



## Custom Info

### 1. Intro

Like you already know, Control4 has an icon-based user experience. That is pretty good in most cases, but sometimes we need to be able to display information in text format or be able to concentrate several pieces of information in one page, like a resume page. This webview driver was created to fill that gap.

#### Features:

- The driver provides the ability to create from 1 to 6 boxes.
- Each box has a custom title, and you can configure from 0 to 3 buttons to trigger Control4 actions.
- Each title can contain real time variables of the system that dynamically change the text to be displayed.
- Each button can contain text or an icon from the driver database.
- The driver grabs all the wallpapers of the project, and can be selected to be displayed as a background image on the page to improve the user experience.
- The icon of the driver can be changed.
- The driver provides backup export/import of the driver configuration (not the programming part), useful if the driver must be re added or create a copy.
- The driver supports “live UI updating”. This means that any change you make of the driver, you can see applied to any UI that has the driver opened.

#### Compatibility:

- T3/T4: Require OS 3.1.3+
- iOS and Android: Require OS 3.2+
- **OS 3.3.0+ is recommended**

### 2. Properties

- *Cloud Status*: Current status of the driver license.
- *Automatic Updates*: Enable driver automatics updates (Recommended!)

- *MAC*: MAC address of the director.
- *Driver*: Driver name.
- *Driver Version*: Current driver version.
- *Driver Information*: Last debug information.
- *Debug Mode*: If On, driver will print debug information on Lua output.
- *UI Icon*: With this property you can choose the icon suits you better.
- *Wallpaper*: The driver polls all the wallpapers that are being used in the project, and allows you to use any of them like a background image.
- *Buttons Background Color*: The boxes button can have black or gray background color.
- *Box Quantity*: From 1 to 6 boxes can be selected. Each box has a title and can also have buttons.
- *Box x Title*: The title of the x box. This text can contain real-time variables (you can use more than one per title) to shown (see “Using Real-Time variables” appendix for more info). TIP: You can use “<br>” or “<br/>” (without quotes) to specify a new line (enter).
- *Box x Button Quantity*: You can choose from 0 to 3 buttons to display in each box.
  - *0*: No buttons.
  - *1*: Button Center will be displayed.
  - *2*: Button Left and Right will be displayed.
  - *3*: Button Left, Center and Right will be displayed.
- *Box x Button Left/Center/Right Text or Icon*: For each button you can select if you want an icon or a text be display.
- *Box x Button Left/Center/Right Text*: For each button, if you selected “Text”, you can set the text you want to display. **Variables cannot be used here.**
- *Box x Button Left/Center/Right Icon*: For each button, if you selected “Icon”, you can set the icon you want to display.
- *Device/Proxy ID*: This property was added to improve finding a variable pattern. Insert the number of the device or proxy ID to be able to see all the variables associated to it.
- *Select Variable*: Here you can select one of all the variables of the previous ID. To be more helpful, the instant value of the variable was added to variable line.
- *Variable Pattern*: Once you selected the variable of the desired Device/Proxy ID, this property appears with the complete pattern. This pattern should be inserted on the right

property/properties so the driver replace this patter with the variable value in real time.

### 3. Actions

- *Update Wallpapers*: This will search again all the wallpapers available in the project. Notice that every time the driver reloads (director reboot) it will search again all the wallpapers.
- *Refresh Titles*: This will reload all the box titles and the monitored variables. This should be used only if support tells you to do so.
- *Print Backup*: This will print an encoded text with the configuration of all the properties of the driver. This is useful if you need to remove the driver but you want to reload all the parameters in a future. NOTE: the programming actions WILL NOT be saved.
- *Restore Backup*: This will load a backup.

### 4. Commands

N.A.

### 5. Events

- ❖ *Box x 1 Left Button Pressed*: It will be triggered when the left button of the x box is pressed. Be aware that Left button is present if 2 or 3 buttons are selected on “Box x Button Quantity” property.
- ❖ *Box x 2 Center Button Pressed*: It will be triggered when the center button of the x box is pressed. Be aware that Center button is present if 1 or 3 buttons are selected on “Box x Button Quantity” property.
- ❖ *Box x 3 Right Button Pressed*: It will be triggered when the right button of the x box is pressed. Be aware that Right button is present if 2 or 3 buttons are selected on “Box x Button Quantity” property.

**NOTE:** Be aware that if you program a button and then that button is eliminated because you changed the “Box x Button Quantity” or delete that box, the programming will be lost despite of added it again.

### 5. Driver Variables

- ❖ *LAST\_MENU\_SELECTED*: Last menu name that selected the driver.
- ❖ *LAST\_ROOM\_SELECTED*: Last ID of the room that selected the driver.

### 6. Using Real-Time variables

To use variables on the properties, we need the variables parameters. These are: deviceID and variableID.

With an update, the finding of the variable was improved. Here you have detailed the new method, and the legacy method (still works of course).

### NEW METHOD

With a recent update, some properties were added to find a desire variable way more faster than before.

- Variable Helper -

Device/Proxy ID	60
Device Name	Living Split
Select Variable	[1104] HVAC_MODE (value='Cool')
Variable Pattern	{60,1104}

For Composer Pro Created Variables use 100001

Insert this pattern on the property to utilize the 'selected' variable. For more information, refer to the documentation.

You only need to insert the Device/Proxy ID. Afterward, a 'Select Variable' property will appear, displaying a list of all the variables associated with that ID. To enhance the search process, an instant value for each variable has been added on each line for reference purposes only. When you select a variable, the corresponding pattern will be presented. This pattern should be inserted into properties that support real-time variables, and the driver will replace it with the correct value.

### LEGACY METHOD

To find them, we need to go to “Variables” Agent. If the variable is a custom variable (was created by the programmer), you will see it on the sheet. If the variable was created by the system (or another driver), you must check the box “Display System Variables”.

The screenshot shows the 'Variables' agent interface. On the left, the 'Agents' list includes 'Variables'. Below it, two red arrows point to the 'Variables' agent, labeled 'Custom Variables' and 'System Variables'. The main area displays a table of variables. At the top of this table, there is a checkbox labeled 'Display System Variables' which is checked. Below this, there are two tables. The top table lists variables with columns for Variable Name, Value, Type, Description, and Last Updated. The bottom table lists system variables with columns for System Variable Name, Device Name, Value, Type, and Description.

Variable Name	Value	Type	Description	Last Updated
cambio_ac2dh	False	Boolean		
CP-DiagnosticsIgnoredDevices	[]	String	Used for Composer Pro Diagnostics	
Doorbell		String		

System Variable Name	Device Name	Value	Type	Description
1001	Color	1Energieze23134630.3170970...	XML	
Active	Living->Spotify Connect	false	String	
Active	Main Room->Spotify Connect	false	String	
ANNOUNCEMENT_DISABLED	Main Room	False	Boolean	
ANNOUNCEMENT_DISABLED	Service	False	Boolean	
ANNOUNCEMENT_DISABLED	Rack	False	Boolean	
ANNOUNCEMENT_DISABLED	Living	False	Boolean	
ANNOUNCEMENT_DISABLED	Guest Room	False	Boolean	
AUXILIARY_HEAT_SOURCE	Main Room->Main Room Split	Not Used	String	
AUXILIARY_HEAT_SOURCE	Living->Living Split	Not Used	String	
AUXILIARY_HEAT_SOURCE	Guest Room->Guest Room Split	Not Used	String	
AWAKE_TIME	Living->System Remote Control SR260	255	Number	
AWAKE_TIME	Main Room->System Remote Control S...	255	Number	
BATTERY_LEVEL	Living->System Remote Control SR260	100	Number	
BATTERY_LEVEL	Main Room->System Remote Control S...	0	Number	
Brightness Percent	Living->Comedor	0	Number	Current light level (0% - 100%
Brightness Percent	Living->Lampara Pie	0	Number	Current light level (0% - 100%

Then, you must find the variable of interest, and then place the cursor above the variable to get the parameters:

LAST_MENU_SELECTED	Living->Disney +		String	
LAST_ROOM_SELECTED	Living->Door	273	Number	
STATE	Living->Door	Open	String	
LAST_MENU_SELECTED	Living->Door	security	String	
StateVerified	Living->Doorbell	True	Boolean	
ContactState	Living->Doorbell	False	Boolean	
ContactState	Living->Doorbell 2	False	Boolean	

STATE  
 Device ID: 215  
 Variable ID: 1001

In this case, deviceID=215 and variableID=1001.

To use it on the properties, we must use this format: **{xxx,yyyy}** where “xxx” is the deviceID of the variable, and “yyy” is the variableID.

In this case would be **{215,1001}**.

## 7. Box titles formatting

Unfortunately, Control4 driver properties don't allow a better way to bring format to the box titles text of the web-based driver. So, you must use HTML tags to do it. This is quite simple, here you have some samples lines to test in the Box x Title properties:

- `<b>This is Bold</b>`
- `<i>This is Italic</i>`
- `<b><i>This is Bold and Italic</i></b>`

Also use `<br>` or `<br/>` (both are equally valid) to make a new line in the box title.

It is recommended to use a combination of this formats to display a good UI.

A complex test could be:

- `<b><i>- House Consumption -</i></b><br><br>Main Circuit: 1235W<br>Secondary Circuit: 999W`

With this formatting and the use of variables, you can concentrate quite much information in one title with a good user experience. In the next images you can see these examples:

\*\*\*\*\* Box 1 \*\*\*\*\*

Box 1 Title

Box 1 Button Quantity

\*\*\*\*\* Box 2 \*\*\*\*\*

Box 2 Title

Box 2 Button Quantity

\*\*\*\*\* Box 3 \*\*\*\*\*

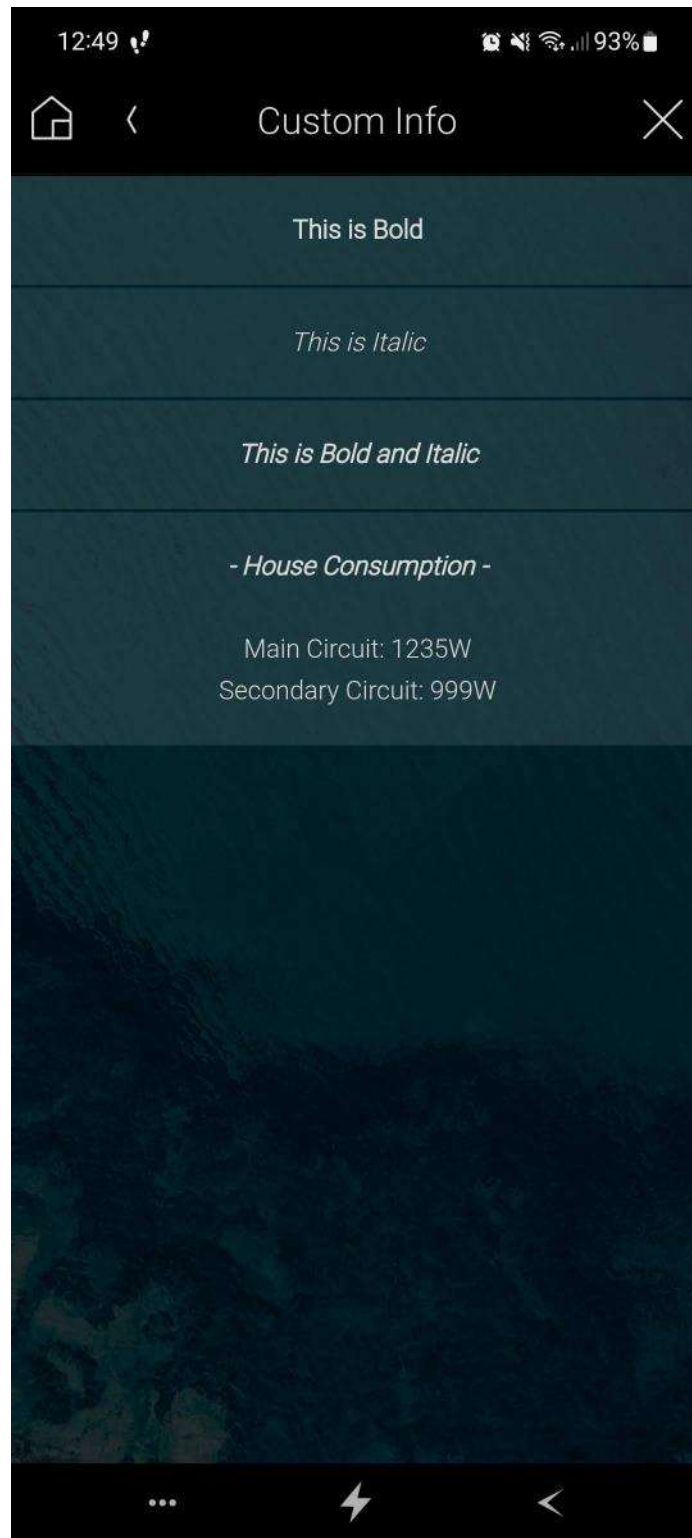
Box 3 Title

Box 3 Button Quantity

\*\*\*\*\* Box 4 \*\*\*\*\*

Box 4 Title

Box 4 Button Quantity



## UPDATE 05/Jan/24

A new function to handle decimals in float numbers was added to improve the variable display performance.

You can use, in the BOX TITLE, this function:

ROUND(FLOAT\_VARIABLE;DECIMALS\_PLACES)

Where:

FLOAT\_VARIABLE: Is the float variable in the format previously showed.

DECIMALS\_PLACES: An int value that indicates the number of decimals desired.

Example of a Box Title:

➤ ***- House Current -***  
Main Circuit: ROUND({435,1002};2) A.

## ***8. Examples***

This driver is so versatile, that shall be a lot of examples of application.

To better understand the use of variables and text formatting, the driver by default will have one box configuration with three buttons.

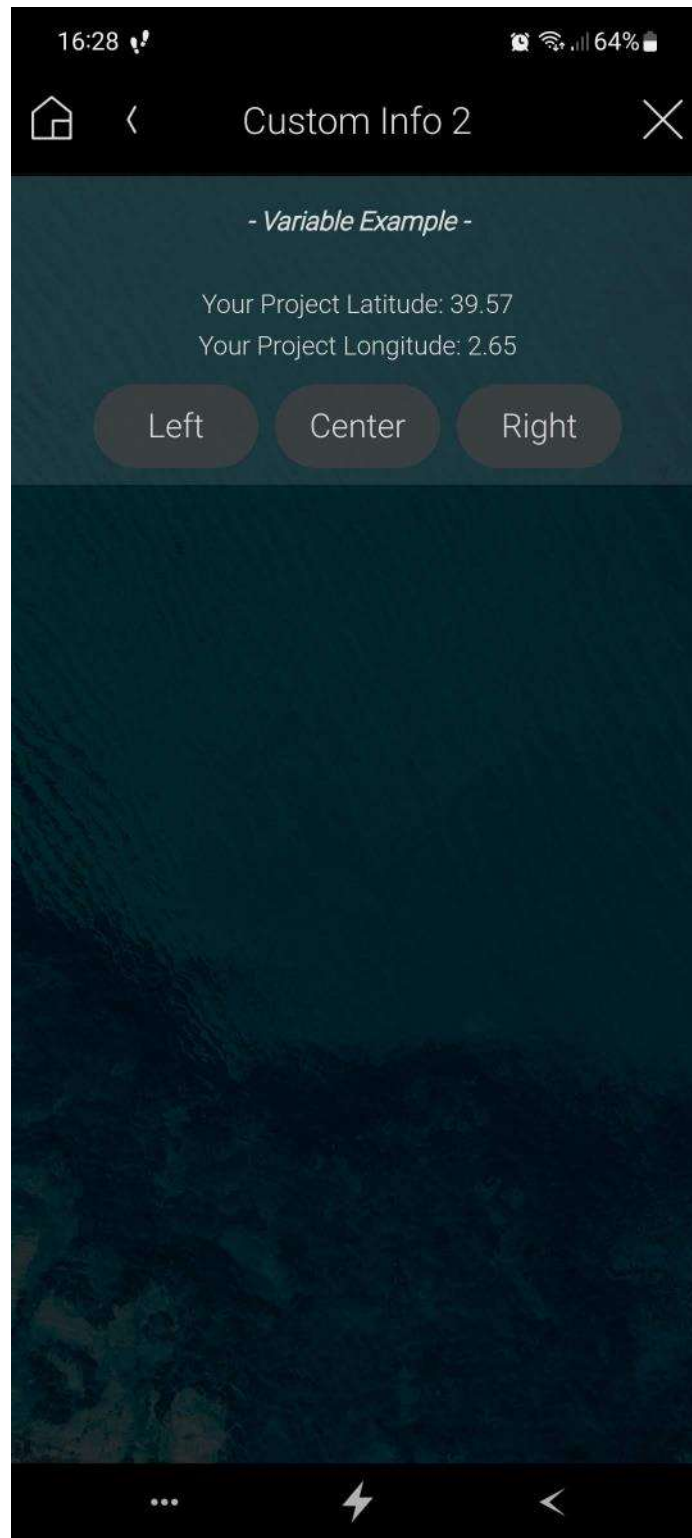
The box 1 title property will have this text:

➤ ***- Variable Example -***  
Your Project Latitude: {1,1001}  
Your Project Longitude: {1,1002}

Note that **{1,1001}** and **{1,1002}** are general variables of all projects that indicate project latitude and longitude respectively.

This should be the view of the driver opened for first time in the UI:





Some other examples of pages:

09:46

92%



# Showroom Info



## - Demo Light Scenes -



## - Showroom Shades -



Stop



## - Showroom Temps -

Living Temp: 22°

Lab Temp: 24°

Cinema Temp: 27.3°

Main Office Temp: 26.8°

Private Office Temp: 26.9°

## - KNX Status -

Connected

## - Last User Authenticated -

Juan Pablo - 10/10/22 09:33:41

## - System Online -

2d 13h:38m



12:54

93%



## System Info



### - Remotes Battery Levels -

Living Room: 98%  
Main Bedroom: 89%

### - WiFi Modules Signal -

Moon Lamp: 100%  
Living Temp Sensor: 100%  
Guest Bedroom Temp Sensor: 76%

### - House Current Power -

2543W

### - System Boot Time -

10d 16h:28m



13:01

92%



# House Climate



## - All House Temps -

Living: 25.6°C

Main Bedroom: 25.38°C

Guest Bedroom: 25°C

## - All House Modes -

Living: Off

Main Bedroom: Off

Guest Bedroom: Off

Off

Heat

Cool

## - All House Set Points -

Living: 25°C

Main Bedroom: 25°C

Guest Bedroom: 25°C

Set Global Set Point: 24°C

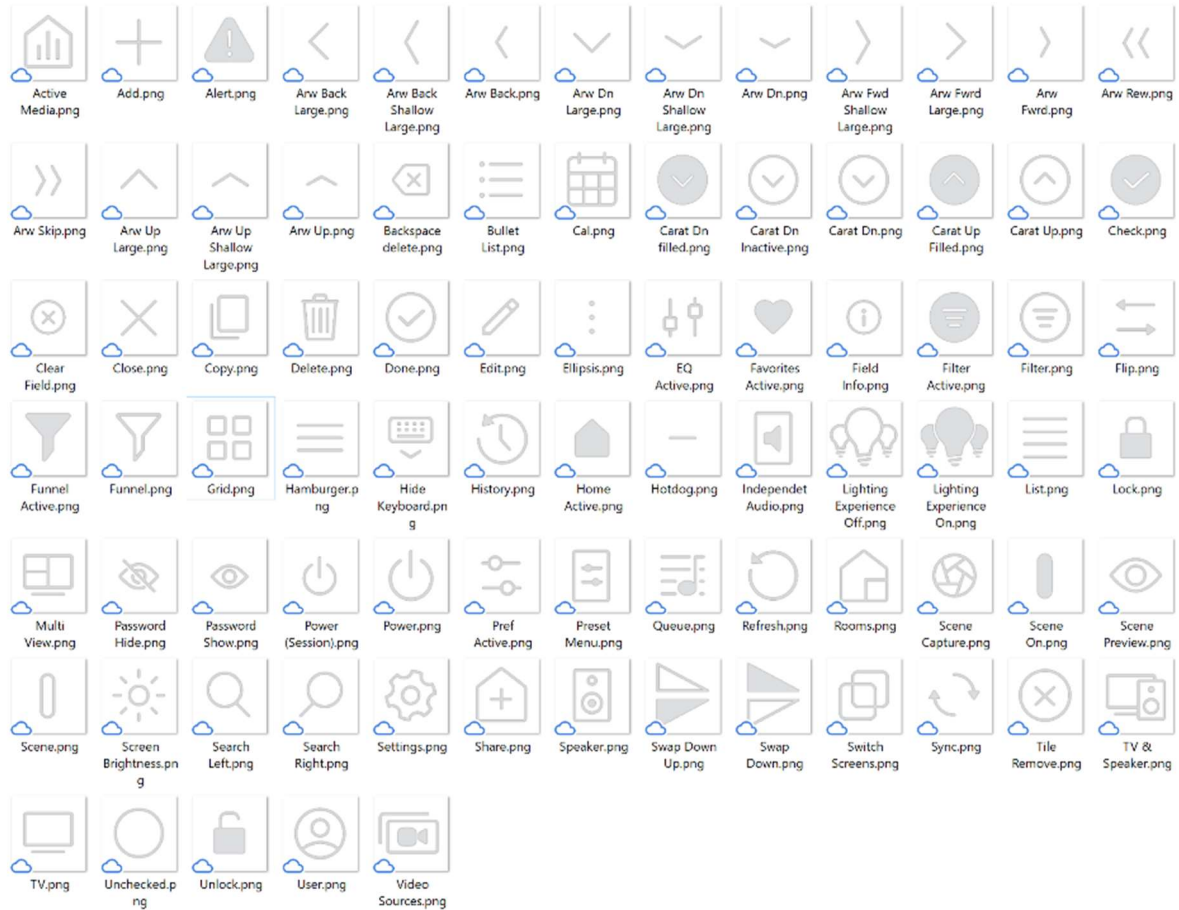


Set



## 9. Button icons

Here you can see the list of icons and their names, available to use in each button if the “icon” option was selected:



\*More icons could be added in a future without updating this documentation

## 10. Limitations

These are the current driver limitations:

- Sometimes, when you select the driver in the UI, a black background and a spinning wheel can appear for a couple of seconds before the page shows up. Apparently, in that cases the Control4 app take more time to load the web-page and that is why you must wait a little bit.

## 11. Warranty and disclaimer

Barbini.dev provide drivers tested as much as possible. However, due to the highly variants of systems setups and the constant changing of the electronics systems, updates and/or modifications may be required to fixing bugs or improve the driver functionality.

Many of our drivers interact with systems APIs (like Control4 and/or any other 3<sup>rd</sup> party system). Therefore, if any API changes, the driver may stop functioning correctly. **Barbini.dev** cannot guarantee long-term functionality of any driver developed that uses APIs. **Barbini.dev** has the right to repair, provide updates, or discontinue a driver at any time. These repairs or updates could be free or required additional expenses. Despite the above, our efforts will always be to maintain the drivers as long as possible and free of charge of upgrades as much as possible.

## ***12. Trial and Showroom Licenses***

All the drivers have 10 days of trial, allowing dealers to test the features of any driver in a customer project before purchasing it. The trial automatically begins when a driver is installed in a project.

We enable all the Showroom licenses to have unlimited access to the drivers.

**Developed By: Barbini.dev**

**Contact: [barbini.dev@highqualityautomation.com](mailto:barbini.dev@highqualityautomation.com)**



-----  
**22101202** - Initial Release

**22101701** - Improved black background

**22102001** - Fixed iOS button icons not loading

**22102102** - Fixed iOS wallpaper image not loading

**22102103** - Added more UI icons

**22111904** - Changed driver default with variable example, improve documentation, added more button icons, remove debug on timer, some other minor improvements.

**23082301** - Added an improvement for boolean variables. Now they are displayed as "True" or "False" rather than "1" or "0".

**230909xx** - A new efficient method for quickly locating variables was introduced, eliminating the need to leave the driver page.

**240105xx** – New Round function added to handle float variables decimals.

**240514xx** – Fixed bug in iOS devices not working properly with the driver if there was an apostrophe symbol inside a string. Additionally, other maintenance fixes were implemented.

**240514xx:** – Fixed bug in iOS devices not working properly with the driver if there was an apostrophe symbol inside a string.

**240516xx** – Fixed a bug where some buttons were not executing the event correctly if they were pressed on the borders.