



HCA Tech Note 714

HCA and Hubitat Communications

I receive a lot of questions about Hubitat and how HCA interacts with it, so I wanted to try and create definitive answers to those type of questions. The full picture is a bit involved but stick with this explanation and you will understand it all.

First let's start with an important term I use: "Manages".

When I say "Devices managed by Hubitat" I mean devices that you have added to the Hubitat list of devices using the Hubitat UI – the big "Add Devices" button in Hubitat.



Those devices typically are ZWave or Zigbee, but Hubitat can operate with some other device types as well.

When I say "Devices managed by HCA" I mean those devices you add to your HCA design using the "Device" button in the "New" panel in the "Design" ribbon category. Of course, HCA has other ways to add devices, but you get the idea.

Hubitat controlling an HCA managed device.

Now that is out of the way, let's start looking at the process by which Hubitat controls an HCA managed device. This is the mechanism that makes voice control of HCA devices and programs work because Alexa and Google Assistant talks to your Hubitat and Hubitat then talks to HCA to control the device.

This works this way: Hubitat wants to control a device, so it looks at the type of the device. Some device types are "built-in" to Hubitat so it always knows how to control them. Other device types are controlled by drivers that you add to Hubitat. As part of the setup described in [HCA Technical note #613](#), you added device drivers to Hubitat: One for dimmable HCA devices and another for non-dimmable HCA devices. You can see this by looking at the device list in Hubitat where it shows device types. For example, these devices are controlled using the HCA device drivers.

<input type="checkbox"/>	Kitchen Hallway	HCA Dim Object
<input type="checkbox"/>	Kitchen Lights	HCA Dim Object
<input type="checkbox"/>	Laundry Fan	HCA Non-Dim Object

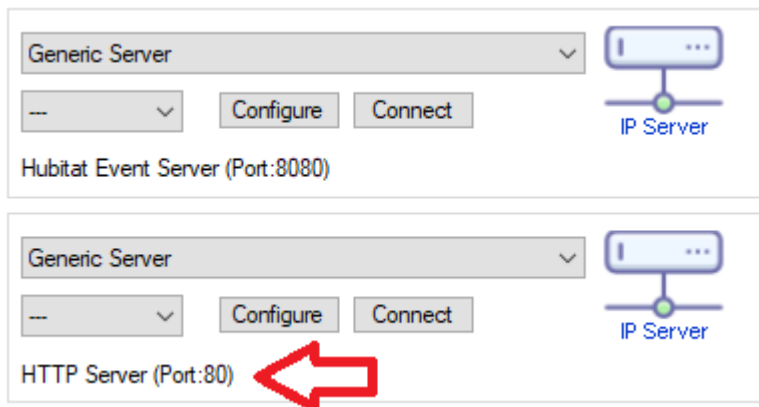
TechSup@HCATech.com

www.homecontrolassistant.com



HCA Tech Note 714

The Hubitat HCA device driver sends a message (a HTTP request) to HCA. HCA operates as a webserver and is listening on port 80 (the standard port for HTTP requests) using the “HTTP Server” interface you configured in HCA.



Those messages are processed by the program “HTTP Server – Receive HTTP”. Lots of information is passed between Hubitat and HCA for this to work, but you need not be concerned about it because the programs do all the work.

HCA controlling a Hubitat managed device.

Next, let’s look at HCA controlling a Hubitat managed device. Hubitat contains a facility where it, like HCA, operates as a webserver and processes HTTP requests sent by other applications to control devices it manages. This part of Hubitat is called the “Maker-API” and you added it to Hubitat as part of [HCA technical note #703](#). In the Hubitat UI, select “Apps”, then “Maker-API”, and then scroll down on that page. You can then see a description of those HTTP commands that it understands.

On the HCA side when you want to control a device that Hubitat manages you don’t need to do anything special just because it is a Hubitat managed device. You can right-click on the device and control it, use it in a schedule, or in a program. All the actual control work happens in one of the classes in the “Hubitat Elevation” package. Those programs send messages to Hubitat to carry out the action, and the Maker-API receives and handles it.

Keeping each side up to date about device states.

That’s the basic flow for commands to be sent for control. But there are more communication paths related to devices changing state. If a device changes state by the side that manages it, the other side must be notified of that state change. Since Hubitat knows about HCA devices and HCA knows about Hubitat devices, it is important that each maintain the correct state of those devices. Here’s an example of why that is important.

TechSup@HCATech.com

www.homecontrolassistant.com



HCA Tech Note 714

Suppose you have a device called “Kitchen Light” that HCA manages and it is also a device that Hubitat knows about. If you have it scheduled to go on at 6am then the device goes on because HCA controls. Why should Hubitat be told? Suppose you want to use the programming facilities in Hubitat and what you create there performs different actions based upon the state of this device. Unless Hubitat knows the true state of the device (is it on or off?) your Hubitat program could operate incorrectly.

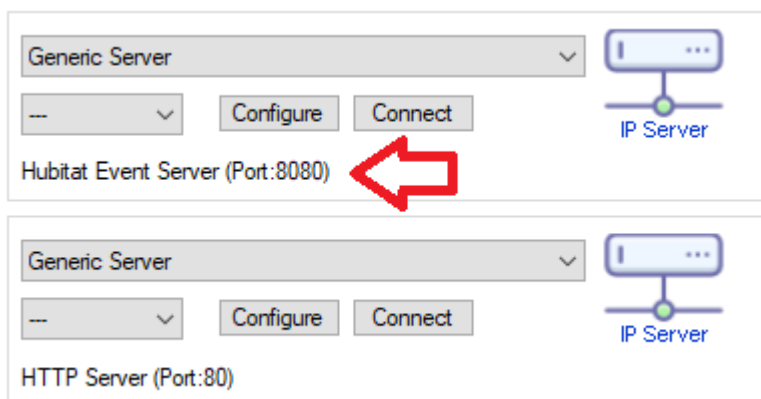
HCA updating Hubitat when a device HCA manages, and Hubitat knows about, changes state.

Any device that is managed by HCA, and is known to Hubitat, when controlled causes an update message sent to Hubitat noting the state change. This happens by the HCA program “Hubitat State Connector - Send State to Hubitat”.

That program sends an HTTP message to Hubitat port 39501. Why that port number? It is just something that the Hubitat people defined for this use. On the Hubitat side, through a mechanism I will not go into, the “HCA State Connector” driver processes that message. Its implementation, based upon details in the message, causes the device state in Hubitat to be updated.

Hubitat updating HCA when a device Hubitat manages, and HCA knows about, changes state.

What about the other direction? Let’s look at the case of a device managed by Hubitat that HCA knows about changes its state. Most Zwave switches, for example, when locally controlled send a state report to Hubitat. The Maker-API gets involved again and sends a message to HCA informing it of that state change. That message is received by the “Hubitat Event Server” interface listening on port 8080.



Messages received by that interface are processed by the program “Hubitat Elevation - Hubitat Event Handler”. That program takes the info from Hubitat and updates the HCA device state.



HCA Tech Note 714

A bit complicated, huh? Let's review:

- When Hubitat wants to control a device managed by HCA, one of the Hubitat HCA drivers sends a message to HCA that is received by the "HTTP Server" interface and processed by the "HTTP Server - Receive HTTP" program.
- When HCA wants to control a device managed by Hubitat, it sends a message to Hubitat using an interface that the Maker-API provides.
- When HCA wants to send a state update of a device it manages and Hubitat knows about, the program "Hubitat State Connector - Send State to Hubitat" is used. It does that by sending a message to the Hubitat port 39501.
- When Hubitat wants to send a state update of a device it manages and HCA knows about, the MakerAPI does that. The message is received by HCA using the "Hubitat Event Server" interface and processed by the program "Hubitat Elevation - Hubitat Event Handler".

How does Hubitat know about HCA devices, and how does HCA know about Hubitat devices?

Now that you have all that down, there is only one more piece to describe. I've said in the above explanation several times "... and HCA knows about" and "...Hubitat knows about". How did that "knowing" happen? Let's look at both parts.

HCA learns about Hubitat devices using the facilities of the Maker API. The Maker API provides a mechanism where an application can ask for a list of Hubitat devices. Not all devices, but only those devices that you designate. When you open the Maker-API there is a list of devices where you can tick or un-tick a box next to it and so designate what devices you want exposed and what ones are hidden.

The HCA program "Hubitat Elevation - #1 Do Discover" uses those facilities and creates the devices that the Maker-API exposes to HCA.

The other direction involves the Hubitat HCA App. You added this app to Hubitat as part of the instructions in [HCA technical note #703](#). What it does is to send a message to HCA asking HCA for all the devices that it should know about. From that list it creates what are called "virtual devices" in your Hubitat list of devices. HCA constructs that list including only those devices that have friendly names or have the "Hubitat" tag in their properties. The Hubitat HCA app does its work by sending messages to HCA using HTTP and those messages are received by the "HTTP Server" interface and then processed by the "HTTP Server - Receive HTTP" program.

If you want to dig deeper into all of this you can! You can look at all the HCA programs in the "Hubitat Elevation" and "HTTP Server" packages. On the Hubitat side the device drivers and the app are all there for you to study. While the Maker-API documents all its commands, its actual implementation isn't available to view.

##end##

TechSup@HCATech.com

www.homecontrolassistant.com