Remote Builder – SmartGrid Help Version 4.3.0 (6/21/25)

Table of Contents

Introduction

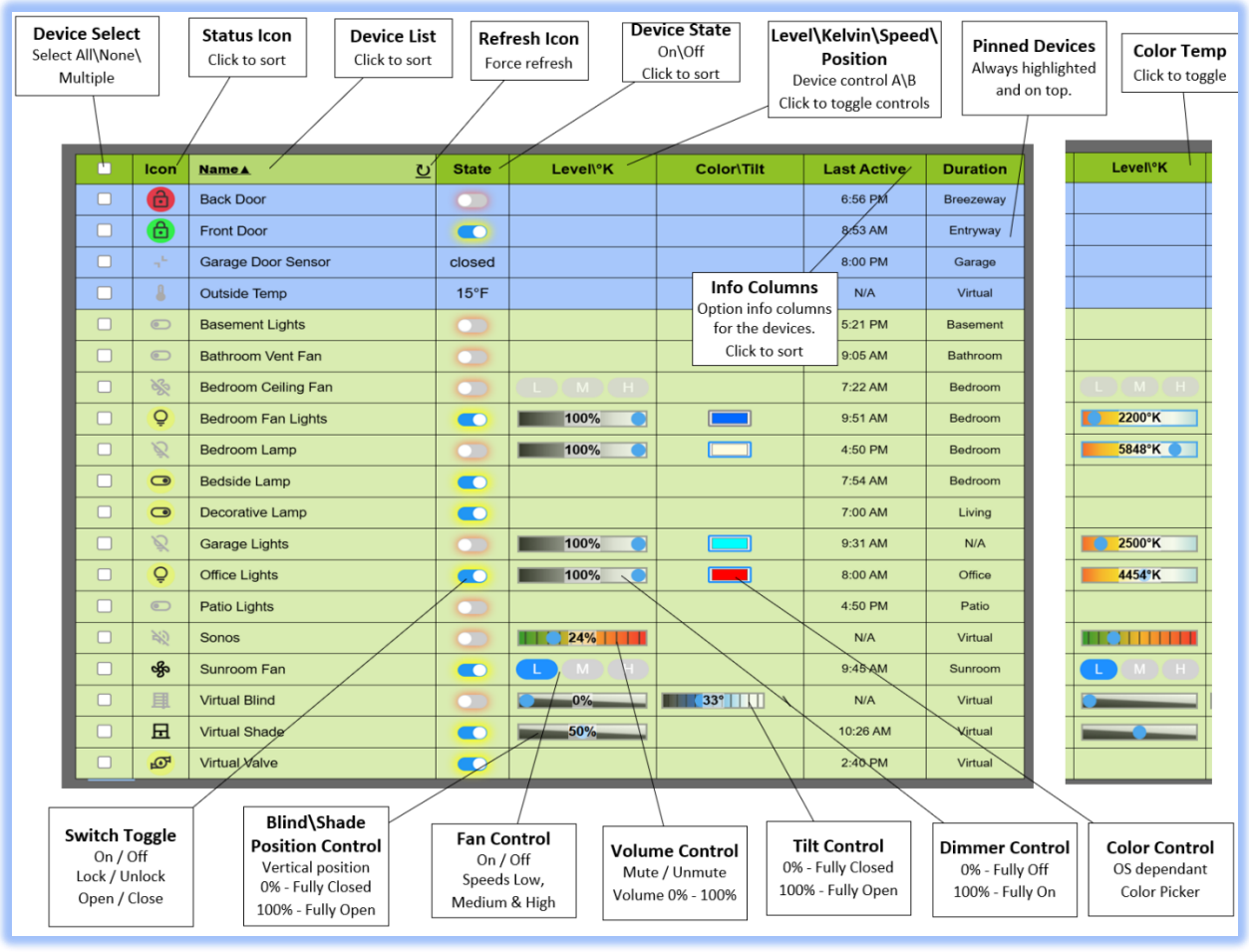
This is a standalone help file for Remote Builder SmartGrid. For a complete introduction to Remote Builder see the help file at this location: https://github.com/GaryMilne/Hubitat-RemoteBuilder/blob/main/Remote%20Builder%20Help.pdf

Errata: Some screenshots may show an older version of the menu system which does not capture all the options.

SmartGrid

SmartGrid combines the tabular format of **Tile Builder** with the ability to control a variety of Hubitat devices via a dense but intuitive interface that can be accessed from anywhere on any device with a modern browser.

SmartGrid Showing Pinned Rows



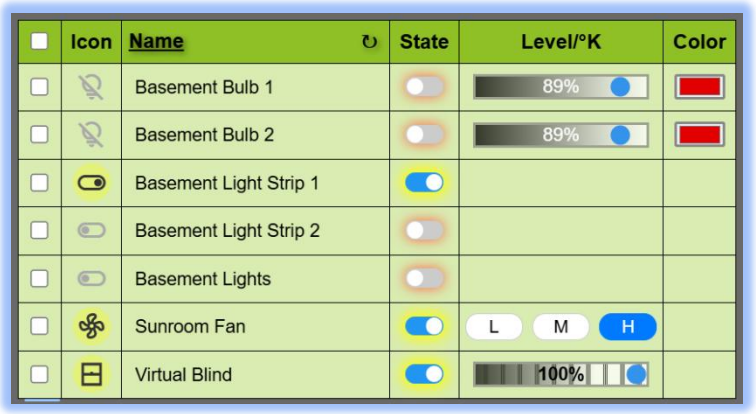
SmartGrid Showing Custom Sort with Variables

Originally designed for use with bulbs it has since been expanded to manage Valves, Locks, Fans, Garage Doors, Blinds, Shades(vertical) and a variety of sensors. The ability to multi-select devices and change them all with one command is a convenient way to work with groups of devices.

Create a New SmartGrid

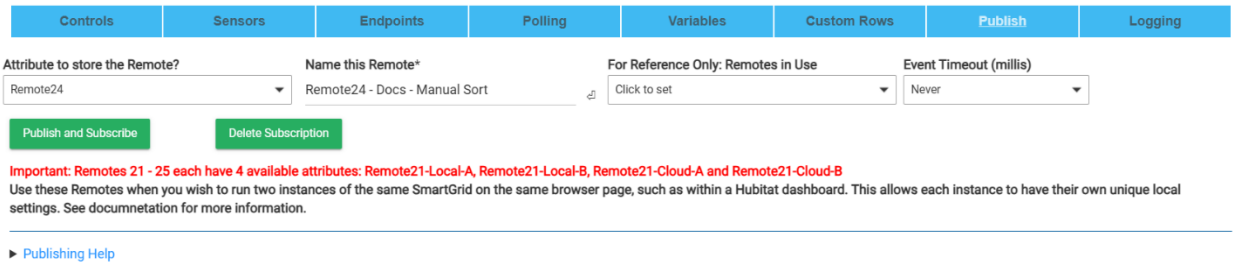
After you have installed the SmartGrid child module you can select to “Add SmartGrid” from the parent.

First you must select the devices you wish to work with in your grid. If you wish you can select from a filtered list by selecting a required capability like “Color Devices” or “Dimmer,” or you can leave it at the default and all supported device types will be available. Once you have selected devices the SmartGrid will be displayed and show all the current values. It might look something like this:

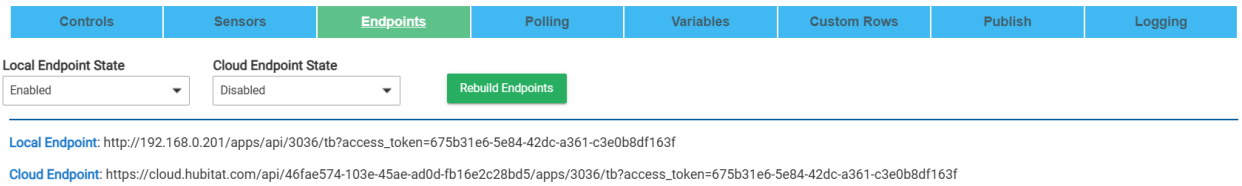


All devices will have a switch so the state column will always be fully populated, but the other columns will only have controls relevant to the device capability, such as color or dimmer.

Before your SmartGrid works correctly you must perform a Publish and Subscribe. Go to the Publish Remote section and give it a name, then select the Attribute to publish to and click on **Publish and Subscribe.** We will come back to the Event Timeout later, but it functions the same way as it does in Tile Builder.



That is all that you need to do to get your SmartGrid up and running. You can now use the applet to control your devices, and the on-screen version will update automatically when changes are detected. To access your SmartGrid from any device you simply need access to the Endpoint links which can be found in the Endpoint section as shown below.



By default, only the Local Endpoint is enabled. You can click on the **Local Endpoint** label or simply copy the link and share it or store it as you see fit.

How Updates Work?

SmartGrid gets its updates in one of three ways.

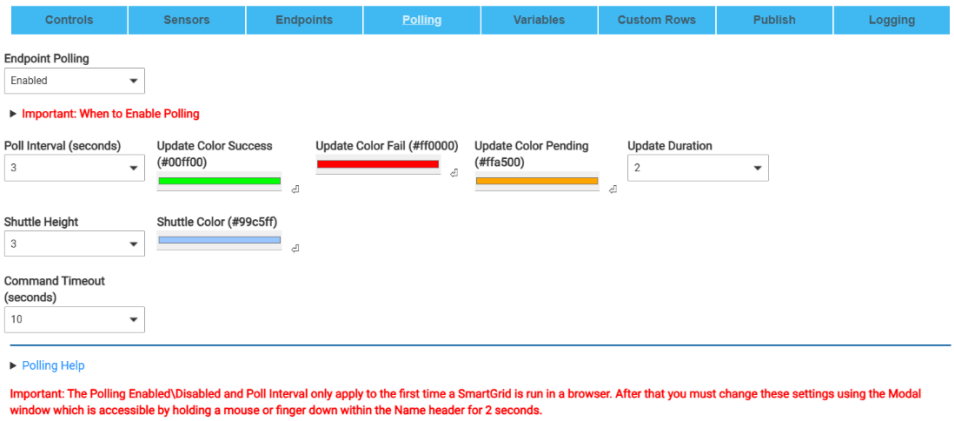
1) You can perform a manual refresh of the device states at any time by clicking on the Refresh icon in the far right of the Name column.
2) SmartGrid can poll the Hub and ask, “Got any changes” to which the Hub will respond “No” or “Yes and here they are.” Polling can be Enabled\Disabled, and the polling frequency can be changed in a combination of the polling settings and the runtime settings (discussed later).
3) If you choose to **embed a SmartGrid into a Hubitat dashboard** then the SmartGrid will be refreshed every time that the underlying data changes, but ONLY if the Event Timeout is set to a value other than NEVER (default).

While there are three methods only methods 2 and 3 are completely automatic so we will focus on those. The rule of thumb is if you are using the SmartGrid within a Hubitat dashboard then you would disable Polling and configure a small **Event Timeout** period. If you plan to launch the SmartGrid mostly from a browser on a phone, tablet, or desktop then you would enable polling, configure an appropriate polling period, and then set the **Event Timeout** to “**Never.**”

Of course, if you plan to use SmartGrid simultaneously in both scenarios then you would leave both methods enabled. The **Event Timeout** method is the same one used in Tile Builder so I will skip over that and talk about polling because that is new.

Polling

The polling configuration interface looks like this:



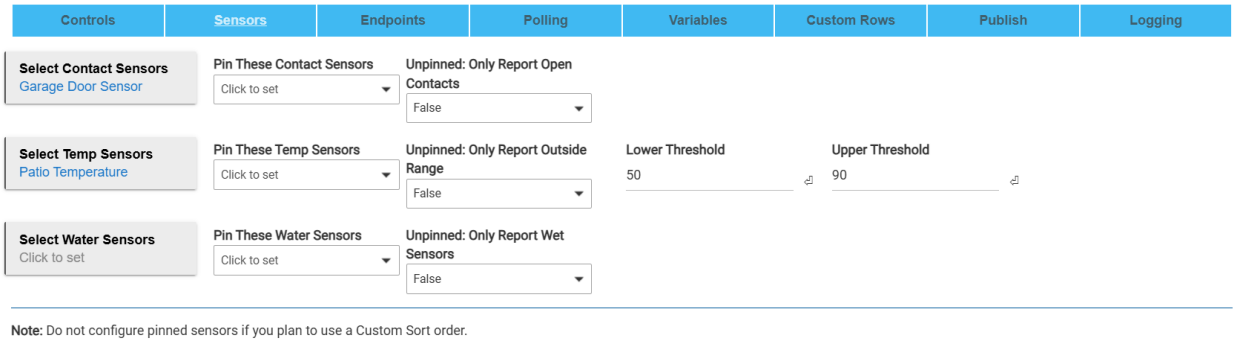
You will find each of these properties explained in the inline Polling Help which I will repeat here:

You can configure the SmartGrid to poll the endpoint and apply any changes that are found. If there are no changes the SmartGrid goes back to sleep until the next poll interval.
Poll Interval: The frequency at which the Hub will be contacted to ask if there are any updates available.
Poll Update Success Color: When updates are applied the Grid will be outlined in the selected color.
Poll Update Failure Color: When updates are requested but no changes are received within the command timeout period the Grid will be outlined in the selected color.
Poll Update Width: This setting has been deprecated. The width of the highlight color will be the same as the table outer border width.
Poll Update Duration: The duration in seconds that the Success\Failure outline is displayed.
Shuttle: In polling mode the Shuttle is displayed at the base of the SmartGrid as a visual indicator of the polling process. When the bar hits either edge then a polling event will occur and any changes will be picked up.
Shuttle Bar Height: The height of the bar at the base of the SmartGrid that identifies the position in the polling cycle.
Shuttle Bar Color: The color of the bar at the base of the SmartGrid that identifies the position in the polling cycle.
Command Timeout: The amount of time allowed to pass without a response from the Hub before a request is deemed to have failed.
When the polling process discovers an update is pending then the SmartGrid is refreshed and the table is outlined for **X** seconds using the Poll Update Duration value configured above.
Note: You can initiate a full refresh of the table at any time regardless of the polling interval using the Refresh Icon ↻.
Important: The Polling Enabled\Disabled and Poll Interval only apply to the first time a SmartGrid is run in a browser. After that you must change these settings using the Modal window which is accessible by holding a mouse or finger down within the Name header for 2 seconds.

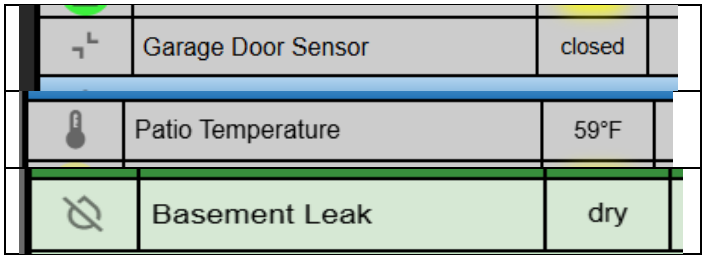
You should now be able to create a SmartGrid and configure access according to your own needs.

Sensors

SmartGrid also supports three sensor types and others will be added soon.

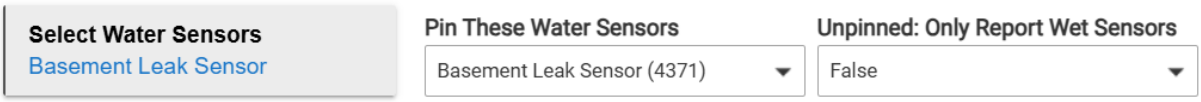


Sensors will show up in the SmartGrid like this with the value in the State column:

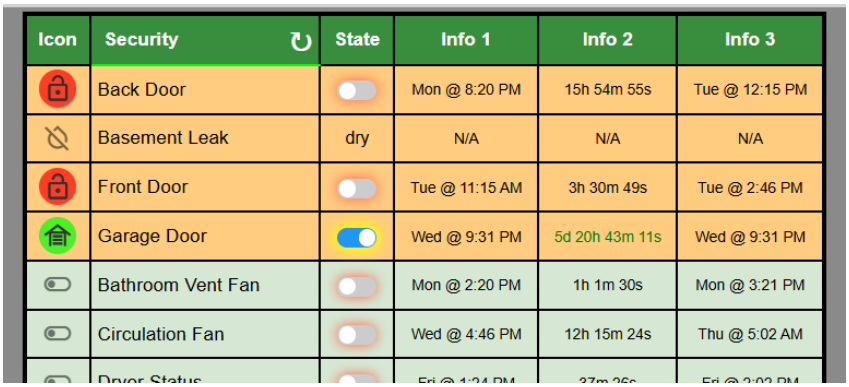


Pinned Entries

A pinned entry is one that will always remain at the top of the list regardless of the sort order. This is used to elevate the visibility of controls and variables that are especially important. You can choose to pin controls or sensors on the appropriate tables.



Pinned entries will then appear at the very top of the SmartGrid as shown below, regardless of the state of the device or the sort order of the grid.



Note: Pinned entries are disabled when a Custom Sort order is configured.

Device Renaming

If you have used Tile Builder you will already be familiar with this capability which can be used to improve the readability of names by removing characters, words etc that you don't want to appear.

But a useful technique you might find useful is to use device renaming in conjunction with Variables. This lets you display the value of Variables within the device label column.

In my case I have some Automatic fans that run based on temperature differentials. Normally the output would look like this.

	Auto Fans		Last On	Run-Time	Last Off
	Attic Vent Fan		Thu @ 5:33 PM	24m 59s	Thu @ 5:58 PM
	Basement Circulation Fan		Sun @ 2:45 PM	4d 19h 3m 24s	Fri @ 9:48 AM
	Bathroom Vent Fan		Thu @ 1:36 PM	6m 4s	Thu @ 1:42 PM

By adding some device renaming strings like this for my Attic Fan and Circulation Fans I can include some information about the current temperatures like this:

Search Device Text #5
Attic Vent Fan

Replace Device Text #5
Attic Vent Fan[br][mark]%var24%°F ⇄ %var31%°F

Search Device Text #6
Basement Circulation Fan

Replace Device Text #6
Basement Circ Fan[br][mark]%var41%°F ⇄ %var81%°F

The net result is that my display now looks like this and lets me check if it's working as expected.

	Auto Fans		Last On	Run-Time	Last Off
	Attic Vent Fan 51°F ⇄ 65°F		Thu @ 5:33 PM	24m 59s	Thu @ 5:58 PM
	Basement Circ Fan 64°F ⇄ 72°F		Sun @ 2:45 PM	4d 19h 3m 24s	Fri @ 9:48 AM
	Bathroom Vent Fan		Thu @ 1:36 PM	6m 4s	Thu @ 1:42 PM

The [br] tag stands for a break which causes it to go to the next line.

The [mark] tag highlights anything with a yellow background by default.

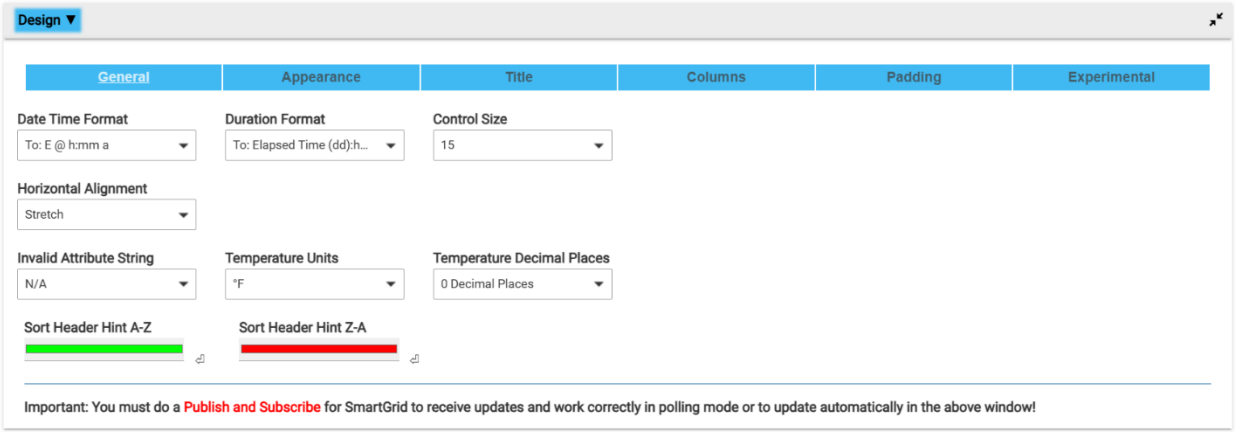
You typically do not need to include closing HTML tags as most are automatically closed when the table cell ends.

Note: You can customize the [mark] tag using the settings on the experimental tab.

Designing a SmartGrid

In the bottom part of the SmartGrid module all the configuration settings that control the appearance of SmartGrid are organized into five main groups. These are described in the following paragraphs.

General Tab



I will go through each group and describe the nature of the settings.

Date Time Settings: These control the appearance of the data in the Info Columns.

Control Size: This is an important setting as it sets the size of the various controls. On a touch device you may find some of the controls to be a little too small for comfort. This can be changed here.

Horizontal Alignment: This controls how a SmartGrid is displayed within an iFrame. The default is **Stretch** but that may not be ideal for every situation.

Invalid Attribute String: The Info columns allow you to display certain types of information about the devices in the SmartGrid. For example, you may have a SmartGrid of devices and have an info column to display the power being used for each device. Devices that do not support the power attribute would display N/A or whichever selection you make.

Temperature Units and Temperature Decimal Places: These control how any temperatures are display within the SmartGrid.

Sort Header Hint: Most SmartGrid column headers function as sort buttons. Sorting occurs on a simple alphabetic basis and the selected column header will be underlined by the selected color to indicate the sort direction.

Appearance Tab

The appearance tab groups together most of the settings that relate to text size, text color, background color and background opacity in some cases.

Most of these are self-explanatory but I will call out a few settings.

Custom Row Background Color #2: This allows the separator rows to be configured with a gradient for improved separation and appearance like this.

If you just wish a solid color, just use the color picker to select the color from the first control.

Device Row Properties: This controls the appearance of Custom Rows that are defined as a Device Row.

Highlight Selected Rows: This selects whether active\selected rows are highlighted.

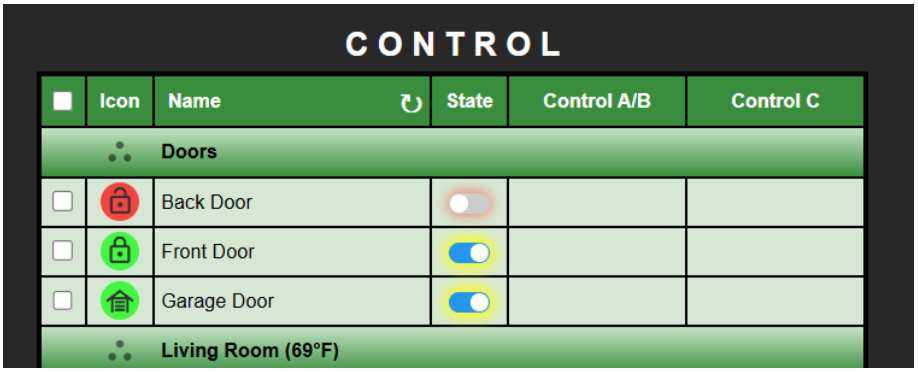
Highlight Pinned Rows: This selects whether pinned rows are highlighted using a distinct color.

Border Color: Controls the settings for the SmartGrid border.

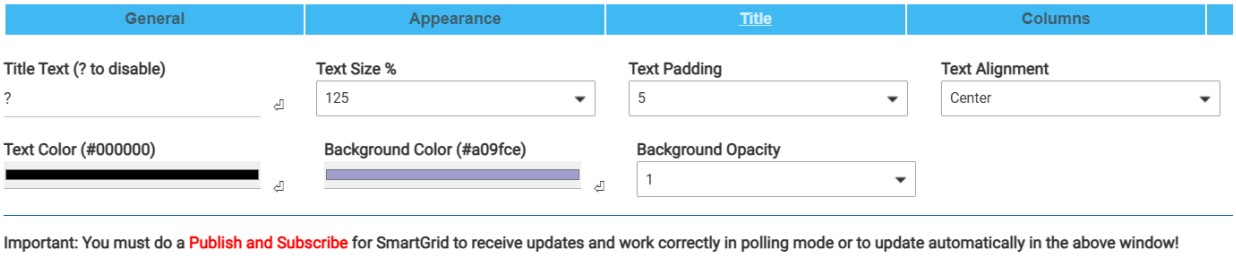
Select Color Theme: Applies a group of preset colors to the current configuration.

Title Tab

You may opt to enable a title on a SmartGrid such as the one shown below.

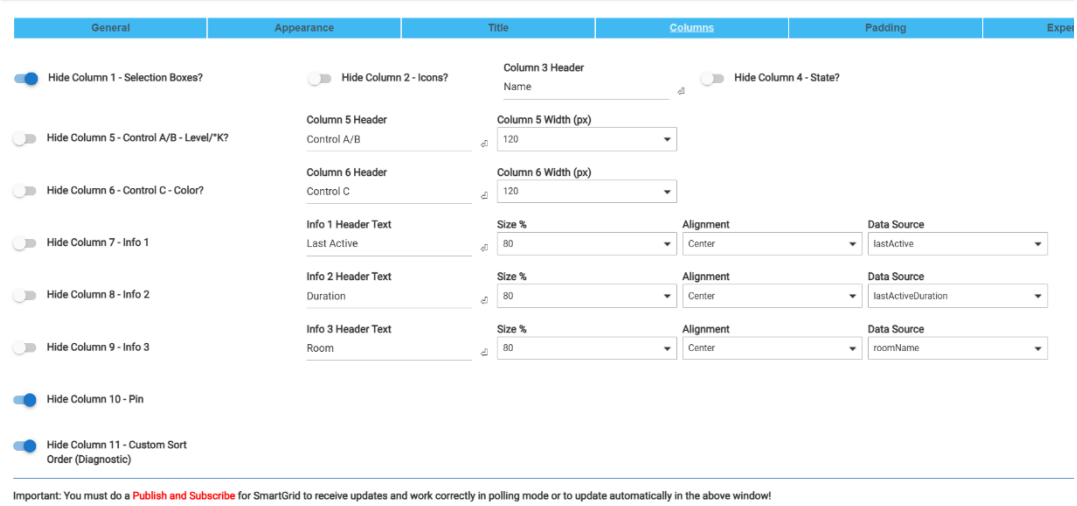


If you choose to enable a Title, all the settings controlling its appearance are on this tab.



Columns Tab

The columns' options look like this:



You can opt to show\hide any column except for column 3, the device name column.

You can configure the title on the header bar for columns 3, 5, 6, 7, 8, 9. You can place HTML within these text entry fields if you wish to make a header bold or underline for example.

You can specify the width for columns 5 and 6 which hold controls and allows the size of things like slider bars to be an appropriate size.

Info Columns

You can configure your SmartGrid to display additional information about a device in one of three Info columns. Available choices are:

Data Source Name	Description
ID	The Hubitat device ID. This is typically a 4-digit number.
battery	The battery level for battery operated devices.
colorMode	The mode that a RGBW bulb is operating in. This can be RGB, CT or EFFECTS.
colorName	The nearest English equivalent name for a bulb color.
colorTemperature	The color temperature in Kelvin for a bulb in CT mode.
deviceTypeName	This is the English name for how a device is recognized such as switch, dimmer, fan etc.
Energy	The is the value of the energy attribute of a device usually in kWh.
healthStatus	The value of the healthStatus attribute of a device, usually online or offline.
lastActive	This is the date\time that the device was last placed into an active mode. This has a different meaning depending on the device. Switch = Last On event Valve = Last Open event Lock = Last Locked event Fan = Last Not Off event Garage Door = Last Not Closed event Shades = Last Open event Contact Sensor = Last Open Event Water Sensor = Last Wet event

lastActiveDuration	If the device is currently active it is the elapsed time from the lastActiveEvent until the present moment (text will be in green). If the device is currently inactive it is the elapsed time from the lastActive event until the lastInactive event.
lastInactive	The inverse state of lastActive. Switch = Last Off event, Valve = Last Closed Event etc.
lastInactiveDuration	If the device is currently inactive it is the elapsed time from the lastInactiveEvent until the present moment (text will be in black). If the device is currently active it is the elapsed time from the lastInactive event until the lastActive event.
lastSeen	This is the date time value of the devices lastActivityAt attribute. This will be formatted with the selected default date time format.
lastSeenElapsed	How much time has passed since this device last had activity.
network	Lists the network by which a device is connected to the Hub. Options are Z-Wave, Zigbee, LAN, Virtual, Other.
power	The current value of the power attribute for a device is usually in Watts.
roomName	The name of the Hubitat room that the device is assigned to.
temperature	The current value of the temperature attribute for a device if present. Temperature will be formatted according to the settings on the General tab.

You can use any combination of attributes across the three info columns.

Columns 10 and 11 are used for diagnostic purposes and should normally remain hidden.

Padding Tab

Padding is used to increase or decrease the horizontal and vertical space around elements in the table and provide a well-balanced appearance.

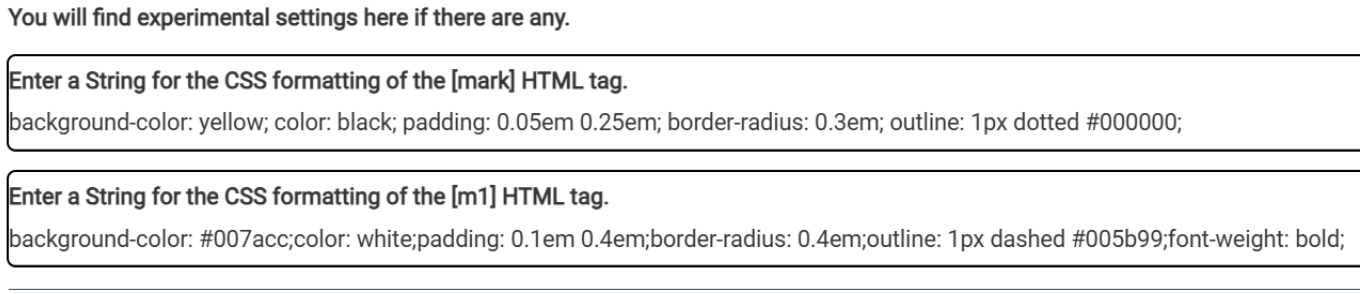
If you want to adjust a SmartGrid to fit a specific space, adjusting these values will give you the best result. Because SmartGrids have many rows adding even a single pixel of Vertical padding can result in a SmartGrid growing vertically by 40 - 60 pixels because padding happens above and below each row.

To provide finer tuning of this you can use the **Row Vertical Padding – Minor** setting**.** You can consider the minor padding to be a decimal place in addition to the value of the **Row Vertical Padding- Major**.

Experimental Tab

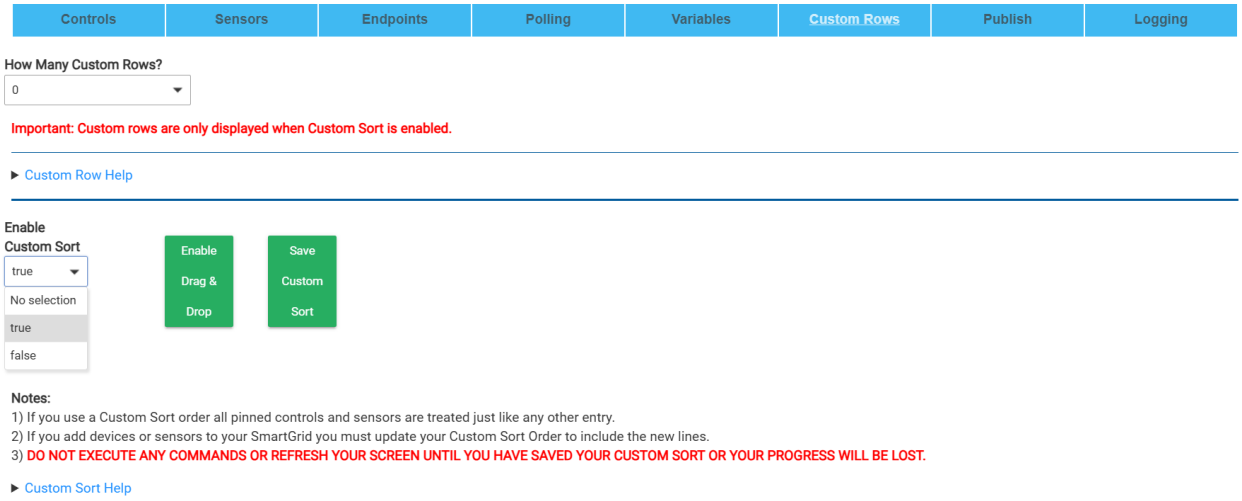
This tab is used for introducing new features\settings for testing purposes that can be assessed without disturbing the UI layout for a change that may not make it into the final product.

Currently there are experimental settings for the customization of the [mark] tag as well as an experimental [m1] tag that can be used as desired.



Custom Sort and Custom Rows

Custom Rows can be used to customize a table and improve its appearance or provide additional information. Custom rows are only display when a SmartGrid is configured with a Custom Sort so we will start there.



Enabling Custom Sort

A Custom Sort lets you arrange all the lines in a SmartGrid in whatever order you choose, and it will always remain that way. In Custom Sort configuration the headers no longer function as sorting mechanism.

To enable Custom Sort, go to the **Custom Rows** tab and turn on **Enable Custom Sort.** This will make any Custom Rows visible within the SmartGrid so that we can sort them later. It also disables polling until the

Add Custom Rows

To add **Custom Rows,** change the dialog below to the number that you want. There are two types of Custom Row **A) Separator Row** and **B) Device Row**.

Separator Rows: These are used to break up a SmartGrid into visual subgroups. This allows one table to have different functional areas for clarity. For example, by room, by functionality or by another organizing principle. The configuration of a separator Row looks like this:

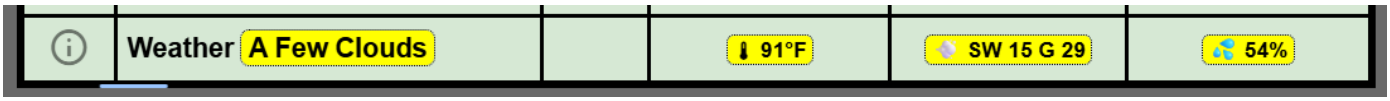
And the resulting Separator row looks like this:

Collapsing Groups: You can temporarily hide groups of information for details that you might only wish to see occasionally by clicking on the three dots icon. Click again and it will revert to the previous state.

Device Rows: These are used to place additional information within the SmartGrid that is not otherwise able to be present because the device type or attribute is not supported. It uses all the same fields, but is formatted to appear like other device rows, with the exception that the device icon which is the information symbol ⓘ.

The Device Row is configured like this to place data within the table.

And it looks like this for this instance (including variables).



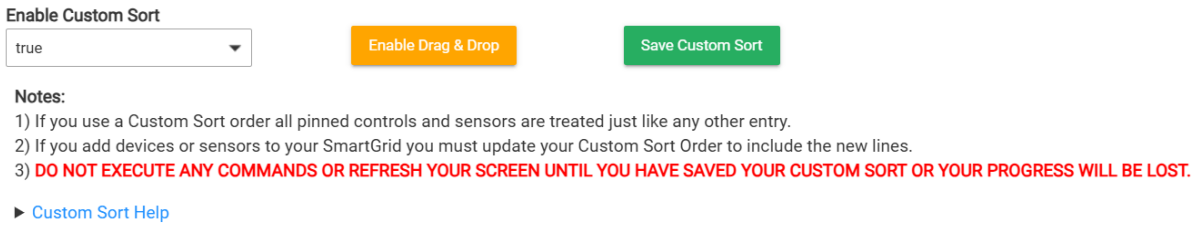
You can add whatever text you wish in these fields including HTML tags as you see fit. Within those fields you will see %var21% and %var22%. These are assignable variables and are covered in the next section.

You can configure up to 10 Custom Rows in total, in any mix of Separator and Device Rows. You can also choose to disable a Row rather than remove it completely as shown below.

Creating and placing variables is covered in the next section.

How to Configure a Manual Sort

Go to the **Custom Rows** tab. Click on **Enable Drag & Drop** and the button will turn orange to show that Drag and Drop are active.



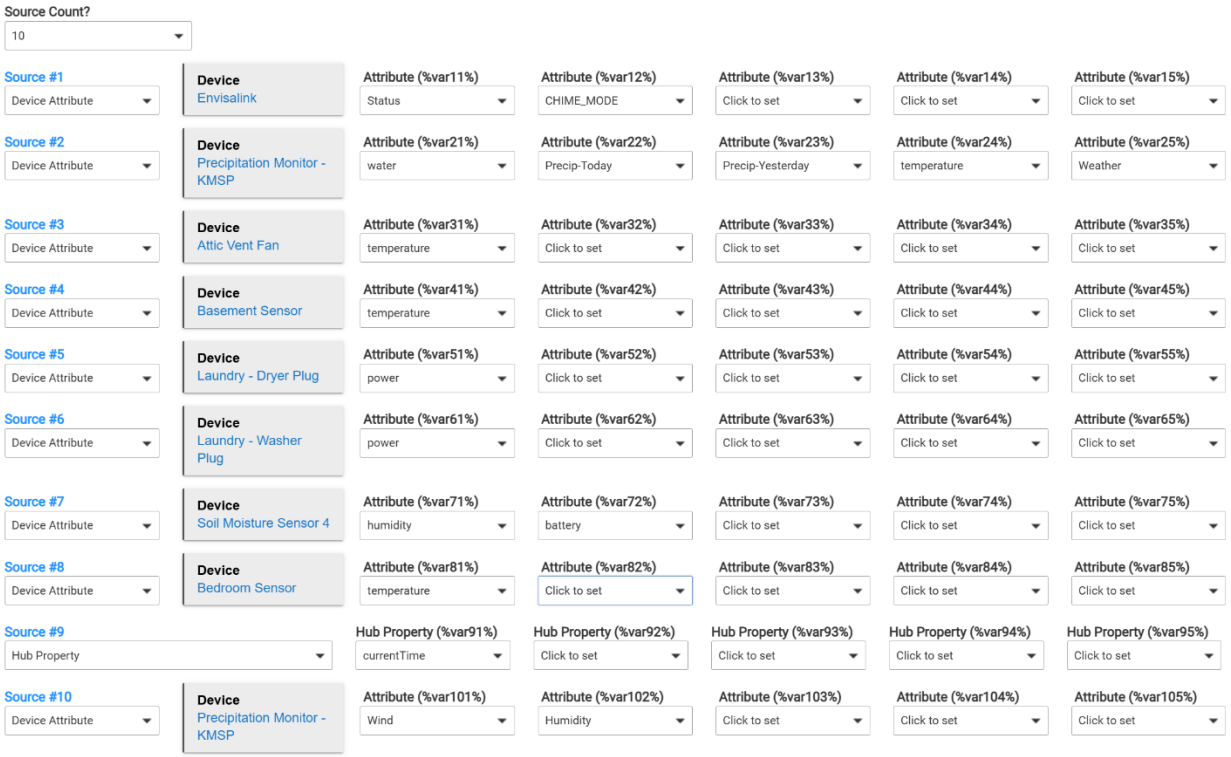
Now when you move the mouse over the SmartGrid you will get the grabber hand icon and you can drag and drop the rows into the order that you want. When you are finished click on the **Save Custom Sort** button and the **Enable Drag & Drop** button will change from orange back to green.

Important: When Drag and Drop mode is active, Polling is automatically suspended to prevent the table from accidentally refreshing. It is restarted when the **Save Custom Sort** is executed. If your SmartGrid is not updating and\or polling is not working, then this is probably the cause.

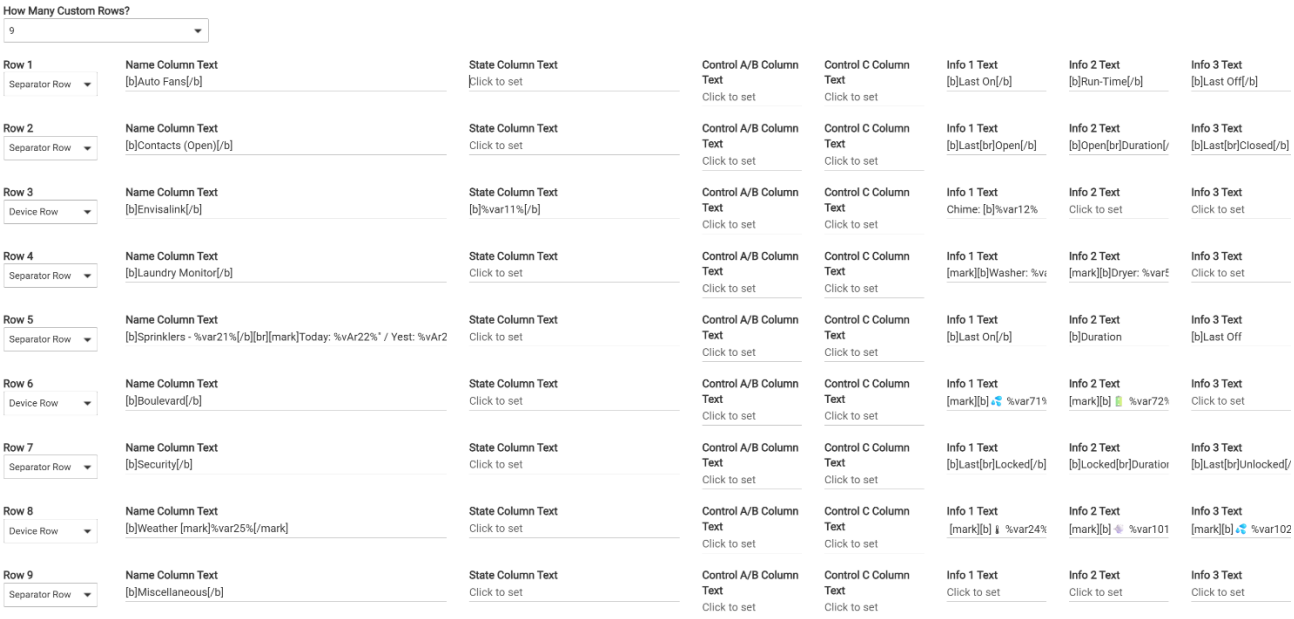
Variables

You can assign the value of any Device attribute, Hub variable or Hub Property to a variable and access that variable by using the %varXX% syntax within any of the Separator or Device Rows as shown in the previous example. Variables can also be placed in the Device Name column using Device Renaming.

To create the variables simply go to the Variables tab, select the number of variables that you with to enable (maximum of 10 sources) and up to 5 variables per source for devices.



You can now place the %varXX% variables anywhere within the Separator or Device Rows like this:



Variables Decimal Places

The number of decimal places is determined by the capitalization of the variable name as described below:
%var22% = 0 decimal places (integer)
%Var22% = 1 decimal place

%vAr22% = 2 decimal places
%vaR22% = 3 decimal places

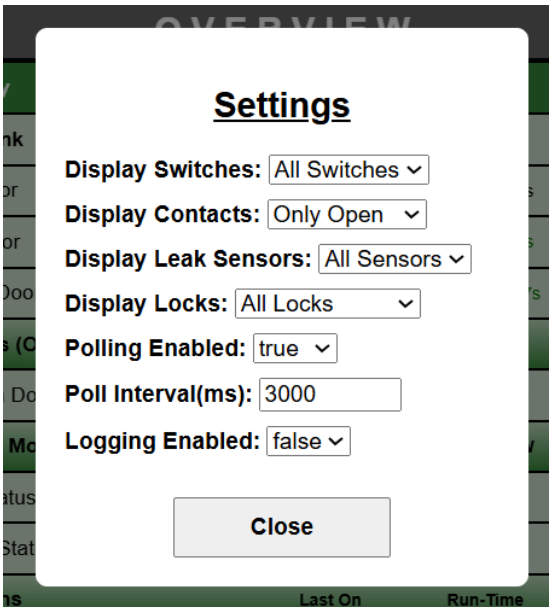
To place variable values within the Device column, use the **Device Renaming** feature previously discussed.

The result of all these configuration choices looks like this:

Run-Time Settings

With SmartGrid version 4 there are now seven controls that can be configured at Run-Time. This means you can configure one SmartGrid and then have it appear differently on multiple devices using these settings that are saved locally to the device.

These settings are available via Modal popup which is accessed by doing a 2-second-long press\click on the Name column header. It looks like this when it pops up.



Using this popup, you can elect to filter out devices in a certain state. A good example is contacts, rather than listing the state of all the contacts in the house you can choose to only list those which are open and create a much more concise SmartGrid.

You can also enable\disable polling and change the interval using this interface and these are saved as local settings on the device. When a SmartGrid is first launched on a device it uses the settings configured in the SmartGrid configurator for polling, **after this initial load polling can only be modified using Run-Time control** until such time as the local app settings are removed.

SmartGrid Examples

Here are a few snippets from my own personal SmartGrids.

I can quickly see which doors and windows are open around the house using the run-time filters.

Contacts (Open)			Last Open	Open Duration	Last Closed
	Sunroom Door	open	Wed @ 6:19 AM	5h 15m 14s	Tue @ 9:41 PM

Laundry: I can monitor the Laundry to see if the Washer and Dryer are still running, how much power they are using at any given moment and how long the last\current cycle took.

Laundry Monitor			Washer: 0W	Dryer: 0W	
	Dryer Status		Fri @ 7:30 AM	53m 1s	Fri @ 8:23 AM
	Washer Status		Fri @ 6:23 AM	1h 5m 59s	Fri @ 7:29 AM

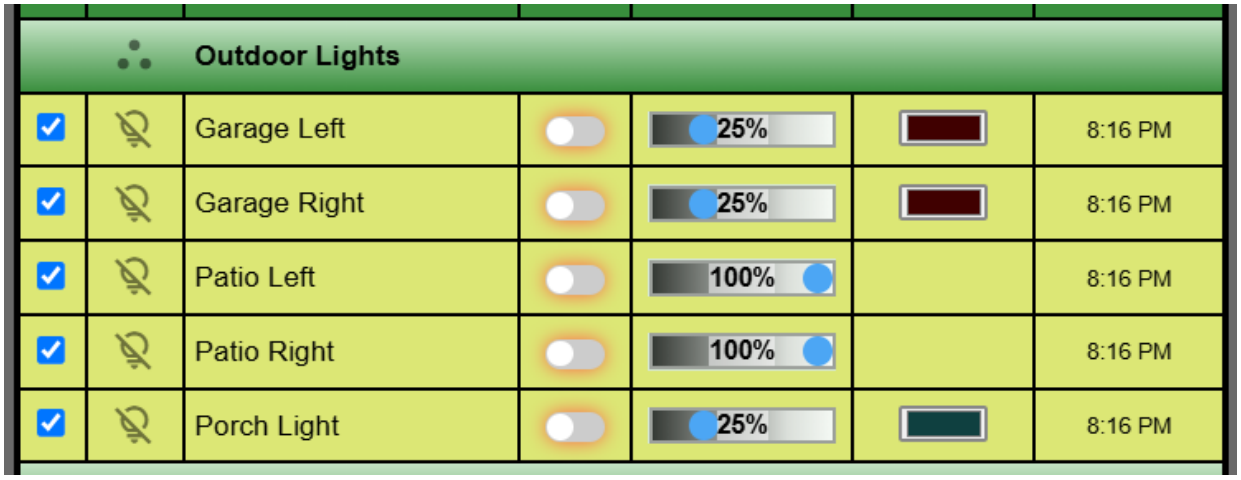
Automatic Fans: I can monitor these automations around the house or force them on\off if I wish to.

Auto Fans			Last On	Run-Time	Last Off
	Attic Vent Fan 91°F ⇄ 114°F (±15°F)		Sat @ 9:59 AM	4h 39m 58s	Fri @ 6:54 PM
	Basement Circ Fan 65°F ⇄ 79°F (±10°F)		Wed @ 11:53 AM	3d 2h 46m 47s	Wed @ 10:12 AM
	Bathroom Vent Fan		Sat @ 1:58 PM	7m 35s	Sat @ 2:06 PM

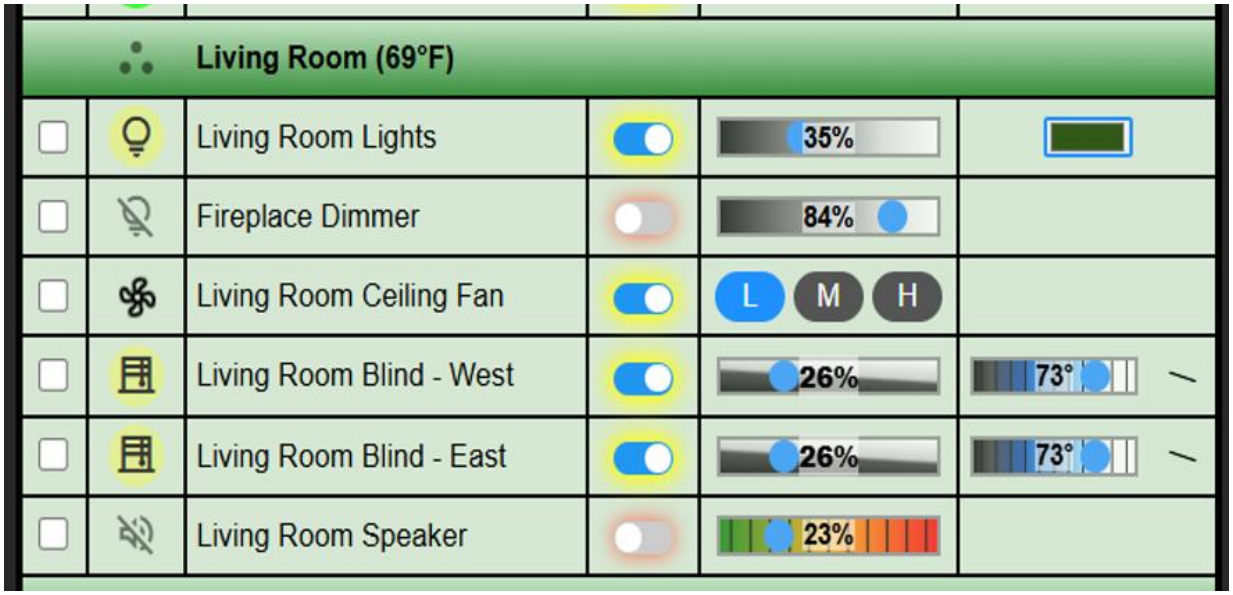
Sprinkler System: Monitor the sprinklers\precipitation or turn on a zone for ad-hoc watering.

Sprinklers - Dry Today: 0.00" / Yest: 0.02"			Last On	Duration	Last Off
	Sprinklers - Pause All Zones		Thu @ 11:46 AM	0m 43s	Thu @ 11:47 AM
	Z1 - Front Lawn North		Sat @ 4:00 AM	40m 0s	Sat @ 4:40 AM
	Z2 - Front Lawn South		Sat @ 5:00 AM	40m 0s	Sat @ 5:40 AM
	Z3 - Back Lawn		Sat @ 6:00 AM	10m 0s	Sat @ 6:10 AM
	Z4 - Flower Beds		Sat @ 7:00 AM	29m 59s	Sat @ 7:30 AM
	Z5 - Patio West		Sat @ 8:00 AM	20m 0s	Sat @ 8:20 AM
	Z6 - Patio East		Sat @ 8:30 AM	10m 51s	Sat @ 8:40 AM

I can check my outdoor lights are following their schedule and modify any number of them at the same time.

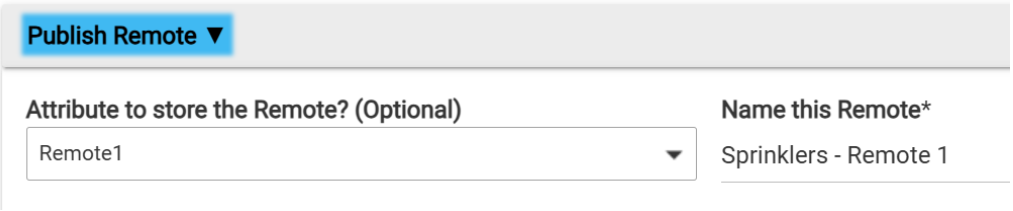


I can group all my commonly used devices together by room for easy access.



Publishing a SmartGrid

With Remote Builder tiles publishing is only required if you are going to place the SmartGrid onto a Hubitat Dashboard. To publish to a dashboard simply select an attribute to store the link and give the remote a name as shown below and click on the **Publish and Subscribe** button.



When a remote is published onto the dashboard it will automatically refresh when any of the values on the table changes. If the table itself changes then you must force a refresh of the table or the browser for the recent changes to be downloaded.

Event Timeout

You can adjust the refresh frequency by changing the **Event Timeout** period to something that matches your personal preferences. The default value for SmartGrid is **Never** so it relies on polling to detect data changes.

If you are using SmartGrid within a Hubitat dashboard you could disable the SmartGrid polling and change the Event Timeout to 1-2 seconds and the table will do a complete refresh when any value changes. This tends to look choppier with larger tables because the whole table is doing a refresh vs just the content of the table which happens using polling.

The Event Timeout setting has no bearing on an Endpoint loaded within a browser (not using a Hubitat dashboard), and these must be refreshed either manually or using polling.

Understanding Local Storage

When a SmartGrid is first launched within a browser it saves a small group of default settings within the local browser storage that looks like this:

5b9f-A_filterContact	onlyOpen
5b9f-A_filterLeak	allLeak
5b9f-A_filterLock	allLock
5b9f-A_filterSwitch	allSwitch
5b9f-A_isLogging	false
5b9f-A_isPolling	true
5b9f-A_pollInterval	3000

The names come from the last four digits of the access token (5b9f) with a -A_ -P_ or -B_ appended to it, followed by the settings in question.

- The prefix -P refers to the instance of Preview of the SmartGrid within the designer.
- The prefix -A refers to configuration A, the first of two possible configurations.
- The prefix -B refers to configuration B, the second of two possible configurations.

This may seem a little complicated, but it is necessary to allow two instances of the same SmartGrid to operate independently within the browser and for each instance to have its own independent settings.

Using A\B Configuration

If you only have a single instance of a SmartGrid per device then you can ignore this entire section.

However, if you wished to publish two instances of a SmartGrid onto a Dashboard, each with different filter settings, then you would need to use an A and B configuration to work correctly.

If you wish to have and A and B configuration, choose to publish your SmartGrid to Remote21 – Remote25. When you do that, it will automatically publish two separate configurations in the form:

Remote21-Cloud-A	Remote21-Cloud-B
Remote21-Local-A	Remote21-Local-B

To have the same SmartGrid loaded twice on a Dashboard you can choose Local-A and then Local-B from the list of attributes. You will then see the same SmartGrid twice and you can configure the Run-Time settings by doing a long press on the Name column header. Now you will have two instances of the same SmartGrid, each running with their own independent settings.

If you are running the same SmartGrid on multiple devices, computer, tablet, phone etc. then it does not matter if they all run the same Endpoint because the Run-Time settings are automatically separated by virtue of them being on different physical devices.

It is a Wrap

Well, if you made it this far you are ready to build your own SmartGrids and get the most out of them. I look forward to seeing the designs that people produce and share on community forums.

This is the day the Lord has made, let us rejoice and be glad.