

Technical Document

Scheduling Guide

August 18, 2015

niagara⁴

Scheduling Guide

Tridium, Inc.

3951 Westerre Parkway, Suite 350
Richmond, Virginia 23233
U.S.A

Confidentiality

The information contained in this document is confidential information of Tridium, Inc., a Delaware corporation ("Tridium"). Such information and the software described herein, is furnished under a license agreement and may be used only in accordance with that agreement.

The information contained in this document is provided solely for use by Tridium employees, licensees, and system owners; and, except as permitted under the below copyright notice, is not to be released to, or reproduced for, anyone else.

While every effort has been made to assure the accuracy of this document, Tridium is not responsible for damages of any kind, including without limitation consequential damages, arising from the application of the information contained herein. Information and specifications published here are current as of the date of this publication and are subject to change without notice. The latest product specifications can be found by contacting our corporate headquarters, Richmond, Virginia.

Trademark notice

BACnet and ASHRAE are registered trademarks of American Society of Heating, Refrigerating and Air-Conditioning Engineers. Microsoft, Excel, Internet Explorer, Windows, Windows Vista, Windows Server, and SQL Server are registered trademarks of Microsoft Corporation. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Mozilla and Firefox are trademarks of the Mozilla Foundation. Echelon, LON, LonMark, LonTalk, and LonWorks are registered trademarks of Echelon Corporation. Tridium, JACE, Niagara Framework, NiagaraAX Framework, and Sedona Framework are registered trademarks, and Workbench, WorkPlaceAX, and AXSupervisor, are trademarks of Tridium Inc. All other product names and services mentioned in this publication that is known to be trademarks, registered trademarks, or service marks are the property of their respective owners.

Copyright and patent notice

This document may be copied by parties who are authorized to distribute Tridium products in connection with distribution of those products, subject to the contracts that authorize such distribution. It may not otherwise, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written consent from Tridium, Inc.

Copyright © 2015 Tridium, Inc. All rights reserved.

The product(s) described herein may be covered by one or more U.S or foreign patents of Tridium.

About this guide

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, see [Chapter 1 Scheduling overview, page 7](#).
- To begin working with schedules, see [Chapter 2 Common tasks, page 11](#).
- For detailed information on scheduling components, see the section on [Chapter 3 Schedule components, page 23](#).
- For detailed information on scheduling views, see the section on [Chapter 4 Schedule plugins, page 47](#).

Document change log

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

- Initial release publication: August 18, 2015

Contents

- About this guide3**
- Document change log3**
- Chapter 1 Scheduling overview7**
 - Schedule component categories7
 - Schedule component views.....7
 - Schedule component links8
 - Schedule special events9
 - Schedule exports and imports (master/slave)..... 10
- Chapter 2 Common tasks.....11**
 - Adding a schedule component from the palette 11
 - Copying a pre-configured schedule 11
 - Configuring weekly schedules..... 12
 - Configuring properties for a weekly schedule 12
 - Configuring normal events for a weekly schedule 12
 - Configuring a special event in a weekly schedule 13
 - Reviewing a weekly schedule’s configuration 14
 - Setting up different permissions for the Special Events tab 14
 - Configuring calendar schedules 16
 - Configuring a calendar schedule 16
 - Deleting a calendar event 17
 - Configuring trigger schedules..... 17
 - Adding a trigger event 17
 - Adding trigger event times..... 18
 - Linking weekly schedules..... 19
 - Linking a weekly schedule using the wire sheet..... 19
 - Linking a weekly schedule using the Nav tree 19
 - Linking trigger schedules..... 20
 - Linking a trigger schedule using the wire sheet..... 20
 - Linking a trigger schedule using the Nav tree 21
 - Importing schedules 21
 - Importing A Schedule Or Calendar From A Supervisor 21
- Chapter 3 Schedule components23**
 - About Weekly Schedules 23
 - Types of weekly schedule components 23
 - Weekly schedule output processing 32
 - Weekly Schedule usage notes 33
 - About Special Events..... 34
 - About schedule properties 36
 - About calendar schedules..... 36
 - Calendar schedule usage 36
 - schedule-CalendarSchedule 37
 - About trigger schedules 38

- schedule-TriggerSchedule 39
- About ScheduleSelector components..... 39
 - Types of ScheduleSelector components..... 40
 - ScheduleSelector configuration examples 44
- Chapter 4 Schedule plugins47**
 - Scheduler..... 47
 - Weekly Schedule tab 48
 - Special events tab 51
 - Properties tab 53
 - Summary tab..... 56
 - Current Day Summary 56
 - Calendar Scheduler view 57
 - Calendar event selections 57
 - TriggerScheduler 60
- Index.....63**

Chapter 1 Scheduling overview

Topics covered in this chapter

- ◆ Schedule component categories
- ◆ Schedule component views
- ◆ Schedule component links
- ◆ Schedule special events
- ◆ Schedule exports and imports (master/slave)

Schedules are created using schedule components found in the **schedule** palette. Place these components in a station, configure, and link as needed to provide scheduling control of other components.

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

The scheduling model is summarized in the following sections.

Schedule component categories

Schedule components may be categorized as follows:

Component category	Description
Weekly schedules	Weekly schedules define regular, repeating, events by "time-of-day" and "day-of-week." Also, you can configure any number of "special events" on a weekly schedule. Weekly schedules are the most used schedule components. Four different types vary by data category (Boolean, Numeric, Enum, and String). Each is identical except for input/output. For more details, see <i>"About weekly schedules"</i> .
Calendar schedules	Calendar schedules define specific days. Use a calendar schedule to define days with scheduling exceptions (e.g. holidays), and reference them in the "special events" setup of weekly schedules. For more details, see <i>"About calendar schedules"</i> .
Trigger schedules	Trigger schedules fire actions or topics. With a set of pre-configured schedules and a ScheduleSelector component, you can simply choose from a list of valid schedules to setup or change the schedule of a device. For more details, see <i>"About trigger schedules"</i> .
Schedule selectors	ScheduleSelector components provide an easy way for you to select a schedule to use for controlling a particular component. As needed, you typically link its output to an action of a control point or extension. For more details, see <i>"About ScheduleSelector components"</i> .

Schedule component views

The default view for any schedule component is its **Scheduler** view, where you define related days and events.

View	Description
Scheduler	<p>Default view for any weekly schedule component. In Workbench the Scheduler view provides four configuration tabs:</p> <ul style="list-style-type: none"> • Weekly Schedule — Sunday-through-Saturday (weekly) event times/values. Use this to define the regular weekly events. • Special Events — All exceptions to the defined weekly schedule, as special events. Use this to define special event times. • Properties — Important properties such as default output, 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. <p>Also available under the Scheduler view dropdown list:</p> <ul style="list-style-type: none"> • Current Day Summary view — provides a simple 24-hour linear graph of schedule event times and values for the current day. <p>NOTE: In a web browser the Scheduler view provides two configuration tabs: Weekly Schedule and Special Events. While the Current Day Summary and Property Sheet views are available under Scheduler view dropdown list.</p>
Calendar Scheduler	<p>Default view for a CalendarSchedule component. The Calendar Scheduler view provides a calendar where you can add, edit, or delete specific calendar days (exceptions to the regular weekly schedule) and establish relative priorities.</p>
Trigger Scheduler	<p>Default view for a TriggerSchedule component. The Trigger Scheduler view provides a two-part view combining a “day picker” and an event “time picker” for specifying when topics are fired on those days.</p>

Schedule component 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 source) to a slot on another component. You can link the same weekly schedule to many target components.

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.

Link type	Usage description
Weekly schedule link	<p>Links a weekly schedule to one or more “writable” control points. Examples: you can link the Out slot of weekly schedule types as follows:</p> <ul style="list-style-type: none"> • 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. <p>Alternatively, 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 time. In this case, the Default Output value of all schedule components (except last in chain) must be <code>null</code>. For more details, see “Linking weekly schedules.”</p>
Trigger schedule link	<p>Links a trigger schedule to an action of a control point or, more commonly, to an action of a point extension. Example:</p> <ul style="list-style-type: none"> • You could link a TriggerSchedule to the “ResetElapsedActiveTime” action of a DiscreteTotalizerExt, a point extension for a BooleanPoint used to accumulate runtime. If the trigger schedule was configured to fire only on the first day of every month (at 12:00am) that extension could be used to hold the current month’s runtime. <p>For more details, see “Linking trigger schedules.”</p>

Schedule special events

Special events apply to weekly schedules only, and are considered any exception to the (normal) weekly schedule.

Characteristics	Description
Where used	Each weekly schedule component has its own special events, configured on the Special Events tab in its Scheduler view. Also, event times (and values) entered for any special event apply only to that schedule.
Occurrence	Special events can be one-time only event changes or recurring event changes, such as holidays.
Configuration	Configuration includes the day(s) of occurrence and related time-of-day events. In the time-of-day event definitions, you can have them “intermingle” with regular weekly events, or completely override the weekly schedule.
Prioritize	You visually prioritize special events, via list order. This allows any overlapping special events to occur in an ordered fashion.
Reference type event	If the special event is a “reference” type (references a separate schedule), days of occurrence are specified in the referenced CalendarSchedule component. This allows you to globally change the days that special events occur in weekly schedules, by editing one or more referenced calendar schedules. For more details, see “ <i>Calendar schedule usage.</i> ”

NOTE: You can provide different permissions for the special events of a weekly schedule (**BooleanSchedule**, **NumericSchedule**, etc.) than for the rest of the schedule’s configuration. For more details see, “*Setting up different permissions for the Special Events tab.*”

Schedule exports and imports (master/slave)

Using the Niagara driver architecture, you can create “master/slave” schedules in order to share schedule configuration between devices. This allows you to globally update the configuration of any slave schedule by simply making changes to its master schedule.

NOTE: Niagara 4.0 Schedule components that are exported to NiagaraAXstations should not contain any additional properties that are not recognized by AX. Adding additional components/slots under a Niagara 4.0 schedule increases the risk that exporting them to AX will fail.

The typical application for this is in a multi-station Niagara network, where you import a schedule component from another station. Typically, you do this in a station, importing 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 a Niagara schedule, a “schedule export descriptor” is automatically created on the sending (master) station, under the Station component that represents the receiving side. This allows for sending-side management of configuration synchronization. For more details, refer to the section “Station Schedules import/export notes” in the *Drivers Guide*.

If 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 Niagara schedule components.

For more details, see the following topics in the *Drivers Guide*:

- “About the Schedules extension”
- “About Schedules extension views”

NOTE: The Bacnet driver lets you export Niagara schedules from the station to existing BACnet Schedule and Calendar objects in a BACnet device, acting as the “master” source. Also, you can expose Niagara 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.” For more details, see the *Bacnet Users Guide*.

Chapter 2 Common tasks

Topics covered in this chapter

- ◆ Adding a schedule component from the palette
- ◆ Copying a pre-configured schedule
- ◆ Configuring weekly schedules
- ◆ Configuring calendar schedules
- ◆ Configuring trigger schedules
- ◆ Linking weekly schedules
- ◆ Linking trigger schedules
- ◆ Importing schedules

Common tasks include the following types:

Adding a schedule component from the palette

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

Prerequisites:

- Open connection to a station

Step 1 Open the **schedule** palette.

Step 2 Click and 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

Step 3 In the **Name** dialog, type a `scheduleName` or use the default name and click **OK**

The schedule or calendar is now in your station, with default values.

Double-click your new schedule to display its **Scheduler** view. At this point, you can configure the schedule by adding events.

Copying a pre-configured schedule

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

Prerequisites:

- Pre-configured schedule component in a bog file or other station database

NOTE: Currently, in a web browser connection you do not have the options to copy or paste components.

Step 1 In the Nav tree, expand `My File System` and navigate to the saved bog file or station `config.bog` file.

Step 2 When you locate the desired schedule component, right-click and select **Copy** (or use drag and drop to copy).

Step 3 Right-click and select **Paste** in either of these locations:

- In the Nav tree, on a folder in your station
- In the view pane, on the **Wire Sheet** or **Property Sheet** view of a folder in your station

The **Name** dialog displays.

Step 4 Type the desired name and click **OK**.

The schedule component with pre-configured values is now in your station.

Configuring weekly schedules

Weekly schedules define the normal, regular (repeating) events by "time-of-day" and "day-of-week." Typically, these are the most used schedule components. There are four different types of weekly schedules that vary by data category (**BooleanSchedule**, **EnumSchedule**, **NumericSchedule**, and **StringSchedule**). Each of the four schedule types is identical except for input/output. Scheduling components are found in the **schedule** palette. Use the **Scheduler** view to configure a weekly schedule.

There are many ways that you can configure a weekly schedule, the following basic procedure is just one approach.

- Double-click a weekly schedule component to open the Scheduler view
- Configure important properties.
- Configure the normal weekly schedule.
- Configure special events, if any.
- Review all configurations.

Configuring properties for a weekly schedule

Use the **Weekly Scheduler** view to configure certain properties of a weekly schedule. The following is one way to approach configuration.

Prerequisites:

- A weekly schedule component in your station

Step 1 Double-click the weekly schedule component.

Step 2 In the **Scheduler** view, click the **Properties** tab, or in a web browser connection select the **Property Sheet** view, and configure the following (leave other properties at default values):

- **Facets** — This is critical if an **EnumSchedule**, and optional if a **BooleanSchedule** or **NumericSchedule**.
NOTE: Facets do not apply if a **StringSchedule**.
- **Default Output** — This is the schedule's output whenever an event (either weekly or special event) is not active. It is also used whenever the component is "not effective."

Step 3 Click **Save**.

Configuring normal events for a weekly schedule

Weekly schedules define regular, repeating events such as the normal hours that a space is occupied. Use the **Scheduler** view to configure a **WeeklySchedule** component.

Prerequisites:

- **Scheduler** view open

Step 1 In the events table on the **Weekly Schedule** tab, click and drag on a series of hours in a day to add an event, and enter its output value .

NOTE: If a Boolean or EnumSchedule, select an output value from the drop-down control. If a Numeric or StringSchedule, type the desired output value and press **Enter**.

Step 2 Right-click the event to display the event menu and use any of the options to configure the event.

Step 3 Continue to add, delete, or adjust events as needed to configure the weekly schedule as you wish.

Step 4 Click **Save**.

Configuring a special event in a weekly schedule

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

Prerequisites:

- A weekly schedule component in your station

Step 1 Double-click the weekly schedule component.

Step 2 In the **Scheduler** view, click the **Special Events** tab.

Step 3 Click **Add** to add a new special event.

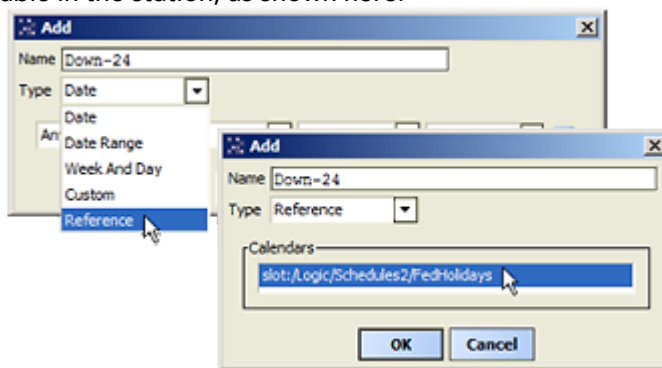
Step 4 In the **Add** dialog, specify:

- **Name** — Type a unique, identifiable name for this event. For example “Christmas Day” or “Half-Day”, etc.

NOTE: By default, the name is `Event`. You can change this value at a later time.

- **Type** — Select the desired Type option, and enter specific day criteria (according to type).

Type	Description
Date	Default selection. Select various combinations of weekday, numerical date, month or month combinations, and year.
Date Range	Select start and end range, using for each a combination of day, month, year.
Week and Day	Select combination of day of week, week in month, month.
Custom	Select various combinations of day, month, weekdays, and year.
Reference	By reference to a specific CalendarSchedule component in the station. Displays a second Add dialog listing all CalendarSchedules available in the station, as shown here.



Step 5 Click **OK**.

The new event is added to this schedule’s special events.

Step 6 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.

Step 7 Use the **Start** and **Finish** fields to fine-tune the event time.

- Step 8 Select the desired **Output** value.
- Step 9 Use the **Priority** controls to set the priority for this event.
- Step 10 Click **Save**.

Reviewing a weekly schedule's configuration

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

Prerequisites:

- Configured weekly schedule
- Step 1 In the **Scheduler** view, click the **Summary** tab.
The calendar shows the current day with its schedule events, including output and output source.
 - Step 2 Click any day on any calendar month to see its schedule events.
 - Step 3 Click the **Weekly Schedule**, **Special Events**, or **Properties** tab (if in Workbench) as needed to make adjustments.
 - Step 4 Click **Save**.

Setting up different permissions for the Special Events tab

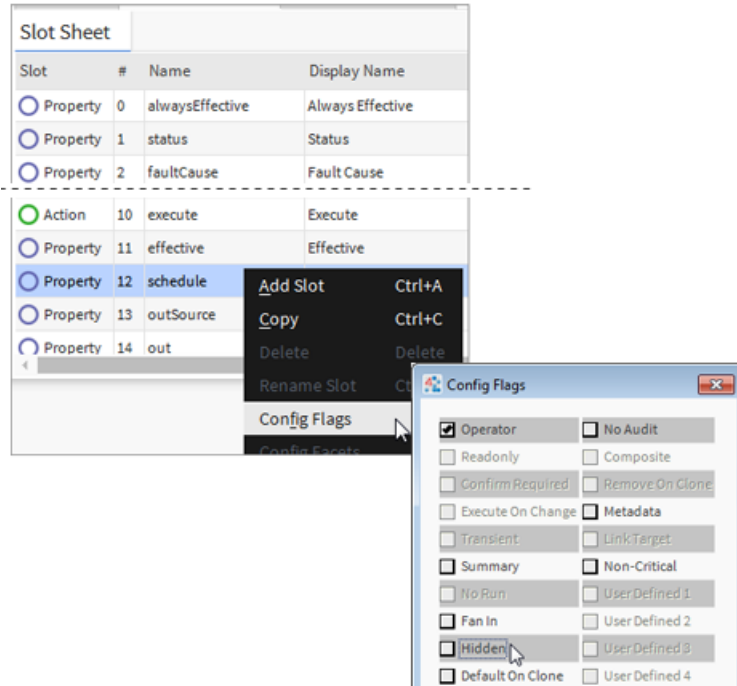
Typically, a user with operator-level **Read** permissions on a weekly schedule would see all of the schedule tabs but would not be able to edit the schedule. However, this optional procedure allows you to configure different permissions for the **Special Events** tab in order to allow the user to have **Write** access only for managing special events.

Prerequisites:

- Existing weekly schedule with operator-level **Read** permissions on all tabs of the **Scheduler** view.

NOTE: You must be logged in as an admin-level user in order to perform this procedure. Also, the permission change described here applies only to the schedule's **Special Events** tab. It does not alter permissions for the other schedule tabs or properties.

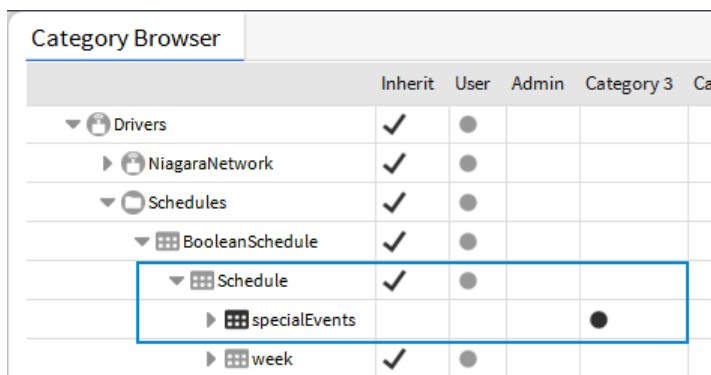
- Step 1 Open **Slot Sheet** view for the schedule component.
- Step 2 Right-click the **Schedule** slot and select **Config Flags**
- Step 3 In the **Config Flags** dialog, click to deselect the **Hidden** check box and click **OK**.



Once the **schedule** slot is no longer hidden, you can expand it in the **CategoryBrowser** and make changes to the schedule component.

Step 4 In **Services**, double-click **CategoryService** to display the **CategoryBrowser** view, expand and the **Schedule** component's child **CompositeSchedule** to expose **specialEvents** and perform the following:

- Click to deselect the **Inherit** option
- Click to clear the assigned category
- Click to select a different category—one that a user could have **operator Write** permissions on, such as **Category 3**, as shown here:



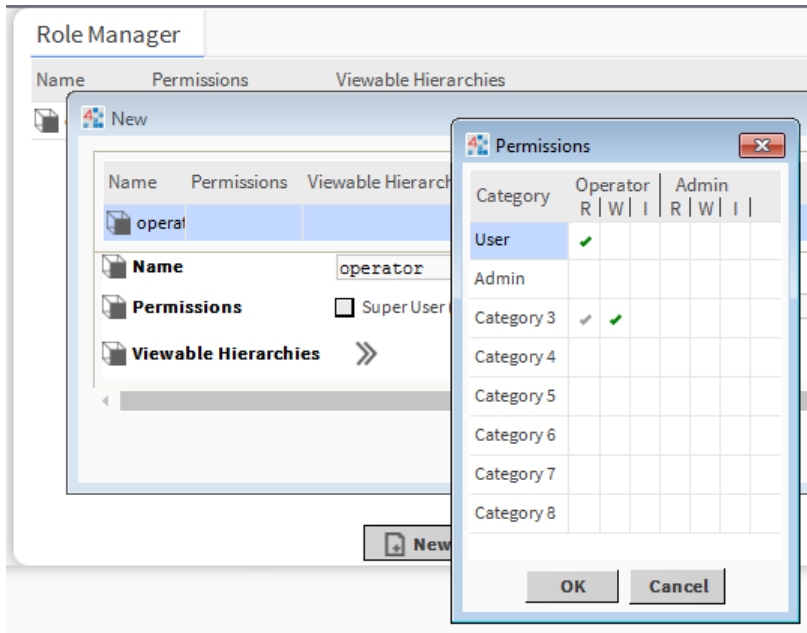
- Click **Save** in the Workbench toolbar.

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

Step 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:

- In **Services**, double-click **RoleService** to display the **RoleManager** view, click **New** and in the **New** dialog click **OK** to add the new role.

- b. In a second **New** dialog, type `Operator` in the **Name** field and click **>>** icon located to the right of the **Permissions** field.
- c. In the **Permissions** dialog, 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 operator role (with necessary permissions) is created and can be assigned to one or more users.

- Step 6** In **Services**, double-click **UserService** to display the **User Manager** view, select a user and click **Edit** then scroll down to **Roles** and 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 schedule's **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.

Configuring calendar schedules

Calendar schedules define specific days. Typically, you use them to define days with scheduling exceptions, for example holidays and other special days. Once you have configured a calendar schedule, you can reference it in the "special events" setup of a weekly schedule.

Scheduling components are found in the **schedule** palette. Use the **Calendar Scheduler** view to configure a `CalendarSchedule` component.

Configuring a calendar schedule

Calendar schedules let you globally specify in the system any exceptions to the normal weekly schedule, for example, holidays and other special days. You configure a calendar schedule by adding one or more events.

Prerequisites:

- `CalendarSchedule` component in a Wire sheet view

Step 1 Double-click the component to open the **Calendar Scheduler** view.

Step 2 Below the calendar events table in the view, click **Add** to add a new calendar event.

NOTE: You can also right-click anywhere on the events table to display an event menu where you can select **Add**.

Step 3 In the **Add** dialog, specify the follow:

- **Name** — Type a unique, identifiable name. For example, `Thanksgiving_Break` or `Cleaning_2`. The default value is simply `Event`.

NOTE: You can change the name later, if needed.

- **Type** — Select the calendar type, and enter specific day criteria (according to type). Type determines selection criteria for day or days, with the following choices:

Type	Description
Date (default)	By various combinations of weekday, numerical date, month or month combinations, and year.
Date Range	By start and end range, using for each a combination of day, month, year.
Week and Day	By combination of day of week, week in month, month.
Custom	By various combinations of day, month, weekdays, and year.

Step 4 Click **OK**.

The new calendar event is added to the events table, with the assigned name.

Step 5 Right-click and select from the event menu to continue adding, editing, renaming, or deleting calendar events as needed.

Step 6 Click **Save**.

Deleting a calendar event

Prerequisites:

Step 1 In the calendar events table, click to select an event to delete

Step 2 Below the calendar events table, click **Delete**.

NOTE: You can also right-click on an event in the calendar events table to display a dropdown menu where you can select `Delete`.

A Delete dialog displays.

Step 3 In the **Confirm** dialog, click **Yes** to delete the event.

Step 4 Click **Save**.

Your changes are saved to the schedule.

Configuring trigger schedules

Trigger schedules provide scheduling control for actions of linked components or their child extensions. **TriggerSchedule** component is found in the **schedule** palette. Use the **Trigger Scheduler** view to configure the schedule.

Adding a trigger event

You can 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.

Prerequisites:

- TriggerScheduler view open

NOTE: Other controls located below the event table allow you to **Edit**, change **Priority** (increase/decrease), **Rename** and **Delete**.

Step 1 Below the trigger events table, click **Add**.

NOTE: You can also right-click anywhere on the events table to display a menu where you can select **Add**.

Step 2 In the **Add** dialog, type a descriptive **Name** for the trigger event.

For example, `FirstDOM` or `Each_WorkHr`. The default value is simply `Event`.

NOTE: You can change the name later, if needed.

Step 3 Select the event **Type** and define as needed. Type determines selection criteria for day or days, with the following choices:

Type	Description
Date (default)	By various combinations of weekday, numerical date, month or month combinations, and year.
Date Range	By start and end range, using for each a combination of day, month, year.
Week and Day	By combination of day of week, week in month, month.
Custom	By various combinations of day, month, weekdays, and year.

The event is added to this calendar's days. It remains selected for further editing, except for Type.

Step 4 Click **OK**.

Step 5 Click **Save**.

Your new trigger event is added to the event table.

Step 6 In the right-side time picker area, add one or more triggers, as needed.

Note the following:

- 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.
- Triggers apply to all trigger events (calendar-side entries).

Step 7 Continue to add, edit, rename, or delete trigger events as needed.

Step 8 Click **Save** when trigger events are configured as you want.

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.

Prerequisites:

- Selected trigger event in the **TriggerScheduler** view

By default, a single "midnight" trigger time already exists (you can delete it if desired).

Step 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.

NOTE: 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.

Step 2 Click **Add**.

The new trigger event time(s) is added to the list in the time-picker pane.

NOTE: To delete a trigger time, click to select the time, then click the **Remove** button. To select multiple trigger times, hold down the **Ctrl** or **Shift** key while you select.

Step 3 Click **Save**.

Your changes are saved to the schedule.

Linking weekly schedules

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

Typically, 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 time. In this case, the **Default Output** value of all schedule components (except last in chain) must be **null**.

Linking a weekly schedule using the wire sheet

Procedure to link a weekly schedule to a component in the same wire sheet.

Prerequisites:

- A configured weekly schedule component and writable target component in a **Wire Sheet** view

Step 1 In the wire sheet, mouse over the weekly schedule’s **Out** slot until highlighted.

Step 2 Click and drag the link wire from the weekly schedule to the bottom of the target component, and release the mouse button.

A **Link** dialog displays. The source (left) side shows the schedule’s **Out** slot preselected.

Step 3 In the target (right) side of the **Link** dialog, click on the desired slot of the target component.

NOTE: 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.

Step 4 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.

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.

Prerequisites:

- Configured weekly schedule and writable target component in the station

Step 1 Expand the Nav tree to locate the source weekly schedule, noting its name.

Step 2 Right-click the weekly schedule and select `Link Mark`.

Step 3 Expand the Nav tree to locate the target component.

Step 4 Right-click the target component and select `Link From <weeklyscheduleName>` from the popup menu.

A **Link** dialog appears.

Step 5 In the source (left) side of the **Link** dialog, click the schedule's `Out` slot.

Step 6 In the target (right) side of the **Link** dialog, click the desired slot of the target component.

NOTE: 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.

Step 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.

Linking trigger schedules

Trigger schedules provide scheduling control for actions of linked components or their child extensions. You can link the output (`Trigger` slot) of a `TriggerSchedule` to an action (or topic) of any component (point or extension) with such a slot. For example, you might link a trigger schedule to the `ResetChangeOfStateCount` action of a `DiscreteTotalizerExt` child of a `BooleanPoint`.

NOTE: Often, you link trigger schedules to point extensions, 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 `BooleanPoint` used to accumulate runtime. If the trigger schedule is configured to fire only on the first day of every month (at 12:00am) that extension could be used to hold the current month's runtime.

Linking a trigger schedule using the wire sheet

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

Prerequisites:

- Configured trigger schedule and target component (point or extension)

Step 1 In the wire sheet, mouse over the trigger schedule's `Trigger` slot until highlighted.

Step 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** dialog displays. The source (left) side has the trigger schedule's `Trigger` slot preselected.

Step 3 In the target (right) side of the **Link** dialog, click on the desired slot of the target component.

Step 4 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.

Linking a trigger schedule using the Nav tree

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

Prerequisites:

- Step 1 Expand the Nav tree to locate the source trigger schedule, noting its name.
- Step 2 Right-click the trigger schedule and select `Link Mark` from the popup menu.
- Step 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.)
- Step 4 Right-click the target component and select `Link From "<target schedule name>"` from the popup menu.
A **Link** dialog appears.
- Step 5 In the source (left) side of the **Link** dialog, click the trigger schedule's **Trigger** slot.
- Step 6 In the target (right) side of the **Link** dialog, click the desired action of the target component.
- Step 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.

Importing schedules

If the station is part of a multi-station Niagara network, you can import Niagara schedule components between stations. You do this using the Niagara driver architecture, specifically, views of the `Schedules` extension under a `NiagaraStation` (device-level) component.

If a Bacnet driver is used, you can also import BACnet Schedules and Calendars (as Niagara schedule components) from a BACnet device. The same basic architecture and methods are used.

Under a `BacnetNetwork`, you can also export Niagara schedules to a specific BACnet device (to configure existing BACnet Schedules and Calendar objects).

For more complete details, refer to the *Drivers Guide*, in sections "About the Schedules extension" and "Station Schedules import/export notes".

Importing A Schedule Or Calendar From A Supervisor

You can import a Niagara or BACnet schedule when working in a station, where the source (master) schedule resides in a remote Supervisor station.

Prerequisites:

- Remote Supervisor station with source schedule
- Station (destination) for imported schedule

- Step 1 In the Nav tree, expand the **Drivers** folder of the station to receive the imported schedule.
- Step 2 Expand the network, either `NiagaraNetwork` or `BacnetNetwork`.
- Step 3 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.
- Step 4 Double-click **Schedules**.
The Niagara or Bacnet **Schedule Import Manager** displays. Any existing schedules (already imported) from this device are listed.
- Step 5 Click **Discover**.

The manager view splits into two panes and a schedule discovery job is started. When complete, schedules available for import appear in the top pane.

Step 6 In the top pane, click to select one or more schedules, then click **Add**.

An **Add** dialog allows you to edit the name of the schedule (as it appears in this station) and several other properties. For more details, refer to “Schedule Import properties” in the *Drivers Guide*.

Step 7 After editing name and properties to suit, click **OK**.

Niagara schedule components are created for the selected items and reside under that device’s **Schedules** extension.

NOTE: You can link into station logic like any other schedules, but you cannot configure these imported schedules (add, delete, or change events or other properties).

Chapter 3 Schedule components

Topics covered in this chapter

- ◆ About Weekly Schedules
- ◆ About calendar schedules
- ◆ About trigger schedules
- ◆ About ScheduleSelector components

Types of scheduling components and their properties are described in the following sections.

About Weekly Schedules

Double-click on a schedule component to display the **Scheduler** view. Use the tabs in this view to edit the schedule. The name of the schedule that you are editing is displayed at the top as the view title.

NOTE: You can move between tabs without losing unsaved data, however, you must click the **Save** button before leaving the view or data is lost.

In Workbench the **Scheduler** view has the following tabs:

- Weekly Schedule tab
These properties specify Sunday-through-Saturday (weekly) event times and output values. Choose an Output option (true or false) and use the Start and Finish fields to specify *when* the Output option is effective.
- Special Events tab
Use these properties to define all exceptions to the defined weekly schedule, as special events.
- Properties tab
This tab is available in Workbench, not in a browser connection. Use these properties to define a default output (output during non-event times), schedule effective times, special event cleanup operation, and schedule facets (display text for outputs).
NOTE: If using a browser connection you can access properties via the **Property Sheet** view.
- Summary tab
This tab is available when using Workbench. For any selected day, provides summary of all schedule events, with source.

In the **Scheduler** view, the **Save** button saves changes on *all tabs* in the view (not just to the one displayed).

- **Save**
The **Save** button is activated only if you have unsaved changes in the component's scheduler. Clicking it downloads your changes to the schedule component's configuration. Immediately following, the **Save** button is deactivated again.
NOTE: For weekly schedules, it is a good practice to save your changes while working in each tab, even though any save applies to changes made on *all tabs*.

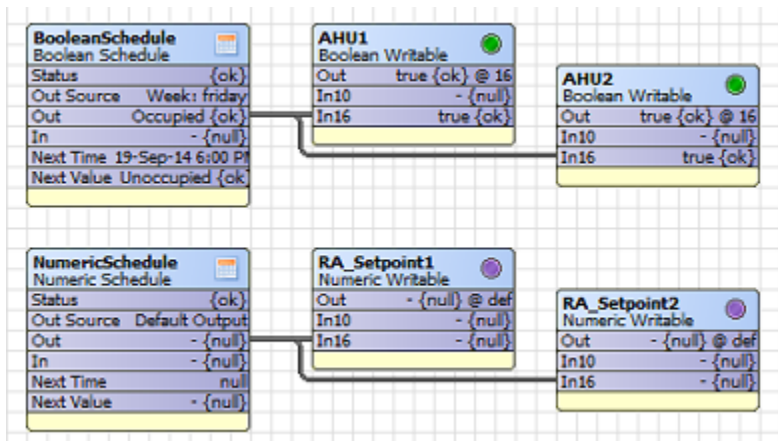
Types of weekly schedule components

There are four different types of weekly schedules that vary by data category (**BooleanSchedule**, **EnumSchedule**, **NumericSchedule**, and **StringSchedule**). Each of the four schedule types is identical except for input/output.

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 of schedules are linked to other components for scheduling control, such as “writable” points of the same type. The figure below shows a **BooleanSchedule** and a **NumericSchedule**, each linked to two writable points. See also, “Schedule component links.”

Figure 1 Schedules linked to writable points



schedule-BooleanSchedule

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. Use it for schedule control of **BooleanWritable** points (typically proxy points). If needed, it also has application for linkage to slots in extensions. For example, using an intermediate **KitControl** object (say, a “Not” logic-type object) use a **BooleanSchedule** for linkage to the **Enabled** slot of an alarm extension.

Property	Value	Description
Status [component]	text	Read-only field. Indicates the condition of the component at last polling. <ul style="list-style-type: none"> • {ok} indicates that the component is polling successfully. • {down} indicates that polling is unsuccessful, perhaps because of an incorrect property. • {disabled} indicates that the Enable property is set to false. • fault indicates another problem.
Fault Cause	text	Read-only field. Indicates why the network, component, or extension is in fault.
Default Output	true or false	Default output of 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.

Property	Value	Description
		For a weekly schedule, set <code>Default Output</code> on the Properties tab in the Scheduler view. Available values must be pre-defined in the component's <code>Facets</code> property.
Cleanup Expired Events	true (default) or false	If true, calendar events of type <code>Date</code> and <code>Date Range</code> that have already occurred, and cannot occur again (as configured), are automatically deleted after they occur. This is recorded in the schedule log, and you no longer see them in the view. If false, all calendar events are retained, 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: <code>ddd hh mm ss</code> , 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	Configuring facets is critical in an <code>EnumSchedule</code> , and optional in a <code>BooleanSchedule</code> or <code>NumericSchedule</code> . Facets do not apply if a <code>StringSchedule</code> . Facets contain additional data applied to input and output values. <ul style="list-style-type: none"> <code>trueText</code> is the text to display when output is true <code>falseText</code> is the text to display when output is false. For example, you might want to set the facet <code>trueText</code> to display "ON" and the facet <code>falseText</code> 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 value. Timestamp of last configuration change.
Out Source	Input, Special Event: <SpecialEventName>, Week: <day_of_week>, (default)	Provides a string source description of the current output, as one of the four options shown left. For example, <code>Week: monday</code> or <code>Special Event: Christmas Break</code>
Out	false {ok} (default)	Output value of schedule component. Output is <code>true</code> during any configured calendar day(s), otherwise it is <code>false</code> Determined by the following, in highest-to-lowest priority: <ul style="list-style-type: none"> 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	- {null} (default)	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. Weekly schedules can be "chained" from Out slot to In slot. Chaining schedules is useful if only one of the schedules is effective at any period in time. In this case, the default output value of all schedule components (except last in chain) must be null.
Next Time	null (default)	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 value. The next scheduled output value, at Next Time. Value is meaningless if Next Time is null.

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 EnumWritable points (typically proxy points). For example, link it to an EnumWritable that proxies a BACnet Multistate Output object, or to an EnumWritable that proxies a LON NVI (using an enumerated SNVT).

Property	Value	Description
Status [component]	text	Read-only field. Indicates the condition of the component at last polling. <ul style="list-style-type: none"> • {ok} indicates that the component is polling successfully. • {down} indicates that polling is unsuccessful, perhaps because of an incorrect property. • {disabled} indicates that the Enable property is set to false. • <code>fault</code> indicates another problem.
Fault Cause	text	Read-only field. Indicates why the network, component, or extension is in fault.
Default Output	true or false	Default output of 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's Facets property.

Property	Value	Description
Cleanup Expired Events	true (default) or false	If true, calendar events of type <code>Date</code> and <code>Date Range</code> that have already occurred, and cannot occur again (as configured), are automatically deleted after they occur. This is recorded in the schedule log, and you no longer see them in the view. If false, all calendar events are retained, 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: <code>ddd hh mm ss</code> , 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	Configuring facets is critical in an <code>EnumSchedule</code> , and optional in a <code>BooleanSchedule</code> or <code>NumericSchedule</code> . Facets do not apply if a <code>StringSchedule</code> . Facets contain additional data applied to input and output values. <ul style="list-style-type: none"> <code>trueText</code> is the text to display when output is true <code>falseText</code> is the text to display when output is false. For example, you might want to set the facet <code>trueText</code> to display "ON" and the facet <code>falseText</code> 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 value. Timestamp of last configuration change.
Out Source	Input, Special Event: <SpecialEventName>, Week: <day_of_week>, (default)	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)	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: <ul style="list-style-type: none"> 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	- {null} (default)	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.

Property	Value	Description
		Weekly schedules can be “chained” from Out slot to In slot. Chaining schedules is useful if only one of the schedules is effective at any period in time. In this case, the default output value of all schedule components (except last in chain) must be null.
Next Time	null (default)	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 value. The next scheduled output value, at Next Time . Value is meaningless if Next Time is null.

schedule-NumericSchedule

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

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

See also, *About weekly schedules*.

Property	Value	Description
Status [component]	text	Read-only field. Indicates the condition of the component at last polling. <ul style="list-style-type: none"> {ok} indicates that the component is polling successfully. {down} indicates that polling is unsuccessful, perhaps because of an incorrect property. {disabled} indicates that the Enable property is set to false. fault indicates another problem.
Fault Cause	text	Read-only field. Indicates why the network, component, or extension is in fault.
Default Output	true or false	Default output of 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's Facets property.
Cleanup Expired Events	true (default) or false	If true, calendar events of type Date and Date Range that have already occurred, and cannot occur again (as configured), are automatically deleted after they occur. This is recorded in the schedule log, and you no longer see them in the view.

Property	Value	Description
		If false, all calendar events are retained, 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	<p>Configuring facets is critical in an EnumSchedule, and optional in a BooleanSchedule or NumericSchedule. Facets do not apply if a StringSchedule.</p> <p>Facets contain additional data applied to input and output values.</p> <ul style="list-style-type: none"> • trueText is the text to display when output is true • falseText is the text to display when output is false. <p>For example, you might want to set the facet trueText to display "ON" and the facet falseText to display "OFF."</p> <p>"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.</p>
Last Modified		Read only value. Timestamp of last configuration change.
Out Source	Input, Special Event: <SpecialEventName>, Week: <day_of_week>, (default)	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)	<p>Output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false</p> <p>Determined by the following, in highest-to-lowest priority:</p> <ul style="list-style-type: none"> • 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	- {null} (default)	<p>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.</p> <p>Weekly schedules can be "chained" from Out slot to In slot. Chaining schedules is useful if only one of the schedules is effective at any period in time. In this case, the default output</p>

Property	Value	Description
		value of all schedule components (except last in chain) must be null.
Next Time	null (default)	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 value. The next scheduled output value, at Next Time . Value is meaningless if Next Time is null.

schedule-StringSchedule

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

Allows schedule control of StringWritable points (possibly proxy points).

See also, *About weekly schedules*.

Property	Value	Description
Status [component]	text	Read-only field. Indicates the condition of the component at last polling. <ul style="list-style-type: none"> {ok} indicates that the component is polling successfully. {down} indicates that polling is unsuccessful, perhaps because of an incorrect property. {disabled} indicates that the Enable property is set to false. fault indicates another problem.
Fault Cause	text	Read-only field. Indicates why the network, component, or extension is in fault.
Default Output	true or false	Default output of 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's Facets property.
Cleanup Expired Events	true (default) or false	If true, calendar events of type Date and Date Range that have already occurred, and cannot occur again (as configured), are automatically deleted after they occur. This is recorded in the schedule log, and you no longer see them in the view. If false, all calendar events are retained, even if they cannot occur again (as configured).

Property	Value	Description
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: <code>ddd hh mm ss</code> , and range is from 1 day (minimum) up. If changed, a value less than the default is typically recommended—for example, 14 days.
Facets	<code>trueText</code> (default), <code>falseText</code>	<p>Configuring facets is critical in an <code>EnumSchedule</code>, and optional in a <code>BooleanSchedule</code> or <code>NumericSchedule</code>. Facets do not apply if a <code>StringSchedule</code>.</p> <p>Facets contain additional data applied to input and output values.</p> <ul style="list-style-type: none"> <code>trueText</code> is the text to display when output is true <code>falseText</code> is the text to display when output is false. <p>For example, you might want to set the facet <code>trueText</code> to display "ON" and the facet <code>falseText</code> to display "OFF."</p> <p>"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.</p>
Last Modified		Read only value. Timestamp of last configuration change.
Out Source	<code>Input</code> , <code>Special Event: <SpecialEventName></code> , <code>Week: <day_of_week></code> , (default)	Provides a string source description of the current output, as one of the four options shown left. For example, <code>Week: monday</code> or <code>Special Event: Christmas Break</code>
Out	<code>false {ok}</code> (default)	<p>Output value of schedule component. Output is <code>true</code> during any configured calendar day(s), otherwise it is <code>false</code></p> <p>Determined by the following, in highest-to-lowest priority:</p> <ul style="list-style-type: none"> 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	<code>- {null}</code> (default)	<p>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.</p> <p>Weekly schedules can be "chained" from <code>Out</code> slot to <code>In</code> slot. Chaining schedules is useful if only one of the schedules is effective at any period in time. In this case, the default output value of all schedule components (except last in chain) must be null.</p>

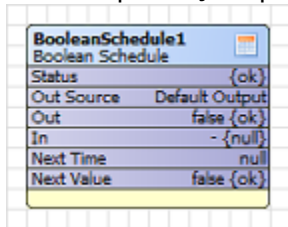
Property	Value	Description
Next Time	null (default)	Time of next scheduled output change for the component. If more than a year away, this value is null. Displays in Baja AbsTime format, for example: 03-Feb-05 5:00 PM. Typical application is for informational display. If needed, slots can be linked into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

Weekly schedule output processing

For any weekly schedule (BooleanSchedule, EnumSchedule, NumericSchedule, StringSchedule), 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

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

For more details on effective schedules, see "Effective Period" in the Properties tab section of the Scheduling Plugins chapter.

Out Source slot

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

- Input

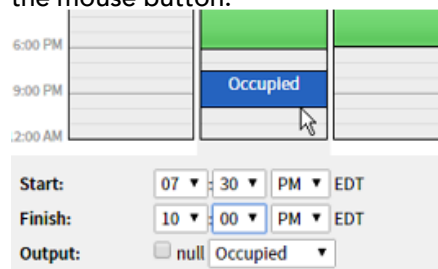
- **Special Event:** <SpecialEventName>
- **Week:**<day_of_week>
- **Default Output**

Examples: "Week: monday" or "Special Event: Christmas Break"

Weekly Schedule usage notes

Use the **Weekly Schedule** tab in the **Scheduler** view to enter regular "normal schedule events" that repeat from week to week, based on the day of the week and the time of day. Once entered, events appear as solid colored blocks, while unscheduled (default output) time is not assigned a color.

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



As needed, click again and drag on the event's top or bottom edge to change its start or finish time (in broad increments). Optionally, while the event is selected you can click on the **Start** and **Finish** dropdown lists located in the lower left corner to adjust start and finish times. You can also 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.

Additional details about the **Weekly Schedule** tab in the **Scheduler** view are described, in the following sections:

- [Event time tuning, page 33](#)
- [Output value, page 33](#)
- [Weekly Scheduler right-click menus, page 33](#)

Event time tuning

With an event selected, you can "fine tune" its start and finish time using the time using the hour, minute, and AM/PM option lists.

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, then a Tuesday event *starts* at midnight, the schedule output is continuous (providing both events have the same Output value).

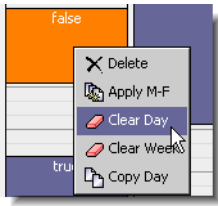
Output value

For any event, you can select a true or false value in the output field. This is the value that is routed to the access device at the scheduled times.

Scheduler right-click menus

Right-click in the weekly schedule area to invoke an event menu. If you have any event *selected*, this menu provides the most commands, as shown here.

Figure 2 Right-click menu with event selected



Event menu options may include the following:

- Delete — Deletes the selected event.
- Apply M-F — Copies all events in the selected day to Mon, Tue, Wed, Thu, and Fri (and *overwrites* any existing events on those days).
- Clear Day — Clears all events in the selected day.
- Clear Week — Clears all events in the entire weekly schedule.
- Copy Day — Copies all events in the selected day, to use with paste day option.
- Paste Day — Appears only if copy day option was used first. Copies all events into selected day.

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 located in the upper left corner of the view.

Event times (and values) entered for any special event apply to that schedule only. If the special event is a “reference” type, days of its occurrence are specified in the **CalendarSchedule** component that is referenced. This allows you to globally change the days that special events occur in weekly schedules, by editing one or more referenced calendar schedules.

In the time-of-day event definitions of special events, you can have them “intermingle” with regular weekly events, or completely override the weekly schedule. In addition, you may visually prioritize special events, via list order (the highest priority on top in the list). This allows any overlapping special events to occur in an priority-ordered fashion.

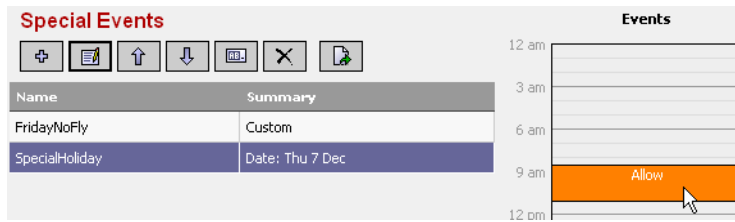
The **Special Events** tab is comprised of monthly calendars at the top, the Special Events table, and a 24-hour time pane. The Special Events table shows a listing of all existing schedules by name and summary. At the bottom of the table, controls are provided for adding, editing, or deleting schedules. In addition, Up and Down links are provided for sorting schedules in order of priority (in case of schedule overlaps). Schedules at the top have the highest priority.

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, page 33](#).

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 weekly schedule, configure a special event for the entire day.

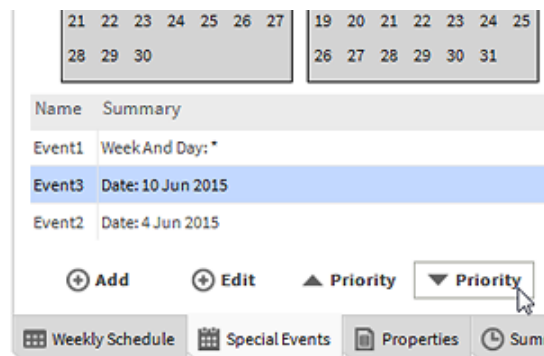
Special event priorities

All special events take priority over regular weekly events. Among special events, you define *relative priorities* by the order of listing in the Special Events table, as follows:

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

Change a special event’s priority by selecting it and using 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 which match the buttons located below the table.

Options may include the following:

- **Add** — Add a new special event
- **Edit** — Edit day(s) selection criteria (but *not changing* special event type).
- **Rename** — Rename selected special event.
- **Priority (up)** — Move special event up in priority list.
- **Priority (down)** — Move special event down in priority list (same as using **Priority** button).
- **Delete** — Removes selected special event from the schedule component (same as using **Delete** button).

About schedule properties

Each schedule component includes a set of properties and options that affect the way that the schedule works and provides information about current and projected schedule values. The **Default Output** and **In** properties are configurable and evaluated, based on priority, along with the **Weekly Schedule** when determining the current **Out** value. The **In** property value has highest priority, followed by a **Special Event**, **Weekly Schedule Event**, and then the default setting.

In Workbench you can access weekly schedule properties via the **Properties** tab in the **Scheduler** view. The **Properties** tab allows you to configure the **Effective Period** of the schedule, as well as the **Default Output** and whether or not to **Cleanup Special Events**.

In a browser connection, you can access schedule component properties by selecting the **Property Sheet** from the **View** dropdown list.

About calendar schedules

Calendar schedules define specific events. Typically, 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:

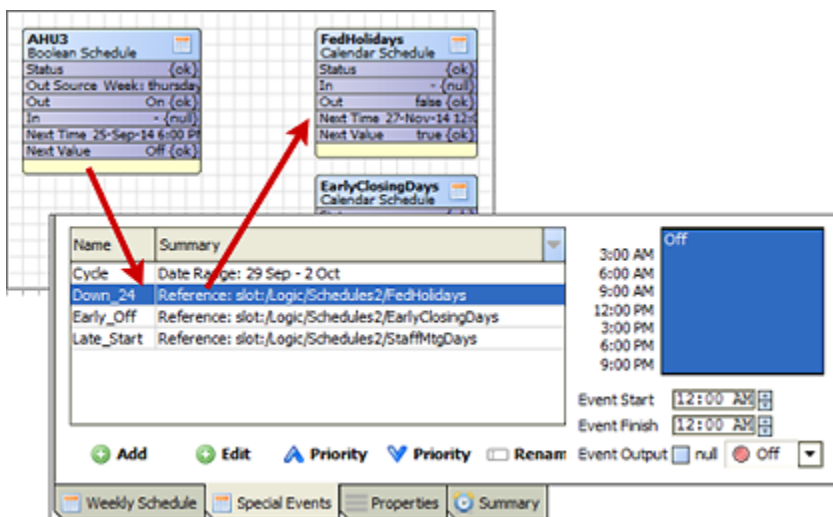
- Date
- Date Range
- Week and Day
- Custom

You can add as many day events as needed in the same **CalendarSchedule**. The following sections provide more details:

Calendar schedule usage

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 5 Example referenced CalendarSchedules



For example, the above figure shows a **BooleanSchedule** and a portion of its special events tab, listing four special events. Three of these are "Reference" types, meaning their calendar day(s) are defined

remotely in the configuration of the referenced calendar schedules. Although all components are shown here 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 containing a special event that references it.

schedule-CalendarSchedule

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

Properties

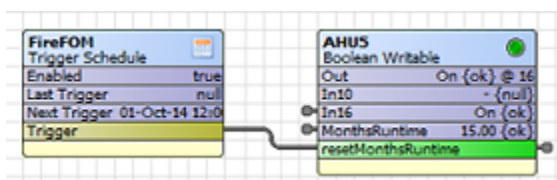
Property	Value	Description
Status [component]	text	Read-only field. Indicates the condition of the component at last polling. <ul style="list-style-type: none"> <code>{ok}</code> indicates that the component is polling successfully. <code>{down}</code> indicates that polling is unsuccessful, perhaps because of an incorrect property. <code>{disabled}</code> indicates that the <code>Enable</code> property is set to false. <code>fault</code> indicates another problem.
Fault Cause	text	Read-only field. Indicates why the network, component, or extension is in fault.
Cleanup Expired Events	true (default) or false	If true, calendar events of type <code>Date</code> and <code>Date Range</code> that have already occurred, and cannot occur again (as configured), are automatically deleted after they occur. This is recorded in the schedule log, and you no longer see them in the view. If false, all calendar events are retained, 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: <code>ddd hh mm ss</code> , 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	Configuring facets is critical in an <code>EnumSchedule</code> , and optional in a <code>BooleanSchedule</code> or <code>NumericSchedule</code> . Facets do not apply if a <code>StringSchedule</code> . Facets contain additional data applied to input and output values. <ul style="list-style-type: none"> <code>trueText</code> is the text to display when output is true <code>falseText</code> is the text to display when output is false. For example, you might want to set the facet <code>trueText</code> to display "ON" and the facet <code>falseText</code> 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.

Property	Value	Description
Last Modified		Read only value. Timestamp of last configuration change.
In	- {null} (default)	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. Weekly schedules can be "chained" from Out slot to In slot. Chaining schedules is useful if only one of the schedules is effective at any period in time. In this case, the default output value of all schedule components (except last in chain) must be null.
Out	false {ok} (default)	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: <ul style="list-style-type: none"> 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.
Next Time	null (default)	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 value. The next scheduled output value, at Next Time. Value is meaningless if Next Time is null.

About trigger schedules

Trigger schedules are special-purpose schedules, providing scheduling control for either linked actions or topics of other components.

Figure 6 Trigger schedule example



The above figure shows a simple example of a **TriggerSchedule** linked to an action slot of a **DiscreteTotalizerExt**, which was "composited" in the parent **BooleanWritable**. This schedule is configured to simply 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.

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
Status [component]	text	Read-only field. Indicates the condition of the component at last polling. <ul style="list-style-type: none"> {ok} indicates that the component is polling successfully. {down} indicates that polling is unsuccessful, perhaps because of an incorrect property. {disabled} indicates that the Enable property is set to false. fault indicates another problem.
Fault Cause	text	Read-only field. Indicates why the network, component, or extension is in fault.
Enabled	true (default) or false	Enables/disables firing of trigger outputs.
Next Trigger	null (default)	Next scheduled trigger firing time.
Last Trigger	null (default)	Read only value. Timestamp of the last firing of the trigger output.
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 value. The next scheduled output value, at Next Time . Value is meaningless if Next Time is null.
Last Modified		Read only value. Timestamp of last 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.

For more details see, *About Trigger schedules*.

About ScheduleSelector components

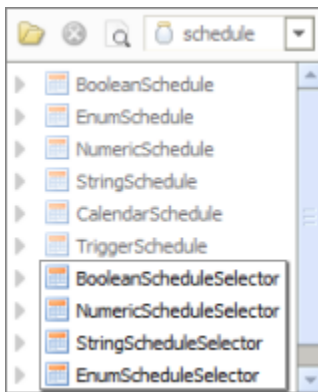
ScheduleSelector components provide an easy way for you to select a schedule to use for controlling a particular component. For example, you can use a **ScheduleSelector** component to link a pre-configured schedule to a component that is controlling equipment, a door schedule, or setpoint temperature. With a set of pre-configured schedules and the **ScheduleSelector** component, users can simply choose from a list of valid schedules when they want to setup or change the schedule of a device. You do not have to actually configure schedule properties.

Each **ScheduleSelector** component contains a property that allows you to target a single location or container that holds all the schedule components that you want 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**.

Types of ScheduleSelector components

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

Figure 7 ScheduleSelector components in schedule palette



ScheduleSelectors 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.

schedule-BooleanScheduleSelector

ScheduleSelector components provide an easy way to select a preconfigured 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.

See also,

- *About ScheduleSelector components*
- *ScheduleSelector configuration examples*

Properties

Property	Value	Description
Container	null (default)	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	0 (default)	<p>Dropdown displays an option list of all available schedule components of the appropriate data type located in the specified Container property.</p> <p>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.</p> <p>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.</p>
Facets		This property does not require configuration. Facets are automatically inherited from whatever schedule is currently selected.
In	- {null} (default)	<p>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.</p> <p>Weekly schedules can be "chained" from Out slot to In slot. Chaining schedules is useful if only one of the schedules is effective at any period in time. In this case, the default output value of all schedule components (except last in chain) must be null.</p>
Out	false {ok} (default)	<p>Output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false</p> <p>Determined by the following, in highest-to-lowest priority:</p> <ul style="list-style-type: none"> 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.

schedule-NumericScheduleSelector

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

See also:

- *About ScheduleSelector components*
- *ScheduleSelector configuration examples*

Property	Value	Description
Container	null (default)	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	0 (default)	<p>Dropdown displays an option list of all available schedule components of the appropriate data type located in the specified Container property.</p> <p>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.</p> <p>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.</p>
Facets		This property does not require configuration. Facets are automatically inherited from whatever schedule is currently selected.
In	- {null} (default)	<p>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.</p> <p>Weekly schedules can be "chained" from Out slot to In slot. Chaining schedules is useful if only one of the schedules is effective at any period in time. In this case, the default output value of all schedule components (except last in chain) must be null.</p>
Out	false {ok} (default)	<p>Output value of schedule component. Output is <code>true</code> during any configured calendar day(s), otherwise it is <code>false</code></p> <p>Determined by the following, in highest-to-lowest priority:</p> <ul style="list-style-type: none"> • 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.

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.

See also:

- *About ScheduleSelector components*
- *ScheduleSelector configuration examples*

Property	Value	Description
Container	null (default)	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	0 (default)	<p>Dropdown displays an option list of all available schedule components of the appropriate data type located in the specified Container property.</p> <p>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.</p> <p>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.</p>
Facets		This property does not require configuration. Facets are automatically inherited from whatever schedule is currently selected.
In	- {null} (default)	<p>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.</p> <p>Weekly schedules can be "chained" from Out slot to In slot. Chaining schedules is useful if only one of the schedules is effective at any period in time. In this case, the default output value of all schedule components (except last in chain) must be null.</p>
Out	false {ok} (default)	<p>Output value of schedule component. Output is true during any configured calendar day(s), otherwise it is false</p> <p>Determined by the following, in highest-to-lowest priority:</p> <ul style="list-style-type: none"> • 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.

schedule-EnumScheduleSelector

ScheduleSelector components provide an easy way for to select a preconfigured schedule of the proper data type for controlling a particular component. **EnumScheduleSelector** components only link

EnumSchedule components to a control component of the Enum data type. **EnumScheduleSelector** is available in the **schedule** palette.

See also:

- *About ScheduleSelector components*
- *ScheduleSelector configuration examples*

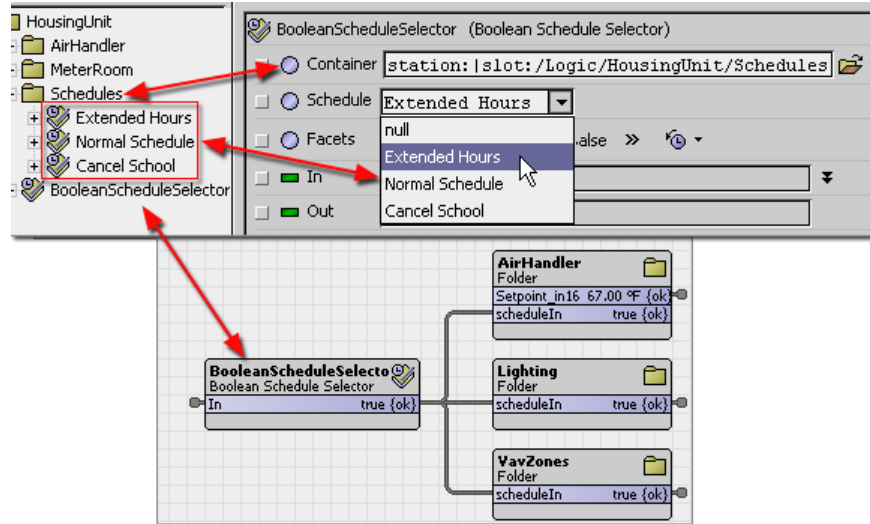
Property	Value	Description
Container	null (default)	Provides a text field that uses an ORD to specify the location of the available schedules. Use the Component chooser at the right side of the field to browse to and select the desired schedule container.
Schedule	0 (default)	<p>Dropdown displays an option list of all available schedule components of the appropriate data type located in the specified Container property.</p> <p>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.</p> <p>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.</p>
Facets		This property does not require configuration. Facets are automatically inherited from whatever schedule is currently selected.
In		When a ScheduleSelector component has been used to select a schedule, the schedule's output supplies this value (via the automatic link) to this slot. The default setting of this slot is the null value, produced when no schedule is selected.
Out		The output value (from the linked schedule) that is passed to whatever is controlled (linked to) this slot.

ScheduleSelector configuration examples

This section provides two examples of **ScheduleSelector** component configurations.

Configuring a BooleanScheduleSelector component

Figure 8 Property Sheet and Wire Sheet Views of a BooleanScheduleConnector Configuration



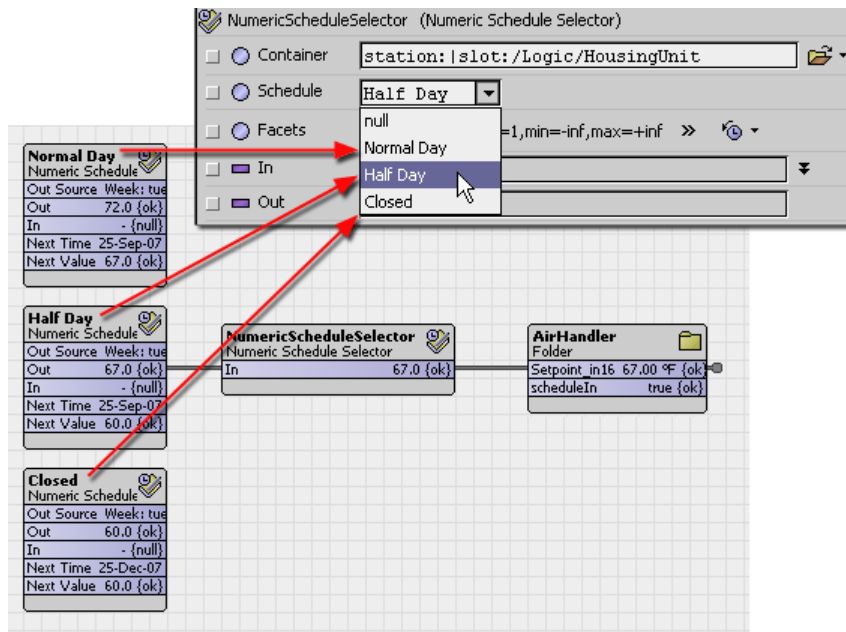
Notice the following points about this configuration:

- The **BooleanScheduleSelector** **Container** property is linked 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** is linked to three device inputs.

Configuring a NumericScheduleSelector component

The following image shows an example of a **NumericScheduleSelector**:

Figure 9 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 displays three numeric schedules available for selection. Notice that 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) is connected from the numeric schedule **Out** to the schedule selector **In**.
- The linked value (67.0 in this example) is passed from the **NumericScheduleSelector** component **Out** to the **AirHandler** device **Setpoint** input.

Chapter 4 Schedule plugins

Topics covered in this chapter

- ◆ Scheduler
- ◆ Current Day Summary
- ◆ Calendar Scheduler view
- ◆ TriggerScheduler

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

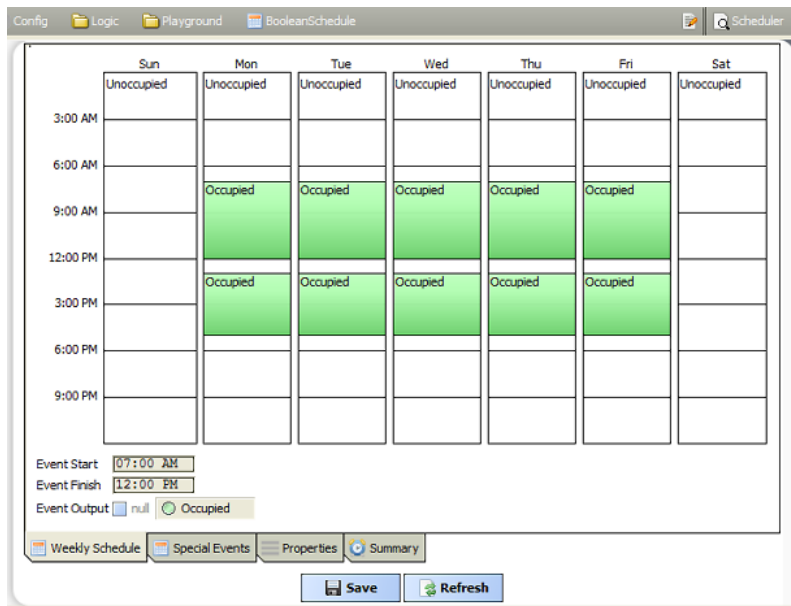
In general, plugins provide views of components and can be accessed in many ways. For example, double-click a component in the Nav tree to see its default view. Also, you can right-click on a component and select from its **Views** menu.

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

Scheduler

The **Scheduler** view allows you to view and edit weekly schedule components, namely **BooleanSchedule**, **EnumSchedule**, **NumericSchedule**, and **StringSchedule**. Each of these components uses the **Scheduler** view. The only difference between these types schedules is the output type.

Figure 10 Scheduler view for weekly schedule



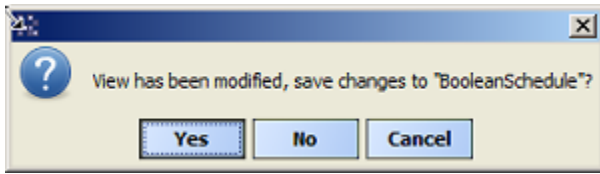
Double-clicking any weekly schedule component in Workbench displays its **Scheduler** view. The view features configuration tabs and at the bottom of the view are additional controls.

The **Refresh** button is always available. When you click **Refresh**, one of two things happens:

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 dialog as shown here:

Figure 11 Refresh confirmation if unsaved changes



The **Save** button is activated only if you have made unsaved changes in the component's scheduler. Clicking it downloads your changes to the schedule component's configuration. Immediately following, the **Save** button is deactivated again.

NOTE: For weekly schedules (tabbed **Scheduler** view), you typically save while working in each tab, even though any save applies to changes made on all tabs.

The four configuration tabs in the **Scheduler** view are described in the following sections.

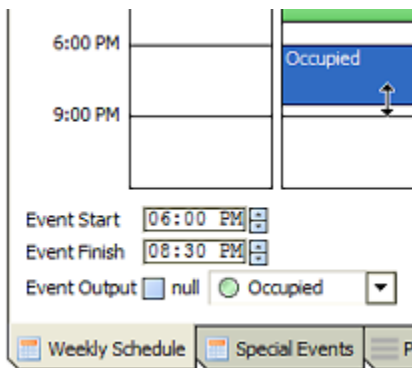
Weekly Schedule tab

Use this tab to enter regular, "normal schedule events" that repeat from week to week, based on the day of the week and the time of day. By default, any existing events appear as colored blocks, while unscheduled (default output) time appears in white.

NOTE: Events in BooleanSchedules and EnumSchedules have specified (and different) colors for each different state value, versus the default "greenish" color seen for all schedule events in prior releases.

You can use a click and drag technique to add a new event. Simply 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, dark blue) when you release the mouse button.

Figure 12 Click and drag to enter weekly events



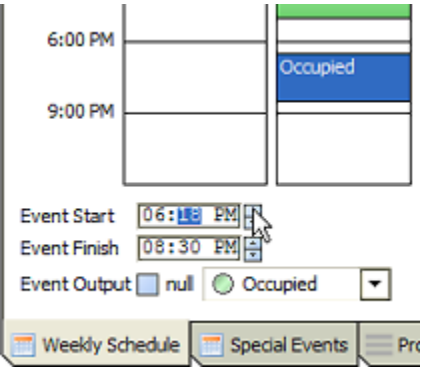
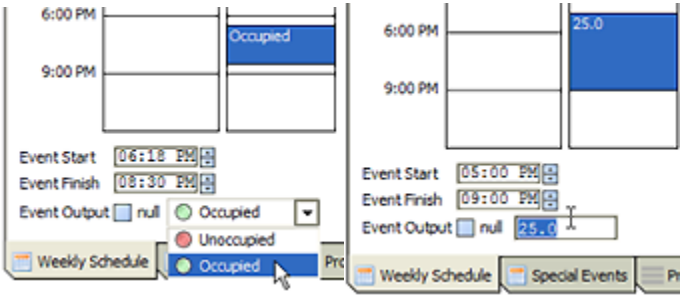
NOTE: If adding an **EnumSchedule**, you should define its range facet from the **Scheduler's Properties** tab before adding events. See "Facets" for more details.

- **Creating A Time Range** — Right-click and drag down in the slider column for desired the day.
- **Deleting A Time Range** — Click on a time range and press the delete or backspace key.
- **Editing A Time Range** — Time can be adjusted four ways: Dragging the top, bottom or center of a time range, or using the fine grained start and finish editors. Fine-tune a time range by clicking on a time range then adjust the times in the editors to the left.

NOTE: The finish time is exclusive; it is the first non-effective time after the effective period. The start time is inclusive.

Output is assigned to each time range and can be edited just below the time editors on the left.

As needed, click again and drag on the event's top or bottom to change its start or finish time (in broad increments). Additional details about the **Weekly Schedule** tab are as follows:

Additional controls	Description				
<p>Event time “tuning”</p>	<p>With any event selected, “fine tune” its Event Start and Event Finish time using the controls, selecting the hours portion or minutes portion. Or, click and type values in directly.</p> <p>Figure 13 Fine tune start and finish time</p>  <p>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, then a Tuesday event starts at midnight, the schedule output is continuous (providing both events have the same Output value).</p>				
<p>Output value</p>	<p>For any event, you can select the “null” checkbox (the schedule’s calculated value is null for that event). However, you typically select or type a value instead, as follows:</p> <p>Figure 14 Select (Boolean, Enum) or type (Numeric, String) output value</p>  <ul style="list-style-type: none"> • Select value or Boolean or Enum schedules <p>NOTE: For an EnumSchedule, first specify its facets (on Properties tab) before entering values. This allows selection of possible values. <ul style="list-style-type: none"> • Enter value for Numeric or String schedules <p>NOTE: If a NumericSchedule or StringSchedule, you type the value in the output field, then press Enter to register it in the event block,</p> </p>				
<p>Right-click menus</p>	<p>Right-click on a day for an event menu. Event menu options are straightforward, and may include the following:</p> <table border="1" data-bbox="479 1759 1416 1856"> <thead> <tr> <th data-bbox="479 1759 711 1808">Option</th> <th data-bbox="711 1759 1416 1808">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="479 1808 711 1856">Delete Event</td> <td data-bbox="711 1808 1416 1856">Deletes the selected event.</td> </tr> </tbody> </table>	Option	Description	Delete Event	Deletes the selected event.
Option	Description				
Delete Event	Deletes the selected event.				

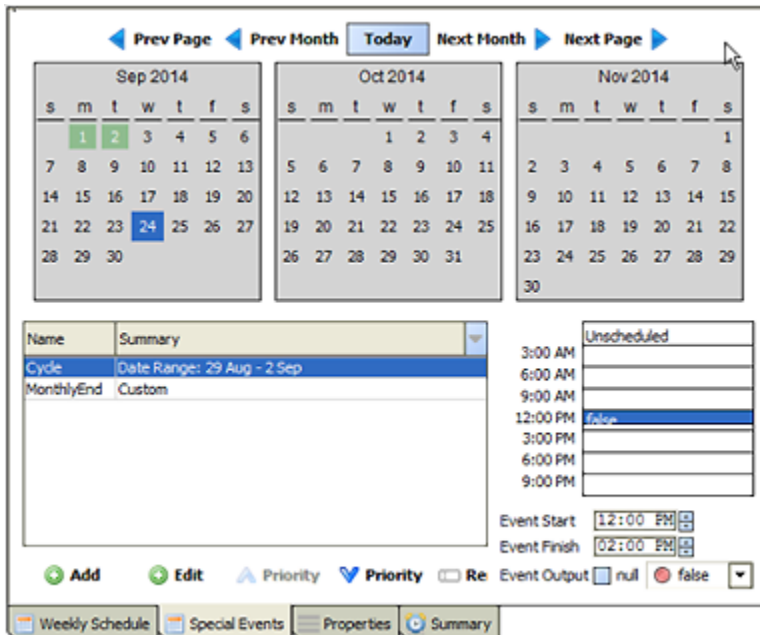
Additional controls	Description	
	Option	Description
	Paste Day	Appears only if copy day option was used first. Copies all events into selected day.
	All Day Event	Makes currently selected (or last entered) event extend to entire day.
	Apply M-F	Copies all events in the selected day to Mon, Tue, Wed, Thu, and Fri (and overwrites any existing events on those days).
	Copy Day	Copies all events in the selected day, to use with paste day option.
	Clear Day	Clears all events in the selected day.
	Clear Week	Clears all events in the entire weekly schedule.

Special events tab

Use this tab to enter all exceptions to the normal weekly schedule, broadly called "special events." Special events override (and intermingle with) events in the normal weekly schedule. Use the controls at the bottom of the tab to Add, Edit, Prioritize, Rename, and Delete.

When you first access the **Special Events** tab, the current day is highlighted in the left-most calendar month at the top of the view. As needed, click on **Next Month** and **Prev Month**, or **Next Page** and **Prev Page** to traverse the calendar ahead or back in time.

Figure 15 Special Events tab in weekly Scheduler



As shown above, existing special events (if any) are listed in the table by name and summary. When you select a special event, its day(s) of occurrence are highlighted in the monthly calendars at the top of the view, and its associated event actions are displayed in the right-side column.

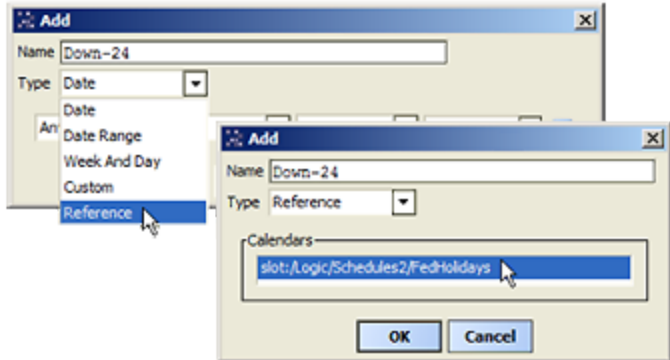
Additional details about the **Special Events** tab are as follows:

Adding special events

When adding special events, an **Add** dialog displays with the following options:

- **Name** — Type a unique, identifiable name for this event. For example “Christmas Day” or “Half-Day”, etc.
NOTE: By default, the name is `Event`. You can change this value at a later time.
- **Type** — Select the desired Type option, and enter specific day criteria (according to type).

Type	Description
Date	Default selection. Select various combinations of weekday, numerical date, month or month combinations, and year.
Date Range	Select start and end range, using for each a combination of day, month, year.
Week and Day	Select combination of day of week, week in month, month.
Custom	Select various combinations of day, month, weekdays, and year.
Reference	By reference to a specific CalendarSchedule component in the station. Displays a second Add dialog listing all CalendarSchedules available in the station, as shown here.



After you have a name and type selected (and defined as needed), clicking OK adds it to this schedule’s special events table. It remains selected for further editing, except for Type.

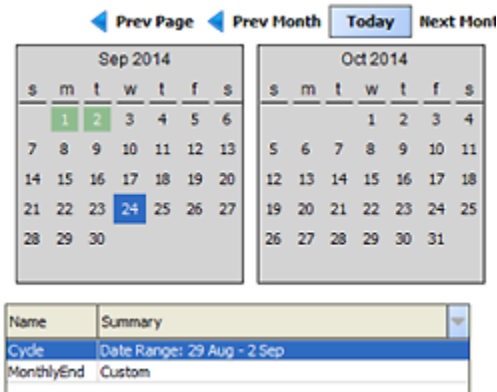
Event times

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.

When you select a special event in the table, if it occurs in any currently displayed month, its associated day or days are highlighted as shown here.

Figure 16 Special event highlights in calendar block



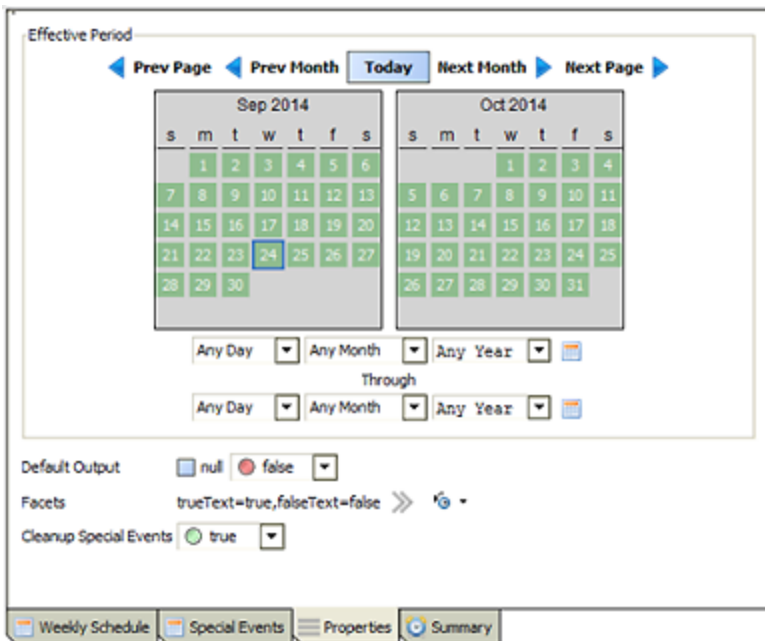
NOTE: A special event must have at least one defined event action to be highlighted in a calendar.

Return to the current calendar month and day by clicking the **Today** button. The events table provides several export options. Clicking the selection icon at the upper-right corner of the events table displays an **Export** dialog. The **Export** dialog allows you to select the desired file format (PDF, CSV, HTML, etc.) and whether to view the exported data internally in Workbench, or with an external application, or to save the data to your file system. Additionally, for PDF exports you can setup the desired page size and orientation.

Properties tab

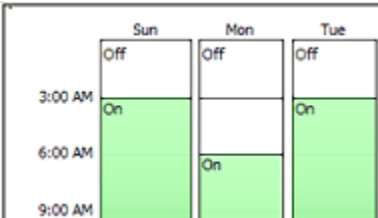
The **Properties** tab is available in the **Scheduler** view when using Workbench. This tab is where you can specify the following properties:

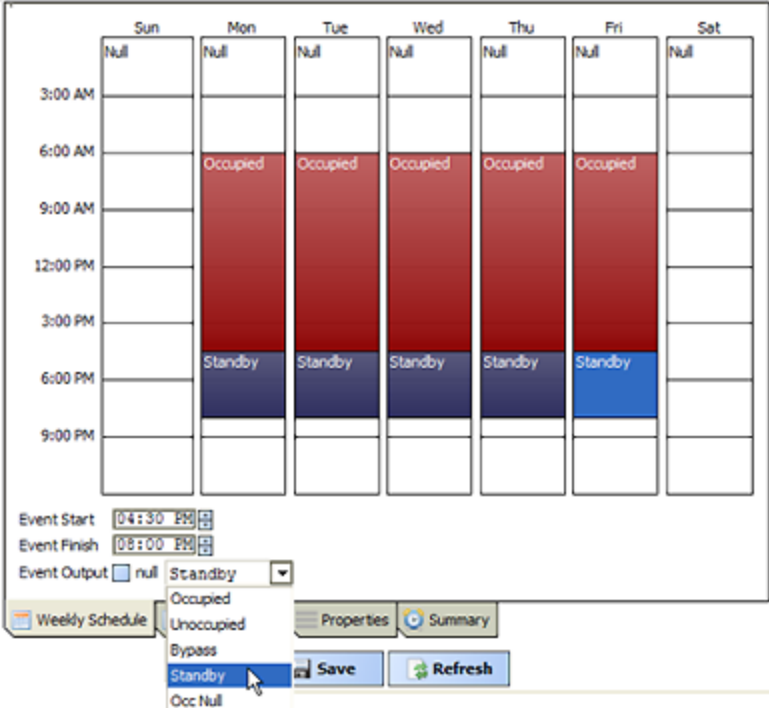
Figure 17 Properties tab in weekly Scheduler



NOTE:

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

Properties tab controls	Description
<p>Effective Period</p>	<p>By default, a weekly schedule added from the schedule palette is always effective. Whenever a schedule component is not effective, its output (Out slot) goes to its default output value, regardless of its weekly schedule or any special events.</p> <p>In most cases, you leave weekly schedules as always effective. However, if you have an application for a schedule effective only at certain times, use the “start” through “end” range fields to limit the effective period. When you Save the changes, only effective days in the calendar months are shown highlighted green.</p>
<p>Default Output</p>	<p>Whenever a schedule event (special or weekly) is not defined, the schedule component’s output (Out slot) is this value. The white area in listed events indicates where the default value is used and displays the current default value (Off, as shown in the image below). The default output value is also used whenever the schedule is not effective.</p> <p>Figure 18</p>  <p>“null” is an available choice—depending on control logic, this may be a valid choice.</p> <p>As copied from the schedule palette, the default “Default Output” varies by schedule type, as follows:</p> <ul style="list-style-type: none"> • BooleanSchedule — false • EnumSchedule — null • NumericSchedule — null • StringSchedule —null

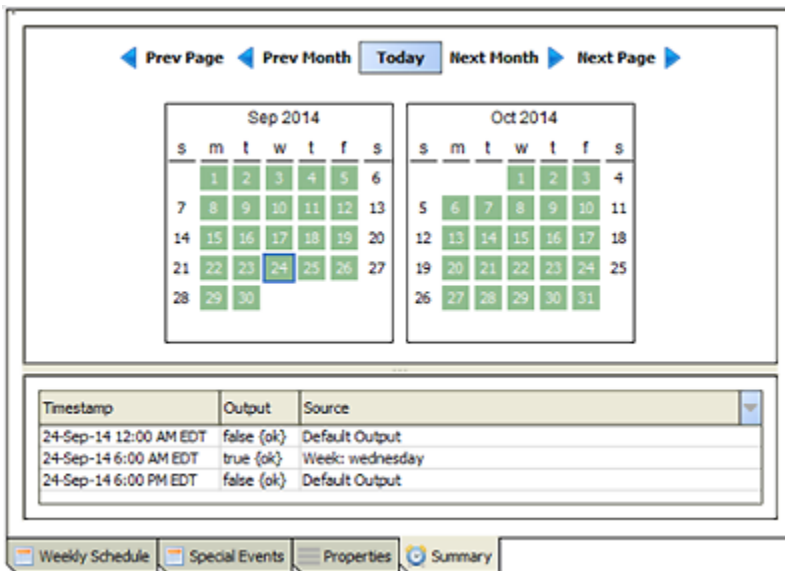
Properties tab controls	Description
<p>Facets</p>	<p>The schedule component's facets determine how its output value is formatted for display. For example, instead of "true" and "false" for a <code>BooleanSchedule</code>, you may need "On" and "Off" instead. Assigned facets appear in scheduler views when adding events, displaying summary data, and so on. For complete details, see "About point facets".</p> <p>By default, facets for schedule components as copied from the schedule palette are as follows:</p> <ul style="list-style-type: none"> • <code>BooleanSchedule</code> — trueText: true, falseText: false • <code>EnumSchedule</code> — range: (not defined) • <code>NumericSchedule</code> — units: (null), precision: 1 • <code>StringSchedule</code> — (not applicable) <p>NOTE: Facets are especially important for <code>EnumSchedules</code>. You need to define "range" facets before you add weekly schedule events (in order to pick an event's enumerated value). Range facets should match those used in any controlled (output-linked) <code>EnumWritables</code>. For related details, see "Facets importance for Enum points" in the <i>User Guide</i>.</p> <p>In the case of <code>StringSchedules</code> (as for all string-type components) facets have no application.</p> <p>The figure X shows output selections for an <code>EnumSchedule</code> with its range facet defined as "lonworks:LonOccupancyEnum," one of the available frozen facets.</p> <p>Figure 19</p>  <p>Configured events shown in the above figure reflect a few "default colors" available for <code>EnumSchedules</code>.</p>
<p>Cleanup Special Events</p>	<p>This property is either true (default) or false.</p>

Properties tab controls	Description
	<ul style="list-style-type: none"> If true, "one-time" special events that have occurred (and will not be effective again) are automatically deleted. When a special event is deleted, a message is sent to the schedule log, and that special event no longer appears in the Special Events tab. If false, "one-time" special events are retained, even though they will not occur again.

Summary tab

The **Summary** tab is available in the **Scheduler** view when using Workbench. The 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 reievw its schedule events. If adjustments are necessary, click the other tabs as needed, to make changes.

Figure 20 Summary tab shows all events for any selected day

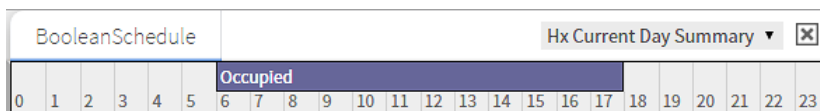


The events table provides several export options. Clicking the selection icon at the upper-right corner of the events table displays an **Export** dialog. The **Export** dialog allows you to select the desired file format (PDF, CSV, HTML, etc.) and whether to view the exported data internally in Workbench, or with an external application, or to save the data to your file system. Additionally, for PDF exports you can setup the desired page size and orientation.

Current Day Summary

The **Current Day Summary** 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.

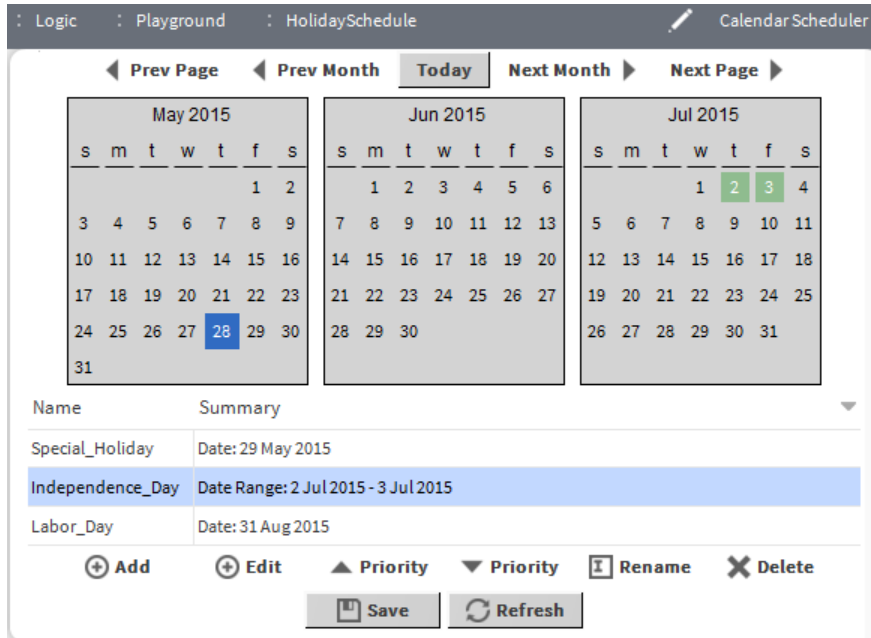
Figure 21 Current Day Summary view (browser connection)



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. Double-click a `CalendarSchedule` component to open the view.

Figure 22 Workbench Calendar Scheduler view



If using Workbench, the **Calendar Scheduler** view displays one or more calendar months, standard controls and a table of existing events in the lower part, as shown in the above image.

When you first access the **Calendar Scheduler**, the current day is highlighted in the left-most calendar month at the top of the view. As needed, click on **Next Month** and **Prev Month**, or **Next Page** and **Prev Page** to traverse the calendar ahead or back in time. Return to the current calendar month and day by clicking the **Today** button.

Existing calendar events (if any) are listed in the table by name and summary. 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.

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.

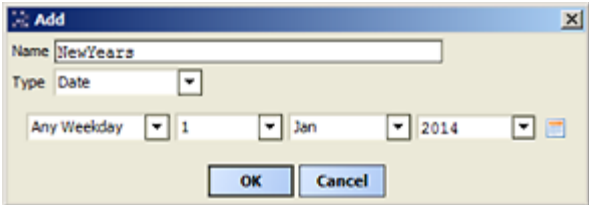
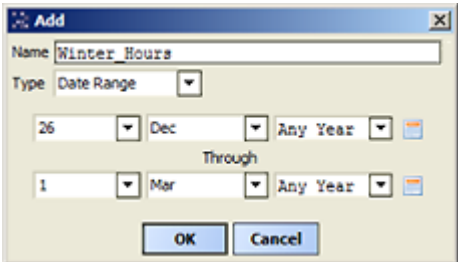
Use the **Add** button to add a calendar event (day or range of days). Each entry requires a unique name, a date type, and other specific calendar data criteria. Other controls in the view allow you to edit priority, rename, or delete calendar date entries.

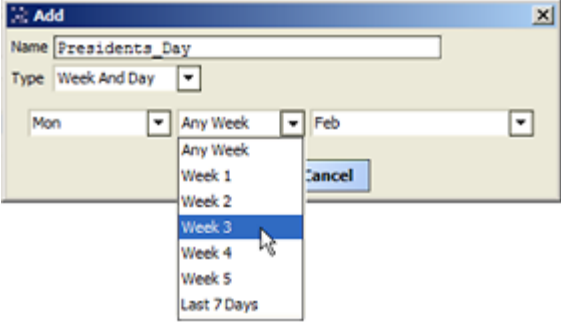
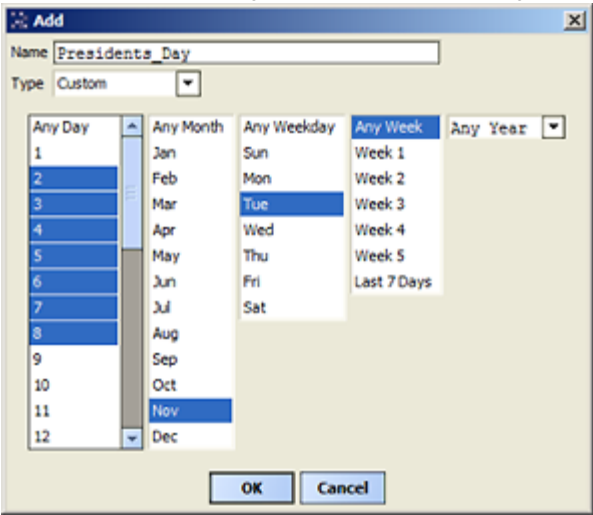
NOTE: Priority selections (via right-click menu or control buttons) affect only the list order of events displayed in a `CalendarSchedule`—true priority applies only to special events (in weekly schedules).

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. Select a row in the table to perform an action using an available control or double click on a row to edit it from the associated edit view.

Calendar event selections

When adding an event in a `CalendarSchedule` the following Type options are available:

Type	Description
<p>Date</p>	<p>This selection has 4 criteria, weekday, day-of-month, month-of-year, and year. You can make only one selection in each criteria (weekday, day-of-month, month, year). Each criteria offers an <i>Any</i> selection, in addition to a specific selection.</p>  <p>The result of selections is by "ANDing" all criteria. For example, if you select weekday of Tuesday, day of month as 5, and remaining criteria "any," the event is specified only on Tuesday, the fifth of any month in any year. If a month does not have Tuesday the fifth, then there is no event that month.</p>
<p>Date range</p>	<p>This selection has a start range and end range, each with 3 criteria, day-of-month, month-of-year, and year. You can make only one selection in each criteria (day-of-month, month, year). Each criteria offers an <i>Any</i> selection, in addition to a specific selection.</p>  <p>In each date range, result is from "ANDing" the criteria. 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.</p>

Type	Description
<p>Week and day</p>	<p>This selection has three criteria: weekday, week-in-month, and month-of-year. You can make only one selection in each criteria (weekday, week-in-month, month). Each criteria offers an <i>Any</i> selection, in addition to a specific selection.</p>  <p>The following criteria offer additional selections, as follows:</p> <ul style="list-style-type: none"> • week-in-month: <ul style="list-style-type: none"> – Last 7 days • month-of-year: <ul style="list-style-type: none"> – Jan-Mar-May-Jul-Sep-Nov – Feb-Apr-Jun-Aug-Oct-Dec <p>The result is from “ANDing” the criteria. For example, as shown in the above image, if selections are for weekday as Monday, the month as February, and the week as 3, the event occurs only on the third Monday in February.</p>
<p>Custom</p>	<p>This selection has 5 criteria: day-of-month, month-of-year, weekday, week-in-month, and year.</p>  <p>Unlike with other calendar types, you can make multiple selections within each criteria (except if you select “any,” which allows only that selection). To select multiples, first select something other than “Any,” then hold down the Ctrl or Shift key while you select more values.</p> <p>Each criteria offers an <i>Any</i> selection, in addition to a specific selection. In addition, the following criteria offer additional selections, as follows:</p> <ul style="list-style-type: none"> • day-of-month <ul style="list-style-type: none"> – Last Day

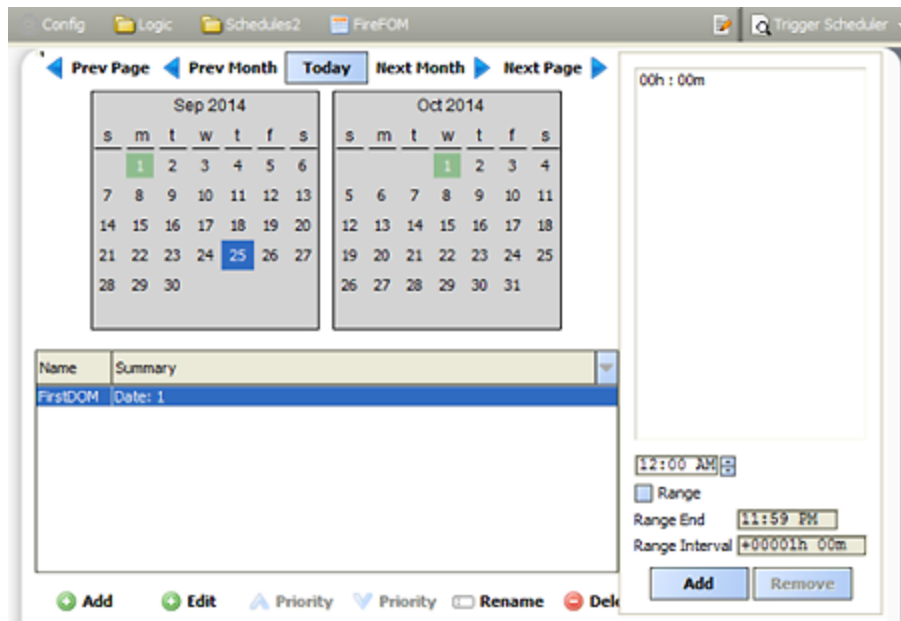
Type	Description
	<ul style="list-style-type: none"> - Last 7 Days • week-in-month - Last 7 Days <p>Within any criteria, selections are "OR'ed." The overall result is from "AND'ing" all criteria. For example, the above image shows a custom selection for U.S. General Election Day, which must be configured as the "first Tuesday after the first Monday in November."</p>

TriggerScheduler

This is the default view of **TriggerSchedule** components. You can use the view to configure schedule events in a **TriggerSchedule**. Trigger schedules are defined by combination of calendar day (or days) and trigger event time(s), including time ranges, each with a repeating interval.

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

Figure 23 Trigger Scheduler view



The Trigger Scheduler has two sides:

- **Calendar** (left) side — where you add events. It operates like the Calendar Scheduler view.
- **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.

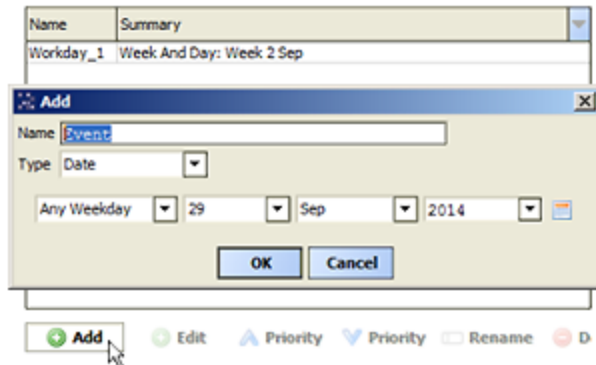
The events table provides several export options. Clicking the selection icon at the upper-right corner of the events table displays an **Export** dialog. The **Export** dialog allows you to select the desired file format (PDF, CSV, HTML, etc.) and whether to view the exported data internally in Workbench, or with an external application, or to save the data to your file system. Additionally, for PDF exports you can setup the desired page size and orientation.

When you first access the **Trigger Scheduler**, the current day is highlighted in the left-most calendar month at the top of the view. As needed, click on **Next Month** and **Prev Month**, or **Next Page** and **Prev Page** to traverse the calendar ahead or back in time.

Return to the current calendar month and day by clicking the **Today** button.

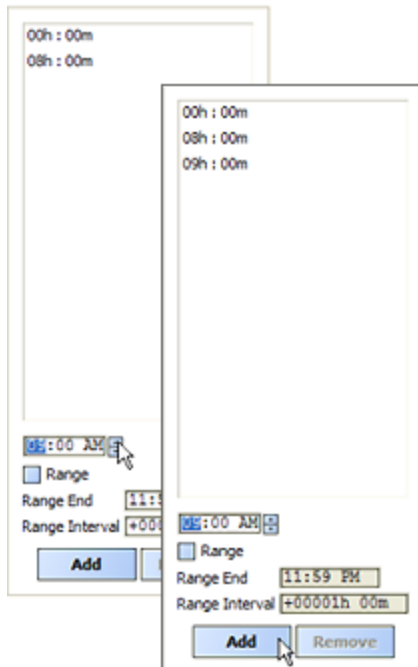
Use the bottom-left **Add** button to add a scheduled day (or range of days). Each entry requires a unique name, a date type, and other specific calendar data criteria. Other controls in the **TriggerSchedule** allow you to edit priority, rename, or delete calendar day entries.

Figure 24 Trigger Event Add dialog



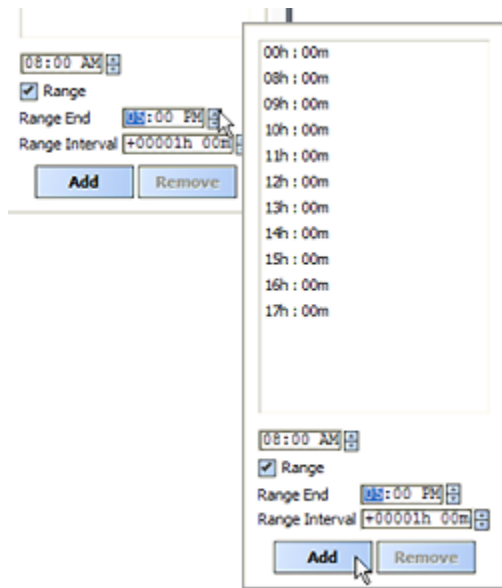
Use the bottom-right **Add** button to add trigger events to any selected scheduled day(s). Use the time selector to specify the individual trigger event. Use the **Range** controls to define a time range with repeating interval for trigger events to occur.

Figure 25 TriggerSchedule time picker controls



Controls in the Time picker side allow you to enter trigger event times. You can enter the desired time in the **hour:minute** editor or use the **Range** option to add multiple triggers that occur at a repeating interval. Selecting the **Range** checkbox enables the **Range End** and **Range Interval** fields for entering values as shown below.

Figure 26



When entering a trigger range, note that the top (hour:minute) editor acts as the first (or **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. To delete a trigger time, click to select, then click the **Remove** button. To select multiple trigger times, hold down the **Ctrl** or **Shift** key while you select.

For more details, see "*About trigger schedules*"

Index

A

Adding a schedule 11

C

Calendar schedule
 configuring 16
 usage 36
Calendar schedules
 configuring 16
 deleting calendar event 17
 configuring weekly (normal) schedule 12
 copying a pre-configured schedule 11

E

exports and imports
 exporting to AX 10

L

Legal notices 2

M

master/slave schedules 10

P

Plugins 47

R

Reviewing weekly schedule configuration 14

S

Schedule component views 7, 60
Schedule special events 9
Scheduler view
 Properties tab 53
 Summary tab 56
 Weekly Schedule tab 48
Schedules
 trigger schedules 38
Scheduling 36
 about configuring 12
 adding a schedule 11
 calendar day selections 57
 Calendar Scheduler view 57
 calendar schedules 36

Calendar schedules
 deleting calendar event 17
components
 BooleanSchedule 24
 BooleanScheduleSelector 40
 CalendarSchedule 37
 EnumSchedule 26
 EnumScheduleSelector 44
 NumericSchedule 28
 NumericScheduleSelector 41
 StringSchedule 30
 StringScheduleSelector 42
 TriggerSchedule 39
configuring Calendar schedules 16
Configuring properties of a weekly schedule 12
Configuring special events of a weekly
 schedule 13
configuring Trigger schedules 17
Importing
 About 21
 importing a schedule or calendar 21
Linking
 a trigger schedule 20–21
 a weekly schedule 19
Linking trigger schedules 20
linking weekly schedules 19
Schedule special events
 permission change 14
Scheduler view 47
 Properties tab 53
 Weekly Schedule tab 48
ScheduleSelector components
 About 39
 configuration examples 44
 Types of 40
trigger schedule
 adding a trigger event 17
Trigger schedule
 adding trigger event times 18
Trigger Schedule view 60
TriggerSchedule 39
Views
 Current Day Summary view 56
 Weekly schedule
 Reviewing configuration of 14
Scheduling component views 57
Scheduling model
 Schedule component categories 7
 Schedule component links 8
 Schedule component views 47
Scheduling model overview 7

T

Trigger schedule

adding trigger event..... 17
adding trigger event times 18

W

weekly schedule
 types of components..... 23
Weekly schedule
 configure weekly (normal) schedule..... 12
 Configuring properties of 12
 Configuring special events of..... 13
 output processing 32
Weekly schedules
 linking..... 19