



Advanced Scenes – Virtual Devices

WHAT ARE VIRTUAL DEVICES?

- Virtual devices are non-physical devices you can control from inside Vera's web interface
- The most common type of virtual device is a **Virtual On/Off Switch**
- Virtual On/Off Switches can be used to provide a user interface for both the **control of your automation** and the **status of your automation**
- They allow you to:
 - Enable or disable functionality on demand

- Provide visual feedback on the current state of an action in the UI
- Allow you to trigger multiple devices and scenes at once
- Test for logic in scenes before carrying out actions





Advanced Scenes – Virtual Devices

CREATING A VIRTUAL ON/OFF SWITCH IN VERA

1. Download the file [I_VirtualSwitch](http://wiki.micasaverde.com/images/0/04/I_VirtualSwitch.zip) and extract its contents to your computer
 - http://wiki.micasaverde.com/images/0/04/I_VirtualSwitch.zip
2. In the Vera Dashboard, navigate to **Apps -> Develop apps -> Luup files** and upload the **I_VirtualSwitch.XML** file, then Click **Done** when complete

The screenshot shows the Vera Dashboard interface. On the left is a navigation menu with options: Dashboard, Devices, Cameras, Scenes, Energy, Settings, Apps (selected), My apps, Install apps, Develop apps, and Users & Account Info. The main content area is titled 'Develop apps' and contains several options: Test Luup code (Lua), Edit Startup Lua, **Luup files** (highlighted), Serial Port configuration, Create device, and Device Simulator (based on JSON). The 'Luup files' section is highlighted with a red box and contains an 'Upload' button, a 'Drag & drop files' area, a list of files (1) I_VirtualSwitch.xml, a 'Done' button, and a checked checkbox for 'Restart Luup after upload'. Below this is a 'Current files' section with a list of files and their actions:

| File Name | Action |
|---------------------------------|--------------------------|
| D_ALTUI.json | Open in Device Simulator |
| D_ALTUI.xml | View download |
| D_ALTUI_UI7.json | Open in Device Simulator |
| D_AvMisc1.xml | View download |
| D_BinaryLight1.json | Open in Device Simulator |
| D_BinaryLight1.xml | View download |
| D_BinaryLightQubino1DRelay.json | Open in Device Simulator |
| D_BinaryOpenClose1.json | Open in Device Simulator |
| D_BluetoothNetwork.xml | View download |



Advanced Scenes – Virtual Devices

CREATING A VIRTUAL ON/OFF SWITCH IN VERA

3. Open the **Create device** page and populate the fields as shown, then click **Create device**

Develop apps

Test Luup code (Lua)
Edit Startup Lua
Luup files
Serial Port configuration
Create device
Device Simulator (based on JSON)

Create device

Device type

Internal ID

Description: My Virtual Switch

Upnp Device Filename*: D_BinaryLight1.xml

Upnp Implementation Filename: I_VirtualSwitch.xml

IP address

MAC

Room: Please select

Parent device: Please select

*Required

Create device

Description:

The name of the virtual switch

Upnp Device Filename:

Vera's configuration file for standard binary switches - "D_BinaryLight1.xml"

Upnp Implementation Filename:

The Vera implementation file for our new virtual switch - "I_VirtualSwitch.xml"



Advanced Scenes – Virtual Devices

CREATING A VIRTUAL ON/OFF SWITCH IN VERA

4. Click **OK** to confirm device creation and navigate to **Settings -> Z-Wave Settings**
5. Click on the **Advanced** tab and click **GO** on the **Reload Engine** task and then **OK**
6. Refresh your browser (**F5**)

The screenshot shows the VERA user interface. On the left is a navigation menu with the following items: Dashboard, Devices, Cameras, Scenes, Energy, Settings (highlighted in green), Rooms, Customer Care, Location, Net & Wi-fi, Backup, Logs, and Z-Wave Settings. The main content area is titled 'Z-Wave Settings' and has two tabs: 'Options' and 'Advanced'. Under the 'Advanced' tab, there is a list of tasks, each with a 'GO' button to its right. The tasks are: Reset Z-Wave network, Copy Z-Wave network from a primary Vera, Vera shift, Reset Z-Wave chip, Get network update from SUC/SIS, Backup Z-Wave Network, Reload Engine (highlighted with a red box), and Update Neighbor Nodes.