

# VisTool MK2

## User Guide

Version: 6.6.0/5

Edition: Friday, November 12, 2021



To obtain the latest documentation and software downloads, please visit:

[www.lawo.com/lawo-downloads](http://www.lawo.com/lawo-downloads)

### Copyright

All rights reserved. Permission to reprint or electronically reproduce any document or graphic in whole or in part for any reason is expressly prohibited, unless prior written consent is obtained from the Lawo AG.

All trademarks and registered trademarks belong to their respective owners. It cannot be guaranteed that all product names, products, trademarks, requisitions, regulations, guidelines, specifications and norms are free from trade mark rights of third parties.

All entries in this document have been thoroughly checked; however no guarantee for correctness can be given. Lawo AG cannot be held responsible for any misleading or incorrect information provided throughout this manual.

Lawo AG reserves the right to change specifications at any time without notice.

© Lawo AG, 2021

## Table of Contents

---

1. Introduction .....	4
2. Product Overview .....	5
3. Installation .....	7
4. Configuring a PC for VisTool Runtime .....	13
5. VisTool Runtime Operation .....	37
6. Editing a VisTool Configuration .....	80
7. VisTool Solo & Solo Plus .....	136
8. VisTool Elements .....	142
9. ON-AIR Designer Functions .....	219
10. Using Ember+ .....	238
11. Configuring an External Matrix .....	255
12. The Logic Engine .....	260
13. The Visual Extension .....	298
14. Troubleshooting .....	311
15. Appendices .....	317

# 1. Introduction

---

Welcome to **VisTool MK2**.

## About this Manual

This document describes how to install, configure and operate **VisTool MK2**.

It provides a general guide to the operation of the software, with screenshots taken from different partnering devices.

Look out for the following which indicate:

**Notes** - points of clarification.

**Tips** - useful tips and short cuts.

**Attention** - alert you when an action should *a/ways* be observed.

## Lawo User Registration

For access to the **Downloads** area and to receive regular product updates, please register at:

[www.lawo.com/registration](http://www.lawo.com/registration).

## 2. Product Overview

**VisTool MK2** is a software application that provides real-time audio visualization and control.

The software can be installed on a Windows PC connected to the Lawo network. During runtime, most of the GUI elements have no actual function of their own, but remotely control parameters within a networked device.

Elements such as faders, meters, timers and signal processing can be viewed and controlled from the VisTool runtime screens. Pages can also include Web elements (to displays web pages); video thumbnails (to preview video from a V\_\_line device); and VSM-Button elements (to control or monitor VSM Panel functions). The range of parameters is determined by the software release, license and the control protocol used by the remote device.

The end-user interface is usually a touch-screen, and a single VisTool PC can support up to 6 screens. Note that VisTool is also used to run the screens on Lawo's Visual Extension (described [later](#)).

*VisTool on External Monitor*



*VisTool Solo on External Tablet*



*sapphire with Visual Extension*

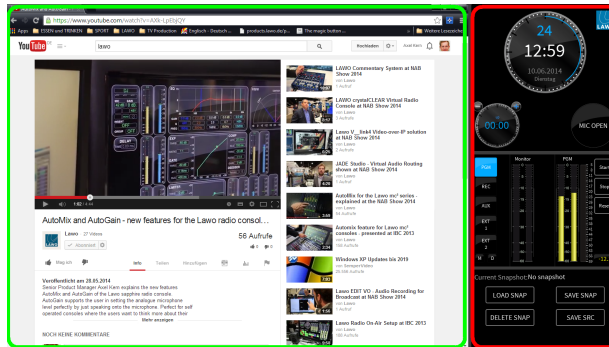


Each configuration can support multiple pages, which can open in full-screen, as a frame-less window, or as a docking bar. This makes it possible to share a single screen with other applications:

*VisTool as a Frame-less Window (in red)*



*VisTool as a Docking Bar (in red)*



## Licensing

Having installed VisTool MK2, you can launch the VisTool GUI by opening a start script. This requires the correct license to be installed on the runtime PC. There are four possible options.

- **VisTool Standard** - is included free of charge with ruby, crystal, sapphire and sapphire compact consoles. This license runs the standard template for your product with a fixed set of pages and elements.
- **VisTool Unlimited** – is a paid upgrade which will run any VisTool start script, including custom configurations designed by VisTool Editor. The resource numbers are "unlimited", and the functionality can be expanded by purchasing the following add-ons:
  - **Ember+/Logic AddOn** - is required to support Ember+ and the Logic Engine.
  - **VSM Button Panel AddOn** - is required to support VSM-Button elements.
- **VisTool Solo** - provides a cost-effective alternative to VisTool Unlimited. This license supports custom configurations but with restrictions on some resources. You cannot use add-on licenses with VisTool Solo.
- **VisTool Solo Plus** - provides a mid-way point, price-wise, between VisTool Solo and VisTool Unlimited. This license supports more resources than VisTool Solo but less than VisTool Unlimited. You cannot use add-on licenses with VisTool Solo Plus.

## VisTool Components

The VisTool MK2.exe file installs four applications which are used as follows:

- **VisToolMK2 Admin** - sets up the network environment.
- **VisToolMK2 Sessions** - configures and generates the start script for runtime operation.
- **VisToolMK2 Editor** - designs and builds custom configurations.
- **VisToolMK2 dbAlias** - exports databases into formats which support multi-user access.

## Custom Configurations

**VisTool Editor** can be used without a license to design and build custom configurations. Thus, you can install VisTool on an unlicensed PC for this purpose. The configuration can be previewed in VisTool Editor by activating an online mode which times out after five minutes. The automatic time-out means that VisTool Editor cannot be used for runtime operation, and so it is essential that you install a VisTool Solo, Solo Plus or Unlimited license if you wish to run a custom start script.

## Preparing the Runtime PC

To prepare a PC for runtime operation, there are three steps:

- Transfer the VisTool project to the runtime PC. This PC must be licensed.
- Use **VisTool Admin** to set up the devices on the network (for DMS control).
- Use **VisTool Sessions** to configure a "session" for each VisTool screen and generate a start script.

The start script can then be used to launch the runtime GUI whenever you double-click on its icon.

## The Control Protocols

VisTool supports a choice of communication protocols (for the remote control of the networked device): either DMS or Ember+.

The DMS (Digital Metering Service) is a Lawo control protocol supported by all radio on-air products. Ember+ is a non-proprietary TCP/IP protocol supported by a wider range of products including Lawo's A\_\_line, V\_\_line and mc<sup>2</sup> devices, plus applications such as R3LAY and Lawo Edit.

During runtime, VisTool can control up to four DMS devices simultaneously (one per session) and/or multiple devices via Ember+. For more details, see [Control Methods](#).

# 3. Installation

---

This chapter describes how to install the software and prepare the host PC.

Topics include:

- [PC Requirements](#)
- [Preparing the User Access in Windows](#)
- [Installing the Software](#)
- [Uninstall & Updates](#)

### 3.1 PC Requirements

Please double-click [here](#) to view the host PC requirements (as a pdf).



### 3.2 Preparing the User Access in Windows

If you wish VisTool to be available to any Windows User, then you should make sure that all files and applications are accessible to all users:

1. With the PC's main monitor connected, make sure that you are logged into Windows as an Administrator.
2. Prepare a "projects" folder that can be seen by all users of the PC.

Later you will use this folder to store all of the relevant project files required to run the VisTool configuration.

We recommend using the following file path:

```
C:\users\public\Lawo
```

In Windows, set up the Users so that they have access to this folder and its contents.

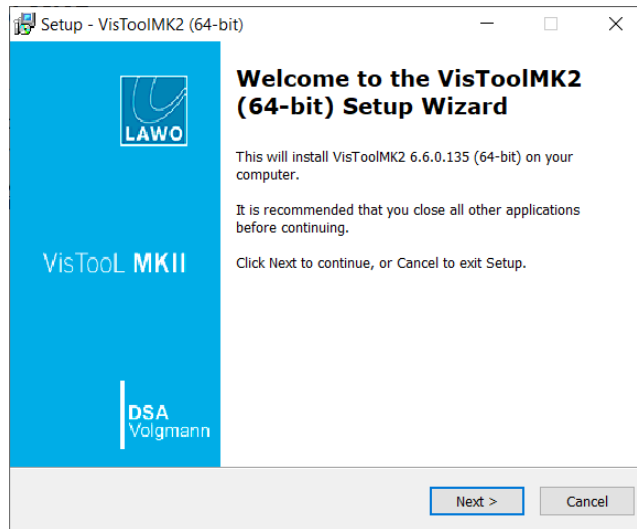
3. When you run the VisTool MK2 installer, make sure that you run the ".exe" file as an Administrator.

### 3.3 Installing the Software

The **VisToolMK2.exe** installer performs an automatic install of all VisTool components and Codemeter Runtime (the software license manager).

1. Copy the installer onto your PC.
2. Either, double click on the installer icon, or right-click and run the ".exe" file as an Administrator.

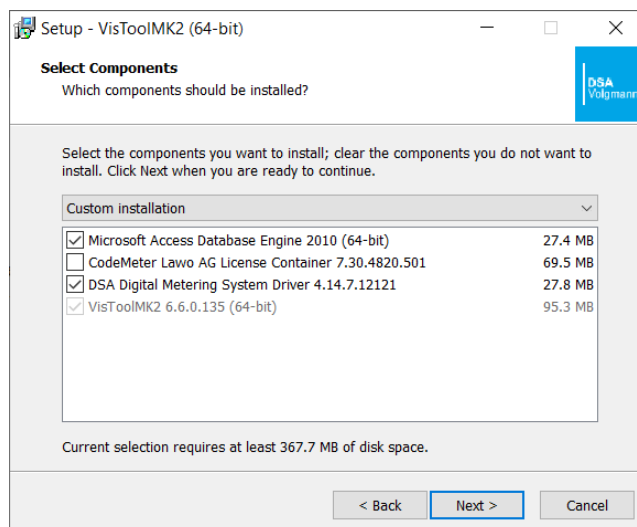
The latter is recommended if you wish VisTool to be accessible to all users of the PC. Both operations start the 'VisToolMK2 Setup Wizard'. Select the language to be used for the installation, either English or German, and the following window appears.



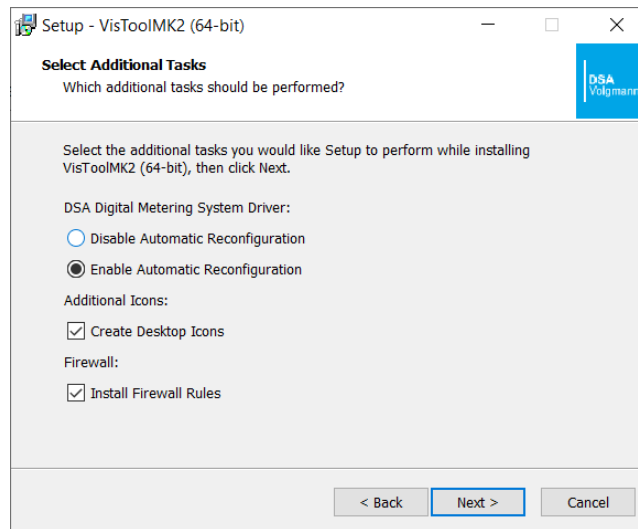
3. Follow the Wizard's instructions accepting the default options provided.

At the 'Select Components' window, all of the components should be ticked.

Note that the **CodeMeter** software is necessary for [licensing](#). If an existing License Container is already installed, then this option can be deselected (as shown below). Leave the option ticked to update the CodeMeter release and create a new container.



At the 'Additional Tasks' window, make sure that the **Create Desktop Icons** and **Install Firewall Rules** options are selected (ticked):

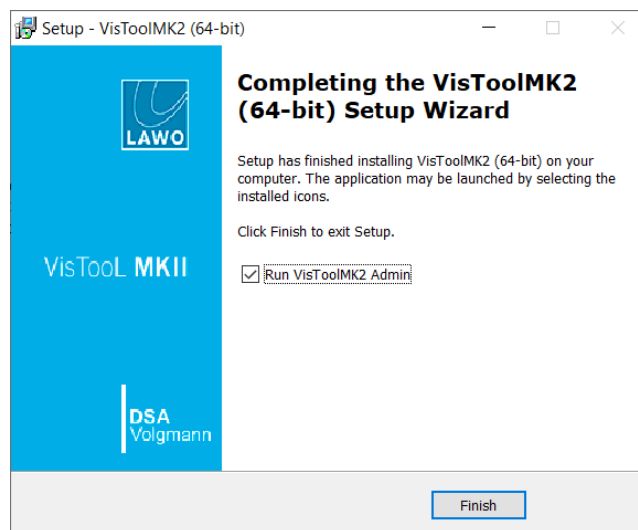


By default, the program will add its firewall rules to ensure smooth operation of VisTool during runtime.

4. When you reach the 'Summary' window, check the options and click **Install** - the software is installed onto your computer; this may take a few minutes.

By default, files are installed in the location: 'C:\Program Files\VisToolMK2'.

5. When the installation is complete, the following window will appear:



It is recommended to leave the default "Run" options selected and click on **Finish**. You can now use **VisToolMK2 Admin** to configure the application.

If you have any problems with the software installation, please contact your local Lawo representative or email [support@lawo.com](mailto:support@lawo.com).

## 3.4 Uninstall & Updates

VisTool MK2 can be uninstalled in the usual Windows manner: from the "Uninstall or change a program" Control Panel, select **VisTool MK2** and click **Uninstall**.

You can update VisTool MK2 to a new software version by following the normal installation procedure described above. There is no need to remove an older version of the application.

# 4. Configuring a PC for VisTool Runtime

---

This chapter describes how to configure a PC for VisTool runtime operation.

Topics include:

- [Introduction](#)
- [Connecting Your PC to the Lawo Control System](#)
- [Starting VisAdmin](#)
- [Configuring the DMS Connections](#)
- [Session Management](#)
- [Licensing](#)

## 4.1 Introduction

Two applications are used to configure a VisTool PC for runtime operation: **VisAdmin** and **VisTool Sessions**.

- **VisAdmin** defines the network for DMS-controlled devices.
- **VisTool Sessions** configures and generates the start script for the runtime GUI.

Both applications are installed automatically when you run the VisToolMK2 installer. They can be launched either by clicking on their desktop icons, or selecting **VisToolMK2 Admin** or **VisToolMK2 Sessions** from the Windows "Start" menu. **VisTool Sessions** can also be launched from the **VisAdmin** home page.

## 4.2 Connecting Your PC to the Lawo Control System

To support runtime operation, you must have a valid network connection between your VisTool PC and the remote device's control system. The connection can be made either directly or via a network switch.

### Network Requirements

If connecting via a network, then this can be shared with other devices, such as in the regular "house network" of a typical broadcast facility. Routers are permitted as long as the minimum requirements below are met. By using routers, or similar devices, the latency of the communication will increase. The network *MUST* meet or exceed the following requirements:

- At least 100MBit/s
- Full Duplex

### Cable Type

For a direct connection, you will need a crossover network cable. For connection via a network switch, use a straight (1:1) network cable. The cable should be STP-CAT 5 with RJ45 connectors.

When using a crossover network cable (direct connection), the "Use of a Proxy Server" browser option must be deactivated. Otherwise access is prevented by looking for a "Proxy".

### TCP/IP Settings

To establish communication, configure the TCP/IP settings on the host PC by opening the network settings for its Network Interface Card.

- The **IP Address** must be unique, and set within the same range as that of the remote device's control system.
- The **Subnet Mask** should be identical to that of the remote device's control system.

The default settings for Lawo radio devices are: **IP Address = 192.168.101.240**; **Subnet Mask = 255.255.255.0**.

If connecting via a larger network, you may need to consult your network administrator to gain access to the Lawo control system network from your device.

### Additional Notes for DMS Clients

Any number of VisTool clients can be connected to the network, as long as the total number of DMS services in use at any one time does not exceed 4. See [Configuring the DMS Connections](#).

### Additional Notes for Ember+ Clients

VisTool supports multiple Ember+ clients (as defined in the Ember Providers list for the project). See [Ember+](#).

### Checking the Network Communication

The simplest way to check that you have a valid network connection to the remote device is to open a Web UI connection: type the IP Address of the device's control system into your web browser's URL field - the device's Web UI Home page should appear. For more details on the Web UI, please refer to your product's User/Technical Manual.

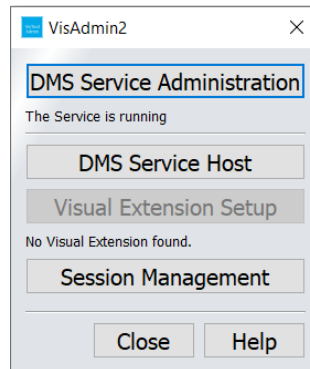
If you having problems establishing a network connection please see the [troubleshooting](#) tips.

### 4.3 Starting VisAdmin

1. **VisAdmin** starts automatically at the end of the software installation procedure.

Alternatively, it can be started by clicking on the **VisToolMK2 Admin** desktop icon, or selecting the program from the Windows "Start" menu. On startup, the **VisAdmin** home page appears:

*VisAdmin Home page*



The home page includes four options:

- **DMS Service Administration** and **DMS Service Host** - define the network for DMS-controlled devices. The setup is described over the next couple of pages.
- **Visual Extension Setup** - launches the setup window for the Visual Extension (described in the [Visual Extension](#) chapter). The option will be greyed out if no Visual Extension is connected.
- **Session Management** - opens the [Session Management](#) window (described later).

2. Select **Close** to close the home page and exit VisAdmin.

You can also select **Help** to display the "VisAdmin Help & Hints".



### 4.4 Configuring the DMS Connections

For all devices controlled via [DMS](#), you will need to define a DMS service.

This is required if you are connecting VisTool to a Lawo radio product for real-time control - for example, to a ruby, crystal, sapphire or sapphire compact. Alternatively, if VisTool is running stand-alone to control Ember+ clients, then the DMS service list should be left empty (to minimize the amount of network traffic).

Please note:

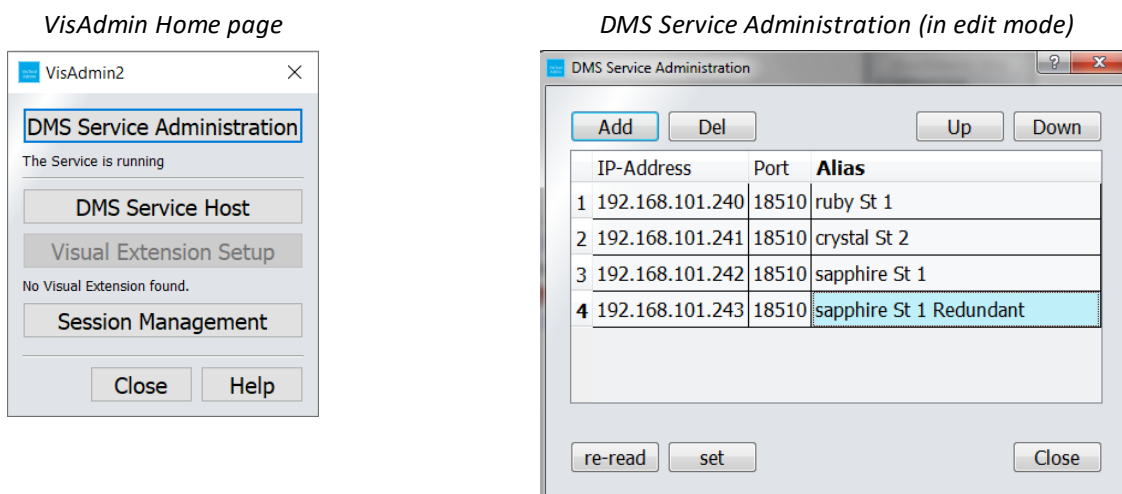
- You can define as many DMS services as you wish, as long as the total number in use at any one time does not exceed 4\*.
- A DMS service host will be required if your VisTool PC does not have a local DMS service.
- Every DMS service increases the amount of network traffic. Therefore, it is recommended to delete all unused DMS services.

\* At any given time, the DMS services in use are determined by the VisTool runtime(s). For each runtime start script, the DMS services are selected using [VisTool Sessions](#).

Note also that VisTool clients are "seen" by each remote device as a single connection as long as they share the same DMS service. So, for example, on a **Visual Extension** with more than 6 displays, you will need two VisTool PCs to drive all of the displays (as one PC can support a maximum of 6 VisTool screens). Therefore, to configure this example correctly, the same DMS service must be assigned to both PCs.

#### 4.4.1 DMS Service Administration

1. Start by selecting the **DMS Service Administration** option from the **VisAdmin** home page (described [earlier](#)):



This window defines all of the DMS services supported by the VisTool PC. If a device is not listed here, then it cannot be controlled via DMS by VisTool. Following a fresh installation, the list will be empty.

2. Select **Add** to add a new device and then double-click in each field to define the connection.

The **IP-Address** and **Port** fields must be completed; the **Alias** field is optional.

- **IP-Address** - this must match the IP address of the connecting device. For example, if the device is set to its default IP address, then enter **192.168.101.240**.
- **Port** - this should be set to **18510** (the default port number).
- **Alias** - this field is used in the [Session Management](#) window. Therefore, it can be useful to enter an alias name to help identify the device.

Under normal circumstances, you should use the default **Port** number (18510). Note that if you use a different **Port** number, then the firewall rules implemented by the [installer](#), will no longer apply.

- Repeat step 2 to define an entry for each of your networked devices.

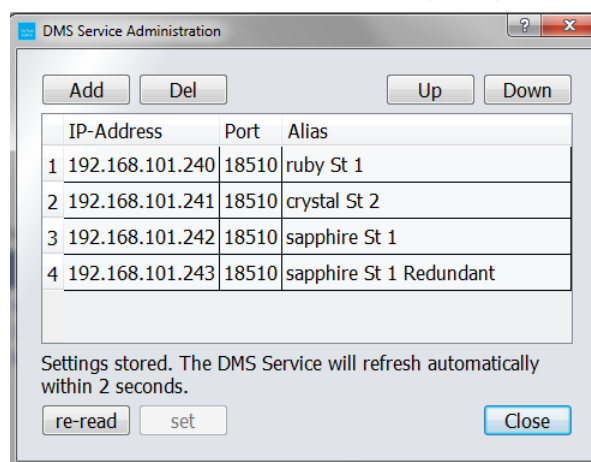
Note that you must add two lines if the device has a main and redundant control system (to define both IP addresses).

While editing, you can:

- Select an entry and then click on **Del** to delete it, or **Up/Down** to move its position in the list.
- Click on **re-read** to re-load the stored settings. Note that this will undo all edits since the last save, so take care when using this function as it will reset *all* of your changes.
- Use keyboard shortcuts to select all (CTRL+A) and then copy (CTRL+C), cut (CTRL+X) or paste (CTRL+V) the contents of a field.

- Once all of the fields are complete, select **set** to store the settings and refresh the DMS service - the list updates and the **set** button becomes greyed out:

*DMS Service Administration (setup complete)*



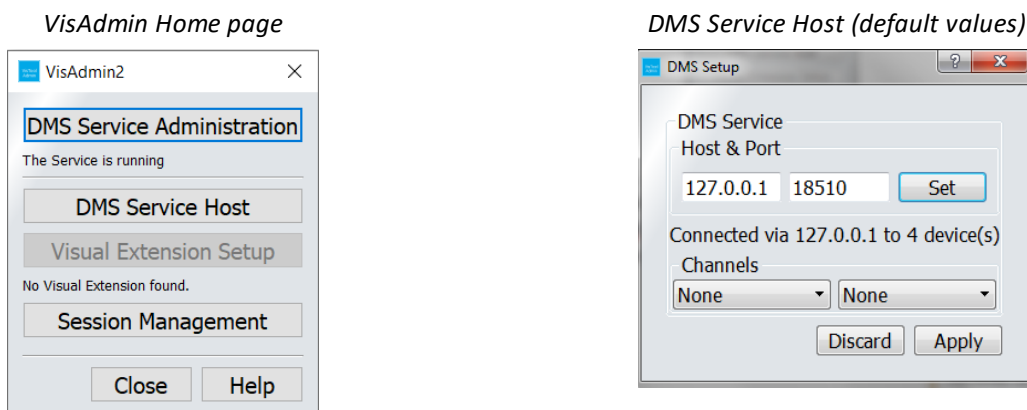
Note that you must use **set** to confirm your changes. If the **set** button is not greyed out, then the settings are still in edit mode and will not be saved if you **Close** the window.

- Once the settings have been stored, select **Close** to close the window and return to the VisAdmin home page.

## 4. Configuring a PC for VisTool Runtime

### 4.4.2 DMS Service Host

- Next, select the **DMS Service Host** option from the **VisAdmin** home page (described [earlier](#)) to define the DMS service host:



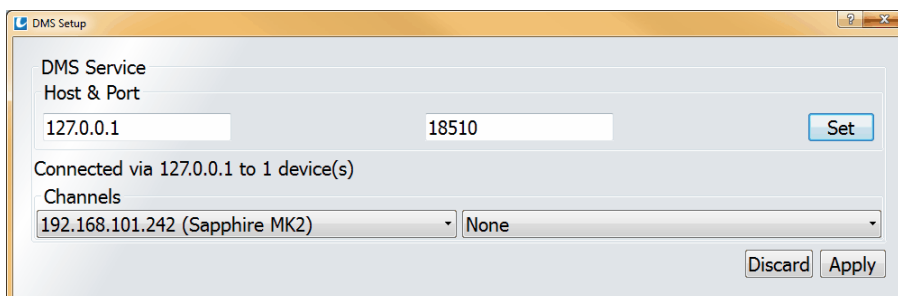
By default, the values for the local host PC will be entered in the upper fields. Under normal circumstances there is no need to change these and you can skip to step 3.

If your VisTool PC does not have a local DMS service, then you can use another computer to provide the service (as a DMS proxy). In this instance, edit the fields as follows:

- Using the upper fields, enter the IP Address and Port number of the computer where the DMS service is running, and select **Set** to activate the new settings.

Once communication with the host has been established, the text will update to show that you are "Connected via %DMS Host IP% to x device(s)".

- Now click on the left **Channels** field and select an available device from the drop-down list - you will see the IP Address of the Lawo control system and its detected hardware in brackets ():



The **Channels** list on the right is for redundant systems. If a redundant control system is fitted, select its IP Address from here. If no redundant control system is fitted, then leave this field set to **None**.

- Once you have finished editing, select **Apply** - the settings are saved and the 'DMS Service Host' window closes.

Alternatively, select **Discard** to discard any changes.

The setup of the DMS service for the network connection is now complete.

## 4.5 Session Management

**VisTool Sessions** is used to configure how VisTool will appear on your PC's monitor(s) and generate a start script

If the PC supports multiple screens, then the start script will open multiple "sessions" where each "session" represents a single screen. Thus, by varying the session options, you can customize the appearance and functionality of each VisTool screen.

When you generate the start script, a link is placed on the desktop of your computer's main monitor (as defined by Windows). The start script can then be used to [launch](#) the runtime GUI whenever you double-click on its icon.

Please note:

- Each start script can contain one or more sessions. This allows you to run several VisTool screens from the same PC, or create a choice of start scripts for different productions/users.
- Sessions can use the same "DMS settings" so that you can use more than one VisTool screen with the same Lawo console.
- Or, sessions can use different "DMS settings" if you wish to control different consoles from a single PC. In this instance, each device must be controlled from a separate VisTool screen.
- Sessions can control only Ember+ devices and no DMS unit providing you disable the DMS session parameters.
- By selecting different "Monitor" options for sessions, you can change how the GUI will appear on each VisTool screen. For example, whether it opens in full-screen, as a frame-less window, as a docking bar, etc.
- Multiple frame-less sessions can be output to the same physical screen.

### 4.5.1 Preparation

In order to configure the sessions and generate a start script, you will need a copy of the VisTool project you wish to use (**.vistool** file). This can be either the standard template delivered with your system or a custom configuration saved by VisTool Editor. A standard template is available for all Lawo radio products: ruby, crystal, sapphire and sapphire compact.

In addition, for DMS-controlled devices, you will need the accompanying ON-AIR Designer project (**.db3** file). Note that this file is not required if your VisTool project uses only Ember+ elements.

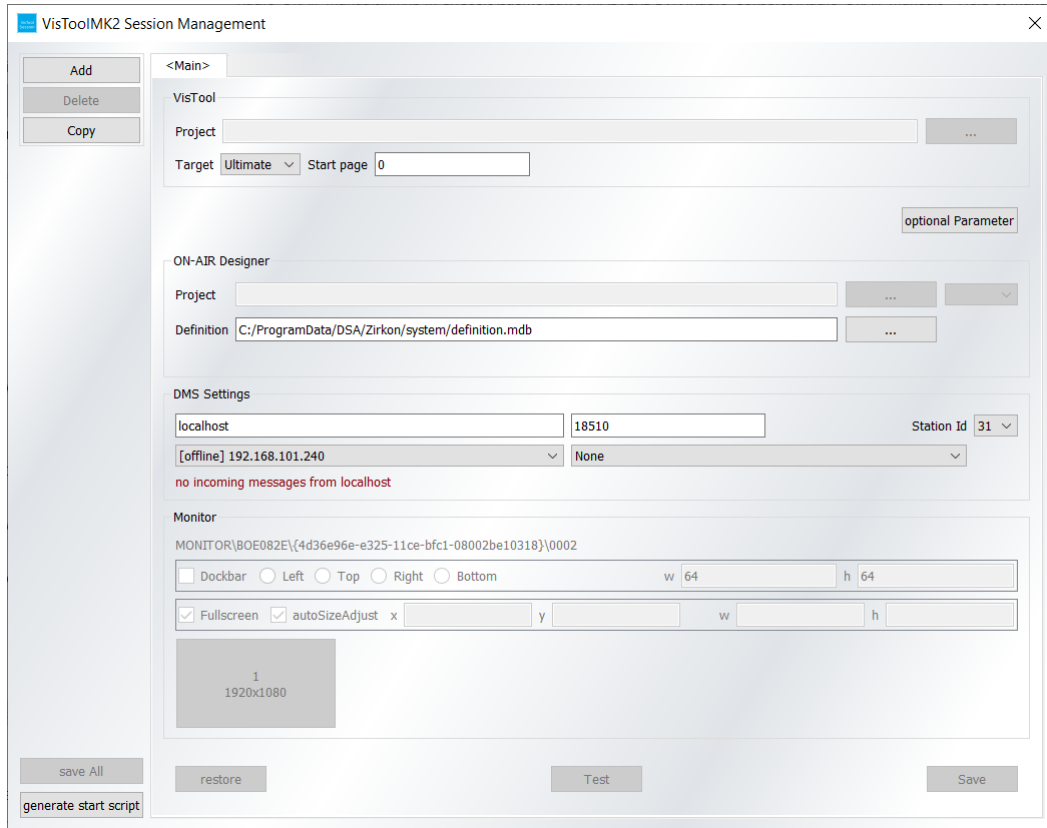
It is best to copy the files into the Windows folder you prepared earlier, so that the VisTool sessions will be accessible to all users. See [Preparing the User Access in Windows](#).

Once the files are in place, you are ready to configure the session(s).

### 4.5.2 Starting VisTool MK2 Sessions

1. Start the application either by clicking on the **VisToolMK2 Sessions** desktop icon, or selecting the program from the START menu (under **Program Files -> VisToolMK2**).

On startup, the "Session Management" window appears:



When you start the tool for the first time, the "Session Management" window contains a single session named **<Main>** (as shown above). Alternatively, the window will re-open in its last saved state.

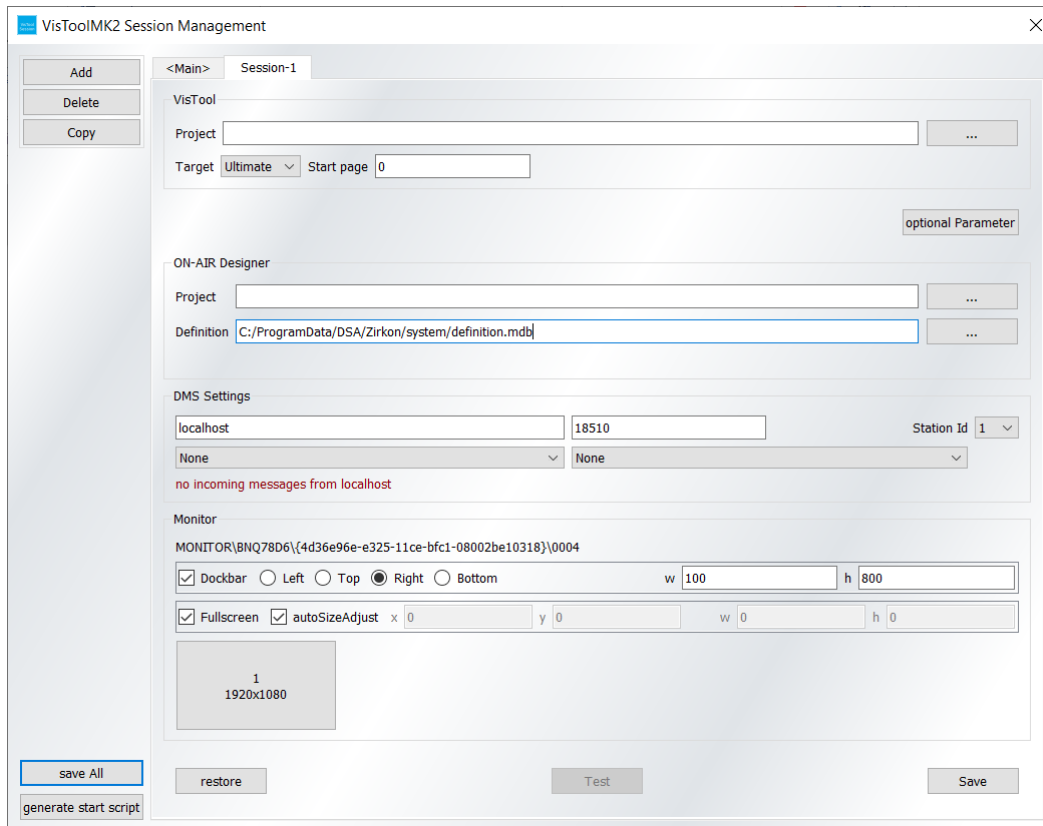
In either case, the session named **<Main>** defines the parameters used to display VisTool Editor. This session is always present and cannot be deleted, renamed or included in a start script. Note also that its "Monitor" settings cannot be adjusted, as VisTool Editor is always displayed in full-screen on your computer's main display.

In our example above, the red error message can be ignored as the DMS Settings are not relevant to the **<Main>** session.

If you have not already done so, the first step is to create a new session for the first VisTool runtime screen.

### 4.5.3 Creating a New Session

1. Select **Copy** to make a copy of the current session - the new session is automatically named and appears as a new tab at the top of the window (e.g. **Session-1**):



You can also select **Add** to create a new empty session (with no parameters specified). However, it is usually better to make a copy so that all existing parameters are carried forward.

At this stage, you could select **Copy** several times to create multiple sessions - they are automatically named: **Session-1**, **Session-2**, **Session-3**, **Session-4**, etc. However, for our example, it makes more sense to define the session parameters first, so that they can be carried forward.

If your VisTool PC supports multiple screens, then you *must* create a new session for each screen.

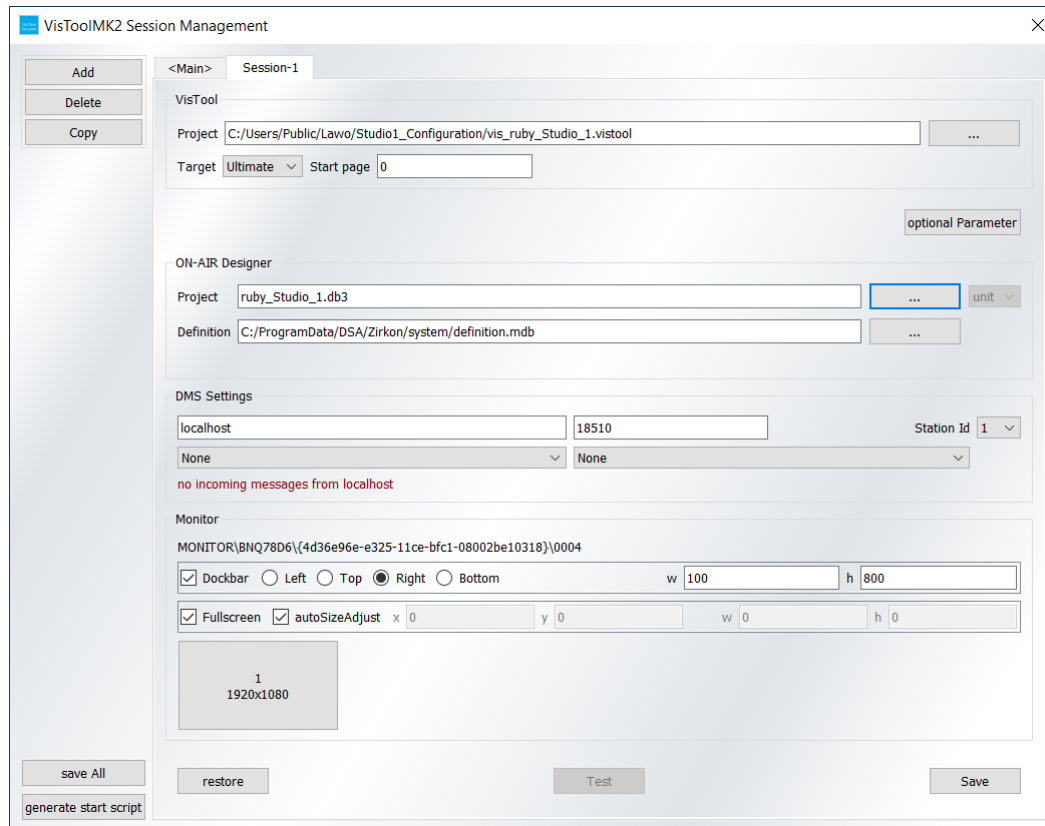
### 4.5.4 Deleting a Session

1. To delete a session, select its name tab at the top of the window, and click on **Delete**. A confirmation dialogue box appears.
2. Select **Yes** to confirm the delete.

### 4.5.5 Editing the Session Parameters

1. Select a session tab to view or edit its parameters:

*Session Parameters (example)*



The screenshot shows the 'VisToolMK2 Session Management' window with the 'Session-1' tab selected. The parameters are organized into several sections:

- VisTool:** Project: C:/Users/Public/Lawo/Studio1\_Configuration/vis\_ruby\_Studio\_1.vistool; Target: Ultimate; Start page: 0.
- ON-AIR Designer:** Project: ruby\_Studio\_1.db3; Definition: C:/ProgramData/DSA/Zirkon/system/definition.mdb.
- DMS Settings:** localhost; 18510; Station Id: 1; None; None; no incoming messages from localhost.
- Monitor:** MONITOR\BNQ78D6\{4d36e96e-e325-11ce-bfc1-08002be10318}\0004; Dockbar (checked); Left (radio); Top (radio); Right (radio, selected); Bottom (radio); w: 100; h: 800; Fullscreen (checked); autoSizeAdjust (checked); x: 0; y: 0; w: 0; h: 0.

Buttons at the bottom include 'save All', 'generate start script', 'restore', 'Test', and 'Save'.

The parameters are divided into the following areas:

- **VisTool** - specifies the VisTool project, target and its default Start page.
- **ON-AIR Designer** - specifies the accompanying ON-AIR Designer files (required for DMS devices only).
- **DMS Settings** - specifies the DMS service and Station Id for the DMS connection (required for DMS devices only).
- **Monitor** - specifies the physical screen where the session will be displayed and its appearance: full-screen, frame-less window, etc.

To control only Ember+ devices and no DMS unit, you must disable the DMS session parameters (as described [later](#)).

If DMS is already disabled, then the **ON-AIR Designer** and **DMS Settings** areas are hidden from view. To reveal them so that DMS can be re-enabled, enter a different project into the **VisTool Project** field.

The next few pages describe all possible parameters.

### VisTool & ON-AIR Designer



#### ➤ VisTool Project

This field specifies the VisTool project. Click on the ... button to select a file, taking care to select a valid file type (**.vistool**).

A red error message appears if the selection is invalid. This can happen if the file path is wrong (because the file has been moved); the file type is invalid (for example, if an ON-AIR Designer file has been selected); or if the project is out of date (because it was created by an older version of software).

#### ➤ Target

This field specifies the type of runtime [license](#). For VisTool Solo, you must select **Solo**. For VisTool Solo Plus, you must select **Solo Plus**. Otherwise, choose **Ultimate** for anything else.

#### ➤ Start page

This field specifies which page from the VisTool project will appear when the start script is opened. The number corresponds to the [page index](#) (which can be viewed from the page properties in VisTool Editor).

#### ➤ ON-AIR Designer Project

This field specifies the ON-AIR Designer project. It is required by DMS-controlled devices to support the VisTool functionality.

Normally, the field is completed automatically when you enter the VisTool Project (according to the [Project Properties](#) defined by **VisTool Editor**). If not, click on the ... button to select the correct **.db3** file. A red error message appears if the selection is invalid.

#### ➤ ON-AIR Designer Definition

This field defines the location of the "definition.mdb" file which is required to support the ON-AIR Designer project.

Normally, the field is completed automatically when you specify the VisTool project (according to the [Global Settings](#) defined by **VisTool Editor**). If not, click on the ... button to select the correct file. A red error message appears if the selection is invalid.

After a standard installation of the ON-AIR Designer software, the file can be found at:

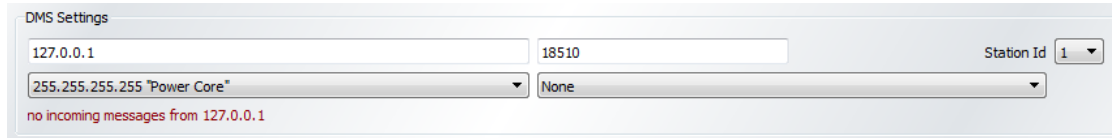
C:\ProgramData\DSA\Zirkon\system\definition.mdb

In Windows, this file path may be hidden from view, and so you may need to [enable](#) the "show hidden files and folders" option.



## 4. Configuring a PC for VisTool Runtime

### DMS Settings & Station Id



#### ➤ DMS Settings

These fields specify the DMS service which the session will connect to.

Click in the left drop-down menu to select the IP Address of the main control system; the options are determined by the [DMS Service Administration](#) window. If you entered an [Alias](#), then it will appear here in brackets ().

If you have multiple DMS services configured, then take care to enter the correct DMS service for the configuration data - for example, a **ruby** start script will not operate when connected to a **crystal** or vice versa.

The drop-down list on the right is for redundant systems. If a redundant control system is fitted, then select its IP Address from here. If no redundant control system is fitted, then set this field to **None**.

#### ➤ Station Id

The **Station Id** maps the session to a "[Logic -> VispageSwitch](#)" element (defined in the ON-AIR Designer project). It allows control signals from the remote device to action VisTool functions such as page switching, snapshot save/load/delete and user access rights.

To configure this function, first define a "Logic -> VispageSwitch" element in the ON-AIR Designer project. Then enter the corresponding **Station Id** number into the VisTool session's **Station Id** field. Any VisTool session set to a matching **Station Id** will then respond to the "Logic -> VispageSwitch" element.

If you want each screen to respond to a different set of control signals, then define one element per screen and make sure that the **Station Id** for each element/session is unique.

Enter a **Station Id** = 0 to disable the session so that the VisTool screen will not react to any "Logic -> VispageSwitch" element.

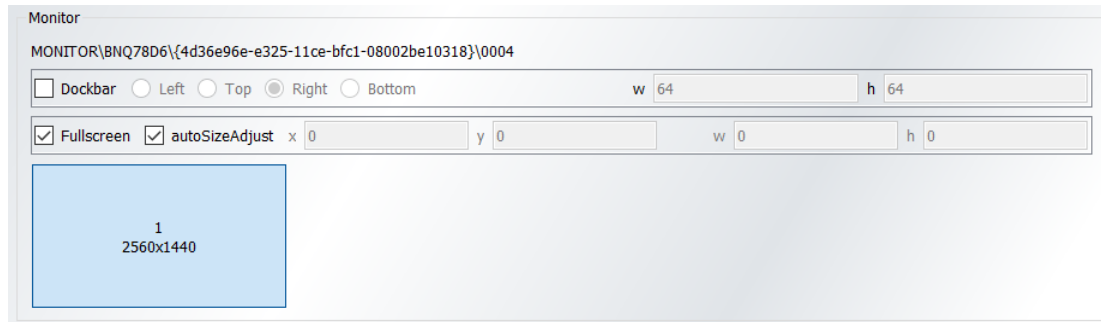
If the **Station Id** is entered incorrectly, then your VisTool snapshots or user access rights may stop working.

### ➤ Monitor

The options in this area specify the physical screen where the session will be displayed and its appearance.

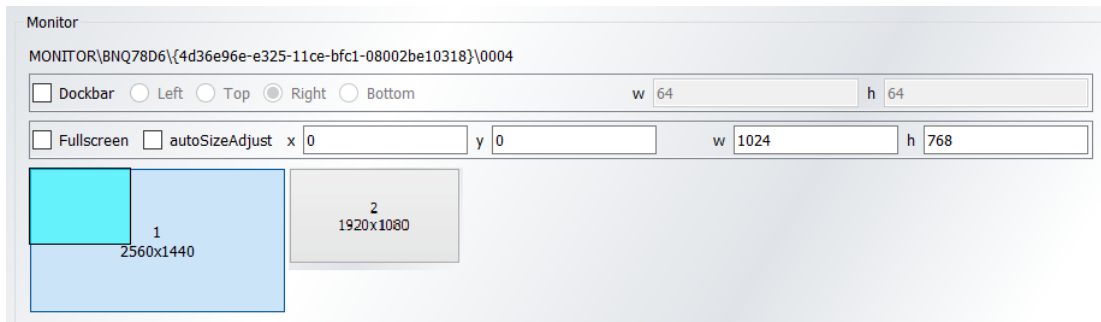
At the bottom you will see a representation of the physical screen(s). The number and resolution depends on the screens connected to the PC and the "Display" settings defined by Windows.

*Example 1*



In the example above, a single 1920 x 1080 monitor is available. In the example below, there are two monitors of different resolutions.

*Example 2*



1. If more than one monitor is available, click on an icon to choose the monitor for the session - an ident appears on the connected screen to help confirm its identity.
2. Then use the remaining options to determine how the session will appear: in full-screen, as a frame-less window or as a docking bar.

To view the session in full-screen, select the **Fullscreen** tickbox (as shown in example 1).

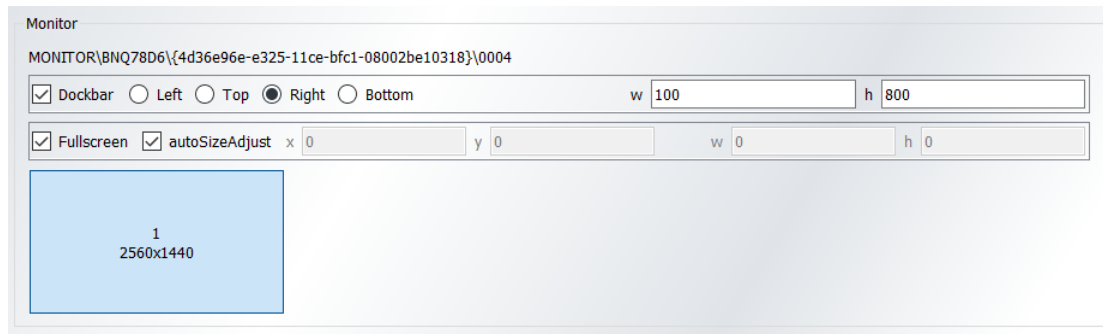
Alternatively, untick **Fullscreen** to view the session as a [frame-less window](#) (as shown in example 2). You can adjust the position and size of the window by entering pixel values into the **x**, **y**, **w** and **h** fields:

- The width (**w**) and height (**h**) fields adjust the window's size.
- The **x** and **y** fields adjust the window's position from **0, 0** (top left corner). Enter a positive **x** value to shift the window to the right. Enter a positive **y** value to shift the window down the screen.

Take care not to make the dimensions too small, as some dialogue boxes require a minimum window size.

## 4. Configuring a PC for VisTool Runtime

Example 3 (DockBar enabled)



In both cases, if you wish to display a [docking bar](#), then set the DockBar position to **Left**, **Top**, **Right** or **Bottom**. Then use the **w** (width) and **h** (height) fields to set the maximum resolution of the DockBar screen. The docking bar itself must be defined by the VisTool project (as a specific page). Therefore, its actual dimensions are set by **VisTool Editor** (via the page properties).

The docking bar [page size](#) should not exceed the **w** (for **Left/Right**) or **h** (for **Top/Bottom**) values, otherwise objects may fail to appear on the docking bar.

Functions which action a dialogue box (such as Login) can be included in the docking bar page, and appear as pop-ups over the main display area.

### 4.5.6 Saving or Discarding Parameter Changes

If changes have been made since the last save, then the **Save**, **restore** and **save All** buttons become available:

1. Click on **Save** to save the current session settings.
2. Click on **Save all** to save settings for all sessions.
3. Or, click on **restore** to discard any changes and restore the previous settings.

Once a save or restore has been actioned, the **Save** and **restore** buttons are greyed out - this shows that all entries are current and no changes have been made since the last save.

### 4.5.7 Testing a Session

Before generating a start script, it is a good idea to test the sessions:

1. Select a session, using the name tabs at the top of the window.
2. Check that the **Monitor** is set to the correct VisTool screen, and that the **Start page** is correct.
3. Then save the settings and click on **Test**.

The configuration data is compiled, and after a few seconds you should see the screen update to show the designated VisTool page. If the remote system is online, then all of the functionality can be tested, including page switching if this is defined by the configuration.

You can use PAGE UP and PAGE DOWN (on your computer keyboard) to select the next/previous page. This provides a quick way to check all of the pages saved in the VisTool project without having to change the session or screen.

4. To exit the online test mode, press ALT + F4 on your computer keyboard. Or wait five minutes until the test mode times out.

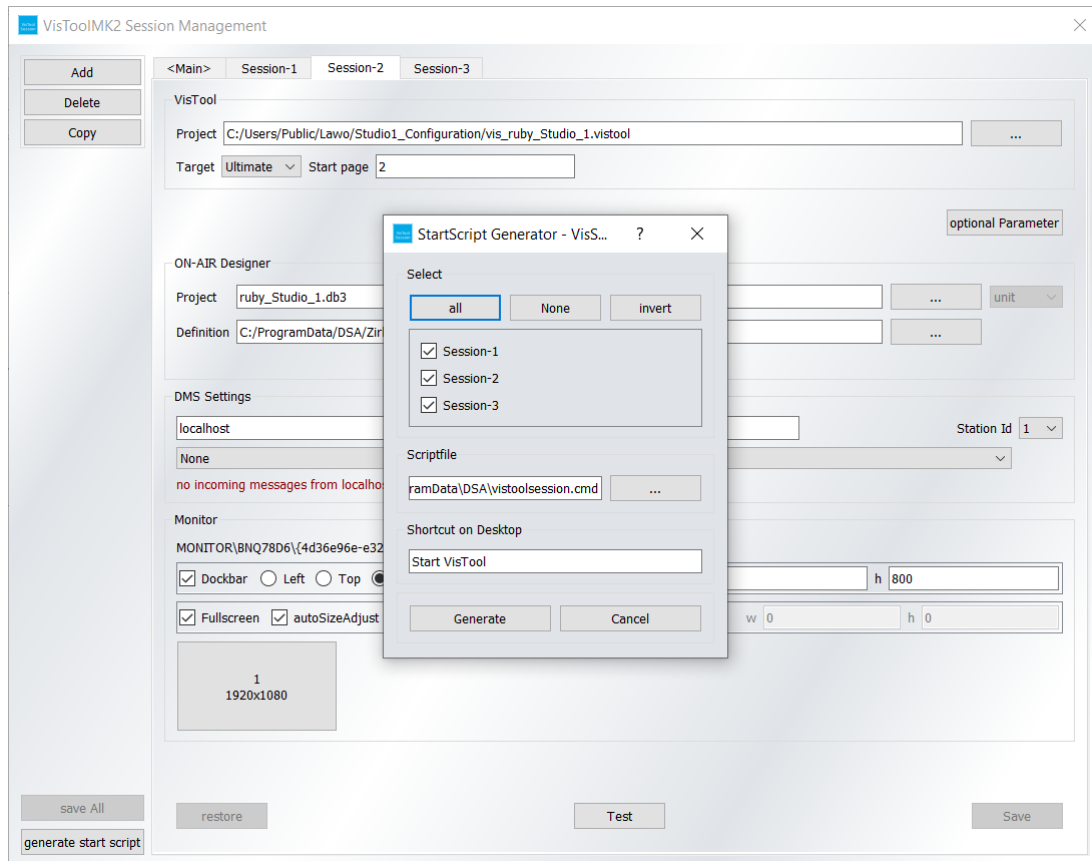
The online test mode always times out after five minutes and, therefore, you cannot use this mode to bypass VisTool runtime.

### 4.5.8 Generating a Start Script

Once you created all of the sessions and edited their parameters, it is time to generate a start script.

1. Click on **generate start script** - if you have made changes since the last save, then a dialogue box appears asking if you wish to save first. Select either **Yes** or **No** to continue.

A pop-up window appears:



2. From here you can define three things:
  - Which sessions will be included in the start script. If you have lots of sessions, then the **all**, **None** and **Invert** buttons provide useful shortcuts.
  - The file name and path for a Script file (.cmd). This can be used to launch the VisTool runtime from an external application.
  - The name used for the desktop shortcut. In our example, this is **Start VisTool**.

When selecting the sessions, remember that each start script can contain one or more sessions. This allows you to run several VisTool screens from the same PC (as shown above), or create different start scripts for different productions and users.

If you are generating start scripts for different Lawo devices, then take care to combine the correct session(s). Otherwise, your start script may not operate correctly.

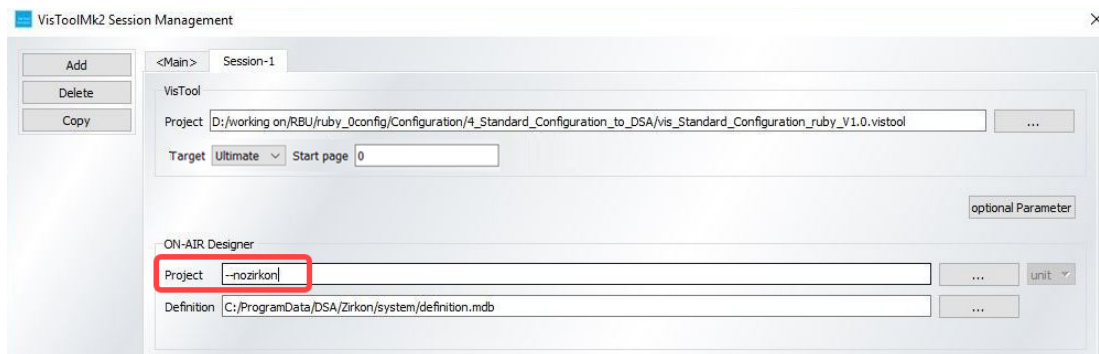
3. Select **Generate** to create the start script - the desktop shortcut and script file are created.
4. You can now double-click on the start script icon to [launch](#) the VisTool runtime.

### 4.5.9 Disabling DMS Parameters

If you wish to control only Ember+ devices, and no DMS unit, then you must disable the "DMS" session parameters. This can be done in one of two ways: via the "Session Management" window or by editing a VisTool start script (for an existing session).

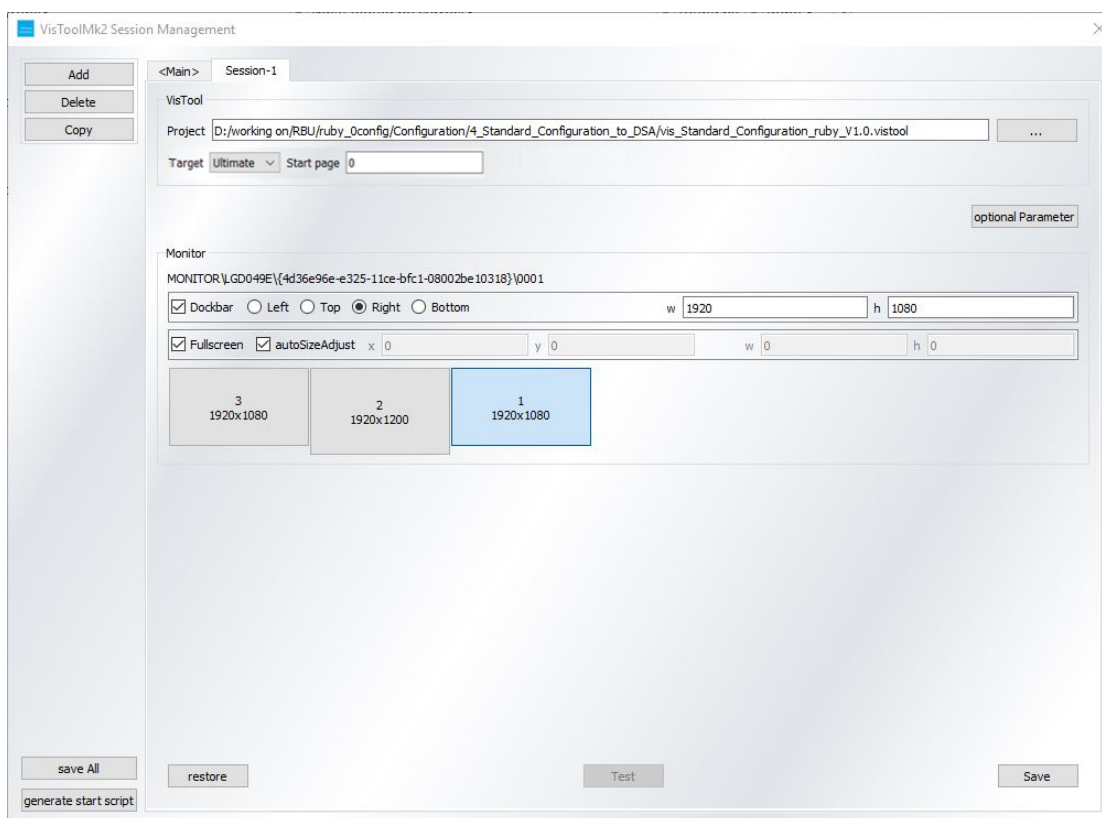
#### ➤ Disabling DMS Parameters via the "Session Management" window

1. Enter a project into the **VisTool Project** field as normal.
2. Then type "--nozirkon" into the **ON-AIR Designer Project** field and press Enter.



The DMS parameters are disabled and the "Session Management" window updates to show only the relevant parameter areas: **VisTool** and **Monitor**.

*Session Parameters (DMS disabled)*



3. Complete the remaining fields in the usual manner.
4. Then save the session and generate a start script.
5. When you launch the VisTool runtime, the DMS parameters will be disabled.

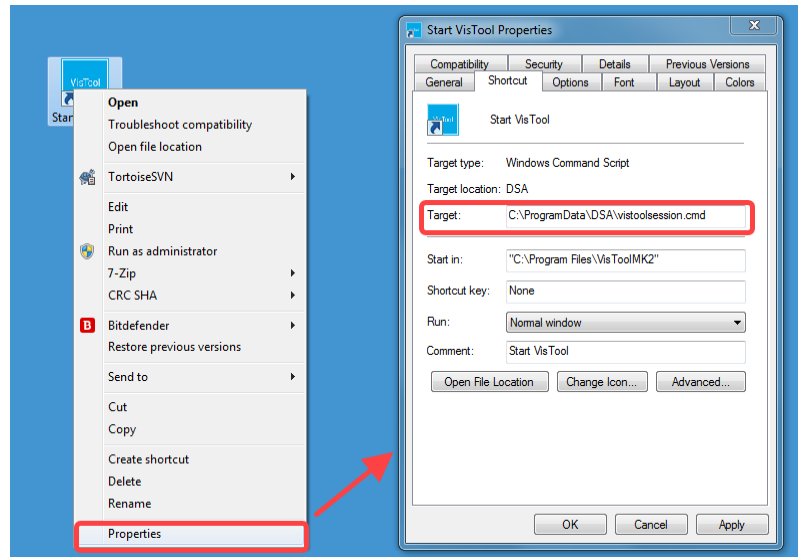
To reveal the **ON-AIR Designer** and **DMS Settings** areas so that DMS can be re-enabled, enter a different project into the **VisTool Project** field.

### ➤ Disabling DMS Parameters by Editing a Start Script

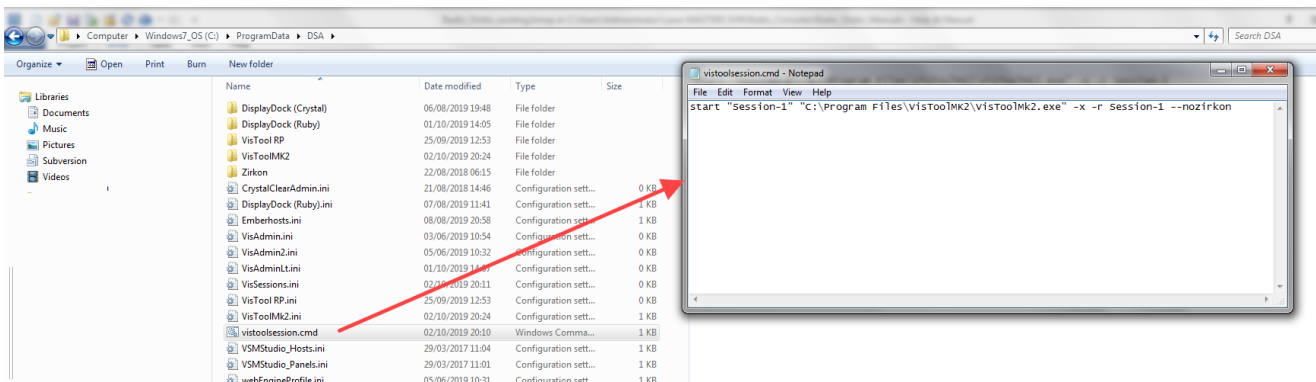
If an existing start script has already been generated, then you can disable its DMS parameters by editing the script.

1. Right-click on the VisTool start script icon and select **Properties**.
2. Then select the "Shortcut" tab to reveal the file path of the "Target" file.

By default this is "C:\ProgramData\DSA\vistoolsession.cmd".



3. Locate the file and open it using a suitable text editor such as Notepad.
4. Add "--nozirkon" to the end of the session description and save the changes.



5. When you launch the VisTool runtime, the DMS parameters will be disabled.

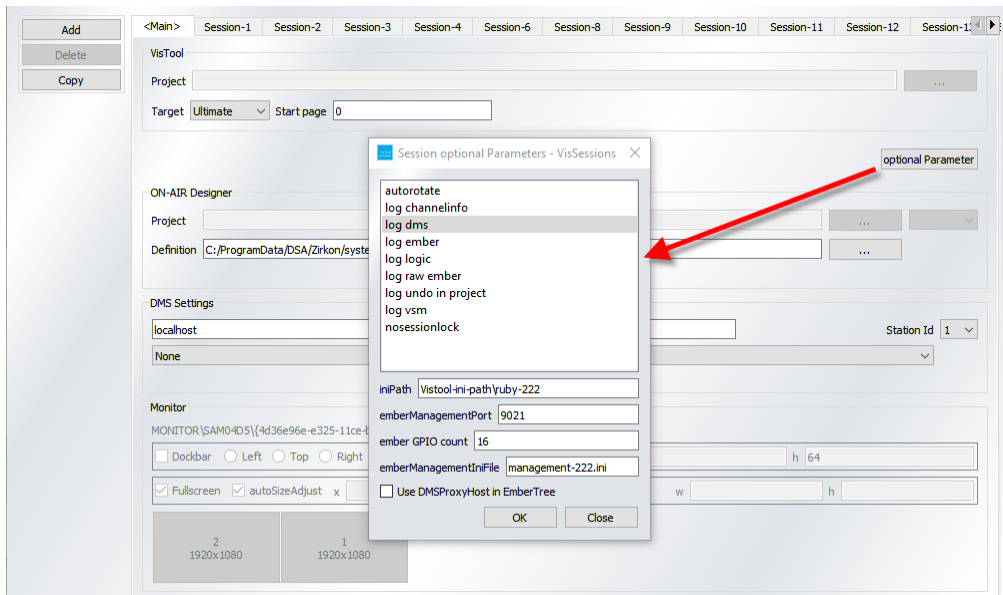
To re-enable DMS, remove the text "--nozirkon" and save the changes.

### 4.5.10 Using a Shared File path for ".ini" Files

If you are running multiple VisTool instances which share the same Ember+, V\_line or VSM devices, then it can be useful to create a single directory for the ".ini" files so that they may be shared between users. Note that this is not required for normal operation.

The ".ini" files store information about each Ember+, V\_line and VSM device, and are updated whenever you add one of those devices to the Ember+, V\_line or VSM Studio devices. If you followed a standard install, then the file path is: **C:\ProgramData\DSA**. The files are named **vsmStudio\_hosts.ini**, **vsmStudio\_Panels.ini**, **Emberhosts.ini** and **V\_line\_series\_Hosts.ini**. They are used by both VisTool runtime and VisTool Editor.

1. First, create a suitable directory for the shared files and copy all of the ".ini" files to that location.
2. Then open the "Session Management" window and select the **optional Parameter** button:



3. Type the desired ".ini" file location into the **iniPath** field and select **OK**.

When VisTool starts up, it will now search for the files in the specified folder (and not in C:/ProgramData/DSA).

Note that a separate file location can be specified for the [Ember Management port](#).

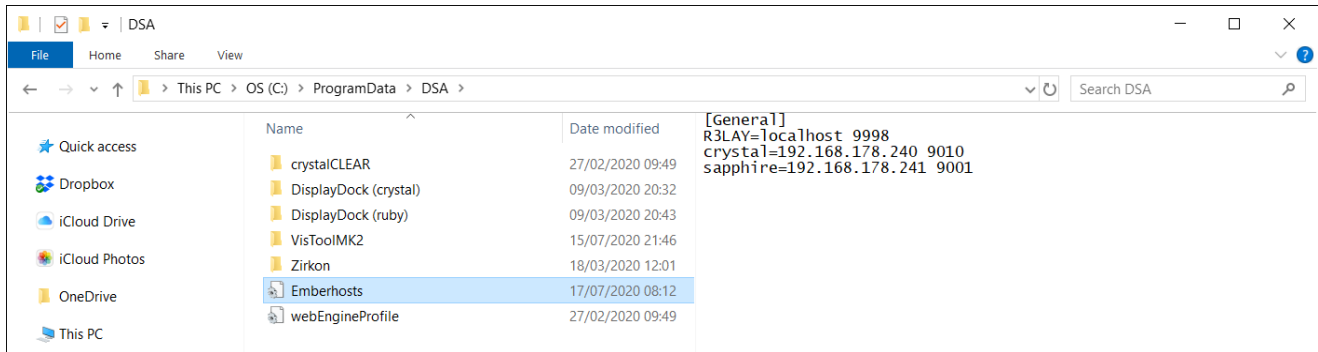
4. Repeat the process for all VisTool instances.

### 4.5.11 Deactivating a Device (in the ".ini" Files)

You can deactivate an individual host (or VSM panel) by editing the corresponding ".ini" file.

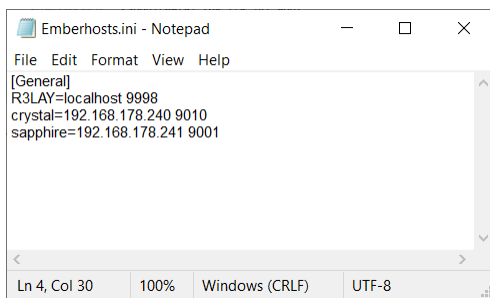
If you followed a standard install, then the file path is: **C:\ProgramData\DSA**. The files are named **vsmStudio\_hosts.ini**, **vsmStudio\_Panels.ini**, **Emberhosts.ini** and **V\_line\_series\_Hosts.ini**. Each file is created once you add one or more devices (in VisTool Editor). For example, the **Emberhosts.ini** file corresponds to the [Ember+ Devices list](#).

1. Locate the file and open it using a suitable text editor such as Notepad.

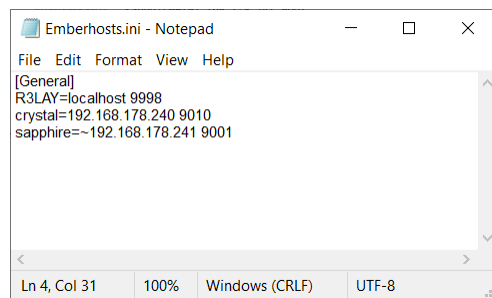


2. To deactivate a host, add the character "~" directly before the IP address.

*All hosts activated*



*sapphire deactivated*



3. Save the file and restart VisTool.



### 4.6 Licensing

To open a start script, the correct license must be installed on the runtime PC. The possible options are:

- **VisTool Standard** - is included free of charge with ruby, crystal, sapphire and sapphire compact consoles. This license runs the standard template for your product with a fixed set of pages and elements.
- **VisTool Unlimited** – is a paid upgrade which will run any VisTool start script, including custom configurations designed by VisTool Editor. The resource numbers are "unlimited", and the functionality can be expanded by purchasing the following add-ons:
  - **Ember+/Logic AddOn** - is required to support Ember+ and the Logic Engine.
  - **VSM Button Panel AddOn** - is required to support VSM-Button elements.
- **VisTool Solo** - provides a cost-effective alternative to VisTool Unlimited. This license supports custom configurations but with restrictions on some resources. You cannot use add-on licenses with VisTool Solo.
- **VisTool Solo Plus** - provides a mid-way point, price-wise, between VisTool Solo and VisTool Unlimited. This license supports more resources than VisTool Solo but less than VisTool Unlimited. You cannot use add-on licenses with VisTool Solo Plus.

The VisTool license can be installed into a local container (on your PC), a remote container (on a networked server), or onto a USB dongle (for portability). If you wish to use a dongle, then this must be ordered separately.

The license is activated and managed by the CodeMeter Runtime licensing system from [WIBU systems](#). The license code can be found on the delivery note shipped with the software.

#### 4.6.1 Using a Dongle

All dongles are specially-configured USB memory sticks which can be purchased from either [Lawo](#) or [WIBU systems](#). The dongles supplied for software and hardware products ship with different file systems, so please take care not to mix up the different types. If you have purchased a dongle for VisTool, then this will have a metal "LAWO logo" tag attached as shown below. Multiple software products can be licensed from a single dongle. Dongles supplied for hardware products such as Power Core are stand-alone (with no metal tag).

*USB Dongle (for Lawo software)*



### 4.6.2 Preparing for Activation

To activate a license, you will need:

- A PC with an internet connection (and USB port if using a dongle). If your PC does not have an internet connection, then the offline activation method can be used.
- The WIBU systems USB memory stick (if using a dongle).
- The license code. This can be found on the delivery note shipped with the system. It takes the form of a 25-digit ticket number such as the one shown below.

*License Code Example*

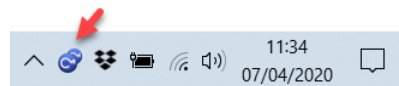
<b>Ticket:</b>
7MAMJ-8HZ95-N9VW5-3MKX6-LWUYM

Once activated, it is strongly recommended that you backup your licenses (using the **CodeMeter** [WebAdmin](#) portal). This will allow you to restore a license if the original is lost or damaged.

### 4.6.3 Installing CodeMeter Runtime

To activate a license, your PC must be installed with **CodeMeter Runtime** (from WIBU systems). The correct version is installed automatically with the setup software for your product.

You can check the installation by looking in the Windows taskbar where you should see the following icon.

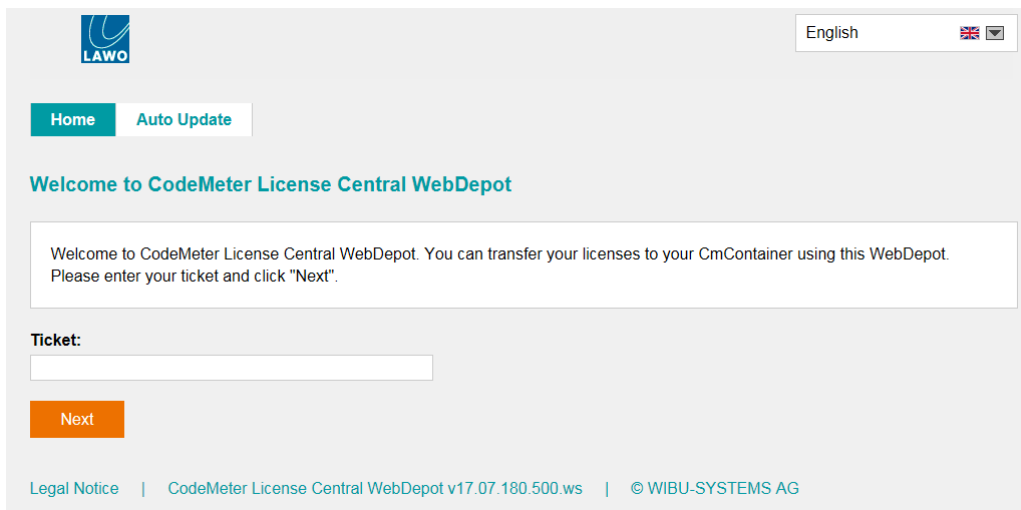


This shows that a Cm container (for local license storage) has been installed.

### 4.6.4 Activating a License Online

To use this method, your PC must have an internet connection. If installing onto a dongle, then this should be connected to the PC's USB port.

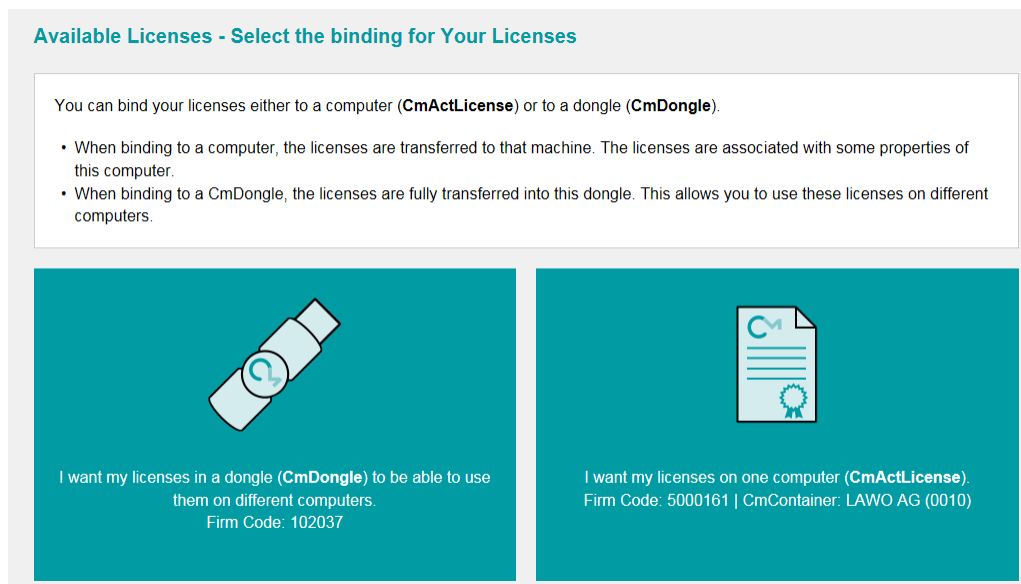
1. Open the Lawo licensing web page by copying the following URL into your web browser:  
<https://licenseportal.lawo.com>



If necessary you can choose a different language using the drop-down menu at the top right of the page.

2. Copy your license ticket number - this is the 25-digit number code - into the **Ticket** field and select **Next**.
3. The WebDepot searches for and displays your licenses - select **Activate Licenses** to continue.
4. At the next page, select the storage method for your license - note that this cannot be altered later.

Choose either **CmDongle** (to create a USB dongle) or **CmActLicense** (to bind the license to the local computer).



5. At the next page, select the licenses you wish to activate and the **CmContainer** to be used for the license storage.

### Available Licenses

**To activate your licenses:**

1. Select the licenses you want to activate.
2. Select the locally connected CmContainer to which you want to transfer the licenses.
3. Click "Activate Selected Licenses Now".

<input checked="" type="checkbox"/>	Name	Activated On	CmContainer	Status
<input checked="" type="checkbox"/>	*Product Name* <small>(License Quantity: 1)</small>			Available

**Select CmContainer**

128-2311304 (LAWO AG) ↻

Activate Selected Licenses Now Offline license transfer

[← Select binding](#)  
[← My Licenses](#)

You can store multiple licenses in the same container. If no Cm containers are available, then you will see an option to **"Get CmContainer automatically"**.

6. Click on **Activate Selected Licenses Now** and wait for a few seconds - a confirmation pop-up appears once the activation is successful:

#### Online License Transfer

**!** **Please wait!** The selected licenses are transferred. **This process may take several minutes to complete.** Please do not remove the CmContainer during this process and do not reload this page.

Starting license transfer.  
Creating license request.

#### Online License Transfer

Starting license transfer.  
Creating license request.  
Downloading license update.  
Importing license update to CmContainer.  
Creating receipt.  
Uploading receipt.

License transfer completed successfully!

OK

7. After selecting **OK**, a summary appears:

Home
My Licenses
Auto Update

### My Licenses

Name	Activated On	CmContainer	Status
*Product Name* <small>(License Quantity: 1)</small>	2018-07-05 18:33:51	128-2311304	Available: 0 (1)

Re-Host Licenses

8. You can now close the browser and return to your Lawo software application or install the USB dongle. For information on re-hosting a license, offline activation, backup/restore and using a license server, please see the [Advanced Licensing Features](#) appendix.

## 5. VisTool Runtime Operation

---

This chapter describes the operation of VisTool runtime.

Topics include:

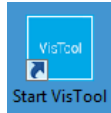
- [Basic Operations](#)
- [Page Navigation](#)
- [Buttons](#)
- [Faders](#)
- [Sliders & Rotary Encoders](#)
- [Metering](#)
- [Source Parameter Control](#)
- [Clock & System Time](#)
- [Timers](#)
- [Master Functions](#)
- [Matrix Control](#)
- [Web Pages](#)
- [Previewing Video](#)
- [Editing User Labels](#)
- [VisTool Snapshots \(PC Snapshots\)](#)
- [User Access Rights](#)

### 5.1 Basic Operations

#### 5.1.1 Starting VisTool

VisTool starts in "runtime" whenever you open a start script.

A shortcut to the start script can be found on the desktop - double-click on its icon to launch the VisTool GUI:



VisTool starts up and loads the prepared elements - you should see your VisTool screen(s) come to life and then the [Connections](#) panel. If not, check the following:

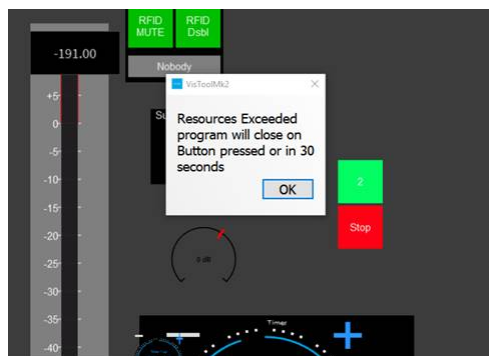
- Do you have a valid network [connection](#) between the VisTool PC and the Lawo control system?
- Is VisTool correctly [installed](#)?
- Is the start script correctly configured by [VisTool Sessions](#)?
- Have you [activated](#) the software license?

If an active license is not found when you launch a start script, then an error message appears. In this instance, please activate your license to continue.

#### 5.1.2 Resources Exceeded Warning (for VisTool Solo or Solo Plus)

If the runtime license is for **VisTool Solo** or **VisTool Solo Plus**, then the following warning message will appear (after 1 minute) if the session exceeds the permitted resources.

*"Resources Exceeded" warning message*



In this instance, the session will close, either on the next button press or after 30 seconds. There are two ways to continue: either activate a higher resource license, or modify the VisTool project (using VisTool Editor) and generate a new start script (using VisTool Sessions).

For more information on the permitted resources, please see [VisTool Solo & Solo Plus](#).

## 5. VisTool Runtime Operation

### 5.1.3 Screen Modes

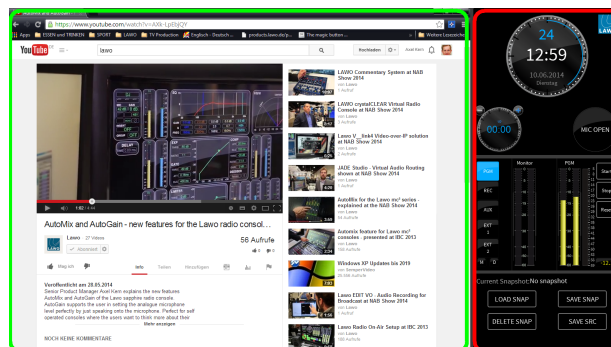
Depending on the [Monitor](#) options defined in the Session Management window, the VisTool session will open in one of three modes: full-screen; as a frame-less window; as a docking bar.

The last two options make it possible to share a single screen with other applications. VisTool occupies part of the screen display, while other applications can be maximized into the remaining display space:

*VisTool as a Frame-less Window (in red)*



*VisTool as a Docking Bar (in red)*



### 5.1.4 Adjusting Parameters

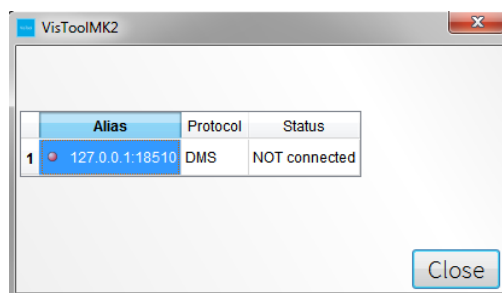
If your external monitor has touch-screen capability, or you are working with the [Visual Extension](#), then the on-screen elements can be touched to operate them. For example, touch a button to enable it; touch and drag an on-screen fader to adjust its position.

If touch-screen functionality is not available, then use your mouse and mouse keys to select and adjust the elements. Some elements can be adjusted using a mouse-wheel: hover over the element and scroll the mouse-wheel to adjust the value.

If the VisTool screen is positioned close to the remote device (e.g. beside a console), then logical triggers from the device may also be programmed to operate common functions. For example, MF Keys to switch pages or enable common options.

### 5.1.5 The Connections Panel

The **Connections** panel appears if the connection to a networked device is lost. The panel shows the status of all networked devices used in the VisTool configuration:



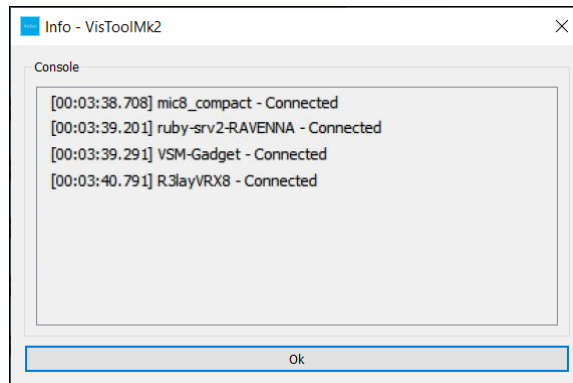
During runtime, each session/screen can control a single networked device via DMS and multiple devices via Ember+.

The status should be online for each device you wish to control. If a status is NOT connected, check that the device is powered and that it has a valid network [connection](#).

If you wish to stop the Connections panel appearing automatically, then change the **showHardwareNotResponding** option in the '[Project Properties](#)'. This can be useful if a DMS device is not currently available.

### 5.1.6 The Ember+ Connections Log

At any time, you can press CTRL + F12 (on your PC keyboard) to open the Ember+ Connections Log.



The log shows the time of every Ember+ connection (and disconnection) during the runtime session. If there are no Ember+ devices configured, then the list will be empty. If a redundant Ember+ connection is configured, then the active IP / hostname is displayed.

Note that the list does not refresh while the log is open, and so you should close the log (by clicking **OK**) once you have checked the list.

### 5.1.7 Closing VisTool

The VisTool runtime GUI can be closed as follows:

1. Press **ALT + F4** on your PC keyboard.
2. Touch the on-screen **Close** button (if one has been configured).

Note that on-screen buttons to switch to **fullscreen**, **minimize**, or **close** (the application) can be configured using VisTool Editor (via the [System](#) options described later).

If a "close password" is enabled for the configuration, then you cannot close the application without entering the password. This prevents you accidentally closing VisTool during an on-air transmission. The **Close PWD** is set within the VisTool Editor [Project Properties](#) window.



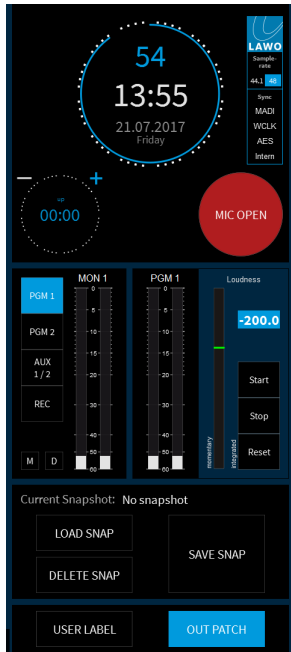
## 5.2 Page Navigation

If the configuration supports multiple pages, then you can change page using the **PAGE UP** and **PAGE DOWN** keys on your PC keyboard.

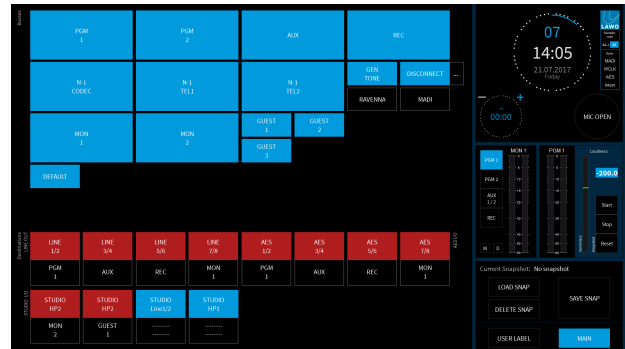
Page switching can also be implemented using VisTool page keys (on-screen) and/or logical triggers received from the remote system.

In the standard template for sapphire compact, the **OUT PATCH** and **MAIN** buttons are used to switch between the output patch and main docking bar (as shown below). In addition, whenever a source is placed in access (on the console), the [Source Parameters](#) page automatically appears.

Standard Template (MAIN dock)



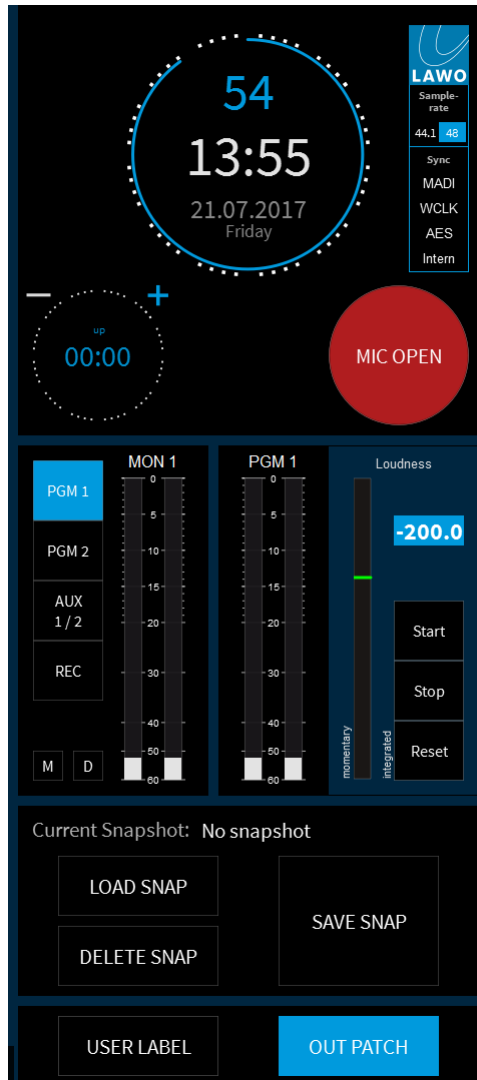
Standard Template (OUT PATCH)



For details on how to configure page keys, see [Configuring Page Switching](#) in the VisTool Editor chapter.

## 5.3 Buttons

Standard Template (MAIN dock)



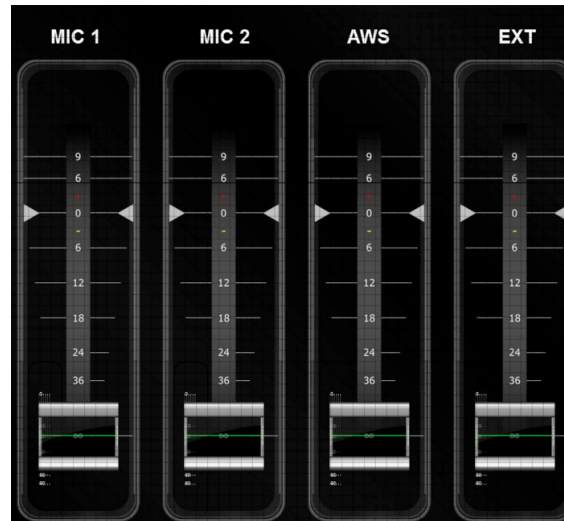
VisTool's on-screen buttons come in a variety of shapes and sizes. Each one can operate a function (when touched/pressed) or provide signaling. Typical applications include timer operation; MIC OPEN signaling; monitor source selection; loudness metering operation; snapshot management; and matrix control.

You can use either [Box](#) or [Button](#) elements in VisTool Editor to configure on-screen buttons. Box elements are recommended as they are much faster to display and scale proportionally.

A special button element ([VSM-Button](#)) can operate buttons on a VSM Panel configured by an external VSM server. This allows you to control or monitor VSM Panel buttons directly from a VisTool screen.

## 5.4 Faders

*VisTool Faders*



VisTool on-screen faders come in a choice of size, scale and style. A VisTool fader can control either a physical fader on the console or a real value from an Ember+ provider. Touch and drag the fader knob to change its position/level.

Use the [Fader](#) element in VisTool Editor to configure on-screen faders.

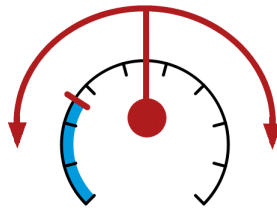
## 5.5 Sliders & Rotary Encoders

Sliders and Rotary Encoders can be linked to variable or stepped parameters within the remote system such as EQ gain, frequency, etc.

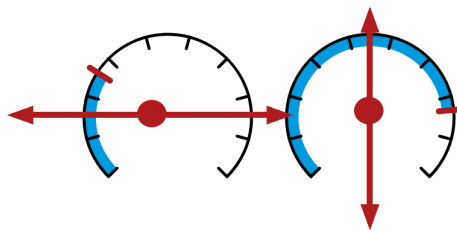
Sliders can appear as horizontal or vertical bars. They are operated just like a fader, so touch and drag to change the slider's position. If the **enable mousewheel** property is turned on, then you can also hover your mouse over the slider and use a mouse-wheel to adjust the slider's position/value.

Rotary Encoders appear as an on-screen encoder. Their operation will vary depending on the configuration of their **pointing mode** property:

- When **pointing mode** is on, the value is changed by a rotational movement outside the circle - touch and hold, and then turn:



- When **pointing mode** is off, the value is changed either by moving up and down, or left and right, relative to the initial position:

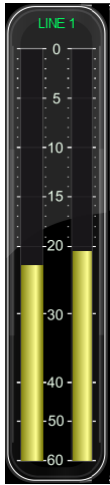


If the **enable mousewheel** property is turned on, then you can also hover your mouse over the encoder and use a mouse-wheel to adjust the encoder's position/value.

Use the [Slider](#) and [Rotary Encoder](#) elements in VisTool Editor to configure these components.

## 5.6 Metering

Channel PPM



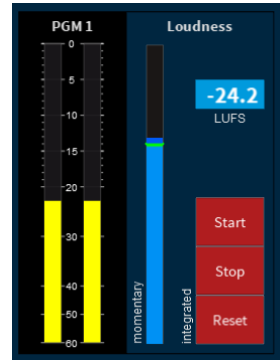
Channel State



PGM VU



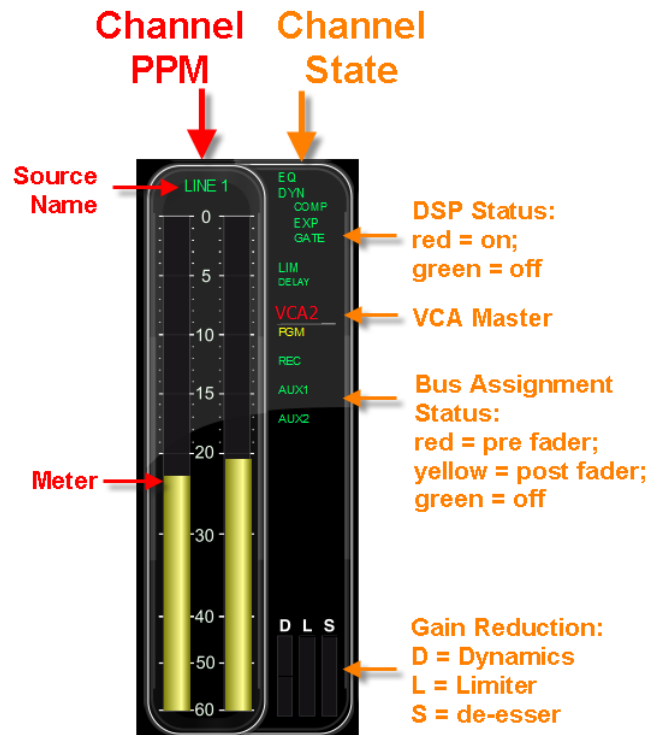
PGM PPM + Loudness



VisTool supports both PPM and VU meters which can link to specific channels, main meters, to the source in access or be freely-configurable (via the "DMS" option). In addition, there are a number of add-on elements which can be used to visualize channel states, display gain reduction, show the stereo audio correlation and measure loudness. The metering elements can be used stand-alone or combined as described in the examples below.

Use the [PPM](#) element in VisTool Editor to configure the components. Loudness metering is configured separately using the [Loudness](#) element.

## 5.6.1 Channel Metering



The example above shows a channel PPM combined with the Channel State add-on. Together they provide input metering for the channel plus feedback and signalisation of the channel states.

The elements have been assigned to a specific fader strip (by VisTool Editor), and include the Display name of the source (e.g. LINE 1). The meter scale can be DIN, UK, NORDIC or DIG (defined by its properties).

The Channel State element includes:

- **DSP Status** (EQ, DYN, COMP, EXP, GATE, LIM, DELAY) – these indicators turn from green to red whenever the DSP module is on. DYN is the status of the master dynamics on/off; COMP, EXP and GATE show the status of the individual sections.
- **VCA Grouping** (not available for crystal) - if the fader is assigned to a VCA, then the VCA master is shown here.
- **Bus assignments** – show the name and status of the first 10 (or 20) bus assignments as defined within the system configuration:
  - Red – indicates a pre-fader assignment.
  - Yellow – indicates a post-fader assignment.
  - Green – indicates no assignment.
- **Gain Reduction** - these three meters show gain reduction for the dynamics (D), limiter (L) and de-esser (S).

## 5.6.2 Output Metering



The example above shows a main meter PPM combined with a Loudness element.

The elements have been assigned to the PGM bus using the main meter options in VisTool Editor.

The PPM is stereo and includes the Display name of the bus (e.g. PGM 1). The meter scale can be DIN, UK, NORDIC or DIG (defined by its properties).

The Loudness components include a bargraph showing the Momentary integration time, and a text read-out showing the integrated loudness measurement (in LUFS). Use the on-screen buttons to **Start**, **Stop** and **Reset** the integrated loudness measurement.

## 5.6.3 Loudness Metering

The loudness metering conforms to the ITU-R BS1770.

### Loudness Metering Bargraphs

When using bargraphs, a single bargraph represents the average energy of the summed component channels: mono, stereo or surround. The average is measured using either a Momentary (400ms sliding window) or Short term (3s sliding window) integration time.

The bargraph color indicates whether loudness is above or below the Target Level. Scale markers indicate a tolerance of +/- 1 LU/LK.

### Integrated Loudness Measurement

The integrated loudness measurement provides a very useful tool for measuring loudness over long periods of time. For example to measure the loudness of a transmission from start to finish.

**Start**, **Stop** and **Reset** functions can be used to control the integrated loudness measurement. The measurement is displayed in LUFS (Loudness Units Full Scale).

### Loudness Range Measurement (LRA)

A loudness range measurement is similar to the integrated loudness measurement, described above, but is gated (at -70dB LUFS) to exclude silence from the measurement.

## 5.7 Source Parameter Control

VisTool Standard Template (Source in Access)

If you are running one of the standard templates, then whenever a source is placed in access (on the console), you can view or control its parameters from the VisTool page shown above. This page uses Parm buttons (for parameter control) and Access Source Information displays (to provide feedback on the source and its parameter values).

To operate our example:

1. Select a source by pressing an **ACCESS** key on the console surface - the DSP Control page appears.
2. For a switched function, press to turn on or off - e.g. **EQ**, **DYN**, **EXP ON**, **COMP ON**, etc.
3. If variable functions are configured, then these can be adjusting by clicking (or touching) on the + or - buttons. In the standard template, these are designed into the parameter fields. So, for example, to increase the Compressor Threshold, click on the right hand side of the COMP THRS value field.

The configuration of Parm buttons requires two stages: first use the ON-AIR Designer to prepare the system configuration; then use VisTool Editor to design the on-screen elements. An example is included [later](#).

The Access Source Information displays are configured using VisTool Editor's [Sources](#) and [Dynamic](#) elements. In the example above, you can see the source name; the status of all input parameters (Mic Gain, 48V, etc.); and graphs for the EQ, Limiter and Dynamics: Expander, Gate, Compressor, De-Esser.

The Dynamics and Limiter graphs include color-coded balls which represent the signal output level. These change dynamically as the signal level varies. On the Dynamics graph, there are two balls; the default colors are green for Dynamics and red for the De-Esser. On the Limiter graph there is a single ball; the default color is green.



### Mouse / Touch Operation via the EQ / Limiter Graphs

For the EQ and/or Limiter, it is possible to adjust parameters from the graphical display if the [mousecontrol](#) property is enabled in VisTool Editor. In this instance, a small circular handle appears for each EQ band and/or the Limiter Threshold. The size of the handles can be adjusted (in VisTool Editor).

1. For the EQ, click and drag (or touch and drag) on a handle to increase or decrease the gain and move the frequency.
2. Double-click (or hold down) on a handle to open a menu where you can adjust the Q value and change the EQ type.
3. For the limiter, click and drag on the handle to increase or decrease the threshold.

## 5.8 Clock & System Time



VisTool supports two elements which can show the internal time (and date) of the remote system. For Lawo's radio products this can be set using SoP Explorer (included with the ON-AIR Designer software).

Use the VisTool Editor [Clock](#) element to configure the example on the left, and [System Time](#) to configure the text display on the right.

In the Clock example on the left, the time is given using a 24 hour digital clock where hours and minutes appear in white and seconds in blue. For greater visibility, the "seconds" are also represented by the blue line/circle around the circumference of the display.

### 5.9 Timers

VisTool timers link to timers defined in the ON-AIR Designer configuration. They can be used to count-up from 00:00 or count-down from a starting value. Each one can be named and offers a choice of three modes:

- **Follow Mic** – the timer will automatically start (or stop) whenever a MIC source is on-air (or off-air).
- **Follow OnAir** – as above but the timer responds to any source: MIC, LINE, etc.
- **Free** – the timer must be controlled manually from **Start**, **Stop** and **Reset** buttons.

Use the VisTool Editor [Timer](#) element to configure the on-screen timers. Two different types of timer are supported: standard and server-based. Server-based timers can be used to synchronize the operation across multiple VisTools. If a parameter such as the timer direction, preset or alarm time is changed from one VisTool, then this is displayed in the other instances. The displayed time is also synchronized.

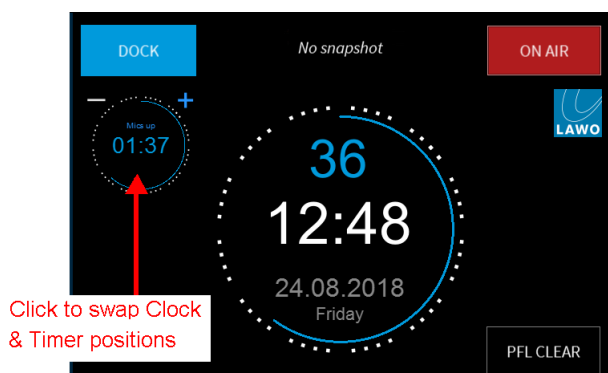
In the standard template, a single standard timer is included in the MAIN page. It is set to 'Follow Mic' which means that it will automatically start when at least one MIC source is on-air. The rest of this topic describes its operation.

#### Reversing the Clock & Timer Positions

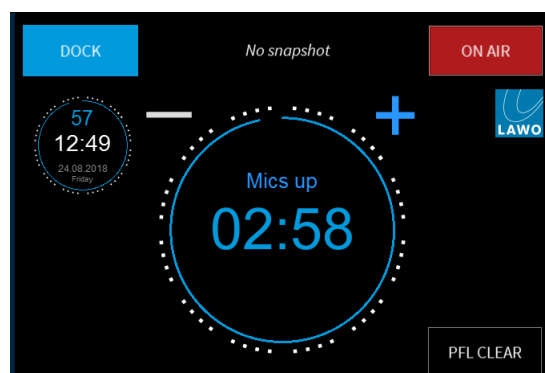
In the standard templates, you will see both a [clock](#) and a timer.

By default, the clock appears (big) with the timer (small). However, the functions can be reversed by clicking on the smaller of the two elements:

*Clock (big) and Timer (small)*



*Timer (big) and Clock (small)*



#### Changing from Count-Up to Count-Down

The timer supports two modes of operation: Count-Up or Count-Down. Select the + sign (it turns blue) to use the timer in count-up mode. Select the - sign (it turns red) to change the timer to count-down mode.

*Count-Up*



*Count-Down*



### Count-Up Timer

In this mode, the timer will count-up from 00:00.

1. The timer starts automatically when a MIC source goes on-air (fader open and channel ON).

For example, open the **MIC 1** fader (on the console) - the timer starts to count-up. If another MIC source opens, then the timer continues to run. At any time you can reset the timer to 00:00 by selecting the **+** sign.

2. The timer stops when all MIC channels are taken off-air, either by closing the faders or de-selecting the channel ON buttons.

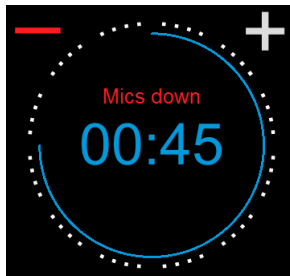
Note that the current time remains on-screen until the timer is reset.

3. The timer will reset automatically when you next open a MIC source. Or, it can be reset using the **+** sign.

### Count-Down Timer

In this mode, the timer will count-down from a starting value (called the **Preset** time), and turn red once a mid-point (the **Alarm** time) is reached:

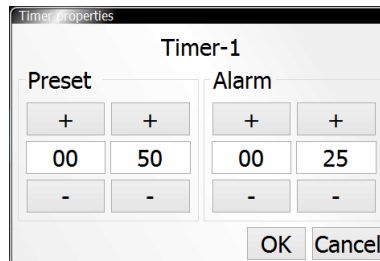
*Count-Down (before Alarm)*



*Count-Down (Alarm active)*



1. To set the **Preset** and **Alarm** time values, select the current count-down time - the Timer properties window opens:



The **Preset** value defines the starting time. When you reset the timer, it will return to this setting - in our example, to 50 seconds.

The **Alarm** value is used for the signalisation. When the count-down reaches this time, it turns red - for example, at 25 seconds.

In both cases, use the minute and second fields to type in a value, or select the **+** and **-** buttons to increase and decrease the current values. Then select **OK** to confirm.

2. The timer starts automatically when a MIC source goes on-air (fader open and channel ON).

For example, open the **MIC 1** fader (on the console) to start the count-down from the **Preset** time.

When the count-down reaches the **Alarm** time, the numbers turn red to warn you that you are almost out of time. If the count-down reaches 00:00, then the timer starts to count back up. This shows you by how much you are overrunning! In this instance, the Alarm status remains active until the timer is reset.

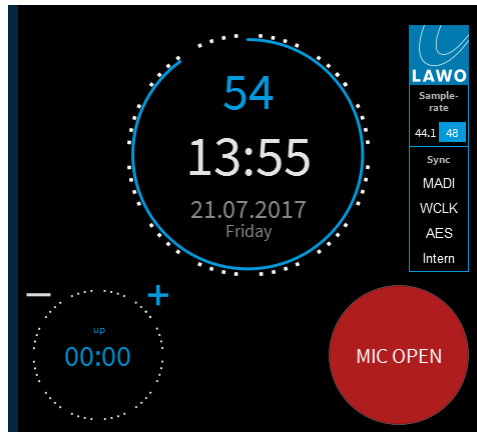
3. The timer stops when all MIC channels are taken off-air, either by closing the faders or de-selecting the channel ON buttons.

Note that the current time remains on-screen until the timer is reset.

4. The timer will reset automatically when you next open a MIC source. Or, it can be reset using the **-** sign.

## 5.10 Master Functions

Standard Template (MAIN dock)



Other master functions included in the standard template are the **MIC OPEN** indicator and the LAWO logo.

### 5.10.1 MIC OPEN (ON-AIR)

The MIC OPEN indicator lights in red whenever a microphone channel is going to air (fader open and channel ON). The state is triggered by any of the console's microphone sources. It is also sometimes labeled as ON AIR.

The MIC OPEN state may also trigger other actions, such as muting the Control Room Speakers and triggering a GPO (wired to an external red light). For specific details, please refer to the User Manual for your console.

The MIC OPEN indicator is configured using a [Box](#) element in VisTool Editor.

### 5.10.2 System Info

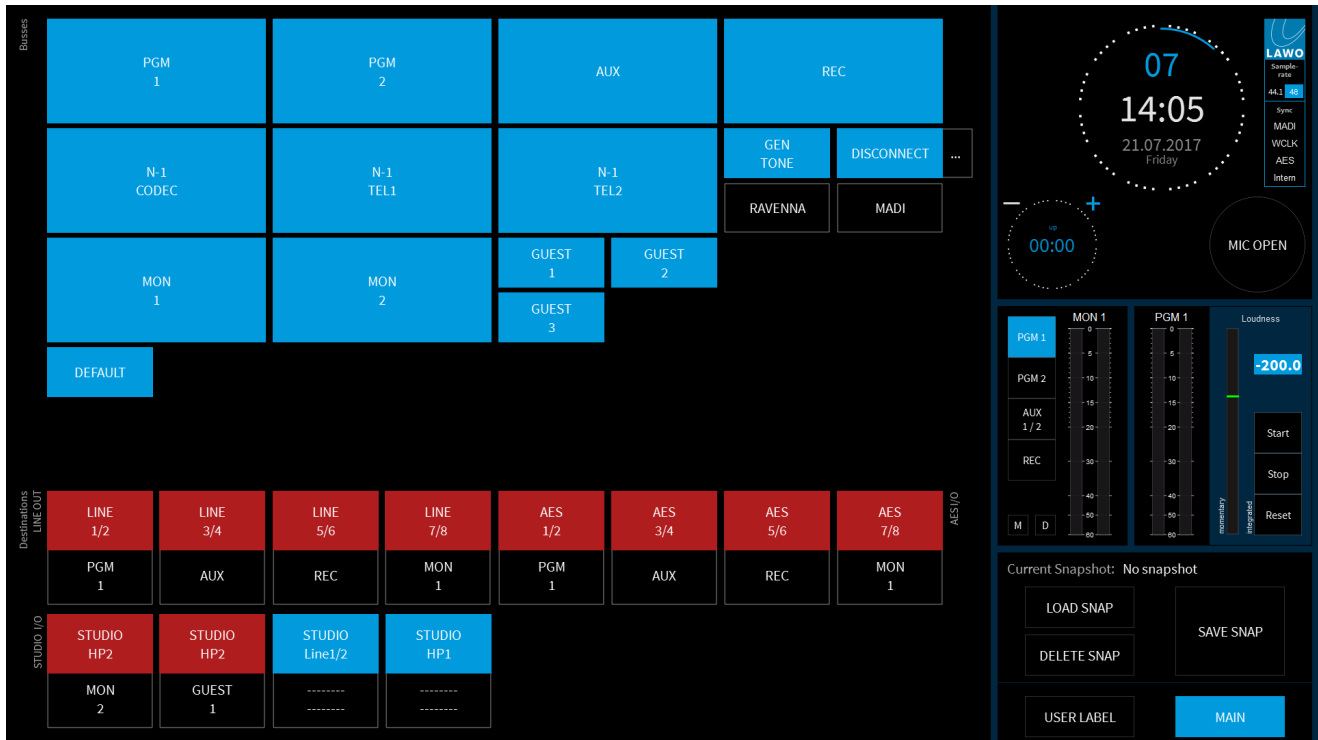
The LAWO logo can be selected (and deselected) to reveal (and hide) information about the **Sync** reference and **Sample rate**. The current reference and sample rate are highlighted in blue. Note that the fields provide information only; you cannot change the sync reference or sample rate from the GUI. For more details, please refer to the User Manual for your console.

The LAWO logo operation is configured using a [Button](#) element in VisTool Editor.

## 5.11 Matrix Control

A common application for VisTool is to create a matrix control panel using on-screen buttons.

*Standard Template (OUT PATCH)*



In the example above, the matrix is divided into "Sources" at the top and "Destinations" at the bottom.

If a destination is already assigned, then its button is red. If no audio is assigned, then the button is blue. The label below each "Destination" button shows the assigned source - for example, PGM 1 is assigned to LINE 1/2.

To change an output assignment:

1. Select a "Destination" button - the button starts to flash.
2. Select an available "Source" (e.g. PGM 1) - the routing assignment is made and the button changes to static (red).
3. Alternatively, repeat step 1 and then select **DISCONNECT** - the current audio source is removed and the button changes to static (blue).

The **DEFAULT** button can be used to reset the matrix to the default audio assignments.

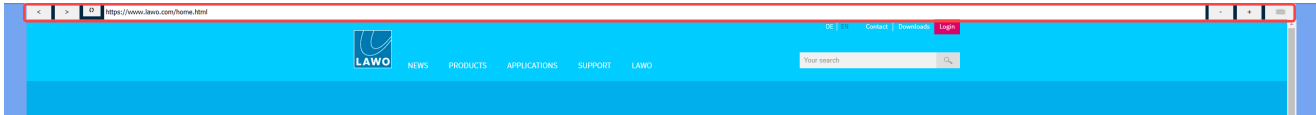
Use the [Matrix-Button](#) element in VisTool Editor to configure the on-screen matrix buttons.

## 5.12 Web Pages

A specific web page, or pages, can be viewed within VisTool. If configured, each web page should appear on a separate VisTool page. The web page address (URL) is defined in the configuration.

This feature is not designed to be a fully-fledged web browser but to display simple web pages, either with or without a navigation bar.

*Web Page (with Navigation Bar)*



If the navigation bar is included, then it provides the following functions:

- < and > (Forward and Back) - to next or previous page.
- Refresh button.
- **Address Bar** - displays the URL of the page. A new address can be entered from the keyboard. For touch-screen operation (with no physical keyboard), open the on-screen keyboard to enter a URL.
- + and - (Zoom in and out).
- On-screen Keyboard - select this icon to open the on-screen keyboard.

Use the [Web](#) element in VisTool Editor to configure this feature.

### 5.13 Previewing Video

You can display video streams from a V\_\_line device in VisTool. For example, to preview a video input to, or output from, a V\_\_pro8, V\_\_link4, etc.

The video elements are designed for preview only, and are not broadcast quality. Multiple thumbnails can be included.

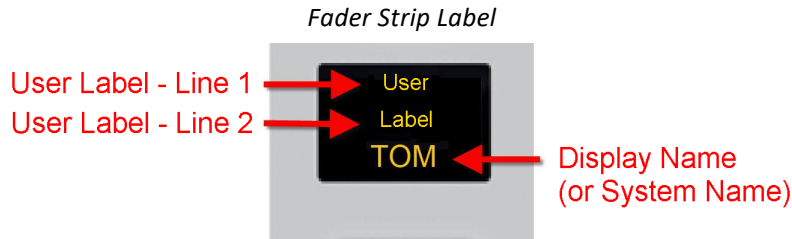
The video source can be either a local playback file or a HTTP stream coming from the network (e.g. from a V\_\_line device). Note that an external converter is required if you wish to display a non-HTTP streaming format.

Use the [Video](#) element in VisTool Editor to configure this feature.

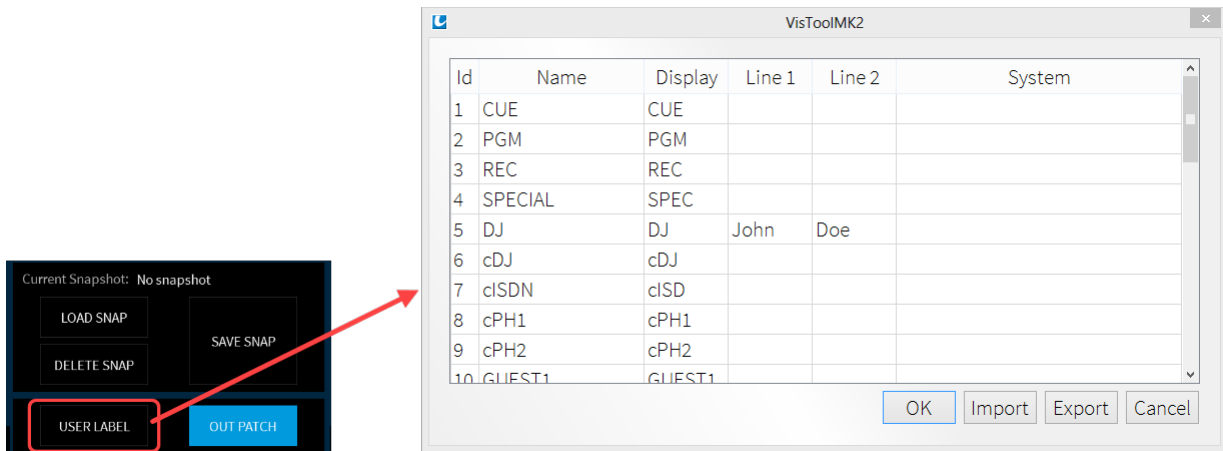


### 5.14 Editing User Labels

If user labels are enabled by the ON-AIR Designer configuration, then the text can be edited using VisTool. Up to three lines of text are supported.



Usually, a dedicated button is configured to open the user labels editor. In the VisTool standard template, this appears on the MAIN dock.



The user labels editor can also be opened from VisTool Editor (via the [Util menu](#)).

The **Id**, **Name** and **Display** fields are defined by the ON-AIR Designer configuration and cannot be changed. By default, a source uses its **Display** name unless a **System** name is entered.

1. Use the **Line 1** and **Line 2** fields to edit the two user label lines (up to 8 characters per line).
2. To replace the **Display** name, enter text into the **System** field. The **System** name replaces the **Display** name until such time as the **System** field is cleared.
3. Once you have finished editing, select **OK** to transfer the labels to the system - the editor closes and the labels on the control surface update.

The **Export** button will save all labels into an **.xml** file. This allows to copy the file to another VisTool PC and then select **Import** to import the labels. When using import, the labels are applied to sources with an identical source **Name** field.

You should take care when deleting a source from the ON-AIR Designer configuration for the following reasons.

Whenever a source is added to the configuration, it is assigned an internal "sys" number. These numbers are remapped if a source with a lower sys number is later removed. System labels are assigned to the sys number rather than the source name. Thus, if a source is deleted from the configuration, the labels of all sources with a larger sys number are moved (shifted downwards). The sys numbers can be viewed in the [Ember+ tree](#) to see what affect this may have on your system.

### Configuring Operator Access to the User Labels Editor

Usually, the user labels editor can be opened in one of two ways: a button on a VisTool page or an MF Key (or other control signal) from the remote system.

For option one, add a [Box](#) element to the VisTool project, and link this to the "[Predefined User Text](#) -> User Label -> Edit" option.

For option two, add a "Logic -> VispageSwitch" element to the ON-AIR Designer project, and define the [Edit PC Labels](#) function. It is important that the **Station ID** (defined for the "[Logic -> VispageSwitch](#)" element) matches the one entered in [VisTool Sessions](#). If not, the trigger from the remote system will not work.

### 5.15 VisTool Snapshots

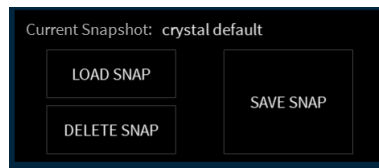
VisTool snapshots are stored on the VisTool runtime PC.

They are stored in the "Visconfigurations.mdb" file defined in VisTool Editor's "Global Settings" (via the [Snapshots](#) field). Usually the default file path is used so that data can be stored independently on each VisTool PC. Alternatively, if you wish to share snapshots between VisTools, then you can create a shared file location. The setup for this is described [later](#).

VisTool snapshots store the same settings as the five internal memories on Lawo's radio on-air consoles. When you save, you can choose to save either a **Full** snapshot (to save settings globally across the console), or **Source** snapshot (to save only the settings for the source in access).

Usually, a set of dedicated buttons are configured to operate the snapshot functions. In the VisTool standard template, these appear on the MAIN dock. The "Current Snapshot" field shows the name of the last snapshot loaded.

*Standard Template (MAIN dock)*



#### Configuring Operator Access to Snapshot Functions

Each snapshot function can be accessed in one of two ways: a button on a VisTool page or an MF Key (or other control signal) from the remote system.

For option one, add a [Box](#) element to the VisTool project, and link this to one of the "[Predefined User Text](#) -> Snapshot" options.

For option two, add a "Logic -> VispageSwitch" element to the ON-AIR Designer project, and define one of the [PC Snap](#) functions. It is important that the **Station ID** (defined for the "[Logic -> VispageSwitch](#)" element) matches the one entered in [VisTool Sessions](#). If not, the triggers from the remote system will not work.

The "Current Snapshot" field is also a [Box](#) element. This is linked to the [DisplaySnaphotName](#) function.

#### Protecting Snapshots (via User Rights)

Optionally, you can set up [User Access Rights](#) to create different levels of snapshot access for administrators and "standard" users. User Rights are not supported by the standard template but can be enabled by creating a user LOGIN button.

### 5.15.1 What's Stored in a Snapshot?

The following settings are stored in all snapshots:

- **Fader Strip Assignments** – the assignment of sources and buses to fader strips.
- **User Labels** – the two User Label text lines for all sources and buses.
- **RAVENNA/AES67 Stream subscriptions** handled via a RAVENNA pool.
- **Input Parameters** – such as mic gain, 48V, etc. for all sources.
- **DSP settings** – EQ, Dynamics, Limiter, Delay and De-Esser for all sources and buses.
- **Fader Levels (optional)** – see below.
- **Bus Assignments** – the assignment of sources to summing buses and their send levels.
- **Insert Assignments** – the assignment of insert devices for all sources.
- **VCA Group Assignments** – the assignment of all sources to VCA Groups.
- **Logical States** – up to 32 logical states defined by the ON-AIR Designer configuration (via "System -> Definition -> Parameter = Logic Snapshot").
- **Matrix Connects** – up to 128 matrix connections defined by the ON-AIR Designer configuration (via "System -> Definition -> Parameter = Matrix Snapshot").

#### Fader Levels

The **Production Mode (Save Value)** option, in the ON-AIR Designer configuration, determines whether fader levels are stored and recalled by snapshots. This option is set in the "System -> Definition -> Parameter = Fader" branch of the 'Tree Definition'.

- **Production Mode (enabled)** = fader levels are loaded. Recommended for production systems.
- **Production Mode (disabled)** = fader levels are ignored (not loaded). Recommended for on-air systems.

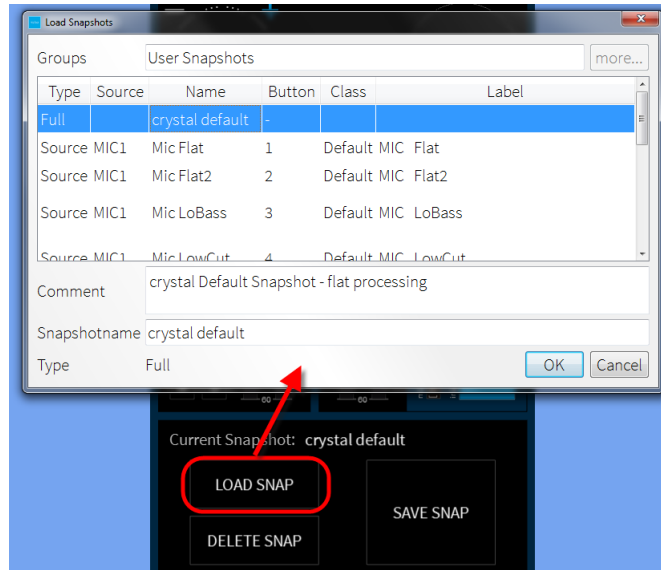
#### Other Recall Options

The following options, in the ON-AIR Designer configuration, affect how snapshot settings are recalled globally across the console. These are set in the "System -> Definition -> Parameter = Snapshots" branch of the 'Tree Definition'.

- **Load max. Fader No.** - specifies whether parameters are loaded for all faders or only up to a certain fader number. You can use this option to exclude fader strips from snapshot recall.
- **Load Audio Always** - if ticked, audio-related parameters are loaded for all faders regardless of the **Load max. Fader No.** You can use this option to load audio-related parameters to all faders, while loading fader strip assignments up to a certain fader number.
- **Enable Load Groups Logic** - can be used to change which source parameters load depending on fader status. By default, all parameters if source faders are closed, and do not load if source faders are open. By assigning control signals to the various load group inputs, you can modify this behaviour as you wish. See Selective Loading of Source Parameters.

### 5.15.2 Loading a VisTool Snapshot

1. Select **LOAD SNAP** - the "Load Snapshot" window appears:



This window lists all of the snapshots stored in the current user group. In the standard template, one group is available: **User Snapshots**.

If your system supports a [user login](#), then different snapshot groups may be available (via the **more..** button). See [User Access Rights](#) for more details.

In the **Type** column you can see that two types of snapshot are available:

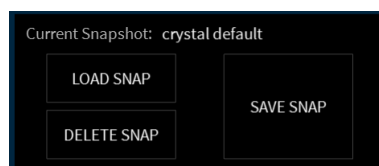
- **Full** snapshots store and recall settings globally across the console (according to the restrictions described [earlier](#)).
- **Source** snapshots store settings for a specific source.

Note that **Source** snapshots can be loaded either to the source in access OR to their original source (if nothing is in access). The original source is identified in the **Source** column.

2. Select a snapshot from the list (e.g. **crystal default**) and select **OK** - the window closes and the settings are loaded to the system.

The console surface resets accordingly and "Current Snapshot" field on the VisTool GUI updates:

*Standard Template (MAIN page)*

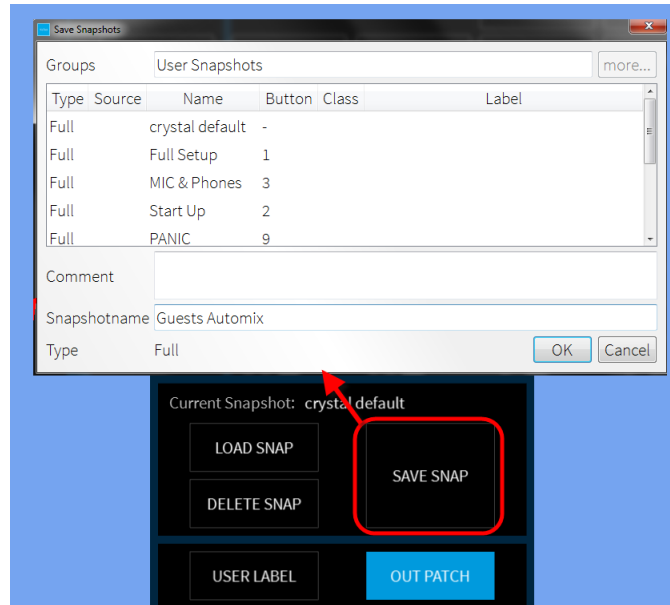


### 5.15.3 Saving a VisTool Snapshot

When you save snapshots, you can choose to save either a **Full** snapshot (to save settings globally across the console) or **Source** snapshot (to save only the settings for the source in access). In the standard template, the **SAVE SNAP** button is configured to support both operations.

➤ **To save a Full snapshot:**

1. Check that there is no source in access and select **SAVE SNAP** - the "Save Snapshot" window appears:



2. Enter a snapshot name (e.g. **Guests Automix**) and select **OK** - the window closes and the settings are saved.

Note that the snapshot **Group** and **Type** cannot be changed. You can enter text to describe the snapshot into the **Comment** field if you wish.

If your system supports a [user login](#), then different snapshot groups may be available (via the **more..** button). See [User Access Rights](#) for more details.

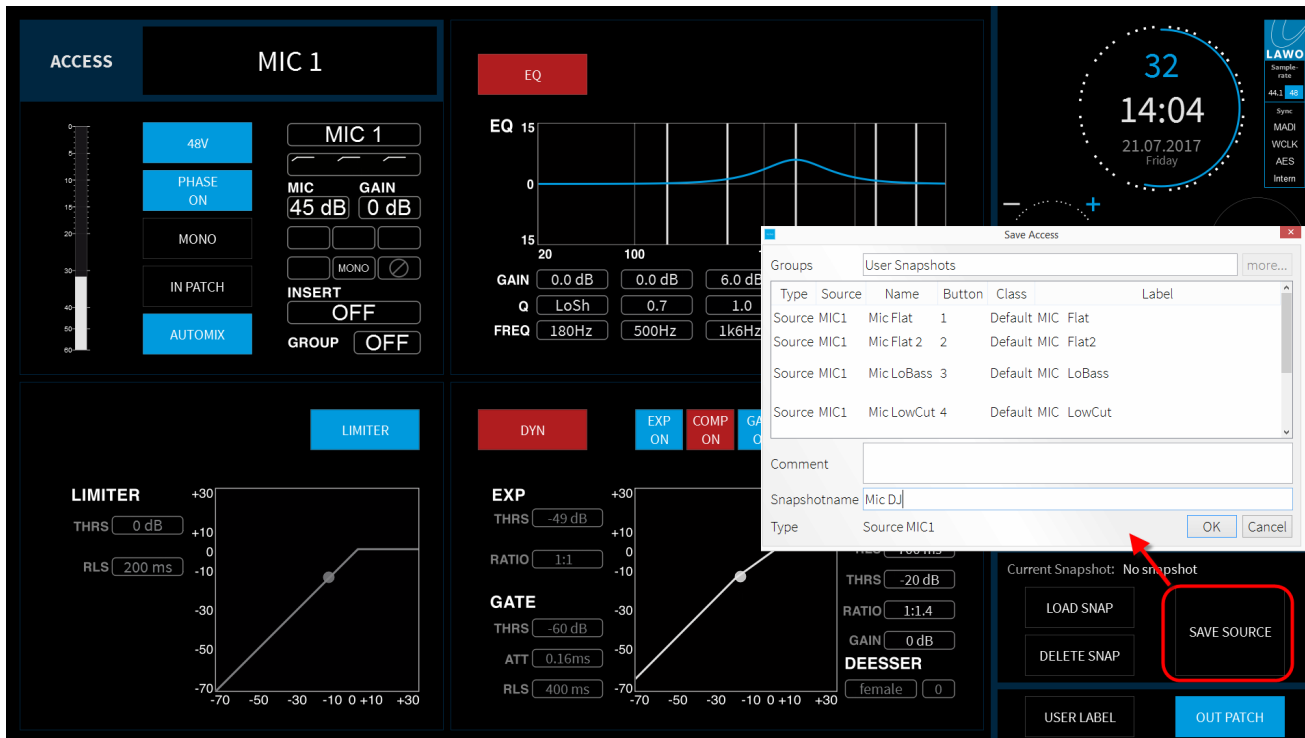
## 5. VisTool Runtime Operation

### ➤ To save a Source snapshot:

1. Press an **ACCESS** key (on the console) to select the source you wish to save.

The **ACCESS** key turns white and the [Source Parameter Control](#) page appears (on VisTool). Notice that the **SAVE SNAP** button has updated to **SAVE SOURCE**.

2. Select **SAVE SOURCE** - the "Save Access" window appears:



2. Enter a snapshot name (e.g. **Mic DJ**) and select **OK** - the window closes and the settings are saved.

Note that the snapshot **Group** and **Type** cannot be changed. You can enter text to describe the snapshot into the **Comment** field if you wish.

If your system supports a [user login](#), then different snapshot groups may be available (via the **more..** button). See [User Access Rights](#) for more details.

#### 5.15.4 Deleting a VisTool Snapshot

You can delete any snapshot, **Full** or **Source**, in a similar manner.

1. Select **DELETE SNAP** - the "Delete Snapshot" window appears.

This window lists all of the snapshots stored in the current user group. In the standard template, one group is available: **User Snapshots**.

If your system support a [user login](#), then different snapshot groups may be available (via the **more..** button). See [User Access Rights](#) for more details.

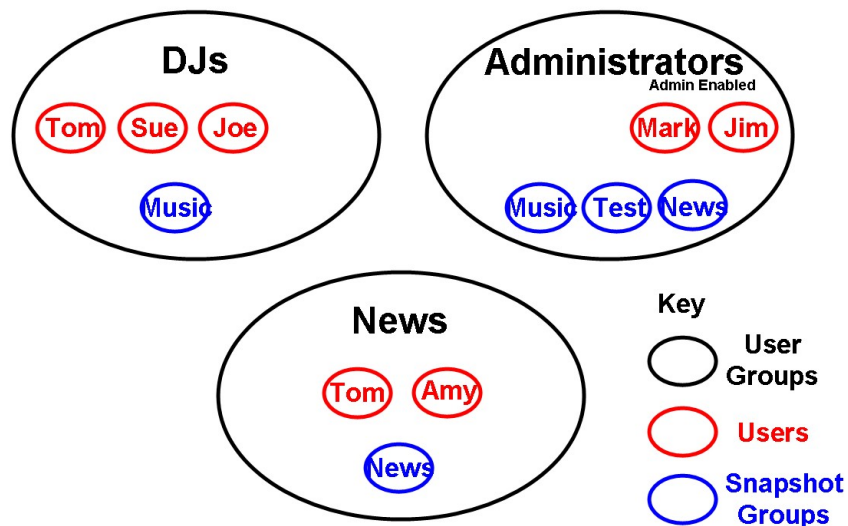
2. Select a snapshot from the list and select **OK** - the window closes and the snapshot is deleted.



## 5.16 User Access Rights

The VisTool MK2 User Access Rights can be used to provide different levels of snapshot access and functionality for administrators and "standard" users.

The system allows administrators to define user groups, users and snapshot groups. A user can belong to one or more user groups, and snapshot groups are available to all users within each group. When a user logs in, they will then have access to the snapshot groups defined for their user group. In the example below, the DJs have access to the "Music" snapshot group, while the Newsreaders have access to "News". Tom is a member of both the DJs and News user groups, and therefore can access both "Music" and "News" snapshot groups:



If you are a standard user, then your user rights will affect the following operations:

- Snapshots - which VisTool snapshot groups you can access, and whether you can save, load and/or delete snapshots.
- Other console operations - whether you can adjust DSP parameters, re-assign fader strips, etc.

Any user group can have administrator access but it makes sense to create a single user group called "Administrators". Administrators can create and edit the user groups, users and snapshot groups, and then determine the user rights for each user. The user rights settings are stored in the "Visconfigurations.mdb" file defined in VisTool Editor's "Global Settings" (via the [Snapshots](#) field).

### Configuring a User Login Button

To provide access to the system, a user **LOGIN** button must be configured. There are two possibilities: a button on a VisTool page or an MF Key (or other control signal) from the remote system.

For option one, add a [Box](#) element to the VisTool project, and link this to the "[Predefined User Text](#) -> Snapshot -> Login" option.

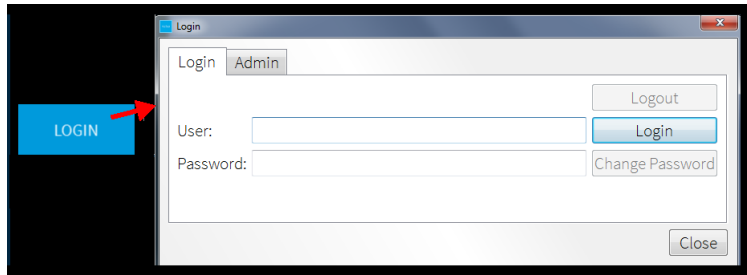
For option two, add a "Logic -> VispageSwitch" element to the ON-AIR Designer project, and define the [PC Login](#) function. It is important that the **Station ID** (defined for the "[Logic -> VispageSwitch](#)" element) matches the one entered in [VisTool Sessions](#). If not, the trigger from the remote system will not work.

Once the configuration is uploaded to the system, users can login via the **LOGIN** button. An administrator will need to prepare the system for other users via the [Admin](#) pages. Non-admin users can then login to access their permitted snapshot groups and operations.

### 5.16.1 User Login

If user access rights are supported by your configuration, then you will find a **LOGIN** button either on VisTool or the console surface.

1. Select **LOGIN** - the 'Login' window appears:



To login, you will need your user name and password, as defined by an administrator. If an administrator has [reset](#) your password, then it will be "**0000**".

If this is the first time anyone has logged in to the system, then the initial administrator user is called "**admin**" and the password is "**default**".

2. Enter your **User** name and **Password**, and then select **Login**.
3. Providing the credentials are accepted, you will see "Login successful". Select **OK** to close the 'Login' window.

If you are an administrator, you can now use the [Admin](#) pages to prepare or edit the user rights for other users.

If you are a standard user, then your user rights will affect the following operations:

- Snapshots - which VisTool snapshot groups you can access, and whether you can save, load and/or delete snapshots.
- Other console operations - whether you can adjust DSP parameters, re-assign fader strips, etc.

One or more snapshots can be set to load whenever a user logs in (via the [User Snapshot Properties](#)). This feature can be used to reset the console automatically.

### 5.16.2 Logout

To leave the console in a safe state, you should always log out:

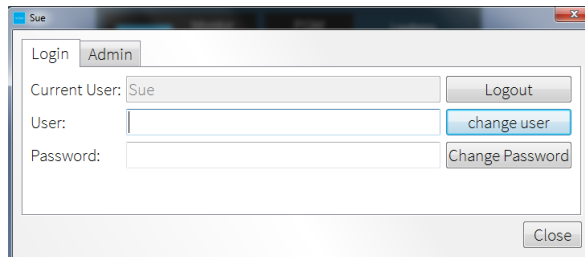
1. Re-open the [Login](#) window - the name of the **Current User** is displayed.
2. Select **Logout** - the console returns to its "logout" state.

The "logout" state is controlled by the Administrator settings for the user called "Nobody". i.e. whenever a user logs out, the "Nobody" user logs in. This allows the console to be reset to a default state and/or its operations restricted, by defining a login snapshot and restricted rights access for the user "Nobody". This is handled from the [Admin](#) pages in the usual manner.

### 5.16.3 Changing Your Password

Once you are logged in, you can change your password as follows:

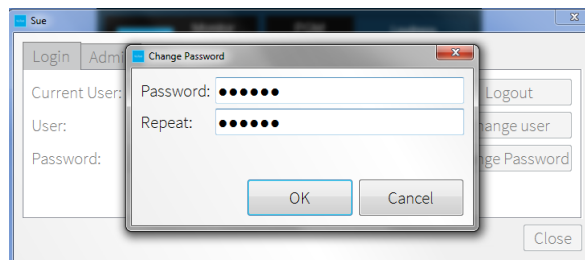
1. Re-open the [Login](#) window - you should see that you are logged in as the Current User:



You cannot change your password if you are not logged in!

If you have forgotten your password, then an Administrator can reset it to **0000**. Administrators, please see [Password Reset](#).

2. Select **Change Password**.
  3. Enter your new password into both the **Password** and **Repeat** fields, and select **OK**.
- There are no restrictions on the characters used for a password.



4. Test the new password by [logging out](#) and [logging in](#).

### 5.16.4 The Admin Pages

To edit the user access rights for other users, you must be logged in as an Administrator.

To login for the first time, enter the following details at the [Login](#) window:

- User name = "admin"
- Password = "default"

Once logged in, you can change the "admin" password at any time. See [Changing Your Password](#).

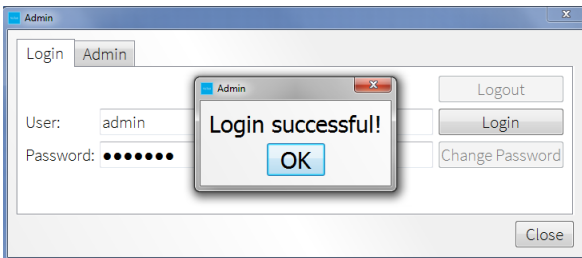
Once you have set up some users and user groups, any group can be given "administration" access (via the [Admin](#) option in the **Edit Group** page). This allows you to provide access to the Admin pages via specific user logins if you wish.

### 5.16.5 User Rights Example

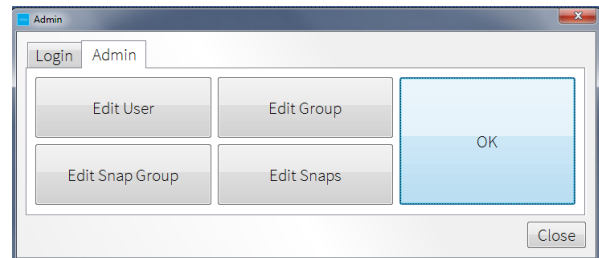
The example below describes a simple system with two users: "Admin" and "Nobody". The "Admin" user has full access to the snapshot group "User Snapshots" so that they can load, save and delete snapshots. The "Nobody" user has restricted access to the same snapshot group so that they can load, but not save or delete snapshots. This creates a system where "Nobody" users have full operational control of the console, but can only load pre-prepared snapshots (defined by the "Admin" user). Remember that the "Nobody" user becomes active whenever another user logs out, and so the "Nobody" restrictions apply automatically once the "Admin" logs out.

1. First make sure that you are [logged in](#) as an administrator, and select **Admin** to access the admin pages:

*Login Screen*



*Admin Pages (Home)*



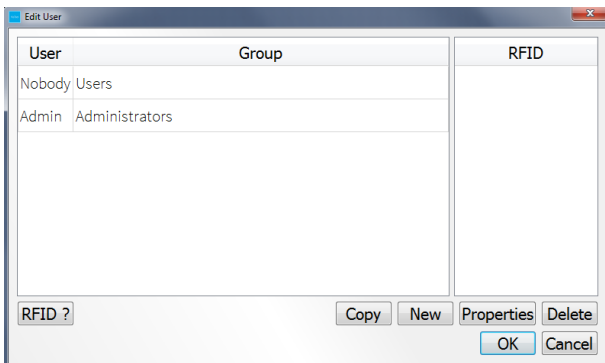
2. Then use the **Edit User**, **Edit Group** and **Edit Snap Group** buttons to check the setup of the users, user groups and snapshot group as follows.

You will need to create two user groups named **Users** and **Administrators**, where the **Users** group has a single user named "Nobody" and the **Administrators** group has a single user named "Admin".

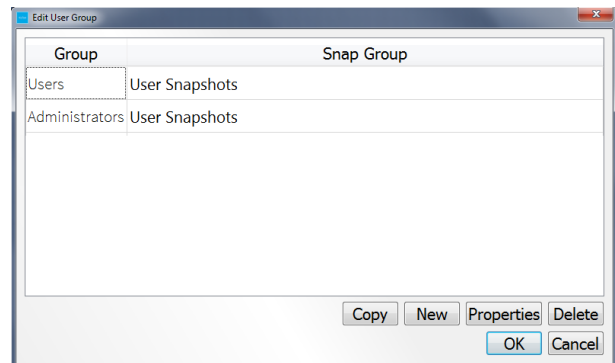
In addition, you will need to create a single snapshot group named **User Snapshots** which is assigned to both user groups: **Users** and **Administrators**.

The correct setup is shown below in the "Edit User" and "Edit Group" pages:

*Edit User*

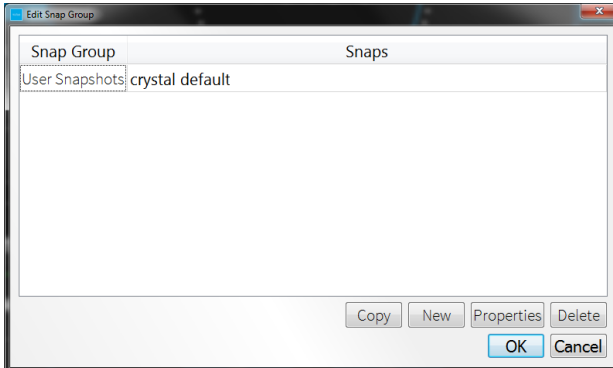


*Edit Group*

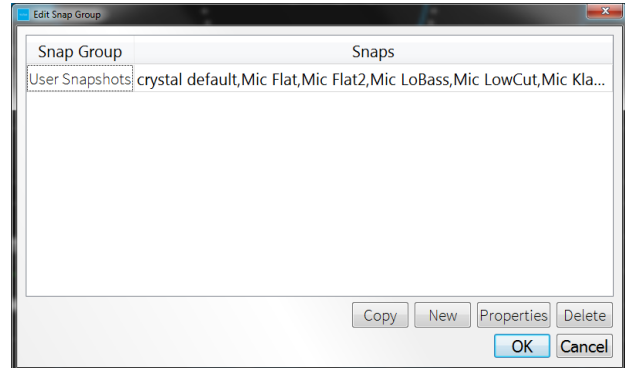


Now open the "Edit Snap Group" window - this will show all of the current snapshots stored in the **User Snapshots** folder. Following a first-time install, this should be empty (apart from any default snapshots loaded by the configuration). If the system is already in use and snapshots have been saved, then the saved snapshots will appear in the "Snaps" list.

*Edit Snap Group (fresh install)*



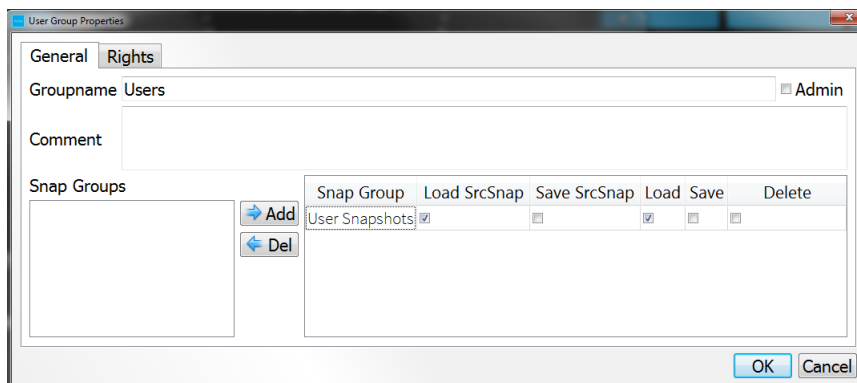
*Edit Snap Group (after snapshot saves)*



If any of the users, user groups or snapshot groups, then you will need to start by adding them and making the correct links. In each case, use the **New** button to add a new user, user group or snapshot group. Then select it and choose **Properties** to configure the linking options.

- Now check that the user access to the snapshot group is defined correctly, by re-selecting **Edit Group**. Select the group **Users**, and click on **Properties**.

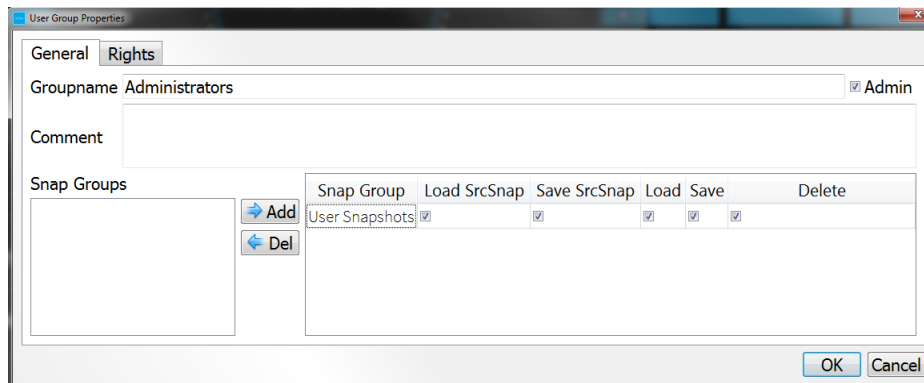
*Edit Snap Group - Properties for "Users"*



The Snap Group named "User Snapshots" should be assigned to the group, and only the **Load SrcSnap** and **Load** boxes should be ticked. Leave all other options unticked, so that any standard users will have load access only.

Then, repeat the steps for the **Admin** user group, but this time tick all of the **Load**, **Save** and **Delete** boxes AND tick the **Admin** option (beside the Groupname). This ensures that all admin users will be able to load, save and delete snapshots.

*Edit Snap Group - Properties for "Administrators"*



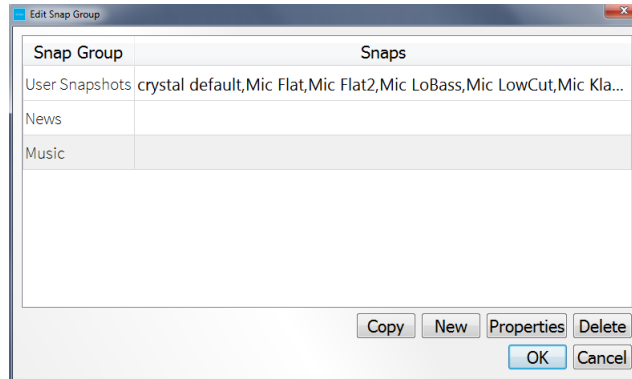
4. The setup is now complete. Log in as the "Admin" user to save snapshots into the "User Snapshots" snap group. Then logout to activate the "Nobody" user.

The rest of this topic describes each of the Admin pages in more detail.

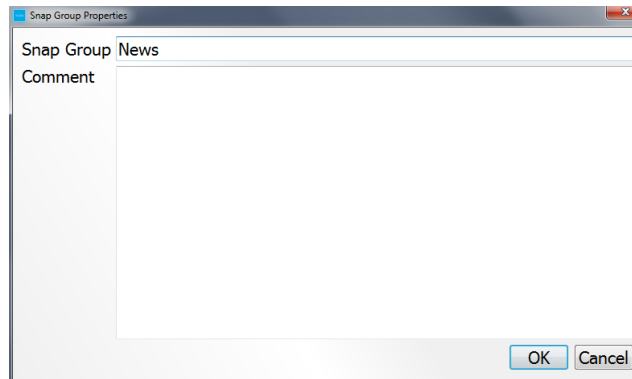
### 5.16.6 Admin -> Edit Snap Group

This page is used to create and rename snapshot groups.

- From the [Admin](#) Home page, select **Edit Snap Group** - the 'Edit Snap Group' window opens, with a list of snapshot groups on the left (**Snap Group**), and snapshots stored within each group on the right (**Snaps**):



- To create a new snapshot group, select **New**. Or, to edit an existing group, select a name and then **Properties** - the 'Snap Group Properties' window appears:



- **Snap Group** - enter a name for the snapshot group, e.g. **News**.
  - **Comment** - you can enter comments if you wish.
- When you have finished editing, select **OK** - the 'Edit Snap Group' window updates to show the new snapshot group name.
  - Continue selecting **OK** to confirm and close all the Admin windows.

For users to access the new snapshot group, you must add it to a user group. All members of the user group will then be able to access the snapshot group, according to the [snapshot group access](#) options.



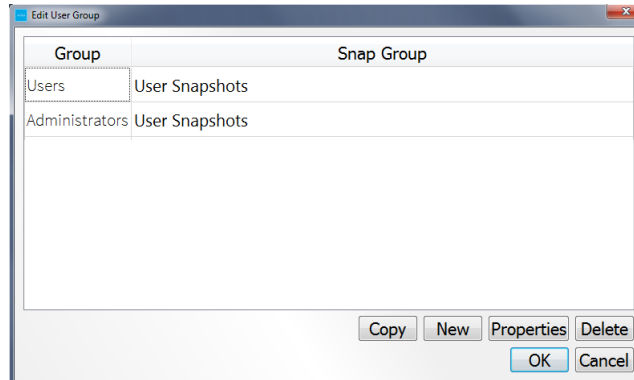
### 5.16.7 Admin -> Edit Group

This page is used to create and edit user groups. You can define:

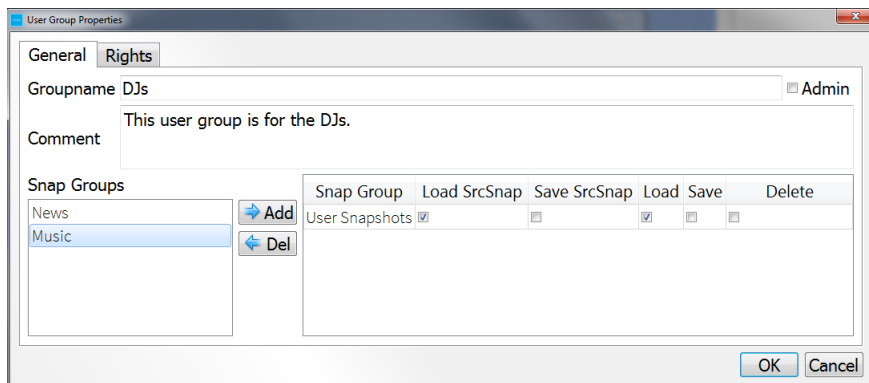
- which snapshot groups can be accessed, and whether users can load, save and delete snapshots.
- whether the user group has Admin page access.
- which console operations ("rights") are available to users within the group.

#### ➤ To create or edit a User Group

1. From the [Admin](#) Home page, select **Edit Group** - the 'Edit User Group' window appears, with a list of user groups on the left (**Group**), and the snapshot groups they may access on the right (**Snap Group**):



2. To create a new user group, select **New**. Or, to edit an existing group, select its name and then **Properties**) - the 'User Group Properties' window appears:

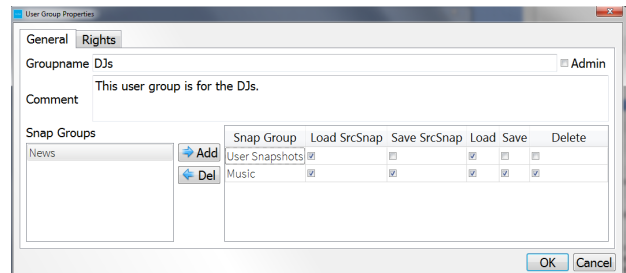
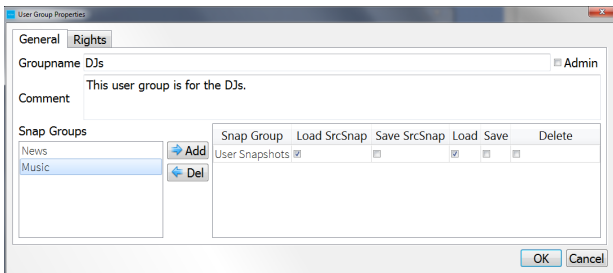


3. Select the **General** tab to edit the:
  - **Groupname** - enter a name for the user group, e.g. **DJs**.
  - **Comment** - you can enter comments if you wish.
  - **Admin** - tick this box to enable Admin access. This allows all users within the group to open and edit the Admin pages.
  - **Snap Groups** - this area defines the snapshot groups which users can access (see next page). On the left are all snapshot groups defined in the VisTool snapshots database. On the right are the snapshot groups which are available to the user group.

### ➤ To add Snapshot Groups to the User Group

1. Select a snapshot group from the list on the left - e.g. **Music** - and then **Add**.

**Music** moves to the right. This will make it available to all users within the **DJs** user group:



2. Select which snapshot operations are allowed for **TomsSnaps** by ticking, or unticking, the relevant boxes:

- **Load SrcSnap** - users can load source snapshots.
- **Save SrcSnap** - users can save source snapshots.
- **Load** - users can load full snapshots.
- **Save** - users can save full snapshots.
- **Delete** - users can delete snapshots.

3. Repeat to add more snapshot groups to the user group.

4. Or, use **Delete** to remove a snapshot group from the user group's access.

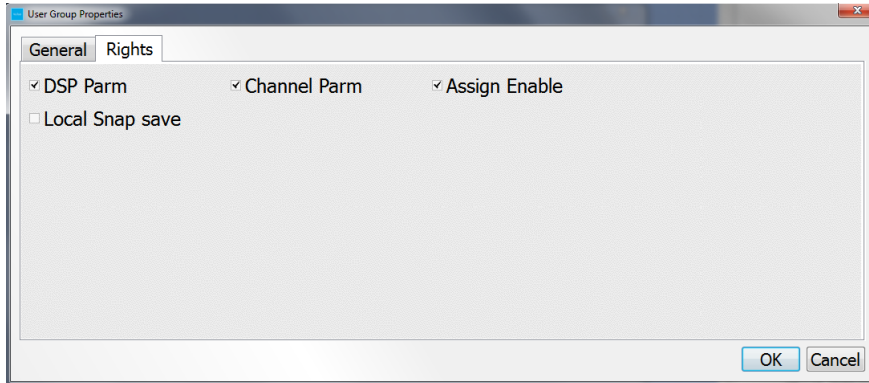
By creating multiple snapshot groups, and defining different snapshot operations for each user group, you can control which users can save and delete important snapshots.

The **Snap Groups** list shows all snapshot groups defined in the VisTool snapshots database. If you have not yet created any snapshot groups, then this list will be empty. Go to [Edit Snap Group](#) to create some snapshot groups.

### ➤ User Group Rights

1. Select the **Rights** tab to edit the user group rights.

What you see in this window depends on your system configuration. Up to 32 "rights" may be named and defined by the configuration (see the [Logic -> VispageSwitch](#) branch of the 'Tree Definition'). Our example includes 4 "rights":



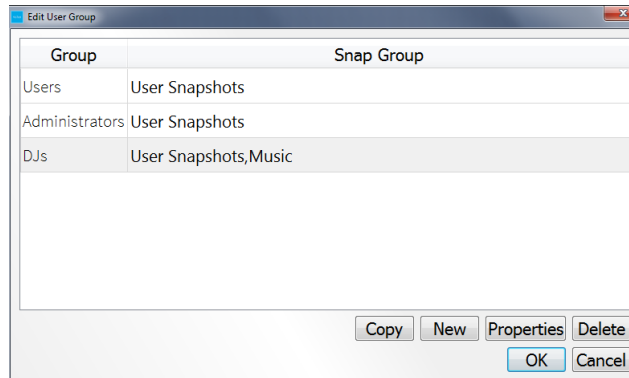
Tick, or untick, the boxes to activate each option - in our example:

- **DSP Parm** - source parameter control.
- **Channel Parm** - channel parameter control.
- **Assign Enable** - fader strip assignment.
- **Local Snap save** - the ability to save (overwrite) the 5 local snapshot memories.

In our example, all users within the **DJs** user group will be able to control source parameters, channel parameters and assign fader strips. They will not be able to save settings to the 5 local snapshot memories.

### ➤ To close and save your changes

1. When you have finished editing both the **General** properties and the user group **Rights**, select **OK** - the 'Edit User Group' window updates to show the new user group - **DJs** - and its available snapshot groups:



2. Continue selecting **OK** to confirm and close all the Admin windows.

The following operations are available from the 'Edit User Group' window:

- **Copy** - select a user group from the list plus **Copy** to make a copy of an existing user group. All **General** properties and **Rights** are copied into the new user group name.
- **Delete** - select a user group and **Delete** to delete an existing user group.

The **Delete** user group operation cannot be undone, so take care when using this function.

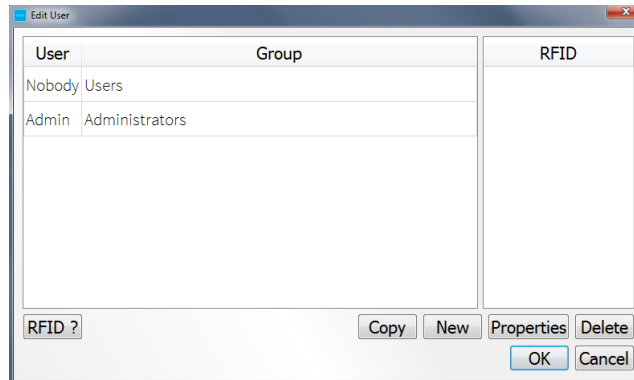
### 5.16.8 Admin -> Edit User

This option is used to create and edit users. You can define:

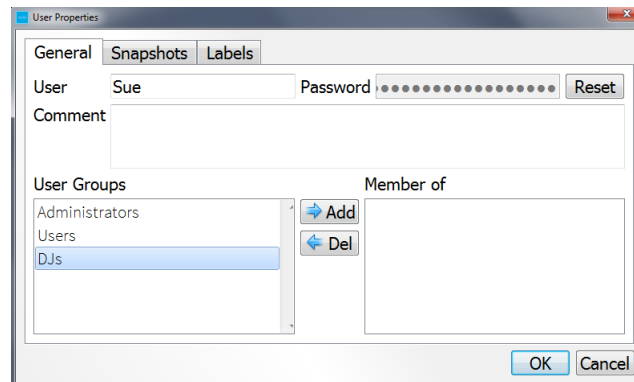
- which user groups they belong to (and therefore which snapshot groups they can access).
- their user name and password.
- their default snapshot group (opens when they choose to load or save snapshots).
- their login snapshots (these are snapshots which load automatically after the user logs in).

#### ➤ To create or edit a User

1. From the [Admin](#) Home page, select **Edit User** - the 'Edit User' window appears, with a list of users on the left (User), and the user groups they belong to on the right (Group):



2. To create a new user, select **New**. Or, to edit an existing user, select a name and then **Properties** - the 'User Properties' window appears:



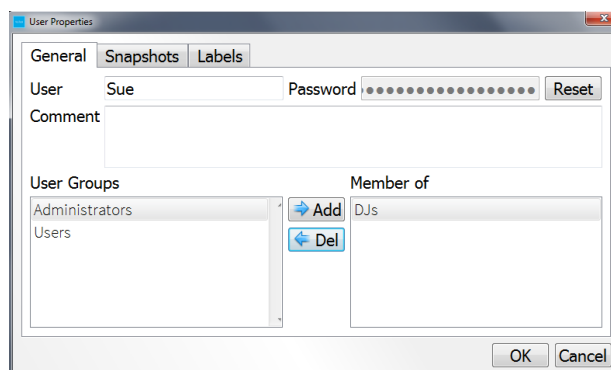
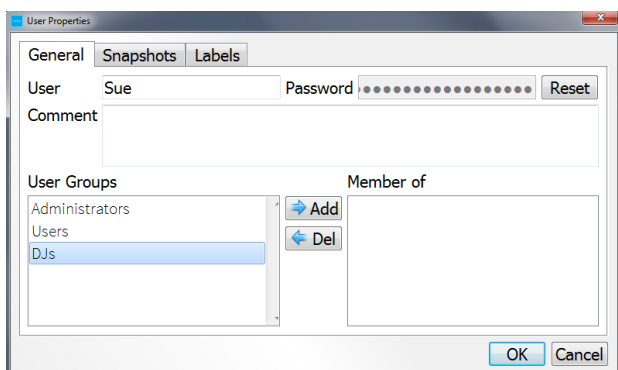
3. Select the **General** tab to edit the:
  - **User** - enter a name for the user, e.g. **sue**.
  - **Password** - enter a password for the user (the default is **0000**). This can be changed by the user once they login, see [Changing Your Password](#).
  - **Reset** - resets the user's password to **0000**. (Very useful if a user forgets their password).
  - **Comment** - you can enter comments if you wish.
  - **User Groups** and **Member of** - defines which user groups the user belongs to (see next page).

## 5. VisTool Runtime Operation

### ➤ To add the User to a User Group

1. Select a user group from the list on the left - e.g. **DJs** - and select **Add**.

**DJs** moves to the **Member of** list on the right; Sue is now a member of the **DJs** user group:

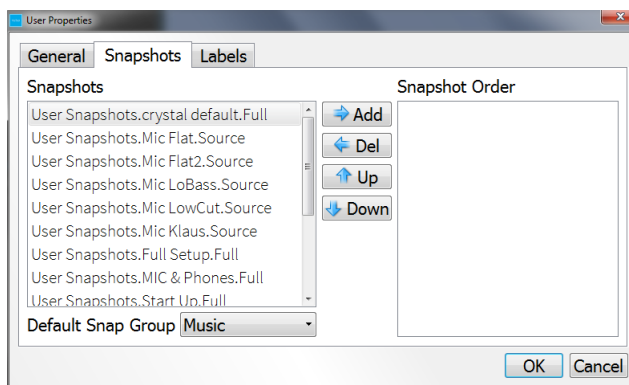


2. Repeat to add the user to more user groups or, use **Del** to remove a user group.

Each user can belong to several user groups. In this instance the user will see all [snapshot groups](#) assigned to the user groups, and access their [user group rights](#).

### ➤ Snapshot Properties

1. Select the **Snapshots** tab to edit the default snapshot group, and order of any login snapshots, for the user:



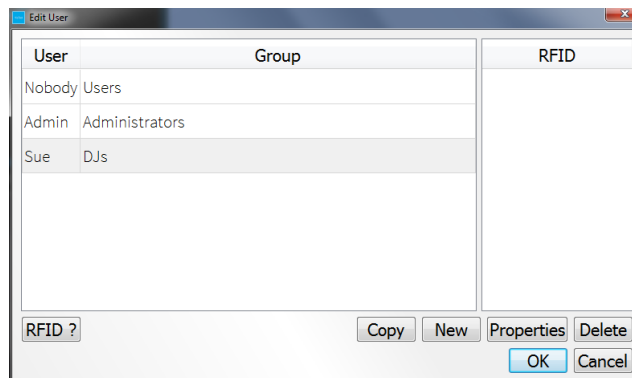
- **Default Snap Group** - select a snapshot group from the drop-down menu - in our example, **Music**. This becomes the group which is automatically selected whenever the user attempts to [Save](#), [Load](#) or [Delete](#) a VisTool snapshot.
  - **Snapshot Order** - snapshots entered in this list are loaded immediately after the user logs in. This can be used to automatically reset the console for the user's show, or type of work, whenever they login to the console.
2. To configure the console to load snapshots after login, use the **Add** and **Del** buttons to move entries from the **Snapshots** list (on the left) to the **Snapshot Order** list (on the right).

The **Snapshots** list (on the left) shows everything available in the user's snapshot groups. Remember that the available snapshot groups are defined by [editing the user group](#).

3. Use the **Up** and **Down** buttons to change the order in the **Snapshot Order** list (on the right). Snapshots load from top to bottom after the user successfully logs in.

➤ **To close and save your changes**

1. When you have finished editing both the **General** and **Snapshots** properties, select **OK**. The 'Edit User' window updates to show the new user - **Sue** - and their user groups - **DJs**:



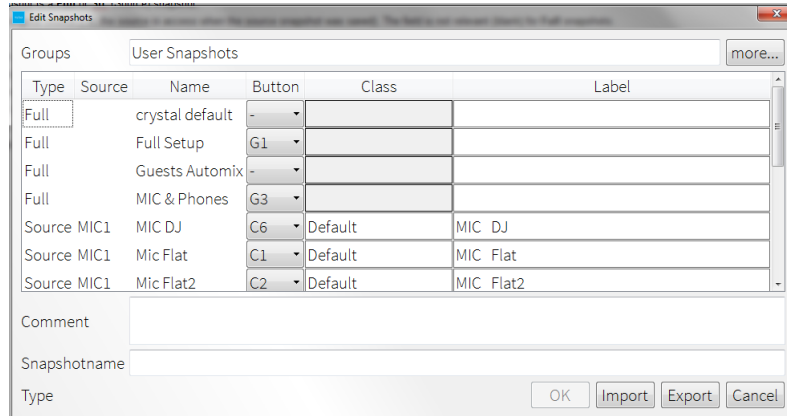
2. Continue selecting **OK** to confirm and close all the Admin windows.

### 5.16.9 Admin -> Edit Snaps

This option is used to:

- Assign snapshots to the Global and Channel Snapshot buttons (defined using a "[Logic -> VispageSwitch](#)" element in the ON-AIR Designer configuration).
- **Import** or **Export** snapshots as .xml files.

1. From the [Admin](#) Home page, select **Edit Snaps** - the 'Edit Snapshots' window appears, listing all the snapshots in the selected [Snapshot Group](#) (e.g. **Snap-Admin**):



Most of the fields are for information only:

- **Type** - indicates whether the snapshot is a **Full** or **Source** snapshot.
- **Source** - shows the name of the original source (i.e. the source in access when the source snapshot was saved). The field is not relevant (blank) for **Full** snapshots.
- **Name** - the snapshot name.

2. Use the **Button** column to assign an entry to one of the 9 Global Snapshot (**G1-G9**) or 9 Channel Snapshot (**C1-C9**) buttons.

These allow users to recall snapshots directly from dedicated buttons (defined using a VisTool Editor [Box](#) element linked to the **Global** or **Channel** snapshot fields in the "[Logic -> VispageSwitch](#)" element).

Note that only **Full** snapshots can be assigned to Global buttons; and **Source-** snapshots to Channel buttons.

3. Use the **Label** field to edit the snapshot name which appears in VisTool. You may enter a **Comment** if you wish.
4. Select **Import** or **Export** to save or load snapshots from/to an .xml file. This allows you to backup or transfer VisTool snapshots between computers.
5. Select a different snapshot group using the **more..** button.
6. When you have finished editing both the **Snapshot** properties, select **OK** to save any changes and return to the Admin page options.

## 6. Editing a VisTool Configuration

---

This chapter describes how to create and edit a custom configuration using VisTool Editor.

Topics include:

- [Introduction](#)
- [Starting the Application](#)
- [Global Settings](#)
- [File -> Open](#)
- [File -> New](#)
- [Project Properties](#)
- [Saving the Project](#)
- [Working with VisTool Editor](#)
- [Arranging the Workspace](#)
- [Creating and Managing Pages](#)
- [Configuring Page Switching](#)
- [Configuring Elements](#)
- [Positioning Elements](#)
- [Monitoring the Resources](#)
- [Using the Project Tree](#)
- [Previewing and Testing the Design](#)
- [Using Color Maps](#)
- [Next Steps](#)
- [The Main Menu](#)
- [Keyboard Shortcuts](#)



### 6.1 Introduction

**VisTool Editor** can be used to design and build custom configurations for use during runtime.

Each project can contain multiple pages containing different elements. The Editor can assign functions to elements and configure their position, size and appearance on the page. The software uses vector graphics allowing elements to be scaled to fit the design. You can even include your own custom graphics and web elements.

Once a project has been saved, you can preview the design using VisTool Editor. Then use [VisTool Sessions](#) to determine how the pages will be displayed on the VisTool runtime screen(s).

The VisTool Editor license is free of charge and so you can prepare and test a project on any Windows PC. However, to operate a custom configuration during runtime, you will need to purchase a license like **VisTool Solo**, **VisTool Solo Plus** or **VisTool Unlimited**.

#### 6.1.1 Control Methods

During runtime, most of the GUI elements have no actual function of their own, but remotely control parameters within a networked device. There are two possible control methods: DMS and Ember+.

The **DMS** (Digital Metering Service) is a Lawo control protocol supported by all radio products. For a VisTool element to control functions within these systems, the function must be supported by the product's software release and defined in the system configuration (using ON-AIR Designer). During runtime, VisTool can control up to four DMS devices simultaneously (one per session). Each session combines data from a VisTool project (**.vistool**) and ON-AIR Designer project (**.db3**). This allows VisTool Editor to assign functions to elements even when the DMS system is offline.

**Ember+** is a non-proprietary TCP/IP protocol supported by a wider range of products. For a VisTool element to control functions within the networked system, parameters must first be published by the Ember+ providing device. An Ember+ Provider Tree, listing all available parameters, can be found in the User Manual for your console/router/application. During runtime, VisTool can control multiple Ember+ devices. These are specified by the VisTool project (via the [Ember Devices](#) list). Note that VisTool Editor requires a valid Ember+ connection to detect the parameters published by the providing device. If you wish, you can save an offline copy of the Provider Tree. This allows VisTool Editor to still access the provider's parameters even if the device goes offline.

Some parameters are only available via DMS.

To support Ember+ control you will need a **VisTool Unlimited** license plus the **Ember+/Logic AddOn**.

To control only Ember+ devices and no DMS unit, you must [disable](#) the DMS session parameters.

#### 6.1.2 File Naming & Organization

VisTool Editor creates a ".vistool" file when you save the VisTool project. It is a good idea to keep the VisTool and ON-AIR Designer projects together, and name the files clearly. For example:

- **ruby\_Studio1\_configuration.db3** - created by the ON-AIR Designer.
- **vis\_ruby\_Studio1.vistool** - created by VisTool Editor.

Once the projects are complete, you will need to generate a start script using [VisTool Sessions](#). You should do this on the runtime PC so that the correct monitor options can be configured. If this is a different PC, then copy all of the relevant files: you will need both the VisTool Editor project (**.vistool** file) and the accompanying ON-AIR Designer project (**.db3** file) to generate a start script.

### 6.1.3 Using the Logic Engine

The [Logic Engine](#) can be used to configure logical outcomes for a range of VisTool elements (both DMS and Ember+). Various building blocks are available, including Any to String, Boolean AND, Boolean NOT, Boolean OR, etc.

Note that the Logic Engine editor opens in its own operating window. However, all settings are saved as part of the VisTool project in the usual manner.

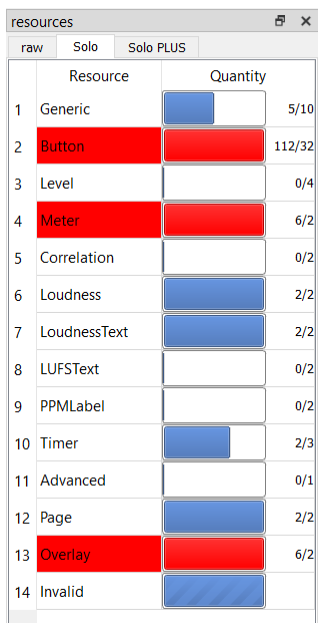
To support functions programmed using the Logic Engine, you will need a **VisTool Unlimited** license plus the **Ember+/Logic AddOn**.

### 6.1.4 Designing a Project for VisTool Solo or Solo Plus

If you are designing a project for **VisTool Solo** or **Solo Plus**, it is important that the project does not exceed the permitted resources. Otherwise the session will not run properly on the runtime PC (unless you upgrade the license).

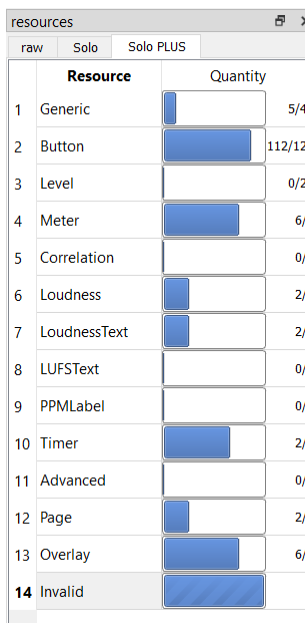
To ensure you remain within the limits, open the [Resources](#) docking panel and select the correct tab: either **Solo** or **Solo Plus**.

*VisTool Solo*



Resource	Quantity
1 Generic	5/10
2 Button	112/32
3 Level	0/4
4 Meter	6/2
5 Correlation	0/2
6 Loudness	2/2
7 LoudnessText	2/2
8 LUFSText	0/2
9 PPMLLabel	0/2
10 Timer	2/3
11 Advanced	0/1
12 Page	2/2
13 Overlay	6/2
14 Invalid	

*VisTool Solo Plus*



Resource	Quantity
1 Generic	5/40
2 Button	112/128
3 Level	0/24
4 Meter	6/8
5 Correlation	0/8
6 Loudness	2/8
7 LoudnessText	2/8
8 LUFSText	0/8
9 PPMLLabel	0/8
10 Timer	2/3
11 Advanced	0/4
12 Page	2/8
13 Overlay	6/8
14 Invalid	

Each line is color-coded to indicate: blue = ok; red = limit exceeded. The counters on the right show how many resources are left out of the maximum permitted. Or, if a line is red, by how many the maximum is exceeded.

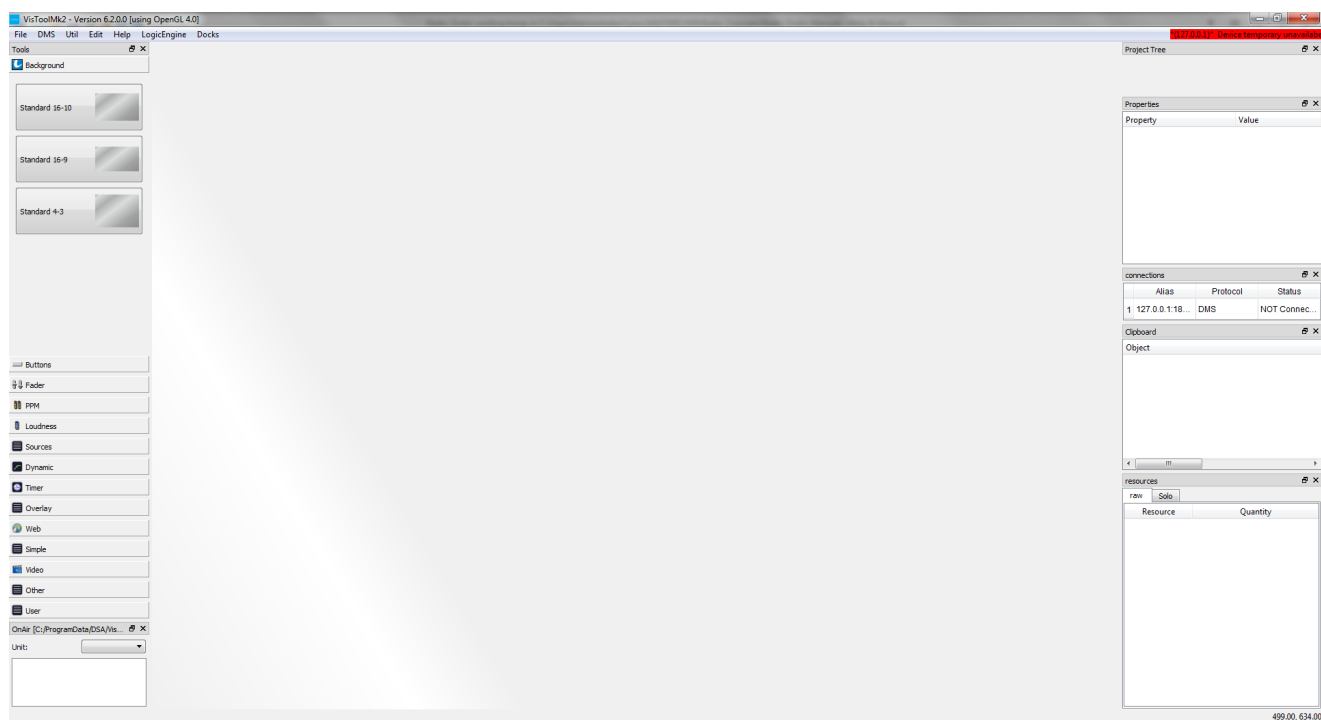
If a limit is exceeded, then you will need to delete the element(s), or purchase a higher license. In the example above, the project can be used with a VisTool Solo Plus license, but not VisTool Solo.

There are no restrictions regarding the setting up or editing of elements, as long as the maximum limits are observed.

### 6.2 Starting the Application

1. Start the program by selecting **VisToolMK2 Editor** from the Windows "Start" menu.

The software opens with a blank operating window:

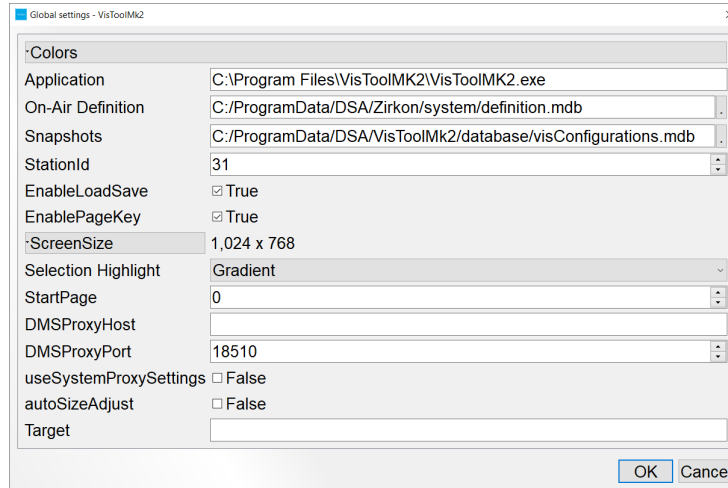


At the top of the page are the main menus; in the middle the main design area; and on the left and right are the docking panels which you will use to add and configure elements within the design.

### 6.3 Global Settings

When starting VisTool Editor for the first time, you should check the 'Global Settings'. These settings affect the operation of the Editor and are stored on the VisTool PC when you close the application.

1. Select **File** -> **Global Settings** from the top menu bar to open the window:



You must be logged in to Windows as an Administrator to edit the 'Global Settings'. If not, then the fields will be greyed out. Close VisTool Editor and re-open it as an Administrator to edit the settings. You can do this by right-clicking on the **VisToolMK2 Editor** icon and selecting "Run as administrator".

2. If necessary, edit the settings as described below.
3. Select **OK** to confirm any changes and exit the **Global Settings** window.

#### Colors

Click to open selectors for the default VisTool Editor color mapping. You may define the RGB and Alpha values for each "color": Black, Red, Green, etc. These are stored locally on your workstation. Note that additional color maps may be created and saved with the project, see [Color Maps](#).

#### Application

This field displays the location of the "VisToolMK2.exe" application. It is included for information purposes only; you cannot edit the file path.

#### On-Air Definition

This field defines the location of the "definition.mdb" file which is required to support the ON-AIR Designer project.

After a standard installation of the ON-AIR Designer software, the file can be found at:

C:\ProgramData\DSA\Zirkon\system\definition.mdb

Usually the default file path is fine, but if you need to change it, click on the ... button and choose the correct location.

If the **On-Air Definition** file path is not specified correctly, then an error message appears when you close the **Global Settings** window.

### Snapshots

This field defines the location of the "Visconfigurations.mdb" file which is used to store [snapshots](#) and [user rights](#) created by the VisTool runtime.

After a standard installation of the VisTool MK2 software, the file can be found at:

C:\ProgramData\DSA\VisToolMK2\database\Visconfigurations.mdb

Usually the default file path is used so that data can be stored independently on each VisTool PC.

Alternatively, if you wish to share snapshots between VisTools, then you can create a shared file location. First make a copy of the "Visconfigurations.mdb" file and paste it onto the shared server. The default file path is given above. In Windows, the default file path may be hidden from view, so you may need to [enable](#) the "show hidden files and folders" option.) Then point all of the VisTool PCs to the same file, by clicking on the ... button and choosing the new location.

If the **Visconfigurations.mdb** file path is not specified correctly, then VisTool cannot save or load snapshots, or access the user rights data.

### Station ID

This field defines the default **Station Id** used in the "Session Management" window. See [Station Id](#) for more details.

### EnableLoadSave

Set this option to **True** (ticked) to enable the load, save and delete snapshot functions. Once enabled, VisTool snapshots and user access rights can be stored in the "Visconfigurations.mdb" file specified in the **Snapshots** field.

Control signals to action the load, save or delete can be defined using a "[Logic -> VispageSwitch](#)" element (in the ON-AIR Designer "Tree Definition").

### EnablePageKey

Set this option to **True** (ticked) to allow VisTool pages to be switched by a control signal.

Control signals to action the page switching can be defined using a "[Logic -> VispageSwitch](#)" element (in the ON-AIR Designer "Tree Definition").

**Screen Size** - click here to define the screen size of the VisTool Editor window.

**Selection Highlight** - click to here to define how selected objects within VisTool Editor are highlighted.

### Start Page

This field specifies the [page index](#) number which VisTool will open when it first starts in runtime.

Note that this can be overwritten if automatic page switching is configured (using the ON-AIR Designer's "Logic -> VispageSwitch" element). This is described [later](#).

### DMSProxyHost, DMSProxyPort & useSystemProxySettings

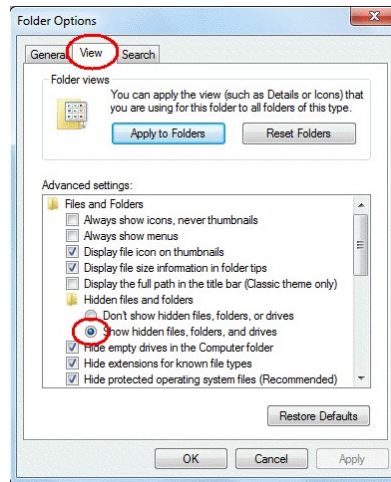
These fields specify whether your computer is running a local DMS service (**localhost**) or DMS proxy (in which case you will see the IP Address and Port of the DMS host.) If **useSystemProxySettings** is unticked, then the settings are automatically taken from **VisAdmin** (see [DMS Service Administration](#) and [DMS Service Host](#)). Tick the **useSystemProxySettings** option to use the system proxy settings (defined in Windows).

**autoSizeAdjust** - set this option to **True** (ticked) to automatically adjust the size of the VisTool Editor screen so that the Windows taskbar remains visible. When deselected, VisTool Editor will fill the complete screen (in full-screen view).

**Target** - in the latest release, this field has no function.

### 6.3.1 Revealing Hidden Files and Folders (in Windows)

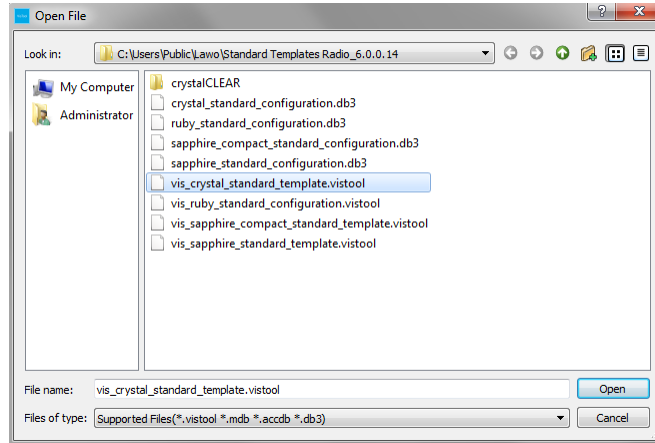
In Windows, the file path "C:\ProgramData..." may be hidden from view. In this instance, enable the "show hidden files and folders" option:



### 6.4 File -> Open

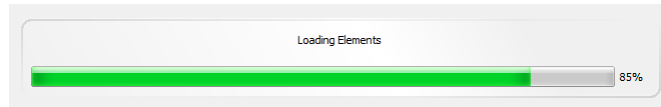
The easiest way to get started with VisTool Editor is to open an existing project.

1. Select **File -> Open** from the main menus.
2. Navigate to the folder where the configuration files are located and select a VisTool project file (**.vistool**) followed by **Open**:



You can also use **File -> recently used** to open a recently used VisTool project file.

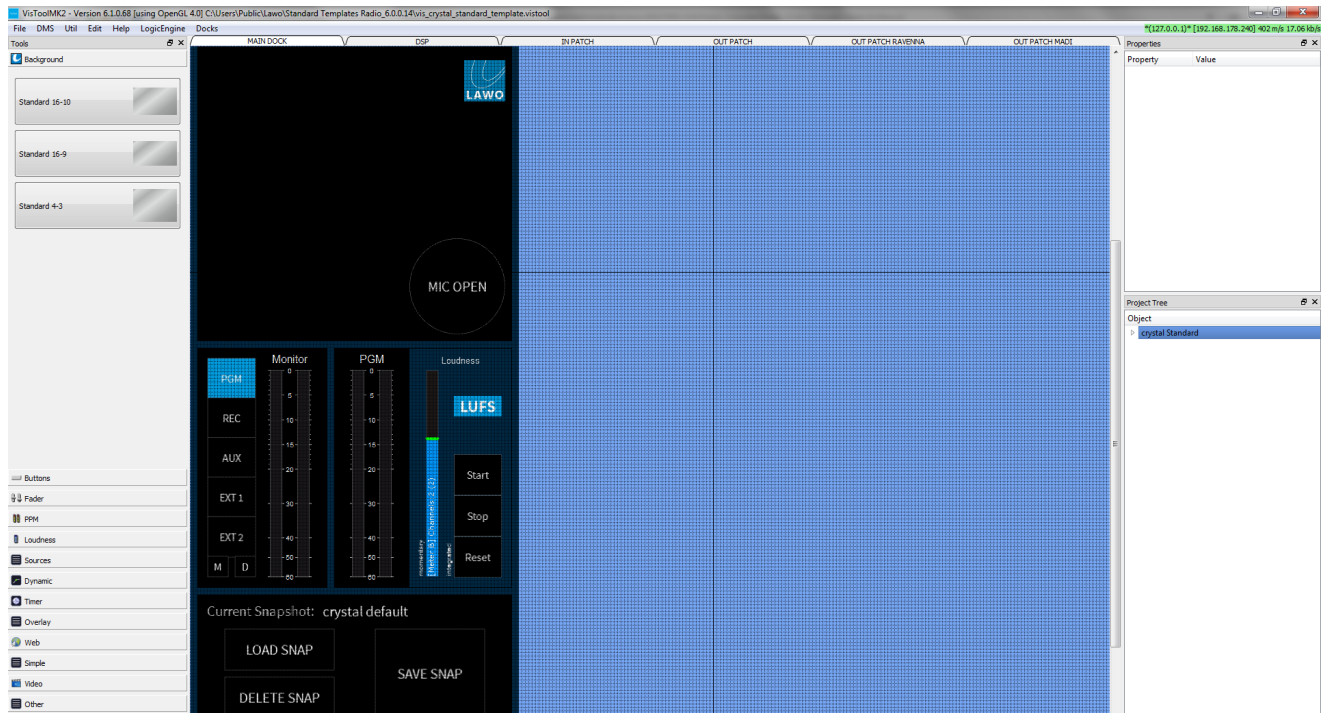
If the project has been opened before, then a link to the ON-AIR Designer project will have been made, and the file opens immediately - you will see the message "Loading Elements" in the middle of the screen:



Alternatively, if a link has not yet been made, then you will be asked to specify one. Select **Yes** to open a 'File Explorer' window, and choose the ON-AIR Designer project file (**.db3**) which will support the VisTool project.

For Lawo's radio on-air products, it is the ON-AIR Designer project which determines what functions can be linked to VisTool elements. Therefore, take care to choose the correct ON-AIR Designer project file!

Once the VisTool project has opened successfully, the Editor's [operating window](#) appears. In the main design area you will see the contents of the first page:



Only one project can be opened at a time (per VisTool instance).



## 6. Editing a VisTool Configuration

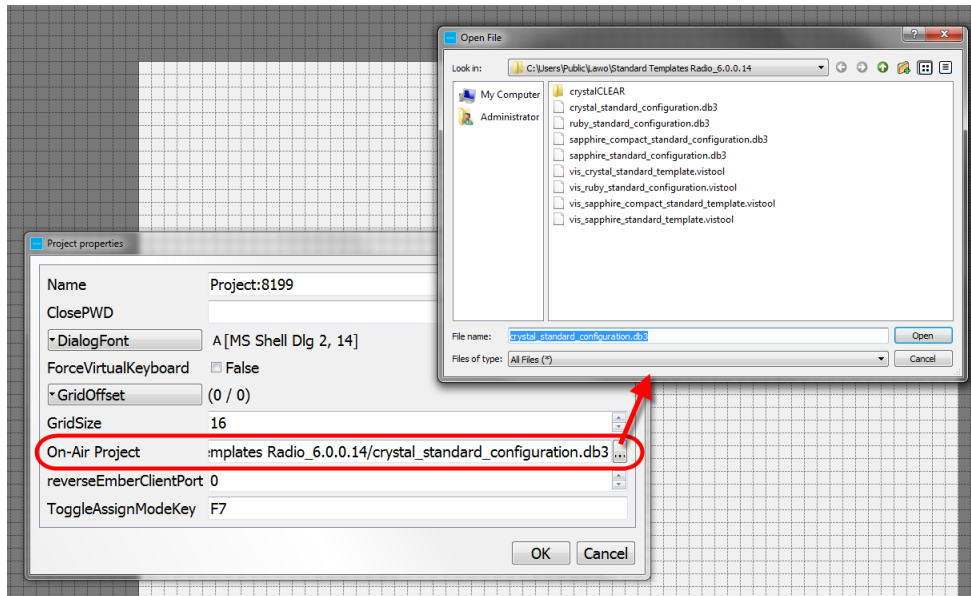
### 6.5 File -> New

To create a new VisTool project:

1. Select **File -> New** from the main menus - the 'Project Properties' window appears.

The 'Project Properties' are described in more detail [later](#). For now, continue as follows:

2. Click on the ... button beside the **On-Air Project** field, and select the ON-AIR Designer project file (.db3) which will support the VisTool project:

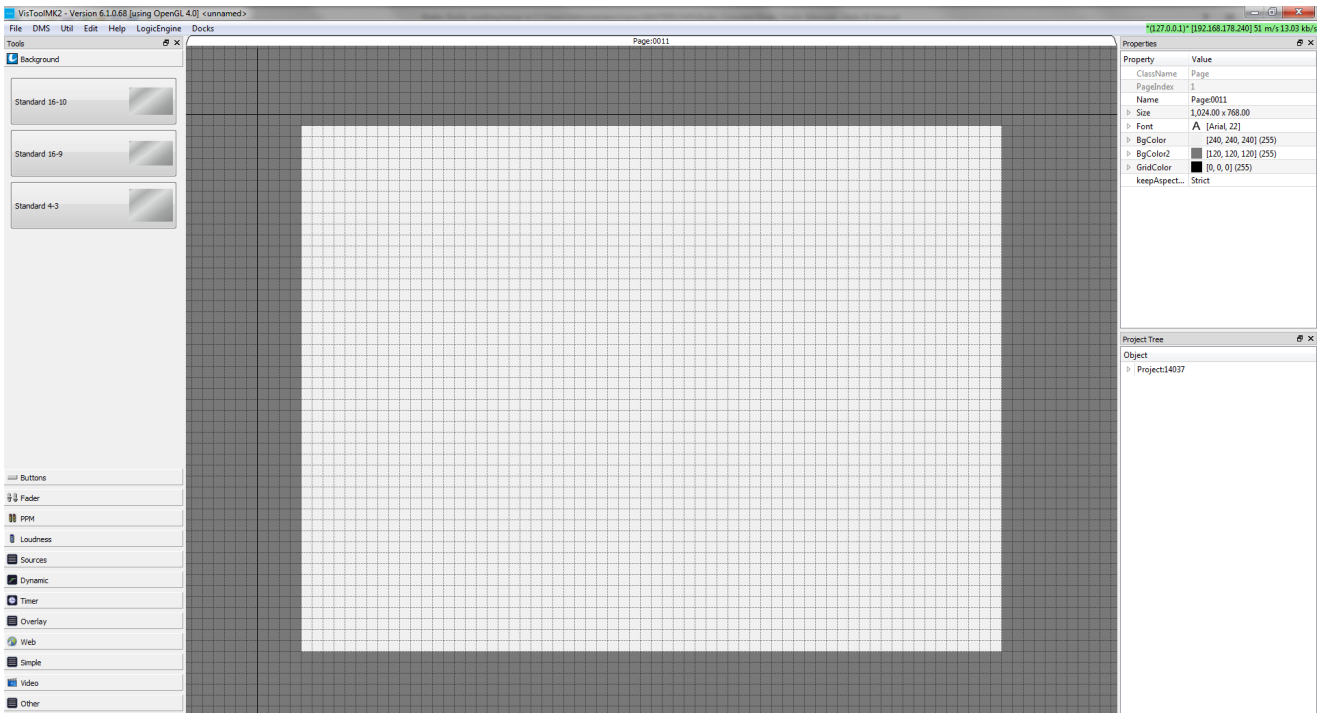


For Lawo's radio on-air products, it is the ON-AIR Designer project which determines what functions can be linked to VisTool elements. Therefore, take care to choose the correct ON-AIR Designer project file!

If you do not have a copy of the ON-AIR Designer project file on your computer, then please ask your System Administrator.

3. Select **Open** to confirm the selection, and then **OK** to save and close the 'Project Properties' window.

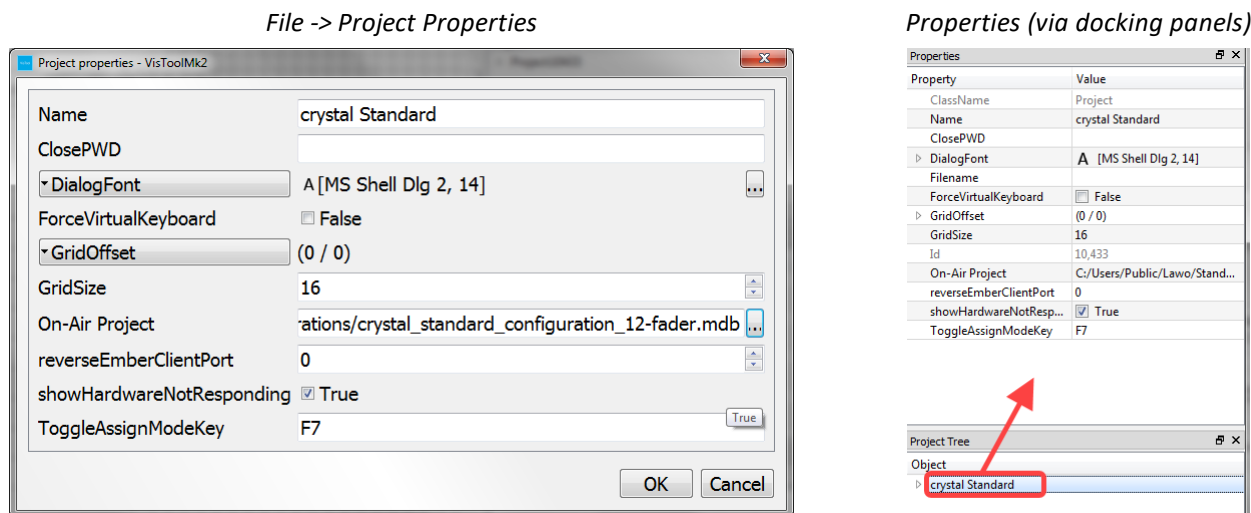
A new project is created with a single blank page:



### 6.6 Project Properties

Now is a good time to check the rest of the 'Project Properties'. These settings affect the operation of the project and are stored whenever you [save](#) the project.

1. Select **File -> Project Properties** from the main menus. Alternatively, select the project in the **Project Tree** and open the **Properties** docking panel:



2. If necessary, edit the settings as described below.
3. Select **OK** to confirm any changes and exit the window. (If you are working in the Properties docking panel, changes are made immediately; there is no need to confirm.)

#### Name

This field shows the name of the VisTool project for display purposes. The name cannot be edited from here as it is defined when you [save](#) the VisTool project.

#### ClosePWD

Use this field to set a "close password".

If a password is entered, then the during runtime operation, you cannot close VisTool without entering the password. This prevents users from accidentally closing VisTool during an on-air transmission.

If the field is left blank, then no password is required to close the application.

#### DialogFont

Click here to reveal the dialog font options. These options change the font used by VisTool.

#### Force Virtual Keyboard

Set this option to **True** (ticked) if you wish to enable the virtual keyboard. When enabled, an on-screen keyboard will appear whenever user input is required. For example, to [login](#) to the user access rights system.

#### GridSize

This field sets the grid size. Enter the number of pixels - e.g. 5 x 5.

#### On-Air Project

This field defines the ON-AIR Designer project file (**.db3**) which will support the VisTool project.

For Lawo's radio on-air products, it is the ON-AIR Designer project which determines what functions can be linked to VisTool elements. Therefore, take care to choose the correct ON-AIR Designer project file!

**reverseEmberClientPort**

This field is used to define a listen port for VisTool (the Ember+ Client), so that an Ember+ Provider can establish a connection.

It is used in conjunction with the Login Observer (on the Provider). For example, in a scenario where each user has a laptop which connects (and disconnects) from the network in the studio, the Login Observer is unknown. By preparing VisTool for a reverseEmberClientPort connection, the user's Login Observer can establish the connection to VisTool when they come online.

**showHardwareNotResponding**

Set this option to **False** (unticked) if you wish to stop the [Connections](#) panel appearing automatically if the connection to a networked device is lost. This can be useful if a DMS device is not currently available.

**ToggleAssignModeKey**

This field specifies the keyboard shortcut which will enter (or leave) the assignment mode. For example, to redefine the function of a Box element during runtime.

### 6.7 Saving the Project

At regular intervals you should save the project, and also use "save as" to keep a copy of the project at different revision stages. This will allow you to test the configuration and revert to an earlier version if necessary.

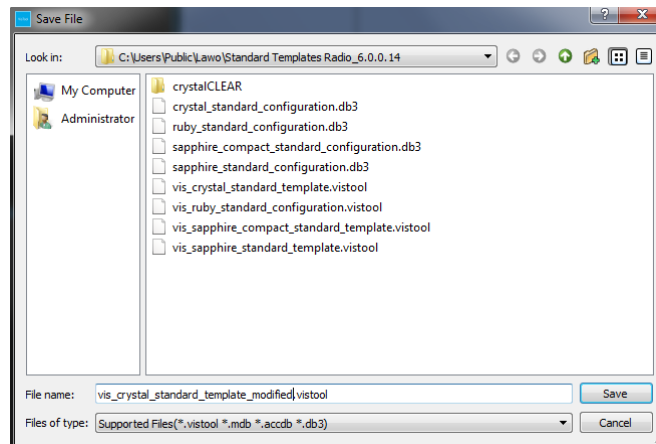
#### ➤ File -> Save

Select **File -> Save** from the main menus, or use the keyboard shortcut (CTRL + S).

This overwrites the existing project (as specified in the headline bar).

#### ➤ File -> Save As

1. Select **File -> Save As** from the main menus, or use the keyboard shortcut (CTRL + SHIFT + S) - the "Save" dialog box opens:

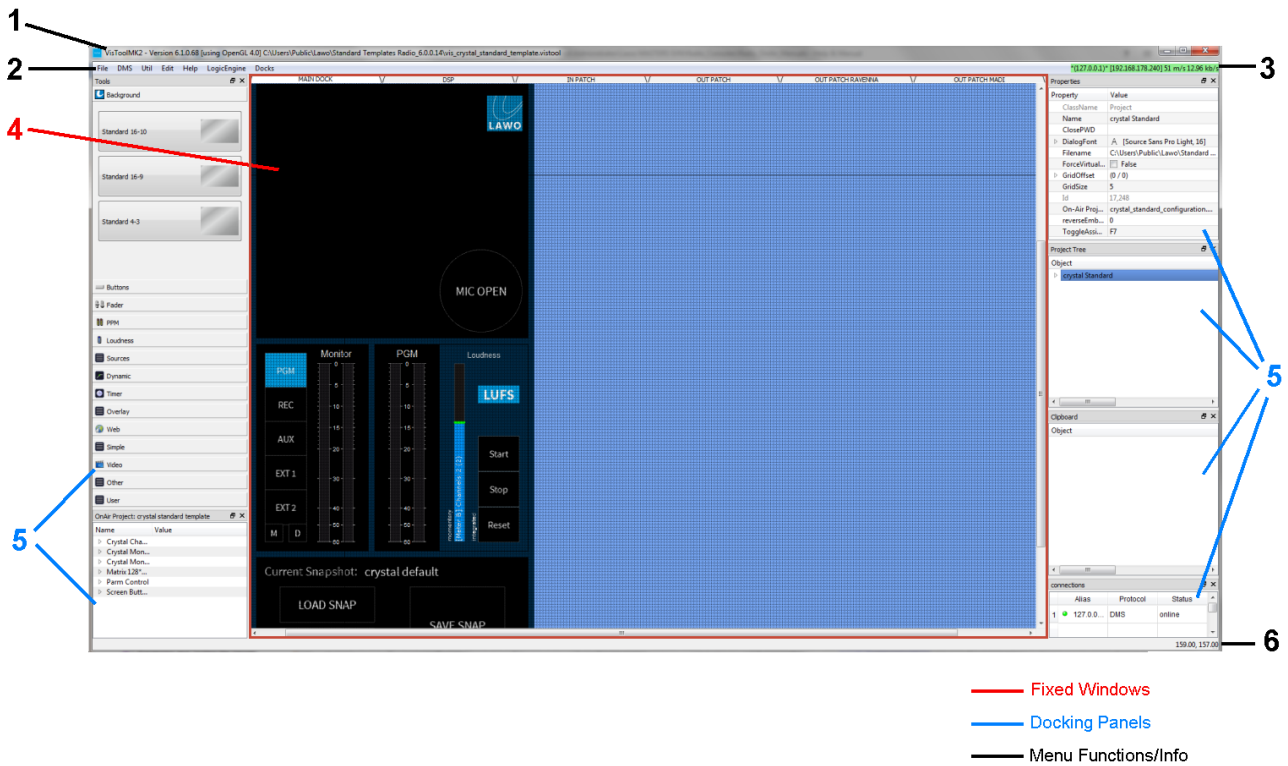


2. Choose a new folder location or enter a different file name.
3. Then click on **Save** to save the project.

The VisTool Editor project is saved on your computer's hard disk as a **.vistool** file.

The project file name is especially important if you have multiple systems to configure. See [File Naming and Organisation](#) for some recommendations.

## 6.8 Working with VisTool Editor



Once a project is open, it can be edited as follows:

### 1 Project Name

In the headline you will see the file path and name of the VisTool project. This is defined when you [save](#) the project.

### 2 Main Menus

Here you can access the [main menus](#): **File, DMS, Util, Edit, Help, LogicEngine** and **Docks**.

### 3 Online Status

This area lights in green if the device specified by ON-AIR Designer project is online. You will also see the IP address and data rate to/from the connected system.

If there is no connection, then the area lights in red and you will see **Device temporarily unavailable**. For details on connecting a system, see [Configuring a PC for VisTool Runtime](#).

### 4 Design Area

The design area shows the elements which have been added to the current page.

If you have just created a new project, then you will see a single page with no elements. In our example, an existing project has been opened with several [pages](#). Click on the page tabs at the top of the design area to view each page.

Left-click on an element to view its properties within the 'Properties' docking panel. Right-click on an element to interrogate or assign its linked function.

To [preview](#) the project, press CTRL + T (on your PC keyboard). Your computer will now show the current page on your main screen. If the connected system is online, then you can test the functionality by clicking the on-screen buttons. Press CTRL + T again to return to edit mode.

### 5 The Docking Panels

To the left and right of the design area are the docking panels which can be opened, closed, resized, floated, docked or stacked in any sidebar position. See [Arranging the Workspace](#).

The docking panels are used as follows:

- **Tools** - contains all the [VisTool elements](#). Drag and drop an element from the Tools panel onto the design area to [add](#) it to the current page.
- **On-Air Project** - contains all the functions defined in the ON-AIR Designer project. This is linked to the VisTool project via the [Project properties](#). You can change the function of a VisTool element by selecting it in the design area and then selecting a function from the On-Air Project tree. If there is no project specified, then you do not have a valid link to an ON-AIR Designer configuration.
- **Properties** - defines the [properties](#) for the selected object. These include general properties such as size, color and position, plus element-specific properties such as meter scale for meters, timer ID for timers, etc.
- **Project Tree** - shows the color mappings, pages and elements used in the project. You can use the [Project Tree](#) to interrogate and select objects.
- **Resources** - shows the number of [resources](#) used by the project. The **Solo** and **Solo Plus** tabs can be used to check if the project is suitable for a VisTool Solo or Solo Plus license.
- **Clipboard** - lists any objects which have been copied. This is useful when performing a [Copy and Paste](#) operation.
- **Connections** - shows the current status of all networked devices. It is similar to the **Connections** panel which appears in VisTool runtime (described [earlier](#)).
- **Ember+ Provider(s)** - if you have added an [Ember+](#) provider to the VisTool project, then you will see a docking panel for each provider. You can change the function of a VisTool element by selecting it in the design area and then selecting a parameter from the provider's tree. If this panel is empty, then no Ember+ providers have been specified.

### 6 Status Bar

On the right of the status bar, you will see the precise coordinates of the cursor position. This can be helpful when positioning objects.

#### 6.8.1 Undo and Redo

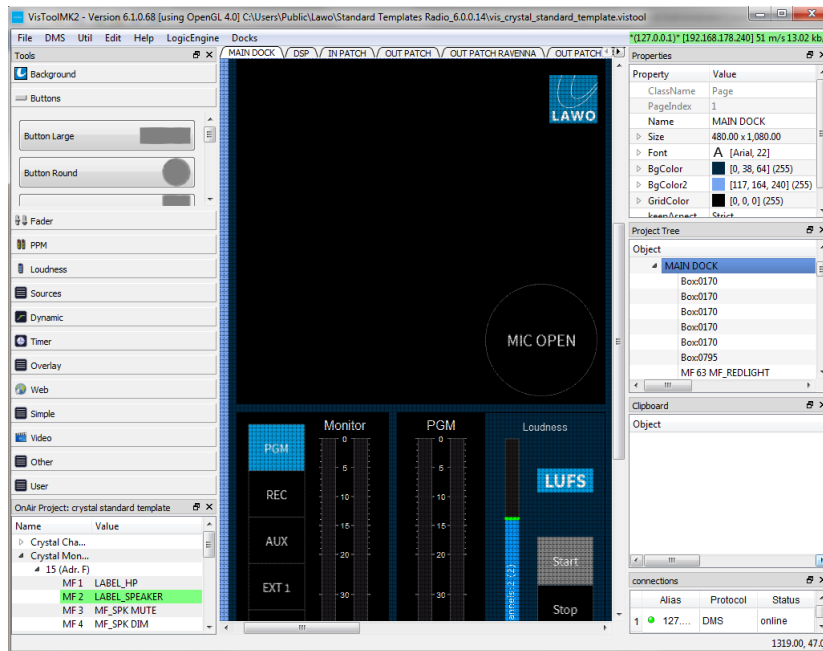
At any time you can press CTRL + Z or CTRL + Y on your PC keyboard to undo or redo the last operation.

The number of undo/redo steps are unlimited and include all VisTool Editor operations: positioning elements, defining element functions, adjusting properties, etc.

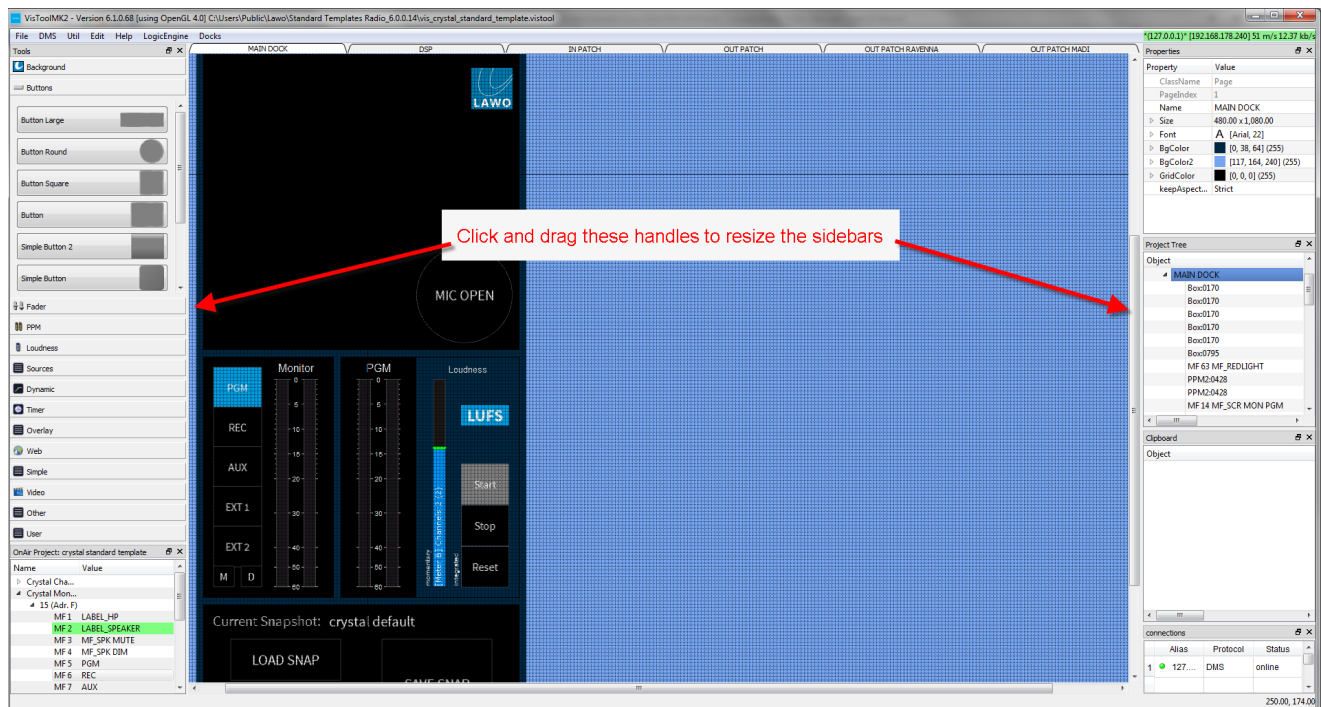
The **Undo** and **Redo** functions can also be accessed from the [Edit](#) menu.

## 6.9 Arranging the Workspace

The VisTool Editor operating window can be resized by selecting the "Restore Down" button (top right), and clicking/dragging on the window boundaries in the usual manner:



Within the operating window, you can change the width of the left and right side bars by clicking and dragging on the appropriate handles:



There is a minimum size for each docking panel which restricts the minimum height and width. If a docking panel is floated and the window size reduced, then you may not be able to re-dock the panel.

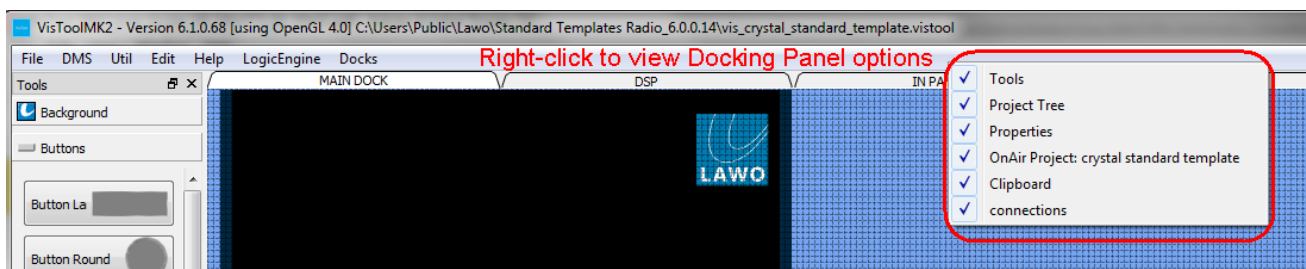


## 6. Editing a VisTool Configuration

### 6.9.1 Arranging the Docking Panels

Any of the [docking panels](#) can be opened, closed, resized, floated, docked or stacked as follows:

1. If you cannot see a panel, then either select the [Docks](#) main menu or right-click anywhere to the right of the main menus. The view options allow you to show or hide each panel:

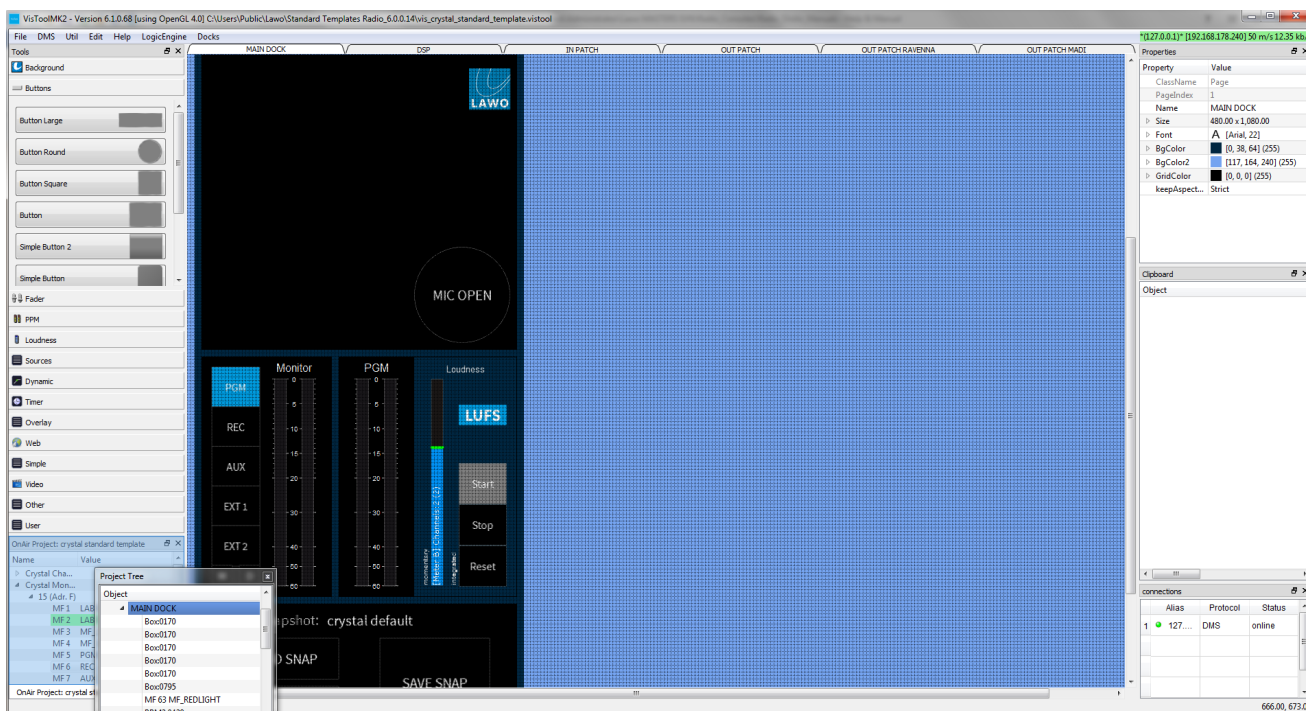


2. To close a panel, click on its close button **x**. To re-open the panel, use the view options described above.
3. To release a docking panel from its current position, click on its float button.

The panel becomes a floating window which can be resized and positioned in the usual manner.

4. To dock a floating panel, release it into the left or right sidebar - a blue background appears when a valid area is reached. When you release, the panel will fill the available space.
5. To stack a floating panel, release it above another panel - the selection turns blue (as shown below). When you release, the panel will be stacked. All of the docking panels can be stacked if you wish.

*'Project Tree' Docking Panel - will be stacked on left with 'OnAir Project'*



6. To unstack a panel, click on its float button to create a floating window, and then re-dock the panel as described above.

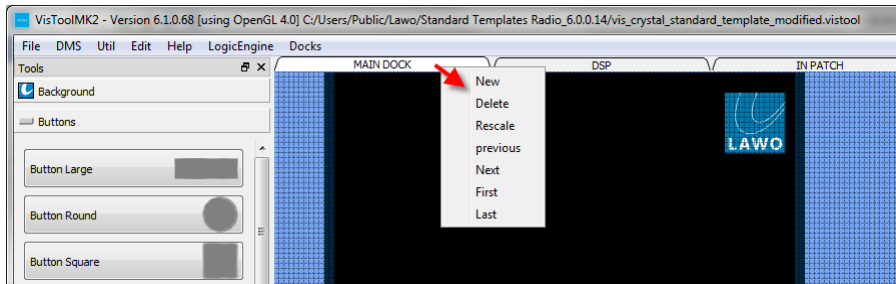
## 6.10 Creating and Managing Pages

A VisTool project can contain several pages. Each page can be displayed on a separate runtime screen by defining different **Start Page** and **Monitor** options per session/screen (in [VisTool Sessions](#)). Alternatively, pages can be displayed on the same runtime screen by [configuring page switching](#) (via page keys or remote triggers).

### 6.10.1 Adding a New Page

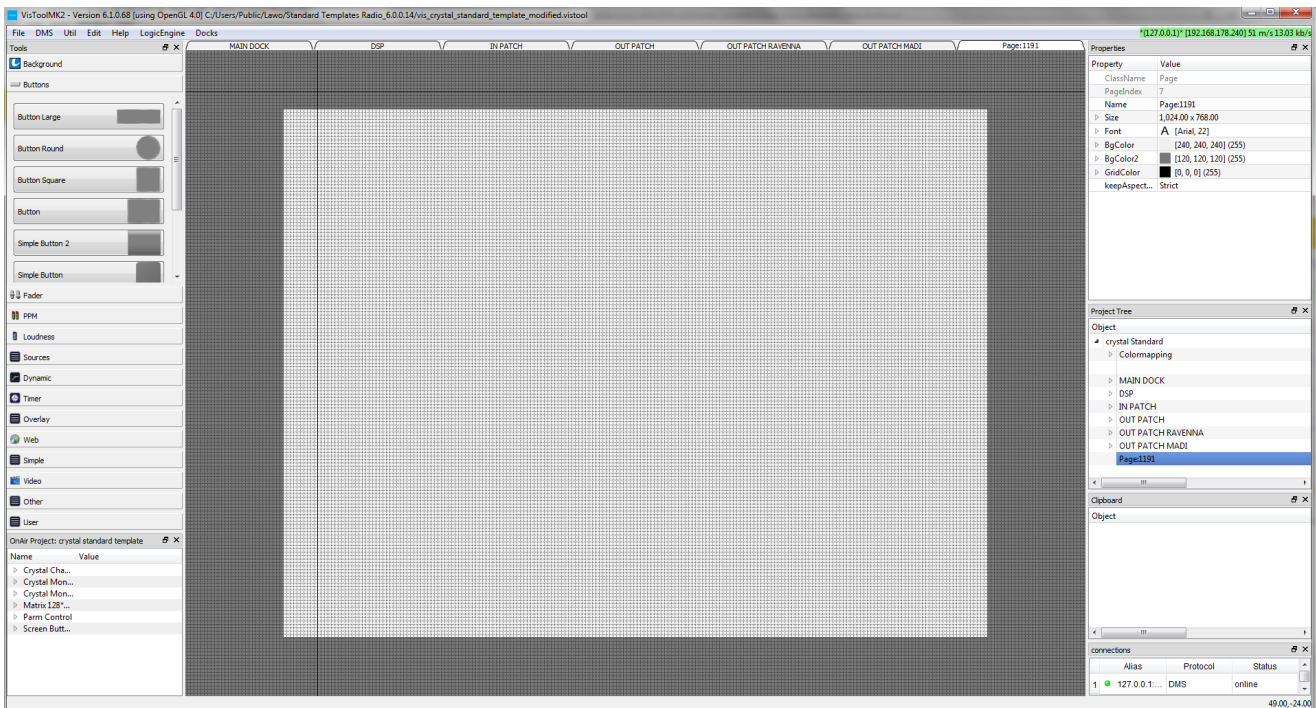
To add a new page:

1. Right-click anywhere in the page tabs area to open up the "page" options:



2. Select **New** to add a new page.

New pages are added to the right of any existing pages and given a default name - in our example, **Page:1191**:

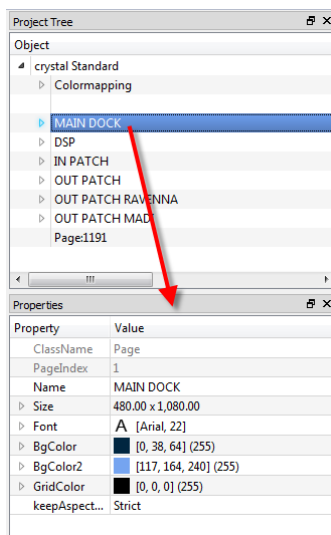


### 6.10.2 Page Properties

Before adding elements, it is best to set the page 'Properties'. In particular, the height and width of the page will affect how it is displayed.

1. Select the Page either by clicking in the general page area (not on an element), or on the Page name in the [Project Tree](#) - the 'Properties' docking panel updates accordingly.
2. Use the panel fields to edit the properties as described below.

The example below shows the properties for the page named "MAIN DOCK" (included in the standard template):



**ClassName** - for information only.

**Page Index** - for information only.

You will need this number to define the [Start Page](#) (in VisTool Sessions) and/or a [Default Page](#) in "Logic -> VispageSwitch" elements.

The page index is generated automatically with page 1 always on the left, page 2 the next page to the right, and so on. Therefore, take care when [moving](#) pages, as the page index may be referenced by the ON-AIR Designer configuration.


**Name** - edit the page name by clicking in the name field.


**Size** - open this branch to set the page width and height.


It is important that the page size matches the ratio of your VisTool runtime screen (defined using the [Monitor](#) options in VisTool Sessions).

So, for example, if the page will be displayed on a 4:3 monitor, set the **Width** to **300** and **Height** to **400**. Alternatively, if the page is to be displayed as a frame-less window or docking bar, set the dimensions to match accordingly. If you are configuring pages for the **Visual Extension** touch-screens, set the **Width** to **800** and **Height** to **480**.

**Font** - sets the font for any displayed text. Click on  to open the font selector, or click on the arrow to access the individual font property fields.

**BgColor** - sets the background color of the page. Click on  to open the color selector, or click on the arrow to access the individual color fields.

**BgColor2** - sets the background color outside of the page (i.e. outside of its visible area). Click on  to open the color selector, or click on the arrow to access the individual color fields.

**Grid Color** - sets the color of the [grid](#). Click on  to open the color selector, or click on the arrow to access the individual color fields.

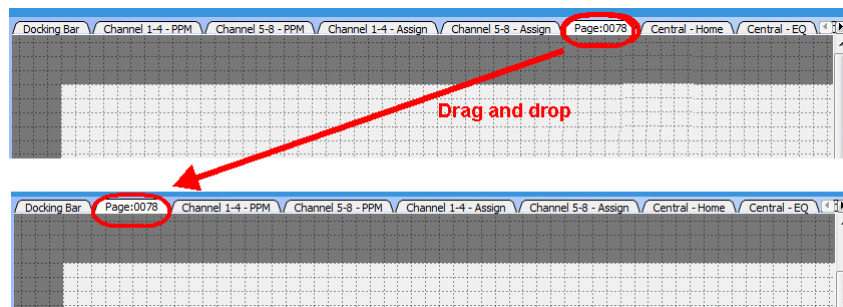
**keepAspect** - if the page is displayed on a bigger monitor than that defined by the Size, then these properties determine how the page will be scaled:

- **Strict** - keeps strictly to the original screen ratio. You will see a black bar either side of the page if the monitor does not match your screen ratio.
- **Loose** - allows a small amount of screen distortion - for example, if 16:9 is configured but a 16:10 monitor used, then the screen will be scaled to fit.
- **None** - the screen is automatically scaled to fit the monitor.

### 6.10.3 Moving a Page

To move a page within the project:

1. Drag and drop the page tab to its new position:

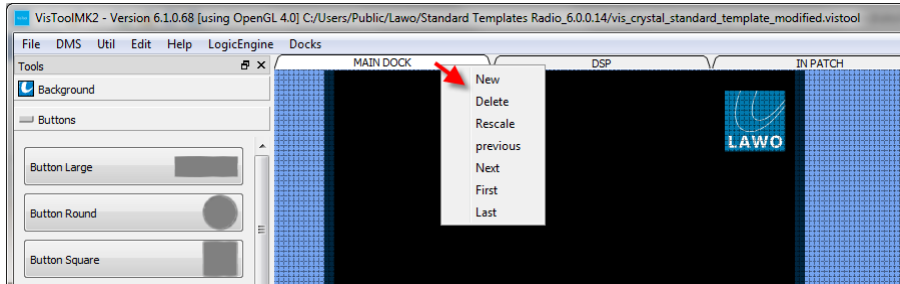


Moving a page alters the [page index](#) for every page in the project. Therefore, take care if pages are already referenced by the system configuration.

Once a page and its elements have been defined, you can use [Copy and Paste](#) to copy all the elements of a page.

### 6.10.4 Other Page Options

1. Right-click anywhere in the page tabs area to open up the "page" options:



- **New** - [adds](#) a new page.
- **Delete** - deletes the current page. Be very sure before deleting, as this will delete the page and all its contents. It will also change the [Page index](#) for all higher numbered pages.
- **Rescale** - opens a pop-up where you can resize the page (this is the same as adjusting the page **Width**, **Height** and **keep Aspect Ratio** from the [Page Properties](#)).
- **previous** - opens the previous page.
- **Next** - opens the next page.
- **First** - opens the first page.
- **Last** - opens the last page.

You can also use [keyboard shortcuts](#) for page navigation.

If your project contains lots of pages, then use the arrow buttons on the right to scroll through the page tabs.

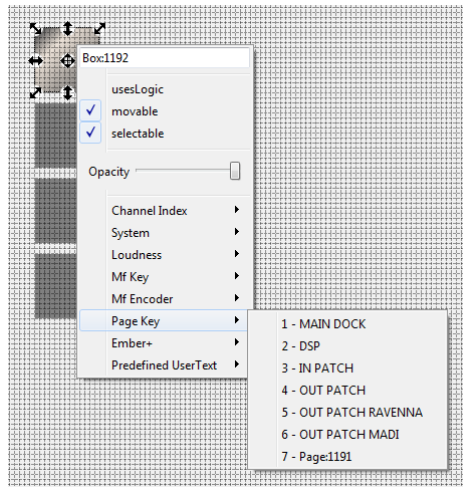
### 6.11 Configuring Page Switching

To switch between pages on a VisTool runtime screen, you will need to configure page navigation buttons for the operator. There are two possibilities:

- VisTool Page Keys (on-screen).
- Automatic page switching triggered by a control signal (from the remote system).

#### 6.11.1 Configuring VisTool Page Keys

1. Add an element for each Page Key to the current page - it is recommended to use **Box** elements. See [Adding Elements to a Page](#).
2. Right-click to assign the element's function and choose the **Page Key** drop-down menu - you will be offered all the pages in the project. Select a page to make the assignment:



Once assigned, the element takes on the name of the page.

3. Adjust the [properties](#) for the VisTool Box element, such as its color, size, font, etc.
4. Then use the [Multipage Object](#) property to repeat the element across the other pages in the project.
5. Test the Page Key operation by [previewing](#) the project.

It is a good idea to position Page Keys in the same place on every page (to achieve a consistent appearance).

## 6. Editing a VisTool Configuration

### 6.11.2 Configuring Automatic Page Switching

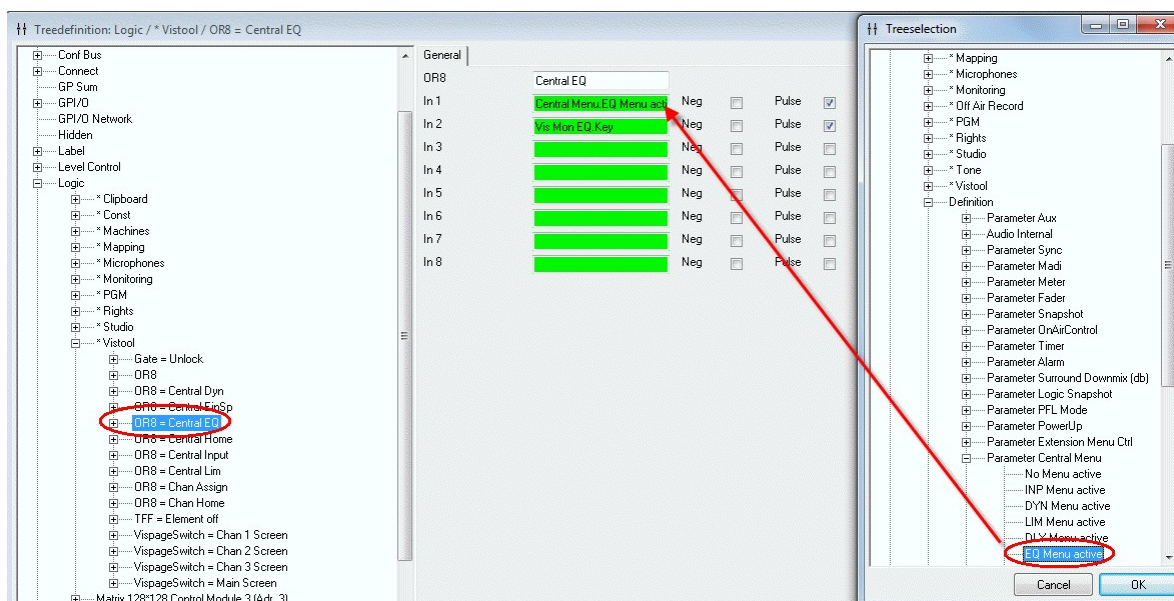
Using a [VispageSwitch](#) element, you can configure automatic VisTool page switching in response to control signals received from the remote system. For example, to show parameters for the source in ACCESS in response to the function buttons on a console surface (INP, EQ, DYN, etc).

There are numerous applications, but they all are programmed in a similar manner:

- First, using VisTool Editor, design the page or pages which you want to appear on the VisTool screen(s). For our example, design a page to show input parameters, a page to show EQ parameters, and so on.
- Then, using ON-AIR Designer, configure a "Logic -> VispageSwitch" element to handle the page switching. Each element can switch any VisTool page to any screen with a matching **Station Id**. The pages are referenced by their index number (from 1 to 32) defined in VisTool Editor.
- Finally, use VisTool's "Session Management" window to assign the **Station Id** to the correct VisTool screen.

Below is an example of the ON-AIR Designer configuration steps. By using a "Logic -> OR8" element, pages may be switched either automatically, in response to the function button mode, or manually, when the operator presses an MF Key:

1. Name the MF Key you wish to use for the manual operation.
2. Create and configure a Logic -> OR8 element to "OR" the control output from the function button (e.g. EQ) and the operator's MF Key:

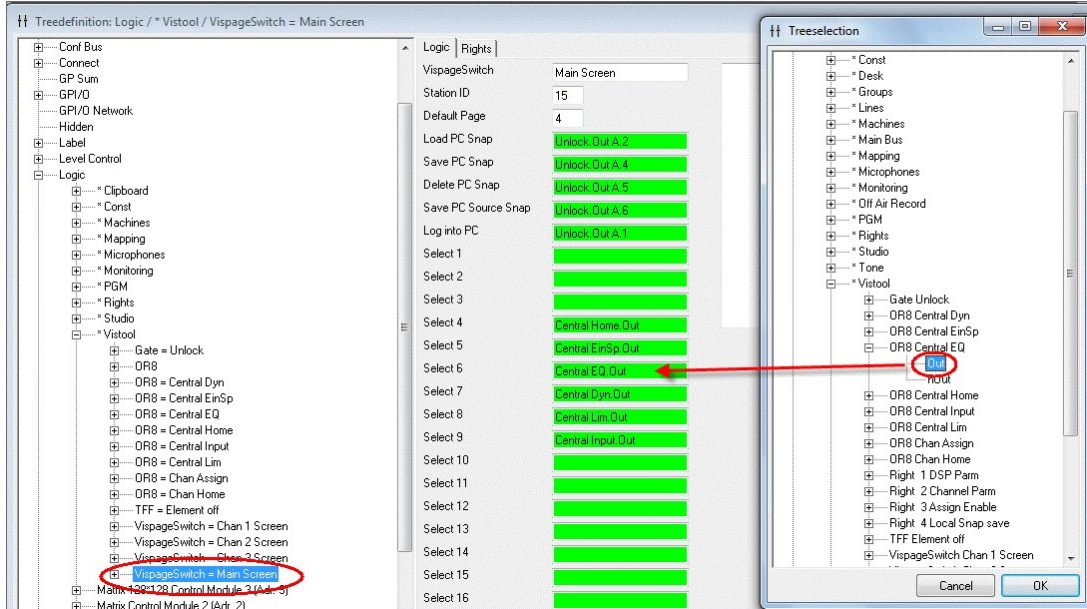


3. Create a [Logic -> VispageSwitch](#) element for the VisTool screen and set the **Station Id**. This should be a unique number.

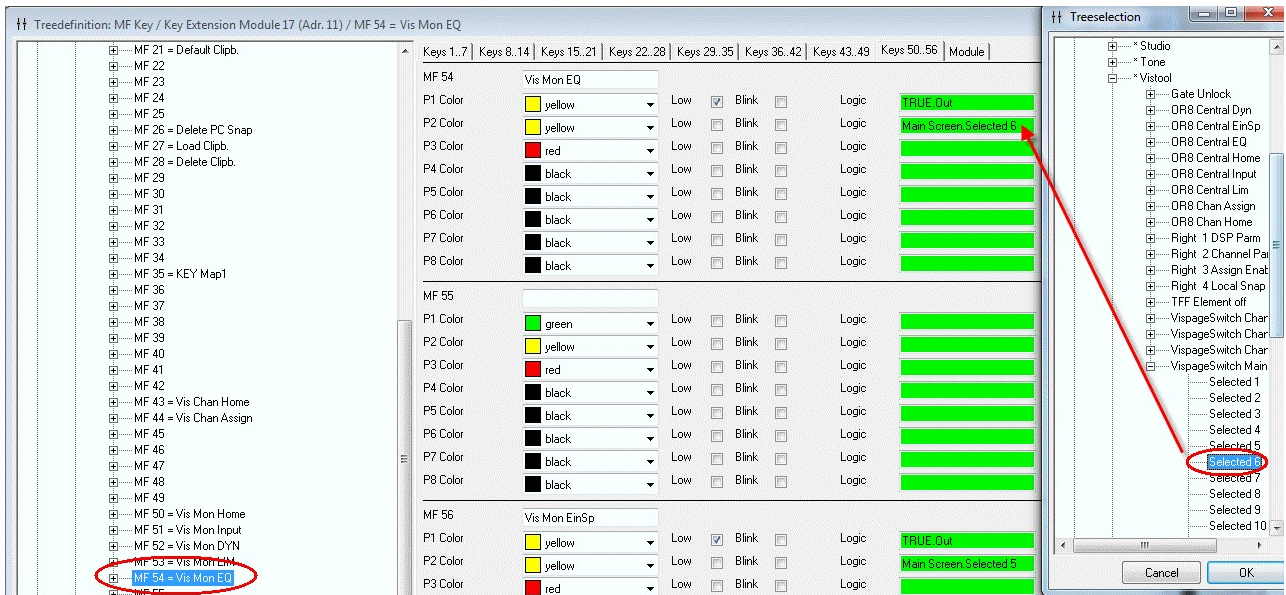
You will need to enter the **Station Id** into VisTool's [Session Management](#) window in order to define the VisTool screen.

**4.** Assign the output of the OR8 element to the correct page **Select** field.

**Select 1** corresponds to page index 1, **Select 2** to page index 2, and so on. Therefore, look at your **VisTool Editor** configuration and take a note of the [page index](#) of your EQ page. In our example, it is page index 6, and so we have assigned the control signal to page **Select 6**:



**5.** Return to the MF Key element, and assign the **Selected 6** control output to a lamp color - this will ensure that the MF Key lights when the EQ page is selected:



**6.** Repeat for each VisTool page you wish to switch - e.g. input parameters, dynamics parameters, pan parameters.



### 6.12 Configuring Elements

VisTool elements are added to each page using the **Tools** docking panel. The individual elements are described in more detail [later](#). This topic describes how to add an element, assign its function and change its appearance.

#### 6.12.1 Adding Elements to a Page

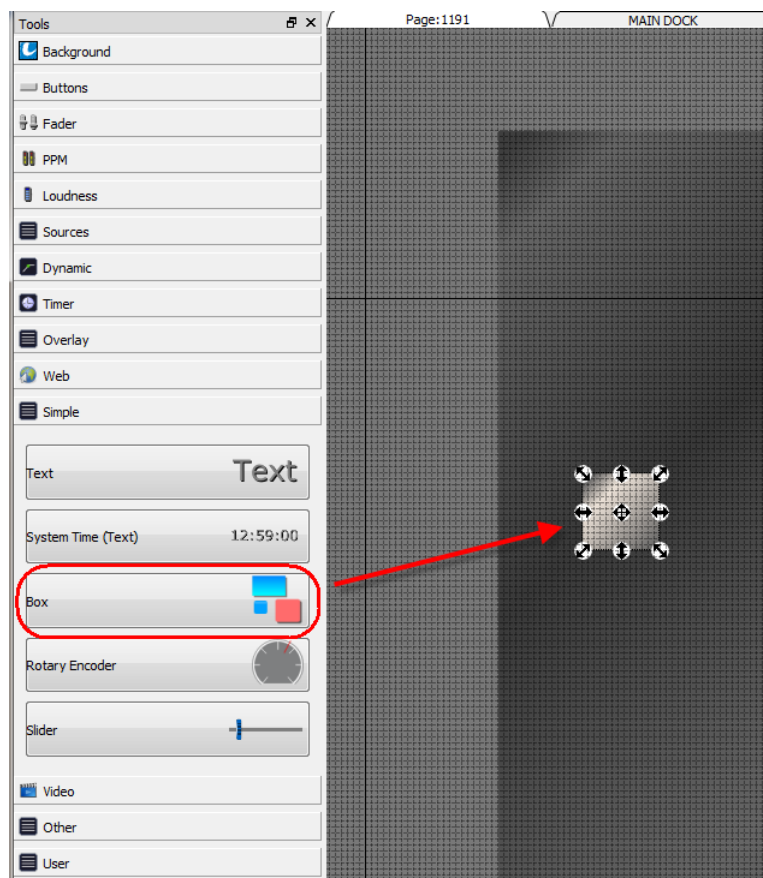
To add a VisTool element to a page:

1. From the **Tools** panel, select a category - e.g. **Simple**.

If you cannot see the **Tools** panel, check the docking panel [view options](#).

2. Click and drag your chosen element onto the page - in our example, a **Box** element. The element is placed when you release your mouse button.

Elements are added at their default size.

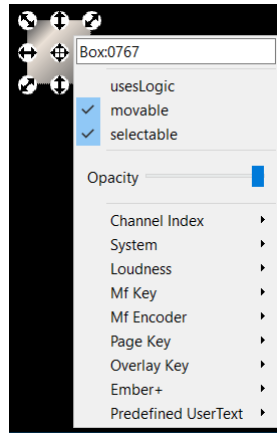


3. Repeat to add more elements.
4. To delete an element, select it and press **DELETE** on your computer keyboard.

Use the [grid](#) and [snap to grid](#) to help position elements. See also [Positioning Elements](#).

### 6.12.2 Defining the Element's Function

- The easiest way to define an element's function, and other common properties, is to right-click on the element - a pop-up window appears.



The first few options define some of the most common properties (also available in the [Properties](#) panel):

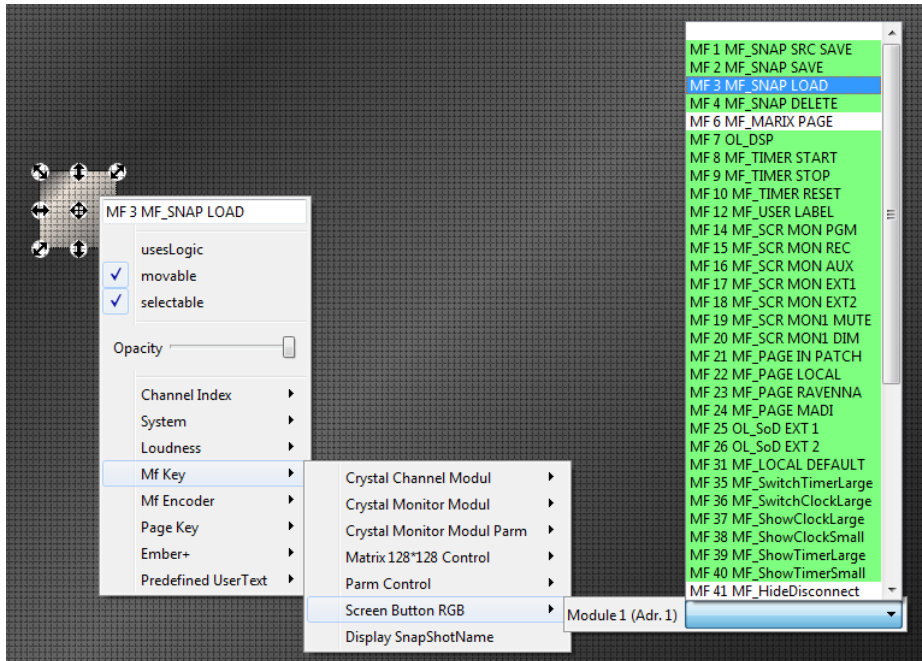
- Box:xxxx** - this is the **Name** of the VisTool element. Edit the name by clicking in the name field.
- useLogic** - this property is element-specific. It allows the Box element to be added to the [Logic Engine](#) Editor.
- movable** - untick this option to prevent the element being moved. This stops you accidentally changing its position while designing your layout.
- selectable** - untick this option to prevent element selection. This renders the element inoperable in the VisTool runtime page. It can also be useful when editing if you wish to overlay elements and stop those at the back from being selected.
- Opacity** - use this slider to change the **Opacity** of the element (1 = fully opaque; 0 = transparent). If you wish to set the opacity to 0, use the [Limit opacity in Edit Mode](#) option to still view the element (in Edit mode).

The lower options assign a function to the element. What you see depends on the type of element and its current configuration. The table below describes the possibilities for our **Box** element.

<b>Channel Index</b>	required for <b>Predefined User Text</b> -> <b>Channel</b> parameters.
<b>System</b>	a function which affects the application: minimize, fullscreen, close, etc.
<b>Loudness</b>	an integrated loudness measurement command: Start, Stop, etc.
<b>MF Key</b>	an MF Key function defined by the ON-AIR Designer project.
<b>MF Encoder</b>	a rotary encoder on the console surface.
<b>Page Key</b>	a VisTool Editor Page Key for on-screen page navigation.
<b>Overlay Key</b>	a key to open an Overlay element.
<b>Ember+</b>	displays existing Ember+ parameter assignments.
<b>Predefined User Text</b>	a predefined user text function. Various options are supported. See <a href="#">Predefined User Text</a> .

## 6. Editing a VisTool Configuration

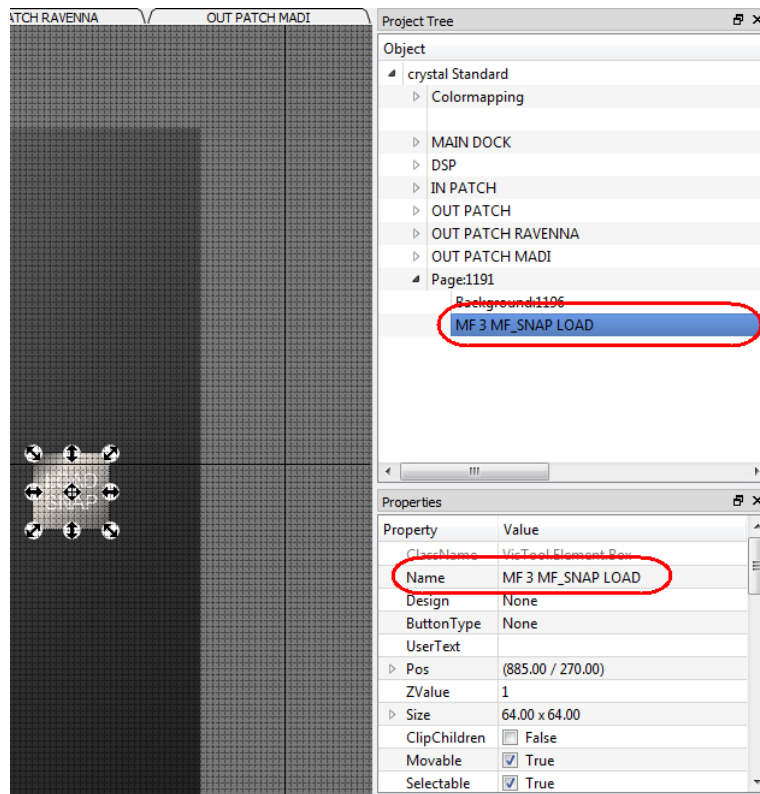
- As an example, select **MF Key** and use the drop-down menus to assign an MF Key from the remote system (e.g. SNAP LOAD):



Most [MF Key functions](#) are read from the ON-AIR Designer configuration (defined in the [Project properties](#)).

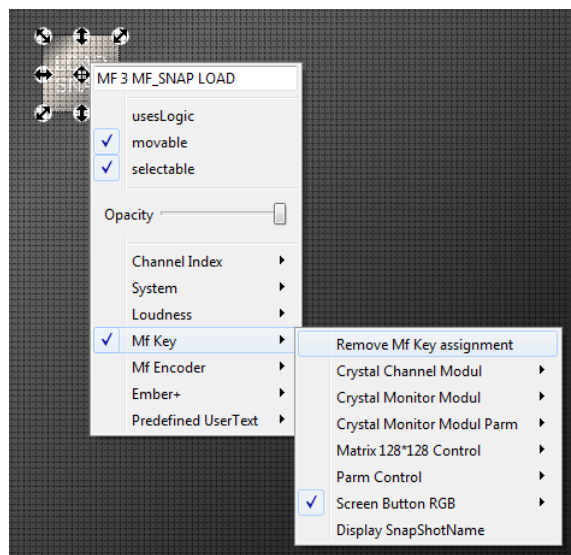
VisTool Editor reads the ON-AIR Designer file on start-up, and updates its available options accordingly. Therefore, if a function does not exist in the ON-AIR Designer project, it will not appear in the MF Key functions list in VisTool Editor. If you make a change to the ON-AIR Designer configuration while VisTool Editor is open, then either restart VisTool Editor or update the 'Project' properties to force VisTool Editor to re-read the configuration data.

Once you have assigned a function, its **Name** appears in the [Project Tree](#) and [Properties](#) panels:



You can use the **UserText** field (in the [Properties](#) panel) to label elements in the design area. If you are running online, press CTRL + T (on your PC keyboard) to test the assigned function. See [Previewing and Testing the Design](#). You can name multiple elements very quickly by defining a [variable](#).

- To remove the function assignment, right-click and select **Remove MF Key assignment**:



- Repeat these steps to assign functions to all the elements on your page.


## 6. Editing a VisTool Configuration

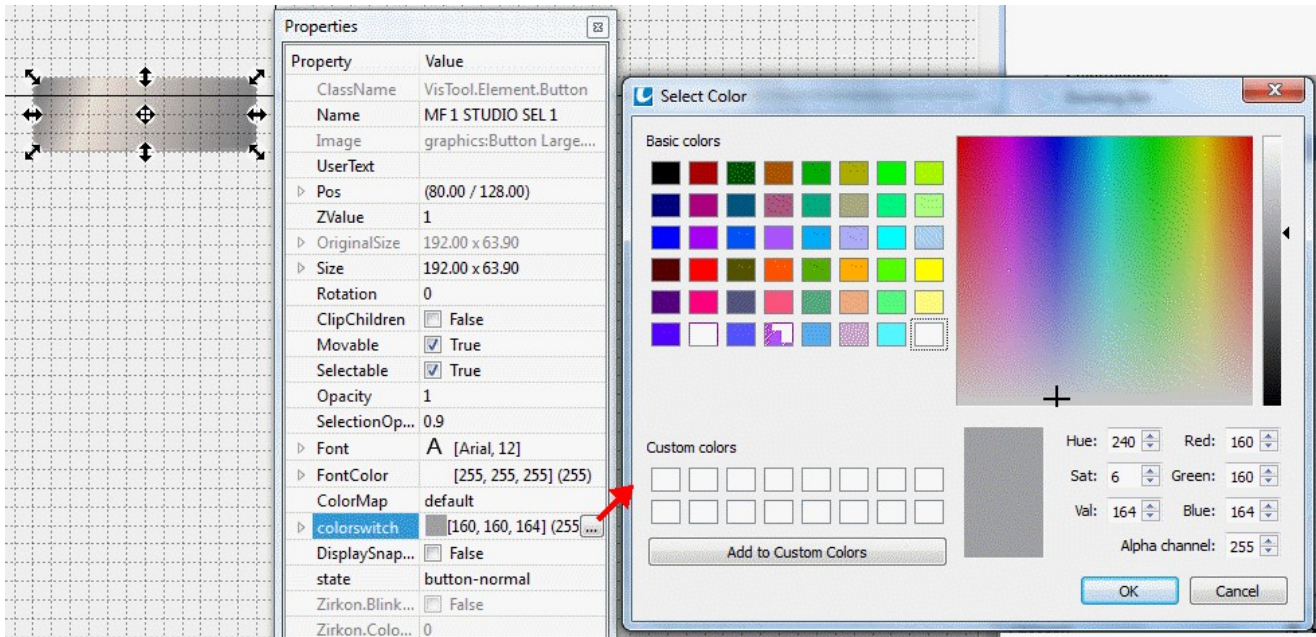
### 6.12.3 Editing the Element's Properties

The **Properties** panel includes general design settings such as size, color and position, plus element-specific properties such as meter scale for meters, timer ID for timers, etc.

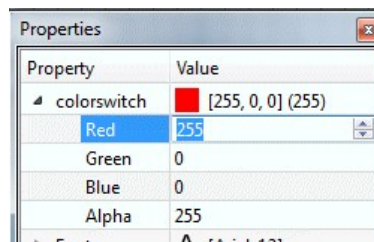
1. Select an element in the design area - the **Properties** panel updates.

If you cannot see the **Properties** panel, check the docking panel [view options](#).

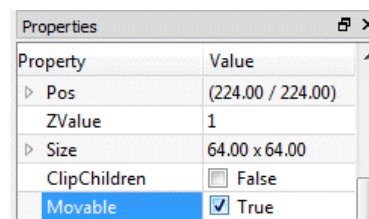
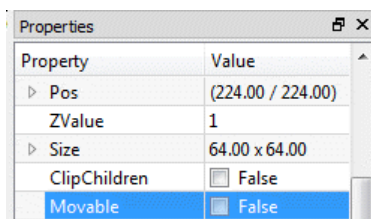
2. To adjust a property, you may need to open a selection window - for example, click in the **colorswitch** field and select the  button to open the color selector:



3. Some properties have an arrow which you may click to reveal child properties - for example, click on the arrow beside **colorswitch** to access the individual **Red**, **Green**, **Blue** and **Alpha** fields:



4. Some properties are enabled or disabled using a checkbox - the current state (True or False) updates accordingly:





### 6.12.4 General Properties

Most elements support the general properties described below. Note that greyed out entries, such as **ClassName**, are for reference only and cannot be adjusted. Element-specific properties are described later for each individual element, see the [VisTool Elements](#) chapter.

*Properties (Box Element)*

Property	Value
ClassName	VisTool.Element.Box
Name	MF 3 MF_SNAP LOAD
Design	Design-4
ButtonType	None
UserText	LOAD SNAP
> Pos	(50.00 / 850.00)
ZValue	1
> Size	160.00 x 60.00
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Glass Look	None
Metal Frame	0
Opacity	1
Roundness-X	0
Roundness-Y	0
TextAlign	Center
> Font	A [Source Sans Pro, 16]
ColorMap	LawoCI
FontColorMap	default-font
FrameColorMap	default
use DMS font color	<input type="checkbox"/> False
use DMS frame color	<input type="checkbox"/> False
> Color	■ [0, 0, 0] (255)
> ColorNone	■ [136, 136, 136] (255)
> ColorOff	■ [2, 253, 99] (255)
> ColorPost	■ [253, 253, 20] (255)
> ColorPre	■ [253, 2, 20] (255)
> FontColor	[255, 255, 255] (255)
> FontColorNone	[255, 255, 255] (255)
> FontColorOff	[255, 255, 255] (255)
> FontColorPost	[255, 255, 255] (255)
> FontColorPre	[255, 255, 255] (255)
Frame	0
> FrameColor	■ [0, 0, 0] (255)
manual-min	nan
manual-max	nan

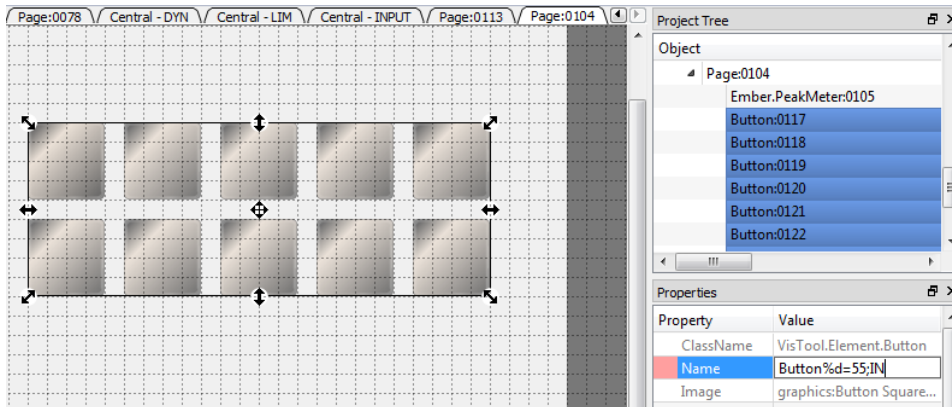
- **ClassName** - shows the type of element. This is for reference only and cannot be adjusted.
- **Name** - this is the name of the VisTool element (displayed in the [Project Tree](#), and if you right-click on an element in the design area). Edit the name by clicking in the name field. Note that, for buttons, the name automatically updates once you [assign a function](#). You can name multiple elements very quickly by defining a [variable](#).
- **UserText** - enter text to annotate the element. You may force a word-wrap by typing "||"; up to three lines of text are supported. The text can be aligned or highlighted using the **Text Align** and **Text Appearance** options.
- **Pos (X and Y)** - enter the XY coordinates for the element's position in pixels. The coordinates set the top left corner position.
- **Z Value** - this value controls whether the element is in front of, or behind, overlapping elements. The higher the **Z Value**, the further front the element will be. Enter a lower **Z Value** to move the element backwards; enter a higher **Z Value** to move the element forwards. If a child object is given a **Z Value** of less than 0, then it will appear "within" the parent object - for example, to produce buttons with "frames" which are controlled by the child's functionality.

- **Size** - this value shows the width and height of the element. Click on the arrow to access the **Width** and **Height** fields, and enter a new size in pixels.
- **Rotation** - enter a value to rotate the element in degrees.
- **ClipChildren** - this option affects what happens to child objects when the parent is cropped. Untick the option (**False**) to leave child objects intact. Tick the option (**True**) to clip the child objects to the new parent size.
- **Movable** - untick this option (**False**) to prevent the element being moved. This stops you accidentally changing its position while designing your layout.
- **Selectable** - untick this option (**False**) to prevent element selection. This is useful if, for example, you want to overlay elements and prevent ones at the back from being selected. This state also renders the element inoperable in the VisTool runtime page
- **Opacity** - enter a value to change the **Opacity** of the element: 1 = fully opaque; 0 = transparent. Use the [Limit opacity in Edit Mode](#) option to view transparent elements in Edit mode.
- **TextAlign** - sets the text alignment for the element's **User Text**.
- **Font** - sets the font for any displayed text. Click on  to open the font selector, or click on the arrow to access the individual font property fields.
- **ColorMap** - sets the [color\\_map](#) for the element (used to overwrite the **Color** property if the element's state is changed from an MF Key). Click to make a selection from the drop-down menu.
- **Color** - sets the color of the element. Click on  to open the color selector, or click on the arrow to access the individual color fields.
- **Text Appearance** - this drop-down menu selects the highlighting option for the element's **User Text**. Options include **Normal** (no change), **Raised**, **Sunken**, **Contour Dark**, **Contour Light**, etc.

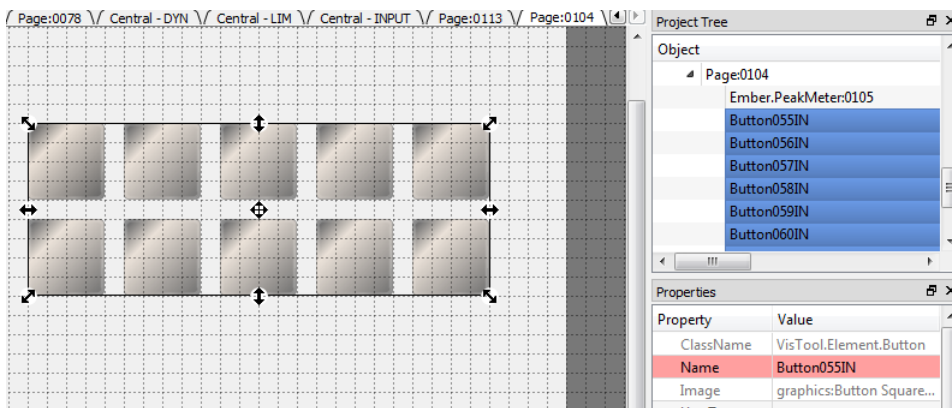
### 6.12.5 Naming Multiple Elements (using a Variable)

By entering a variable into the **Name** field of the [Properties](#), you can name several consecutive elements very quickly:

1. Select all the elements you wish to name - in our example, several button elements.
2. Then enter the following syntax into the **Name** field: "Element"%d=nn;"Name Text" where:
  - "Element" describes the element type (e.g. **Button**)
  - nn= the starting number for the incremental naming (e.g. **55**)
  - "Name Text" is the text suffix applied (e.g. **IN**).



3. Press Enter to confirm - the selected elements are automatically named accordingly:





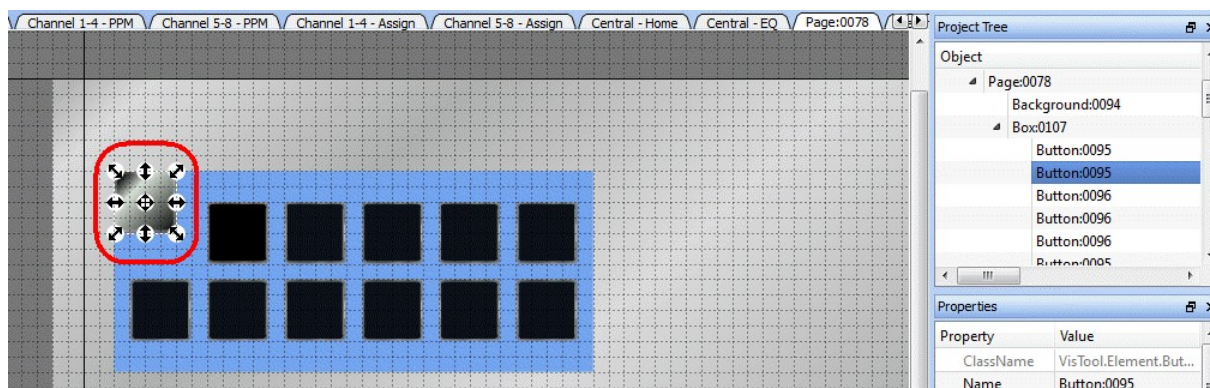
### 6.13 Positioning Elements

VisTool Editor includes several tools to help position elements.

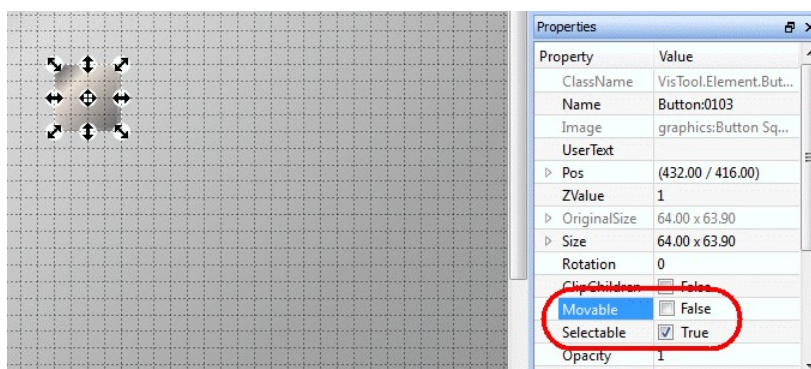
At any time you can press CTRL + Z to undo the last operation, see [Undo and Redo](#).  
Use the [grid](#) and [snap to grid](#) to help positioning.

#### 6.13.1 Moving Elements

1. In the design area, click and drag the central  handle to move an element. Note that the element is highlighted in the [Project Tree](#) as confirmation of your selection:

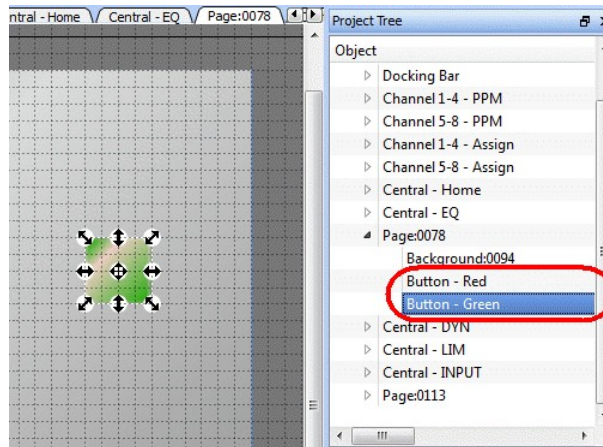



If an element cannot be moved, then check the **movable** and **selectable** options in the [Properties](#) panel:




Press **SHIFT** and click to select an unselectable element! This acts as a temporary override if **selectable = False** is active.

If an element is covered by another element, then click once to select the upper element; click again to select the lower element - the [Project Tree](#) follows accordingly. Once the lower element is selected, it can be moved in the usual manner:



Note that the central  handle moves the selected element; the other handles will [resize](#) the selection.

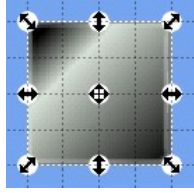
Once an element is selected, you can also press the **Left/Right/Up/Down** arrow keys (on your PC keyboard) to nudge the selection.

2. To select multiple elements, use any of the following methods:
  - Press CTRL + A (on your PC keyboard) to select all elements on the current page.
  - Press and hold CTRL (on your PC keyboard) and then select individual elements either in the design area or [Project Tree](#).
  - Click and drag around the elements you wish to select (in the design area).
  - Press and hold SHIFT (on your PC keyboard) and select the first and last element in the [Project Tree](#) to select a range.
3. Having selected multiple elements, drag the selection (using the  handle in the design area) to move all your selections together.

Use the [right-click options](#) to align multiple objects.

### 6.13.2 Resizing Elements

To resize an element, select it and then click and drag on one of the outer handles:



Press SHIFT (on your PC keyboard) while resizing to maintain the aspect ratio.

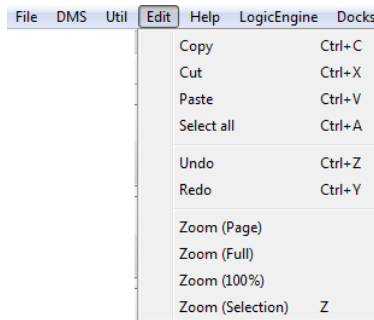
If a selection includes text, then the font is not resized. Press CTRL, and then click and drag, to adjust the font size.

If multiple elements are selected, then all selections are resized together.

The [right-click options](#) can be used to apply the same width or same height to all selected elements.

### 6.13.3 Zooming In and Out

You can zoom the design area using the **Zoom** options in the [Edit](#) menu:



- **Zoom (Page)** - displays the page.
- **Zoom (Full)** - displays the page plus any overlays (situated outside the page area).
- **Zoom 100%** - zooms to the actual size.
- **Zoom (Selection)** - zooms in on the current selection.

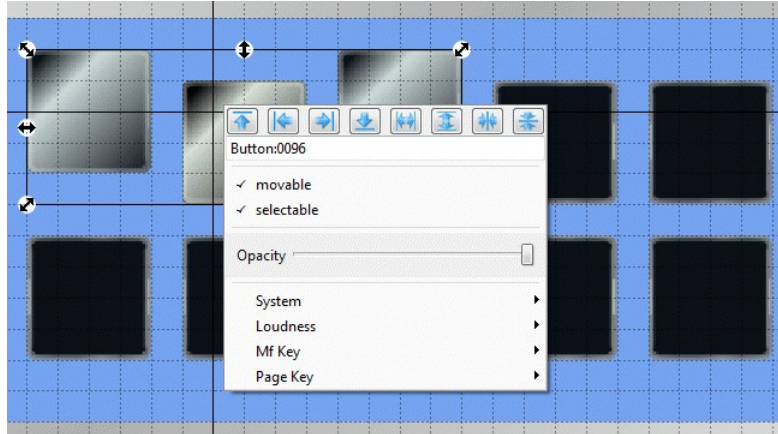
You can press Z (on your PC keyboard) to zoom to the current selection.

If no elements are selected, press Z to zoom to the page.

You can also use the LEFT/RIGHT/UP/DOWN keys (on your PC keyboard) to scroll the design area window if the page is selected.

### 6.13.4 Aligning Elements

1. Select the elements you wish to align.
2. Right-click on the element you wish to use as the alignment reference - in our example, on the second element from the left:



3. Then select a right-click option using the buttons running across the top of the window:

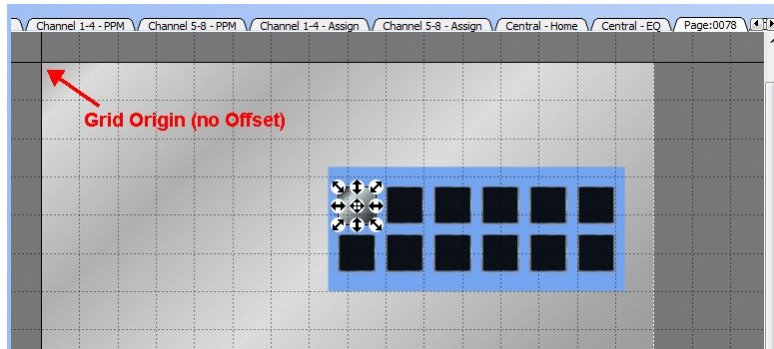


Working from left to right, the options are:

- Align to Top
- Align to Left
- Align to Right
- Align to Bottom
- Apply Same Width to all Selections
- Apply Same Height to all Selections
- Center all to X-center of this one
- Center all to Y-center of this one

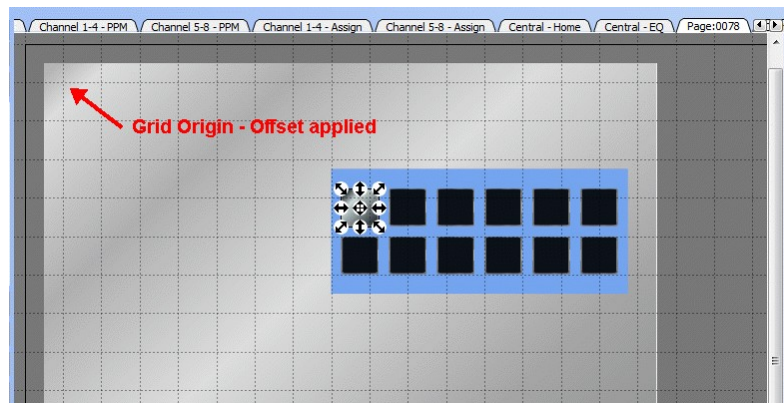
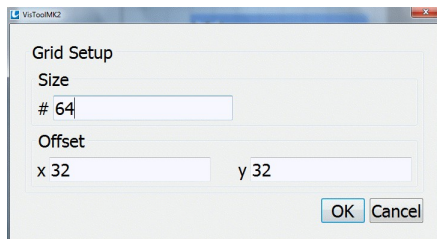
### 6.13.5 Using the Grid

Select **Util** -> **Show Grid** (from the main menus) or press # (on your PC keyboard) to show or hide the grid. The grid is applied to the whole of the design area; its origin being the top left corner of the page:



You can adjust the color of the grid for each page in the [Page properties](#).

Select **Util** -> **Grid Size** (from the main menus) or press ALT + # to open the grid setup window. Here you can specify the **Size** of the grid; the smaller the number, the closer the grid lines. Use the **Offset X** and **Y** fields to apply an offset to the grid origin - for example:

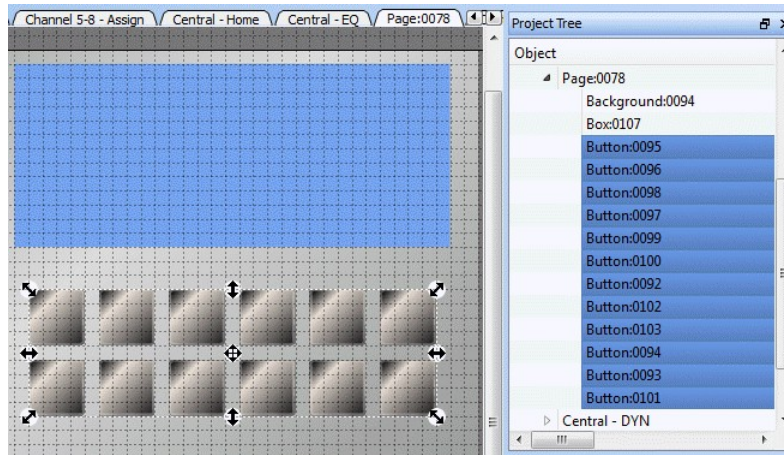


### 6.13.6 Nesting Elements

Elements can be nested in order to group elements and simplify the [Project Tree](#). In the example below, a **Box** element has been added to create a frame for a series of **Button** elements.

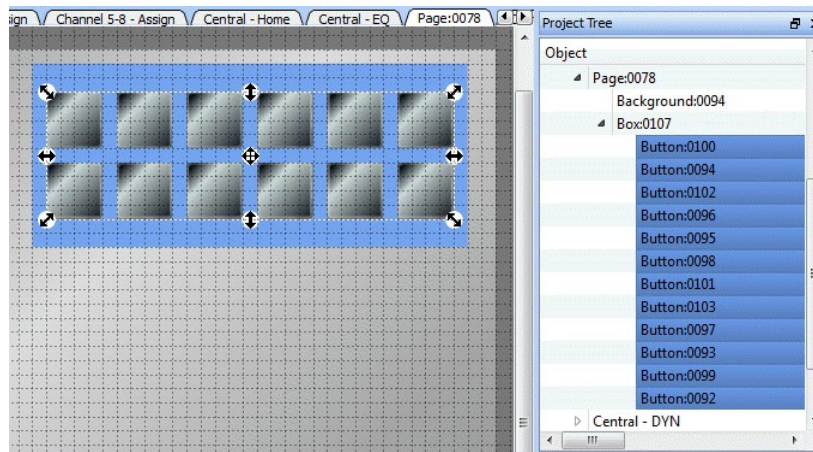
To nest all the **Buttons** within the **Box**:

1. Select the elements you wish to nest - e.g. click and drag around the **Buttons**:



2. Press and hold the ALT key (on your PC keyboard) and drag the selection onto the parent element (in our example, the **Box**). The parent element changes color to indicate that you are nesting objects, rather than simply moving them.

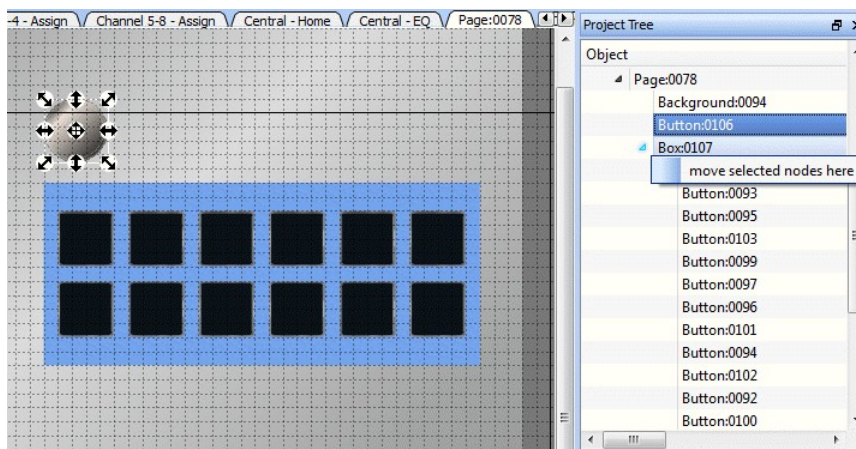
When you release the ALT key, the elements are nested within the parent branch of the **Project Tree**:



## 6. Editing a VisTool Configuration

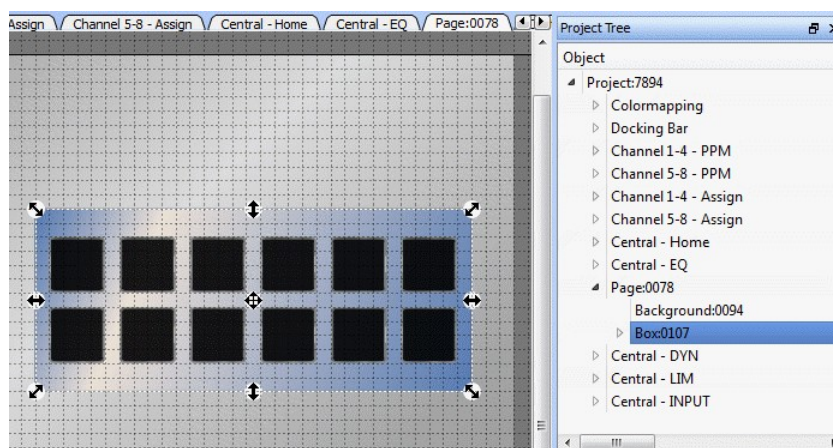
To add a single element to a parent - for example, to add another **Button** to the **Box**:

1. Select the element you wish to add, then right-click on the parent (in the [Project Tree](#)) and select **move selected nodes here**:



### Nested Element Behaviour

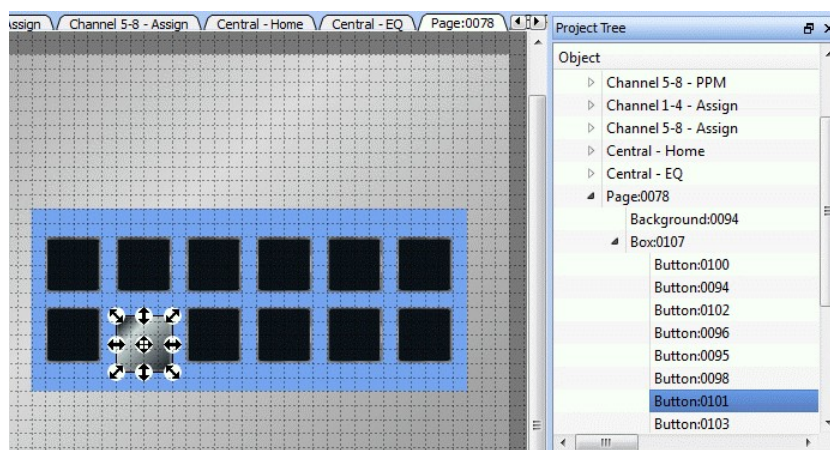
The parent (**Box**) and its subnodes (**Buttons**) now move together if you click and drag the **Box** in the design area:



If you [resize](#) the parent element, then only the parent resizes.

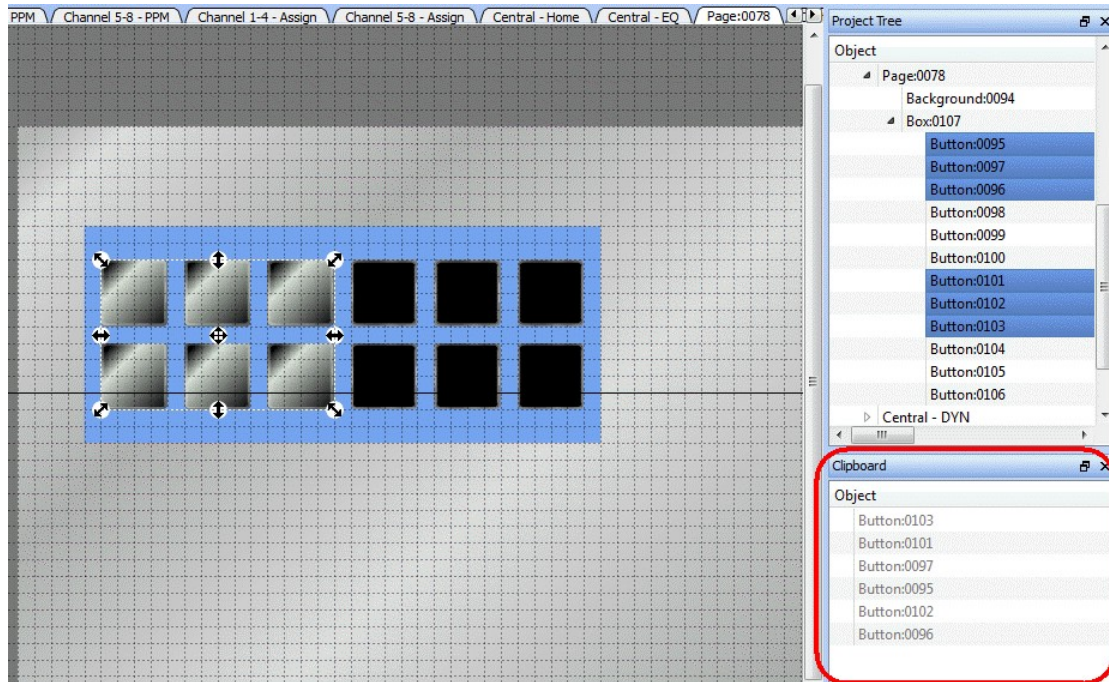
Press CTRL (on your PC keyboard) while resizing to resize both the parent and its subnodes. Press CTRL + SHIFT while resizing to also maintain their aspect ratio.

At any time, you can select an individual element in order to change its position and/or [Properties](#):



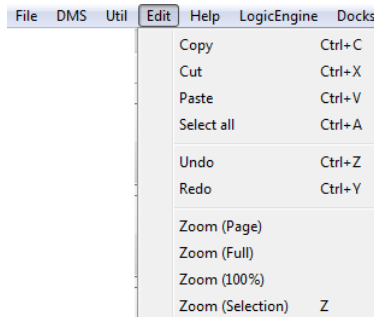
### 6.13.7 Copy, Cut and Paste

On your PC keyboard, press **CTRL + C** to copy or **CTRL + X** to cut the current selection. The copied elements appear in the **Clipboard docking panel** as confirmation:



Press **CTRL + V** to paste the elements from the **Clipboard**. Pasted elements are placed in exactly the same position as the original, and include all properties and functionality.

**Copy, Cut and Paste** can also be accessed from the [Edit menu](#):



You can use **Copy** and **Paste** to copy an existing page. The pasted page appears to the right of any existing pages, and takes the same name as the copied page. Use the page [Properties](#) to rename the page and adjust its functionality.



### 6.13.8 Using Multipage Objects

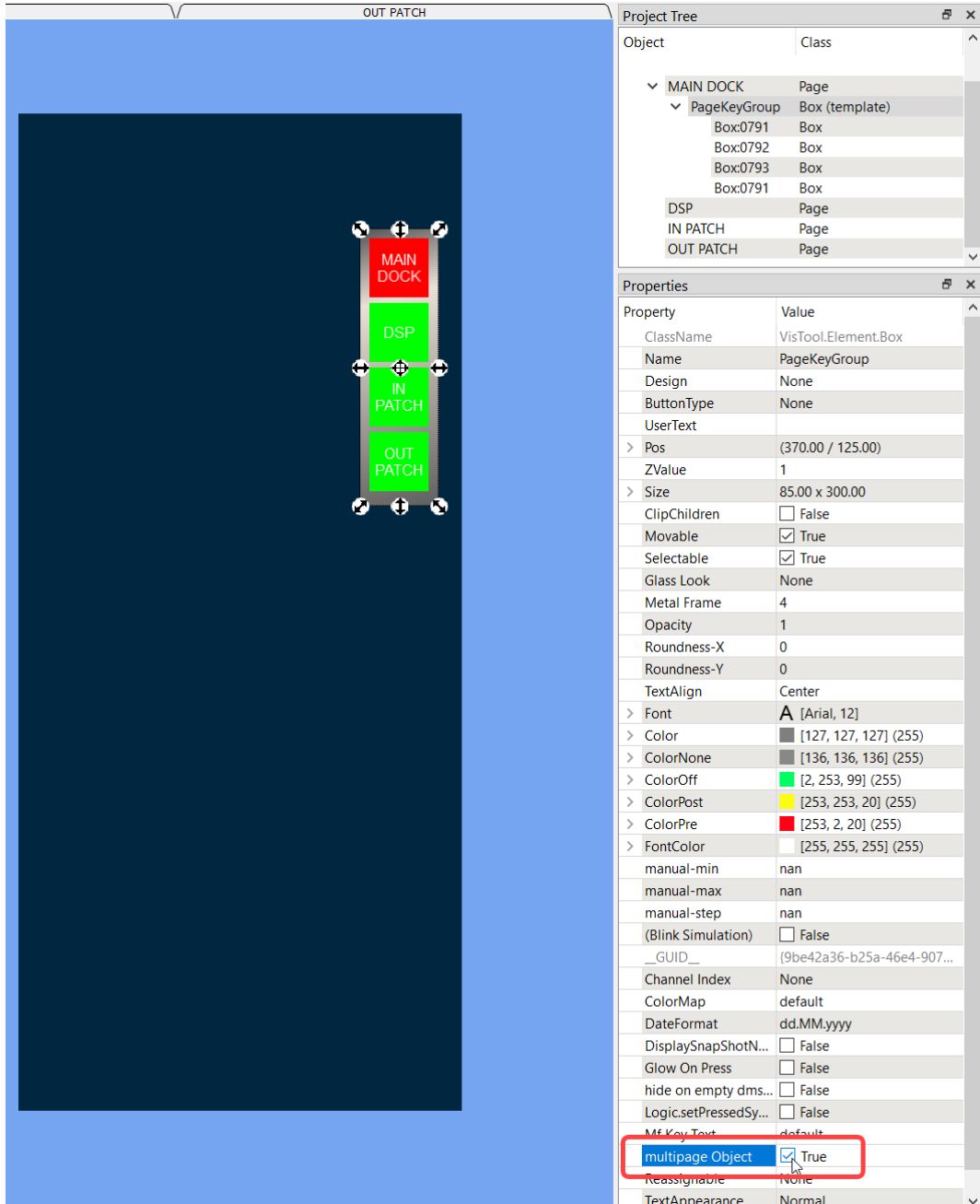
The **Multipage Object** property can be used to repeat an element (or group of elements) across several pages. It can be enabled for [Box](#) and [Overlay](#) elements, and is ideal for page switching keys or dock bar elements such as Timers and PPMs. Once enabled, all occurrences of the element can be customized from just one page. Using this feature also minimizes the size of the project and improves performance.

1. Start by adding the elements to a page.

In our example, a nested group of four page keys has been created using a Box.

2. Select the element (or group of elements) and enable the **multipage Object** property.

Notice that in the [Project Tree](#), the element Class changes to a template.

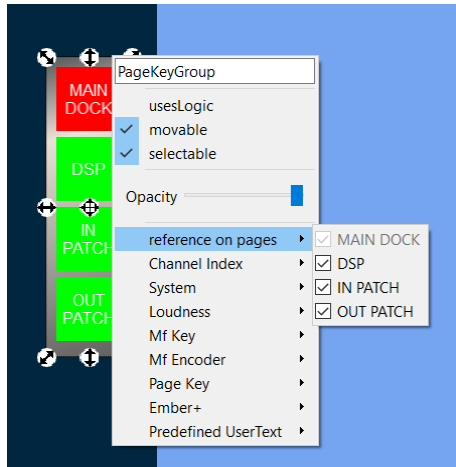


The screenshot shows a VisTool configuration window with a dark blue background. A vertical dock bar is visible, containing four buttons: 'MAIN DOCK' (red), 'DSP' (green), 'IN PATCH' (green), and 'OUT PATCH' (green). The dock bar is labeled 'OUT PATCH' at the top. To the right, the 'Project Tree' panel shows a hierarchy: 'MAIN DOCK' (Page) containing a 'PageKeyGroup' (Box (template)) which contains four 'Box' elements (Box:0791, Box:0792, Box:0793, Box:0791). Below the Project Tree is the 'Properties' panel, which shows various settings for the selected element. The 'multipage Object' property is highlighted with a red box and is set to 'True'.

Property	Value
ClassName	VisTool.Element.Box
Name	PageKeyGroup
Design	None
ButtonType	None
UserText	
Pos	(370.00 / 125.00)
ZValue	1
Size	85.00 x 300.00
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Glass Look	None
Metal Frame	4
Opacity	1
Roundness-X	0
Roundness-Y	0
TextAlign	Center
Font	A [Arial, 12]
Color	[127, 127, 127] (255)
ColorNone	[136, 136, 136] (255)
ColorOff	[2, 253, 99] (255)
ColorPost	[253, 253, 20] (255)
ColorPre	[253, 2, 20] (255)
FontColor	[255, 255, 255] (255)
manual-min	nan
manual-max	nan
manual-step	nan
(Blink Simulation)	<input type="checkbox"/> False
_GUID_	{9be42a36-b25a-46e4-907...
Channel Index	None
ColorMap	default
DateFormat	dd.MM.yyyy
DisplaySnapshotN...	<input type="checkbox"/> False
Glow On Press	<input type="checkbox"/> False
hide on empty dms...	<input type="checkbox"/> False
Logic.setPressedSy...	<input type="checkbox"/> False
Mf Key Text	default
<b>multipage Object</b>	<input checked="" type="checkbox"/> True
Reassignable	None
TextAppearance	Normal

- Right-click on the element (or group of elements) and use the **reference on pages** drop-down menu to choose where the template should appear.

The current page (e.g. MAIN DOCK) is automatically selected.



The element (or group of elements) can now be customized. For example, to change the function, position, color, etc. Any changes made are applied equally to all instances.

Note that deleting the template will delete the element (or group of elements) from all pages.

If you disable the **multipage Object** property, then the element (or group of elements) is reinstated as a single object on the original page.

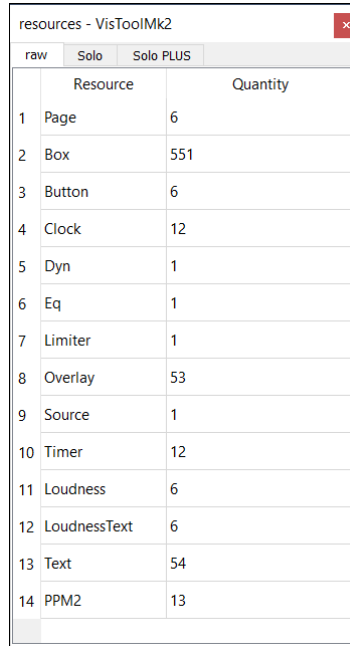
### 6.14 Monitoring the Resources

The **Resources** docking panel can be used to check the resources used by the project.

1. Select **Docks** -> **resources** to open the panel.

The panel can be resized, floated, docked or stacked in the [usual](#) manner.

2. The **raw** tab can be used for general information, and shows the number of resources used by the project.



resources - VisToolMk2		
raw	Solo	Solo PLUS
	Resource	Quantity
1	Page	6
2	Box	551
3	Button	6
4	Clock	12
5	Dyn	1
6	Eq	1
7	Limitier	1
8	Overlay	53
9	Source	1
10	Timer	12
11	Loudness	6
12	LoudnessText	6
13	Text	54
14	PPM2	13

3. The **Solo** and **Solo Plus** tabs can be used to check if the project is suitable for a VisTool Solo or Solo Plus license (described [later](#)).

## 6.15 Using the Project Tree

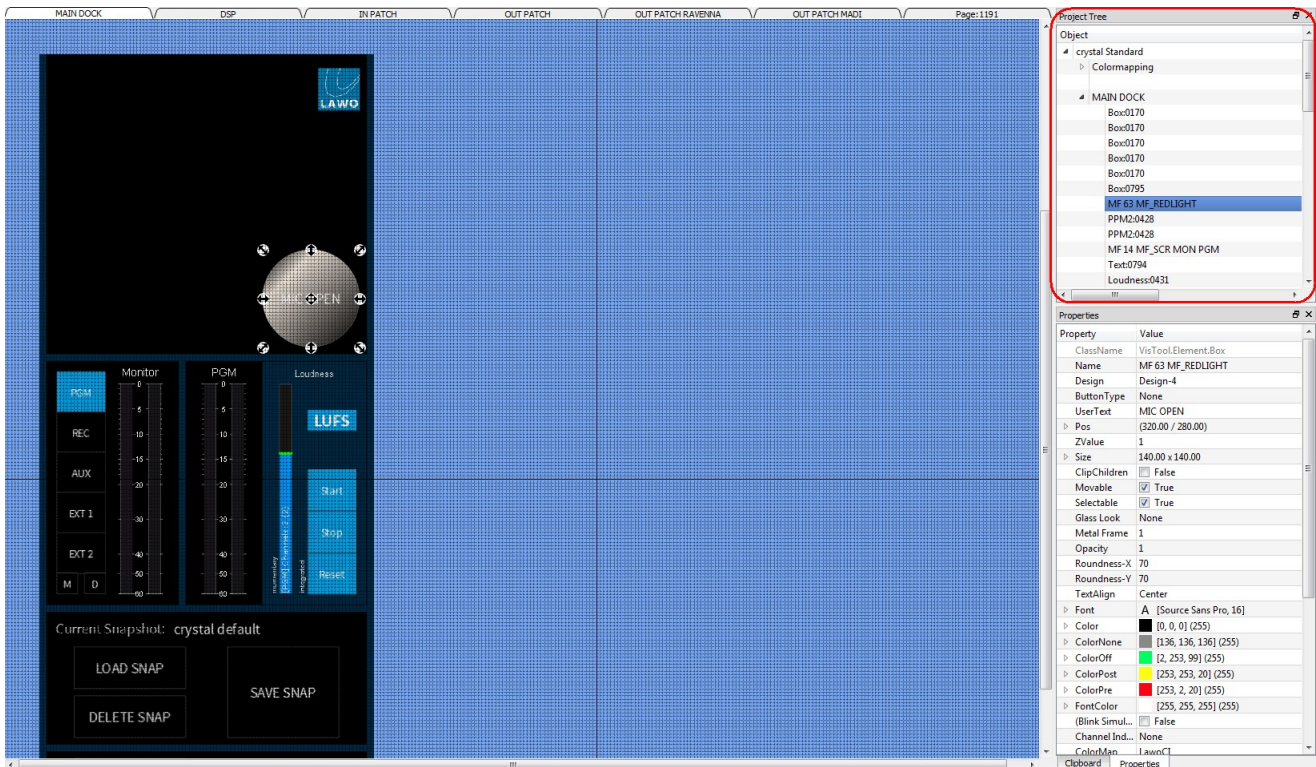
The **Project Tree** shows all the pages and elements used in the project, and provides another way to select elements.

If you cannot see the **Project Tree**, check the docking panel [view options](#).

1. Open and close the branches of the **Project Tree** to view the project's structure.

Each Project can contain multiple Pages, and each Page may contain multiple elements. In our example, the Page named **MAIN DOCK** is open.

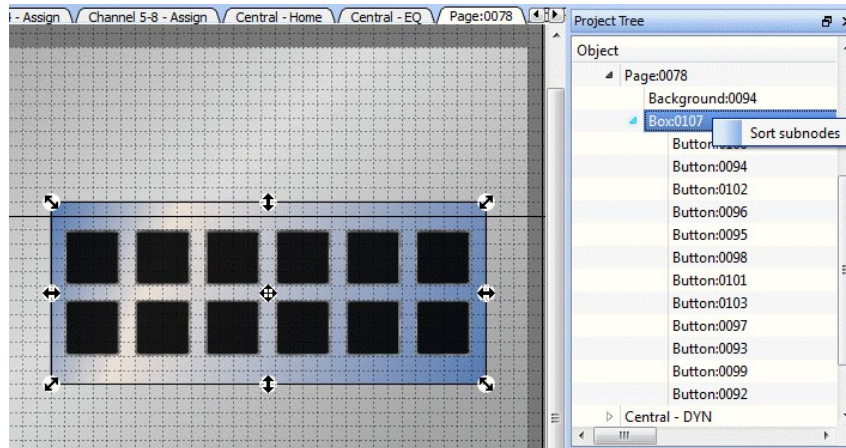
2. Click on an element from the **Project Tree** to select it - for example, the Box element named **MF 63 MF\_REDLIGHT**:



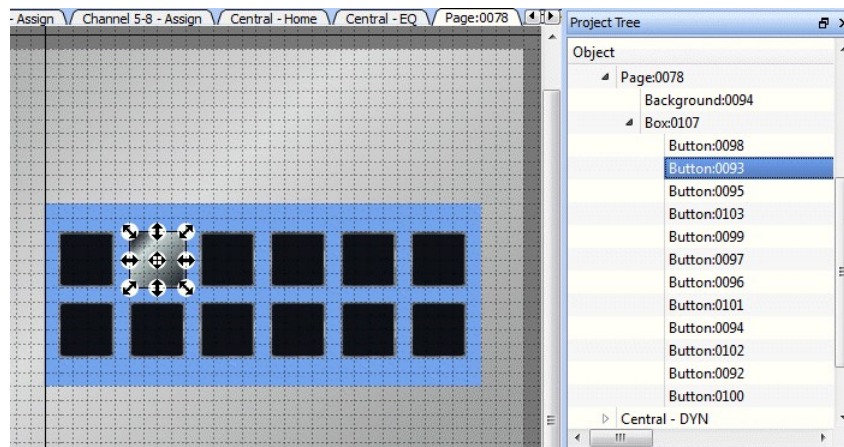
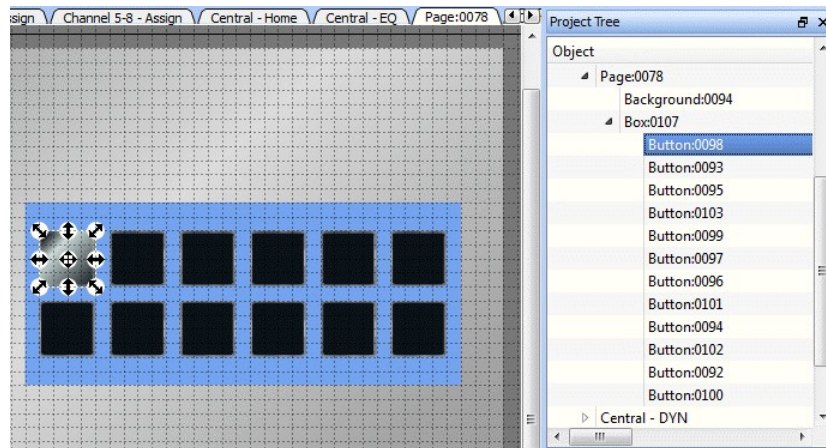
You can now [position](#) the element, or edit its [Properties](#) in the usual manner.

## 6. Editing a VisTool Configuration

- To sort elements within the tree, right-click on a parent element (e.g. **Box 0107**) and select **Sort subnodes**:



All child elements are reordered from top to bottom using their **Z value** (in front/behind), then **Y and X** coordinates. In our example, this makes it easy to select each **Button** element from left to right and top to bottom in order:



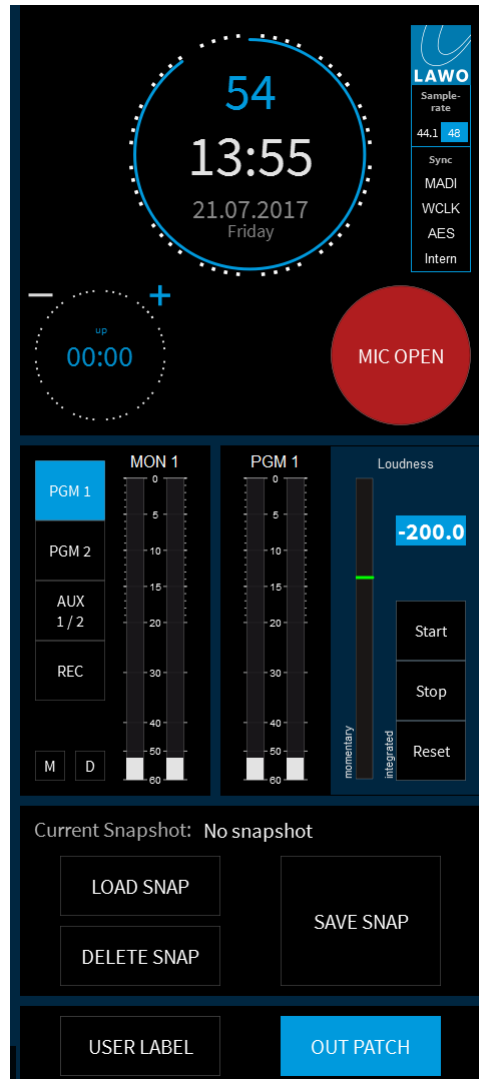
You can use the ALT + UP/DOWN/LEFT/RIGHT arrow keys (on your PC keyboard) to select elements quickly within the Project Tree.

### 6.16 Previewing and Testing the Design

1. Press CTRL + T (on your PC keyboard) to preview the design.

Your computer will now show the current page on your main screen (as it will appear in VisTool runtime):

*MAIN DOCK (in preview mode)*



If the connected system is online, then you can test the functionality by clicking the on-screen buttons, sending audio to a meter, etc.

2. Press PAGE UP and PAGE DOWN to scroll through the other pages defined in the VisTool Editor project.
3. Press CTRL + T again to return to edit mode.

The online mode times out after five minutes and, therefore, VisTool Editor cannot be used for runtime operation.

If you wish to use the project in runtime, please see [Configuring a PC for VisTool Runtime](#).

### 6.17 Using Color Maps

Color maps can be used to customize the 16 colors triggered by a remote system. For example, to change the color of an element from an MF Key (or other logical control signal).

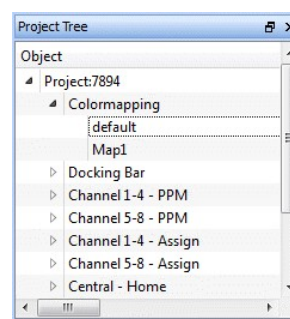
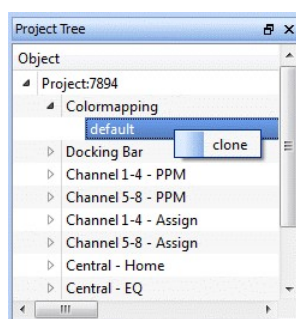
Multiple color maps can be created (from the **Project Tree**) and assigned to individual elements (from the [Properties](#) panel). Every project includes a **default** color map that can only be edited from the [Global Settings](#) window.

Note that, for **Box** elements, color maps can also be applied to the displayed text using the **Font Color Map** property.

#### ➤ To define a new color map:

1. From the [Project Tree](#), open the **Colormapping** branch.
2. Right-click on an existing entry - e.g. **default** - and select **clone**.
3. Enter a name (e.g. **Map 1**) and select **OK**.

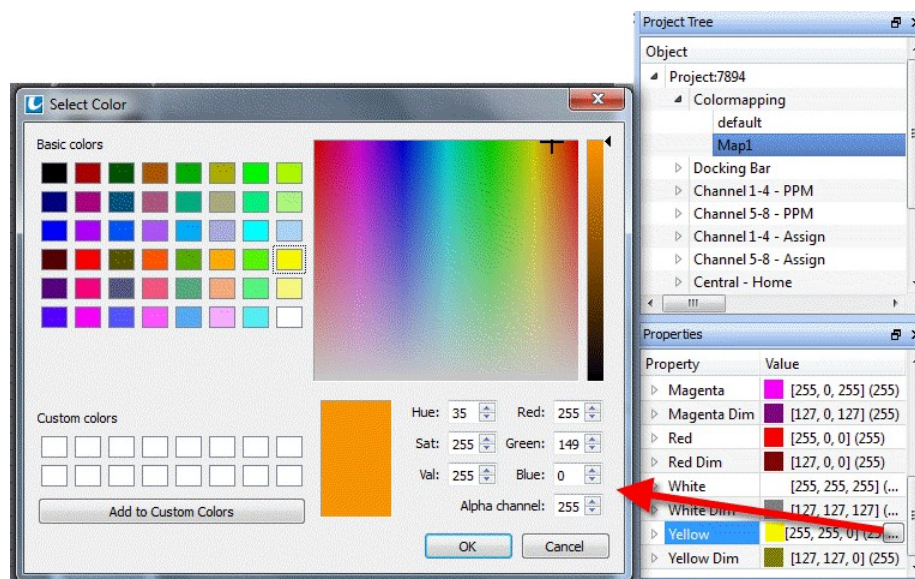
The color map is added to the **Project Tree**.



4. Now select the color map (in the **Project Tree**), and edit its **Properties** to define each color.

If you cannot see the **Properties** panel, check the docking panel [view options](#).

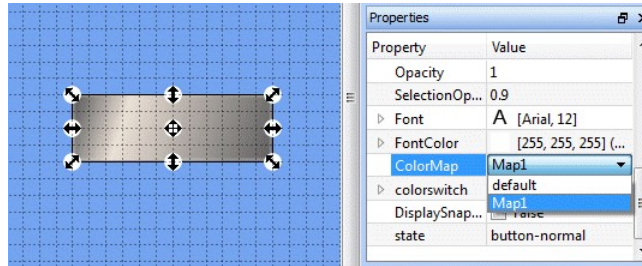
In the example below, we are changing the color used for **Yellow**. Each color is referenced by name only, and so the values specified for **Yellow** can be anything!



Once a color map is created, it can be assigned to one or more elements.

➤ **To assign a color map to an element:**

1. Select the element and open its [Properties](#) panel.
2. Use the **ColorMap** drop-down menu to assign one of the color maps created earlier.



The element will now use the [Color](#) property for its normal state, and the 16 colors defined in the **ColorMap** when the state is changed from the remote system.

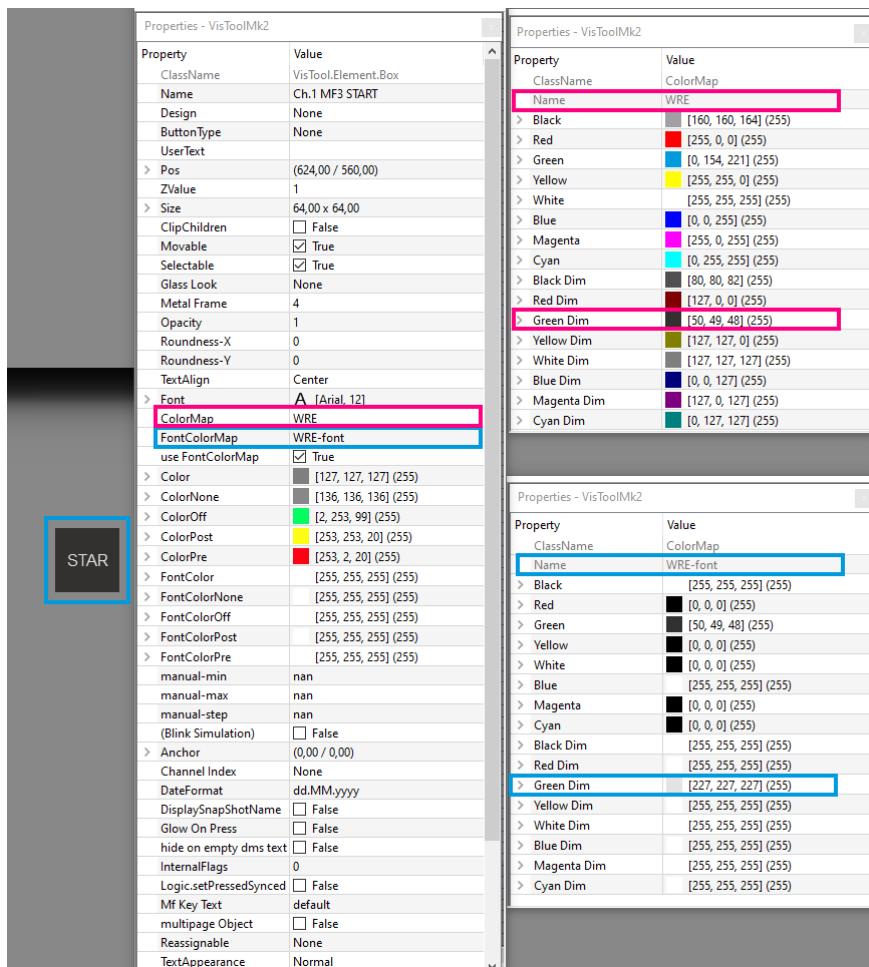
➤ **Changing the Color of Displayed Text (Box Elements only)**

For [Box](#) elements, color maps can also be applied to the displayed text so that both the color of the element and its text can be changed separately.

In this instance, turn on the **use FontColorMap** option and use the **FontColorMap** field to assign one of the color maps created earlier. Every project includes a **default-font** color map that can only be edited from the [Global Settings](#) window. This can be cloned to create color maps specifically for text if you wish. Right-click on the **default-font** color map and follow the customization instructions described earlier.

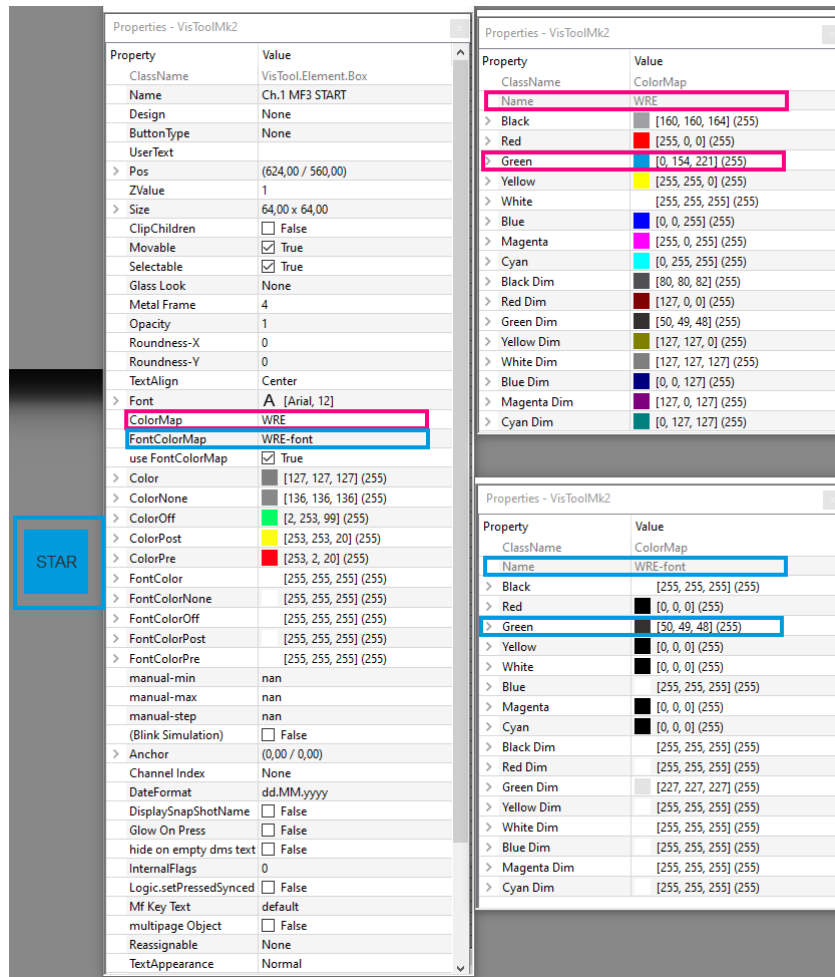
In the example below, if the remote system triggers a change of color from **Green Dim** to **Green**, then the **Box** will change color from black to blue and its text will change color from white to black.

*Customization of "Green Dim"*



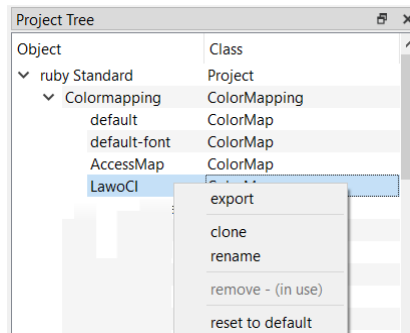


Customization of "Green"



➤ Other Color Map Operations

Color Map (right-click options)



Right-click on a color map (in the **Project Tree**) to perform the following operations:

- **export** - exports the color map to a ".xml" file. This can be imported into another project by right-clicking on **Colormapping** and selecting **import**.
- **clone** - creates a copy of the selected color map.
- **rename** - renames the selected color map.
- **remove** - removes the color map from the project. This option can only be selected if the Color Map is not in use.
- **reset to default** - resets the selected color map to the default colors.

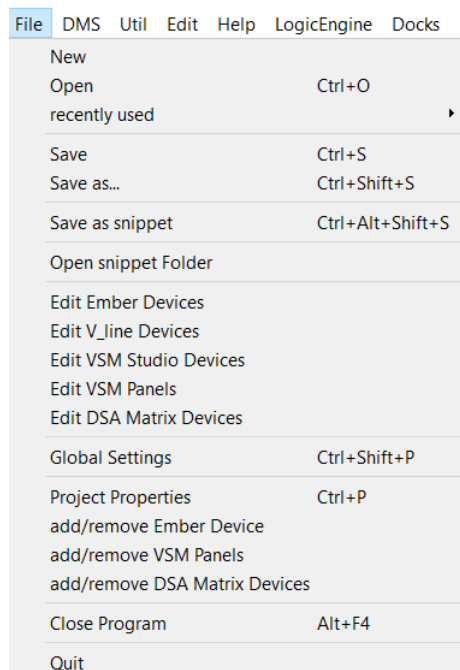
## 6.18 Next Steps

Once the VisTool Editor project is complete, you can generate a start script using [VisTool Sessions](#). You should do this on the VisTool runtime PC so that the correct monitor options can be configured.

If this is a different PC, then make sure you copy all of the relevant files: the VisTool Editor project (**.vistool** file) and the accompanying ON-AIR Designer project (**.db3** file). Once you have transferred the files, you are ready to configure the session(s).

## 6.19 The Main Menus

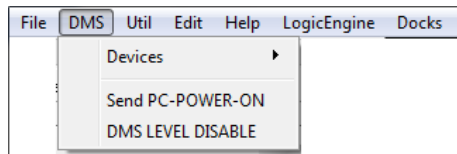
### 6.19.1 File



Use the **File** menu options to:

- **New** - create a [New](#) VisTool project.
- **Open** - [open](#) an existing VisTool project.
- **recently used** - open a recent project from the historic entries.
- **Save** and **Save as** - [save](#) the VisTool project.
- **Save as snippet** - [save](#) the current selection as a snippet.
- **Open snippet Folder** - [open](#) Windows Explorer to select a snippet from the Snippets directory (C:\Program Data\DSA\VisToolMK2\Snippets).
- **Edit Ember Devices** - [edit](#) the Ember+ providers known to VisTool Editor.
- **Edit V\_line Devices** - [edit](#) the V\_line devices known to VisTool Editor.
- **Edit VSM Studio Devices** - [edit](#) the VSM Studio devices known to VisTool Editor.
- **Edit VSM Panels** - [edit](#) the VSM Panels known to VisTool Editor.
- **Edit DSA Matrix Devices** - [edit](#) the DSA Matrix devices known to VisTool Editor.
- **Global Settings** - open the [Global Settings](#) for the VisTool Editor application.
- **Project Properties** - open the [Project Properties](#) for the current project.
- **add/remove Ember Device** - [add or remove](#) Ember+ providers to the current project.
- **Add VSM Panels** - [add or remove](#) VSM Panels to the current project.
- **add/remove DSA Matrix Device** - [add or remove](#) DSA Matrices to the current project.
- **Close Program** or **Quit** - close the VisTool Editor application. If you have made changes and not saved, you will be prompted to save before closing.

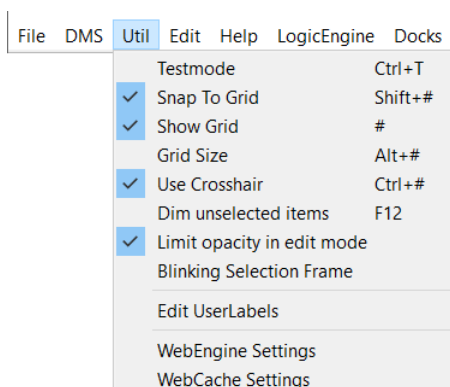
### 6.19.2 DMS



Use the **DMS** menu options to:

- **Devices** - select a DMS device (as defined by **VisAdmin**, see [Configuring the DMS Connections](#)). The option **offline** can be selected to stop the transfer of outgoing messages to the DMS service.
- **Send PC-POWER-ON** - tells the connected device to send its complete status.
- **DMS LEVEL DISABLE** - disables DMS level information (metering) from the connected device. You can use this option to reduce the amount of network traffic if metering is not required.

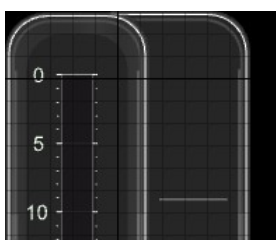
### 6.19.3 Util



Use the **Util** menu options to select:

- **Testmode** - switches to test mode so that you can [test](#) the configuration. Press CTRL + T (on your PC keyboard) to return to edit mode.
- **Snap to Grid** - elements snap to the [grid](#) when adjusting their position.
- **Show Grid** - turns the [grid](#) on or off (shown below)
- **Grid Size** - opens the [Grid Setup](#) window.
- **Use Crosshair** - turns the crosshair on or off (shown below).
- **Dim unselected items** - dims all items other than the one selected (shown below).
- **Limit opacity in edit mode** - allows you to see elements, with an opacity of 0, which would otherwise be hidden. The option applies to VisTool Editor only; once the project is opened in runtime, the real opacity values are used.
- **Blinking Selection Frame** - makes the selected frame flash (blink) for better visibility.
- **Edit UserLabels** - opens the [user labels](#) editor.
- **WebEngine Settings** - opens the web engine settings (applicable to [Web Elements](#)).
- **WebCache Settings** - opens the web cache settings (applicable to [Web Elements](#)).

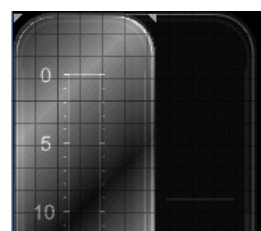
*Grid & Crosshair ON*



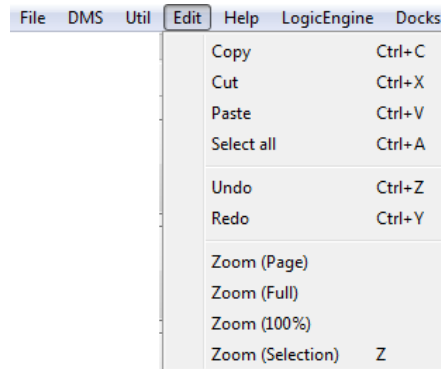
*Grid & Crosshair OFF*



*Dim ON*



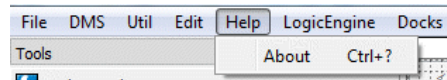
### 6.19.4 Edit



Use the **Edit** menu options to:

- **Copy, Cut and Paste** - copy, cut or paste elements as described [earlier](#).
- **Select All** - selects all elements on the current page.
- **Undo/Redo** - undo or redo the last operation. See [Undo and Redo](#).
- **Zoom** - to the Page, Full-screen, etc. See [Zooming In and Out](#).

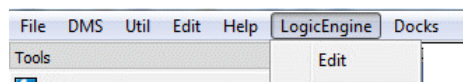
### 6.19.5 Help



Select **About** to check the release version of the VisTool software and other information about your project and file paths.

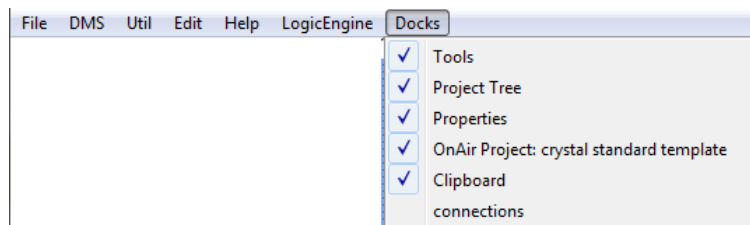
Select **OK** to close the window.

### 6.19.6 Logic Engine



This main menu provides access to the [Logic Engine](#) editor. From here you can program logical conditions for Ember+ parameter values.

### 6.19.7 Docks



Use the **Docks** menu options to show or hide the various docking panels, see [Arranging the Workspace](#).

### 6.20 Keyboard Shortcuts

Key	Function
CTRL + ?	Opens the <a href="#">Help</a> window.
ALT + F4	<a href="#">Closes</a> the VisTool application.
PgUp	Opens the previous page, see <a href="#">Page Navigation</a> .
PgDown	Opens the next page.
Home	Opens the first page.
End	Opens the last page.
CTRL + O	<a href="#">Opens</a> an existing VisTool project.
ALT + Left Arrow Key	Selects the next parent element within the <a href="#">Project Tree</a> .
ALT + Right Arrow Keys	Selects the next child element within the <a href="#">Project Tree</a> .
ALT + Up/Down Arrow Keys	Selects the next element (up or down) within the current branch of the <a href="#">Project Tree</a> .
CTRL + SHIFT + O	<a href="#">Opens</a> the most recent VisTool project.
CTRL + S	<a href="#">Saves</a> the project.
CTRL + SHIFT + S	Opens the <a href="#">Save As</a> dialogue box.
CTRL + ALT + SHIFT + S	<a href="#">Saves</a> the selected elements as a snippet.
CTRL + ALT + SHIFT + O	<a href="#">Opens</a> the snippet dialogue box.
CTRL + P	Opens the <a href="#">Project Properties</a> window.
CTRL + SHIFT + P	Opens the <a href="#">Global Settings</a> window.
CTRL + T	Switches between <a href="#">Test mode</a> and Edit mode.
SHIFT + #	Toggles the <a href="#">Snap to Grid</a> option.
#	Toggles the <a href="#">Show/Hide Grid</a> option.
ALT + #	Opens the <a href="#">Grid Setup</a> window.
CTRL + #	Toggles the <a href="#">Use Crosshair</a> option.
F12	Toggles the <a href="#">Dim Unselected Objects</a> option.
CTRL + C	Copies the selected elements, see <a href="#">Copy, Cut and Paste</a> .
CTRL + X	Cuts the selected elements
CTRL + V	Pastes the copied elements onto the current page.
CTRL + A	Selects all elements on the current page.
CTRL + Z	Undo (the last action), see <a href="#">Undo and Redo</a> .
CTRL + Y	Redo (the previous action).
Z	<a href="#">Zooms</a> in or out to the current selection(s).

## 7. VisTool Solo & Solo Plus

---

This chapter describes VisTool Solo and VisTool Solo Plus.

Topics include:

- [Introduction](#)
- [Resource Limitations](#)
- [Generating a Start Script](#)
- [Runtime Operation](#)



### 7.1 Introduction

**VisTool Solo** and **VisTool Solo Plus** are licensing options for VisTool runtime that provide a cost-effective alternative to VisTool Unlimited. They support custom configurations, designed by VisTool Editor, but with restrictions on the number and type of resources.

## 7.2 Resource Limitations

The table below shows the maximum number of resources for each license.

Resource Type	VisTool Solo	VisTool Solo Plus
Generic elements (Background, Graphics, Text)	10	40
Buttons (Box, Button)	32	128
Levels (Fader, Slider, Rotary)	4	24
Stereo Meters (PPM, PPM2, EmBER+ peak meter, EmBER+ PPM, EmBER+ VU)	2	8
Correlation Meters	2	8
Loudness Meters	2	8
Loudness Value Labels	2	8
LUFS Value Labels	2	8
PPM Value Labels	2	8
Event Timers	3	3
Advanced (WebView)	1	4
Pages	2	8
Overlays	2	8















In addition, the following elements are prohibited (invalid):

- Graphical DMS elements (Source, Bus, Dyn, Lim, EQ, etc.)
- V\_\_line thumbnails
- VSM-Buttons















VisTool Solo Plus also supports a large time-of-day clock.

### 7.2.1 Monitoring the Resources

While designing a project in VisTool Editor, it is the responsibility of the customer to ensure the limits are not exceeded. This can be done by opening the [Resources](#) docking panel and selecting the correct tab: either **Solo** or **Solo Plus**.

raw	Solo	Solo PLUS	Quantity
1	Generic		5/10
2	Button		112/32
3	Level		0/4
4	Meter		6/2
5	Correlation		0/2
6	Loudness		2/2
7	LoudnessText		2/2
8	LUFSText		0/2
9	PPMLabel		0/2
10	Timer		2/3
11	Advanced		0/1
12	Page		2/2
13	Overlay		6/2
14	Invalid		

raw	Solo	Solo PLUS	Quantity
1	Generic		5/40
2	Button		112/128
3	Level		0/24
4	Meter		6/8
5	Correlation		0/8
6	Loudness		2/8
7	LoudnessText		2/8
8	LUFSText		0/8
9	PPMLabel		0/8
10	Timer		2/3
11	Advanced		0/4
12	Page		2/8
13	Overlay		6/8
14	Invalid		

Each line is color-coded to indicate: blue = ok; red = limit exceeded. The counters on the right show how many resources are left out of the maximum permitted. Or, if a line is red, by how many the maximum is exceeded.

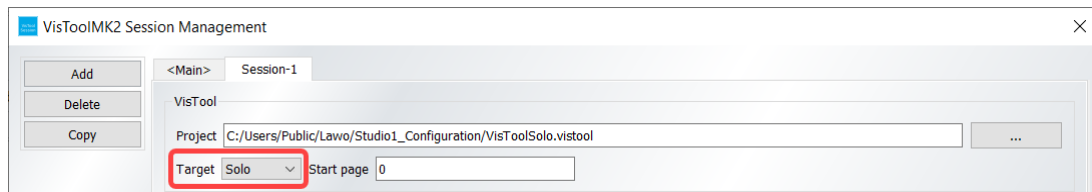
If a limit is exceeded, then you will need to delete the element(s), or purchase a higher license. In the example above, the project can be used with a VisTool Solo Plus license, but not VisTool Solo.

There are no restrictions regarding the setting up or editing of elements, as long as the maximum limits are observed.

### 7.3 Generating a Start Script

Once the project has been saved, use VisTool Sessions to generate a start script. When completing the Session Parameters, the **Target** field must be set to either **Solo** or **Solo Plus**. All other fields should be completed in the [usual](#) manner.

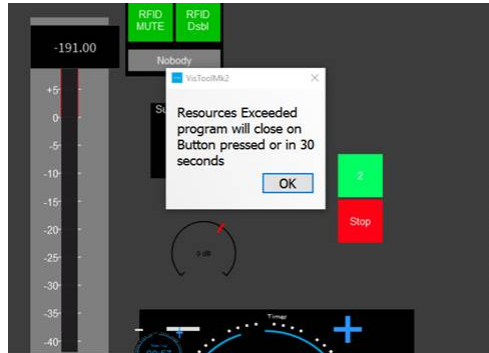
*Session Parameters (for VisTool Solo project)*



## 7.4 Runtime Operation

On the runtime PC, the correct license must be [activated](#). If not, then following warning message will appear (after 1 minute) if the session exceeds the permitted resources.

*"Resources Exceeded" warning message*



In this instance, the session will close, either on the next button press or after 30 seconds. There are two ways to continue: either activate a higher resource license, or modify the VisTool project (using VisTool Editor) and generate a new start script (using VisTool Sessions).

## 8. VisTool Elements

---

This chapter describes all of the elements supported by VisTool Editor.

Topics include:

- [Introduction](#)
- [Background](#)
- [Buttons](#)
- [Fader](#)
- [PPM](#)
- [Loudness](#)
- [Sources](#)
- [Dynamic](#)
- [Timer](#)
- [Overlay](#)
- [Web](#)
- [Simple](#)
- [Video](#)
- [Other](#)
- [User](#)
- [Snippets](#)

## 8.1 Introduction

Elements are [added](#) to a page from the **Tools** docking panel. They can be used as follows:



- [Background](#) – add a background to the VisTool page
- [Buttons](#) – create on-screen buttons which can action a function or provide signaling.
- [Fader](#) – create on-screen faders that can control physical faders on a Lawo console or real values from an Ember+ provider.
- [PPM](#) – create on-screen metering and visualization for channel states and parameters.
- [Loudness](#) – configure loudness metering bargraphs and/or integrated loudness measurements.
- [Sources](#) – display parameters for the source "in access".
- [Dynamic](#) – display DSP parameters for the source "in access": Delay, Dynamics, EQ, Limiter, etc.
- [Timer](#) – create on-screen timers or a system clock.
- [Overlay](#) – create an area that appears temporarily such as a pop-up menu.
- [Web](#) – view a web page within VisTool.
- [Simple](#) – this category includes text elements (to annotate your design), the system clock (as a numeric), box elements (ideal for framing other elements or creating on-screen buttons), and Sliders and Rotary Encoders (for controlling variable parameters).
- [Video](#) – display thumbnails from a V\_\_line device within the VisTool page.
- [Other](#) – add other predefined elements to customize the design.

In addition, it is possible to add [user](#) graphics to customize the design, and create [snippets](#) to reuse collections of elements.

## 8.2 Background

These elements can be used to add a background to the VisTool page.



There are three **Background** elements to choose from offering a choice of aspect ratio:

- **Standard 16-10**
- **Standard 16-9**
- **Standard 4-3**

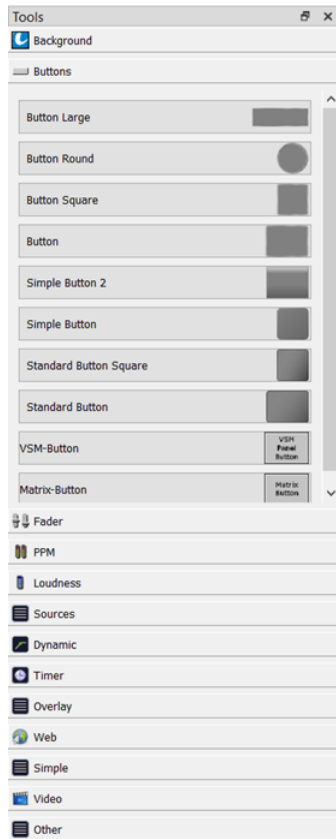
These elements create a background for the VisTool page. They are configured from the **Properties** panel. All options are covered in the [general properties](#) topic.



## 8.3 Buttons

These elements can be used to create on-screen [buttons](#) that action a function or provide signaling.

Button elements used fixed graphics and therefore do not scale proportionally. If you wish to scale the graphic or adjust its visual appearance, it is recommended to use [Box](#) elements.



There are several "regular" elements offering a choice of button size and shape:

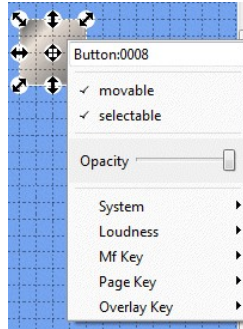
- **Button Large**
- **Button Round**
- **Button Square**
- **Button**
- **Simple Button 2**
- **Simple Button**
- **Standard Button Square**
- **Standard Button**

In addition, you will find two special elements:

- **VSM-Button** - control or monitor VSM Panel buttons (configured by an external VSM server).
- **Matrix-Button** - control crosspoints within an external matrix.

### 8.3.1 Button Functions (Right-click)

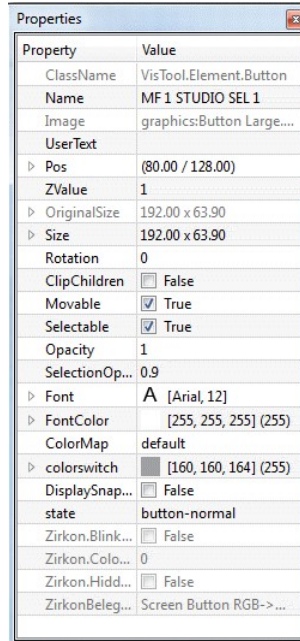
A regular **Button** element can be linked to any of the following functions:



<b>System</b>	a function which affects the application: minimize, fullscreen, close, etc.
<b>Loudness</b>	an <a href="#">integrated loudness measurement</a> command: Start, Stop, etc.
<b>MF Key</b>	an <a href="#">MF Key</a> defined by the ON-AIR Designer project.
<b>Page Key</b>	a VisTool Editor <a href="#">Page Key</a> for on-screen page navigation.
<b>Overlay Key</b>	a key to open an <a href="#">Overlay</a> element.

### 8.3.2 Button Properties

In addition to the [general properties](#), **Button** elements support the following special properties:



<b>Display SnapshotName</b>	Tick this box to activate the <a href="#">DisplaySnapshotName</a> option.
<b>state</b>	<p>Each "push" button has two states: pressed and not pressed which are controlled by the connected hardware. Using the state menu, you can see what each button state will look like in VisTool:</p> <ul style="list-style-type: none"> <li>• button-normal = not pressed.</li> <li>• button-pressed = pressed.</li> </ul>

### 8.3.3 VSM-Buttons

The **VSM-Button** element can operate buttons on a VSM Panel configured by an external VSM server. This allows you to control or monitor VSM Panel buttons directly from a VisTool screen.

To support this feature you will need a **VisTool Unlimited** license plus the **VSM Button Panel AddOn**.

#### VisTool & VSM

The functionality of each button is defined by the VSM server and not by VisTool Editor. This includes not only the button's function but also the image and text displayed when the button is active/inactive. The button size and layout can be adjusted in the usual manner. This allows specific page layouts to be created for VSM buttons, or VSM buttons to be combined with other VisTool elements.

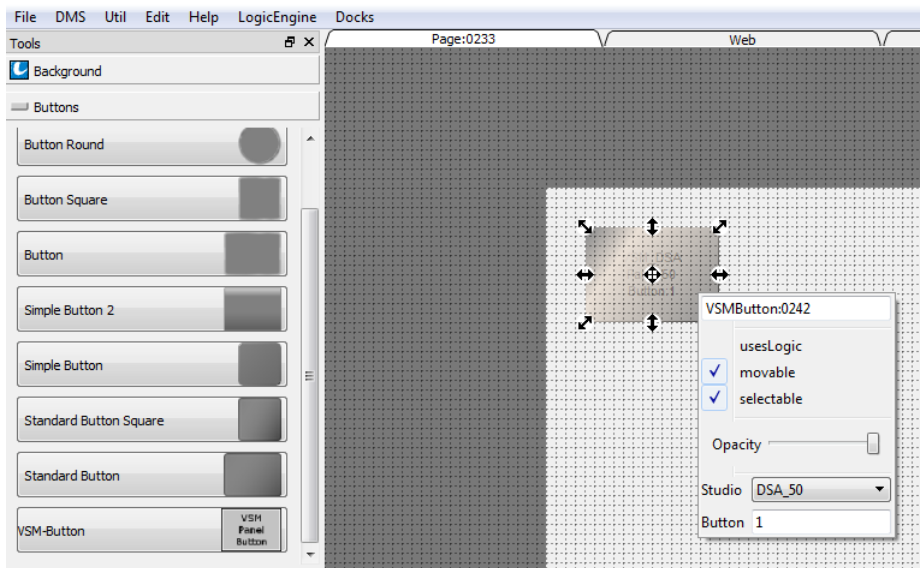
A single VisTool instance can connect to up to six VSM servers at the same time, and main/redundant VSM servers are supported.

#### Preparation & Configuration

The VSM server(s) must be connected to the Lawo system network. You will need to know the hostname or TCP/IP Address and Port Number of each VSM server, and the VSM Panel IDs which are supported.

Then use VisTool Editor to perform the following configuration steps:

1. Specify the VSM server(s) in the [VSM Studio Devices](#) list.
2. [Define](#) the **Panel ID** and number of available buttons (**Button #**) for each VSM panel.
2. [Add](#) the VSM Panel(s) to the current VisTool project.
3. [Assign](#) each **VSM-Button** element to a specific VSM Panel ID and Button number:

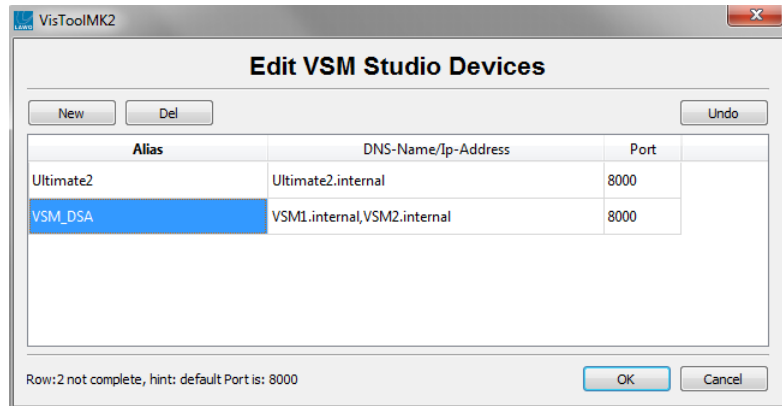
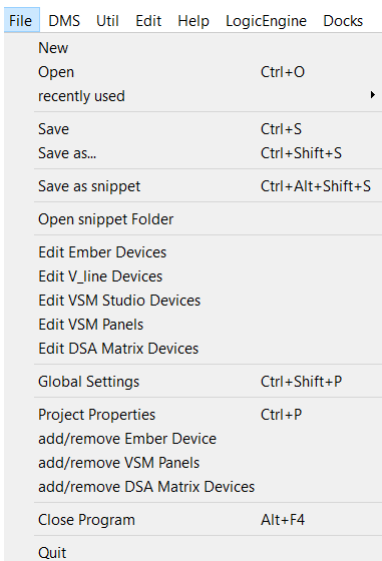


If you wish, you can use the [Logic Engine](#) to program logical conditions for VSM-Button elements. The next few pages describe the steps above in more detail.

### The VSM Studio Devices List

This list defines the TCP/IP settings of each VSM server on the network.

1. Select **File -> Edit VSM Studio Devices** to open the list:



2. Select **New** to add a new entry.
3. Click in each of the fields to enter an **Alias** name, the **DNS-Name** or **IP-Address**, and **Port** number of each VSM server - take care to press Enter to confirm each field.

If main and redundant VSM servers are in use, then create a single entry and specify both servers in the **DNS-Name/Ip-Address** field (separated by a comma) - for example:

- **Alias** = VSM\_DSA
- **DNS-Name** of main server = VSM1.internal
- **DNS-Name** of redundant server = VSM2.internal

4. Use **Del** to delete an existing line. **Undo** will undo all changes. Select **OK** to save all changes and exit.

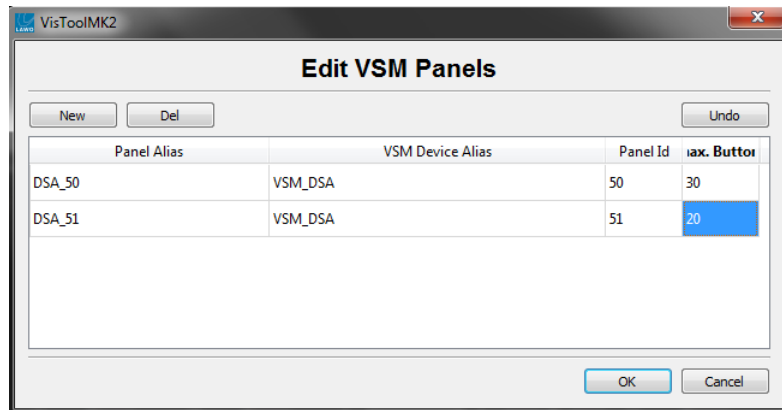
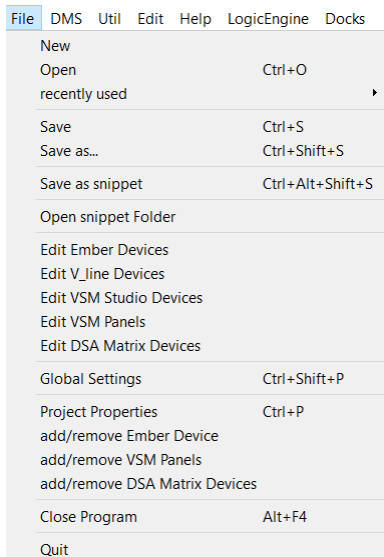
In our example, two VSM server clusters have been configured; the second entry supports a main and redundant server.

The VSM Studio devices list is stored in the **vsmStudio\_hosts.ini** file. This may be shared between VisTool instances. See [Using a Shared File path for ".ini" Files](#). You can also deactivate a host by [editing](#) the file.

## The VSM Panels List

Once you have defined the VSM device(s), the next step is to define the **Panel ID** and number of available buttons for each VSM panel. This is handled by the **VSM Panels** list:

1. Select **File -> Edit VSM Panels** to open the list:



2. Select **New** to add a new entry.
3. Click in each of the fields to enter a **Panel Alias** name, the **VSM Device Alias** (as defined in the [VSM Studio Devices](#) list), the **Panel Id** (of each panel supported by the server) and its number of available buttons in **Max. Button** - take care to press Enter to confirm each field.

Note that the number of available buttons (up to 1024) should match the number of buttons configured in the VSM server.

4. Use **Del** to delete an existing line. **Undo** will undo all changes. Select **OK** to save all changes and exit.

In our example, we have two panels, one with 30 and the other with 20 buttons.

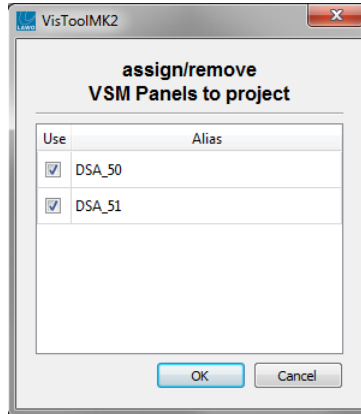
The VSM Panels list is stored in the **vsmStudio\_Panels.ini** file. This may be shared between VisTool instances. See [Using a Shared File path for ".ini" Files](#). You can also deactivate a host by [editing](#) the file.

### Adding a VSM Panel to the VisTool Project

The next step defines which VSM Panels are available in the current project.

1. Select **File -> Add VSM Panels** and tick the 'Use' boxes to make your selections.

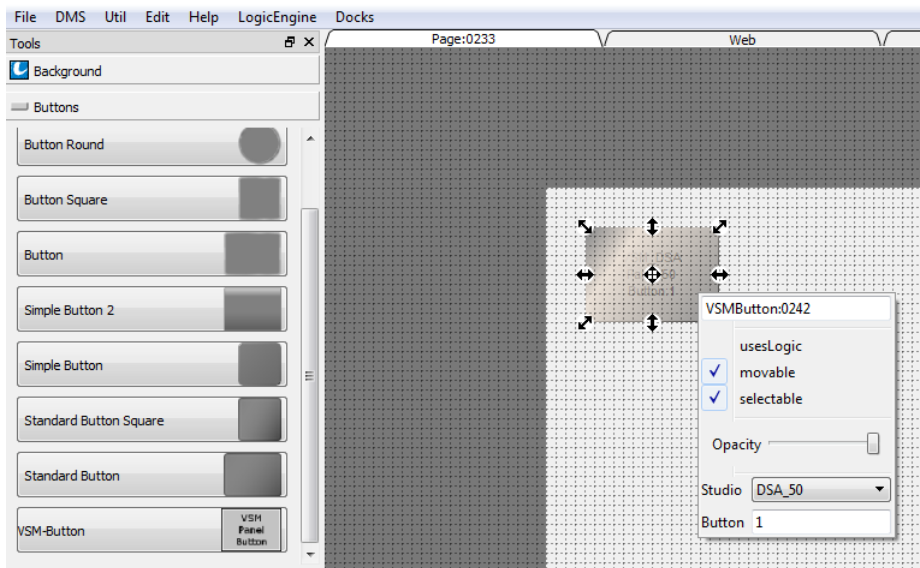
Note that you can only add panels which have already been defined in the [VSM Panels](#) list. In our example, we have selected two panels:



2. Select **OK** to save all changes and exit.

### VSM-Button Functions (Right-click)

You can now add a **VSM-Button** element to the design area, and right-click to assign the button's function:



- **Studio** - use the drop-down menu to choose the **Panel Alias** (defined in the [VSM Panels](#) list).
- **Button** - enter the button number which you wish to control.

The VisTool button is now linked to the VSM Panel button and, in [Test mode](#) or runtime, will action the VSM Panel button's function.

### VSM-Button Properties

In addition to the [general properties](#), **VSM-Button** elements support the following special properties:

Property	Value
ClassName	VisTool.Element.VSMButton
Name	VSMButton:0242
▷ Pos	(32.00 / 32.00)
ZValue	1
▷ Size	160.00 x 88.00
Rotation	0
Scale	1
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Opacity	1
▷ Font	A [Arial, 8]
▷ FontColor	■ [0, 0, 0] (255)
ColorMap	default
▷ LBP_dotColor	■ [0, 0, 0] (255)
LBP_image...	4
LBPxx_Button	1
usesLogic	<input type="checkbox"/> False
VsmPanelN...	DSA_50

<b>LBP_dotColor</b>	Defines the outline color of the VSM button (around the image supplied by the VSM server).
<b>LBP_imageMargin</b>	Sets the thickness of the outline margin (around the image supplied by the VSM server).
<b>LBPxx_Button</b>	Enter a button number (to link the VisTool button to a different VSM Panel button number) - see <a href="#">Function Options (Right-click)</a> .
<b>usesLogic</b>	Tick this option to add the VSM-Button element to the <a href="#">Logic Engine</a> .
<b>VsmPanelNumber</b>	Select a VSM panel from the drop-down menu - see <a href="#">Function Options (Right-click)</a> .

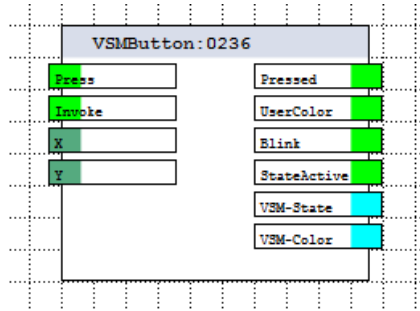
You can use the **Font** field to change the font used for any display text (if applicable).

All other properties - such as the function, image or text displayed when the button is active/inactive - are defined by the VSM server configuration and not by VisTool Editor.



### VSM-Button Logic

When added to the [Logic Engine](#), a **VSM-Button** element provides the following logical inputs and outputs:



#### ➤ Logical Inputs

<b>Press</b>	Use this input to press the VSM Button.	boolean
<b>Invoke</b>	Use this input to invoke the VSM Button.	boolean
<b>X and Y</b>	Sets the position of the VSM Button element to the X and Y values.	double

#### ➤ Logical Outputs

<b>Pressed</b>	Is active when the VSM Button is pressed.	boolean
<b>UserColor</b>	Is active when the VSM Button user color is active.	boolean
<b>Blink</b>	Is active when the VSM Button is blinking.	boolean
<b>StateActive</b>	Is active when the VSM Button is active.	boolean
<b>VSM-State</b>	Use this field to trigger another logic element (e.g. Prio, TextPrio, DoublePrio, Button16) when the VSM Button state is active.	integer
<b>VSM-Color</b>	Use this field to trigger another logic element (e.g. Prio, TextPrio, DoublePrio, Button16) when the VSM Button user color is active.	integer

### 8.3.4 Matrix-Buttons

**Matrix-Button** elements can be used to create source and target buttons to control matrix crosspoints within an external device.

The external matrix must first be added to the VisTool project. See [Configuring an External Matrix](#). A number of **Matrix-Button** elements can then be added and configured.

#### Operation

Each **Matrix-Button** can operate as one of the following:

- **Source** - selects a matrix source (can be multi-channel).
- **Target** - selects a matrix target (can be multi-channel).
- **Target Connection** - shows the connected source for a target.
- **Mute / No Source** - selects no source.

As soon as a **Source** and **Target** button are selected, the connection is made. A **Target Connection** button can be used to show the current assignment.

The **Mute / No Source** option can be used to create a "Disconnect" button - select a target and then press "Disconnect" (assigned to **Mute / No Source**).

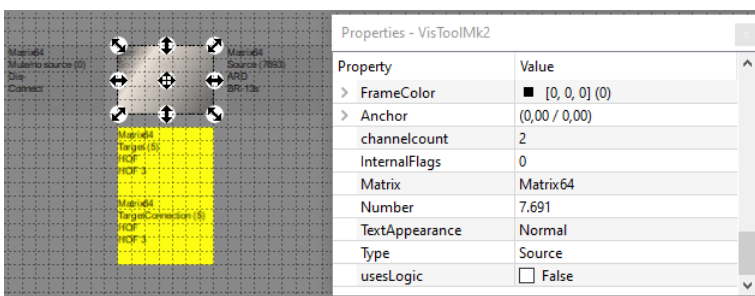
To create a multi-channel source or target, you must specify the first matrix address and the channel count. The connections are then made sequentially from the first address upwards. If there is a mismatch between the channel counts, the connections are made on a best-effort basis. For example:

- If a 2-channel source is connected to a 6-channel target, the source channels are repeated: Target receives source channels 1, 2, 1, 2, 1, 2.
- If a 6-channel source is connected to a 4-channel target, the first available source channels are used: Target receives source channels 1, 2, 3, 4.

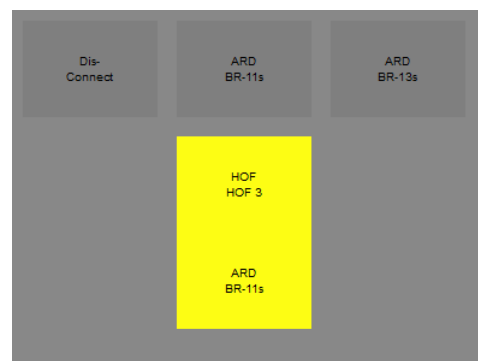
#### Labeling

In VisTool Editor, the **Matrix-Button** shows the configuration and, if available, the matrix group and signal name. In VisTool Runtime, the group and name are displayed using two lines. This may differ for Ember+ providers.

VisTool Editor (Matrix-Button Properties)



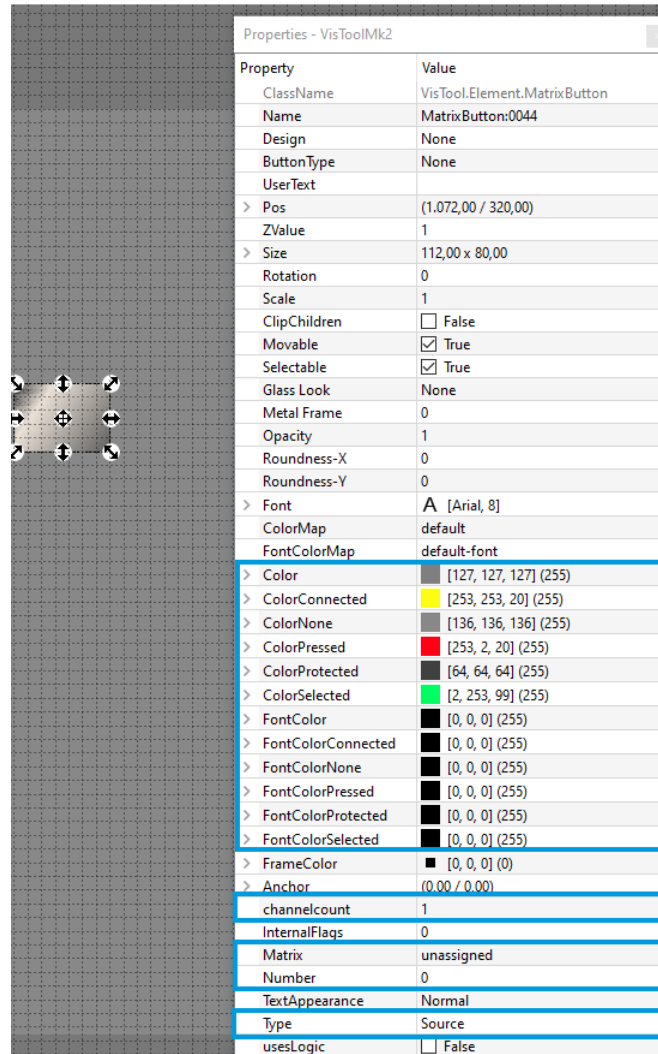
VisTool Runtime (operation)



All of the functionality is defined from the **Properties** panel.

### Matrix-Button Properties

In addition to the [general properties](#), **Matrix-Button** elements support the following special properties:



### Functionality

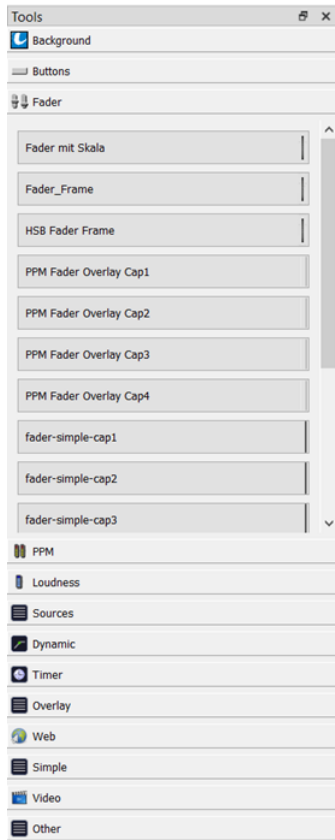
<b>Type</b>	Set the button type to <b>Source</b> , <b>Target</b> , <b>Target Connection</b> or <b>Mute/no source</b> .
<b>Matrix</b>	Select the matrix to be controlled. The drop-down menu lists all of the matrices added to the VisTool project (from the <a href="#">Ember+ Provider Tree</a> or <a href="#">DSA Matrix devices</a> ).
<b>Number</b>	Enter the matrix address number for the Source or Target (see <a href="#">Matrix Numbers</a> ).
<b>channelcount</b>	Enter the number of channels to be connected.

### Visual Appearance

The **Color** and **FontColor** properties can be used to customize the button and text label colors for different **Matrix-Button** states: connected, pressed, etc.

## 8.4 Fader

These elements can be used to create on-screen [faders](#) that can control either physical faders on a Lawo console or real values from an Ember+ provider.



There are several different **Fader** elements to choose from, offering a choice of size, scale and style:

- **Fader mit Skala** - a frame-less fader with standard cap and dB scale.
- **Fader\_Frame** - a framed fader with standard cap and dB scale.
- **HSB Fader Frame & Fader Small HSB** - a project-specific design created for HSB.
- **PPM Fader Overlay Cap 1 to 4** - an overlay fader, designed to fit on top of a PPM meter.
- **Fader Simple Cap 1 to 4** - a frame-less fader with simple cap and no scale.
- **Fader Small Cap 1 to 4** - a thin frame-less fader with simple cap and no scale.

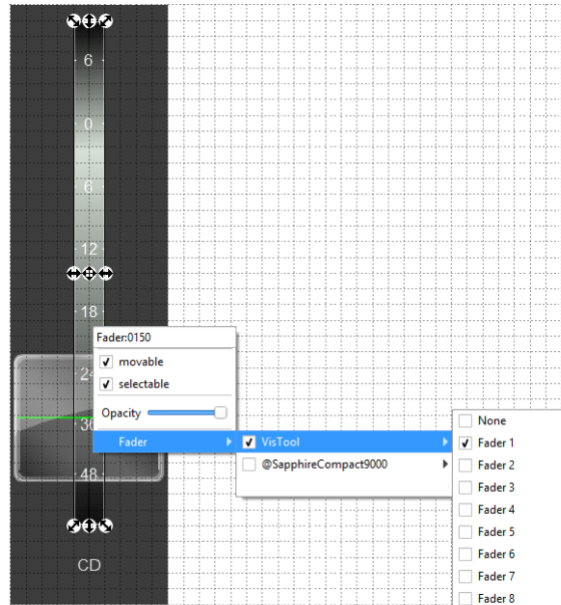
## 8. VisTool Elements

### 8.4.1 Fader Functions (Right-click)

A **Fader** element can be linked to either a physical fader on the console, or a real value from an Ember+ provider.

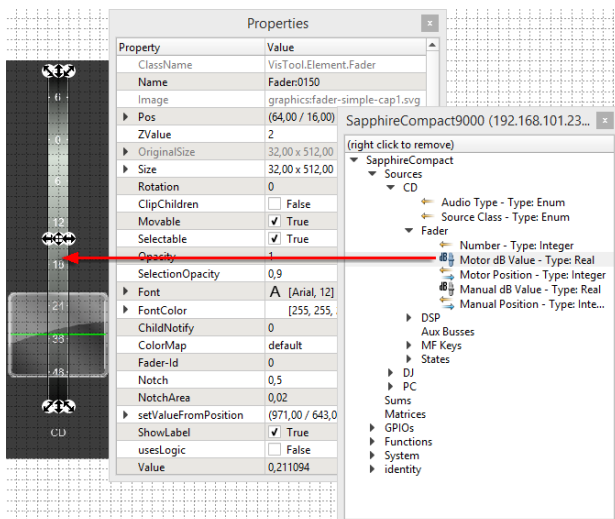
To assign a console fader, right-click and choose a channel fader number from the **VisTool** drop-down menu. The number of faders available depends on the console specified by the ON-AIR Designer project.

*Fader Element linked to Physical Fader*

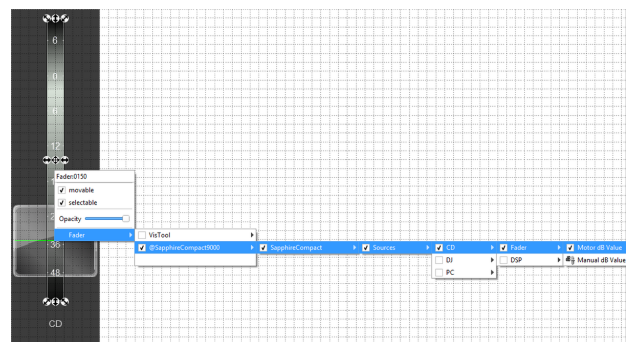


Alternatively, to assign an Ember+ value, drag and drop the value from the Ember+ provider tree onto the fader element. You can choose any Ember+ Real value (including [dB values](#)). A prerequisite for numerical values is that the minimum and maximum values must be transmitted by the provider. The Ember+ provider must be defined in the [Ember+ Devices list](#) and [added](#) to the project. Once an assignment has been made, you can interrogate or change it by right-clicking on the element.

*Drag & drop to assign Ember+ Real Value*



*Right-click to interrogate assignment*



### 8.4.2 Fader Properties

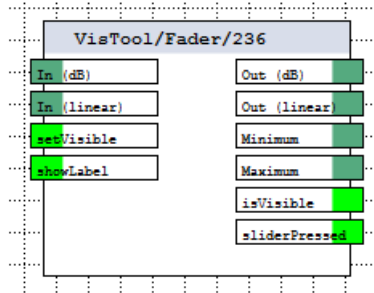
In addition to the [general properties](#), **Fader** elements support the following special properties:

Property	Value
ClassName	VisTool.Element.Fader
Name	Fader:1191
Image	graphics:Fader mit Skala...
Pos	(165.00 / 40.00)
ZValue	1
OriginalSize	176.00 x 662.00
Size	176.00 x 662.00
Rotation	0
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Opacity	1
SelectionOp...	0.9
Font	A [Arial, 12]
FontColor	[255, 255, 255] (255)
ColorMap	default
Fader-Id	1
hide on em...	<input type="checkbox"/> False
Notch	0.79545
NotchArea	0.02
ShowLabel	<input type="checkbox"/> False
usesLogic	<input type="checkbox"/> False
Value	0.5

<b>Fader-Id</b>	Enter a number to <a href="#">assign</a> a fader.
<b>Notch</b>	This field is reserved for future implementation.
<b>NotchArea</b>	This field is reserved for future implementation.
<b>usesLogic</b>	Tick this option to add the Fader element to the <a href="#">Logic Engine</a> .

### 8.4.3 Fader Logic

When added to the [Logic Engine](#), a **Fader** element provides the following logical inputs and outputs:



#### ➤ Logical Inputs

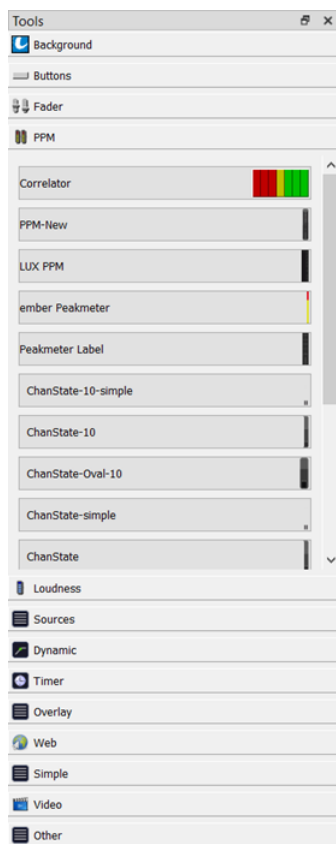
<b>In (dB)</b>	Sets the Fader level (in dB).	double
<b>In (linear)</b>	Sets the Fader level using a linear value.	double
<b>setVisible</b>	Sets the visibility of the Fader element. A logical false at the input makes the fader disappear during runtime; a logical true signal displays the fader.	boolean
<b>showLabel</b>	Sets the visibility of the Fader Label. A logical false at the input makes the label disappear during runtime; a logical true signal displays the label.	boolean

#### ➤ Logical Outputs

<b>Out (dB)</b>	Shows the Fader level (in dB).	double
<b>Out (linear)</b>	Shows the Fader level as a linear value.	double
<b>Minimum / Maximum</b>	Shows the minimum and maximum Fader positions.	double
<b>isVisible</b>	Is active when the Fader element is visible.	boolean
<b>sliderPressed</b>	Is active when the Fader's button cap (slider) is pressed.	boolean

## 8.5 PPM

These elements can be used to create on-screen [metering](#) and [visualization](#) for channel states and parameters.



There are several different types of element to choose from:

- **Correlator** - a stereo audio correlation meter.
- **PPM-New, LUX PPM** and **PPM** - audio Peak Program Metering (in a choice of styles). The **LUX PPM** offers the most design options.
- **ember Peakmeter** and **EmberPPM** - metering of an external device via Ember+.
- **Peakmeter Label** - a meter scale only (designed to accompany the **ember Peakmeter**).
- **ChanState-10** and **ChanState** - an overview of channel states (in a choice of styles). This element is designed to fit alongside a channel PPM.
  - The **ChanState** element includes all 20 Aux bus assignments defined in the "System -> Definition -> Parameter = Aux" branch of the ON-AIR Designer's 'Tree Definition'.
  - The **ChanState-10** element includes the first 10. This makes it suitable for use with smaller screen sizes such as the Visual Extension.
- **GainRed** - displays the gain reduction components (from the **ChanState** element).
- **VU, VU2** - audio VU metering (in a choice of styles).

The available functions and properties vary depending on the type of element.



### 8.5.1 Audio Metering Functions (Right-click)

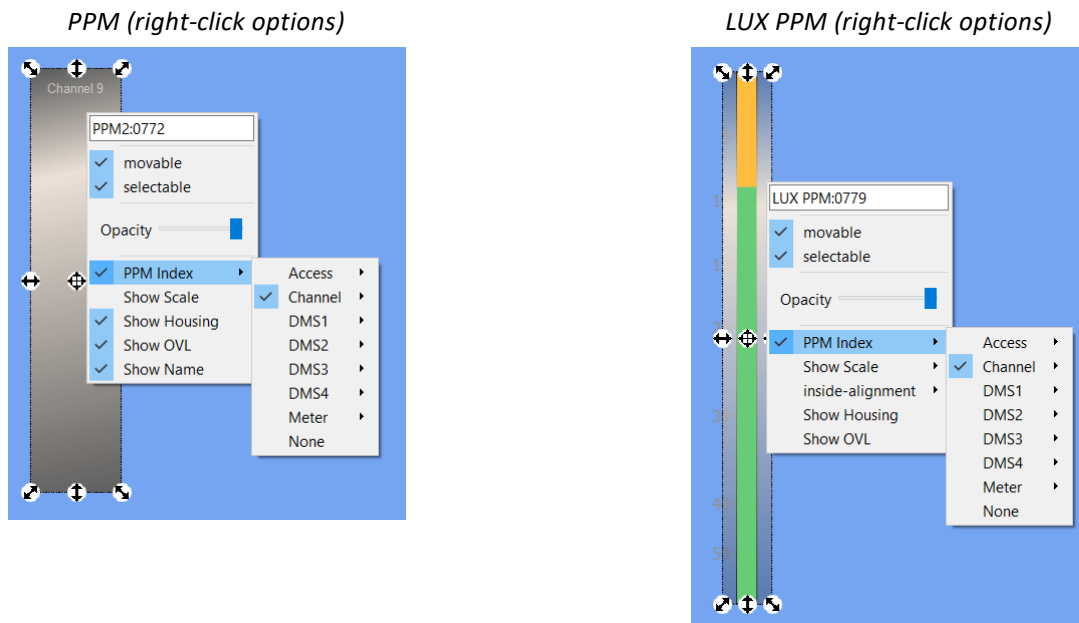
#### Ember+ Elements

An Ember+ metering element is assigned to a "PeakValue" data stream from an Ember+ provider. The assignment is made using a drag and drop from the Ember+ tree, see [Ember+ Metering](#). Right-click on the element to interrogate the Ember+ stream connections.

#### DMS Elements

A PPM, VU or Correlator is linked to an audio source from the DMS device. Right-click on the element and choose **PPM Index** (or **VU Index**) to access the options.

Note that the number of assignable DMS meters varies depending on the product. The examples below are taken from a ruby system.



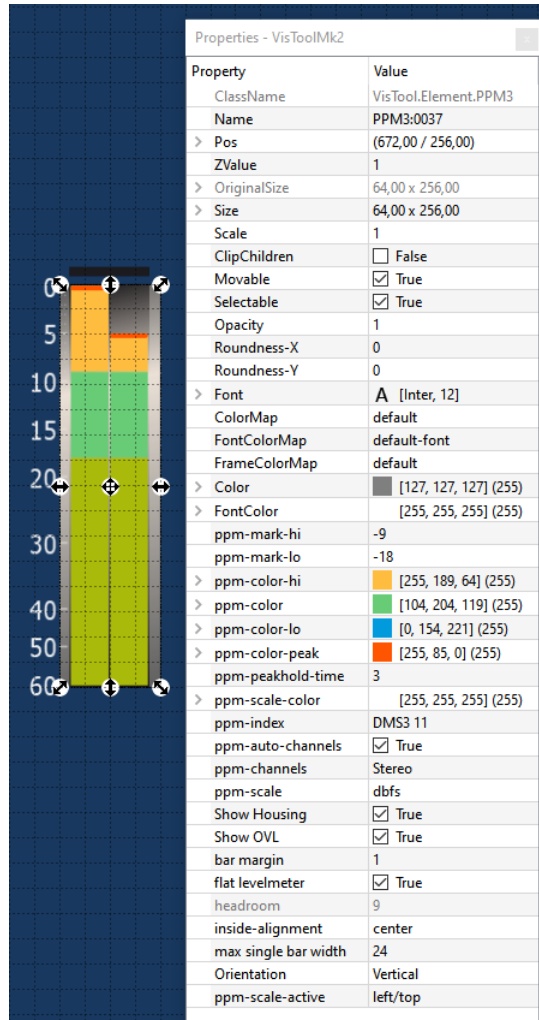
- **Access** - links the meter to the source in access. You can choose the meter's pickup point from the drop-down menu.
- **Channel** - links the meter to a specific fader strip. The number of faders available depends on the console specified by the ON-AIR Designer project. The meter will be mono, stereo or 5.1 according to the source or bus assigned to the fader strip. Channel meters include the Display name of the source/bus.
- **DMS** - links the meter to a DMS meter defined in the ON-AIR Designer project (via the "Audio Input -> DMS" branch of the 'Tree Definition'). DMS meters are freely configurable and allow VisTool to meter any signal - for example, a physical IO, conference bus, Minimixer, etc. If more than 64 meters are available, then they are presented in groups (e.g. **DMS1** to **DMS4**), where each group contains 64 meters. Once assigned to a VisTool element, the meter will display the label defined by the ON-AIR Designer configuration. Each DMS meter also has a Silence detect control output. The silence detect thresholds are defined by the ON-AIR Designer "Audio Input -> DMS" branch (under the **Loopback/Silence** tab).
- **Meter (A, B, and 1 to 4)** - links the meter to one of the six stereo main meters defined in the ON-AIR Designer project (via the "System -> Definition -> Parameter = Meter" branch of the 'Tree Definition'). Main meters include the Display name of the source/bus.
- **None** - removes the link so that the meter is unassigned.

The global time constant applied to all VisTool meters is adjusted by the **DMS Integration Mode** parameter defined in the ON-AIR Designer project (via the "System -> Definition -> Parameter = Settings" branch of the 'Tree Definition').

## 8.5.2 Audio Metering Properties

In addition to the [general properties](#), PPM elements support the following special properties. The LUX PPM offers the most design options.

LUX PPM (properties)



Property	Value
ClassName	VisTool.Element.PPM3
Name	PPM3:0037
Pos	(672,00 / 256,00)
ZValue	1
OriginalSize	64,00 x 256,00
Size	64,00 x 256,00
Scale	1
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Opacity	1
Roundness-X	0
Roundness-Y	0
Font	A [Inter, 12]
ColorMap	default
FontColorMap	default-font
FrameColorMap	default
Color	■ [127, 127, 127] (255)
FontColor	[255, 255, 255] (255)
ppm-mark-hi	-9
ppm-mark-lo	-18
ppm-color-hi	■ [255, 189, 64] (255)
ppm-color	■ [104, 204, 119] (255)
ppm-color-lo	■ [0, 154, 221] (255)
ppm-color-peak	■ [255, 85, 0] (255)
ppm-peakhold-time	3
ppm-scale-color	[255, 255, 255] (255)
ppm-index	DMS3 11
ppm-auto-channels	<input checked="" type="checkbox"/> True
ppm-channels	Stereo
ppm-scale	dbfs
Show Housing	<input checked="" type="checkbox"/> True
Show OVL	<input checked="" type="checkbox"/> True
bar margin	1
flat levelmeter	<input checked="" type="checkbox"/> True
headroom	9
inside-alignment	center
max single bar width	24
Orientation	Vertical
ppm-scale-active	left/top

### Meter Functionality

As a first step, it is a good idea to check the following:

- **ppm-index** - sets the audio source. Identical to the right-click options described [earlier](#).
- **Orientation** - sets the orientation to either Vertical or Horizontal. After a change of orientation, the meter dimensions must be adjusted manually.
- **headroom** - is for information purposes only. It displays the amount of headroom supported by the hardware.

### Channel Format

Each meter can be mono, stereo or 5.1.

To set the format automatically, turn on **ppm-auto-channels**. VisTool evaluates the number of channels received via DMS and sets the format of the meter accordingly. This is particularly useful if the number of channels changes during operation.

Alternatively, turn off **ppm-auto-channels** and use the **ppm-channels** menu to set the format manually. The options are mono, stereo, 5.1 or 5.1+2. Note that the **ppm-channels** option is ignored if **ppm-auto-channels** is enabled.

### Meter Scale

For a standard PPM, the position of the scale is fixed (on the left of the meter). The scale can be turned on or off using **ppm-scale-visible**.

For a LUX PPM, you can choose if the scale appears on the left/top or right/bottom of the meter using **ppm-scale-active**. Set this option to off to hide the scale.

The other ppm-scale properties are the same for all types of PPM:


- **ppm-scale** sets the meter scale. Options include DIN, UK, NORDIC, dBFS and astro.
- **ppm-scale-color** sets the color (as described earlier).

### Meter Color-Coding and Peak Level

Each meter supports three different colors and a peak level indicator.

The three colors are defined by the following properties:

- **ppm-color / color-hi** and **color-lo** set the colors.
- **ppm-mark-hi / mark-lo** set the levels at which the meter changes color (in dBFS).

To set the colors, click on  to open the color selector, or click on the arrow to access the individual color fields. For DMS channels, **ppm-color-lo** is replaced by the color defined in the ON-AIR Designer configuration.

To set the levels, enter a value in dBFS:

- **ppm-mark-hi** sets the change from **ppm-color-hi** to **ppm-color**.  
If a value is not defined (nan), then the configured headroom of the hardware is used.
- **ppm-mark-lo** sets the change from **ppm-color** to **ppm-color-lo**.  
If a value is not defined (nan), then only two colors are used: **ppm-color-lo** and **ppm-color-hi**.

The peak level indicator is defined by the following properties:

- **ppm-color-peak** sets the color (as described earlier).
- **ppm-peakhold-time** sets the length of time that the peak level indicator remains visible. The time is set in seconds from the drop-down menu.

### Visual Appearance

The following properties can be used to fine tune the appearance:

- **show Housing** - show or hide the meter housing.
- **show OVL** - show or hide the overload indicator.
- **flat levelmeter** - show or hide the 3D effects: flat = no 3D effects.

For a standard PPM, you can also change the shape of the meter using **style-1.0**.

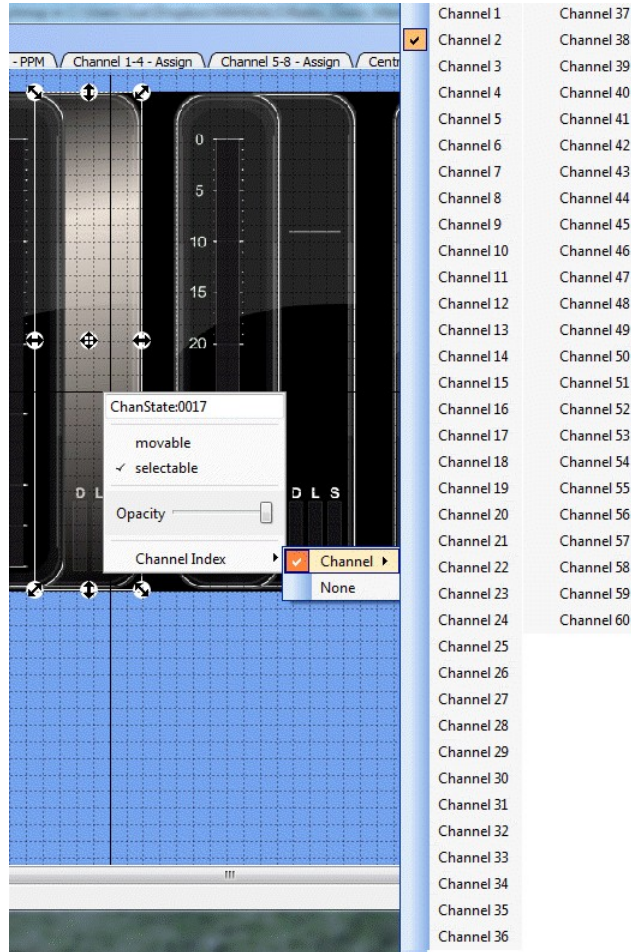
For a LUX PPM, you can also change the internal positioning of the bargraphs. This is particularly useful if the number of channels changes during operation.

- **bar margin** = the distance between the bargraphs and edge of the LUX PPM element (in pixels).
- **inside-alignment** = the position of the bargraphs within the LUX PPM element. The options are left/top, centre or right/bottom.
- **max single bar width** = the maximum width of the individual bargraphs (in pixels).

### 8.5.3 Channel State Functions (Right-click)

A **ChanState**, **ChanState-10** and **Gain Reduction** element can be linked to any fader strip on the console by choosing a **Channel Index**:

*Right-click options (for sapphire)*



The number of faders available depends on the console specified by the ON-AIR Designer project. Once assigned the element will show channel [status](#) information for the source or bus assigned to the fader strip.

### 8.5.4 Channel State Properties

In addition to the [general properties](#), **Channel State** and **Gain Reduction** elements include color properties to differentiate between the bus assignment states (**ChanState**) and gain reduction metering (**GainRed**).

The properties for a **ChanState** element are shown below:

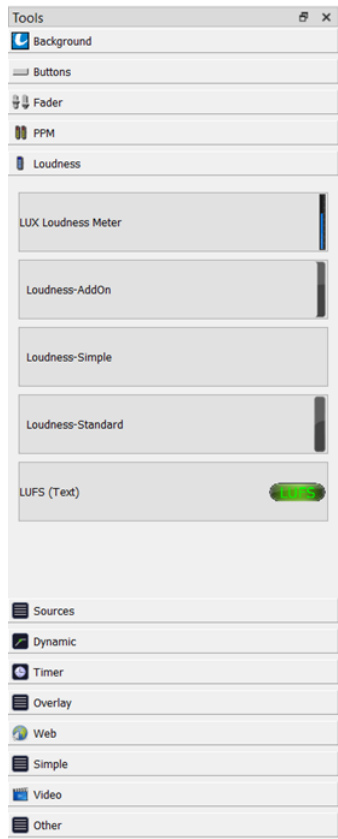
Property	Value
ClassName	VisTool.Element.ChanS...
Name	ChanState:0109
Image	graphics:ChanState.svg
Pos	(400.00 / 64.00)
ZValue	1
OriginalSize	112.48 x 525.00
Size	112.48 x 525.00
Rotation	0
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Opacity	1
SelectionOp...	0.9
Font	A [Arial, 12]
ColorNone	[136, 136, 136] (255)
ColorOff	[2, 253, 99] (255)
ColorPost	[253, 253, 20] (255)
ColorPre	[253, 2, 20] (255)
FontColor	[255, 255, 255] (255)
ppm-color	[2, 253, 99] (255)
ppm-color1	[253, 2, 20] (255)
ppm-color2	[253, 253, 20] (255)
ColorMap	default
ppm-index	Channel 3

<b>ColorNone</b>	Sets the color used if the Aux bus position is unused.
<b>ColorOff</b>	Sets the color used when bus assignments are switched off.
<b>ColorPre</b>	Sets the color for pre-fader bus assignments.
<b>ColorPost</b>	Sets the color for post-fader bus assignments.
<b>ppm-color</b>	Sets the color of the PPM meter for different signal thresholds.

The Aux buses displayed by the **ChanState** and **ChanState-10** elements are defined in the "System -> Definition -> Parameter = Aux" branch of the ON-AIR Designer's 'Tree Definition'. The **ChanState** element displays all 20 bus assignments, while the **ChanState-10** element displays the first 10.

## 8.6 Loudness

These elements can be used to configure [loudness](#) metering.



The first four elements create a loudness metering bargraph:

- **LUX Loudness Meter** - similar to **Loudness-Simple** but offers more design options.
- **Loudness-AddOn** - designed to fit alongside a channel [PPM](#).
- **Loudness-Simple** - a simple loudness metering bargraph without a background. This option works well when overlaid onto a [PPM](#).
- **Loudness-Standard** - a complete stand-alone element.

The **LUFS (Text)** element creates a text box which displays the integrated loudness measurement. It is usually accompanied by buttons to start, stop and reset the measurement.

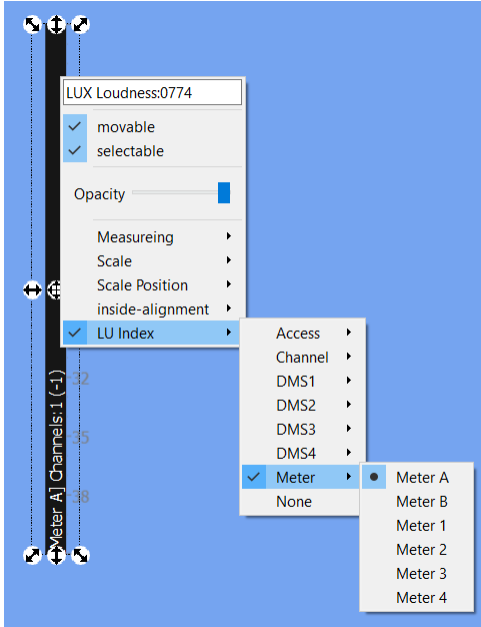
The available functions and properties vary depending on the type of element.

## 8.6.1 Loudness Metering Functions (Right-click)

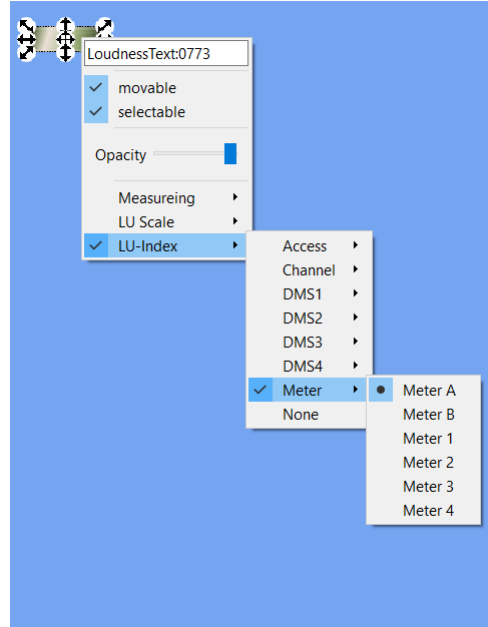
A loudness element is linked to an audio source in the DMS device. The options are the same as those described [earlier](#) (for a PPM).

Right-click on the element and choose **LU Index** to access the options.

*LUX Loudness Meter (right-click options)*



*LUFS Text only (right-click options)*

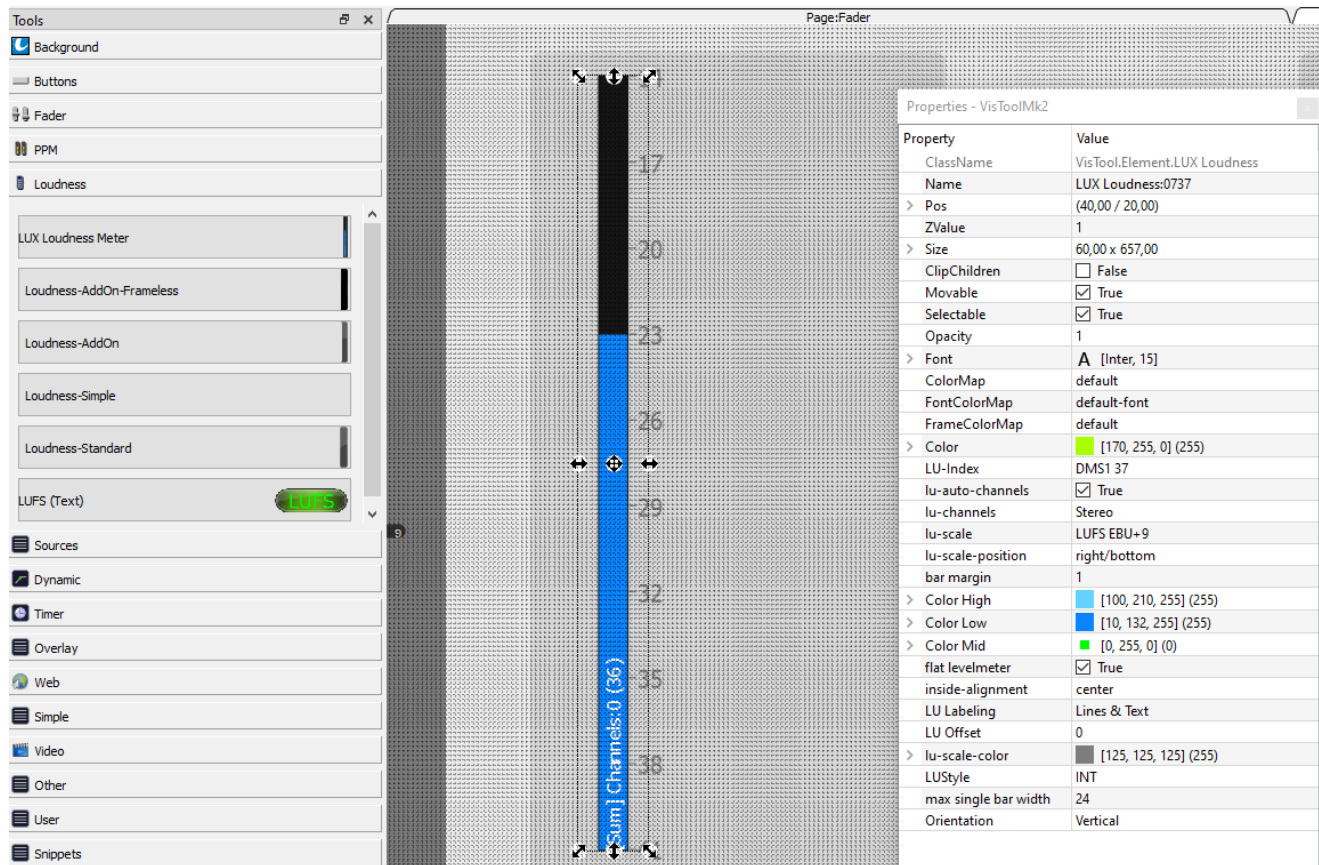


## 8.6.2 Loudness Metering Bargraph Properties

In addition to the [general properties](#), Loudness metering bargraph elements support the following special properties. The LUX element offers the most design options.

Note that all loudness metering parameters conform to the ITU-R BS1770 standard.

LUX Loudness Meter (properties)



### Meter Functionality

As a first step, it is a good idea to check the following:

- **LU-index** - sets the audio source. Identical to the right-click options (described [earlier](#)).
- **LU Offset** - sets the Target Level relative to 0 = -23 LUFS.
- **Orientation** - sets the orientation to either Vertical or Horizontal. After a change of orientation, the meter dimensions must be adjusted manually.

### Channel Format

The loudness metering element can be mono, stereo or 5.1. Note that the format affects the number of channels processed, and not the number of bargraphs displayed. For loudness metering, a single bargraph represents the average energy of the summed component channels: mono, stereo or surround.

To set the format automatically, turn on **lu-auto-channels**. VisTool evaluates the number of channels received via DMS and sets the format of the element accordingly. This is particularly useful if the number of channels changes during operation.

Alternatively, turn off **lu-auto-channels** and use the **lu-channels** menu to set the format manually. The options are mono, stereo, 5.1 or 5.1+2. Note that the **lu-channels** option is ignored if **lu-auto-channels** is enabled.



### Measurement Type

The **LU Style** property determines the loudness measurement type:

- Select either **MOM** or **SHORT** for bargraph elements to determine a fixed integration time: Momentary (400ms sliding window) or Short term (3s sliding window).
- Select **INT** to configure an integrated loudness measurement (to be started/stopped by the user).
- Select **LRA** to configure a loudness range measurement. This is similar to an integrated loudness measurement, but is gated (at -70dB LUFS) to exclude silence from the measurement.

### Meter Scale

The loudness metering scale can display values in either LUFS (absolute) or LU (relative to the Target Level). In addition, it can use either the EBU +9 or EBU +18 scale.

For a standard loudness element, the scale type is defined by two properties: **LU Scale** and **LU EBU**. The **LU Labeling** property determines whether the scale shows Lines only, Text only, Lines & Text or None. Set **LU Labeling** to None to turn off the scale.

For a LUX Loudness Meter, the scale type is defined by a single property: **lu-scale**. Set **lu-scale** to None to turn off the scale. The **LU Labeling** property has no function.

The LUX Loudness Meter supports two additional options:

- **lu-scale-position** - sets the position of the scale to either left/top or right/bottom.
- **lu-scale-color** - sets the color of the scale labeling.

### Meter Color-Coding

The loudness metering scale also supports three different colors:

- **Color High** - sets the color of the scale above the Target Level.
- **Color Low** - sets the color of the scale below the Target Level.
- **Color Mid** - sets the color of the scale at the Target Level (by default, Alpha = 0).

### Visual Appearance

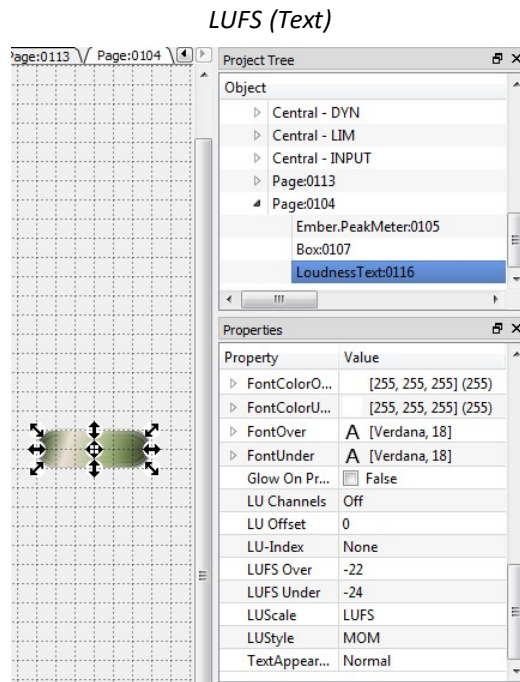
The following properties can be used to fine tune the appearance:

- **flat levelmeter** - show or hide the 3D effects: flat = no 3D effects.

For a LUX Loudness Meter, you can also change the internal positioning of the bargraph:

- **bar margin** = the distance between the bargraph and edge of the LUX element (in pixels).
- **inside-alignment** = the position of the bargraph within the LUX element. The options are left/top, centre or right/bottom.
- **max single bar width** = the width of the individual bargraphs. This option is not relevant for loudness metering (as there is only one bargraph).

### 8.6.3 LUFS (Text Only) Properties



For the **LUFS (Text)** element, you can change how the integrated loudness measurement is displayed when the measured level reaches a specific value. Two thresholds can be specified: **LUFS Over** and **LUFS Under**. This allows you to alert the operator when the loudness measurement exceeds or falls below a certain value.

Start by using the **LUFS Over** and **LUFS Under** properties to define the two levels.

Then use the following properties to define how the integrated loudness measurement is displayed:

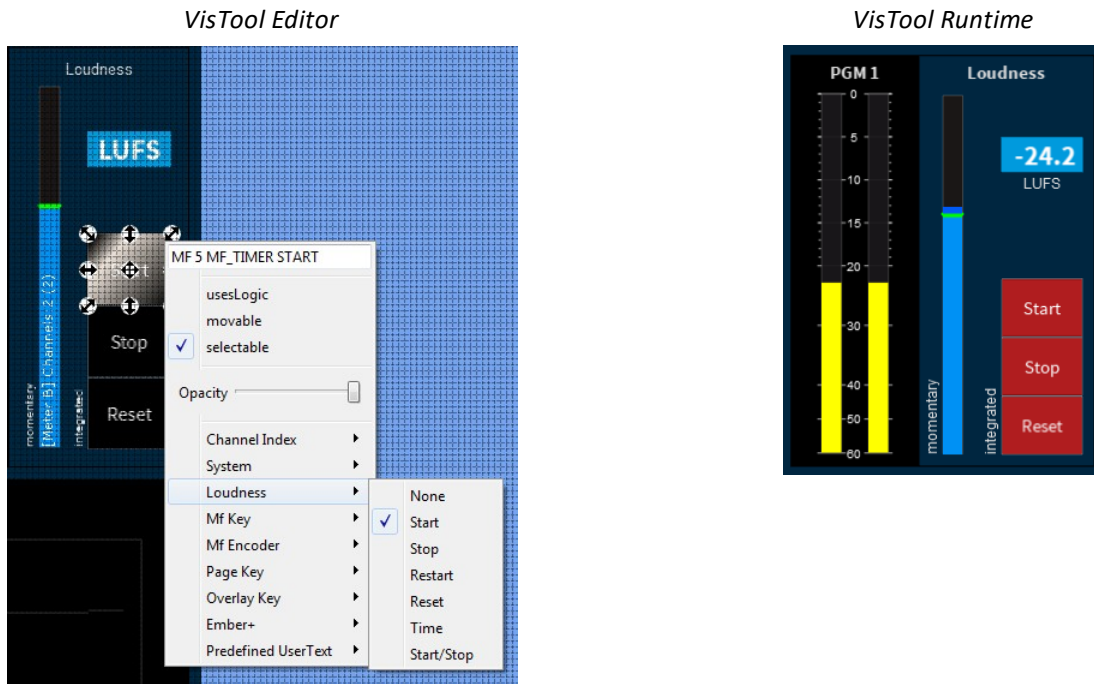
- **FontColor** - sets the font color for a normal reading: between **LUFS Over** and **LUFS Under**.
- **FontColorOver** - sets the font color used when the measured level is above **LUFS Over**.
- **FontColorUnder** - sets the font color used when the measured level is below **LUFS Under**.
- **Font** - sets the font type (including size) for a normal reading: between **LUFS Over** and **LUFS Under**.
- **FontOver** - sets the font type (including size) used when the measured level is above **LUFS Over**.
- **FontUnder** - sets the font type (including size) used when the measured level is below **LUFS Under**.

### 8.6.4 Integrated Loudness Measurement Start, Stop, etc.

When using the **LUFS (Text)** element, it is a good idea to create some buttons to control the measurement. For example:

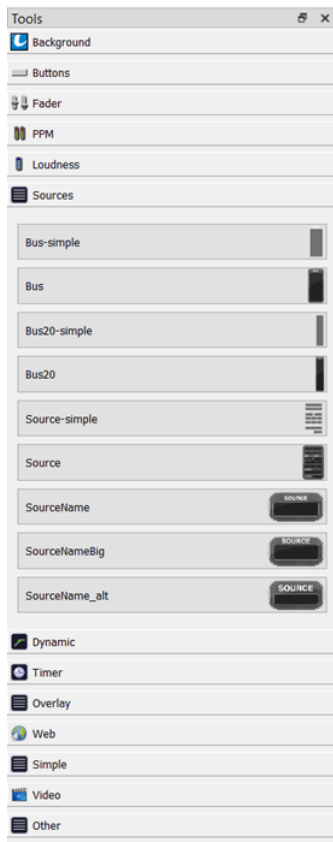
1. Add some [Box](#) elements to the page in the usual manner.
2. Right-click and assign one of the **Loudness** measurement functions: Start, Stop, Restart, Reset, Time or Start/Stop.
3. Repeat for each function you wish to use.

The example below uses the **Start**, **Stop** and **Reset** functions:



## 8.7 Sources

These elements can be used to display parameters for the source "in access". They are often included in [Source Parameter Control](#) pages.



The **Source** elements include:

- **Bus** and **Bus20** - displays the Aux bus assignments (in a choice of styles).

The Aux buses are defined in the "System -> Definition -> Parameter = Aux" branch of the ON-AIR Designer's 'Tree Definition'.

The **Bus20** element displays all 20 bus assignments, while the **Bus** element displays the first 10.

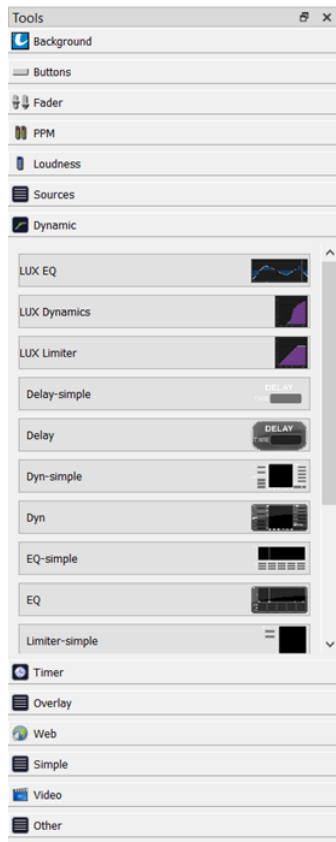
- **Source** - displays source parameters such as mic gain (in a choice of styles).
- **SourceName** - displays the source name in a choice of sizes and design.

These elements always follow the source in access.

Their properties are covered in the [general properties](#) topic, with the exception of the **Bus** elements which can be color-coded for different bus assign states. These are configured in an identical manner to a [ChanState](#) element.

## 8.8 Dynamic

These elements can be used to display DSP parameters for the source "in access". They are often included in [Source Parameter Control](#) pages.



The **Dynamic** elements include:

- **EQ** – EQ graph and parameters.
- **Dynamics** - Expander, Gate, Compressor (and De-Esser) graph and parameters.
- **Limiter** – Limiter graph and parameters.
- **Delay** – delay time
- **Panning** - XY Pan position and parameters.
- **PWB** – Pan, Balance or Width parameters.

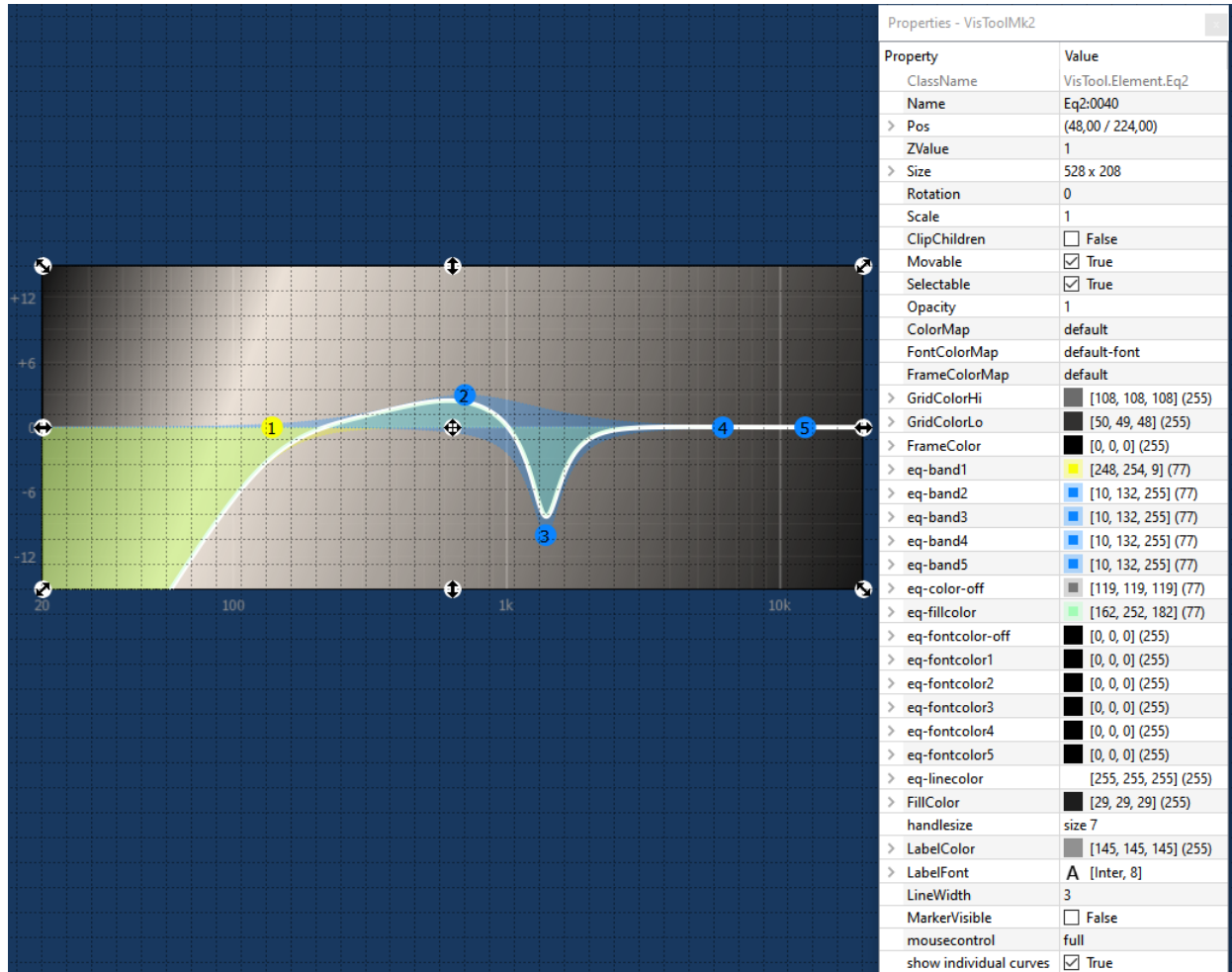
The **EQ**, **Dynamics** and **Limiter** are available in standard or LUX versions. The LUX versions offer the most design options.

**Dynamic** elements always follow the source in access. Therefore, there is no need to define a function (via a right-click). The properties vary depending on the type of element.

### 8.8.1 EQ Properties

In addition to the [general properties](#), EQ elements support the following special properties.

LUX EQ (properties)



#### EQ Graph

The **LUX EQ** provides a graphical display that can be customized in many ways. The **EQ** and **EQ-simple** elements support a subset of these properties.

Use the following properties to define the background, grid and scale.

- **FillColor** sets the color of the background.
- **GridColorHi** / **GridColorLo** sets the colors used for the main and auxiliary grid lines.
- **LabelColor** sets the color used for the scale.
- **LabelFont** sets the font used for the scale.

The overall frequency response is shown as a single EQ line / curve. Its color, width and fill are defined by:

- **eq-linecolor** sets the color of the EQ curve (when EQ is active).
- **LineWidth** sets the width of the EQ curve.
- **eq-fillcolor** sets the color used to fill above and beneath the EQ curve.

If **show individual curves** is enabled, then individual curves for each EQ band are also displayed.


If **MarkerVisible** is enabled, then vertical lines at each EQ band frequency are displayed.

The individual EQ bands are color-coded as follows:

- **eq-band1 (to 5)** sets the color of the "point" for the EQ band.  
+ the individual fill color (if **show individual curves** is enabled).  
+ the vertical frequency marker color (if **MarkerVisible** is enabled).
- **eq-fontcolor1 (to 5)** sets the font color used for the parameter values.

The following options can be used to change how the graph looks when EQ processing is turned off:

- **eq-color-off** sets the color of the EQ curve (when EQ is off).
- **eq-fontcolor-off** sets the font color used for all parameter values (when EQ is off).

To set the colors, click on  to open the color selector, or click on the arrow to access the individual color fields.

### Mouse / Touch Operation

By enabling **mousecontrol**, operators can adjust EQ parameters from the graphical display. This option is supported by all EQ elements.

The property has three possible states:

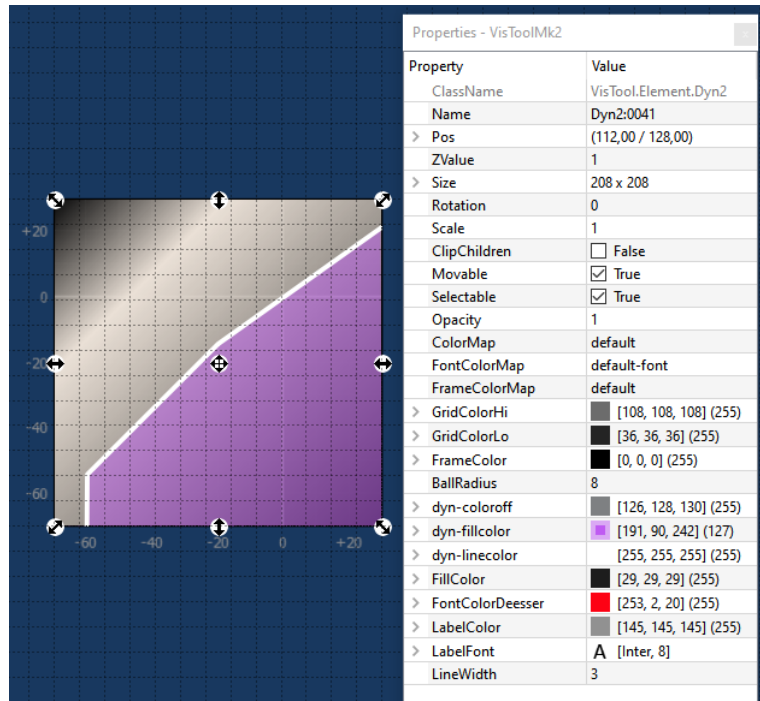
- **full** = mouse / touch control works independently of EQ On/Off.
- **if active** = mouse / touch control is only possible when the EQ is On.
- **disabled** = mouse / touch control is disabled.

When **mousecontrol** is enabled, a small circular handle appears for each EQ band. The **handle size** sets the diameter of the circles from 1 to 7, where 1 = large (ideal for touch operation) and 7 = small (better for mouse operation).

## 8.8.2 Dynamics Properties

The graphical displays for **Dynamics** elements are configured in a similar manner to the EQ. Note that mouse control can be enabled for the Limiter but not for the Dynamics.

LUX Dynamics (properties)



### Dynamics Graph

The **LUX Dynamics** provides a graphical display that can be customized in many ways. The **Dyn** and **Dyn-simple** elements support a subset of these properties.

The background, grid and scale are defined as follows:


- **FillColor** sets the color of the background.
- **GridColorHi** / **GridColorLo** sets the colors used for the main and auxiliary grid lines.
- **LabelColor** sets the color used for the scale.
- **LabelFont** sets the font used for the scale.

The overall shape of the Dynamics processing is shown as a single line. In addition, there are two balls that move dynamically as the signal level varies: one for the Dynamics and one for the De-Esser.

- **BallRadius** sets the size of both balls (in pixels).
- **dyn-linecolor** sets the color of the line and Dynamics ball (when Dynamics is active).
- **LineWidth** sets the width of the line.
- **dyn-fillcolor** sets the color used to fill beneath the line.
- **FontColorDeesser** sets the color of the De-Esser ball.

The following option can be used to change how the graph looks when Dynamics processing is turned off:

- **dyn-color-off** sets the color of the line (when Dynamics is off).

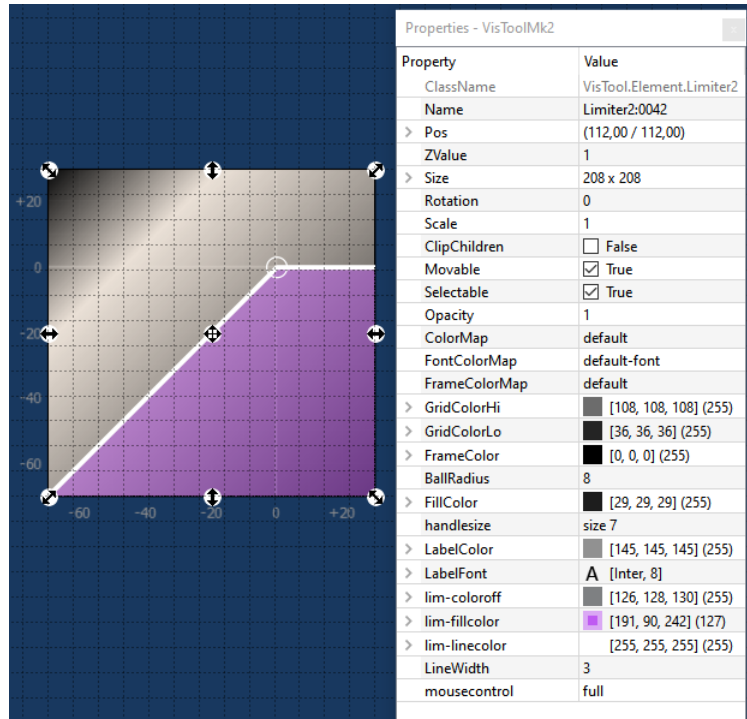
To set the colors, click on  to open the color selector, or click on the arrow to access the individual color fields.



### 8.8.3 Limiter Properties

The graphical displays for **Limiter** elements are configured in a similar manner to the EQ.

*LUX Limiter (properties)*



Property	Value
ClassName	VisTool.Element.Limiter2
Name	Limiter2:0042
Pos	(112,00 / 112,00)
ZValue	1
Size	208 x 208
Rotation	0
Scale	1
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Opacity	1
ColorMap	default
FontColorMap	default-font
FrameColorMap	default
GridColorHi	[108, 108, 108] (255)
GridColorLo	[36, 36, 36] (255)
FrameColor	[0, 0, 0] (255)
BallRadius	8
FillColor	[29, 29, 29] (255)
handlesize	size 7
LabelColor	[145, 145, 145] (255)
LabelFont	A [Inter, 8]
lim-coloroff	[126, 128, 130] (255)
lim-fillcolor	[191, 90, 242] (127)
lim-linecolor	[255, 255, 255] (255)
LineWidth	3
mousecontrol	full

#### Limiter Graph

The **LUX Limiter** provides a graphical display that can be customized in many ways. The **Lim** and **Lim-simple** elements support a subset of these properties.

The background, grid and scale are defined as follows:


- **FillColor** sets the color of the background.
- **GridColorHi** / **GridColorLo** sets the colors used for the main and auxiliary grid lines.
- **LabelColor** sets the color used for the scale.
- **LabelFont** sets the font used for the scale.

The overall shape of the Limiter is shown as a single line. In addition, there is a ball that moves dynamically as the signal level varies.

- **BallRadius** sets the size of the ball (in pixels).
- **lim-linecolor** sets the color of the line and ball (when Limiter is active).
- **LineWidth** sets the width of the line.
- **lim-fillcolor** sets the color used to fill beneath the line.

The following option can be used to change how the graph looks when the Limiter is turned off:

- **lim-color-off** sets the color of the line (when Limiter is off).

To set the colors, click on  to open the color selector, or click on the arrow to access the individual color fields.

### Mouse / Touch Operation

By enabling **mousecontrol**, operators can adjust the Limiter threshold from the graphical display. This option is supported by all Limiter elements.

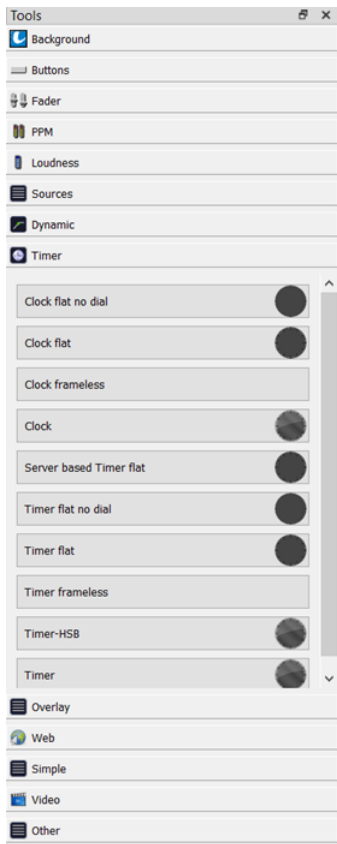
The property has three possible states:

- **full** = mouse / touch control works independently of Limiter On/Off.
- **if active** = mouse / touch control is only possible when the Limiter is On.
- **disabled** = mouse / touch control is disabled.

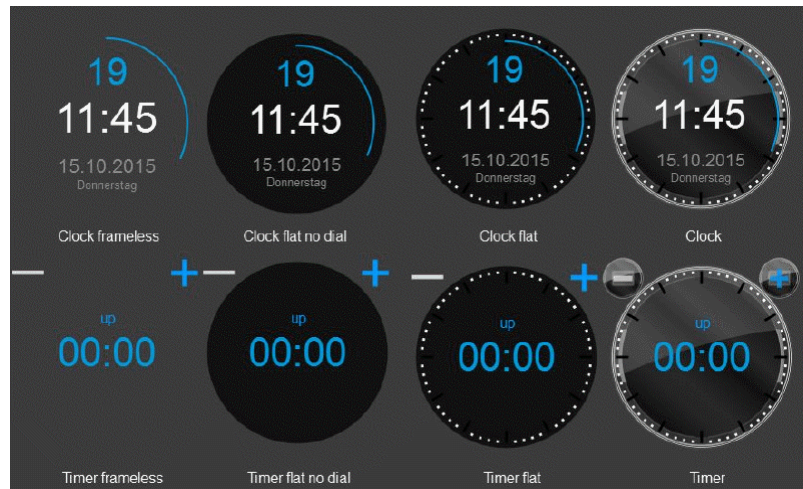
When mousecontrol is enabled, a small circular handle appears for the Limiter threshold. The **handle size** sets the diameter of the circles from 1 to 7, where 1 = large (ideal for touch operation) and 7 = small (better for mouse operation).

## 8.9 Timer & Clock

These elements can be used to create on-screen [timers](#) or a [system clock](#).



They are available in standard, flat, flat (no dial) or frameless variants:



VisTool timers link to timers defined in the ON-AIR Designer configuration. Two different types of timer are supported: standard and server-based. Use a standard **Timer** for stand-alone operation. **Server-based Timers** can be used to synchronize timer operation across multiple VisTools.

A **Clock** always shows the internal time and date of the remote system. For Lawo's radio products this can be set using SoP Explorer (included with the ON-AIR Designer software).

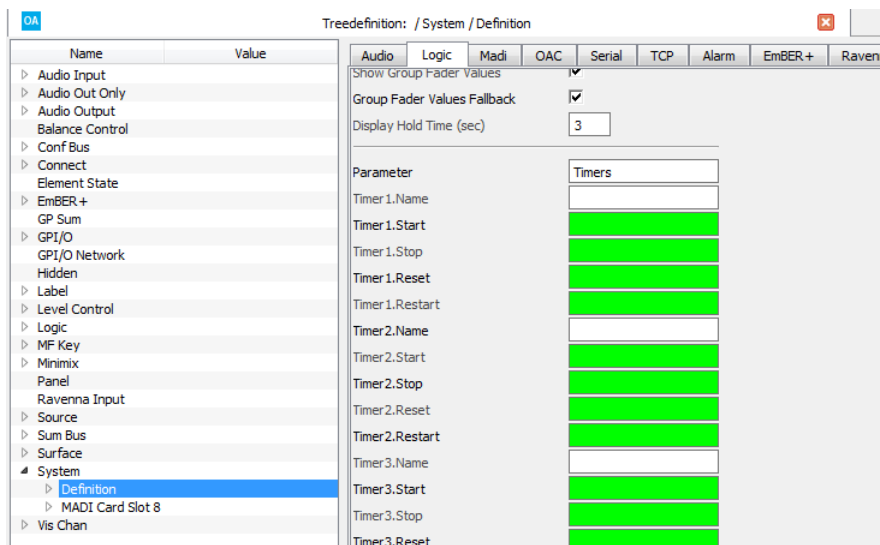
### 8.9.1 Standard Timers

"System -> Definition -> Parameter = Timers"

This branch of the ON-AIR Designer 'Tree Definition' sets the options for the six standard timers included in every configuration. Choose a standard timer if you do not need synchronized operation.

1. Start by selecting "**System -> Definition**" in 'Tree Definition' and the **Logic** tab. Scroll down to access the timer parameters.

ON-AIR Designer Standard Timers



The fields can be used to name each timer and define how it is triggered:

<b>Parameter</b>	Timer – reference name for the element.
<b>Time N Name</b>	Enter a reference name for the timer.
<b>Timer N Start</b>	Assigns a control signal to start the timer. The timer will start from its current position. If you want a key to always start the timer from 00:00:00:00, assign the same key to both the <b>Timer N Start</b> and <b>Timer N Reset</b> functions.
<b>Timer N Stop</b>	Assigns a control signal to stop the timer.
<b>Timer N Reset</b>	Assigns a control input signal to reset the timer to 00:00:00:00.
<b>Timer N Restart</b>	Assigns a control input signal to restart the timer from its current position.

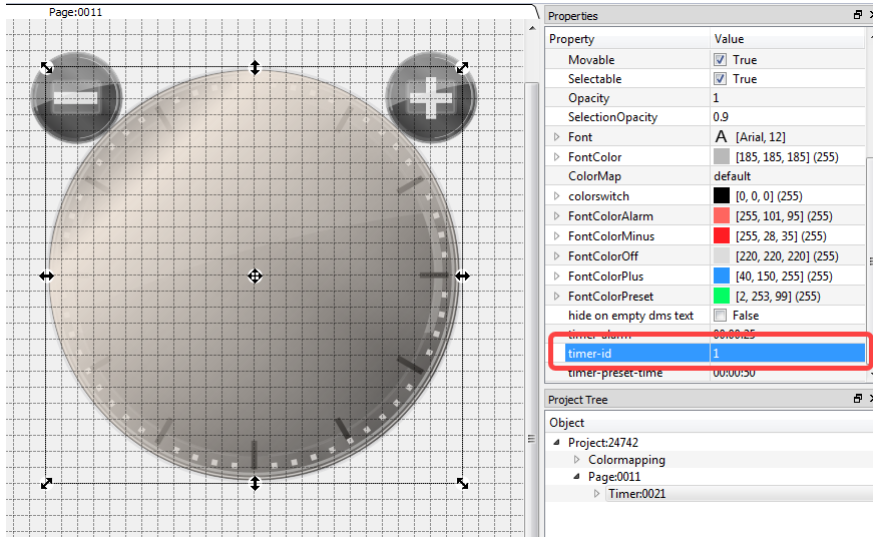
In the 'Tree Selection' window, the following outputs appear under "Logic -> Definition -> Parameter Timer". These can be used to illuminate the trigger keys and/or signal the timer alarm status:

<b>Timer N Run</b>	Active when the timer is running. Can be used to illuminate the MF Key controlling the timer.
<b>Timer N Alarm</b>	Active when the timer alarm is active.

2. Then open VisTool Editor and insert a **Timer** element in the usual manner.

Edit the **Timer ID** in the 'Properties' panel to link the element to an ON-AIR Designer timer. For standard timers, the ID numbers are fixed: Timer 1 = ID 1; Timer 2 = ID 2; and so on.

*VisTool Editor Standard Timer (linked to Timer 1)*



### 8.9.2 Server-based Timers

"Logic -> Server based Timer"

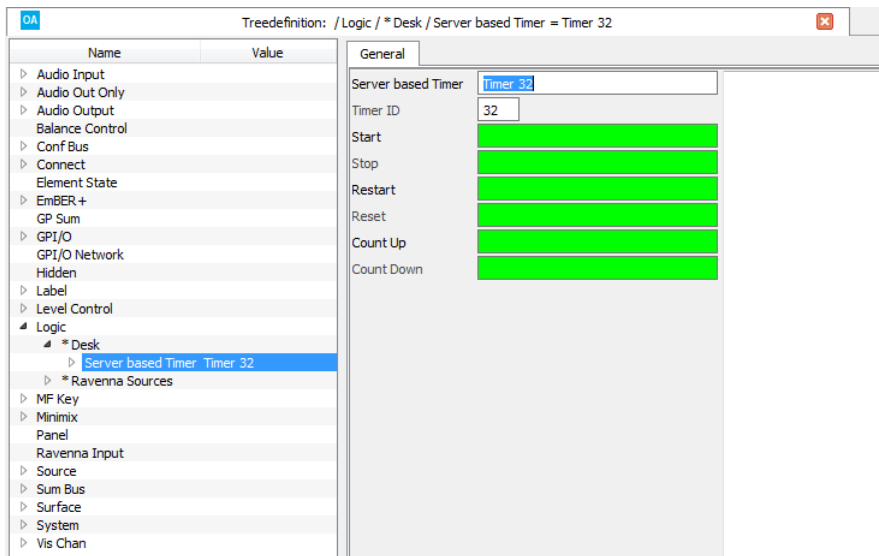
Server-based timers can be used to synchronize the operation across multiple VisTools. If a parameter such as the timer direction, preset or alarm time is changed from one VisTool, then this is displayed in the other instances. The displayed time is also synchronized.

Up to 32 independent server-based timers can be configured.

1. Start by adding a **Server based Timer** element to the Logic branch of the 'Tree Definition', and enter a **Timer ID** number from 1 to 32.

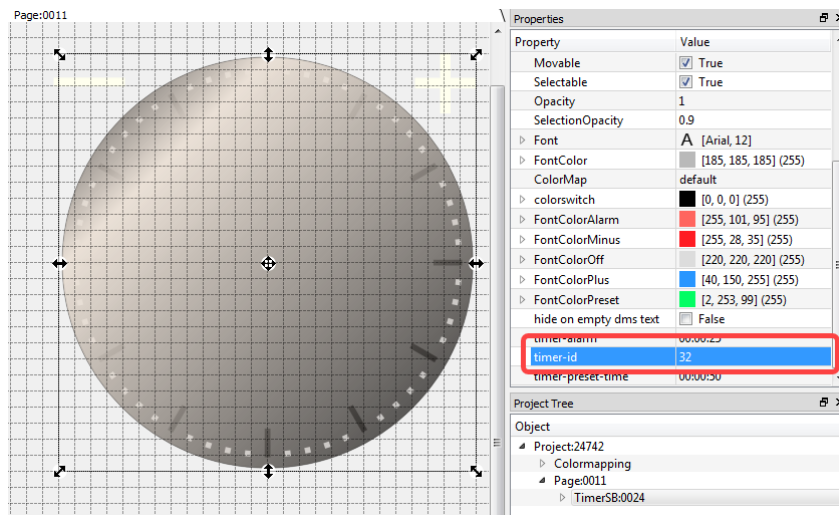
You can use the remaining fields to define how the timer is triggered.

ON-AIR Designer Server-based Timer (ID 32)



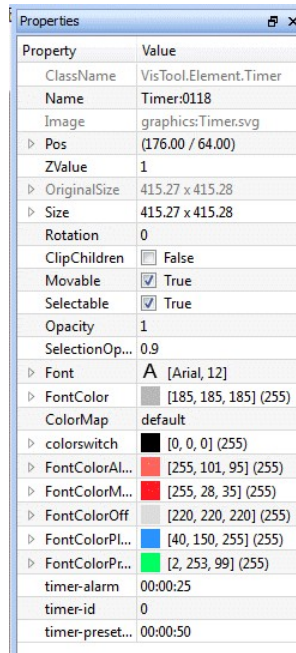
2. Then open VisTool Editor and insert a **Server based Timer** element in the usual manner. Edit the **Timer ID** in the 'Properties' panel to match the one entered in the ON-AIR Designer:


VisTool Editor Server-based Timer (linked to Timer 32)



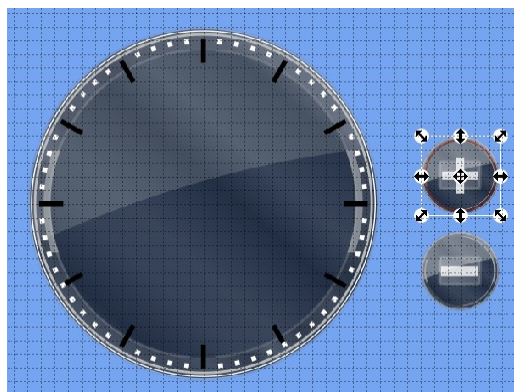
### 8.9.3 Timer Properties

In addition to the [general properties](#), **Timer** elements support the following special properties:



<b>FontColorAlarm</b>	Sets the font color used when the Timer reaches its <b>Alarm</b> value (set below). Click on  to open the color selector, or click on the arrow to access the individual color fields.
<b>FontColorMinus</b>	Sets the font color for Timer minus state.
<b>FontColorOff</b>	Sets the font color for Timer off state.
<b>FontColorPlus</b>	Sets the font color for Timer plus state.
<b>FontColorPreset</b>	Sets the font color used when the Timer reaches its <b>Preset-Time</b> (set below).
<b>timer-alarm</b>	Use this option to enter the <a href="#">Alarm</a> value for the Timer.
<b>timer-id</b>	Links the on-screen timer to a timer defined in the ON-AIR Designer configuration.
<b>timer-preset-time</b>	Use this option to enter the <a href="#">Preset</a> time for the Timer.

You can select the individual plus and minus Timer handles - press **SHIFT** and click to select. This allows you to customize the position of the handles:



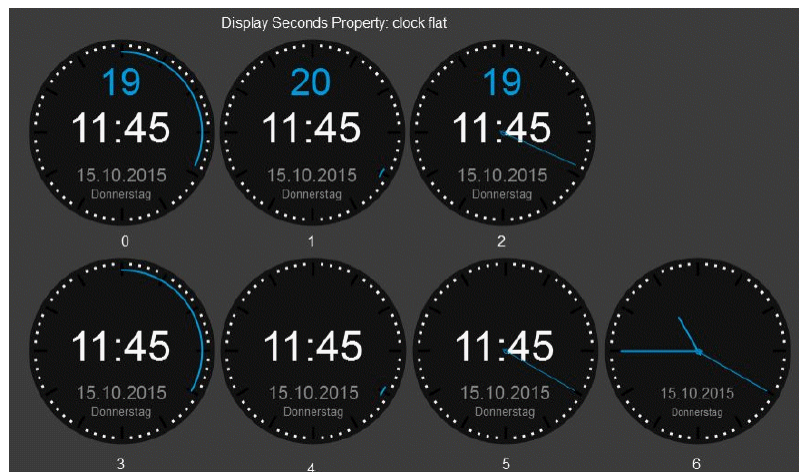
### 8.9.4 Clock Properties

In addition to the [general properties](#), **Clock** elements support the following special properties:

Property	Value
ClassName	VisTool.Element.Clock
Name	Clock:0740
Image	graphics:Clock flat.svg
Pos	(0.00 / 0.00)
ZValue	1
OriginalSize	412.00 x 412.00
Size	260.00 x 260.00
Rotation	0
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Opacity	1
SelectionOp...	0.9
Font	A [Source Sans Pro, 16]
FontColor	[227, 227, 227] (255)
ColorMap	LawoCI
colorswitch	[0, 0, 0] (255)
DateFormat	dd.MM.yyyy
DisplaySeco...	0
FontColorD...	[137, 137, 137] (255)
FontColorSec	[0, 154, 221] (255)
hide on em...	<input type="checkbox"/> False
OffsetinHours	0

<b>DateFormat</b>	Sets the format used to display the date. Enter a valid format using dd, MM and yy.
<b>DisplaySeconds</b>	Allows you to customize how the seconds are displayed. The options are numbered (as described below).
<b>FontColorDate</b>	Sets the font color used for the date. Click on  to open the color selector, or click on the arrow to access the individual color fields.
<b>FontColorSec</b>	Sets the font color used for the seconds.
<b>OffsetinHours</b>	Defines an offset (in hours).

#### Options for DisplaySeconds



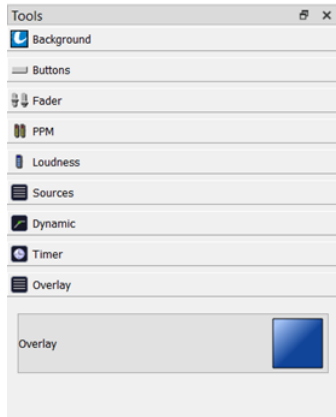
Figures 0, 1 & 2 display the seconds as a number (in blue). In addition:

- Figure 0 displays a continuous seconds ring (in blue). This is the default design.
- In Figure 1, a short seconds ring indicates the active second (in blue).
- In Figure 2, a seconds hand indicates the active second (in blue).

Figures 3, 4, & 5 are similar those above, but do not show the seconds as a number. Figure 6 removes all numerical digits and shows the time using hour, minute and seconds hands only.



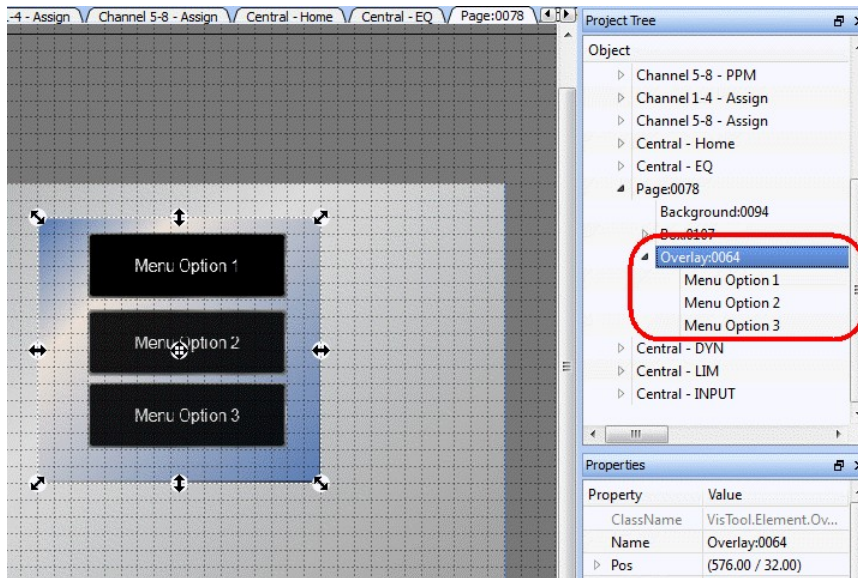
## 8.10 Overlay



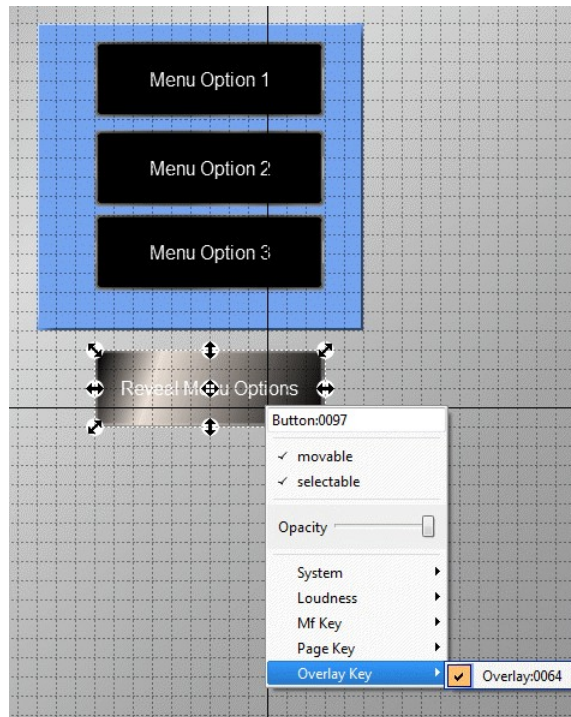
The **Overlay** element can be used to create an area that appears temporarily when the overlay is active. For example, to create a pop-up menu.

1. Add an **Overlay** element to the page in the [usual](#) manner.
2. Then add and [nest](#) the element(s) you wish to include in the Overlay - for example, the contents of the pop-up menu.

Any number of elements can be nested within a single Overlay, and Overlays can appear at any position on the page (using the [Set Popup Pos](#) option). In our example, three menu option buttons are nested in **Overlay:0064**:



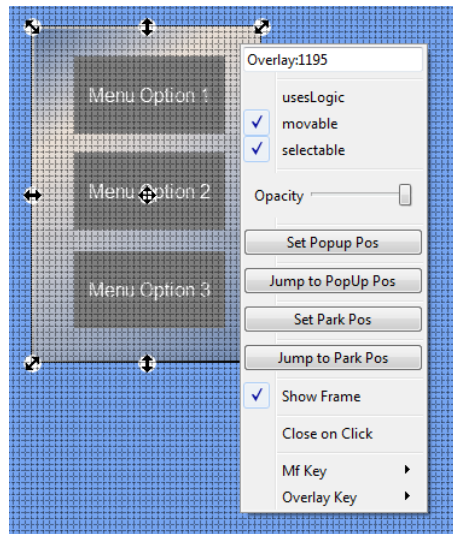
- Now select the trigger for the Overlay (e.g. another **Button** element) and right-click to assign it to the Overlay Key:



- Press CTRL + T (on your PC keyboard) to [test](#) the runtime operation - the Overlay and all of its linked elements should appear and disappear when you click on the trigger button.

### 8.10.1 Overlay Functions (Right-click)

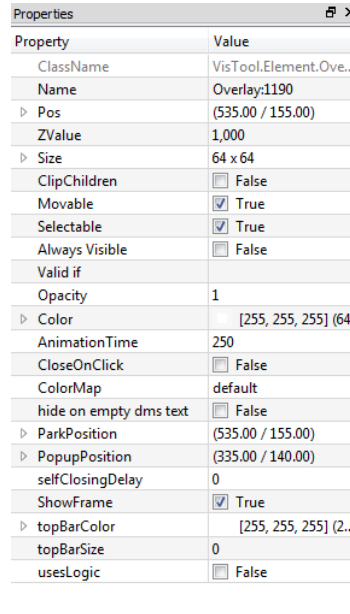
The following options are adjusted by right-clicking on the **Overlay** element:

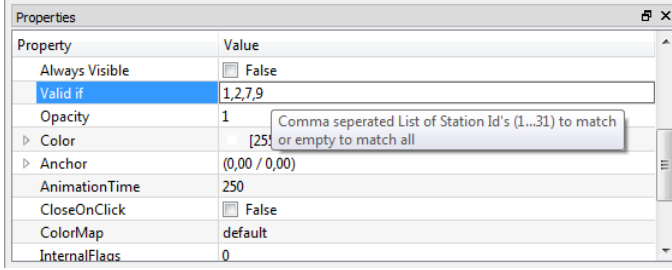



<b>Set Popup Pos</b>	This is the pop-up position where the Overlay element appears in VisTool when it is active. Click to set the x and y coordinates to the current position. The <b>PopUp Position</b> can be adjusted from the Overlay <a href="#">Properties</a> .
<b>Set Park Pos</b>	This is the parked position where the Overlay element appears in VisTool when it is not active. Click to set the x and y coordinates to the current position. The <b>Park Position</b> can be adjusted from the Overlay <a href="#">Properties</a> .
<b>Jump to Popup Pos</b> <b>Jump to Park Pos</b>	Once the <b>PopUp Position</b> and <b>Park Position</b> have been set, you can click on the <b>Jump to...</b> options to check the final positioning of the element.
<b>Show Frame</b>	When ticked, the Overlay element is displayed with a frame. Deselect this option to remove the frame.
<b>Close on Click</b>	Tick this option to hide the Overlay element when the user clicks on it. You can use the <a href="#">Animation Time</a> property to control the fade in/fade out time.
<b>MF Key</b>	Use this option to assign an <a href="#">MF Key</a> as the trigger for the Overlay.
<b>Overlay Key</b>	Use this option to interrogate or change the trigger for the Overlay. You will see all of the other Overlay Keys defined in the project.

### 8.10.2 Overlay Properties

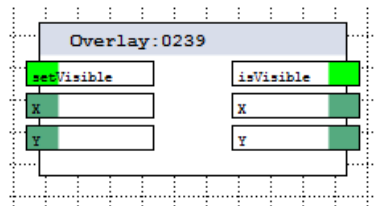
In addition to the [general properties](#), **Overlay** elements support the following special properties:



<b>Always Visible</b>	Determines whether the Overlay element is always visible (True) or conditional (False) according to the <b>Valid if</b> field.
<b>Valid if</b>	<p>Used in conjunction with the <b>Always Visible</b> field to make the appearance of the Overlay element conditional for a specific <a href="#">Station ID</a> (or IDs).</p> <p>Use a comma separated list to enter the Station ID numbers (in the range from 1 to 31). Or, leave the field empty to make the Overlay element non-conditional (match all Station IDs). In the example below, the Overlay will appear for Station IDs 1, 2, 7 and 9:</p> 
<b>Animation Time</b>	Sets the time the Overlay takes to fade in or fade out.
<b>CloseOnClick</b>	Enable this option to hide the Overlay element when the user clicks on it. You can use the <b>Animation Time</b> to control the fade in/fade out time.
<b>ParkPosition</b> <b>PopupPosition</b>	Set the inactive (park) and active (popup) position of the Overlay element (as described on the <a href="#">previous</a> page).
<b>selfClosingDelay</b>	Set this value >100, and the Overlay element will automatically close after the time specified (in milliseconds).
<b>ShowFrame</b>	When ticked, the Overlay element is displayed with a frame. Deselect this option to remove the frame.
<b>topBarColor</b>	Sets the color used for the top bar. Click on  to open the color selector, or click on the arrow to access the individual color fields.
<b>topBarSize</b>	Sets the size of the top bar.
<b>usesLogic</b>	Tick this option to add the Overlay element to the <a href="#">Logic Engine</a> .

### 8.10.3 Overlay Logic

When added to the [Logic Engine](#), an Overlay element provides the following logical inputs and outputs:



#### ➤ Logical Inputs

<b>setVisible</b>	Sets the visibility of the Overlay element. A logical false at the input makes the element disappear during runtime; a logical true signal displays the element.	boolean
<b>X and Y</b>	Sets the position of the Overlay element to the X and Y values.	double

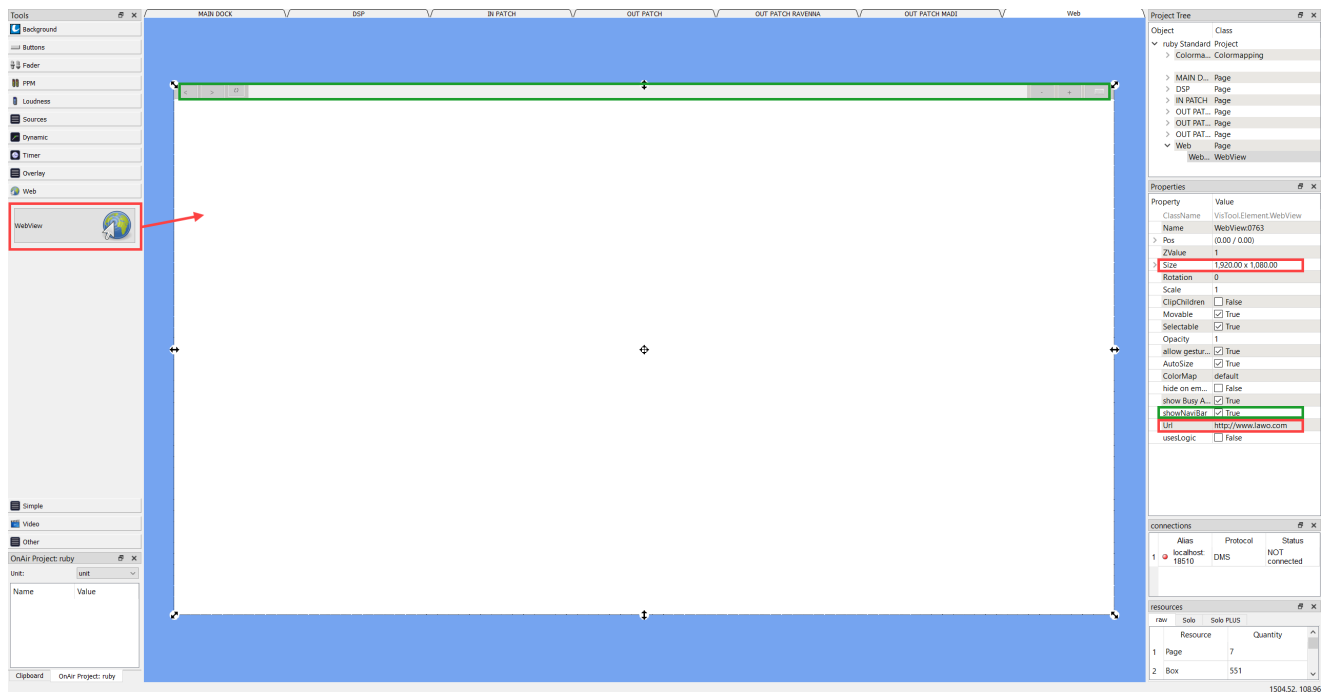
#### ➤ Logical Outputs

<b>isVisible</b>	Is active when the Overlay is visible.	boolean
<b>X and Y</b>	Shows the position of the Overlay element.	double

## 8.11 Web

The **WebView** element can be used to add a web page to the VisTool runtime. It is recommended to use only one element per page to avoid overloading the Graphics Processing Unit (GPU) of the host PC<sup>1</sup>. For the best user experience, it is recommended to set the element size to match that of the VisTool page.

1. Add a **WebView** element to the page in the [usual](#) manner.



Note that there are no functions available via a right-click, and so the page's **URL** and other options are defined from the [Properties](#) panel.

Each **WebView** element appears as a place holder (in VisTool Editor). In the example above, the navigation bar is enabled and the **URL** will open the Lawo home page. You can check how the design will look by entering [test mode](#) (press CTRL + T).

The settings used by the web engine can be customized by selecting **Util -> WebEngine Settings** (from the main menus). This allows you to enable (or disable) options such as javascript and plugins.

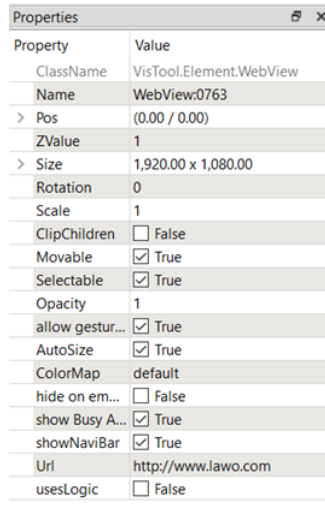
From Version 6.6.0, the web engine uses Chromium 80 (for improved graphics and speed). The settings used by the web cache can be customized by selecting **Util -> WebEngine Settings** (from the main menus). A button can be configured to clear the cache by linking a [Box](#) element to the "[Predefined User Text](#) -> Web -> Clear cache" option.

<sup>1</sup> Although it is possible to add multiple **WebView** elements to a VisTool page, this is not recommended (especially if dynamic content is to be shown) for the following reasons.

A visible change in a **WebView** element forces the whole element to be redrawn and, as a result, a redraw of the VisTool page. By default, each **WebView** element uses about 400MB of memory which can grow depending on the content. If several elements are present (whether visible or not), then this can cause a very high frame rate resulting in a very high graphics memory usage. This can cause a problem for onboard or weak GPUs (especially if the memory is shared).

### 8.11.1 WebView Properties

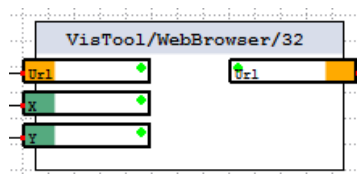
In addition to the [general properties](#), **WebView** elements support the following special properties:



<b>allow gestures</b>	When ticked, the web interface will support gestures (from a touch-screen interface).
<b>AutoSize</b>	When ticked, the web page will automatically re-size to fit the available space.
<b>show Busy Animation</b>	When ticked, the web page will indicate when it is busy (e.g. downloading a page).
<b>showNaviBar</b>	When ticked, the web page appears with a navigation bar, see <a href="#">VisTool Operation: Web Pages</a> . Untick this option to remove the navigation bar from the user's view.
<b>Url</b>	Enter the full address of the web page you wish to display. For example: <b>http://www.lawo.com</b> will open the Lawo home page. <b>http://192.168.101.240</b> will open the status page for a Lawo radio system using the default IP address.
<b>usesLogic</b>	Tick this option to add the WebView element to the <a href="#">Logic Engine</a> .

### 8.11.2 WebView Logic

When added to the [Logic Engine](#), a WebView element provides the following logical inputs and outputs:



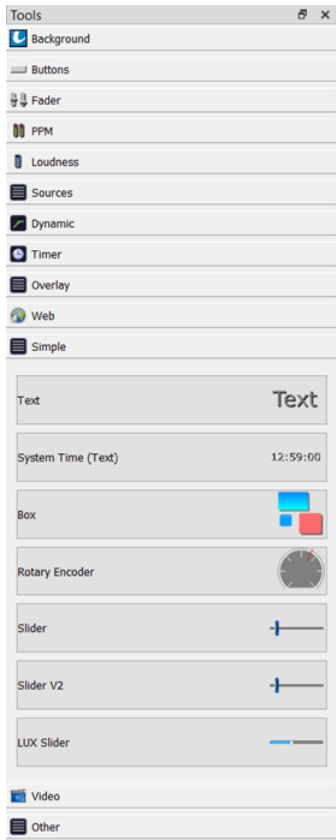
#### ➤ Logical Inputs

<b>Url</b>	Defines the target URL. If the input is not connected, then the URL is defined in the element's <a href="#">Properties</a> .	string
<b>X and Y</b>	Sets the position of the WebView element to the X and Y values.	double

#### ➤ Logical Outputs

<b>Url</b>	Shows the active URL of the WebView element.	string
------------	--	--------

## 8.12 Simple



This category includes the following elements.

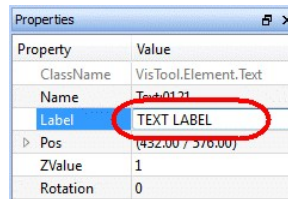
- **Text** - use this element to add text to a page (to annotate your design).
- **System Time (Text)** - displays the system time as a numeric.
- **Box** - these elements are ideal for framing other elements or creating on-screen buttons.
- **Rotary Encoder** - these elements can be used to create on-screen encoders to control variable parameters within an external device.
- **Slider** - these elements can be used to create on-screen sliders to control variable parameters within an external device. Both standard and LUX Sliders are supported. The LUX Slider offers more design options.

The available functions and properties vary depending on the type of element.



### 8.12.1 Text & System Time

The **Text** and **System Time** elements add text to a page, and therefore do not link to a function (via a right-click). To edit the text of a **Text** element, select the Text object and then edit the **Label** field in the Properties panel:



Property	Value
ClassName	VisTool.Element.Text
Name	Text.0121
Label	TEXT LABEL
Pos	(452.00 / 576.00)
ZValue	1
Rotation	0

The **System Time** is displayed as a numeric as described [earlier](#).

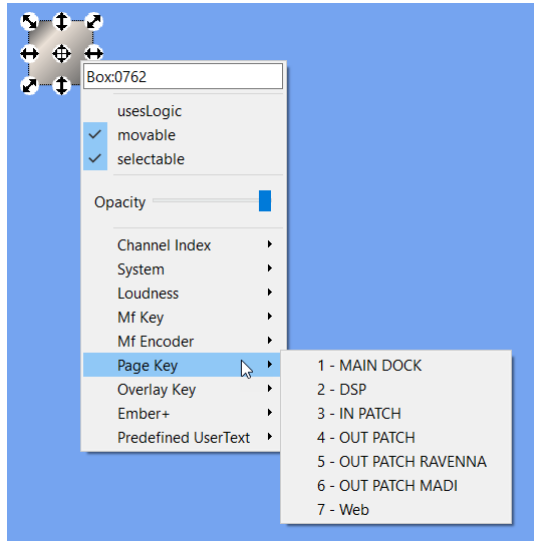
### 8.12.2 Box Elements

**Box** elements are ideal for framing other elements, such as a group of buttons, or creating on-screen [buttons](#) which action a function or provide signaling. They can be linked to any of the usual **Button** element [functions](#) plus a rotary encoder value (MF encoder), switched functions from an Ember+ provider or a predefined user text function.

Compared to **Button** elements, they are much faster to display and scale proportionally. You can also adjust the roundness of their corners and other visual properties.

#### Box Functions (Right-click)

Using a right-click, a **Box** element can be linked to any of following functions:



<b>Channel Index</b>	required for <b>Predefined User Text</b> -> <b>Channel</b> parameters.
<b>System</b>	a function which affects the application: minimize, fullscreen, close, etc.
<b>Loudness</b>	an <a href="#">integrated loudness measurement</a> command: Start, Stop, etc.
<b>MF Key</b>	an <a href="#">MF Key</a> defined by the ON-AIR Designer project.
<b>MF Encoder</b>	a rotary encoder on the console surface.
<b>Page Key</b>	a VisTool Editor <a href="#">Page Key</a> for on-screen page navigation.
<b>Overlay Key</b>	a key to open an <a href="#">Overlay</a> element.
<b>Ember+</b>	displays existing Ember+ parameter assignments.
<b>Predefined User Text</b>	a predefined user text function. Various options are supported. See <a href="#">Predefined User Text</a> .

The following options apply only to **Box** elements and are described in more detail over the next few pages: **Channel Index**, **MF Encoder**, **Ember+** and **Predefined User Text**.

#### >> MF Encoder

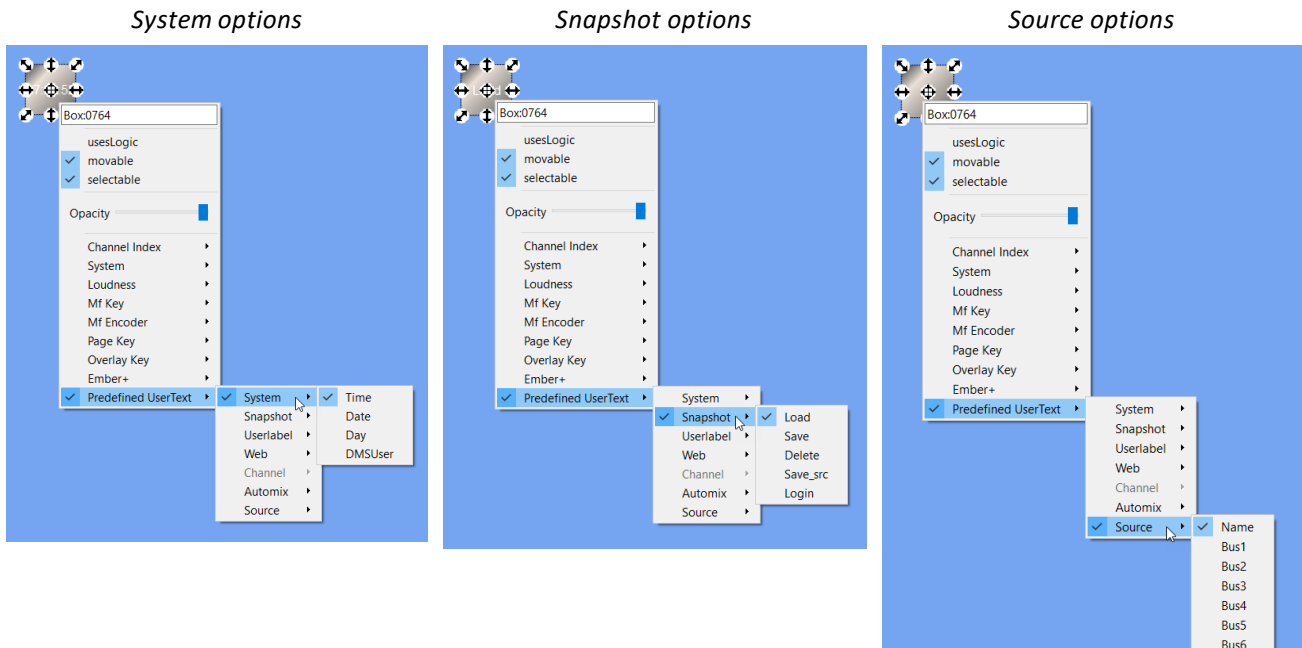
The **MF Encoder** branch can be used to link a **Box** element to a rotary encoder on the console surface. The available options are dependent on the ON-AIR Designer project. The options are divided into sub categories for each physical module.

#### >> Ember+

If an **Ember+** parameter has been assigned (via a drag and drop), then the **Ember+** branch can be used to interrogate the Ember+ provider and parameter. For information on how to make the assignments, see the [Ember+](#) chapter.

## 8. VisTool Elements

### >> Predefined User Text

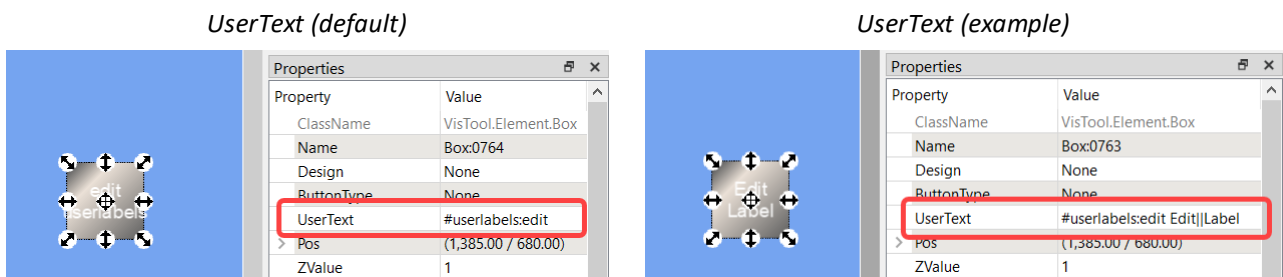


The **Predefined User Text** branch can be used to link a **Box** element to one of the following functions:

- **System** - to display the **Time, Date, Day** or **DMS User**.
- **Snapshot** - to perform a VisTool snapshot function (**Load, Save**, etc).
- **UserLabel** -> **Edit** - to open the user labels editor.
- **Web** -> **Clear cache** - to clear the cache of the web engine.
- **Channel** - to display the status of a channel parameter. (A **Channel Index** must first be assigned to specify the channel type/number).
- **Automix** - to display an automix parameter: **Weight, Speed, Group** or **On/off** status (for the source in access).
- **Source** - to display the **Name** or a **Bus** assignment (for the source in access).

### Editing the Button Labels

You can change the visible label for buttons with Predefined User Text by editing the **UserText** field in the Box Properties:



The label must not contain a leading space or a space at the end. To create an empty string, use a non-breaking space (ALT + 255). You can force a word-wrap by typing "|"; up to three lines of text are supported.

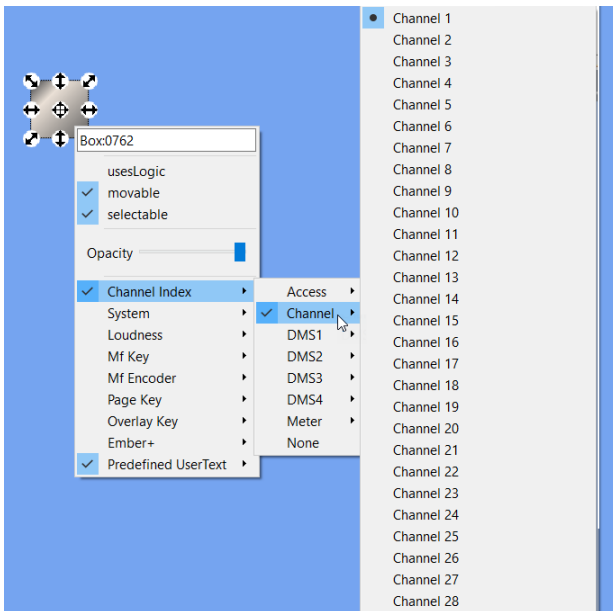
In the current release, this feature applies to the following functions: **UserLabel** -> **Edit** and **Snapshot** -> **Load, Save, Save\_Src, Delete, Login**.

## >> Displaying Channel Parameters

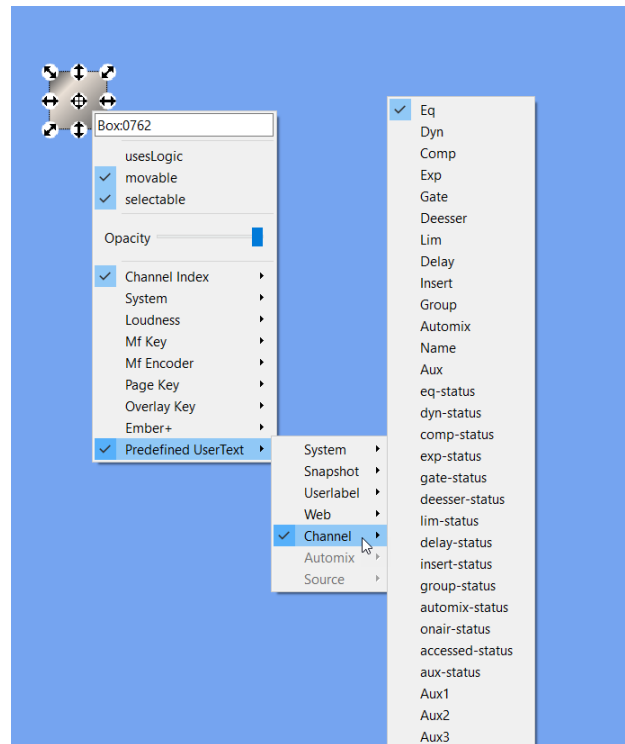
The **Predefined User Text -> Channel** options can be used to provide signalisation of channel parameters.

Start by assigning a **Channel Index** to the **Box** element, and then choose the parameter you wish to display from the **Predefined User Text -> Channel** menu.

Step 1 - choose the fader strip number



Step 2 - choose the parameter



### Assigning a Channel Index

When choosing the Channel Index, there are four possible categories:

- **Access** - the source in access.
- **Channel** - the source assigned to a specific fader strip. The number of faders available depends on the console specified by the ON-AIR Designer project.
- **DMS** - a DMS meter signal defined in the ON-AIR Designer project (via the "Audio Input -> DMS" branch of the 'Tree Definition'). These signals are freely configurable. The number available depends on the DMS system. If more than 64 DMS signals are available, then they are presented in groups (e.g. **DMS1** to **DMS4**), where each group contains 64 signals.
- **Meter (A, B, and 1 to 4)** - one of the six stereo main meter signals defined in the ON-AIR Designer project (via the "System -> Definition -> Parameter = Meter" branch of the 'Tree Definition').

Choose **None** to remove the Channel Index.

### Visualization of Parameters

The options in the **Predefined User Text -> Channel** menu allow parameter states to be indicated in different ways. For example, there are two possible ways to indicate EQ on/off:

- **Eq** displays the text "EQ", where the text color changes from ColorPre to ColorOff.
- **eq-status** displays the text "On" or "Off".

In each case, the colors are defined by the [Box Properties](#).

VisTool runtime



The list below describes all of the possible options:

- **Eq, Dyn, Comp, Exp, Gate, Deesser, Lim, Delay, Automix** - displays the DSP name, where the text changes color to indicate the on/off state.
  - DSP on - text color is ColorPre.
  - DSP off - text color is ColorOff.
- **Group** - displays the active VCA group.
  - VCA assigned - text color is ColorPre.
  - No VCA - display area is left empty.
- **Insert** - displays the active insert name.
  - Insert assigned - text color is ColorPre.
  - No insert - display area is left empty.
- **eq-status, dyn-status, comp-status, exp-status, gate-status, deesser-status, lim-status, delay-status, insert-status, group-status, automix-status**
  - State on - displays the text "On".
  - State off - displays the text "Off".
- **Name** - displays the Source Name.
  - ACCESS active - text color is ColorPre.
  - ACCESS inactive - text color is ColorOff.
- **accessed-status**
  - ACCESS active - displays the text "On".
  - ACCESS inactive - displays the text "Off".
- **Aux** - displays "AUX" if the channel has an aux. The text color indicates the aux type.
  - Audio source (source/sum) - text color is ColorPre.
  - Source Minimix, VCA or ProTools - text color is ColorOff.
- **aux-status**
  - Audio source (source/sum) - displays the text "On".
  - Source Minimix, VCA or ProTools - displays the text "Off".
- **Fadervalue** - displays the fader value (in dB).
- **onair-status**
  - Fader open - displays the text "On".
  - Fader closed - displays the text "Off".
- **Aux1 to Aux20** - displays the Aux Name.
  - Aux off - text color is ColorOff.
  - Aux on, pre-fader - text color is ColorPre.
  - Aux on, post-fader - text color is ColorPost.

### Box Properties

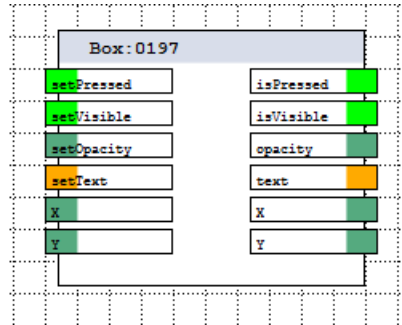
In addition to the [general properties](#), **Box** elements support the following special properties:

Property	Value
ClassName	VisTool.Element.Box
Name	MF 3 MF_SNAP LOAD
Design	Design-4
ButtonType	None
UserText	LOAD SNAP
> Pos	(50.00 / 850.00)
ZValue	1
> Size	160.00 x 60.00
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Glass Look	None
Metal Frame	0
Opacity	1
Roundness-X	0
Roundness-Y	0
TextAlign	Center
> Font	A [Source Sans Pro, 16]
ColorMap	LawoCI
FontColorMap	default-font
FrameColorMap	default
use DMS font color	<input type="checkbox"/> False
use DMS frame color	<input type="checkbox"/> False
> Color	■ [0, 0, 0] (255)
> ColorNone	■ [136, 136, 136] (255)
> ColorOff	■ [2, 253, 99] (255)
> ColorPost	■ [253, 253, 20] (255)
> ColorPre	■ [253, 2, 20] (255)
> FontColor	[255, 255, 255] (255)
> FontColorNone	[255, 255, 255] (255)
> FontColorOff	[255, 255, 255] (255)
> FontColorPost	[255, 255, 255] (255)
> FontColorPre	[255, 255, 255] (255)
Frame	0
> FrameColor	■ [0, 0, 0] (255)
manual-min	nan
manual-max	nan

<b>Design</b>	Choose a design from one of the drop-down options.
<b>Glass Look</b> <b>Glow On Press</b> <b>Metal Frame</b>	These options adjust the visual appearance of the design.
<b>Roundness-X and Y</b>	Enter a value to increase the roundness of the corners.
<b>Channel Index</b>	See <a href="#">Channel Index</a> .
<b>DisplaySnapshotName</b>	Tick this box to activate the <a href="#">Display SnapshotName</a> option.
<b>Reassignable</b>	See <a href="#">Reassignable "Buttons"</a> .
<b>usesLogic</b>	Tick this option to add the Box element to the <a href="#">Logic Engine</a> .

### Box Logic

When added to the [Logic Engine](#), a **Box** element provides the following logical inputs and outputs:



#### ➤ Logical Inputs

<b>setPressed</b>	Sets the Box active state (defined by <b>color pre</b> / <b>color off</b> in the Vistool Editor). Note that this input only changes the visual state of the box in order to visualize a boolean input. The <b>isPressed</b> output is not influenced by this input.	boolean
<b>setVisible</b>	Sets the visibility of the Box element. A logical false on the input makes the box disappear in the runtime, while a true signal shows the box.	boolean
<b>setOpacity</b>	Sets the opacity to the input value.	double
<b>setText</b>	Displays a string	string
<b>X and Y</b>	Sets the position of the Box element to the X and Y values.	double

#### ➤ Logical Outputs

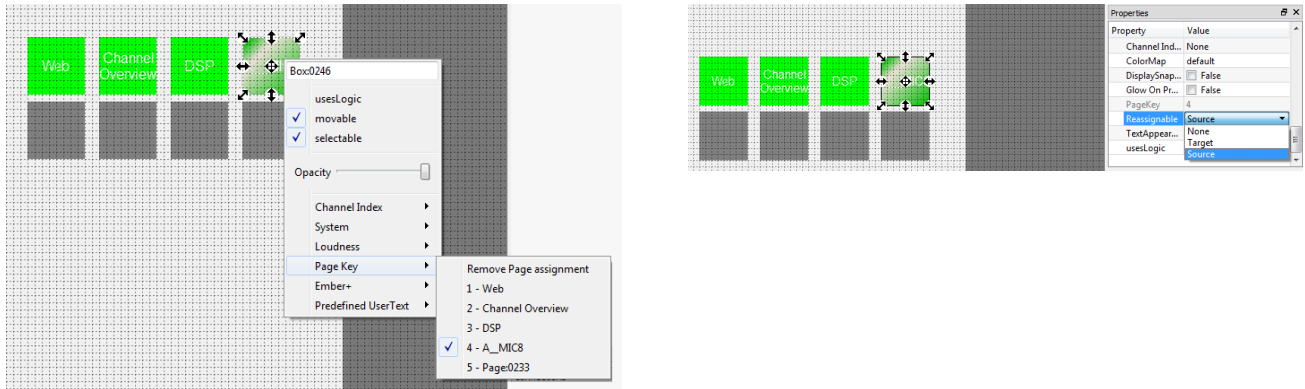
<b>isPressed</b>	Shows whether the Box is pressed manually. The <b>setPressed</b> input does not influence the output	boolean
<b>isVisible</b>	Is active when the Box is visible.	boolean
<b>opacity</b>	Shows the opacity value of the Box element.	double
<b>text</b>	Displays the <b>UserText</b> entered in the <a href="#">Properties</a> for the Box element.	string
<b>X and Y</b>	Shows the position of the Box element.	double

### Reassignable Buttons

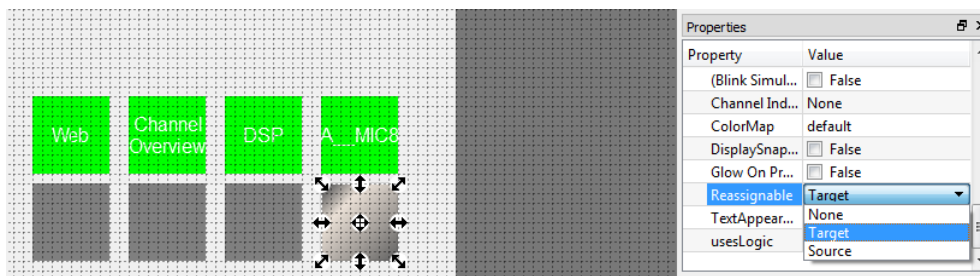
The **Reassignable** property for each Box element can be used to create a bank of reassignable buttons for the VisTool operator. For example, to create a [Matrix Control](#) panel.

The system works by defining a set of source and target Box elements as follows, and can be combined with any type of function (DMS, Ember+, etc).

First, define the function of each source "button" (with a right-click on the Box element), and set its **Reassignable** property to **Source**:



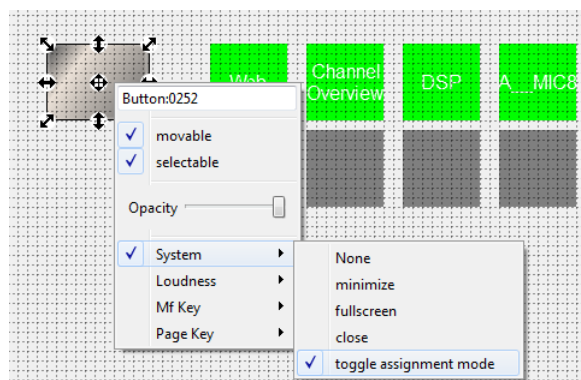
Then, define each target "button" by setting its **Reassignable** property to **Target**:



Note that the target's other properties (e.g. Color, Size, etc.) are not affected by the reassignment operation and, therefore, can be customized in the usual manner.

Repeat for each of your source and target buttons to create a "pool" of sources and targets. During runtime, an operator can now enter the reassignment mode to map sources to targets.

To configure a dedicated VisTool key to enter (and exit) the reassignment mode, create another Button or Box element and assign it to the **System** -> **toggle assignment mode**:





### 8.12.3 Rotary Encoder

Rotary Encoder elements can be used to create on-screen encoders to control variable parameters.

#### Rotary Encoder Functions (Right-click)

Each rotary encoder can be linked either to a console encoder (defined in the system configuration) or a parameter from an Ember+ provider.

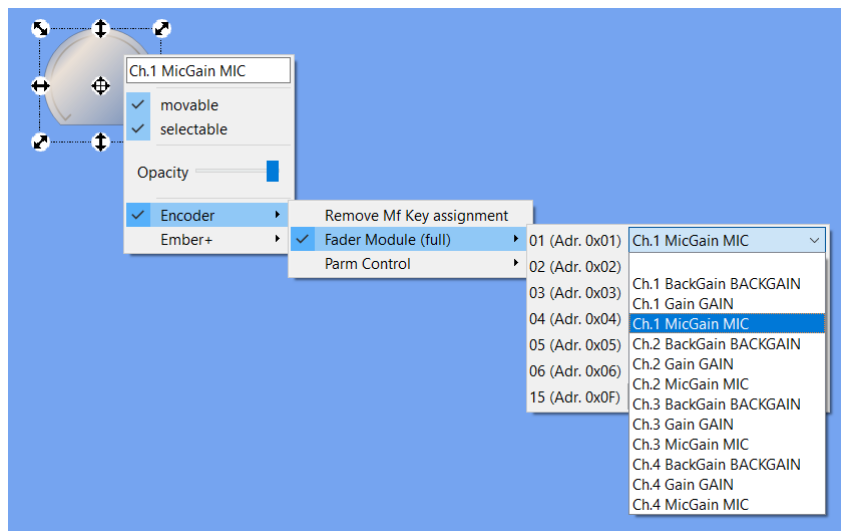
#### Console Encoder

In this category, the possible options are as follows. In each case, the parameters must be defined in the ON-AIR Designer project before they appear in VisTool Editor.

- **MicGain**, **Gain** or **BackGain** - for each configured Fader Module.
- **Parm Control** parameters for the Source In Access - requires a PARM control panel to be defined in the "Frames -> Screen" menu.

To make the assignment, right-click and choose an option from the **Encoder** menu:

*Rotary Encoder assigned to Channel 1 Mic Gain*

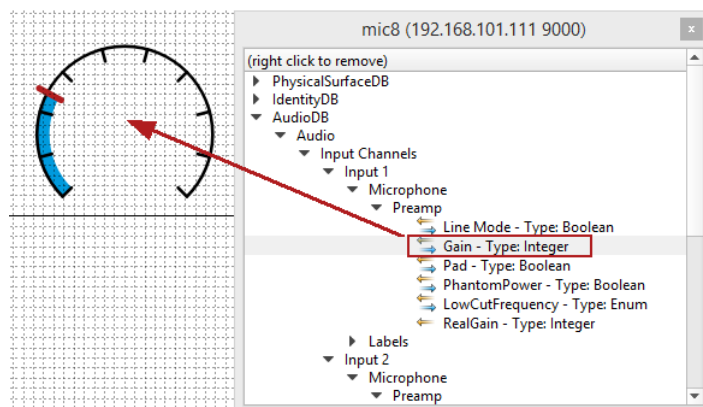


#### Ember+ Parameter

In this category, you can assign any parameter from an Ember+ provider except "PeakValue". A prerequisite for numerical values is that the minimum and maximum values must be transmitted by the provider.

The Ember+ provider must be defined in the [Ember+ Devices list](#) and [added](#) to the project. The assignment is made using a drag and drop from the Ember+ provider tree:

*Rotary Encoder assigned to A\_mic8 Input 1 Mic Gain*

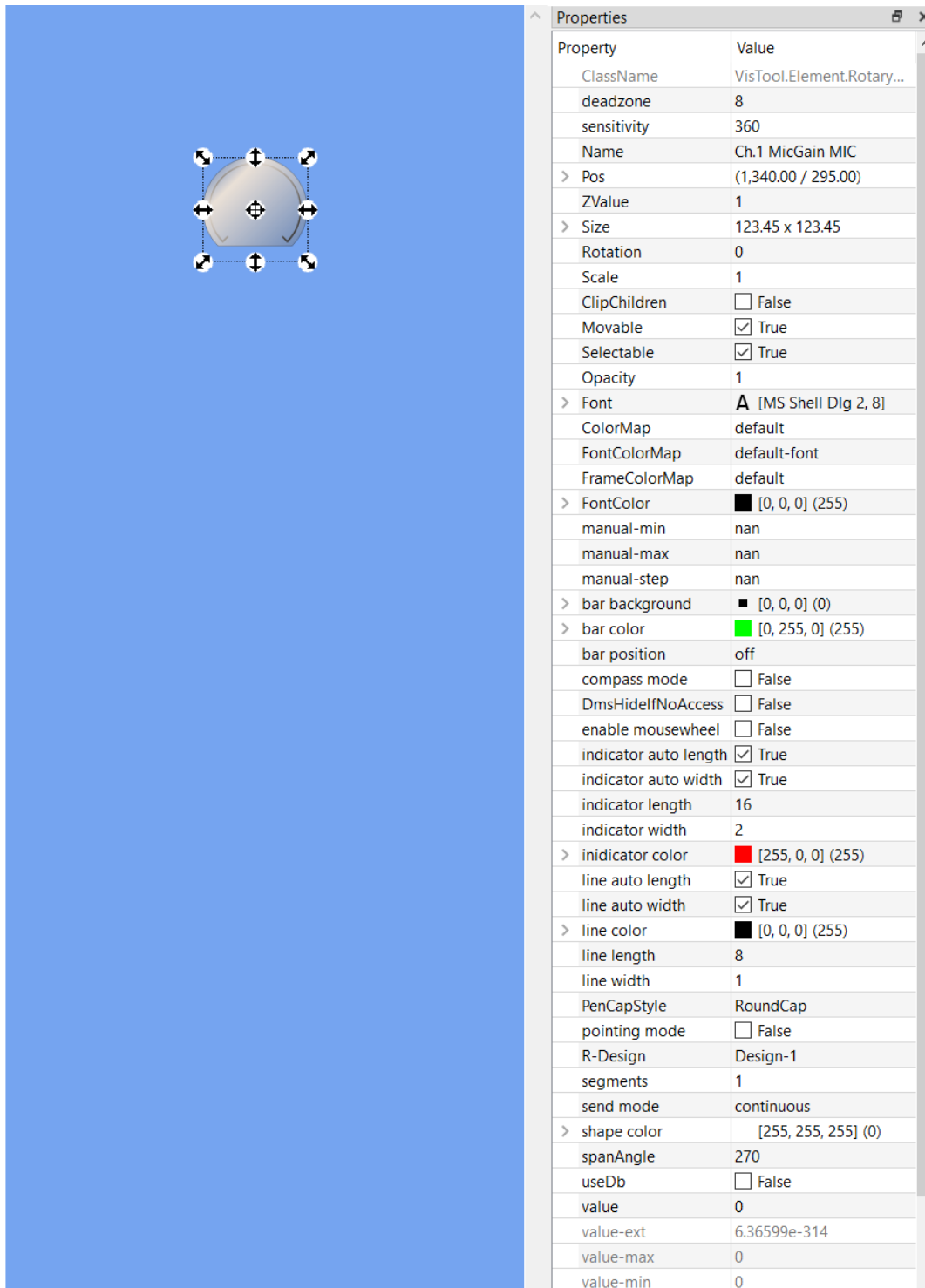


Right-click on the Rotary Encoder and choose **Ember+** to interrogate an existing assignment.

### Rotary Encoder Properties

In addition to the [general properties](#), Rotary Encoder elements support the following special properties.

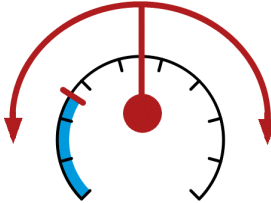
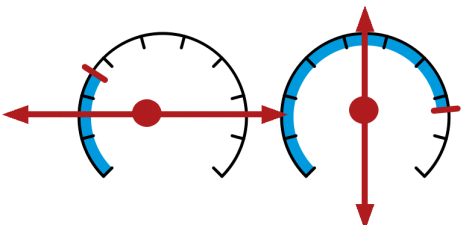
*Rotary Encoder (properties)*



Property	Value
ClassName	VisTool.Element.Rotary...
deadzone	8
sensitivity	360
Name	Ch.1 MicGain MIC
> Pos	(1,340.00 / 295.00)
ZValue	1
> Size	123.45 x 123.45
Rotation	0
Scale	1
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Opacity	1
> Font	A [MS Shell Dlg 2, 8]
ColorMap	default
FontColorMap	default-font
FrameColorMap	default
> FontColor	■ [0, 0, 0] (255)
manual-min	nan
manual-max	nan
manual-step	nan
> bar background	■ [0, 0, 0] (0)
> bar color	■ [0, 255, 0] (255)
bar position	off
compass mode	<input type="checkbox"/> False
DmsHidelfNoAccess	<input type="checkbox"/> False
enable mousewheel	<input type="checkbox"/> False
indicator auto length	<input checked="" type="checkbox"/> True
indicator auto width	<input checked="" type="checkbox"/> True
indicator length	16
indicator width	2
> inidicator color	■ [255, 0, 0] (255)
line auto length	<input checked="" type="checkbox"/> True
line auto width	<input checked="" type="checkbox"/> True
> line color	■ [0, 0, 0] (255)
line length	8
line width	1
PenCapStyle	RoundCap
pointing mode	<input type="checkbox"/> False
R-Design	Design-1
segments	1
send mode	continuous
> shape color	[255, 255, 255] (0)
spanAngle	270
useDb	<input type="checkbox"/> False
value	0
value-ext	6.36599e-314
value-max	0
value-min	0

## Behaviour

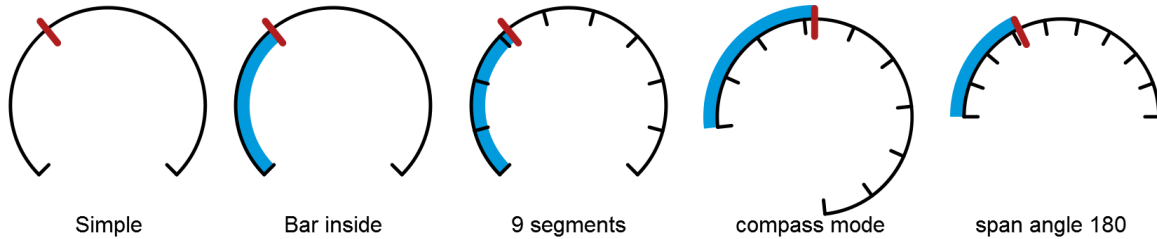
The following properties define the behaviour of the encoder.

<b>deadzone</b>	Sets the number of pixels, in coordinates, from which a reaction will take place. The higher the number, the further the encoder must be moved, from its starting position, to achieve a change in value.
<b>sensitivity</b>	Sets the number of pixels, in coordinates, which define a complete 360 degree rotation. The higher the number, the more rotation is required to turn the encoder.
<b>manual-min</b> <b>manual-max</b>	Sets the value range of the rotary encoder. Can be used if an Ember+ device does not provide minimum and maximum values, or if the control range is to be restricted. By default, "nan" (not a number) is entered so that the values are set automatically by the external provider. If a numerical value is entered, then the range is set manually and the provider's data is ignored. In this instance, the <b>manual-min/max</b> values must be within the provider's range. If the provider uses a scaling factor, then the configured values must be calculated using the factor. For example, if the factor is 32 and the required minimum is -128dB, then the <b>manual-min</b> value must be set to -4096.
<b>manual-step</b>	Sets the step size for adjustable values.
<b>DmsHideIfNoAccess</b>	When ticked, the rotary encoder is hidden from view until its assigned parameter is available. For example, to hide Parm Control parameters when there is no source in access.
<b>enable mousewheel</b>	Tick this option to allow the rotary encoder to be operated by a mouse-wheel.
<b>pointing mode</b>	Tick this option to enable pointing mode. In this mode, the value is changed by a rotational movement of the mouse outside the circle - click and hold, and then turn:  When pointing mode is disabled, the value is changed by moving the mouse upwards or downwards relative to the initial position- click and hold, and then move up or down: 
<b>send mode</b>	Enter the mode in which values will be transferred to the Ember + provider: <ul style="list-style-type: none"> <li>• <b>Continuous</b> = the values are transmitted continuously.</li> <li>• <b>On release</b> = the value is transmitted on release only.</li> </ul> The latter may result in a break in audio for say a fader value, but may be a good idea if you want to choose a stereo mode.
<b>value</b>	This field shows the current internal value of the rotary encoder (between 0 and 1).
<b>value-ext</b>	This field shows the current provider value of the encoder (between the <b>value-max</b> and <b>value-min</b> ).
<b>value-max/min</b>	This field shows the maximum and minimum values of the encoder as supplied by the external provider.

### Appearance

The following properties affect the appearance of the encoder.

*Rotary Encoder Examples*



<b>shape color</b>	Sets the background color. In the examples above, this is set to white.
<b>spanAngle</b>	Sets the expansion angle in degrees.
<b>bar color</b> <b>bar position</b>	The Rotary Encoder includes a bar that shows the amount of deviation from the origin. In the examples above, this is shown in blue. Use <b>bar color</b> to set the color and <b>bar position</b> to set the position. The position options are <b>off</b> (no bar), <b>inside</b> (bar appears inside the baseline), <b>centered</b> (bar is centered over the baseline); <b>outside</b> (bar appears outside the baseline).
<b>indicator color</b>	The Rotary Encoder includes an indicator that shows the current position. Use <b>indicator color</b> to set the color of the indicator. In the examples above, this is set to red.
<b>indicator auto length</b> <b>indicator auto width</b> <b>indicator length</b> <b>indicator width</b>	The indicator width and length can be adjusted either automatically or manually: <ul style="list-style-type: none"> <li>• Enable the <b>auto width</b> and/or <b>auto length</b> options to adjust the dimensions automatically, according to the size of the element.</li> <li>• Alternatively, turn off the <b>auto</b> option(s) and use <b>indicator width</b> and/or <b>indicator length</b> to set the dimensions manually. Note that the manual value is ignored if the <b>auto</b> option is enabled.</li> </ul>
<b>compass mode</b>	Tick this option to enable compass mode. In this mode, the indicator remains in a fixed position, while the rotary scale moves around it. Deselect compass mode for the reverse operation.

### Scale

For a Rotary Encoder element, the scale must be designed manually. In the examples above, the scale lines (and baseline) are shown in black.

<b>segments</b>	Defines the number of segments to be used for the scale.
<b>useDb</b>	Sets the spacing of the scale lines: false = linear; true = dB scale.
<b>line position</b>	Sets the position of the scale lines. The options are <b>off</b> (no scale lines), <b>top</b> (scale lines appear above the baseline), <b>centered</b> (scale lines are centered over the baseline), <b>bottom</b> (bar appears below the baseline).
<b>line color</b>	Use <b>line color</b> to set the color of the scale lines (and baseline).
<b>line auto length</b> <b>line auto width</b> <b>line length</b> <b>line width</b>	The width and length of the scale lines (and baseline) can be adjusted either automatically or manually. These options work in the same way as the indicator width and length described above.

### Text Label

The **Font** and **FontColor** properties are applied if a label is assigned to the Rotary Encoder. If the **Alpha** value is less than 5, the label is not displayed.

### 8.12.4 Slider

Slider elements can be used to create on-screen sliders to control variable parameters. Both standard and LUX Sliders are supported. The LUX Slider offers more design options.

#### Slider Functions (Right-click)

Each slider can be linked either to a console encoder (defined in the system configuration) or a parameter from an Ember+ provider.

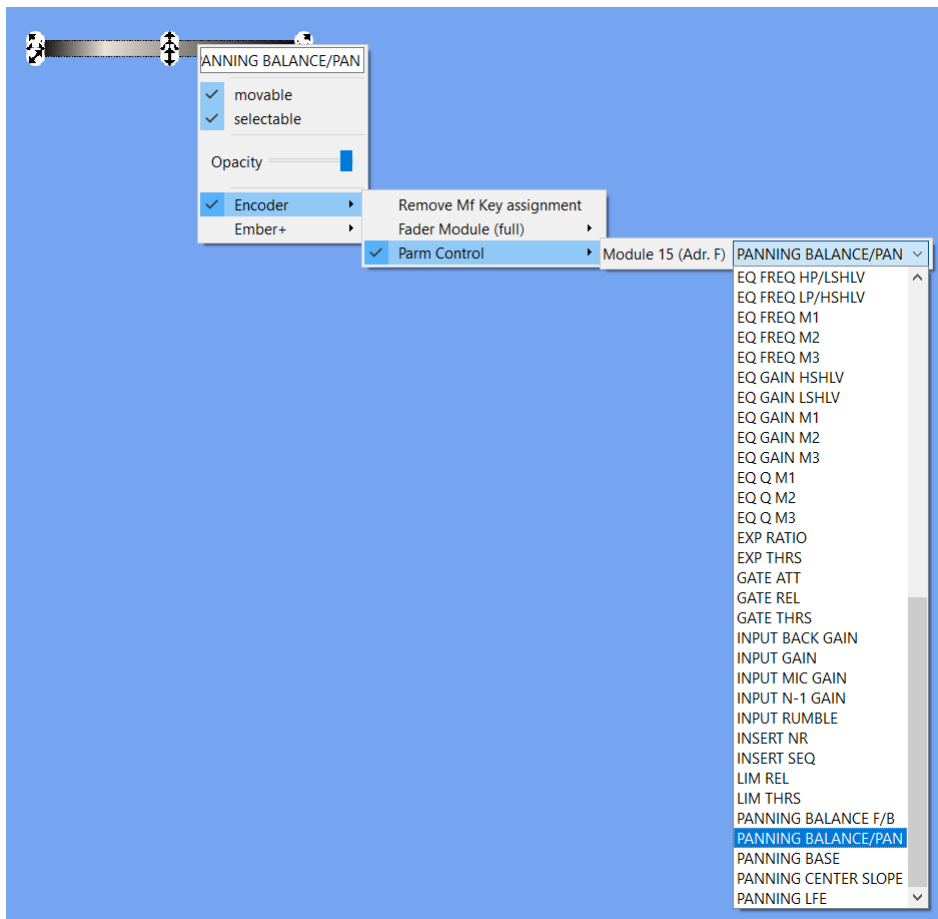
#### Console Encoder

In this category, the possible options are as follows. In each case, the parameters must be defined in the ON-AIR Designer project before they appear in VisTool Editor.

- **MicGain, Gain or BackGain** - for each configured Fader Module.
- **Parm Control** parameters for the Source In Access - requires a PARM control panel to be defined in the "Frames -> Screen" menu.

To make the assignment, right-click and choose an option from the **Encoder** menu:

*LUX Slider assigned to Balance/Pan (for the Source in Access)*



#### Ember+ Parameter

In this category, you can assign any parameter from an Ember+ provider except "PeakValue". A prerequisite for numerical values is that the minimum and maximum values must be transmitted by the provider, and parameters must be Read/Write.

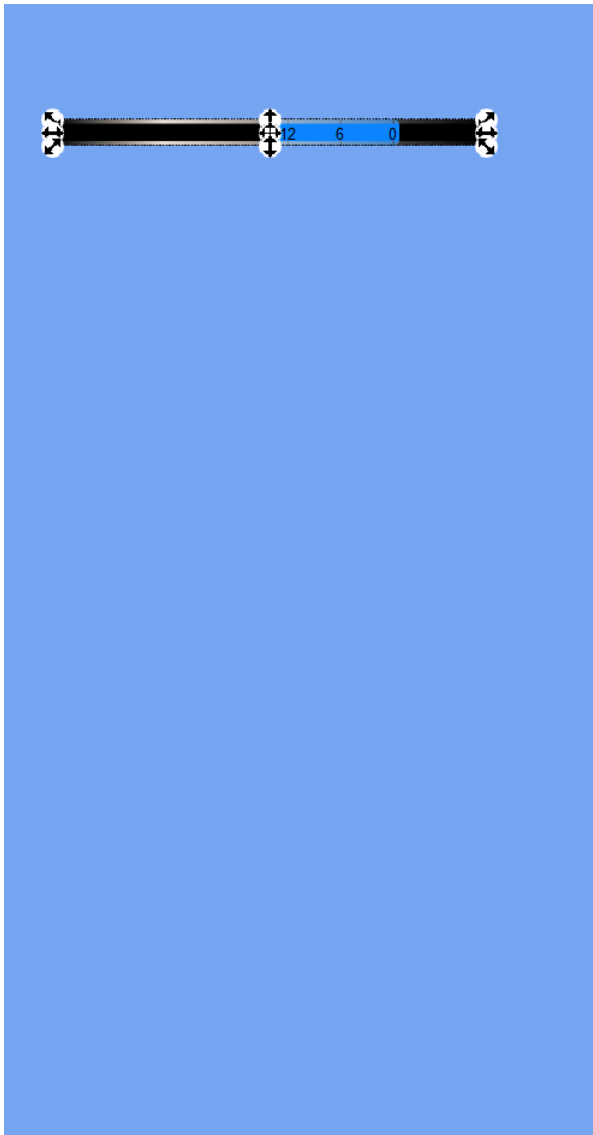
The Ember+ provider must be defined in the [Ember+ Devices list](#) and [added](#) to the project. The assignment is made using a drag and drop from the Ember+ provider tree.

Right-click on the Slider and choose **Ember+** to interrogate an existing assignment.

## Slider Properties

In addition to the [general properties](#), Slider elements support the following special properties.

*LUX Slider (properties)*



Property	Value
ClassName	VisTool.Element.LUX Sli...
deadzone	8
sensitivity	360
Name	AUX 01 GAIN
> Pos	(1,430.00 / 835.00)
ZValue	1
> Size	447.33 x 27.11
Rotation	0
Scale	1
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Opacity	1
Roundness-X	6
Roundness-Y	6
> Font	A [Arial, 12]
ColorMap	default
FontColorMap	default-font
FrameColorMap	default
manual-min	nan
manual-max	nan
manual-step	nan
> bar color	<span style="color: blue;">■</span> [10, 132, 255] (255)
bar origin	0
DmsHidelfNoAccess	<input type="checkbox"/> False
enable mousewheel	<input type="checkbox"/> False
> indicator color	<span style="color: blue;">■</span> [255, 255, 255] (255)
> line color	<span style="color: black;">■</span> [0, 0, 0] (255)
margin	0
Orientation	Horizontal
pointing mode	<input type="checkbox"/> False
> shape color	<span style="color: black;">■</span> [0, 0, 0] (255)
slider-scale	Aux-Bus-gain[dB] (Pow...
value	0.5
value-ext	0
value-max	0
value-min	0

## Behaviour

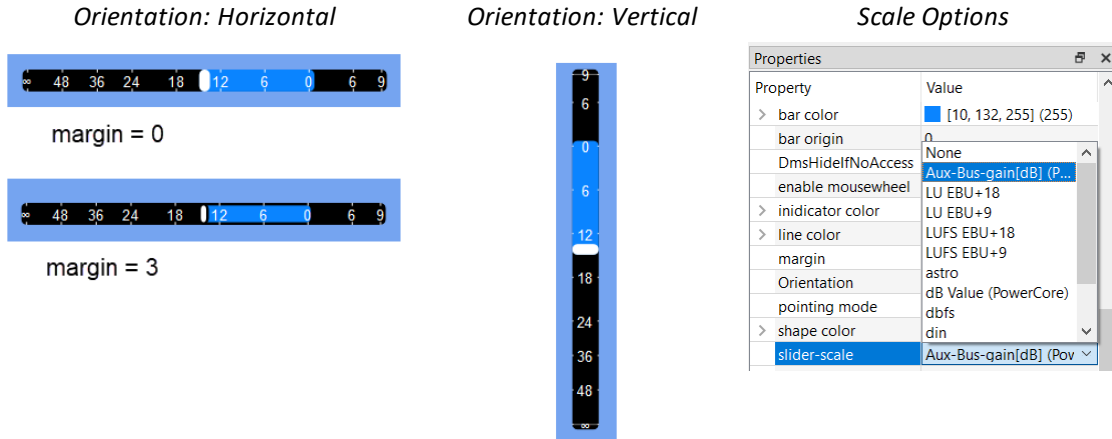
The following properties define the behaviour of the slider.

<b>deadzone</b>	Sets the number of pixels, in coordinates, from which a reaction will take place. The higher the number, the further the slider must be moved, from its starting position, to achieve a change in value.
<b>sensitivity</b>	Sets the number of pixels, in coordinates, which define a complete open to close "rotation". The higher the number, the further the slider must be moved to open or close the fader.
<b>manual-min</b> <b>manual-max</b>	<p>Sets the value range of the slider. Can be used if an Ember+ device does not provide minimum and maximum values, or if the control range is to be restricted.</p> <p>By default, "nan" (not a number) is entered so that the values are set automatically by the external provider.</p> <p>If a numerical value is entered, then the range is set manually and the provider's data is ignored. In this instance, the <b>manual-min/max</b> values must be within the provider's range. If the provider uses a scaling factor, then the configured values must be calculated using the factor. For example, if the factor is 32 and the required minimum is -128dB, then the <b>manual-min</b> value must be set to -4096.</p>
<b>manual-step</b>	Sets the step size for adjustable values.
<b>DmsHideIfNoAccess</b>	When ticked, the slider is hidden from view until its assigned parameter is available. For example, to hide Parm Control parameters when there is no source in access.
<b>enable mousewheel</b>	Tick this option to allow the slider to be operated by a mouse-wheel.
<b>pointing mode</b>	<p>Tick this option to enable pointing mode. In this mode, the value jumps to the absolute position of the pointing device. So, for example, the operator can touch a slider position to instantly jump to that value.</p> <p>When disabled, the operator must click and drag to adjust the value - the slider moves left/right or up/down relative to its current position.</p>
<b>value</b>	This field shows the current internal value of the slider (between 0 and 1).
<b>value-ext</b>	This field shows the current provider value of the slider (between the <b>value-max</b> and <b>value-min</b> ).
<b>value-max/min</b>	This field shows the maximum and minimum values of the slider as supplied by the external provider.

## Appearance

The following properties define the appearance of the slider. These vary depending on the slider type.

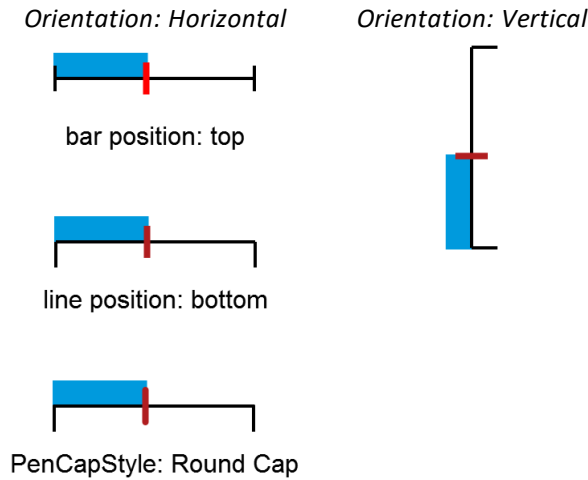
### Appearance: LUX Slider



<b>Orientation</b>	Sets the orientation of the slider to either Horizontal or Vertical. After a change of orientation, the slider dimensions must be adjusted manually.
<b>shape color</b>	Sets the background color. In the examples above, this is set to black.
<b>Roundness-X and Y</b>	Enter a value to increase the roundness of the corners.
<b>bar color</b> <b>bar origin</b>	The LUX Slider includes a bar that shows the amount of deviation from the origin. In the examples above, this is shown in blue. Use <b>bar color</b> to set the color and <b>bar origin</b> to set the origin.
<b>indicator color</b>	The LUX Slider includes an indicator that shows the current position. Use <b>indicator color</b> to set the color. In the examples above, this is set to white.
<b>indicator auto size</b> <b>indicator roundness</b> <b>indicator size</b>	In the current release, the indicator size and shape are adjusted automatically and so these options have no function.
<b>margin</b>	The <b>margin</b> sets the width of the margin around the bar and indicator (in pixels). You can this to separate the bar and indicator from the edge of the element.
<b>slider scale</b> <b>line color</b>	Use the <b>slider scale</b> menu to add a scale. The options are pre-defined (as shown above). Select <b>None</b> to turn off the scale. Use <b>line color</b> to set the color. In the examples above, this is set to white.

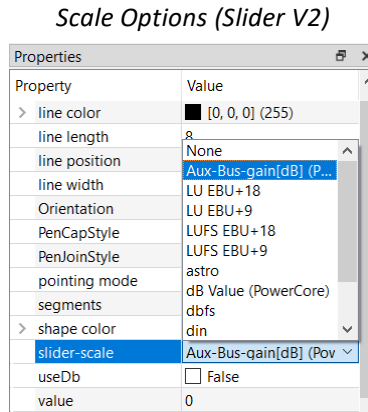


Appearance: Slider & Slider V2



<b>Orientation</b>	Sets the orientation of the slider to either Horizontal or Vertical. After a change of orientation, the Slider dimensions must be adjusted manually.
<b>shape color</b>	Sets the background color. In the examples above, this is set to white.
<b>bar color</b> <b>bar position</b>	The slider includes a bar that shows the amount of deviation from the origin. In the examples above, this is shown in blue. Use <b>bar color</b> to set the color and <b>bar position</b> to set the position. The position options are <b>off</b> (no bar), <b>top</b> (bar appears above the baseline), <b>centered</b> (bar is centered over the baseline); <b>bottom</b> (bar appears below the baseline).
<b>indicator color</b>	The slider includes an indicator that shows the current position. Use <b>indicator color</b> to set the color of the indicator. In the examples above, this is set to red.
<b>indicator auto length</b> <b>indicator auto width</b> <b>indicator length</b> <b>indicator width</b>	The indicator width and length can be adjusted either automatically or manually: <ul style="list-style-type: none"> <li>• Enable the <b>auto width</b> and/or <b>auto length</b> options to adjust the dimensions automatically, according to the size of the element.</li> <li>• Alternatively, turn off the <b>auto</b> option(s) and use <b>indicator width</b> and/or <b>indicator length</b> to set the dimensions manually. Note that the manual value is ignored if the <b>auto</b> option is enabled.</li> </ul>
<b>PenCapStyle</b> <b>PenJoinStyle</b>	Use these options to change the style of the indicator.

Scale Options: Slider V2



For a Slider V2 element, the scale is handled in a similar manner to the LUX Slider.

<b>slider scale</b>	Use the <b>slider scale</b> menu to add a scale. The options are pre-defined (as shown above). Select <b>None</b> to turn off the scale.
<b>line color</b>	Use <b>line color</b> to set the color of the scale. The size and spacing is defined automatically and so the following options have no function:
<b>segments</b>	
<b>useDb</b>	
<b>line position</b>	
<b>line auto length</b>	
<b>line auto width</b>	
<b>line length</b>	
<b>line width</b>	

Scale Options: Slider

*Scale Design (Slider)*



Segments: 9



useDb

For a Slider element, the scale must be designed manually. In the examples above, the scale lines (and baseline) are shown in black.

<b>segments</b>	Defines the number of segments to be used for the scale.
<b>useDb</b>	Sets the spacing of the scale lines: false = linear; true = dB scale.
<b>line position</b>	Sets the position of the scale lines. The options are <b>off</b> (no scale lines), <b>top</b> (scale lines appear above the baseline), <b>centered</b> (scale lines are centered over the baseline), <b>bottom</b> (bar appears below the baseline).
<b>line color</b>	Use <b>line color</b> to set the color of the scale lines (and baseline).
<b>line auto length</b> <b>line auto width</b> <b>line length</b> <b>line width</b>	The width and length of the scale lines (and baseline) can be adjusted either automatically or manually. These options work in the same way as the indicator width and length described above.

### 8.13 Video

The **Vline Thumbnail** element can be used to display video streams from a V\_\_line device in VisTool. There is only one element option in this category called **Vline Thumbnail**. Multiple elements can be added, and positioned and sized in the usual manner.

Note that thumbnails received as an Ember+ stream, via the Gadget Server, can be assigned to a **Box** element directly (using a drag and drop from the Ember+ tree). In this instance, a separate **Vline Thumbnail** element is not required. See [Using Ember+](#) and [Assigning Parameters to VisTool Elements](#).

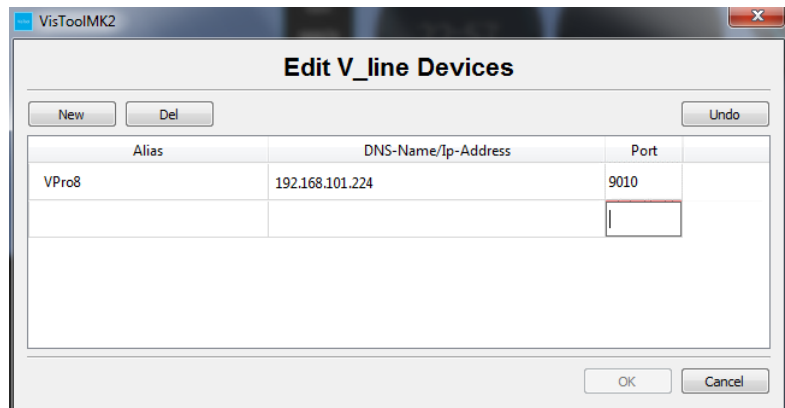
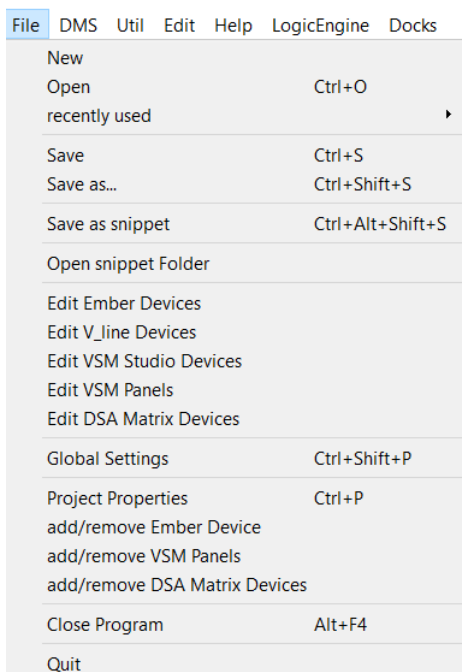
#### Preparation & Configuration

The V\_\_line device must be connected to the network and specified in the [V\\_line Devices](#) list. Each element can then be assigned to a specific input or output channel (from 1 to 8).

#### 8.13.1 The V\_line Devices List

This list defines the TCP/IP settings of each V\_\_line device on the network. Devices must be added to this list before their video channels can be [assigned](#) to a **Vline Thumbnail** element.

1. Select **File -> Edit V\_line Devices** to open the V\_\_line Devices list:



2. Select **New** to add a new entry.
3. Click in each of the fields to enter an **Alias** name, the **DNS-Name** or **IP-Address**, and **Port** number of each V\_\_line unit - take care to press Enter to confirm each field.

Use **Del** to delete an existing line. **Undo** will undo all changes. Select **OK** to save all changes and exit.

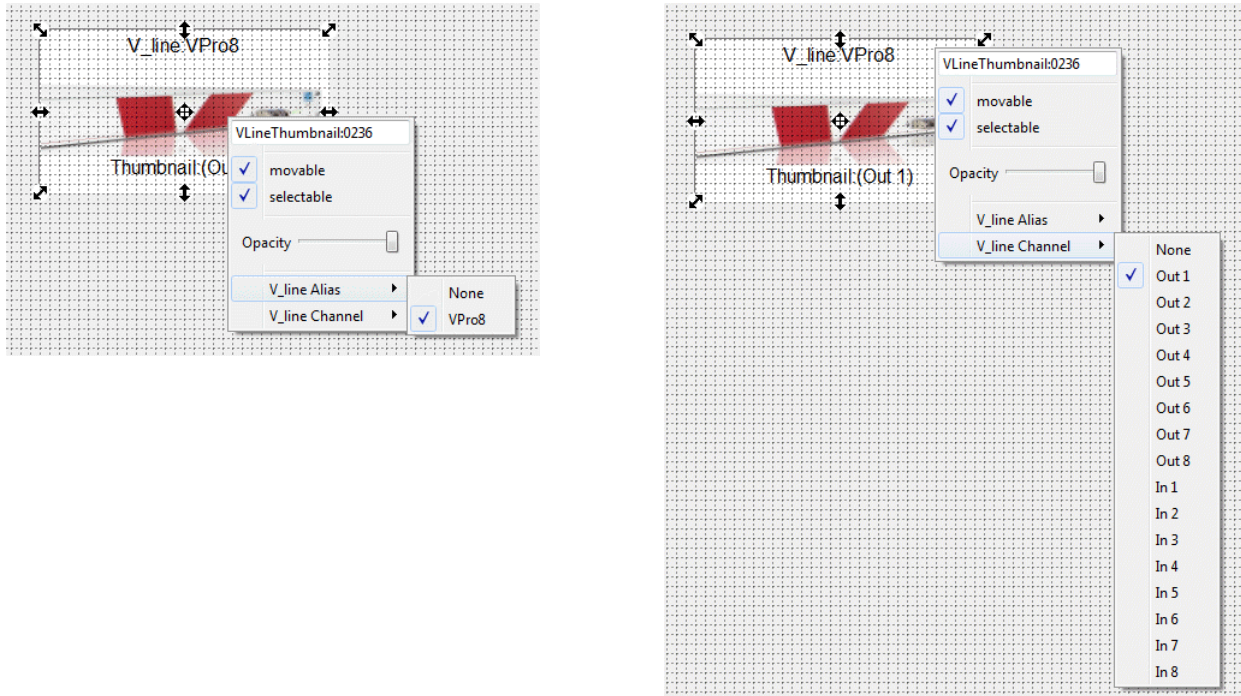
In our example, a single **V\_\_pro8** has been configured.

The V\_line devices list is stored in the **V\_line\_series\_Hosts.ini** file. This may be shared between VisTool instances. See [Using a Shared File path for ".ini" Files](#). You can also deactivate a host by [editing](#) the file.

### 8.13.2 Vline Functions (Right-click)

Right-click on the **Vline Thumbnail** element to select a device from the **V\_line Alias** drop-down menu. Note that you can only add devices which have already been defined in the [V\\_line Devices list](#).

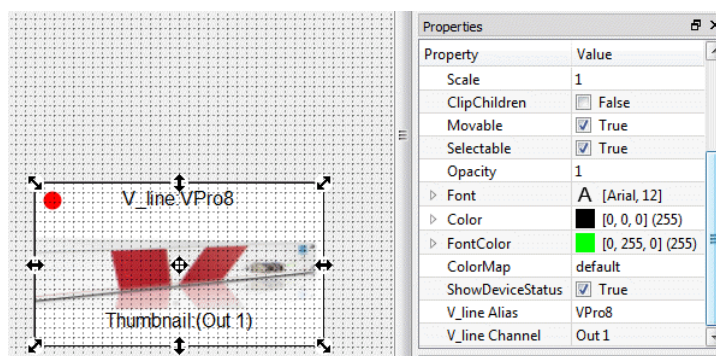
Then assign an input or output channel from 1 to 8:



Video will be streamed from the connected **V\_line** device. The thumbnail element appears as a place holder until a valid connection is established. Once this occurs and video is present, you will see the live video in VisTool Editor.

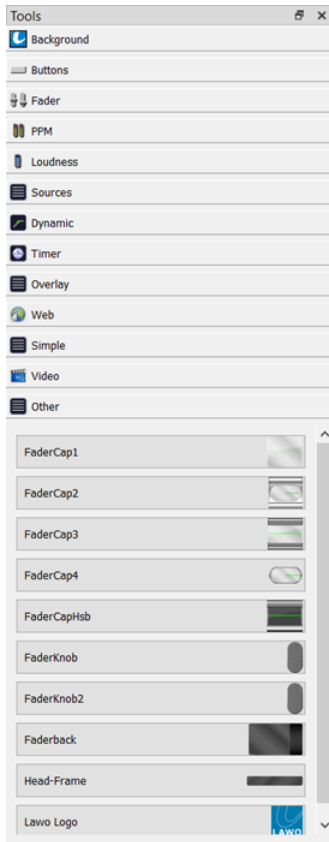
### 8.13.3 Vline Properties

In addition to the [general properties](#), **Vline Thumbnail** elements support the following special property:



<b>ShowDeviceStatus</b>	Tick this box to add a device status indicator to the video thumbnail. This is a circle which appears at the top left of the thumbnail and is color-coded to indicate the status of the network connection: green = valid; red = invalid or no connection.
-------------------------	--

## 8.14 Other



This category of elements contains other predefined elements which you may use to customize your design.

The elements offer the same [Functions](#) and [Properties](#) as **Buttons** but cannot indicate a state (color).

Options include a range of fader caps, fader knobs and standard icons.

## 8.15 User

You can use this category to add user graphics to your design.

All standard graphic file formats are supported (e.g. JPEG, PNG, TIFF, BITMAP).

You must be logged in to Windows as an Administrator to drag and drop graphics files into VisTool Editor. If not, then Windows may not permit the drag and drop operation.

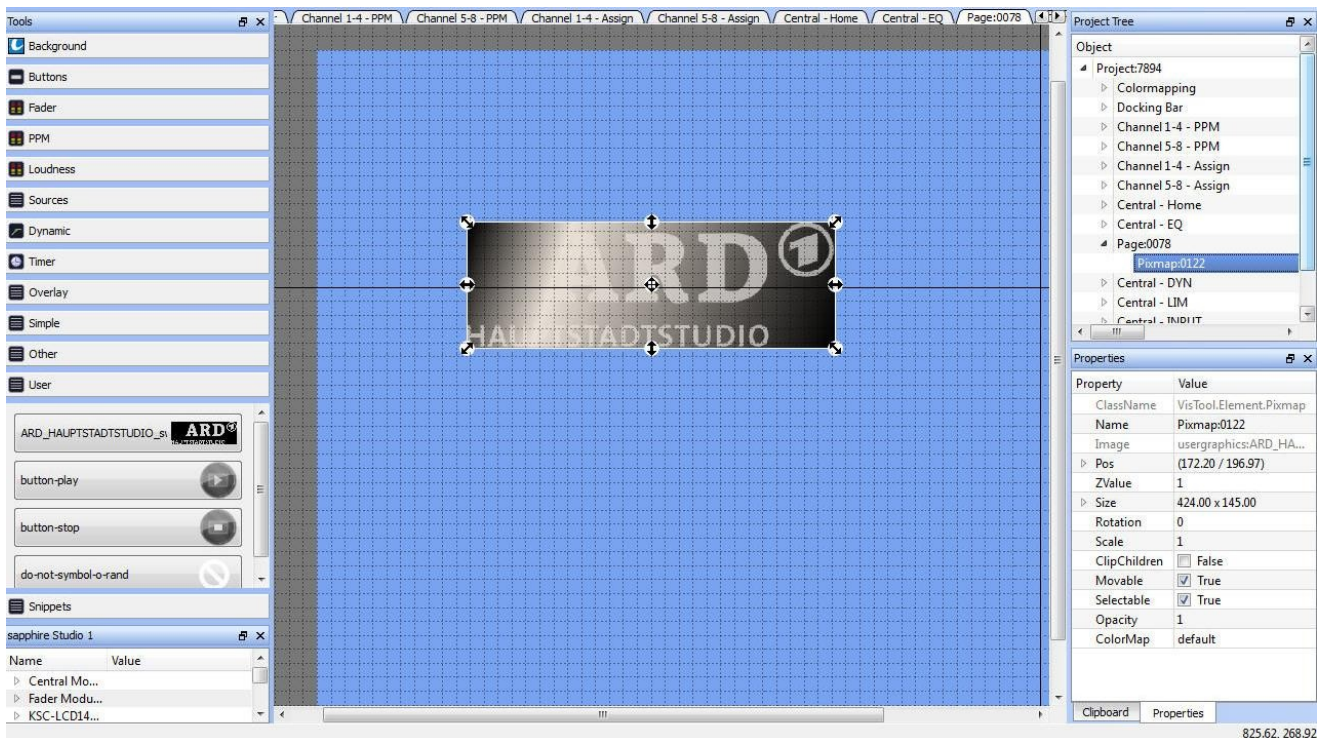
When using formats such as jpg or png, make sure that the graphic is set to the required size, as scaling the graphic may affect the image resolution.

User graphic files are saved with the VisTool Project within the "User Graphics" directory.

1. First, drag and drop the graphic file onto a **VisTool Editor** page - this adds the graphic to the **User** elements list.
2. Then [add](#) the element to any page in the usual manner.

**User** elements offers the same [Function Options](#) and [Properties](#) as **Buttons** but cannot indicate a state (color).

*User Element (example)*

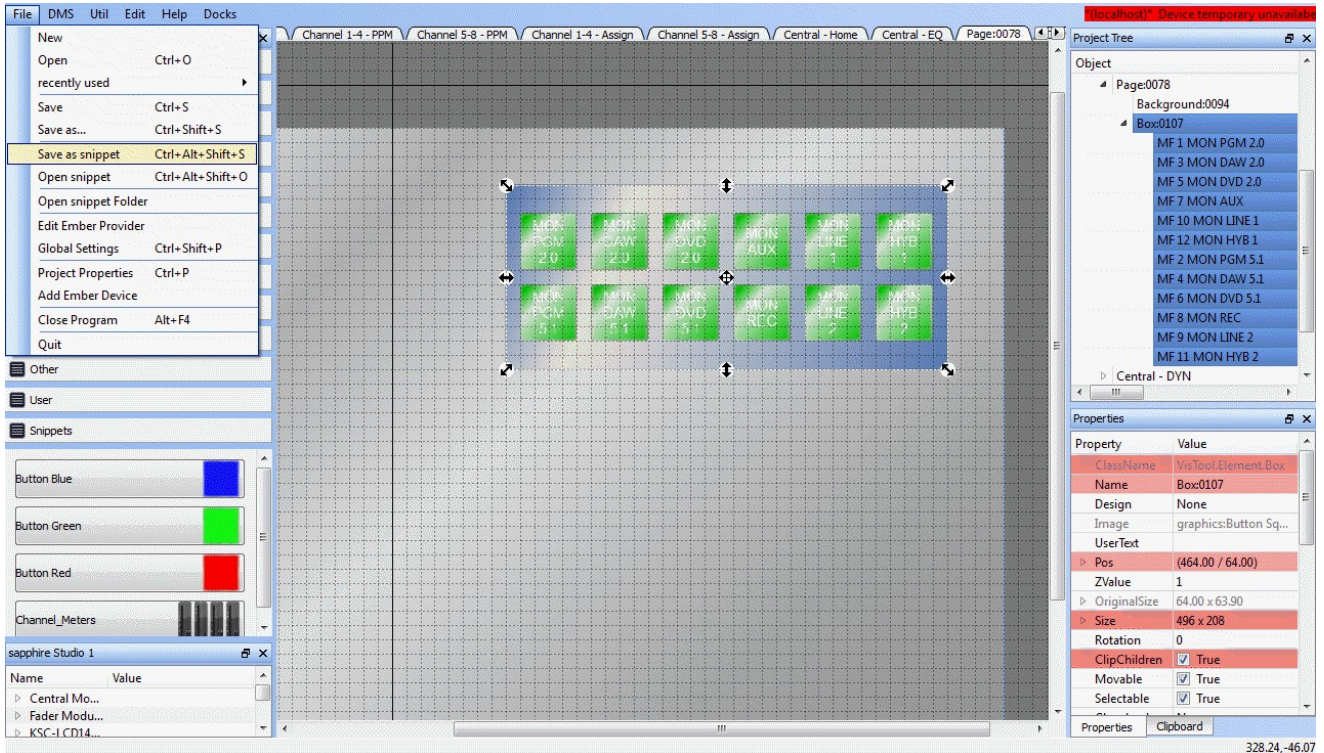


## 8.16 Snippets

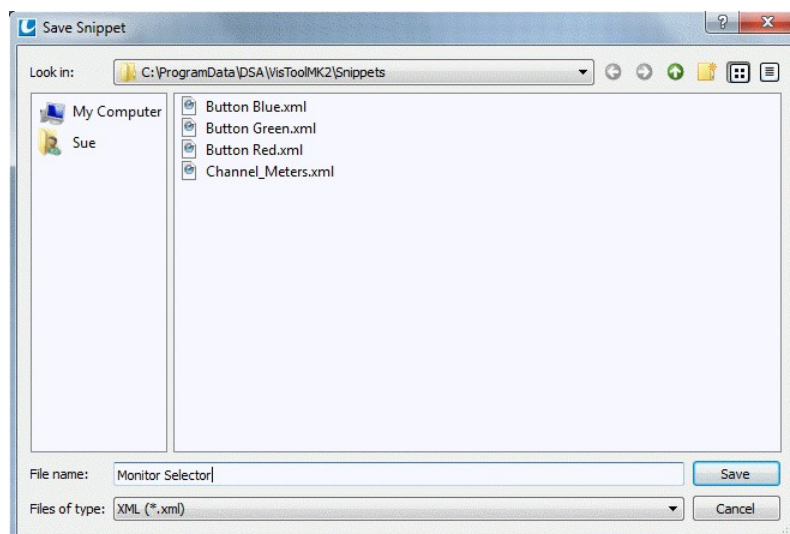
Snippets store elements, or combinations of elements in files which are stored independently of the VisTool project. This allows you to reuse their content within any configuration at a later date. For example, to create a library of useful components such as custom buttons, meters, timers, selector panels, etc.

### 8.16.1 Saving a Snippet

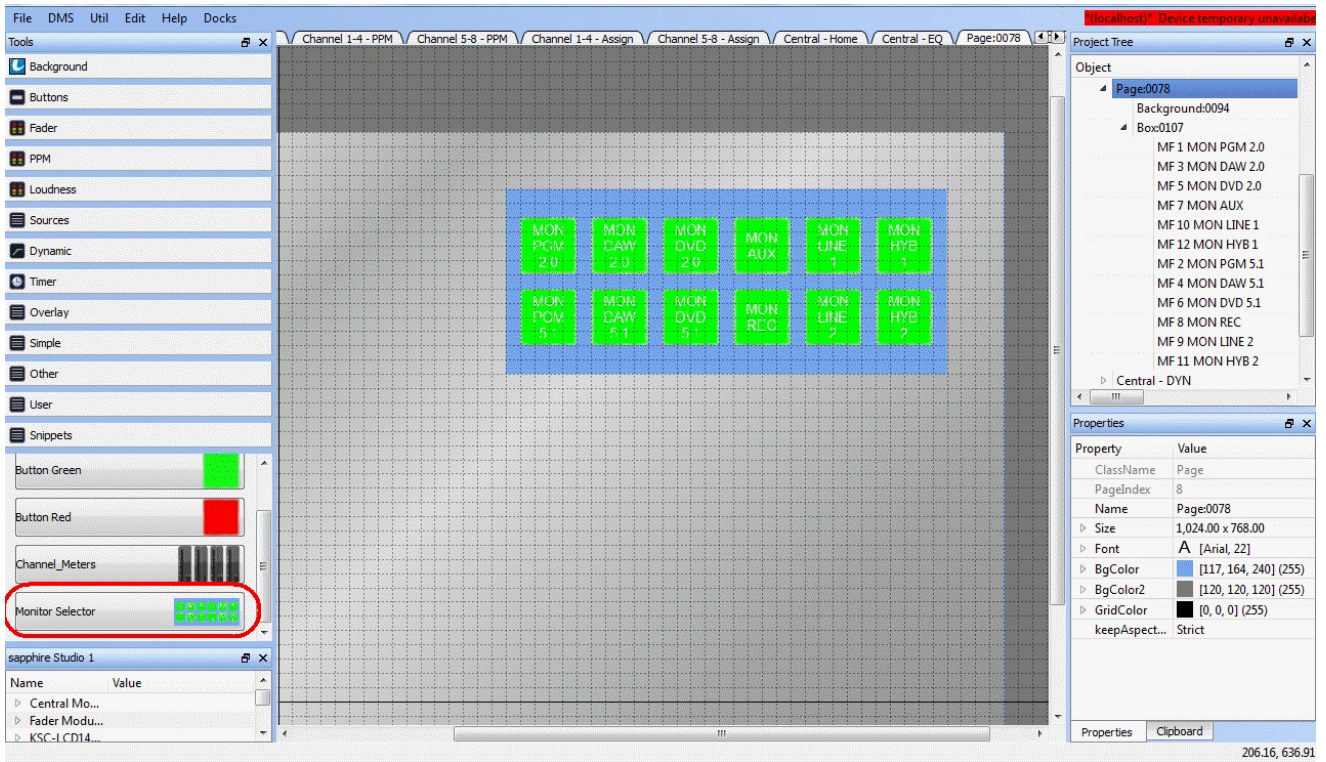
1. Select the elements you wish to store.
2. Press CTRL + ALT + SHIFT + S (or select **File** -> **Save as snippet** from the [main menus](#)):



3. Enter a name for the snippet file and select **Save**:



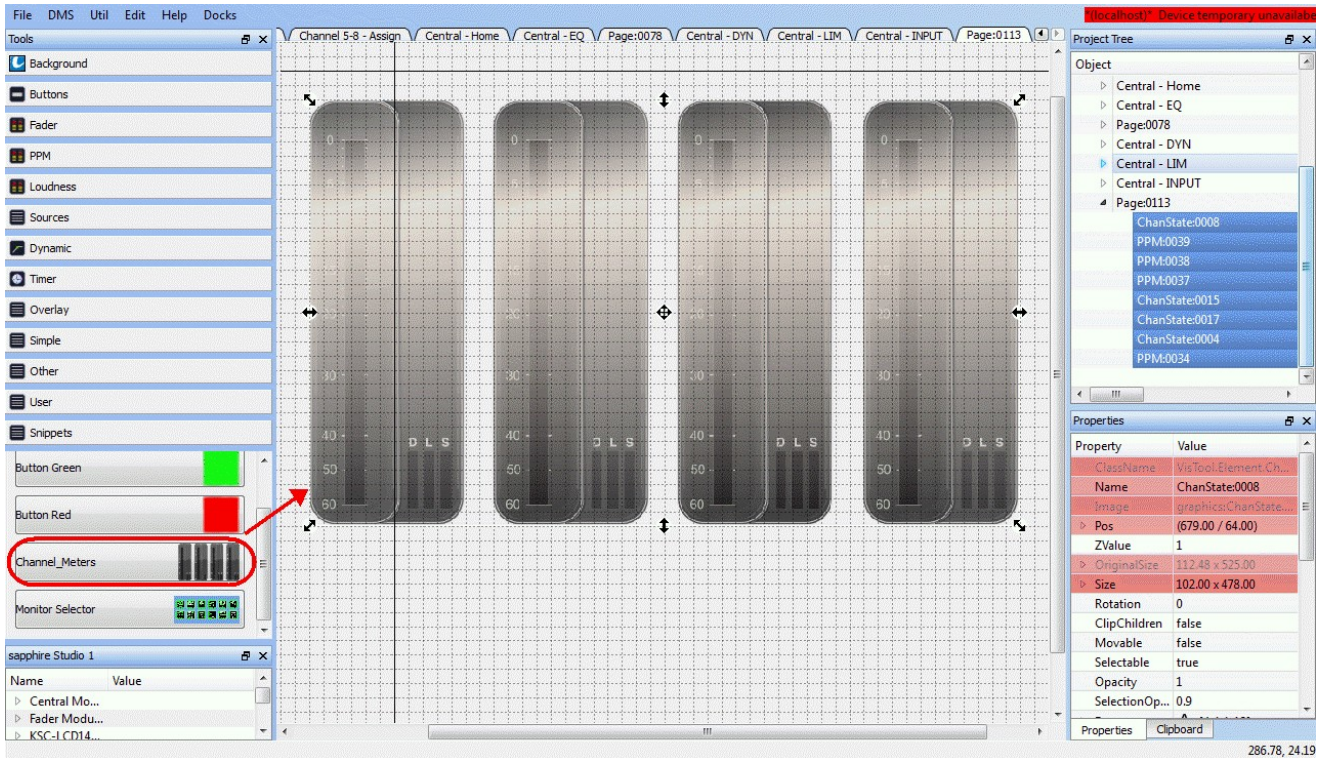
The snippet is stored on your computer (as an **.xml** file in the "C:\Program Data\DSA\VisToolMK2\Snippets" directory), and appears under the **Snippets** category in the **Tools** panel:





### 8.16.2 Loading a Snippet

To load a snippet, drag and drop the element from the **Tools** panel in the usual manner:



Loading a snippet pastes all the individual component elements, including their properties and functionality. You can now [edit](#) or [reposition](#) individual elements as required.

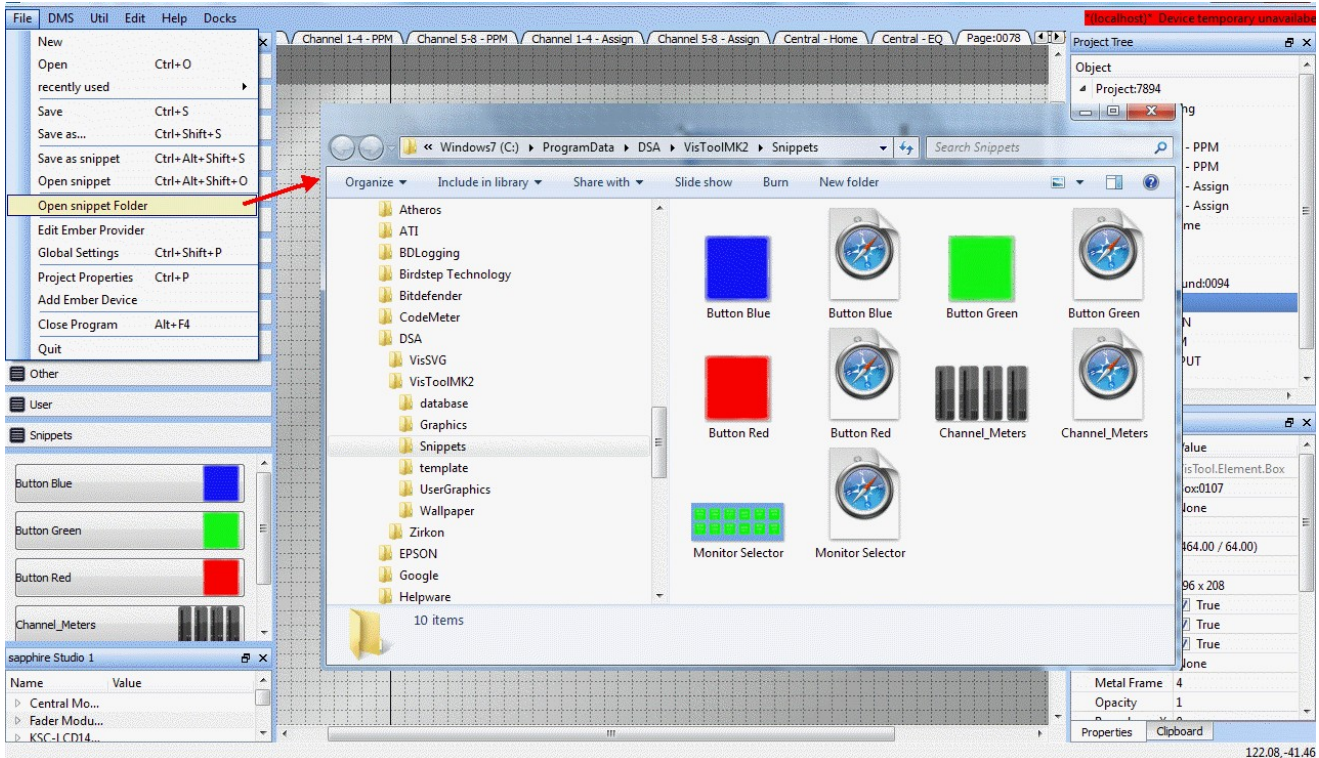
Note that an element's functionality will only load if the project supports the correct hardware. For example, a snippet containing elements linked to **sapphire** functions may not load within a **crystal** project.

### 8.16.3 Transferring Snippets

Snippets are stored on your computer in the "C:\Program Data\DSA\VisToolMK2\Snippets" directory. Note that two files exist for each snippet: **.xml** (data file) and **.png** (graphical preview).

You can transfer snippets between VisTool Editor computers as follows:

1. Open the snippets directory by selecting **File -> Open snippet folder** from the [main menus](#):



2. Use Windows Explorer to copy and paste the data (**.xml**) and preview (**.png**) files.
3. Restart VisTool Editor - any new snippet files are read from the "C:\Program Data\DSA\VisToolMK2\Snippets" directory and appear under the **Snippets** category in the **Tools** panel.

The selected snippet appears under the **Snippets** category in the **Tools** panel, and is stored with the project when you save.

# 9. ON-AIR Designer Functions

---

This chapter describes VisTool functions which are dependent on the ON-AIR Designer configuration.

Topics include:

- [Introduction](#)
- [MF Key Functions](#)
- [Fader Module Parameters](#)
- [Fader Strip Assign](#)
- [DSP Parameter Control](#)
- [Fader Strip ACCESS Keys](#)
- [Advanced VisTool Integration](#)
- [VisTool General Settings](#)

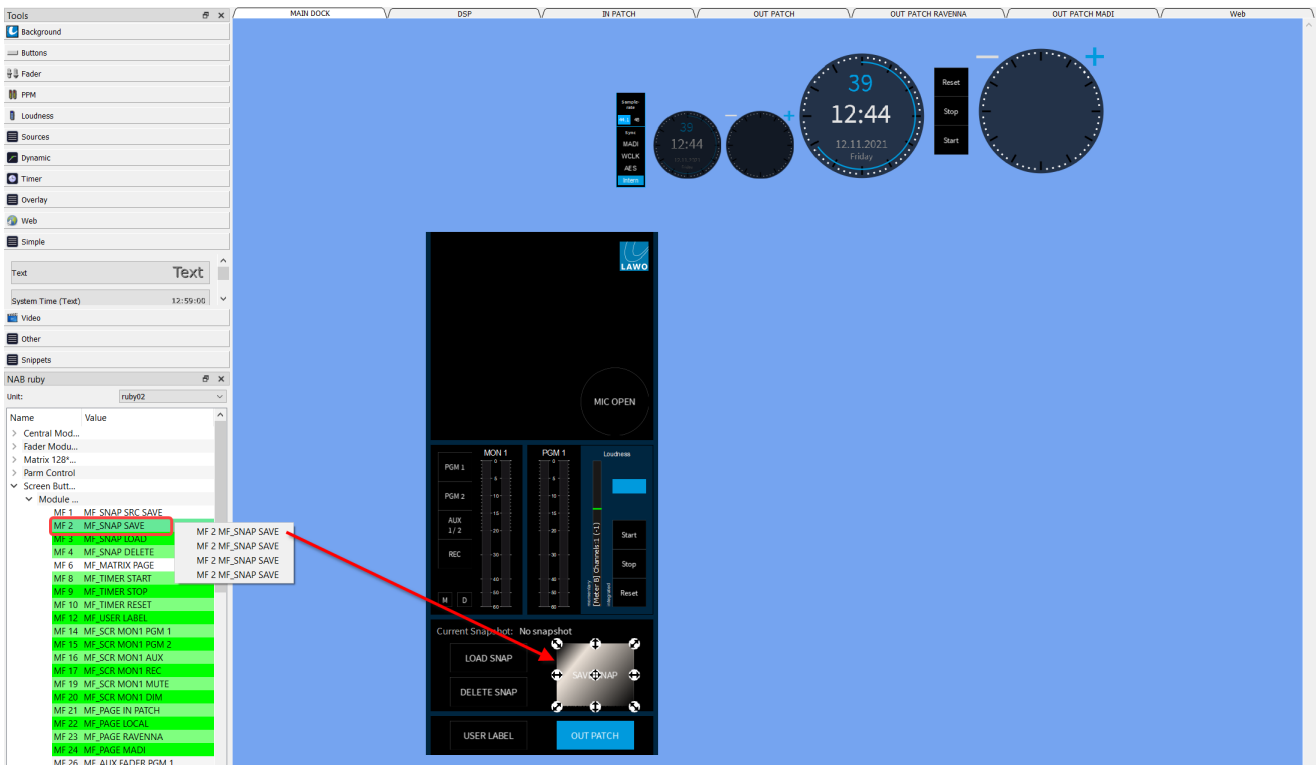
## 9.1 Introduction

To control (or respond to) functions within a remote DMS system, VisTool elements must be linked to parameters defined in the system's ON-AIR Designer configuration.

The ON-AIR Designer configuration is specified in the [Project properties](#) window. VisTool Editor reads the contents on start-up, and updates its available options accordingly. If you make a change to the ON-AIR Designer configuration while VisTool Editor is open, then either restart VisTool Editor or update the 'Project properties to force VisTool Editor to re-read the configuration data.

The functions defined in the ON-AIR Designer configuration can be interrogated by opening the system's Docking panel. In the example below, the ON-AIR Designer project is named "NAB ruby" and the selected Unit is "ruby02". Any entries highlighted in green are already used somewhere in the VisTool project. Right-click on a green entry to interrogate its uses. In the example below, the **MF\_SNAP\_SAVE** key is used 4 times. If you select a use (in the right-click list), the corresponding page opens and the object is selected.

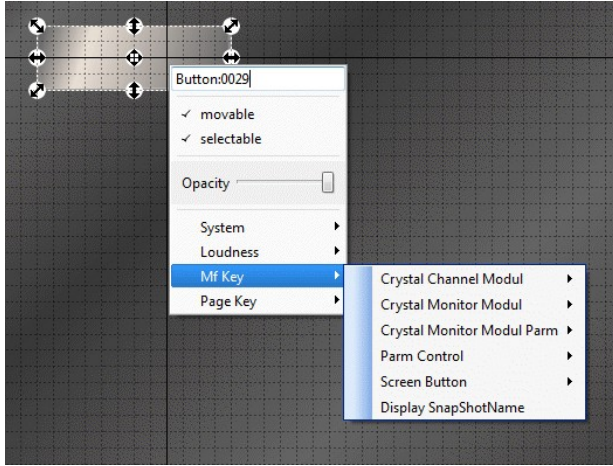
*VisTool Editor: Interrogate MF Key Uses*



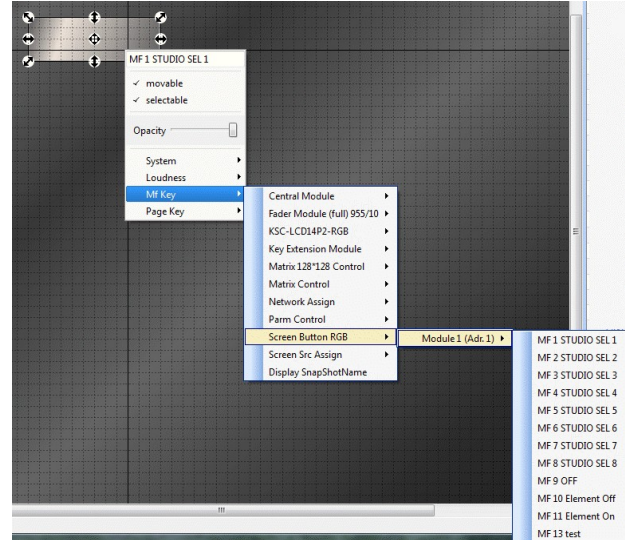
## 9.2 Assigning MF Key Functions

When assigning an MF Key to a Vistool element, the available options are divided into sub categories for the physical and screen modules. At the bottom of the list you will also find the **DisplaySnapshotName** function.

Example 1 (crystal)



Example 2 (sapphire)



### 9.2.1 Physical MF Keys

Use the module options (Central Module, Fader Module, etc.) to link the VisTool element to a physical MF Key on the console - the VisTool element will then mirror the operation of the MF Key.

The available MF Keys depend on the control surface and key panel configuration defined by the ON-AIR Designer (using the menus "Frames -> Surface" and "Frames -> Panels").

Note that to access MF Keys on Fader, Channel or Rotary Extension Modules, they must be named in the "MF Key" branch of the ON-AIR Designer's 'Tree Definition'.

The MF Key parameters are defined by the "MF Key" and "Source -> Keys" branches of the ON-AIR Designer's 'Tree Definition'.

### 9.2.2 Screen Modules

The screen modules allows you to create virtual MF Keys which can then be linked to VisTool elements.

Each module must be defined in the ON-AIR Designer (using the "Frames -> Screen" window).

The following modules can be used several times to create multiple instances:

- **Matrix 128 \* 128 Control** - an XY matrix panel of 128 x 128.
- **Matrix Control** - an XY matrix panel of 90 x 30 (switched in pages).
- **Screen Blank Panel** – a blank cover plate.
- **Screen Button** – 32 MF Key functions (defined by the "MF Key" branch of the ON-AIR Designer's 'Tree Definition').
- **Screen Button RGB** – 64 MF Key functions (as above) but with self-labeling, RGB buttons.

The following modules can be used only once. In each case, you must choose the correct panel slot (in the ON-AIR Designer "Frames -> Screen"):

- **Slot 11: Screen Matrix TB** - a talkback switching panel, with 32 buttons.
- **Slot 12: Screen Matrix Monitor** - a monitor matrix control panel, with 32 buttons.
- **Slot 13: Network Assign** - a network assign control panel, with 32 buttons.
- **Slot 14: Screen Src Assign** - a fader strip assign control panel, with 32 buttons.
- **Slot 15: Parm Control** - a DSP control panel, with 32 buttons.

Some examples of how to use the screen modules are included later in this chapter.

### 9.2.3 Display Snapshot Name

Use this option to display the current snapshot name in the VisTool element. The element will then display the last snapshot [loaded](#).

### 9.3 Fader Module Parameters

The "Vis Chan" branch of the ON-AIR Designer's 'Tree Definition' names the physical controls on Fader/Channel Modules so that they can be mapped to VisTool elements. This allows controls from the module such as MF Keys to be mirrored on the VisTool runtime GUI. You must enter a unique reference name for each control you wish to access, and repeat the procedure for each Fader/Channel Module. Then save and link the ON-AIR Designer project to VisTool Editor.

Within VisTool Editor, the named functions appear under the corresponding drop-down menus - for example, MF Keys appear under the **MF Key** branch when [defining](#) a VisTool Button or Box element.

The following parameters on the Fader/Channel Module can be accessed: MF Keys, signal present indicator, ACCESS keys, display name, user label text, analog input gain, channel DSP gain and back gain.

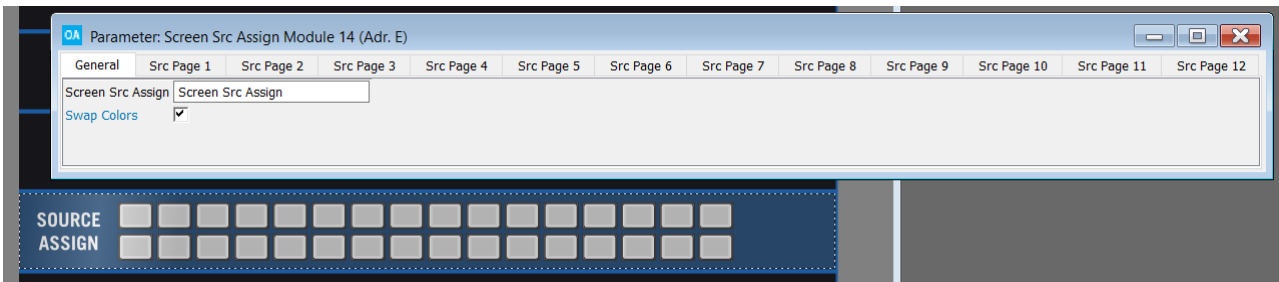
## 9.4 Fader Strip Assign

The buttons on a **Screen Source Assign** module can be used to perform fader strip assignments from a VisTool runtime GUI.

The assignment always works on the fader in access, and so you may wish to add some ACCESS keys to the VisTool page (as described [later](#)). The configuration requires two stages: first use the ON-AIR Designer to prepare the system configuration, and then use VisTool Editor to edit the VisTool project.

### ON-Air Designer Configuration

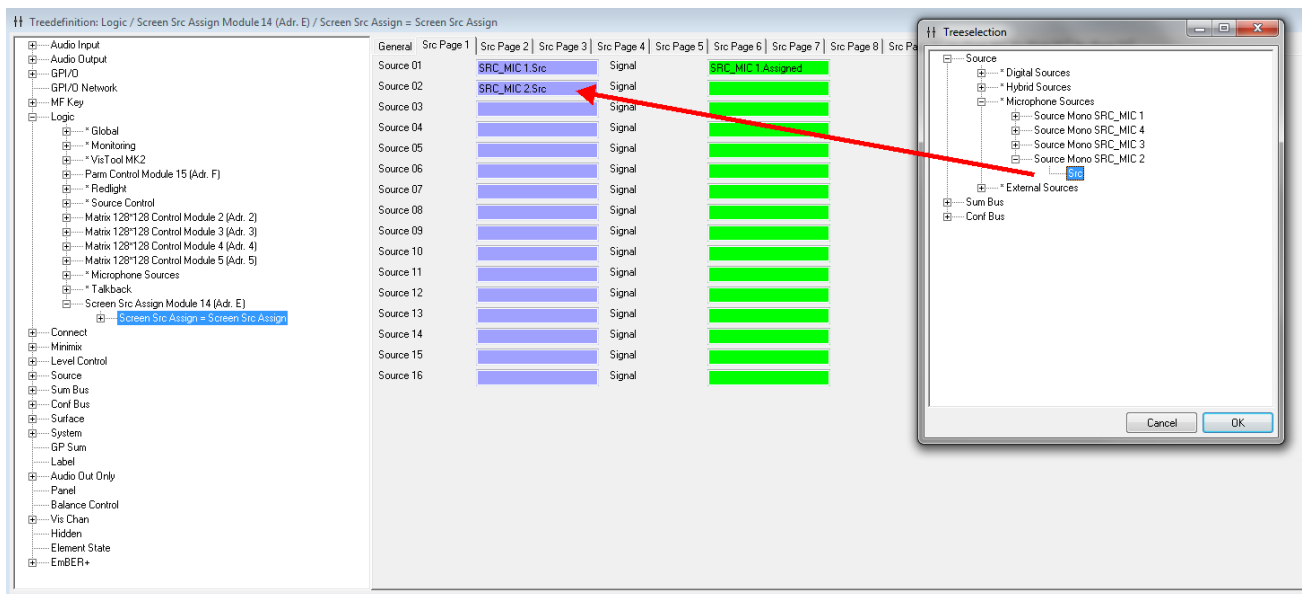
1. Add a **Screen Src Control** panel to slot 14 of the Screen configuration.
2. Name the panel from its **General** parameters tab:



If the **Swap Colors** option is enabled, then the on-screen "Assign Src" button in VisTool will use more colors to indicate the fader strip assign status. These are described [later](#).

3. Select the **Screen Src Control** module from the "Logic" branch of the 'Tree Definition' and assign the sources you wish to make available for VisTool assignment from the **Src Page** parameter tabs.

You can define up to 192 sources.





# 9. ON-AIR Designer Functions

Then assign the signal which will be used for the signalisation of the VisTool element - the **"Source.Assigned"** output is normally used:

The screenshot shows the 'Screen Src Assign = Screen Src Assign' window with a table of source assignments. A red arrow points from the 'SRC\_MIC 2.Assigned' cell in the 'Src Page 6' column to the 'SRC\_MIC 2' entry in the 'Treeselection' dialog box.

General	Src Page 1	Src Page 2	Src Page 3	Src Page 4	Src Page 5	Src Page 6	Src Page 7	Src Page 8	Src Page 9
Source 01		SRC_MIC 1.Src	Signal			SRC_MIC 1.Assigned			
Source 02		SRC_MIC 2.Src	Signal			SRC_MIC 2.Assigned			
Source 03			Signal						
Source 04			Signal						
Source 05			Signal						
Source 06			Signal						
Source 07			Signal						
Source 08			Signal						
Source 09			Signal						
Source 10			Signal						
Source 11			Signal						
Source 12			Signal						
Source 13			Signal						
Source 14			Signal						
Source 15			Signal						
Source 16			Signal						

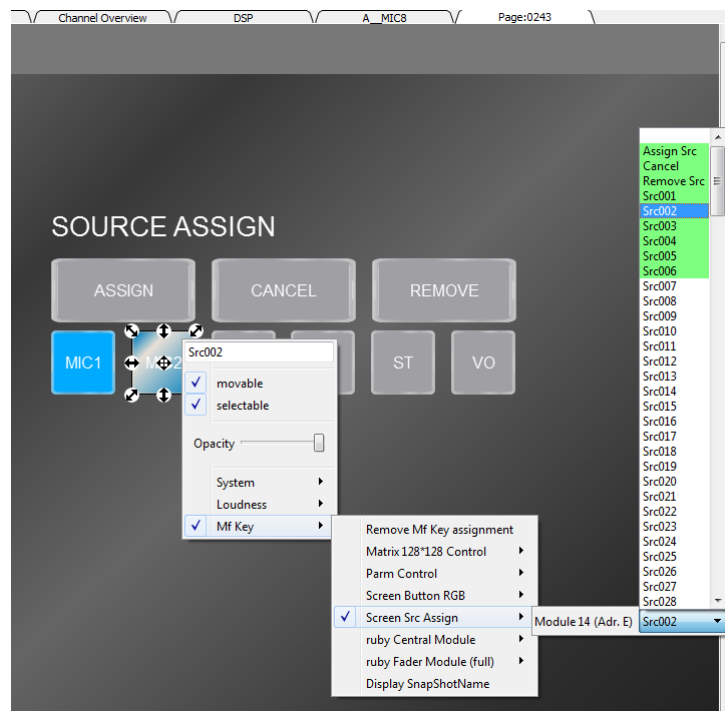
## VisTool Editor Configuration

1. Add some [Box](#) elements to a page.
2. Right-click on a button, and open up the **MF Key** -> **Screen Src Assign** branch. Note that any functions which have already been used are highlighted in green.

First, add the three buttons which will action the "fader assign" mode:

- **Assign Src** - mimics a double-press of the console's **ACCESS** key. This puts the current fader in access into "fader assign" mode. Once active, you can choose a source using a source select button (see below), press **Cancel** to cancel the operation, or press **Remove Src** to remove the current source.
- **Cancel** - cancels an active **Assign Src** operation.
- **Remove Src** - removes the source from the fader in access and leaves it unassigned.

Then, add the source select buttons (**Src001**, **Src002**, etc.). Each of these relates to the Screen sources defined earlier by the ON-AIR Designer - in our example, **Src002** corresponds to our **MIC\_2** source:



As you assign each button, it updates to show the assigned function. Edit the **User Text** property to define the name displayed on the VisTool screen.

3. Remember to [save](#) the project and switch to [Test mode](#) to test the fader strip assignments.

If the **Swap Colors** option is enabled in the ON-AIR Designer, then the on-screen "Assign Src" button shows four states:

- Red (P4) = fader result start (i.e. fader open).
- Yellow (P3) = configured signal active.
- Green (P2) = source assigned to fader.
- Black (P1) = true.

If the **Swap Colors** option is disabled, then only the last two states are shown:

- Green (P2) = source assigned to fader.
- Black (P1) = true.

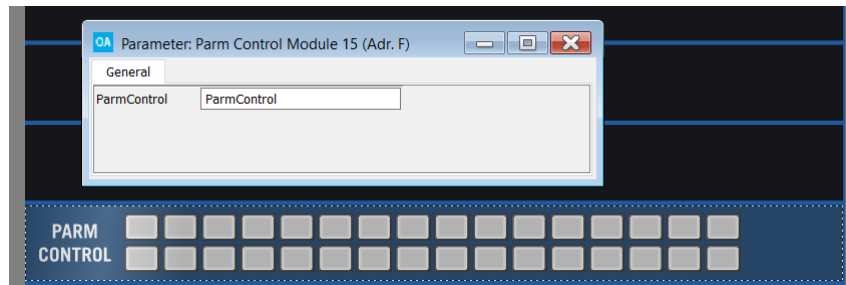
## 9.5 Source Parameter Control

The buttons on a **PARM Control** module can be used to adjust signal processing parameters from a VisTool runtime GUI.

The controls always work on the fader in access, and so you may wish to add some **ACCESS** keys to the VisTool page (as described [later](#)). PARM MF Keys are usually combined with Access Source Information to provide feedback on the source and its parameter values. The configuration requires two stages: first use the ON-AIR Designer to prepare the system configuration, and then use VisTool Editor to edit the VisTool project.

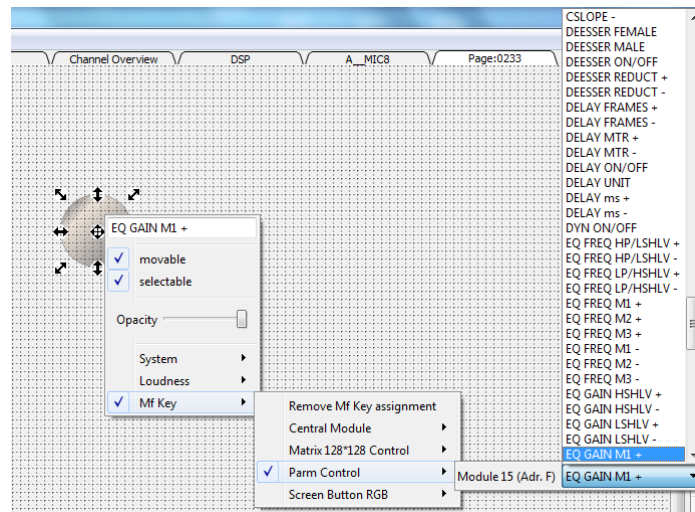
### ON-Air Designer Configuration

1. Add a **PARM Control** panel to slot 15 of the Screen configuration.
2. Name the panel from its **General** tab:



### VisTool Editor Configuration

1. Add some [Box](#) elements to a page.
2. Right-click on a button, and open up the **MF Key -> Parm Control** branch. Select the parameter you wish to assign to the VisTool button - there are many options including Aux on/off and +/- for most DSP parameters:



As you assign each button, it updates to show the name of the function.

The **Access 01** to **Access 40** options are for backwards compatibility only. To assign the Access key for a fader strip to a VisTool button use the method described below.

3. Remember to [save](#) the project and switch to [Test mode](#) to test your parameter control.

## 9.6 Fader Strip ACCESS Keys

By adding fader strip ACCESS keys to the VisTool GUI, the end-user will be able to change the source in access. The configuration requires two stages: first use the ON-AIR Designer to prepare the system configuration, and then use VisTool Editor to edit the VisTool project.

### ON-Air Designer Configuration

1. Name the relevant Access key(s) in the "Vis Chan" branch of the 'Tree Definition'.

You will need to name the **Access** key for each of the 4 fader strips on each Fader Module, and the **LayerSelect** key if you wish to access sources on Layer 2.

### VisTool Editor Configuration

1. Add some [Box](#) elements to a page.
2. Right-click on the element and select the **MF Key** -> **Fader Module** branch to assign the named keys.

## 9.7 Advanced VisTool Integration

### "Logic -> VispageSwitch"

A **VispageSwitch** element can be used to configure MF Keys, or other control signals, to action VisTool logical functions. For example, to create page switching buttons, snapshot save/load/delete and User Access Rights.

VispageSwitch elements are added to the "Logic" branch of the 'Tree Definition'. You should define an element for each VisTool screen, and set the **Station ID** to match the **Station ID** defined in VisTool's [Session Management](#) window. Any VisTool screen set to a matching **Station ID** will then respond to the VispageSwitch functions.

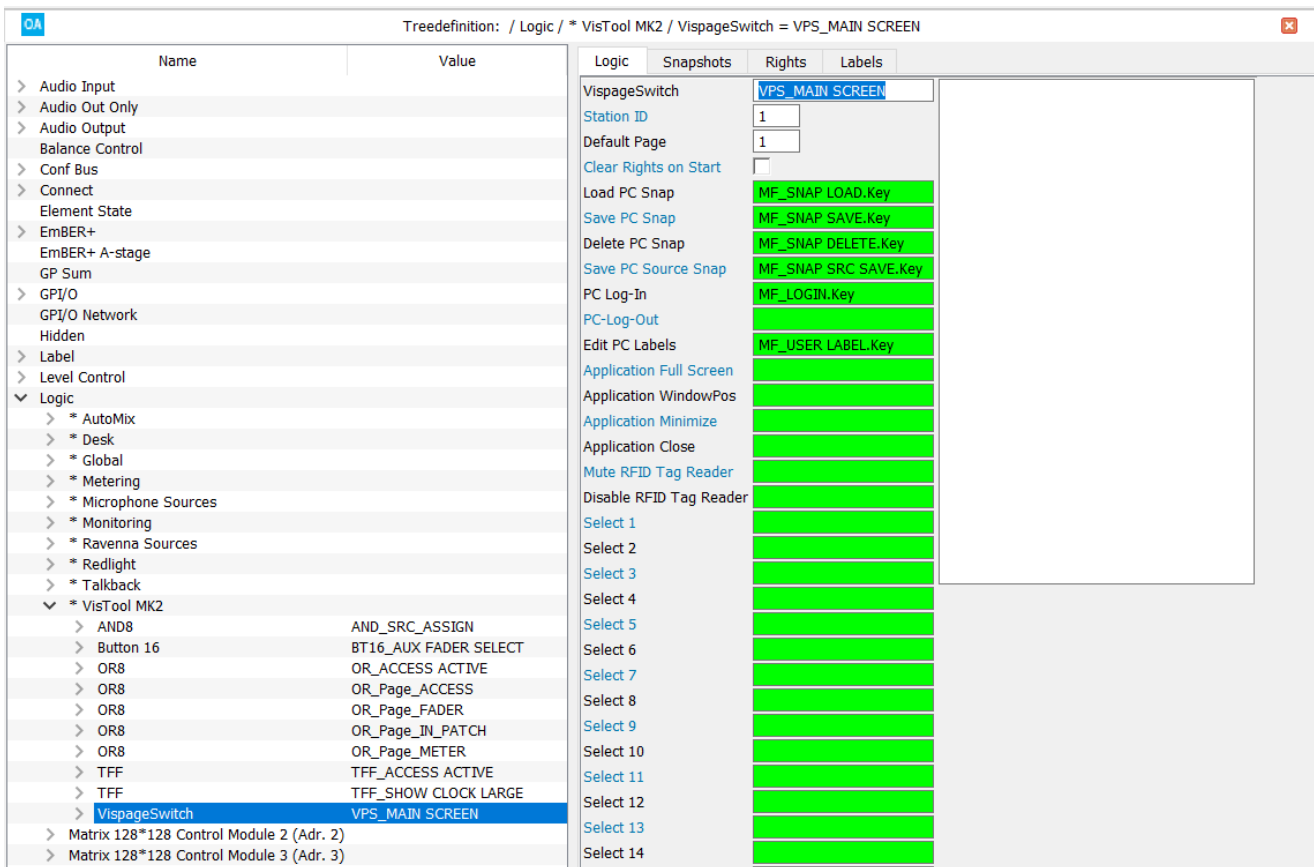
To enable VisTool page switching and snapshot functions you must tick the **EnablePageKey** and **EnableLoadSave** checkboxes within the VisTool Editor "[Global Settings](#)" window.

VisTool page switching and snapshot functions can also be configured using the "[System -> Definition -> Parameter = Screen](#)" and "System -> Definition -> Parameter = Snapshot" elements. These methods switch all VisTool screens regardless of their **Station ID**.

The **Nova29** does not support snapshots.

Each VispageSwitch element supports four sets of parameters: Logic, Snapshots, Rights and Labels.

### 9.7.1 VispageSwitch -> Logic Parameters



The screenshot shows the 'Logic' tab for a VispageSwitch element. The tree view on the left shows the hierarchy: Logic > VisTool MK2 > VispageSwitch. The central panel lists various logic actions with their corresponding MF keys, such as 'Load PC Snap' (MF\_SNAP LOAD.Key) and 'Save PC Snap' (MF\_SNAP SAVE.Key). The right-hand panel shows a large empty area, likely for visual feedback or a preview of the actions.

<b>Vispage Switch</b>	Reference name for the element.
<b>Station ID</b>	Assigns an ID number to the element. Any VisTool screen set to a matching <b>Station ID</b> (in <a href="#">VisTool Sessions</a> ) will respond to the VispageSwitch functions.

<b>Access Group</b>	This option assigns an Access Group to the element. Access Groups are supported by Power Core Max to link the control surface ACCESS keys to different VisTool instances. Any Fader Modules set to a matching Access Group will respond to the corresponding VispageSwitch functions.
<b>Default Page</b>	This page appears on the VisTool screen following a cold start of the remote system. You should enter the number which corresponds to the <a href="#">page index</a> within the VisTool Editor configuration.
<b>Clear Rights on Start</b>	If checked, defined user rights are reset after a warm start (e.g. a power cycle). If not checked, user rights will remain intact. Note that you must be using VisTool MK2 to support the <a href="#">user rights</a> system.
<b>Load PC Snap</b>	When this control signal is active, a window opens on the VisTool screen to load a snapshot.
<b>Save PC Snap</b>	As above, but to save a snapshot.
<b>Delete PC Snap</b>	As above, but to delete a snapshot.
<b>Save PC Source Snap</b>	As above, but to save a source snapshot.
<b>PC Log-In</b>	When this control signal is active, the user <a href="#">log in</a> window opens on the VisTool screen. You must activate this function to provide access to the user rights supported by VisTool MK2. The function should only be active on a single VisTool screen - i.e. for one VispageSwitch element mapped to one Station Id.
<b>PC Log-Out</b>	As above, but the control signal logs out the current user.
<b>Edit PC Labels</b>	When this control signal is active, the <a href="#">user label</a> edit window opens on the VisTool screen.
<b>Application Full Screen</b>	Sets the application to full screen view.
<b>Application WindowPos</b>	Toggles the application between full screen and minimized.
<b>Application Minimize</b>	Minimizes the application.
<b>Application Close</b>	Closes the application.
<b>Mute RFID Tag Reader</b>	Mutes the RFID Tag Reader. The RFID Tag Reader is an optional Add On for VisTool MK2 which permits user login via an RFID tag. This control signal can be used to mute the reader.
<b>Disable RFID Tag Reader</b>	As above, but the control signal disables the RFID Tag Reader.
<b>Select 1 to 32</b>	When the <b>Select 1</b> control signal is active, the VisTool screen is switched to page 1, where page 1 is the <a href="#">page index</a> within the VisTool Editor configuration. When the <b>Select 2</b> control signal is active, the VisTool screen is switched to page 2. And so on up to page 32.

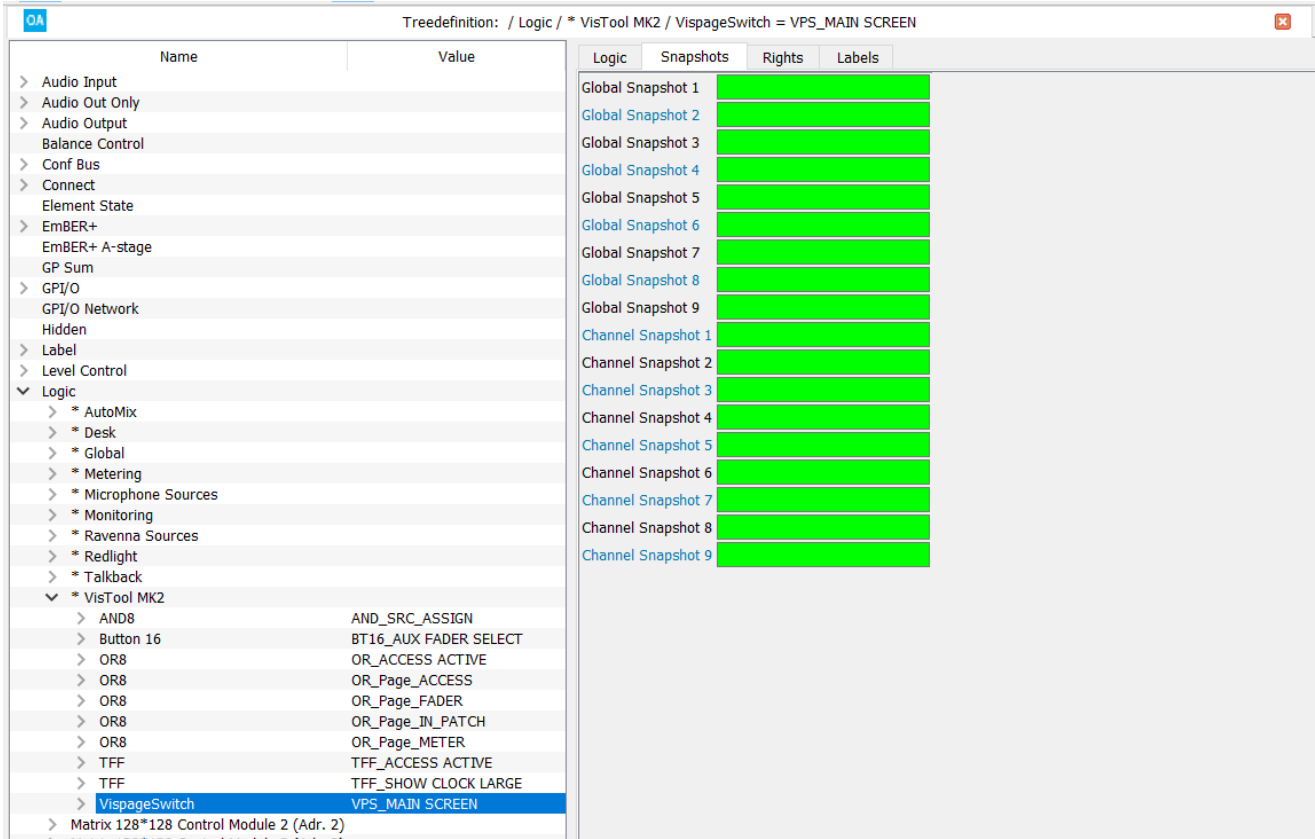
The following control outputs appear under "Logic -> <GroupName> -> VispageSwitch" in the 'Tree Selection' window:

<b>Selected 1</b>	True when Page 1 is selected.
<b>Selected 2</b>	True when Page 2 is selected.
<b>etc.</b>	Etc. up to 32.

Use the **Selected 1** to **32** outputs as feedback to the control signal (for example, to light the MF Key which switched the page).

## 9.7.2 VispageSwitch -> Snapshot Parameters

These parameters assign control signals to the nine Global/Channel snapshot buttons:



<b>Global Snapshot 1 to 9</b>	When this control signal is active, the corresponding global snapshot is loaded. The 9 Global Snapshot buttons can load any Full snapshot. The assignment of snapshots to buttons is made in the <a href="#">Edit Snaps</a> window.
<b>Channel Snapshot 1 to 9</b>	As above, but Channel snapshot buttons can load any Src snapshot with a matching Class field.

The control outputs appear under “Logic -> <GroupName> -> VispageSwitch” in the 'Tree Selection' window:

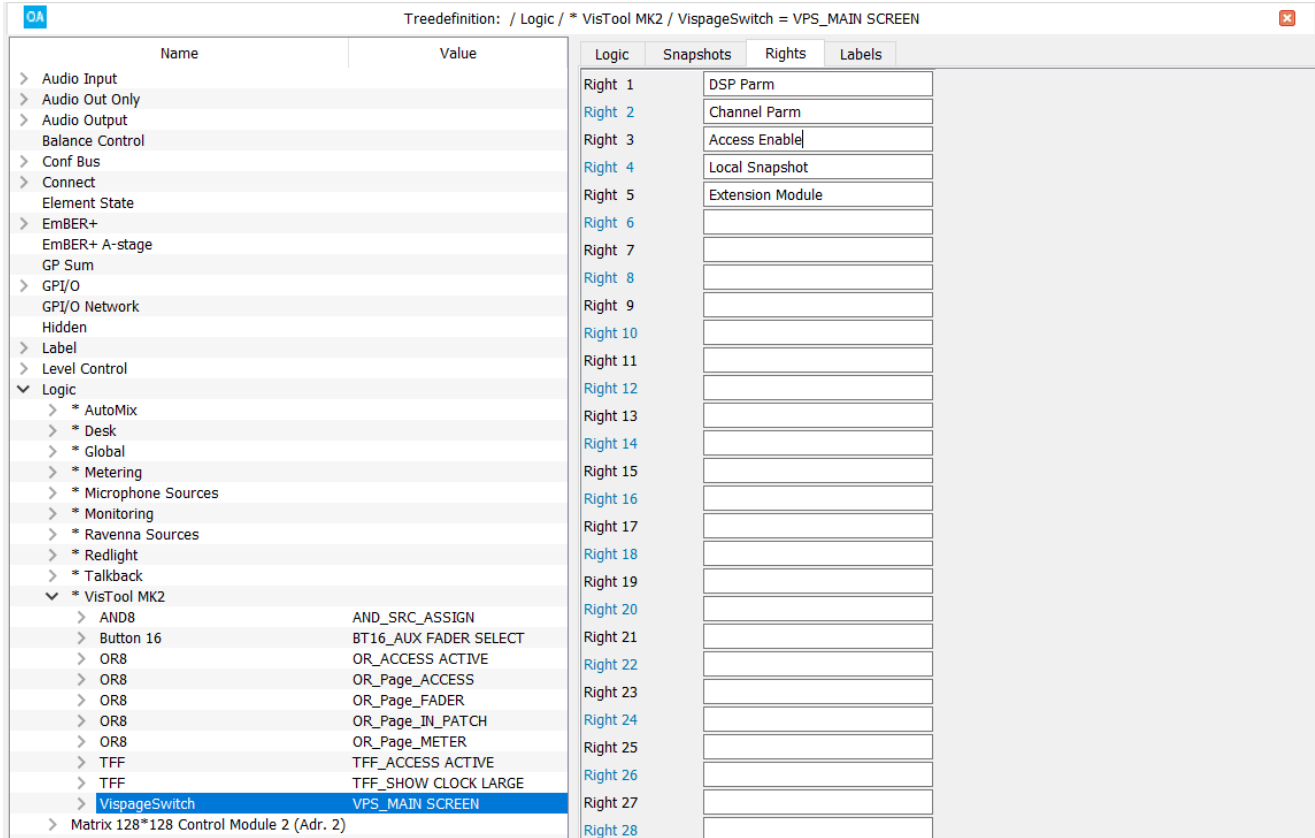
<b>GlbSnap n defined</b>	True when global Snapshot n is defined within the <a href="#">user rights</a> system supported by VisTool MK2.
<b>ChanSnap n defined</b>	True when channel Snapshot n is defined within the <a href="#">user rights</a> system supported by VisTool MK2.

### 9.7.3 VispageSwitch -> Rights Parameters

These parameters name each of the [user access rights](#) supported by VisTool MK2.

You should *only* name Rights for the same **VispageSwitch** element which opens the user log-in window (via [PC Log-In](#)). Otherwise you will not be able to access the rights from the user management Admin pages!

The name in the 'Tree Definition' is a reference name only; up to 32 Rights may be defined:



Once named, the control outputs appear under “Logic -> <GroupName> -> Right” in the 'Tree Selection' window. For each right, there are two states: granted and denied:

<b>Right 1 Granted</b>	True when user permission is granted (defined by the <a href="#">User Group Rights</a> administrator).
<b>Right 1 Denied</b>	True when user permission is denied.
<b>etc.</b>	etc. up to 32.

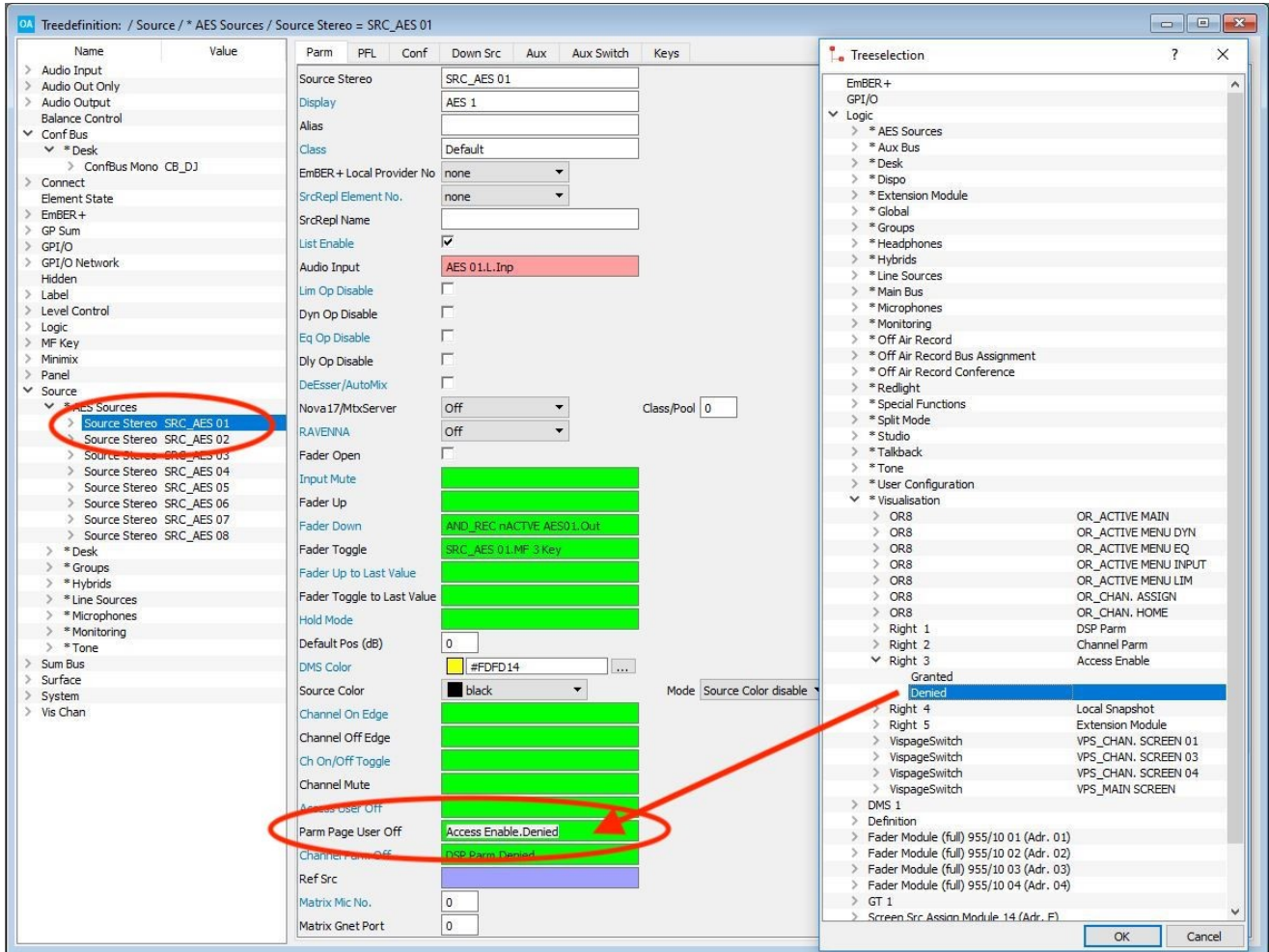


## Example of "Rights" Configuration

The naming of the user "Right" simply makes it appear in the [User Group Rights](#) Admin pages. To define what each "Right" does, you must configure the appropriate 'Tree Definition' functions.

In our example, **Right 3** (named **Access Enable**) allows permitted users to select the **ACCESS** button. If users are denied access to this right, then they cannot select **ACCESS** and the source cannot be assigned to the center section.

To configure this example, assign the **Denied** control output for user **Right 3** to the **Access User Off** function. If you wish to apply this to every source and sum bus, then repeat for every source and sum in the configuration:



### 9.7.4 VispageSwitch -> Labels Parameters

These parameters can be used to label the 32 user Rights described [earlier](#).

Tree definition: / Logic / \* VisTool MK2 / VispageSwitch = VPS\_MAIN SCREEN

Name	Value	Logic	Snapshots	Rights	Labels
> Audio Input		Label 1			
> Audio Out Only		Label 2			
> Audio Output		Label 3			
> Balance Control		Label 4			
> Conf Bus		Label 5			
> Connect		Label 6			
> Element State		Label 7			
> EmBER+		Label 8			
EmBER+ A-stage		Label 9			
GP Sum		Label 10			
> GPI/O		Label 11			
> GPI/O Network		Label 12			
> Hidden		Label 13			
> Label		Label 14			
> Level Control		Label 15			
▼ Logic		Label 16			
> * AutoMix		Label 17			
> * Desk		Label 18			
> * Global		Label 19			
> * Metering		Label 20			
> * Microphone Sources		Label 21			
> * Monitoring		Label 22			
> * Ravenna Sources		Label 23			
> * Redlight		Label 24			
> * Talkback		Label 25			
▼ * VisTool MK2		Label 26			
> AND8	AND_SRC_ASSIGN	Label 27			
> Button 16	BT16_AUX FADER SELECT	Label 28			
> OR8	OR_ACCESS ACTIVE	Label 29			
> OR8	OR_Page_ACCESS	Label 30			
> OR8	OR_Page_FADER	Label 31			
> OR8	OR_Page_IN_PATCH	Label 32			
> OR8	OR_Page_METER				
> TFF	TFF_ACCESS ACTIVE				
> TFF	TFF_SHOW CLOCK LARGE				
> VispageSwitch	VPS_MAIN SCREEN				
> Matrix 128*128 Control Module 2 (Adr. 2)					
> Matrix 128*128 Control Module 3 (Adr. 3)					
> Matrix 128*128 Control Module 4 (Adr. 4)					
> Matrix 128*128 Control Module 5 (Adr. 5)					
> Parm Control Module 15 (Adr. F)					
> Screen Src Assign Module 14 (Adr. E)					
> MF Key					

## 9.7.5 Configuring Automatic Page Switching

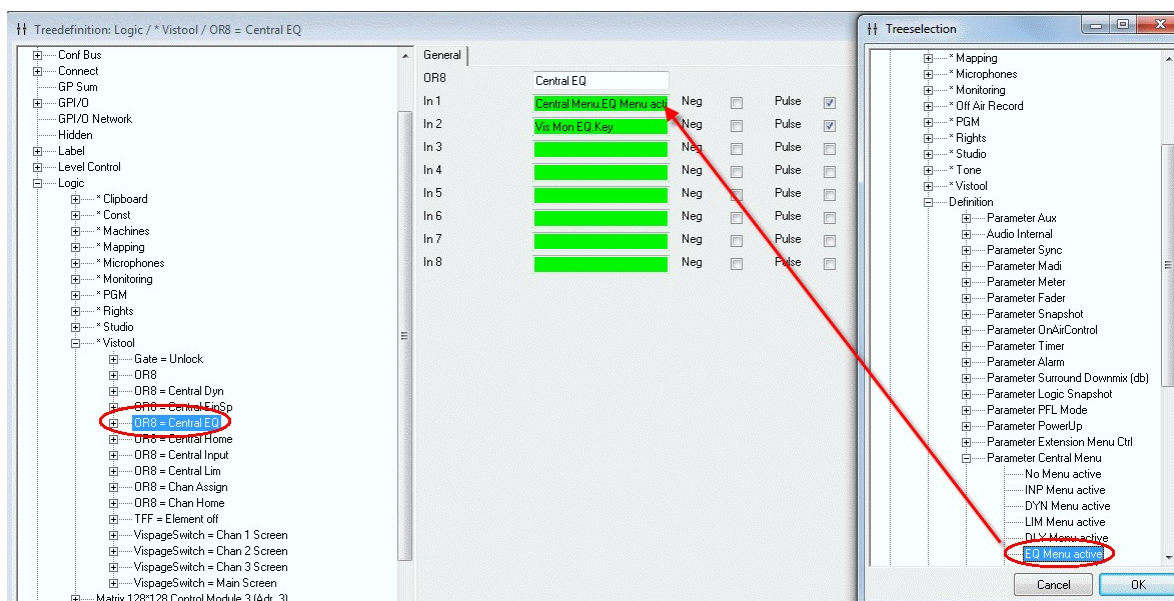
Using a [VispageSwitch](#) element, you can configure automatic VisTool page switching in response to control signals received from the remote system. For example, to show parameters for the source in ACCESS in response to the function buttons on a console surface (INP, EQ, DYN, etc).

There are numerous applications, but they all are programmed in a similar manner:

- First, using VisTool Editor, design the page or pages which you want to appear on the VisTool screen(s). For our example, design a page to show input parameters, a page to show EQ parameters, and so on.
- Then, using ON-AIR Designer, configure a "Logic -> VispageSwitch" element to handle the page switching. Each element can switch any VisTool page to any screen with a matching **Station Id**. The pages are referenced by their index number (from 1 to 32) defined in VisTool Editor.
- Finally, use VisTool's "Session Management" window to assign the **Station Id** to the correct VisTool screen.

Below is an example of the ON-AIR Designer configuration steps. By using a "Logic -> OR8" element, pages may be switched either automatically, in response to the function button mode, or manually, when the operator presses an MF Key:

1. Name the MF Key you wish to use for the manual operation.
2. Create and configure a Logic -> OR8 element to "OR" the control output from the function button (e.g. EQ) and the operator's MF Key:

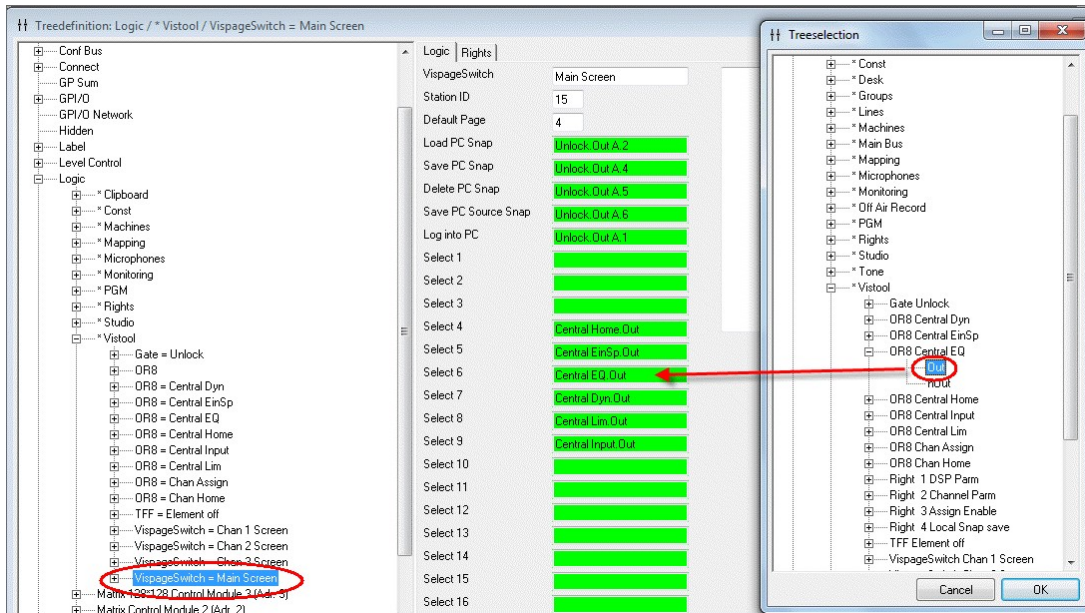


3. Create a [Logic -> VispageSwitch](#) element for the VisTool screen and set the **Station Id**. This should be a unique number.

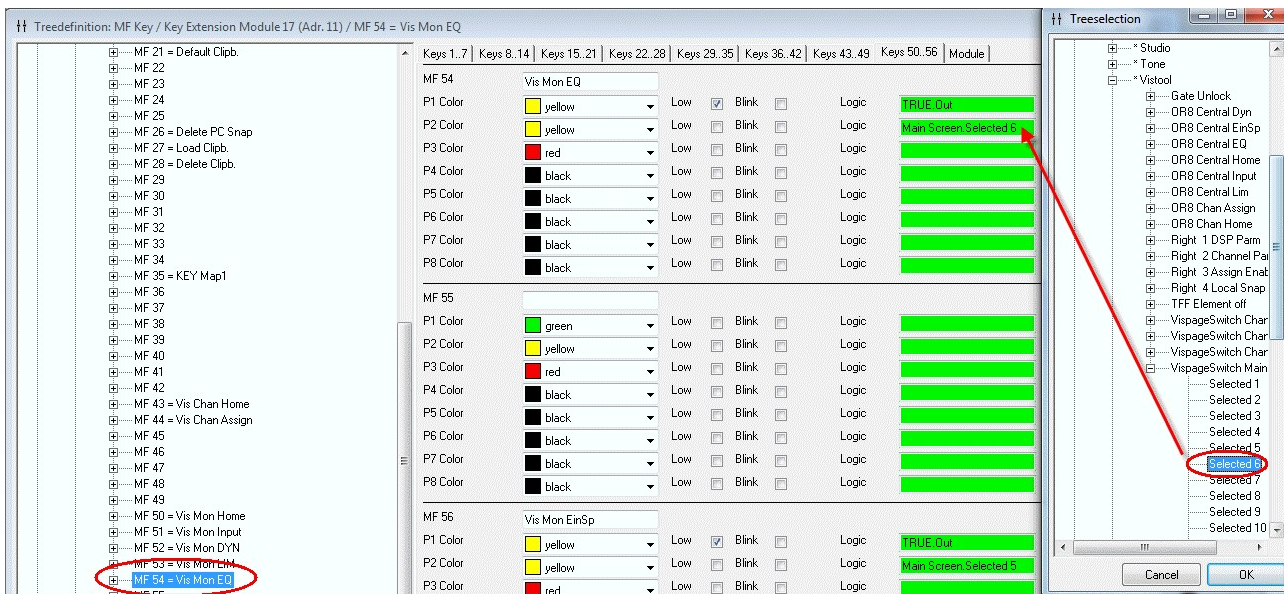
You will need to enter the **Station Id** into VisTool's [Session Management](#) window in order to define the VisTool screen.

**4.** Assign the output of the OR8 element to the correct page **Select** field.

**Select 1** corresponds to page index 1, **Select 2** to page index 2, and so on. Therefore, look at your **VisTool Editor** configuration and take a note of the [page index](#) of your EQ page. In our example, it is page index 6, and so we have assigned the control signal to page **Select 6**:



**5.** Return to the MF Key element, and assign the **Selected 6** control output to a lamp color - this will ensure that the MF Key lights when the EQ page is selected:



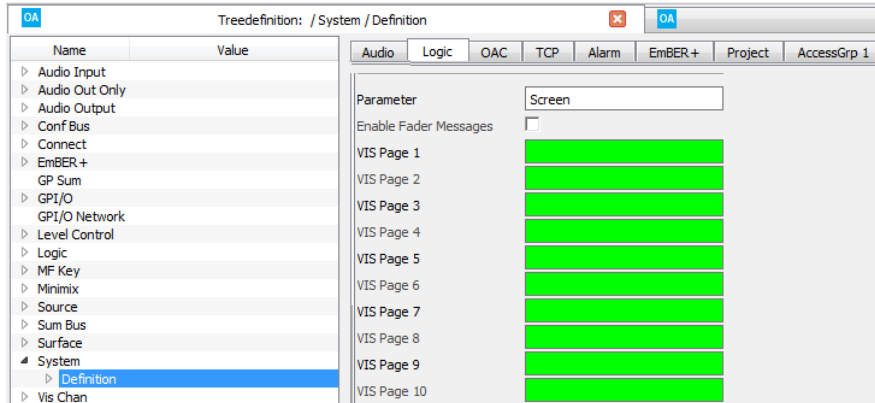
**6.** Repeat for each VisTool page you wish to switch - e.g. input parameters, dynamics parameters, pan parameters.

## 9.8 VisTool General Settings

"System -> Definition -> Parameter = Screen"

This branch of the 'Tree Definition' defines general settings for VisTool functions. The options apply globally to all VisTool instances.

Start by selecting "**System -> Definition**" in 'Tree Definition' and the **Logic** tab. Scroll down to access the "**Parameter = Screen**" options:



<b>Parameter</b>	Screen – reference name for the element.
<b>Enable Fader Messages</b>	When ticked, messages concerning fader levels are exchanged via DMS. When unticked, fader messages via DMS are disabled. The option must be ticked in order to exchange fader values with VisTool (e.g. to/from <a href="#">VisTool Fader</a> elements). When enabled, there will be an increase in the amount of network traffic. If fader messages are not required (to/from other DMS devices), then the option should be disabled to reduce the volume of network traffic.
<b>VisPage 1 to 10</b>	Available for <b>crystal</b> , <b>sapphire</b> and <b>sapphire compact</b> . Assigns a control signal (for example, an MF Key) to switch VisTool to a particular page. You can configure switching for the first 10 VisTool pages – <a href="#">page index</a> 1 to 10. Pages are switched across all VisTool instances. Thus, if you are connecting more than one VisTool and wish to configure independent page switching, please use the " <a href="#">Logic -&gt; VispageSwitch</a> " element instead.
<b>ASSIGN State Key Pressed Transition</b>	Available for <b>ruby</b> . Defines whether the console ASSIGN state triggers ACCESS functions in VisTool. There are two possible states: ASSIGN triggers ACCESS or ASSIGN triggers the normal state.

## 10. Using Ember+

---

This chapter covers VisTool's Ember+ implementation.

Topics include:

- [Introduction](#)
- [The Ember+ Devices List](#)
- [Adding an Ember+ Provider to the VisTool Project](#)
- [The Provider Tree](#)
- [Assigning Ember+ Parameters to VisTool Elements](#)
- [Working with an Offline Ember+ Provider](#)
- [Using the Ember Management Port](#)

To support Ember+ control you will need a **VisTool Unlimited** license plus the **Ember+/Logic AddOn**.

## 10.1 Introduction

VisTool MK2 is an Ember+ consumer which can control, or respond to, parameters published by an Ember+ providing device.

Ember+ is a non-proprietary TCP/IP protocol supported by a wide range of Lawo products.

For a VisTool element to control functions within the networked system, parameters must first be published by the Ember+ providing device. An Ember+ Provider Tree, listing all available parameters, is included in the User Manual for your console/router/application.

During runtime, VisTool can control multiple Ember+ devices. These are specified by the VisTool project (via the [Ember Devices](#) list).

### Preparation & Configuration

VisTool Editor requires a valid Ember+ connection to detect the parameters published by the providing device. This means that, in the first instance, you must connect both the VisTool Editor PC and the Ember+ provider to the Lawo system network.

You will need to know the hostname or TCP/IP Address and Port Number of each Ember+ provider on the network.

Then open VisTool Editor and perform the following configuration steps:

1. Specify the Ember+ providers in the [Ember+ Devices list](#).
2. [Add](#) the Ember+ provider(s) to the current VisTool project.
3. [Assign](#) parameters to your VisTool elements (using the Ember+ provider tree).

You can save an [offline copy](#) of the Provider Tree. This allows VisTool Editor to still access the provider's parameters even if the device goes offline.

You can use the [Logic Engine](#) to program logical conditions for Ember+ elements.

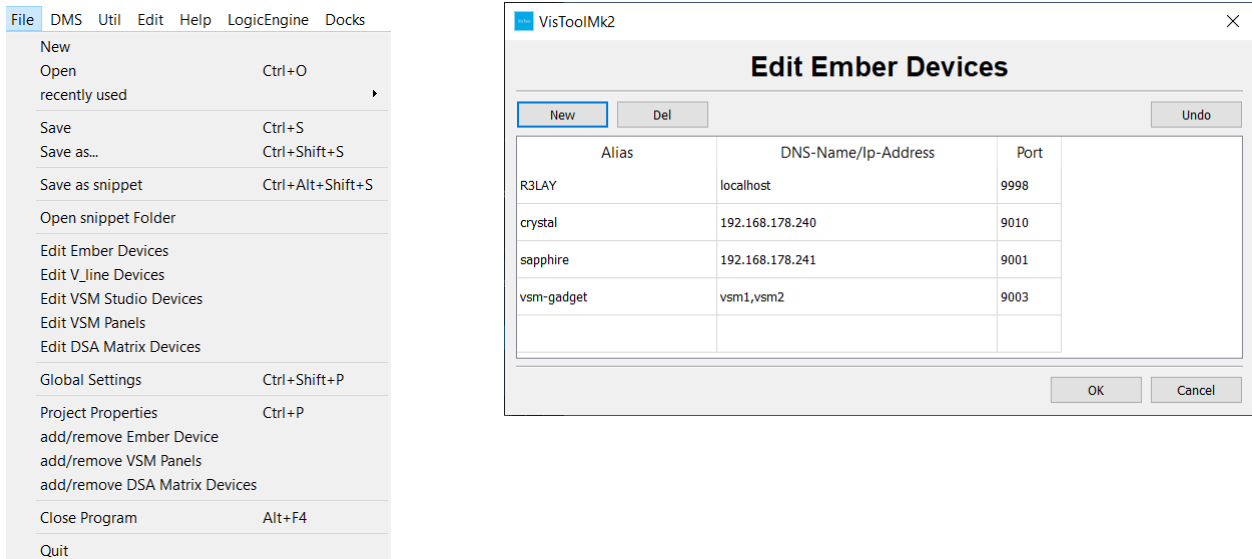
The rest of this chapter describes the steps above in more detail.

## 10.2 The Ember+ Devices List

This list defines the TCP/IP settings of each Ember+ provider on the network.

Note that you can define more devices than you wish to use in the current project - for example, to define all Ember+ providers on the network. A subset can then be added to the project later.

1. Select **File -> Edit Ember Devices** to open the list:



2. Select **New** to add a new entry.
3. Click in each of the fields to enter an **Alias** name, the **DNS-Name** or **Ip-Address**, and **Port** number of each Ember+ provider - take care to press Enter to confirm each field.
4. Use **Del** to delete an existing line. **Undo** will undo all changes. Select **OK** to save all changes and exit.

In our example, the providers include a Lawo **R3LAY**, **crystal**, **sapphire** and **vsm-gadget**.

The Ember+ devices list is stored in the **Emberhosts.ini** file. This may be shared between VisTool instances. See [Using a Shared File path for ".ini" Files](#). You can also deactivate a host by [editing](#) the file.

### Configuring a Redundant Connection

It is possible to configure redundant Ember+ connections so that if, the active connection fails, VisTool will automatically switch to the backup device. Only one of the configured devices is ever active, and there is no comparison of parameters or device mirroring. Thus, a prerequisite is that the Ember+ Provider trees must match.

1. In the **Edit Ember Devices** list, enter the two IP addresses (or DNS names) separated by a comma.
2. Select **OK** to save the changes and exit.

A redundant connection can also be configured by editing the **Emberhosts.ini** file. For the vsm-gadget shown above, the file looks like this: `vsm-gadget="vsm1, vsm2 9003"`

3. Once configured, the system behaves as follows.

On start-up, VisTool attempts to connect to the first address. If this is unsuccessful, then VisTool tries the second address. This step alternates until communication is established.

VisTool then tries to access the Ember+ provider's root node:

- If there is no traffic, then a ping-request is sent. If there is no ping-return or no traffic (for about 1 minute), VisTool considers the client dead and disconnects. The software then starts over with a new connection attempt.
- If the root node is available, then the software looks up all of the Ember+ parameters defined in the VisTool project and reads the data from the new device. If there is nothing to look up, then only the root node is read.

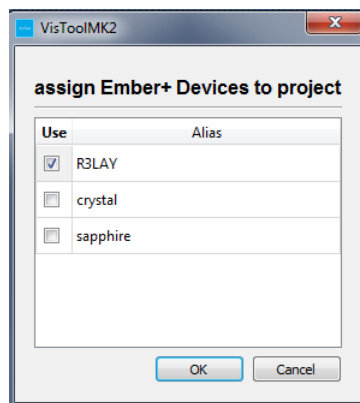
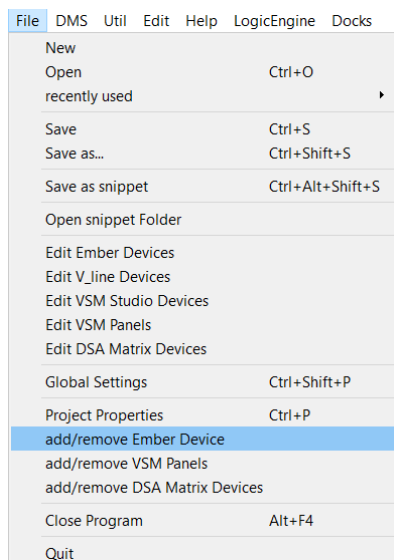


## 10.3 Adding an Ember+ Provider to the VisTool Project

The next step defines which Ember+ providers are available in the current project.

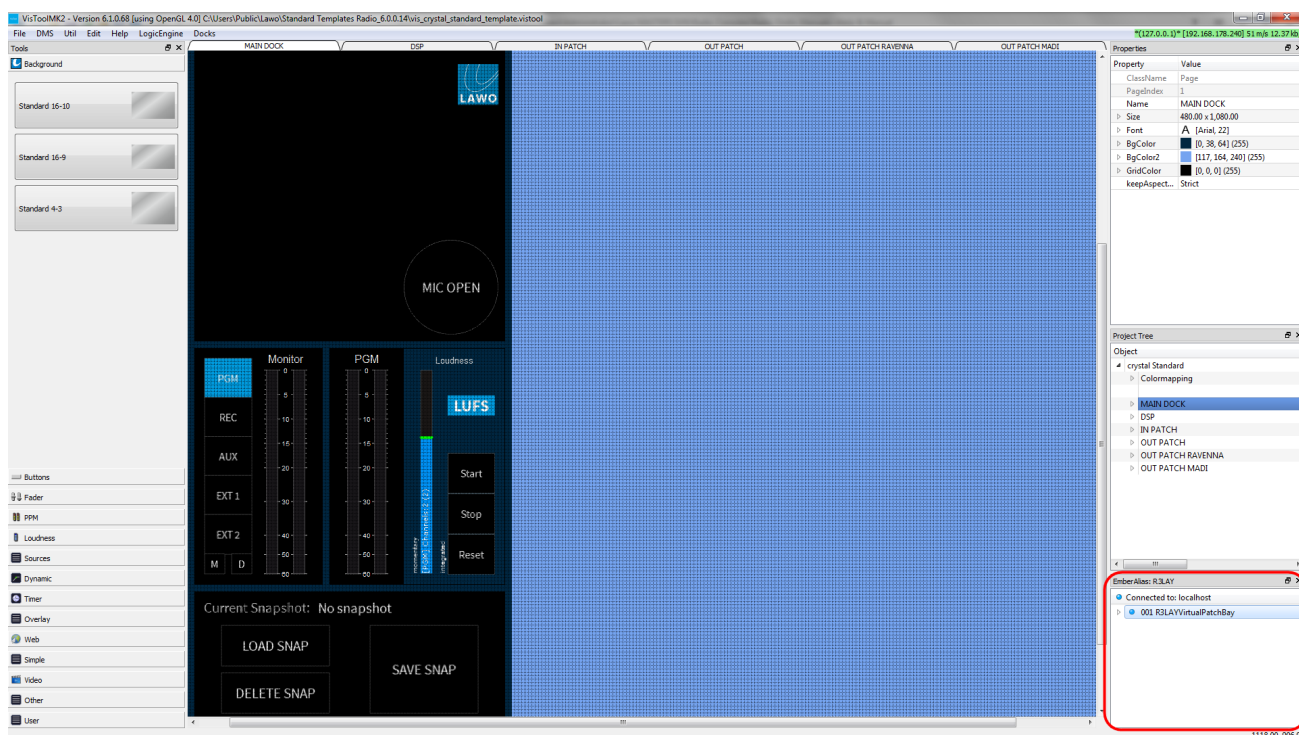
1. Select **File -> add/remove Ember Device** and tick the boxes to make your selections.

Note that you can only add devices which have already been defined in the [Ember+ Devices list](#). In our example, we have selected R3LAY:

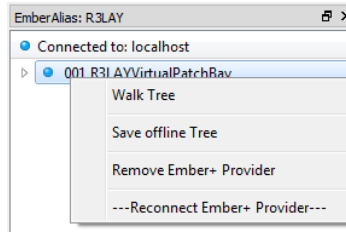


2. Click **OK** to save the changes and exit - a new docking panel is added to the workspace for each selected provider.

This provides access to the Ember+ Tree described on the next page. The docking panel can be opened, closed and repositioned in the usual manner, see [Changing Your Workspace](#).



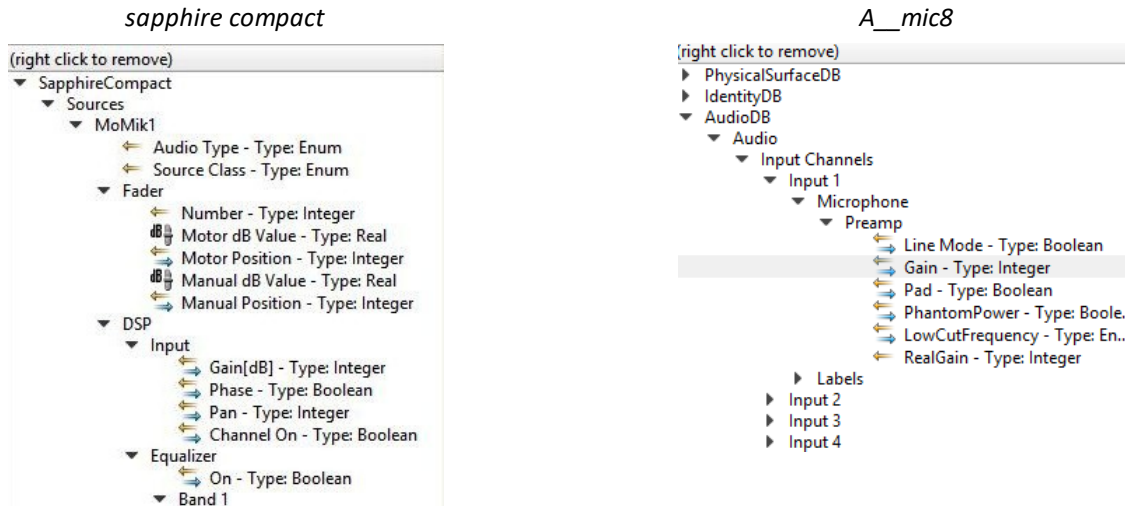
3. To remove a provider: right-click on its name and then select **Remove Ember+ Provider**:



## 10.4 The Provider Tree






1. Click to open the Ember+ provider tree - here you will see all published parameters from the provider.

If the provider is not connected, or no parameters have been published, then the tree will be empty. The examples below show parameters published from a sapphire compact and A\_\_mic8:



Within each branch, you will see an icon, the parameter name (e.g. **Audio Type**) and its data type (e.g. **Type: Enum**). These are important as they identify the parameter, and indicate how it can be used.

### Icons

-  Read only
-  Read & Write
-  PPM (Peak Value). These entries appear in green to indicate that this is a data stream.
-  Values with data type Real
-  dB value (Real values with minimum from -200 to -120 and maximum from 0 to +20).

### Data Types

- Type: Enum** Enumerations.  
This data type consists of a set of named values. For example, the **Audio Type** in a sapphire compact system might be **mono**, **stereo**, **surr 5.1**, etc.
- Type: Integer** Whole numbers.  
For example, the input **Gain** from an A\_\_mic8 might be **-12**, **0**, **+32**, etc.
- Type: Real** Values which increment continuously (including whole numbers and fractions).  
For example, the **Fader Manual dB Value** in a sapphire compact system might be **+3.5**, **+3.6**, etc.
- Type: Boolean** True or False values, output from a switched function.  
For example, the **Phase** in a sapphire compact system might be **On** (true) or **Off** (false).

The complete Ember+ Provider Tree for Lawo products can be found in the product's User Manual.

### Refreshing the Provider Tree

The Ember+ provider tree is detected when an Ember+ device is added to the project, or (for existing devices) when VisTool Editor restarts or the device comes online. This means that once the provider tree has been detected, the parameters continue to appear in the Docking Panel until either VisTool Editor or the device restarts.

If the provider's published parameters have been altered, then the simplest way to refresh the tree is to remove the device: right-click on the Ember+ provider, and then add the device back to the project via **File -> Add Ember Device**.

Note that for products configured by the ON-AIR Designer, the only way to change their published parameters is to edit and upload a new system configuration.

## 10.5 Assigning Parameters to VisTool Elements

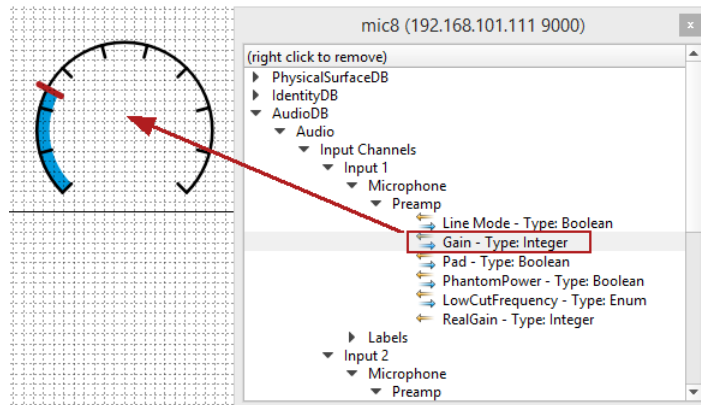
Once an Ember+ provider has been [added](#) to the project, the Provider Tree can be used to assign parameters to the following VisTool elements: [Box](#), [Slider](#), [Rotary Encoder](#), [Fader](#) and [Ember+PPM](#). Please follow the links for full details on each element. Here, we will use some examples to illustrate the operating principles.

Please note:

- **Box** elements - can display any Ember+ parameter value as text. Alternatively, they can be used as a button to control, or respond to, Boolean (true/false) parameter values.
- **EmberPPM** elements - can be assigned to any Ember+ PeakValue.
- **Fader** elements - can be assigned to dB values.
- **Slider & Rotary Encoder** elements - can be assigned to any Ember+ parameter except PeakValues.
- An Ember+ parameter can be assigned to multiple VisTool elements. Each VisTool element can be assigned to a single Ember+ parameter.
- Once VisTool Editor has connected to the Ember+ provider, you can save the provider tree so that you can continue to work when the device is offline. See [Working with an Offline Ember+ Provider](#).

To make a new assignment:

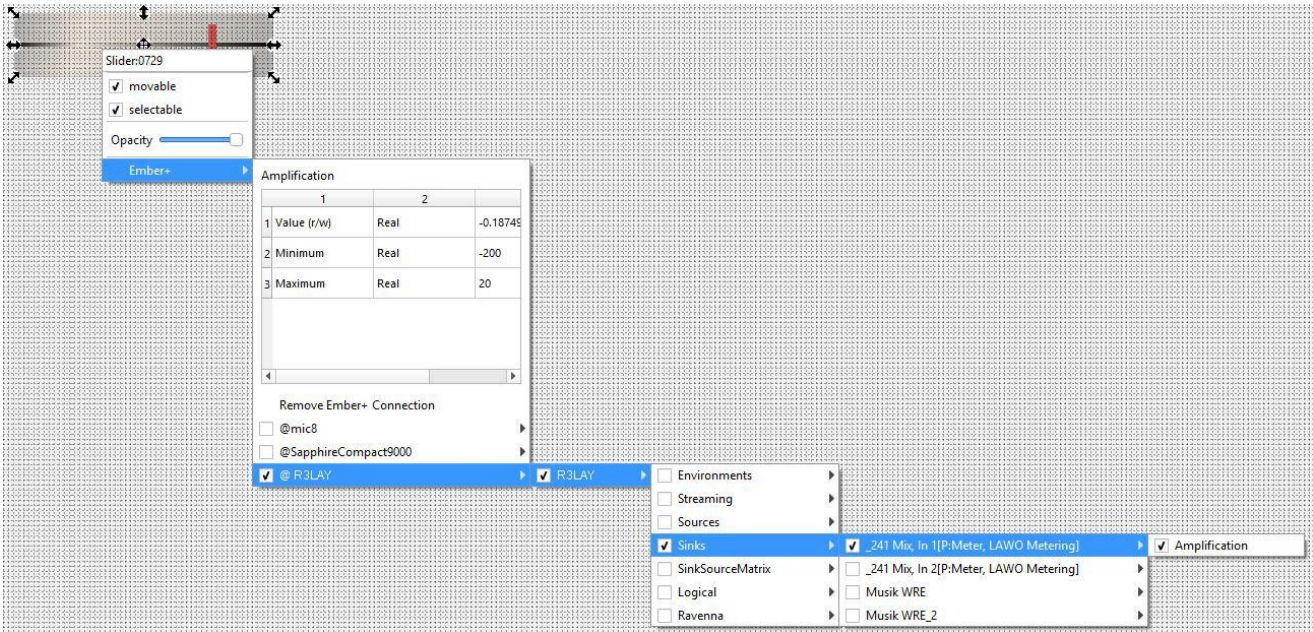
1. First, add a VisTool Ember+ element to the page - in our example, a **Rotary Encoder**.
2. Then drag and drop a parameter from the Ember+ provider tree onto the element - in our example, the **Gain** parameter from an A\_\_mic8 input:



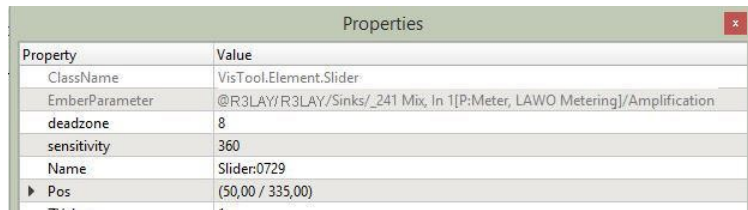
If you see a "No Entry" symbol, then the assignment is not valid (e.g. the VisTool element does not support Ember+ parameters).

To interrogate an existing assignment:

1. Right-click on the element and select **Ember+**. The Slider below is assigned to the **Amplification** from one of the **Sinks** (Destinations) in **R3LAY**:



2. Alternatively, select the element and look in the Properties list - the assignment appears in the **EmberParameter** field:



The next few pages show some more examples of Ember+ assignments.

### 10.5.1 A\_\_mic8 Input Gain

In this example, a **Box** element is assigned from the input Gain of an A\_\_mic8. This will allow the operator to adjust mic gain from VisTool.

The screenshot displays the VisTool interface with a 'Box' element configuration window. The 'Box' element is named 'Box:0703' and is assigned to the 'Gain' parameter of the 'mic8' device. The 'mic8' device is located at IP 192.168.101.111 port 9000. The configuration window shows the following properties:

Property	Value
ClassName	VisTool.Element.Box
EmberParameter	@mic8/AudioDB/Audio/Input Channels/Input 1/Microphone/Preamp/Gain
Name	Box:0703
Design	None
ButtonType	None
UserText	
Pos	(60,00 / 95,00)
ZValue	1
Size	64,00 x 30,00
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input type="checkbox"/> False
Glass Look	None
Metal Frame	4
Opacity	1
Roundness-X	0
Roundness-Y	0
TextAlign	Center
Font	A [Source Sans Pro, 16]
Color	[1, 1, 1] (255)
ColorNone	[136, 136, 136] (255)
ColorOff	[2, 253, 99] (255)
ColorPost	[253, 253, 20] (255)
ColorPre	[253, 2, 20] (255)
FontColor	[227, 227, 227] (255)
(Blink Simulation)	<input type="checkbox"/> False
Channel Index	None
ColorMap	default
DisplaySnapShotName	<input type="checkbox"/> False
Ember.Text	0
Glow On Press	<input type="checkbox"/> False
Reassignable	None
TextAppearance	Normal
usesLogic	<input type="checkbox"/> False

The 'mic8' device configuration window shows the following structure:

- PhysicalSurfaceDB
- IdentityDB
- AudioDB
  - Audio
    - Input Channels
      - Input 1
        - Microphone
          - Preamp
            - Line Mode - Type: Boolean
            - Gain - Type: Integer
            - Pad - Type: Boolean
            - PhantomPower - Type: Boolean
            - LowCutFrequency - Type: Integer
            - RealGain - Type: Integer
- Labels
  - fixed - Type: String
  - user - Type: String

The interface also shows a 'MIC 1' control panel with a 'Gain' knob and buttons for 'Mic', 'Line', 'Low Cut', '48V', and 'Pad'.

### 10.5.2 A\_\_mic8 Source Label

In this example, a **Box** element is assigned from the **user Label** of an A\_\_mic8. This provides identification of the source within VisTool:

The screenshot displays the VisTool interface with three main components:

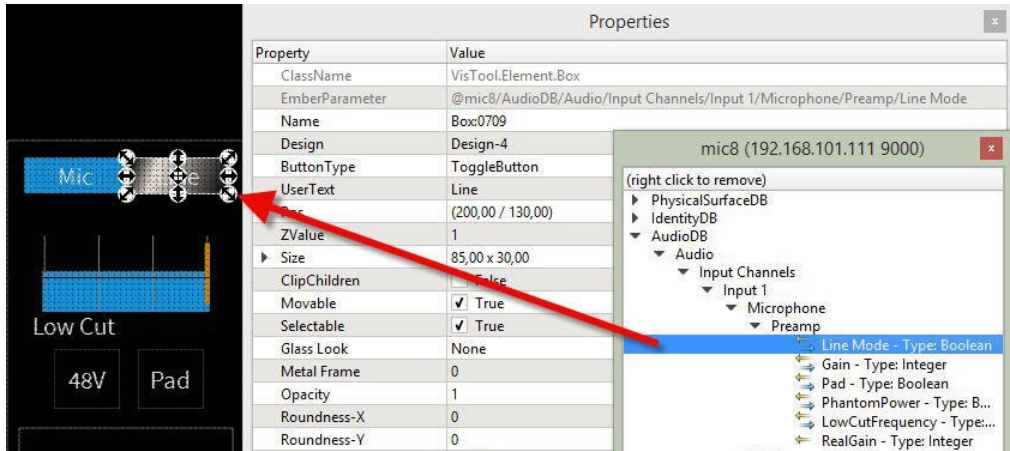
- Control Panel (Left):** Features a 'Mic Line' selector, a 'Low Cut' slider, '48V' and 'Pad' buttons, a 'Gain' meter showing '0', and a 'Mic/Line In 1' input selector.
- Properties Window (Center):** A table listing properties for the selected 'Box' element.
 

Property	Value
ClassName	VisTool.Element.Box
EmberParameter	@mic8/AudioDB/Audio/Input Channels/Input 1/Labels/user
Name	Box:0708
Design	None
ButtonType	ToggleButton
UserText	
Pos	(10,00 / 435,00)
ZValue	1
Size	180,00 x 30,00
ClipChildren	<input type="checkbox"/> False
Movable	<input checked="" type="checkbox"/> True
Selectable	<input checked="" type="checkbox"/> True
Glass Look	None
Metal Frame	0
Opacity	1
Roundness-X	0
Roundness-Y	0
TextAlign	Center
Font	A [Source Sans Pro, 16]
Color	[1, 1, 1] (255)
ColorNone	[136, 136, 136] (255)
ColorOff	[0, 154, 221] (255)
ColorPost	[253, 253, 20] (255)
ColorPre	[1, 1, 1] (255)
FontColor	[227, 227, 227] (255)
(Link Simulation)	<input type="checkbox"/> False
Channel Index	None
ColorMap	LawoCI
DisplaySnapshotName	<input type="checkbox"/> False
Ember.Text	MIC 1
Glow On Press	<input type="checkbox"/> False
Reassignable	None
TextAppearance	Normal
usesLogic	<input type="checkbox"/> False
- mic8 Object Tree (Right):** A hierarchical tree view for 'mic8 (192.168.101.111 9000)'. It includes sections for PhysicalSurfaceDB, IdentityDB, AudioDB, Audio, Input Channels, Input 1, Microphone, Preamp, Labels, and Output Channels. The 'Labels' section contains 'fixed - Type: String' and 'user - Type: String', with the latter highlighted in blue. A red arrow points from this 'user' label to the corresponding entry in the Properties window.

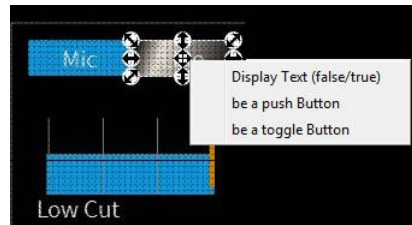


### 10.5.3 A\_\_mic8 Input Line Mode

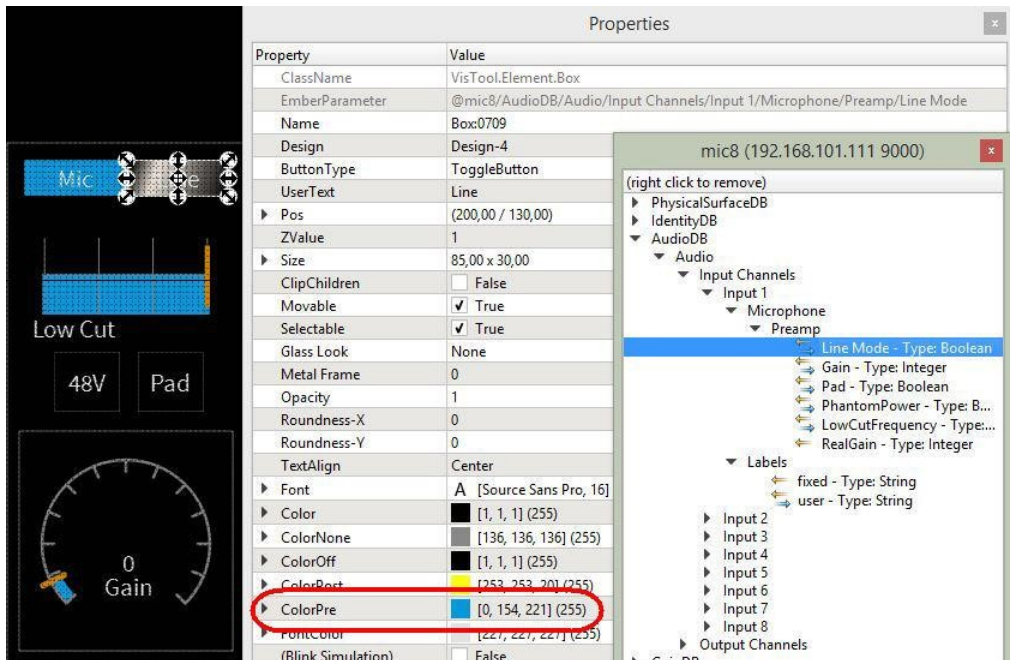
In this example, a **Box** element is assigned from the **Line Mode** of an A\_\_mic8:



For Boolean (on/off) parameters, you are asked how you wish the Box to behave:



- **Display Text (false/true)** - choose this option to display the parameter value as text only.
- **be a push button** or **be a toggle button** - choose this option to display the text and use the Box as a button (push = momentary, toggle = latching). The parameter state is displayed by the Box color, where the off state is defined by the **color off** option, and the on state is defined by the **color pre** option in the Properties list:



The operator will see the Line box illuminate (in blue) whenever the Line Mode becomes active.

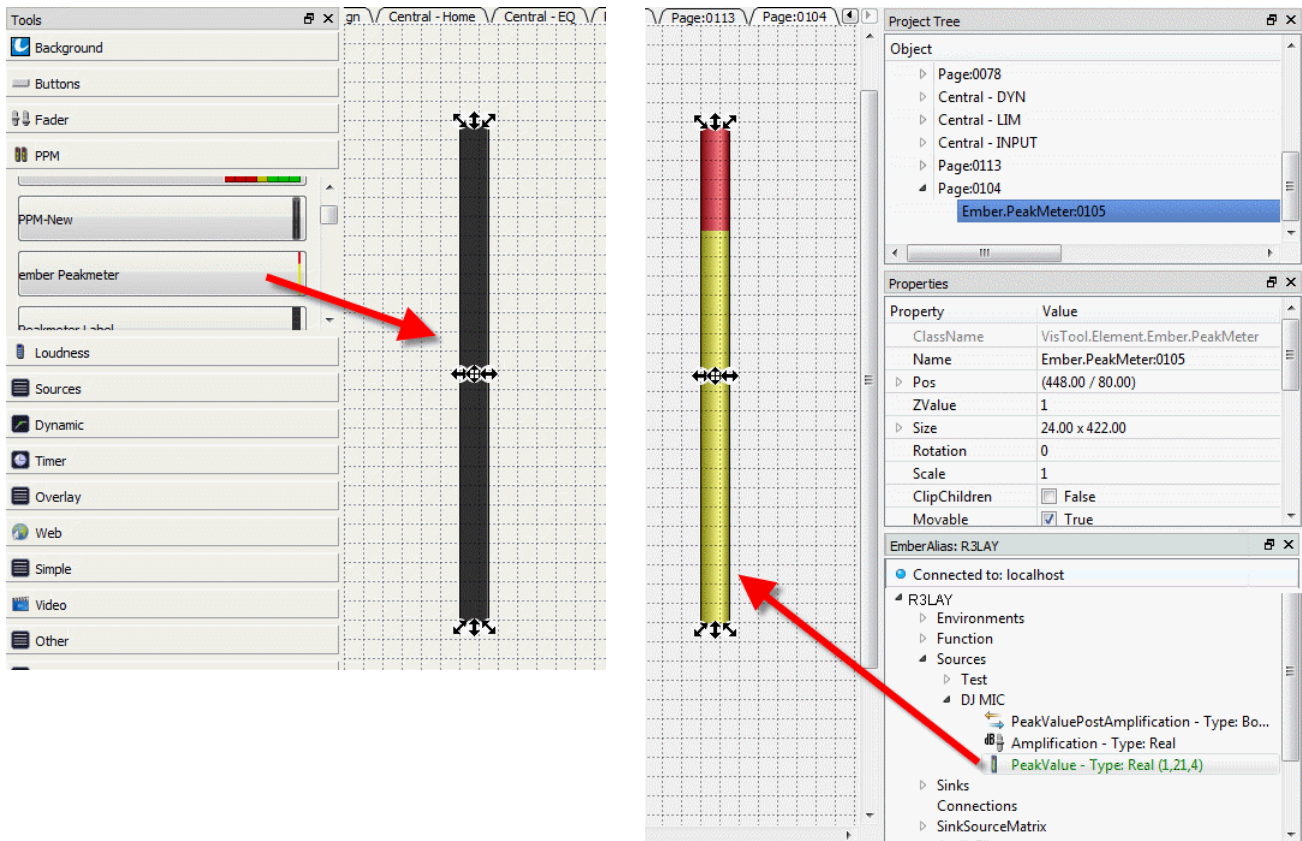
### 10.5.4 Ember+ Metering

VisTool supports a number of Ember+ PPM elements that can be assigned to a "PeakValue" data stream from an Ember+ provider. This allows VisTool to provide metering of an external device (via Ember+).

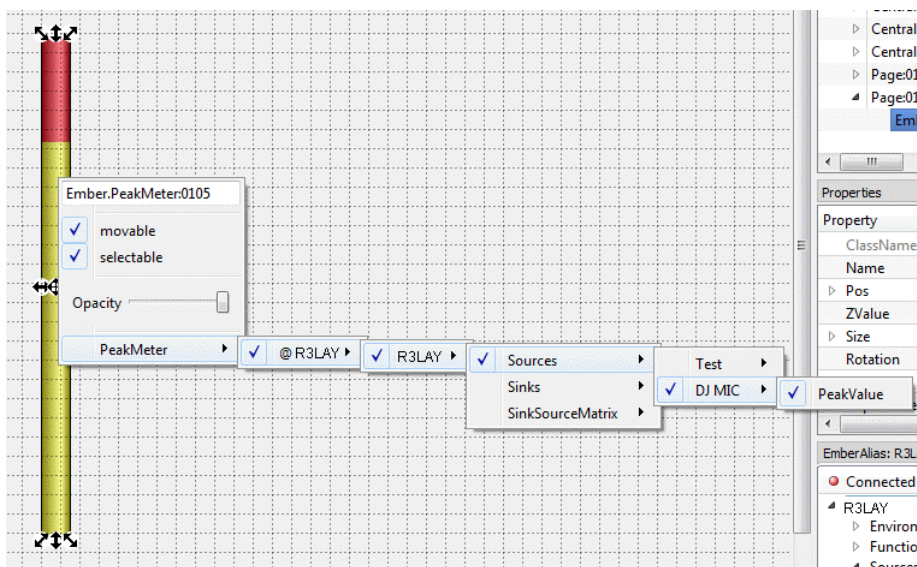
1. First, add one of the Ember+ [PPM elements](#) to the page.

You can choose to add just the meter (**ember Peakmeter**), or something more complete: meter + scale + housing (**EmberPPM**).

2. Then drag and drop a **PeakValue** parameter onto the PPM - the meter illuminates to indicate a valid assignment. In the example below, the meter will follow the PeakValue data stream from R3LAY's DJ MIC:



3. Right-click on the element to interrogate the assignment.

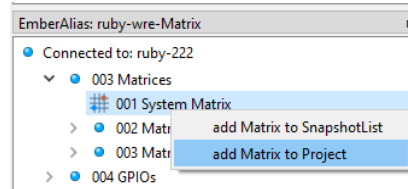


## 10.6 Controlling Ember+ Matrix Crosspoints

The matrix crosspoints within an Ember+ provider can be controlled directly from VisTool.

First, add the matrix to the VisTool project (via the right-click options):

*Ember+ Provider Tree*



Once the matrix is added, its crosspoints can be controlled using one of the following elements: [Matrix Button](#) or [Matrix Connect](#).

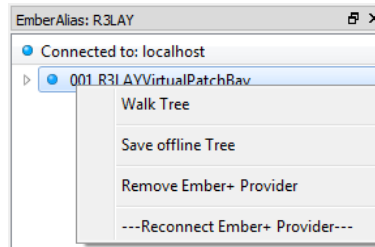
For more information, see [Configuring an External Matrix](#).

## 10.7 Working with an Offline Ember+ Provider

Once the Ember+ provider tree has been detected by VisTool Editor, you can choose to save the tree so that it can be loaded (if the device goes offline).

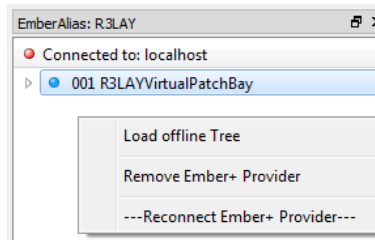
To save the Ember+ provider tree:

1. Right-click on the Ember+ provider, and select **Save offline Tree** - the data is saved automatically.



Once an Ember+ provider tree has been saved, it can be loaded once the device is offline:

1. Right-click on empty provider tree, and select **Load offline Tree** - the saved data is loaded:



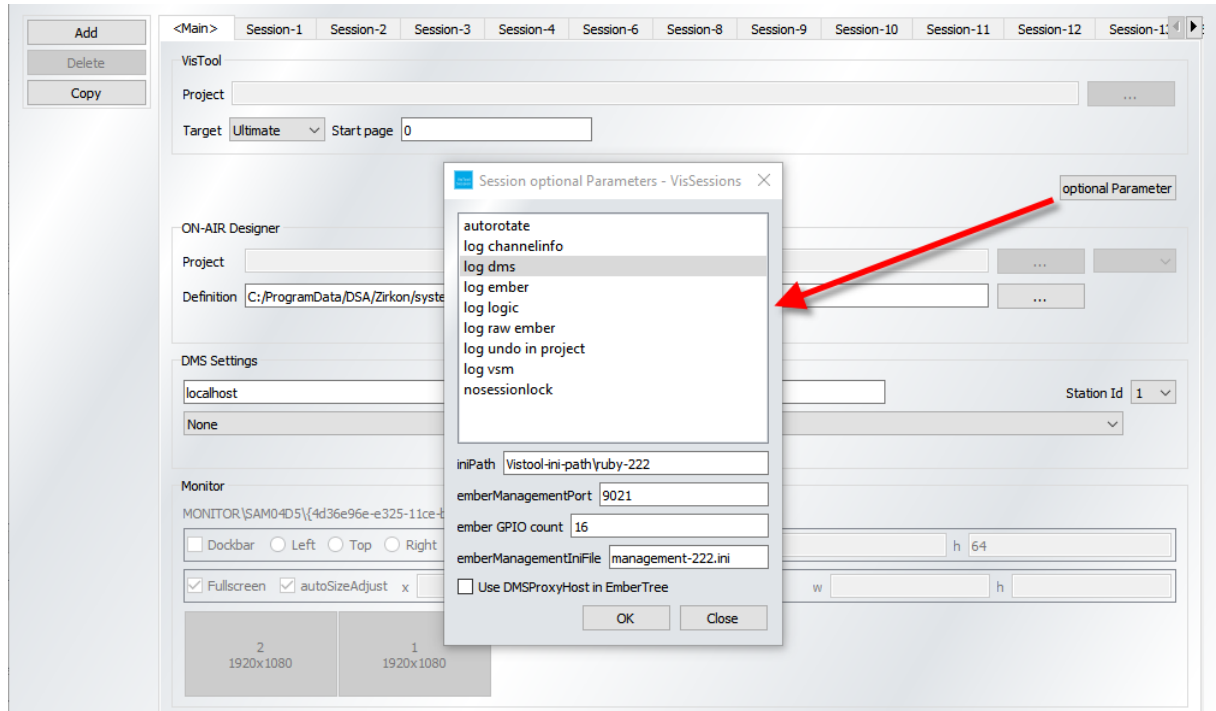
Note that this function loads the latest saved data only. Therefore, if new parameters are published by the provider while it is offline, you will need to reconnect the device in order to update the Provider Tree.

## 10.8 Using the Ember Management Port

The Ember Management Port is an Ember+ provider through which the Station ID and DMS connection of a VisTool session can be adjusted during runtime. In addition, "GPIOs" (Boolean, Integer, Real, String) can be configured to transfer information to other VisTool sessions or Ember+ devices.

The port can be defined for VisTool Editor (via its [Global Settings](#)), or for a VisTool session (via the 'optional Parameters' in the [Session Manager](#)).

*VisTool Session Manager*



In each case, define the **emberManagementPort** and number of GPIOs (via **ember GPIO count**). The port must be unique (on the host computer).

To use the management port within the same VisTool session, the automatically generated Ember device "self" can be used. This is always the configured management port of the respective session on the localhost (or editor).

The values of the EmberManagementPort GPIOs are stored in an "ini" file that can be specified in the **EmberManagementIniFile** field. You can enter either an absolute or relative file path. Relative paths refer to the [iniPath](#) (if one is defined) or to **C:\ProgramData\DSA** (if no iniPath is defined). If an **EmberManagementIniFile** is defined but not available, then a file with default values is created when Vistool starts up.

All adjustments made by registry entries are evaluated once only, when VisTool starts up. The following parameters can be adjusted.

### Station ID

The [Station ID](#) of the current Vistool session can be customized via the EmberManagementPort (/ VisToolMk2 / Management / Station Id).

### DMS Connection

Two modes are available for switching over the DMS connection. The mode is controlled via the registry entry EmberManagementFlags (bit1) which must be created manually (EmberManagementFlags - REG\_DWORD).

For VisTool Editor:

```
Computer \ HKEY_LOCAL_MACHINE \ SOFTWARE \ DSA \ VisToolMk2 \ EmberManagementFlags
```

For VisTool Sessions:

```
Computer \ HKEY_LOCAL_MACHINE \ SOFTWARE \ DSA \ VisToolMk2 \ Session-n \ EmberManagementFlags
```

#### ➤ Mode 1: Switching the first DMS connection

(Default: EmberManagementFlags not available / bit1 = 0).

The EmberManagementPort (/ VisToolMk2 / Management / Target Ip) switches the IP address of the first DMS connection (CONN-ADR-1). The use is only permitted if localhost as DMSProxyHost is used.

The change of the IP address takes approximately 30 seconds, after which all parameters of the new device are received. Any further change is prevented during this period (/ VisToolMk2 / Management / Target Ip isOnline = false).

The active host is displayed in the Ember tree under / VisToolMk2 / Project / Target Host.

The isOnline status, in combination with the target host, can be used to signal the switchover (using the Logic Engine).

#### ➤ Mode 2: Switching the DMS ProxyHost

(EmberManagementFlags bit1 = 1).

The EmberManagementPort (/ VisToolMk2 / Management / DMSProxyHost) switches the DMS proxy host. The DMS connections on both hosts must be identical. After the switchover, the corresponding DMS connection is used on the selected host.

A renewed switchover is prevented for approximately 30 seconds (/ VisToolMk2 / Management / DMSProxyHost isOnline = false).

The active host is displayed in the Ember tree under / VisToolMk2 / Project / Target Host.

The isOnline status, in combination with the target host, can be used to signal the switchover (using the Logic Engine).

### GPIOs

Input / output options are available under / VisToolMk2 / Project / gpio / [Boolean | Integer | Real | String]. These can be used to transfer information to other VisTool sessions or Ember+ devices.

For example, via the Logic Engine, they can be used to minimize or maximize the Vistool session, control the display of overlays, or swap strings for display.

By default, there are 8 GPIO channels but the number can be adjusted using the **ember GPIO count** (described earlier).

# 11. Configuring an External Matrix

---

This chapter describes how to configure an external matrix in VisTool.

Topics include:

- [Introduction](#)
- [Configuring an Ember+ Matrix](#)
- [Configuring a DSA Matrix \(for CAN / KPF\)](#)
- [Next Steps](#)

## 11.1 Introduction

The matrix crosspoints within an external device can be controlled directly from VisTool. This provides a much simpler alternative to using the [Matrix Control](#) elements in the ON-AIR Designer.

Two possible communication protocols are supported:

- **EmBER+** - for supporting EmBER+ devices.
- **CAN / KPF** - for ruby, sapphire, crystal, Nova29 or Power Core.

The configuration is done in two parts. First, configure the device and add the matrix to the VisTool project. Then, use one of the following methods to configure the crosspoint control:

- [Matrix Buttons](#) - create on-screen source and target buttons (for direct operator control).
- [Matrix Connects](#) - create matrix connects in the Logic Engine (for indirect control from another logical function).

This chapter describes how to configure the device and add the matrix to the VisTool project. For more information on configuring the crosspoint control, follow the links above.

Multiple matrices of different types can be addressed: Ember+ and CAN / KPF.  
Additional network drivers are not required, as the connection is established directly through VisTool.

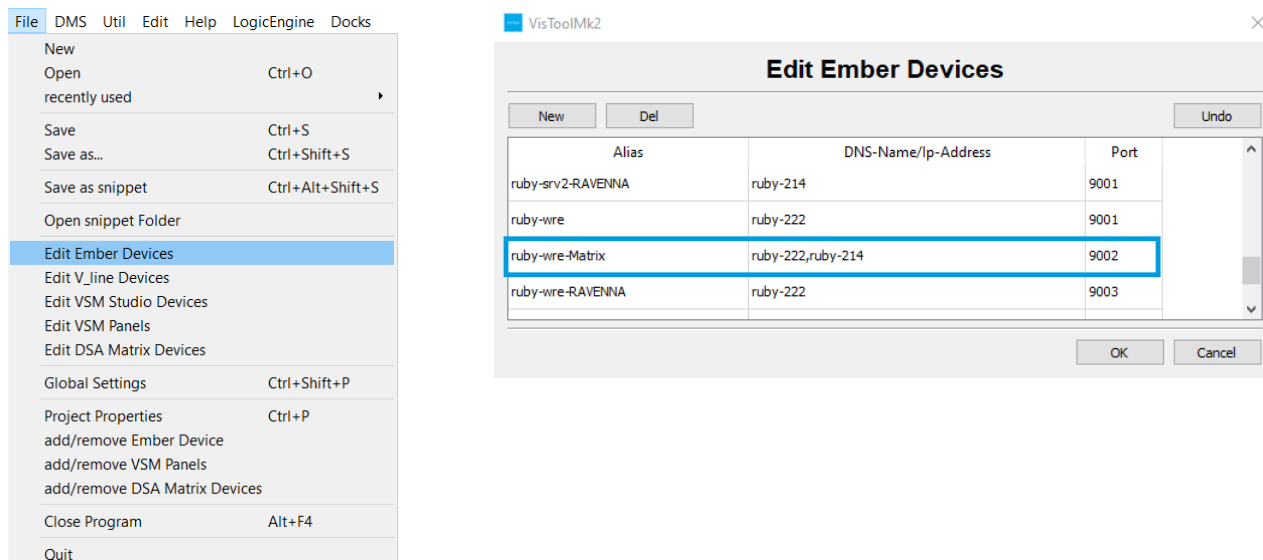


## 11.2 Configuring an Ember+ Matrix

1. Add the device to the 'Ember+ Devices List' (via **File -> Edit Ember Devices**).

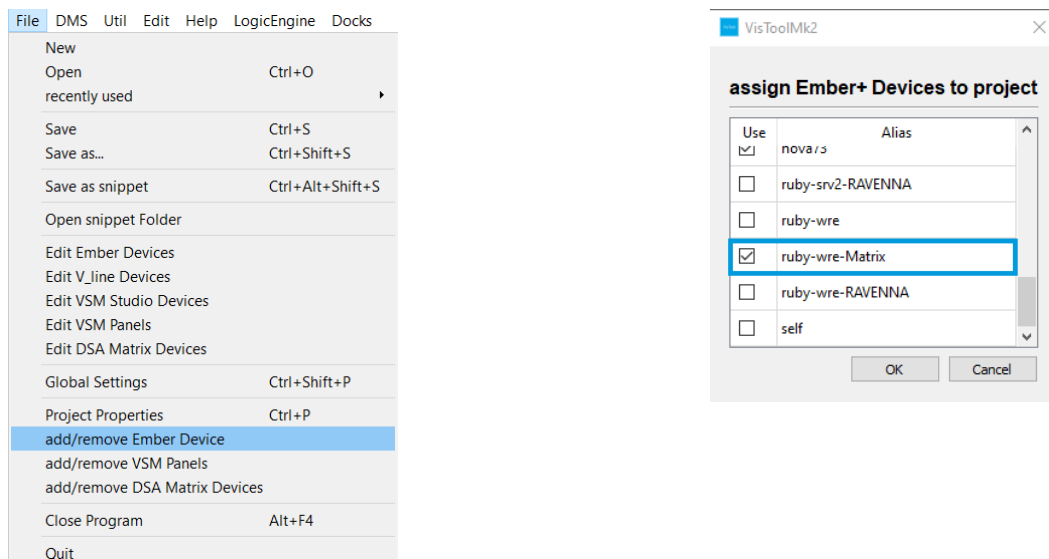
The method is the same as for other Ember+ devices. See [The Ember+ Devices List](#).

Redundant connections can be specified by entering the two IP addresses (or DNS names) separated by a comma.



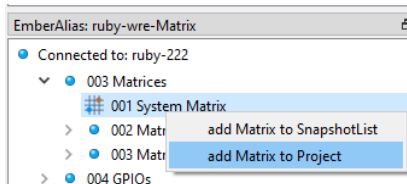
2. Add the Ember device to the VisTool project (via **File -> add/remove Ember Device**).

The method is the same as for other Ember+ devices. See [Adding an Ember+ Provider to the VisTool Project](#).



A new docking panel is added for the selected provider.

3. Click to open the Ember+ Provider Tree, and add the matrix to the project (via the right-click options):

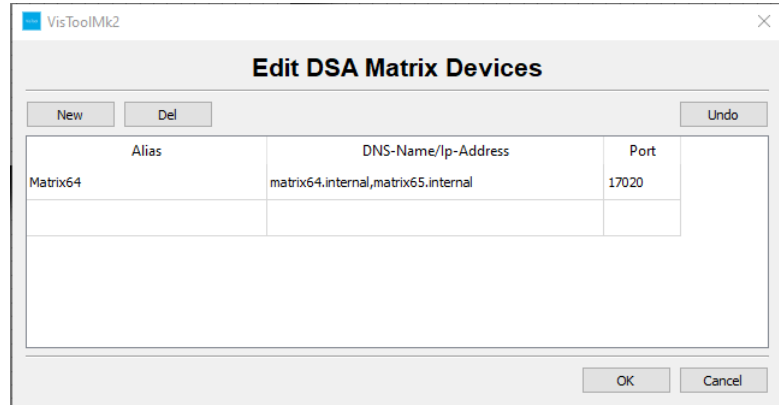
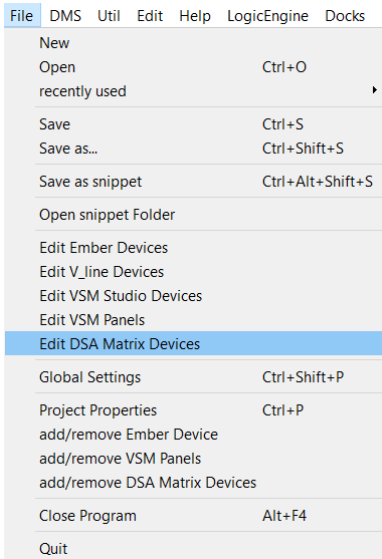


## 11.3 Configuring a DSA Matrix (for CAN / KPF)

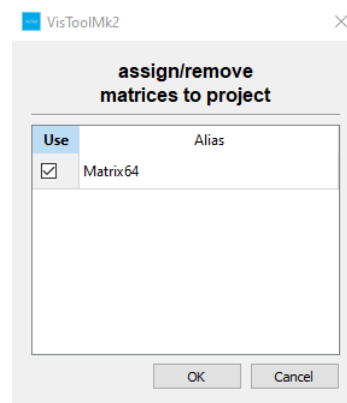
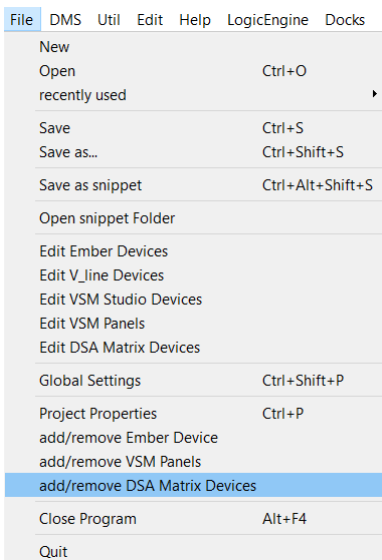
A DSA Matrix must be configured if the method of communication is CAN / KPF. The configuration method is similar to that used for Ember+.

1. Add the device to the 'DSA Matrix Devices List' (via **File -> Edit DSA Matrix Devices**).

Redundant connections can be specified by entering the two IP addresses (or DNS names) separated by a comma.



2. Add the DSA Matrix device to the VisTool project (via **File -> add/remove DSA Matrix Devices**).



### 11.4 Next Steps

Once a matrix is added to the VisTool project, its crosspoints can be controlled from [Matrix Button](#) elements or a [Matrix Connect](#).

## 12. The Logic Engine

---

This chapter describes the Logic Engine.

Topics include:

- [Introduction](#)
- [Adding VisTool Elements to the Logic Engine](#)
- [Operating Principles](#)
- [Logical Elements](#)
- [Functions](#)

To support functions programmed using the Logic Engine, you will need a **VisTool Unlimited** license plus the **Ember+/Logic AddOn**.

### 12.1 Introduction

The Logic Engine can be used to configure logical outcomes for a range of VisTool elements (both DMS and Ember+).

Various logical building blocks are available, including Any to String, Boolean AND, Boolean NOT, Boolean OR, etc.

Note that the Logic Engine editor opens in its own operating window. However, all settings are [saved](#) as part of the VisTool project in the usual manner.

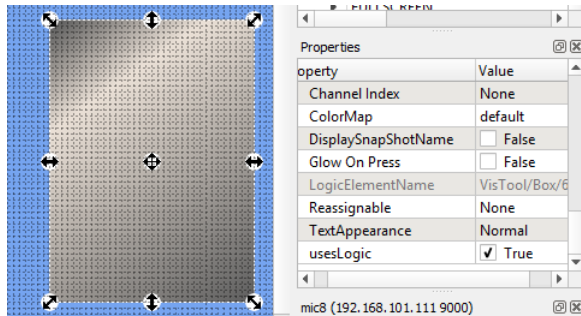
If you have used the Logic Engine to program functions within your configuration, then your VisTool runtime PC will require a **VisTool Unlimited** [license](#) with the **Ember+ / Logic AddOn**.

## 12.2 Adding VisTool Elements to the Logic Engine

The following VisTool elements can be added to the Logic Engine editor: [Box](#), [Overlay](#), [Fader](#), [WebView](#) and [VSM-Button](#).

To add an element, return to the main VisTool Editor window and:

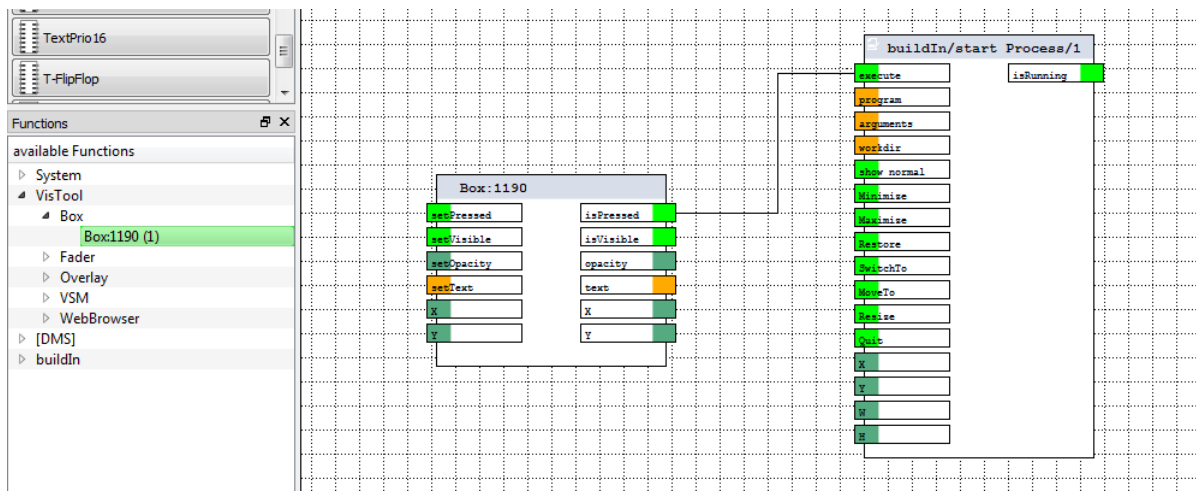
1. Select the element (e.g. a **Box**).
2. Open its [Properties](#) panel, or right-click on the element, and set the **usesLogic** option to **True**:



3. [Save](#) the project.

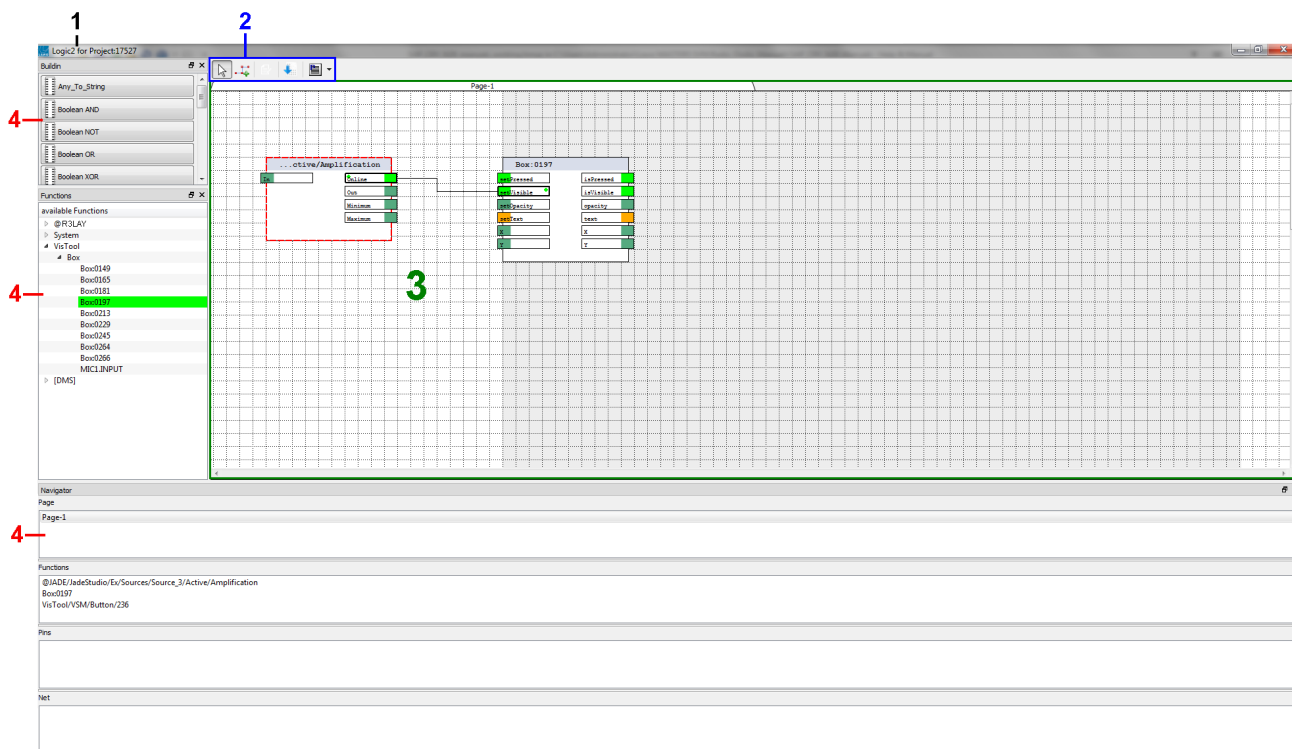
If you now open the Logic Engine editor, the element appears under the **VisTool** branch of the available [Functions](#) docking panel. From here it can be used to connect to other logical elements as described later.

*VisTool Box element added to Logic Engine editor*



## 12.3 Operating Principles

The Logic Engine editor is opened by selecting **File -> Logic Engine -> Edit** from the main menus - the editor opens in its own operating window (providing you have a valid license). At any time, you can switch between the Logic Engine and VisTool Editor using the icons on the Windows taskbar.



### 1 Project Name

In the headline you will see "Logic2 for xxx" where xxx identifies the VisTool project.

### 2 Tools

Use the **Selection Mode** tool (keyboard shortcut = S) to drag and drop elements onto the design area, move an element, select a connection, etc.

Use the **Create Connector** tool (keyboard shortcut = C) to draw the connecting lines between nodes.

Click on **Compiler** to run the [Compiler](#).

### 3 Design Area

The design area shows the elements added to the current page, and their available nodes and connections.

Nodes are color-coded to reflect their logical data type:

- Green = boolean
- Orange = string
- Dark blue = long
- Light blue = integer
- Dark green = double.

Connected pins are indicated by a green dot. Note that the green dot does not update in real time but appears after running the [Compiler](#).

Multiple pages can be [added](#) if you wish to divide the logic for different parts of the network (to simplify the programming).

#### **4** The Docking Panels

To the side or below the design area are the docking panels:

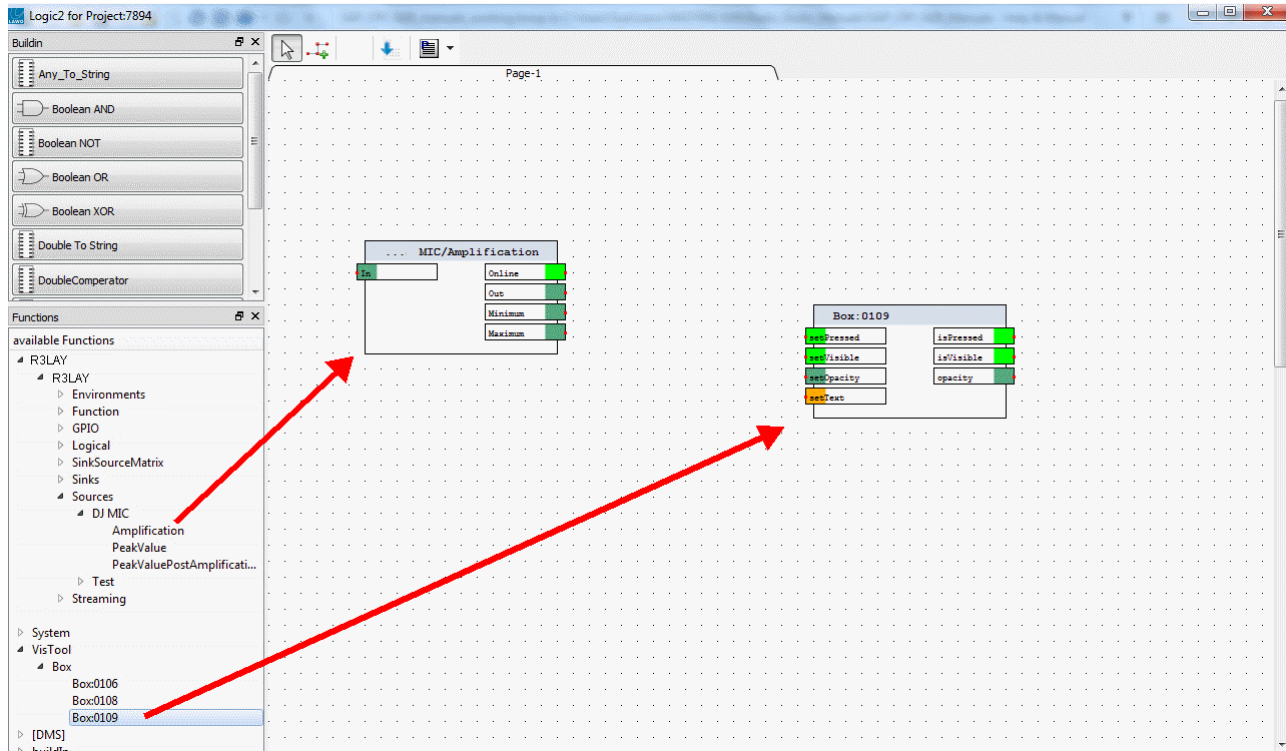
- **Buildin** - contains all the [logical elements](#): Any to String, AND, NOT, OR, etc.
- **Functions** - lists all [functions](#) available to the Logic Engine editor. These include Ember+ provider parameters; System parameters (SysClock and Boolean); VisTool elements (added from the project); DMS MF Keys and Settings; and any logical elements already used in the Logic Editor.
- **Navigator** - lists all the **Pages, Functions, Pins** and **Nets**.

Each docking panel can be opened, closed, resized, floated or docked in the usual manner, see [Changing Your Workspace](#). Right-click on any docking panel to choose which panels are visible in the Logic Editor window.



### 12.3.1 Adding and Positioning Elements

Using the **Selection Mode** tool (arrowhead cursor), drag and drop elements into the design area:



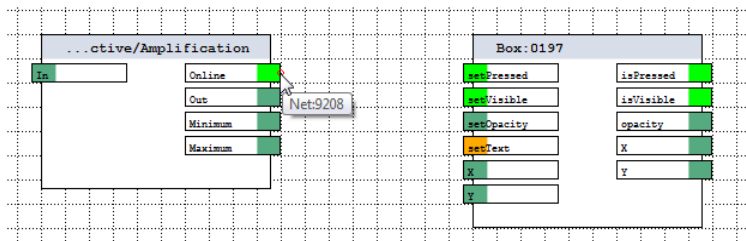
Once added, you can select an element - indicated by the dotted red outline - and click and drag to move it. Note that all elements snap to the grid (shown by the black dotted background).

If you move an element which is already connected, then all connections to and from the element are removed. Therefore, it is important to place your elements carefully before making any connections.

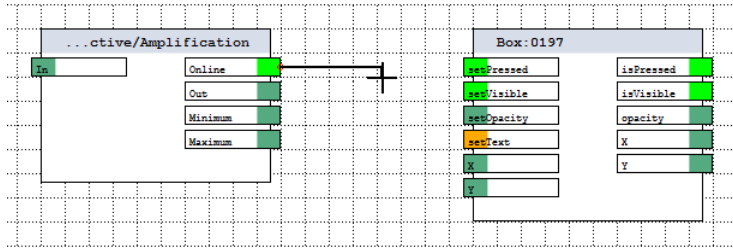
### 12.3.2 Making Connections

Once you have added and positioned the elements, use the **Create Connector** tool (crosshair cursor) to draw the connection paths and program the logic.

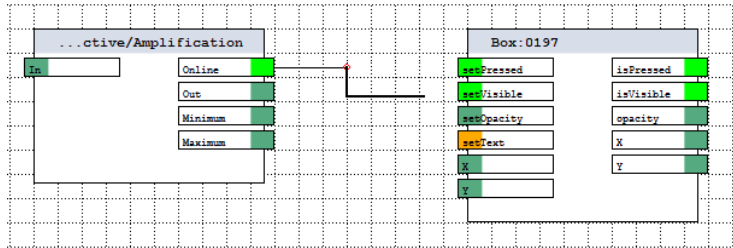
Start by clicking on an input (or output) node. You must click on the border of the node - a small red dot and the default connection name (e.g **Net9208**) appears when you have selected the correct position:



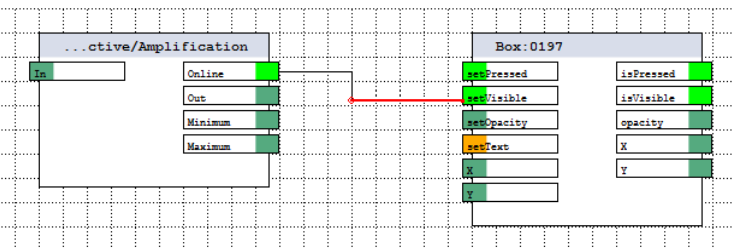
As you move your mouse, a black line is drawn:



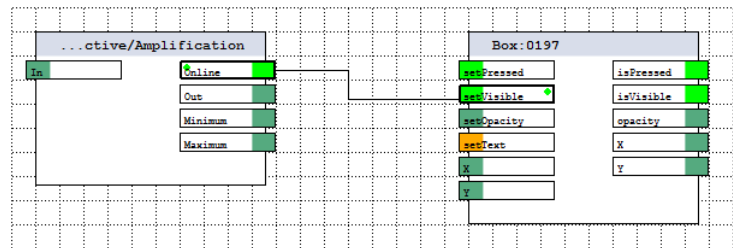
You can turn 90 degrees (if you wish) and click in "mid-space" to confirm the change of direction - a perpendicular line is added:



Keep clicking and turning until you reach the destination node - the line turns red to indicate that this is a valid connection:



Click on the border of the destination node to complete the connection - the connecting line turns black:



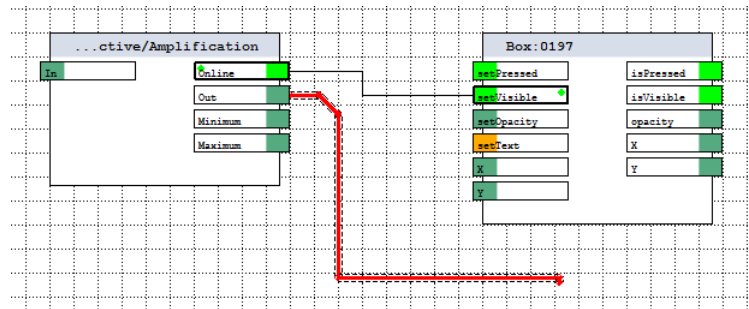
Connected pins are indicated by a green dot, and appear in the [Navigator](#) docking panel. Note that the green dot does not update in real time but appears after running the [Compiler](#).

## Illegal Operations

If, at any time, you see a red No Entry symbol, then the connection is either not valid or you are trying to perform an illegal operation:

- An input node can be connected to only one output.
- Logic elements and other connections cannot be crossed.

## Tips



While drawing a connection path with the **Create Connector** tool:

- press the SPACE bar to change a line's direction - the angle of the line cycles through the available options: horizontal, 45 degree angle or vertical.
- press **ESC** to undo the last action.
- single-click to select an individual line - it highlights in red.
- double-click to select the complete connection path - it highlights in red (as shown opposite).

Or, using the **Selection Mode** tool:

- single-click to select an individual line or object - it highlights in red.
- double-click to select a complete connection path - it highlights in red.
- press and hold **SHIFT** and then click to select several lines/objects, or click and drag around an area to select all lines/objects within it - the selections highlight in red.

Press **DELETE** to remove all highlighted lines, paths and objects.

If your configuration requires a lot of logic, then [Net Labels](#) are recommended. They allow you to scatter a logical signal flow across multiple pages, and make the Logic Engine editor window easier to read.

Use the [Free Text](#) element to annotate the Logic Engine editor window.

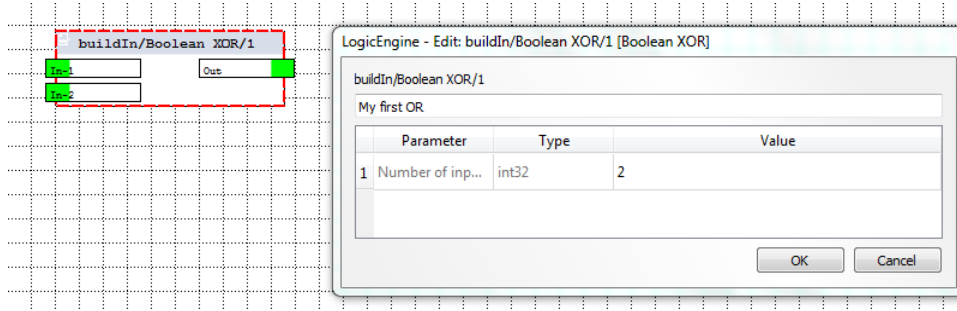
Once you have finished editing, run the [Compiler](#) to check the configuration.

## Connecting Different Data Types

The Logic Engine can automatically convert the following data types: boolean, integer, long and double. This allows you to connect nodes of different types. For example, if an integer output is connected to a boolean input, then values below 0 are converted to False, and values equal to 0 or above are converted to True.

### 12.3.3 Defining the Properties

Using the **Selection Mode** tool (arrowhead cursor), right-click on an element, or select it and press F2, to open its Properties window. From here you can assign a name and adjust the element's characteristics - these vary depending on the selection. In our example, the Boolean XOR can support different numbers of inputs:



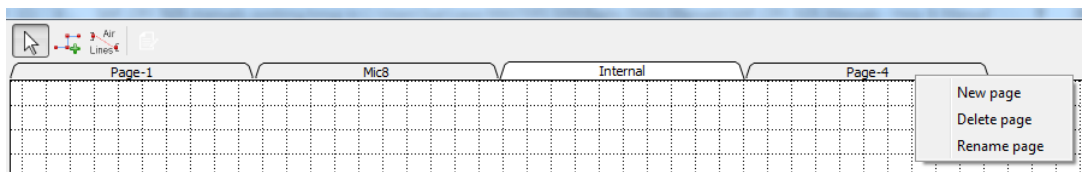
See [Logical Elements](#) for more details.

### 12.3.4 Creating and Managing Pages

You can add multiple pages to the Logic Engine editor if you wish to divide the logic. This can help to simplify the programming in a complex configuration. For example, to divide the logic for the A\_\_mic8 units from internal console functions, and so on.

To link the logic between pages, you can use [Net Label](#) elements. For cross-page configuration, the **Net Label** name must have "\$" in front.

Pages are created, renamed and deleted in a similar manner to the main VisTool Editor window - right-click on the page bar and choose an option:



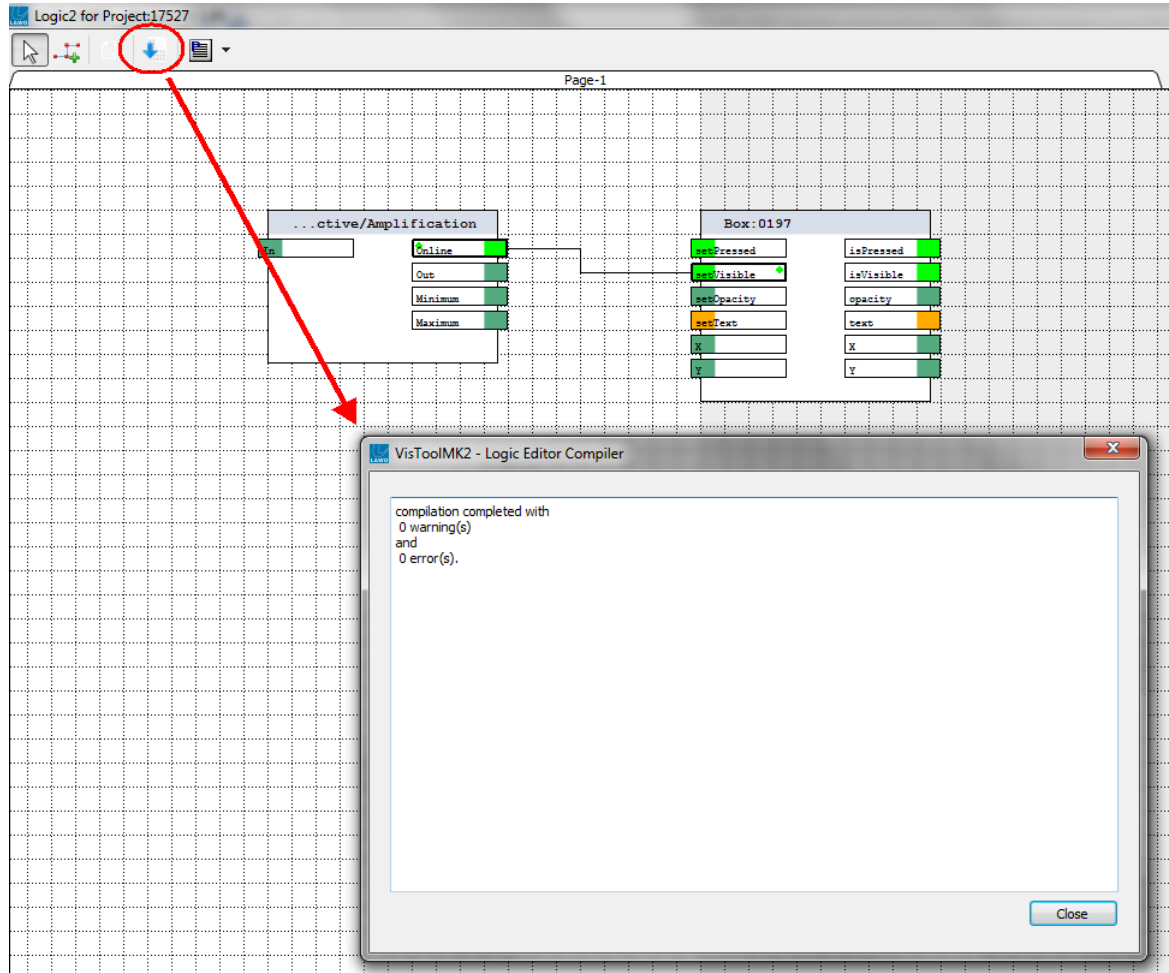
### 12.3.5 Saving the Configuration

Once you have completed all the logical connections, the settings should be [saved](#) as part of the VisTool project in the usual manner.

### 12.3.6 Running the Compiler

The Compiler must then be used to check the Logic Engine configuration. It reports any errors and also updates the connected pin states.

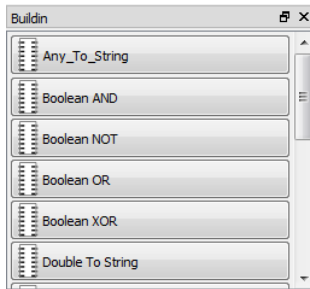
To run the Compiler, select the **Compile** button from the main tool bar - the Compiler window opens reporting any warnings and errors:



In addition, any changes to the connection status of individual pins are updated - in our example, the green dots beside the **OnLine** output and **setVisible** input pins confirm that they are correctly connected.

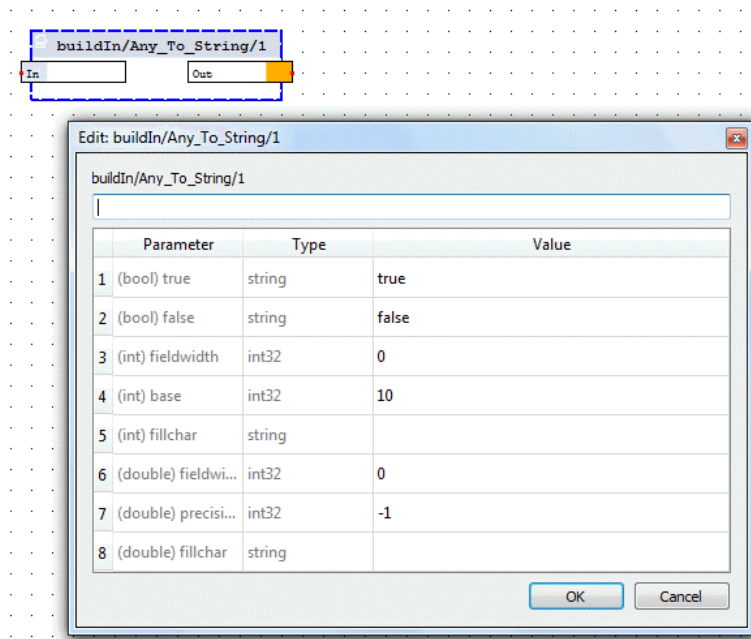
## 12.4 Logical Elements

The following logical elements are available in the **Buildin** docking panel:



- [Any to String](#)
- [AND / NOT / OR / XOR](#)
- [Double to String](#)
- [Double / Integer / UInt Comparator](#)
- [Constants: Boolean, Double, Integer, String](#)
- [String Comparator](#)
- [JsonObjectModifier](#)
- [Button 16](#)
- [DialPad](#)
- [DoublePrio 16](#)
- [MonoFlop](#)
- [Prio 16](#)
- [start Process](#)
- [TextPrio 16](#)
- [T-Flip Flop](#)
- [Matrix Connect](#)
- [LS Command](#)
- [Free Text](#)
- [Net Label](#)

## 12.4.1 Any to String



This element converts any parameter value into a string.

All possible data types can be converted; the output will be either a fixed character string for Boolean (true / false) values, or a fixed-length number for Integers or Doubles (fractional values).

Typical applications might be to output the text string "MIC ACTIVE" when the incoming parameter "Mic Active" is True. Or, to output the first two digits of a continuous "PeakValue" reading as a number.

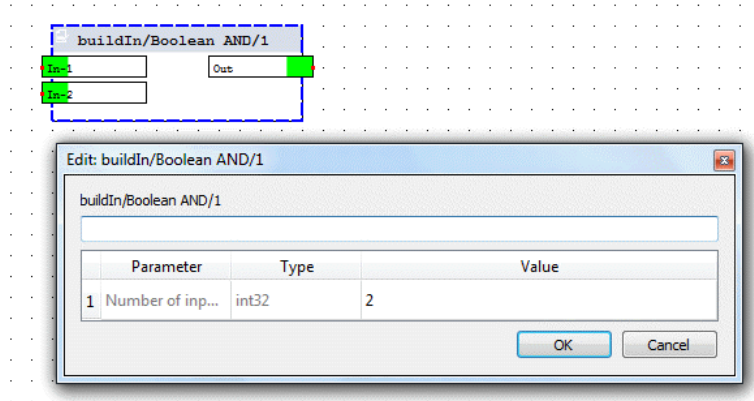
Right-click on the element to define its Properties - here you will see all options for the supported data types. The **Type** column indicates the type of **Value** that should be entered:

Parameter	Value
(Boolean) true / false	Enter the string which will be output for each parameter state: true (and false). Any string can be entered: text or numbers.
(int) fieldwidth (int) base (int) fillchar	Enter the number of digits for the output string. Enter the number base which will be used for the output string: 10 = Decimal; 16 = Hexadecimal; 2 = Binary. The fill characters (if entered) appear at the front of the output string: for example, if the <b>fieldwidth</b> = 3, the input value = 2, and the <b>fillchar</b> = 0, then the output string = 002. You may also use a blank or any other character as the <b>fillchar</b> .
(double) fieldwidth (double) precision (double) fillchar	Enter the number of digits for the output string (including decimal places, '.' and '-'). Enter the number of decimal places for the output string: -1 = infinite; 0 = none; 1 = one decimal place; and so on. The fill characters (if entered) appear at the front of the output string: for example, if the <b>fieldwidth</b> = 6, the input value = 1.25, and the <b>fillchar</b> = 0, then the output string = 001.25. You may also use a blank or any other character as the <b>fillchar</b> .

### 12.4.2 AND / NOT / OR / XOR

The next four elements can be used to process Boolean parameters - i.e. those with only two conditions: true (active) or false (inactive).

#### Boolean AND



This element can be used to 'AND' several input parameter values.

Right-click on the element to define its Properties:

Parameter	Value
Number of inputs	Enter the number of logical inputs. Then select <b>OK</b> - the element updates accordingly.

#### Boolean NOT

This element can be used to invert a logical signal. It has no special Properties, other than to name the element.

#### Boolean OR

This element can be used to 'OR' several input parameter values.

Right-click on the element to define its Properties:

Parameter	Value
Number of inputs	Enter the number of logical inputs. Then select <b>OK</b> - the element updates accordingly.

#### Boolean XOR

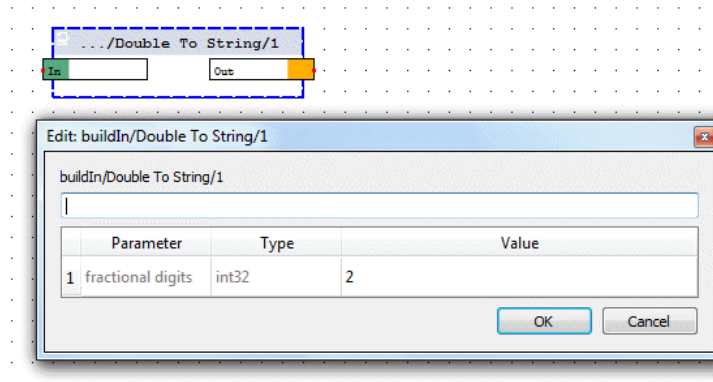
This element can be used to 'EITHER OR' several input parameter values.

Right-click on the element to define its Properties:

Parameter	Value
Number of inputs	Enter the number of logical inputs. Then select <b>OK</b> - the element updates accordingly.



### 12.4.3 Double to String



This element converts Double parameters - i.e. those using fractional values - into a string. For example, to output a continuous "PeakValue" reading using say 2 decimal places.

Right-click on the element to define its Properties:

Parameter	Value
fractional digits	Enter the number of decimal places for the output string: -1 = infinite; 0 = none; 1 = one decimal place; and so on.

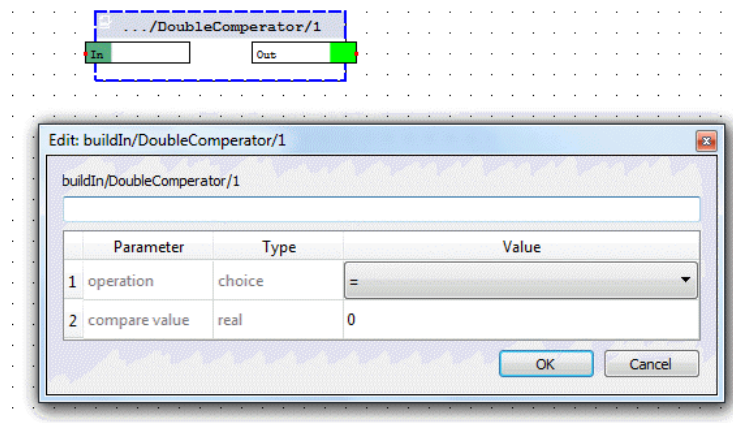
### 12.4.4 Double / Integer / UInt Comparator

The double, integer and unit integer comparators can be used to compare a numerical input with a defined value, and output a Boolean value: either true or false.

Choose the **Double Comparator** to process fractional values or the **Integer Comparator** for whole numbers. The **UInt Comparator** can be used to compare whole numbers within a range.

A typical application might be to light a VisTool Overload indicator, once an incoming 'PeakValue' meter reading exceeds a certain threshold level.

#### Double Comparator



Right-click on the element to define its Properties - the **Type** column indicates the type of **Value** that should be entered:

Parameter	Value
operation	Select the type of comparison from the drop-down menu. The options are: = (equal to) < (less than) > (greater than) != (not equal to) <= (less than or equal to) >= (greater than or equal to)
compare value	Enter the reference value to be used for the comparison. You can enter any value.

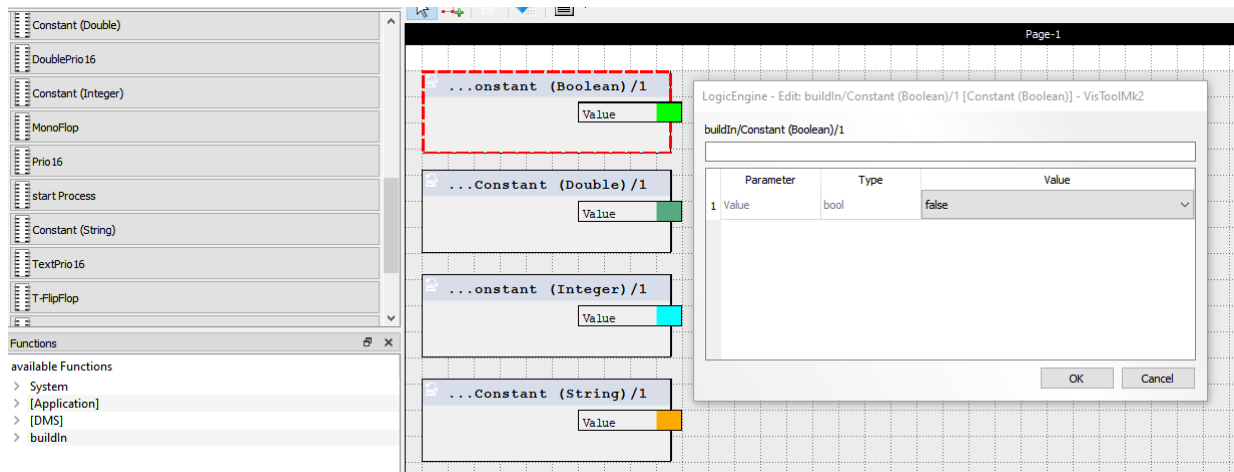
#### Integer Comparator

Identical to the above, except the input and **compare value** must be integers (whole numbers).

#### UInt Comparator

Identical to the Integer Comparator, but the **compare value** determines the range.

### 12.4.5 Constants: Boolean, Double, Integer, String



The following elements can be used to define constants in the Logic Engine:

- **Constant (Boolean)**
- **Constant (integer)**
- **Constant (double)**
- **Constant (string)**

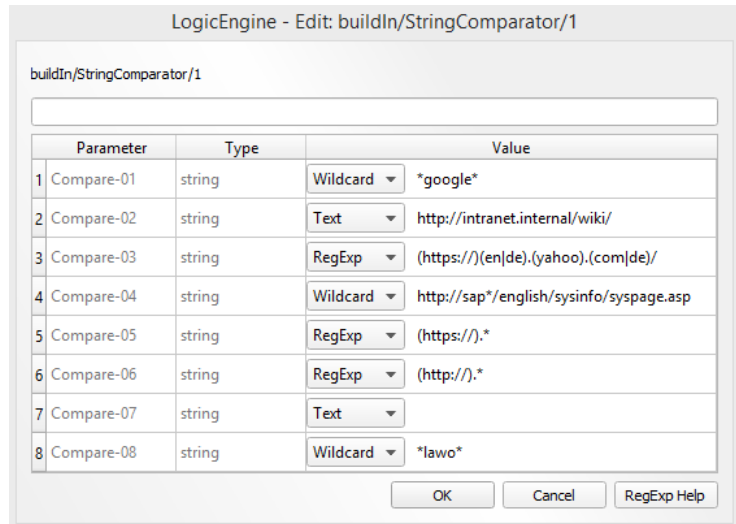
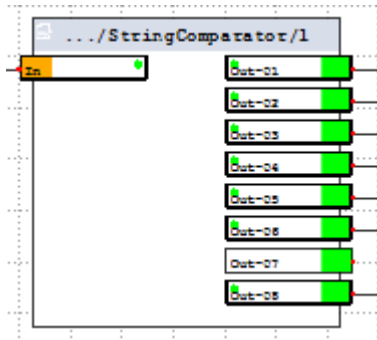
In each case, right-click on the element to defines its Properties and use the **Value** output to connect to other elements.

### 12.4.6 String Comparator

This element compares the input string with up to 8 defined strings and outputs a Boolean value (either true or false) at the corresponding output(s).

Each comparison can work in one of three modes:

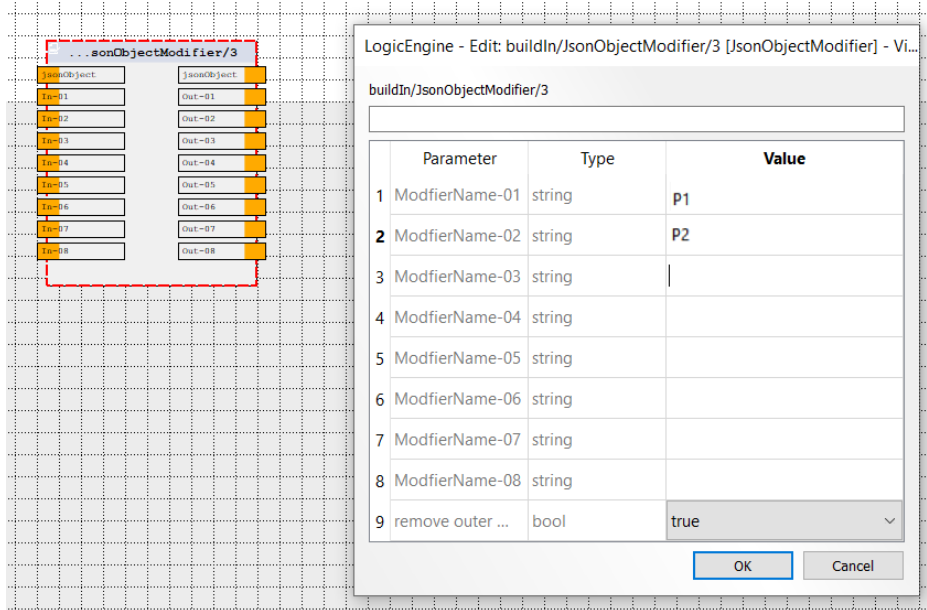
- **Text** - requires an exact match between the input and defined value.
- **Wildcard** - searches for a simple wildcard match between the input and defined value.
- **RegExp** - searches for a match of Regular expressions (explained below).



Select the **RegExp Help** button for an explanation of the different modes.

### 12.4.7 Jjson Object Modifier

This element provides support for the third-party application [LUCI Studio](#). It acts as a JSON object encoder and decoder as follows.



A JSON object string attached to [JJsonObject] is broken down and, if a (string) name is set, the "ModifierName-01" to "ModifierName-08" values are written to OUT-1 to OUT-8. The other way round, a JSON object is written from the inputs IN-01 to IN-08 with the names "ModifierName-01" to "ModifierName-08".

For example, if the JJsonObject input -> OUT-01 to OUT-08, and IN-01 to IN-08 -> JJsonObject output, then the following applies:

- If a "ModifierName" is set, the input and its corresponding output are included in the JSON object.
- If a "ModifierName" is set but there is no string (or an empty one), then the input and its corresponding output are NOT included in the JSON object.

Right-click on the element to define its Properties. The **Type** column indicates the type of **Value** that should be entered:

Parameter	Value
ModifierName-0x	Enter the string to be included in the JSON object.
remove outer brackets	Set this value to <b>false</b> to remove the outer object brackets.

*Example 1*

ModifierName-01 = "P1"

ModifierName-02 = "P2"

IN-01 = "Marius"

IN-02 = "Mustermann"

*Generates the output:*

```
{"P1" : "Marius", "P2" : "Mustermann"}
```

*Example 2*

ModifierName-01 = "P1"

ModifierName-02 = "P2"

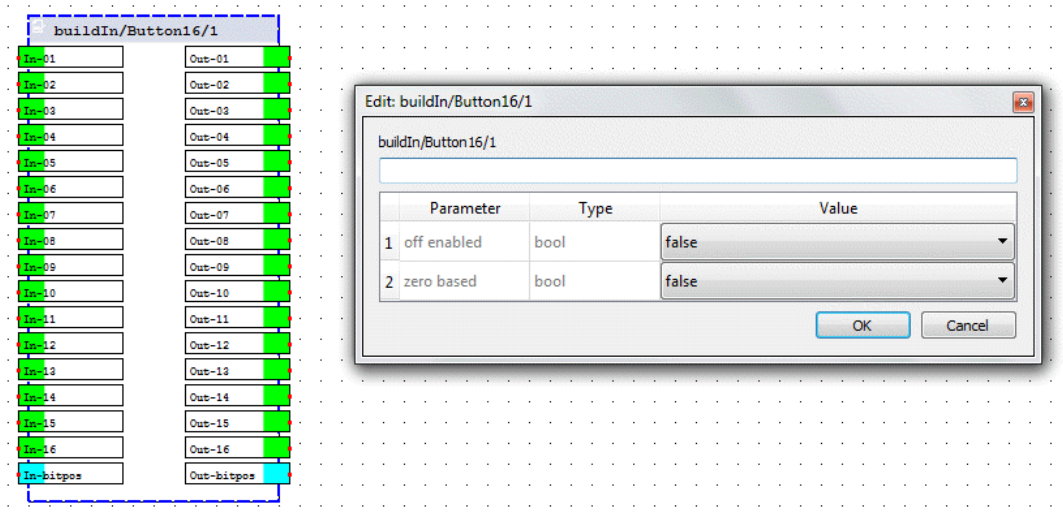
IN-01 = ""

IN-02 = "Mustermann"

*Generates the output:*

```
{"P2" : "Mustermann"}
```

### 12.4.8 Button 16



This element has 16 (Boolean) inputs and outputs (connected 1:1), where only one output can be active at any given time. A rising edge at one of the inputs sets the output of the same number to a logical "1". It remains set until a rising edge is received at another input. When this occurs the first output is reset to "0", and the new output is set to "1".

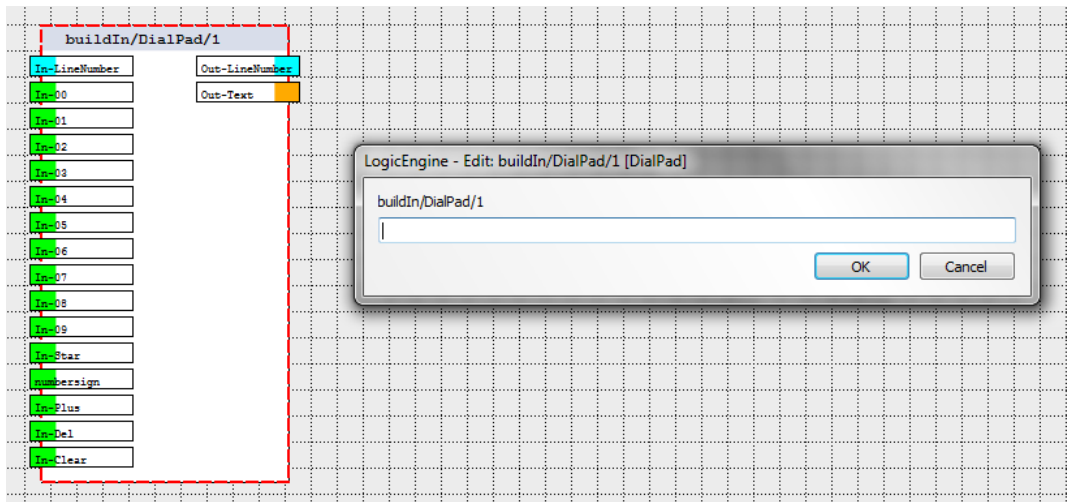
Typically the element can be used to create an intercanceling button or signaling group. The functionality is similar to the "Logic -> Button16" element in the ON-AIR Designer.

The **Out-bitpos** output shows the number of the active output, depending on the **zero based** parameter. The **In-bitpos** integer input may be used to set the output by a number.

Right-click on the element to define its Properties. The **Type** column indicates the type of **Value** that should be entered (bool = Boolean):

Parameter	Value
off enabled	Sets whether outputs can be switched to "off": <b>false</b> = The inputs are inter-canceling. Outputs may not be switched to "off". There is always an active output. <b>true</b> = The active input may be switched to "off" or inter-cancelled by another input. There is no need to have an active output.
zero based	Sets whether the range of values for <b>In-bitpos</b> and <b>Out-bitpos</b> start at 0 or 1 <b>false</b> = The bitpos of the first output is 1, the outputs range from 1 to 16. <b>true</b> = The bitpos of the first output is 0, the outputs range from 0 to 15.

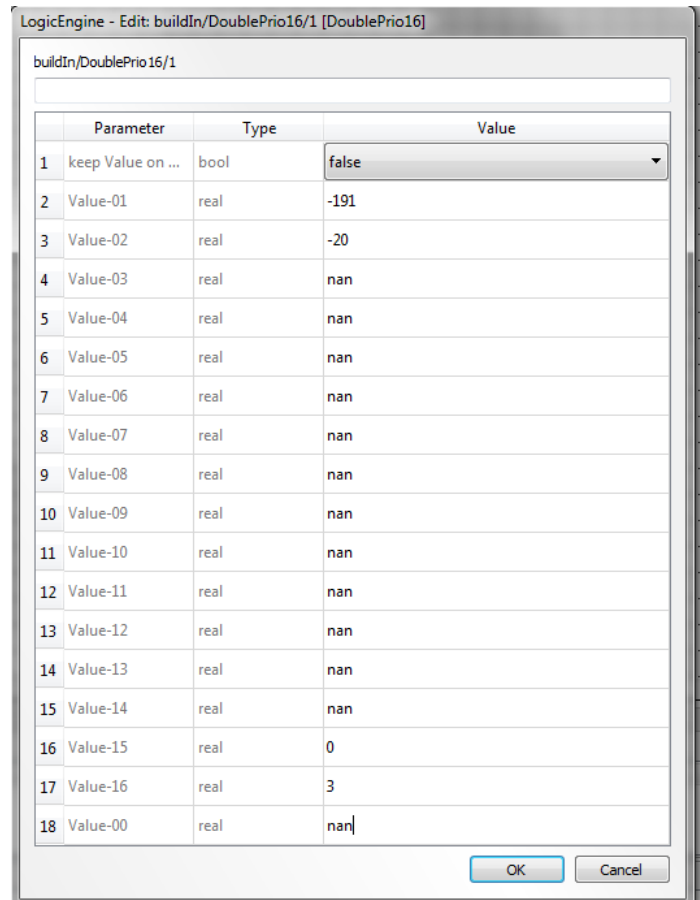
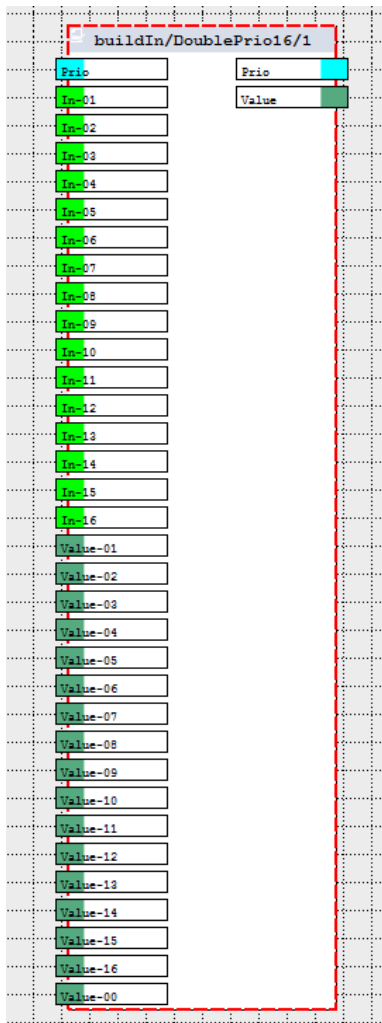
### 12.4.9 DialPad



This element can be used to create a "dial pad" so that VisTool users can make a telephone call from a VisTool page. The inputs should be linked to VisTool button or box elements (to enter the telephone number). The values are then assembled into the **Out-LineNumber** output (to dial the call).

In the Logic Engine, the element has no special Properties, other than to name the element.

## 12.4.10 DoublePrio 16



This element can issue different values (**Value-00** to **-16**) to a single output (**Value**). The switching is controlled by the corresponding logical inputs (**In 01** to **16**), where the highest index has the highest priority.

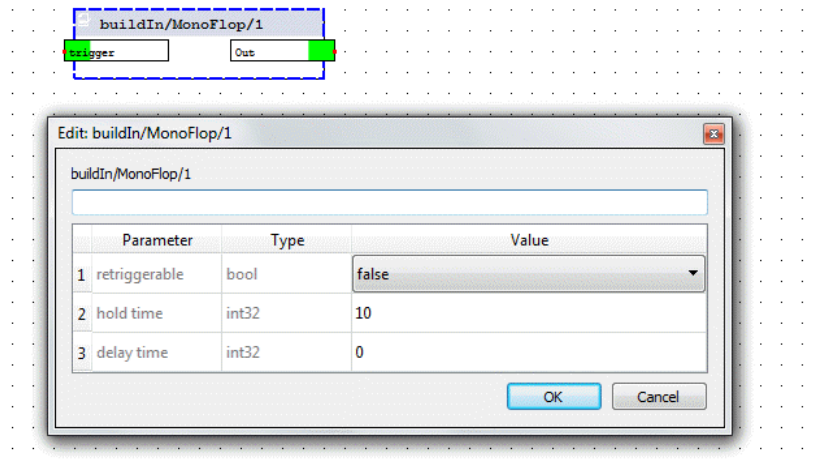
In addition, there is a **Prio** input (integer), which can also change the output **Value** - an integer value of 0 to 16 is equivalent to a **true** signal on the corresponding input.

The **Prio** output (integer) can be used to cascade multiple elements (e.g. Prio, TextPrio, DoublePrio, Button16).

Parameter	Value
keep Value on prio 0	Defines whether the last output value is retained when all inputs are inactive. <b>true</b> or <b>false</b> .
Value-01 to Value-16	Enter the value corresponding to each logical input.
Value-00	Enter the default value to be used when no other value is active.



### 12.4.11 MonoFlop

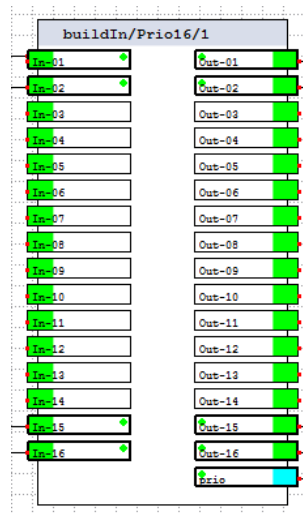


This element can be used to delay the incoming parameter value or to create a pulse of a defined length from a static signal.

Right-click on the element to define its Properties. The **Type** column indicates the type of **Value** that should be entered: bool = Boolean; int32 = an integer.

Parameter	Value
retriggerable	= <b>true</b> - the hold time counter is reset by an active <b>retrigger</b> input during the hold time. = <b>false</b> - the hold time counter cannot be reset.
hold time	Enter the hold time (in multiples of 10ms - i.e. 10 = 100ms). The Output is held by the entered value.
delay time	Enter the delay time (in multiples of 10ms). The Output is delayed by the entered value.

## 12.4.12 Prio 16



This element is similar to the [Button 16](#) but with prioritised switching.

It has 16 (Boolean) inputs and outputs (connected 1:1), where only one output can be active at any given time, and the highest index has the highest priority.

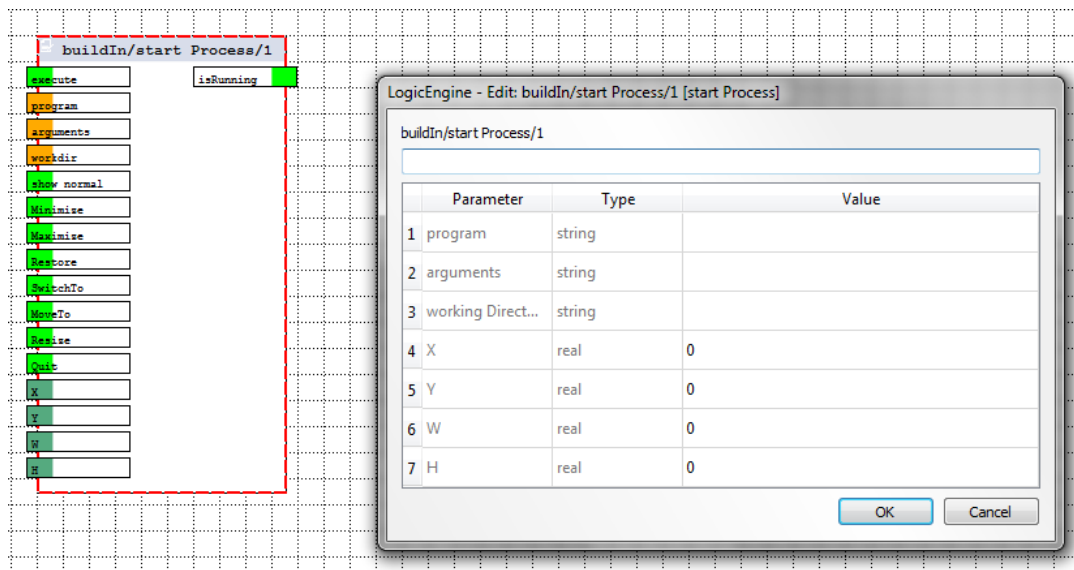
For example: if **In 1** is set to a logical "1", and all other inputs set to "0", then **Out 1** is active and all other outputs are set to "0". If **In 3** then becomes set to a logical "1", **Out 3** becomes active and **Out 1** is set back to logical "0". If **In 1** and **In 3** are both set to logical "1", then **In 3** has higher priority so **Out 3** is the active output.

Typically the element can be used to create an intercanceling button or signaling group. The functionality is similar to the "Logic -> Prio" element in the ON-AIR Designer.

It has no special Properties, other than to name the element.

The **Prio** output (integer) can be used to cascade multiple elements (e.g. Prio, TextPrio, DoublePrio).

## 12.4.13 start Process



This element can be used to start any application from VisTool - for example, to start up R3LAY, Skype, etc.

Programs can be started, stopped and their screen size adapted. Note that, depending on the program, it may not be possible to change the screen size or interact from the keyboard. However, normal programs should work as expected.

Use the **Execute** input to start the program. The program (location and) name can be entered dynamically via the orange text inputs (**program**, **arguments** and **workdir**), OR statically via the corresponding Parameter fields in the 'Properties' window (see below). The **isRunning** output is active when the program has been started at least once.

The following inputs adjust the (running) program's appearance:

- **Show normal**: the window of the program is displayed at its original size.
- **Minimize**: the window of the program is minimized.
- **Maximize**: the window of the program is maximized.
- **Restore**: the window of the program is displayed at its original size and at the original position.
- **Switch to**: the program is displayed in the foreground (in front of any other running programs).
- **Move to**: moves the program window to the position defined under **X** and **Y**.
- **Resize**: scales the program window to the size defined under **W** and **H**.

The **Quit** input can be used to quit the (running) program.

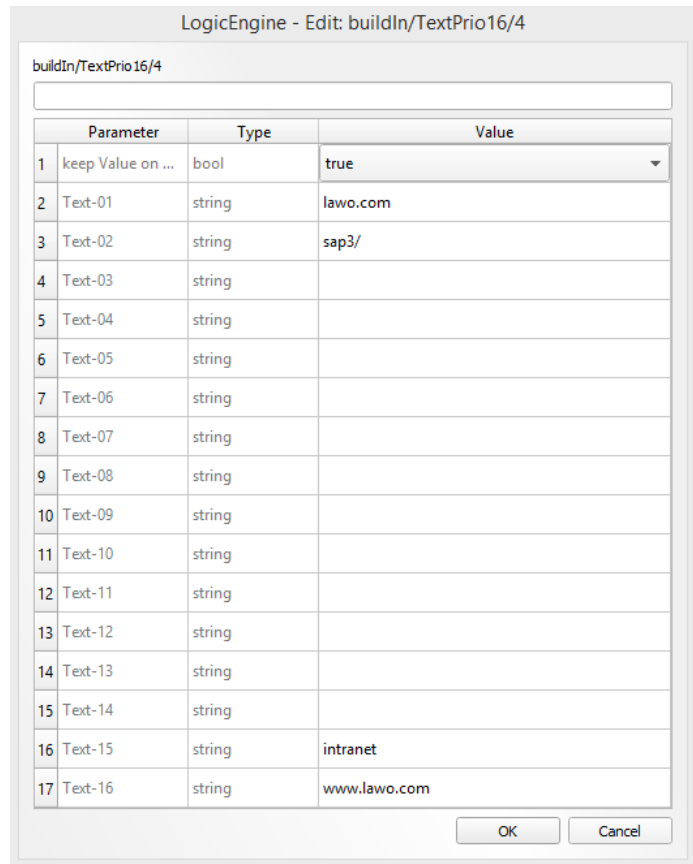
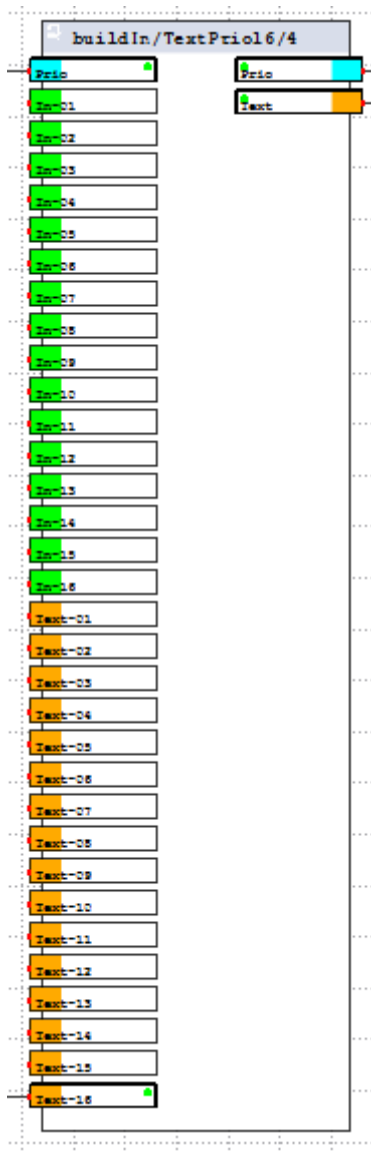
Note that the screen size inputs do not work if several instances of the program are started. To avoid conflicts, you should use logic to prevent multiple instances being started.

Right-click on the element to define its Properties. The **Type** column indicates the type of **Value** that should be entered:

Parameter	Value
program	Enter the location and name of the executable file as a string. Alternatively, this can be defined dynamically using the <b>program</b> text input.
arguments	Enter any start up parameters as a string. Alternatively, this can be defined dynamically using the <b>arguments</b> text input.
working Directory	Enter the working directory for the program as a string. Alternatively, this can be defined dynamically using the <b>workdir</b> text input.

X & Y	Enter the <b>X</b> and <b>Y</b> coordinates to be used when the <b>Move to</b> input is triggered. Alternatively, these can be defined dynamically using the <b>X &amp; Y</b> inputs.
W & H	Enter the <b>Width</b> and <b>Height</b> coordinates to be used when the <b>Resize</b> input is triggered. Alternatively, these can be defined dynamically using the <b>W &amp; H</b> inputs.

12.4.14 TextPrio 16



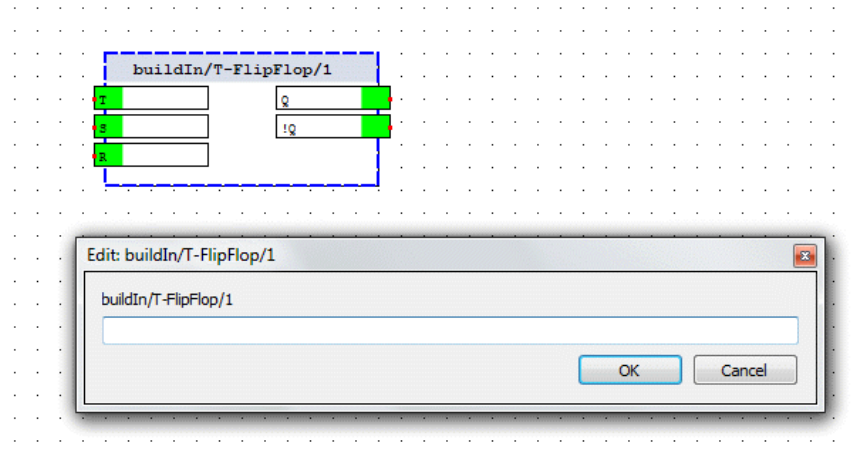
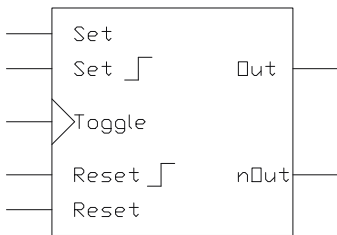
This element can issue up to 16 different strings (**Text-01** to **-16**) to a single output (**Text**). The switching is controlled by the logical inputs (**In 01** to **16**), where the highest index has the highest priority. The functionality is similar to the "Logic -> Text Prio" element in the ON-AIR Designer.

In addition, there is a **Prio** input (integer), which can also change the output **Value** - an integer value of 0 to 16 is equivalent to a **true** signal on the corresponding input.

The **Prio** output (integer) can be used to cascade multiple elements (e.g. Prio, TextPrio, DoublePrio, Button16).

Parameter	Value
keep Value on prio 0	Defines whether the last output value is retained when all inputs are inactive: <b>true</b> or <b>false</b> .
Text-01 to Text-16	Enter the string corresponding to each logical input. Any string can be entered: text or numbers.

### 12.4.15 Toggle Flip Flop



This element can change its output state in response to an input trigger. A typical application is to create a latching function.

The inputs can be connected as follows:

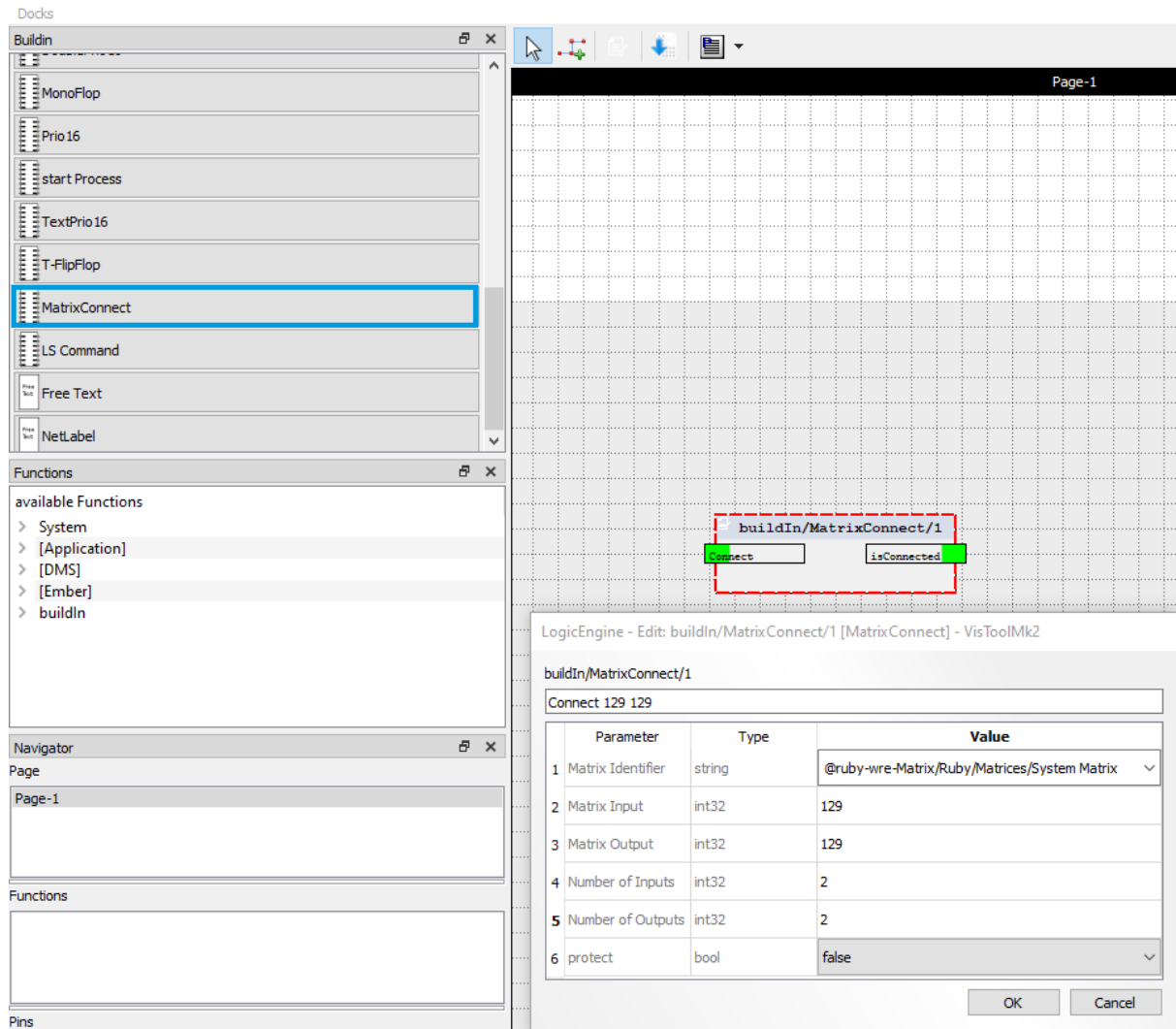
- **T** (Toggle) - an active Toggle input will invert the state of the output.
- **S** (Set) - an active input here will set the output to "1".
- **R** (Reset) - an active input here will reset the output to "0".

Each toggle flip flop supports two outputs:

- **Q** is the output.
- **!Q** is an inverted output.

The functionality is similar to the "Logic -> TFF" element in the ON-AIR Designer. It has no special Properties, other than to name the element.

## 12.4.16 Matrix Connect



This element can be used to control matrix crosspoints within an external device. The functionality is similar to the "Connect -> Matrix Multi Connect" element in the ON-AIR Designer.

The external matrix must first be added to the VisTool project. See [Configuring an External Matrix](#). A **Matrix Connect** element can then be added and configured.

The element can connect single or multiple crosspoints. For the input(s) and output(s), you must specify the first matrix address and the number of channels. The connections are then made sequentially from the first address upwards. If there is a mismatch between the channel counts, the connections are made on a best-effort basis. For example:

- If a 2-channel source is connected to a 6-channel target, the source channels are repeated: Target receives source channels 1, 2, 1, 2, 1, 2.
- If a 6-channel source is connected to a 4-channel target, the first available source channels are used: Target receives source channels 1, 2, 3, 4.

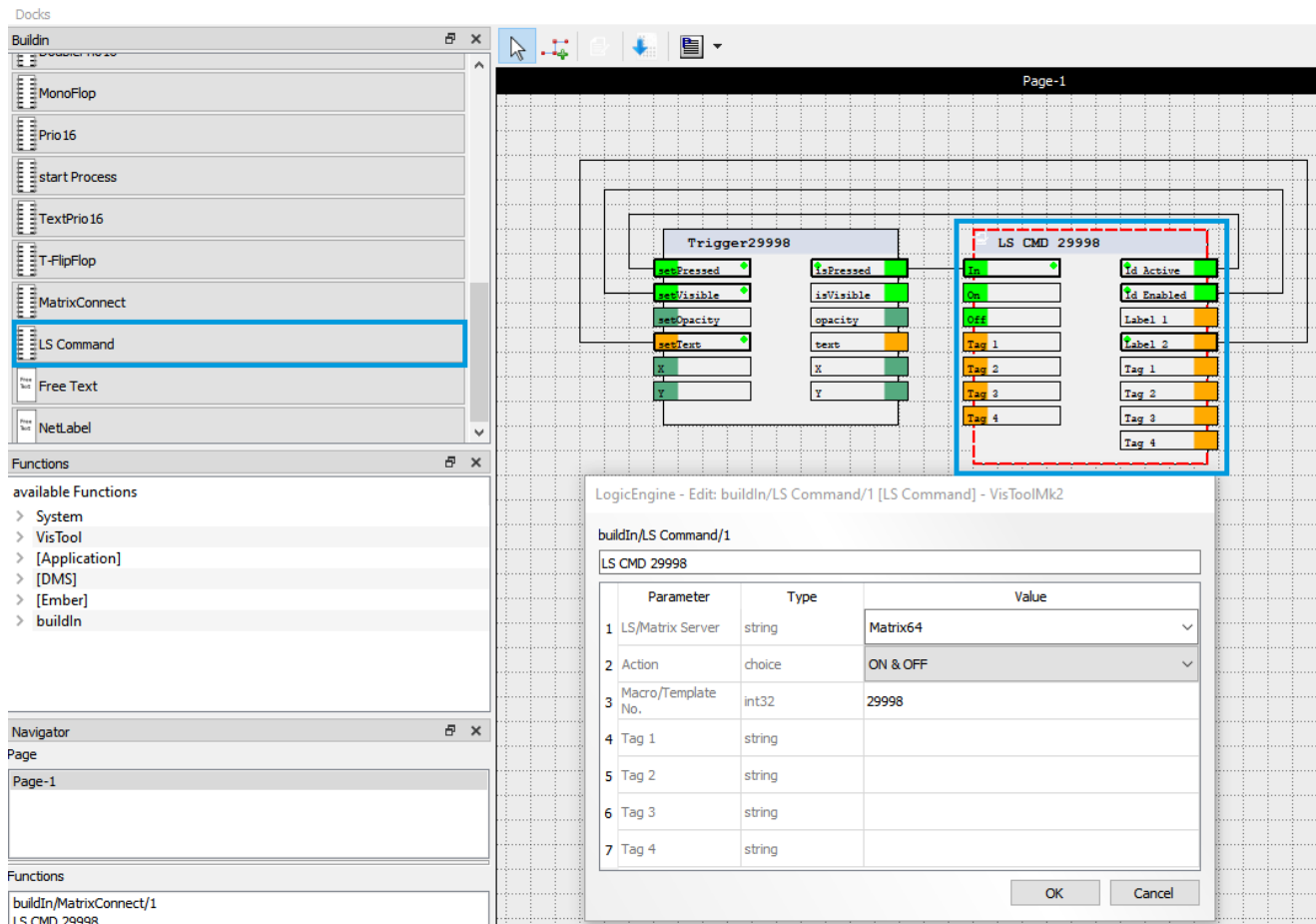
The **Connect** input can be used to trigger the matrix connections from another logical element. The **isConnected** output becomes true once the connections are active.

Right-click on the element to define its Properties. The **Type** column indicates the type of **Value** that should be entered:

Parameter	Value
Matrix Identifier	Select the matrix to be controlled. The drop-down menu lists all of the matrices added to the VisTool project (from the <a href="#">Ember+ Provider Tree</a> or <a href="#">DSA Matrix devices</a> ).
Matrix Input	Enter the matrix address number of the first input (see <a href="#">Matrix Numbers</a> ).
Matrix Output	Enter the matrix address number of the first output (see <a href="#">Matrix Numbers</a> ).
Number of Inputs	Enter the number of inputs to be connected.
Number of Outputs	Enter the number of outputs to be connected.
protect	Set this option to <b>true</b> to protect the connected outputs (targets). The protection can only be implemented if this feature is supported by the external matrix.



## 12.4.17 LS Command



This element can be used to change and monitor Triggers on a Line Scheduler (LS). The functionality is similar to the "Logic -> LS Command" element in the ON-AIR Designer.

Vistool establishes a direct connection to a matrix server for this purpose. The external matrix must first be added to the VisTool project. See [Configuring an External Matrix](#). An **LS Command** element can then be added and configured.

The control inputs can be connected as follows:

- **In** - set this input to true to perform the configured **Action**.
- **On** (or **Off**) - set these inputs to true to activate (or deactivate) the Macro / Template.
- **Tag 1** to **4** - use these inputs to dynamically set the days.

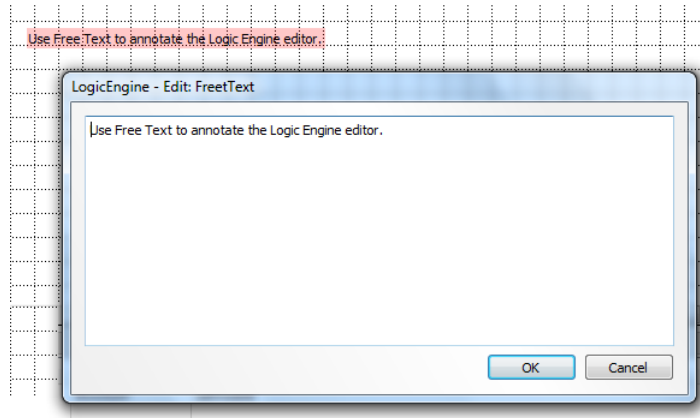
The control outputs provide feedback from the Line Scheduler:

- **ID Active** - becomes true when the Trigger (Macro / Template) is active.
- **ID Enabled** - becomes true when the Trigger (Macro / Template) is configured.
- **Label 1** & **Label 2** - outputs the LS button label.
- **Tag 1** to **4** - outputs the active days.

Right-click on the element to define its Properties. The **Type** column indicates the type of **Value** that should be entered:

Parameter	Value
LS Matrix Server	Select the matrix server to be used for the connection. The drop-down menu lists all of the matrices added to the VisTool project (from the <a href="#">Ember+ Provider Tree</a> or <a href="#">DSA Matrix devices</a> ).
Action	Select the action that will happen when the <b>In</b> control input becomes true.
Macro / Template No.	Enter the Line Scheduler's trigger number for the Macro / Template (usually found under Administration / Trigger names).
Day 1 to 4	Enter the strings to control the Merge Show function.

### 12.4.18 Free Text



This element can be used to add notes to the Logic Engine design. Click and drag to place the **Free Text** element. Then double-click to open its parameter window - the text entered inside the window appears on your design. The Free Text element is highlighted in pink when selected.

### 12.4.19 Net Labels

Net labels can be used to simplify the connections in the Logic Engine editor window. They can also be used to link the logic between pages.

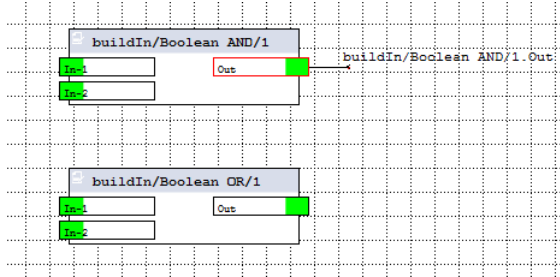
Instead of connecting an output to an input directly, as described [earlier](#), an intermediate Net Label node can be used. In the example below, a Net Label named **AND 1** has been created. This is connected from the **Out** of the Boolean AND/1, and to the **In-1** of the Boolean OR/2 and **In-2** of the Boolean OR/3.

Using the Net Label makes it easy to interrogate the connections and understand the design:

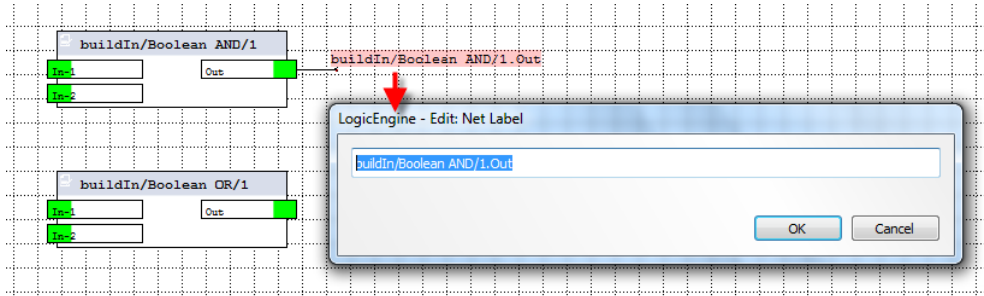
Net	Type	Page	Element	Pin	Page	Element	Pin
Net:12480	bool	-	@mic8/AudioD...	Out	-	Box:0006	setPressed
Net:13069	bool	-	Box:0006	isPressed	-	buildIn/T-FlipFL...	T
Net:13074	bool	-	Box:0007	isPressed	-	buildIn/T-FlipFL...	T
Net:13089	bool	-	buildIn/T-FlipFL...	Q	-	@mic8/AudioD...	In
Net:36423	bool	-	@Sapphire242/...	Out	-	Box:0007	setPressed
Net:106932	bool	-	buildIn/T-FlipFL...	Q	-	@Sapphire242/...	In
AND 1	bool	-	buildIn/Boolean AND/1	Out	-	buildIn/Boolea...	In-1
						buildIn/Boolea...	In-2
buildIn/Boolean OR/2.Out	bool	-	buildIn/Boolea...	Out	-	buildIn/Boolea...	In

The simplest way to create a Net Label is as follows:

1. Select the output you wish to connect from and press **CTRL + L** (keyboard shortcut). This creates an automatically named Net Label element - in our example, from the **Out** of Boolean AND/1:

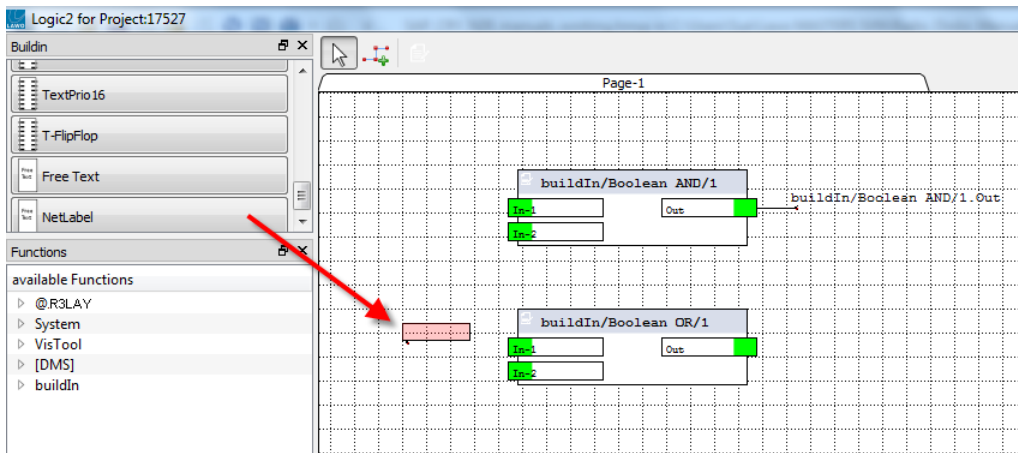


2. Double-click on the Net Label text to open its parameter window, and copy the Net label name:

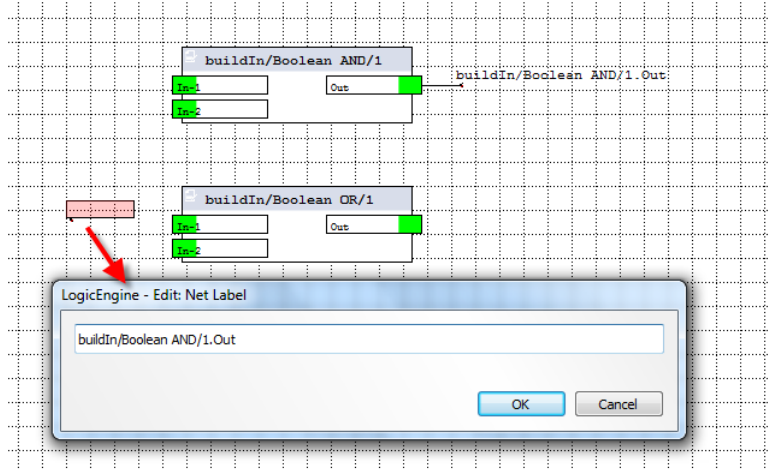


If you wish to use the Net Label on a different page, then make sure you add a "\$" to the front of the name.

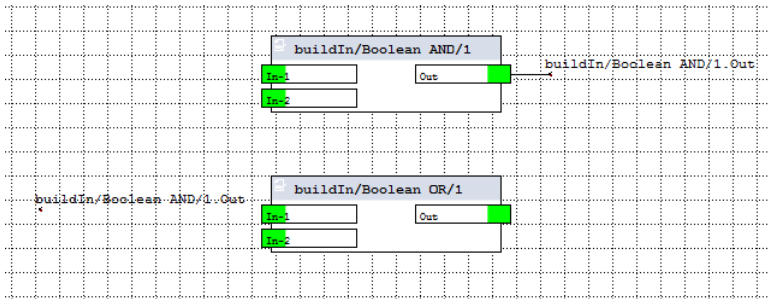
3. Click **OK**, and then manually position a new Net Label before the input - in our example, the **In-1** of Boolean OR/1:



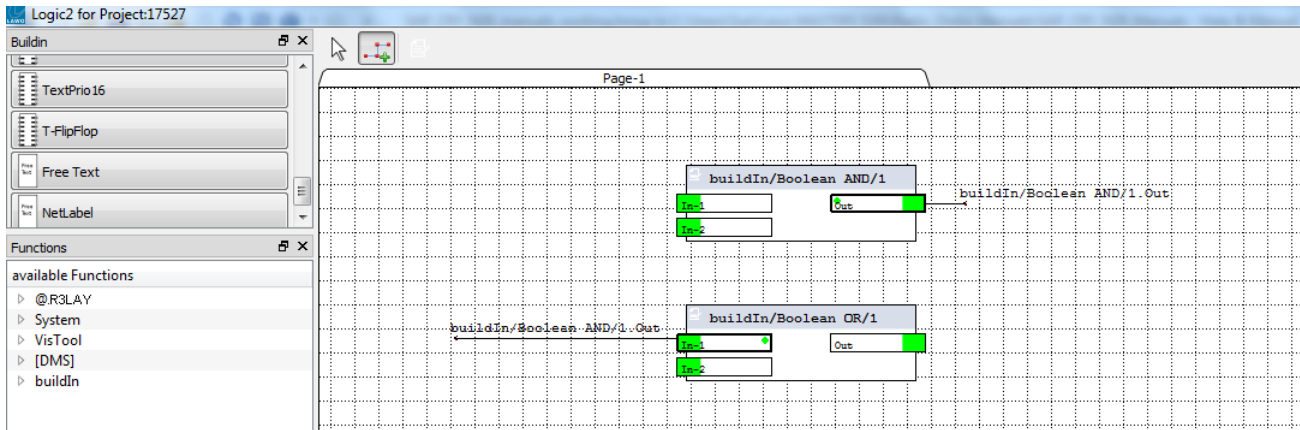
- Double-click to open its parameter window, and paste in the name (copied in step 2) into the Label field:



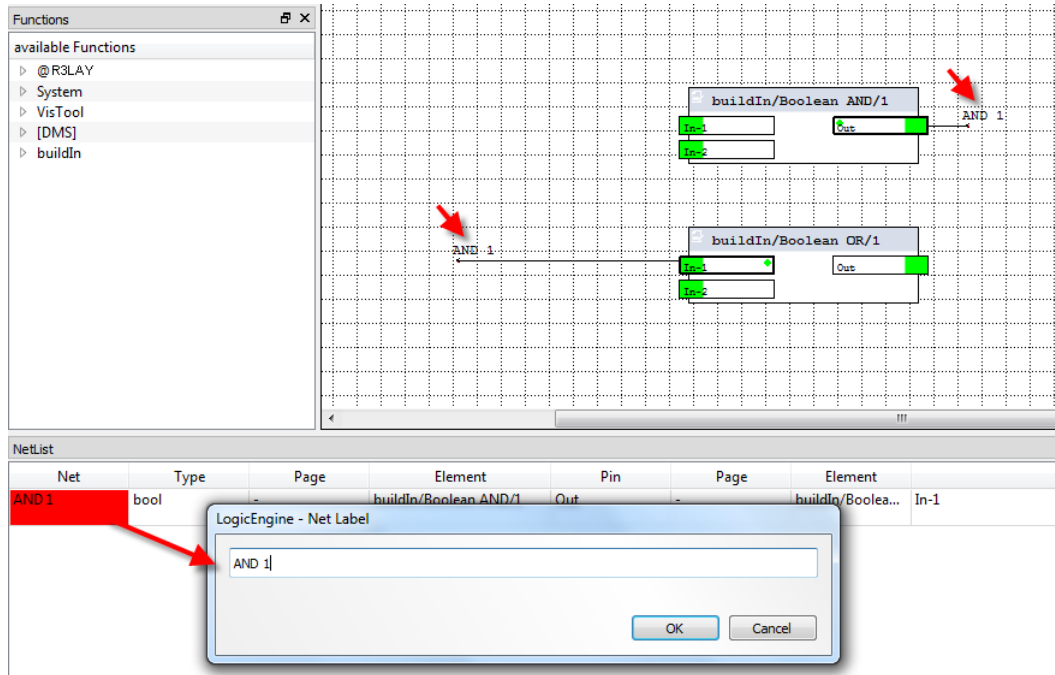
- Click **OK** to confirm - you should now have an identically named Net Label element which appears in two different places:



- Now connect the Net Label to the input(s) of your logical element(s) - in our example, to the **In-1** of the Boolean OR/1 - the "through" connection is made:



7. If you wish, you can rename the Net Label element from the Navigator - in our example, to **AND 1**. Click **OK** to confirm. The Net Label name is automatically updated throughout the Logic Engine design:



### Tips

When making connections, position your Net Label element on the same line as the input (or output).

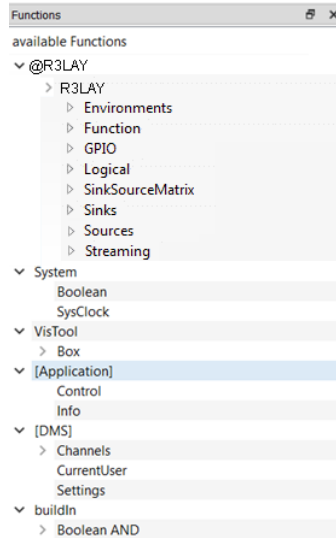
You can use **CTRL + L** to create an automatically named Net Label for the selected output.

Do **NOT** rename a Net Label element positioned at the input or output, unless you wish to break its logical connections. Instead, always use the Navigator (as described in step 7). This method will update all occurrences of the Net Label and keep all existing connections intact.

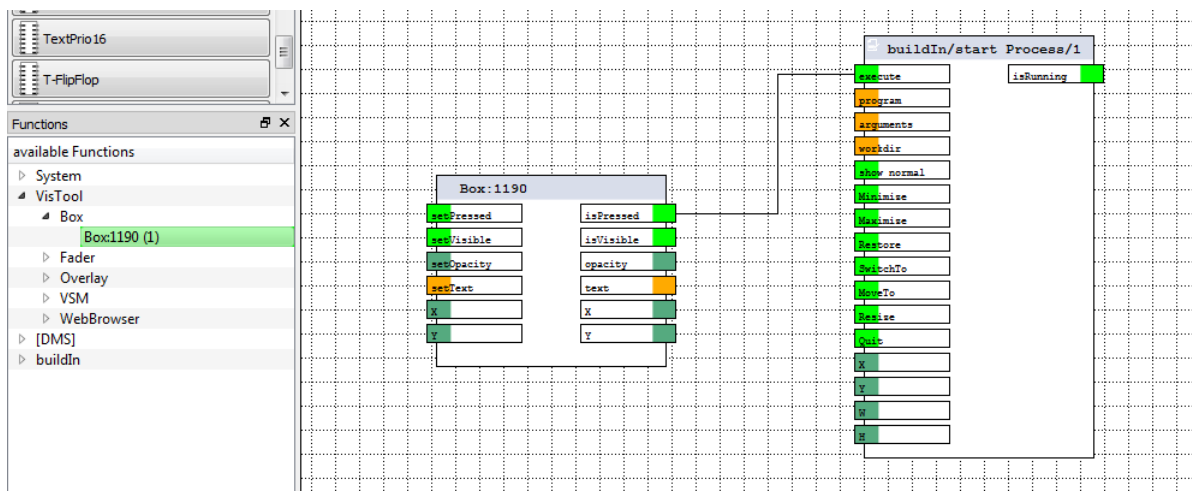
You can use Net Label elements to scatter a logical signal flow across multiple pages. To use a Net Label on a different page, you must add a "\$" to the front of the name.

## 12.5 Functions

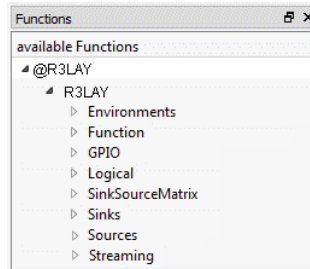
The **Functions** docking panel contains all of the available functions.



Any function can be added to a page and connected to other logical elements as shown in the example below:



### 12.5.1 @xxx



The @xxx branch includes the Ember+ provider parameter values.

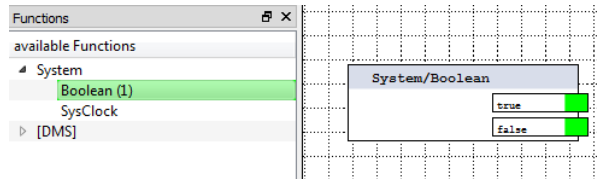
If no Ember+ providers have been assigned to the Logic Engine, then you will not see this branch. A branch appears for each Ember+ provider once it has been assigned (via drag and drop) to the project. The functions and their logical inputs and outputs vary depending on the provider.

### 12.5.2 System

The **System** branch includes the boolean "true" and "false" states (**Boolean**) and the network's system clock (**SysClock**).

#### Boolean

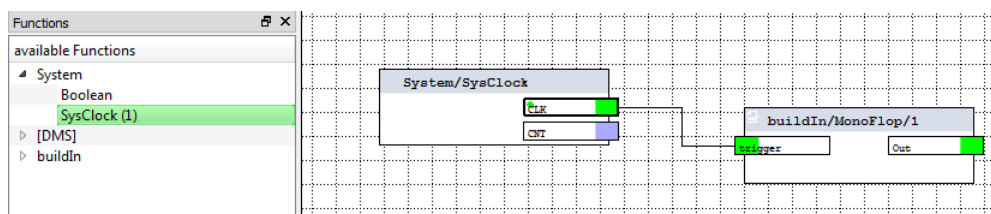
You can use the **Boolean** outputs whenever you need to trigger a "true" or "false" state.



<b>true</b>	"true" output	Boolean
<b>false</b>	"false" output	Boolean

#### SysClock

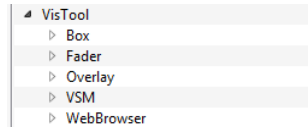
You can use the **SysClock** in combination with a [MonoFlop](#) to create recurring edges:



<b>CLK</b>	System Clock output	Boolean
<b>CNT</b>	Counts how often the CLK pin has been activated since the last restart of the Logic Engine. It will increase by one every 10 ms.	Value



## 12.5.3 VisTool



The **VisTool** branch includes VisTool elements which have been [added](#) to the Logic Engine editor.

If no elements have been added, then you will not see the **VisTool** branch. Once elements have been added, they appear in the corresponding sub menus. The logical inputs and outputs vary depending on the element.

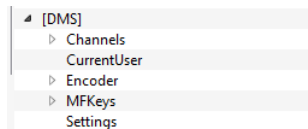
## 12.5.4 [Application]



The **[Application]** branch includes elements that allow you to control, or show information about, the VisTool application.

- **Control** - use the inputs to this element to minimize or restore a VisTool instance.
- **Info** - use the outputs from this element to show if a VisTool instance is minimized and its active page.

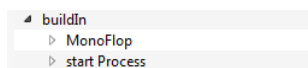
## 12.5.5 [DMS]



The **[DMS]** branch includes elements from the "DMS" system (defined by the ON-AIR Designer project specified in VisTool Editor's [Project Properties](#)).

The elements are divided into five sub menus: **Channels**, **CurrentUser**, **Encoder**, **MFKeys** and **Settings**. The logical inputs and outputs vary depending on the element.

## 12.5.6 buildIn



The **buildIn** branch contains all [Logical Elements](#) which have already been added to the design.

## 13. The Visual Extension

---

This chapter describes how to install, configure and operate the Visual Extension.

Topics include:

- [Introduction](#)
- [The Hardware](#)
- [Wiring](#)
- [Configuration](#)
- [Using the Standard Template](#)
- [Using a Custom Configuration](#)

## 13.1 Introduction

Lawo's Visual Extension is available as an integrated option on sapphire consoles or as a stand-alone unit. It houses a series of 7" touch-screen displays:

*sapphire with Visual Extension Overbridge*



*Stand-alone Visual Extension*



The displays show VisTool GUI pages, and can be operated directly from the screens using their touch-screen functionality.

Note that the VisTool MK2 software must run on an external PC which is not included and must be provided by the customer. If more than 6 screens are fitted, then you will need two VisTool PCs (as a single PC can support up to 6 runtime instances).

The exact functionality depends on the VisTool project. If you have a **VisTool Standard** license, then you can run the standard template delivered with your system. If you have a **VisTool Unlimited** license, then the GUI pages can be customized by editing project using VisTool Editor.

## 13.2 Hardware

*Visual Extension with 4 Screens*



### General Notes

The Visual Extension touch-screens are housed in "frames", and a single frame may contain up to 6 screens. Frames may be placed next to each other and handled as "one unit" to create larger Visual Extensions.

To drive the screen displays, you will need an external PC running VisTool MK2. Each PC can support up to 6 screens. You may install a second PC to drive more screens.

Each Visual Extension frame requires three connections: power, CAN bus (to the System Unit) and USB (to the VisTool PC). Please make sure that the connectors on the rear are accessible!

Special care should be taken when carrying frames. The Visual Extension is designed with flexibility in mind, so it is perfectly normal for it to bend a little when lifting! Nonetheless, you should handle it carefully.

### Dimensions and Weight

The dimensions and weight of each frame depend on the specification. Please refer to the relevant data sheet for details.

### Air Conditioning

The Visual Extension frame is designed for normal studio installation and needs no special air conditioning.

### Grounding

A grounding screw is provided beside the CAN bus connector to ground the frame. Use a M4x8 screw only.

### Power

Each Visual Extension frame is fitted with a DC power connector and delivered with its own power supply unit (PSU). Alternatively, you may connect power from the Redundant Power System (951/25). For power consumption and electrical voltage details, please refer to the PSU datasheet. Note that the maximum power consumption must be considered when using the redundant PSU system.

The PSU *MUST* be connected to the mains using the adapters supplied with the system.

## 13.3 Wiring

From each Visual Extension frame there are three connections: power, CAN bus and USB. In addition, the VisTool PC must connect to the Lawo control system network via Ethernet. Note that the order in which you power the VisTool PC and its Visual Extension is important. Please see the [Installation Procedure](#).

### Power

Independent DC power connectors are provided from each Visual Extension frame. These can connect either to the PSU delivered with each frame, or to the optional Redundant Power System (951/25).

### CAN Bus

The CAN bus transports the touch-screen data and connects to the sapphire Master Board, along with the other control surface frames and Key panels in a daisy chain fashion. In our sapphire system shown below, there are two Master Boards, and therefore the external CAN terminator is not required. For systems with one Master Board, the CAN bus must be terminated at the last device.

### USB

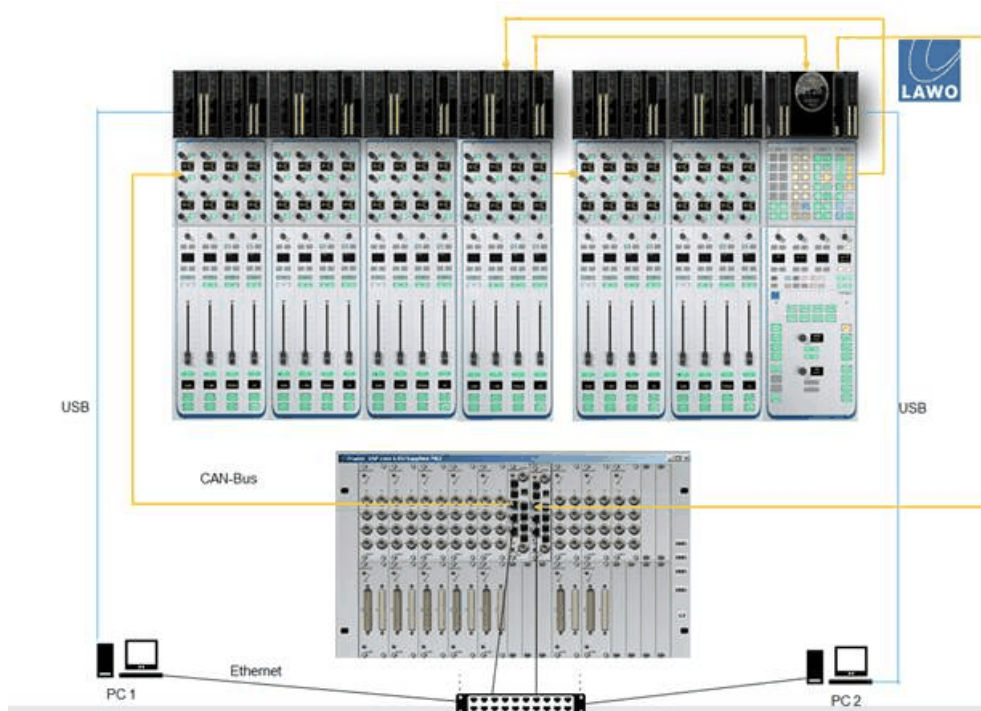
The USB connection transports the graphical data and connects to the the VisTool PC. The USB through outlet can be used to connect a second Visual Extension frame.

### Ethernet

The VisTool PC must connect to the Lawo control system network via Ethernet, see [Connecting to the Lawo Control System](#).

### Wiring Diagram

The example below shows the connections required for a multi-frame sapphire fitted with two Master Boards. Two PCs are used to drive the Visual Extension screens. Note that the power connections are not shown.



## 13.4 Configuration

Each VisTool PC must be installed with VisTool MK2 and configured for runtime operation. While these procedures are similar to a standard installation, there are some additional steps to correct the screen orientation (in Windows) and set the touch-screen IDs (in VisAdmin). The complete procedure is described below.

Note that you will need to connect an external monitor to the PC to configure the system (via DVI or VGA).

The steps below can be used for a Visual Extension with 6 screens or less. If the Visual Extension has more than 6 screens, then please also read the [Additional Information](#).

1. Check that the VisTool PC meets the [hardware requirements](#) (for "In-Studio VX-Quality Screen Designs").
2. Connect the PC to the Lawo control system, see [Network Connection](#).
3. Connect the PC to the Visual Extension via USB, and make the other system connections, see [Wiring](#).

**BEFORE** you power up the VisTool PC, make sure that the Visual Extension is powered and connected to both the VisTool PC (via USB) and to the sapphire Master Board (via CAN Bus).

4. Power up the PC and install the "DisplayLink" driver supplied with the system.

To do this, double-click on the **DisplayLink.exe** installer and run the installation, accepting the default settings. After the driver installation is complete, the Visual Extension screens should light up. If not, restart the PC.



When you first power on the PC, Windows "sees" the screens in the wrong order and orientation. You should correct this **BEFORE** proceeding any further (in step 5).

**DO NOT** attempt to physically rotate or reorder the screens! This is unnecessary, as the screens can be easily corrected from the Windows Control Panel.

5. Using the Windows Control Panel, correct the orientation of each screen and its addressing.

The steps are described in more detail [later](#). The screens must be "turned" and addressed from left to right in the correct order.

6. Using the Windows Control Panel, disable any USB energy saving options.

This is described in more detail [later](#). The USB energy saving options must be disabled before launching VisTool runtime. Otherwise Windows may "shut down" the USB communication between the Visual Extension and VisTool PC during runtime.

7. Install the VisTool MK2 software as described [earlier](#).
8. Activate the appropriate [software license](#) for your VisTool configuration.

You will need a **VisTool Unlimited** license to run a custom configuration.

9. Configure the DMS services using VisAdmin.

This is done via the [DMS Administration](#) window. You should enter the **IP-Address** for each Lawo system you wish to connect to. The default IP address is **192.168.101.240**.

10. Configure the touch-screen IDs using VisAdmin.

The steps are described in more detail [later](#). During the setup, you are prompted to touch each screen so that the system can "learn" the touch IDs.

11. Generate a start script from your configuration data using [VisTool Sessions](#).

You must configure a "session" for each Visual Extension screen connected to the VisTool PC. Then generate a start script - a link to the start script appears on your computer desktop.

12. Double-click on the start script to [launch](#) VisTool runtime on the Visual Extension screens.
13. Check the [User access](#) in Windows.

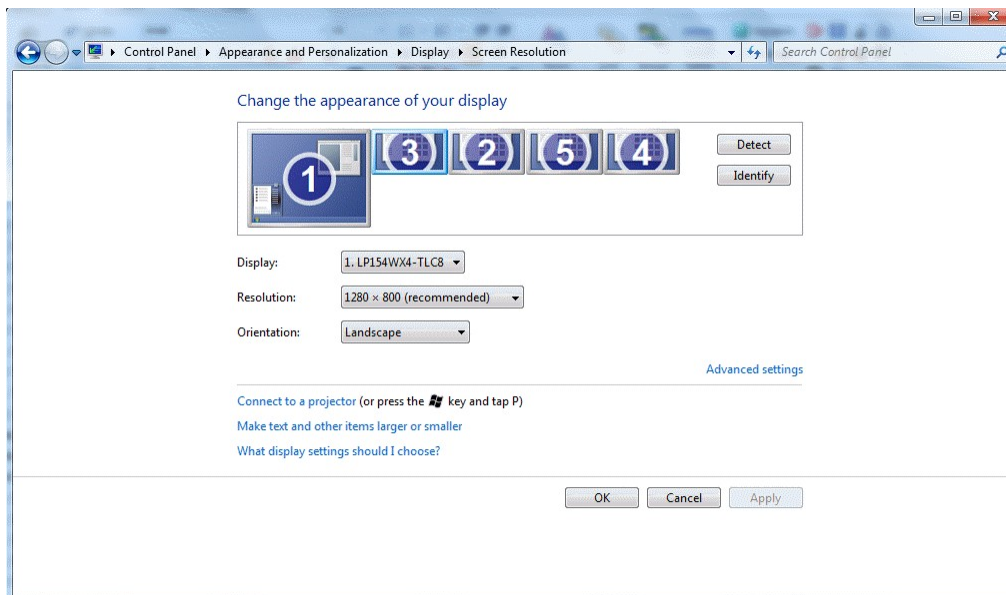
The next topics deal with steps 5, 6 and 10 in more detail.

If you have any problems, please see [Troubleshooting the Visual Extension](#).

### 13.4.1 Correcting the Screen Orientation & Order

1. In Windows, open the "Control Panel -> Display -> Screen Resolution".

Providing the Visual Extension is powered and connected, you should see a representation of the connected screens - in our example, we have a main monitor (Monitor 1) plus a Visual Extension with 4 screens (Monitors 2 to 5):



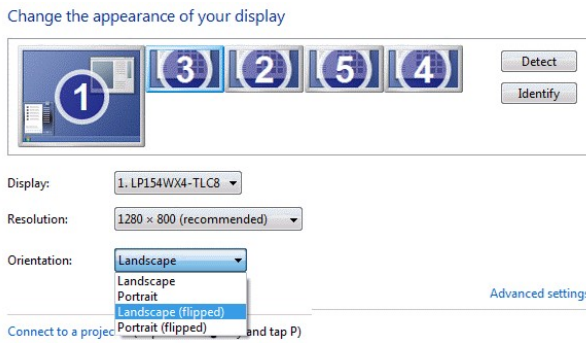
2. Click on **Identify** - an ident appears on each screen showing the current orientation.

If the idents are upside down, then the screen orientation needs correcting!



- To correct the orientation, select a screen and then select **Landscape (flipped)** from the **Orientation** menu. Click **Apply** to confirm.
- Then click on **Identify** again to check the new orientation - the screen ident should now be the right way up:

*Windows Control Panel - change orientation*



*Visual Extension - screen 3 corrected*



- Repeat steps 3 and 4 to correct all of the screens.
- Next, check the screen order.

The order is not important for normal touch-screen operation. However, if you wish to use your PC's mouse with the Visual Extension, and move it across the screens from left to right, you will need to set the correct order.

Note that it is the order from left to right, and not the numbering, which is important. i.e. it does not matter that, in our example, the screens are numbered 3, then 2, then 5, then 4, as long as the order in the Windows Control Panel matches the actual layout. The non-sequential numbering is a Windows anomaly which cannot be altered!

To adjust the order, click and drag a screen to its correct position, and click **Apply** to confirm any changes. The pictures below show that a correct setup - the screen ident's are correctly orientated and the order on the Visual Extension matches that in the Windows Control Panel:

*Windows Control Panel*



*Visual Extension*



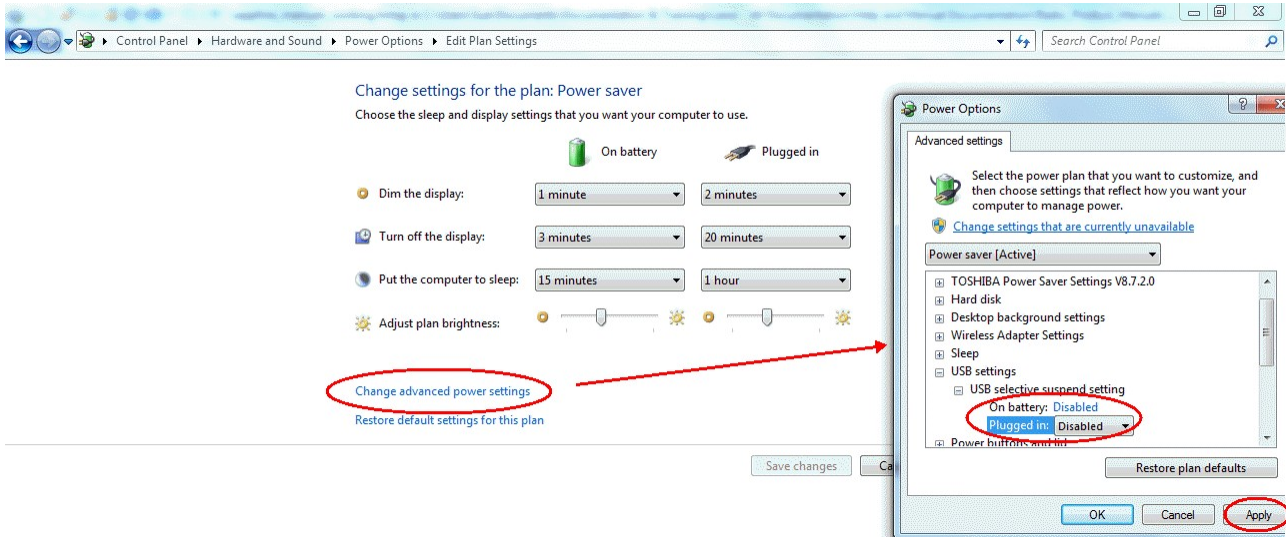
- Once the setup is complete, you can close the Windows Control Panel.



### 13.4.2 Disabling the USB Energy Saving Option

1. In Windows, open the "Control Panel -> Power Options", and select your power plan.
2. Select "Change advanced power settings" to open the 'Power Options' window.

Under "USB settings", select **Disabled** for all available power options. Then click **Apply**:



### 13.4.3 Setting Up the Touch-Screen IDs

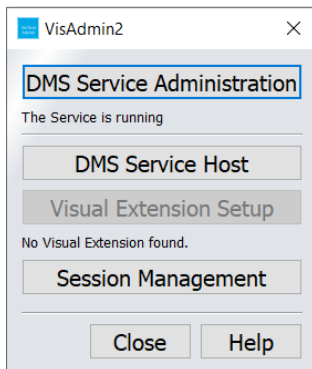
Each touch-screen within the Visual Extension frame must be assigned to the VisTool PC by the **Visual Extension Setup** window.

If your VisTool PC has other monitors connected with touch-screen capability, then these must be configured by Windows or the manufacturer's software.

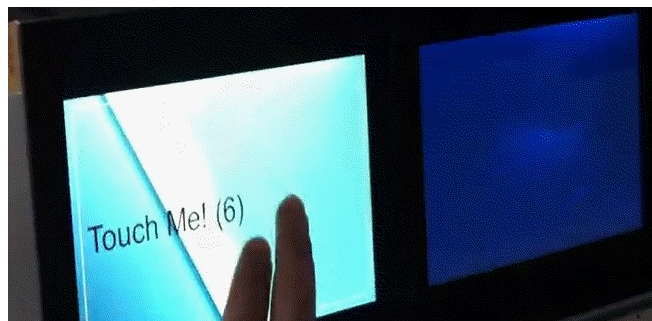
Before you begin, make sure that the screen orientation and order is correct (as described [earlier](#)). Then continue as follows.

1. Start by selecting the **Visual Extension Setup** option from the **VisAdmin** home page (described [earlier](#)) - the "Touchscreen Setup" window appears.

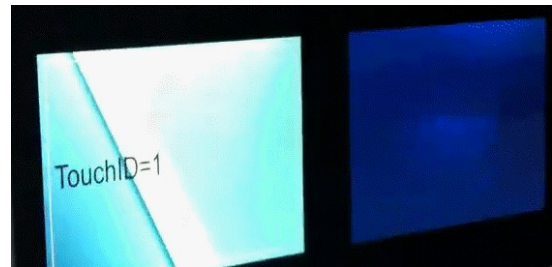
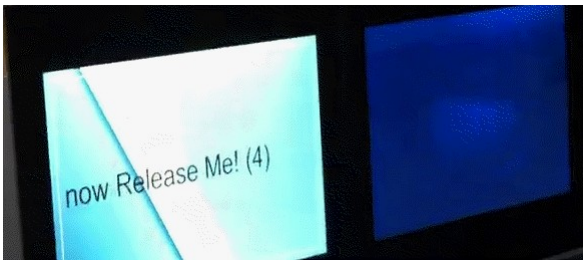
*VisAdmin Home page*



2. Click on the **Start** button to start the assignment process - an ident appears on one of the **Visual Extension** touch-screens asking you to touch it:



3. Touch the screen - the ident confirms the touch by showing "now Release Me!", and then confirms the assigned ID:



You are now asked to touch the next screen, so repeat the process.

When the operation is complete, the **Visual Extension Setup** window updates to show the assigned IDs.

4. Select **Back** to return to the **VisAdmin** Home page.

The **TouchID** is determined by the Visual Extension hardware (set by a rotary wheel within each screen display). The default IDs are **15** = the central screen (above the Central Module) and **1, 2, 3**, etc. = screens above Fader Modules counting from left to right.

If the system detects a **TouchID** conflict, then you will be informed during step 3; each screen must have a unique ID. This could occur if you forget to set the touch ID when replacing a display unit.

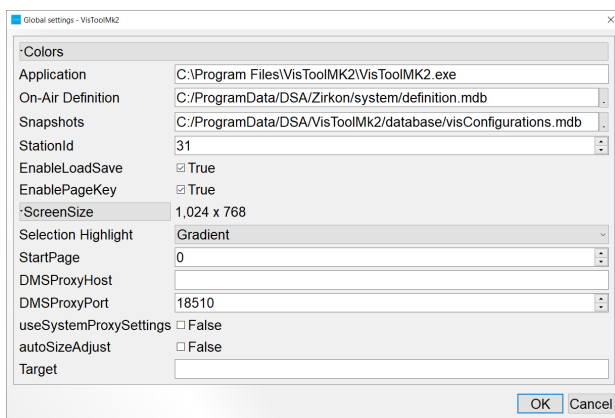
### 13.4.4 Additional Information for Visual Extensions Requiring Two PCs

If the Visual Extension is fitted with more than 6 screen displays, then a second VisTool PC is required (as a single PC can support up to 6 runtime instances). See [Wiring](#) for details on how to connect up the system.

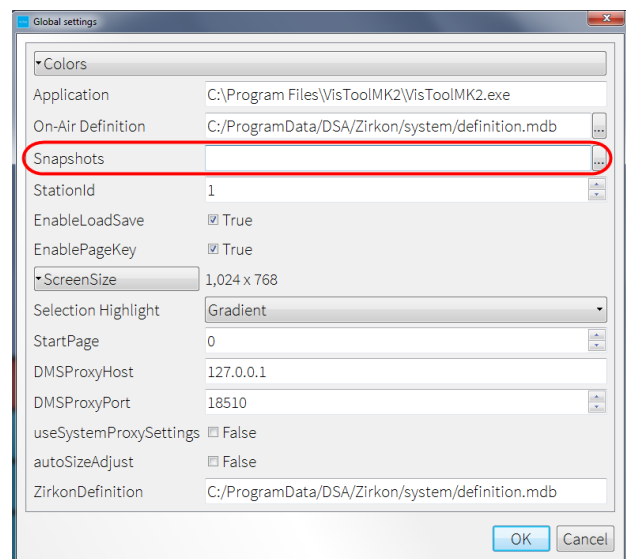
On both PCs follow the configuration steps described [earlier](#), noting the following:

- You must use the same configuration data on both PCs. i.e. the same VisTool project (**.vistool**) and ON-AIR Designer project (**.db3**). You can use the standard templates delivered with your system.
- Open VisTool Editor and the [Global Settings](#) window, and check the configuration.
  - On the PC connected to the main Visual Extension screen (above the Central Module), the **Snapshots** field must specify a valid path for the [VisConfigurations.mdb](#) file. The default file path is usually fine.
  - On the second PC, clear the **Snapshots** field. This ensures that all snapshots and user rights data is stored in a single file location (defined on the main VisTool PC).

*Global Settings (on main PC)*



*Global Settings (on second PC)*



- Then generate the start scripts from the configuration data using [VisTool Sessions](#). You must configure a separate "session" for each screen connected to the PC, and enter independent [Station IDs](#) for each screen/session.

## 13.5 Using the Standard Template

The standard template delivered with your system can be used to prepare a start script in the usual manner (using [VisTool Sessions](#)). The template supports the following functionality.

### 13.5.1 Default Pages

Above the Fader Modules you will see [channel metering](#) and status information for the assigned sources.

Above the Central Module is the MAIN page. This contains a Lawo logo (to define system options), a [Clock & Timer](#) (which can be reversed) and "MIC OPEN" indicator (which lights whenever a MIC source fader is open).

## 13.6 Using a Custom Configuration

Alternatively, you can purchase a **VisTool Unlimited** license to run a custom configuration prepared by [VisTool Editor](#).

## 14. Troubleshooting

---

This chapter covers some typical problems and their solutions.

Topics include:

- [Solving Connectivity Issues](#)
- [Performance Issues](#)
- [Operational Problems](#)
- [Troubleshooting the Visual Extension](#)

## 14.1 Solving Connectivity Issues

### 14.1.1 The GUI is not responding

If the VisTool GUI is not responding, then this could be due to an invalid network connection:

1. Try running a Windows PING test.

If you do not receive an answer then continue as follows.

2. Is the remote device connected to the mains supply?

If not, connect and power on the system.

3. Try restarting the remote device with a warm start.

For details, please refer to your device's User Manual.

4. Is there a network connection between the PC and remote device's control system?

Check your connections. If the system and your PC are connected via a network switch, then try bypassing this and use a direct connection from the unit's ETHERNET port to the PC, remembering to use a crossover network cable for this connection.

5. Is the IP Address assigned to your PC within the correct address range and subnet mask as the remote device's control system?

If not, you will need to change the IP settings on your PC accordingly.



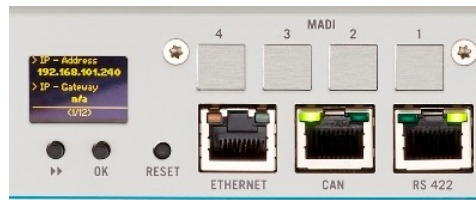
## 14.1.2 What is my system's IP address?

On Power Core and the Compact Engine, the IP address of the control system is shown on the unit's front panel display:

*Power Core (front view)*



*Compact Engine (front view)*



If you cannot see the IP address, then use the menu controls (rotary knob on Power Core, or >> button on the Compact Engine) to cycle through the pages of information.

For sapphire systems, there is no front panel display on the System Unit. You can either check the IP address defined in the configuration (using the ON-AIR Designer's 'Project' window), or reset the IP settings as follows.

Press the MODE button on the front of the Master Board:

*sapphire Master Board (front view)*



The IP settings are reset to the factory defaults:

- IP Address = 192.168.101.240
- Subnet mask = 255.255.255.0

## 14.2 Performance Issues

### 14.2.1 The graphics take a long time to update

This can occur if the PC does not meet the minimum system requirements. For real time operation, the PC *must* meet the [hardware requirements](#) specified earlier.

## 14.3 Operational Problems

### 14.3.1 VisTool Snapshots or User Logins are not working

If VisTool Snapshots or User Logins are not working, then check the following:

1. In the VisTool Editor 'Global Settings', is the correct file path specified for the "Visconfigurations.mdb" file in the [Snapshots](#) field?

If not, VisTool cannot save or load snapshots, or access the user rights data.

2. Is the correct [Station ID](#) defined for the session?

This must match the corresponding **Station ID** in the ON-AIR Designer's "[Logic -> VispageSwitch](#)" element IF you are using VispageSwitch elements.

3. Are the correct ON-AIR Designer and VisTool project files specified in the start script?

These are defined for the screen/session by [VisTool Sessions](#).

4. Are the user rights in Windows preventing access to the configuration files?

Please see [Preparing the User Access in Windows](#).

## 14.4 Troubleshooting the Visual Extension

### 14.4.1 The screens do not light up

If the screens do not light up, please check the following:

1. Are the Display Link [drivers](#) installed?

They are required to transport the graphics.

2. Is the VisTool PC connected to the Visual Extension frame via USB?

If not, check the [wiring](#).

3. Is an unsupported device used to lengthen the USB connection?

Only a “USB Ranger 2212” USB 2.0 Extension may be used as an extender.

4. Is power connected to the Visual Extension frame?

### 14.4.2 Screens are not found or appear in the wrong order

It can occur that not all of the screens are found if you attempt to run more than 6 screens from a single VisTool PC. To run more than 6 screens, you must install a second PC. See [Additional Information](#).

Similarly, the screens may appear in the wrong order when you first power on the VisTool PC. This is perfectly normal and can be easily corrected, see [Correcting the Screen Orientation & Order](#).

**DO NOT** attempt to physically rotate or reorder the screens! This is unnecessary, as the screens can be easily corrected from the Windows Control Panel.

### 14.4.3 The touch-sense is not working

In this instance, check the following:

1. Is the CAN bus connected and correctly terminated? Please see [wiring](#).
2. Is the DMS service running on the PC? You can check this on the [VisAdmin](#) Home page.
3. Is the CAN cable too long? (60m maximum at 500 kb/s).
4. Have the touch screens been detected by VisAdmin? Please see [Setting up the Touch-screen IDs](#).
5. Is the firewall blocking the connection? (Port 18510 for TCP and port 18511 for UDP must be open. Please contact your system administrator).
6. Have the touch-screens been initialised? If not, this can be done via a terminal command. Please contact the [Lawo support](#) department for assistance.

## 15. Appendices

---

This chapter includes further information which you may find useful.

Topics include:

- [VisTool MK2 dBAlias](#)
- [Files & Organization](#)
- [Matrix Numbers](#)
- [Advanced Licensing Features](#)

## 15.1 VisTool MK2 dbAlias

**VisTool MK2 dbAlias** is installed automatically with the VisTool MK2 [installer](#). It can be used to export VisTool databases into formats which support multi-user access. For example, to create a snapshots or user database that can be accessed from lots of different workstations.

The program can be launched from the Windows START menu: select **Program Files -> VisToolMK2 -> VisTool dbAlias**.

Then create a database, with an alias name, for your intended application. Supported formats besides SQLite and MDB are rdbms like Microsoft SQL Server and MySQL, if ODBC drivers are installed on the computers.

## 15.2 Files & Organization

### Configuration Files

The ON-AIR Designer creates configuration files offline and saves them as database files on your computer. A number of different files are created with different suffixes. All files are saved to the same folder as the main project (.db3 file).

**.db3** – this is the file which you create when you save a project, and it is this file you edit using the ON-AIR Designer software.

**.mdb** – this is the format used by the previous configuration tool. It is supported for backwards compatibility.

**.prm** – this is the binary file which is created when you perform a Transfer -> Setup to unit. The file contains all the setup parameters for the system and is transferred and stored on the control system.

**.cfg** – this is the binary file which is created when you perform a Transfer -> Config to unit. The file contains all the configuration settings for the system. It is stored in the "unit" folder of the project directory.

**\_\_Kpf.mdb** – this is the file which stores source and target names for inputs and outputs which are matrix enabled. It is stored in the "unit" folder of the project directory.

**before\_convert.mdb** – this is a file which is created when the ON-AIR Designer converts an existing file. It allows you to keep a copy of the original project before conversion.

It is a good idea to create a project folder to store all the files required for a particular system, and to make backup copies of your work to avoid losing any data.

### VisTool Files

If you are using VisTool Editor, then this will create a **.vistool** file when you save the VisTool project. It is a good idea to keep the VisTool and ON-AIR Designer projects together, and name the files clearly as you will need both files to configure a VisTool runtime session.

VisTool also uses a file named **Visconfigurations.mdb** file to store any snapshots and user rights created by the VisTool PC. This is created automatically when you install the VisToolMK2 software (in the folder "C:\ProgramData\DSA\VisToolMK2\database\"). The file path can be changed from VisTool Editor's "Global Settings" (via the [Snapshots](#) field).

In addition a number of **.ini** files to store other variable such as the list of Ember+ hosts, V\_\_line devices, etc. By default, these are stored in the folder "C:/ProgramData/DSA". The file path can be changed from VisTool's Session Manager (via the 'optional Parameters' [.iniPath](#) field).

### 15.3 Matrix Numbers

When using matrix functions, the inputs and outputs within the Lawo system are addressed as shown below. Note that the matrix address is the same as the **ID** number shown in the Command -> Inputs/Outputs lists (in ON-AIR Designer), and so this can be a useful way to determine the address for an individual signal. To control crosspoints within an external matrix, please contact the manufacturer for details on their matrix addressing system.

Matrix Address	DSP Core			
	Power Core	NOVA17	NOVA29	COMPACT ENGINE
1	IO CARDS	IO CARDS	MADI 1	LOCAL IO
65			MADI 2	
129	MADI 1	MADI 1	MADI 3	MADI 1
193	MADI 2	MADI 2	MADI 4	MADI 2
257	MADI 3	MADI 3	MADI 5	MADI 3
321	MADI 4	MADI 4	MADI 6	MADI 4
385	MADI SLOT 1	MADI 5	MADI 7	LOOPBACKS
449		MADI 6	MADI 8	
513	MADI SLOT 2	LOOPBACKS	MADI 9	
577			MADI 10	
641	MADI SLOT 3		MADI 11	
705			MADI 12	
769	MADI SLOT 4	UDP	MADI 13	
833			MADI 14	
897	MADI SLOT 5		MADI 15	
961		GT	MADI 16	
1025	MADI SLOT 6		LOOPBACK PORTS	
1089				
1153	MADI SLOT 7			
1217				
1281	MADI SLOT 8			
1345			DMS	
1409	RAVENNA		GT 1	
1473			GT 2	
1537				
1601				
1665	LOOPBACKS			
1729				
1793				
1857				
1921	GT			
1985				
2049	UDP			



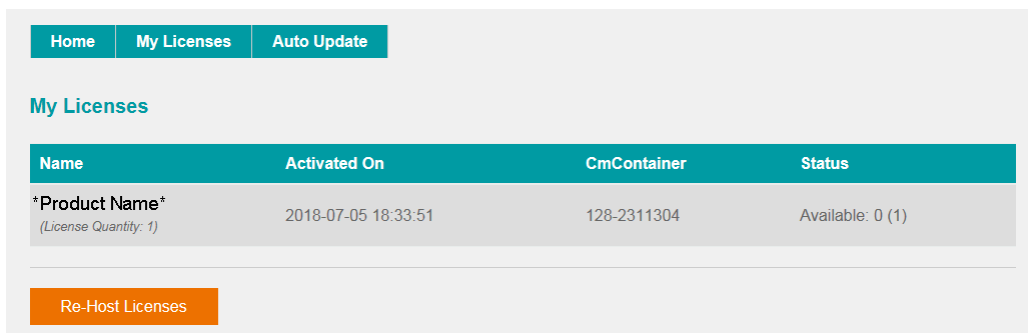
## 15.4 Advanced Licensing Features

This section describes the more advanced features of the CodeMeter Runtime licensing system.

### 15.4.1 Re-Hosting a License

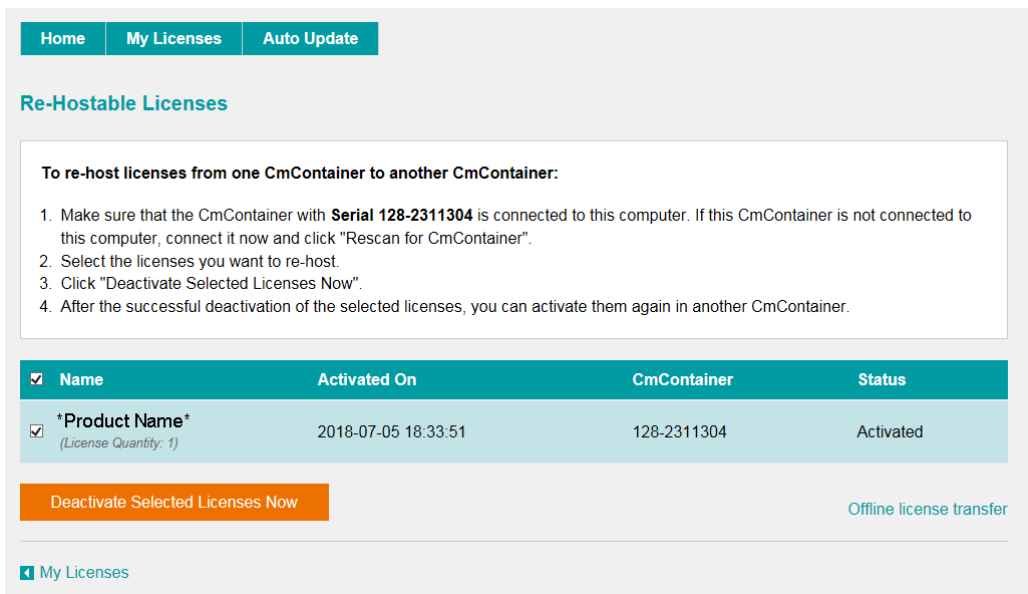
To move a license from one Cm container to another, you will need to re-host the license as follows. This requires you to first de-activate the license from its existing Cm container, and then choose the new storage container.

- Follow the first two steps from the [online activation](#) method:
  - Open the 'Lawo License' web browser page at <https://licenseportal.lawo.com>.
  - Copy your license ticket number - this is the 25 number code - into the **Ticket** field and select **Next**.
- At the "My Licenses" summary window, select **Re-Host Licenses**:



Name	Activated On	CmContainer	Status
*Product Name* <small>(License Quantity: 1)</small>	2018-07-05 18:33:51	128-2311304	Available: 0 (1)

- Make sure that the Cm container is connected to the computer, select the licenses you wish to re-host and select **Deactivate Selected Licenses Now**.



**To re-host licenses from one CmContainer to another CmContainer:**

- Make sure that the CmContainer with **Serial 128-2311304** is connected to this computer. If this CmContainer is not connected to this computer, connect it now and click "Rescan for CmContainer".
- Select the licenses you want to re-host.
- Click "Deactivate Selected Licenses Now".
- After the successful deactivation of the selected licenses, you can activate them again in another CmContainer.

<input checked="" type="checkbox"/>	Name	Activated On	CmContainer	Status
<input checked="" type="checkbox"/>	*Product Name* <small>(License Quantity: 1)</small>	2018-07-05 18:33:51	128-2311304	Activated

Wait for a few seconds - a confirmation pop-up appears once the de-activation is successful.

You can follow steps 3 to 7 from the [online](#) activation method, or perform an [offline](#) license transfer, to activate the license using a new storage container.

## 15.4.2 Activating a License Offline

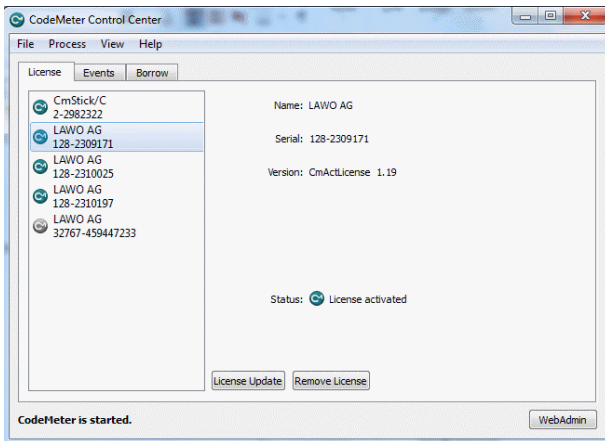
If your PC has no internet access, then you can activate a software license offline. This method involves three stages:

- **Create a license request file** - for the Cm storage container. Then copy the file onto a computer with internet access.
- **Activate the license** - using the 'Lawo License' web portal, copy the license update file back to the original computer.
- **Import the license update file** - to the Cm storage container.

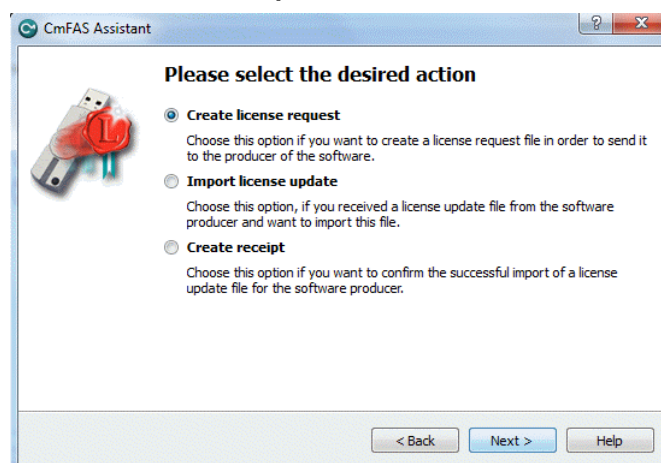
### Creating the License Request File

On the computer you wish to license:

1. Open the '[CodeMeter Control Center](#)' (by clicking on the Cm taskbar icon), and select the container you wish to use for the license storage.
2. If the container is empty, select **Activate License**. Or, if the container already holds an active license, select **License Update**. This starts the 'CodeMeter Field Activation Service (CmFAS) assistant':

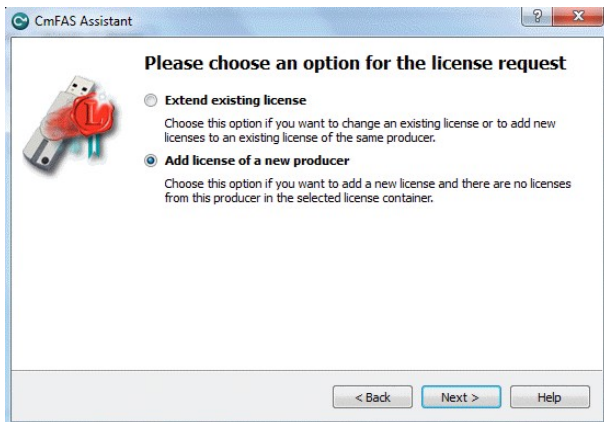


3. Select **Next:** and then **Create license request:**



At this stage, there are two additional steps (to add the correct firmcode) if you have selected a **CmStick** container. Enter the following Lawo FirmCode to create the license request file:

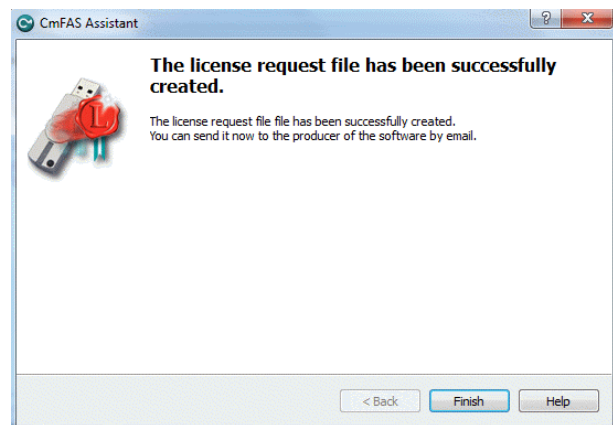
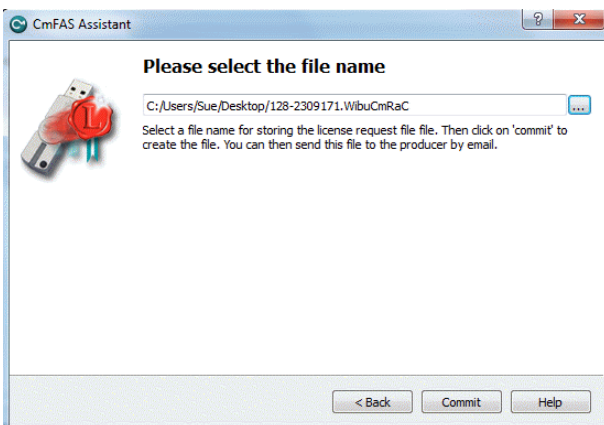
*Select Add license of a new producer*



*Enter the Lawo FirmCode = 102037*



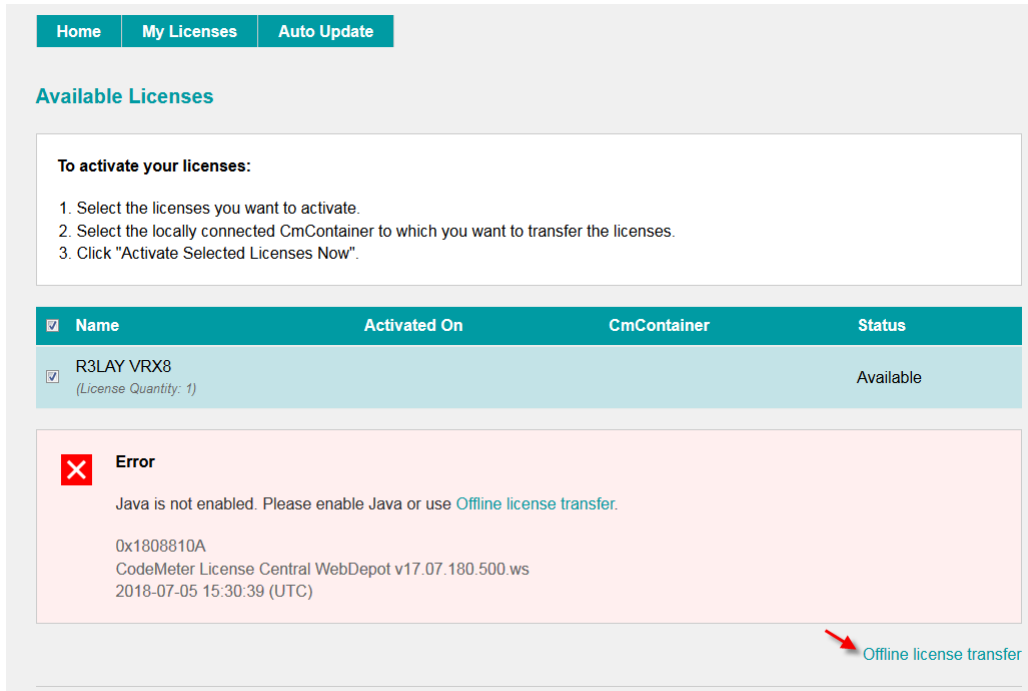
4. Select **Next**, and using Windows Explorer, enter a file path for the license request file. Choose somewhere easy to find, such as the Desktop. Then select **Commit** to create the request file:



5. Select **Finish** and copy the request file onto a computer with internet access. You will need both the license request file and the 25-digit ticket number (received with your license purchase).

## Activating the License

1. On a computer with internet access, follow the first four steps from the [online activation](#) method:
  - Open the 'Lawo License' web browser page at <https://licenseportal.lawo.com>.
  - Copy your license ticket number - this is the 25 number code - into the **Ticket** field and select **Next**.
  - When your license is displayed, select **Activate Licenses** to continue.
  - Select the storage method for your license - either USB dongle or single computer.
2. At the **Available Licenses** page, select **Offline license transfer** to continue:



Home My Licenses Auto Update

### Available Licenses

**To activate your licenses:**

1. Select the licenses you want to activate.
2. Select the locally connected CmContainer to which you want to transfer the licenses.
3. Click "Activate Selected Licenses Now".

<input checked="" type="checkbox"/>	Name	Activated On	CmContainer	Status
<input checked="" type="checkbox"/>	R3LAY VRX8 <small>(License Quantity: 1)</small>			Available

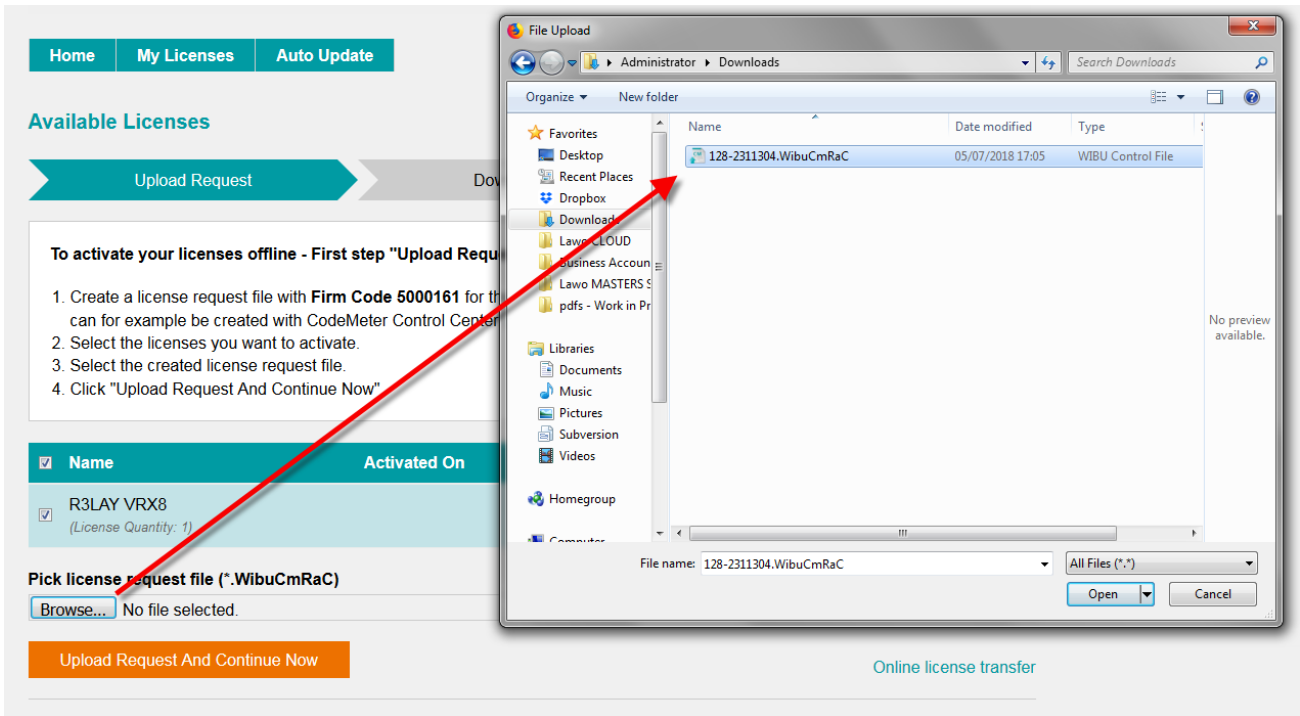
**Error**

Java is not enabled. Please enable Java or use [Offline license transfer](#).

0x1808810A  
CodeMeter License Central WebDepot v17.07.180.500.ws  
2018-07-05 15:30:39 (UTC)

[Offline license transfer](#)

3. On the next page, select the license(s) you wish to activate, select **Browse...** and, using Windows Explorer, choose the request file (created earlier):

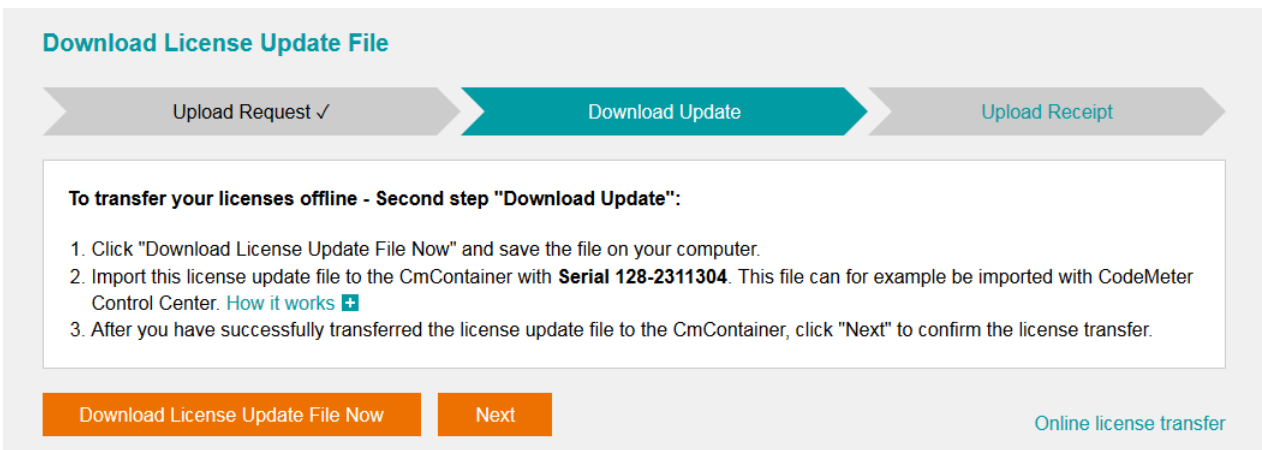


The screenshot shows the 'Available Licenses' section with a table:

Name	Activated On
<input checked="" type="checkbox"/> R3LAY VRX8 (License Quantity: 1)	

Below the table, the 'Pick license request file (\*.WibuCmRaC)' section has a 'Browse...' button. A red arrow points from this button to a Windows File Upload dialog box. The dialog box shows the file '128-2311304.WibuCmRaC' selected in the Downloads folder. The file name is '128-2311304.WibuCmRaC' and the file type is 'WIBU Control File'.

4. Then select **Upload Request And Continue Now** - the license request is processed and, if successful, you will have the option to Download the update:



The screenshot shows the 'Download License Update File' section with a progress bar:

Upload Request ✓ → **Download Update** → Upload Receipt

**To transfer your licenses offline - Second step "Download Update":**

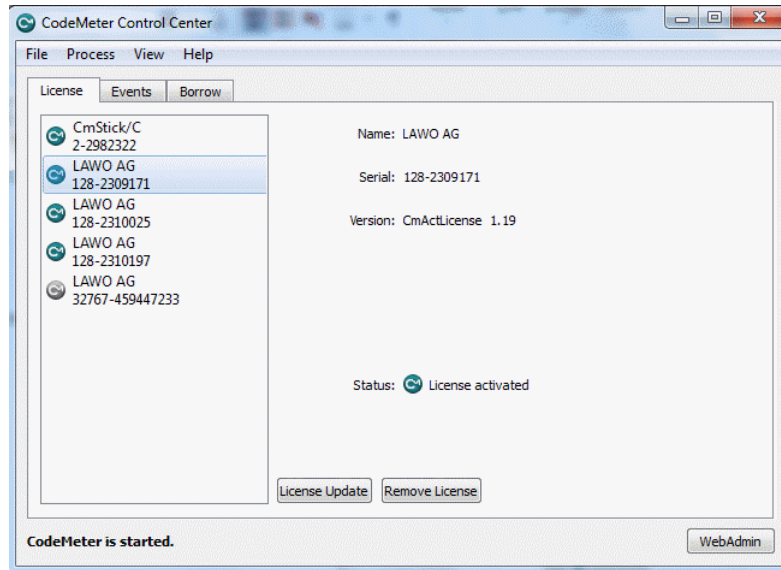
1. Click "Download License Update File Now" and save the file on your computer.
2. Import this license update file to the CmContainer with **Serial 128-2311304**. This file can for example be imported with CodeMeter Control Center. [How it works](#) +
3. After you have successfully transferred the license update file to the CmContainer, click "Next" to confirm the license transfer.

Buttons: **Download License Update File Now** and **Next**

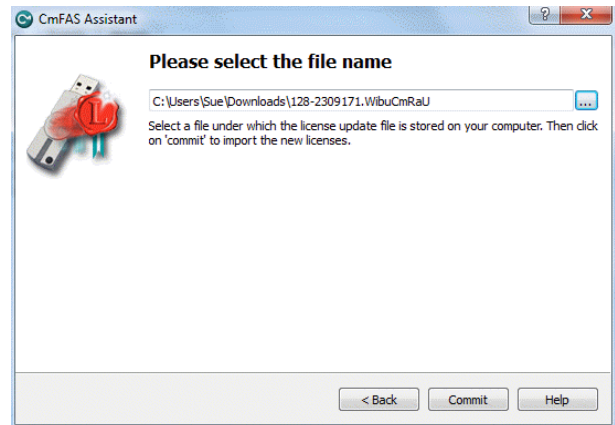
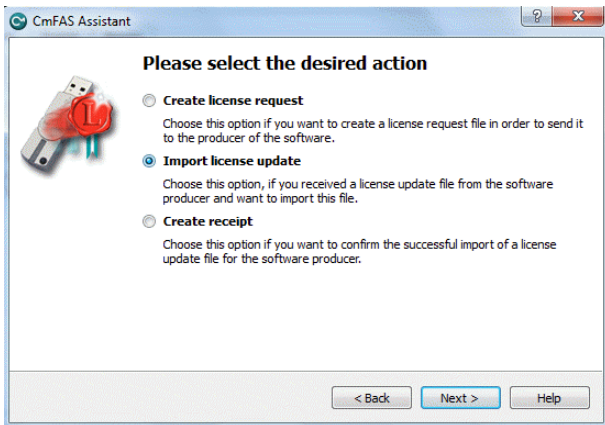
5. Select **Download License Update File Now** and, when prompted, choose the **Save** file option - the file is downloaded.
6. Copy the License Update file back to the original PC.

## Importing the License Update

1. On the original computer re-open the 'CodeMeter Control Center' and select the Cm container for the license update. Note that this must be the same container as the one selected earlier (during the License Request).



2. Select **License Update** and follow the instructions given by the 'CmFAS Assistant' - when prompted, select Import License update and choose the update file (downloaded from the License portal):

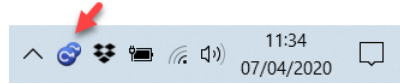


3. Select **Commit** to action the update - the license is activated and you can close the 'CodeMeter Control Center'.
4. You can now return to your Lawo application or install your USB license dongle - all licensed features should be available.

### 15.4.3 The CodeMeter Control Center

The 'CodeMeter Control Center' is used to manage the license containers and perform a backup or restore.

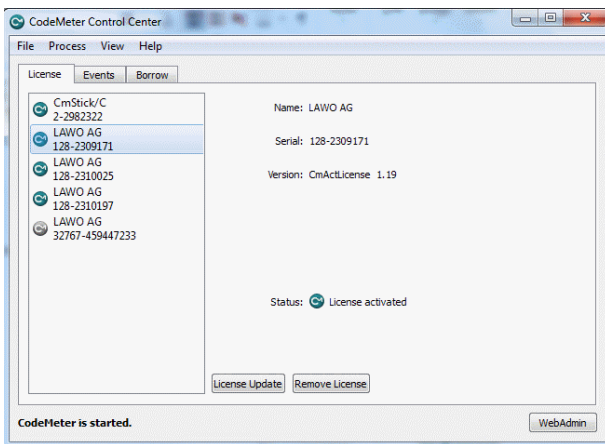
1. Click on the Windows taskbar Cm icon to open the 'CodeMeter Control Center' - the icon may be hidden from view or vary in color (depending on your taskbar configuration).



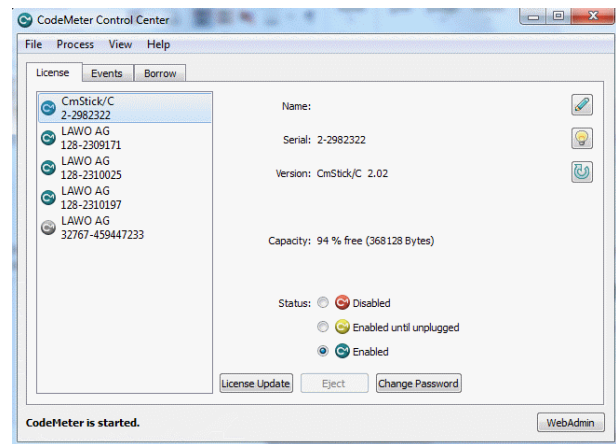
The 'CodeMeter Control Center' shows all the Cm containers which can be used for license storage.

A new local computer container is created each time you run the **CodeMeter Runtime** install wizard. Therefore, if you have installed multiple Lawo products or software versions, you will see several **LAWO AG** containers. If a USB dongle is connected, you will see a container labelled **CmStick**.

Local Computer Container (LAWO AG)



USB Dongle Container (CmStick)

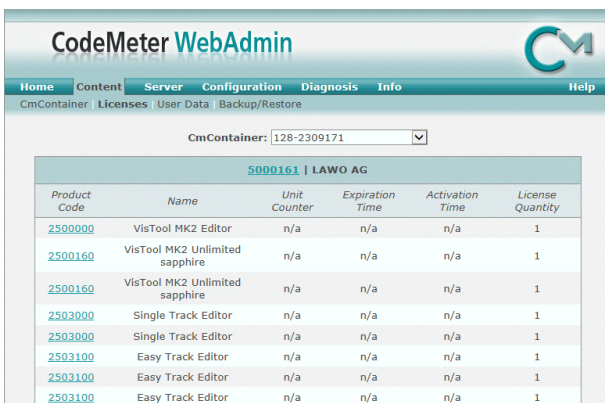


The **Serial** numbers identify each container. The icon colours indicate: green = license activated; grey = container is empty; red = license deactivated.

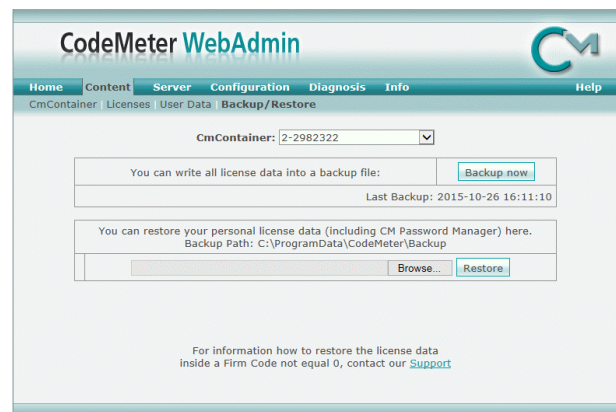
A single Cm container can contain multiple licenses - simply select the same container during the [activation](#) process.

2. Select **WebAdmin** (bottom right) to open the WebAdmin portal in your default browser. The portal has many functions including license interrogation, and backup/restore functions for licenses stored on a **CmStick** (USB Dongle):

WebAdmin Content Cm Container



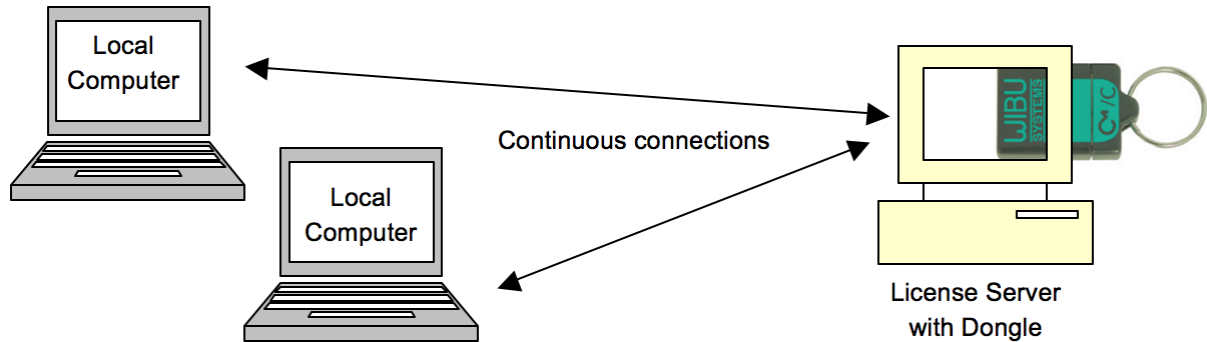
WebAdmin Backup/Restore



For further information, please refer to the **CodeMeter Runtime** documentation at [wibu.com](http://wibu.com).

### 15.4.4 License Configuration via a Central Server

This license storage method can be used to administrate licenses centrally within a local network. For example, when starting a Lawo application such as **VisTool**, the local computer asks the server to borrow the relevant license. The license is then used by the **VisTool** client until the application is closed. On closing, the license is handed back to the server where it may then be used by a different **VisTool** client.



#### Preparing the License Server

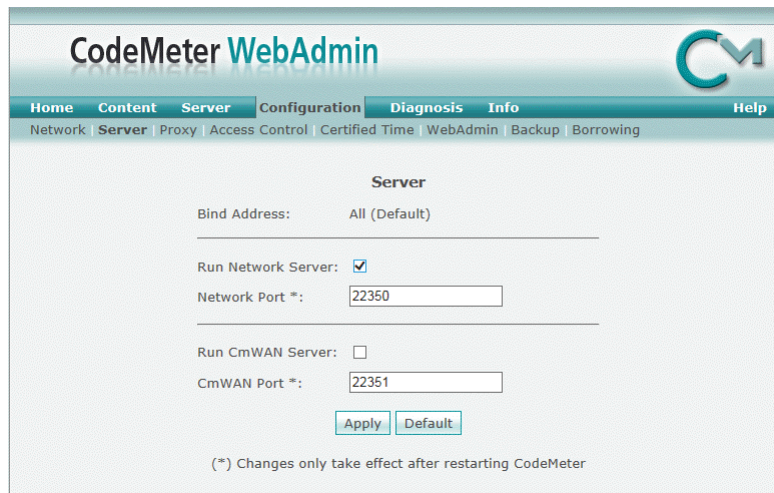
1. First, install the **Codemeter Runtime** software (Version 5.10 or above) on the central server. (Or, on a virtual machine on the server).

The latest release of CodeMeter Runtime can be downloaded from WIBU systems at [wibu.com](http://wibu.com).

2. Activate all licenses in the usual manner.

If a virtual machine is being used, set up a connection between the virtual machine and USB dongle.

3. Open the [CodeMeter Control Center](#) and select **WebAdmin**.
4. Under **Configuration** -> **Server**, select the **Run Network Server** checkbox:

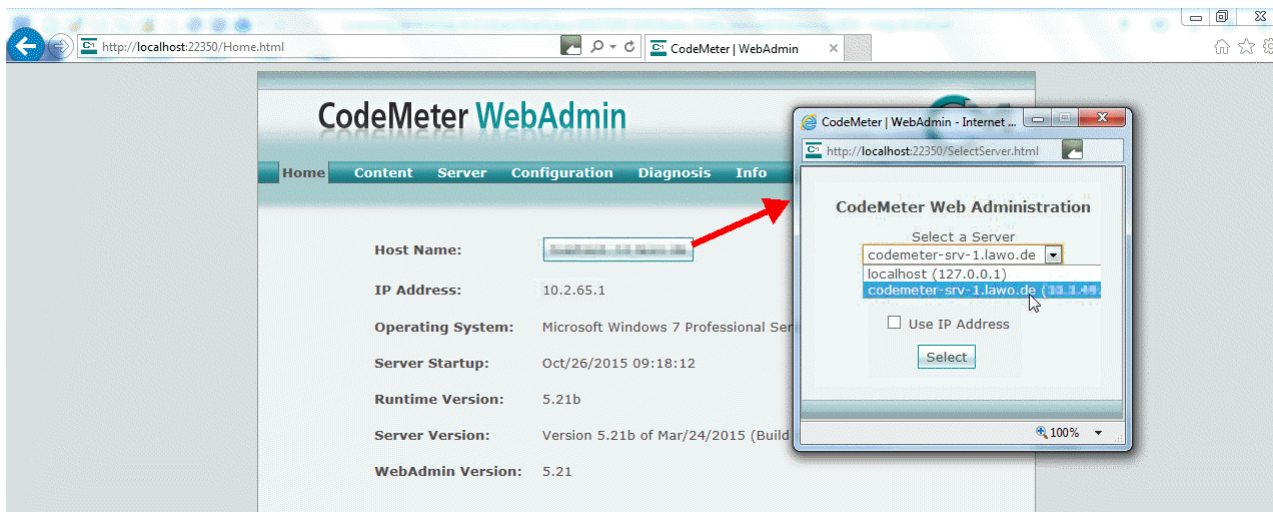


5. Click **Apply** and restart the **CodeMeter Runtime** software.



## Preparing the Clients

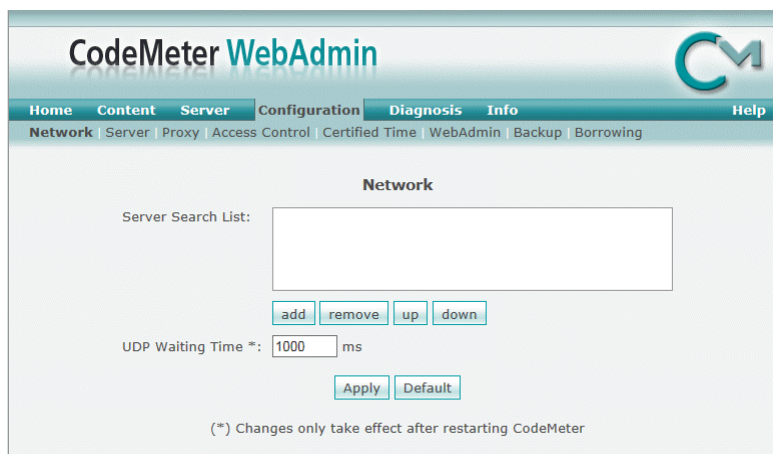
1. Install the **Codemeter Runtime** software (Version 5.10 or above) on each of the client computers.
2. Connect the clients (and server) to the network - on startup, the **CodeMeter Runtime** software sends a broadcast message across the network to find all license servers.
3. On each client computer, open the [CodeMeter Control Center](#) and select **WebAdmin**.
4. From the **Home** page, select the name of the computer - a dialogue box should appear listing all the available license servers:



5. Choose the correct server from the drop-down list and click on **Select**.

If the license server is not found, then your firewall or network policy may not allow messages to be broadcast. In this case, you will need to add the server's IP settings to the search list, manually, as follows:

Under **Configuration** -> **Network**, add the correct IP settings into the **Server Search** list:



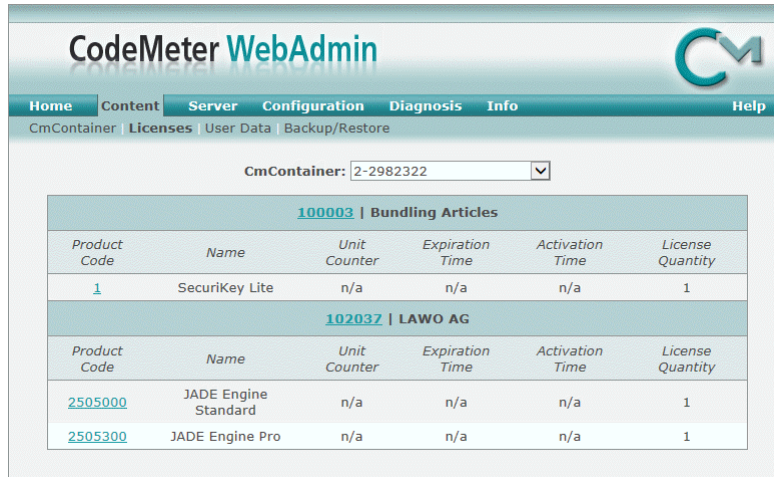
Click **Apply** and restart the **CodeMeter Runtime** software.

Note that once the **Server Search** list has an entry, all other license servers (announced automatically to the network) will be ignored.

## Choosing a License

Once a connection to the license server is established, you can borrow one of the server licenses as follows:

1. On the client computer, open the [Codemeter Control Center](#) and select **WebAdmin**.
2. Under **Content** -> **Licenses**, choose the **CmContainer** (holding the server licenses) and select the desired license file:



The screenshot shows the CodeMeter WebAdmin interface. The 'Content' menu is selected, and the 'Licenses' sub-menu is active. A dropdown menu shows 'CmContainer: 2-2982322'. Below this, there are two tables of licenses. The first table is for '100003 | Bundling Articles' and the second is for '102037 | LAWO AG'.


100003   Bundling Articles					
Product Code	Name	Unit Counter	Expiration Time	Activation Time	License Quantity
<a href="#">1</a>	SecuriKey Lite	n/a	n/a	n/a	1

102037   LAWO AG					
Product Code	Name	Unit Counter	Expiration Time	Activation Time	License Quantity
<a href="#">2505000</a>	JADE Engine Standard	n/a	n/a	n/a	1
<a href="#">2505300</a>	JADE Engine Pro	n/a	n/a	n/a	1

If the licenses is in use, then a warning message appears.

You can check which licenses are available (free) by selecting **Server** and **Cluster**:



The screenshot shows the CodeMeter WebAdmin interface with the 'Server' menu selected and the 'Cluster' sub-menu active. The page title is 'Available Network Licenses at 'codemeter-srv-1.lawo.de''. Below this, there is a table showing available licenses with their status.

Product Code	Name	Feature Map	Licenses	Status						
				User Limit (Borrowed)	No User Limit	Exclusive	Shared	Free		
102037   LAWO AG										
2503000	Single Track Editor	0x0	1	0 (-)	0	0	0	1	<a href="#">Details</a>	
2505400	JADE VST Option (Dongle)	0x0	1	0 (-)	0	0	0	1	<a href="#">Details</a>	