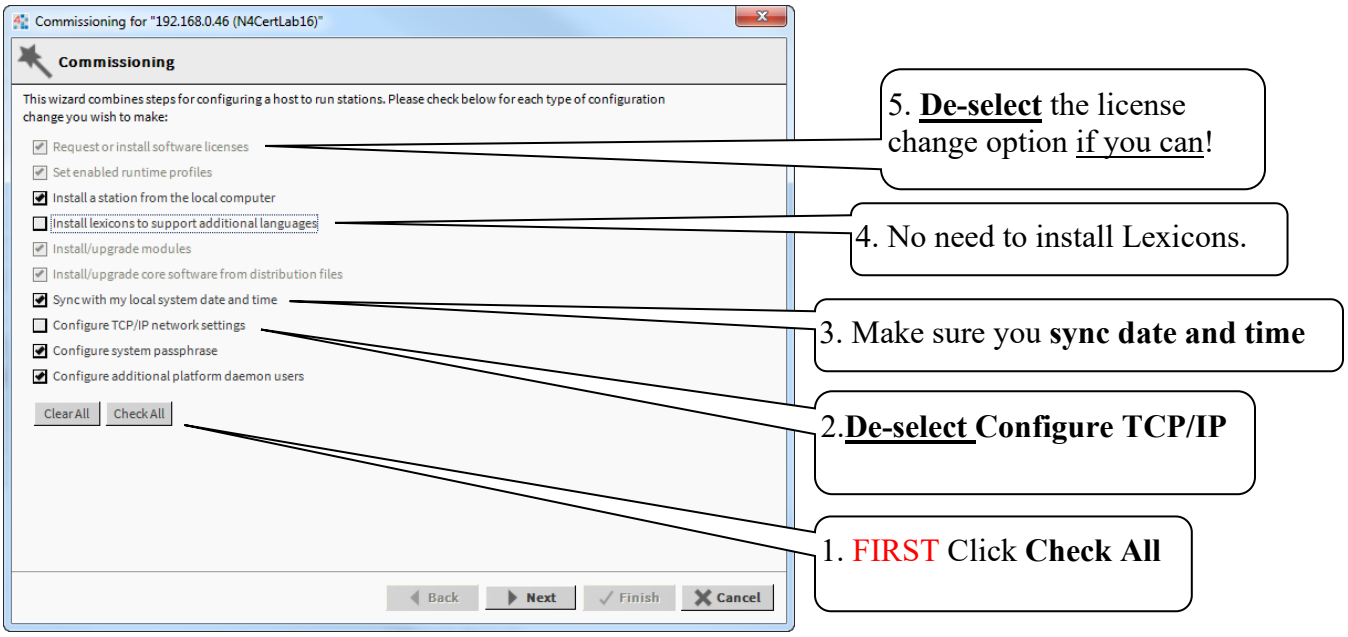


Note: If the JACE already has a Niagara4 station running on it, you may not be forced to change the platform defaults. If the JACE is in factory default condition, or has been cleaned, the Change Platform Defaults Wizard appears.

Additional Note: The window of opportunity to configure a system passphrase is time limited. If it times out (SystemPw:Failed to check system password) simply cancel and repeat the Step.



6. If your JACE asks you to change the default passphrase let the teacher know. They will use your station as a learning opportunity.
7. In the list of platform views, double-click **Platform Administration** to access those tools.
8. In the platform administration window, click the **Commissioning** button.
9. When the Commissioning Wizard opens, modify the selections to resemble this picture:



10. Click the **Next** button. **If asked**, **Select *Don't Change Any Licenses***, then click **Next.** If prompted to install a **certificate**, check the box to allow the install.
11. In the **Enabled Runtime Profiles** window, select **UX** and **WB**. Click **Next**.
12. When prompted to choose a station to install on the JACE, choose **TrainingXX_HalfWay**.
    a) **Rename** the station to **TrainingXX** in this process. Verify the station is set to **Auto Start** and **Start After Install**.
    b) If asked for a passphrase, type **Admin12345**. This was the PC's system passphrase at the time you copied the station from Daemon Home to User Home on your local host. If that doesn't work, ask your instructor for assistance!
13. When prompted, use the **Copy Every File in the Station Directory and Subdirectories** option. If this option is not available, use the **Copy files from selected directories** option, and plan to leave all directories selected in the following step.

> Note: **Lexicons** are used to allow individuals who speak a language other than English to view Workbench in their native tongue. We will not use Lexicons in this course, but you should feel free to install lexicons if you would like.

14. If there is already a station on the JACE a warning will let you know you are about to an existing station. This is ok. Click next.
15. The wizard analyses the station and schedules the installation of modules that are required to support the station as well as the platform.
16. There are other modules that must be copied to the JACE to support the project, and you will need to identify these modules. Click **Next** when ready to proceed.

a) You should select these additional modules if they are not selected automatically:

i. dashboard-rt (notice when you select the **rt** option, **wb** is also auto-selected)
ii. email-rt
iii. exportTags-rt
iv. kitPxHvac-wb
v. lonHoneywell-rt
vi. lonworks-rt

These you'll need to select yourself. As you select these modules, remember that dependent modules are selected automatically!

17. In the **Distribution File Installation** view, there is nothing to do, just click the **Next** button.
18. If you see the Re-install the core software check box, **don't**. Click **Next** if you see this view**.**
19. If you see the **Configure TCP / IP Configuration** view, **don't** change anything!! Click **Next** if you see this view.

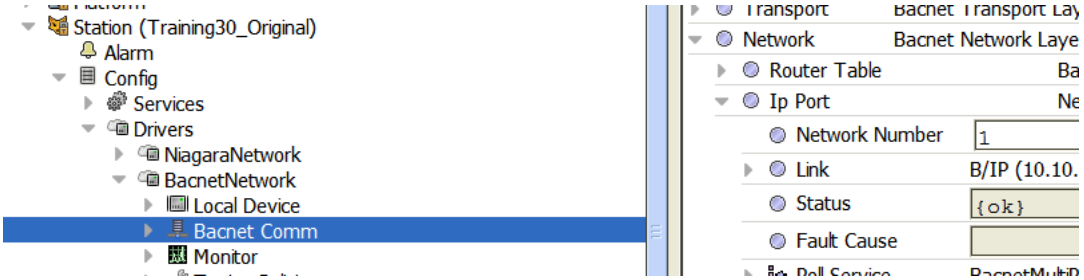
Note: All passwords and passphrases are case sensitive.

20. In the **System Passphrase** window that you see next:
    a) Type in **Admin12345** as the **Current Passphrase** (Note: This may not be necessary if required in an earlier step.)
    b) Set the **New Passphrase** to **Admin12345**
    c) Click **Next** to move on
21. In the next window you can define an additional user account for the platform
    a) Click the **New User** button
    b) Name the new user **TheOwner**
    c) Assign this user the password **TheOwner123**
    d) Click **OK** to create the new user
22. After clicking the **Next** button, you should see the final view titled **Review Changes**.
    a) Click the **Finish** button if you don't need to make changes.
23. Once the commissioning process is complete, wait two minutes for the JACE to reboot. Attempt to reconnect the platform. It could take as long as 5 minutes for the JACE to boot up.
    a) Reestablish a platform connection to the JACE. Remember the new platform username is **admin** and the new password is **Admin12345.**
    b) Open the **Application Director** and verify the station is starting. Once the station is running, connect to the station by right clicking the JACE's IP address in the Nav SideBar and choosing **Open Station**.
    c) Accept the self-signed security certification if the option is presented by clicking the **Accept** button. This default certificate originates with Tridium, a known and trusted entity, and its use is acceptable in a training situation.

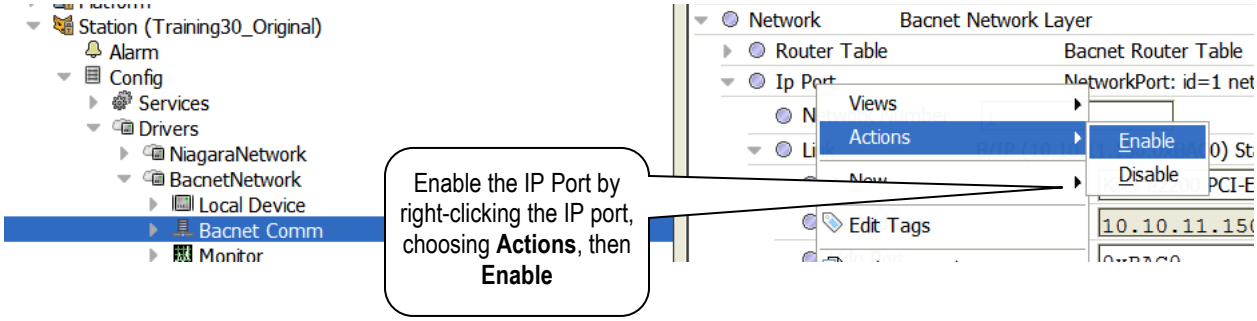
    Note: On a real job site you should think twice before using non-secure connections based on default certificates. It is better not to get into this habit. The best practice is to use secure connections with server certificates for which a chain of trust can be established by one of the larger, well known Certificate Authorities.

    d) Establish a secure connection (TLS) to the station. Remember the user name is **admin**, and the password is **Admin12345**.

24. Right-click the station, and select (click on) **Spy**, then **util**, then **gc**.
25. Right-click the station in the Nav SideBar. Choose **Views**, then **Resource Manager**.
26. Note the **CPU Usage** display. This should be less than 80% on a continuous basis.
27. Note the **heap.used** display value. On a JACE, this should be less than 75% of the **heap.max** value (you won't see heap.max when looking at a station running on a PC.
28. You may not need to re-enable the **BACnet** network adapter, but you should verify it. To do this, navigate into the **Bacnet Comm** component in the BACnet Network of your station.



27. Expand the **Link** slot container. Select the appropriate **Network Adapter.** We'll choose **en0.** **Save the change.**
28. Verify that the correct IP Address is displayed.
29. Enable the IP Port by right-clicking the IP Port, then choose **Actions**, then **Enable**. Verify that the IP Port is Enabled! (*Port status should report OK*)



30. In the BACnet Device Manager, verify that your station is now talking to the BACnet device. You

(*End of lab, stop here*)

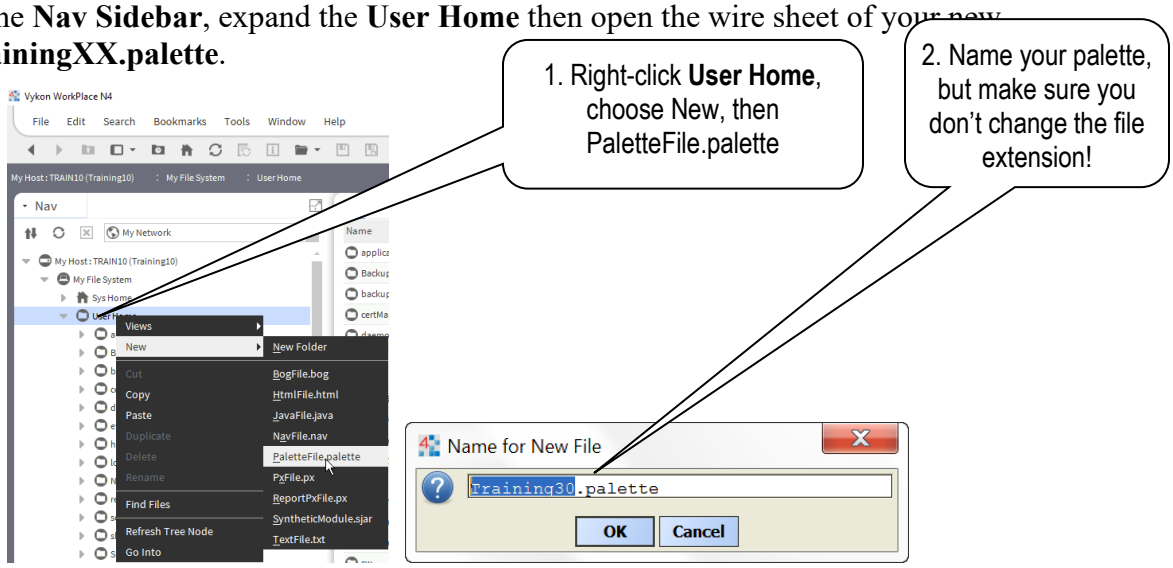
This page is intentionally left blank!

LAB #21: Customization Lab	30 min
Description:	In this lab, you will explore several ways to customize the Workbench user interface, including creating a personal palette (allowing you to create your own library of objects) as well as customizing the New components menu.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create a personal palette under the User Home. 2. Add objects to your personal palette. 3. Add extensions and folders to your personal palette. 4. Open your palette in the palette sidebar. 5. Customize the New Components menu. 6. Create.

Part 1, Creating a Personal Palette

Procedure 1, Creating the palette. This palette will be available for use throughout the class. Feel free to return to the palette file later and add any items you think may be useful.

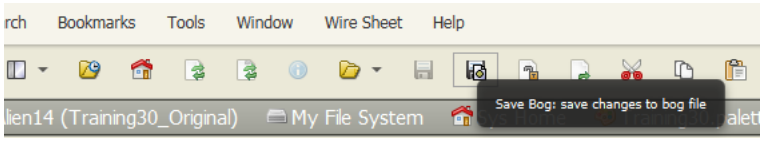
1. In the Nav Sidebar, Expand My Host My File System and locate **User Home**
2. Right click on the **User Home** node and select **New**, then **PaletteFile.palette**.
3. Rename the file **Training*XX*.palette** where *XX* is your workstation number.
4. In the **Nav Sidebar**, expand the **User Home** then open the wire sheet of your new **TrainingXX.palette**.



Procedure 2, Adding objects

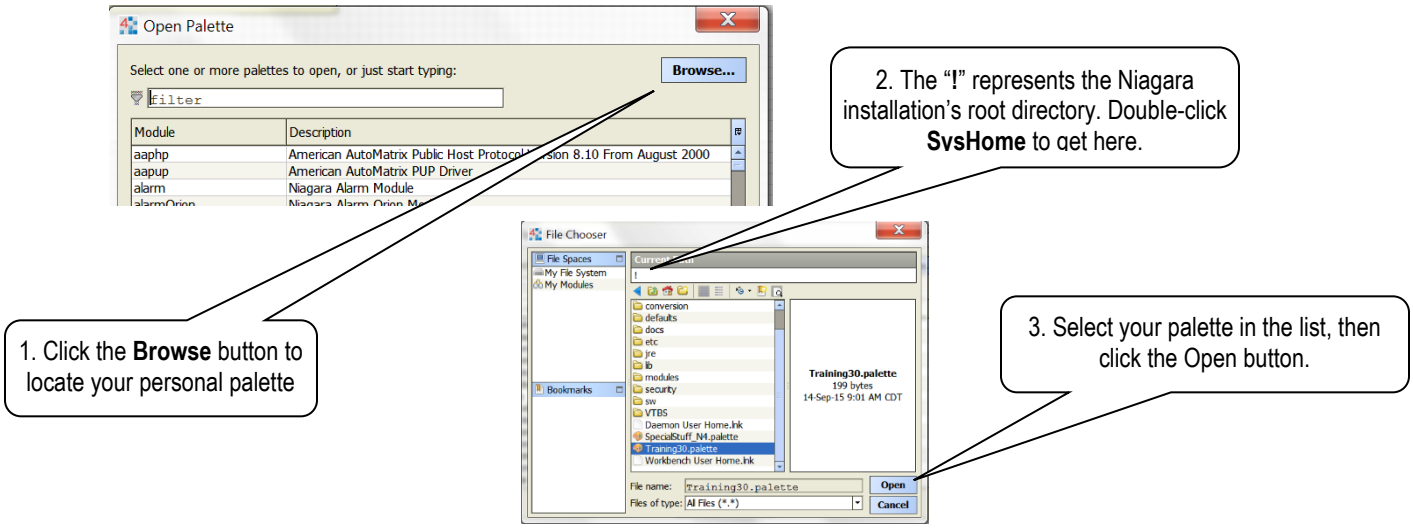
1. Right-click in the wire sheet of your personal palette, choose **New**, then **Folder**. Create a new folder called **kitControl**. Open the wire sheet of the new folder by double-clicking the folder.
2. Open the **kitControl** palette in the Palette SideBar. Select all the folders in the kitControl palette, control container. Drag the folders onto the open **Training*XX*.palette's** kitControl wire sheet.
3. Re-open your personal palette's wire sheet.

4. In the Palette SideBar, open the **Alarm** palette (click the Open Palette button, choose the Alarm module).
5. Drag the **Extensions** folder onto the **Training*XX*.palette** wire sheet.
6. Rename the folder **AlarmExtensions**.
7. Create a new folder in the personal palette wire sheet called Schedules. Open the wire sheet of this folder.
    a) Add to this folder the **PumpSchedule** for the **PumpControl** folder
    b) The **HolidayCalendar** schedule from the **PumpControl** folder
    c) The **ZoneSetpoint** schedule from the **ZoneTempControl** folder
8. Return to the wire sheet of your personal palette. From your class station in the Nav SideBar, drag the **PumpControl** folder into your personal palette.
    a) Open the wire sheet view of the folder you just added and note that the links to the proxied points have been lost
9. Return to the wire sheet of your personal palette. Add the **Boiler Control** folder to the wire sheet.
10. Save the Palette by clicking the **Save Bog** button.



Completed Palette

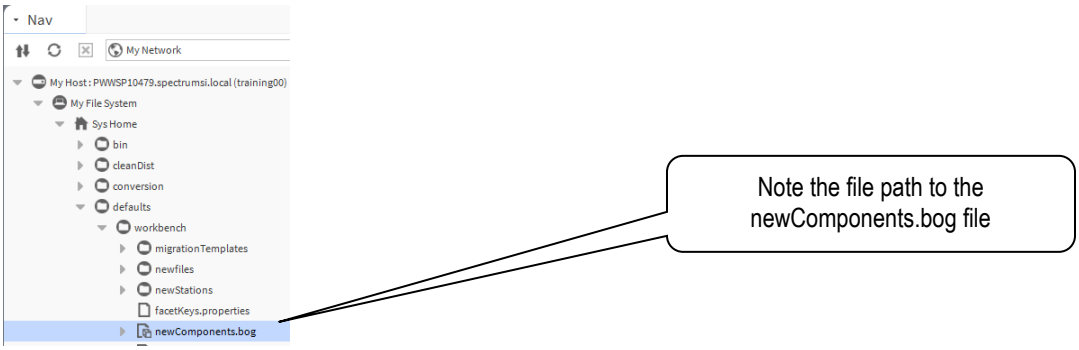
11. When completed, open the palette in the Palette SideBar by clicking the **Open Palette** button and clicking the **Browse** button. Double-click **User Home** to access the Niagara installation directory. Select your palette, click the **Open** button to open the palette.



Note: <u>You will have to browse to the location in which you created your new palette</u> (*it should be in the root of the SysHome folder*)

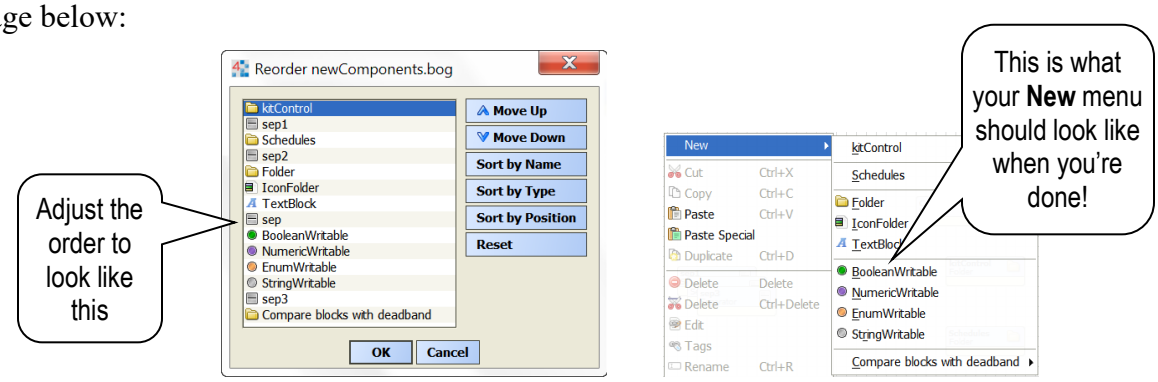
Part 2, Customizing the New Menu

1. In the Nav SideBar, expand: My Host-->My File System-->Sys Home-->defaults-->workbench. **newComponents.bog**.



2. Double click on **newComponents.bog**. The wire sheet for new components opens.
3. **From your personal palette** in the Palette SideBar, drag the **kitControl** palette onto the wire sheet.
4. Copy the **Separator** object that is already on the **newComponents** wire sheet. Using the **Paste Special** option, create 2 more separators.
5. From your **personal palette**, drag the entire **Schedules** folder onto the **newComponents.bog** wire sheet.
6. Reordering items in the **New Menu**
   a) Right click in the wire sheet of the **newComponents.bog** file.
   b) Select the **Reorder** option.
   c) Move the **kitControl** folder to the top of the list, with the first new separator below it.
   d) Arrange the list so the **Schedules** folder is right below this separator, with the second new Separator below the Schedules folder.
   e) Save your changes by clicking the **Save Bog** button on the button bar.

When completed, right click in the wire sheet, choose **New**. Your **New** menu should resemble the image below:

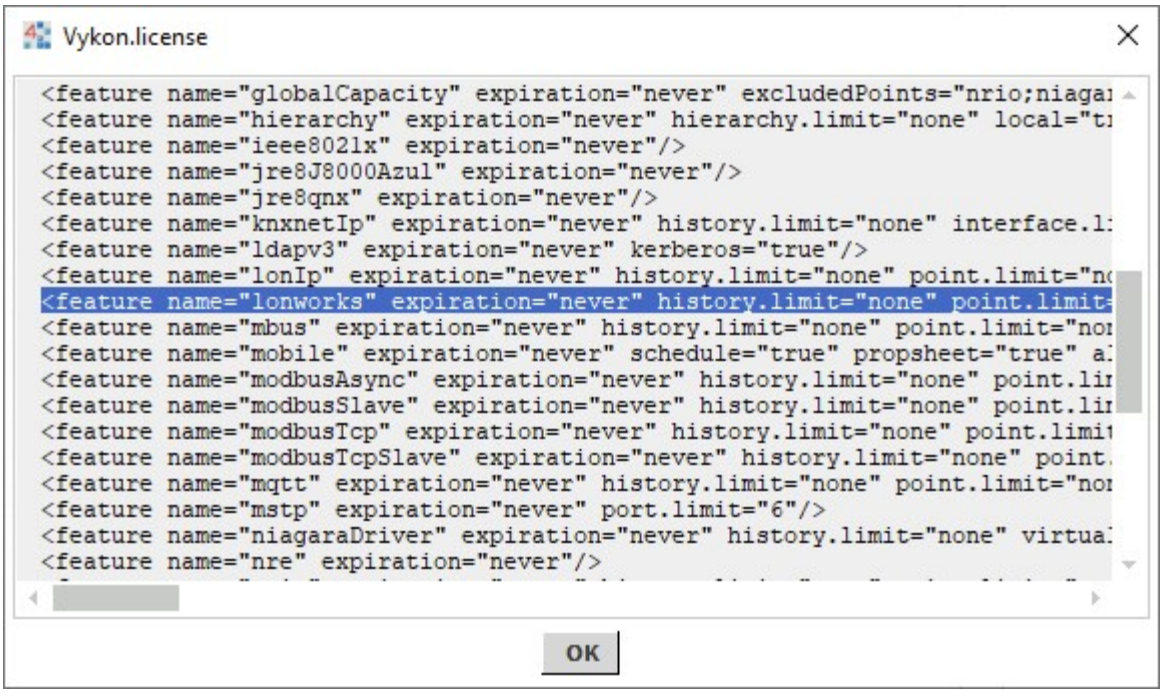


(*End of lab, **stop here**!*)

Lab #22: LON Integration, Part 1	10 min
Description:	In this lab you will Create and Configure a Lon Network.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Verify that a Niagara platform is Lon capable. 2. Create and configure a Lon Network. 3. Verify that the Lon Network is fully functional. **Note:** Verifying that the JACE is ready for Lon may have already been accomplished during the class.

Verify that the JACE is licensed for Lon.

1. Open a platform connection to the JACE. Open the License Manager. Select the item under Licenses, then click the View button and look for the ***feature name="lonworks"*** entry in the license. Driver features on the JACE themselves are not typically limited, however the feature globalCapacity does limit the total number of devices and points.
2. Make special note of any limits to the number of devices or points that are allowed by the license.

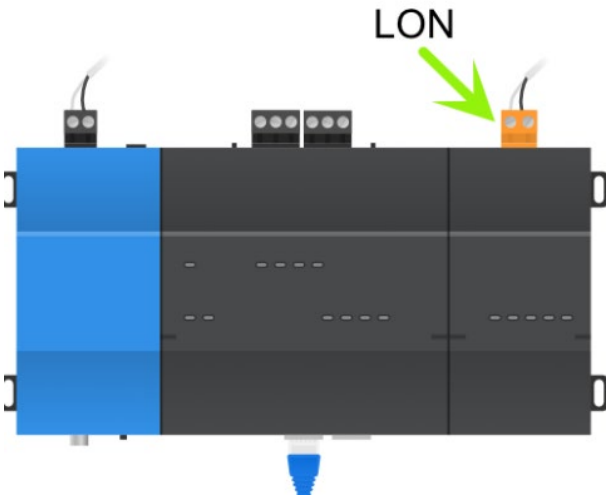


Verify the software modules required for a Lon integration.

3. Using the same platform connection to the JACE, open the Software Manager and verify that the Lonworks module is installed and up to date.

Verify that you have a physical connection to the Lon trunk.

4. See the image below. You can also use the classroom video feed to visually inspect the JACE board in the classroom studio. Open a web browser and go to https://192.168.86.51 and login using *username* ***guest*** *and password* ***guest1234***
    a) You are using a JACE-8000, the Lon Module will be connected to the JACE. The orange connector protrudes from the top of the Lon module on the right side of the JACE.
5. There is a two-wire, twisted pair attaching the Lon port to the Lon trunk. The Lon port is not polarity sensitive, so it is not possible to connect the wires backward**. See the image below**.



Create a LON Network

6. Expand the station **Config** container in the Nav SideBar and locate the **Drivers** container. Expand the view of the **Drivers** container; note the existing Niagara, and Bacnet Networks.
7. Open the **lonworks** palette and locate the **LonNetwork** object.
8. Drag a **LonNetwork** out of the lonworks palette and drop it onto the **Drivers** component. Leave the name as is. (*Note: The LonNetwork could also be added to your station by opening the Driver Manager window, then selecting the New button at the bottom of the window.*) Ensure that the Network displays a status of OK.
9. Expand the **LonNetwork** in the Nav Tree and verify that a **Local Lon Device** appears. This is the LonCard the station will use to connect to the LON network.

(*End of lab section, save your station and* ***stop here****!*)

Lab# 23: LON Integration, Part 2	25 min
Description:	In this lab you will add a Lon device template using the Device Manager view.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Locate a template in the User Home Templates folder. 2. Place the Lon Device Manager into Dev Template Mode. 3. Deploy the Template to add a Lon device. 4. Follow the Wizard-based process for deploying the template. 5. Add a haystack tag to the newly added controller. 6. Verify that the graphical view of the device is functional.

Adding a device from a *Template*

There are several ways to add a device to a station from a *template*. One way is to locate the template file in the Nav SideBar and drag and drop it into the station. Another is to add the device in the Device Manager view of the network driver.

1. Locate and expand the **Drivers** component in the JACE station, Locate the **LonNetwork**.
2. Double-click the **LonNetwork** to open the **Lon Device Manager**.
3. Click on the **Device Template** button on the Tool Bar.

*Note: In the Nav SideBar, expand **My Host**/ **My File System**/, then **User Home**. Expand the **Templates** folder to see the templates stored in your user directory (you can't access the templates created by other users unless they are moved to a common location on the computer).*

4. In the **Templates** pane that opens, note the template(s) that are available to use. Select the *AHU* template file, drag and drop it to the **LonNetwork**.
   a) On the Deploy dialog box click the OK button to accept the default name of AHU. (FYI a controller added from a templet will be disabled by default)
   b) On the templet dialog box, you could modify settings, however, leave all the settings at the default values and click the ok button.
   c) The device is Grayed out(status: disabled) in the Database list. The statuses will be updated in a future lab.
5. Apply the *haystack* "**ahu**" and the c:graphic tags to the AHU device you just deployed to the Lon network. (Remember to right click the device and edit tags.)
6. In the Nav SideBar, locate and expand the **AHU** controller and double-click the **Points** folder Note the Points are grayed out (status: disabled).
7. Use the View Selector to view the Wiresheet of the **Points** container. Note the out slot is grayed.
8. Double click the AHU in the Lon network to see the Px. points are gray (status: disabled).

(*End of lab section, save your station and **stop here**!*)

LAB: 24 LON Integration, Part 3	10 min
Description:	In this lab you will complete the process of enabling the controller that was added to the Lon Network with a template.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Discover the Lon Network to locate the actual controller that is on the network. 2. Match the Device Candidate (added from template) with the corresponding Lon controller. 3. Verify the correct Subnet / Node settings were implemented. 4. Commission the Lon Controller, as needed. 5. Bind Lon Links with the Lon Link Manager.

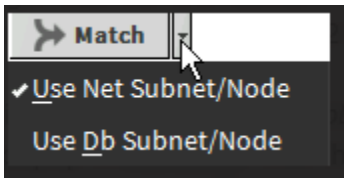
Matching the LON Controllers on the Network with the Database.
Commissioning, and Binding.

Enable the Lon Controller

1. Right-click the **AHU** controller in the Nav side bar, choose **Views** then **Ax Property Sheet**.
2. Set the **Enabled** setting to **True** (when added using a template, controllers are disabled by default)
3. Double click on the LonNetwork in the Nav Sidebar. Note that the device's status color is now orange. This is because the Neuron id is zero.

Match the LON Controllers on the Network with the Database

4. In the Lon Device Manager window, double click the AHU controller. Verify the controller is set to Channel 1,Subnet 1, Node Number 1 . Click the OK button to save any changes.
5. Click the **Discover** button to discover devices on the **LonNetwork**.
6. Review the LON device(s) that were discovered on the network.
7. Select the *honeywell* controller discovered on the network.
8. Select the AHU controller in the Database list.
9. Note the **subnet and node IDs** of the two devices.



Note*: In the lower right corner of the screen, drop the menu next to the **Match** button (look for the little black arrow). You can pre-set to use the Database or network subnet and node IDs.* (*Note: if the selection you want is already selected, click off the menu to close it! If you make either selection, the selection will change to the other selection!*)

10. Click the **Match** button, then the **OK** button.
11. Click the **Discover** button again. Note that the controller does not show up again in the **Discovered** window because it is now part of the database.

12. Right click the AHU device

    Commission the controller.

13. Select the **AHU** controller and the **Local LON Device** in the **Database** window. Click the *Commission* button. Note the status indicator at the top of the window.

Complete the Controller Bindings

14. Right-click the **LonNetwork** in the Nav SideBar. Choose Views, then **Lon Link Manager**.
15. Select all the items in the Network Variable Links list. Click the **Set Service Type** button and set the Service Type popup to **Standard** Click the **OK** button.
16. Unselect all list items. Click the Bind button at the bottom of the window to bind the connections of the selected points.

(*End of lab section, save your station and **stop here**!*)

Lab# 25: LON Integration Alarming	10 min
Description:	In this lab you will continue to develop your ability to extend the functionality of a Niagara object.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Add an Alarm Notification capability to the Lon AHU Controller. 2. Verify that the Alarm notifications are generated and appear on the Alarm Console.

Alarm Notification

1. Locate the **AHU** controller on the **LonNetwork**. Find the **SpaceTemp** point in the controller's Points folder. Open the Property sheet of this point.
2. From the **Extensions** folder in the Alarm palette, add an **OutOfRangeAlarmExt** to the Property sheet of the **SpaceTemp** points property sheet. Name the extension **SpaceTemp_Alarm.**
3. Expand the SpaceTemp_Alarm property sheet, then set the Source Name to **%parent.parent.parent.displayName%_%parent.displayName%**
4. Set To Normal Text to ***SpaceTemp is in the normal range.***
5. In Offnormal Algorithm of the alarm extension
   a) Set the High Limit to 68°F
   b) Set the Low Limit to 65°F
   c) Set the Deadband to 2
   d) Set High Limit Text to SpaceTemp > 68°F
   e) Set Low Limit Text to SpaceTemp < 65°F
   f) Check **Low Limit Enable** and **High Limit Enable** check boxes.
6. Set the Alarm Class to **High_Priority_Alarms**.
7. Click the **Save** button to commit changes.
8. Open the wire sheet of the AHU Points folder. Watch the SpaceTemp point until it goes into alarm.
9. Navigate back to the SpaceTemp-->SpaceTemp_Alarm extension and change the **OffNormal Algorithm High Limit** to **80°F** and the High Limit Text to SpaceTemp > 80°F (The SpaceTemp points alarm should clear)
10. Copy the SpaceTemp_Alarm extension to the SpaceTemp point in the ZoneTempControl folder. Change the extensions source name to **Zone_1_%parent.displayName%**
11. Open Config-->Services-->AlarmService->All_Alarms console recipient to verify the alarming activity.

(*End of lab, save your station and **stop here**!*)

LAB# 26: Emailing Alarms (OPTIONAL)	
Description:	In this lab you will explore the ability of your Niagara solution to send alarm notifications to external users via email. **Note**: The Email service will not be able to actually transport the email traffic when this lab is complete. There are additional steps the IT department are required to do to enable connectivity via an Email server that we cannot do in a classroom.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Add the Email Service to the Niagara station. 2. Configure the Email Service's incoming and outgoing accounts. 3. Reconfigure the Alarm Service as needed to forward the alarm notifications. 4. Implement an Email Acknowledger that can silence alarms upon any reply to the Alarm notification email message.

1. Open the **email** palette. From the palette, add an **EmailService** to the **Services** container in your station (drag the service from the palette and drop it onto the **Services** container in your station).

2. Add an **IncomingAccount** and an **OutgoingAccount** to the new email service (drag from the palette, drop on the **Email Service** in the station, you don't need to configure them in any way for this class).

3. Add an **EmailAlarmAcknowledger** to the **EmailService**.

4. From the **email** palette, add an **email recipient** to the **Alarm Service** wire sheet. Connect the output of the **High_Priority_Alarms** alarm class object to the new **email recipient**.

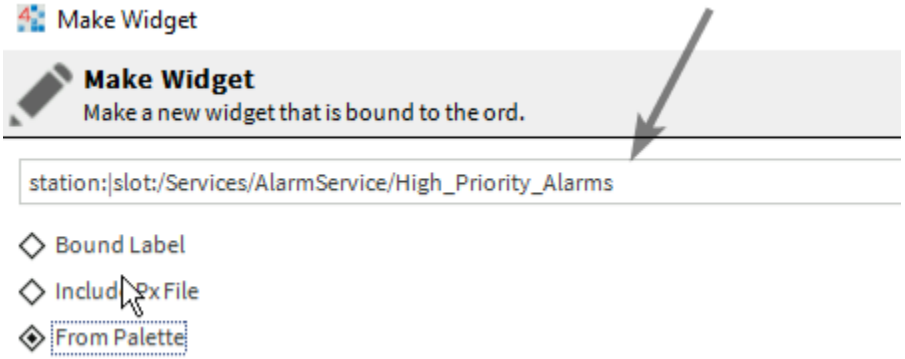
5. Open the **email recipient** to view the setup of the e-mail message (note the BFormat text in the e-mail). Add to the end of the text in the **Subject** line: **UUID:%uuid%**. It is appropriate to put a space in the line before this text. This text is required for the **EmailAlarmAcknowledger** to work properly.

(*End of lab,* ***Backup the station*** *and stop here!*)

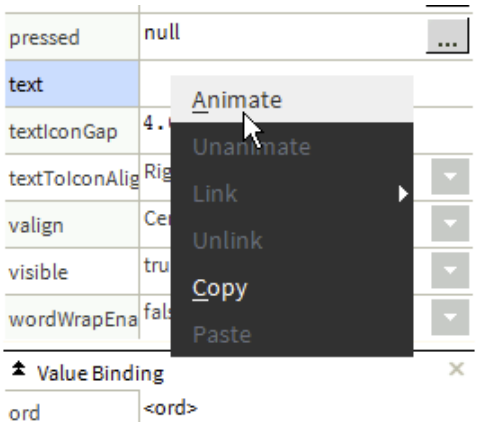
LAB# 27: Banner PX Modifications	
Description:	In this lab you will add additional functionality to the PX Include created earlier.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create a Hyperlink button that displays the Alarm Count and Hyperlinks to the Alarm Console. 2. Display Pump Runtime.

BannerPX procedures

1. Open the **Edit BannerPX** view of our Banner. (Found in **Config-->Home_Page-->Banner**)
2. Using the Nav Sidebar, expand **Config-->Services-->AlarmService**, then drag the **High_Priority_Alarms** onto the Edit BannerPX view.
3. On the Make Widget Wizard, select the From Palette radio button.
   a. Double click on the ord at the top of the widget.
   b. On the **Select Ord** dialog, choose **Unacked Alarm Count** under **High_Priority_Alarms**, then click the **OK** button.
   c. Using the **kitPx** palette, select the **HyperlinkButto**n and be sure to check the Hyperlink check box at the bottom of the wizard.



   d. On the right-hand side of the dialog, right click on the text slot and choose **Animate** from the popup menu.



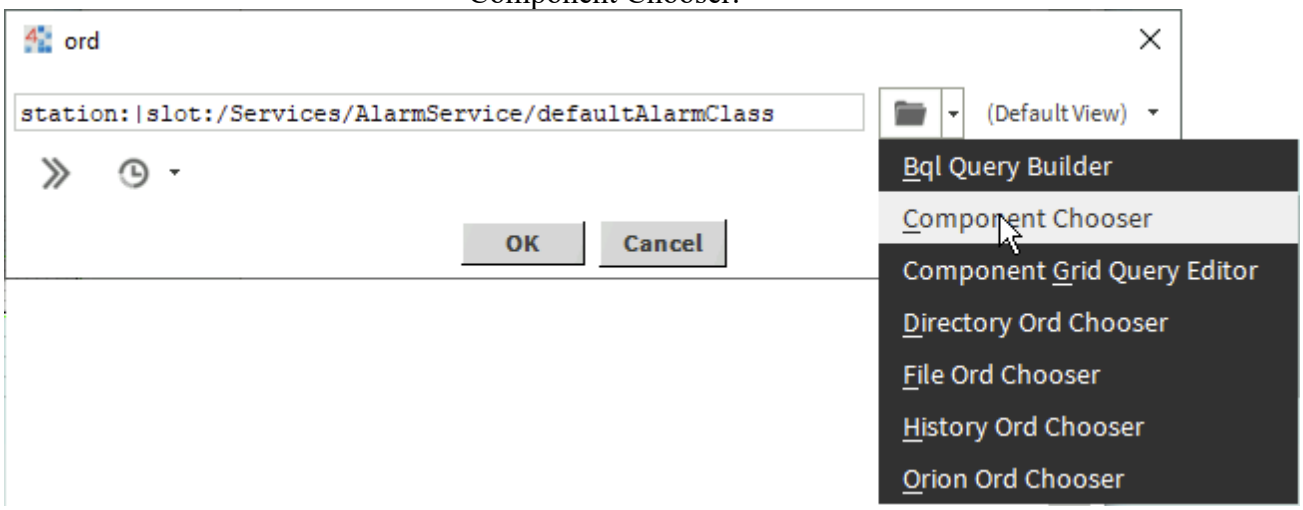
(Continued on the next page)

e. Modify the Object to String **Format** field to **Unacknowledged\nCritical Alarms:%.%** , then click the **OK** button.

f. Click the next **OK** button to close the wizard.

4. Adjust the button size, then double click it to open its **Properties**. Under the Value Binding section, click the **hyperlink** slot.

   a. In the hyperlink dialog box click the down arrow next to the folder icon and select Component Chooser.



   b. Collapse **the Default Alarm Class** node, then scroll and select **All_Alarms**. This will change the view to the **All_Alarms Console** when the users click the button.

   c. Click the **Ok** button on the Select Ord dialog, then **Ok** on the hyperlink dialog, and finally click the **Ok** button on the Properties dialog.

5. Save and Preview the Banner PX view. Make size and object placement changes as necessary.

6. Locate and open the PX view for **PumpControl**. Note the modifications.

7. Expand the PumpControl folder in the Nav, then locate and Expand **HotWaterPump_1**.

8. Toggle to edit mode by clicking the pencil icon in the toolbar, then drag the **DescreteTotalizerExt** onto the PX page.

9. Double click to change the ord at the top of the Make Widget dialog. Find and select **Elapsed Active Time** and click the **OK** button.

10. Choose the Bound Label radio button, put a check in Format Text and update the value to **Pump 1 Runtime:\n%.%.** Change any text and font values on the right as desired, then click the OK button to close the Make Wizard dialog.

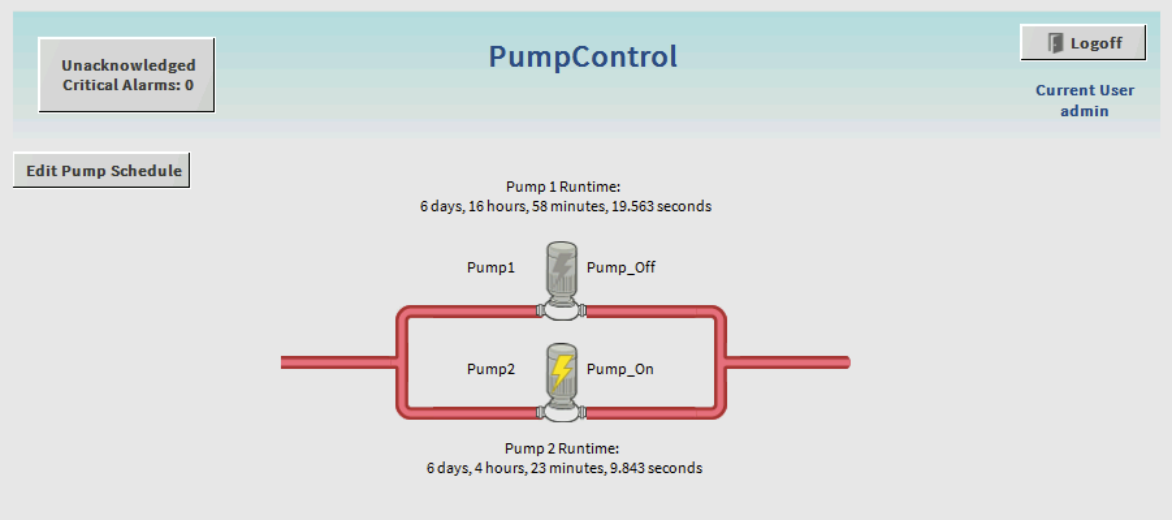
11. Resize the newly created label so all the text is visible and move it above **Pump 1**.

12. Repeat steps 7-10 for **HotWaterPump_2** making appropriate changes.

13. Resize the label and move it below **Pump 2**.

14. Save and preview the PX page.

15. When you are done it should look similar to the screenshot below.



*(End of lab, **save station** and stop here!)*

LAB #28: Navigational Hierarchies	15 min
Description:	In this lab you will practice the basic techniques for building a navigation mechanism based upon a Niagara4 Hierarchy.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Configure the Hierarchy Service 2. Create and test a simple Hierarchy Tree 3. Configure a couple basic Hierarchy Definitions a. QueryLevel b. RelationLevel

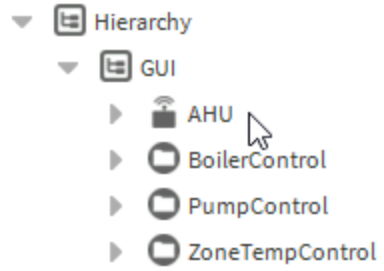
Things you need to know:

- Hierarchies use queries based on tagging and relationships to search for data to be displayed.
- A hierarchy requires at least one query definition.
- Hierarchies can have group level or list level filters that allow matched data from queries to be shown in branches of the hierarchy structure.
- The order of definitions is important.

Create a Hierarchy using a simple query.

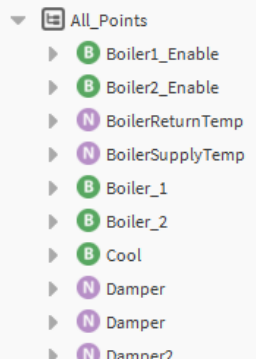
Hierarchies are managed under the **HierarchyService** beneath the **Services** container.

1. Using the Nav Sidebar, expand **Config-->Services--> double click the Hierarchy Service** this will open the open the AX Property sheet view of the **Hierarchy Service** .
2. In the Palette SideBar open the **Hierarchy** palette.
3. Drag a **Hierarchy** out of the hierarchy palette and drop onto the **Hierarchy Service** AX Property sheet. Name this new hierarchy **GUI**. This hierarchy will be used for easy access to the PX views.
4. Drag a **QueryLevelDef** of the **hierarchy palette** and drop onto the **GUI** Hierarchy (No need to rename the QueryLevelDef).
   a) Expand **GUI-->QueryLevelDef** then set the query slot to **c:graphic**
   b) Check that the **IncludeGroupQueries** is set to **true**
   c) Click the Save button.
   d) Right-click on the **Hierarchy** folder in the station and select **RefreshTreeNode**.

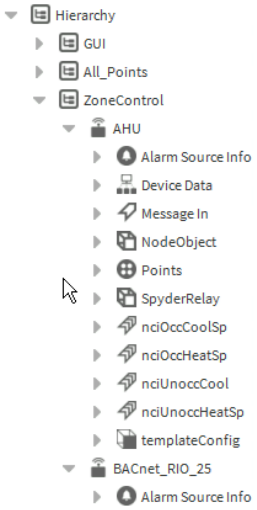


   e) Observe the result. You should see a list of graphical views that were created earlier, but only those views that were tagged with the **c:graphic** tag from the Custom dictionary. The Banner graphic is embedded into the other graphical views.

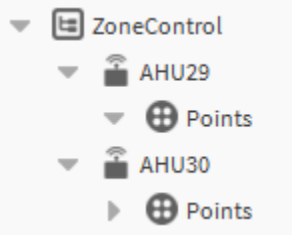
5. Drag a **Hierarchy** from the palette and drop onto the **Hierarchy Service** property sheet. Name the new hierarchy **All_Points**.
6. Drag a **QueryLevelDef** from the palette and drop onto the **All_Points** Hierarchy (No need to rename the **QueryLevelDef**).
    f) Set the query to **n:point** (this is an implied tag)
    g) Set **IncludeGroupQueries** to **true**
    h) Click the Save button.
    i) Right-click on the **Hierarchy** folder in the station and select **RefreshTreeNode**.
    j) Observe the results, you should see a list of all the components in the station that have the implied **n:point** tag from the Niagara dictionary.



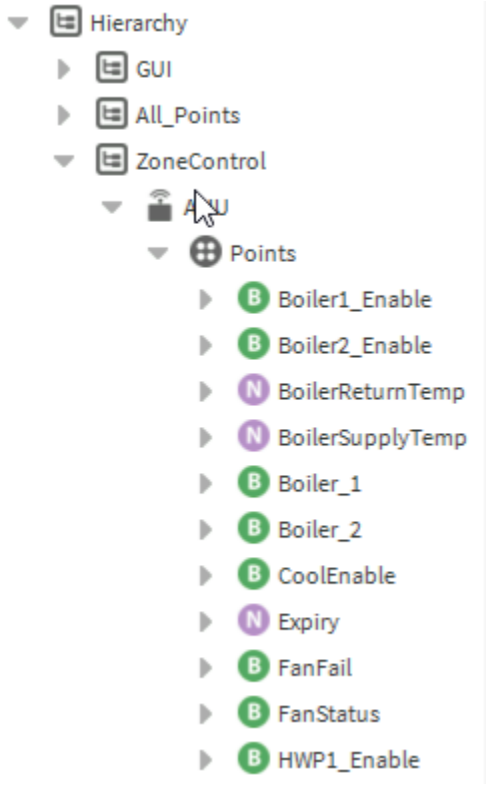
7. Drag a Hierarchy from the palette and drop onto the **Hierarchy Service** property sheet. Name this new hierarchy **ZoneControl**. **RefreshTreeNode** and observe the changes.
8. Drag a **QueryLevelDef** from the palette onto the **ZoneControl** Hierarchy. Set the Query to **hs:vav or hs:ahu.**
9. Drag out a **RelationLevelDef** from the palette onto the **ZoneControl** Hierarchy.
    a) In the **Outbound Relation Ids** field, set the RelationID to **n:child**
    b) Click the Save button.
    c) **RefreshTreeNode** and note that the RelationLevelDef produced a change to the hierarchy. However, every possible property is now listed under each device!



10. Edit the **Filter Expression** of the RelationLevelDef to restrict the hierarchy so that only the Points container will appear on the Hierarchy tree for each device. Define the Filter Expression by typing in the following string: **n:type="bacnet:BacnetPointDeviceExt"**
11. **RefreshTreeNode** and observe the change. Notice that the Points container still appears; however, none of the proxy points are listed in the hierarchy.



12. Drag another **QueryLevelDef** from the palette onto the **ZoneControl** Hierarchy. Set the Query to **n:point. RefreshTreeNode** and observe the changes.



(*End of lab, save station and stop here!*)

OPTIONAL LAB: Additional Hierarchy Exercises	
Description:	In this lab you will practice the process of creating Hierarchies.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Heating / Boiler Control Hierarchy 2. Geographical Hierarchy

Part 1, Heating / Boiler Control Hierarchy

1. Drag a new **Hierarchy** from the palette and drop onto the **Hierarchy Service** property sheet. Name this new hierarchy **HeatingSystem**.
2. To the new **HeatingSystem** hierarchy, add a new **QueryLevelDef** called **BuildingHeatSystem**. Set the query to: **hs:heat or hs:heating or hs:boiler**
3. **RefreshTreeNode** and observe the changes.
4. Drag another new **Hierarchy** from the palette and drop onto the **Hierarchy Service** property sheet. Name this new hierarchy **BoilerControl**.
5. To the new **BoilerControl** hierarchy, add a new **QueryLevelDef** called **BoilerPoints**. Set the query to: **hs:boiler or hs:steamPlant**
6. **RefreshTreeNode** and observe the changes.

Part 2, Geographical Hierarchy

1. Use the BQL Query Builder (refer to Lab #18) to add the following tags to all BACnet proxy points in a batch fashion:
    a. hs:geoState and set the Default value to **VA**
    b. hs:geoCity and set the Default value to **Richmond**
2. Using the BQL Query Builder once again, add the following tags to all LON proxy points in a batch fashion:
    a. hs:geoState and set the Default value to **MN**
    b. hs:geoCity and set the Default value to **Minneapolis**
3. Drag a new **Hierarchy** from the palette and drop onto the **Hierarchy Service** property sheet. Name this new hierarchy **GEO**.
4. To the new **GEO** hierarchy, add a **GroupLevelDef** called **State**. Set the query to: **hs:geoState**
5. To the same hierarchy, add a **GroupLevelDef** called **City**. Set the query to: **hs:geoCity**
6. Add a **QueryLevelDef** called **Variables**. Set the query to: **n:point**
7. **RefreshTreeNode** and observe the changes.

Part 3, Add a Hierarchy to a Palette

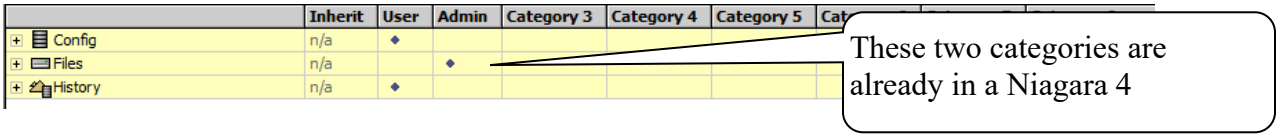
1. Open your personal palette's wire sheet. Create a new folder in the palette called **Hierarchies**. Open the wire sheet of this new palette.
2. Drag the entire **HierarchyService** from the station on your JACE drop it onto the wire sheet of the new Hierarchies wire sheet. Save the changes to your personal palette.
3. Review your personal palette in the Palette SideBar.

LAB #29: Security and User Administration, Part 1	15 min
Description:	In this lab you will define a system of Categories that are useful for granting or denying access to specific systems, subsystems, equipment, or components.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create a system of Security Categories. 2. Assign Categories to various levels of the Station's Architecture. 3. Configure Passwords in the Authentication Service

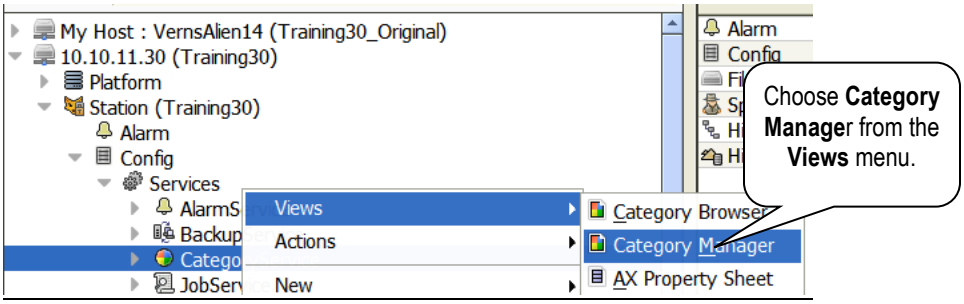
Security Categories

Do this lab in the station running on the JACE.

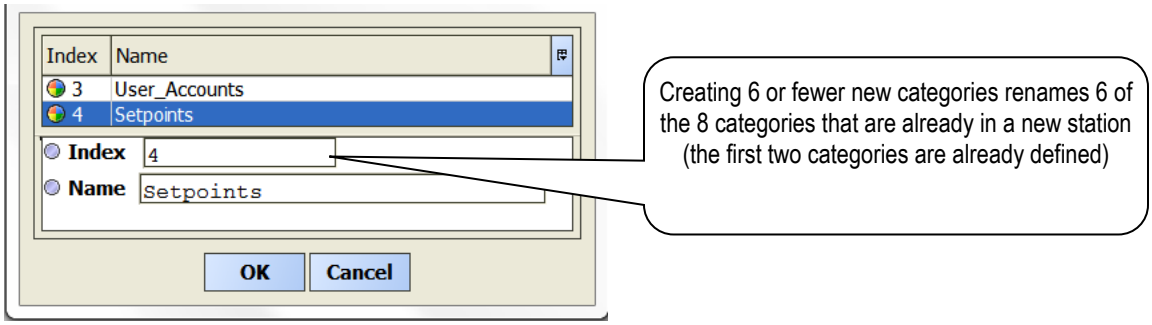
1. In the Nav SideBar, open the **Services** container. Right-click the **CategoryService** object. Choose **Views**, then **Category Browser**. There are already two categories defined in a station. Parts of the station are already assigned to these categories.



2. Right-click the **CategoryService** object and choose **Views**, then **Category Manager**.

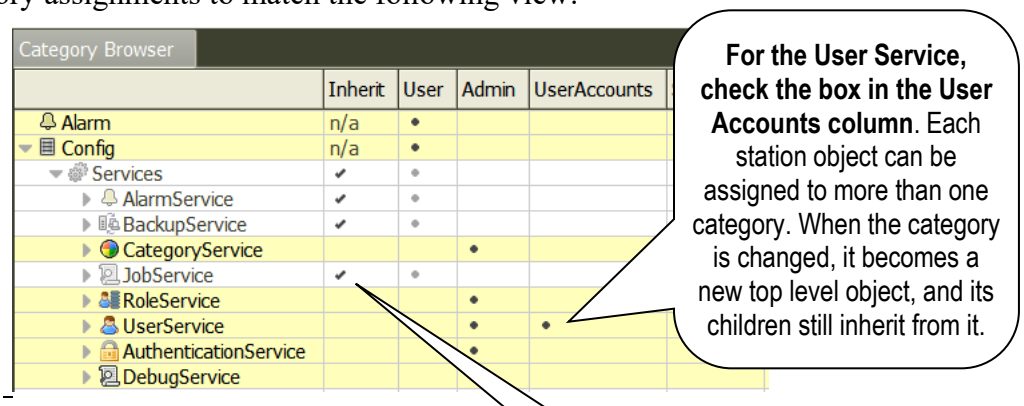


3. In the **Category Manager**, click the **New** button on the bottom of the window. Create **2 new categories**. Call the categories **Users_Accounts** and **Setpoints**.

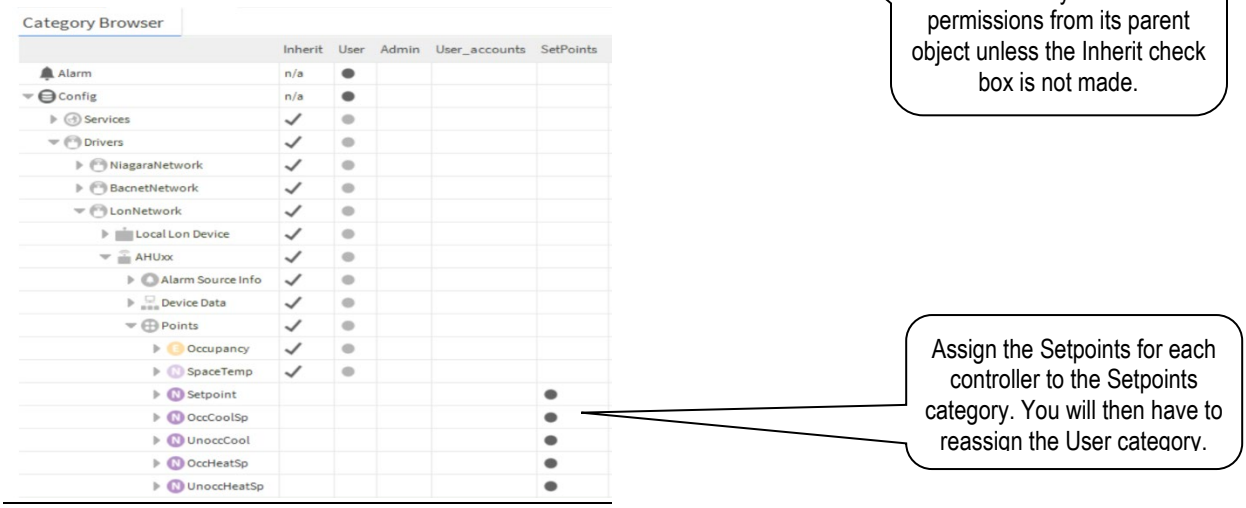


4. Change your view to the **Category Browser** (right-click **CategoryService** in the Nav SideBar, choose **Views**, then **CategoryBrowser**).

5. Adjust the category assignments to match the following view:



6. Make these **additional changes** to the category assignments:



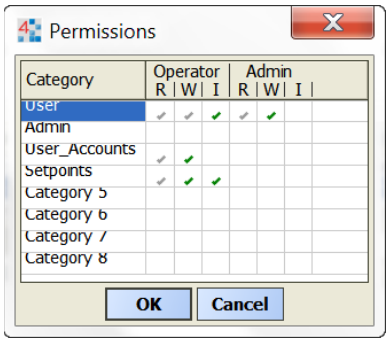
7. Browse into the folders in **Control_Logic** folders. Assign all setpoint objects that you find to the **Setpoints** category.
8. **Save** the changes made to the category file (click the **Save** button on button bar).
9. In the **Nav SideBar**, locate the **AuthenticationService** in the **Services** directory in your station.
10. Double-click the **AuthenticationService** to open its Property Sheet view.
    a) Unfold the **Authentication Schemes** folder, then the **Digest Scheme** folder.
    b) Unfold the **Global Password Configuration** slot container. Then **Password Strength**.
    c) Change the **Minimum Length** setting to **8** and save your changes
    d) Repeat these setting for the **AXDigestScheme**. Save your changes.
11. For both schemes, set the **Password Expiration Interval** to 90 days and the **Warning Period** to 14 days.

(*End of lab, save station and stop here!*)

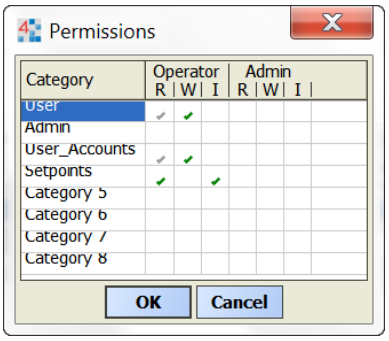
LAB #30: Security & User Administration, Part 2	10 min
Description:	In this lab you will learn how to create User Roles that define the permissions and viewable hierarchies assigned to individual users.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create new User Roles. 2. Assign Permissions to those Roles. 3. Declare which Hierarchies are viewable to each Role.

User Roles

1. In the Nav SideBar, locate the **RoleService** in the Services container in your station. Double-click the Role Service to open the AX Role Manager.
2. Click the **New** button at the bottom of the view. Enter 3 as the number of new Roles to create. Edit the new Roles this way
   a) Name the first new Role **AdminLevel**.
      i. Set **Permissions** at **SuperUser**
      ii. Viewable **Hierarchies**, select all in the list
   b) Name the second Role **Engineering**.
      i. Set permissions to match this view:

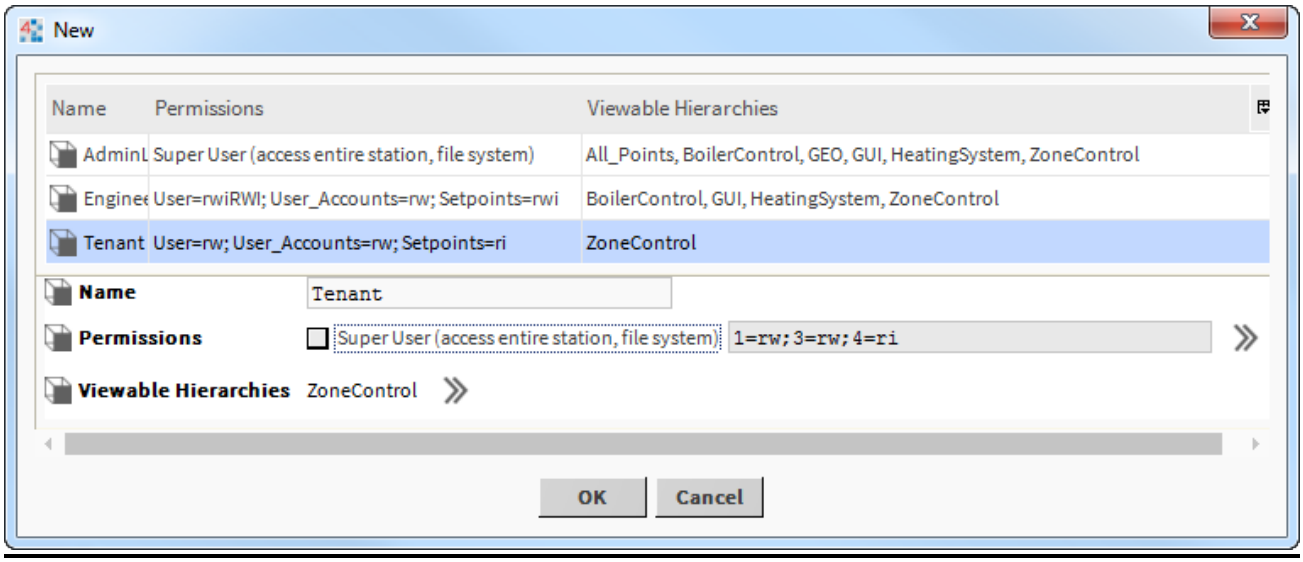


      ii. Set **Viewable Hierarchies** to **GUI, HeatingSystem**, **BoilerControl** and **ZoneControl**.
   c) Name the third Role **Tenant**.
      i. Set permissions to match this view:



      ii. Set **Viewable Hierarchies** to **ZoneControl**.

3. Your New menu should look like this when you're ready to create the new roles:



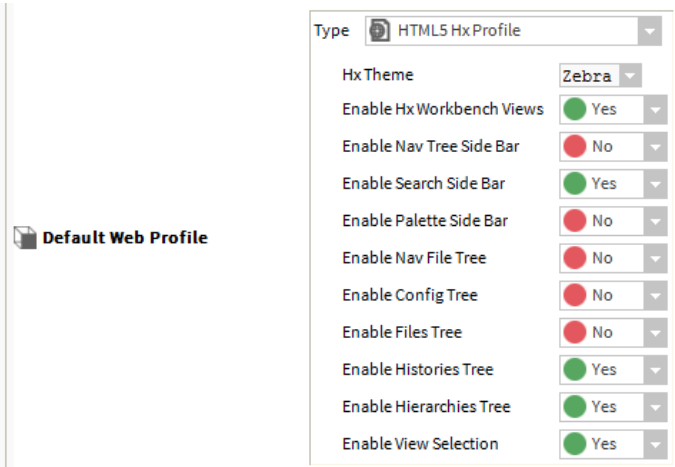
4. Click **OK** to create the new Roles.

(*End of lab, save station and stop here!*)

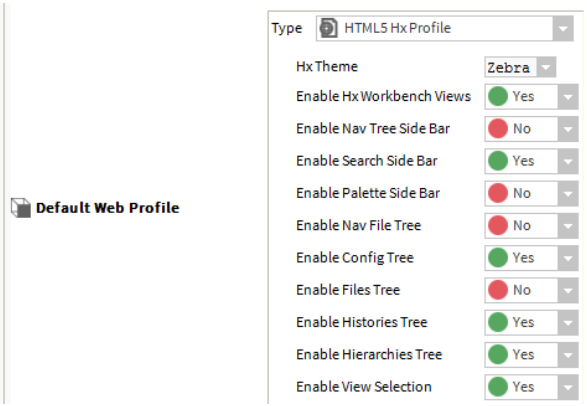
LAB #31: Security and User Administration, Part 3	20 min
Description:	In this lab you will create User accounts, and define the parameters that govern the User's access to the Niagara station:
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Configure Password Lockout 2. Assign User Role 3. Establish Initial Password 4. Configure Default Web Profile 5. Test / Verify the User account from the Workbench

Creating User Accounts

1. Right-click the **UserService**. Choose **Views**, then **AX Property Sheet**.
   a) Change the **Lock Out Period** setting to 1 minute
   b) Change the **Lock Out Window** setting to 1 minute and 30 seconds
2. Open the **Ax User Manager** view of the **User Service**.
3. Click the **New** button at the bottom of the view to create a single new user account. Make these settings for the new user:
   a) Name: **JohnD** (*this is the login username.* ***It is*** *case sensitive*)
   b) Full name: **John Doe**
   c) Set the account to expire in 30 days from now.
   d) Assign the **Tenant** role.
   e) Set the Authenticator **Password** and **Confirm** to: JohnD1234
   f) Set Authenticator **Force Reset at Next** Login to **True**
   g) Select the Authenticator **Expires On** radio button.
   h) Set **Default Web Profile** settings as shown in the below.
   i) Click the **OK** button to create the user.



4. Create a second user, user information:
    a) Name: **SueS**
    b) Full name: **Sue Smith**
    c) Set the account to **Never expire**.
    d) Assign the **Engineering** role.
    e) Set the Authenticator **Password** and **Confirm** to: **SUES1234.**
    f) **Set Authenticator Force Reset Next Login to True** Click **OK** to create the user.
    g) Select the Authenticator **Expires On** radio button.
    h) Set the facets **unit conversion** to **English.**
    i) Set **Default Web Profile** settings as shown below.
    j) Click the **Ok** button to create the user.



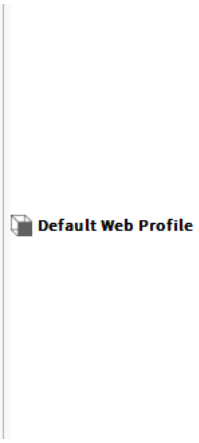
5. Create a third user, user information:
    a) Name: **BillJ**
    b) Full Name: **Bill Jones**
    c) Set the account to **Never Expire**
    d) Assign the **AdminLevel** role.
    e) Set Authenticator **Password** and Confirm to: **BILLJ1234**.

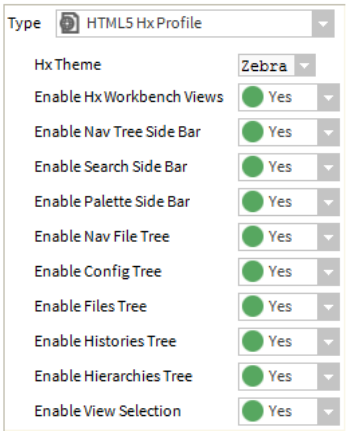
f) Set Authenticator **Force Reset At Next** Login to **False**.

g) Select the Authenticator **Expires On** radio button.

h) Set Facets **Unit Conversion** to **Metric**.

i) Set Default Web Profile settings as shown in the image right.





j) Click **OK** to create the user

6. Create a fourth user, user information:

a) Name: **JACE** (Remember! Case sensitive!)

b) No full name.

c) Set the account to **Never Expire.**

d) Assign the **AdminLevel** role.

e) Set Authenticator **Password** and **Confirm** to: **Jace1234**

f) Set Authenticator **Force Reset At Next** Login to **False.**

g) Set Facets **Unit Conversion** to **None**.

h) Leave **Default Web Profile** settings at the default.

i) Click the **OK** button to create the user.

7. After completing the creation of the users, **Disconnect** from the station (**right-click** the station in the Nav SideBar and choose **Disconnect**).

8. Sign on to the station as **JohnD**. When prompted to change your password, change it to **JohnD2345**. Review the following parts of the station to see how the view and options have changed:

a) If you can see the **User Service**, can you modify John's **Default Workbench Profile** settings? _____

b) Can you see the **Category Service** in the Services directory in the station? _____

c) Double-click the **BoilerControl** folder, check the View menu, can you see the Wire Sheet? _____ (you need Admin read/write permissions to see a wire sheet) Open the **AXPropertySheet** of the folder.

d) Expand the view of **Boiler_1**. Can you see the **Fallback** slot? _____

e) In the **PumpControl** folder, can you perform any Actions on the HotWaterPump_1? _____

f) In the folder, can you modify the **BooleanSchedule**? _____

g) Can you see the **Files** directory in the station? _____

h) Expand the **Alarm Service** component in the Nav SideBar. Double-click the **All_Alarms** 

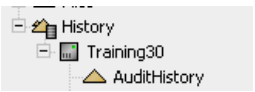
console recipient. Can you acknowledge an Alarm? _____ Can you Force Clear an Alarm? _____

i) Locate the **Setpoints** for the **AHU** controller in your station. Can you change the setpoint using the Action Menu?_____

9. Sign off from the station, then sign on to the station as **SueS**. Review the following parts of the station to see how the view and options have changed:
    a) Can you modify Sue's **Default Workbench Profile** settings? _____
    b) Double-click the **BoilerControl** folder, check the **View** menu, can you see the **Wire Sheet**? _____ (you need Admin read/write permissions to see a wire sheet) Open the **AXPropertySheet** of the folder.
    c) Expand the view of **Boiler_1**. Can you see the **Fallback** slot? _____
    d) In the **PumpControl** folder, can you put the **HotWaterPump_1** point in manual mode (using the Actions menu) _____
    e) Can you modify the **BooleanSchedule**? _____
    f) Can you see the Files directory in the station? _____
    g) Open the **All_Alarms** 

console recipient. Can you acknowledge an Alarm? _____ Can you Force Clear an Alarm? _____
    h) Locate the **Setpoints** for the **RTU** controller in your station. Can you change the setpoint (use the Actions / Set command if you can)? _____
10. Sign off as **Sue** and sign on as **Bill**.
11. Review the following parts of the station to see how the view and options have changed:
    a) In **User Service**, could you modify Bill's **Unit Conversion** setting? _____
    b) Can you see the **Category Service** in the Services directory in the station? __
    c) Can you open the **Files** directory? _____
    d) Double-click the **PumpControl** folder, can you see the **Wire Sheet**? _____ (you need Admin read/write permissions to see a wire sheet)
    e) Expand the view of **HotWaterPump_1**. Can you see the **Fallback** slot? _____
    f) In the **PumpControl** folder, can you put the **HotWaterPump_1** point in manual mode (using the Actions menu) _____
        i. If you do see the **Actions** menu, can you put the point in **Emergency Override**? _____
    g) In the **PumpControl** folder, can you modify the **PumpSchedule**? _____
    h) Expand the **Alarm Service** component in the Nav SideBar. Double-click the **All_Alarms** 

console recipient. Can you see any alarms? _____ If so, can you acknowledge them? _____
    i) Right-click any alarm. Do you see the **Force Clear** option? _____
    j) Can you successfully save the station? _____
12. Sign off as **Bill** and sign back on as **admin (**user name = **admin,** password = **Admin12345)**.
13. Open the **Audit History** file and review the history for the users signing in and out of the station.



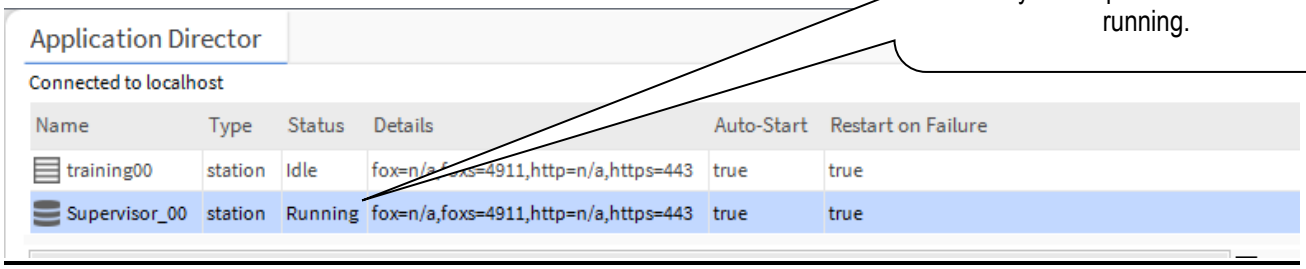
(*End of lab,* ***stop here***)

LAB #32: Niagara Network Integration	25 min
Description:	In this lab you will establish station-to-station communication between the station running on your JACE, and a new Supervisor station running on your PC.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create a new station based on a New Supervisor template. 2. Place the Station Manager into Learn Mode and initiate a Discovery process. 3. Establish and Configure the Primary connection. 4. Edit and Enable the Reciprocal connection. 5. Verify two-way connectivity between the stations.

Station-to-Station Communication

1. **Create a new station** on your remote PC. Call the new station **Supervisor_XX**, where the XX represents your workstation number [*click the **Tools** menu, select **New Station***].
   a) Use the **NewSupervisorStationWindows.ntpl** template when creating the station.
   b) Assign a password of **Admin12345** to the **admin** user account.
   c) Select the radio button selection for **Copy it to secure platform for "local host" with the station copier**.
2. When you click the **Finish** button to create the station, you will be prompted to authenticate to the platform. After connecting to the platform, the Station Transfer Wizard dialog will open. Deselect **the AUTO START** checkbox, then click the **Next** then the **Finish** buttons.
3. The station will transfer from **User Home** to the **Platform Home**. When finished click the **Close Window** button.
4. On the **Open** the **Application Director** dialog click the **Yes** button. The Application Director will open, and the station will start.
5. Verify that the station is running.

Note the **Status** of the stations on your PC. If any are running, except the Supervisor, stop them! Only the Supervisor station should be running.



6. Connect to the new Supervisor_XX station as **admin**.
7. In the new **Supervisor_XX** station, expand the **Services** container.

   Double-click the **UserService** in **Supervisor Station** to open the **User Manager** view.

   a) **Create a new user**:
      i. The new user should be called **M2M**.

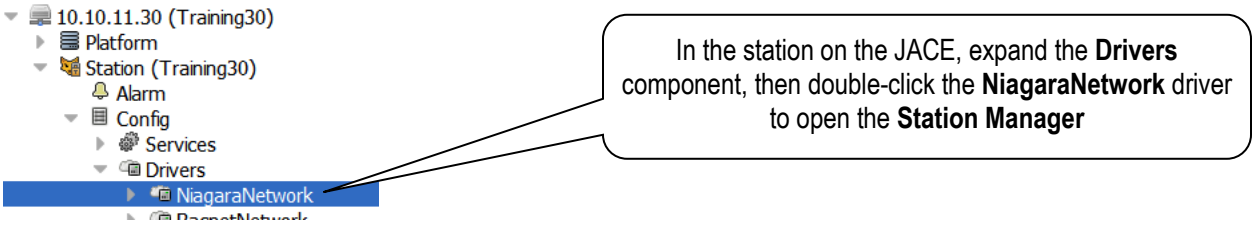
ii. Password = **M2MAdmin12**

iii. **Role:** check the **admin** role selection

iv. Leave all the other settings at the defaults

v. Click **OK** to create the new user

8. In the **station running on your JACE**, expand the view of the **Drivers** component, then **double-click** the **NiagaraNetwork** to open the **Station Manager** view.



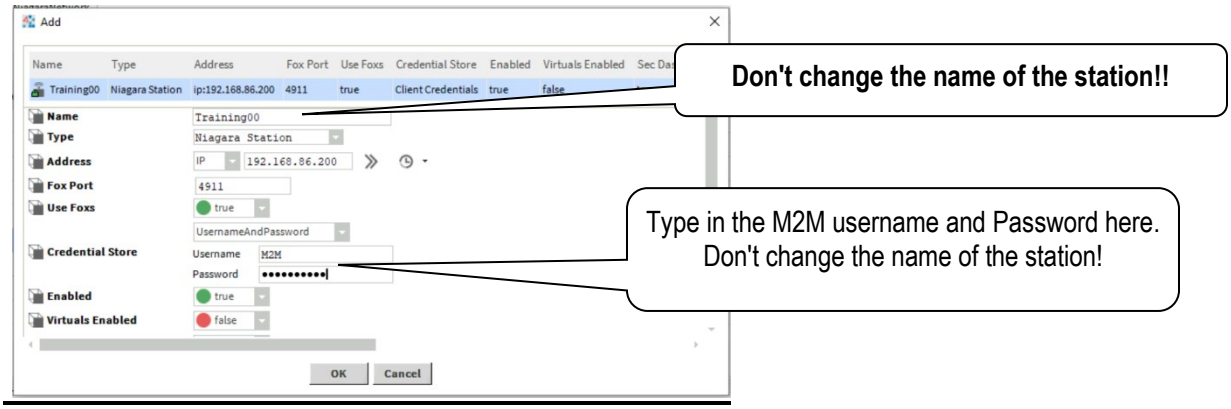
9. Click the **Discover** button to find other stations available on the network.

10. In the **Discovered** window, locate **your Supervisor_XX** station.

a) Drag and drop your station onto the **Database** window.

b) Note: **Use Foxs** is set to true and the **Foxs Port** setting to **4911**

c) In the **Add** window, locate the **Username** field (DON'T change the **Name** field!). Type in the user name **M2M**, set the Password field to **M2MAdmin12.** The **TrainingXX** station on the JACE will use this user account to connect to the **Supervisor** station. Click the **OK** button to add the Supervisor to the JACE's Niagara Network.



d) Right-click the station in the **Database** view, choose **Actions**, then **Ping** to force a connection. Note, the connection only lasts a moment.

e) The connection will go into a **Fault** condition because the TLS security certificate provided by the remote station has not been approved. Here's how we fix that:

i. In the Station from which the Niagara Network connection originated (In this case the station running on your Remote PC) open the CertManagerService located in PlatformServices.

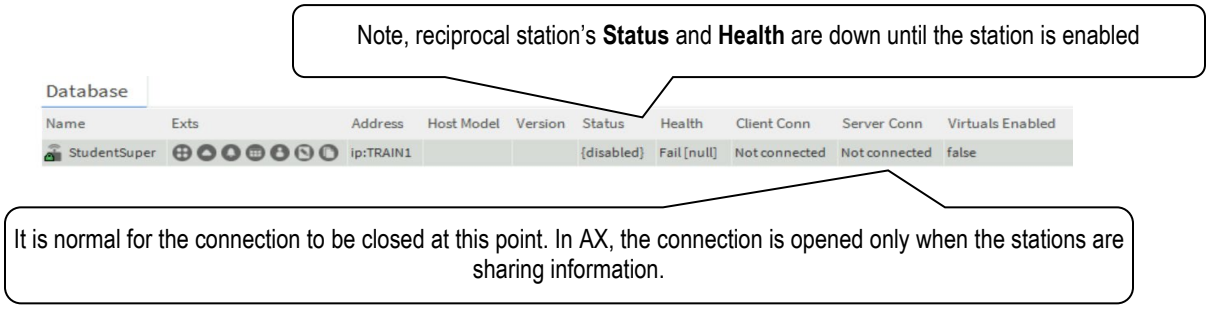
ii. Click the **Allowed Hosts** tab. Select the certificate you see there, then click the **Approve** button. Click **OK** in the confirm popup window.

iii. Return to the **Station Manager** view of the station on the Supervisor. Ping the JACE again. This time a connection should be made. If not, it's probably an authentication issue.

f) Verify the **TrainingXX** station connected to the **Supervisor_XX** station. If the **Supervisor_XX** station shows with a yellow highlight, try pinging the it (right-click the Station, choose **Actions**, then **Ping**). If the connection still doesn't work, verify the user name in the station's user manager matches the user name you assigned to the Niagara Network connection. If it still doesn't work, consult the instructor.

11. In the **TrainingXX** station, expand the view of the **Drivers** component. **Double-click** the **NiagaraNetwork** to open the **Station Manager** view.

12. Locate your **Supervisor_XX** station (this is the Reciprocal Station) in the **Database** window. Note the **Status** and **Health** of the **Supervisor_XX** station.



13. Select (click on) the **Supervisor_XX** station, then click the **Edit** button.

a) Enter the IP Address of the Remote PC. (Reference you desktop) *192.168.86.1xx*

b) In the **Username** field, type in **M2M. Set the password to M2MAdmin12** (this is the user account we created earlier in the Remote PC and the JACE). Enable the connection.

c) Click **OK** to save the changes.

d) Click **OK**. Note the connection health and status. **Ping** the station to initiate a handshake. You will need to repeat the approval of the security certificate like you did in step 10-e.

e) **Ping** to verify the connection works.

f) Repeat b) through e) until you have it working!

(*End of lab, **stop here***)

LAB #33: Configuring a Master Schedule	10 min
Description:	In this lab you will explore the Niagara Network functions that facilitate the sharing of a Schedule object with remote stations.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create and configure a Master Schedule. 2. Import a Master Schedule into destination stations. 3. Link the Subordinate Schedule into local control logic.

Part 1, Creating the Schedule that will be Shared

1. In the **Supervisor Station**, under the **Config** node, create a new folder called **MasterSchedules**.
2. Add a **Boolean schedule** called **MasterBooleanSchedule** to the wire sheet of the new **Master Schedules** folder.
3. Configure facets on On/Off and set the schedule to On at 8:00AM and Off at 5:00PM Mon-Fri.

Part 2, Sending the Schedule to the JACE

1. In the TrainingXX station, expand the Niagara Network on the Nav Tree.
2. Under the **Supervisor_XX** station, double-click the **Schedules** node to open its Import Manager. Click the **Discover** button to see which schedules in the Supervisor can be imported.
3. Drag the Boolean Schedule into the **Database** window. Note the **Last Success** display is null.
4. Right-click the Boolean Schedule in the Database window. In the menu, click **Import**. Note the **Last Success** display note has a time and date stamp.
5. In the Nav SideBar, expand the view of the station in your JACE. Expand the view of the Niagara Network. Expand the view of the **Schedules** node of the Supervisor_XX station. You should see the Boolean Schedule that was just imported.
6. Double-click the Boolean Schedule to see the scheduled events. Note that the events can't be modified here (special events can be set up though!).
7. In the Niagara Network of the Training Station on the JACE, right-click the **MasterBooleanSchedule** node. Choose **Views**, then **AX Property Sheet**. Expand the **ext** object, then expand the **Execution Time** folder. Note the **Trigger Mode** and **Interval** settings.

Part 3, Using the Master Schedule in the JACE

1. Right-click the imported schedule in the JACE's Niagara Network and choose **Link Mark**.
2. Create a new folder called **MasterScheduleLab** under the Config node in the station running on your JACE. Open the wire sheet of this folder.
3. Add a **BooleanWriteable** point to the wire sheet. Assign facets of Occupied / UnOccupied.
4. Right-click the new BooleanWritable point. Choose **Link From "MasterBooleanSchedule"** and link to **in16**. Note the updating of the points value when the link is complete.
5. Verify functionality by modifying the Master Schedule in the Supervisor and verifying that the Subordinate updates. You will need to manually Import again.

(*End of lab, **stop here**!*)

LAB #34: Routing Alarms to the N4 Supervisor	5 min
Description:	In this lab you will be able to configure the Niagara Network to route alarm notifications from one station to another station.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Configure the Alarm Service to route alarms to the Supervisor Station. 2. Verify that the Alarms are received in the Alarm Console of the Supervisor Station. 3. Verify that the Alarms can be acknowledged in the Supervisor, and that the acknowledgement is sent to the point of origin (alarm console in the JACE.)

This lab will be done using the station running on your JACE.

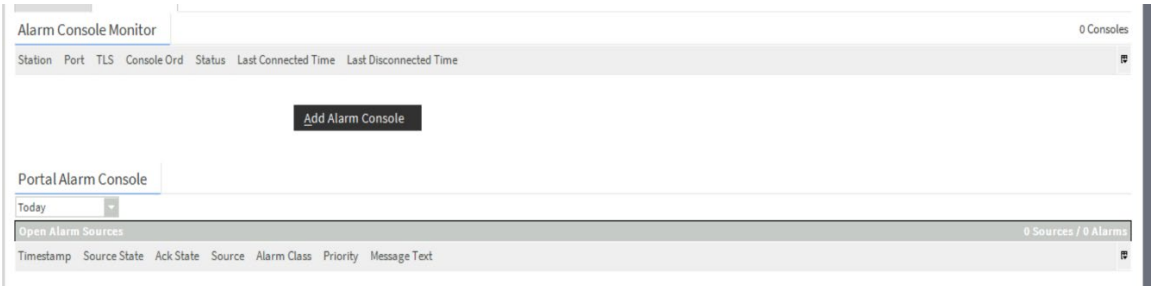
1. Open the wire sheet of the **AlarmService** in the station running on the JACE.
2. From the **Alarm** palette's **Recipients** folder, add a **StationRecipient** to the wire sheet.
3. Connect the output of **all Alarm Class objects** to the input of the new **StationRecipient**.
4. Open the **Property Sheet** of the **Station Recipient**. Configure the **Remote Station** setting to send the alarms to your Supervisor Station (use the drop-down menu).
5. Open the wire sheet of the **AlarmService** in the **Supervisor_XX** station. Add a **ConsoleRecipient** to the wire sheet. Connect the output of the **Default Alarm Class** to the input of the new **Console Recipient**.
6. Open the alarm console that you just added to the **Supervisor_XX** station to view alarms from the supervisor station and from the training station on the JACE.
7. Acknowledge the alarms from the Alarm Console in the Supervisor.
8. Verify that the alarm is also acknowledged in the Alarm Console in the JACE.

Note: Two-way communication is required for alarm routing and alarm acknowledgment. If the **Ack State** column in the Supervisor's Alarm Console indicates "Ack Pending" but the never counts down to Zero, then you have a Niagara Network communication issue that needs to be resolved.

(*End of lab, **save station** and **stop here**!*)

LAB #35: Configuring the Alarm Portal	5 min
Description:	In this lab you will learn how to configure the Alarm Portal to simultaneously monitor multiple alarm consoles.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Open the Alarm Portal. 2. Configure the Alarm Console Monitor 3. Configure Alarm Portal Options

1. In the Workbench under the **Tools** menu, select **Alarm Portal**.
2. In the white space in the **Alarm Console Monitor** window, right-click in the Alarm Console Monitor area and click **Add Alarm Console**.



3. Type in the IP address of your lab PC: 192.168.86.1xx (your PC is running your Supervisor Station). Accept the certificate. Add a username and password as needed (use **M2M** as the username, password = **M2MAdmin12**). Click **Next**.
4. Select the console you wish to monitor and click the **Finish** button.
5. Review and acknowledge the alarms that are displayed.
6. Remove the **Alarm Console** from the **Monitor** window (right-click on the console, choose Remove Alarm Console).
7. Close the Alarm Portal by opening a different tab window in the workbench and then closing the alarm portal tab.

(*End of lab, **stop here**!*)

This page is intentionally left blank!

LAB #36: Archiving Histories to the Supervisor	10 min
Description:	In this lab you will practice the importing and exporting of historical trend data from the JACE to the Supervisor.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Import or Export Historical Trend data from one station to another station. 2. Schedule and configure the Import / Export. a. Capacity restrictions b. Full Policy c. On Demand Polling 3. Manually initiate the Importing or Exporting of data.

***Note**: Histories can be exported from a JACE to a Supervisor, or the histories can be imported to the Supervisor from the JACE. History importing to a Supervisor station is more configurable and therefore more desirable in most applications.*

Part1, Exporting Histories from the JACE to the Supervisor

1. In the **TrainingXX** station running on your JACE, expand the view of the **Niagara Network**.
2. Expand the view of the **Supervisor_XX** station in the JACE's **Niagara Network**.
3. Locate and double-Leftclick the **Histories** node.
4. Note the name of this view. Change the view to the **Niagara History Export Manager**.
5. Click the **Discover** button.
6. Expand the view of the **Trainingxx** history container in the **History ID** window.
7. Select, drag and drop the Boilers_1 and Boiler_2 histories onto the **Database** window.
8. Select both the histories displayed in the **Add** window. Change the **Export Time** to **8:00 am**, randomized by 20 minutes. Click **OK** to add the histories to the database.
9. Select all the histories in the **database (**the lower window). Click the **Archive** button in the bottom right corner of the window and observe the progress of the export.
10. In the **Supervisor_XX** station, expand the **History** container directly under the Station node.
11. Expand the **TrainingXX** folder. Note the histories in the folder. Double-click one of the history files to view the data.

Part 2, Importing Histories to the Supervisor from the JACE

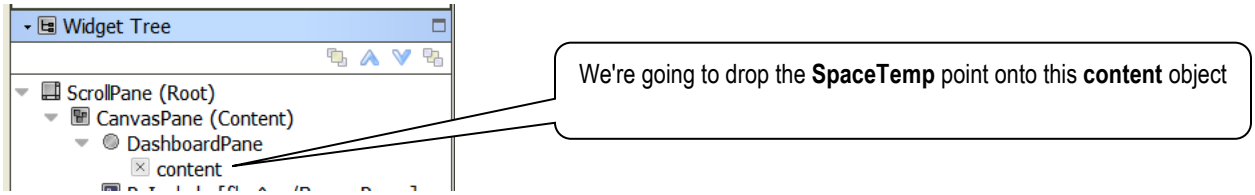
12. Open the TrainingXX Station in the Niagara Network of your Supervisor Station.
13. Double-click the **Histories** object in the TrainingXX station. Note the view is the Niagara **History Import Manager**.
14. Click the **Discover** button.
15. Expand the **TrainingXX** history folder in the **Discovered** window. Drag the **Audit** and **Log** histories and all the remaining system histories into the **Database** window. Don't add Boiler_2 and 2 we already did!

16. Set all the histories to import at 6:00 am, randomized by 30 minutes.
    a) Set the **Capacity** for all histories to a record count of **5,000** records.
    b) Force an archive of these histories.

(*End of lab, **Backup the station on the JACE**, and **stop here**!*)

LAB #37: Dashboard Views	10 min
Description:	In this lab you will practice the creation and custom configuration of User Dashboards.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Verify that the Dashboards module is installed. 2. Add the Dashboard Service to the Services container. 3. Add a Dashboard Pane to a PX View. 4. Embed a webChart widget into a User Dashboard. 5. Modify a Dashboard for the current session only. 6. Permanently modify the Dashboard for a specific User.

1. Perform this work in the station that is running on your JACE.
2. Open the Dashboards palette, and drag the Dashboard Service into the Services container.
3. In the Config container, create a New Folder and name it Dashboards.
4. Right-click the new Dashboards folder. From the **Views** menu, select **New View**.
5. Name the new PX view **DashboardPx,** and leave everything else set to the defaults.
6. From the **Custom** tag dictionary, add the **graphic** tag to this object.
7. Set the view size to 1100 x 900. Set a reasonable background color. Add the **BannerPx** file to this view.
8. From the palette, drag a **Dashboard Pane** on to the edit view of the **DashboardPx** view.
9. Adjust the size of the dashboard pane appropriately for the view.
10. In the right-hand SideBar called **Widget Tree**, expand the **Scroll Pane**, then the **Canvas Pane**, then the **Dashboard Pane**. Note the **content** object inside the dashboard pane.



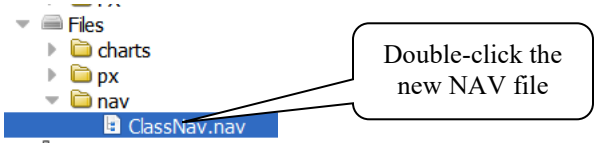
11. Unfold the **ZoneTempControl** folder in the Nav SideBar.
12. Drag the **SpaceTemp** point from the folder in the SideBar and drop it onto the **content** object in the **dashboard pane** in the **Widget Tree** SideBar.
    a) In the Make Widget Wizard, choose **From Palette**.
    b) Select the **WebChart** palette
    c) Select the **Chart** option in the palette.
    d) Click **OK** to add the point value to the dashboard.
13. Run the view to see the chart that is built.
14. While in the **View** mode, drag from the Nav SideBar the **VavDamper** point. Drop it onto the chart.
    a) Note the additional data that displays
    b) Click the **Save** button to save the change made to the dashboard
    c) Drag any other numeric data point onto the chart. **Save**, and note the change in the chart.

15. Open the **Edit** view of the **DashboardPx** file.
16. Drag the **ZoneControl_PID** onto the view. In the Make Widget Wizard, choose **From Palette**.
17. Pick the **webChart** palette, choose the **Circular Gauge** as the widget.
18. Resize the gauge to fit the view.
19. Add the Vav Damper point to the view, using the same **Circular Gauge** option.
20. Save the view and check out your dashboard.

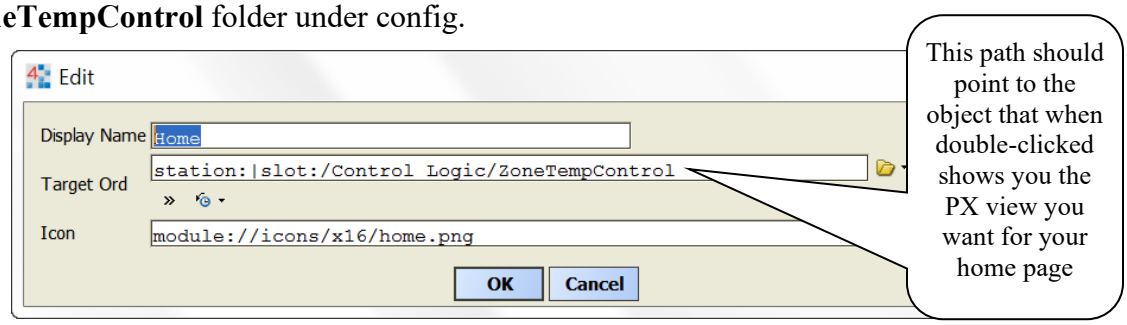
(*End of lab, stop here!*)

LAB #38: NAV Files	15 min
Description:	In this lab you will design a Navigation scheme and embed it's functionality into the Graphical User Interface.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Create a Navigation File. 2. Specify a Home Page. 3. Assign the NAV file to a User account. 4. Test the Navigation features in the GUI.

1. Create a new folder called **nav** under the **Files** folder in the station running on your JACE.
2. Right-click on the new **nav** folder and choose **New**, then **NavFile.nav**. Call the new file **ClassNav.nav**. Open the new Nav file for editing by double-clicking it.



3. **To set the home page in a Nav file:** Double-click on the row labeled "**Home**" in the edit window. In the **Target ORD** window, browse to the **ZoneTempControl** folder, (DON'T select the actual PX view in the file container or the slot that links to the PX view). Select (click-on) the **ZoneTempControl** folder under config.



4. Click the **Show Components** button. Drag the objects that display in the **Source Objects** window into the **Result Tree** window, except for the **Banner** folder (these are objects / folders that have a PX view linked to them.
5. Right-click one of the ORDs added to the **Result Tree** view and choose **Edit**. Change the **Display Name** field to a more user friendly name if you think it appropriate.
    a) Save the Nav file by clicking the **Save** button on the Tool button bar
6. Open the **AX User Manager** view (right-click user service).
7. Double-click the **admin** user account to open the **Edit** view of the account.
8. Assign the new Nav file to the **admin** user. Click **OK** to save the change.

(*Continued on next page*)

9. Open a web browser on your PC. Access the station through the browser (put your JACE's IP address in the Address field as follows: https://192.168.86.2xx, click Go).
    a) Click the Advanced button on the warning page that pops up, and then click Continue to 192.168.86.2xx
    b) Sign in with the admin account to view the station.
    c) You may be presented with a security certificate concern. You will be informed that the security certificate is not trusted and you will be asked to accept a security exception for this site. You must do so to access the server.

(*End of lab, stop here!*)

LAB #39: Copying Stations	5 min
Description:	In this lab you will create a backup of the platform (including the station), and provide a copy to the Instructor. This is a critical final step that you will also perform when you have completed your Certification Test.
Learning Objectives: Upon completion of this lab exercise you will be able to:	1. Backup the JACE. 2. Copy the Training Station from the JACE to the PC. 3. Copy the station from your PC to the Instructor's PC. 4. Clean the JACE.

Part 1, Copying the Station from the JACE to the PC

1. Right-click your station and choose **Backup Station** (click SAVE to complete).
2. Open a **Platform** connection to the JACE. Double-click the **Station Copier** tool.
3. Locate the **TrainingXX** station you've been working on all week and select that station (right side pane in the station copier view).
4. Click the active **Copy** button (the one pointing at the column titled **Stations on This computer**) to begin the copy process.
    a. Rename the station to TrainingXX_EOC (end of class)
5. When the copy process has completed, verify your station shows up in the list on your PC (left side pane of the station copier), then close the copy window.

(*End of lab, stop here*)