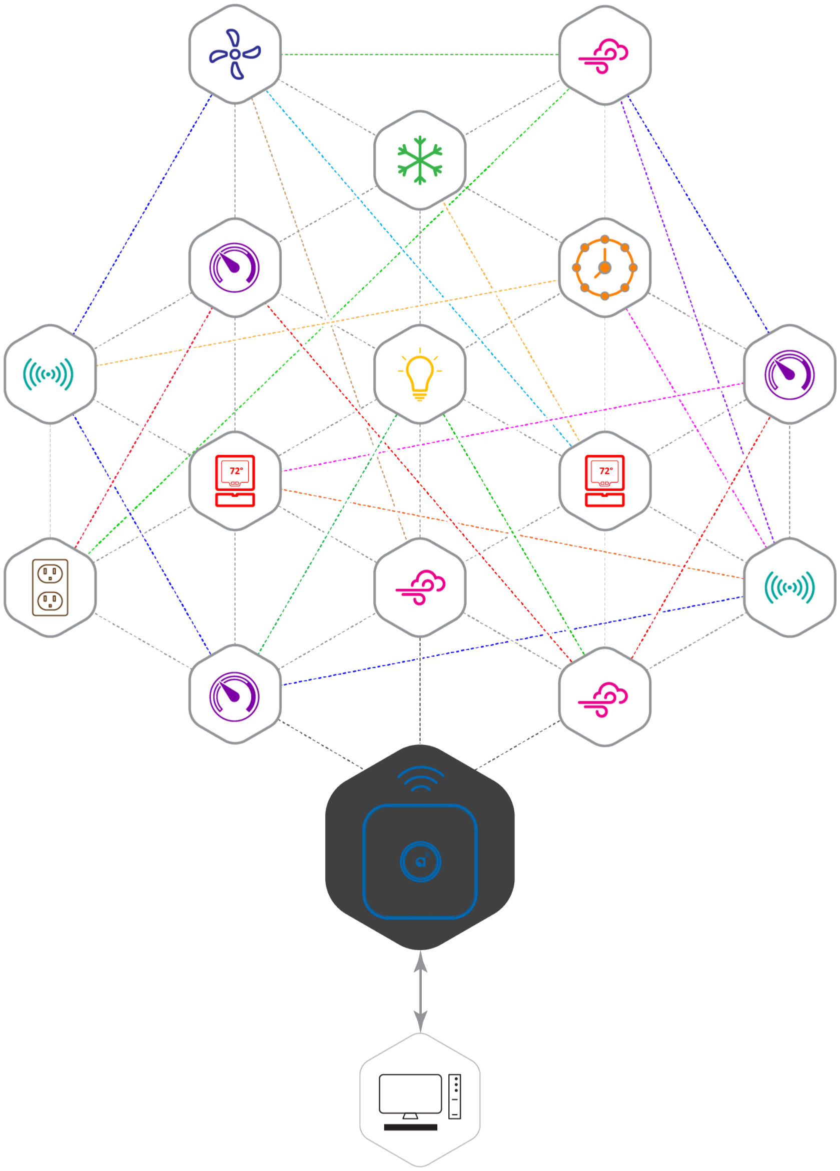
DOC #: 80305, REV: 03

User Guide

EnergyCenter® PRO

Custom Scripting



Autani LLC
7001 Columbia Gateway Drive, Suite 210, Columbia, MD 21046 USA, (443) 320-2233 | (240) 755-0092 (fax) | www.autani.com.

Table of Contents

1. **Introduction** 3

2. **Using EnergyCenter®** 3

   2.1. Working with Event Rules 3

   2.2. Creating a Custom Script 4

3. **Scripting Language** 5

   3.1. Basics 5

   3.2. Rule Types 8

   3.3. Data Types 8

   3.4. Dropdowns and Selection Menus 9

   3.5. Variables 9

   3.6. Group Operations 9

   3.7. Script Errors 10

   3.8. Reference 11

   3.8.1. Rule Type 11

   3.8.2. Logic 11

   3.8.3. Math 13

   3.8.4. Text 14

   3.8.5. Lists 15

   3.8.6. Monitoring, General 16

   3.8.7. Monitoring, HVAC 16

   3.8.8. Monitoring, Lights 17

   3.8.9. Monitoring, Sensors 17

   3.8.10. Monitoring, Plugs 18

   3.8.11. Monitoring, Meter 18

   3.8.12. Monitoring, Load 19

   3.8.13. Control, HVAC 20

   3.8.14. Control, Lights 21

   3.8.15. Control, Plugs 22

   3.8.16. Control, Load 22

   3.8.17. Alerts 23

   3.8.18. Overrides 23

   3.8.19. Groups 24

   3.8.20. Devices 24

4. **Examples** 25

List of Tables

Table 1 : Monitoring, General 16

Table 2 : Monitoring, HVAC 16

Table 3 : Monitoring, HVAC 17

Table 4 : Monitoring, Sensors 17

Table 5 : Monitoring, Plugs 18

Table 6 : Monitoring, Meter 18

Table 7 : Monitoring, Meter 19

Table 8 : Control, HVAC 20

Table 9 : Control, Lights 21

Table 10 : Control, Plugs 22

Table 11 : Control, Load 22

1. Introduction

PRO rules allow a user to provide custom programming (or *scripting*) to extend the built-in capabilities of a Manager appliance. Rules are assembled by linking puzzle-piece-like programming blocks together in a graphical editor. This simplifies the programming process as puzzle pieces are selected from a built-in menu and will only link with acceptable companion pieces. There is no need to connect wires, write code, or learn a programming language.

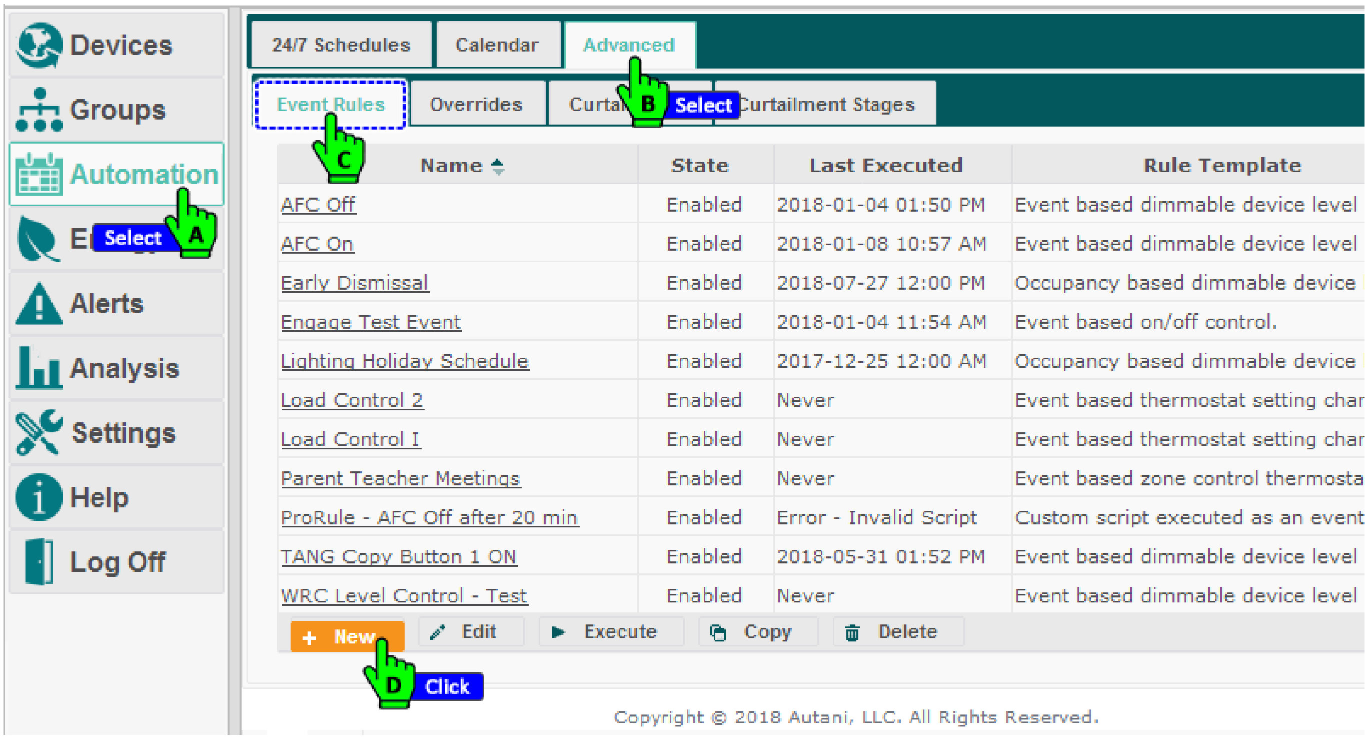
The goal is to allow any user to be able to quickly get started creating and maintaining custom rules, while providing the flexibility to allow power users to create advanced rules.

2. Using EnergyCenter®

2.1. Working with Event Rules

PRO rules are a special type of event rules and can be created using the New Event Rule GUI.

1. Select **Automation** > **Advanced** > **Event Rules**.



Name	State	Last Executed	Rule Template
AFC Off	Enabled	2018-01-04 01:50 PM	Event based dimmable device level
AFC On	Enabled	2018-01-08 10:57 AM	Event based dimmable device level
Early Dismissal	Enabled	2018-07-27 12:00 PM	Occupancy based dimmable device
Engage Test Event	Enabled	2018-01-04 11:54 AM	Event based on/off control.
Lighting Holiday Schedule	Enabled	2017-12-25 12:00 AM	Occupancy based dimmable device
Load Control 2	Enabled	Never	Event based thermostat setting char
Load Control I	Enabled	Never	Event based thermostat setting char
Parent Teacher Meetings	Enabled	Never	Event based zone control thermosta
ProRule - AFC Off after 20 min	Enabled	Error - Invalid Script	Custom script executed as an event
TANG Copy Button 1 ON	Enabled	2018-05-31 01:52 PM	Event based dimmable device level
WRC Level Control - Test	Enabled	Never	Event based dimmable device level

Copyright © 2018 Autani, LLC. All Rights Reserved.

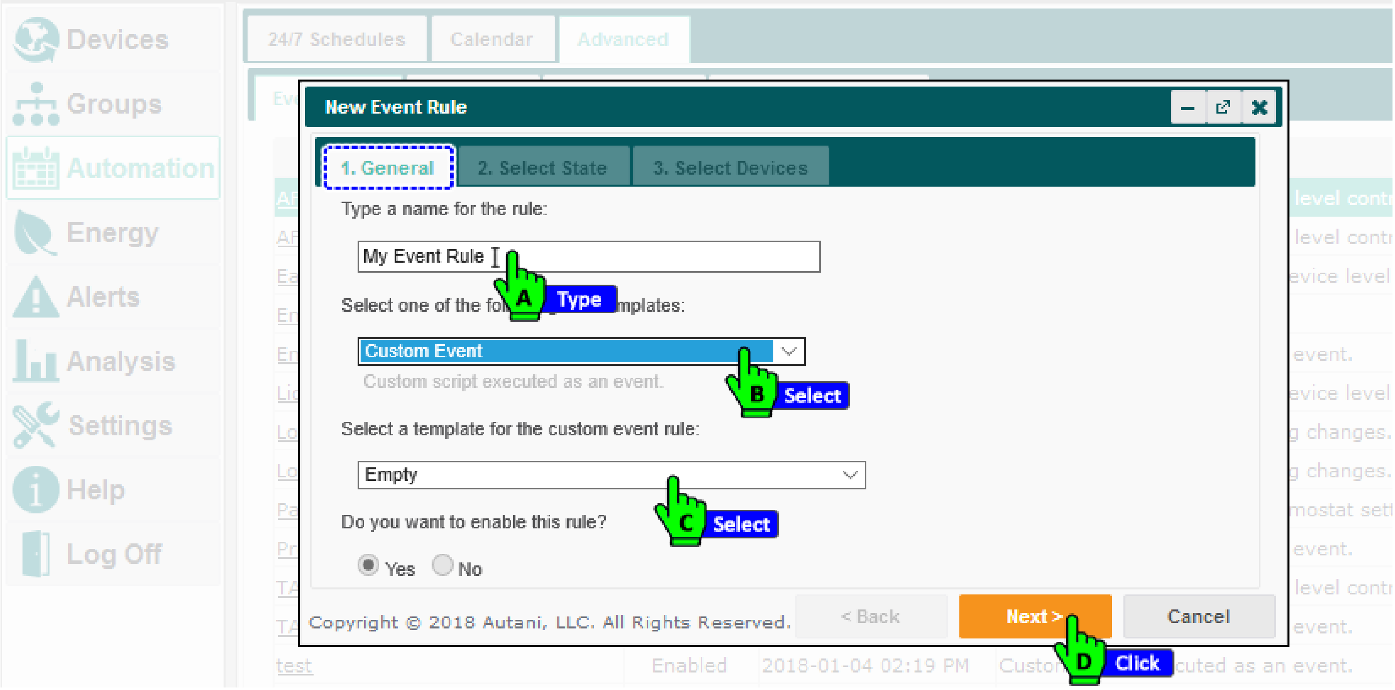
New event rules can also be created as part of the process of creating a new override or scheduling an override on the calendar.

An event rule is not executed until the override that contains the rule is activated. This is typically done in one of two ways:

- **Standing overrides** – Overrides that are always in effect. A PRO rule that is part of a standing override will always be executed. For example, this would be useful for a rule monitoring the communication status of a device.
- **Calendar-based overrides** – Overrides that are scheduled to execute on specific days/times. For example, a holiday override might be created to curtail energy usage when a building is unoccupied for an extended period of time.

2.2. Creating a Custom Script

2. From the **New Event Rule** GUI, a new custom scripting can be created by selecting **Custom Event** as the rule type. A pre-existing template can also be selected to make it easier to create commonly-used scripts.



New Event Rule

Type a name for the rule: My Event Rule

Select one of the fo[…]mplates: Custom Event

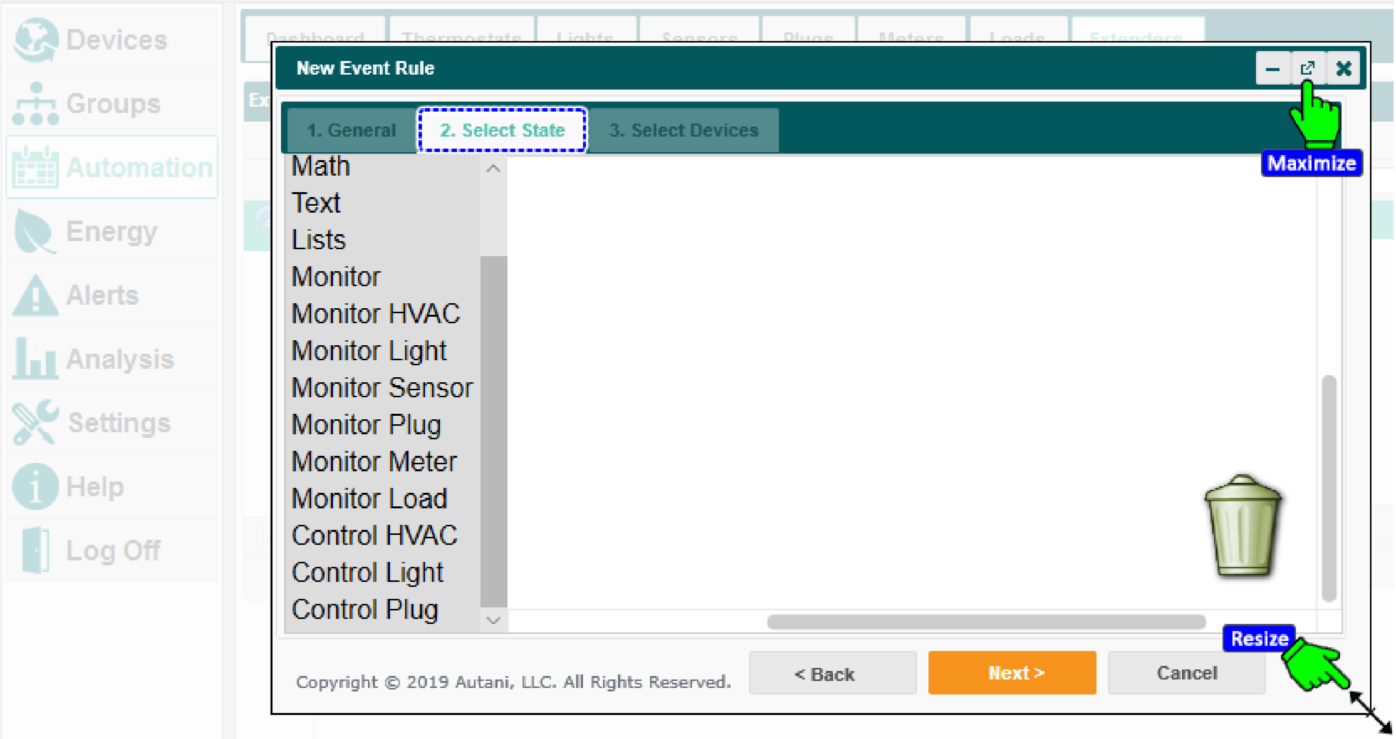
Custom script executed as an event.

Select a template for the custom event rule: Empty

Do you want to enable this rule? ● Yes ○ No

Copyright © 2018 Autani, LLC. All Rights Reserved.

3. The size of the **New Event Rule** dialog can be increased to increase the size of the PRO rule workspace, which can make it easier to work with more complex rules. There is also a maximize button in the upper right-hand corner to make the dialog as large as possible.



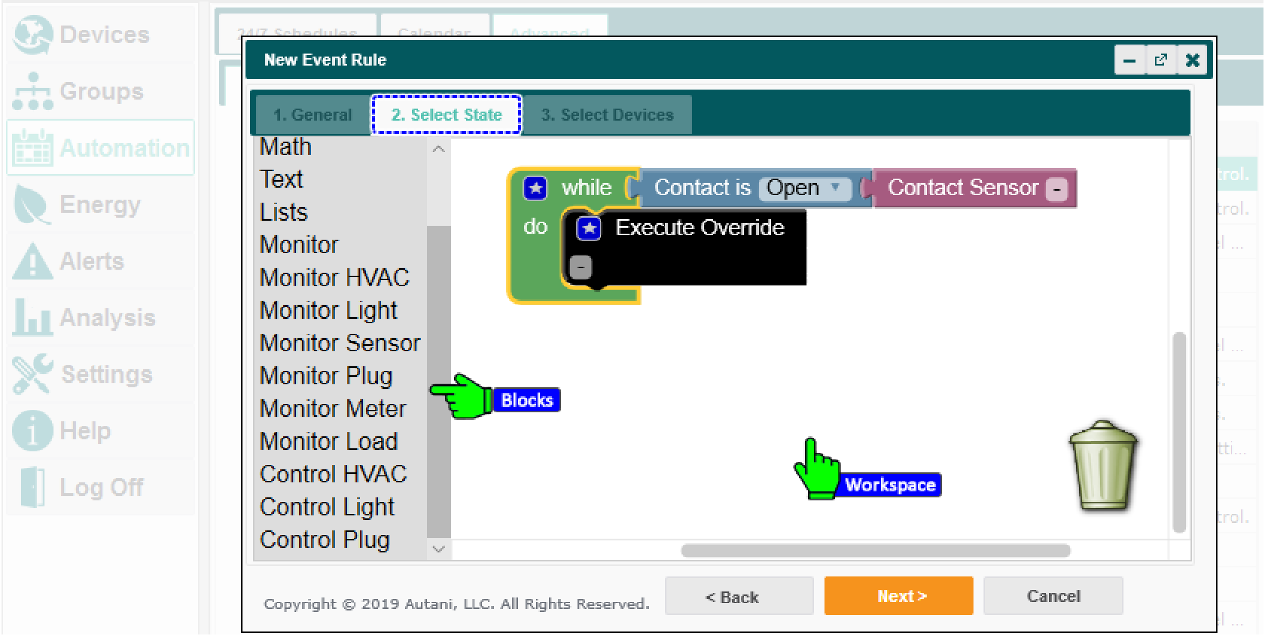
New Event Rule

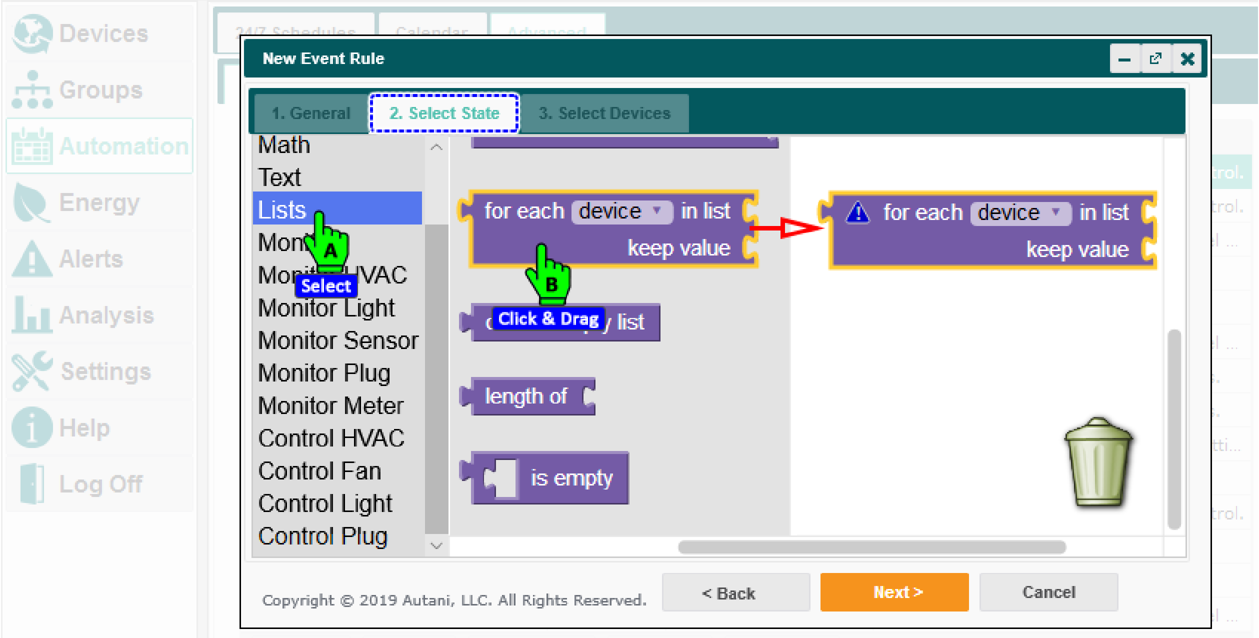
Copyright © 2019 Autani, LLC. All Rights Reserved.

3. Scripting Language

3.1. Basics

When you first open the PRO rule editor, the rule itself will be displayed in the editor's workspace. There is a menu on the left-hand side of the editor that can be used to select new blocks to use for the rule. Just drag a block from the menu to the workspace, and connect it to the other blocks in the rule:





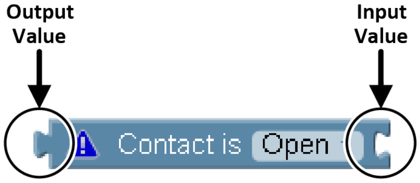
Blocks are color-coded to indicate how they are used, based on the following categories:

- Rule types
- Logic
- Math
- Text
- Lists
- Monitoring
- Control and Alerting
- Groups and Devices

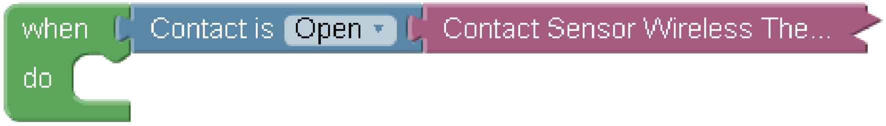
NOTE: Some of the blocks will not be displayed if an appliance does not have any devices of the corresponding type.

Blocks can be connected in two ways:

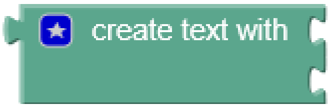
- Notches on the left and right are used to pass values between blocks. A block that accepts values will have one or more notches on its right-hand side to receive input values. A block that returns a value will have a single notch on its left-hand side to pass along its output value:



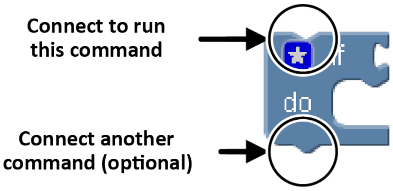
  For example, the **contact-is-open** block below receives the device to monitor (a wireless contact sensor) as an input value, and passes an output value to the **when** block indicating whether the contact is open.

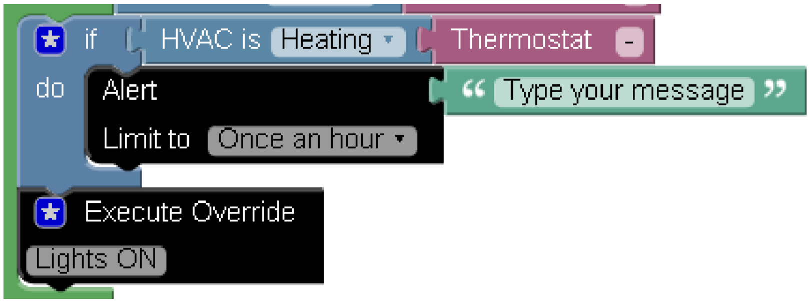


  Some blocks can receive more than one input value but a block can never have more than one output value:



- Notches on the top and bottom can be used to execute the blocks sequentially, one after the other:

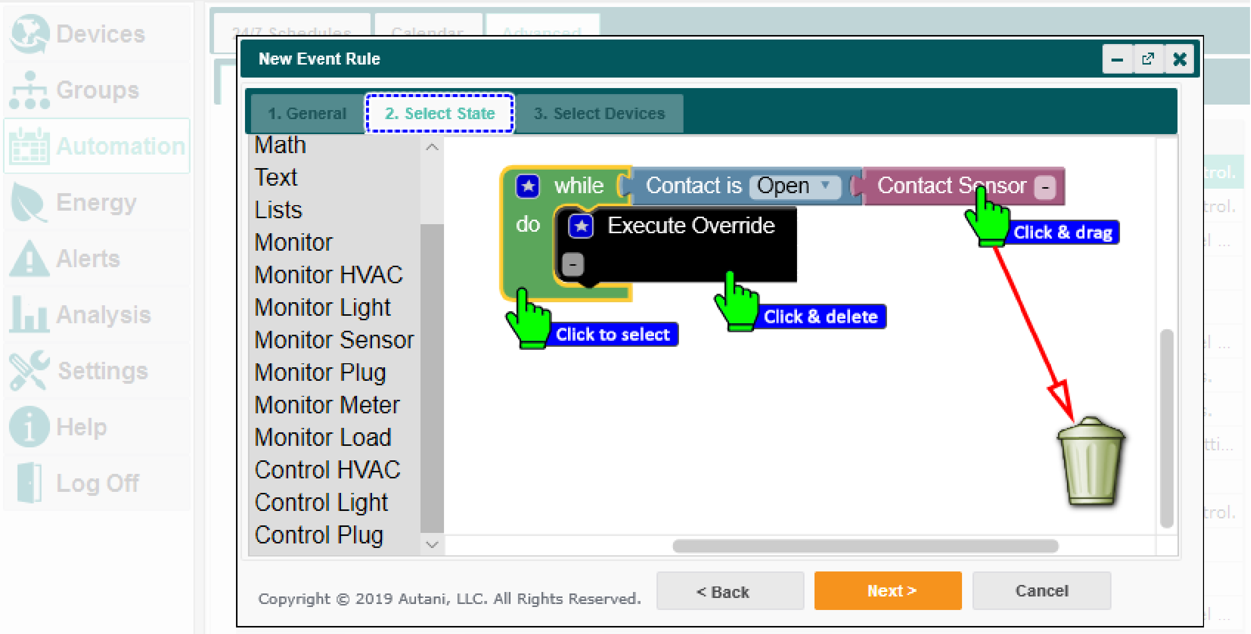

Connect to run this command
if
do
Connect another command (optional)



  For example, the **if** block above is connected to a rule type block (top) and an **execute-override** block (bottom). If the rule type condition is true, control will pass to the **if** block and then to the **execute-override** block.

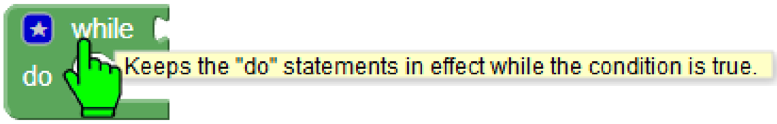
A block may be selected by clicking on it.

A block may be deleted by selecting it and pressing the delete key, dragging it to the trash can, or right-clicking on it and selecting delete.



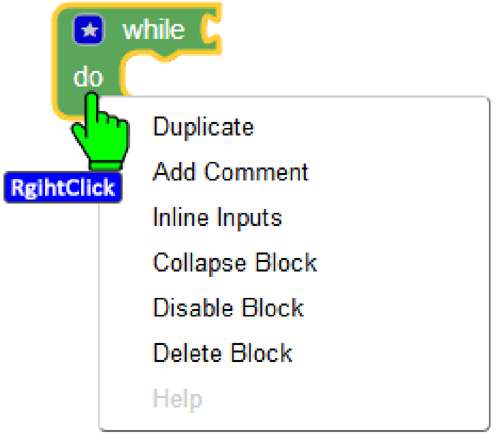
Copyright © 2019 Autani, LLC. All Rights Reserved.

Inline help may be displayed by hovering over a block:

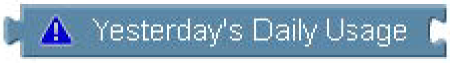


Keeps the "do" statements in effect while the condition is true.

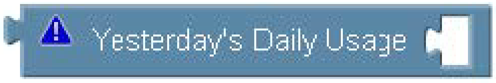
Right-clicking on a block displays a menu to perform the following options:



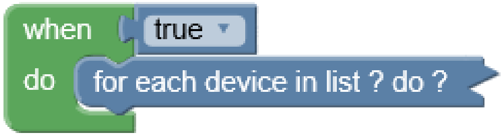
- Duplicate – Create a copy of the block.
- Add Comment – Add free-form text to a block, for example to explain what the rule is doing or why a particular block is needed.
- Inline Inputs / External Inputs – Change how a block's input notch is displayed:
  - □ External:

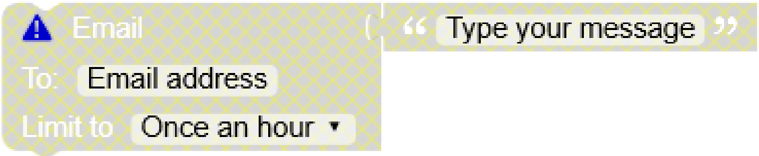


  - □ Inline:



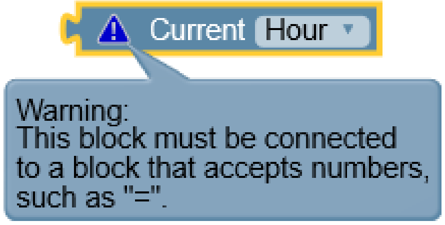
- Collapse Block – Reduces the size of a block visually to save space on the screen. This can be useful when working with a large rule.
- Disable Block / Enable Block – Toggle whether the selected block is used by the rule. It is an error if a disabled block is connected to other blocks that are not disabled.





- Delete Block – Delete the selected block.

If there is a problem with a block that needs to be corrected, a triangular warning icon will be displayed on the block. The icon can be clicked for more information:



Warning:
This block must be connected to a block that accepts numbers, such as "=".

3.2. Rule Types

A PRO rule must be based on a single Rule Type block. All of the other blocks in the rule are connected to the rule type. PRO does not allow a single rule to contain multiple rule types or other blocks not connected to the rule type.

There are two types of PRO rules:

- **While** rules detect a condition is true and place devices into an override state (overriding 24/7 schedules and lower priority overrides) until the condition is no longer met.
- **When** rules trigger an immediate response as soon as a condition is detected but do not place any devices into an override state.

3.3. Data Types

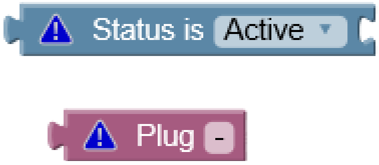
Data is exchanged when a block is plugged into the input of another block. This data can be any of the following types:

- Boolean – A logical value that is either **true** (*yes*) or **false** (*no*).
- Number
- Text
- List – A list consists of zero or more elements of any data type.
- Device – A reference to a device. Some blocks accept any device, and others only accept a specific type (Sensor, Thermostat, Light, etc).
- Group – A reference to locations and/or collections defined elsewhere in EnergyCenter®.

Each block's input has a limit on what block types with which it can be connected. If you try connecting a block to an input that does not accept blocks of that type, the blocks will be repelled and the connection will not be allowed.

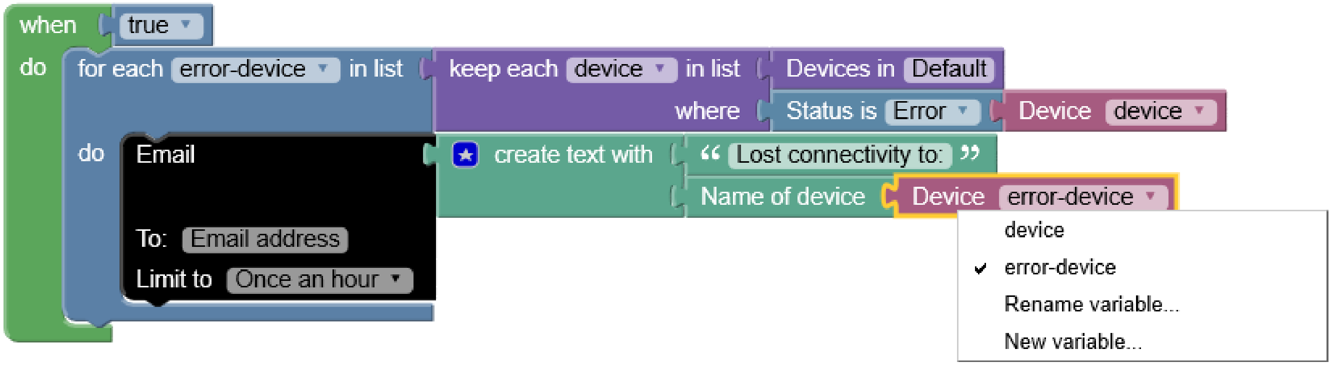
3.4. Dropdowns and Selection Menus

Many blocks have a lighter-colored area inside them with either an arrow (to indicate a dropdown menu) or a dash (by default; to indicate a selection menu). Clicking upon this area will bring up a menu to select the intended value.



3.5. Variables

Some blocks create a variable to refer to a device at a later time. Each variable is assigned a different name and may be referred to later on in this script using this name.


when true
do for each error-device in list keep each device in list Devices in Default
where Status is Error Device device
do Email create text with " Lost connectivity to: "
Name of device Device error-device
To: Email address
Limit to Once an hour
device
✔ error-device
Rename variable...
New variable...

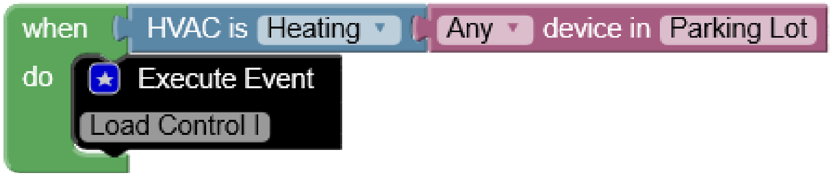
For example, in the above script, the ***for each*** block executes its do section for every member of the input list from the ***keep each*** block. The ***for each*** creates a variable ***error device*** to refer to a single member of the list on the current pass. The **error device** variable is used later by a ***name of device*** block to create an alert email.

In addition, a variable called ***device*** is used by the ***keep each*** block to refer to each member of its input list. The device variable is then used inside the ***where*** clause to determine if a list member is in ***error*** state.

A variable can only be used after it has been created (or *declared*). For example, in the above script it would be an error to use either variable in the **when** block's condition section (the part with the ***true*** block).

3.6. Group Operations

A group block may be used in most places that a device block would be used, to let the logic refer to *any* or *every* applicable device from the group.

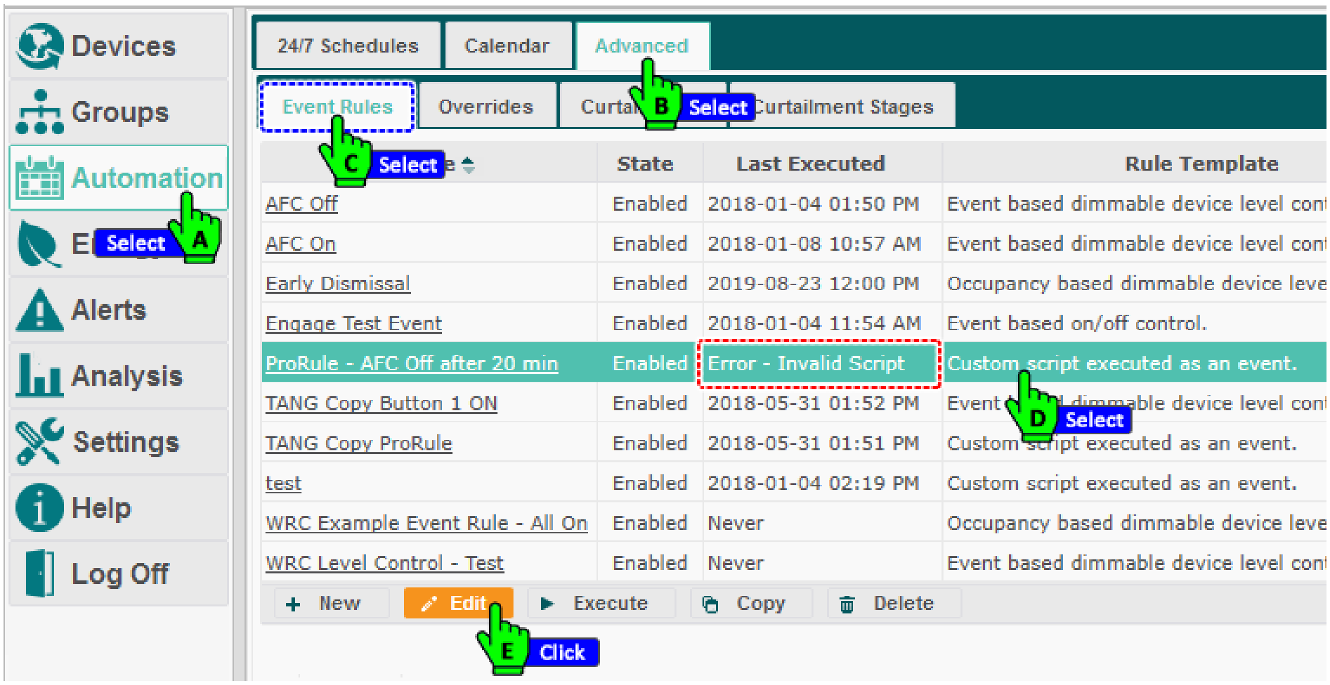


For example, in the above script the event Loads ON will be executed if any thermostat is heating.

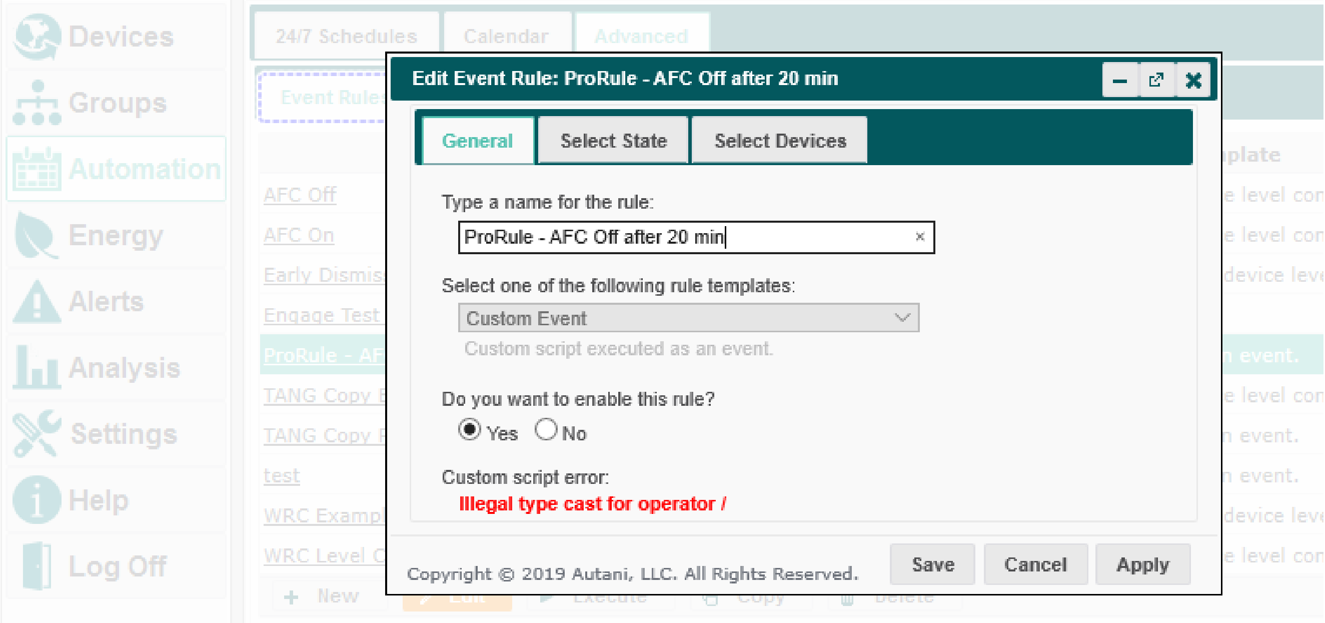
When using a group in this manner, the group should only contain devices of the applicable type. If a suitable group does not exist it may be necessary to create a new collection group.

3.7. Script Errors

The rule editor prevents most script errors by preventing blocks from being connected to other blocks of the wrong type. Usually this is enough to prevent an error. But sometimes it is possible for the blocks to be connected in ways that are not allowed. In this case, shortly after the rule is saved, an error will be displayed in the event rules grid. Opening up the rule will display more information about the error.



e	State	Last Executed	Rule Template
AFC Off	Enabled	2018-01-04 01:50 PM	Event based dimmable device level cont
AFC On	Enabled	2018-01-08 10:57 AM	Event based dimmable device level cont
Early Dismissal	Enabled	2019-08-23 12:00 PM	Occupancy based dimmable device leve
Engage Test Event	Enabled	2018-01-04 11:54 AM	Event based on/off control.
ProRule - AFC Off after 20 min	Enabled	Error - Invalid Script	Custom script executed as an event.
TANG Copy Button 1 ON	Enabled	2018-05-31 01:52 PM	Event dimmable device level cont
TANG Copy ProRule	Enabled	2018-05-31 01:51 PM	Custom script executed as an event.
test	Enabled	2018-01-04 02:19 PM	Custom script executed as an event.
WRC Example Event Rule - All On	Enabled	Never	Occupancy based dimmable device leve
WRC Level Control - Test	Enabled	Never	Event based dimmable device level cont



Edit Event Rule: ProRule - AFC Off after 20 min

Type a name for the rule:
ProRule - AFC Off after 20 min

Select one of the following rule templates:
Custom Event
Custom script executed as an event.

Do you want to enable this rule?
● Yes ○ No

Custom script error:
Illegal type cast for operator /

Copyright © 2019 Autani, LLC. All Rights Reserved.

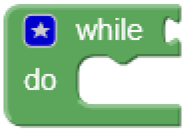
3.8. Reference

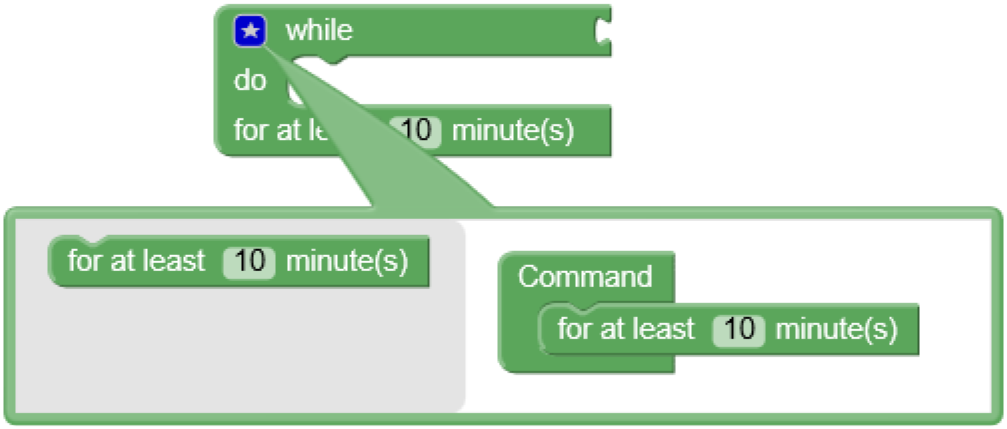
This **Reference** contains all of the blocks available for PRO rules. Note that some of these blocks will not be displayed if an appliance does not have any devices of the corresponding type.

3.8.1. Rule Type

3.8.1.1. While

A **while** block receives an input condition and executes its do section while the condition is true. Any changes to device settings will be maintained at the device as an override until the condition is no longer true. A time modifier may also be used to maintain the override for a minimum amount of time, even if the condition is no longer true.

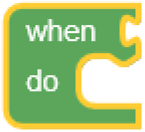




The PRO rule will periodically test the Rule Type condition to see if it is true. The condition will also be tested as soon as any reports come in from devices that are being monitored by the condition.

3.8.1.2. when

when also receives an input and executes a **do** section. Like the **while** block, **when** will immediately change device settings **when** the condition is met. However, **when** will not maintain an override state on any affected devices.

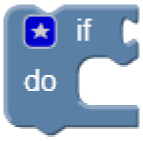


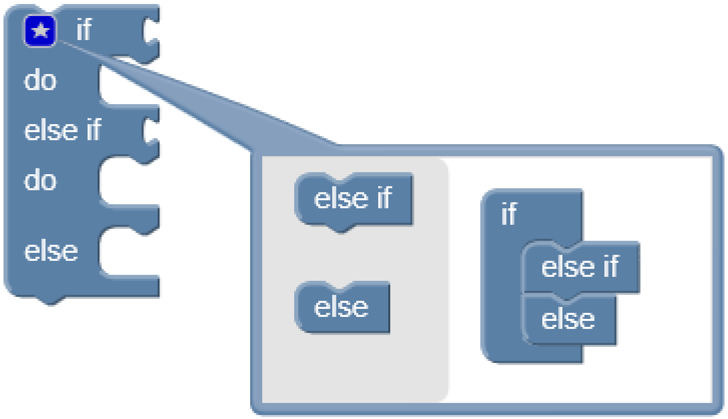
3.8.2. Logic

3.8.2.1. If

The **while/when** blocks only have a single condition input. But sometimes a rule needs to take different actions depending upon the state of many conditions. This is where the **if** block comes into play.

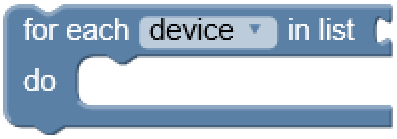
The **if** block contains a Boolean condition input and a do section. Multiple condition's/do's may be added by adding new **else if** blocks, and an else may be added to execute a behaviour if none of the conditions is true.





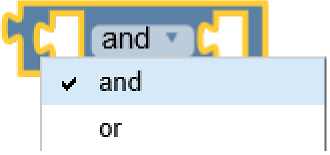
3.8.2.2. For each

This block may be used to loop over an input list, executing the do section once for each member of the list.



3.8.2.3. Boolean logic

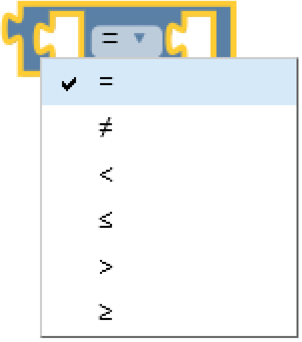
The Boolean logic block may be used to test multiple conditions and return true if both of the conditions are true (**and**) or if at least one of them is true (**or**).



These blocks may be combined if more than two conditions must be tested.

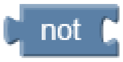
3.8.2.4. Numeric equality

This block allows two numbers to be compared for equality, not equal, less than, less than equal, greater than, or greater than equal.



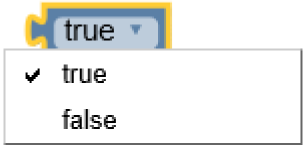
3.8.2.5. Not

The **not** block inverts a Boolean value.



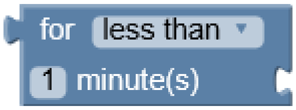
3.8.2.6. Boolean

This block refers to a Boolean value directly. Sometimes this is useful if a **while**/**when** rule should always be executed and another block such as an **if** statement is used to check the conditions.



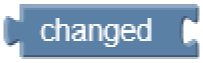
3.8.2.7. For at least / less than

This block is used to test that a condition is present for at least (or less than) a specific number of minutes.



3.8.2.8. Changed

This block will return true if the input value changes.

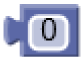


3.8.3. Math

Each of the math blocks takes one or more numbers as input, performs some kind of operation, and outputs another number as the result.

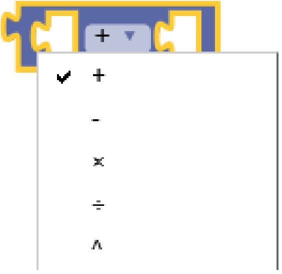
3.8.3.1. Number

This block refers to the number entered into its text box.



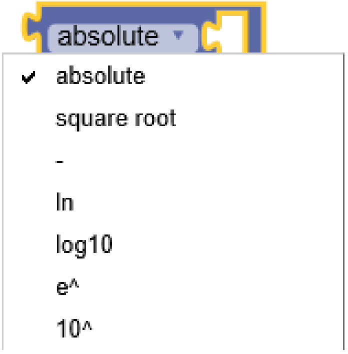
3.8.3.2. Operations

This block allows operations to be performed on to numbers: addition, subtraction, multiplication, division, and exponentiation.



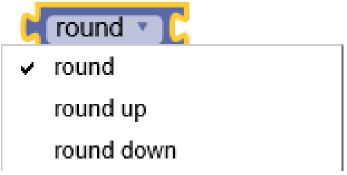
3.8.3.3. Functions

This block performs the selected mathematical function on a single input number and outputs the result.



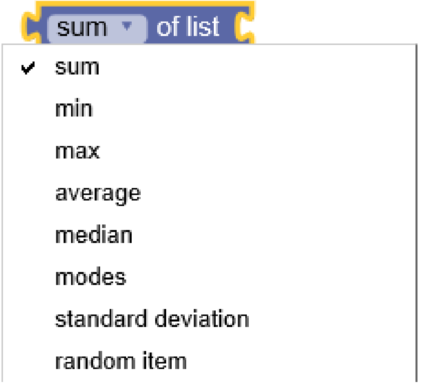
3.8.3.4. Rounding

This function rounds the input to a whole number.



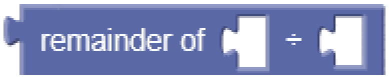
3.8.3.5. Aggregate Functions

This function receives a list of numbers and performs the selected function on them.



3.8.3.6. Remainder

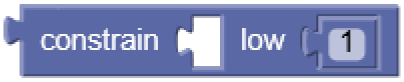
This block computes the remainder of dividing two input numbers.

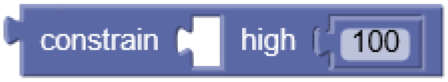


3.8.3.7. Constraints

These blocks return the input number, unless:

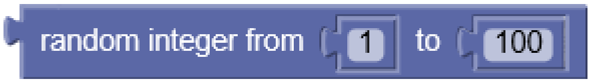
- If the number is below “low”, the low number will be returned instead.
- If the number is above “high”, the high number will be returned.





3.8.3.8. Random Numbers

This block generates a random number.

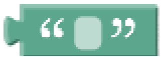


3.8.4. Text

These blocks are used to build up text, for example to output as an alert or email.

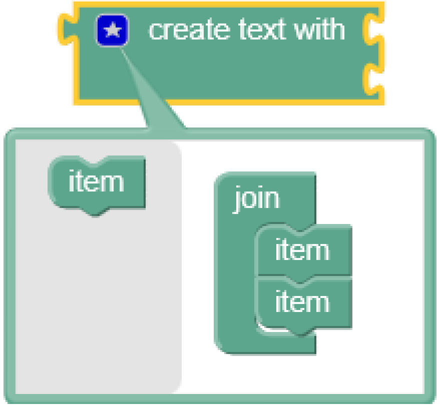
3.8.4.1. Create Text

This block creates a simple string of text:



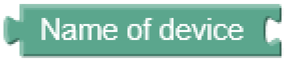
3.8.4.2. Join Text

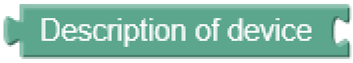
This block takes one or more strings of text as input and joins them together. As many items as necessary can be added to allow more text to be joined.

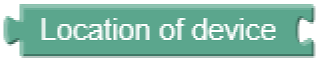


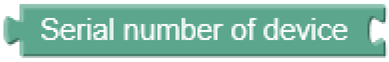
3.8.4.3. Device Identification

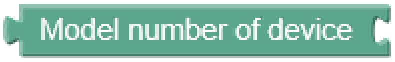
Each of these blocks receives a device block as input and outputs text corresponding to the given device.











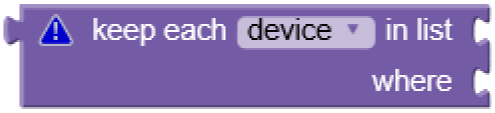
3.8.5. Lists

These blocks are responsible for working with lists. A list is simply zero or more elements, generally of the same type.

Lists should not be confused with groups. A list can be created from a group using the *List of Devices in Group* block. A group may be used in place of a device, but a list cannot. List and group blocks are color-coded to try to avoid confusion.

3.8.5.1. Filter

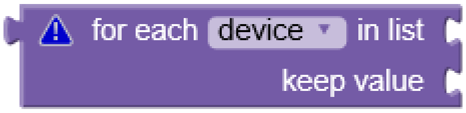
The keep block receives a list as input and loops over the list, creating another output list. A variable is created to refer to the current list member in the where clause. A condition in the where clause is executed once for each member of the list. A list member is only added to the output list if the condition is true for that member.



This is useful to, for example, find all the thermostats that are reporting a temperature that does not meet their setpoint.

3.8.5.2. Iteration

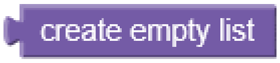
The **for each** block is similar to the keep block, however instead of executing a condition for each list member, it executes a keep value clause. The value returned by this clause will be added to the output list.



A **for each** block might be useful for receiving a list of thermostats and returning another list containing their current temperatures.

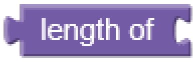
3.8.5.3. Create empty list

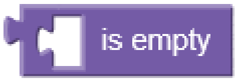
This block creates and empty list.



3.8.5.4. List length

The length block returns the **length of** the list, and the **is empty** block returns a Boolean indicating whether the list is empty:

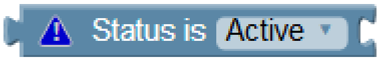
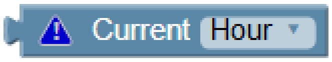
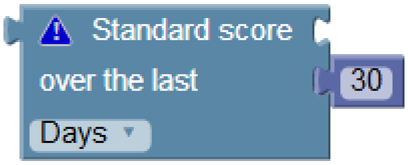
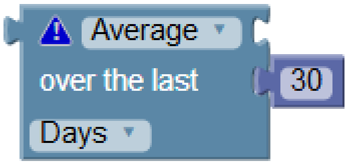
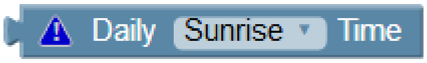




3.8.6. Monitoring, General

Blocks in this section are used to monitor a control point on a device. Note that these blocks will only be displayed if at least one of the corresponding devices exists on the Manager.

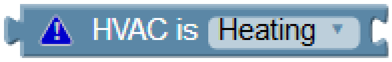
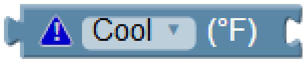
Table 1 : Monitoring, General

Block	Inputs	Output	Notes
 **Status is** Active | Warning | Error	Device or Group	Boolean	Determine the communication status of a device.
 **Current** Hour | Minute | Second | Month | Day | Year | Day of the week | Time		Number	
 **Standard score over the last** Days | Hours | Minutes	Control Point, Interval	Number	Compute the standard score using historical data for the given control point over the given interval. Control point is a numeric monitoring block with an attached device. For example, temperature from a thermostat.
 **Average** | **Count** | **Max** | **Min** | **Standard Deviation** | **Sum** Days | Hours | Minutes	Control Point, Interval	Number	Compute the selected aggregate function using historical data for the given control point over the given interval. Control point is a numeric monitoring block with an attached device. For example, temperature from a thermostat.
 **Daily** Sunrise | **Daily** Sunset **Time**	Control Point, Interval		

3.8.7. Monitoring, HVAC

Blocks in this section are used to monitor a control point on a device. Note that these blocks will only be displayed if at least one of the corresponding devices exists on the Manager.

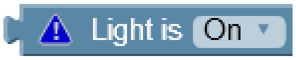
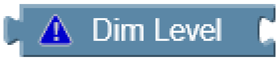
Table 2 : Monitoring, HVAC

Block	Inputs	Output	Notes
 **HVAC is** Heating | Cooling | Idle	Thermostat or Group	Boolean	Determine if the thermostat is currently performing the selected operation.
 **Setpoint** Cool | Heat | Occupied Cool | Unoccupied Cool | Occupied Heat | Unoccupied Heat	Thermostat or Group	Number (degrees)	Retrieve the value of the thermostat's current setpoint.

3.8.8. Monitoring, Lights

Blocks in this section are used to monitor a control point on a device. Note that these blocks will only be displayed if at least one of the corresponding devices exists on the Manager.

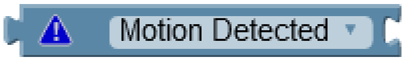
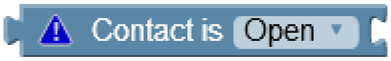
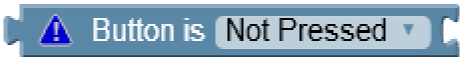
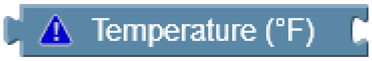
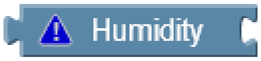
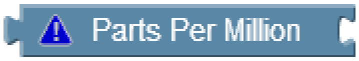
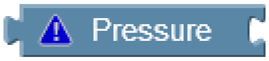
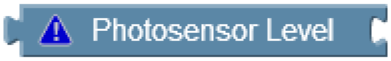
Table 3 : Monitoring, HVAC

Block	Inputs	Output	Notes
 **Light is** On | Off	Light or Group	Boolean	
 **Dim Level**	Dimmable Light or Group	Number (%)	

3.8.9. Monitoring, Sensors

Blocks in this section are used to monitor a control point on a device. Note that these blocks will only be displayed if at least one of the corresponding devices exists on the Manager.

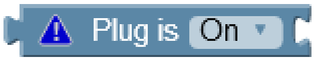
Table 4 : Monitoring, Sensors

Block	Inputs	Output	Notes
 Motion Detected | No Motion	Occupancy Sensor or Group	Boolean	
 **Contact is** Open | Closed	Contact Sensor or Group	Boolean	
 **Button is** Pressed | Not Pressed	Button or Group	Boolean	
 **Temperature**	Temperature Sensor, Thermostat, or Group	Number (degrees)	
 **Humidity**	Humidity Sensor or Group	Number (rh)	
 **Parts Per Million (PPM)**	PPM Sensor or Group	Number (ppm)	
 **Pressure**	Pressure Sensor or Group	Number (inch wc)	
 **Photosensor Level**	Photosensor, Light, or Group	Number (%)	

3.8.10. Monitoring, Plugs

Blocks in this section are used to monitor a control point on a device. Note that these blocks will only be displayed if at least one of the corresponding devices exists on the Manager.

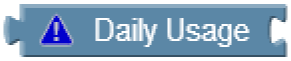
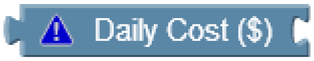
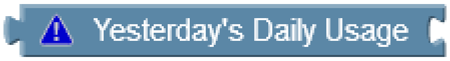
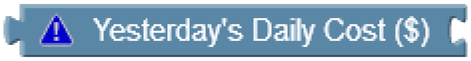
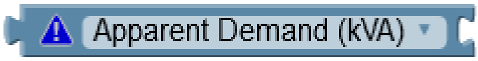
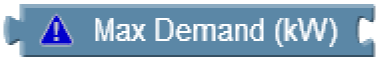
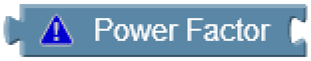
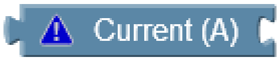
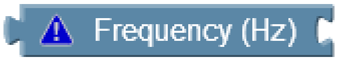
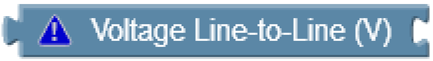
Table 5 : Monitoring, Plugs

Block	Inputs	Output	Notes
 Plug is On ▾ **Plug is** On | Off	Plug or Group	Boolean	

3.8.11. Monitoring, Meter

Blocks in this section are used to monitor a control point on a device. Note that these blocks will only be displayed if at least one of the corresponding devices exists on the Manager.

Table 6 : Monitoring, Meter

Block	Inputs	Output	Notes
 Daily Usage **Daily Usage**	Meter, Plug, VFD, or Group	Number	
 Daily Cost ($) **Daily Cost**	Meter, Plug, VFD, or Group	Number ($)	
 Yesterday's Daily Usage **Yesterday's Usage**	Meter, Plug, VFD, or Group	Number	
 Yesterday's Daily Cost ($) **Yesterday's Cost**	Meter, Plug, VFD, or Group	Number ($)	
 Apparent Demand (kVA) ▾ Apparent Demand (kVA) | Reactive Demand (kVAR) | Real Demand (kW)	Meter or Group	Number	
 Max Demand (kW) **Max Demand**	Meter or Group	Number (kW)	
 Power Factor **Power Factor**	Meter or Group	Number	
 Current (A) **Current**	Meter or Group	Number (A)	
 Frequency (Hz) **Frequency**	Meter or Group	Number (Hz)	
 Voltage Line-to-Line (V) **Voltage Line-to-Line**	Meter or Group	Number	

Block	Inputs	Output	Notes
Voltage Line-to-Neutral (V) **Voltage Line-to-Neutral**	Meter or Group	Number (V)	
Apparent Energy Delivered (kVAh) **Apparent Energy Delivered**	Meter or Group	Number (kVAh)	
Reactive Energy Delivered (kVARh) **Reactive Energy Delivered**	Meter or Group	Number (kVARh)	
Instantaneous Apparent Power (kVA) **Instantaneous Apparent Power**	Meter or Group	Number (kVA)	
Instantaneous Reactive Power (kVAR) **Instantaneous Reactive Power**	Meter or Group	Number (kVAR)	
Instantaneous Real Power (kW) **Instantaneous Real Power**	Meter or Group	Number (kW)	

3.8.12. Monitoring, Load

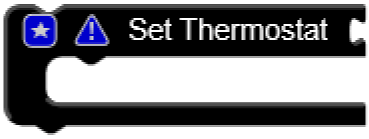
Blocks in this section are used to monitor a control point on a device. Note that these blocks will only be displayed if at least one of the corresponding devices exists on the Manager.

Table 7 : Monitoring, Meter

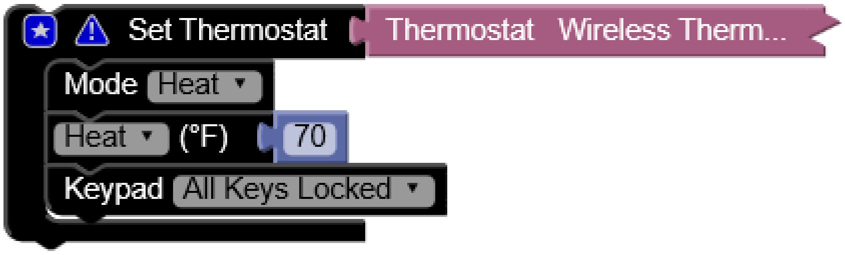
Block	Inputs	Output	Notes
Load is On ▾ **Load is** On | Off	Load or Group	Boolean	

3.8.13. Control, HVAC

Blocks in this section are used to change the value of device control points. Each of these blocks must be added to a corresponding Set block that refers to the device to change. For example, the **Set Thermostat** block would be used to specify a thermostat to change, and a sequence of control blocks to indicate the setpoint and other control points that will be changed:

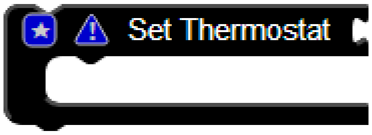
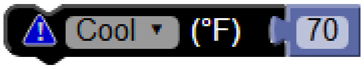
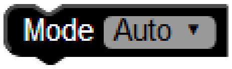
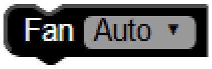
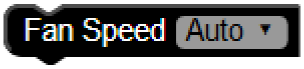
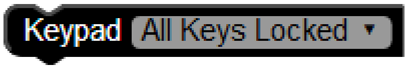


Set Thermostat



Set Thermostat Thermostat Wireless Therm...
Mode Heat ▾
Heat ▾ (°F) 70
Keypad All Keys Locked ▾

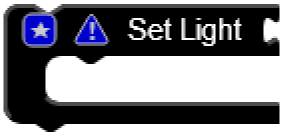
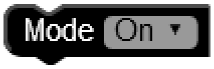
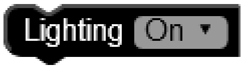
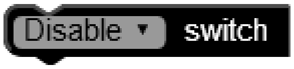
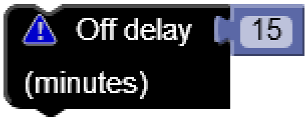
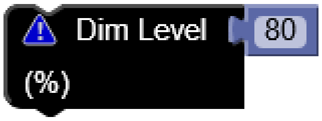
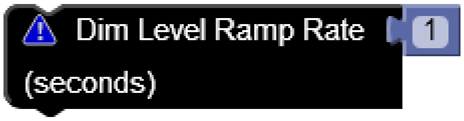
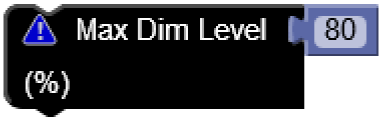
Table 8 : Control, HVAC

Block	Inputs	Notes
 Set Thermostat **Set Thermostat**	Thermostat Devices	
 Cool ▾ (°F) 70 **Setpoint** Cool | Heat | Occupied Cool | Unoccupied Cool | Occupied Heat | Unoccupied Heat	Number (degrees)	Sets value of the selected setpoint.
 Mode Auto ▾ **Mode**	Auto | Cool | Heat | Off	
 Fan Auto ▾ **Fan**	Auto | On	
 Fan Speed Auto ▾ **Fan Speed**	Auto | High | Medium | Low	
 Keypad All Keys Locked ▾ **Keypad**	All Keys Locked | Partial Lock | All Keys Unlocked	

3.8.14. Control, Lights

Blocks in this section are used to change the value of device control points. Each of these blocks must be added to a corresponding Set block that refers to the device to change.

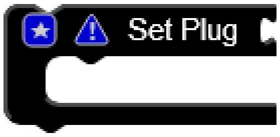
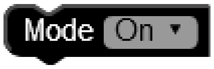
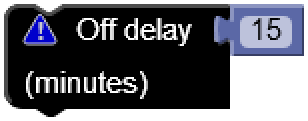
Table 9 : Control, Lights

Block	Inputs	Notes
 **Set Light**	Lighting Devices	
 **Mode**	On | Off | Smart On/Off | Vacancy	
 **Lighting**	On | Off	
 **Switch**	Disable | Enable	
 **Off Delay**	Number (minutes)	
 **Dim Level**	Number (%)	
 **Dim Level Ramp Rate**	Number (seconds)	
 **Max Dim Level**	Number (%)	

3.8.15. Control, Plugs

Blocks in this section are used to change the value of device control points. Each of these blocks must be added to a corresponding Set block that refers to the device to change.

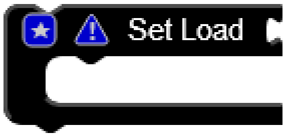
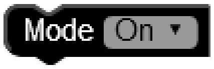
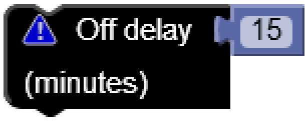
Table 10 : Control, Plugs

Block	Inputs	Notes
 **Set Plug**	Plug devices	
 **Mode**	On | Off | Smart On/Off | Vacancy	
 **Off Delay**	Number (minutes)	

3.8.16. Control, Load

Blocks in this section are used to change the value of device control points. Each of these blocks must be added to a corresponding Set block that refers to the device to change.

Table 11 : Control, Load

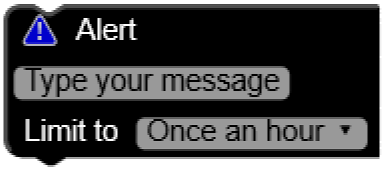
Block	Inputs	Notes
 **Set Load**	Load Devices	
 **Mode**	On | Off |Smart On/Off | Vacancy	
 **Off Delay**	Number (minutes)	
Output	Number (%)	

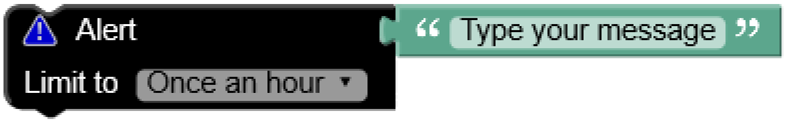
3.8.17. Alerts

Blocks in this section may be used to raise some form of an alert. To avoid generating too many alerts, an alert may be limited to once an hour, once a day, or once a week.

3.8.17.1. User Defined Alert

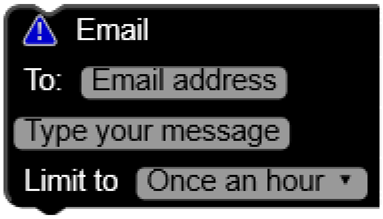
These blocks may be used to raise a user-defined alert. These alerts can be viewed in EnergyCenter® under Alerts | Recent Alerts.

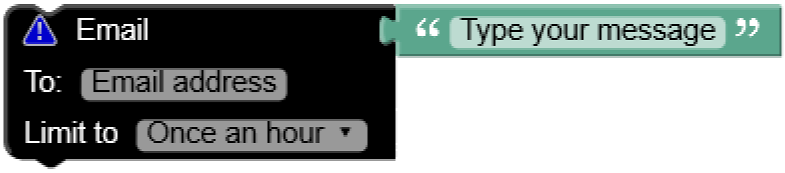


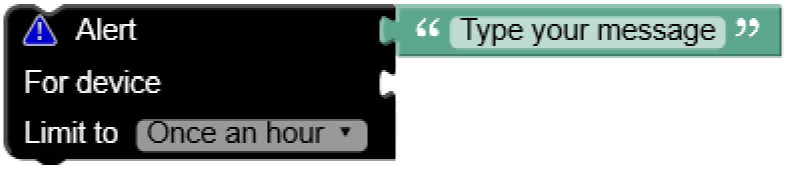


3.8.17.2. Email Alert

These blocks may be used to send an email alert:



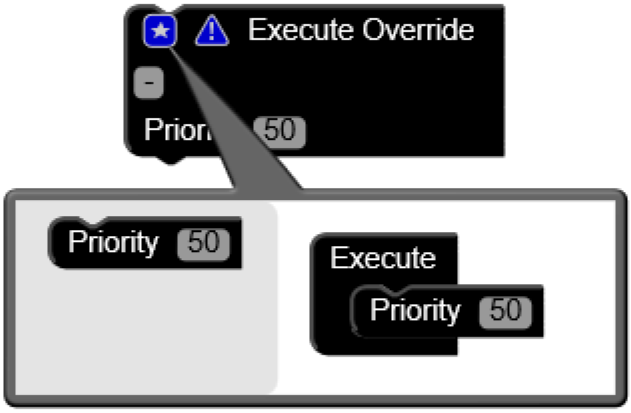


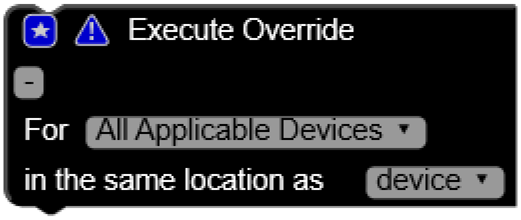


3.8.18. Overrides

3.8.18.1. Execute Override

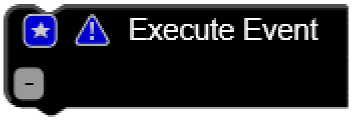
These blocks may be used to execute all of the event rules contained in the given override. Note that an override is not actually raised by this block – if this block is executed by a while rule then any affected devices will be placed in override state with the given priority. However, if the block is executed by a when rule the affected devices will not be placed into an override.



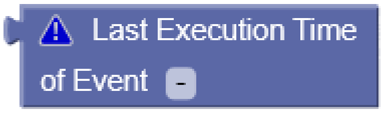


3.8.18.2. Execute Event

This block has the same effect as execute override, but only the given event rule will be executed.



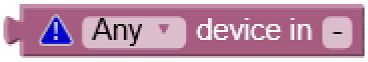
The **last execution time** block may be used to return the last time an event was executed, as a number of seconds. This can be used by a script to, for example, determine which event to execute based on which one was executed most recently.

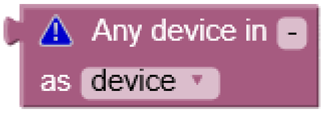


3.8.19. Groups

3.8.19.1. Any / Every Device in Group

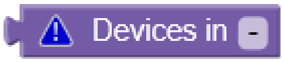
These blocks may be used anywhere a device block is accepted, to refer to either any device or every device in the group.

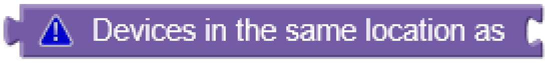




3.8.19.2. List of Devices in Group

This block may be used to create a new list containing a reference to each device in the given group. This is useful to combine with a filtering or iteration block.

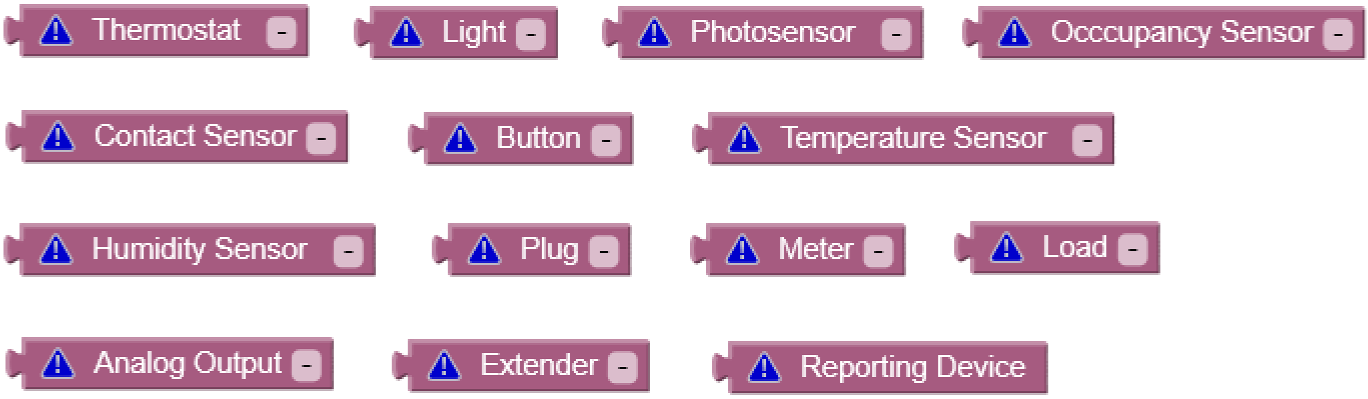




3.8.20. Devices

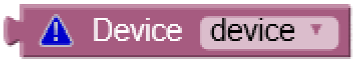
3.8.20.1. Device Blocks

These blocks may be used to refer to a specific device.



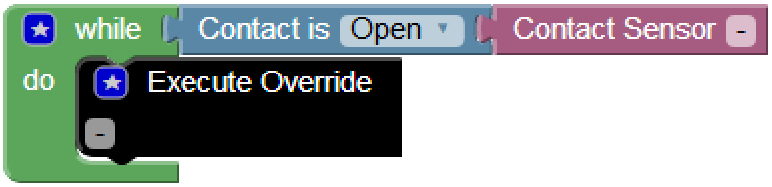
3.8.20.2. Device Variable

This block may be used to refer to a single device referenced by a variable.



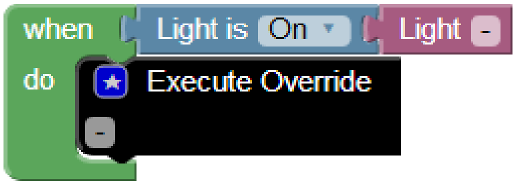
4. Examples

Example 1: Begin an override using a contact sensor



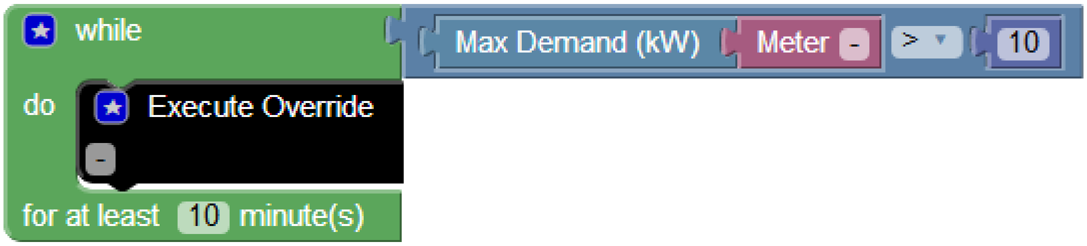
```
while  Contact is Open ▾  Contact Sensor -
do
    Execute Override
    -
```

Example 2: Trigger other events using an on/off light



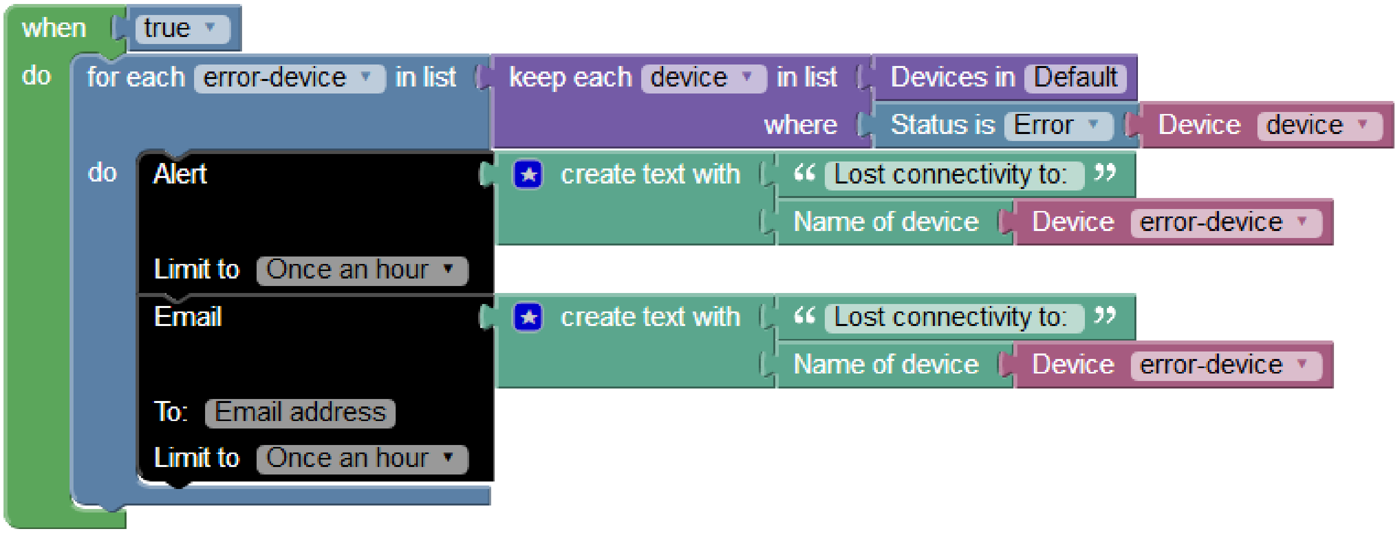
```
when  Light is On ▾  Light -
do
    Execute Override
    -
```

Example 3: Reduce energy usage if max demand exceeds threshold



```
while  Max Demand (kW)  Meter -  > ▾  10
do
    Execute Override
    -
for at least 10 minute(s)
```

Example 4: Alert when device connectivity is lost



```
when  true ▾
do  for each error-device ▾ in list  keep each device ▾ in list  Devices in Default
                                                          where  Status is Error ▾  Device device ▾
    do  Alert   create text with  " Lost connectivity to: "
                                  Name of device  Device error-device ▾
        Limit to Once an hour ▾
        Email   create text with  " Lost connectivity to: "
                                  Name of device  Device error-device ▾
        To: Email address
        Limit to Once an hour ▾
```

Contact Information

- □ **Autani Support**

  Phone: 443.320.2233 x2
  Address: 7001 Columbia Gateway Drive, Suite 210, Columbia, MD 21046 USA
  Support/Commissioning Services: support@autani.com

- □ **Autani Sales**

  Phone: 443.320.2233 x1
  Sales/Quotations: sales@autani.com, quotes@autani.com
  General Inquiries: information@autani.com

Hours of Operation: Monday to Friday, 9am to 5pm, Eastern Standard Time

Intentionally Left Blank.

Autani LLC
7001 Columbia Gateway Drive, Suite 210, Columbia, MD 21046 USA,
(443) 320-2233 | (240) 755-0092 (fax) www.autani.com.