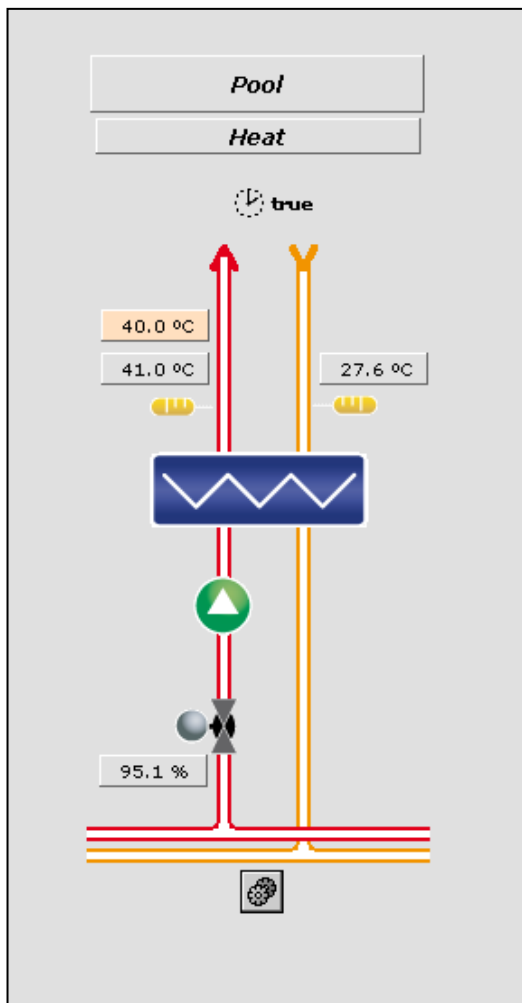


EN-30: Automating the creation of custom applications

- Vykon HVAC is designed to automate the process of application generation. However if an application is required which is not in the Vykon HVAC library then it needs to be created as a custom application.
- Vykon HVAC contains a range of automation facilities that the engineer can use to create custom applications, which can be made to have the same look-and-feel as library generated applications. This engineering note shows how to create the example of a heat delivery circuit supplying controlled hot water to a swimming pool heat exchanger and it illustrates the use of some of the custom application automation facilities contained within Vykon HVAC.
- First of all, let's see what we are aiming to create. Here is the overall performance specification of the pool heat delivery circuit:
 - Circuit enabled by a time schedule
 - A two port supply water control valve is modulated by a PID loop using the measured supply water temperature and a calculated supply temperature setpoint
 - The calculated supply temperature is determined by a PID loop using the measured return water temperature and a desired pool temperature setpoint
 - An on/off circulation pump in the supply is enabled when the valve is in a controlled mode ($\neq 0\%$)
 - Provide an adjustable 'heat demand' signal to the heat plant, enabled whenever the pump is on
 - All PID settings and desired pool temperature setpoint to be available for the appropriate user
 - A graphic which illustrates the current state of the circuit and all its control elements
- The heat circuit graphic and its settings tabs should look like this:



Desired value | **Valve control**

Valve

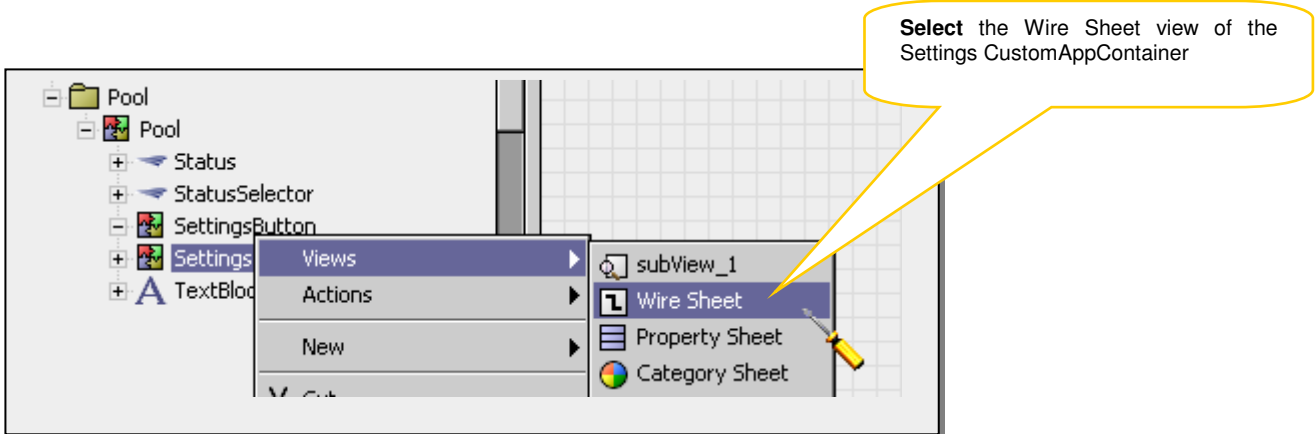
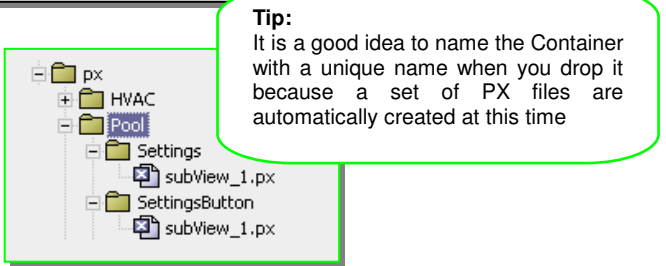
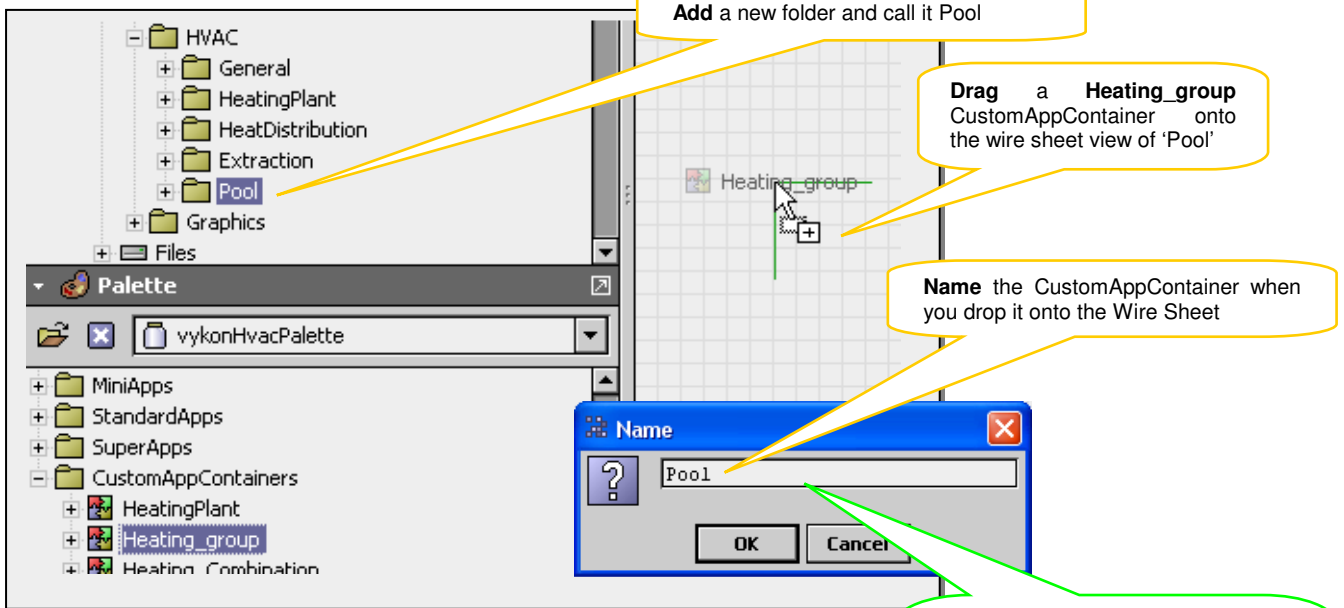
Desired supply temperature	40.0 °C
Actual supply temperature	37.9 °C
Dead band	0.0 °C
Offset	0.0 °C
Gain [Kr]	4.0
Integral time [Ti]	01:00 mm:ss
Derivative time [Td]	00:00 mm:ss
Minimum Valve position	0.0 %
Maximum Valve position	100.0 %
Valve position	10.6 %

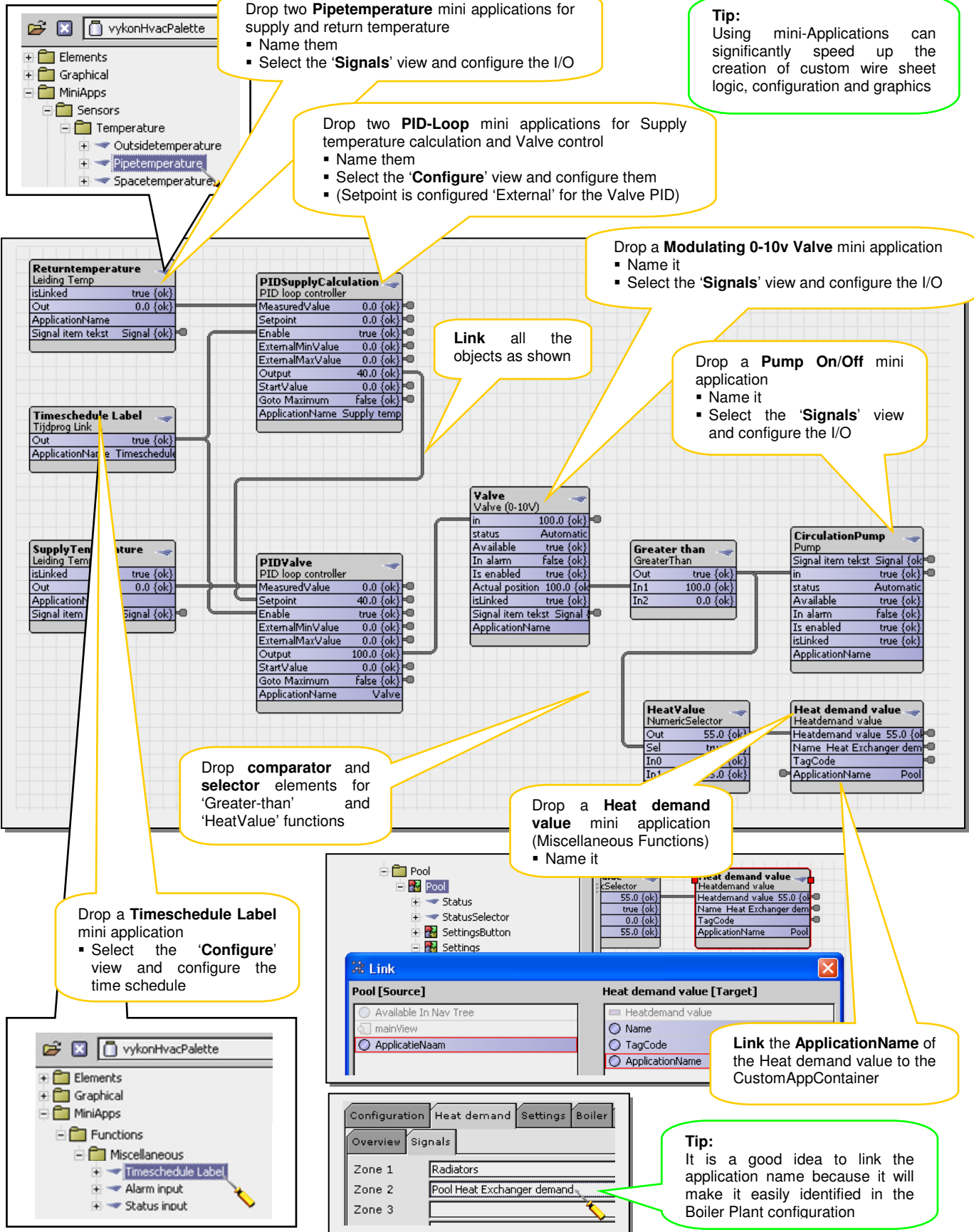
Desired value | **Valve control**

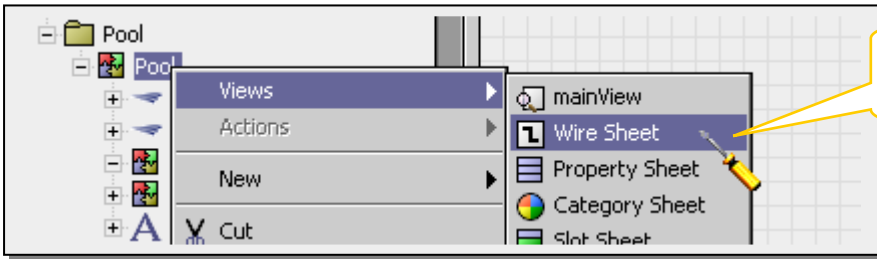
Supply temperature

Desired return temperature	28.0 °C
Actual return temperature	0.0 °C
Dead band	0.0 °C
Offset	0.0 °C
Gain [Kr]	4.0
Integral time [Ti]	01:00 mm:ss
Derivative time [Td]	00:00 mm:ss
Minimum Desired supply temperature	24.0 °C
Maximum Desired supply temperature	40.0 °C
Desired supply temperature	40.0 °C
Heat demand value	55.0 °C

- Let's now create our custom application

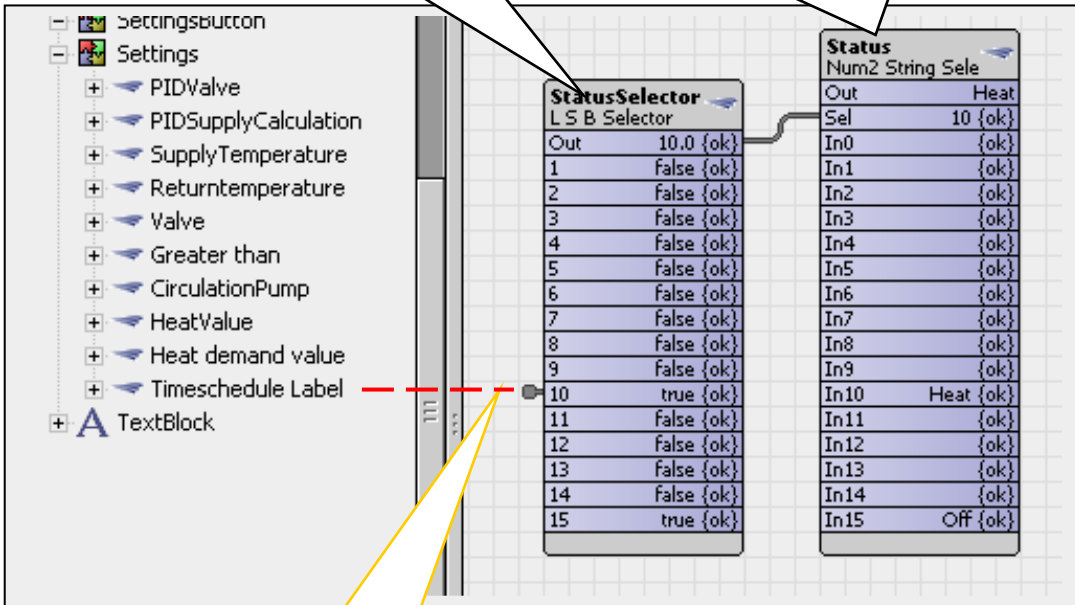
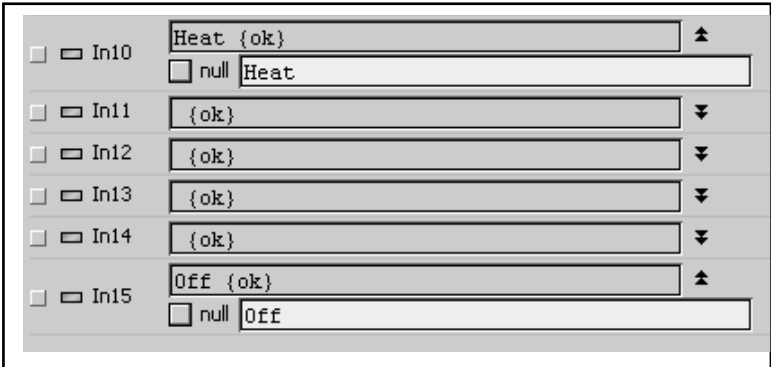
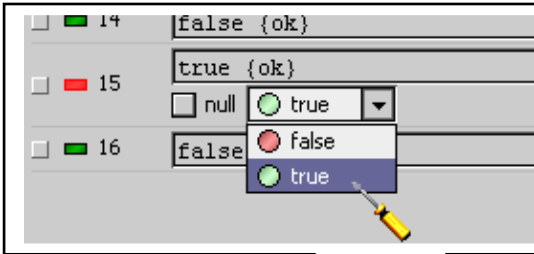






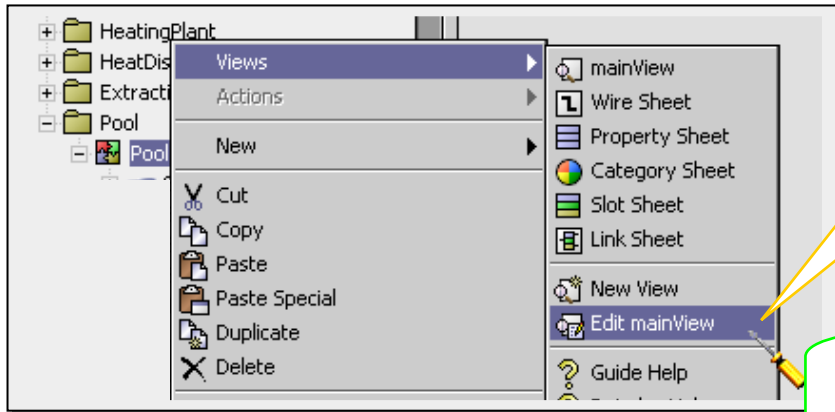
Select the Wire Sheet view of the Pool CustomAppContainer

Configure the property views accordingly, of the two selector objects, you will find prepared on the wire sheet view



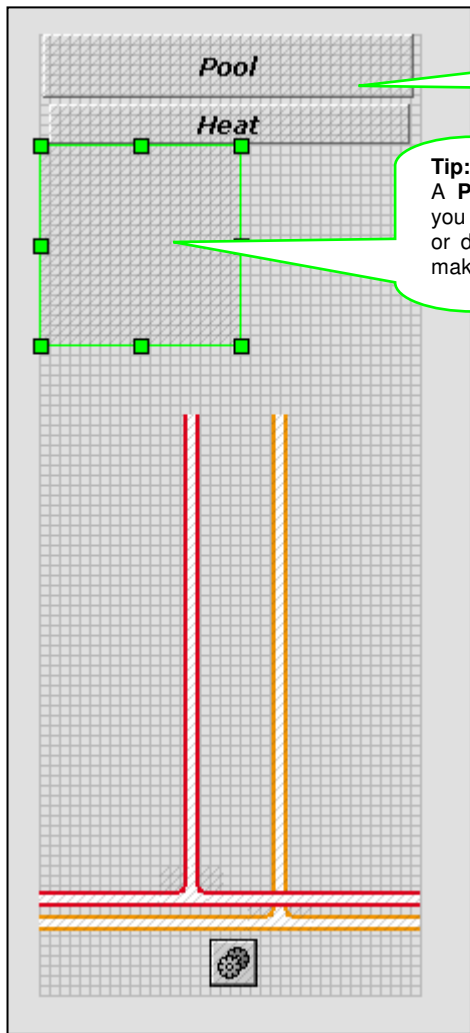
Link the Timeschedule Label 'Out' to In10 of the selector object

Tip:
By using these preloaded selectors on the Wire Sheet view of the CustomAppContainer you can easily define what the current status of the custom application is and display it on the mainView graphic



Select the Edit mainView of the Pool CustomAppContainer

Tip:
The mainView has already been populated with some basic widgets to speed up the graphic creation process



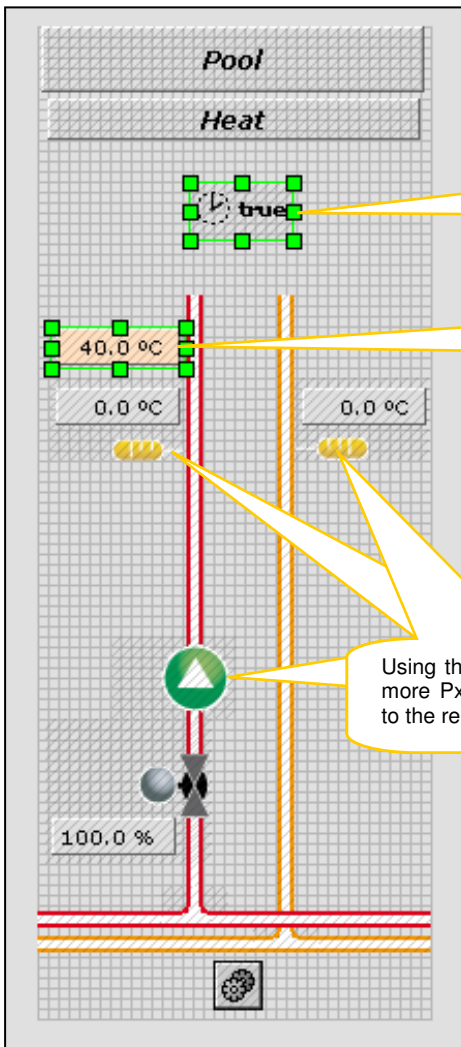
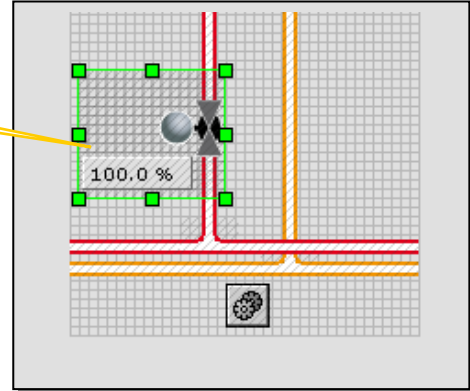
Tip:
The Header and Status widgets are already bound and operational

Tip:
A PxPane widget is already loaded for you to use and then you can use the copy or duplicate function in the Px Editor to make it easier to generate more of them

Select the PXPane Widget that is provided then bind the PX-view binding to TwoPortMotorLeft view of the Valve

Using all your artistic skills carefully **position** the valve on the pipe-work

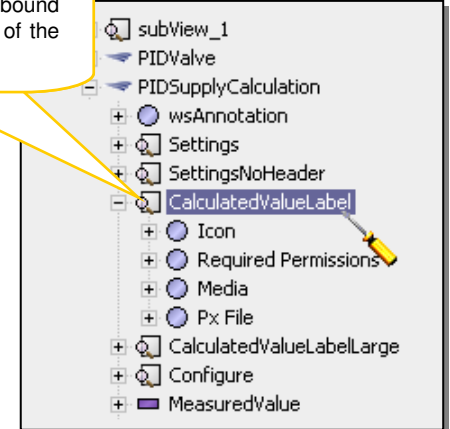
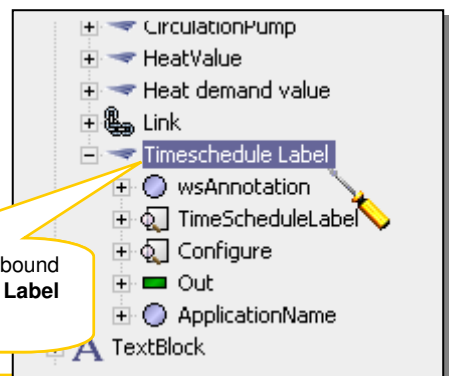
Tip:
If the snap/grid settings prevent you from getting the position exactly right you can 'Nudge' the PxPane by using the combination of <CTRL> key and ←→↑↓ keys



This is also a PXPane widget bound directly to the **Timeschedule Label** mini-application

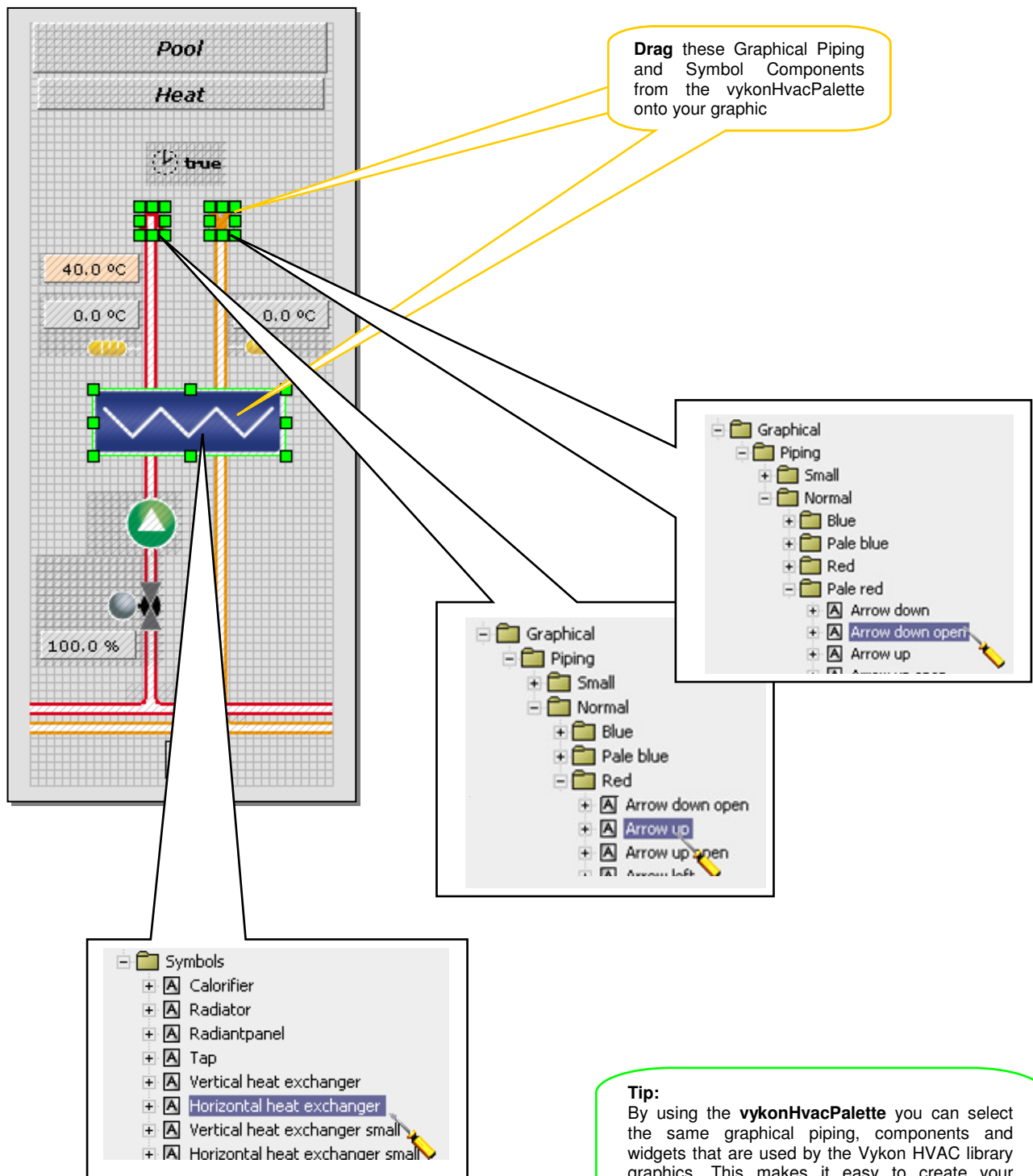
This is also a PXPane widget bound to the **CalculatedValueLabel** of the PID mini-application

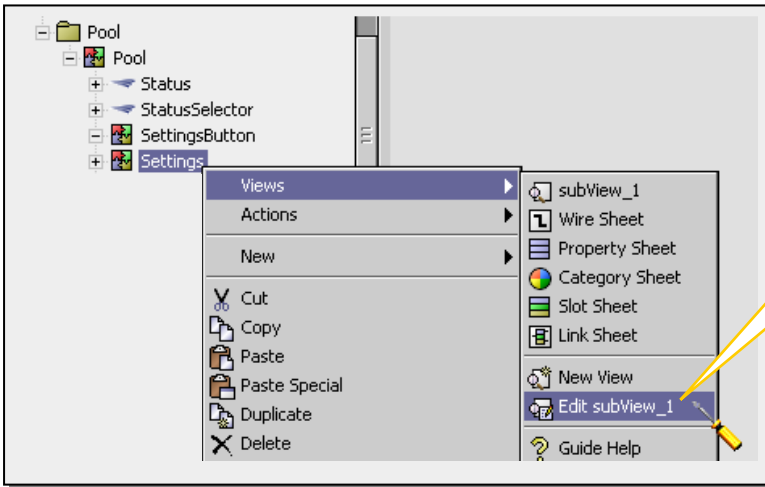
Using the **Duplicate** function create more PXPane widgets with bindings to the remaining control equipment



Tip:
Vykon HVAC makes extensive use of the PXPane widget. This is a **special widget unique** to Vykon HVAC. It allows you to 'nest' views, or put another way, embed one view within another view within another view and so on. This makes it much easier to manipulate graphics and re-use views over and over again within the application graphics

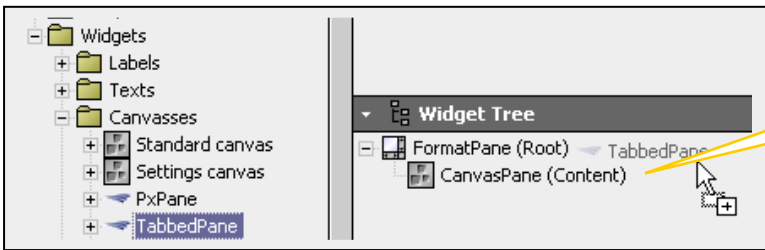
Tip:
By using mini applications and PXPane widgets you can **quickly** create custom graphics to match the Vykon HVAC graphics.



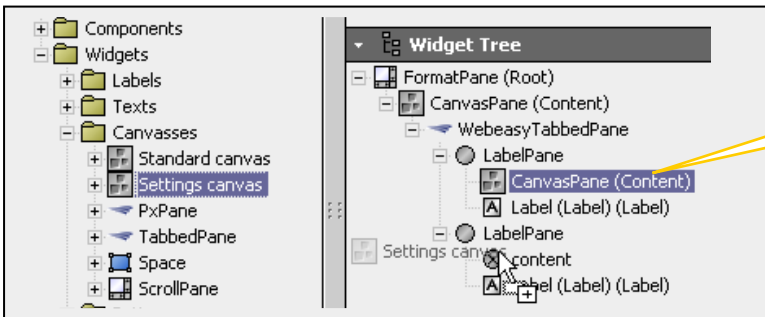


Select the Edit subView_1 of the Settings CustomAppContainer

Tip:
A subView_1 has already been created with default properties ready for your settings widgets

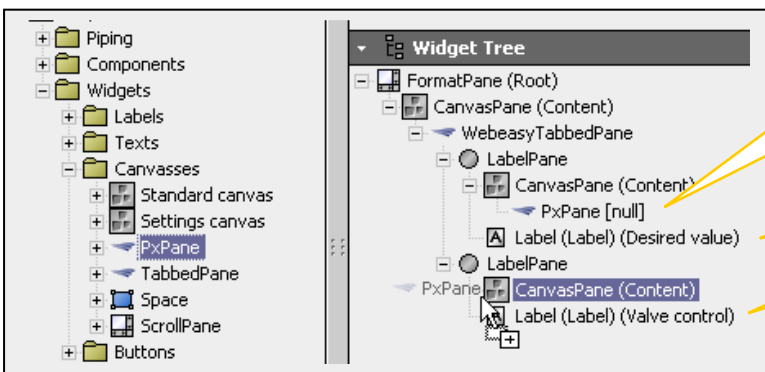


Drop a **TabbedPane** Canvas onto the CanvasPane of subView_1



Drop a **Settings canvas** Canvas onto each LabelPane

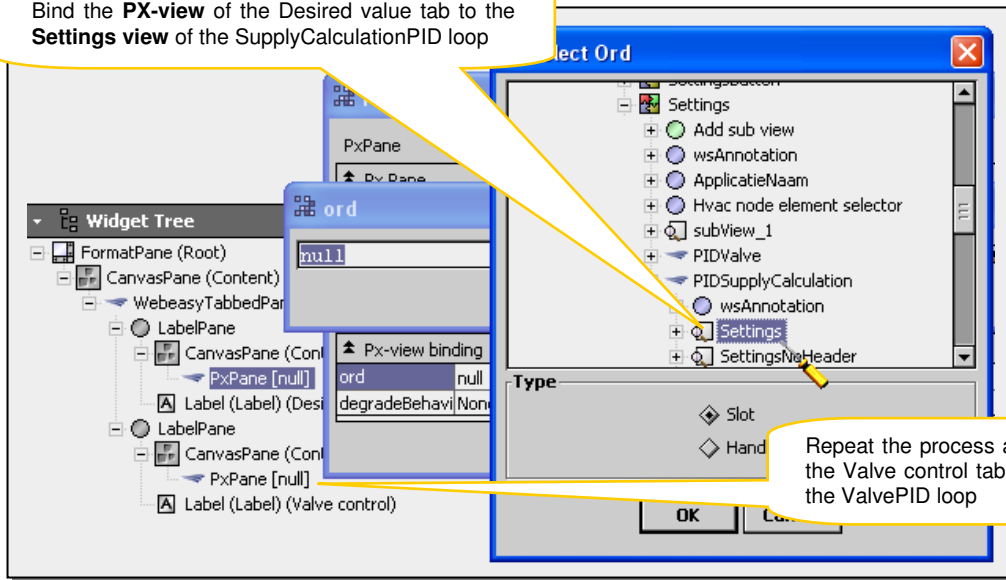
Tip:
It is good practice to use a Canvas because it then allows you to add multiple widgets to the Pane



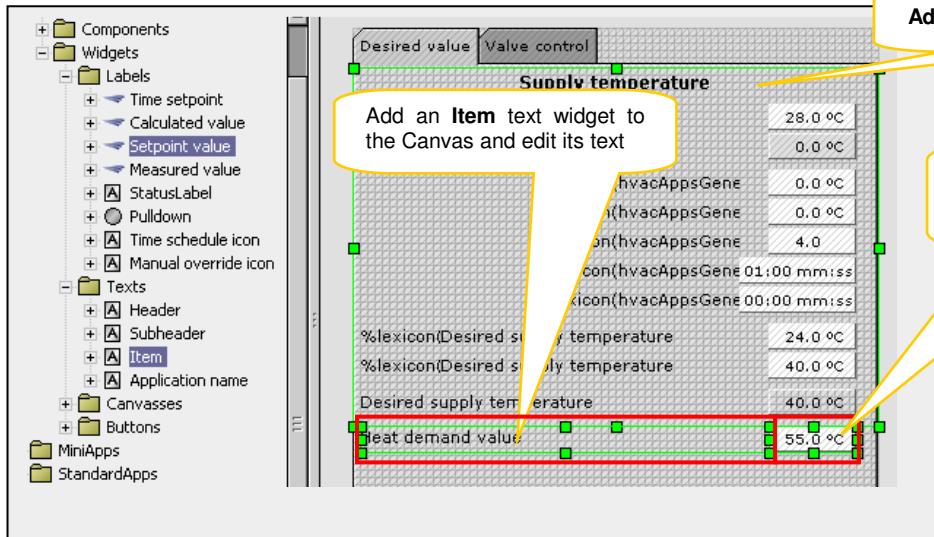
Drop a **PXPane** Canvas onto the CanvasPane of each LabelPane

...and set the **Text** of each Label accordingly

Bind the **PX-view** of the Desired value tab to the **Settings view** of the SupplyCalculationPID loop



Repeat the process and bind the **PX-view** of the Valve control tab to the **Settings view** of the ValvePID loop



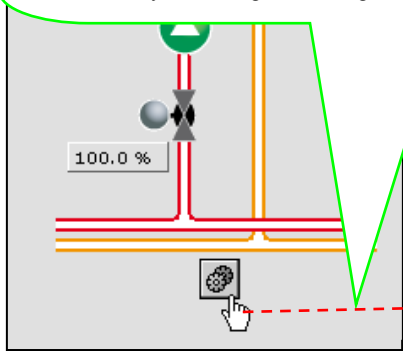
Adjust the position of the PX-views

Add an **Item** text widget to the Canvas and edit its text

Add a **Setpoint value** widget to the Canvas and bind it to the 'Heat Value' In1 value

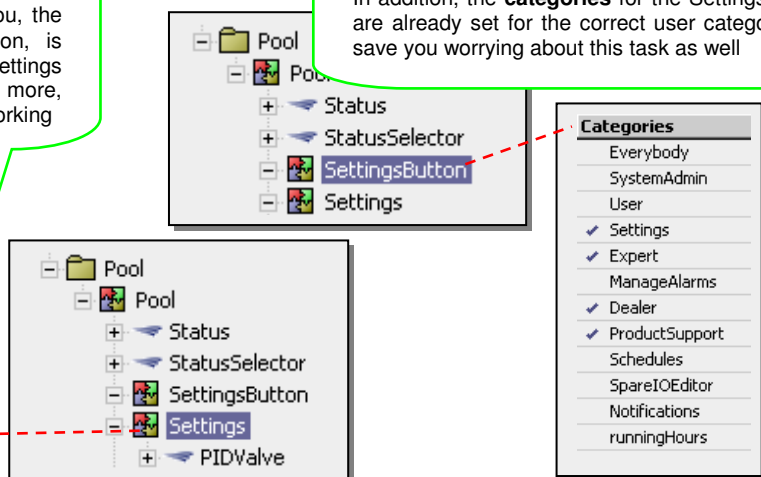
Tip:

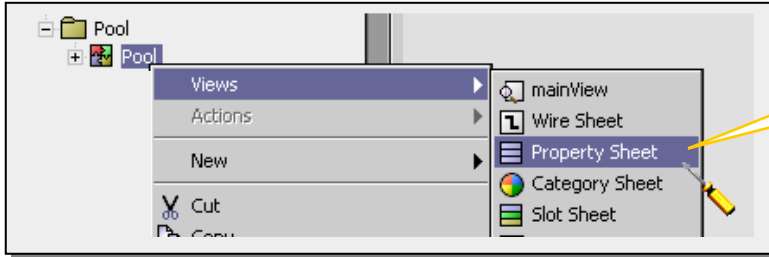
To make life as **easy** as possible for you, the **hyperlink** property of the settings icon, is automatically bound to the view on the Settings container. You do not have to do any more, than create your settings view, to get it working



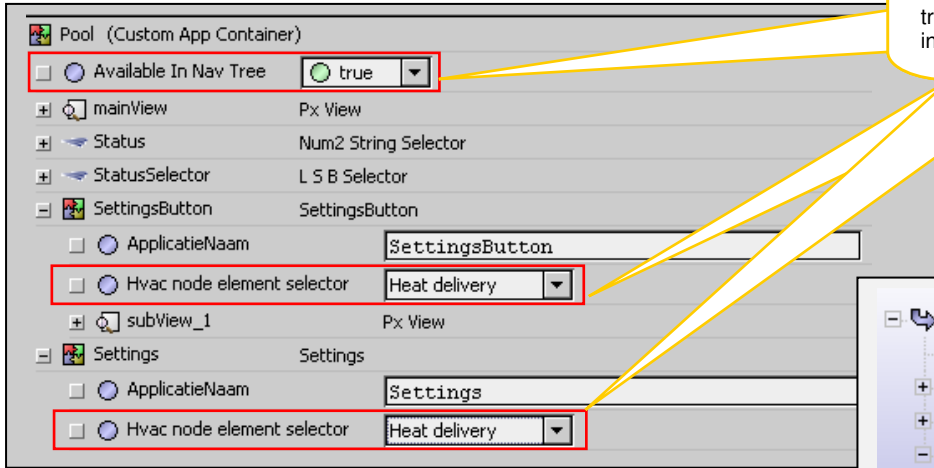
Tip:

In addition, the **categories** for the SettingsButton are already set for the correct user categories to save you worrying about this task as well

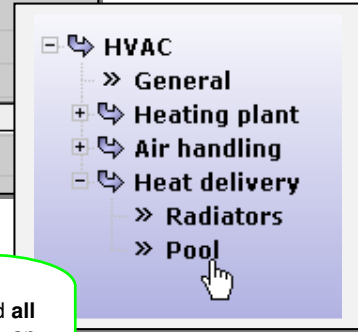




Select the Property Sheet view of the **Pool** CustomAppContainer



Ensure that your Custom container is made available for automatic inclusion in the Nav tree and that it will be positioned into the correct Node element



Tip:
All the Vykon HVAC CustomAppContainers and all the library generated Graphics views have an 'Available in Nav Tree' property which enables you to control which applications are available to view and in which node element they are positioned

And Finally... Action the Synchronize Nav Trees function in the hvacProfileService

