Technical Document

Niagara Scheduling Guide

About this Guide

This topic contains important information about the purpose, content, context, and intended audience for this document.

Product Documentation

This document is part of the Niagara technical documentation library. Released versions of Niagara software include a complete collection of technical information that is provided in both online help and PDF format. The information in this document is written primarily for Systems Integrators. To make the most of the information in this book, readers should have some training or previous experience with Niagara software, as well as experience working with JACE network controllers.

Document Content

This document provides information on the scheduling tool that is available in Workbench or via web browser connection to a station.

Included in the guide are basic procedures for working with schedules, as well as the scheduling references which describe scheduling components and views.

- For an overview of scheduling concepts and functionality, refer to Scheduling overview.
- To begin working with schedules, refer to Weekly schedules.
- For detailed information on scheduling components, refer to the topic on Schedule components.
- For detailed information on scheduling views, refer to the section on Schedule plugins.
- **Document change log**
  Updates (changes and additions) to this document are listed below.
- **Related documents**
  The following documents provide related information.

Document change log

Updates (changes and additions) to this document are listed below.

June 16, 2021

Added HTML5 color support, reorganized and edited topics.

April 9, 2019

In the topic, “About Special Events”, edited description of differences between numbered weeks and calendar weeks.

21 February 2018

Updates for Niagara 4.6.

4 August 2017

In the topic Weekly Schedule Usage Notes, All Day Event added in Scheduler right click menu.

18 August 2015

Initial release publication.

Parent topic: About this Guide

Related documents

The following documents provide related information.

- *Getting Started with Niagara*
- *Niagara Drivers Guide*

Parent topic: About this Guide

Weekly schedules

To provide scheduling control of station components, you place a schedule component from the schedule palette in a station, configure it, and link it as needed to the other component(s).

Types of schedules

Scheduling functionality is the same whether using Workbench or an HTML5 web browser with one exception. In a web browser, you do not have the options to copy and paste components.

There are four types of schedules:

- *Weekly schedules* are the most used schedule components. They define regular, repeating, events and include any number of special events. Four types of weekly schedules correspond to the standard data types: Boolean, numeric, enum, and string. All are identical except for input and output.
- *Calendar schedules* define specific days with scheduling exceptions (for example, holidays) You reference them in the special events setup of weekly schedules.
- *Trigger schedules* initiate actions or topics. With a set of pre-configured schedules and a ScheduleSelector component, you can choose from a list of valid schedules to set up or change the schedule of a device.
- *ScheduleSelector components* select a schedule to use for controlling a component. As needed, you link a ScheduleSelector's output to an action of a control point or extension.

Weekly schedules

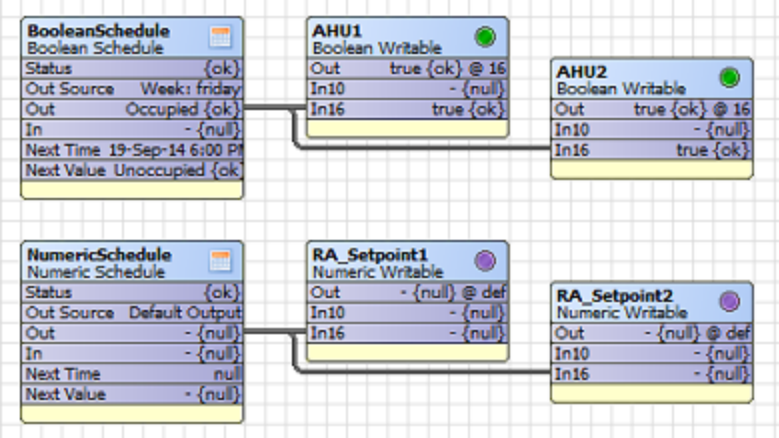
Weekly schedules define the normal, regular (repeating) events by the time of day and the days of the week. These are the most used schedule components.

There are four types of weekly schedules that vary by data category:

- **BooleanSchedule**s set up schedules where two states are possible: true or false, on or off, up or down, etc.
- **EnumSchedule**s set up schedules where more than two conditions apply. For example, you would use an enum schedule to configure fan speed: slow, medium and fast.
- **NumericSchedule**s output numeric values for each of the event blocks.
- **StringSchedule**s output text information for each of the event blocks.

Each of the four is identical except for input and output, which are compatible with the schedule component type. Typically, only outputs from schedules link to other components for scheduling control, such as writable points of the same type.

Figure 1. Schedules linked to writable points



The screen capture shows a **BooleanSchedule** and a **NumericSchedule**, each linked to two writable points. Weekly schedules support status coloring. You can configure colors in **Lexicon** and using Facets.

- **Scheduling overview**
  To provide scheduling control of station components, you place a schedule component from the schedule palette in a station, configure it, and link it as needed to the other component(s).
- **Weekly schedule output processing**
  Each component type varies only by data value category, meaning its Out slot (generally called output) and its In slot (generally called input). The schedule's output and input are Status <Type> according to schedule component type. For example, StatusBoolean if a **BooleanSchedule**, StatusEnum if an **EnumSchedule**.
- **Adding a schedule component from the palette**
  In your station you can copy a new schedule component directly from the schedule palette sidebar.
- **Copying a pre-configured schedule**
  Using Workbench, you can copy a previously configured schedule component from a saved bog file or other station database.
- **Setting up a weekly schedule**
  Weekly schedules define regular, repeating events, such as the normal hours that a space is occupied. The following is one way to approach configuration.
- **About Special Events**
  Special events apply to weekly schedules only, and are considered any exception to the (normal) weekly schedule. Special events can be one-time only event changes or recurring event changes, such as holidays. Configuration includes both day(s) of occurrence and related time-of-day events. Each weekly schedule component has its own special events, configured on the **Special Events** tab of the scheduler view. If using Workbench the tab is located at the bottom-left corner of the view. If using a web browser, the tab is in the upper left corner of the view.
- **Configuring a special event in a weekly schedule**
  Special events are exceptions to the regular weekly schedule, and typically include recurring holidays and one-time events.
- **Setting up permissions for the Special Events tab**
  Typically, a user with operator-level read permissions on a weekly schedule would see all the schedule tabs but would not be able to edit the schedule. However, this optional procedure configures permissions for the **Special Events** tab to allow operations users write access only for managing special events.
- **Event colors in weekly schedules**
  Configured normal and special events in weekly schedules (Boolean, Enum, Numeric and String) display different colors according to the event state.
- **Reviewing a weekly schedule's configuration**
  In Workbench you use the read-only **Summary** tab to view a weekly schedule's configuration.
- **Weekly schedules links**
  Schedule component links provide scheduling control over other components. Using a weekly schedule and a trigger schedule, you can link the weekly schedule's Out slot as a source to a slot on another component. You can link the same weekly schedule to many target components.

- **Schedule exports and imports (master/slave)**
  Using the driver architecture, you can create master/slave schedules to share schedule configuration among devices. This allows you to globally update the configuration of any slave schedule by making changes to its master schedule.

Scheduling overview

To provide scheduling control of station components, you place a schedule component from the schedule palette in a station, configure it, and link it as needed to the other component(s).

Scheduling functionality is the same whether using Workbench or an HTML5 web browser with one exception. In a web browser, you do not have the options to copy and paste components.

There are four types of schedules:

- *Weekly schedules* are the most used schedule components. They define regular, repeating, events and include any number of special events. Four types of weekly schedules correspond to the standard data types: Boolean, numeric, enum, and string. All are identical except for input and output.
- *Calendar schedules* define specific days with scheduling exceptions (for example, holidays) You reference them in the special events setup of weekly schedules.
- *Trigger schedules* initiate actions or topics. With a set of pre-configured schedules and a ScheduleSelector component, you can choose from a list of valid schedules to set up or change the schedule of a device.
- *ScheduleSelector components* select a schedule to use for controlling a component. As needed, you link a ScheduleSelector's output to an action of a control point or extension.

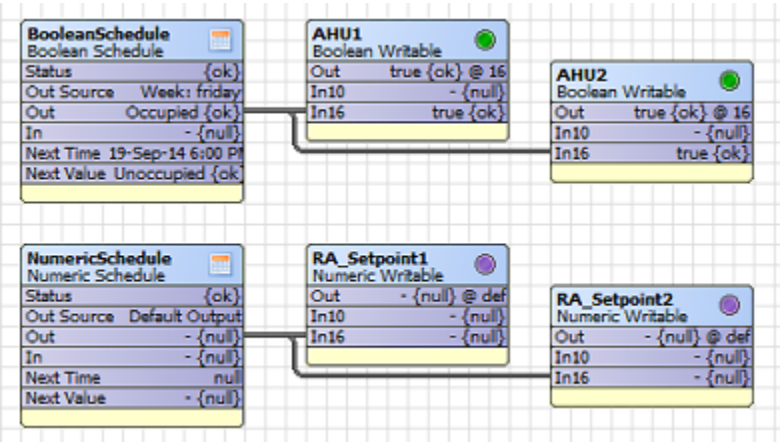
Parent topic: Weekly schedules

Weekly schedule output processing

Each component type varies only by data value category, meaning its Out slot (generally called output) and its In slot (generally called input). The schedule's output and input are Status <Type> according to schedule component type. For example, StatusBoolean if a **BooleanSchedule**, StatusEnum if an **EnumSchedule**.

Typically, only outputs from schedules link to other components for scheduling control, such as writable points of the same type.

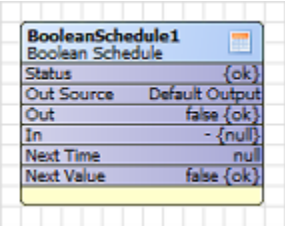
Figure 1. Schedules linked to writable points



The screen capture shows a **BooleanSchedule** and a **NumericSchedule**, each linked to two writable points. Weekly schedules support status coloring. You can configure colors in **Lexicon** and using Facets.

Output

Each component has an Out slot and an Out Source slot. In addition, Next Time and Next Value slots are available. By default, these slots are pinned on the component's shape on the **Wire Sheet**, as shown below.



Output recalculation occurs upon any of the following:

- Any saved change to its configuration
- Any change at its input
- Station startup
- Any change to the system clock

Upon any output change, all of these slots are updated.

Out slot

A schedule's output value is determined by the following, in highest-to-lowest priority:

Priority
Description

1

To any non-null value at its In slot (if linked) This value is immediately passed to its output. Otherwise (if null), processing continues.

2

If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any).

3

To the active weekly schedule event (if any).

4

To the default output value.

Out Source slot

Out Source provides a string "source description" of the current output, as one of the following:

- Input

- Special Event: <SpecialEventName>
- Week:<day_of_week>
- Default Output

Examples: “Week: monday” or “Special Event: Christmas Break”

Parent topic: Weekly schedules

Adding a schedule component from the palette

In your station you can copy a new schedule component directly from the schedule palette sidebar.

You are connected to an open station.

1. Open the schedule palette.
2. Drag a schedule component, such as **BooleanSchedule** or CalendarSchedule, to one of these locations:
   - In the Nav tree, onto a folder in your station
   - In the view pane, onto the **Wire Sheet** or **Property Sheet** view of a folder in your station
3. In the **Name** window, give the schedule a name or use the default name and click **OK**
   The schedule or calendar is now in your station, with default values.
4. Double-click your new schedule to display its scheduler view. At this point, you configure the schedule by adding events.

Note: You can open a chart view in the schedule component. A **BooleanSchedule**, **EnumSchedule** and **NumericSchedule** is chart-able. This is useful in determining when a state change is expected to occur.

Parent topic: Weekly schedules

Copying a pre-configured schedule

Using Workbench, you can copy a previously configured schedule component from a saved bog file or other station database.

A pre-configured schedule component exists in a bog file or other station database.

Note: Currently, in a web browser connection you do not have the option to copy and paste components.

---

1. In the Nav tree, expand **My File System** and navigate to the saved bog file or station config.bog file.
2. When you locate the desired schedule component, right-click and select **Copy** or drag the file to copy itto either of these locations:
   - A folder in your station’s Nav tree
   - The **Wire Sheet** or **Property Sheet** view of a folder in your station

   The Name window opens.
3. Type a name and click **OK**.
   The schedule component with pre-configured values is now in your station.

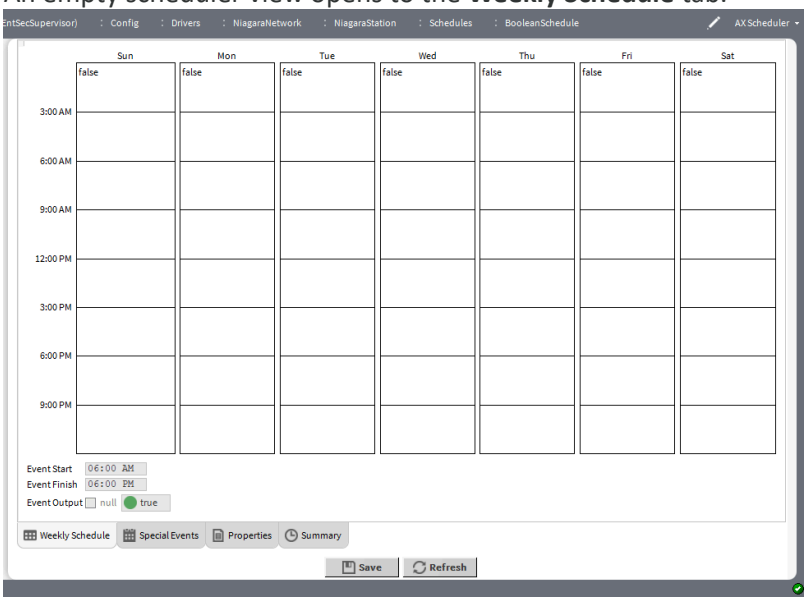
Parent topic: Weekly schedules

Setting up a weekly schedule

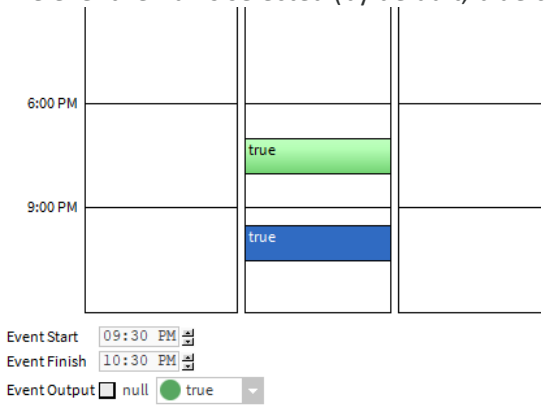
Weekly schedules define regular, repeating events, such as the normal hours that a space is occupied. The following is one way to approach configuration.

A weekly schedule exists in your station.

1. If you are adding an enum schedule, right-click **EnumSchedule**, click **Views > AX Property Sheet** and define its range Facets before continuing to add events.
2. Click the schedule component and select the **Weekly Schedule** tab.
   An empty scheduler view opens to the **Weekly Schedule** tab.

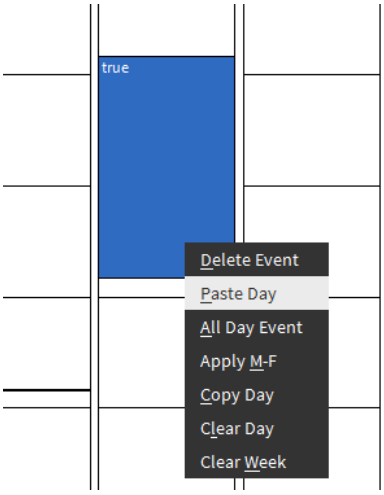


3. To add a new event time range, click in a day at the approximate event start time, and drag down to define the start and finish time.
   The event remains selected (by default, blue colored) when you release the mouse button.



   The framework assigns default colors to events depending on the type of weekly schedule. You may configure these colors for each schedule using Facets or globally using the Lexicon.

   Right-clicking an event opens this menu, which provides the most event-scheduling commands.



4. As needed, do one of the following:
   - Click again and drag on the event's top or bottom edges to change its start or finish times (in broad increments).
   - Right-click on a scheduled block and use the popup menu to copy and paste event times across predefined days of the week, to delete an event or to clear a day or week.
   - To fine-tune an event's start and finish times, use the hour, minute and AM/PM option lists.
   - To configure the event to occur all day long, right-click the day and click All Day Event.
   - To clear the whole week, right-click anywhere in the schedule and click Clear Week.

Note:

For any event, start time is inclusive, and the event extends to (but is exclusive of) the end time. In other words, there is no output blip between adjacent events, even if across days. For example, if a Monday event ends at midnight, a Tuesday event starts at midnight. The schedule output is continuous (providing both events have the same Output value).

---

5. If a you have a Boolean or enum schedule, select the Event Output value from the drop-down control. If yours is a numeric or string schedule type a value.
   The framework routes this value to the access device at the scheduled time(s).
6. Click the **Properties** tab, or, in a web browser connection, select the **Property Sheet** view, and configure the following properties. Leave other properties at their default values:

   - Facets — This is critical if you are configuring an **EnumSchedule**, and optional if you have a **BooleanSchedule** or **NumericSchedule**.

     Note: Facets do not apply if you are configuring a **StringSchedule**.

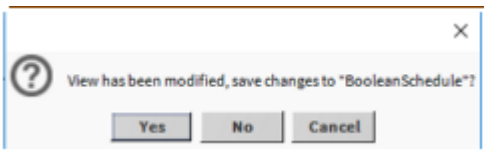
   - Default Output — This is the schedule's Event Output when an event (either weekly or special event) is not active. It framework also uses it for the hours that are not scheduled (not effective).

7. To continue, click **Save**.
   For weekly schedules (tabbed scheduler view), you should save while working in each tab, even though any save applies to changes made on all tabs.
8. To refresh the view, click **Refresh**.

   Refresh does one of two things:

   - If the **Save** button is not available (there are no unsaved changes), clicking **Refresh** re-synchronizes the view with the component's current configuration.

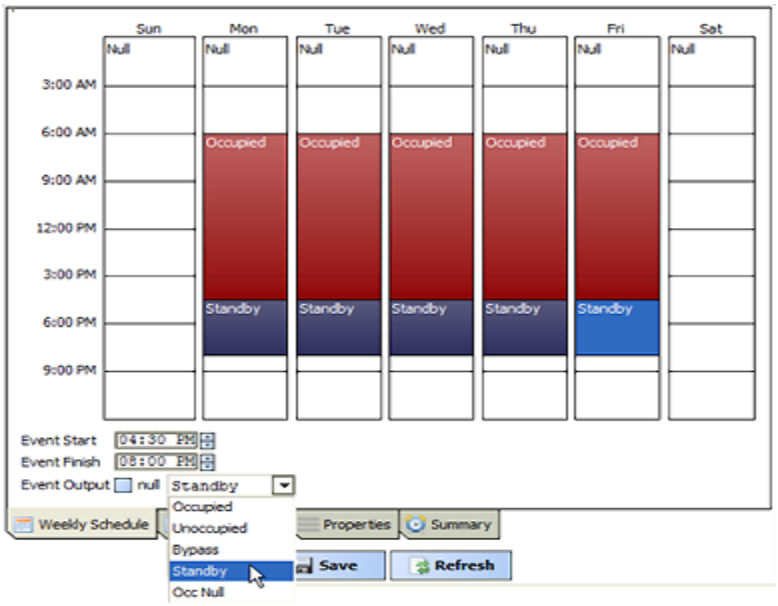
- If the **Save** button is available (unsaved changes exist), clicking **Refresh** produces a confirmation window:



  You can move among tabs without losing unsaved data, however, you must click the **Save** button before leaving the view or data are lost.

9. If the schedule contains unsaved changes, click **Save**.

This screen capture shows output selections for an **EnumSchedule** with its range facet defined as "lonworks:LonOccupancyEnum," one of the available frozen facets.



Configured events reflect a few default colors available for **EnumSchedule**s.

Parent topic: Weekly schedules

About Special Events

Special events apply to weekly schedules only, and are considered any exception to the (normal) weekly schedule. Special events can be one-time only event changes or recurring event changes, such as holidays. Configuration includes both day(s) of occurrence and related time-of-day events. Each weekly schedule component has its own special events, configured on the **Special Events** tab of the scheduler view. If using Workbench the tab is located at the bottom-left corner of the view. If using a web browser, the tab is in the upper left corner of the view.

Each weekly schedule component has its own special events configured on the **Special Events** tab in its scheduer view. Event times (and values) entered for any special event apply to that schedule only.

Special events can occur one-time only or recurr, such as holidays.

The configuration of special events includes the day(s) of occurrence and related time-of-day events. In the time-of-day event definitions, you can intermingle them with regular weekly events, or completely override the weekly schedule.

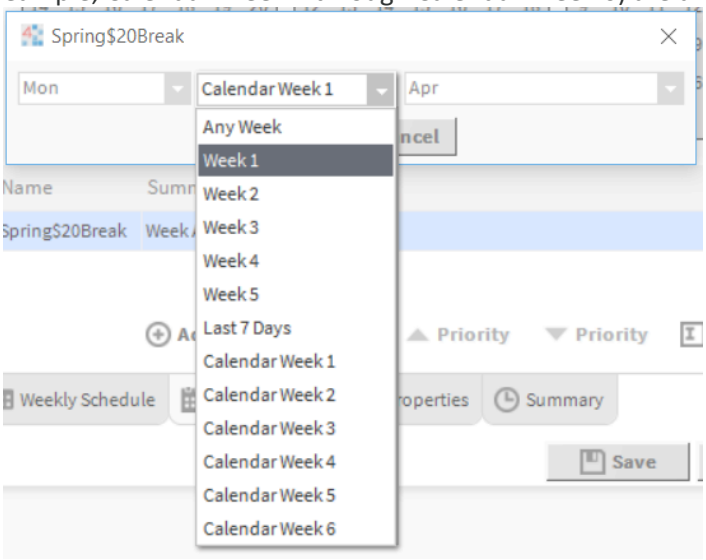
You can visually prioritize special events via list order. This allows any overlapping special events to occur in an ordered fashion.

If the special event is a reference type event (it references a separate schedule), the **CalendarSchedule** component specifies the days of occurrence. This globally changes the days that special events occur in weekly schedules by editing one or more referenced calendar schedules.

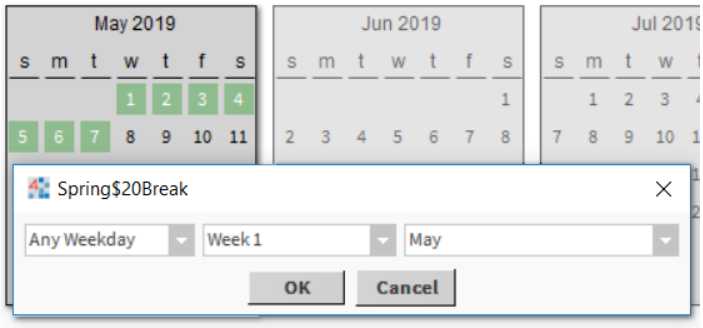
Note: You can configure permissions for the special events of a weekly schedule (**BooleanSchedule**, **NumericSchedule**, etc.) than for the rest of the schedule's configuration.

The **Special Events** tab has monthly calendars at the top, the special events table and a 24-hour time pane. The special events table lists all existing schedules by name and summary. At the bottom of the table, controls add, edit and delete schedules. In addition, up and down links sort schedules in order of priority (in case of schedule overlaps). Schedules at the top have the highest priority.

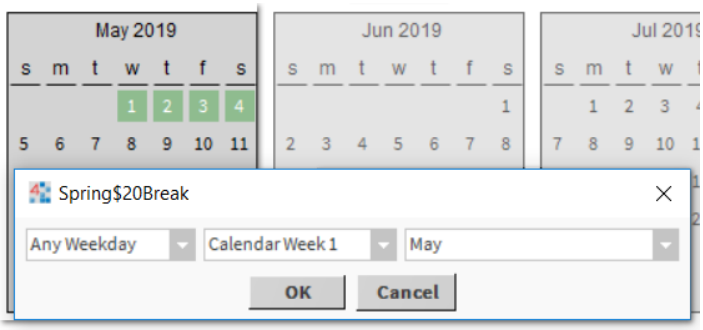
Note: In special events, a numbered Week (for example, Week 1 through Week 5) and a calendar week(for eample, Calendar Week 1 through Calendar Week 6) are available.



The Week 1 through Week 5 options divide the month into five seven-day increments, where each week does not necessarily start and end Sunday-to-Saturday. Instead, the week starts on the specified day one and ends after day seven. The exception would be Week 5, which could have fewer than seven days when the end of the month occurrs mid-week.
Figure 1. Week 1 comprised of days highlighted in green (May 1–May 7)



By contrast, a calendar week typically starts and ends Sunday-to-Saturday. Although, if the first day of the month occurs in the middle of the week, Calendar Week 1 contains only the remaining days of that first week, ending in Saturday, as shown here.
Figure 2. Calendar Week 1 comprised of days highlighted in green (May 1–May 4)

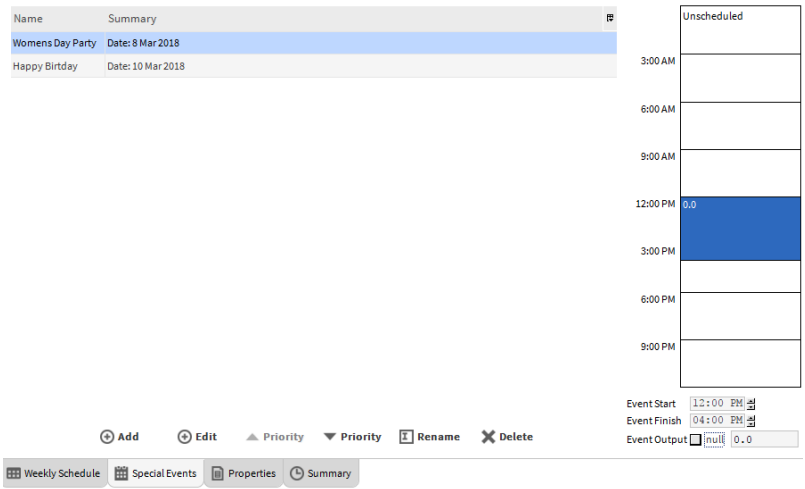


Event times and output values

A newly-created special event has no events defined. With the special event selected, click in the right-side events column and enter events as necessary. Start, finish, and output controls work the same as in the **Weekly Schedule** tab.

You can also right-click in the column for an event menu. This is useful to add an all-day event or set the entire day to the schedule's default value.

Figure 3. Special event schedule actions



Note: You must specify events for any special event to occur. Where nothing is scheduled, the special event relinquishes control back to any lower-priority schedule events, and finally intermingles with the weekly schedule. To completely override the events configured for a day in the weekly schedule, configure a special event for the entire day.

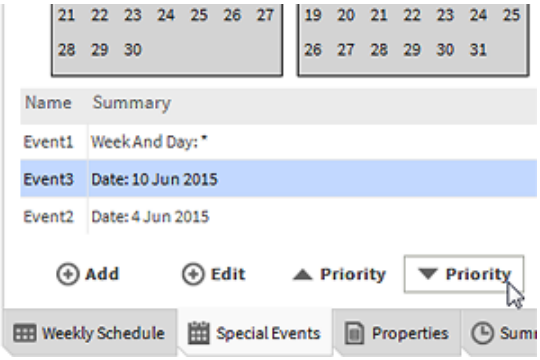
Special event priorities

All special events take priority over regular weekly events. Among special events, the order in which the events appear in the table defines relative priorities among special events table, as follows:

- Highest priority is at the top of the list. This special event, when active, always occurs.
- Lowest priority is at bottom of list. Events occur only if not overlapped by other special events that are active during the same period.

To change a special event's priority, select it and use the priority arrow buttons, as shown in the following image.

Figure 4. Change priority by listing order



Right-click menus and other controls

Right-clicking in the special events table invokes menu options that match the buttons located below the table.

Options may include the following:

- **Add** creates a new event.
- **Edit** opens an existing event so you can update it.
- **Priority** up moves the event higher in the table.
- **Priority** down moves the event lower in the table.
- **Rename** opens a window so you can change the event name.
- **Delete** removes the event from the calendar.

Parent topic: Weekly schedules

Configuring a special event in a weekly schedule

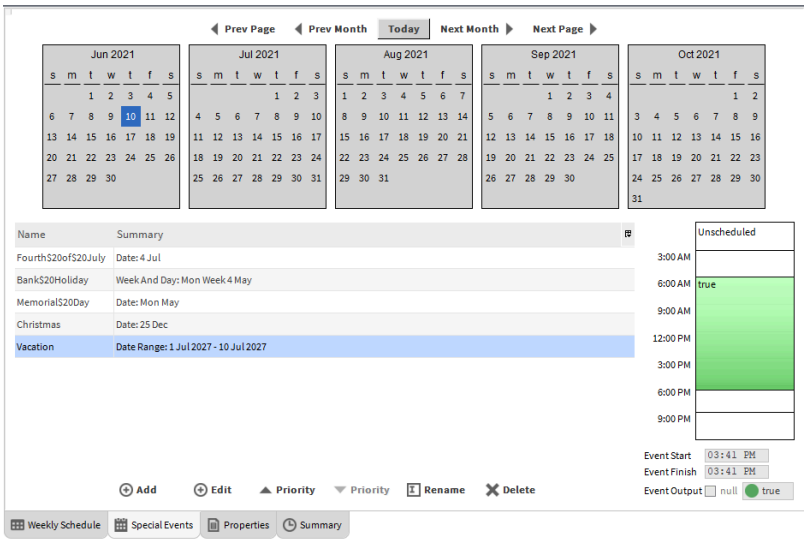
Special events are exceptions to the regular weekly schedule, and typically include recurring holidays and one-time events.

A weekly schedule component exists in your station.

1. Double-click the weekly schedule component.
   One of the scheduler views opens.
2. Click the **Special Events** tab.
   When you first access this tab, the framework highlights the current day in the left-most calendar month at the top of the view.
3. As needed, click on **Next Month** and **Prev Month**, or **Next Page** and **Prev Page** to traverse the calendar ahead or back in time.
4. To add a new special event, click **Add**.
   The **Add** window opens.
5. Provide a unique, identifiable Name for the event and select the Type.
   Name examples: “Christmas Day” or “Half-Day.” Name defaults to `Event`. You can leave the name at its default value and change it later.

   Type controls the additional properties in the **Add** window.

6. Configure the remaining properties and click **OK**.
   The framework adds the new event to this schedule’s special events table. The event remains selected for further editing, except for Type.
   Figure 1. Special Events tab in weekly Scheduler



   When you select a special event, if the event occurs in any currently displayed month, the framework highlights its day(s) of occurrence in the monthly calendars at the top of the view, and displays its associated event actions in the right-side column. It remains selected for further editing, except for Type.

Note: A special event must have at least one defined event action to be highlighted in a calendar.

---

7. To manage events in the table choose one of the following:
   - To edit an existing event, click **Edit**.
   - To change the location of the event row in the table, select the event and click the **Priority** up and **Priority** down arrows.
   - To rename the event, select it and click **Rename** ( ).
   - To delete a special event, click **Delete**.
8. With the new event selected, on the far right-side of the tab click and drag in the time-picker (or right-click) to configure the time for this event.

A newly-created special event has no events defined. You must specify at least one time range for any special event to occur. Where nothing is scheduled, the special event relinquishes control back to any lower-priority schedule events, and finally intermingles with the weekly schedule.

Note: To completely override the weekly schedule, configure a special event for the entire day.

9. Use the Event Start and Event Finish properties to fine-tune event times.
10. Select the desired Output value and click **Save**.
    The Event Output value is the number or text you want to identify the event in the event block.
11. To return to the current calendar month and day by clicking the **Today** button above the gray calendars.

Parent topic: Weekly schedules

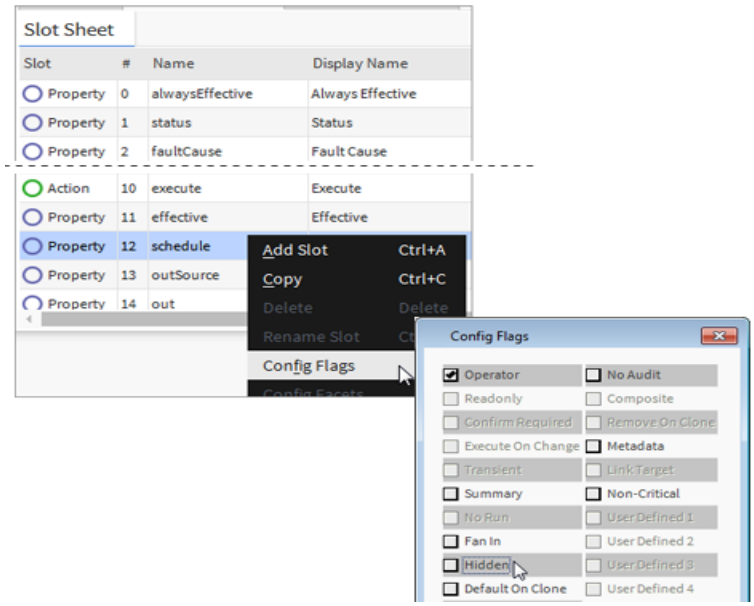
Setting up permissions for the Special Events tab

Typically, a user with operator-level read permissions on a weekly schedule would see all the schedule tabs but would not be able to edit the schedule. However, this optional procedure configures permissions for the **Special Events** tab to allow operations users write access only for managing special events.

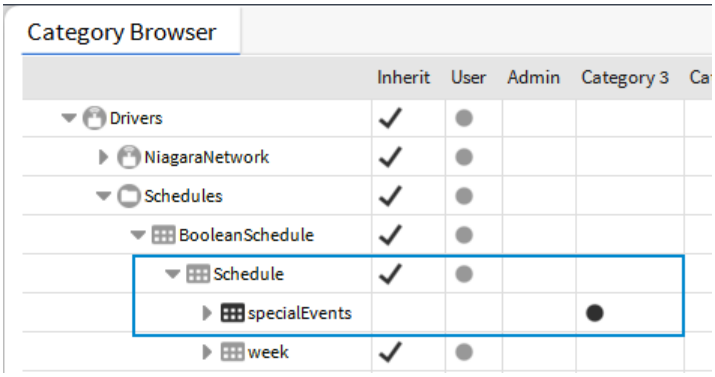
A weekly schedule exists with operator-level read permissions on all tabs of the scheduler view. You are logged in as an admin-level user.

The permission change described here applies only to the schedule's **Special Events** tab. It does not alter permissions for the other schedule tabs and properties.

1. Open **Slot Sheet** view for the schedule component.
2. Right-click the Schedule slot and select Config Flags
   The **Config Flags** window opens.



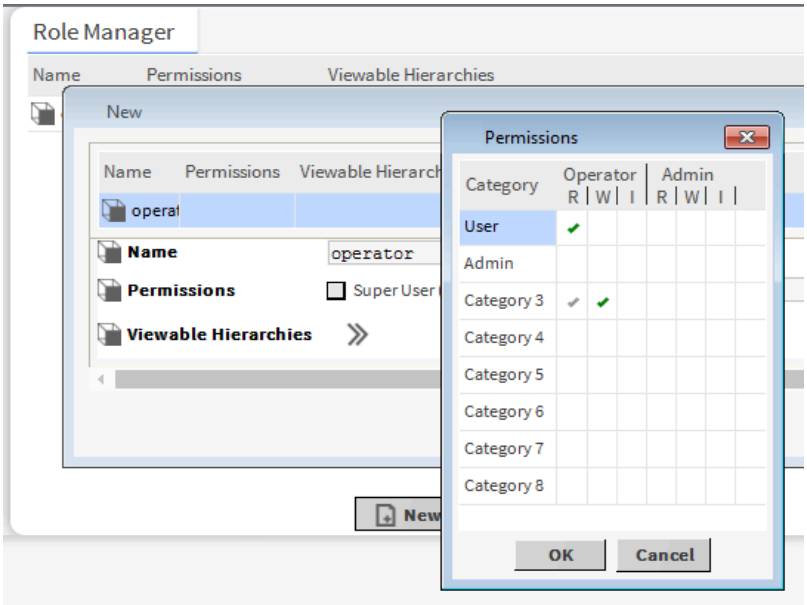
3. Click to deselect the Hidden check box and click **OK**.
   Once the schedule slot is no longer hidden, you can expand it in the **CategoryBrowser** and make changes to the schedule component.
4. To display the **CategoryBrowser** view, in **Services**, double-click **CategoryService**, expand and the schedule component's child **CompositeSchedule** to expose specialEvents and perform the following:
   a. Click to deselect the Inherit option.
   b. Click to clear the assigned category.
   c. Click to select a different category—one that a user could have operator write permissions on, such as Category 3, as shown here:



d. To continue, click **Save** in the Workbench toolbar.

The child specialEvents CompositeSchedule is now assigned to a different category than that of the top-level schedule component.

5. If you have an operator role already configured with read and write permissions, skip to the next step to assign the role to a user, otherwise, to configure the role now perform the following steps:
   a. To add the new role, expand **Config > Services**, double-click **RoleService**, click **New** and click **OK**.
   b. In the second **New** window, type Operator in the Name property and click » icon located to the right of Permissions.
   c. In the **Permissions** window, click to select the following operator-level permissions: User (default category 1), click R, and for Category 3 click R, W, (as shown here), and click **OK**.



The framework creates operator role (with necessary permissions). You can now assign the role to one or more users.

6. To assign the role to a user, under **Services**, double-click **UserService**, select a user in the **User Manager** view, click **Edit**, then scroll down to Roles click the check box to select the **Operator** role and click **OK**. The user now has read access on all of the tabs on this scheduler view and write access only for managing events on the **Special Events** tab.

Once logged in to the station, the user can manage that schedule's special events as needed.

Parent topic: Weekly schedules

Event colors in weekly schedules

Configured normal and special events in weekly schedules (Boolean, Enum, Numeric and String) display different colors according to the event state.

Default colors and schedule types

These are the default colors for all four schedules:

Schedule Name	Color	Value
Boolean Schedule	Green	true
	Red	false
	Gray	null
Enum Schedule	14 default colors	This has 14 default colors. The color cycle repeats if there are more that fourteen enum values.
	silver gray	null
Numeric Schedule	Green	any event
	Gray	null
String Schedule	Green	any event
	Gray	null

- **Configuring current schedule event colors**
  Event colors that are configured at the component level, apply to the current schedule. You use facets to configure these colors. Facet-configured event colors override any colors globally sourced from schedule lexicon keys (if applicable), as well as default colors.
- **Configuring global event colors**
  The Lexicon configures colors that apply globally across all schedules, this is not a station specific change.

Parent topic: Weekly schedules

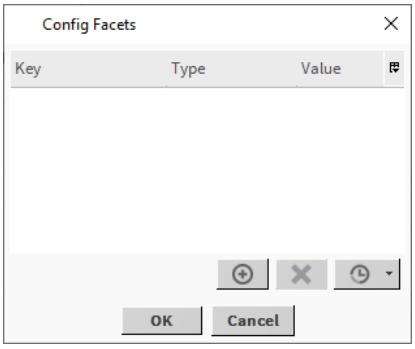
Configuring current schedule event colors

Event colors that are configured at the component level, apply to the current schedule. You use facets to configure these colors. Facet-configured event colors override any colors globally sourced from schedule lexicon keys (if applicable), as well as default colors.

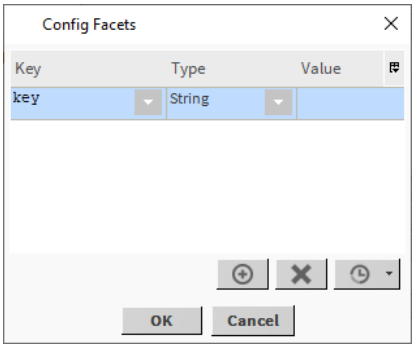
Your schedule exists in the station.

This procedure uses the **EnumSchedule** as an example.

1. Expand **Config > Drivers > NiagaraNetwork > NiagaraStation > Schedules**, right-click **EnumSchedule** and click **Views > AX Property Sheet**.
2. To open the **Config Facets** window, click the Facets chevron (»).
   The **Config Facets** window opens.

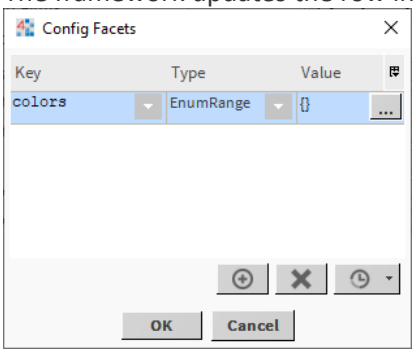


3. To add a new key for colors, click ⊕.
   The framework adds a row to the table.

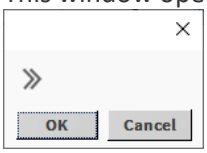


4. Specify the following properties:
   a. For Key, enter the word colors in the key column.
   b. For Type, select EnumRange from the drop-down list.

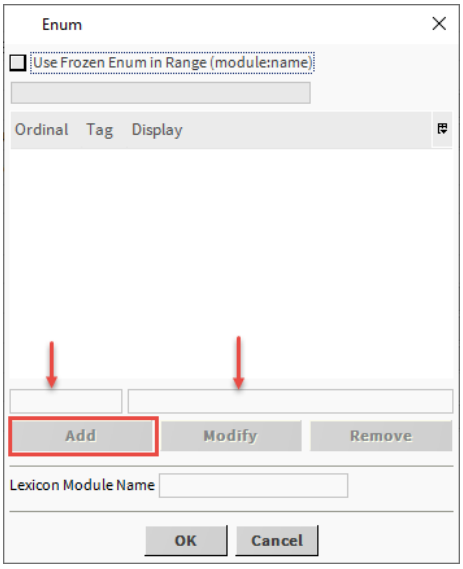
   The framework updates the row in the table.



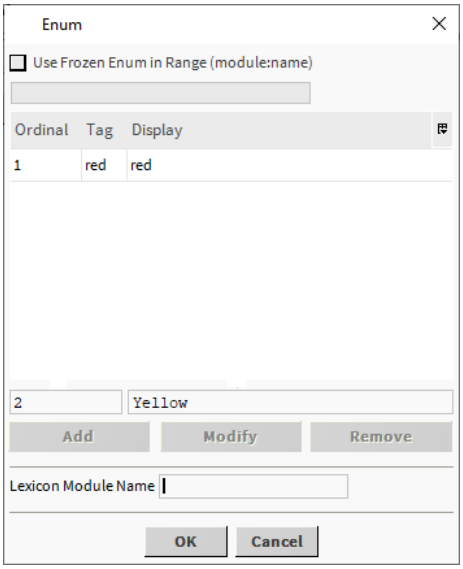
5. To define the range of colors for enum schedules, click … at the right end of the row.
   This window opens:



6. To open the Enum window, click ».
   The **Enum** window opens.

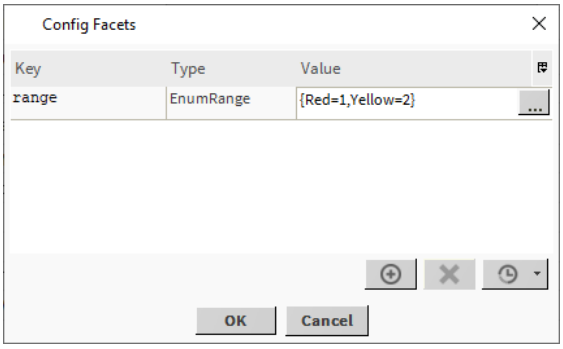


7. To add facet entries for colors click the left-side (Ordinal) field and enter a number, then in the right-side field enter either a color name or color code and click **Add**.
The framework adds the color definition as a row in the enum facets table.

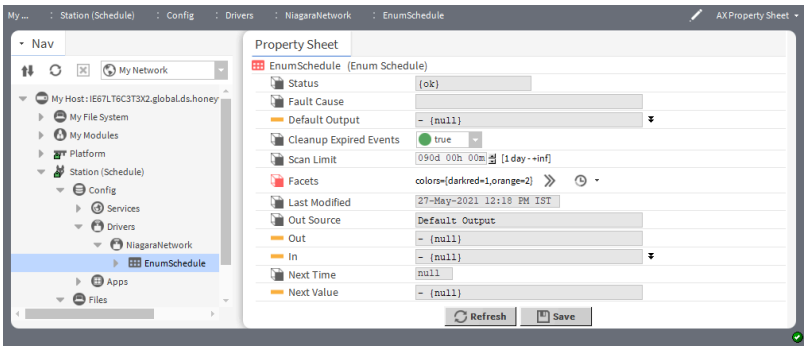


The example above shows colors assigned to two enum states corresponding to ordinals 1 and 2. For **BooleanSchedule**s, the ordinals used are 0 (false) and 1 (true). For numeric and string schedules use the ordinal value 0 (bajaux only).

8. After defining all the colors you need (enum facets can assign many different colors), return to the **Config Facets** window by clicking **OK**.
The framework displays the color values you added in the Value column of the table.



9. To return to the schedule's **Property Sheet** click **OK**, then, to save the **Property Sheet**, click **Save**.



The schedule immediately reflects the new event colors. Each enum schedule supports a total of 14 colors beginning with zero (0). If you specify an ordinal number greater than 13, the assigned colors repeat beginning with the color associated with zero (0).

Ordinal Value, Color

0, dark red

1, dark green

2, dark purple

3, dark blue

4, dark indigo

5, dark orange

6, gray

7, red

8, green

9, purple

10, blue

11, indigo

12, orange

13, khaki

null, silver gray

14, repeat 0

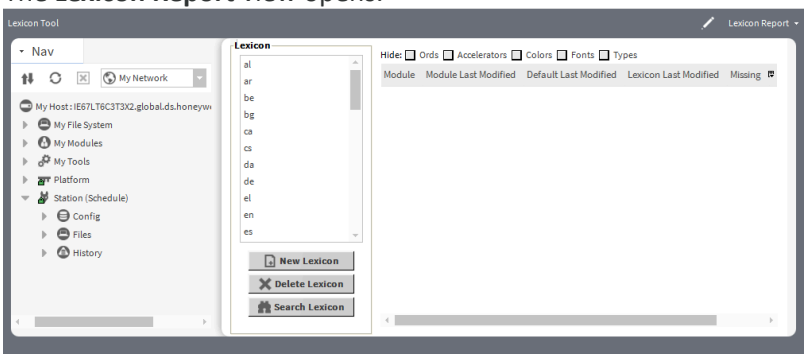
15, repeat 1

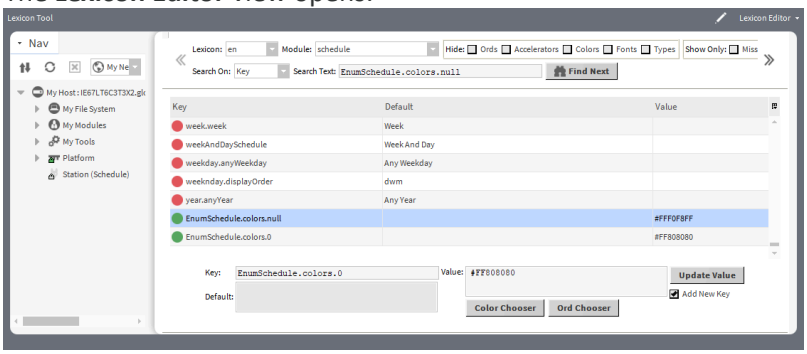
Parent topic: Event colors in weekly schedules

Configuring global event colors

The Lexicon configures colors that apply globally across all schedules, this is not a station specific change.

You are working in Workbench connected to a Supervisor station.

1. Click **Tools** on the menu bar and select **Lexicon Tool**.
   The **Lexicon Report** view opens.
   

2. Using the drop-down list in the upper right corner of the window, click **Lexicon Editor**.
   The **Lexicon Editor** view opens.
   

3. From each drop-down list, select these properties:
   a. For Lexicon select the language (en for English).
   b. For Module select schedule.
   c. For Search On select key.
4. To add a new lexicon key, enable **Add New key** at the bottom of the **Lexicon Editor** window.
5. For key, type a schedule description, such as TypeSchedule.colors.null.
   Below are the examples for key for the different schedules:

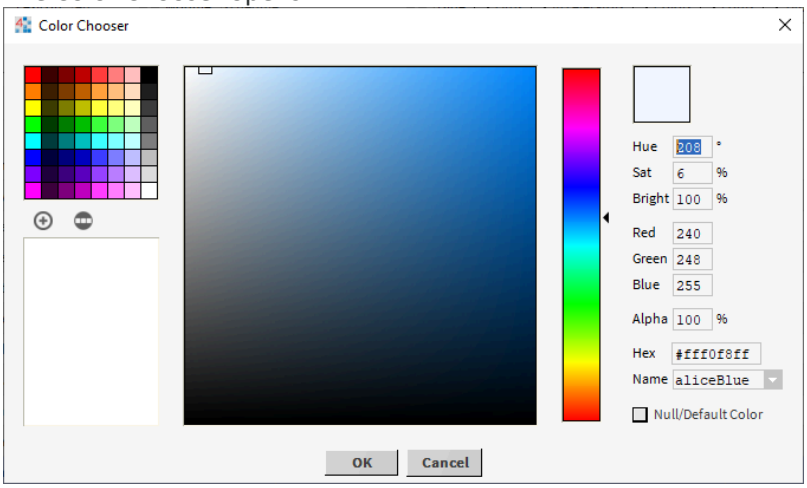
   - For an enum schedule the key is EnumSchedule.colors.n , where n is null or an ordinal integer, for example: 0, 1, 2 and so on. Each integer value represents a color for a scheduled event.

   - For a Boolean schedule the key value is BooleanSchedule.colors.n, where n is 0 (true) or

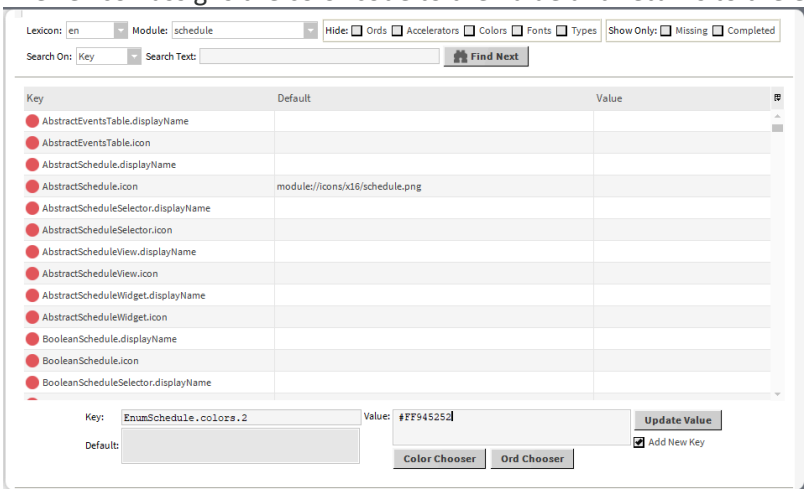
1 (false) or null. True defaults to green, false defaults to red and null defaults to gray.

- For a numeric schedule the value for key is NumericSchedule.colors.n , where n is 0 (true) or null. Zero defaults to green, which is for any configured event and null defaults to gray.
- For a string schedule the value for key is StringSchedule.colors.n , where n is 0 (true) or null. Zero defaults to green, which is for any configured event and null defaults to gray.

6. For Value click the **Color Chooser** button.
   The Color Chooser opens.



7. Select the color and click **OK**.
   The Lexicon assigns the color code to the Value and returns to the editor.



8. Click **Update Value**.
   This updates the Lexicon key for a single color.
9. After configuring all colors, restart the station and Workbench.
   The changes become effective.

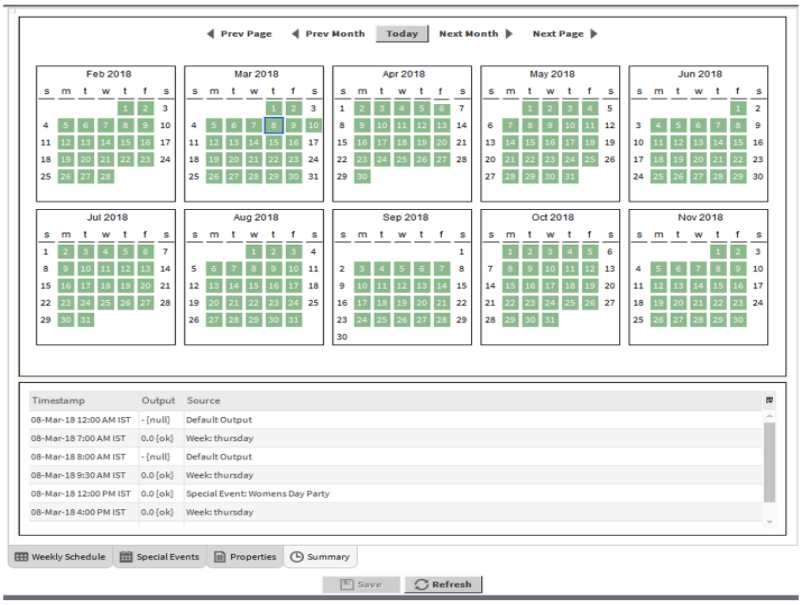
Parent topic: Event colors in weekly schedules

Reviewing a weekly schedule's configuration

In Workbench you use the read-only **Summary** tab to view a weekly schedule's configuration.

Your station has a configured weekly schedule.

1. In the scheduler view, click the **Summary** tab.
   The calendar shows the current day with its scheduled events.



   The table below the calendar shows the Output and output Source.

2. Click any day on any calendar month to see its scheduled events.
3. Click the **Weekly Schedule**, **Special Events**, or **Properties** tabs to update events and click **Save**.

Parent topic: Weekly schedules

Weekly schedules links

Schedule component links provide scheduling control over other components. Using a weekly schedule and a trigger schedule, you can link the weekly schedule's Out slot as a source to a slot on another component. You can link the same weekly schedule to many target components.

You link the output (out slot) of a weekly schedule to one or more writable control points with a like `Status<type>` input. For example, linking the output of a **BooleanSchedule** to a BooleanWritable point. More specifically, you could link the Out slot of weekly schedule types as follows:

- **BooleanSchedule** to a BooleanWritable that is a proxy point for a Binary Output object in a BACnet device.
- **NumericSchedule** to a NumericWritable that is a proxy point for setpoint NVI in a LON device.

By convention, when linking to a target writable point (with 16-level priority array), you select In16 among its different priority array inputs. However, you are free to select any available input level.

In a few cases, you may wish to "chain" weekly schedules from Out slot to In slot. This technique is typically useful only if one of the chained schedules is effective during any period. In this case, the Default Output value of all schedule components (except last in chain) must be null.

You can chain weekly schedules from Out slot to In slot. This technique is useful only if one of the chained schedules is effective during any period. In this case, the Default Output value of all schedule components (except last in chain) must be null.

Note: Typically, you do not link a CalendarSchedule component. Instead, you reference one or more calendar schedules from a weekly schedule in its special events setup. This allows global editing of event definitions.

- **Linking a weekly schedule using the Wire Sheet**
  Linking weekly schedules to other components provides scheduling control. You can link a single schedule to many target components.
- **Linking a weekly schedule using the Nav tree**
  Often, a weekly schedule is in a different container than the target component. In this case, the easiest way to link is using the Nav tree.

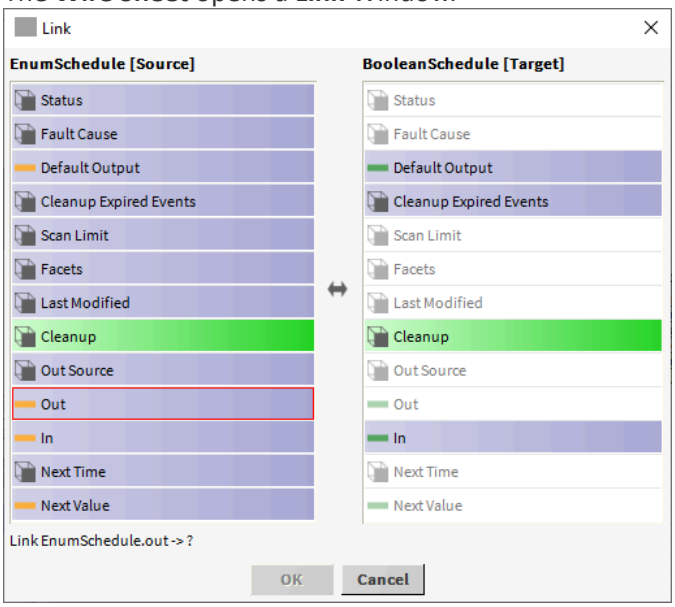
Parent topic: Weekly schedules

Linking a weekly schedule using the Wire Sheet

Linking weekly schedules to other components provides scheduling control. You can link a single schedule to many target components.

You have a configured weekly schedule component and writable target component in a Wire Sheet view.

1. In the **Wire Sheet**, mouse over the weekly schedule's Out slot until highlighted.
2. Drag the link wire from the weekly schedule to the bottom of the target component and release the mouse button.
   The **Wire Sheet** opens a **Link** Window.



   The source (left) side shows the schedule's Out slot preselected.

3. In the target (right) side of the **Link** window, click the desired slot of the target component.
   By convention, if linking to a priority array input of a writable point, In16 is used for schedule level control. However, you can select any available level desired.
4. With both source and target sides selected, click **OK**.

The target point or component is now linked to the weekly schedule.
Parent topic: Weekly schedules links

Linking a weekly schedule using the Nav tree

Often, a weekly schedule is in a different container than the target component. In this case, the easiest way to link is using the Nav tree.

You have a configured weekly schedule and writable target component in the station.

1. Expand the Nav tree to locate the source weekly schedule, noting its name.
2. Right-click the weekly schedule and select Link Mark.
3. Expand the Nav tree to locate the target component.
4. Right-click the target component and select Link From <weeklyscheduleName> from the popup menu. A **Link** window opens.
5. In the source (left) side, click the schedule's Out slot.
6. In the target (right) side, click the desired slot of the target component.
   By convention, if linking to a priority array input of a writable point, In16 is used for schedule level control. However, you can select any available level desired.
7. With both source and target sides selected in the **Link** dialog, click **OK**.

The target point or component is now linked to the weekly schedule.
Parent topic: Weekly schedules links

Schedule exports and imports (master/slave)

Using the driver architecture, you can create master/slave schedules to share schedule configuration among devices. This allows you to globally update the configuration of any slave schedule by making changes to its master schedule.

The typical application for export and import is a multi-station **NiagaraNetwork**, where you import a schedule component from another station. Typically, you import a schedule component that resides in a Supervisor station. This creates a local copy that you can use and link into control logic, but cannot otherwise configure (change events, and so on).

When you import the schedule, the scheduler automatically creates a schedule export descriptor in the sending (master) station under the **Station** component that represents the receiving side. This allows for sending-side management of configuration synchronization.

If you are using the BACnet driver, the same basic architecture is available. You can import BACnet Schedule and Calendar objects from a BACnet device, and model them as schedule components.

Note: The BACnet driver lets you export schedules from the station to existing BACnet Schedule and Calendar objects in a BACnet device, acting as the master source. Also, you can expose schedule components as BACnet Schedule or Calendar objects for access by any networked BACnet device. You do this through configuration of the **Export Table** under the BacnetNetwork's **Local Device.**

- **Importing a Schedule or calendar**
  If the station is part of a multi-station **NiagaraNetwork**, you can import a schedule from a Supervisor station. The schedule in the local station functions as the slave schedule and the schedule in the Supervisor station as the master schedule. This procedure uses views of the **Schedules** extension under a **NiagaraStation** (device-level) component. It works the same for the schedules and calendars of the BACnet driver.
- **Exporting a weekly schedule**
  The events table provides several export options.

Parent topic: Weekly schedules

Importing a Schedule or calendar

If the station is part of a multi-station **NiagaraNetwork**, you can import a schedule from a Supervisor station. The schedule in the local station functions as the slave schedule and the schedule in the Supervisor station as the master schedule. This procedure uses views of the **Schedules** extension under a **NiagaraStation** (device-level) component. It works the same for the schedules and calendars of the BACnet driver.

You are connected to the local (remote) station. The Supervisor has a schedule to import.

1. In the Nav tree, expand **Config > Drivers**. expand the network component (such as **NiagaraNetwork** or **BacnetNetwork**, of the station to receive the imported schedule.
2. Expand the device that contains the source schedule (**NiagaraStation** or **BacnetDevice**).
   In the Nav tree, the device's children include: **Points**, **Histories**, **Alarms**, and **Schedules**.
3. Double-click **Schedules**.
   The **Schedule Import Manager** opens. This manager lists any existing schedules already imported from this device.
4. Click **Discover**.
   The manager splits into two panes and starts a schedule discovery job. When complete, the **Discovered** pane (top) lists the schedules that are available for import.
5. In the top pane, click to select one or more schedule(s), then click **Add**.
   An **Add** opens.
6. After editing name and properties to suit, click **OK**.
   The job creates schedule components for the selected items. They reside under the device's schedules extension.

Note: You can link in to station logic as you can with any other schedule, but you cannot configure imported schedules (add, delete, or change events and other properties).

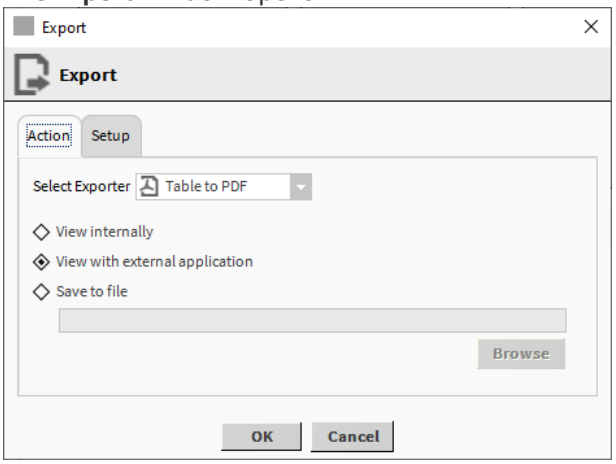
Parent topic: Schedule exports and imports (master/slave)

Exporting a weekly schedule

The events table provides several export options.

The **Scheduler** view is open.

1. Click the **Special Events** tab.
2. To export an event, click the selection icon at the upper-right corner of the events table and click **Export**.
   The **Export** window opens.



3. Select the file format (Select Exporter).

Supported formats include: oBix, PDF, CSV, text and HTML.
4. Select to view the exported data internally in Workbench or with an external application or to save the data to your file system.
5. If you selected PDF, set up the page size and orientation.
It supports in-line editor, highlight the time-slot and make changes at the bottom.

Parent topic: Schedule exports and imports (master/slave)

Other schedules and schedule selector

Along with weekly schedules, other components in the schedule palette add calender features.

These components define holidays, fire actions and link schedules for greater flexibility.

- Calendar schedules define holidays and other special days as scheduling exceptions. You can add as many day events as needed in the same **CalendarSchedule**. Once you configure a calendar schedule, you reference it using the special events setup of a weekly schedule. The **Calendar Scheduler** view manages these schedules.
- Trigger schedules fire actions or topics to control linked components or their child extensions. For example, you might link a trigger schedule to the ResetChangeOfStateCount action of a DiscreteTotalizerExt child of a BooleanPoint.
- Schedule selector components select a schedule to control a component. They work together with trigger schedules to link their output to an action on a control point or extension.

About calendar schedules

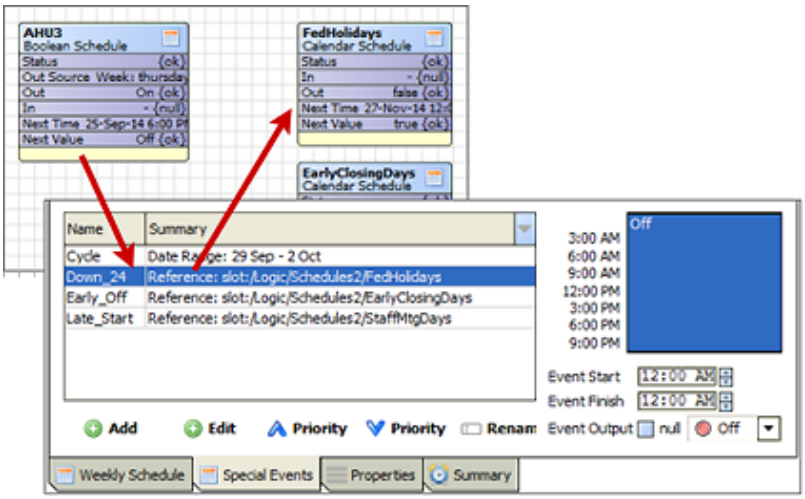
Calendar schedules define specific events. You use them to define days with scheduling exceptions, for example, holidays and other special days. You reference calendar schedules in the special events setup of weekly schedules.

Calendar schedules specify entire days, using the following types of day event selections: You can add as many day events as needed in the same **CalendarSchedule**.

- Date
- Date Range
- Week and Day
- Custom

The method for linking a **CalendarSchedule** is by referencing it from the special events configuration of one or more weekly schedules. Each referenced **CalendarSchedule** defines the day portion of a special event. Then, you configure time-of-day events in each special event, as needed.

Figure 1. Example referenced CalendarSchedules



For example, the figure above shows a **BooleanSchedule** and a portion of its special events tab, listing four special events. Three of these are reference events. Remote calendar schedules referenced by this schedule define this schedule's calendar day(s). Although the figure shows all components in the same container, quite often calendar schedules are located elsewhere in the station.

CalendarSchedule usage by special event reference allows global changes of day definitions, where multiple weekly schedules can reference one or more calendar schedules. Any edit of a **CalendarSchedule** affects all weekly schedules with a special event that references it.

- **Setting up the same event every month or year**
  Using the Calendar Scheduler you can configure the same event days every month or year.
- **Deleting a calendar event**
  Calendar events include special events and exception days configured on a calendar schedule.

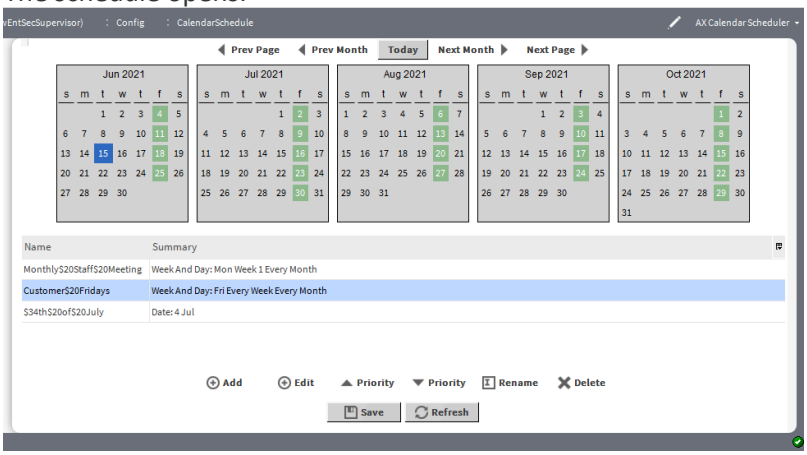
Parent topic: Other schedules and schedule selector

Setting up the same event every month or year

Using the Calendar Scheduler you can configure the same event days every month or year.

A weekly schedule exists in the station.

1. Locate the **CalendarSchedule** component in the station and double-click it.
   The schedule opens.



When you first access this scheduler, the current day is highlighted in the left-most calendar month at the top of the view.

2. To view previous and next months, click on **Next Month** and **Prev Month**, or **Next Page** and **Prev Page**.
3. To return to the current calendar month and day, click the **Today** button above the calendars.
4. Do one of the following:
   - To add an event, click **Add**.
   - To edit an event, select it and click **Edit**.

Note: You can also right-click anywhere on the events table to display an event menu from which to select **Add** or **Edit**.

When you select a calendar event, its day(s) of occurrence are highlighted in green in the monthly calendars at the top of the view.

If you added an event, the **Add** window opens. If you are editing an existing event, the **Edit** window opens.

5. If you are adding an event, give it a unique Name.
   For example, Thanksgiving_Break or Cleaning_2" This value defaults to `Event`. You can change its name later.
6. If you are adding an event, select its date Type to configure.
   These event types are available:

   - Date provides various combinations of weekday, numerical date, month, month combinations and year.
   - Date Range defines the event by start and end dates using for each a combination of day, month and year.
   - Week and Day provide a combination of day of the week, week in month and month.
   - Custom provides various combinations of day, month, weekdays and year.

7. To edit an event, select its row in the table and double-click it.
8. To perform an action using an available control, right-click the event.
   This opens the action menu.

Note: Priority selections via right-click menu or control buttons affect only the list order of events displayed in a **CalendarSchedule**. True priority applies only to special events in weekly schedules.

Parent topic: About calendar schedules

Deleting a calendar event

Calendar events include special events and exception days configured on a calendar schedule.

The calendar event exists.

1. Locate and double-click the CalendarSchedule in the station.
2. In the calendar events table, select an event to delete
3. Do one of the following:
   - Below the calendar events table, click **Delete**.
   - Right-click on an event in the calendar events table and select Delete from the drop-down menu.

   A **Delete** window opens.

4. To confirm, click **Yes**.
5. To save the updated schedule, click **Save**.
   Your changes are saved to the schedule.

Parent topic: About calendar schedules

About trigger schedules

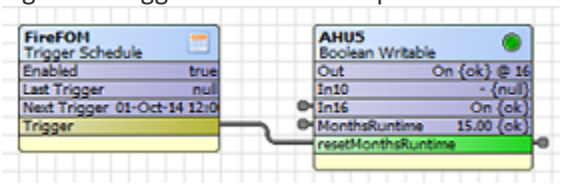
Trigger schedules are special-purpose schedules that control either the linked actions or topics of other components. A trigger schedule may link to an action of a control point or, more commonly, to an action of a point extension.

Often, you link a trigger schedule to a point extension, not directly to a parent container point or component. In this case, before linking you could first composite the target container component, selecting to expose the extension's action up in the parent point. Although optional, this can help reusability and link clarity.

Target actions or topics, unlike target property slots, can accept multiple link sources. In some cases, you may wish to link the Trigger Missed slot of the trigger schedule to the same target.

For example, you could link a **TriggerSchedule** to the ResetElapsedActiveTime action of a DiscreteTotalizerExt, a point extension for a Boolean point used to accumulate runtime. If the trigger schedule is configured to fire only on the first day of every month (at 12:00 am) that extension could be used to hold the current month's runtime.

Figure 1. Trigger schedule example



The above figure shows a simple example of a **TriggerSchedule** linked to an action slot of a DiscreteTotalizerExt, which was composite in the parent Boolean writable. This schedule is configured to fire once at midnight on the first day of every month. The trigger at the ResetElapsedActiveTime slot zeroes the runtime accumulated from the previous month. You can add individual trigger times to trigger schedule. Trigger times populate all triggered events once added. Use **Ctrl** + click to select the triggered events.

- **Adding a trigger event**
  You add one or more trigger events to a trigger schedule using controls located below the event table in the calendar side of the **Trigger Scheduler** view.
- **Adding trigger event times**
  Add one or more trigger event times to a trigger schedule using controls in the Time Picker side of the **Trigger Scheduler** view.
- **Linking a trigger schedule using the wire sheet**
  This procedure links a trigger schedule to a component in the same wire sheet.
- **Linking a trigger schedule using the Nav tree**
  Often a trigger schedule is in a different container from the target component container. In this case, the easiest way to link is using the Nav tree.

Parent topic: Other schedules and schedule selector

Adding a trigger event

You add one or more trigger events to a trigger schedule using controls located below the event table in the calendar side of the **Trigger Scheduler** view.

A schedule exists. The Trigger Scheduler view is open.

1. Below the trigger events table, click **Add** or right-click anywhere on the events table to open a pop-up menu and click **Add**.
   The **Add** window opens.
2. In the Name property, type a descriptive Name for the trigger event.
   For example, FirstDOM (first day of the month) or Each_WorkHr. You can change this name later, if needed.
3. Select the date Type, define the selection criteria and click **OK**.
   Type determines selection criteria provide these choices:

   - Date provides various combinations of weekday, numerical date, month, month combinations and year.
   - Date Range defines the event by start and end dates using for each a combination of day, month and year.
   - Week and Day provide a combination of day of the week, week in month and month.
   - Custom provides various combinations of day, month, weekdays and year.

   The framework adds the event to this calendar and the event row remains selected for further editing, except for Type.
4. In the right-side time picker area, add one or more triggers.
   By default, a midnight trigger may exist (`00h:00m`); you can delete it if needed. Using the Range option, you can add multiple triggers at some repeating interval. Trigger times apply to all trigger events (calendar-side entries).
5. Continue to add, edit, rename, or delete trigger events as needed and click **Save** when you are finished.

Parent topic: About trigger schedules

Adding trigger event times

Add one or more trigger event times to a trigger schedule using controls in the Time Picker side of the **Trigger Scheduler** view.

Your schedule exists with trigger events. The Trigger Scheduler view is open.

1. In the time picker (right) side of the **TriggerScheduler** view, set the desired time in the hour: minute editor, either by clicking up/down controls or typing the time.
   By default, a single midnight trigger time already exists (you can delete it if desired).
2. To enter a time range instead of a specific time, click the Range check box and set the desired Range End and Range Interval times.
   When entering a trigger range, the hour: minute editor acts as the Range Begin trigger time. By default, the Range Interval is set to one hour (`+00001h 00m 00.000s`). You can set this to whatever interval is needed.
3. Click **Add**.
   The new trigger event time(s) is added to the list in the time-picker pane.
4. To select multiple trigger times, hold down the `Ctrl` or `Shift` key while you select.
5. To delete a trigger time, select the time, then click the **Remove** button.
6. Click **Save**.
   Your changes are saved to the schedule.

Parent topic: About trigger schedules

Linking a trigger schedule using the wire sheet

This procedure links a trigger schedule to a component in the same wire sheet.

You have a configured trigger schedule and a target component (point or extension).

1. In the **Wire Sheet**, mouse over the trigger schedule's Trigger slot until highlighted.
2. Click and drag the link wire from the trigger schedule to the bottom of the target component, and release the mouse button.
   A **Link** window opens. The source (left) side has the trigger schedule's Trigger slot preselected.
3. In the target (right) side of the **Link** window, click on the slot of the target component.
4. With both source and target sides selected in the **Link** window, click **OK**.

The target component (point or extension) is now linked to the trigger schedule.
Parent topic: About trigger schedules

Linking a trigger schedule using the Nav tree

Often a trigger schedule is in a different container from the target component container. In this case, the easiest way to link is using the Nav tree.

You have a trigger schedule

1. Expand the Nav tree to locate the source trigger schedule, noting its name.
2. Right-click the trigger schedule and select Link Mark from the popup menu.
3. Expand the Nav tree to locate the target component.
   (In some cases, the target may be an extension of a point or other component.)
4. Right-click the target component and select Link From "<target schedule name>" from the popup menu.
   A **Link** dialog appears.
5. In the source (left) side of the **Link** dialog, click the trigger schedule's Trigger slot.
6. In the target (right) side of the **Link** dialog, click the desired action of the target component.
7. With both source and target sides selected in the **Link** dialog, click **OK**.

The target component (point or extension) is now linked to the trigger schedule.
Parent topic: About trigger schedules

About schedule selector components

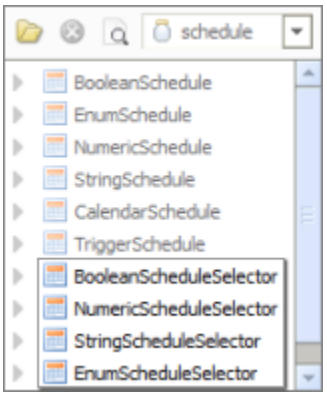
These componentsselect a pre-configured schedule for controlling a component.

For example, you can use a ScheduleSelector component to link a pre-configured schedule to a component that controls equipment, a door schedule, or setpoint temperature. With a set of pre-configured schedules and the **ScheduleSelector** component, you can set up or change the schedule of a device by choosing from a list of valid schedules. You do not have to configure schedule properties.

Each **ScheduleSelector** component contains a property to target a single location or container that holds all the schedule components to offer as options for the **ScheduleSelector** component. After you define the container location, the component populates the Schedule property option list with all valid schedule components. You can choose any of the options or leave the property set to the default null value. When you select a schedule, the ScheduleSelector component creates a link between the selected schedule and the component that is linked to the output of the **ScheduleSelector**.

There are four types of **ScheduleSelector** components available in the schedule palette, representing each of the four data types (Boolean, Enum, etc.).

Figure 1. ScheduleSelector components in schedule palette

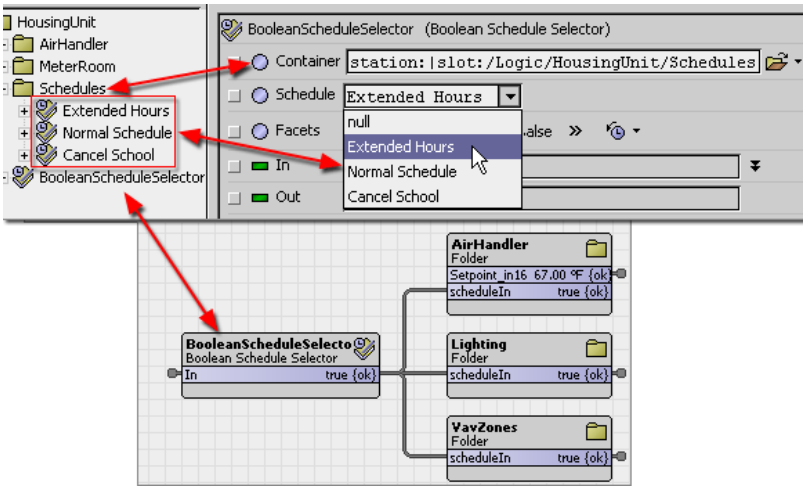


ScheduleSelector provide links between schedules and control components that are of the same data type. For example, a **BooleanScheduleSelector** component only links a **BooleanSchedule** component to a control component of the Boolean data type. An **EnumScheduleSelector** links between an **EnumSchedule** component and a control component of the enum data type. The same relationship applies to numeric and string schedule selectors.

If you have schedules of different data types in the same container, the **ScheduleSelector** only displays valid schedule components in the Schedule property options list.

BooleanScheduleSelector component

Figure 2. Property Sheet and Wire Sheet Views of a BooleanScheduleConnector Configuration

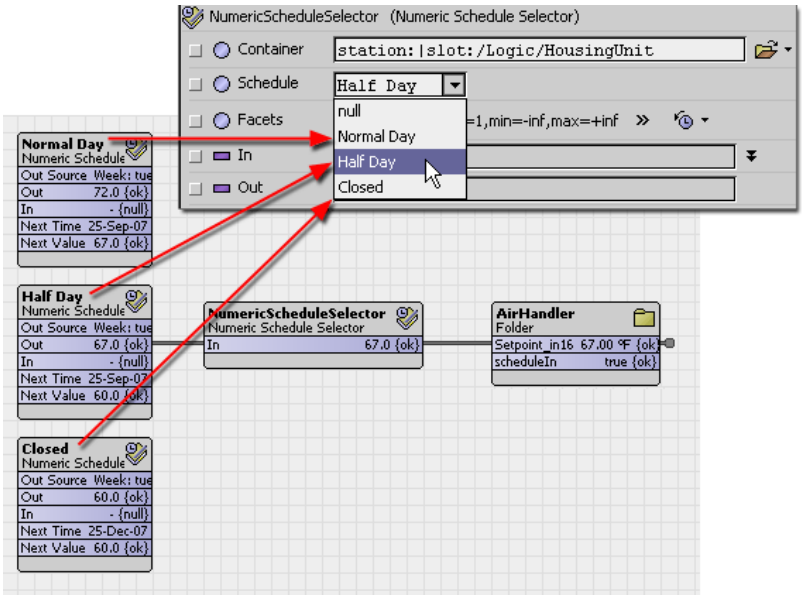


Notice the following points about this configuration:

- The **BooleanScheduleSelector** Container property links to the **Schedules** container (see Nav tree).
- The three **BooleanSchedule** components in the container are available as options in the Schedule property option list.
- On the **Wire Sheet** view, the **BooleanScheduleSelector** shows a knob link from the selected schedule to the schedule selector In.
- The **BooleanScheduleSelector** Out links to three device inputs.

NumericScheduleSelector component

The following image shows an example of a **NumericScheduleSelector**:
Figure 3. Property Sheet and Wire Sheet Views of a NumericScheduleSelector Configuration



Notice the following points about this configuration:

- The Container property is set to the station:|slot:/Logic/HousingUnit ORD. This is the container that holds the three numeric schedules.
- The Schedule property option list contains three numeric schedules available for selection. The schedules are the same ones shown on the **Wire Sheet** view of the folder component labeled HousingUnit.
- The link (visible in the **Wire Sheet** view) connects from the numeric schedule Out to the schedule selector In.
- The linked value (67.0 in this example) passes from the NumericScheduleSelector component Out to the AirHandler device Setpoint input.

Parent topic: Other schedules and schedule selector

Schedule components

Components include services, folders and other model building blocks associated with a module. You drag them to a property or wire sheet from a palette. Views are plugins that can be accessed by double-clicking a component in the Nav tree or right-clicking a component and selecting its view from the **Views** menu.

The component and view topics that follow appear as context-sensitive help topics when accessed by:

- Right-clicking on the object and selecting **Views > Guide Help**
- Clicking **Help > Guide On Target**

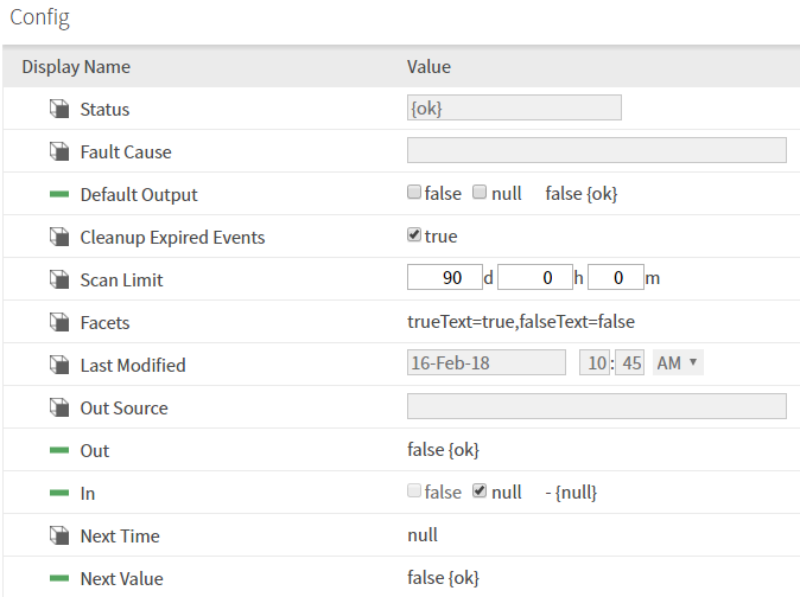
- **schedule-BooleanSchedule**
  This component is a deployable weekly schedule that provides a continuous StatusBoolean output. The **BooleanSchedule** is available in the schedule palette.
- **schedule-EnumSchedule**
  A deployable weekly schedule that provides a continuous StatusEnum output. Other weekly schedule types include **BooleanSchedule**, **NumericSchedule**, and **StringSchedule**. **EnumSchedule** is available in the **schedule** palette.
- **schedule-NumericSchedule**
  This component is a deployable weekly schedule that provides a continuous StatusNumeric output. **NumericSchedule** is available in the schedule palette.
- **schedule-StringSchedule**
  This component is a deployable weekly schedule that provides a continuous StatusString output. **StringSchedule** is available in the **schedule** palette.
- **schedule-CalendarSchedule**
  This component provides a calendar for scheduling holidays or other schedule overrides. CalendarSchedule is available in the **schedule** palette.
- **schedule-TriggerSchedule**
  TriggerSchedules are special-purpose schedules, providing scheduling control for either linked actions or topics of other components. TriggerSchedule fires actions or topics, there is no continuous output. TriggerSchedule is available in the **schedule** palette.
- **schedule-BooleanScheduleSelector**
  These components provide an easy way to select a pre-configured schedule of the proper data type for controlling a particular component. **BooleanScheduleSelector** components only link **BooleanSchedule** components to a control component of the Boolean data type. **BooleanScheduleSelector** is available in the schedule palette.
- **schedule-NumericScheduleSelector**
  ScheduleSelector components provide an easy way to select a preconfigured schedule of the proper data type for controlling a component. **NumericScheduleSelector** components only link **NumericSchedule** components to a control component of the numeric data type. **NumericScheduleSelector** is available in the schedule palette.
- **schedule-StringScheduleSelector**
  ScheduleSelector components provide an easy way to select a preconfigured schedule of the proper data type for controlling a particular component. **StringScheduleSelector** components only link StringSchedule components to a control component of the String data type. **StringScheduleSelector** is available in the **schedule** palette.
- **schedule-EnumScheduleSelector**
  ScheduleSelector components provide an easy way for to select a preconfigured schedule of the proper data type for controlling a component. **EnumScheduleSelector** components only link EnumSchedule components to a control component of the Enum data type. **EnumScheduleSelector** is available in the schedule palette.

schedule-BooleanSchedule

This component is a deployable weekly schedule that provides a continuous StatusBoolean output. The

BooleanSchedule is available in the schedule palette.

BooleanSchedule is the most commonly used weekly schedule component type. You use it for schedule control of Boolean writable points (typically proxy points). If needed, you can link it to to slots in extensions. For example, using an intermediate kitControl object (say, a Not logic-type object) use a **BooleanSchedule** to link to the Enabled slot of an alarm extension.



To access these properties, right-click the **BooleanSchedule** in the Nav tree and click **Views > AX Property Sheet**.

In addition to the standard properties (Status and Fault Cause), these properties support Boolean schedules.

Property	Value	Description
Default output	true or false	Sets up default output for the component. In scheduling, the schedule's output value whenever an event (either weekly or special event) is not active. It is also used whenever the component is not effective. For a weekly schedule, set Default Output on the **Properties** tab in the **Scheduler** view. Available values must be pre-defined in the component'sFacets property.
Cleanup Expired Events	true (default) or false	Automatically deletes (true) calendar events of type Date and Date Range that have already occurred and cannot occur again (as configured). This is recorded in the schedule log, and you no longer see them in the view. false retains all calendar events even if they cannot occur again (as configured).
Scan Limit	`90 days` (default)	Specifies a limit on how far ahead the component searches to find the next event output change. This can prevent excessive

Property	Value	Description
		CPU usage. Format is: ddd hh mm ss, and range is from 1 day (minimum) up. If changed, a value less than the default is typically recommended--for example, 14 days.
Facets	trueText (default), falseText	Configure true and false text for schedules. Configuring facets is critical in an EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule. Facets contain additional data applied to input and output values. • trueText is the text to display when output is true • falseText is the text to display when output is false. For example, you might want to set the facet trueText to display "ON" and the facet falseText to display "OFF." "Units of measurement" is also a type of facet. View facets on the **Slot Sheet** and edit them from a component Property sheet by clicking the **>>** icon to display the **Config Facets** window.
Last Modified	read-only	Reports the timestamp of last configuration change.
Out Source	read-only	Provides a string source description of the current output, as one of the four options shown left. For example, Week: monday or Special Event: Christmas Break
Out	false {ok} (default)	Reports the output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false Determined by the following, in highest-to-lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value.

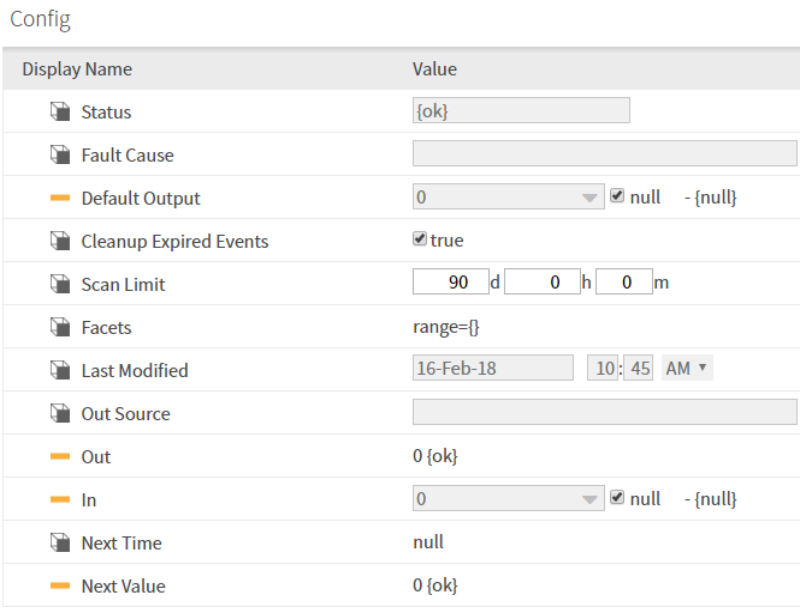
Property	Value	Description
In	(defaults to `null`)	Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Next Time	(defaults to `null`)	Displays the time of next scheduled output change for the component. If more than a year away, this value is null. Displays in Baja AbsTime format, for example: `03-Feb-05 5:00 PM`. Typical application is for informational display. If needed. slots can be linked into control logic. For example TimeDifference and CurrentTime objects (kitControl, Timer) provide AbsTime slots too.
Next Value	read-only	Reports the next scheduled output value, at Next Time. Value is meaningless if Next Time is null.

Parent topic: Schedule components

schedule-EnumSchedule

A deployable weekly schedule that provides a continuous StatusEnum output. Other weekly schedule types include **BooleanSchedule**, **NumericSchedule**, and **StringSchedule**. **EnumSchedule** is available in the **schedule** palette.

Allows schedule control of enum writable points (typically proxy points). For example, link it to an enum writable that proxies a BACnet Multistate Output object, or to an enum writable that proxies a LON NVI (using an enumerated SNVT).



To access these properties, right-click the **EnumSchedule** in the Nav tree and click **Views > AX Property Sheet**.

In addition to the standard properties (Status and Fault Cause), these properties support enum schedules.

Property	Value	Description
Default output	true or false	Sets up default output for the component. In scheduling, the schedule's output value whenever an event (either weekly or special event) is not active. It is also used whenever the component is not effective. For a weekly schedule, set Default Output on the **Properties** tab in the **Scheduler** view. Available values must be pre-defined in the component'sFacets property.
Cleanup Expired Events	true (default) or false	Automatically deletes (true) calendar events of type Date and Date Range that have already occurred and cannot occur again (as configured). This is recorded in the schedule log, and you no longer see them in the view. false retains all calendar events even if they cannot occur again (as configured).
Scan Limit	`90 days` (default)	Specifies a limit on how far ahead the component searches to find the next event output change. This can prevent excessive CPU usage. Format is: ddd hh mm ss, and range is from 1 day (minimum) up. If changed, a value less than the default is typically recommended--for example, 14 days.
Facets	trueText (default), falseText	Configure true and false text for schedules. Configuring facets is critical in an EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule. Facets contain additional data applied to input and output values. • trueText is the text to display when output is true • falseText is the text to display when output is false. For example, you might want to set the facet trueText to display "ON" and the facet falseText to display "OFF." "Units of measurement" is also a type of facet. View facets on the **Slot Sheet** and edit them from a component Property sheet by clicking the **>>** icon to display the **Config Facets** window.
Last Modified	read-only	Reports the timestamp of last configuration change.

Property	Value	Description
Out Source	read-only	Provides a string source description of the current output, as one of the four options shown left. For example, Week: monday or Special Event: Christmas Break
Out	false {ok} (default)	Reports the output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false Determined by the following, in highest-to-lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value.
In	(defaults to `null`)	Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Next Time	(defaults to `null`)	Displays the time of next scheduled output change for the component. If more than a year away, this value is null. Displays in Baja AbsTime format, for example: `03-Feb-05 5:00 PM`. Typical application is for informational display. If needed. slots can be linked into control logic. For example TimeDifference and CurrentTime objects (kitControl, Timer) provide AbsTime slots too.
Next Value	read-only	Reports the next scheduled output value, at Next Time. Value is meaningless if Next Time is null.

Parent topic: Schedule components

schedule-NumericSchedule

This component is a deployable weekly schedule that provides a continuous StatusNumeric output. **NumericSchedule** is available in the schedule palette.

Allows schedule control of numeric writable points (typically proxy points), which may represent setpoints, limits, or any number of other variables.

Config

Display Name	Value
Status	{ok}
Fault Cause	
Default Output	0.00 null - {null}
Cleanup Expired Events	true
Scan Limit	90 d 0 h 0 m
Facets	units=null,precision=1,min=-inf,max=+inf
Last Modified	16-Feb-18 10 : 45 AM
Out Source	
Out	- {null}
In	0.00 null - {null}
Next Time	null
Next Value	0 {ok}

To access these properties, right-click the **NumericSchedule** in the Nav tree and click **Views > AX Property Sheet**.

In addition to the standard properties (Status and Fault Cause), these properties support numeric schedules.

Property	Value	Description
Default output	true or false	Sets up default output for the component. In scheduling, the schedule's output value whenever an event (either weekly or special event) is not active. It is also used whenever the component is not effective. For a weekly schedule, set Default Output on the **Properties** tab in the **Scheduler** view. Available values must be pre-defined in the component'sFacets property.
Cleanup Expired Events	true (default) or false	Automatically deletes (true) calendar events of type Date and Date Range that have already occurred and cannot occur again (as configured). This is recorded in the schedule log, and you no longer see them in the view. false retains all calendar events even if they cannot occur again (as configured).
Scan Limit	`90 days` (default)	Specifies a limit on how far ahead the component searches to find the next event output change. This can prevent excessive CPU usage. Format is: ddd hh mm ss, and range is from 1 day (minimum) up. If changed, a value less than the default is typically recommended--for example, 14 days.
Facets	trueText (default), falseText	Configure true and false text for schedules. Configuring facets is critical in an

Property	Value	Description
		EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule. Facets contain additional data applied to input and output values. • trueText is the text to display when output is true • falseText is the text to display when output is false. For example, you might want to set the facet trueText to display “ON” and the facet falseText to display “OFF.” "Units of measurement" is also a type of facet. View facets on the **Slot Sheet** and edit them from a component Property sheet by clicking the **>>** icon to display the **Config Facets** window.
Last Modified	read-only	Reports the timestamp of last configuration change.
Out Source	read-only	Provides a string source description of the current output, as one of the four options shown left. For example, Week: monday or Special Event: Christmas Break
Out	false {ok} (default)	Reports the output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false Determined by the following, in highest-to-lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value.
In	(defaults to `null`)	Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Next Time	(defaults to `null`)	Displays the time of next scheduled output change for the component. If more than a

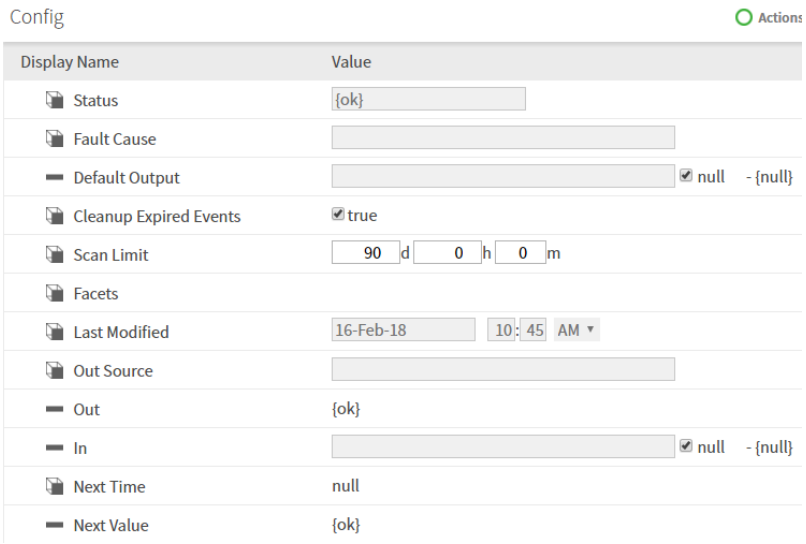
Property	Value	Description
		year away, this value is null. Displays in Baja AbsTime format, for example: `03-Feb-05 5:00 PM`. Typical application is for informational display. If needed. slots can be linked into control logic. For example TimeDifference and CurrentTime objects (kitControl, Timer) provide AbsTime slots too.
Next Value	read-only	Reports the next scheduled output value, at Next Time. Value is meaningless if Next Time is null.

Parent topic: Schedule components

schedule-StringSchedule

This component is a deployable weekly schedule that provides a continuous StatusString output. **StringSchedule** is available in the **schedule** palette.

Allows schedule control of string writable points (possibly proxy points).



To access these properties, right-click the **StringSchedule** in the Nav tree and click **Views > AX Property Sheet**.

In addition to the standard properties (Status and Fault Cause), these properties support string schedules.

Property	Value	Description
Default output	true or false	Sets up default output for the component. In scheduling, the schedule's output value whenever an event (either weekly or special event) is not active. It is also used whenever the component is not effective. For a weekly schedule, set Default Output

Property	Value	Description
		on the **Properties** tab in the **Scheduler** view. Available values must be pre-defined in the component'sFacets property.
Cleanup Expired Events	true (default) or false	Automatically deletes (true) calendar events of type Date and Date Range that have already occurred and cannot occur again (as configured). This is recorded in the schedule log, and you no longer see them in the view. false retains all calendar events even if they cannot occur again (as configured).
Scan Limit	`90 days` (default)	Specifies a limit on how far ahead the component searches to find the next event output change. This can prevent excessive CPU usage. Format is: ddd hh mm ss, and range is from 1 day (minimum) up. If changed, a value less than the default is typically recommended--for example, 14 days.
Facets	trueText (default), falseText	Configure true and false text for schedules. Configuring facets is critical in an EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule. Facets contain additional data applied to input and output values. • trueText is the text to display when output is true • falseText is the text to display when output is false. For example, you might want to set the facet trueText to display "ON" and the facet falseText to display "OFF." "Units of measurement" is also a type of facet. View facets on the **Slot Sheet** and edit them from a component Property sheet by clicking the **>>** icon to display the **Config Facets** window.
Last Modified	read-only	Reports the timestamp of last configuration change.
Out Source	read-only	Provides a string source description of the current output, as one of the four options shown left. For example, Week: monday or Special Event: Christmas Break
Out	false {ok} (default)	Reports the output value of schedule component. Output is true during any

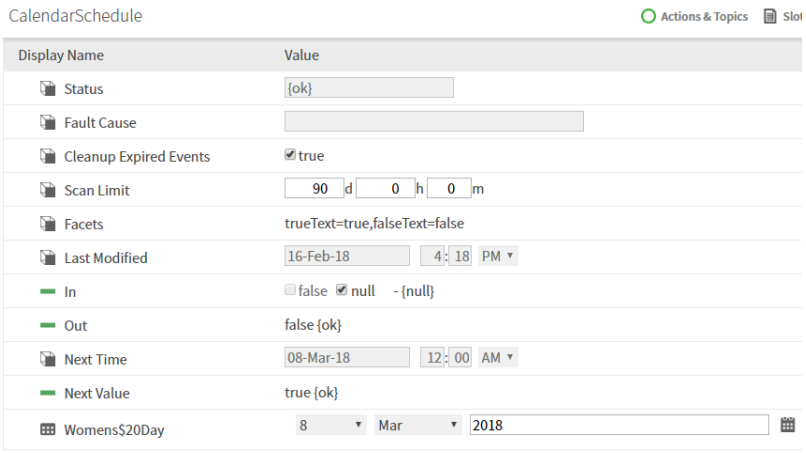
Property	Value	Description
		configured calendar day(s), otherwise it is false Determined by the following, in highest-to-lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value.
In	(defaults to `null`)	Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Next Time	(defaults to `null`)	Displays the time of next scheduled output change for the component. If more than a year away, this value is null. Displays in Baja AbsTime format, for example: `03-Feb-05 5:00 PM`. Typical application is for informational display. If needed. slots can be linked into control logic. For example TimeDifference and CurrentTime objects (kitControl, Timer) provide AbsTime slots too.
Next Value	read-only	Reports the next scheduled output value, at Next Time. Value is meaningless if Next Time is null.

Parent topic: Schedule components

schedule-CalendarSchedule

This component provides a calendar for scheduling holidays or other schedule overrides. CalendarSchedule is available in the **schedule** palette.

Figure 1. CalendarSchedule properties



To view these properties, right-click the CalendarSchedule component in the station and click **Views > AX Property Sheet**.

In addition to the standard properties (Status, and Fault Cause), these properties configure calendar schedules.

Property	Value	Description
Default output	true or false	Sets up default output for the component. In scheduling, the schedule's output value whenever an event (either weekly or special event) is not active. It is also used whenever the component is not effective. For a weekly schedule, set Default Output on the **Properties** tab in the **Scheduler** view. Available values must be pre-defined in the component'sFacets property.
Cleanup Expired Events	true (default) or false	Automatically deletes (true) calendar events of type Date and Date Range that have already occurred and cannot occur again (as configured). This is recorded in the schedule log, and you no longer see them in the view. false retains all calendar events even if they cannot occur again (as configured).
Scan Limit	`90 days` (default)	Specifies a limit on how far ahead the component searches to find the next event output change. This can prevent excessive CPU usage. Format is: ddd hh mm ss, and range is from 1 day (minimum) up. If changed, a value less than the default is typically recommended--for example, 14 days.
Facets	true Text (default), falseText	Configure true and false text for schedules. Configuring facets is critical in an EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule. Facets contain additional data applied to

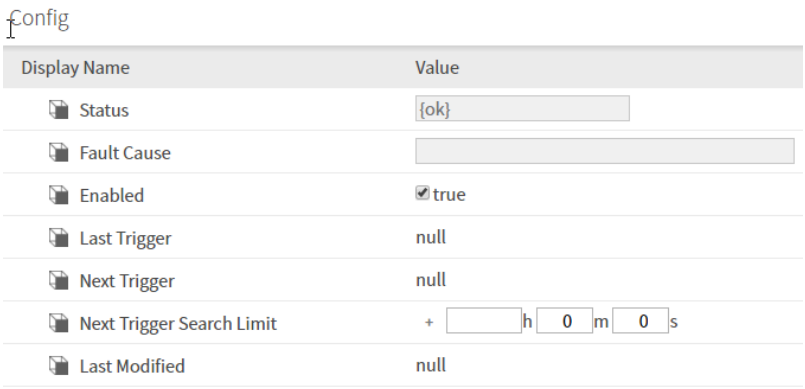
Property	Value	Description
		input and output values. • trueText is the text to display when output is true • falseText is the text to display when output is false. For example, you might want to set the facet trueText to display “ON” and the facet falseText to display “OFF.” "Units of measurement" is also a type of facet. View facets on the **Slot Sheet** and edit them from a component Property sheet by clicking the **>>** icon to display the **Config Facets** window.
Last Modified	read-only	Reports the timestamp of last configuration change.
Out Source	read-only	Provides a string source description of the current output, as one of the four options shown left. For example, Week: monday or Special Event: Christmas Break
Out	false {ok} (default)	Reports the output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false Determined by the following, in highest-to-lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value.
In	(defaults to `null`)	Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Next Time	(defaults to `null`)	Displays the time of next scheduled output change for the component. If more than a year away, this value is null. Displays in Baja AbsTime format, for example: `03-Feb-05 5:00 PM`. Typical application is for informational

Property	Value	Description
		display. If needed. slots can be linked into control logic. For example TimeDifference and CurrentTime objects (kitControl, Timer) provide AbsTime slots too.
Next Value	read-only	Reports the next scheduled output value, at Next Time. Value is meaningless if Next Time is null.

Parent topic: Schedule components

schedule-TriggerSchedule

TriggerSchedules are special-purpose schedules, providing scheduling control for either linked actions or topics of other components. TriggerSchedule fires actions or topics, there is no continuous output. TriggerSchedule is available in the **schedule** palette.



By default, for any TriggerSchedule copied from the schedule palette, the main Trigger output slot is pinned on the component's glyph (shape on the wire sheet), as well as the following additional slots:

Property	Value	Description
Enabled	true (default) or false	Enables/disables firing of trigger outputs.
Next Trigger	(defaults to `null`)	Reports next scheduled trigger firing time.
Last Trigger	read-only	Reports the timestamp of last configuration change.
Next Trigger Search Limit	`2160h 00m 00s` (default)	Limits how far into the future to search for the next trigger. This setting prevents an infinite search due to a poorly configured schedule. Default is 90 days (2160 hours). Format is:`hhhh mm ss`
Next Value	read-only	Reports the next scheduled output value, at Next Time. Value is meaningless if Next Time is null.
Last Modified	read-only	Reports the timestamp of last

Property	Value	Description
		configuration change.

Trigger Missed slot

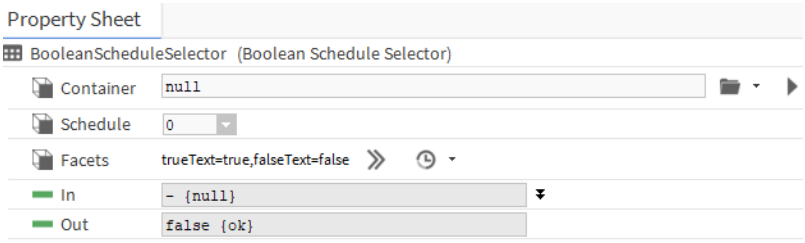
In addition to the main Trigger output slot, the TriggerSchedule has a Trigger Missed slot, also a topic type slot. If the station was not running when a scheduled trigger was to occur (appeared previously in the Next Trigger property), upon station startup the Trigger Missed slot fires once.

Note: Trigger Missed always fires only once, no matter many triggers may have been missed.

Parent topic: Schedule components

schedule-BooleanScheduleSelector

These components provide an easy way to select a pre-configured schedule of the proper data type for controlling a particular component. **BooleanScheduleSelector** components only link **BooleanSchedule** components to a control component of the Boolean data type. **BooleanScheduleSelector** is available in the schedule palette.



Properties

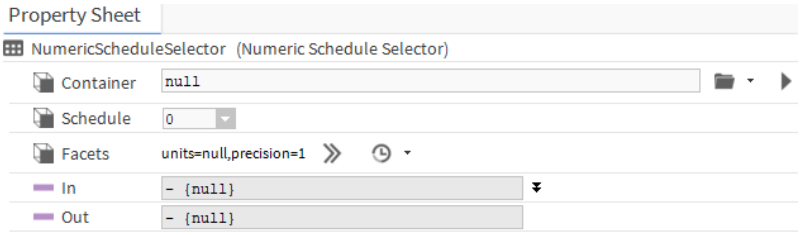
Property	Value	Description
Container	opens a chooser for selecting an ORD (defaults to `null`)	Provides a text field that uses an ORD to specify the location of the available schedules. Use the Component chooser at the right side of the field to browse to and select the desired schedule container.
Schedule	drop-down list (defaults to zero (0))	Displays an option list of all available schedule components of the appropriate data type located in the specified **Container** property. Selecting and saving a schedule using this property's option list automatically creates a link from the out slot of the selected schedule to the in slot of the **ScheduleSelector** component. Selecting null for this property causes the out property to generate a null value, and automatically removes any link from a schedule component to the in slot of the

Property	Value	Description
		ScheduleSelector component.
Facets	opens the **Config Facets** window for configuring trueText (default), falseText	Reports schedule facets. This property does not require configuration. The schedule selector automatically inherits Facets from the currently selected schedule.
In		Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Out	read-only and check-mark (defaults to the incoming value from the device)	Reports the output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false Determined by the following, in highest-to-lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value. When null is enabled, reports the incoming value from the device. You cannot change this value. To change this value, click the double-arrow to the right and remove the null check mark.

Parent topic: Schedule components

schedule-NumericScheduleSelector

ScheduleSelector components provide an easy way to select a preconfigured schedule of the proper data type for controlling a component. **NumericScheduleSelector** components only link **NumericSchedule** components to a control component of the numeric data type. **NumericScheduleSelector** is available in the schedule palette.



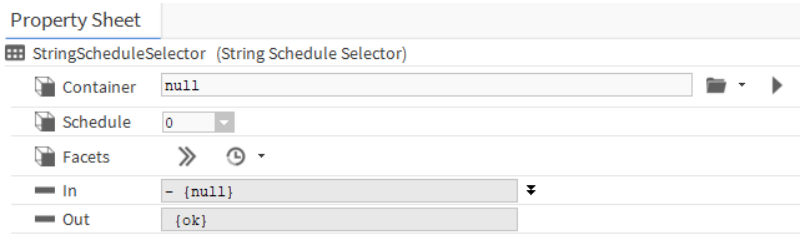
Property	Value	Description
Container	opens a chooser for selecting an ORD (defaults to `null`)	Provides a text field that uses an ORD to specify the location of the available schedules. Use the Component chooser at the right side of the field to browse to and select the desired schedule container.
Schedule	drop-down list (defaults to zero (0))	Displays an option list of all available schedule components of the appropriate data type located in the specified **Container** property. Selecting and saving a schedule using this property's option list automatically creates a link from the out slot of the selected schedule to the in slot of the **ScheduleSelector** component. Selecting null for this property causes the out property to generate a null value, and automatically removes any link from a schedule component to the in slot of the **ScheduleSelector** component.
Facets	opens the **Config Facets** window for configuring trueText (default), falseText	Configure true and false text for schedules. Configuring facets is critical in an EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule. Facets contain additional data applied to input and output values. • trueText is the text to display when output is true • falseText is the text to display when output is false. For example, you might want to set the facet trueText to display “ON” and the facet falseText to display “OFF.” "Units of measurement" is also a type of facet. View facets on the **Slot Sheet** and edit them from a component Property sheet by clicking the **>>** icon to display the **Config Facets** window.
In		Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Out	read-only and check-mark (defaults to the incoming value from the device)	Reports the output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false Determined by the following, in highest-to-

Property	Value	Description
		lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value. When null is enabled, reports the incoming value from the device. You cannot change this value. To change this value, click the double-arrow to the right and remove the null check mark.

Parent topic: Schedule components

schedule-StringScheduleSelector

ScheduleSelector components provide an easy way to select a preconfigured schedule of the proper data type for controlling a particular component. **StringScheduleSelector** components only link StringSchedule components to a control component of the String data type. **StringScheduleSelector** is available in the **schedule** palette.



Property	Value	Description
Container	opens a chooser for selecting an ORD (defaults to `null`)	Provides a text field that uses an ORD to specify the location of the available schedules. Use the Component chooser at the right side of the field to browse to and select the desired schedule container.
Schedule	drop-down list (defaults to zero (0))	Displays an option list of all available schedule components of the appropriate data type located in the specified **Container** property. Selecting and saving a schedule using this property's option list automatically creates a link from the out slot of the selected schedule to the in slot of the **ScheduleSelector** component.

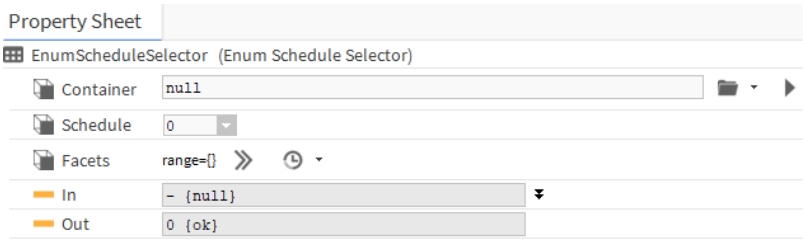
Property	Value	Description
		Selecting null for this property causes the out property to generate a null value, and automatically removes any link from a schedule component to the in slot of the **ScheduleSelector** component.
Facets	opens the **Config Facets** window for configuring trueText (default), falseText	Configure true and false text for schedules. Configuring facets is critical in an EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule. Facets contain additional data applied to input and output values. • trueText is the text to display when output is true • falseText is the text to display when output is false. For example, you might want to set the facet trueText to display "ON" and the facet falseText to display "OFF." "Units of measurement" is also a type of facet. View facets on the **Slot Sheet** and edit them from a component Property sheet by clicking the **>>** icon to display the **Config Facets** window.
In		Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Out	read-only and check-mark (defaults to the incoming value from the device)	Reports the output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false Determined by the following, in highest-to-lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value. When null is enabled, reports the incoming value from the device. You cannot change

Property	Value	Description
		this value. To change this value, click the double-arrow to the right and remove the null check mark.

Parent topic: Schedule components

schedule-EnumScheduleSelector

ScheduleSelector components provide an easy way for to select a preconfigured schedule of the proper data type for controlling a component. **EnumScheduleSelector** components only link EnumSchedule components to a control component of the Enum data type. **EnumScheduleSelector** is available in the schedule palette.



Property	Value	Description
Container	opens a chooser for selecting an ORD (defaults to `null`)	Provides a text field that uses an ORD to specify the location of the available schedules. Use the Component chooser at the right side of the field to browse to and select the desired schedule container.
Schedule	drop-down list (defaults to zero (0))	Displays an option list of all available schedule components of the appropriate data type located in the specified **Container** property. Selecting and saving a schedule using this property's option list automatically creates a link from the out slot of the selected schedule to the in slot of the **ScheduleSelector** component. Selecting null for this property causes the out property to generate a null value, and automatically removes any link from a schedule component to the in slot of the **ScheduleSelector** component.
Facets	opens the **Config Facets** window for configuring trueText (default), falseText	Configure true and false text for schedules. Configuring facets is critical in an EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule. Facets contain additional data applied to input and output values. • trueText is the text to display

Property	Value	Description
		when output is true • falseText is the text to display when output is false. For example, you might want to set the facet trueText to display “ON” and the facet falseText to display “OFF.” "Units of measurement" is also a type of facet. View facets on the **Slot Sheet** and edit them from a component Property sheet by clicking the **>>** icon to display the **Config Facets** window.
In		Provides a string source description of the current input, such as a linked schedule. If the in property is linked and its value is non-null, then this value overrides the scheduled output.
Out	read-only and check-mark (defaults to the incoming value from the device)	Reports the output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false Determined by the following, in highest-to-lowest priority: Any non-null value at its In slot (if linked). This value is immediately passed to its output. Otherwise (if null), processing continues. If the schedule is not effective, the output goes to the default output value. If the schedule is effective, the output goes to the (highest priority) active special event (if any). The active weekly schedule event (if any). The default output value. When null is enabled, reports the incoming value from the device. You cannot change this value. To change this value, click the double-arrow to the right and remove the null check mark.

Parent topic: Schedule components

Schedule plugins

This section provides information on the plugins in the schedule module.

In general, plugins provide views of components and can be accessed in many ways. For example, double-click a component in the Nav tree to see its default view. Also, you can right-click on a component and select from its **Views** menu.

For summary documentation on any view, select **Help > On View** (**F1**) from the Workbench menu or press **F1** while the view is open.

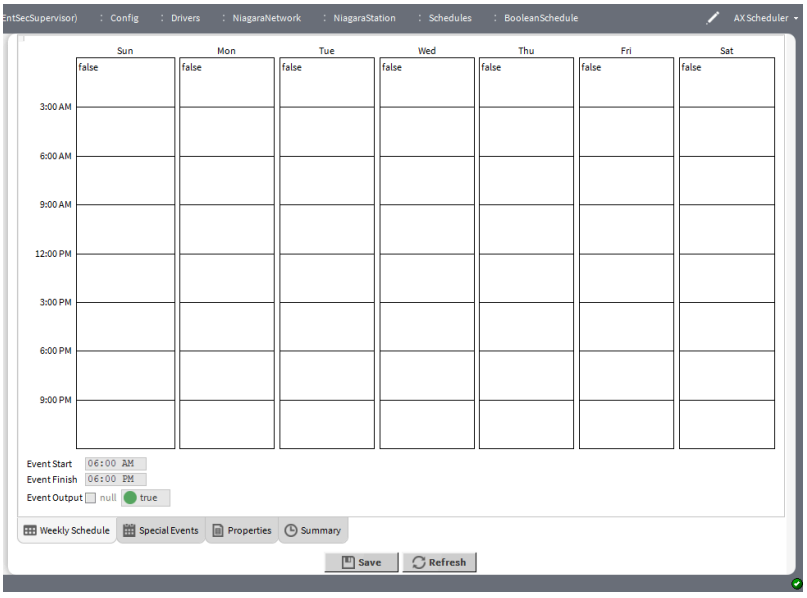
- **Scheduler**
  This view supports the four weekly schedule components: **BooleanSchedule**, **EnumSchedule**, **NumericSchedule**, and **StringSchedule**. The only difference among them is the output type.
- **Current Day Summary**
  This view is available only on weekly schedule (**BooleanSchedule**, **EnumSchedule**, **NumericSchedule**, and **StringSchedule**) components. It provides a simple graphical display of all schedule events for the current day, moving left-to-right from 0-to-24 hours.
- **Calendar Scheduler view**
  This is the default view for CalendarSchedule components. In this view you can configure event days per month. Typically, calendar events represent holidays.
- **TriggerScheduler**
  This is the default view of TriggerSchedule components. You use this view to configure schedule events in a TriggerSchedule. Trigger schedules are defined by a combination of calendar day (or days) and trigger event time(s). Trigger times apply to all trigger events (calendar-side entries).

Scheduler

This view supports the four weekly schedule components: **BooleanSchedule**, **EnumSchedule**, **NumericSchedule**, and **StringSchedule**. The only difference among them is the output type.

Buttons

Figure 1. Empty weekly schedule



The Scheduler defaults to the **Weekly Schedule** view. A separate topic documents this view.

All Scheduler views share the same buttons:

- **Save** downloads your changes to the schedule component's configuration. Immediately following, it activates the **Save** button again. This button is active only if you have unsaved changes in the component's scheduler.

- **Refresh** does one of two things:

  If the **Save** button is not available (no unsaved changes), clicking **Refresh** re-synchronizes the view with the component's current configuration.

  If the **Save** button is available (unsaved changes), clicking **Refresh** produces a confirmation window:

- **Weekly Schedule tab**
  This tab schedules regular events that repeat from week to week based on the day of the week and the time of day. By default, existing events appear as colored blocks, while unscheduled (default output) time appears in white.
- **Special Events tab**
  This tab configures exceptions to the normal weekly schedule as special events. These events override and intermingle with events in the regular weekly schedule. To manage these events you use the controls at the bottom of the tab to Add, Edit, Prioritize, Rename, and Delete.
- **Properties tab**
  This tab configures schedule properties.
- **Summary tab**
  This tab displays a read-only summary, which you can use to review a weekly schedule's configuration.

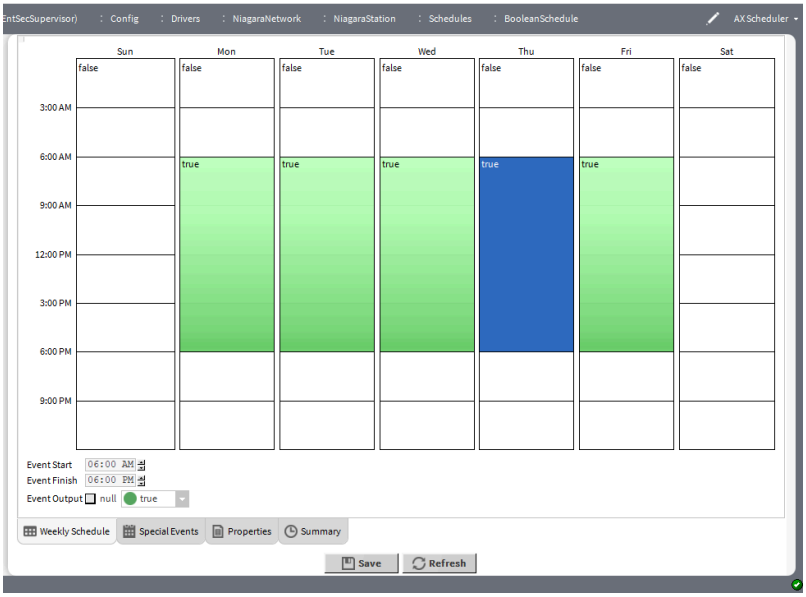
Parent topic: Schedule plugins

Weekly Schedule tab

This tab schedules regular events that repeat from week to week based on the day of the week and the time of day. By default, existing events appear as colored blocks, while unscheduled (default output) time appears in white.

Properties

Figure 1. Scheduler view for a weekly schedule



To view this schedule, double-click the schedule component and select the **Weekly Schedule** tab.

The view title at the top displays the name of the schedule.

The grid provides a visual representation of the schedule.

You use the tabs in this view to edit the schedule.

Note: The finish time for an event is the first time value after the event time range. The start time is inclusive.

Property	Value	Description
Event Start	time of day, AM and PM with up and down arrows	Fine tunes the start time for the event.
Event Finish	time of day, AM and PM with up and down arrows	Fine tunes the end time for the event.
Event Output	null and true (default) or false	Displays numbers or text in each schedule block. What to display depends on the type of schedule. If you intend nothing to display, click the null check box. An enum schedule displays a facet name. This is why it is important to configure facets before configuring an enum schedule. If a **NumericSchedule** or **StringSchedule**, you type the value in this property, then press **Enter** to register it in the event block.

Tabs

- **Weekly Schedule** manages Sunday through Saturday (weekly) event times and values. Use this to define regular weekly events.

- **Special Events** manages all exceptions to the defined weekly schedule as special events. Use this to define special event times.
- **Properties** configures important properties, such as default output, the schedule effective times, special event cleanup operation, and schedule facets.
- **Summary**, for any selected day, provides a tabular summary of all schedule events with source.

Right-click action menu

Right-click on a day for an event menu. Event menu options are straightforward, and may include the following:

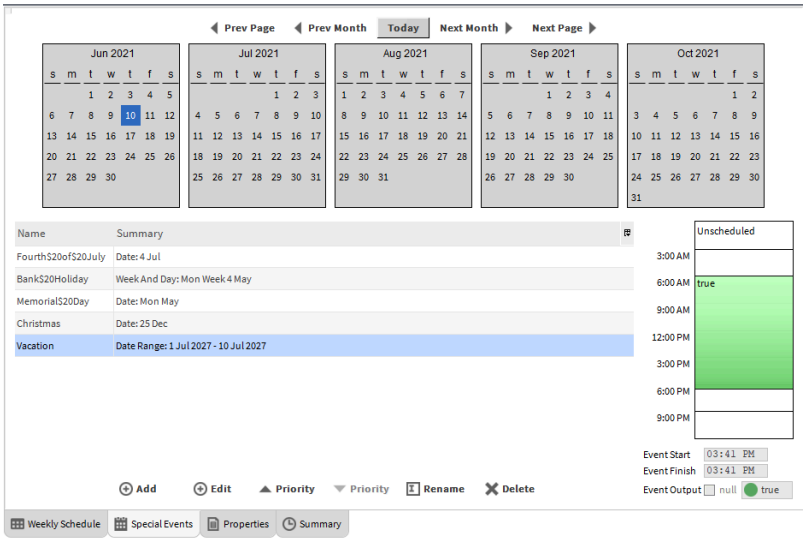
- Delete Event removes the selected event from the schedule.
- Paste Day works with Copy Day to paste the selected day's event into another day.
- All Day Event extends the currently selected or last entered event for the entire day.
- Apply M-F applies the selected event to each day from Monday through Friday.
- Copy Day copies the selected event in preparation to paste it elsewhere in the schedule.
- Clear Day removes all events for the selected day.
- Clear Week removes all events for the entire week.

Parent topic: Scheduler

Special Events tab

This tab configures exceptions to the normal weekly schedule as special events. These events override and intermingle with events in the regular weekly schedule. To manage these events you use the controls at the bottom of the tab to Add, Edit, Prioritize, Rename, and Delete.

Figure 1. Special Events tab in weekly Scheduler



To view this schedule, double-click the schedule component and select the **Special Events** tab. The view title at

the top displays the name of the schedule.

The table lists existing special events (if any) by name and summary. When you select a special event, the schedule highlights its day(s) of occurrence in the monthly calendars at the top of the view, and displays its associated event actions in the right-side column.

Property	Value	Description
Event Start	time of day, AM and PM with rotator	Fine tunes the start time for the event. To configure this property, select the event and either type in the time and AM, PM or use the up and down arrows.
Event Finish	time of day, AM and PM with rotator	Fine tunes the end time for the event. To configure this property, select the event and either type in the time and AM, PM or use the up and down arrows.
Event Output	null and true (default) or false	Displays numbers or text in each schedule block. What to display depends on the type of schedule. If you intend nothing to display, click the null check box. An enum schedule displays a facet name. This is why it is important to configure facets before configuring an enum schedule. If a **NumericSchedule** or **StringSchedule**, you type the value in this property, then press **Enter** to register it in the event block.

Controls

The controls above the tabs manage the configuration of each event.

- **Add** creates a new event.
- **Edit** opens an existing event so you can update it.
- **Priority** up moves the event higher in the table.
- **Priority** down moves the event lower in the table.
- **Rename** opens a window so you can change the event name.
- **Delete** removes the event from the calendar.

Right-click action menu

If you right-click a selected event, this menu of options opens:

- **All Day Events**
- **Clear Days**

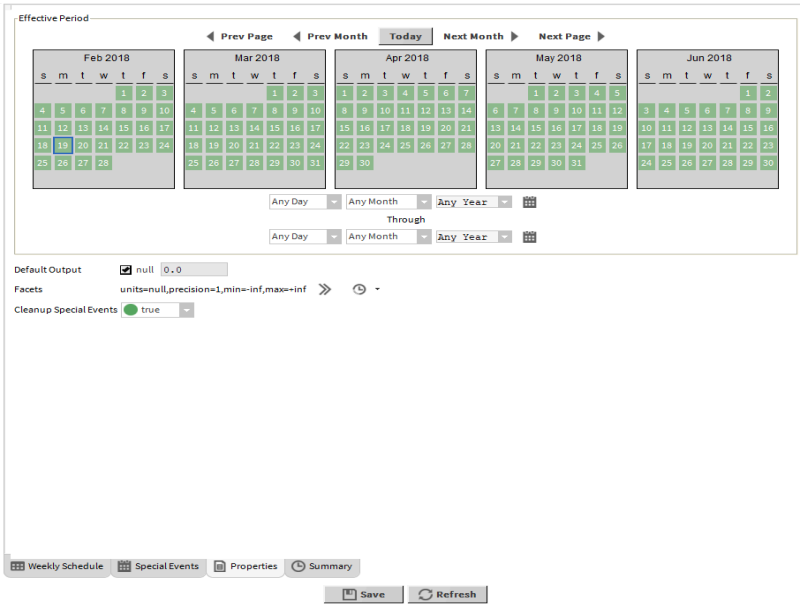
- **Schedule Defaults**

Parent topic: Scheduler

Properties tab

This tab configures schedule properties.

Figure 1. Properties tab in weekly Scheduler



To view this schedule, double-click the schedule component and select the **Properties** tab.

Note:

Another configuration property is also available, but only on the **Property Sheet** view of a weekly schedule.

Property	Value	Description
Effective Period	calendars	Configures when the special event applies in the schedule. By default, a weekly schedule is always in effect. When not in effect, the event's output (Out slot) changes to its default output value, regardless of its weekly schedule or any special events. In most cases, you leave weekly schedules always in effect. However, if you have an application for a schedule that is in effect only at certain times, use the drop–down from and to lists below the calendars to limit the effective period. When you save the changes, the calendar highlights only effective days.

Property	Value	Description
Default Output	null or value (defaults based on the type of schedule: false for **BooleanSchedule**s and null for the other three schedule types	Defines what the Event Output value should be when the time in the schedule is undefined. The white area in the calendar indicates where the default value displays. The default output value is also used whenever the schedule is not effective. null may be a valid choice depending on control logic.
Facets	Config Facets window (default based on the type of schedule:**BooleanSchedule** trueText: true, falseText: false**EnumSchedule** — range: (not defined)**NumericSchedule** — units: (null), precision: 1 **StringSchedule** — (not applicable)	Determine how its output value is formatted for display. For example, instead of true and false for a **BooleanSchedule**, you may need On and Off. Assigned facets appear in scheduler views when adding events and displaying summary data.Facets are especially important for **EnumSchedule**. You need to define range facets before you add weekly schedule events (to pick an event's enumerated value). Range facets should match those used in any controlled (output-linked) enum writables. In the case of string schedules (as for all string-type components) facets have no application.
Cleanup Special Events	true (default) or false	Determines what happens to events that have already occurred. true automatically deletes one-time special events that have occurred (and will not be effective again). Deleted special events send a message to the schedule log, and the event no longer appears in the **Special Events** tab. false retains one-time even though they will not occur again.

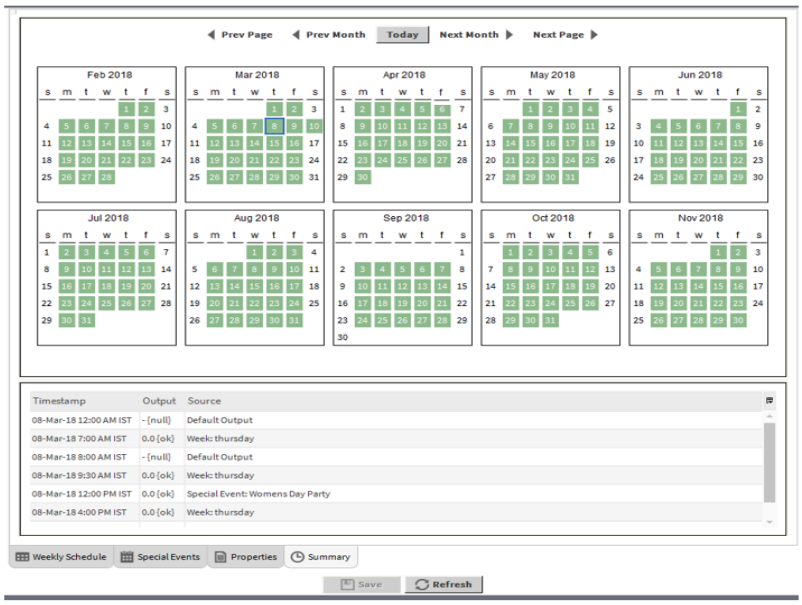
Parent topic: Scheduler

Summary tab

This tab displays a read-only summary, which you can use to review a weekly schedule's configuration.

Displayed on the tab are schedule events for the current day, including output and output source. You can click on any calendar month to review its schedule events. To see all combined active dates view, click outside the table. If adjustments are necessary, click the other tabs as needed, to make changes.

Figure 1. Summary tab shows all events for any selected day



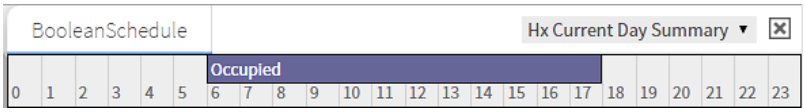
To view this schedule, double-click the schedule component and select the **Summary** tab.

Parent topic: Scheduler

Current Day Summary

This view is available only on weekly schedule (**BooleanSchedule**, **EnumSchedule**, **NumericSchedule**, and **StringSchedule**) components. It provides a simple graphical display of all schedule events for the current day, moving left-to-right from 0-to-24 hours.

Figure 1. Current Day Summary view (browser connection)

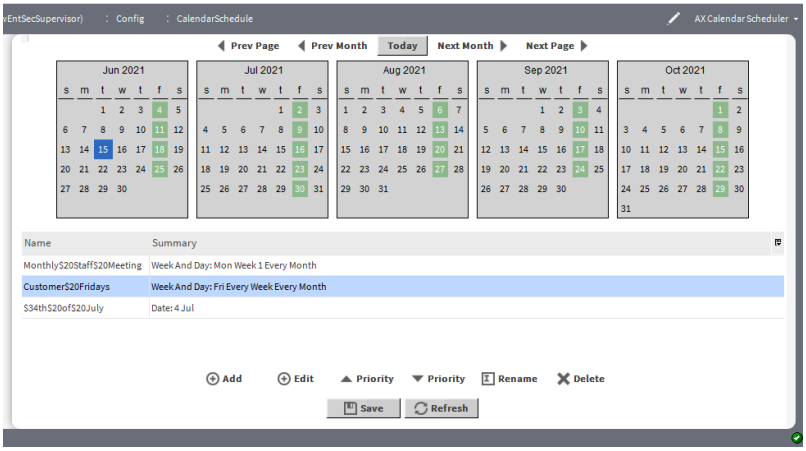


Parent topic: Schedule plugins

Calendar Scheduler view

This is the default view for CalendarSchedule components. In this view you can configure event days per month. Typically, calendar events represent holidays.

Figure 1. Calendar Scheduler view



To open this view, double-click a CalendarSchedule component in the Nav tree. If this component is not in your Nav tree, drag it in from the schedue palette.

The calendar identifies the events.

The controls below the calendar configure calendar events.

The table lists existing calendar events (if any) by name and summary.

Note: When using a web browser connection, the view contains standard controls and a table of existing events. It does not include calendar month depictions.

Also, you can use the standard table controls to add or remove table columns, as well as configure, filter, search, refresh, or export tables as .CSV or .PDF files by initiating actions.

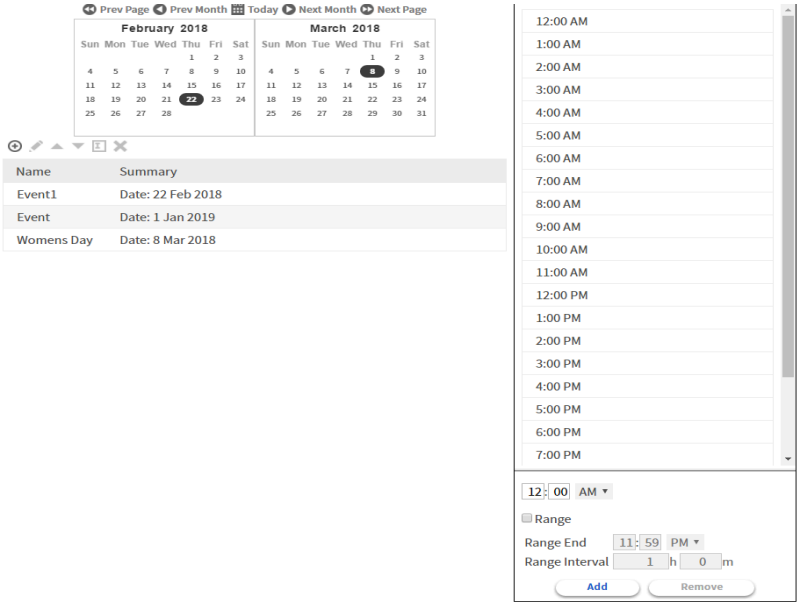
Parent topic: Schedule plugins

TriggerScheduler

This is the default view of TriggerSchedule components. You use this view to configure schedule events in a TriggerSchedule. Trigger schedules are defined by a combination of calendar day (or days) and trigger event time(s). Trigger times apply to all trigger events (calendar-side entries).

Double-click a TriggerSchedule to see its default **Trigger Scheduler** view, as shown below.

Figure 1. Trigger Scheduler view



The Trigger Scheduler has two sides:

- The calendar (left) side where you add events. It operates like the **CalendarScheduler** view.
- The time picker (right) side where you add trigger times for the schedule to fire its trigger output. Includes the ability to add repeating intervals.

Note: Trigger times, as set in the time picker, apply to all calendar events (if more than one).

Existing trigger events (if any) are listed in the event table by name and summary. When you select a trigger event, its day(s) of occurrence are highlighted in green in the monthly calendars at the top of the view. Trigger times are listed in the time picker area. The screen capture shows a trigger for each hour from 12 AM to 7 PM.

Property	Value	Description
Time	hours minutes and AM/PM (defaults to 12:00 AM)	Sets a time in the time picker.
Range	check box (defaults to null)	Selects time picker configuration using a range end and interval.
Range End	hours minutes and AM/PM (defaults to 11:59 PM)	Sets the last time entry for the time picker.
Range Interval	hours and minutes (defaults to 1 hour 00 minutes)	

Controls

The controls above the tabs manage the configuration of each event.

- **Add** creates a new event.
- **Edit** opens an existing event so you can update it.

- **Priority** up moves the event higher in the table.
- **Priority** down moves the event lower in the table.
- **Rename** opens a window so you can change the event name.
- **Delete** removes the event from the calendar.

Right-click action menu

Right-click on a day for an event menu. Event menu options are straightforward, and may include the following:

- Add creates a trigger event.
- Edit updates a trigger event.
- Rename changes the name of the trigger event.
- Delete removes the selected event from the schedule.

Time picker buttons

- **Add** creates a trigger time.
- **Remove** deletes a trigger tag.

Parent topic: Schedule plugins

Windows

Windows create and edit database records or collect information when accessing a component. You access them by dragging a component from a palette into a station or by clicking a button.

Windows do not support **On View (F1)** and **Guide on Target** help. To learn about the information each contains, search the help system for key words.

- **Add event based on Date**
  When adding event to a weekly or calendar schedule, clicking Add defaults to this window.
- **Add event based on Date Range**
  This window has a start and end range, each with three range values, day of the month, month of the year and year.
- **Add event based on Week and Day**
  This special event type has three values: weekday, week in the month, and month of the year.
- **Add custom event**
  These controls select various combinations of day, month, weekdays and year with five custom values: day of the month, month of the year, weekday, week in the month and year.
- **Add reference event**
  These controls select a specific **CalendarSchedule** component by reference and add it to the table.

Add event based on Date

When adding event to a weekly or calendar schedule, clicking Add defaults to this window.

Figure 1. Add and Edit event window for type Date

To open this window, double-click the schedule component in the station and click **Add**.

You can make only one selection in each date value (weekday, day of month, month, year). Each date value offers an Any selection, in addition to a specific selection.

The schedule combines all date values. For example, if weekday is Tuesday, day of month as 5, and the remaining date values are any, the schedule sets up the event only on Tuesday, the fifth of any month in any year. If a month does not have a Tuesday the fifth, no event is scheduled that month.

Property	Value	Description
Name	text (defaults to `Event`	Identifies the event. For example, “Christmas Day” or “Half-Day”, etc.
Type	drop-down list (defaults to Date)	Selects the desired type of event. The properties that follow depend on the Type you choose. Date provides various combinations of weekday, numerical date, month, month

Property	Value	Description
		combinations and year. Date Range defines the event by start and end dates using for each a combination of day, month and year. Week and Day provide a combination of day of the week, week in month and month. Custom provides various combinations of day, month, weekdays and year.
Day	drop-down list	Selects Any Weekday or any specific weekly day.
Weekday	drop-down list	Selects the day of the week.
Day of the month	drop-down list	Selects the date in the month.
Month	drop-down list	Selects the month.
Year	drop-down list	Selects the year

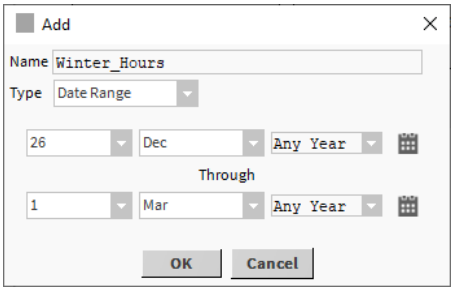
Parent topic: Windows

Add event based on Date Range

This window has a start and end range, each with three range values, day of the month, month of the year and year.

You can make only one selection in each date range value (day of the month, month and year). Each date range value offers an Any selection, in addition to a specific selection.

Figure 1. Add event window for Date Range



To open this window, double-click the schedule component in the station, click **Add** and select Date Range for Type.

The schedule combines all date values. In addition, the start day can be after the end date. For example, as shown in the above figure, the start day can be in December and the end date in March. This event occurs December, January and February.

In addition to Name and Type, these properties configure the date range.

Property	Value	Description
Day (from and to)	drop-down list (defaults to Any Day)	Selects the day of the month.
Any Month (from and to)	drop-down list (defaults to Any Month)	Selects the month in the year.
Year (from and to)	drop-down list	Selects the year.

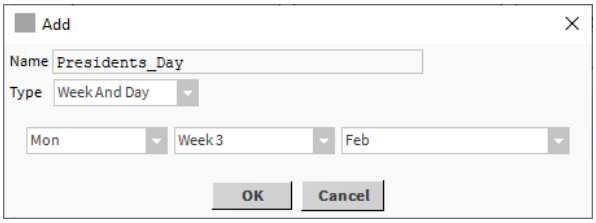
Parent topic: Windows

Add event based on Week and Day

This special event type has three values: weekday, week in the month, and month of the year.

You can make only one selection in each week and day value (week in the month, month of the year). Each date range value offers an Any selection, in addition to a specific selection.

Figure 1. Add event window for Week and Day



This window opens when you click the add control and select Week And Day for Type.

The schedule combines all date values. For example, as shown in the above image, if selections are for weekday as Monday, week as Week3 and month as February, the event occurs only on the third Monday in February.

Property	Value	Description
Week day	drop-down list (defaults to Any Weekday)	Selects the day during the week
Any Week	drop-down list (defaults to Any Week)	Selects the week.
Any Month	drop-down list (defaults to Any Month)	Selects the month.

Parent topic: Windows

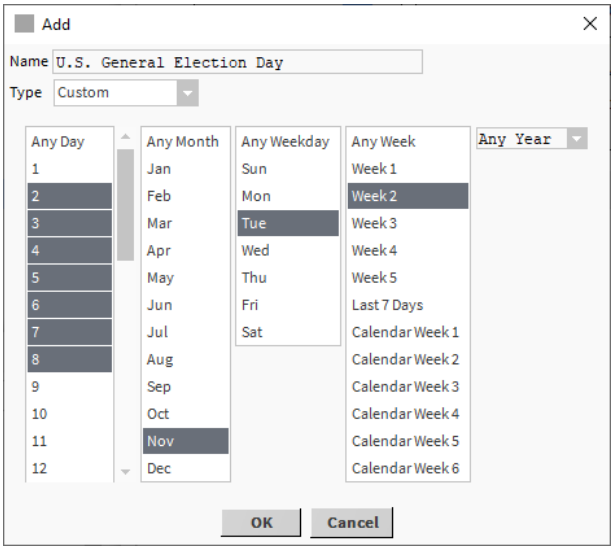
Add custom event

These controls select various combinations of day, month, weekdays and year with five custom values: day of the month, month of the year, weekday, week in the month and year.

Unlike the other calendar types, you can make multiple selections within each custom value(except if you select one of the any options). To select multiples, first select something other than an any option, then hold the **Ctrl** or **Shift** key while you select more values.

Each custom value offers an Any selection, in addition to a specific selection.

Figure 1. Add custom event



This window opens when you click the add control and select Custom for Type.

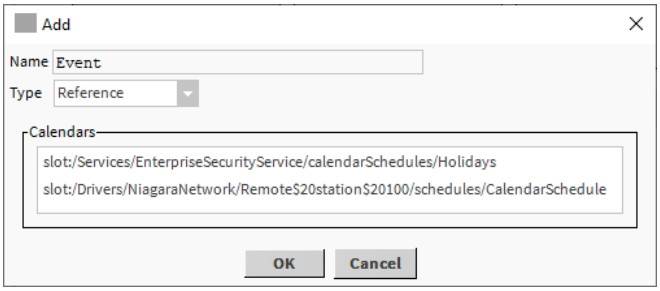
Within any value the schedule sets up an OR relationship. The overall result is from combining all values. For example, the above image shows a custom selection for U.S. General Election Day, which occurs on the first Tuesday after the first Monday in November.

Parent topic: Windows

Add reference event

These controls select a specific **CalendarSchedule** component by reference and add it to the table.

Figure 1. Add reference event window



This window opens when you click the add control and select Reference for Type.

Property	Value	Description
Name	text (defaults to `Event`)	Identifies the name of the event.
Type	drop-down list	Selects the type of data to add to the table.
Calendars	slot list	Selects an existing calendar.

Parent topic: Windows