



# Remote Builder Documentation

Version 4.14.0

## Requirements

You will need Java 8 in order to run this software. Java 7 and under is no longer supported. Java 9 and above is incompatible with our software.

Remote Builder works on Windows 7 and up, macOS, and most Linux OS's.

## About Remote Builder

Remote Builder is a program made by Physics Systems for configuring servers running Physics Systems' Installation Server software. It is a drag-and-drop WYSIWYG (What You See Is What You Get) editor where users use buttons on the toolbar and options in the menu bar to add items and components, and drag the items into the preferred places. Depending on the level of complexity of the installation, knowledge in HTML, CSS, and/or JavaScript may be required.

In the future, Remote Builder will be embedded into the web interface of the Installation Server for easy editing anywhere without having to install Java and download Remote Builder on whichever computer(s) a user may wish to use to edit the Installation.

## Table of Contents

1. [Layout](#)
  - a. [Fresh Start](#)
  - b. [Menus](#)
  - c. [Keyboard Shortcuts](#)
2. [Installations](#)
  - a. [Creating a New Installation](#)
  - b. [Saving and Uploading an Installation](#)
  - c. [Opening a Previous Installation](#)
  - d. [Closing an Open Installation](#)
  - e. [Installation Tab](#)
3. [Toolbar](#)
  - a. [New](#)
  - b. [Open](#)
  - c. [Close](#)
  - d. [Save](#)
  - e. [Upload](#)
  - f. [Scheduler](#)
  - g. [NVR](#)
  - h. [Alexa](#)
  - i. [Internal APIs](#)
  - j. [Asynchronous I/O](#)
  - k. [Console Log](#)
4. [Tree](#)
5. [Installation](#)
  - a. [Settings](#)
  - b. [PSNET](#)
  - c. [NVR](#)
  - d. [HTTP & IP Triggers](#)
  - e. [Remote Control Servers](#)
  - f. [Common Script](#)
  - g. [Server-side Script](#)
6. [Notes](#)
7. [Users](#)
  - a. [General](#)
  - b. [Auto Login](#)
  - c. [Alert Notifications \(Defaults\)](#)

d. [Notification Delivery Methods](#)

8. [Images](#)

9. [Sounds](#)

10. [Internal APIs](#)

- a. [Legrand MR232 API](#)
- b. [Nest API](#)
- c. [Pentair IntelliCenter API](#)
- d. [Philips hue API](#)
- e. [Sony BDP API](#)
- f. [WebOS API](#)
- g. [WeMo API](#)

11. [Widgets](#)

- a. [File Structure](#)
- b. [Resource Files](#)
- c. [Widget XML](#)
- d. [Resources](#)
- e. [Implementation Notes](#)
- f. [Example Widget XML](#)
- g. [Example Widget XHTML](#)

12. [Modules](#)

- a. [Aloha Spy Server](#)
- b. [Hai Controller \(UDP\)](#)
- c. [Jukebox](#)
- d. [MyQ](#)
- e. [OPD Server](#)
- f. [Sonos](#)

13. [Drivers](#)

- a. [General Settings](#)
- b. [Serial Settings](#)
- c. [IP Settings](#)
- d. [Script](#)
- e. [Asynchronous I/O](#)

14. [Cameras](#)

- a. [General](#)
- b. [Video](#)
- c. [Motion](#)
- d. [Overlay](#)
- e. [Audio](#)

- f. [Inputs](#)
- g. [Alerts](#)
- 15. [Servers](#)
  - a. [Settings](#)
  - b. [Cameras](#)
  - c. [Alerts](#)
- 16. [Commanders](#)
  - a. [Setup](#)
  - b. [IR Triggering](#)
- 17. [Matrices](#)
  - a. [Port](#)
- 18. [IR Remotes](#)
  - a. [Button](#)
- 19. [Components](#)
  - a. [Driver Components](#)
  - b. [UPnP Components](#)
- 20. [Scripting](#)
  - a. [Scripts](#)
  - b. [Script API](#)
- 21. [Macros](#)
- 22. [Triggers](#)
- 23. [Intents](#)
- 24. [Schedules](#)
  - a. [Schedule](#)
  - b. [Excluded Months](#)
  - c. [Excluded Weeks](#)
  - d. [Excluded Days](#)
- 25. [Pages](#)
  - a. [Pages Taskbar](#)
  - b. [Buttons](#)
  - c. [Labels](#)
  - d. [Videos](#)
  - e. [Sliders](#)
  - f. [Select Boxes \(Beta\)](#)
  - g. [Widgets](#)
  - h. [XHTML](#)
  - i. [Page Properties](#)
  - j. [Layers](#)

k. [Elements](#)

l. [Timers](#)

26. [Motion Detection](#)

a. [Overview: How Motion Detection Works](#)

b. [How to Properly Setup Motion Detection](#)

c. [The Stop Delay](#)

d. [The Refresh Interval](#)

e. [The Detection FPS](#)

27. [IR Command](#)

28. [Serial Command](#)

29. [IP Command](#)

30. [Relay Command](#)

31. [Number Property](#)

a. [Properties](#)

b. [Conversion Functions](#)

32. [Enum Property](#)

a. [Properties](#)

b. [Add Item](#)

c. [Conversion Functions](#)

# 1. Layout

## FRESH START

Whenever a user opens Remote Builder (by opening it with Java), they will be presented with an empty layout. (Figure 1.1)

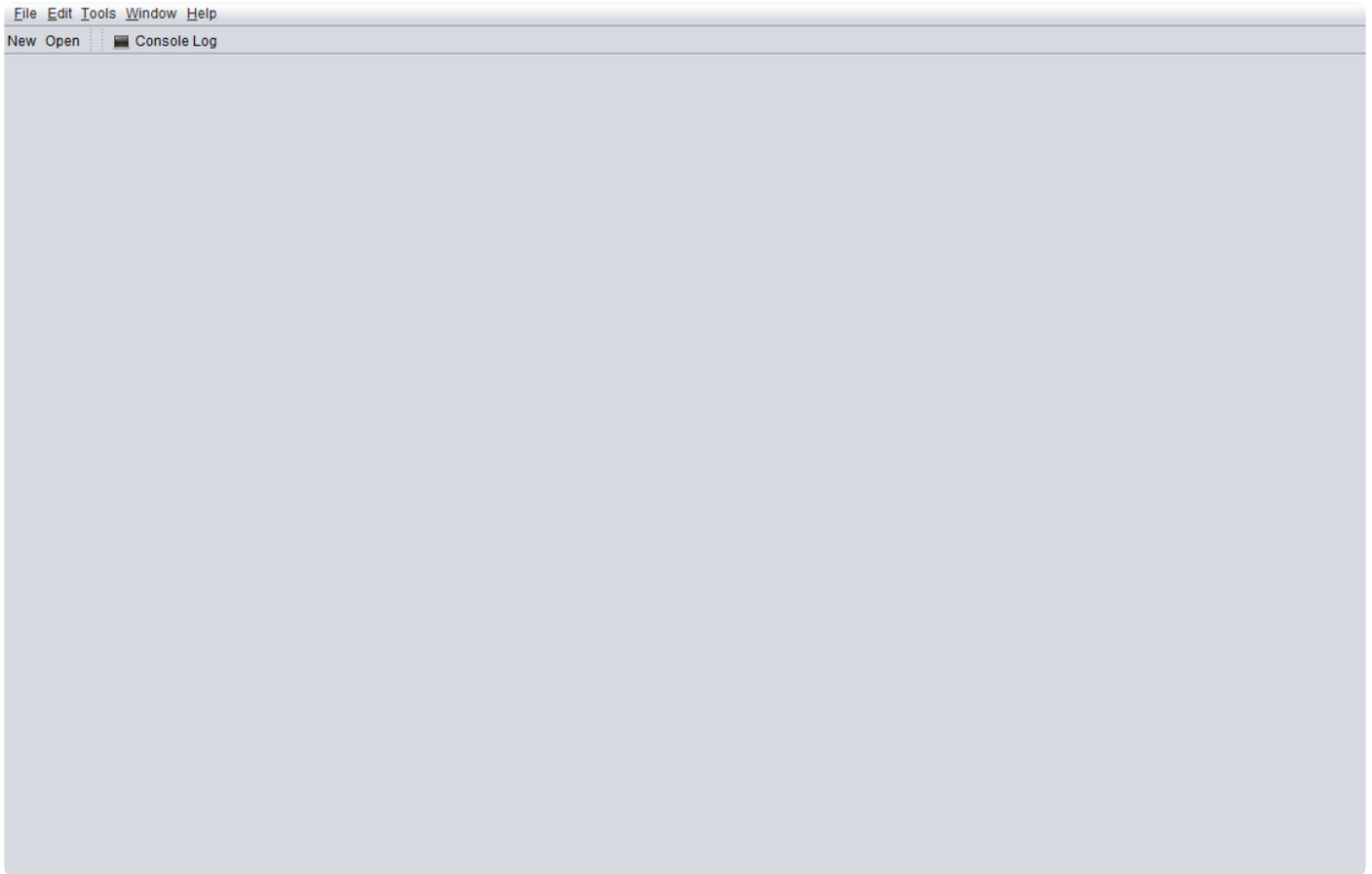
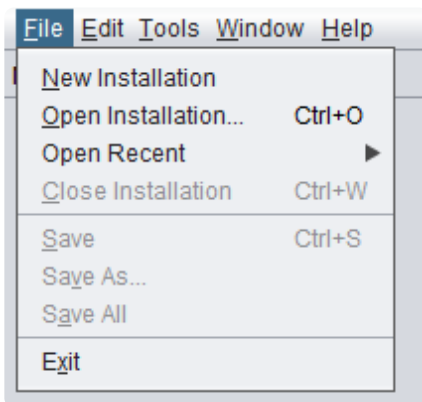


Figure 1.1 | Fresh Start

Users should notice in the title bar of the instance of Remote Builder the ports for previewing the open Installation. For example, if it says it is listening on ports 8000 and 8443 (which is the default), then a preview of the open Installation can be browsed to at localhost:8000 and localhost:8443.

## MENUS

Initially, there are five (5) Menu bar tabs: File, Edit, Tools, Window, and Help.



## FILES

The **File** tab has the options to create a new Installation, to open an Installation, to open recently opened Installations, to close the current Installation, to save the current Installation, to save the current Installation as a new file, to save all open Installations, and to exit Remote Builder. (Figure 1.2)

## EDIT

The **Edit** tab has the options to undo, redo, cut, copy, paste, delete,

Figure 1.2 | File tab

select all, and to change the Grid and Live Preview preferences. (Figure 1.3)

Items cannot be deleted by the Delete button on the keyboard or by the Delete option in the Edit tab if the item is referred to elsewhere. A user must first remove the references before deleting the item.

When a user clicks on “Preferences...”, they are shown a dialog box with a couple tabs containing settings that can be changed.

### Grid

The first tab is labelled “Grid”; these settings change the Grid shown on Pages. (Figure 1.4) See the section Pages for additional information.

Grid X: 8  
Grid Y: 8  
Show Grid: unchecked  
Snap to Grid: unchecked

- The **Grid X** input box controls how much space in pixels is in between each vertical line of the Grid.
- The **Grid Y** input box controls how much space in pixels is in between each horizontal line of the Grid.
- The **Show Grid** checkbox shows (checked) or hides (unchecked) the Grid on Pages.
- The **Snap to Grid** checkbox controls whether Page items “snap” to the Grid lines or can be moved around freely.

### Live Preview

The second tab is labelled “Live Preview”; these settings change for testing and previewing the Installation locally. (Figure 1.5)

Enabled: checked  
HTTP Port: 8000  
HTTPS Port: 8443  
Offline Caching: unchecked

- The **Enabled** checkbox toggles whether the Live Preview feature is enabled (checked) or not (unchecked).

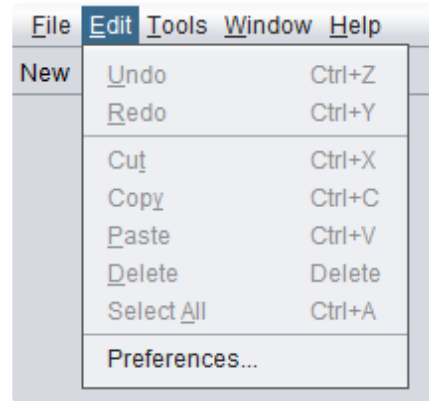


Figure 1.3 | Edit tab

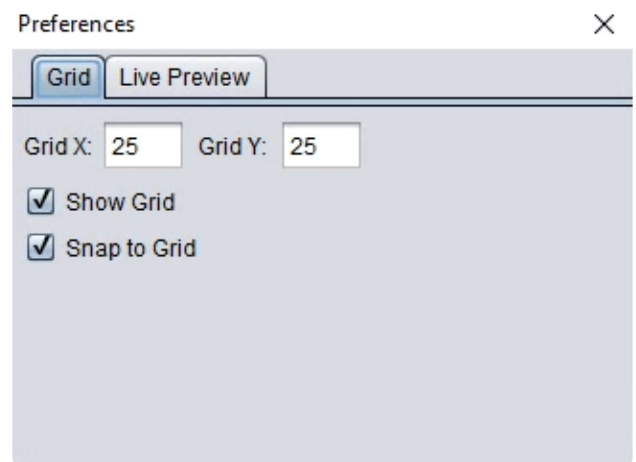


Figure 1.4 | Preferences - Grid tab

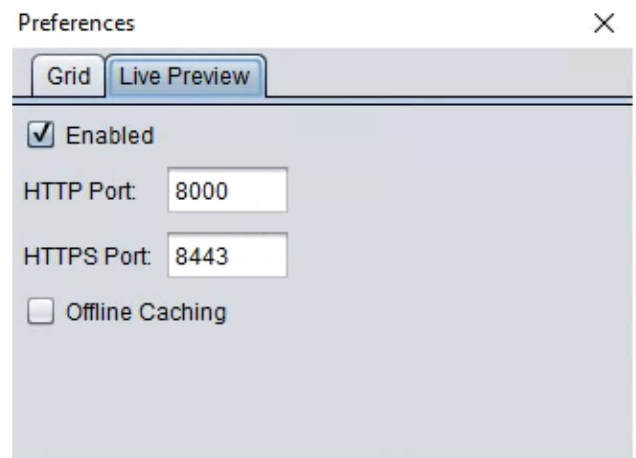


Figure 1.5 | Preferences - Live Preview tab

- The **HTTP port** input box defines the port on localhost that the Live Preview can be viewed at. To view it at this location, open a browser and enter localhost:8000 in the URL bar.
- The **HTTPS Port** input box defines the port on localhost that the Live Preview can be viewed at securely, if Remote Builder can install an HTTPS certificate.
- The **Offline Caching** checkbox toggles whether caching is enabled (checked) or not (unchecked) in the Live Preview.

## TOOLS

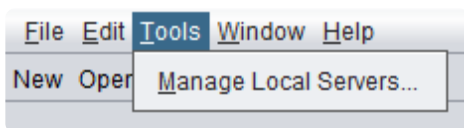


Figure 1.6 | Tools tab

The **Tools** tab has the option to manage local servers. (Figure 1.6) This allows a user to see what servers are on the local network and change their upload password if needed.

## WINDOWS

The **Window** tab has the options to close all open installations, and to open the Console Log. (Figure 1.7) The Console Log shows any alerts and errors that are generated while running the Installation locally. This is good to use when testing scripts.

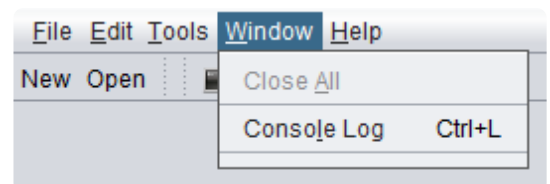


Figure 1.7 | Window tab

## HELP

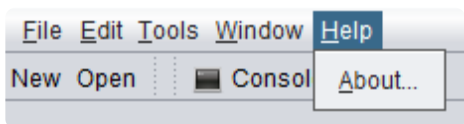


Figure 1.8 | Help tab

The **Help** tab has the option to see information about Remote Builder, which includes the version number of Remote Builder and contact information for Physics Systems. (Figure 1.8)



## KEYBOARD SHORTCUTS

The following table documents the keyboard shortcuts available in Remote Builder for both Windows and macOS, along with their appropriate functions. Mastering these shortcuts can help the user be more productive while building an Installation.

	Function	Windows	macOS
1	Open Installation	Ctrl+O	Command+O
2	Close Installation	Ctrl+W	Command+W
3	Save Installation	Ctrl+S	Command+S
4	Undo	Ctrl+Z	Command+Z
5	Redo	Ctrl+Y	Command+Y
6	Cut	Ctrl+X	Command+X
7	Copy	Ctrl+C	Command+C
8	Paste	Ctrl+V	Command+V
9	Delete	Delete	
10	Select All	Ctrl+A	Command+A
11	Open Console Log	Ctrl+L	Command+L
12	Add Internal API	Ctrl+Shift+A	Command+Shift+A
13	New Note	Ctrl+Shift+N	Command+Shift+N
14	New User	Ctrl+Shift+U	Command+Shift+U
15	New Driver	Ctrl+Shift+D	Command+Shift+D
16	New Driver Component	Ctrl+Shift+C	Command+Shift+C
17	New Macro	Ctrl+Shift+M	Command+Shift+M
18	New Schedule	Ctrl+Shift+S	Command+Shift+S
19	New Page	Ctrl+Shift+P	Command+Shift+P
20	Upload	Ctrl+U	Command+U
21	Add Button	Alt+B	
22	Add Label	Alt+L	
23	Add Video	Alt+V	
24	Add Select	Alt+E	
25	Add Slider	Alt+S	
26	Add Widget	Alt+W	
27	Add XHTML	Alt+X	
28	Open Properties	Alt+P	
29	Open Layers	Alt+A	
30	File tab	Alt+F	
31	Edit tab	Alt+E	
32	Installation tab	Alt+I	
33	Page tab	Alt+P	
34	Tools tab	Alt+T	
35	Window tab	Alt+W	
36	Help tab	Alt+H	
37	Insert ASCII	Alt+A	

Table 1 | Keyboard Shortcuts Available in Remote Builder

## 2. Installations

### CREATING A NEW INSTALLATION

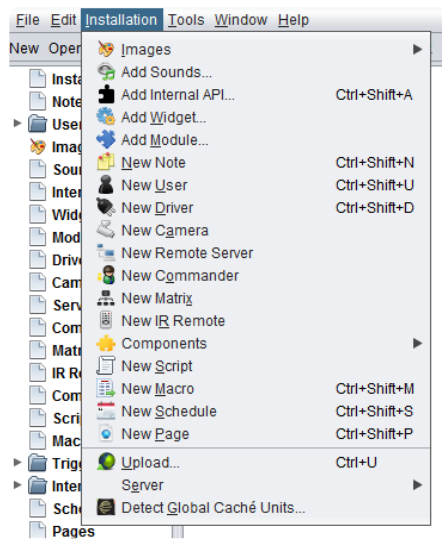


Figure 2.1 | Installation tab

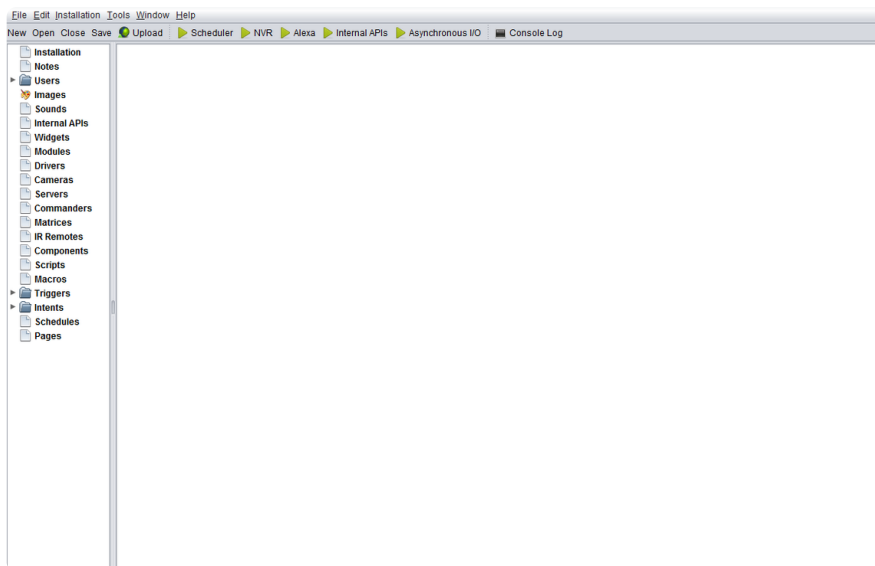


Figure 2.2 | New Installation (above)

Whenever the user creates a new installation, or opens an existing installation, the user will then be presented with a few more options than before: an Installations tab in the Menu bar, (Figure 2.1) additional options in the Task bar, and a Menu panel to the left. (Figure 2.2)

### SAVING AND UPLOADING AN INSTALLATION

To save an installation, a user can use a variety of methods to do this. There is the keyboard shortcut Ctrl/Command+S, the Save button on the Taskbar (Figure 2.3), and the Save menu option in the File tab. Whenever a user saves a new installation, Remote Builder will prompt the user to specify the location and name of the file.

For the Installation created in Remote Builder to be applied in the server, the user needs to upload the Installation file. Remote Builder makes this easy to do.

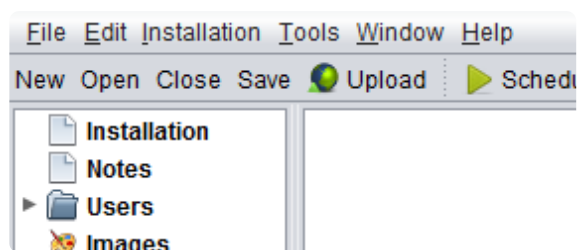


Figure 2.3

The user ought to make sure the server's IP address and port in the Installation settings have been set (see the **Installation** section for more details). On the Taskbar there is a button that says "Upload." (Figure 2.3) When a user clicks this, it will prompt them for a password. The default value is "password". Type this in or the custom set password (see below on setting the server's password) and click the "OK" button. It will upload the file and show the user a dialog box notifying the user of the status of the

upload and whether it failed or succeeded in uploading the file.

### OPENING A PREVIOUS INSTALLATION

To open an existing installation, a user would click on the File menu option, and click on Open Installation and proceed to browse to it with the File dialog box (this dialog box can also be opened

with Ctrl/Command+O), or if they have opened it recently, click to Open Recent and then to one of the recent options. (Figure 2.4)

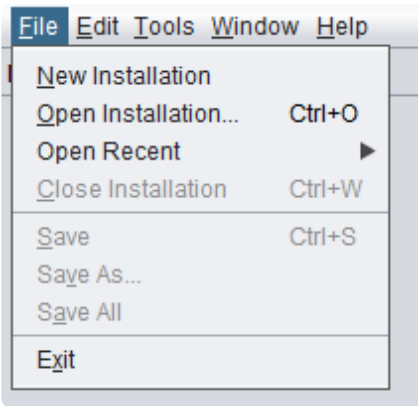


Figure 2.4

## CLOSING AN OPEN INSTALLATION

To close the open Installation, a user would click the “Close Installation” option in the File menu option, use the Close button on the Taskbar, or use the keyboard shortcut Ctrl/Command+W. After closing the Installation, Remote Builder will remain open, showing either the next open Installation (if another one is open) or it will look like it did when it was first opened (see Figure 1.1 in the **Fresh Start** section under the **Layout** section).

## INSTALLATION TAB

The Installation tab contains multiple options for adding items to the Installation, as well as other actions.

- **Images** - Contains options for adding and changing images. See the section **Images** for more information.
  - **Add...** - Allows for adding Images to the Installation.
  - **Change...** - Used for replacing Images. New Image must already have been added.
- **Add Sounds...** - Allows for adding Sounds. See the section **Sounds** for more information.
- **Add Internal API...** (Ctrl/Command+Shift+A) - Allows for adding Internal APIs. See the section **Internal APIs** for more information.
- **Add Widget...** - Allows for adding Widgets. See the section **Widget** for more information.
- **Add Module...** - Allows for adding Modules. See the section **Modules** for more information.
- **New Note** (Ctrl/Command+Shift+N) - Allows for creating new Notes. See the section **Notes** for more information.
- **New User** (Ctrl/Command+Shift+U) - Allows for creating new Users. See the section **Users** for more information.
- **New Driver** (Ctrl/Command+Shift+D) - Allows for creating new Drivers. See the section **Drivers** for more information.
- **New Camera** - Allows for creating new Cameras. See the section **Cameras** for more information.
- **New Remote Server** - Allows for creating new Remote Servers. See the section **Servers** for more information.
- **New Commander** - Allows for creating new Commanders. See the section **Commanders** for more information.
- **New Matrix** - Allows for creating new Matrices. See the section **Matrices** for more information.

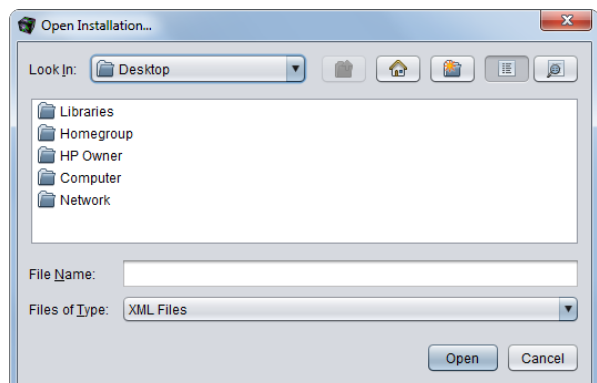
- **New IR Remote** - Allows for creating new IR Remotes. See the section **IR Remotes** for more information.
- **Components** - Contains options for creating new Driver Components and adding UPnP Components. See the section **Components** for more information.
  - **New Driver Component** (Ctrl/Command+Shift+C) - Allows for creating new Driver Components.
  - **Add UPnP Component** - Allows for adding UPnP devices as Components.
- **New Script** - Allows for creating new Scripts. See the section **Scripting** for more information.
- **New Macro** (Ctrl/Command+Shift+M) - Allows for creating new Macros. See the section **Macros** for more information.
- **New Schedule** (Ctrl/Command+Shift+S) - Allows for creating new Schedules. See the section **Schedules** for more information.
- **New Page** (Ctrl/Command+Shift+P) - Allows for creating new Pages. See the section **Pages** for more information.
- **Upload...** - Allows for uploading the open Installation to its Installation Server. See the section **Saving and Uploading an Installation** for more information.
- **Server** - Contains options for managing the Installation's Installation Server.
  - **Change Password...** - Allows for changing the Installation Server's upload password. The factory default is password. There are three (3) input boxes, one (1) for the current password, and two (2) for the new password. If the current password is set to the factory default, do NOT change it over the internet or any other network. The new password must be nine or more characters and have at least one upper case letter, one symbol, and one number. Losing this password will require a physical reset to the factory default.
  - **Upload SSL Certificate...** - Allows for uploading an SSL Certificate to the Installation Server. You will need a PEM encoded, PKCS #1 or PCS #8 private key and a PEM encoded, X.509 certificate. Having multiple certificates are allowed. The first text box is for the private key, and the second text box is for the certificate(s). and If the current password is set to the factory default, do NOT upload your private key over the internet or any other insecure network. Change your password first.
  - **Get Log Files...** - Allows for retrieving the Installation Server's log files. You will need to type in the Installation Server's password in order to retrieve the log files.
- **Detect Global Caché Units...** - Allows for detecting Global Caché Units that are on the current network. As they are discovered, they will be shown in the popup window, with their MAC Address, IP Address, Model, and any other Info. Depending on the unit, you may have options to "Configure in Browser...", "Blink Power LED...", and/or "Add To Commanders...". See **Commanders** for more information.

### 3. Toolbar

#### NEW

Clicking “New” on the Toolbar creates a new Installation, ready to be edited in Remote Builder. It is not saved until it is either uploaded or manually saved (see the **Saving and Uploading an Installation** section under the **Installations** section).

#### OPEN



*Figure 3.1 | Open Installation... Dialog Box*

Clicking “Open” on the Toolbar opens the File dialog box for a user to select and open an existing Installation. (Figure 3.1) The default location is the current user's Documents folder. The user can browse from there to the location of the Installation file using the controls at the top or by clicking through the directories shown in the middle. Do note that it will only show drives, directories, and XML files.

A user can jump to multiple locations in the directory tree by clicking on the drop-down box with the label “Look In:” to its left.

To the right of the drop-down box is a button with an icon of a folder with an upwards pointing arrow on it. When clicked, shows the contents of the parent directory of the shown directory. (This button is grayed out in Figure 3.1)

To the right of that button is a button with a house icon. When clicked, this is show the home directory. In the case of Figure 3.1, which was run on a Windows 7 PC, the home directory is the Desktop directory.

To the right of this button is a button with an icon of a folder with a shining, orange star. Clicking it will create a new directory within the currently shown directory.

The next two buttons changes how the directories are shown. The first one (the default button) shows the directories as a list. The second one shows the directories with more details.

When the user has browsed to the location of the Installation file, they are to click on the file and click the Open button or press Enter.

#### CLOSE

Clicking “Close” on the Toolbar closes the currently open Installation. Other open Installations will not be closed.

#### SAVE

Clicking “Save” on the Toolbar will save the current, open Installation. Please note that it will not attempt to save if no changes were made. If it has not been saved to the hard drive yet (as would be the case if it was just created, changes were made and it had not been saved since), it will open a File dialog box in which the user ought to choose the location and file name of the Installation file. The default location is the current user's Documents folder. The user can browse from there to the desired location for the

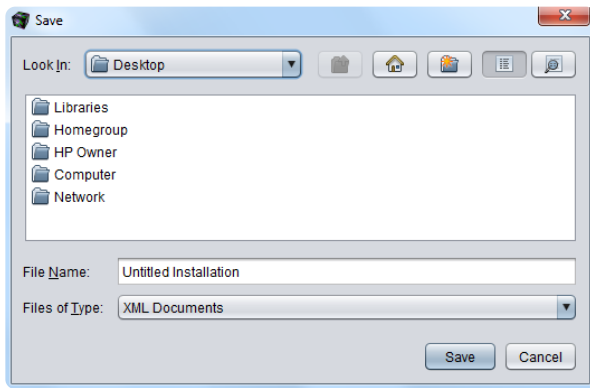


Figure 3.2 | Save Dialog Box

Installation file using the controls at the top or by clicking through the directories shown in the middle. Do note that it will only show drives, directories, and XML files. A user can jump to multiple locations in the directory tree by clicking on the drop-down box with the label “Look In:” to its left.

To the right of the drop-down box is a button with an icon of a folder with an upwards pointing arrow on it. When clicked, shows the contents of the parent directory of the shown directory. (This button is grayed out in Figure 3.1)

To the right of that button is a button with a house icon. When clicked, this is show the home directory. In the case of Figure 3.1, which was run on a Windows 7 PC, the home directory is the Desktop directory.

To the right of this button is a button with an icon of a folder with a shining, orange light. Clicking it will create a new directory within the currently shown directory.

The next two buttons changes how the directories are shown. The first one (the default button) shows the directories as a list. The second one shows the directories with more details.

## UPLOAD

Clicking “Upload” on the Toolbar will upload the Installation file to the server. The user ought to make sure the server's IP address and port in the Installation settings have been set (see the **Installation** section for more details). It will first prompt the user for a password. The default value is “password”. Type this in or the custom set password (see below on setting the server's password) and click the “OK” button. It will upload the file and show the user a dialog box notifying the user of the status of the upload and whether it failed or succeeded in uploading the file.

## SCHEDULER

Clicking “Scheduler” on the Toolbar will start the set schedules in the Installation for testing purposes. See the section **Schedules** for more information.

## NVR

Clicking “NVR” on the Toolbar will start the NVR feature in Remote Builder for testing purposes. It will check for a canary file in the nvr directory in the directory Remote Builder is located in. Remote Builder will create the directory if it does not exist, but it will not create a canary file if it does not exist. It is important for the user to create this file in the nvr directory for this feature to work.

The reason Remote Builder checks for the canary file is so it can know whether or not it has read and write access to the directory. If it can read and write the canary file, it can record video files. This behavior is the same with Installation Server.

## ALEXA

Clicking “Alexa” on the Toolbar will start the Alexa feature for testing purposes. This feature allows for local communication between Remote Builder and an Alexa device.

## INTERNAL APIS

Clicking “Internal APIs” on the Toolbar will start the added Internal APIs for testing purposes. For adding Internal APIs, see the section **Internal APIs**.

#### **ASYNCHRONOUS I/O**

Clicking “Asynchronous I/O” on the Toolbar will start Asynchronous I/O feature for testing purposes.

#### **CONSOLE LOG**

Clicking “Console Log” on the Toolbar will open the Console Log in Remote Builder. From here, a user can see the recorded logs and errors. This can be very useful when writing scripts. (See the section **Scripting**)

## 4. Tree

The Tree on the left side appears whenever a new installation is created or an existing installation is opened. Each item either controls certain settings or stores files or items. The following are the available Tree items:

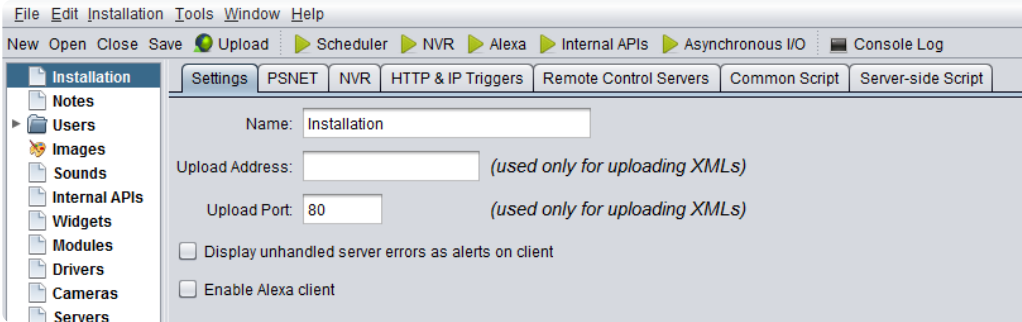
- Installation - contains settings about the current Installation; see the section **Installation**.
- Notes - each Note is a blank text box that can be used for storing information, taking notes, or recording documentation; see the section **Notes**.
- Users - add and edit Users who can log into the Installation when it is live; see the section **Users**.
- Images - add Images that can be used as backgrounds for Page items; see the section **Images**.
- Sounds - add Sounds that can be played as part of Actions; see the section **Sounds**.
- Internal APIs - easily integrate common APIs; see the section **Internal APIs**.
- Widgets - add Widgets that can be placed on pages to show specific content; see the section **Widgets**.
- Modules - easily integrate special solutions; see the section **Modules**.
- Drivers - Drivers control connected devices by sending added Commands; see the section **Drivers**.
- Cameras - add Cameras of multiple models; see the section **Cameras**.
- Servers - connect other Installation Servers to the current Installation; see the section **Servers**.
- Commanders - the central piece of much automation; see the section **Commanders**.
- Matrices - add a network Matrix; see the section **Matrices**.
- IR Remotes - add an IR Remote with Buttons to send IR commands; see the section **IR Remotes**.
- Components - send Driver and UPnP commands; see the section **Components**.
- Scripts - use the Script API to automate whatever you need; see the section **Scripting**.
- Macros - combine multiple actions into a single Macro; see the section **Macros**.
- Triggers - Do stuff when stuff happens; see the section **Triggers**.
- Intents - run a script when a command is run; see the section **Intents**.
- Schedules - run a Macro at specific times and intervals; see the section **Schedules**.
- Pages - these are the main user interface of the Installation; see the section **Pages**.



## 5. Installation

The Installation option contains seven (7) tabs. Each tab is meant to categorize the different parts of the Installation. Some tabs may contain settings, and most tabs act as folders for parts of the Installation such as pages, images, and scripts.

### SETTINGS



The Settings tab (Figure 5.1) contains input boxes and checkboxes for configuring the Installation.

Figure 5.1 | Installation Settings tab

*Name: Installation*

*Upload Address: (unset)*

*Upload Port: 80*

*Display unhandled errors as alerts on client: unchecked*

*Enable Alexa client: unchecked*

- The **Name** input box defines the name of the Installation.
- The **Upload Address** input box sets the address used for uploading the Installation XML file. It can be the IP address or domain name of the server.
- The **Upload Port** input box sets the port used with the **Upload Address** to upload the Installation XML file.
- The **Display unhandled errors as alerts on client** checkbox defines whether unhandled errors generate JavaScript alerts in the browser (checked) or not (unchecked).
- The **Enable Alexa client** checkbox enables (checked) or disables (unchecked) the Alexa client feature. Only check if an Alexa device is available on the network.

### PSNET

The PSNET tab (Figure 5.2) contains fields for inserting the PSNET ID and Key associated with the Installation and the external ports the Installation is accessed on for viewing and interaction purposes.

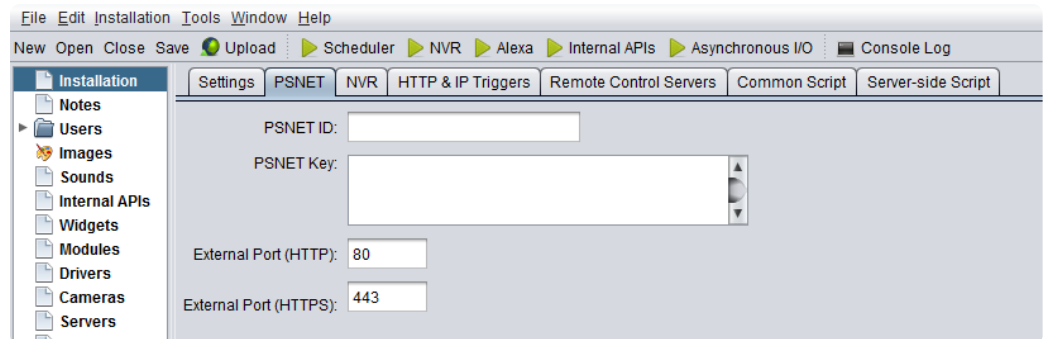


Figure 5.2 | Installation PSNET tab

PSNET ID: (unset)  
PSNET Key: (unset)  
External Port (HTTP): 80  
External Port (HTTPS): 443

- The **PSNET ID** input box sets the physicsystems.net ID a user was assigned for this Installation. Users ought to leave this blank if they were not assigned one. This is paired with the PSNET Key. A user will need to contact us to receive a PSNET ID and PSNET Key.
- The **PSNET Key** input box sets the physicsystems.net Key a user was assigned for this Installation. Users ought to leave this blank if they were not assigned one. This is paired with the PSNET ID. A user will need to contact us to receive a PSNET ID and PSNET Key.
- The **External Port (HTTP)** input box sets the port number your router is listening to for HTTP traffic. If you have set up your router to port forward to the Installation Server, enter the port number the router is listening to on the external (WAN) side for HTTP connections. Ordinarily, this would be port 80.
- The **External Port (HTTPS)** input box sets the port number your router is listening to for HTTPS traffic. If you have set up your router to port forward to the Installation Server, enter the port number the router is listening to on the external (WAN) side for HTTPS connections. Ordinarily, this would be port 443.

If the PSNET ID and Key fields are filled out, a user can click the “Check Account Status...” button to verify that the PSNET ID and Key are correct and see the status of the associated account.

## NVR

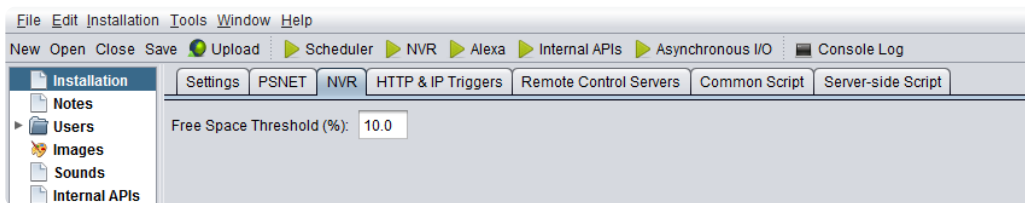


Figure 5.3| Installation NVR tab

The NVR tab (Figure 5.3) contains settings for the Installation Server's NVR feature.

Free Space Threshold (%): 10.0

- The **Free Space Threshold (%)** input box defines the minimum percentage of space that must be free on the server's hard drive(s) before the video files are written over. The oldest files are written over first.

## HTTP & IP TRIGGERS

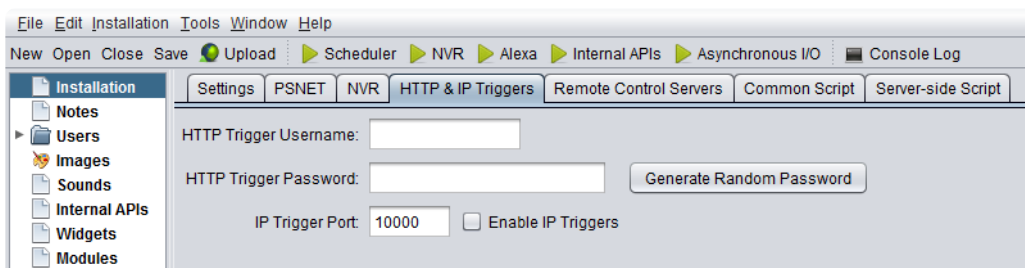


Figure 5.4 | Installation HTTP & IP Triggers tab

The HTTP & IP Triggers tab (Figure 5.4) contains settings for Triggers. Devices can send Triggers to the server over the network using this username and password. The server can capture it, and depending on what is set

up in the Installation, it is able to perform an action based on that Trigger. See the section [Triggers](#) for more information.

*HTTP Trigger Username: (unset)*

*HTTP Trigger Password: (unset)*

*IP Trigger Port: 10000*

*Enable IP Triggers: unchecked*

- The **HTTP Trigger Username** input box defines the username to be used with HTTP Triggers.
- The **HTTP Trigger Password** input box defines the password to be used with HTTP Triggers. There is a button next to it for generating a random password that fills this box.
- The **IP Trigger Port** defines which port the Installation Server should listen on for IP Triggers.
- The **Enabled IP Triggers** checkbox defines whether the Installation Server should listen for IP Triggers on the specified port (checked) or not (unchecked).

## REMOTE CONTROL SERVERS

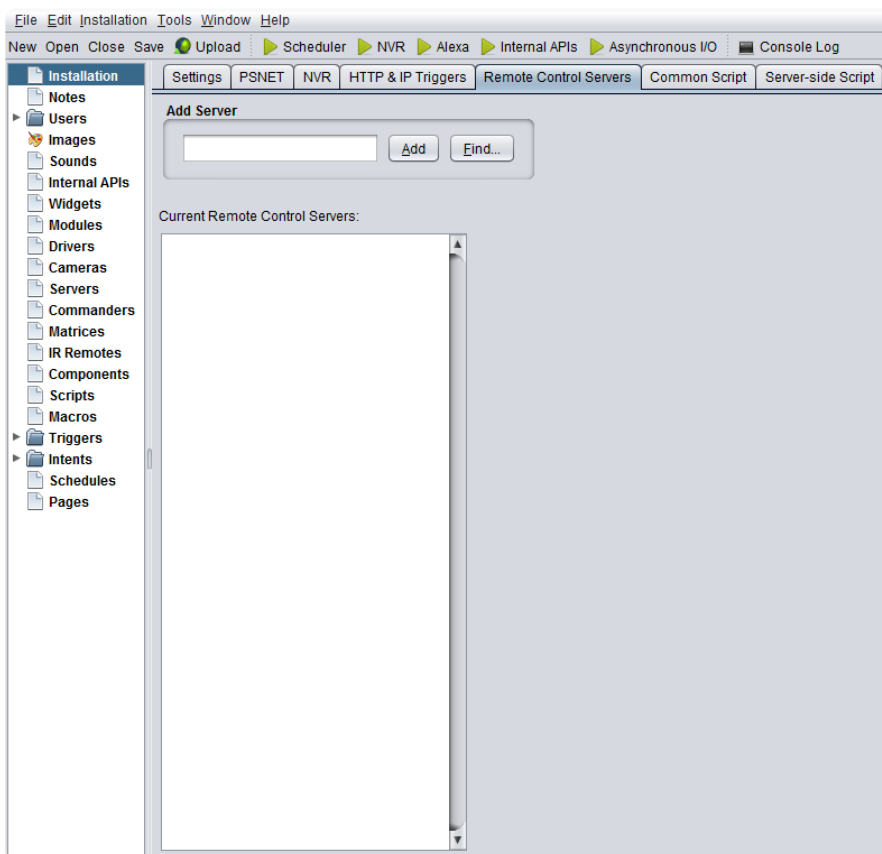


Figure 5.5 | Installation Remote Control Servers tab

The Remote Control Servers tab (Figure 3.5) contains a field for adding servers that have the Physics Systems Remote Control software installed and are set up to work with remote controllers. Such servers can receive input from physical remote controllers and send their input to the Installation server, which then can use that input as a trigger for a certain function. This field allows you to manually type in the server's IP address, or you can click the "Find" button and select one that is on the network. Then you click the "Add" button to add it to the "Current Remote Control Servers" list. To remove a server from the list, select it by clicking on it, and press the Delete key on your keyboard.

## COMMON SCRIPT

The Common Script tab (Figure 3.6) contains a text entry box for a client-side script written in JavaScript that will be available to page, client and server-side scripts. See the section on **Scripting** for more information.

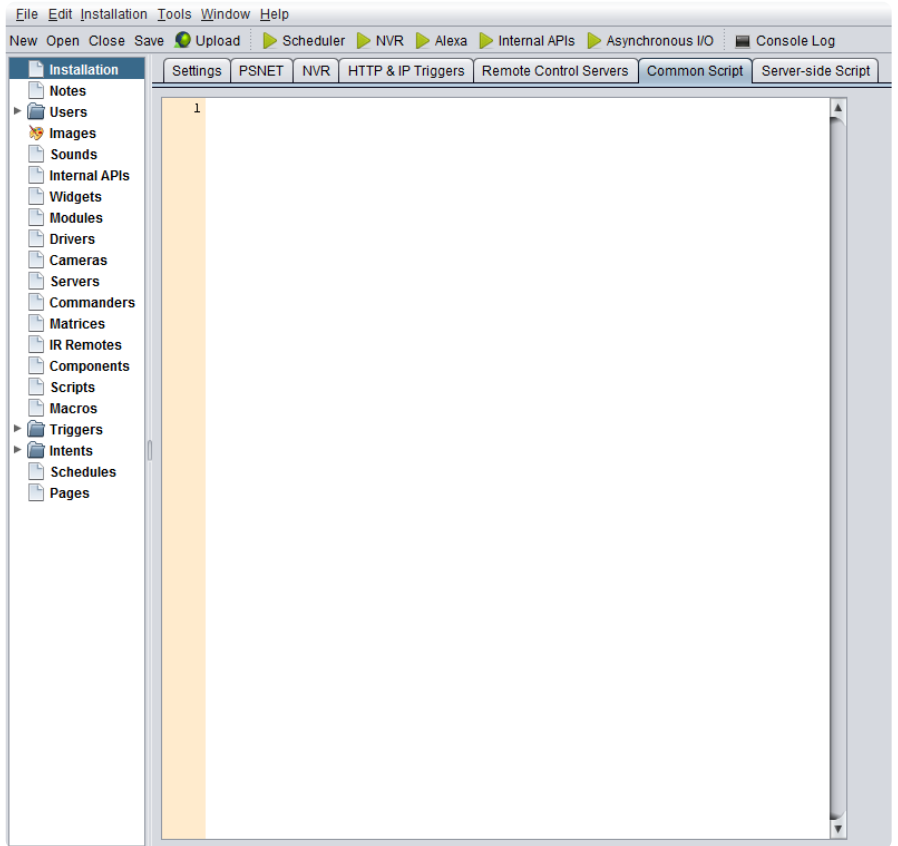


Figure 5.6 | Installation Common Script tab

## SERVER-SIDE SCRIPT

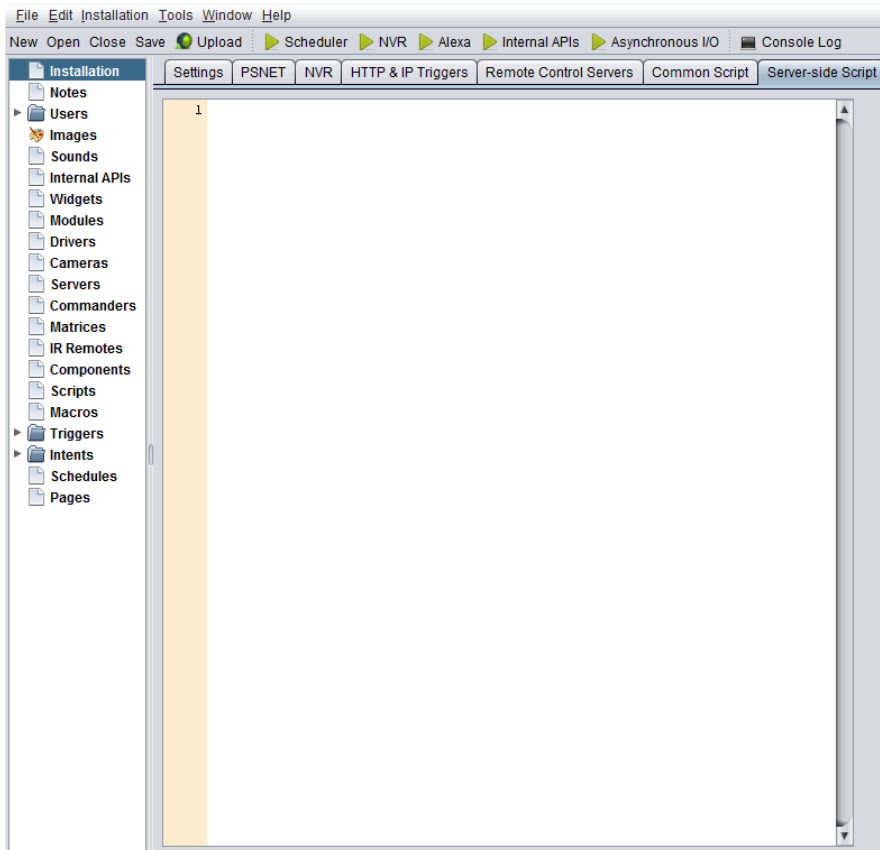


Figure 5.7 | Installation Server-side Script tab

The Server-side Script tab (Figure 3.7) contains a text entry box for a server-side script written in JavaScript (using the Physics Systems Platform Script API) that is available to other server-side scripts. See the section on **Scripting** for more information.

## 6. Notes

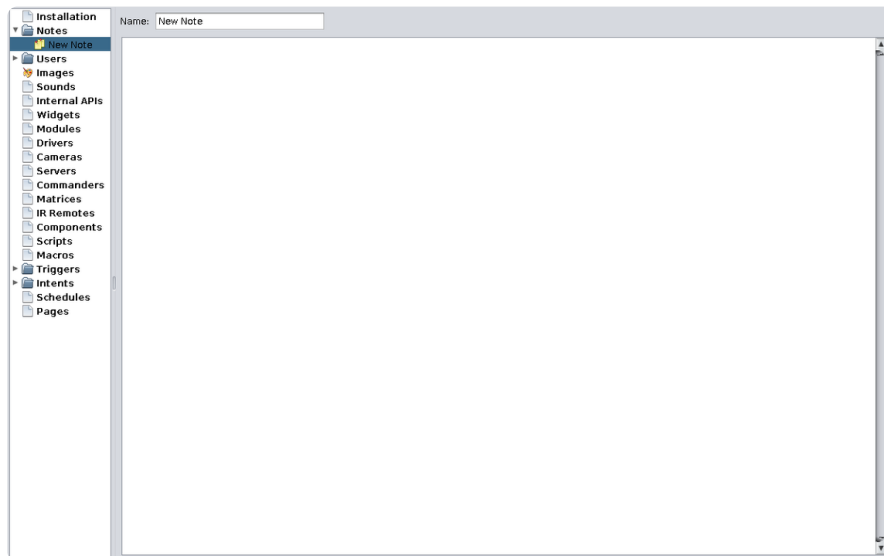


Figure 6.1 | A New Note

Notes can be used to record pretty much anything a user may need to record and store on the server. A note can be added by going to the Installation menu option, and clicking on the Note option. Alternatively, a new note can be created with the keyboard shortcut `Ctrl/Command+Shift+N`. Anyone who copies the Installation file to edit will have the Notes available to them as well. This can be useful for documenting how something works, or pretty much anything.

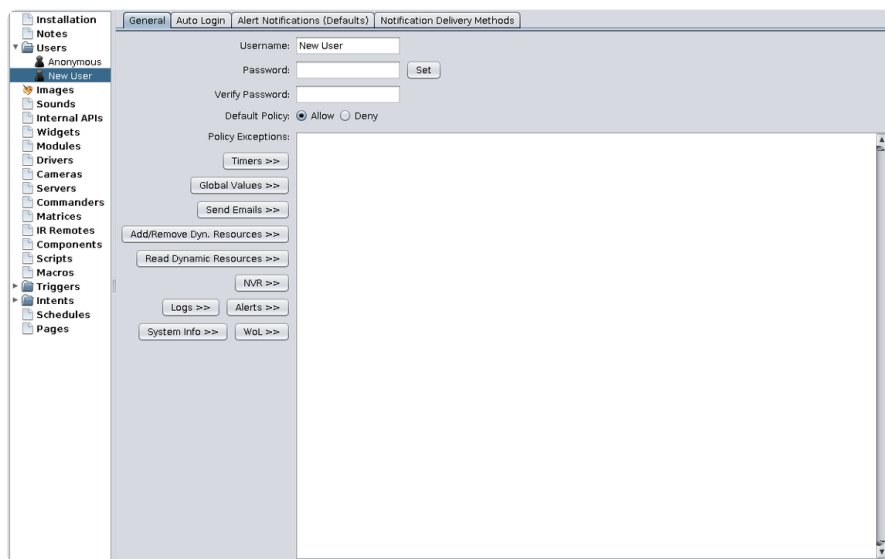
When a new Note is created, the user will see two text boxes. The first one is a single line text box, labelled as "Name" with the default value of "New Note". The second text box takes up the entire space available. It is for the contents of the note, thus it has no default value.

## 7. Users

A User can be added by clicking on the Installation tab in the Menu bar, and then clicking on New User. All Users listed under this folder can access the Installation according to the rules set per User. This can include not being able to access the Installation at all. By default, the Installation has a single user named “Anonymous” for anonymous users. This user cannot be deleted.

When a user is clicked, four (4) tabs will be shown, each with different settings for different functions.

### GENERAL



The first tab is labelled “General” and has three (3) input boxes and two (2) radio buttons, along with multiple buttons.

Figure 7.1 | The General tab

Username: New User

Password: (unset)

Verify Password: (unset)

Default Policy: Allow

- The **Username** input box defines the username that a user would use to login to the Installation with. Changes to this reflect on the label in the Tree immediately; it also serves as the display name of the user.
- The **Password** input box defines the password to be associated with this User.
- The **Verify Password** input box should contain the same text that was entered in **Password**. The password is then set by clicking on the “Set” button. Once set, the password disappears in both boxes but is saved within the Installation.

The radio buttons have a label to the left which says “Default Policy”. Whichever radio button is set defines the default policy of that user. The first radio button is labelled “Allow” and is default. Having this selected allows the user to have access to entire Installation, with the exception of the Policy Exceptions defined below (more on that in a bit). The second radio button is labelled “Deny”. Having this selected prevents the user from having any access to the Installation, with the exception of the Policy Exceptions.

*For security purposes, one ought to set the Anonymous user's Default Policy to “Deny” to prevent any unwelcome Internet browsers from exploring the Installation and potentially changing something.*

If the Default Policy is set to “Allow”, the user is denied access to all of the items in the Policy Exceptions. Likewise, if the Default Policy is set to “Deny”, the user is allowed access to all of the items in the Policy Exceptions. Some of the commonly used items for the exceptions can be seen to the left of the big white box. Clicking on these buttons adds those items to the Policy Exceptions. These buttons are:

- **Timers >>**: either allows or denies access to the timers (see the section on **Timers** under **Pages**).
- **Global Values >>**: either allows or denies access to the Global Values.
- **Send Emails >>**: either allows or denies access to the API feature that allows for sending emails.
- **Add/Remove Dyn. Resources >>**: either allows or denies access to the API feature that allows for adding or removing dynamic resources.
- **Read Dynamic Resources >>**: either allows or denies access to the API feature that allows for reading dynamic resources.
- **NVR >>**: either allows or denies access to the NVR page (/nvr).
- **NVR Delete >>**: either allows or denies users from deleting NVR timeline entries.
- **Logs >>**: either allows or denies access to the Logs page (/logs).
- **Alerts >>**: either allows or denies access to the Alerts page (/alerts).
- **System Info >>**: either allows or denies access to the System Info page (/info).
- **WoL >>**: either allows or denies access to the Wake on LAN feature.

Along with the Privileges mentioned above, there are other items that can be added to the Policy Exceptions. These include:

- Module Privileges
- Cameras
- Servers
- Commanders
- Matrices
- Components
- Macros
- Pages

## **AUTO LOGIN**

The second tab is labelled “Auto Login”. It contains a text box and a button labelled “Add”. An IP address can be added to the text box, and added to the user profile by clicking on the “Add” button. This adds the IP address to a list of IP addresses that automatically login the user profile in. This feature is disabled for Anonymous users.

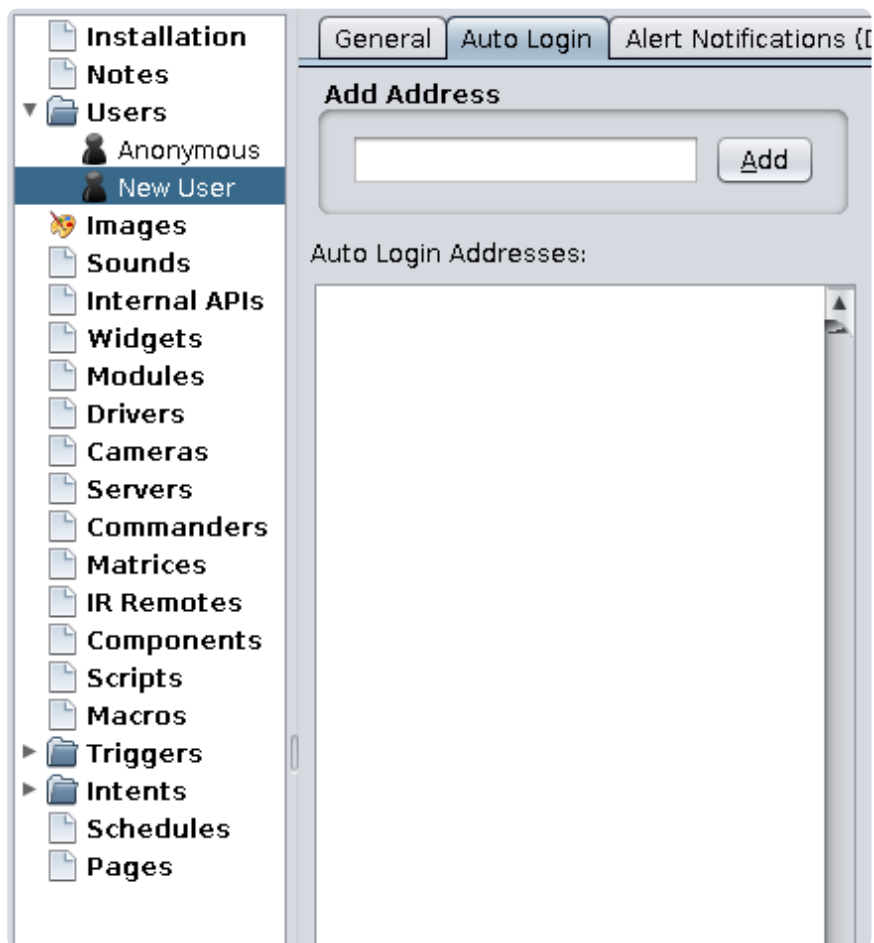
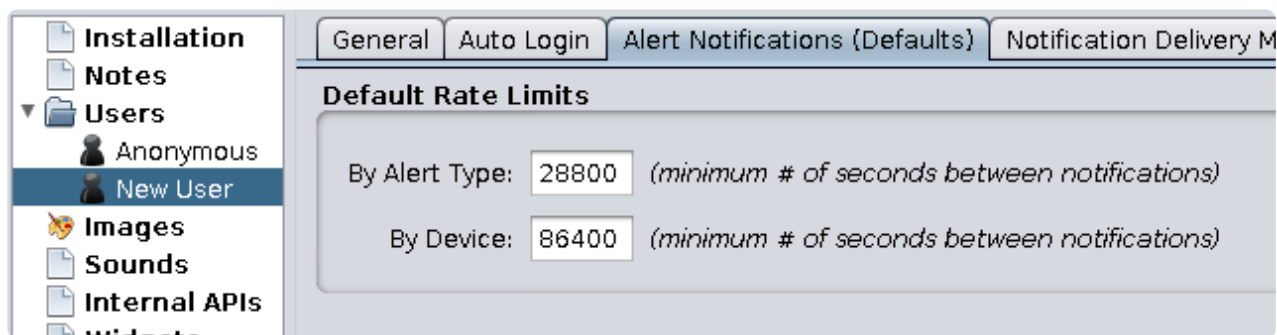


Figure 7.2 | The Auto Login tab

#### ALERT NOTIFICATIONS (DEFAULTS)



The third tab is

Figure 7.3 | Alert Notifications (Defaults) tab

labelled “Alert Notifications (Defaults)”. The settings under this tab how much time (in seconds) are in between notifications by default.

*By Alert Type: 28800 seconds*

*By Device: 86400 seconds*

- The **By Alert Type** input box controls how often a notification is sent for a type of alert. For example, if a user is notified for Camera alerts, they will be notified only once every so often (depending on the value of this setting) if the alert is triggered.
- The **By Device** input box controls how often a notification is sent for a type of device. For example, if a user is notified for when a device is no longer connected to the network, they will only be



notified once every so often (depending on the value of this setting).

## NOTIFICATION DELIVERY METHODS



Figure 7.4 | Notification Delivery Methods tab

The fourth tab is labelled “Notification Delivery Methods”. Using the settings under this tab, one can enable notifications for a user profile that are sent as an email from a Gmail account or as a text message from a phone number. By default, both options are disabled. To enable one option, click on the check box next to that option.

### Gmail Notifications

*Enabled: unchecked*  
*Rate Limit: 180 seconds*  
*Gmail Account: (unset)*  
*Gmail Password: (unset)*  
*Recipient Address: (unset)*

- The **Enabled** checkbox defines whether Gmail notifications are enabled (checked) or not (unchecked).
- The **Rate Limit** input box defines the minimum number of seconds between notification emails.
- The **Gmail Account** input box defines the full Gmail address (including “@gmail.com” that should be used to send the notification emails.
- The **Gmail Password** input box defines the password to use with **Gmail Account** to send the notification emails.
- The **Recipient Address** input box defines which email address the notification emails should be sent to. If left blank, the notification emails will be sent to the **Gmail Account**.

### Plivo/SMS Notifications

*Enabled: unchecked*  
*Rate Limit: 3600 seconds*  
*Auth ID: (unset)*  
*Auth Token: (unset)*  
*Sender Phone #: (unset)*  
*Recipient Phone #: (unset)*

- The **Enabled** checkbox defines whether Plivo/SMS notifications are enabled (checked) or not (unchecked).
- The **Rate Limit** input box defines the minimum number of seconds between notification texts.
- The **Auth ID** input box defines the Plivo Auto ID to use when sending the notification texts. The **Auth Token** input box defines the Plivo Auth token to use when sending the notification texts.

- The **Sender Phone #** input box defines the phone number that will send the notification texts. It must include the country code and area code.
- The **Recipient Phone #** input box defines which phone number the notification texts should be sent to. If left blank, the notification texts will be sent to the **Sender Phone #**. It must include the country code and area code.

## 8. Images

An Image can be added by clicking on the Installation tab in the Menu bar, and then clicking on Images and then on Add Image. Supported image file types are JPEG, PNG, and GIF files. Images can be used as background images for Pages, Buttons, Labels, Slider elements, and Videos. To display an image on a Page, create a Label item, remove the default Text, and add the Image as the Background Image.

## 9. Sounds

A Sound can be added by clicking on the Installation tab in the Menu bar, and then clicking on Add Sounds. Supported audio file types are MP3, WAV, and AIFF files. Sounds can be played when certain actions trigger a series of commands, such as with Macros, Buttons, and Page entering and exiting.

Sounds can be played in Remote Builder to preview it by double-clicking on the Sound. Sounds can be renamed by triple-clicking it.

## 10. Internal APIs

Internal APIs are essentially built-in Drivers, Commanders, and Components, all in one item, with an interface to set them up. An Internal API can be added by clicking on the Installation tab in the Menu bar, and then clicking on Add Internal API. Available Internal APIs include:

- Legrand MR232 API
- Nest API
- Pentair IntelliCenter API
- Philips hue API
- Sony BDP API
- WebOS API
- WeMo API

A user would select an API they wish to add and click on the OK button to add it. Otherwise, they can click the Cancel button. More APIs may be added in the future depending on demand.

### LEGRAND MR232 API

The Legrand/Miro MR232 based lighting system uses RS232 to communicate with the controller. Setup is a multi-step process:

1. First, one or more RS232 ports must be specified for use by the API. Automatically discovered ports that have not been added to the API will appear in the top table. Please allow up to 60 seconds for Global Cache units on the network to be discovered. You may also add an RS232 controller manually by clicking **Manually Add Ports...**
2. Select the port you wish to use from the top table and click **Use Selected Port**. Repeat this process until all of the RS232 ports that have MR232 Network Controllers connected to them have been added.
3. After the desired port(s) have been added, you must select a port from the bottom list and click **Setup Device(s)...** to begin device discovery for the controller connected to that port.
4. During discovery, press the switch or dimmer you wish to add. The system will flash it and request that you enter a name.
5. Continue this process until you have added all of the devices.

### NEST API

To set up the Nest API, you must obtain a PIN code to authorize the server. To obtain a PIN, log into the user's Nest account in a web browser at the following URL:

[https://home.nest.com/login/oauth2?client\\_id=7a0338cc-de1a-481a-bd20-f2d01989c51a&state-STATE](https://home.nest.com/login/oauth2?client_id=7a0338cc-de1a-481a-bd20-f2d01989c51a&state-STATE)

*NOTE: If you copy the URL, please copy everything, including all query parameters.*

After obtaining a PIN, click on the **Setup...** button and enter the PIN in the **PIN** input box and click **OK**.

### PENTAIR INTELLICENTER API

To set up a Pentair IntelliCenter controller, determine its IP address and port (usually 6681) and then click **Setup**. The controller's address should be set to static or given a reservation in the DHCP server as this driver currently has no way to detect address changes.

#### **PHILIPS HUE API**

To set up a Philips Hue Bridge, you must press the link button on the Bridge no more than 30 seconds before clicking **Setup**. Once it appears in the device list, select it and click the **Setup** button.

#### **SONY BDP API**

To set up a Sony Blu-ray player, make sure the device is powered on and at the home screen. The device's name can be set in the **System Settings** menu. Ensure that **Remote Start** is on in the **Network Settings** menu.

*NOTE: Menu options and setting names may be different depending on the device model and/or firmware version*

After selecting the player from the device list below and clicking **Setup**, you may have to enter the PIN code that appears on the TV screen, depending on model/firmware version.

#### **WEBOS API**

WebOS TVs cannot be controlled over the network until **LG Connect Apps** are turned on in the **Network** menu of the TV. The name of the TV should also be set there.

WebOS TVs use WoL to power on over the network. To enable this, activate **Turn on via Wi-Fi** in the **Mobile TV On** setting in the **General** menu. You must turn this setting on even if the TV is connected to the network via ethernet.

*TVs will not appear in the device list unless they are powered on!*

To set up a WebOS TV, select it from the device list, click the **Setup** button and then accept the pairing request that appears on the TV screen. You must accept the request within 30 seconds or the setup will have to be repeated.

#### **WEMO API**

To set up a WeMo switch or dimmer, ensure the device is on the local network and is discoverable on your machine with UPnP.

## 11. Widgets

Widgets provide configurable client-side capability that would otherwise be inappropriate or clunky to add using Remote Builder's page editor. For instance, stock tickers, news feeds, photo album viewers and social networking interfaces can be easily created using standard HTML/CSS/JavaScript tools and then imported as widgets, as opposed to being hand-coded using Remote Builder client scripts and XHTML page objects.

Widgets are written in HTML5 with strict XHTML syntax. They can be fully designed and tested outside the Remote Builder environment and then bundled without modification into a ZIP file to be imported into an installation. Once imported they can be added, dragged, resized, copied, pasted and removed just like any other object on a page.

Remote Builder provides version checking and in the future will automatically update widgets as newer versions become available from our widget repository.

A user can add a new Widget by clicking on the Installation tab in the Menu bar, and then clicking on New Widget.

### FILE STRUCTURE

A widget is to be packaged in an unencrypted ZIP file containing the following files at its root:

- **widget.xml** - Contains the widget specification. See the "Widget XML" section for details.
- **widget.png** - The placeholder image that is displayed in the builder application when a user adds the widget to a page. If this file is not present a checkerboard image will be used instead. The image should be sized to the preferred display size of the widget. In other words, its dimensions should match the "**width**" and "**height**" attributes specified in the widget.xml.
- **widget.xhtml** - Contains the XHTML of the widget. The widget must use strict XHTML 1.0 syntax. The DOCTYPE is:

```
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml11/DTD/xhtml11-strict.dtd">
```

This content is inlined directly into a document body, so your XHTML does not have to be a full document. If it is a standalone document (i.e., it contains the <html> and <body> tags), you MUST wrap the widget markup like this:

```
<!-- WIDGET BEGINS HERE -->

<p>Your widget XHTML would be here.</p>

<!-- WIDGET ENDS HERE -->
```

The widget's container tag will have margin and padding set to 0, and the width and height set to whatever size the user set it to (or the widget's specified size if fixed).

### RESOURCE FILES

Any number of files may be utilized by your widget. Their names and directory structure are arbitrary, but root-level resource files may NOT be named **widget.xml**, **widget.png** or **widget.xhtml**. Those names are ONLY to be used for files conforming to the previously mentioned descriptions. Root-level resource directories SHOULD NOT be named “proxy”, this path is reserved for future use. See the **Resources** section for supported file types.

## WIDGET XML

At the root level of the widget ZIP file structure there MUST be a file named **widget.xml** that describes the widget. The information in this file is used to present the widget to the user, allowing them to specify settings and determine if the version they are using is up-to-date.

All element and attribute names are case-sensitive. See the end of this document for an Example Widget XML.

Tag descriptions:

### <widget>

The widget tag is the top level tag in the XML file. There must be only one widget tag per file.

## Attributes

- **width**

*Expected Value: String, number in pixels. MANDATORY*

*Default Value: None*

*Usage: <widget ... width='300'>*

The preferred width of the widget, in pixels. This attribute is MANDATORY.

- **fixed-width**

*Expected Value: String, either "true" or "false". Optional*

*Default Value: "false"*

*Usage: <widget ... fixed-width='false'>*

If set to “true”, the user will not be allowed to resize the width of the widget from the GUI. This attribute is optional. If omitted, the width is assumed to be fixed.

- **height**

*Expected Value: String, number in pixels. MANDATORY*

*Default Value: None*

*Usage: <widget ... height='300'>*

The preferred height of the widget, in pixels. This field is MANDATORY.

- **fixed-height**

*Expected Value: String, either "true" or "false". Optional*

*Default Value: "false"*



Usage: `<widget ... fixed-height='false'>`

If set to **“true”**, the user will not be allowed to resize the height of the widget from the GUI. This attribute is optional. If omitted, the height is assumed to be fixed.

## Elements

- **<UUID>** - SHOULD be a UUID conforming to RFC4122 (<http://tools.ietf.org/html/rfc4122>). This UUID MUST remain the same across all versions of the widget. Do NOT switch the representation format of the UUID between versions. UUIDs can be generated using operating system specific tools, or on helpful websites such as <https://www.uuidgenerator.net/>. Do NOT reuse UUIDs with different widgets. This element is MANDATORY.
- **<name>** - A friendly name for the widget. SHOULD be short and concise. SHOULD NOT contain version information. This element is MANDATORY.
- **<version>** - The version of the widget contained in the ZIP file. How you wish to represent your version numbers is at your discretion, but subsequent version numbers MUST compare as “greater than” all previously released version numbers using a standard case-sensitive string matching algorithm (assume C `strcmp()`, Java `compareTo()` or JavaScript `>` operator). This element is MANDATORY.
- **<author>** - A friendly name for the author. May be an individual or company name. This element is MANDATORY.
- **<URL>** - A link the user can go to for more information about the widget. This element is optional.
- **<description>** - An in-depth description of the widget. If you choose to also document the widget in the description, the first paragraph MUST be the actual description. Subsequent paragraphs may be used for documentation. This element is optional.
- **<settings>** - Contain names, descriptions and default values for the settings the widget uses to configure itself. The **type** attribute value may be either `global` or `page`. Global settings are set once by the user and apply across all usages of the widget within an Installation. Page settings are set by the user for each page the widget is used in.

Settings are passed to the widget as quoted strings by a JavaScript variable. The name of this variable is set in the **scriptid** attribute for the setting. For instance, if you had a variable with `scriptid MESSAGE` and the user set it to "Hello", the following code would appear in a `<script>` tag before the widget's XHTML:

```
var MESSAGE = "Hello";
```

Use care when picking script IDs in order to avoid collisions with other widgets or script variables created by the user. A recommended convention would be to prefix the ID with the initials of your company name, the product name and an underscore. Do NOT use a double underscore to prefix your script IDs, the double-underscore namespace is reserved.

If no default value is specified it is assumed to be a blank string.

The **scriptid** and **type** attributes are MANDATORY. Changing the type between versions will cause the user to have to reconfigure the widget after an upgrade.

The **name** element is MANDATORY. The name SHOULD NOT be changed across versions unless its purpose has changed. Should you decide to change a setting name, any existing installations will lose

their configuration for that setting and will require manual intervention by the user after an upgrade. Setting names MUST be unique within the widget (i.e., global and page settings cannot share the same name).

The **description** element, if present, SHOULD provide a helpful description of what the setting does and what type of values are expected.

## RESOURCES

A widget may utilize any number of resources such as images, plain text files or XHTML. Resources MUST be referenced relatively by the widget if they are packaged in the zip file. The proper base URL for all resources is /widget/WIDGET-UUID/, where WIDGET-UUID is the widget UUID as specified in the widget.xml.

The server recognizes the type of the resource by its filename extension and will set the Content-Type HTTP header appropriately when it is retrieved. The following extensions are currently supported:

	Extension(s)	Content-Type
1	jpg, jpeg	image/jpeg
2	png	image/png
3	gif	image/gif
4	txt	text/plain
5	css	text/css
6	xml	text/xml
7	xhtml	application/xhtml+xml
8	js	application/javascript

Table 2: Supported Resource File Types

## IMPLEMENTATION NOTES

Widgets should be as compact, straight-forward and single-purpose as possible. Each widget in an Installation will consume as much RAM on the server as at least the size of the ZIP file plus the uncompressed size of its contents. As a rule of thumb, try to keep widget ZIPs under 1MB. Anything larger is potentially violating the “spirit” of widgets.

If possible, widgets should fill their available area automatically. Often times, simply setting the CSS width and height to 100% will do the trick. When this isn't sufficient, the variables `__widget_width` and `__widget_height` are preset with the width and height set from Remote Builder.

## EXAMPLE WIDGET XML

```
<?xml version='1.0' encoding='UTF-8' ?>
<widget width='320' fixed-width='false' height='322' fixed-height='false'>
  <UUID>363BCA43-A3EE-99EF-5D34DD7B7B00A4A1</UUID>
  <name>Jukebox Interface</name>
  <version>1.0</version>
  <author>Physics Systems LLC</author>
  <URL>http://www.physicssystems.com</URL>
  <description>An interface to the Jukebox component.</description>
  <settings>
    <setting type='page' scriptid='PS_JUKEBOX_ZONE_ID'>
      <name>Zone ID</name>
      <description>Which zone this particular instance of the widget
```

```
        should control. Zone numbers start at 1 for the first zone.
        </description>
        <default>1</default>
    </setting>
</settings>
</widget>
```

#### EXAMPLE WIDGET XHTML

You can use this template when writing your own widgets from scratch:

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html
  PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns='http://www.w3.org/1999/xhtml'>
  <head>
    <style type='text/css'>
      body {
        padding: 0;
        margin: 0;
        border: 0;
      }
      #widget {
        width: 100%;
        height: 100%;
      }
    </style>
  </head>
  <body>
    <div id='widget'>
      <!-- WIDGET BEGINS HERE -->

      <!-- WIDGET ENDS HERE -->
    </div>
  </body>
</html>
```

## 12. Modules

A Module can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Module. This brings up a dialog box with a drop-down menu with the available modules. The available modules that can be added are:

- Aloha Spy Server
- HAI Controller (UDP)
- Jukebox
- MyQ
- OPD Server
- Sonos

A user would select a module they wish to add and click on the OK button to add it. Otherwise, they can click the Cancel button. Some Modules have Privileges that can be added to User Default Policy Exceptions.

Each module has settings for controlling the connected devices. Each has a settable URI, along with settings that are unique to the module. Some Modules include certain privileges that can be added to users' Policy Exceptions.

### ALOHA SPY SERVER

*URI: /module/alohaspy/3999*  
*Port: 3999*

- The **URI** is the internet location, relative to the server's root URL, that can be used to send commands to the Aloha Spy Server through this module using the specified settings.
- The **Port** is the internet port used to connect to the Aloha Spy Server.

The Aloha Spy Server module also includes the following module privilege:

- Access Transactions

### HAI CONTROLLER (UDP)

*URI: /module/hai/ebb667fe*  
*Address: 192.168.0.101*  
*Port: 4369*  
*Encryption Key: 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00*  
*Security Code: 0000*

- The **URI** is the internet location, relative to the server's root URL, that can be used to send commands to the Hai Controller through this module using the specified settings.

The Hai Controller (UDP) module also includes the following module privileges:

- Control Alarm
- Control Audio
- Control Lighting

- Control Thermostats

## JUKEBOX

*URI: /module/jukebox/d7b7e7a8*  
*Address: 127.0.0.1*  
*Port: 8082*

- The **URI** is the internet location, relative to the server's root URL, that can be used to send commands to the Jukebox through this module using the specified settings.

The Jukebox module also includes the following module privileges:

- Control Any Zone
- Browse Music Collection
- Manage Media
- Get Zone Status

## MYQ

*URI: /module/myq/0*  
*Username: (unset)*  
*Password: (unset)*

- The **URI** is the internet location, relative to the server's root URL, that can be used to send commands to the MyQ device through this module using the specified settings.

## OPD SERVER

*URI: /module/opd/OPD*  
*Settings File: ~/opd.xml*

- The **URI** is the internet location, relative to the server's root URL, that can be used to send commands to the OPD server through this module using the specified settings.
- The **Settings File** is the path on the server pointing to the XML file that contains the settings for this module.

## SONOS

*URI: /module/sonos/7f9000cf*  
*Root Device UDN: (unset)*

- The **URI** is the internet location, relative to the server's root URL, that can be used to send commands to the Sonos device through this module using the specified settings.
- The **Root Device UDN** is the Sonos device's Unique Device Name (UDN).

The Sonos module also includes the following module privilege:

- Control Sonos unknown (none)

*Note: This module privilege's label may change depending on the settings*

## 13. Drivers

A Driver can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Driver or by using the keyboard shortcut Ctrl/Command+Shift+D. A Driver has 5 tabs each containing settings and code for communicating with hardware. These tabs are “General Settings”, “Serial Settings”, “IP Settings”, “Script”, and “Asynchronous I/O”. There are also buttons above the tabs for adding an IR command, a Serial command, an IP command, a Number property, and an Enum property.

Drivers can only be utilized through Components. See **Components** for more information.

Commands and properties can be copied to other Drivers.

### GENERAL SETTINGS

*Manufacturer: New Driver*

*Model: (unset)*

- The **Manufacturer** input box should contain the name of the manufacturer of the device this driver is for.
- The **Model** input box should contain the model number/name of the device this driver is for.

The above fields define the name of the driver, which begins with the Manufacturer name and ends with the Model name. This allows the Driver to be easily identified throughout the Installation wherever it is used.

The interface for Commands and Properties is displayed below the above settings when they are selected. The last viewed Command or Property of the Driver is always shown until the Driver has been unselected.

### SERIAL SETTINGS

*Data Rate (baud): 9600*

*Data Bits: 8*

*Stop Bits: 1*

*Parity: None*

*Flow Control: None*

- The **Data Rate (baud)** input box defines the data rate of the serial connection.
- The **Data Bits** drop-down box defines whether the data bits are either 7 or 8.
- The **Stop Bits** drop-down box defines whether the stop bits are either 1 or 2.
- The **Parity** drop-down box defines whether the parity is “Even”, “Odd” or “None”.
- The **Flow Control** drop-down box defines if the flow control is either through the hardware or none.

### IP SETTINGS

*Stay Connected: 0 seconds after last IP command*

*IP Command Type: Normal*

*HTTP Authentication Type: None*

*On IP Connect Command: (unset)*

- The **Stay Connected** text box defines how many seconds the Installation Server should stay connected to the device after the last IP command has been sent. Subsequent commands that occur before this time has elapsed will reuse the connection and reset the timer. 0 or less will cause a new connection to be made for each command sent.
- The **IP Command Type** drop-down box defines the command type, whether it is “Normal” or “Telnet”. This setting does not apply to HTTP commands.
- The **HTTP Authentication Type** drop-down box defines the type of HTTP authentication the Installation Server should use when connecting to the device. Options are “None”, “Basic”, and “Digest”. If HTTP IP commands need authentication, specify the type here. The username and password values are set per Component. This setting only applies to HTTP commands.
- The **On IP Connect Command** script text box should contain a Server-side script that will be sent the first time the Installation Server connects to the device via IP. Every time a Component using the Driver is first connected to via IP, this script will be sent first and only once before any other IP commands are sent. This can be used, for instance, to log a user into the system before issuing any real commands. The script may be raw ASCII or a (JavaScript) function generator that returns a UTF-8 string. Raw ASCII strings may use the escape sequences for newline (“\n”) and carriage return (“\r”). Function generator parameters are assigned in the Component, under the “IP Settings” tab.

## SCRIPT

See **Scripting** for more information.

## ASYNCHRONOUS I/O

*Driver is Asynchronous: unchecked*  
*Parser Script: (unset)*

- The **Driver is Asynchronous** checkbox defines whether the Driver is asynchronous (checked) or synchronous (unchecked). If it is asynchronous, you will need a special parser script (see below).
- The **Parser Script** text box is disabled when the “Driver is Asynchronous” checkbox is unchecked, but enabled when it is checked. The script is ran to parse the response from the device.

## 14. Cameras

A Camera can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Camera or with the keyboard shortcut **Alt+I+A**. A Camera has seven (7) tabs, some of which may not be accessible at all times, depending on certain values set in other tabs.

Numerous models are supported, and some unlisted models may still work with similar models.

Cameras can be added to User Default Policy Exceptions.

### GENERAL

*Enabled: checked*

*Name: New Camera*

*Model: ONVIF*

*IP Address / UPnP UDN: (unset)*

*HTTP Port: 80*

*RTSP Port: 554*

*ONVIF Port: 80*

*Username: (unset)*

*Password: (unset)*

*Default Mode: Armed*

- The **Enabled** checkbox defines whether this Camera is enabled (checked) or disabled (unchecked). When it is disabled, NVR is prevented from accessing it; no connections will be attempted to it and it will not be available for viewing, recording, or motion detecting.
- The **Name** input box defines the name of the Camera. This reflected throughout the Installation, wherever the Camera is used. It would be a good idea to not have any Cameras share the same name.
- The **Model** drop-down defines which camera model this is being connected to. This is important to set correctly so the Installation can properly communicate with the connected camera. The following are the available options:
  - Arecont Vision AV20175DN
  - Arecont Vision AV20185DN
  - Arecont Vision AV20365
  - Arecont Vision 40185DN
  - Arecont Vision AV6656DN
  - Arecont Vision AV815DN
  - AXIS A8105-E
  - AXIS M3113-R
  - AXIS M3114
  - AXIS M5014
  - AXIS M5054
  - AXIS M7014 Encoder
  - AXIS P3707-PE



- AXIS P5512
  - AXIS VAPIX V3 720p
  - AXIS VAPIX V3 1080p
  - Dahua Generic HTTP API V1 NTSC (HTTP)
  - Dahua Generic HTTP API V1 1080p NTSC (RTSP)
  - Dahua Generic HTTP API V1 720p NTSC (RTSP)
  - DoorBird D202
  - Foscam FI8910
  - Grandstream GXV3504
  - Grandstream GXV3504 (4\*D1 Mode)
  - HIKVISION HNC304-XD
  - ONVIF
  - Physics Systems Screen Capture
- The **IP Address / UPnP UDN** input box records the Camera's IP address. If you are using UPnP, you can enter the camera's UPnP UDN instead and the IP address will be automatically determined.
  - The **HTTP Port** input box defines the port used for HTTP communications to the Camera.
  - The **RTSP Port** input box defines the port used for RTSP communications to the Camera. This is only available when **Model** is set to "Dahua Generic HTTP API V1 1080p NTSC (RTSP)", "Dahua Generic HTTP API V1 720p NTSC (RTSP)", or "ONVIF".
  - The **ONVIF Port** input box defines the port used for ONVIF communications to the Camera.
  - The **Username** input box defines the username to use with the **Password** when connecting to the Camera.
  - The **Password** input box defines the password to use with the **Username** when connecting to the Camera.
  - The **Default Mode** drop-down defines the default recording mode of the Camera. Options are "Stopped" (doesn't record unless specifically instructed to), "Armed" (stays connected to the camera, records immediately upon motion detection or some other trigger), or "Continuous" (stubbornly records at all times until camera is placed into stopped or armed mode).

## VIDEO

*Channel: 1*

*Resolution: 1280x720*

*FPS: 30.0*

*Quality: 70*

*Bitrate: 16384*

*Use Binning When Appropriate: unchecked*

- The **Channel** drop-down box defines which Camera channel is monitored/recorded. This is only applicable to multi-channeled Cameras. Thus this is only enabled when **Model** is set to any *Arecont Vision*, "AXIS M7014 Encoder", "AXIS P3707-PE", any *Grandstream*, and "Physics Systems Screen Capture".
- The **Resolution** drop-down box sets the Camera's video resolution. Available options depend on the **Model**.

- The **FPS** input box sets the desired frames per second. Setting to -1 uses the Camera's default FPS value.
- The **Quality** input box defines the compression quality. Setting to -1 uses the Camera's default value.
- The **Bitrate** input box defines the bitrate in kbps. Some cameras use this instead of **Quality**. This only applies to ONVIF cameras.
- The **Use Binning When Appropriate** checkbox defines whether to use pixel binning when appropriate. This option is only available when **Model** is set to any *Arecont Vision* except "Arecont Vision AV8185DN".

## MOTION

*Motion Detect in NVR: checked*  
*Detection FPS: 4.0*  
*Refresh Interval (seconds): 1.0*  
*Motion Threshold (RMSE %): 8.0*  
*Stop Delay (seconds): 2.0*  
*Preview FPS: 1*

- The **Motion Detect in NVR** checkbox defines whether motion detection will be done in NVR. If unchecked, motion detection will be disabled except on some supported camera models, and the settings under **NVR Motion Detection Settings** will be disabled.
- The **Detection FPS** input box is under **NVR Motion Detection Settings**. This defines how many times per second motion will be checked for. Lower values use less CPU but could miss fast moving objects that enter and leave the scene entirely between motion checks. Motion will never be checked at a rate greater than the video frame rate.
- The **Refresh Interval (seconds)** input box is under **NVR Motion Detection Settings**. Motion detection works by comparing recent frames to an older frame (the reference frame). The **Refresh Interval** specifies how many seconds to use the same reference frame before replacing it with a newer frame. Larger intervals will make the detector more sensitive to slower changes that occur over the span of many frames, but also more sensitive to camera noise.
- The **Motion Threshold (RMSE %)** input box is under **NVR Motion Detection Settings**. This ranges from 0 to 100. Higher values make the detector harder to trip. When motion detecting, the current frame is split up into regions and each region is compared to the reference frame. The difference between them is the RMSE. The higher the **Motion Threshold**, the greater a change is required between the frames in order for the motion detector to trip. A threshold of 100% requires a region to go from completely white to completely black, or vice versa. The correct value to use must be determined by monitoring the RMSE over a period of time.
- The **Stop Delay (seconds)** input box is under **NVR Motion Detection Settings**. The number of seconds that no motion must occur before NVR will be notified that motion has stopped. If any motion occurs during the stop delay, the delay is reset. This value is limited to a maximum of 15 seconds.
- The **Preview FPS** input box is under **Motion Windows**.

For more information about motion detection, see the section **Motion Detection**.

## OVERLAY

*Overlay Data/Time: checked*

*Overlay Title: checked*

*Set Title: (unset)*

- The **Overlay Date/Time** checkbox, when checked, instructs the camera to overlay the current data and time on top of the image.
- The **Overlay Title** checkbox, when checked, instructs the camera to overlay the channel title on top of the image.
- The **Set Title** input box defines the camera title to be set when connecting to the camera. If left blank, the camera title will not be changed.

These settings are only enabled when **Model** is any *Axis* or *Dahua* camera. If **Model** is a *Grandstream* camera, **Overlay Date/Time** is enabled.

## AUDIO

*Enable streaming of live audio (via NVR API): unchecked*

*Audio Channel: 1*

*Audio Format: G711Mu, 8000Hz, 640000bps*

- The **Enable streaming of live audio (via NVR API)** checkbox
- The **Audio Channel** drop-down defines which audio channel is to be used. Available values depends on the selected **Model**. This option is usually disabled.
- The **Audio Format** drop-down usually has only one option “G711Mu, 8000Hz, 640000bps”, which (of course) is default.

This tab is only enabled when **Model** is set to “AXIS A8105-E”, “AXIS M5014”, “AXIS M5054”, or “DoorBird D202”.

## INPUTS

*Trigger Event(s): None*

- Each cell in the **Trigger Event(s)** column can be set to either be “None”, “Active”, “Inactive”, or “Both”.

This tab is only enabled when **Model** is set to “AXIS A8105-E” or “DoorBird D202”. The number of inputs depends on the **Model**: “AXIS A8105-E” has three (3) ports and “DoorBird D202” has one (1) port.

## ALERTS

*Max Connection Fail Time: 120 (seconds)*

*Max Lost Connections Hourly: 1*

*Max lost Connections Daily: 3*

- The **Max Connection Fail Time** is the maximum number of seconds of sequential failed connections allowed before alerting.
- The **Max Lost Connections Hourly** is the maximum number of times in one hour that the connection may be lost to this camera without generating an alert.
- the **Max Lost Connections Daily** is the maximum number of times in one day that the connection may be lost to this camera without generating an alert.

## 15. Servers

A Server is another device with Installation Server installed and running. By adding a Server to an Installation, commands can be sent to the connected Server, and its Cameras can be viewed on the Installation. A Server can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Remote Server. Servers can be added to User Default Policy Exceptions.

Each Server has three (3) tabs: Settings, Cameras, and Alerts.

### SETTINGS

*Enabled: checked*  
*Name: New Server*  
*Address: (unset)*  
*HTTP Port: 80*  
*HTTPS Port: 443*  
*Use HTTPS: unchecked*  
*Username: (unset)*  
*Password: (unset)*

- The **Enabled** checkbox enables (checked) or disables (unchecked) connecting to this STS (site-to-site) server. If it is disabled, no connections will be attempted to it and its cameras will not be available for viewing on local video walls.
- The **Name** input box defines the display name of the server that is used to identify it throughout the Installation.
- The **Address** input box defines the URI of the server, which can be an IP address or a domain/subdomain. Only use the host section of the URI, without the scheme and port.
- The **HTTP Port** input box defines which port to use for HTTP connections.
- The **HTTPS Port** input box defines which port to use for HTTPS connections.
- The **Use HTTPS** checkbox defines whether to connect to the server using the HTTPS port (checked) or using the HTTP port (unchecked). It can only use HTTPS if the server has a valid site certificate and is not on the LAN.
- The **Username** input box defines the username that the Installation should use to connect to the server with.
- The **Password** input box defines the password that the Installation should use to connect to the server with.

There is also a button labelled “Test Connection” that, when clicked on, attempts to connect to the specified server using the above settings. The results from the attempt are shown in a popup window.

### CAMERAS

The Camera(s) that are on the Server are listed in the tab, showing the Cameras’ name, width, height, FPS, whether or not audio capturing is enabled, and whether or not audio playback is enabled. At the bottom, there is a button labelled “Reload Camera List” that, when clicked on, queries the Server for a list of Cameras and updates this list accordingly.

### ALERTS

*Max Connection Fail Time: 900 (seconds)*

*Max Lost Connections Hourly: 1*

*Max lost Connections Daily: 3*

- The **Max Connection Fail Time** is the maximum number of seconds of sequential failed connections allowed before alerting.
- The **Max Lost Connections Hourly** is the maximum number of times in one hour that the connection may be lost to this server without generating an alert.
- the **Max Lost Connections Daily** is the maximum number of times in one day that the connection may be lost to this server without generating an alert.

## 16. Commanders

A Commander can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Commander or with the keyboard shortcut **Alt+I+O**. Commanders can be added to User Default Policy Exceptions.

A Commander has two (2) tabs: Setup, and IR Triggering. It can also have IP Commands, and Relay Commands. For IP Commands, see the section **IP Commands**. For Relay Commands, see the section **Relay Commands**.

### SETUP

*Name: New Commander*  
*Profile: Webports*  
*Address: (unset)*  
*Port: 8080*  
*# IR Ports: 0*  
*# Serial Ports: 0*  
*# Relays: 0*  
*# GPIO Pins: 0*

- The **Name** input box defines the display name of the Commander that is used to identify it throughout the Installation.
- The **Profile** drop-down box defines the profile to use. Since the Webports profile is currently the only profile that exists, this drop-down box is disabled.
- The **Address** input box sets the URI to use to connect to the Commander's server.
- The **Port** input box sets the port to use with the **Address** when connecting to the Commander's server.
- The **# IR Ports** input box defines how many IR ports the Commander has.
- The **# Serial Ports** input box defines how many serial ports the Commander has.
- The **# Relays** input box defines how many relays the Commander has.
- The **# GPIO Pins** input box defines how many GPIO pins the Commander has.

### IR TRIGGERING

There are 16 checkboxes, each labelled with its corresponding number, beginning with 0. Each are initially, disabled. How many are enabled depends on the number of **IR Ports** specified in the **Setup** tab. For each that are enabled, they can be checked to enabled IR triggering on the corresponding IR port.

Beneath the checkboxes is a text box labelled "Live IR Trigger Watcher" that contains received IR codes on the triggering ports. This requires Asynchronous I/O to be started. There is also a button beneath this labelled "Clear" that will empty the text box.

## 17. Matrices

A Matrix can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Matrix.

Matrices can be added to User Default Policy Exceptions.

*Name: New Matrix*  
*Model: Netgear M4100 Series Switch*  
*Address: (unset)*  
*Port: 23*  
*Username: (unset)*  
*Password: (unset)*  
*Base VLAN ID: 1000*

- The **Name** input box defines the display name of the Matrix that is used to identify it throughout the Installation.
- The **Model** drop-down box defines which API to use with this Matrix. Currently, only one is available: “Netgear M4100 Series Switch”.
- The **Address** input box is the URI to the switch.
- The **Port** input box is the port that the Matrix should use when communicating with the switch, with the **Address**.
- The **Username** input box is the username of the switch.
- The **Password** input box is the password of the switch.
- The **Base VLAN ID** input box contains the base VLAN ID of the VLANs. VLANs generated for matrix switching purposes will have VLAN IDs, which are the base VLAN ID + the port number.

### PORT

*Name: New Port*  
*Type: Destination*  
*Port Number: 1*

- The **Name** input box defines the display name of the Port that is used to identify it throughout the Installation.
- The **Type** drop-down box sets the type of Port this is. It can either be “Destination” or “Source”.
- The **Port Number** input box defines the port number of the Port.

## 18. IR Remotes

An IR Remote can be added by clicking on the Installation tab in the Menu bar, and then clicking on New IR Remote or by using the keyboard shortcut **Alt+I+R**.

*Remote Name: New IR Trigger Remote*

- The **Remote Name** input box defines the display name of the IR Remote that is used to identify it throughout the Installation.

There is also a button labelled “Add Button” for adding an IR remote button.

### **BUTTON**

*Button Name: New IR Trigger Remote*

*IR Code: (unset)*

- The **Remote Name** input box defines the display name of the IR Remote that is used to identify it throughout the Installation.
- The **IR Code** input box defines the IR code to be sent. The button labelled “Capture” is used to capture IR trigger codes, but only when Asynchronous I/O has been started.



## 19. Components

### DRIVER COMPONENTS

A Driver Component can be added by clicking on the Installation tab in the Menu bar, clicking on Components, and then on New Driver Component, or by using the keyboard shortcut Ctrl/Command+Shift+C. Driver Components can be added to User Default Policy Exceptions.

*Name: New Component*

*Driver: (unset)*

- The **Name** input box defines the display name of the Component that is used to identify it throughout the Installation.
- The **Driver** drop-down box defines which Driver this Component works with.

### Commander Settings

*Commander: (unset)*

*IR Port: (unset)*

*Serial Port: (unset)*

- The **Commander** drop-down box
- The **IR Port** drop-down box
- The **Serial Port** drop-down box

### IP Settings

*IP Address / UPnP UDN: (unset)*

*Port: 0*

- The **IP Address / UPnP UDN** input box
- The **Port** input box

### Authentication

*Username: (unset)*

*Password: (unset)*

- The **Username** input box
- The **Password** input box

### UPNP COMPONENTS

A UPnP Component can be added by clicking on the Installation tab in the Menu bar, clicking on Components, and then on Add UPnP Component. A popup window appears, showing UPnP devices on the network. When a device is selected, and depending on the device, the “Add to Components” button will be enabled, allowing the user to add that device as a Component. UPnP Components can be added to User Default Policy Exceptions.

*Name: (no default)*

*Root Device UDN: (no default)*

- The **Name** input box defines the display name of the Component that is used to identify it throughout the Installation. This is initially set based on the advertised name of the UPnP device.
- The **Manufacturer** label shows the manufacturer of the UPnP device.
- The **Model Number** label shows the model number of the UPnP device.
- The **Root Device Type** label shows the root device type of the UPnP device.
- The **Root Device UDN** input box shows the root device UDN. This is initially set based on the advertised UDN of the UPnP device.
- The **Current Address** label shows the current network address of the UPnP device.

## 20. Scripting

### SCRIPTS

Scripts are written in JavaScript, and can be included in many different elements, such as Pages, Buttons, and Schedules. They are client-side scripts, thus they are run by the browser.

A Script can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Script.

### SCRIPT API

Please refer to this [page](#) for information about our Script API.

## 21. Macros

A Macro can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Macro. Macros can be added to User Default Policy Exceptions.

*Name: New Macro*

*Actions: (unset)*

- The **Name** input box defines the display name of the Macro that is used to identify it throughout the Installation.
- The **Actions** box contains four (4) buttons for adding Actions to the Macro. These Actions are shown in the white space beneath it. Other items in the Tree that are Actions can be dragged-and-dropped here. Actions can be rearranged as needed. The available buttons are “Add Delay”, “Add Server Script”, “Add Timer Action”, and “Add Matrix Action”.
  - The “Add Delay” button gives a user the option to add a delay of any amount in milliseconds, the default being 500 milliseconds.
  - The “Add Server Script” button adds a Server Script. (see the section **Scripting**)
  - The “Add Timer Action” button adds a Timer. (see the section **Timers** under the **Pages** section)
  - The “Add Matrix Action” button adds a Matrix action. There are drop-down boxes for selecting a Matrix, selecting a Destination Port, and selecting a Source Port, each of which are unset by default. (see the section **Matrices**)

## 22. Triggers

Triggers are server-side scripts that are run when an event of specified conditions occurs, such as a user logging into the Installation.

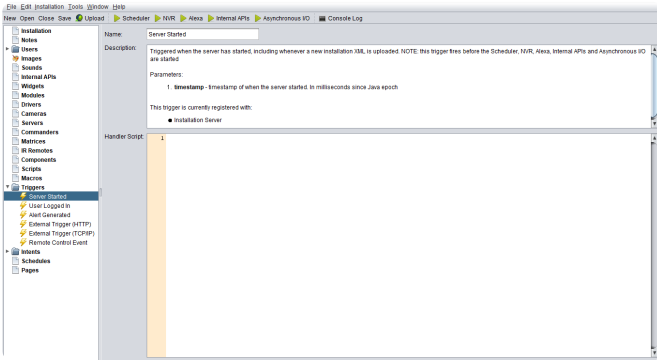


Figure 22.1

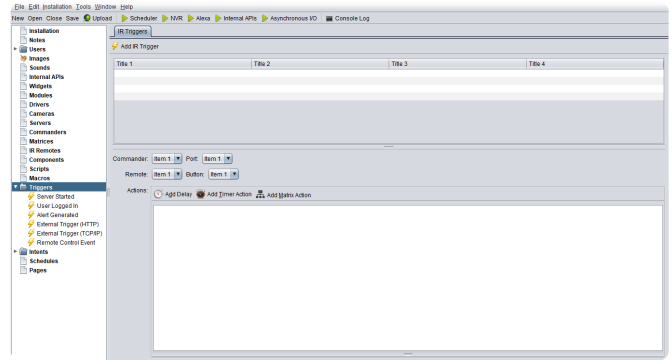


Figure 22.2

Default Triggers are:

- **Server Started** - The handler script is run when the server starts, including whenever a new Installation XML is uploaded, but before Scheduler, NVR, Alexa, Internal APIs, and Asynchronous I/O starts. The script has the variable `timestamp` available to it, which is a timestamp of when the server started in milliseconds since Java epoch.
- **User Logged In** - The handler script is run when a user logs in. The script has the variable `username` available to it, which is the name of the user.
- **Alert Generated** - The handler script is run when any alert is generated. The script has the variables `timestamp` (timestamp of the alert, in milliseconds since Java epoch), `subsystem` (alert subsystem), `typeName` (alert type name), `deviceName` (alerting device, which can be null), and `message` (alert message) available to it.
- **External Trigger (HTTP)** - The handler script is run when the server receives an HTTP request using the URL format `http://SERVER_ADDRESS/trigger?type=Custom&event=EVENT&data=DATA`. For this to work, the username and password in the 'HTTP & IP Triggers' tab in the Installation settings must be set. The parameter names in the trigger URL are case sensitive (e.g. `event=EVENT` is not equal to `Event=EVENT`) The script has the variables `event` (value of the event parameter in the URL), and `data` (value of the data parameter in the URL) available to it. External pages (as well as internal pages, using client scripts or XHTML, which should rarely be necessary) or external devices can send a GET HTTP request using this URL format and the specified authentication in the Installation settings to set off this trigger.
- **External Trigger (TCP/IP)** - The handler script is run when a remote device sends a TCP/IP packet to the IP trigger port. It only fires if 'IP triggers' are enabled in the 'HTTP & IP Triggers' tab in the Installation settings. See the 'External Triggers (TCP/IP)' section of the Script API manual for instructions on how to utilize this trigger. The script has the variables `host` (the IP address of the sending device), `port` (the remote port of the sending device), and `IO` (the IO script API object that can be used to consume any available data that was sent by the device) available to it.
- **Remote Control Event** - The handler script is run by a remote control device when a button or key is pressed or released. The script has the variables `remote` (name of the remote device, which is the 'Device name' assigned in the setup page of the Remote Control Server; if no device name has been assigned yet, this value will be null), `type` (the type of the button pressed, either 'number',

'letter', 'symbol', or 'button'), state (the current state of the button, either 'up', 'repeat', or 'down'), and value available to it.

Triggers can be created, using the interface shown when the Triggers folder is clicked.

Macros run by a trigger are run with the username of “Trigger”.

## 23. Intents

A user cannot add a new Intent.

*Name: (no default)*

*Handler Script: (unset)*

- The **Name** text input defines the display name of the Intent that used to identify it throughout the Installation.

## 24. Schedules

A Schedule is typically used to schedule events or scripts to be started.

A Schedule can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Schedule.

### SCHEDULE

*Name: Schedule*

*Macro: (unset)*

*Run Every: Once*

*Second: (current second)*

*Hour: (current hour)*

*Minute: (current minute)*

*AM/PM: (based on current time)*

*Time Zone: (current time zone)*

*Year: (current year)*

*Month: (current month)*

*Day: (current day)*

*Exactly: 0 minutes*

*Before/After: Before*

*Latitude: 0.0 degrees*

*Longitude: 0.0 degrees*

- The **Name** input box defines the display name of the IR Remote that is used to identify it throughout the Installation.
- The **Macro** drop-down box sets which Macro should be run when the Schedule runs.
- The **Run Every** drop-down box defines how often the Macro should be run. Depending on this value, the following options may be visible. The next three (3) tabs are enabled or disabled depending on this value as well. Possible values are “Minute”, “Hour”, “Day”, “Week”, “Month”, “Year”, “Once”, “Sunrise”, “Sunset”.
- The **Second** input box defines which second of every minute the Macro should run. This is only available when **Run Every** is set to “Minute”.
- The **Hour** input box defines which hour of the specified day the Macro should run. This is only available when **Run Every** is set to “Day”, “Week”, “Month”, “Year”, or “Once”.
- The **Minute** input box defines which minute of the specified hour the Macro should run. This is only available when **Run Every** is set to “Hour”, “Day”, “Week”, “Month”, “Year”, or “Once”.
- The **AM/PM** drop-down box defines whether the specified time is in the AM or PM. This is only available when **Run Every** is set to “Day”, “Week”, “Month”, “Year”, or “Once”. Available options are “AM” and “PM”.
- The **Time Zone** drop-down box defines which time zone the time specified is in. This is only available when **Run Every** is set to “Day”, “Week”, “Month”, “Year”, or “Once”. There are many available options, certainly too much to list all of them here.
- The **Year** input box defines which year the Macro should run. This is only available when **Run Every** is set to “Once”.



- The **Month** drop-down box defines which month the Macro should run. This is only available when **Run Every** is set to “Year” or “Once”. The available options are “January”, “February”, “March”, “April”, “May”, “June”, “July”, “August”, “September”, “October”, “November”, and “December”.
- The **Day** drop-down box defines which day the Macro should run. This is only available when **Run Every** is set to “Week”, “Month”, “Year”, or “Once”. The available values depend on **Run Every** and sometimes the **Year** and **Month**. If **Run Every** is set to “Week”, the available options are “Sunday”, “Monday”, “Tuesday”, “Wednesday”, “Thursday”, “Friday”, and “Saturday”. If **Run Every** is set to “Month”, the available options range from 1 to 30. If **Run Every** is set to “Year” or “Once”, it depends on the **Year** (if “Once”) and/or **Month**. If the **Year** is a leap year, and/or the **Month** is a February, the available options could range from 1 to 28 or 1 to 29. If the **Month** is set to a month that has 31 days, then the available options range from 1 to 31. Otherwise, the available options range from 1 to 30.
- The **Exactly** input box defines how many minutes before/after sunrise/sunset the Macro should be run. This is only available when **Run Every** is set to “Sunrise” or “Sunset”.
- The **Before/After** drop-down box defines if the Macro should be run before or after sunrise/sunset. This is only available when **Run Every** is set to “Sunrise” or “Sunset”.
- The **Latitude** input box defines the latitude in degrees where the sunrise or sunset should be measured by. This is only available when **Run Every** is set to “Sunrise” or “Sunset”.
- The **Longitude** input box defines the longitude in degrees where the sunrise or sunset should be measured by. This is only available when **Run Every** is set to “Sunrise” or “Sunset”.

When **Run Every** is set to “Sunrise” or “Sunset”, there are also two (2) buttons available for automatically filling in the **Latitude** and **Longitude**. These buttons are labelled as “Lookup By Zip Code...” and “Guess By IP...”. The “Lookup By Zip Code...” button gets the latitude and longitude from a zip code. The “Guess By IP...” button guesses the latitude and longitude based on the given IP address.

#### EXCLUDED MONTHS

This tab contains a checkbox for each month of the year. By default, all of them are unchecked, and by checking them, that month is excluded in the scheduling.

#### EXCLUDED WEEKS

By Week of: Month

- The **By Week of** drop-down box defines how the weeks to include/exclude are listed. This option is currently disabled. Otherwise, potential values could be “Month” and “Year”.

This tab also contains a checkbox for each week of the month. By default, all of them are unchecked, and by checking them, that week is excluded in the scheduling. Week 0 is the first week of the month, but only if it is a partial week. Week 1 is the first full week.

#### EXCLUDED DAYS

By Day of: Week

- The **By Day of** drop-down box defines how the days to include/exclude are listed. This option is currently disabled. Otherwise, potential values could be “Week”, “Month”, and “Year”.

This tab contains a checkbox for each day of the week. By default, all of them are unchecked, and by checking them, that day is excluded in the scheduling.

## 25. Pages

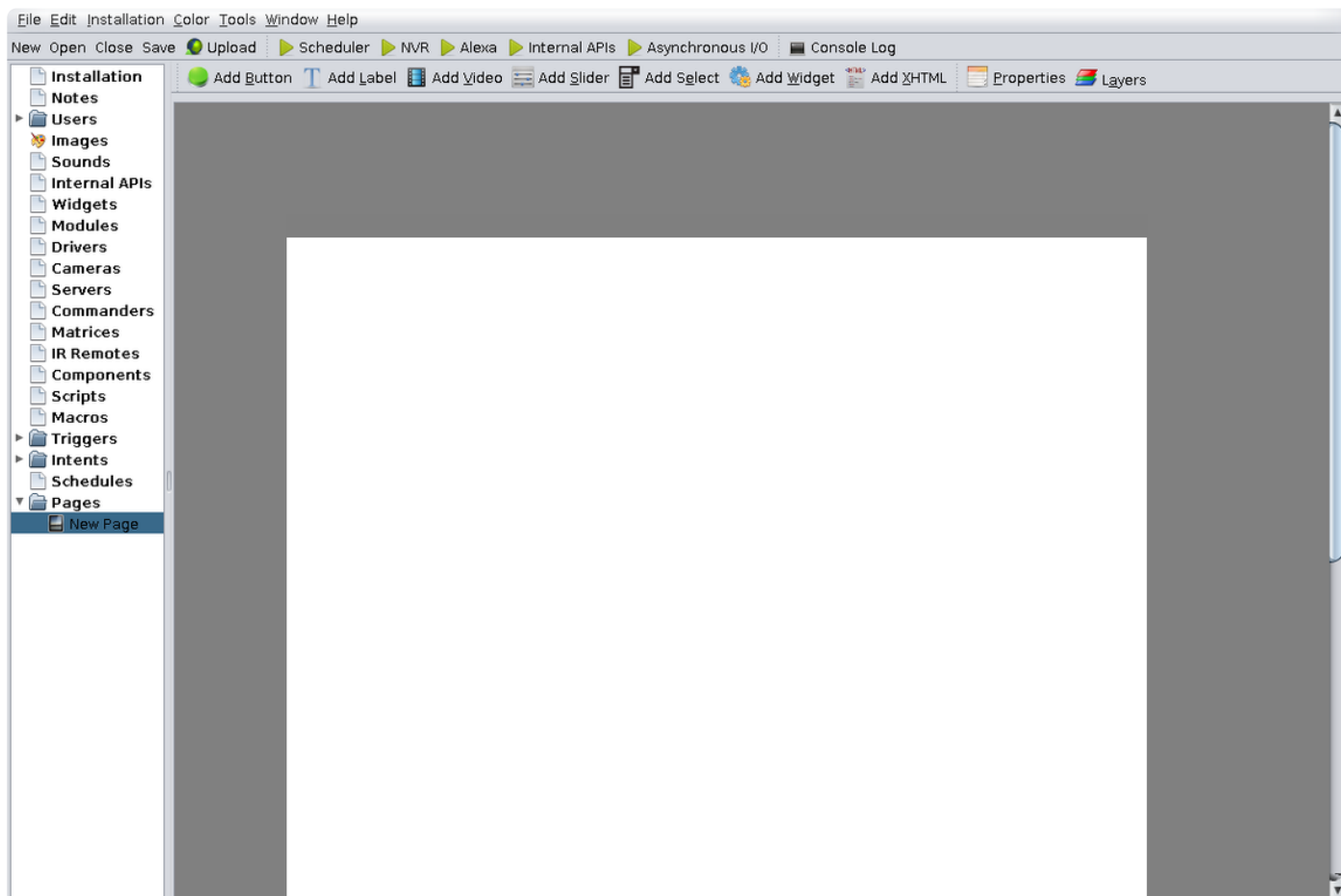


Figure 25.1 | New Page

A Page can be added by clicking on the Installation tab in the Menu bar, and then clicking on New Page or by using the keyboard shortcut `Ctrl/Command+Shift+P`. Pages are the main user-interface of the Installation. A user can set up a page for the purpose of controlling connected devices. Pages can be added to User Default Policy Exceptions.

After creating a Page, the user will be shown a blank page with the default dimensions of 768x1024, optimized for an iPad running iOS 4.3 or greater. The pages origin (where the coordinate (0, 0) would be) is located at the top left corner of the page. (Figure 5.1)

### PAGES TASKBAR

An additional Taskbar can be seen at the top containing nine (9) buttons: "Add Button", "Add Label", "Add Video", "Add Slider", "Add Select", "Add Widget", "Add XHTML", "Properties", and "Layers". Clicking on each of the "Add..." buttons will add elements to the page, and are there to make editing and designing pages easier. The "Properties" button shows a dialog box for configuring the selected element, or if no element is selected, it shows options for configuring the page. (See the section on **Properties** for more information)

### BUTTONS

Clicking on the "Add Button" button in the Pages Taskbar creates a new button element. The Properties dialog box of a Button has four (4) tabs: "Initial", "Press", "Repeat", and "Release".

## Initial

*Script ID: (unset)*

*Position: semi-random set of two numbers: 1st is X-coordinate, 2nd is Y-coordinate, in pixels*

*Label: "Button"*

*Style: Normal*

*Font Size: 18 pixels*

*Label Color: #000000 (black)*

*Background Color: #FFFFFF (white)*

*Z-index: 0*

*Size: (100px, 100px)*

*Opacity: 100%*

*Font: SansSerif*

*Weight: Normal*

*Image: (unset)*

The "Initial" tab contains options for configuring the design of the Button.

- The **Script ID** text box defines a string of text that can be used to refer to the Button in scripts. (See **Scripting**)
- The **Position** defines where the Button is located, in pixels, in relation to the page origin, which is the top left corner.
- The **Label** defines the text on the Button. To remove the Label, leave the text box empty.
- The **Style** defines whether the text is italicized (*Italic*) or not (*Normal*).
- The **Font Size** is the size of the text in pixels.
- The **Label Color** defines the text color.
- The **Background Color** defines the color of the Button.
- The **Z-index** defines which layer the Button is on (see **Layers**).
- The **Size** defines the width and length of the Button in pixels.
- The **Opacity** defines how opaque the Button is in percentages (0% is transparent, 100% is opaque).
- The **Font** defines the text font, which can be "Monospaced", "Serif", or "SansSerif". Other font options are not available.
- The **Weight** defines if the text is bold (**Bold**) or not (*Normal*).
- The **Image** defines an Image that would be the background of the Button. This image is on top of the Background Color, and is underneath the Label. Images must have been added already for it to show in this drop-down menu. (See **Images** for more information)

## Press

*Label Color: #FFFFFF (white)*

*Background Color: #808080 (a shade of gray)*

*Image: (unset)*

Items specified here are run as soon as the Button is pressed. When pressed, the Button's text color and background color and image are changed to the specified values. Items that can be added are:

- Delays (add with button labelled “Add Delay”)
- Client Scripts (add with button labelled “Add Client Script”)
- Timer Actions (add with button labelled “Add Timer Action”)
- Matrix Actions (add with button labelled “Add Matrix Action”)
- Sounds (drag and drop from Tree)
- Component Commands (drag and drop from Tree)
- Commander Commands (drag and drop from Tree)
- Pages (drag and drop from Tree)

## Repeat

*Wait time: 250 milliseconds*  
*Repeat after every: 500 milliseconds*

Items specified here are run when the Button is held for the specified “Wait” time and run continuously every so milliseconds specified in the text box next to “repeat after every”. Items that can be added are:

- Delays (add with button labelled “Add Delay”)
- Client Scripts (add with button labelled “Add Client Script”)
- Timer Actions (add with button labelled “Add Timer Action”)
- Matrix Actions (add with button labelled “Add Matrix Action”)
- Sounds (drag and drop from Tree)
- Component Commands (drag and drop from Tree)
- Commander Commands (drag and drop from Tree)
- Pages (drag and drop from Tree)

## Release

Items specified here are run after the Button is released. Items that can be added are:

- Delays (add with button labelled “Add Delay”)
- Client Scripts (add with button labelled “Add Client Script”)
- Go Back Action (add with button labelled “Add Go Back”)
- Timer Actions (add with button labelled “Add Timer Action”)
- Matrix Actions (add with button labelled “Add Matrix Action”)
- Sounds (drag and drop from Tree)
- Component Commands (drag and drop from Tree)
- Commander Commands (drag and drop from Tree)
- Pages (drag and drop from Tree)

## LABELS

Clicking on the “Add Label” button in the Pages Taskbar creates a new label element. The Properties dialog box doesn't have any tabs, but only has similar options like that of a Button's “Initial” tab.

*Script ID: (unset)*  
*Position: semi-random set of two numbers: 1st is X-coordinate, 2nd is Y-coordinate, in pixels*

*Font: SansSerif*  
*Text: Label*  
*Font Size: 18 pixels*  
*Bold: Unchecked*  
*Italic: Unchecked*  
*Text Color: #000000 (black)*  
*Background Color: unset (transparent)*  
*Z-index: 0*  
*Size: (160px, 32px)*  
*Opacity: 100%*  
*Text Position: Middle*  
*Image: (unset)*

- The **Script ID** text box defines a string of text that can be used to refer to the Label in scripts. (See **Scripting**)
- The **Position** defines where the Label is located, in pixels, in relation to the page origin, which is the top left corner.
- The **Font** defines the text font, which can be “Monospaced”, “Serif”, or “SansSerif”. Other font options are not available.
- The **Text** defines the text on the Label. To remove the Text, leave the text box empty, although doing that defeats the purpose of having a Label.
- The **Font Size** is the size of the text in pixels.
- The **Bold** checkbox defines whether the Text is bold (checked) or normal (unchecked).
- The **Italic** checkbox defines whether the Text is italicized (checked) or normal (unchecked).
- The **Text Color** defines the text color.
- The **Background Color** defines the color of the Label.
- The **Z-index** defines which layer the Label is on (see **Layers**).
- The **Size** defines the width and length of the Label in pixels.
- The **Opacity** defines how opaque the Label is in percentages (0% is transparent, 100% is opaque).
- The **Text Position** defines where the text is anchored in the Label. The position of the selected Radio button correlates with this.
- The **Image** defines an Image that would be the background of the Label. This image is on top of the Background Color, and is underneath the text. Images must have been added already for it to show in this drop-down menu. (See **Images** for more information)

## VIDEOS

Clicking on the “Add Video” button in the Pages Taskbar creates a new video element. The Properties dialog box of a Video has four (4) tabs: “General”, “Live View Settings”, “HTTP Settings”, and “KCAM Settings”.

## General

*Script ID: (unset)*  
*Position: semi-random set of two numbers: 1st is X-coordinate, 2nd is Y-coordinate, in pixels*  
*Load Delay: 0*  
*Source Type: Secure Kubincam*

*Poster Image: (unset)*  
*Z-index: 0*  
*Size: (160px, 120px)*  
*Opacity: 100%*

The “General” tab contains options for configuring the design of the Video.

- The **Script ID** text box defines a string of text that can be used to refer to the Video in scripts. (See **Scripting**)
- The **Position** defines where the Video is located, in pixels, in relation to the page origin, which is the top left corner.
- The **Load Delay** defines how much delay there is in seconds when loading the Video.
- The **Source Type** defines the source type of the video, which can be “Built-in NVR Live View”, “Site-to-site Live View”, “MP4 over HTTP”, “MJPEG over HTTP” or “Secure Kubincam”. This value changes which tabs are accessible and what options in those tabs are accessible.
- The **Poster Image** defines an Image that would be the background of the Button when the Source is not available. Images must have been added already for it to show in this drop-down menu. (See **Images** for more information)
- The **Z-index** defines which layer the Video is on (see **Layers**).
- The **Size** defines the width and length of the Video in pixels.
- The **Opacity** defines how opaque the Video is in percentages (0% is transparent, 100% is opaque).

## Live View Settings

*Server: Local or unset*  
*Camera: (unset)*  
*Unzoomed FPS: 1*  
*Zoomed FPS: 15*  
*Unzoomed Quality: 40*  
*Zoomed Quality: 70*  
*On Click Script: (unset)*  
*On Zoom Script: (unset)*  
*On Unzoom Script: (unset)*

The “Live View Settings” tab becomes accessible when the Source Type in the “General” tab has a value of “Built-in NVR Live View” or “Site-to-site Live View”. The options in this tab are settings for those Source Types.

- The **Server** drop-down box (accessible with “Site-to-site Live View” Source Type) shows the connected servers. See **Servers** for information about adding and configuring Servers.
- The **Camera** drop-down box shows the camera of the current Installation (with “Built-in NVR Live View” Source Type) or the cameras of the selected Server (with “Site-to-site Live View” Source Type). The output from the selected Camera is shown as the content of the Video.
- The **Unzoomed FPS** defines how many Frames Per Second (FPS) the Video is displayed with when it is unzoomed. Minimum: 1, maximum: 30.
- The **Zoomed FPS** defines how many Frames Per Second (FPS) the Video is displayed with when it is zoomed. Minimum: 1, maximum: 30.

- The **Unzoomed Quality** defines the quality of the Video when it is unzoomed. Minimum: 1, maximum: 97.
- The **Zoomed Quality** defines the quality of the Video when it is zoomed. Minimum: 1, maximum: 97.
- The **On Click Script** text box can contain a Client script (See **Scripting**) that runs when the Video is clicked.
- The **On Zoom Script** text box can contain a Client script (See **Scripting**) that runs when the Video is zoomed.
- The **On Unzoom Script** text box can contain a Client script (See **Scripting**) that runs when the Video is unzoomed.

## HTTP Settings

*URL: (unset)*  
*HTTP Authentication: None*  
*Username: (unset)*  
*Password: (unset)*

The “HTTP Settings” tab becomes accessible when the Source Type in the “General” tab has a value of “MP4 over HTTP” or “MJPEG over HTTP”. The options in this tab are settings for those Source Types.

- The **URL** defines the location of the source video, as would be accessed over the Internet.
- The **HTTP Authentication** defines what kind of authentication the Installation should use when attempting to access the source video, which can be “None”, “Basic”, or “Digest”.
- The **Username** text box sets what username the Installation should use when attempting to access the source video. This is not used when HTTP Authentication is set to “None”.
- The **Password** text box sets what password the Installation should use when attempting to access the source video. This is not used when HTTP Authentication is set to “None”.

## KCAM Settings

*Input: 0*  
*Quality: 70%*  
*FPS: 1*

The “KCAM Settings” tab becomes accessible when the Source Type in the “General” tab has a value of “Secure Kubincam”. The options in this tab are settings for that Source Type.

- The **Input** text box defines which Input channel the Installation server should use when attempting to connect to a Secure Kubincam camera.
- The **Quality** text box defines the quality the Video should be displayed with.
- The **FPS** text box defines the Frames Per Second (FPS) the Video should be displayed with.

## SLIDERS

Clicking on the “Add Slider” button in the Pages Taskbar creates a new slider element. The Properties dialog box of a Slider has four (4) tabs: “Slider”, “Thumb”, “Rail”, and “Overrides”.

## Slider

*Script ID: (unset)*

*Position: semi-random set of two numbers: 1st is X-coordinate, 2nd is Y-coordinate, in pixels*

*Delayed Load checkbox: unchecked*

*Set Value on Slide checkbox: checked*

*Z-index: 0*

*Size: (48px, 192px)*

*Opacity: 100%*

*Vertical checkbox: checked*

*Number Property: (unset)*

The “Slider” tab contains options for configuring the general design of the Slider.

- The **Script ID** input box defines a string of text that can be used to refer to the Slider in scripts. (See **Scripting**)
- The **Position** defines where the Slider is located, in pixels, in relation to the page origin, which is the top left corner.
- The **Delayed Load** checkbox defines whether or not the value of the Slider should be retrieved immediately or delayed. When checked, its value is retrieved immediately; when unchecked, the retrieval is delayed.
- The **Set Value on Slide** checkbox defines whether or not the value of the Slider is set while it is being slid. When checked, its value is updated while the Slider is sliding; when unchecked, its value is updated after the slider is finished sliding.
- The **Z-index** defines which layer the Slider is on (see **Layers**).
- The **Size** defines the width and length of the Slider in pixels.
- The **Opacity** defines how opaque the Slider is in percentages (0% is transparent, 100% is opaque).
- The **Vertical** checkbox defines the orientation of the Slider. When checked, the slider is vertical; when unchecked, the slider is horizontal. Its size and position remains the same when this is changed, thus a user may need to change its size and position after changing the checkbox's value.
- The **Number Property** defines what is being controlled by the Slider. A Number Property can come from a Component. To add it to the Slider, drag and drop the Number Property from the Component drop-down onto the Slider.

## Thumb

*Size: (40px, 20px)*

*Corner Radius: 3px*

*Color: #4682B4 (a shade of blue)*

*Text Color: #FFFF00 (yellow)*

*Font Size: 10px*

*Image: (unset)*

The “Thumb” tab contains options for configuring the design of the Thumb on the Slider.

- The **Size** defines the width and length of the Thumb in pixels.
- The **Corner Radius** defines how round the corners of the Thumb are, in pixels.
- The **Color** defines the background color of the Thumb in the hexadecimal format.
- The **Text Color** defines the color of the text that is displayed on the Thumb.
- The **Font Size** defines the size of the text on the Thumb in pixels.



- The **Image** defines the image displayed on top of the Thumb.

## Rail

*Thickness: 10px*  
*Corner Radius: 0px*  
*Color: #808080 (a shade of gray)*

The “Rail” tab contains options for configuring the design of the Slider's Rail.

- The **Thickness** defines how thick the Rail is in pixels.
- The **Corner Radius** defines how round the corners of the Rail are, in pixels.
- The **Color** defines the background color of the Rail.

## Overrides

*Override Device Settings checkbox: unchecked*  
*Range: 0.0 to 100.0*  
*Step: 1.0*

The “Overrides” tab contains options for changing the default range and step of the Slider.

- The **Override Device Settings** checkbox defines whether or not the following options should change the default values of the Slider's range and step. When checked, the following options are available to be changed; when unchecked, the following options are not available to be changed.
- The **Range** input boxes define the minimum and maximum values of the slider. These numbers can be a decimal number.
- The **Step** input box defines how many units are in between each jump or step in the Slider. For example, if the Step value is 5, the value may go from 15 to 20 when the Slider is being slid.

## SELECT BOXES (BETA)

*Note: This is a beta feature, and the API is not finalized and could easily change at any time.*

Clicking on the “Add Select” button in the Pages Taskbar creates a new select drop-down box element. The Properties dialog box doesn't have any tabs, but only has similar options like that of a Button's “Initial” tab.

*Script ID: (unset)*  
*Position: semi-random set of two numbers: 1st is X-coordinate, 2nd is Y-coordinate, in pixels*  
*Font: SansSerif*  
*Font Size: 18px*  
*Text Color: #000000 (black)*  
*Background Color: (unset)*  
*Z-index: 0*  
*Size: (160px, 32px)*  
*Opacity: 100%*

- The **Script ID** input box defines a string of text that can be used to refer to the Select box in scripts. (See **Scripting**)
- The **Position** defines where the Select box is located, in pixels, in relation to the page origin, which is the top left corner.

- The **Font** defines the text font, which can be “Monospaced”, “Serif”, or “SansSerif”. Other font options are not available.
- The **Font Size** is the size of the text in pixels.
- The **Text Color** defines the text color.
- The **Background Color** defines the color of the Select box .
- The **Z-index** defines which layer the Select box is on (see **Layers**).
- The **Size** defines the width and length of the Select box in pixels.
- The **Opacity** defines how opaque the Select box is in percentages (0% is transparent, 100% is opaque).

## WIDGETS

Clicking on the “Add Widget” button in the Pages Taskbar adds a Widget element to the Page. Be sure to have added a Widget to the Installation prior to attempting to add a Widget to a Page, else an error message will appear in dialog box saying there are currently no Widgets loaded in the Installation with instructions on how to add one.

When the “Add Widget” button is pressed, a dialog box with the title “Choose a Widget” appears. It contains a drop-down box for selecting an available Widget along with a button labelled “Cancel” to cancel adding a Widget, and a button labelled “OK” to add to the selected Widget. Once the Widget is selected and the “OK” button is pressed, the Widget is added to the page. The Widget has its own Properties as well, with no tabs like a Label's Properties. (See the **Widget** section for more information)

*Widget: name of widget, immutable*

*Script ID: (unset)*

*Position: semi-random set of two numbers: 1st is X-coordinate, 2nd is Y-coordinate, in pixels*

*Z-index: 0*

*Size: the default size is defined by the Widget*

*Opacity: 100%*

*Page Settings: depends on Widget*

- The **Widget** label shows the name of the Widget. This cannot be changed.
- The **Script ID** input box defines a string of text that can be used to refer to the Widget in scripts. (See **Scripting**)
- The **Position** defines where the Widget is located, in pixels, in relation to the page origin, which is the top left corner.
- The **Z-index** defines which layer the Widget is on (see **Layers**).
- The **Size** defines the width and length of the Widget in pixels.
- The **Opacity** defines how opaque the Widget is in percentages (0% is transparent, 100% is opaque).
- The **Page Settings** are a group of settings that are defined by the Widget. The name of the setting is shown in the “Setting column” and its value can be added/modified by double clicking on the cell under the “Value” column in the same row. Hovering over the setting shows its description.

## XHTML

Clicking on the “Add XHTML” button in the Pages Taskbar creates a new select XHTML element. The Properties dialog box doesn't have any tabs, but only has similar options like that of a Button's “Initial” tab.

*Script ID: (unset)*

*Position: semi-random set of two numbers: 1st is X-coordinate, 2nd is Y-coordinate, in pixels*

*Overflow: Auto*

*Z-index: 0*

*Size: (100px, 100px)*

*Opacity: 100%*

- The **Script ID** input box defines a string of text that can be used to refer to the Widget in scripts. (See **Scripting**)
- The **Position** defines where the Widget is located, in pixels, in relation to the page origin, which is the top left corner.
- The **Overflow** drop-down box defines how content that is overflowing the XHTML item's boundaries are handled.
- The **Z-index** defines which layer the Widget is on (see **Layers**).
- The **Size** defines the width and length of the Widget in pixels.
- The **Opacity** defines how opaque the Widget is in percentages (0% is transparent, 100% is opaque).

## PAGE PROPERTIES

Pressing the Properties button shows a window with options for configuring the selected item. When no item is selected, the properties for the Page are shown. A Page's Properties has five (5) tabs: "Properties", "Script", "Enter Actions", "Leave Actions", and "Included Scripts".

## Properties

*Name: New Page*

*Script ID: (unset)*

*Profile: iPad iOS 4.3+*

*Orientation: Portrait*

*Size: (768px, 1024px)*

*Background Color: #FFFFFF (white)*

*Background Image: (unset)*

*Background Repeat: No*

*Exclude from Page History: unchecked*

The "Properties" tab contains options for designing the Page.

- The **Name** input box defines the Title of the Page that is seen in a browser tab and also in the Page's label under the Pages folder in the Tree.
- The **Script ID** input box defines a string of text that can be used to refer to the Page in Scripts. (See **Scripting**)
- The **Profile** drop-down box contains a number of devices and browsers that the page can be optimized for. These are:
  - Chrome 12+
  - iPad iOS 4.3+
  - iPhone & iPod 3G/4G iOS 4.2+
  - iPhone 5/5C/5S

- iPhone 6/6s/7/8
- iPhone 6/6s/7/8 Plus
- iPhone X
- iPhone Xs Max / XR
- Firefox 4.0+
- Safari 5.0+
- The **Orientation** drop-down box defines whether the page should be orientated as Portrait or Landscape.
- The **Size** input box defines the height and width of the Page in pixels.
- The **Background Color** input box defines the background color of the Page in a hexadecimal format.
- The **Background Image** defines an Image to be used as the background of the Page. This Image overlays the Background Color.
- The **Background Repeat** drop-down box defines whether or not and how the Background Image repeats across the Page. Options are:
  - Yes: The Background Image repeats both Horizontally and Vertically across the Page.
  - Horizontally: The Background Image only repeats Horizontally across the Page, and not Vertically.
  - Vertically: The Background Image only repeats Vertically across the Page, and not Horizontally.
  - No: The Background does not repeat across the Page at all and only appears once.
- The **Exclude from Page History** checkbox defines whether or not the Page should appear in the browser's history when browsed to. When checked, it does not appear in the browser's history; when unchecked, it appears in the browser's history.

## Script

The “Script” tab contains a text box for a client-side script written in JavaScript that will run when the Page loads. See the section on **Scripting** for more information.

## Enter Actions

Items specified here are run as soon as a user enters the Page. Items that can be added are:

- Delays (add with button labelled “Add Delay”)
- Client Scripts (add with button labelled “Add Client Script”)
- Timer Actions (add with button labelled “Add Timer Action”)
- Matrix Actions (add with button labelled “Add Matrix Action”)
- Sounds (drag and drop from Tree)
- Component Commands (drag and drop from Tree)
- Commander Commands (drag and drop from Tree)
- Pages (drag and drop from Tree)

## Leave Actions

Items specified here are run as soon as a user leaves the Page. Items that can be added are:

- Delays (add with button labelled “Add Delay”)
- Client Scripts (add with button labelled “Add Client Script”)
- Timer Actions (add with button labelled “Add Timer Action”)
- Matrix Actions (add with button labelled “Add Matrix Action”)
- Sounds (drag and drop from Tree)
- Component Commands (drag and drop from Tree)
- Commander Commands (drag and drop from Tree)
- Pages (drag and drop from Tree)

### Included Scripts

Any Scripts that are created and listed under the Scripts folder in the Tree will be shown here. Highlighted Scripts are included. Click on one to include it in the Page; it will be loaded and ran when the Page loads. To select multiple, press and hold the Control key on the keyboard and click on the ones you want to include. To select multiple that next to each other in the list, press and hold the Shift key on the keyboard and click on the first Script in the list that you want to include and then click on the last Script you want to include; all Scripts in between, along with the ones that were clicked on, will be included. To exclude a previously included Script, press and hold the Control key on the keyboard and click on the highlighted Script that you wish to exclude.

### LAYERS

Page elements can be layered upon each other. By default, every element is on the base layer (numbered 0); they overlap each other in the order that they were added. Alternatively, each element can be placed at a specific level in the layering. This is controlled by the elements' “Z-index” property. Things to keep in mind:

- A Z-index of -1 puts the element beneath the surface, causing it to be untouchable from the user.
- A widget is layered above everything with a smaller Z-index than itself, and below everything with a larger Z-index than itself.

There is a “Layers” button on the Pages Taskbar that when clicked shows how many layers are present on the Page. Layers are added by changing the Z-index value of a widget to a number other than 0.

You can learn more about this at the MDN Docs on [z-index](#).

### ELEMENTS

Once an element is added to a page, it can be moved around freely (depending on the Grid settings in Preferences) and resized as desired. Double-clicking on an element, or single-clicking on it followed by clicking the “Properties” button in the Pages Taskbar reveals a Properties dialog box for configuring the element.

### TIMERS

Timers can be added to certain actions, like a Page's Enter and Leave Actions or a Button's Press, Hold, and Release actions or in a Macro. Timers are created by clicking on “Add Timer Action” in the Action tabs of the Page or Button's Properties or in a Macro. Timers generally have these options:

*Timer Name: (unset)*  
*Do What: Start*

Macro: (unset)

Duration / Amount: 0 SECONDS

- The **Timer Name** input box is for giving the Timer a name to distinguish it from other timers in the logs. Other Timers that should affect each other must have the same name.
- The **Do What** drop-down box has three (3) options for what the Timer can do: “Start”, “Cancel”, “Increment”. The options below may or may not be available depending on this value. When it is set to “Start”, the timer is started, running for the defined length of time specified. When the timer is finished, it calls the specified Macro, running whatever commands the Macro has. If this value is set to “Cancel”, the Timer with the same name is canceled. If this value is set to “Increment”, the Timer's duration is lengthened by the amount specified.
- The **Macro** drop-down box defines which Macro should be run when the Timer is run after the Duration/Amount value. This is only available when the Do What option is set to “Start”.
- The **Duration / Amount** input box and drop-down box defines the length of time until a Macro is run, if the Do What option is set to “Start”. If it is set to “Increment”, the Timer with the same name is adjusted according to this value. It can be a positive number to increase the length of time or a negative number to decrease the length of time. Options for types of time lengths are “MILLISECONDS”, “SECONDS”, “MINUTES”, “HOURS”, and “DAYS”. This option is not available when the Do What option is set to “Cancel”.

## 26. Motion Detection

### OVERVIEW: HOW MOTION DETECTION WORKS

At regular intervals (set by the **Detection FPS** in Remote Builder), the motion detector compares the current frame to a previous frame (called the **reference frame**) it has seen some time ago. The comparison is done by calculating the RMSE (Root Mean Square Error) between the pixel intensities of the two frames. The RMSE is then converted into a percentage value between 0% and 100%.

The RMSE is calculated independently in 64 equally sized regions of the frame.

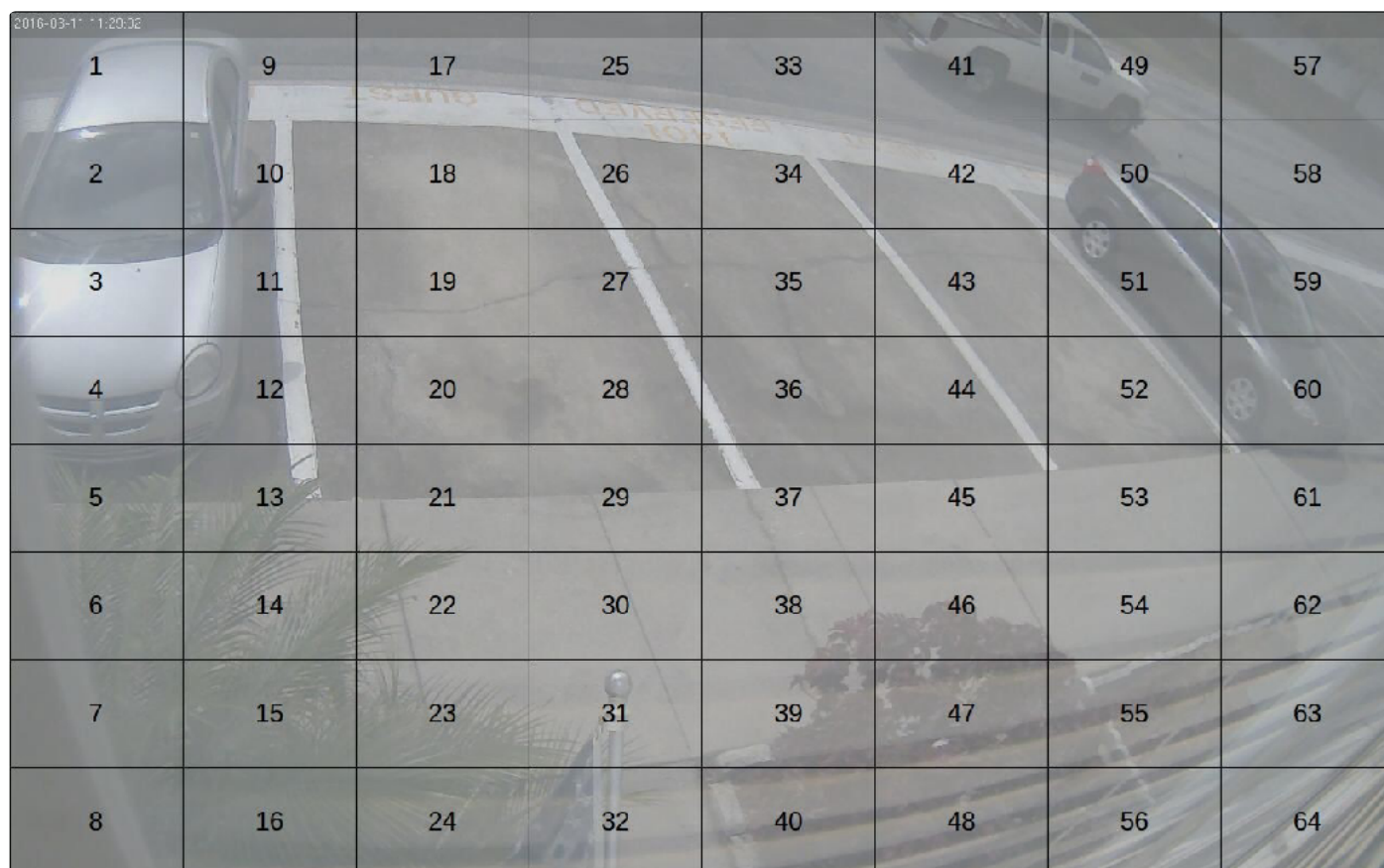


Figure 26.1 | The motion detection regions

The highest RMSE from all 64 regions is taken and compared to the **Motion Threshold (RMSE %)** setting. If the RMSE is equal to or greater than the motion threshold, the NVR is informed that motion has started. If the camera's **Default Mode** is set to **Armed**, then recording will begin immediately. The recording will include approximately two seconds of video leading up to the time that motion was first detected.

Once motion is detected, the NVR will not be informed that motion has stopped until the RMSE remains below the motion threshold for a period of time at least equal to the **Stop Delay**. The stop delay begins the moment the RMSE falls below the motion threshold. During the stop delay, if the RMSE rises above the motion threshold the stop delay is reset (starts over).

Once motion has been below the motion threshold for at least the stop delay, the NVR is notified motion has stopped and the **finish countdown** begins. During the finish countdown, new frames continue to be written to the current recording, and if the detector indicates motion has started again, the finish countdown is reset. The finish countdown is currently internally set to two seconds.

Thus, the combination of the stop delay and finish countdown determine how long after motion has truly stopped before the recording is ended. Using the default stop delay of two seconds, a recording will include approximately two seconds of video leading up to the motion triggering event, and four seconds of video following the last detection of motion – two seconds for the stop delay and two seconds for the finish countdown.

## HOW TO PROPERLY SETUP MOTION DETECTION

The most important setting for motion detection is the **Motion Threshold (RMSE %)**. If set properly (as described below) you should have very few spurious recordings and also very few recordings that are cut short when motion slows down by doesn't stop. *A good installation will NOT rely on the default value.* To do so could cause important motion events to be missed and/or excessive wasted space.

Motion detection should be configured from the “Motion” table of the Camera editor in Remote Builder. The rough outline of steps to take is:

1. Configure the camera's network settings, resolution and quality first. The resolution and quality can affect the camera's average RMSE.
2. Activate the “Preview”.
3. Create Exclusion Windows (the red button with the white dash) over any objects that are frequently moving but should not trigger motion detection. Common examples are: TV screens, bushes and distant high-traffic roads. Do NOT be concerned about missing import motion just because you've excluded a TV screen or bush – remember that motion is checked independently in reach of the 64 regions!
4. AFTER the exclusion windows have been created, you can then determine the camera's average RMSE by observing the “**Current RMSE %**” field over a few seconds. Most cameras have an innate noise floor of around 2-5% RMSE. It is sometimes higher in low light conditions, but should not be drastically so unless the camera is being incorrectly utilized e.g. using a high-lux camera at nighttime with no or insufficient IR illumination.

Ensure that no major motion occurs in the scene during your observation! Otherwise, simply ignore the peak RMSE value and try to get an appreciation for the average value when motion is not occurring.

5. Set the **Motion Threshold (RMSE %)** value to around 2% higher than the average RMSE you observed in the previous step. For instance, if you observe an average RMSE of around 2.6%, you might try setting the motion threshold to 4.6% or even 5%. There is no exact rule here, but the general idea is to set the motion threshold lower (i.e. close to the average RMSE) if you want very high sensitivity to motion and set the threshold higher to conserve space by preventing spurious recordings.
6. Upload the XML (or use the “Start NVR” feature in Remote Builder if the server is not running yet) and verify that the camera is not continuously tripping motion detection from its built-in noise. If you performed the previous steps correctly you should only have to check this occasionally after you have set up the entire Installation. Observe recordings at both daytime and nighttime! A good Installation is not blindly left alone for months or years on end without a periodic review of its recording tendencies

## THE STOP DELAY

Some customers may want certain cameras to be stubbornly recorded for a longer period after motion has slowed/stopped in order to create more continuous recordings of events. Increase the **Stop Delay** in



this case for the specific camera is question. If you have properly configured the motion threshold, the default stop delay of two seconds is sufficient for most purposes. It will cause about 33 minutes of extra footage to be recorded per 1000 recordings.

The stop delay is currently limited by the system to no more than 15 seconds.

#### **THE REFRESH INTERVAL**

The refresh interval is the length of time the same old reference frame is used for RMSE calculations before being replaced with the current frame. In most Installations, this value can be safely left at the default of one second.

Regardless of the refresh interval setting, if the current RMSE rises above the motion threshold then the reference frame is immediately replaced with the current frame. This is to prevent the motion detector from continuously firing after an object has relocated in a scene. For instance, if a car pulls into a parking lot and then remains motionless for 45 seconds before backing back out, you don't generally want to record most of the time that the car is parked if nothing else is going on in the scene. To do so would waste space with identical frames and reviewing footage that much more tedious.

#### **Reasons to increase this value:**

- To increase the sensitivity to slow or small changes to the scene over time (like recording a time lapse of the sun rising).

#### **THE DETECTION FPS**

The detection FPS determines how often motion detection occurs. The default value of 4FPS will cause the detector to check for motion approximately every 250 milliseconds (number of milliseconds =  $1000.0 / \text{detection FPS}$ ). This is sufficient for most purposes and is a good balance between fast moving object detection and CPU usage. It will cause the detector to (sometimes) ignore events that occur and return the scene back to its original state in under 250ms.

#### **Reasons to increase this value:**

- You are trying to capture very quick events. It is rare when the default 4FPS is not sufficient to detect motion in a situation where useful footage could have actually been recorded.

#### **Reason to decrease this value:**

- You are recording lots of short, mostly useless little clips. The presence of a blinking light or quickly wavering plant in the scene is a good reason to lower this value.
- CPU usage from motion detection is causing other issues with the system. This is not likely, however, on modern server machines – 1280x720 detection @ 4FPS should use 1-2% of a single core per camera on a 6th generation Intel i5/i7.

If the detection FPS value causes detection to occur less often than the current setting of the refresh interval to effectively be once every 20 seconds.

Regardless of the Detection FPS setting, motion will never be detected more often than once per new frame from the camera.

## 27. IR Command

*Name: New Command*

*Format: Pronto*

*Code: (unset)*

- The **Name** input box defines the name of the IR Command.
- The **Format** drop-down box defines whether the format of this IR Command should be in Pronto or Tira-2.
- The **Code** text box defines the code that should be sent when communicating with the device over this IR connection. Can be a Client Script.

An IR Command also has buttons for testing the code, capturing an IR command, and generating NEC code.

## 28. Serial Command

*Name: New Command*

*Is HEX?: No*

*Want Response?: No*

*Response Timeout: 500 milliseconds*

*Command: (unset)*

- The **Name** input box defines the name of the Serial Command.
- The **Is HEX?** drop-down box defines whether the communication between the serial device and this command is in a HEX format or not. If “Yes” is selected, a button to “Insert ASCII” is shown (with keyboard shortcut of **Alt+A**).
- The **Want Response?** drop-down box defines whether the Driver should expect a response from the device or not.
- The **Response Timeout** input box defines how long (in milliseconds) the Driver should wait for a response from the device.
- The **Command** text box defines the command that should be sent when communicating with the device over this serial connection. Can be a Client Script.

## 29. IP Command

*Name: New Command*

*Format: HTTP GET*

*Want Response?: No*

*Response Timeout: 500 milliseconds*

*Expected Response Pattern: (unset)*

*Command: (unset)*

- The **Name** input box defines the name of the IP Command.
- The **Format** drop-down box defines whether the format of the IP Command should be “HTTP Get”, “Raw HEX”, or “Raw ASCII”. If “Raw HEX” is selected, a button to “Insert ASCII” is shown (with keyboard shortcut of Alt+A).
- The **Want Response?** drop-down box defines whether the Driver should expect a response from the device or not. If “Yes” is selected, the **Response Timeout** input box is shown.
- The **Response Timeout** input box is only shown when **Want Response?** is set to “Yes”. It defines how long (in milliseconds) the Driver should wait for a response from the device.
- The **Expected Response Pattern** is only shown when **Format** is set to “Raw ASCII” **Want Response?** is set to “Yes”. It should contain a RegEx pattern (using the [Java syntax](#)) that the expected response should match.
- The **Command** text box defines the command that should be sent when communicating with the device over this IP connection. Can be a Client Script.

## 30. Relay Command

*Name: New Command*

*Relay: 1 - port 0, pin 0*

*Action: On*

*Momentary Time: 500 milliseconds*

- The **Name** input box defines the display name of the Relay Command that is used to identify it throughout the Installation.
- The **Relay** drop-down box sets the relay this Command uses. How many options that are available depends on **# Relays** in **Setup**.
- The **Action** drop-down box defines the action associated with this Command. Options are “On”, “Off”, “Toggle”, “Momentary On/Off”, “Momentary Off/On”, and “Momentary Toggle”.
- The **Momentary Time** input box defines how long the momentary action takes in milliseconds.

## 31. Number Property

Number Properties can only be used in Sliders.

### PROPERTIES

*Name: New Property*

*Capability: Two Way (Get & Set)*

*Get Command: (unset)*

*Set Command: (unset)*

*Range: 0.0*

*to: 100.0*

*Prefix: (unset)*

*Suffix: (unset)*

- The **Name** input box defines the name of the Number Property.
- The **Capability** drop-down box defines whether the Number Property can get values (“One Way (Get Only)”), set values (“One Way (Set Only)”), or both (“Two Way (Get & Set)”).
- The **Get Command** drop-down box defines which Driver command will be used to get values. This can be either a Serial or an IP Command that expects a response. This option is only shown when **Capability** is set to “One Way (Get Only)” or “Two Way (Get & Set)”.
- The **Set Command** drop-down box defines which Driver command will be used to set values. This can be either a Serial or an IP command that does not expect a response and has a function with at least one argument in the Command text box. This option is only shown when **Capability** is set to “One Way (Set Only)” or “Two Way (Get & Set)”.
- The **Range** and **to** input boxes defines the maximum and minimum values of the Number Property.
- The **Prefix** input box defines what should be prepended to the Property display. This is reflected on the Slider’s Thumb.
- The **Suffix** input box defines what should be appended to Property display. This is reflected on the Slider’s Thumb.

### CONVERSION FUNCTIONS

*Device to Value: (unset)*

*Value to Device: (unset)*

- The **Device to Value** text box contains the function to convert the response from the device to a usable value. This is only available if **Capability** is set to “One Way (Get Only)” or “Two Way (Get & Set)”.
- The **Value to Device** text box contains the function to convert the value to a value that the device can understand. As other values are set, this script is automatically set. This is only available if **Capability** is set to “One Way (Set Only)” or “Two Way (Get & Set)”.

## 32. Enum Property

Enum Properties can be used in action environments, such as Button's Release or a Page's Leave Actions to set values easily with some more advanced yet simple control than with other means.

### PROPERTIES

*Name: New Property*

*Capability: Two Way (Get & Set)*

*Get Command: (unset)*

*Set Command: (unset)*

- The **Name** input box defines the name of the Enum Property.
- The **Capability** drop-down box defines whether the Enum Property can get values ("One Way (Get Only)", set values ("One Way (Set Only)", or both ("Two Way (Get & Set)").
- The **Get Command** drop-down box defines which Driver command will be used to get values. This can be either a Serial or an IP Command that expects a response. This option is only shown when **Capability** is set to "One Way (Get Only)" or "Two Way (Get & Set)".
- The **Set Command** drop-down box defines which Driver command will be used to set values. This can be either a Serial or an IP command that does not expect a response and has a function with at least one argument in the Command text box. This option is only shown when **Capability** is set to "One Way (Set Only)" or "Two Way (Get & Set)".

### ADD ITEM

*Device Value: (unset)*

*Display Label: (unset)*

- The **Device Value** input box defines the value that should be sent to the device when this item is selected. It should be the same as what would be returned by the response parser function.
- The **Display Label** input box defines the user-friendly label that should be shown instead of the value.

### CONVERSION FUNCTIONS

*Response Parser: (unset)*

- The **Response Parser** text box contains a function for parsing the response returned from the get Command. As other values are set, this script is automatically set. This is only available if **Capability** is set to "One Way (Get Only)" or "Two Way (Get & Set)".