MAPIO 2 PRO/LITE

User manual

Revision 2

MAPIO 2.3.0

contact@visution.com

Table of contents

1 Introduction

Welcome to Visution MAPIO 2 User Manual. This is a software tool to project multimedia (video, animation, and interactive) data via screen projection units and addressable RGB LED lights onto complex surfaces. Primarily, the software is meant to be used for 3D mapping projections on buildings and other objects. It is also possible to communicate with the software interactively and integrate it with other systems. The software provides convenient tools to make it possible for it to be used in both manual and automatic modes, and to control it and carry out its maintenance via network (Ethernet).

MAPIO 2 is available as both PRO and LITE editions. This user manual covers PRO version features. For those features unavailable in LITE edition, **PRO [text]** or **PRO|** will be shown on the left of a paragraph or a heading.

Terms:

“**Mapping**” means a process of transformation from one geometry into another.

“**Slice**” is a primal element of geometry transformation.

“**Source**” is any media content you can attach to be shown in a slice. Among examples of sources are video files, images, data captured from a device, QML script, etc.

“**Destination**” is reserved for data output methods, such as window, screens, video stream interchange facilities (Spout, NDI), LED lights. Some methods can be combined.

“**Input/Output map**” are sets of coordinates needed for mapping process. Input map are triangle (quadrangle) vertices coordinates. They describe an area to be taken from a source (video frame, image, etc.). Output map are quadrangle (triangle) vertices coordinates where a slice from a source on a screen (or screens) should be put.

“**QML**”, “**Qt Meta Language**” is a User Interface Markup Language (http://doc.qt.io/qt-5/qmlapplications.html).

“**Pixel-mapping**” is data output to LED lights.

“**Sensor**” is an element of a project which sets a method and geometry to read data from a render to output on pixels/LEDs.

“**Console**” is a software interface section with mapping content controls. It consists of two control units, Deck and Mixer.

“**Deck**” is a console element. As with a hardware of the same name used by DJs, Deck is a video/audio player with playlists (they can be automated), preview functionality, controls and settings.

“**Mixer**” is a console element very similar to a DJ hardware of the same name. Mixer is used to mix visual stream for two or more Decks.

2 System Requirements

Operating System: Windows 7 SP1/8.1/10 (32/64 bit). Windows 7 requires DirectX to be installed.

CPU: Intel Core i3/i5/i7 or higher, or their AMD counterparts; 2 cores are minimum, 4 cores are recommended.

GPU: NVidia GeForce 8xxx or higher, and ATI Radeon HD 2xxx, HD 3xxx, HD 4xxx, HD 5xxx or higher, Mobility Radeon HD 43xx or higher, Intel HD Graphics 3000 or higher.

Make sure your GPU (integrated too) drivers are up to date!

While the requirements listed above guarantee the software can be run on the hardware provided, its eventual efficiency (FPS, playback smoothness, and overall workload of your system) may vary depending heavily on many factors, such as GPU/CPU, PCI Express and SSD/HDD performance, file resolution and codec, as well as output resolution of your project.

To make sure MAPIO will perform on hardware available, you can test it in Demo mode before making a purchase.

While the software is run in Demo mode, the only limitation is your video output will be watermarked. Unless the software was activated, MAPIO works in Demo mode. The latest installation package is always available at: http://visution.com/downloads.

It should be noted, that MAPIO is shipped with integrated video and audio codecs, developed and released by FFmpeg open project [ffmpeg.org]. To use the software, no third-party codecs are required. This saying, MAPIO doesn’t make use of your system codecs, so it’s impossible to play back files encoded with proprietary codecs installed in your system, if they haven’t been implemented by FFmpeg.

3 Installation and Launching

Before installation, please make sure to use an installation package intended for your operating system. There are two installation packages provided: x32 and x64. For 64-bit operating systems, both x32 and x64 MAPIO versions can be used, but we recommend installing the latter package to get around limitations of operating memory made available to the software. This is particularly true, if you use high-resolution video files. For 32-bit operating systems, only x32 version of the software is available.

There's a standard and convenient way to install a package in Windows. Please start an msi-file and follow the procedure. It's recommended to install MAPIO in folder suggested by default. After installation process finishes, an application icon will appear on your Desktop. The icon looks as follows:

Notes:

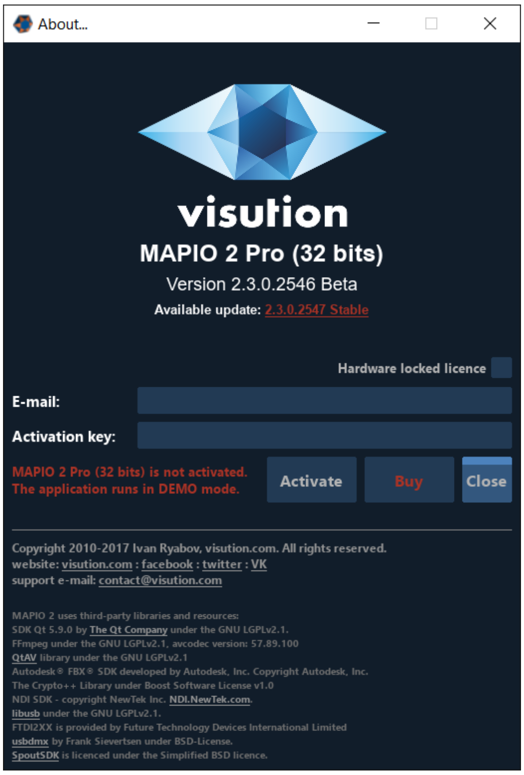
1) Since you can use MAPIO as a FFGL plug-in in Resolume software, to register it as a plug-in, please close Resolume before you launch MAPIO installation process.
2) If you want to install Resolume after MAPIO software is installed, please wait until Resolume installation process is finished, then launch a standalone instance of MAPIO. Then, it will detect Resolume and ask for permission to register a MAPIO plug-in there.

Upgrading and Uninstalling

To upgrade your MAPIO to the newest version you can install the new installation package without uninstalling currently installed package. The old version will be uninstalled automatically. You can uninstall MAPIO via default "Apps & features" panel or by launching an "Uninstall MAPIO2 Pro (x64)" icon from "Start" menu.

It's recommended to stop MAPIO Observer before uninstalling or updating the software.

4 License Activation and Deactivation



MAPIO 2 Pro (32 bits)

Version 2.3.0.2546 Beta

Available update: 2.3.0.2547 Stable

MAPIO 2 Pro (32 bits) is not activated.
The application runs in DEMO mode.

Copyright 2010-2017 Ivan Ryabov, visution.com. All rights reserved.
website: visution.com : facebook : twitter : VK
support e-mail: contact@visution.com

MAPIO 2 uses third-party libraries and resources:
SDK Qt 5.9.0 by The Qt Company under the GNU LGPLv2.1.
FFmpeg under the GNU LGPLv2.1, avcodec version: 57.89.100
QtAV library under the GNU LGPLv2.1
Autodesk® FBX® SDK developed by Autodesk, Inc. Copyright Autodesk, Inc.
The Crypto++ Library under Boost Software License v1.0
NDI SDK - copyright NewTek Inc. NDI.NewTek.com.
libusb under the GNU LGPLv2.1.
FTDI2XX is provided by Future Technology Devices International Limited
usbdmx by Frank Sievertsen under BSD-License.
SpoutSDK is licenced under the Simplified BSD licence.

Unless you activate MAPIO license, the software works in demo mode. It never expires, and it's fully-functional, albeit the video output will be watermarked.

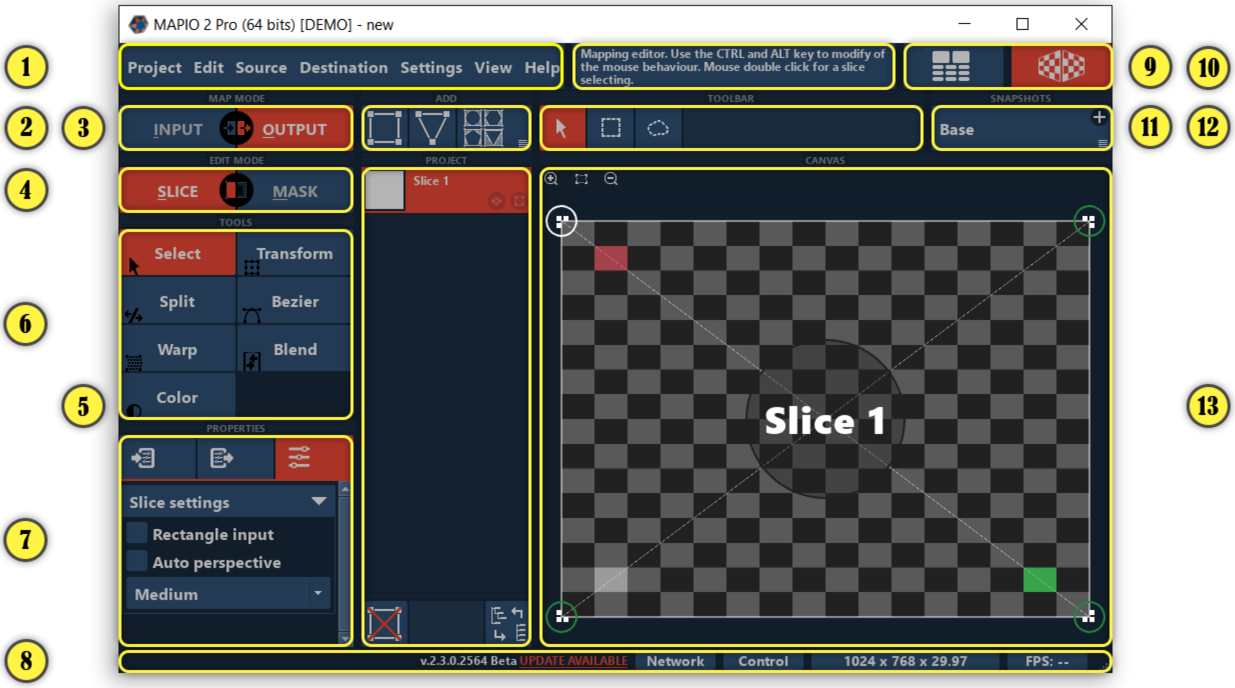
You can activate a license purchased by entering your credentials in "About" window. To open the window, please open "Help → About/Registration" (see Fig. above). Please enter the email given upon purchase. In the following box please enter your activation key received, and click "Activate". If the activation is successful, you'll see a notification in green. To apply your license and lift the restrictions, please restart MAPIO.

To deactivate your license please open "About" again and click "Deactivate".

The license installed will be saved for a currently logged user. Please note: you may want to enter your credentials again if you run MAPIO under another user.

5 Quick Start

5.1 Interface Overview



Mapping editor. Use the CTRL and ALT key to modify of the mouse behaviour. Mouse double click for a slice selecting.

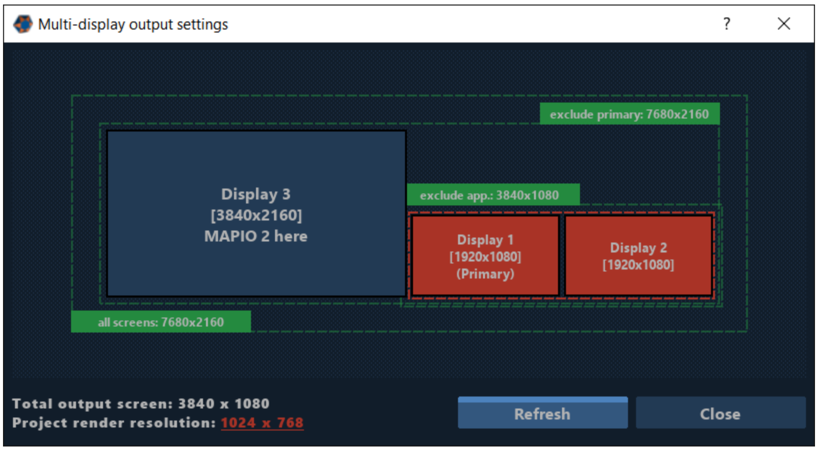
1) Main menu, see Section 6
2) Mapping map mode — either input or output map, see Section 5.4
3) Add elements to your project, see Section 7.3.1
4) Slice or Mask Geometry Editing Mode, see Section 5.4
5) Element List of your project, see Section 7.3
6) Tools, list contents depend on current map mode and editor, see Section 7.1
7) Current element Properties. Properties are divided into three pages: vertices coordinates for Input map, vertices coordinates for Output map, Other properties of the element, and properties and settings of the tool currently in use, see Section 7.2
8) Main Window status bar displays (L-R): version of the software, **PRO [**Network button with networking functionality, see Section 12.1**]**, Controls settings button, Project resolution settings, Output FPS, **PRO [**Rendering, CPU Load, and Memory Usage statistics are available upon click**]**.
9) Tip on UI element your mouse pointer is pointing at.
10) **PRO [**Interface toggle to switch between Console (media manager) and Mapping Editor. In PRO, the software interface is split into two. The first one is Mapping Editor . The editor is used to set transformation geometry of slices and its masks, and locations of other elements. The other part is Console , interface to interact with media content.**]** There's no Console in LITE version. There, you attach media sources to a slice directly, this means you cannot switch between files in a slice immediately, and every slice matches only one source (file, image, or any other media source).
11) Additional settings toolbar for the current tool
12) Mapping configuration snapshots, see Section 7.4
13) Editor Working area (Canvas). For an input map, Working area is a current source frame, with its aspect ratio usually corresponding aspect ratio of the source, if applicable. Output map represents

resulting output of all monitors, i.e. if there are two monitors, the working area will be divided into two parts with aspect ratio of the area corresponding your project's resolution (see Output Settings).

By default, MAPIO is initialized using a project with a slice expanded to fit in Canvas in both input and output maps. Practically, it works like a bypass and it means there's no transformations.

You can't have no slices in your project. Every time you delete the last slice remaining, new default slice will be added automatically.

5.2 Output Settings



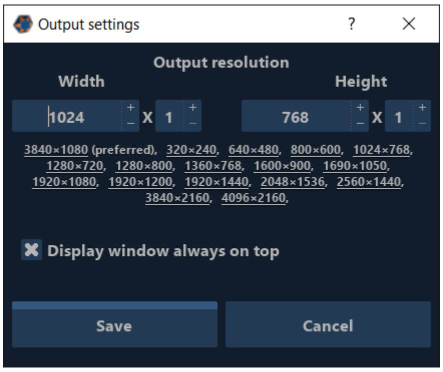
Total output screen: 3840 x 1080
Project render resolution: 1024 x 768

Since the purpose of projection mapping is to output images to projectors, it's usually a good idea to set up both output methods and resolutions. You may want to set one or more destinations, i.e. a monitor with MAPIO interface enabled AND at least one projector. A dedicated UI monitor is essential when mapping editing and setting are under way, but you can run MAPIO in autonomous mode with no dedicated MAPIO UI monitor (see Sections 13 and 18).

Output Settings can be found in "Destination" submenu under Main menu. If you are to use only a monitor for output, "Display" will be enough. Once selected, a window opens. You may leave it as it is or move to projector's desktop. Double-click inside the window to maximise it.

In case you would like to use several projectors, you will need Multi-Display (see Section 10). Here, you can select monitors as output. You can NOT position your monitors in MAPIO. To set their properties and positioning, please open your system settings or GPU control panel. You may want to restart MAPIO after the configuration has been changed. To enable and disable output to a certain monitor, click on the monitor in "Multi-Display settings" window. Double-click on the window to close it temporarily. When the project is loaded or Multi-Display is re-opened, output settings will be restored.

Next, you should set resolution of your project, i.e. the frame size to be used to render resulting images. Normally, to ensure the best results, you should set its resolution to be equal to output resolution, i.e. screen(s) resolution. In the figure above, you can see both total screen resolution and project resolution displayed (bottom left). To open resolution settings you can either find "Output Settings..." in "Destination" submenu within Main menu; or press CTRL+O; or click the output resolution button in your main window status bar. You can also open the settings by clicking on resolution in Multi-Display window.



Output resolution

Width: 1024 X 1

Height: 768 X 1

3840×1080 (preferred), 320×240, 640×480, 800×600, 1024×768, 1280×720, 1280×800, 1360×768, 1600×900, 1690×1050, 1920×1080, 1920×1200, 1920×1440, 2048×1536, 2560×1440, 3840×2160, 4096×2160,

☒ Display window always on top

In "Output Settings" window, you can set both frame width and height in pixels. You will find a tip with a list of the most common resolutions just below, and if you've already set any, the recommended resolution will be shown as the first item on the list.

5.3 Source Adding

Every slice needs a source, i.e. visual media content to be used as content of the slice to output it via projectors. By default, every new slice shows chequered pattern, this means there's no media content attached to the slice.

You can either attach a different source to every slice, or the same source to several slices. The most convenient way to attach a source (for example, a video file) is to select slice(s) and then to attach a Video to them using "Source" submenu in Main menu.

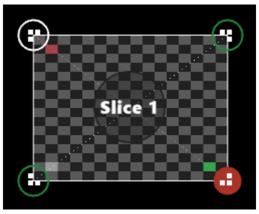
For selected slices with no video files previously attached, that's what happens behind the scenes: a Deck with the video file added is created in Console, then the Deck is attached to the slices as a source. Now, you can play video by pressing Alt+Enter (Ctrl+Enter to stop all sources).

Another way is to open Console (use F8 as a hotkey) and add new Deck by clicking button, then you can add a source with "Select Source" menu in a status bar of the Deck. By now, the Deck is ready but still unattached to a slice. To attach it, please go back to Editor and select desired slices. Then you can select the Deck you just created in "Attach…" submenu in "Source" menu. You can also attach the Deck by using deck menu. It opens on click on a button with the Deck number on it.

For **LITE**, there are neither consoles nor decks, and a source will be attached to a slice directly. To access properties and controls of the source, you may use "Properties" panel on the left.

5.4 Mapping Editor and Tools

Canvas, also mapping editor or working area, is a geometry editor for slices, masks and other elements. Selected mode and editing target are set by two mode switches (see Fig. at the beginning of Section 5, items 2 and 4), and by current tool (item 6). Among common Canvas features, you can drag a canvas with a left mouse button by any canvas area free from active elements (active elements are usually highlighted), or, if it's not possible, with CTRL key. To zoom in/out Canvas, you can use either mouse wheel or +/- keyboard keys. To reset zoom (and to fit the canvas to a window), please use N key on your keyboard. All the functionality is also accessible from a bar at the top of the canvas.



In a colour scheme used by default, active elements of a canvas are highlighted in green. Selected current elements are highlighted in orange. Slice vertices circled in green with current vertex (i.e. current focus to be controlled with keyboard keys) circled in white. You can use "<" and ">" keys to change focus, while "Space" bar is used for selecting and deselecting vertices. Selected vertices are filled with green.

While dragging vertices around with your mouse pointer or keyboard, precision of such movements can be increased by zooming the working area in.

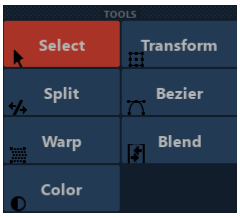
By default, vertices of two or more slices stick together, i.e. they are selected and dragged as if they were one. To temporarily switch the behaviour off, press and hold CTRL key. To disable the behaviour altogether, go to "Edit" menu and enable "Only the Current" option.

A vertex dragged will be sticking to other vertices and marks on guides when possible. To disable the behaviour, press and hold ALT key, while dragging.

By default, it's not permitted to drag slice vertices outside the canvas. To disable it, you should check "Allow to drag outside" option in "Edit" menu.

You can switch between slices either by using double click on a desired one in a slice list, or by using Tab/Shift – Tab keys and 1, 2, 3, ... 0 keys on your keyboard to select one of the first 10 slices directly.

Tools:



Select — Basic tool to edit slice geometry, allowing you to select vertices one way or another, and to drag them.

Transform — Tool for editing vertices in groups, and for dragging, scaling and rotating selected vertices.

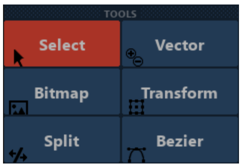
Split — It splits slices and masks into two with a straight line; the tool is available in any mode but not in any case (see Section 7.1.3).

Bezier — Curve transformations for slices (output map only) and masks (both input and output maps).

Warp — Morphing and warping tool to apply to slice contents. The tool is useful for adjusting slice geometry and available for slices on output map.

Blend — Tool for blending slice edges; available for slices on both input and output maps.

Color — Color and level correction, brightness, contrast and other characteristics of slice contents; available for slices on an output map.



Vector — Tool for adding/removing vector masks and their vertices; available for masks only.

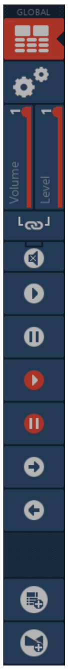
Bitmap — Image masks adding/removing; available for masks only.

Working with Masks

By default, masks are bound to a slice, its position and geometry. Being a part of a slice, masks are not identified as a separate element of a project. In case you will need it, you can unbind a mask from a slice by enabling “Absolute coordinates” option in “Properties” panel, but still, the mask will be applied to its owner slice.

If you need a global mask to apply it to every slice, you should add a slice filled with black (“Source → Color”), place it on top of all other slices and shape it with masks.

5.5 PRO | Media Console. Content Playback

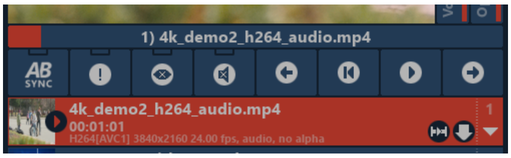


Media console, or Console, consists of a regular block plus a varying number of Deck and Mixer type blocks. A vertical block on the left contains project management elements and settings, and buttons used to add Decks/Mixers at the bottom (see Fig. above). For more information, please refer to Section 8.3.

A deck consists of media sources playlist and playlist controls. You can set up every playlist element to play/stop media playback automatically. For more information, please refer to Section 8.3.

A mixer allows you to switch decks smoothly with some automation tools provided. For more information, please refer to Section 8.48.4.

To channel content generated by a deck or a mixer to a slice, you should attach it as a source to the slice. You can do that by either choosing a slice needed in “Editor” mode, and then choosing a desired source in “Source → Attach…”, or by opening a menu with the deck or mixer number in Console mode, and to select “Attach to slices”. As a result, a deck/mixer chosen will be attached to a current or selected slice.


1) 4k_demo2_h264_audio.mp4
AB SYNC
4k_demo2_h264_audio.mp4
00:01:01
H264[AVC1] 3840x2160 24.00 fps, audio, no alpha
1

Basic controls (see Fig. above). An indicator displaying playback of a current playlist item (if its duration is known) can be found just below Preview display. Left-click on it to open an even bigger playback indicator with options to zoom in/out and to set starting/end points for playback repeat functionality.

There is a block with function keys below (left to right): Clock bus selection, Solo mode, Mute video, Mute sound, Go to previous, Rewind, Play/Pause, Go to next. The block is followed be a media elements list.

6 Main menu Overview

6.1 Project menu

- New
  - With one Quad — Initialize project with one slice. By default. Initialization resets mapping grid but doesn't affect Console.
  - With 1x2 — Initialize project with 2 slices, useful for edge blending of two projectors.
  - With 1x3 — Same as previous but with 3 slices.
  - With 2x2 — Same as previous but configured as 2x2.
  - With custom grid… — Custom grid.
  - With radial grid… — Custom grid but with slices placed radially.
- Load… — Load project, switches to the project loaded without further notice.
- Save — Save project. If you have already saved the project, it will be saved to the same parent folder and the file will be overwritten.
- Save As… — Save project with a file name and/or parent folder specified.
- 3D import… — Import FBX and Collada (DAE) 3D files. See Section 15.
- Export… — Export slices and masks to SVG and PNG as outline sets.

6.2 Edit menu

Mapping editing options and settings. Available in mapping editing mode only.

- Undo/Redo — Undo and redo changes made to a canvas.
- Copy/Paste — Copy and paste selected elements and masks to/from clipboard.
- Select All — Select all elements of your project.
- Only the current — Enable/disable editing mode for current element only. Other elements on the canvas will be ignored.
- Sync IO edit — Synchronized editing of both input and output maps.
- Allow to drag outside — Allow to drag vertices outside your working area.
- Lock/Unlock — Lock/unlock editing for selected elements to prevent them from being accidentally changed.
- Group — Create a group and add selected elements to it.
- Reorder -> Move up/Move down — Move up and down your project's list. It changes rendering order. Rendering is preformed from the bottom of the list upwards, thus, the first element on the list is on the top.
- Make equal IO — Copy current element coordinates from another map and apply to a map in use.

6.3 Source menu

Menu used to attach sources quickly. Available in mapping editing mode only.

- **PRO [**Attach… → New Deck/Mixer — Create new Deck/Mixer and attach it to selected slices.**]**
- **PRO [**Attach… → List of decks and mixers — Attach a Deck/Mixer to selected slices.**].** In **LITE**, the submenu can be found under Existent sources… — it contains a list of files loaded.
- Detach — Detach source(s), revert selected slices back to chequered pattern. It is known as "None" in **LITE**.
- Image — Choose an image file as a source.

- Video — Choose a video file as a source.
- Color — Choose color filler generator. See Generator properties in media source properties in Deck.
- Spout submenu — Choose Spout video as a source.
- NDI submenu — Choose NDI video as a source.
- QML submenu — Choose a QML script as a source. For more information, please refer to Section 16.
- Capture devices submenu — Video capture devices list.
- Adjustment patterns submenu — Temporarily switch to adjustment patterns. It will be applied to every slice, but won't replace sources attached to the slices.
- Play/Pause all Media — Play/pause every playlist element for every slice.

PRO [Choosing a source from the menu results in creation of a Deck with a source chosen. The Deck will be automatically attached to selected slice as a source.**]**

For more information on sources, please refer to Section 9.

6.4 Destination menu

MAPIO output selection and settings menu.

- None — Do not output anywhere/disable output.
- Display — Output to a window. Can be used in window mode or maximized to a display.
- **PRO [**Multi-Display — Output to several displays.**]**
- NDI — Output for video streaming communication system between applications in NDI network.
- Spout — Output for video sharing between Spout applications.
- **PRO [**Pixel-mapping… — Output to LED or other pixels devices via DMX or Serial Port. See Section 11.**]**
- Output settings… — Output settings, rendering resolution.

Display and Multi-Display are mutually exclusive output directions.

6.5 Settings menu

- Show mode — Special mode designated to disable mapping editor (to release system resources) and hide all accessories before the beginning of a show. The mode is also enabled when you use MAPIO in console mode.
- Auto-start options
  - Auto load current project — Set and save current project as a default project. On MAPIO launch it will be loaded automatically. The original file of the project will be used so please don't delete this.
  - Reset auto-load project — Reset default project you set previously.
  - Start all media after loading — Start all items on a playlist automatically upon loading a project.
  - **PRO [**Launch observer — Launch MAPIO Observer. For more information, please refer Section 13.**]**
- Color scheme — Choose from a set of color schemes available for MAPIO.
- Performance… — Performance settings window, see Section 17.

6.6 View menu

Mapping Editor Accessories appearance settings. Available in mapping editing mode only.

- Zoom In/Out/Reset — Working Area Zooming management.
- Output overlay — Settings auxiliary elements in output overlay.
    - Highlight slice mode — Sharp contrast highlighting of the current slice. Slices will be filled with color to highlight its borders.
    - Show mouse — Show/hide mouse pointer as a cross inside your working area.
    - Show vertices — Show/hide vertices and edges of current slice and highlight them in green. It helps to adjust slice using its projection and not your editor window.
    - Show grid — Show/hide white grid over current slice.
    - Show info — Show/hide project resolution and actual output resolution.
- Show labels — Show/hide circled labels with the name of a slice.
- Show content of slice — Show/hide slice contents in editor.
- Show guide lines — Show/hide guides in editor.
- Show tools panel — Show/hide left bar, it can be useful for low resolution displays.
- Background image... — Upload background image. It allows to create your own layout.

6.7 Меню Help

- Feedback — Feedback form with an option to send system configuration and MAPIO logs.
- About/registration — Information on MAPIO version, reserved rights and copyrights, and license activation form.

7 Mapping Editor

7.1 Tools

7.1.1 Select

This is a basic slice geometry editing tool. It allows to select vertices in different ways and drag them. There are three modes displayed in additional toolbar:

↖ — Basic mode, it allows to pan working area in/out and selecting nodes by clicking on slice vertices.

⬚ — Rectangle select mode. By clicking in free space, you can start to create a rectangle vertices selection area.

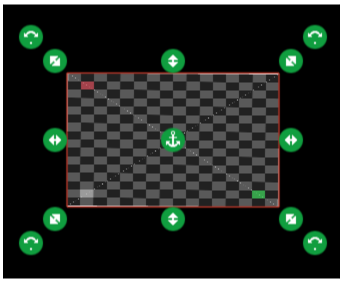
◌ — Free select tool. By clicking in free space, you can start to create a freeform vertices selection area.

In ⬚ and ◌ modes, panning is also available. You can hold CTRL key to perform this.

In slice mode, you can select all nodes of a slice by clicking on a circle with the slice name in it. Hold SHIFT key to select more than one slice.

To select all outline nodes in mask mode, click on the mask outline. Hold SHIFT key to select more than one outline.

7.1.2 Transform



Tool to move, scale and rotate selected vertices.

Along with standard active elements to scale and rotate, there are also options to set central point of coordinates, an anchor point to be used for rotation or scaling up/down operations. An anchor point can be dragged around with a mouse. You can also select a place for it from “Properties” in a bar on the left.

The tool looks and works the same way for both slice and mask vertices.

7.1.3 Split

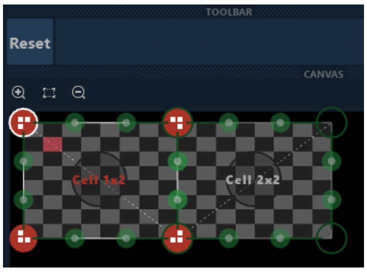
Tool to split slices and masks into two with a straight line. Available in all modes.

Split works only for slices with no curvilinear correction or blending, i.e. you cannot apply Split to a slice modified with either Bezier, Warp or Blend. To split the slice, you should either reset all changes made or create a new slice.

7.1.4 Bezier

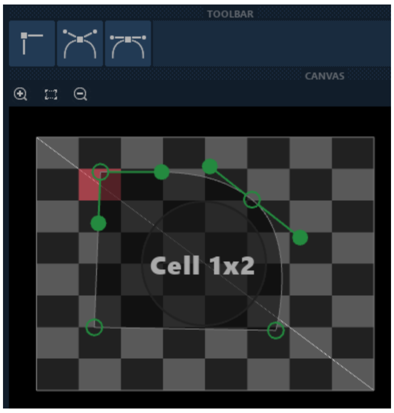
Tool to create curvilinear transformations of a slice and mask boundaries. Available for slices in input maps and for masks in both input and output maps. The tool doesn't work the same way for slices and masks.

For slices



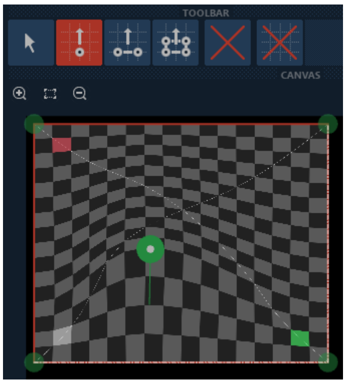
When activated for the first time, two handles for every vertex are created. They are inactive until you try to move a handle. After this, "Bezier" mode for vertices will be activated. Same as with slice vertices, neighbouring slices will be edited simultaneously. Two Bezier handles of neighbouring slices will be moved together. To disable the behaviour, hold ALT key or enable "Only the current" option in "Edit" menu. Use "Reset" button to reset selected vertices to their regular linear look. See Fig. above. Curvilinear transformations depend on slice quality.

For masks



Bezier tool for masks comes with three buttons to switch between vertices modes. Those are: basic linear, free handles Bezier mode, and fixed handles mode with Bezier handles fixed at 180 degrees. You can switch mode for vertices by selecting the ones you need and then pressing a button in a toolbar provided. See Fig. above. Curvilinear transformations quality depends on "Performance" panel settings.

7.1.5 Warp



Tool to morph and warp slice contents, i.e. to adjust contents geometry. Available for slices in output map only. See Fig. above.

There are 4 operational modes available:

↖ — Select and modify corrections made.

↑ — Add a correction, based on an anchor point.

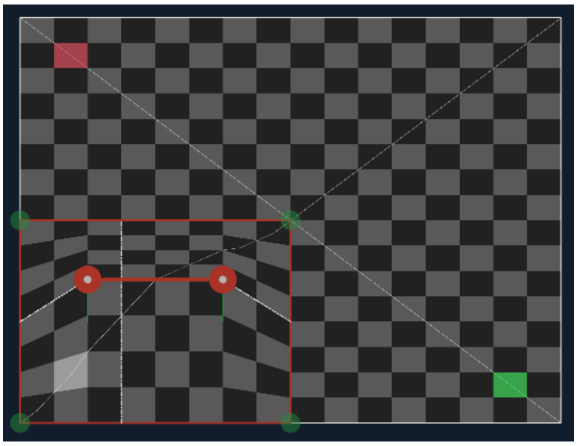
↑ — Add a correction, based on two anchor points, i.e. a line.

↑ — Add a correction based on four anchor points, i.e. a rectangle.

And also following actions:

☒ — Remove current correction.

☒ — Remove all corrections.

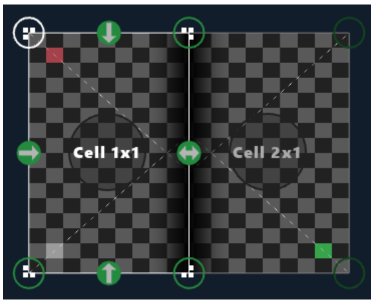


For every correction you can set elasticity area boundaries. There will be no changes made outside the area. The boundaries are highlighted in orange and match slice edges by default. To move the frame around, drag it by green dots. See Fig. above.

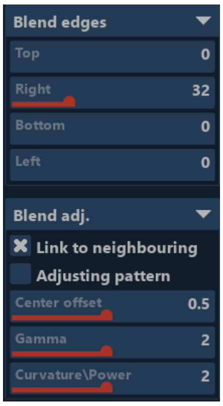
Curvilinear transformation quality (number of nodes) depends on slice quality.

For any modification you can set "Smoothness" parameter to define smoothness factor of transformation. Use arrow keys on your keyboard to move anchor points; to switch between anchor points use "<" and ">" keys.

7.1.6 Blend



Tool to blend edges of slices. Available in both input and output maps. All features of the tool available from its "Properties" panel. Key parameters of the tool are width of areas with alpha-gradient for every side of a quadrangle. You can also set blending width on canvas: click on a green dot corresponding to the edge you need, and, while holding a left button pressed, drag it to set the desired width. See Fig. above.



Blend edges

Top 0

Right 32

Bottom 0

Left 0

Blend adj.

☒ Link to neighbouring

□ Adjusting pattern

Center offset 0.5

Gamma 2

Curvature\Power 2

In Blend adj. block you will find additional settings. See Fig. above.

Link to neighboring — It defines special behaviour to be applied when blending width changes. It works as follows: if an edge being changed is in line with an edge of another slice, the same blending width will be applied to both slices. And input maps for both slices will be changed to form correct overlapping.

Adjusting pattern — Check it to enable high-contrast pattern on every side's blending area to make seaming easier.

Center offset, Gamma, Curvature/Power — Those are parameters to correct alpha-gradient calculation function. You don't usually need to change the settings.

7.1.7 Color

Tool to correct color, levels, brightness, contrast and other slice contents parameters. Available for slices in output map.

The tool parameters are as follows: Opacity, Gamma, Red 1/Gamma, Green 1/Gamma, Blue 1/Gamma, Saturation, Contrast, Bright, White level, Black level.

7.1.8 Vector

Tool to create vector masks, and to add and remove nodes. Available for masks in both input and output maps. There are 3 buttons in additional toolbar: ⊞ — Add a default mask. Its outline will match the slice outline. It comes in handy when you need to adjust the outline of the slice. ⊕ — Click the button to set up a new mask outline. Double-click to wrap up outline entering. ⊗ — Remove all vertices selected.

You can also use the tool to add a vertex to an edge by simply clicking on it, or to remove a vertex by double-clicking on the node.

7.1.9 Bitmap

Tool to create masks from images (bitmap). Available for masks in both input and output maps. There are five buttons in additional toolbar:

— Select an image and upload it as a mask.

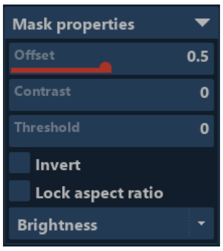
— Add dynamic mask based on data provided by Kinect sensor. Mask will be created from depth map data. Settings are provided. See below.

— Remove current bitmap mask.

— Fit to working area; first click to fit a mask in with aspect ratio ignored, second click to fit it in with the ratio kept intact.

— Fit bitmap mask to current slice. Works only in input map and for slices with “Rectangle input” option checked.

Bitmap mask positioning is always absolute and is not reliant on a slice position and geometry. To shape a slice with a bitmap mask, apply the mask in input map. By doing this, you will be able to work with a slice shaped by the mask.



Mask properties

Offset 0.5

Contrast 0

Threshold 0

☐ Invert

☐ Lock aspect ratio

Brightness

There are also following parameters, see Fig. above:

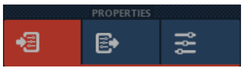
Offset — Change mask “brightness”. Contrast — Change mask “contrast”. Threshold — Change level of full opacity. Invert — Invert mask. Lock aspect ratio — Lock aspect ratio of a mask on editing.

Next, there’s a channel select box. You can choose a base channel to create a mask from “Alpha”, “Red”, “Green”, “Blue”, “Brightness”. If there is an alpha channel in a file, it will be used by default. If there’s no alpha available, “Brightness” will be chosen.

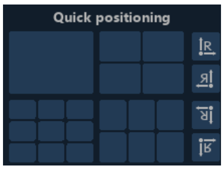
7.2 “Properties” panel

There are parameters of different types in “Properties" panel. Every numeric value is represented with a “Slider” graphical element. For all sliders, you can double-click on it to open a text box and edit its value (only for horizontal sliders). You can also change its value with mouse wheel (press CTRL to lower increment size). To reset a slider to its default value, right-click on it.

Input map and Output map properties



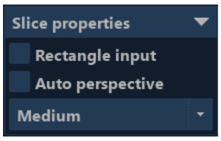
“Properties” panel is divided into three pages, first and second of these are input and output settings, i.e. current element coordinates. Both input and output maps settings pages look the same but either control only the corresponding map.



Besides sliders standing for axes of every vertex, there’s a Quick Positioning block used to set parameters quickly. The settings are divided into two conceptual blocks: geometry modification buttons on the left and orientation modification column on the right. Four possible output configurations are shown on the left. Click on a cell to apply the geometry. You can also click and drag one cell to another to select various cells. It’s a good idea, if you want to use more complex geometry configurations. Also, if Multi-Display is in use and output to more than one monitor is enabled, a square with a set-up matching your monitor configuration will be displayed in output properties. It allows you to quickly output your slice to a desired monitor with great level of precision.

Current element and tool properties

There are current element and tool properties shown on the third page. For MAPIO LITE, media sources properties and controls will also be displayed here.

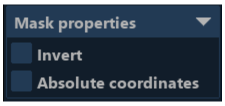


Slice properties (see Fig. above):

Rectangle input — Check if you want to lock slice geometry in input map as rectangle. It’s not uncommon for an input map to be a rectangle, and restricting it to it with the checkbox helps to edit its geometry.

Auto perspective — The settings changes slice calculation algorithm in output map, assuming a resulting projected image should be rectangular, and perspective correction can be applied to the slice content automatically.

Low, Medium, High — Slice quality selector. It affects geometry quality in Warp and Bezier, and also while performing extreme transformations of the slice.



Vector mask properties (see Fig. above):

Invert — Invert current vector mask.

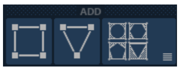
Absolute coordinates — Absolute coordinates for current outline. Use to unbind mask position and geometry from a slice.

Mask properties are displayed when mask editing mode is activated and there is at least a vector mask outline (except where Bitmap tool is being used).

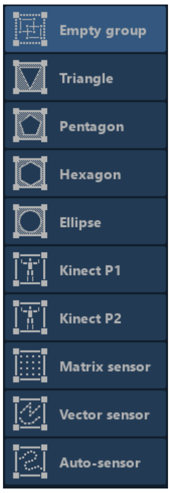
7.3 Working with list of elements (slices)

7.3.1 Adding and Removing elements

Adding elements



There are three buttons grouped at the top of a slice list. The buttons are used to add elements to your project (see Fig. above). The first button (I) adds a quadrangle, the second one adds a triangle, and the last button opens a menu with other elements available.



In the menu you will find the following elements (see Fig. above):

Empty group — Add an empty group to your project. You can fill it by moving elements to the group.

Triangle, Pentagon, Hexagon, Ellipse — Add quadrangular slices with an initialized vector mask of a certain shape to an input map.

Kinect P1/Kinect P2 — Add a fixed group of slices with their position defined by data provided by a Kinect sensor.

Matrix sensor, Vector sensor, Auto-sensor — Add elements to set data capture method and geometry used for pixel mapping. For more information, please refer to Section 11.

Removing elements

To remove selected slices, click ☒ button placed at the bottom of the elements list. When "OK?" button appears on the right, press it to confirm.

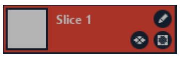
7.3.2 Order Changing

To change order of elements in your project, you can either use "Reorder/Move up/Move down" functionality from "Edit" menu, or press Ctrl+Shift+PageUp/PageDown keys, or drag elements with your mouse.

7.3.3 Grouping

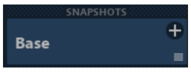
You can group elements within a project to make it more convenient to work with sets of slices. It makes geometry of the slices dependent on geometry of their respective groups. To create a group, you can select two or more slices (please note, they should be on the same tier of group hierarchy) and select "Group" in "Edit" menu, or press CTRL+G. You can also add an empty group and drag desired slices there.

7.3.4 Element properties. Rename



Any element of a project can be renamed ✎. Press Enter to confirm the name typed or ESC to cancel changes you made. You can also change element visibility. ◉ — Hide element from an editor; if the editor is full of elements, you may find it useful to hide some of them, while also keeping the elements in resulting output. ⊗ — Leave element out of render, i.e. hide from resulting output.

7.4 Snapshots



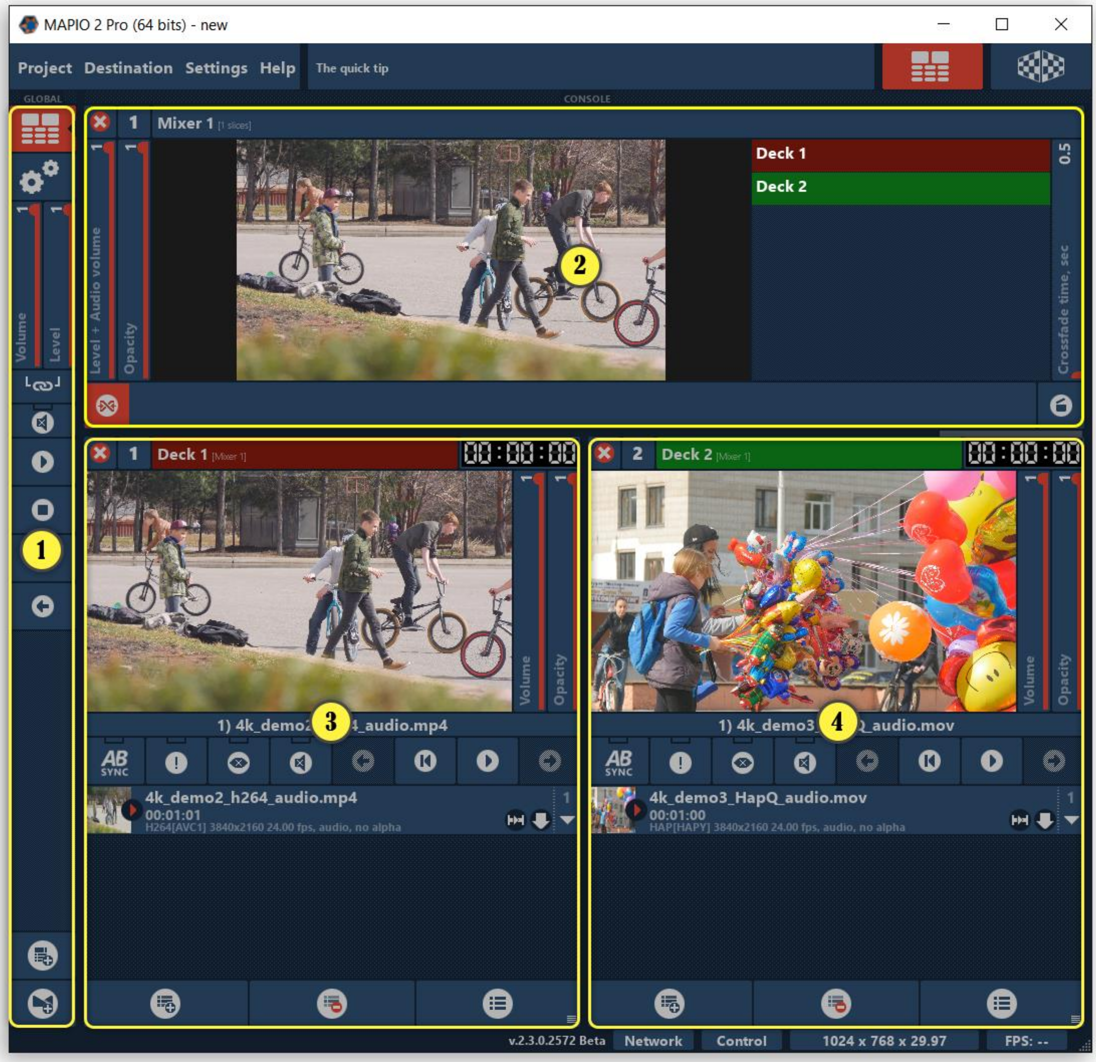
MAPIO software provides a tool to save mapping configurations as snapshots and switch between the snapshots instantly. A list of available snapshots is accessible from a top right menu (see Fig. above). By default, there's a Base snapshot (permanent, you can't delete this). To add a snapshot, click ⊕, enter its name and confirm action by pressing Enter. New snapshot will be created and initialized with current configuration of your project. Every time you edit the project it will be stored to current snapshot automatically. To remove a snapshot, click ⊗ in the snapshot list.

Input and output configurations, the project's elements displaying statuses, sources attached to slices, modifications made with Bezier, Warp, Blend and Color tools, and all masks will be stored in a snapshot.

Please note, console state as well as project output settings, including pixel-mapping, network configuration and control assignment settings will NOT be stored in a snapshot.

8 Console

8.1 Overview



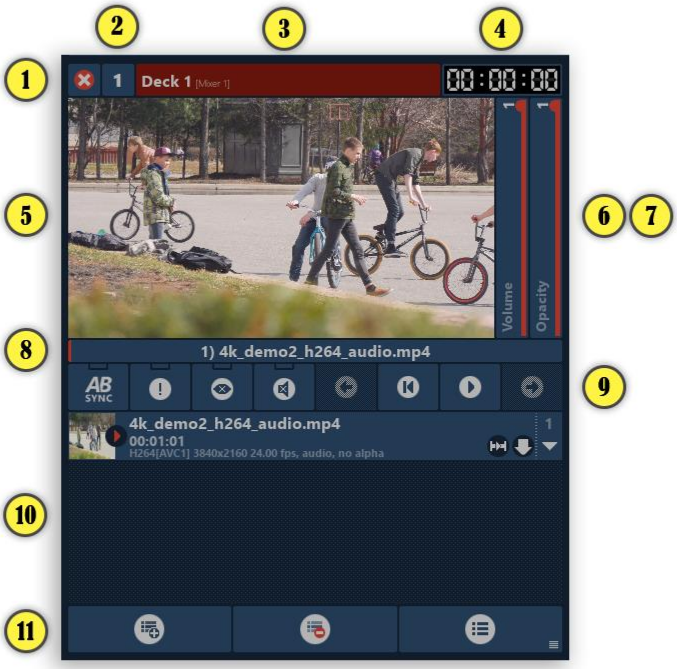
1) Left sidebar contains project management controls
2) Mixer block (you can add up to 4 mixers)
3) Deck block 1 (you can add up to 8 decks)
4) Deck block 2

Console is available in MAPIO **PRO** edition only.

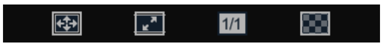
8.2 Left sidebar

From top to bottom (see Fig. above): first two buttons are used to switch between console page and settings page (for details, see below). Then, there are volume level and video level (video output opacity/brightness) sliders of your project. They can be linked by clicking on a linking chain button 🔗 just below them. 🔇 button — Mute the project. ▶ button — Play current file for every deck. ⏸ button — Pause playback in all decks. Then, there are two buttons highlighted with colour to play/pause only selected elements in decks. Then, ➡ — Go to next video in all decks. ⬅ — Go to previous video in all decks. There are two more buttons at the bottom of the bar used to add a new Deck and a new Mixer, respectively.

8.3 Deck



1) Delete Deck button. You should confirm the action, when you click on this.
2) Deck number (used in OSC addresses). Click on this to open a menu. The menu contains: Attach to slices — Attach the deck as a source to a current slice or selected slice(s); Select slices — If the desk is attached to any slices as their source, upon choosing the option, the slices will be selected (double-click on the deck name to the same effect); Bind in network… — Add/remove the deck from (network) synchronization for selected slave instance(s); Move to previous, Move to next, Move to begin, Move to end — Change order of the deck.
3) Deck name and information on where it's being used. A button to edit the name is displayed on mouse over. If the deck is used in a mixer, red background means it's on the air, i.e. its video goes live to the mixer's output. Green background means its content is not on the air, but the deck will be next in the mix.
4) Timecode of the current video. Left-click to toggle modes: elapsed time (in white) and remaining time (in red)
5) Preview. Left-click to open/close window mode settings. Left to right: fit to window (ignore aspect ratio); fit to window (keep aspect ratio), double/half the size; enable chequered pattern in background.

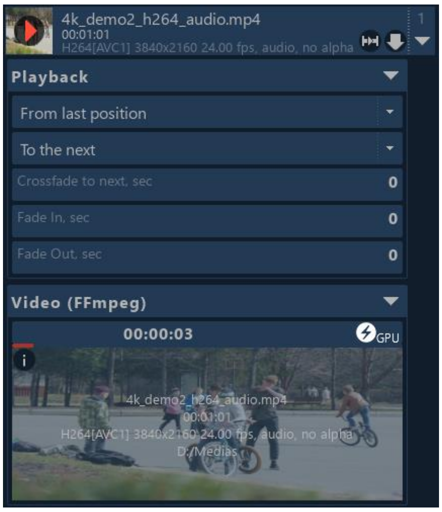


6) Volume slider of the deck.
7) Opacity slider of the deck.
8) Index and name of a source in use, playback duration timeline (if applicable). Left-click (if available for the source) to open loop playback settings (see below) and additional controls. Right-click to FF/rewind video/audio file.
9) Settings and controls of the deck. Left to right: Clock bus selection (for more information see below), Solo mode (only video and audio from the deck will be sent to output), Mute video, Mute sound, Go to previous video, Rewind, Play/Pause, Go to next video.
10) List of sources (videos and other media sources). Every rectangular playlist item contains following information: video/media source name (editable); video clip duration (if there's a loop, its duration is separated with a slash character); information (for video clips, codecs, resolution, FPS, audio and alpha statuses); playback modes icons. Using a mouse, you can change order of video clips, drag them between decks, or even copy a video (press and hold CTRL key to perform the latter).

11) Three buttons: Add sources menu, Remove sources, and Save/Load playlist menu + set dynamic external playlist (for details, see below).

List of sources

To add elements, either use a menu, or drag them. Using a mouse, you can also change order of the list and drag elements between decks.



4k_demo2_h264_audio.mp4
00:01:01
H264[AVC1] 3840x2160 24.00 fps, audio, no alpha

1

Playback

From last position

To the next

Crossfade to next, sec 0

Fade In, sec 0

Fade Out, sec 0

Video (FFmpeg)

00:00:03

GPU

4k_demo2_h264_audio.mp4
00:01:01
H264[AVC1] 3840x2160 24.00 fps, audio, no alpha
D:/Medias

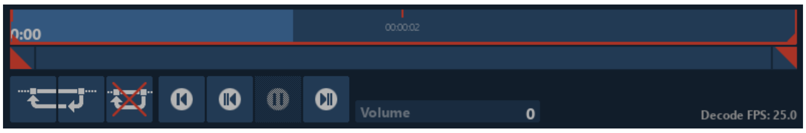
You can open settings of any element by clicking on a triangle on the right. For every media type Playback group of properties is available. It is usually displayed above others and contains the following:

- Resume mode — Playback starting point mode. The parameter is available only for seekable sources, video and audio files. It can be set to: always from the start, resume from the last stop, choose randomly.
- Behavior at the end — Action to be performed when playback reaches the end: Once (stop); To the next; To the previous; Repeat; To any other; No play (skip); To the first.
- Duration — Duration in seconds. The parameter is useful only for media sources with no duration provided.
- Crossfade — Duration of cross-fade (smooth mixing technique) in seconds with the next element on the list or the next user selected.
- Fade In — Duration of fade-in from black in seconds at the beginning.
- Fade Out — Duration of fade-out to black in seconds at the end.

There are a group of video file properties (see Fig. above), containing timecode and codec operation mode status (⊗CPU — No acceleration, i.e. codec uses CPU resources; ⚡GPU — GPU accelerated), frame preview, information on file format, ℹ button to show full information on the file (formatted as MediaInfo). Mouse click on preview or timecode to open Loop editor.

Here you can also find various properties specific to different media sources and grouped separately.

Loop Editor



Loop editor consists of three blocks, see Fig. above:

1) Upper block — Video timeline, and starting and end of a loop (marked in orange). To set these, drag either starting or end of a loop to a desired position on a timeline. You can also drag either beginning or end marked with orange vertical lines. To skip video to a certain position, either use right-click, or left-click with CTRL key held, or move your mouse pointer to the position while holding both CTRL and ALT keys.
2) Middle block — zooming and scrolling bar combined in one element. To zoom in/out, move your mouse pointer over the bar or the timeline and use your mouse wheel. To achieve this, you can also drag ends of a scroller.
3) Additional controls: ↰ — Set current frame as starting point of a loop; ↲ — Set current frame as end of a loop; ☒ — Reset loop, i.e. whole file will be used; ⏮ — Rewind to the start of a file/loop; ⏪ — Go to the previous frame; ▶ or ⏸ — Play/Pause; ⏭ — Go to the next frame; Volume level slider, if audio track is available; on the right, actual decoding process FPS of the file is displayed.

The loop editor also works for audio files but with some elements disabled.

Synchronization

This section covers local timecode synchronization and synchronous management of console decks, in other words, index of a source being played and its timecode will be synchronised for selected decks. This feature is useful in case you want separate audio files from video files and then play them synchronously. You can also slice a file and synchronise its playback, to perform operations of reading and decoding files concurrently and, thus, to improve bandwidth in case a codec doesn't support multi-threaded decoding.

Local synchronization allows to play video files with "frame-to-frame" level of precision, if their FPS match each other.

To put synchronous playback into action for several decks, you need to set local synchronization bus to either "A" or "B" for every of them using AB SYNC button. Press the button once to enable "A" bus AB SYNC for the deck, press it again to switch to "B" bus AB SYNC. To disable synchronization (i.e. to revert the deck to its initial state), press the button once again.

Every bus needs a master to work as its command/timecode source. For a deck, master will be selected automatically, and its synchronization bus selection button will be marked with an orange letter. Every time you make any changes to a deck (i.e. configure its settings, start or stop playback, or switch between videos), master deck switches to the deck you modify. Note that video playback parameters set in slave decks will be ignored.

Since there are two synchronization buses (A and B) available, you can create two groups of synchronised decks.

Regular and dynamic playlists

MAPIO allows to save and load playlist contents from a separate file (see the first Fig. of the section for “Save playlist”, “Load playlist” options in drop-down menu listed as #11). It allows you to switch between several media contents sets without changing mapping configuration. Playlists are stored in binary format, and files contain all playlist elements settings.

Additionally, there are tools to create dynamic playlists (‘Set dynamic playlist’, ‘Reset dynamic playlist’), using human-readable plain text file format. After dynamic playlist is loaded/set, a deck switches to special read-only mode, and MAPIO starts to monitor changes made to its file in real-time and to update playlist accordingly. Currently, the mode doesn’t support individual settings for playlist elements, and you can only set default settings for all playlists (see Section 8.6).

To store playlist data in dynamic playlist files, we use strings formatted as following:

<type>|*<parameter>*|*<file>*, where

<type> — File type abbreviated with a single uppercase letter. (“V” is used for video files, “A” is for audio files, and “I” means the file is an image).

<parameter> — Additional parameter. Currently, only image exposure duration (in milliseconds) is supported as a parameter, it defaults to 10,000ms. Otherwise, the field should be left empty.

<file> — Absolute or relative path to a media file of a said type; with the latter path is relative to the playlist file location.

Strings with no source type set are permittable. By default, it implies the source with no type set is a video file. Lines beginning with “;” or “#” characters are comments and will be ignored.

You can merge playlists into one and use the resulted file with several decks. To specify a deck, add a line with its name in square brackets (i.e. [DECK1], [DECK2], etc., a number corresponds to deck number) just above a group of files.

Example 1, with no decks specified:

```
;Example of dynamic playlists
I|6000|c:\imagefile.png
V||localvideofile.avi
C:\absolute\located\file.mov
```

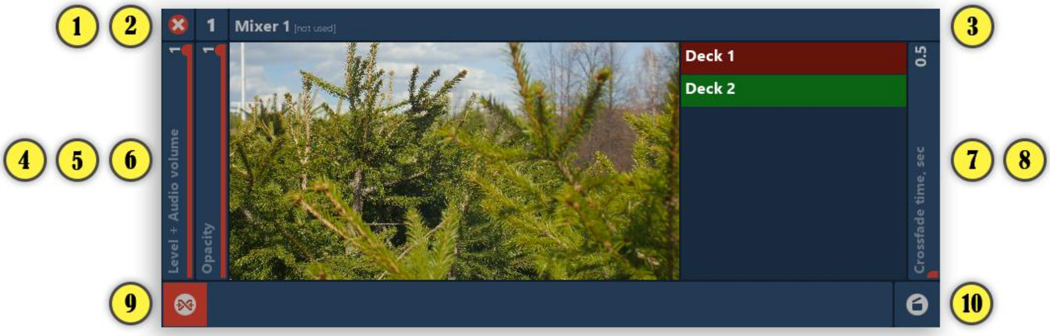
Example 2, with two decks specified:

```
;Example of dynamic playlists
;Playlist of deck 1
[DECK1]
I||c:\imagefile.png
C:\absolute\located\videofile1.mov

;Playlist of deck 2
[DECK2]
A||localaudiofile.wav
A||C:\absolute\located\file.mp3
```

8.4 Mixer

Mixer allows you to switch smoothly between signals received from several decks, and to add some automation.



1) "Delete Mixer" button. You should confirm the action, when you click on this.
2) Mixer number (used in OSC addresses). Click on this to open a menu. The menu contains:
   Attach to slices — Attach the mixer as a source to a current slice or selected slice(s);
   Select slices — If the mixer is attached to any slices as their source, upon choosing the option, the slices will be selected (double-click on the mixer's name to the same effect);
   Auto-stop the deck — Auto-stop the deck currently on the air after mixing is complete;
   Auto-start the deck — Auto-start the next deck when mixing starts;
   Auto-crossfade — Auto-start crossfade to the next deck when video playback from current deck ends;
   Move to previous, Move to next, Move to beginning, Move to end — Change order of the mixer.
3) Mixer name and information on where it's being used. A button to edit the name is displayed on mouseover.
4) Combined slider for volume level and image brightness (fade to black).
5) Opacity slider. Fade to transparent.
6) Preview window. For details, please refer to Deck documentation.
7) List of available decks and deck selector. The deck on the air marked in red, the next deck to go live via crossfade is marked in green.
8) Auto-mixing speed.
9) Crossfader — drag the handle from left to right to switch to the next deck.
10) Auto-crossfader launching button.

8.5 Workflow example

By default, the following console control keys are defined:

Space — Launch auto-crossfader for Mixer 1.

Left / Right — Select next deck for Mixer 1.

Up / Down — Select video in the next deck of Mixer 1.

Home / End – Select the first/last video in the next deck of Mixer 1.

Enter — Switch between Select/Play/Pause modes of selected video in the next deck of Mixer 1.

Backspace — Rewind current video in the next deck of Mixer 1.

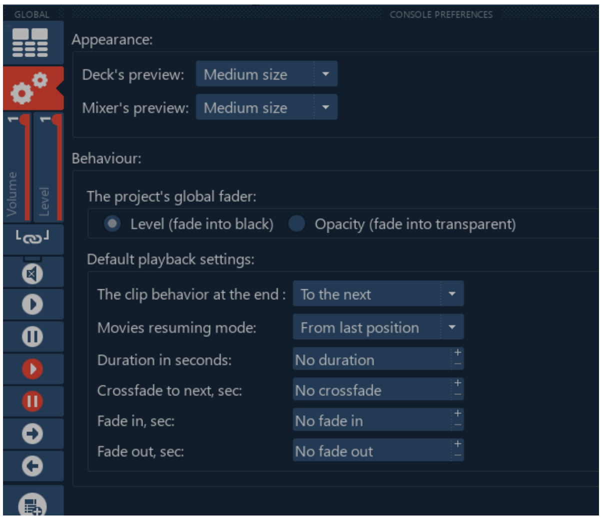
To reassign hotkeys, please refer to Section 14, in subsections */mapio/mixer1 and /mapio/mixer1/nextdeck/.*

The hotkeys set allows to perform the most of live montage actions you will usually need. An exemplary workflow could be as follows:

- Select next deck with **Left / Right** arrows (for three or more decks).
- Select a video with **Up / Down** arrows.
- Press **Enter** to set it as "current video", press again to play it, and again to pause the video. Press **Backspace** to rewind the video (if the media source selected supports the action).
- At the right time, press **Space** to launch auto-crossfade and **Enter** to play the video. Or, alternatively, enable **Auto-start the deck** mixer option to start the deck automatically.
- Repeat the procedure for another deck.

8.6 Settings

Click on the second from the top button in the left sidebar to open a panel where you can find appearance and behaviour settings for mixers, decks, etc.:



Appearance:

Deck's preview: Medium size

Mixer's preview: Medium size

Behaviour:

The project's global fader:

● Level (fade into black) ○ Opacity (fade into transparent)

Default playback settings:

The clip behavior at the end : To the next

Movies resuming mode: From last position

Duration in seconds: No duration

Crossfade to next, sec: No crossfade

Fade in, sec: No fade in

Fade out, sec: No fade out

Appearance block: Decks and mixers' preview windows size.

Behaviour block: Switch behaviour for Level/Opacity slider in the left sidebar. Default playback properties group settings. Properties will be applied to new sources you will add to decks.

9 Sources

9.1 Image

A source represents a static image and is loaded as a file. The following image formats are supported: JPEG, PNG, BMP, GIF, TIFF, WEBP, SVG, ICO, TGA, WBMP. There's only a setting available to the source — "Smooth". It allows to improve quality of large high-resolution images when you output them with significantly lower resolutions. In other words, the setting enables MIPMAP [wikipedia.org/wiki/Mipmap] generation to the image.

9.2 Video

A source represents a video file in one of the following formats: AVI, MOV, MP4, M4V, WMV, DIVX, MP2, MPEG, FLV, MKV, WEBM, TS, GIF and other containers, formats and codecs supported by FFmpeg, including DXV and HAP. If you cannot find a file in "Select File" dialogue window, please check "All Files (*.*)" to reset filters.

For some GPU, codec and video file parameters combinations, your GPU capabilities may significantly improve and accelerate decoding. For more information on decoding statuses, please refer to Section 8.3 on p.26.

9.3 Audio

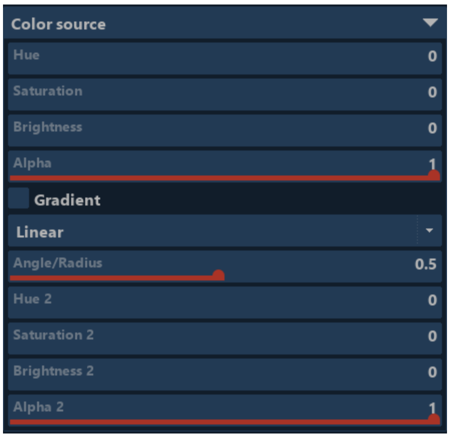
An audiofile-type source could be one of the following: WAV, MP3, OGG, FLAC, APE, AC3, RA, WMA, M4A and other containers, formats and codecs supported by FFmpeg. If you cannot find a file in Select File dialogue window, please check "All Files (*.*)" to reset filters.

Audio source generates no visual data. A slice will be filled with black if a deck with an audio file is in use.

The source is not available in MAPIO LITE, because audio sources can be added and used with decks only.

9.4 Color

A source represents a solid fill/gradient generator.



Color source

Hue 0

Saturation 0

Brightness 0

Alpha 1

□ Gradient

Linear

Angle/Radius 0.5

Hue 2 0

Saturation 2 0

Brightness 2 0

Alpha 2 1

Colours and shapes of gradients can be set with the following parameters (see Fig. above):

- Primary colour is defined with four sliders: Hue, Saturation, Brightness, Alpha.
- Check "Gradient" box to enable gradient generation.
- Then, select your gradient type: Linear, Radial, Conical.

- Angle/Radius slider to customize gradient.
- Additional colour to form a gradient: Hue 2, Saturation 2, Brightness 2, Alpha 2.

9.5 Spout

A submenu with a list of available sources compatible with Spout [spout.zeal.co].

Spout is a protocol designated for video sharing between Windows applications. If supported by your hardware, it communicates with zero-copying, i.e. with almost zero memory/processor utilisation and minimal latency. Spout works in both DX9 and DX11 modes. To select mode, run SpoutDXmode.exe available from Spout installation package.

In case you use more than one GPU (e.g. an integrated and a dedicated video card), please make sure, both "signal source" and "signal receiver" applications are run on the same GPU.

You can find resolution of video frames received, and set desired refresh rate in the source properties.

9.6 NDI

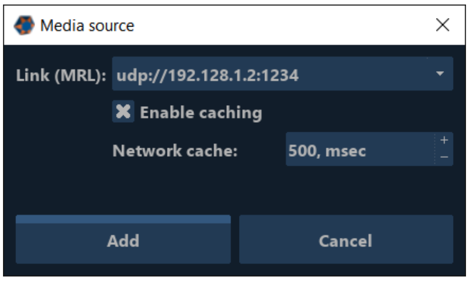
A submenu with a list of available sources compatible with NDI [www.newtek.com/ndi/].

NDI is a standard for audio and video streaming communication between applications and devices connected to Ethernet. In MAPIO, you cannot set any additional properties for NDI sources. Please note, MAPIO does NOT support sending and receiving audio via NDI.

9.7 QML

In QML submenu you can find a list of available sketches. For more details, see Section 16.

9.8 Capture devices



Media source

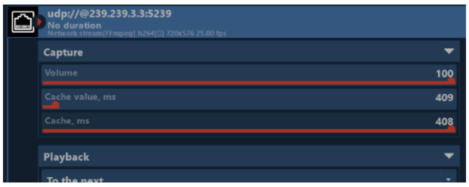
Link (MRL): udp://192.128.1.2:1234

☒ Enable caching

Network cache: 500, msec

In "Capture devices" menu, you will find a list of video capture devices available in your system. The list is divided by two based on device interaction implementation: "Base on FFmpeg" and "Quick DirectShow". "Base on FFmpeg" consists of following device types: DirectShow devices installed in the system; Blackmagic DeckLink devices; screen capture split by monitors; and Network URL… — video stream capture from network by URL specified (FFmpeg supports numerous protocols). "Quick DirectShow" consists of DirectShow devices to be run with default settings. When choosing from "Base on Ffmpeg" list, you can configure some properties, depending on the device type.

Available device types:

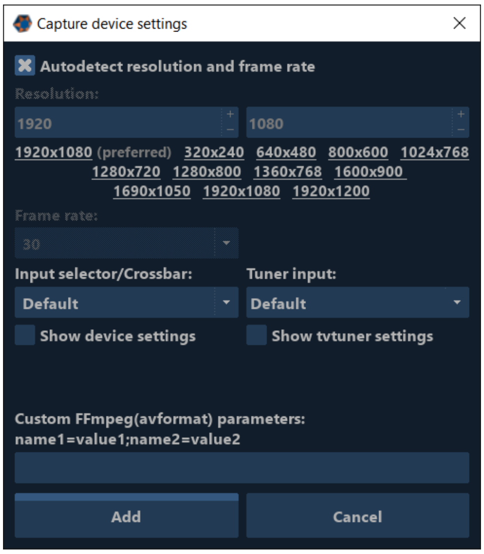


udp://@239.239.3.3:5239
No duration
Network stream[FFmpeg] h264[] 720x576 25.00 fps

Capture
Volume 100
Cache value, ms 409
Cache, ms 408

Playback
To the next

Network URL — Read and output video stream accessed by URL. FFmpeg supports various networking and local protocols like IP-TV broadcasting, IP-Web cameras and other devices. Among available protocols are file, ftp, gopher, hls, http, https, mmsh, mmst, pipe, rtmp, rtmps, rtmpt, rtmpts, rtp, srtp, subfile, tcp, tls, udp, udplite, etc. For a source of Network URL type, there are following properties you can configure: volume level slider (if applicable), cache buffer size slider (in milliseconds) and another slider to display current cache buffer size (noneditable).



☒ Autodetect resolution and frame rate

Resolution:

1920 1080

1920x1080 (preferred) 320x240 640x480 800x600 1024x768 1280x720 1280x800 1360x768 1600x900 1690x1050 1920x1080 1920x1200

Frame rate:

30

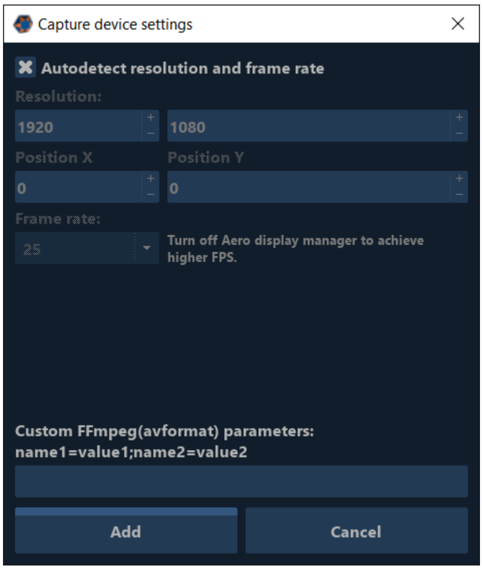
Input selector/Crossbar: Default

Tuner input: Default

□ Show device settings □ Show tvtuner settings

Custom FFmpeg(avformat) parameters:
name1=value1;name2=value2

You can see the dialog for adding DirectShow capture device in the figure above. Users can set preferable resolution and frame rate values. If the device doesn't support them, the closest available parameters will be chosen. For tuner type devices, you can also choose from inputs available, and, by checking "Show device settings" and "Show tv tuner settings", open their native Settings windows (if there's any available for the device).



☒ Autodetect resolution and frame rate

Resolution:

1920 1080

Position X 0

Position Y 0

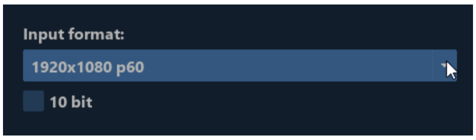
Frame rate:

25

Turn off Aero display manager to achieve higher FPS.

Custom FFmpeg(avformat) parameters:
name1=value1;name2=value2

You can see the dialog for adding screen capture in the figure above. By default, it captures the whole screen of a monitor selected. You can also set capture area geometry, its size, position and frame rate. Frame rate may be limited by system performance. For those using Windows 7, we suggest disabling Aero desktop theme. Additionally, to optimize performance, you can downsize the area you want to capture.



Input format:

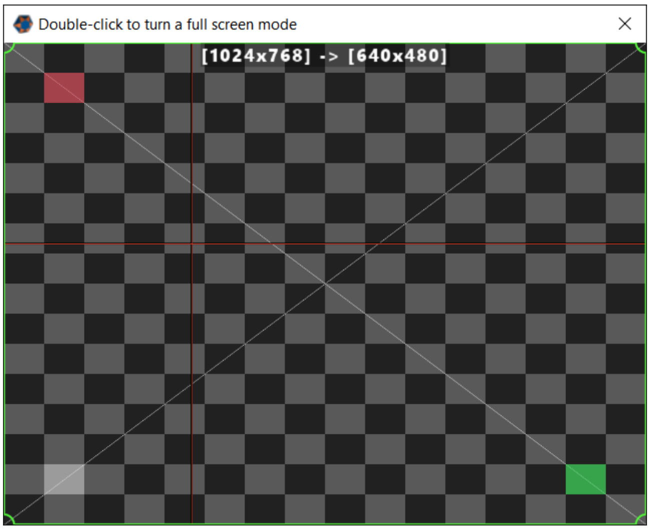
1920x1080 p60

□ 10 bit

You can see DeckLink device settings in the figure above. The dialogue allows you to select from capture video formats available and to enable 10bit mode. Other parameters are set to default and can be accessed and configured via “official” settings panel of the device.

10 Output destination

10.1 Display

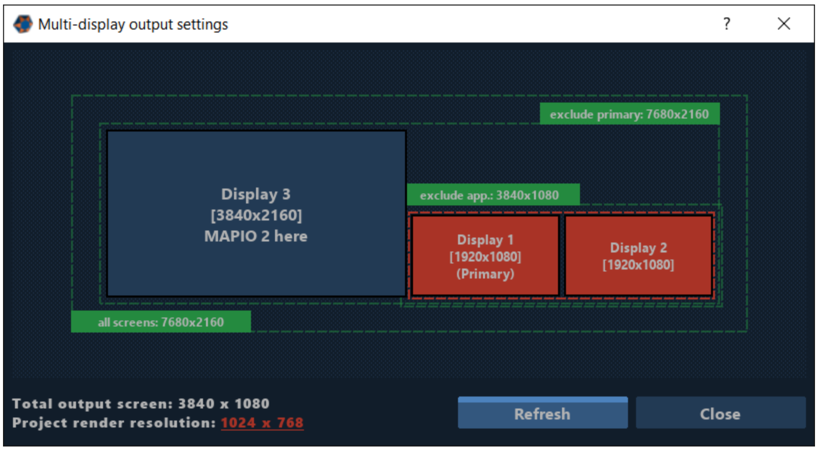


Double-click to turn a full screen mode

[1024x768] -> [640x480]

This is a simple output window (see Fig. above) that can be maximized in one monitor by double-clicking inside the window. You can use same double-clicking to restore the size of the window if the “Show” mode is on. Otherwise you should first turn the “Show” mode off by pressing CTRL+B. By default, in the window you will see different auxiliary elements, such as border colored green, current slice vertices, yellow lines whose cross marks the position of the mouse in the editor, etc. Two resolutions are shown above as well: the first one is the resolution of the project, and the second one is the resolution of the window or the monitor where it is deployed. These and other auxiliary elements can be either disabled or enabled in the “View/Output Overlay” menu. Furthermore, all the auxiliary elements can be automatically hidden in the “Show” mode which is enabled in the “Settings” menu.

10.2 PRO | Multi-Display



Multi-display output settings

Total output screen: 3840 x 1080
Project render resolution: 1024 x 768

By enabling “Multi-Display” output you will open additional setting window (see Fig. above). Here all the monitors / projectors available for the system are shown in the configuration of mutual arrangement that is identical to that of either GPU drivers or OS settings. It is not possible to change the arrangement of monitors with MAPIO tools. In the rectangles representing the monitor, enter the sequence number of the monitor (normally, it coincides with the numbers in the system settings), as well as the monitor resolution, “Primary” marks for the main monitor (where OS main taskbar is located), and “MAPIO 2 is here” for the monitor where MAPIO window is located.

To enable the output, click on the respective monitor. You can select several monitors if you observe the following rule: the selected monitors must be positioned in a group that can be entirely covered by a rectangle without voids.

When the output is set up the “Multi-display” output setting window can be closed using “Close” button or cross button.

The output can be temporarily disabled by double-clicking on it. The “Multi-Display” output will restore after project’s upload or reopening. This option can be used to automatically open the output on a display with MAPIO (i.e. “on top” MAPIO). To do that, enable the output, then temporarily close it by double-clicking and save the project as is. At the next project uploading, the output will restore.

When the “Show” mode is on you cannot use double-clicking to close the window. First, you need to turn the “Show” mode off by pressing CTRL+B.

For more details on auxiliary graphical elements in the output window, please refer to “Display” section above.

10.3 NDI

This functionality allows to output project rendering results on a NDI data exchange system. Frame resolution will be equal to project resolution, frame frequency will be equal to actual rendering frame frequency. Sound transmission is not supported. NDI output can be combined with other destinations.

10.4 Spout

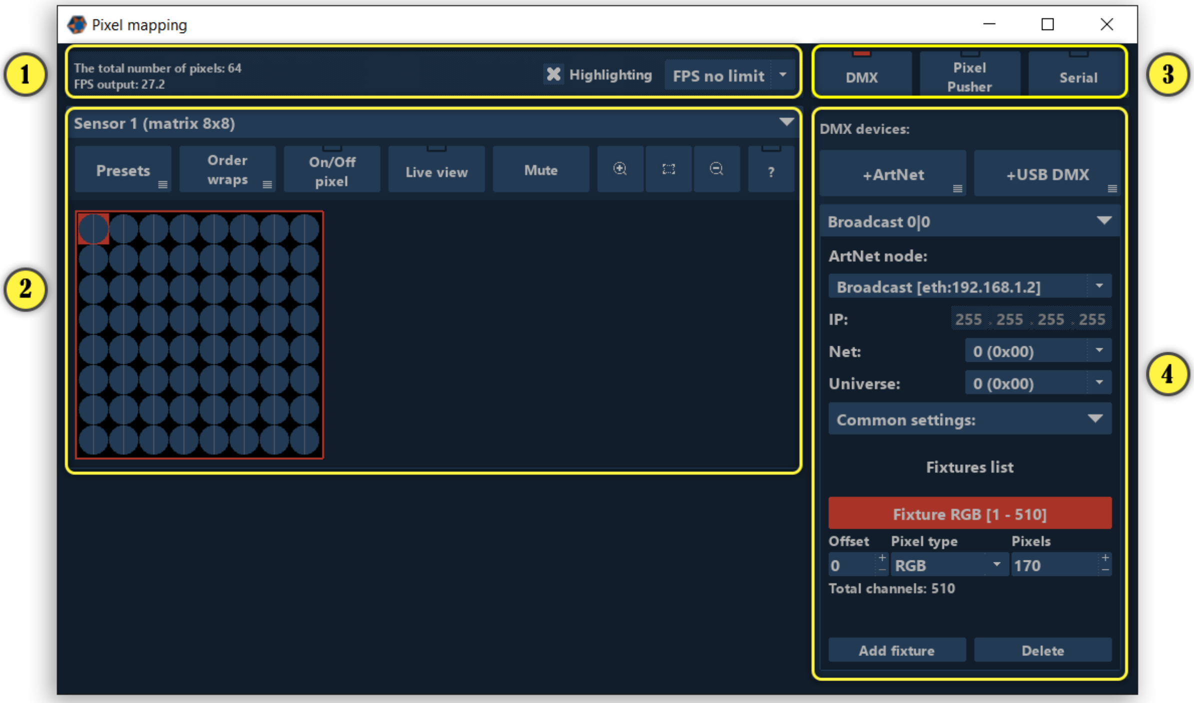
This functionality allows to output project rendering results on a Spout data exchange system. Frame resolution will be equal to project resolution, frame frequency will be equal to actual rendering frame frequency. Sound transmission is not supported. Spout output can be combined with other destinations.

10.5 PRO | Pixel-Mapping

This opens pixel-mapping window, which is a system that allows to output on LEDs and other pixel-mapping devices, such as PixelPusher controllers, DMX devices via ArtNET or any other devices via a serial port. Data for the listed devices is read in a special way from MAPIO resulting output by special project elements called “sensors”. Pixel-Mapping output can be combined with other destinations. For more details on Pixel-Mapping, please refer to Section 11.

11 PRO | Pixel-Mapping

11.1 Pixel-Mapping window: general info



The total number of pixels: 64
FPS output: 27.2

ArtNet node: Broadcast [eth:192.168.1.2]
IP: 255 . 255 . 255 . 255
Net: 0 (0x00)
Universe: 0 (0x00)

Offset	Pixel type	Pixels
0	RGB	170

Total channels: 510

1) Information and setting field. It contains information about total number of pixels and project's rendering frame frequency, as well as a window for enabling / disabling of "Highlighting" mode (LED-highlighting of the current elements pointed with the mouse), and a selector containing FPS limitation to output on devices. This selector contains the following values: "FPS unlimited", i.e. data is immediately sent to devices as soon as project data is updated; 1/2, 1/3, 1/4 — part of a project update frequency; 5, 10, 15, 20, 25, 30 — frame frequency fixed values.
2) Setting panel for linking of project pixels with device pixels, specifically with fixture location (i.e. pixel sets described by the address of starting pixel, and their quantities) within sensor's pixel array. Three sensor types will be displayed here: "Matrix", "Vector" and "Auto". By default, a project has no sensors, so there will be nothing here and you will need to create a sensor (see Section 7.3.1.).
3) Three panel-enabling buttons responsible for the following types of devices: DMX — devices based on DMX protocol, USB DMX controllers and ArtNET devices are supported; Pixel Pusher — devices used for outputting on LED pixels and based on a PixelPusher controller; Serial — allows to output on any devices that support serial ports as the means of data exchange, e. g. Arduino.
4) Panel with LED pixel output device settings.

11.2 Sensors

As project elements, sensors define methods and geometry for reading the data from project's resulting output to send them to pixels / LEDs. Sensors consist of sensitive points of different types and sizes that determine where and how the final value for a pixel will be calculated.

Sensors can occupy any position in the slice hierarchy, this hierarchy does not affect sensor functioning. It is convenient to place sensors in a separate group.

The sensitive points can be of the following types:

One pixel — the simplest and the quickest way to read data. One pixel in a specific position in a frame is used as data for a LED pixel. Sensor´s size parameter is ignored.

Average box — all pixels are read from a square whose size is specified in “Size of sensor”, and then averaged.

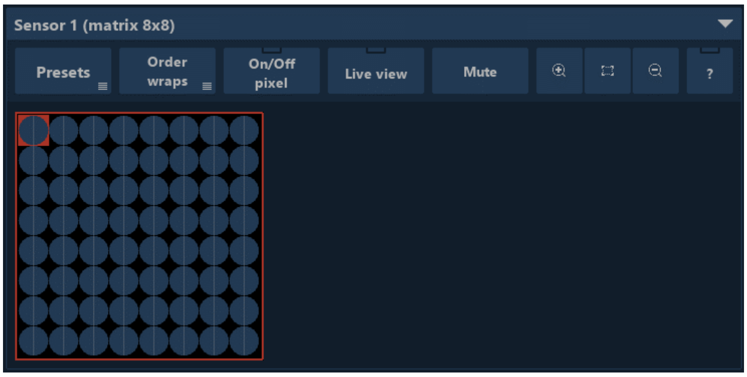
H line / V line — averaged value of pixels read from horizontal / vertical line of a specified size.

Diagonal cross / Cross — averaged value of pixels read from the lines of a specified size located in a diagonal or regular cross.

Sensor type determines the performance of a data preparation stage: the more points a sensitive element has, the more resources are required. Sensor´s complexity factor affects when you have more than 100,000pixels; if this number is less, the difference in the performance is insignificant. Since calculation is made on GPU, the reserve of performance is high.

The sensor type needed in a specific situation depends on the nature of an image and movement in it, as well as on the requirements for the image on LEDs. In most cases, the most general (Average Box) will do.

11.2.1 Matrix sensor



Matrix sensor is a matrix of sensitive elements of a specified number of columns and lines. The number of columns and lines is specified in the property bar of a project element in the main MAPIO window. There you can also specify the type and size of a sensitive element. The quantity of columns and lines cannot exceed 512.

After adding a sensor to the project “Pixel mapping” window will show a block containing settings to bind the sensor to the devices (see Fig. above).

At the top, there is a bar with tool buttons (from right to left):

- Menu to handle presets — Save and load a configuration to bind devices.
- Selection of the “Wrap order” mode — Rule for automatic positioning of the pixel chain in the matrix, sets the direction and the way of positioning. The icon of each rule clearly reflects its meaning.
- On / Off pixel — Enable/disable a special mode for handling pixels in the matrix. Enable this mode to disable (or enable) a pixel by double-clicking on it. Disabled pixels are skipped during automatic positioning of a chain. This allows to form more complex configurations basing on the matrix.
- Live view — Preview pixel values in the matrix. This mode overloads the system and should be disabled when you don’t need it.

- Mute — Exclude a sensor from the output. It is identical to the ⊗ button in project's element list.
- Change of the matrix size — Allows to either increase, reset or decrease matrix size.
- The button opens a simple hint on action sequence.

To position a Fixture / Strips (a current one, highlighted in a color on the device panel) make a left double click and then, holding the mouse button, drag to set the pixel chain automatic positioning direction. Horizontal or vertical positioning of LED lines, as well as the direction and ways to switch to a new line can be set in the "Order wraps" menu.

11.2.2 Vector sensor

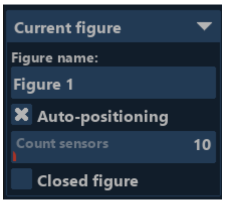
Vector sensor is a set of separate chains of sensitive elements located according to a pattern formed out of arbitrary rectilinear or curved lines.

After creating a Vector sensor, you need to set a pattern to position the sensors. For this purpose, in the OUTPUT – SLICE mode you can use the following buttons on the tool bar:

— Press to either enable entry mode for new vector primitives or add nodes in the existing primitives. To complete entering a figure and to start entering a new figure, make a double click; to cancel entry mode press this button again or use mouse right button.

— Delete current figure.

— Three buttons to switch the modes of the curved line for all nodes of a figure: the first one is for simple linear mode; the second one is for "Bezier" mode with free handles; the third one is for the mode where the angle between handles is fixed at 180 degrees.



Current figure

Figure name: Figure 1

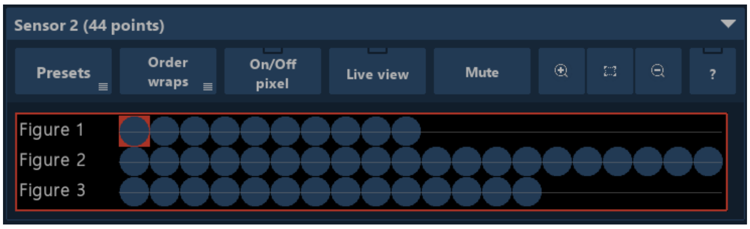
☒ Auto-positioning

Count sensors 10

□ Closed figure

Apart from the sensor element standard properties, on the left side of the property bar you can find properties of the current figure (see Fig. above).

In the figure properties you can set a contour name (max. 12 characters). "Auto-positioning" allows you to automatically position sensitive elements uniformly across the figure; quantity is set by the next slider called "Count sensors". If "Auto-positioning" mode is disabled, the sensitive elements are only positioned in figure's nodes. "Closed figure" allows you to automatically link the last point with the first point to get closed contour.



Sensor 2 (44 points)

After adding a sensor and one or more figures to the project, the "Pixel mapping" window will show a block containing settings to bind the sensor to the devices (see Fig. above).

At the top, there is a bar with tool buttons identical to the “Matrix sensor” line (see above). The only difference is that the “Order wraps” list is limited to only horizontal modes. The way of positioning for Fixtures is completely identical as well.

11.3 Working with Matrix and Vector sensors

When you have one or more sensors on the left and one or more output devices on the right, you need to bind sensitive points of the sensor to the LED pixels of the devices in the mapping window. To do so, make the following steps:

1) Turn on and set up the devices on the right, determine the right set of Fixtures in line with your devices.
2) Set up the right type of “Wrap order” for the sensors.
3) Select a Fixture of interest on the left.
4) Left-click on the first pixel location and, holding the button, drag in the necessary direction. (Please remember that depending on the “Wrap order” mode there will be either vertical or horizontal direction available.)
5) Repeat the steps 3-4 and 2-4 for all Fixtures.
6) The data are sent immediately.

You can bind several Fixtures to the same location in the sensor if identical data are implied.

Avoid using the same Fixture in different sensors since in this case two different sets of data will be sent for one Fixture that could lead to an unexpected result. However, this option can be useful if you control the data submission (using either “Mute” button or hide-from-render button placed on each project element) so that pixel data enters the Fixture from only one sensor at once.

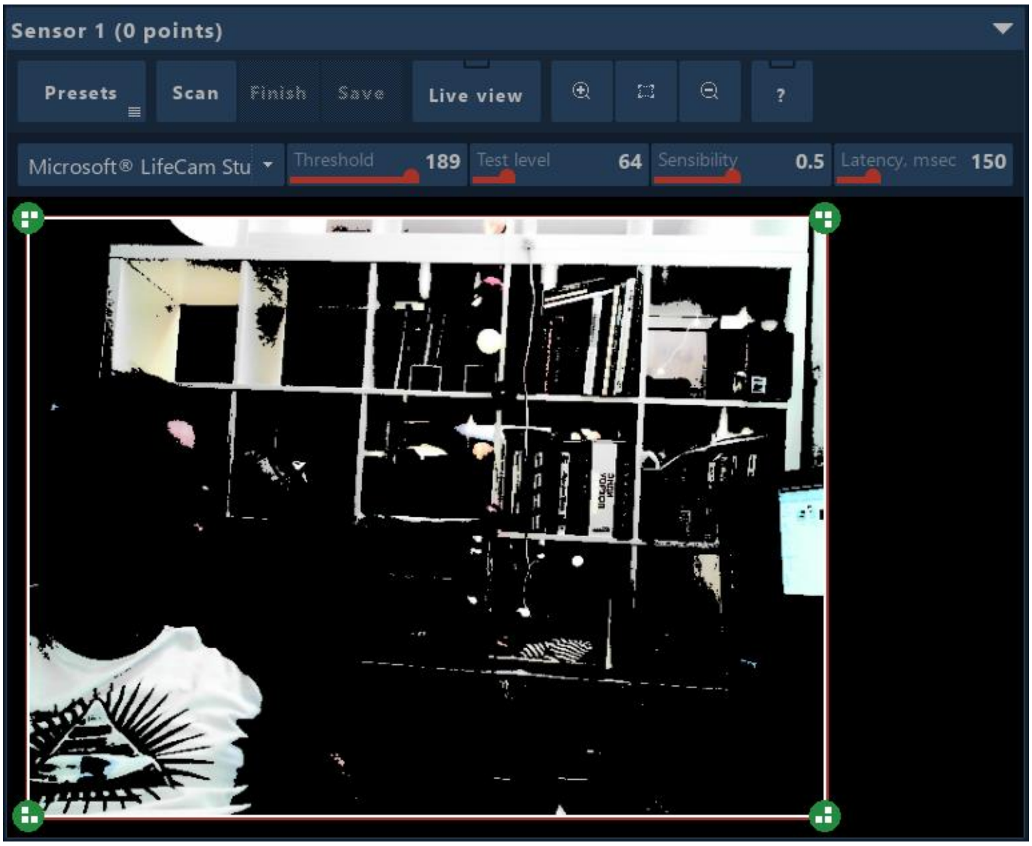
To clean bound Fixtures in the sensor, use the context menu in the grid and select “Unbind all”. There you can also unbind or change a specific Fixture.

11.3.1 Auto sensor

Auto sensor is a project element that allows to build a dynamic map of sensitive points of an actual LED pixels’ positioning based on feedback provided by a web camera or other DirectShow-compatible image capture.

In the sensor properties you should only set sensitive points’ size and type, while their quantity and positioning are determined by scanning physical LED installation.

After adding Auto sensor to the project, the “Pixel-mapping” window will show the following block:



Microsoft® LifeCam Stu

Threshold 189

Test level 64

Sensibility 0.5

Latency, msec 150

To start scanning, you need to choose capture devices. Preferably, use high resolution and low latency devices for more precise and faster scanning.

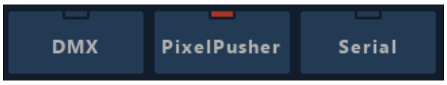
While scanning, all available pixel devices and pixels are used. To improve scanning speed and quality, turn off the devices that will not be included in the sensor.

Use the following sequence of actions:

1) Turn on and set up the output devices fixtures.
2) Select a camera or other means of image capture. In the operational field, there will be an image preview from camera, which has been put through the Threshold effect. Point the camera to an object of interest. You can scan this object either entirely or partially. It is important to remove other sources of light from the frame, especially blinking ones, as well as flares and reflections of light from the LEDs. If possible, it is highly recommended to turn off camera's auto exposure (some web cameras allow to do it through driver settings or original software).
3) Select a scanning zone, if it is smaller than the frame. This will help you to optimize the algorithm and to cut the area that does not have LEDs but introduces distortion. This zone is set by moving green points from the corners of the image preview.
4) Set up scanning parameters. Firstly, you should set the Threshold parameter so that, apart from the white point from a LED, there is an as-black-as-possible field; the value is used to be between 250 and 254. Test level — brightness of the test pixel; the value may depend on LEDs' brightness and the distance from the camera. Sensibility — sensibility of the algorithm for the pixel BLOB detection. Latency — it is needed for the complete cycle: data submission->LED lighting up->result in the captured frame; this latency depends on camera speed (FPS) and controller.
5) Press "Scan" button to start scanning. Wait.
6) Check scanning quality and pixel positioning precision. If deviations are insignificant, fix them manually after pressing "Finish" button by moving the point with the mouse, otherwise you can change scanning conditions and parameters and repeat scanning.

7) To complete the cycle of scanning, press “Finish” button.
8) Fix pixel positions manually and set right sensor geometry.
9) Press “Save” to save the result. Auto sensor pixel mapping is done.

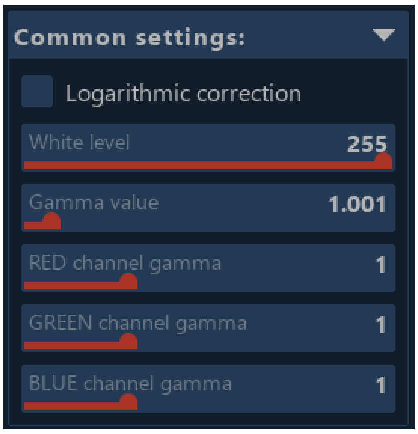
11.4 Output devices



To output on LED strips, MAPIO uses different methods and devices. These devices are divided into three groups:

1) DMX — is a group of devices that use DMX protocol. These are some types of USB devices and ART-Net devices that operate via Ethernet.
2) Pixel-Pusher — a Heroic Robotics device with its own exchange protocol via Ethernet.
3) Serial — any devices that operate via serial port and can be programmed, such as Arduino or similar devices.

To start working with these devices, it is necessary to enable one or several groups of devices using the buttons represented in the above figure. For the description of work with each type of devices available, see below.



Common settings:

□ Logarithmic correction

White level 255

Gamma value 1.001

RED channel gamma 1

GREEN channel gamma 1

BLUE channel gamma 1

All devices have a common setting block for the preparation of pixels (see Fig. above).

Logarithmic correction — corrects brightness of a LED by means of inverse logarithmic function; allows to have more natural colors and increase LED brightness.

White level — highest level of white; by reducing this value you can reduce LED brightness and energy consumption.

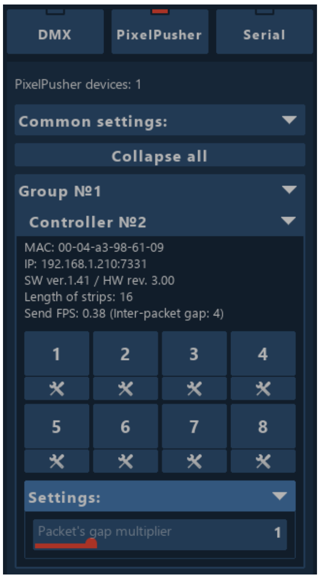
Gamma value — RGB value gamma correction. **RED/GREEN/BLUE channel gamma** — gamma correction values for each color channel.

11.4.1 Pixel-Pusher pixel devices

Pixel-Pusher is a device by Heroic Robotics that allows to output data on various address LED strips (for more details, please see www.heroicrobotics.com/products/pixelpusher).

MAPIO supports this device “out of the box”, so you don’t need to install any software. You only need to set up the device to work with your strips (for more details, please see Pixel-Pusher documentation).

Communication with the device is done via Ethernet connection with one native Pixel-Pusher protocol. There is no limit to the number of devices operating simultaneously, but the maximum number may depend on different subsystems' performance and user-friendliness of a multiple-device user interface. In practice, the maximum number for one computer with MAPIO equals to tens of Pixel-Pusher devices with 50,000–150,000pixels in total.



PixelPusher devices: 1

MAC: 00-04-a3-98-61-09
IP: 192.168.1.210:7331
SW ver.1.41 / HW rev. 3.00
Length of strips: 16
Send FPS: 0.38 (Inter-packet gap: 4)

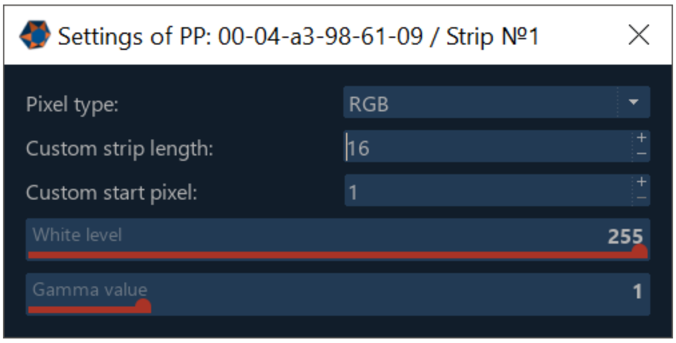
Packet's gap multiplier 1

Devices detected in Ethernet network are automatically shown on MAPIO device panel divided into groups and sorted by controller number (number of a group or controller is specified in device's configuration file; see Pixel-Pusher documentation).

Text field shows device information: MAC, IP, firmware version, hardware version, total length of strips in pixels, actual frequency of data submission to the device. "Inter-packet gap" is a current value of the pause between packets with data (in milliseconds).

Then, there are eight or less pairs of buttons, which correspond to the device's eight ports.

The button with a number selects a respective strip to position it in the sensor matrix. The button below opens settings related to the port (see Fig. below).



Settings of PP: 00-04-a3-98-61-09 / Strip №1

Pixel type: RGB
Custom strip length: 16
Custom start pixel: 1
White level 255
Gamma value 1

Pixel type can be any set of characters consisting of up to six characters from the following list: R, G, B, A, W, L, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, where RGB are respective color channels; A is the alfa channel; W shows the white level; L is the lightness channel equal to (R+G+B)/3; - and 0 are fixed values equal to 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 are fixed values from the following row, respectively: 8, 16, 32, 64, 128, 160, 192, 224, 255. You can either select from the list or enter any combination manually.

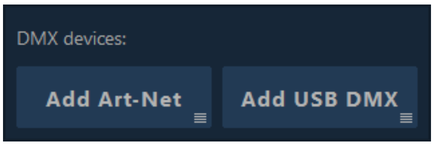
Custom strip length — length of a segment; it can be less than the total length of the strips.

Custom start pixel — starting pixel in a strip.

White level / Gamma value — white level and gamma to make a separate data correction for the port.

Custom strip length/ Custom start pixel parameters — are used for specifying an arbitrary strip segment to place these segments in different sensor matrix locations. These values are only used during the positioning of the chain in the matrix. After positioning, you can set other values to position the next segment.

11.4.2 DMX pixel devices



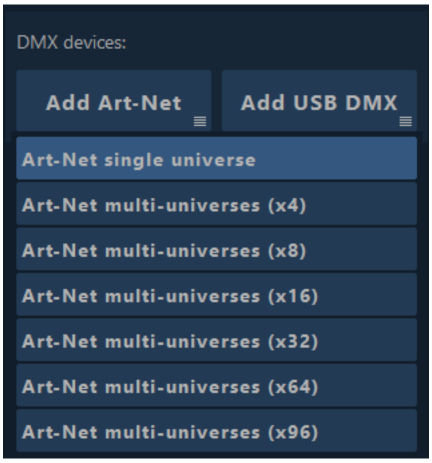
DMX devices:

Add Art-Net

Add USB DMX

DMX devices group includes two types of devices divided by the way of connection: Art-Net (DMX via Ethernet) and USB DMX devices.

Art-Net



DMX devices:

Add Art-Net

Add USB DMX

Art-Net single universe

Art-Net multi-universes (x4)

Art-Net multi-universes (x8)

Art-Net multi-universes (x16)

Art-Net multi-universes (x32)

Art-Net multi-universes (x64)

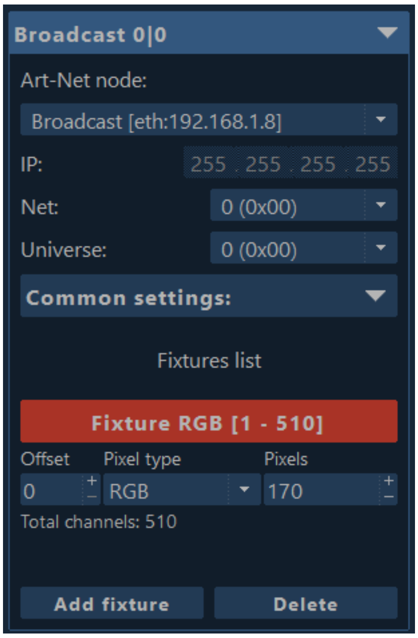
Art-Net multi-universes (x96)

When working with DMX devices via Art-Net, there is a possibility to use one of two modes: single universe addressing or multi-universe addressing. When creating a device in Multi-universe mode, there are presets for groups of 4, 8, 16, 32, 64 and 96 universes available, however this number can be changed at the subsequent setup (see below).

The figure below shows the interface of an Art-Net single-universe device. From top to bottom there are the following elements:

Art-Net node — list of destinations for Art-Net data. Here you can select an Ethernet interface and a way of addressing: “Unicast” or “Broadcast”. In this list you can also select an immediate device detected in the local network. “Unicast” mode serves for addressing a device by IP address. “Broadcast” is a broadcast way of addressing for all network devices at a time.

IP — addressable device address. If “Unicast” mode is selected, you can set your device’s address here, in other cases, this field is not available for editing.



Broadcast 0|0

Art-Net node: Broadcast [eth:192.168.1.8]

IP: 255 . 255 . 255 . 255

Net: 0 (0x00)

Universe: 0 (0x00)

Common settings:

Fixtures list

Fixture RGB [1 - 510]

Offset	Pixel type	Pixels
0	RGB	170

Total channels: 510

Net — Art-Net network number.

Universe — Art-Net universe number.

Please note that “Net” and “Universe” numbers are counted off from 0. To refer to other devices where the counting starts from 1, add 1 to the number.

Common settings — pixel preparation settings described in the beginning of the Section 11.4.

Fixture list — list of fixtures defined in DMX universe address area. By default, contains one fixture for 170 RGB pixels (512 DMX channels include 170 triplets with RGB values). Each fixture has the following properties:

Offset — first fixture channel.

Pixel type — defines pixel data type and size and can be of any set of characters (up to six) from the following list: R, G, B, A, W, L, -, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, where RGB are respective color channels; A is the alfa channel; W shows the white level; L is the lightness channel equal to (R+G+B)/3; - and 0 are fixed values equal to 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 are fixed values from the following row, respectively: 8, 16, 32, 64, 128, 160, 192, 224, 255. You can either select from the list or enter any combination manually.

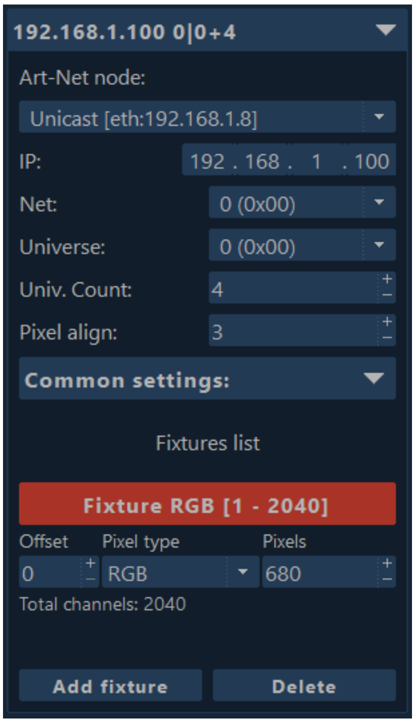
Pixels — number of pixels in a fixture.

Total channels — total number of DMX channels occupied by this fixture (for information) and calculated as follows: Pixels * PS, where PS is the pixel size specified in the “Pixel type” field.

Add fixture — a button to add fixtures. If there are unoccupied channels after the last channel in the fixture list, a new Fixture by default occupies the entire space to the end of the universe.

Delete — delete the last fixture from the list. When deleting the last fixture, you are asked if you wish to delete Art-Net device at all.

Art-Net multi-universe device interface is similar. See Fig. below:



Art-Net node: Unicast [eth:192.168.1.8]

IP: 192 . 168 . 1 . 100

Net: 0 (0x00)

Universe: 0 (0x00)

Univ. Count: 4

Pixel align: 3

Offset	Pixel type	Pixels
0	RGB	680

Total channels: 2040

Art-Net multi-universe device has only two additional properties:

Univ. count — number of consecutive universes joint in a single address area. The first universe is specified in the “Universe” field.

Pixel align — indicates how to align pixel data on universe borders. If this number is equal to the pixel size, pixel data will never “rupture” during the transition from one universe to another. Use 1 or 0 to disable incomplete pixel transfer to the next universe.

USB DMX

This group of devices supports the following DMX USB devices: Enttec (and other devices based on FTDI)/Digital Enlightenment/FX5/Velleman/Sunlite/uDMX/Eurolite.

Each DMX USB device has “Common settings” and “Fixture list” block described above in the “Art-Net devices” section.

FTDI-based devices have the following additional properties:

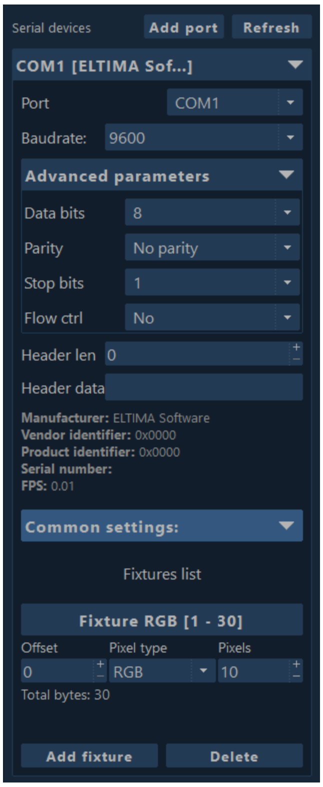
Device type — operational mode selection: “Enttec Open DMX” or “Enttec DMX Pro”.

Device port — available if you select “Enttec DMX Pro” in the “Device type” field. You can select from two available ports: Port 1 or Port 2.

Other device types have no specific properties.

For this menu to reflect actual list of connected devices, you should restart MAPIO.

11.4.3 Serial pixel devices



Port: COM1
Baudrate: 9600
Data bits: 8
Parity: No parity
Stop bits: 1
Flow ctrl: No
Header len: 0
Header data:

Manufacturer: ELTIMA Software
Vendor identifier: 0x0000
Product identifier: 0x0000
Serial number:
FPS: 0.01

Offset	Pixel type	Pixels
0	RGB	10

Total bytes: 30

When enabling Serial devices group, one port adds automatically (see Fig. above). There is an option to add another port or several ports by pressing “Add port” button. You can also remove ports by removing all their fixtures — when you remove the last fixture the port removes as well. “Refresh” button refreshes available COM port list.

Serial device properties:

Port — COM port

Baud rate — exchange speed

Advanced parameters — a group of port setting enhanced parameters containing the following parameters: **Data bits**, **Parity**, **Stop bits**, **Flow ctrl**

Header len — length of a header placed at the beginning of a device’s data block

Header data — text cue placed at the beginning of a device’s data block header; its length may not exceed header’s length specified in “Header len”

Data header is needed to detect new data for pixels in the data stream, i.e. it indicates that a header is immediately followed by first pixel data.

Next, there are “Common settings” and “Fixture list” blocks described above.

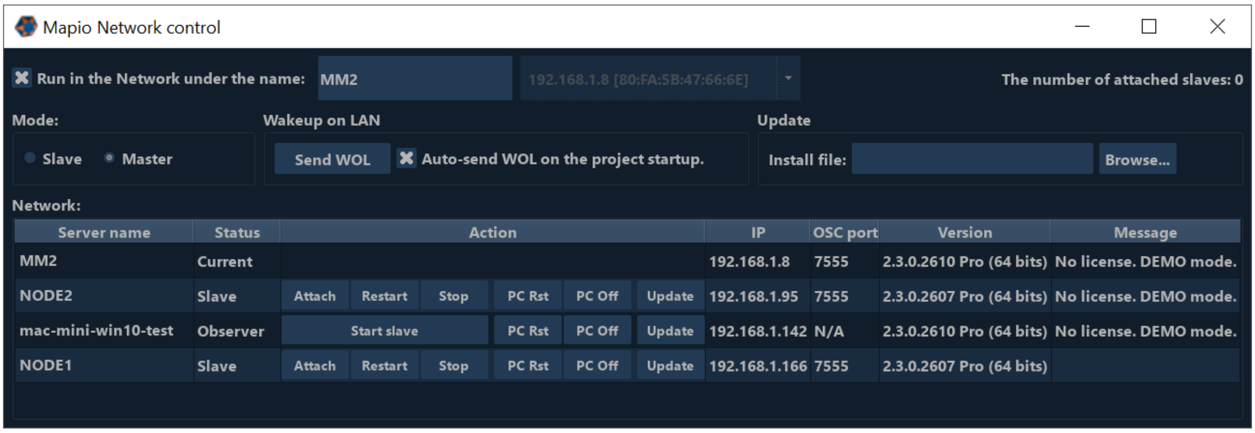
12 PRO | Network

12.1 Quick start

To be able to operate in master-slave pair, MAPIO PRO must be installed in every computer. After installation on a slave machine, make sure that there is a MAPIO Observer icon in the notification area of the task bar.

1) Start MAPIO on the master machine.
2) At the bottom of the main window you will see the “Network” button. Press on it.
3) In the new opened window, you need to select “Master” mode and enable networking in the “Run in the network” box.
4) The list of machines in the network will appear in the table below after a few seconds.
5) Since MAPIO has not been started on any slave machine, press “Start slave” button to start MAPIO in “Slave” mode on the respective machine remotely with the help of the Observer.
6) The table will update after a few seconds and once you see “Attach” button, press on it. Now this slave machine is involved in the current process. Repeat 5) and 6) for all the slave machines you need.
7) Now you can close network control window.
8) In the “Destination” menu select “Multi-Display” to set up the output. Enable output for every machine including, if needed, the current (local) one. To select the current machine, use the selector located at the bottom of the window next to the buttons.
9) In the “Destination” menu, select “Output settings” and, using the similar pattern, set up project resolution for every machine.
10) Mapping, blending, warping and other settings are done for each machine in turn; changes are applied online, on-the-fly; the current machine can be selected in the top right corner where normally snapshot selection is located. Please note that in the network mode the snapshot functionality is not available.
11) Add media content to MAPIO. On the slave machines, media content will upload following the same path, which means that video files in every machine must be identical in terms of location and name, while their contents can be of any type.
12) After setting up the project, you can store it on the master machine. At the next project start, slave machines will receive the settings automatically over the network.

12.2 Network control window



☒ Run in the Network under the name: MM2 192.168.1.8 [80:FA:5B:47:66:6E]

The number of attached slaves: 0

Mode: ○ Slave ● Master

Wakeup on LAN: ☒ Auto-send WOL on the project startup.

Update: Install file:

Network:

Server name	Status	Action	IP	OSC port	Version	Message
MM2	Current		192.168.1.8	7555	2.3.0.2610 Pro (64 bits)	No license. DEMO mode.
NODE2	Slave	Attach Restart Stop PC Rst PC Off Update	192.168.1.95	7555	2.3.0.2607 Pro (64 bits)	No license. DEMO mode.
mac-mini-win10-test	Observer	Start slave PC Rst PC Off Update	192.168.1.142	N/A	2.3.0.2610 Pro (64 bits)	No license. DEMO mode.
NODE1	Slave	Attach Restart Stop PC Rst PC Off Update	192.168.1.166	7555	2.3.0.2607 Pro (64 bits)	

Elements shown on the above figure, top to bottom and left to right:

- Turning on the network functionality
- Name of a MAPIO instance in the network
- IP and MAC address of the current network interface
- Number of connected slave instances if MAPIO operates in "Master" mode, or station status if the current mode is "Slave"
- Networking mode: "Slave" — slave instance mode, "Master" — master instance mode (for more information, see below)
- Wakeup-on-LAN features — automatic waking-up/turning on for slave machines
- MAPIO automatic remote update feature — in this field the installation file should be specified

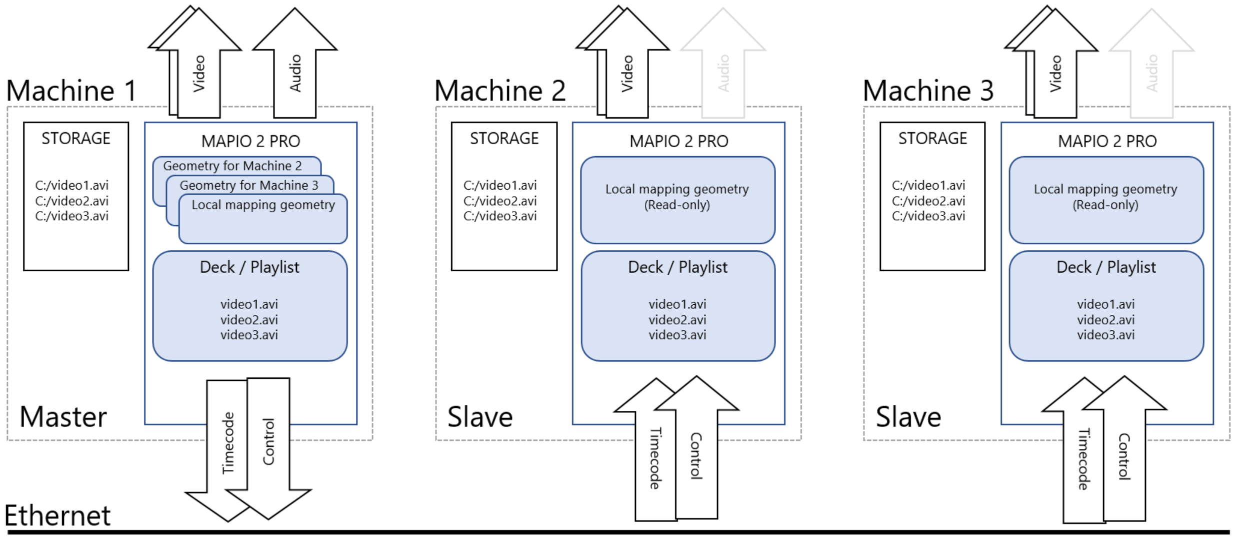
Then, there is a table of MAPIO instances detected in the network, including the current instance. Apart from the current instance named MM2, the table includes slave MAPIO instances named NODE1 and NODE2. Though the "mac-mini-win10-test" machine does not include any started MAPIO instance, MAPIO Observer can run MAPIO instance in "Slave" mode.

For the successful performance of network features, the following network interface ports must be available to send and receive: 8362, 8363, 8364. By default, in Windows 7-10 systems MAPIO performs setup of the embedded firewall. If you have a third-party firewall, make sure it is properly set up.

12.3 Basic principles of operation

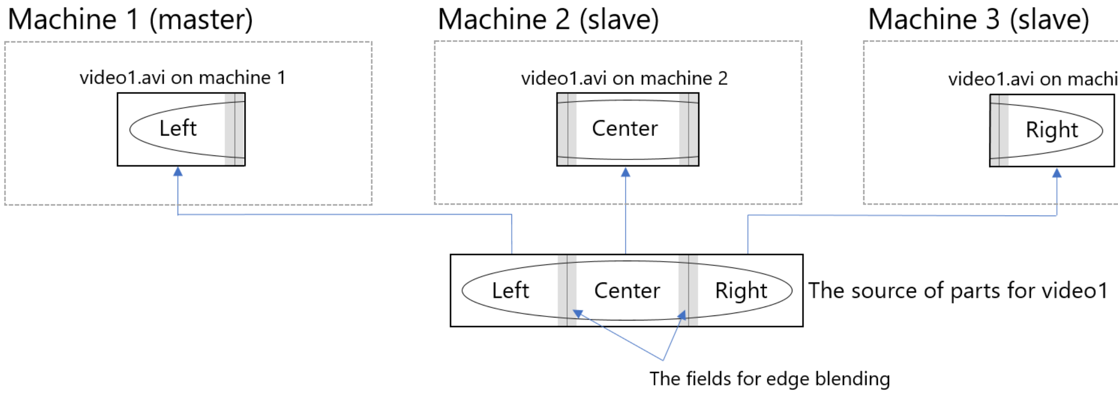
"Slave" is a mode of a secondary instance fully controlled by a master-type instance. A slave instance does not require a control monitor or any monitor at all. Neither you need to save your project since initialization and editing are performed by the master instance. All project settings are gathered on the master instance. Mapping configurations and media content control: automatic start-up for slave instances can be set with the help of MAPIO Observer. On the master machine you can also set automatic start-up for MAPIO with a project file. Additionally, slave instance can run with a command-line parameter: network slave.

This arrangement reflects one of several possible MAPIO usage configurations:



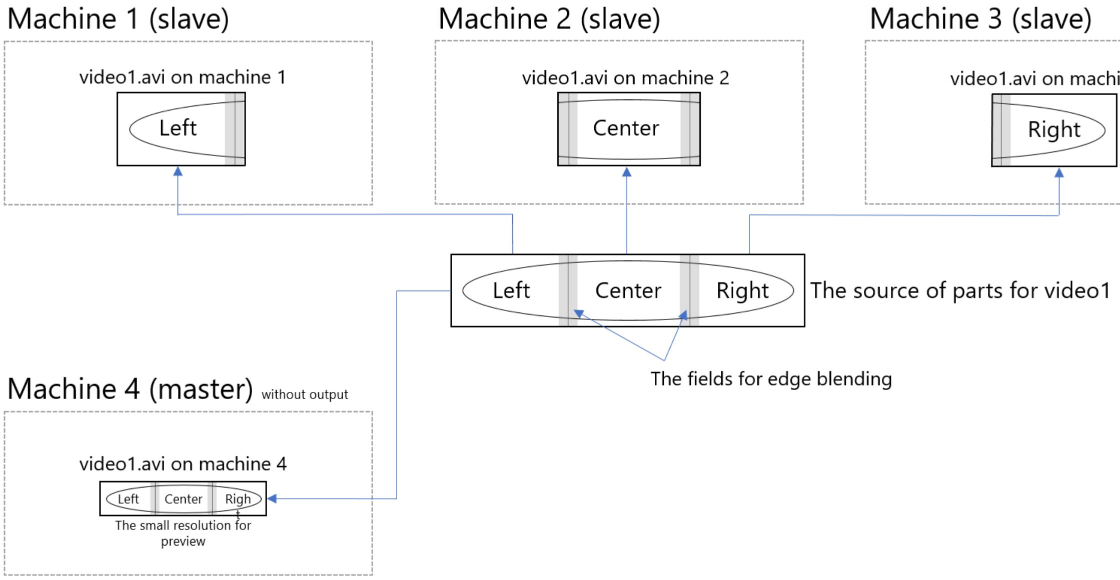
In this option, the master machine outputs content on par with slave machines. It is recommended to output sound exclusively from the master machine without duplicating sound output on slave machines, especially if cross audibility of several machines is possible, to avoid echo effect, which can arise in some cases. Media content files must be in similar locations and have similar names on all the machines, while the contents can be of any type. One of the important requirements for the file contents is that the numbers of frames per second in the files that play in parallel must match exactly. If the content on a slave machine is not found you will be offered to solve the problem by changing the file or file location folder (see below).

The arrangement below shows a possible option to prepare content for the output of a single panoramic video with the need for blending:

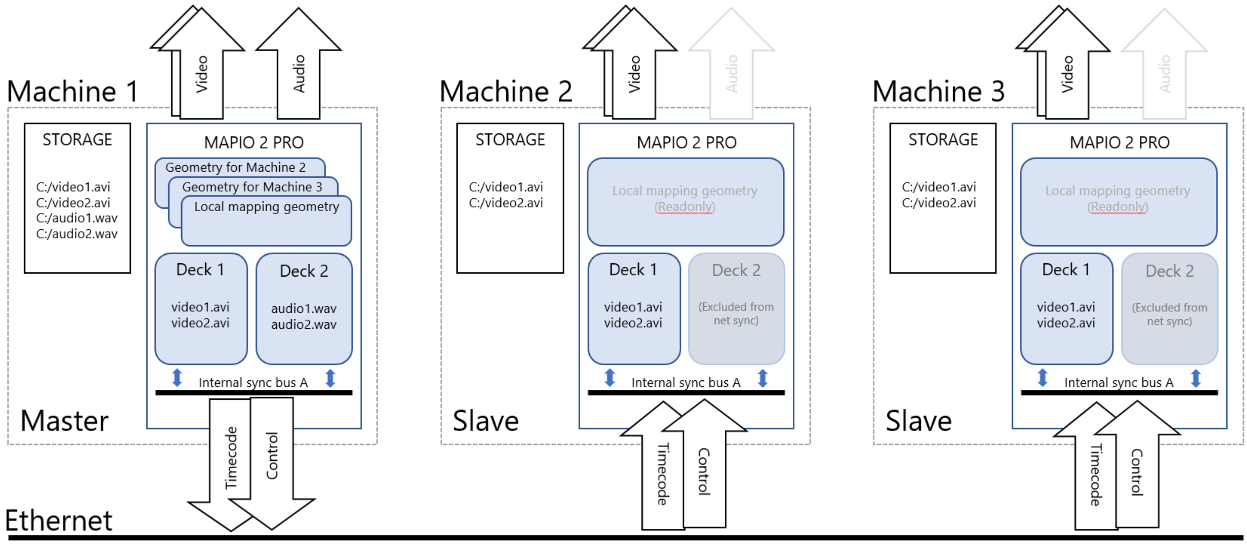


Please note that to successfully blend neighboring parts of the screens, you need to prepare video files that contain not only 1/3 of the screen, but the blending zone field as well.

On the arrangement, the Machine 1 (master) outputs the content along with the slave machines, however a configuration where the master machine is intended only to control and not to output image (only sound, if necessary) is possible as well.



Shared use of local A/B synchronization between decks with the synchronization over the network is also possible. The arrangement reflects the option to use synchronous decks on the master machine with separated sound, while on slave machines these decks are turned off (see “Bind in work” functionality) since sound output on slave machines is not required:



In the current version of MAPIO, only "Multi-display" mode is available for the remote setup of output methods. Other output methods, including Pixel-Mapping, are not available for slave instances. It is planned to add them in upcoming versions.

12.4 Working with the project

After you have connected one or more slave instances to the project, you can start working with the project. Now, in the distributed network mode MAPIO user interface always represents one of the instances, either remote or local, in MAPIO mapping editing section. You can select current instance in the top right corner of the window labelled NET CLIENTS where previously the snapshot selection option was located.

When you choose remote instance, the local output freezes. It is not an error, but an operational aspect.

All project changes, such as geometry editing, tool parameter editing, project element visibility editing, are applied immediately to the selected instance or, if these changes concern media console, to all instances.

Once the project is set up you can save it to a file, this file will contain settings of all remote instances. While a network project is loading all operational modes will restore and the settings will be sent to remote instances, if they are already available or once they are available. Furthermore, at the project startup all connected instances will receive "Wakeup-On-LAN" message to wake the machines up if they are turned off.

Binding decks to specified instances

For complex cases of media content distribution over instances, there is a possibility to determine what deck is available and in what instance. This is implemented through an option in the additional deck menu: **Bind in network**. By default, a new deck is active in all instances, but for a remote instance, by unticking the "Bind in network" box, you can disable media content uploading (the deck will be present, but won't fill with contents).

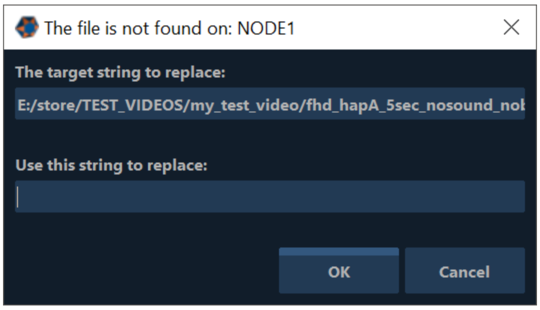
File naming

To facilitate video files handling, you can insert special expressions into file or folder name. When searching files on slave machines, such expressions are replaced by certain values. Currently, the following expressions are used:

1) **@node@** — when searching files on slave machines, this expression will be replaced by the name of the instance;
2) **@ip4@** — will be replaced by full IP address of the instance;
3) **@ipNode@** — will be replaced by the last number from the IP address of this instance.

For example, if the video file that is being loaded on the master machine is named ***show-@node@.mov***, MAPIO will search for a file named ***show-NODE1.mov*** on the slave machine titled NODE1.

These replacement keys can be used in folder names as well.



The file is not found on: NODE1

The target string to replace:

E:/store/TEST_VIDEOS/my_test_video/fhd_hapA_5sec_nosound_nol

Use this string to replace:

If it is impossible to find a file on a slave machine, you will be offered (on the master machine) to replace this file or folder. This replacement will be saved in the project and used later on. In this dialog you will be offered two editable fields:

- First field with the value to be replaced: by default, this field displays the full path and the name of the file that was not found on the computer, so, by default, you will be offered to replace a specific file;
- Second field with the value that should be used instead of the value found above.

Therefore, in this way you can either specify another file for replacement or specify only a part of the file path for replacement, for example it is possible to replace only disc and file location folder. Subsequently, this rule applies to all files in the project on a given machine.

Remote machines control functions

If MAPIO Observer runs on a remote machine, you can perform the following steps:

When you hover your mouse on the field with the instance's name a button appears. By pressing it you can rename the instance.

Start slave — Run MAPIO in "Slave" mode and remember auto startup settings. This button appears when MAPIO application is not found in the machine's running processes. Please note that remote startup is only available for slave machines.

Attach — Attach remote instance to the current project. Available when the instance is not attached to the master machine.

Detach — Detach from the instance and remove its settings from the project. Available not only when the instance is attached to the current machine, but also when it is attached to another master machine so that you can reattach the instance to the current master machine.

Refresh — Restart the project on the respective machine without restarting the application.

Restart — Restart MAPIO application on the respective machine. If the instance is not attached, it will automatically receive the project from the master machine.

Stop — Finish MAPIO application and disable auto startup.

PC Rst — Restart machine; executes immediately.

PC Off — Turn off machine; executes immediately.

Update — this button starts update installation; msi-file with MAPIO distribution must be specified at the top of the window in the “Update” block. This file is not checked whether it is a MAPIO distribution, so, if another file is used for updating, the following will happen: immediately after installation starts, the Observer will terminate and, if it is not a MAPIO distribution, won’t restart and you won’t be able to perform remote control.

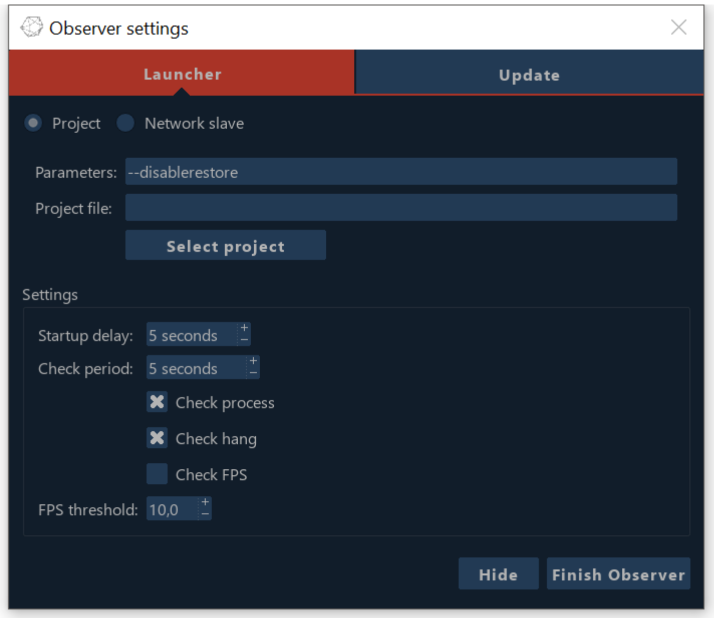
12.5 License issues

In demo mode, there is no limit to the number of connected instances. If the licenses are installed in a connected MAPIO network, each license can be used only once. Duplicates won’t work. Therefore, to form a network you will need individual licenses for each network participant, both for master and slave machines. For all instance types PRO version is needed. If you buy several licenses, you can get a discount. Please email at contact@visution.com.

12.6 Security issues

All network features are intended to operate in a safe, closed environment. The communication protocol and the control feature are not safe in an open environment, so it is recommended neither to arrange access to local network from Internet nor connect local network with public Wi-Fi access. To archive safest and most stable result, use local Ethernet 100/1000 network not occupied by other services and with no Internet access.

13 PRO | MAPIO Observer



● Project ○ Network slave

Parameters: --disablerestore

Project file:

Startup delay: 5 seconds

Check period: 5 seconds

☒ Check process

☒ Check hang

□ Check FPS

FPS threshold: 10,0

MAPIO Observer is a simple tool that allows to observe and automate operation of the main MAPIO application.

MAPIO Observer is only available in the PRO version.

By default, when MAPIO PRO is being installed Observer’s auto startup is set up with the help of Task Scheduler in the Windows system. After startup, the tool is represented by the icon in the task bar notification area. By left-clicking on the icon or via “Settings” menu (opens by right-clicking) you can open the setting window (see Fig. above).

At the top of the window, you can choose between two pages:

1) **MAPIO Launcher tab** — Set up and control MAPIO startup. Interface description (top to bottom):
   - MAPIO startup mode selection: “Project” — A simple mode to run with the parameters and project file specified below; “Network” slave — Run in network mode as a slave instance and wait for data from the master. The “Project” mode is also used on the master machine for a network project.
   - Command-line parameters. For the list of available MAPIO parameters, see Section 18.
   - MAPIO project file with “mio” file extension; it is used only in “Project” startup mode.
   - Startup delay — Delay after startup, before the Observer starts to operate.
   - Check period — Frequency for MAPIO check.
   - Check process — Checks if MAPIO process is running in the system.
   - Check hang — Checks for a possible MAPIO hang-up.
   - Check FPS + FPS threshold — Checks MAPIO output FPS rendering that must be at least equal to the specified value.

   Hide — Button to close the window. Observer will continue to operate. Finish Observer — Close the window and terminate Observer.

2) **Update tab** — Intended for checking and installing updates. There you will find information on the current version and available update, as well as its description. There are also “Check for

updates" and "Install" buttons. Please note that, when you press "Install", MAPIO closes automatically without "Save changes" prompt, even if the project has not been saved.

Observer — execution logic

Observer starts automatically after logging in (automatic login should be set up). Next, if a project is set for startup or the "Network-slave" mode is selected, MAPIO runs in the respective mode. Then, the Observer tool periodically checks MAPIO application process using the methods specified in settings. If the application was not found, hung up or if FPS is below the minimum permitted level, the application will be forcibly terminated and restarted.

Observer — context menu

Right-clicking on the Observer icon opens a menu that include the following options:

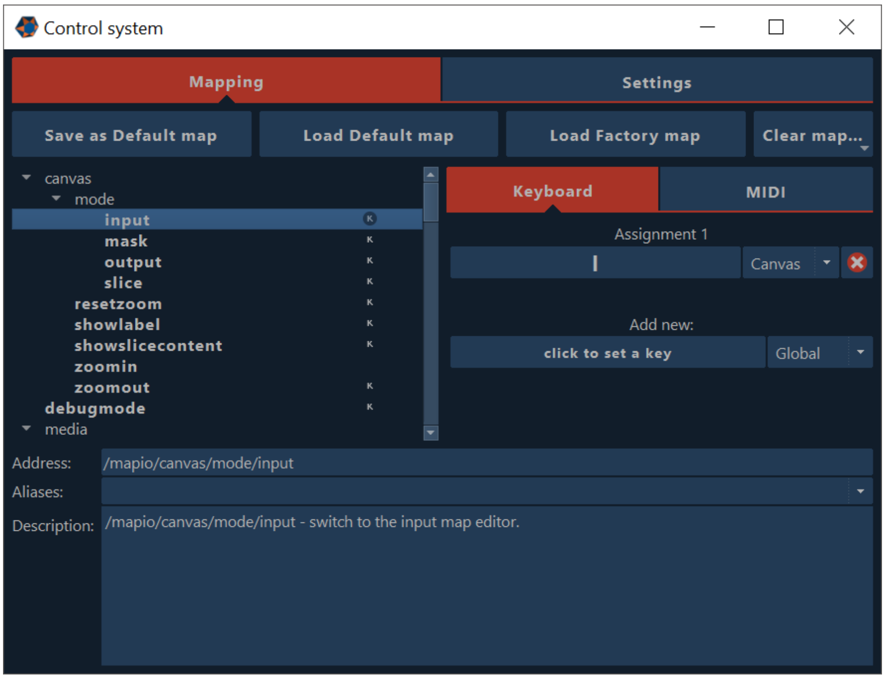
1) **Suspend/Resume** — Suspend and resume MAPIO process check.
2) **Settings** — Open setting window.
3) **Update** — Open update window.
4) **Run at Windows startup** — Set up automatic launching for the Observer at system startup; untick to prevent Observer from starting automatically.
5) **Finish Observer** — Terminate the Observer.

14 Control (OSC, MIDI, Keyboard)

Control system main window (see Fig. below) is divided into two tabs: “Mapping” and “Settings”. Here the features to control MAPIO from outside the application are gathered. These features are available for both PRO and LITE versions. All control features are always available via OSC protocol [wikipedia.org/wiki/Open_Sound_Control]. By default, port **7555** is used. It can be changed in the “Settings” tab. MIDI devices and keyboard can be arbitrarily assigned to any controlled elements represented in the element tree of the “Mapping” tab. Each control element has its own OSC address.

“Mapping” tab

Top button line is a tool bar for saving, uploading and resetting all the control assignment settings described below. By default, all control assignment settings are saved in the project but there are also global application settings that are applied to newly created projects.



Address: /mapio/canvas/mode/input

Aliases:

Description: /mapio/canvas/mode/input - switch to the input map editor.

Tool bar description (from left to right):

Save as Default map — Save current project settings as application settings; will apply to newly created projects (no changes for previously created projects).

Load Default map — Upload application settings to the current project, i.e. reset to default global settings.

Load Factory map — Reset the current project to default factory settings.

Clear map... — Remove all settings from the selected map, i.e. fully clean all assignments.

Tip: to reset global assignment maps to factory settings, open a new project, then press **Load Factory map** and **Save as Default map.**

Below, there are three blocks:

1) Element tree on the left — Tree-like list of control elements available. The list is dynamic and reflects current project situation, current project element list, decks, mixers and media sources.

2) On the right — List of assignments for the current selected control element. Currently, MAPIO supports control assignment to keyboard and MIDI devices with every device type located on a separated tab.
3) At the bottom you will find information about the selected element, where "Address" is the OSC address of the selected control element; "Aliases" are alternative OSC addresses added to support old versions and not recommended for new projects; "Description" is the description of the selected element where parameter type and limits are specified.

In the address tree, there is a context menu with the following options:

- **Collapse all** — Collapse and minimize all address tree branches.
- **Expand all** — Expand all address tree branches.
- **Clear Keyboard assignments** — Clear all keyboard assignments for the selected address.
- **Clear MIDI assignments** — Clear all MIDI assignments for the selected address.
- **Remove all duplicate Key assignments** — Make the use of the keys assigned to this address exclusive. This will ensure that all key assignments used for the current address will remove from other addresses.

OSC address description contains the following conventions for the parameters: **i** — integer value; **f** — floating point value; **s** — text string.

How to set keys

1) Select desired control element on the tree.
2) Press "Click to set a key" button on the right.
3) Now press a key or a combination of keys that you want to set.
4) Once the key is set you can also set the key scope from the following list: Global, Canvas, Console, Canvas and console, Pixel-Mapping, QML window.
5) If the element has a parameter, you need to specify desired value that will be assigned when you press the key that has been set.

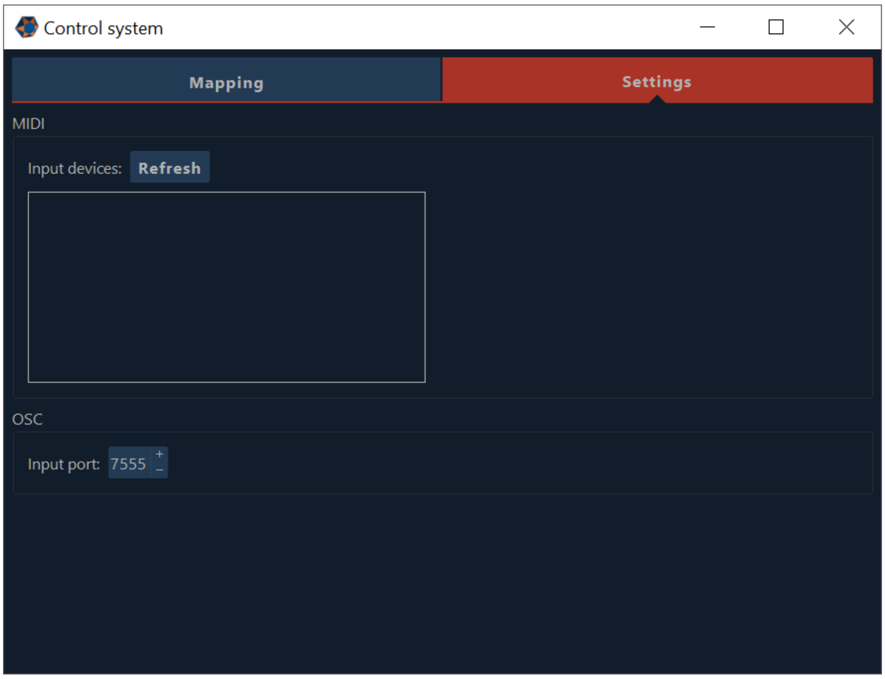
You can specify key scope and parameter before you set a key. You can set up to 50 keys for one control element with each key having its scope and value.

How to set MIDI controllers

1) Select desired control element on the tree.
2) Press "Click to midi learn" button on the right.
3) Now press and move MIDI controller element that you wish to set.
4) Once it is set, you can also set its scope from the following list: Global, Canvas, Console, Canvas and console, Pixel-Mapping, QML window.
5) If the element has a parameter, you can set up parameter value normalization range, as well as invert the value.

You can set up to 50 MIDI controllers for one control element with each one having its scope and parameters.

"Settings" tab



First MIDI section — Includes a list of available MIDI devices and enables necessary devices to work in the program.

Second MIDI section — Data-reading port via OSC protocol.

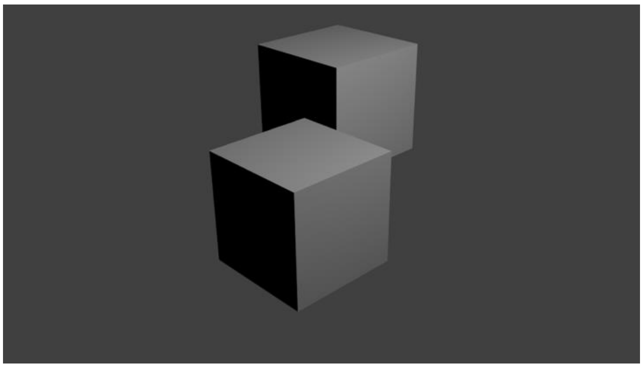
Setting data are saved and restored along with the project.

15 3D Import

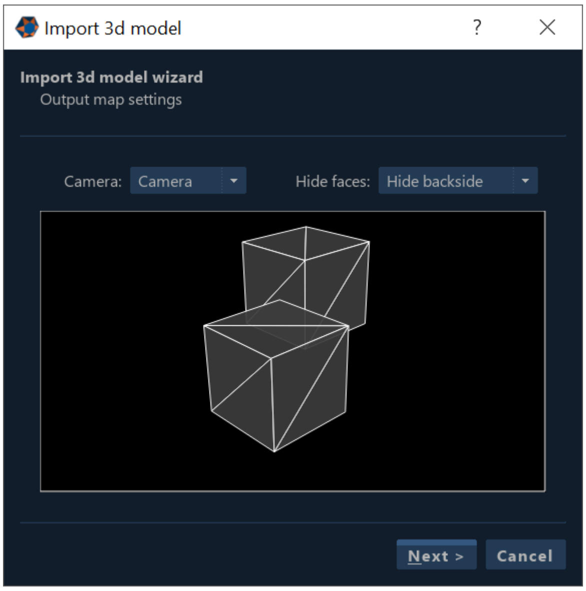
To facilitate slicing of a complex project, MAPIO supports 3D file import to generate respective maps. When importing, every triangle turns into a slice with its position and configuration in the output map depending on the selected camera. The camera must be embedded in the file. Input map can be equal either to output map or to a texture UV map. This import method places restrictions on a 3D model: it should be a low polygonal model (preferably, no more than 100 triangles) with at least one embedded camera. FBX by Autodesk and DAE (Collada) files are supported.

Simple import example

Original FBX file and its render — The view corresponds with an embedded camera named "Camera" — Frame aspect ratio 16:9.



First import page: here you have a framework model; the projection corresponds with the selected camera. You can also select what edges not to use for the import: by default, all backside edges are rejected. Frame aspect ratio is not considered since it is normalized. The aspect ratio final value is set further by the resolution of the entire project (to archive perfect result and coordinate matching, it should be equal to your render aspect ratio).



Import 3d model

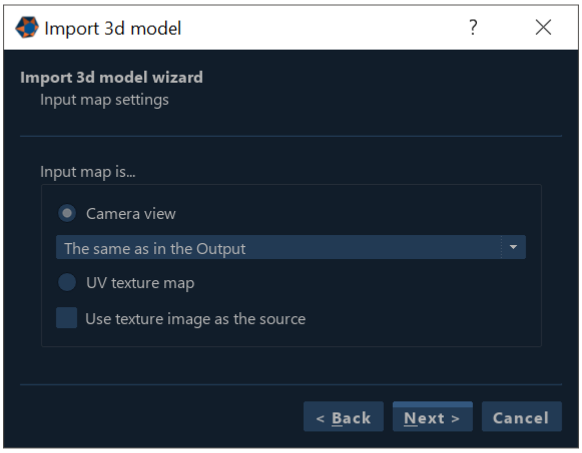
Import 3d model wizard

Output map settings

Camera: Camera

Hide faces: Hide backside

Second page: slice input map settings. Select from two options, either a camera or a texture UV map.



Import 3d model

Import 3d model wizard

Input map settings

Input map is...

● Camera view

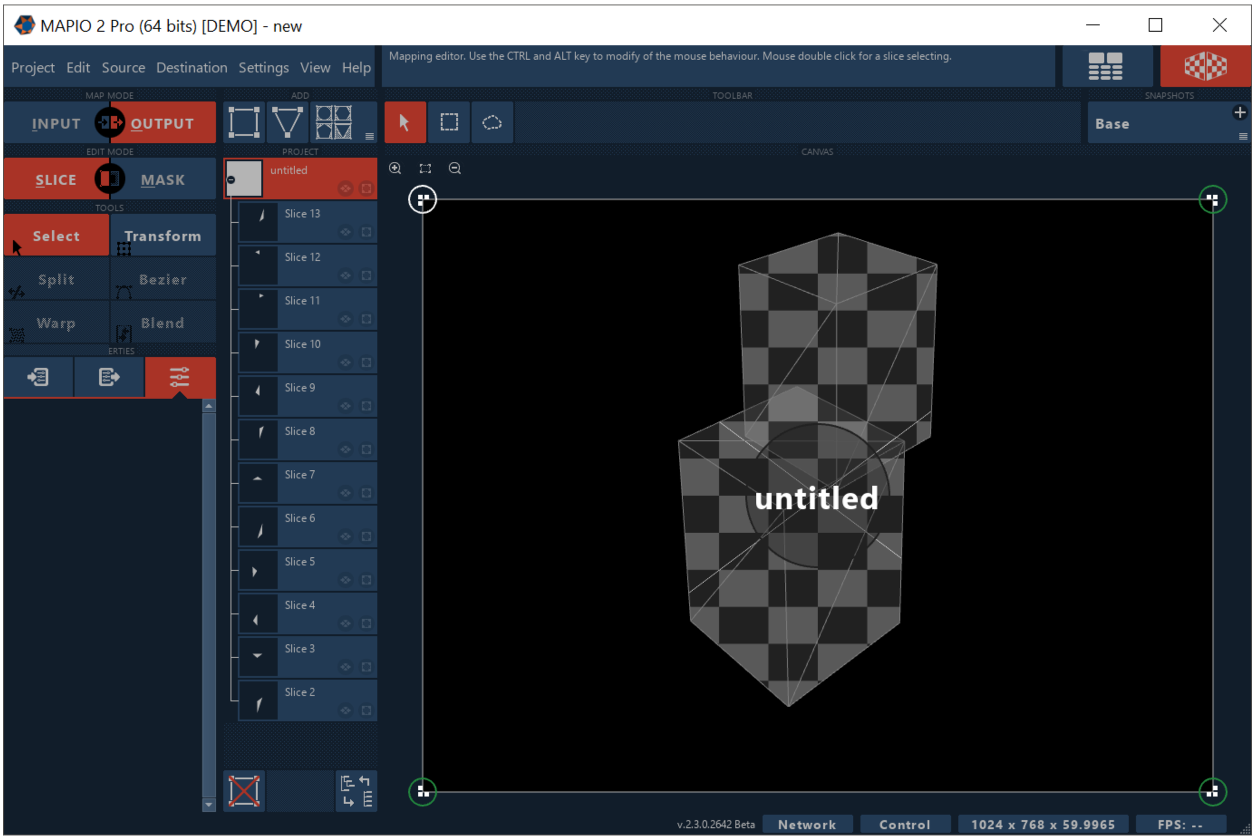
The same as in the Output

○ UV texture map

□ Use texture image as the source

The third page contains one option: delete or do not delete project existing slices.

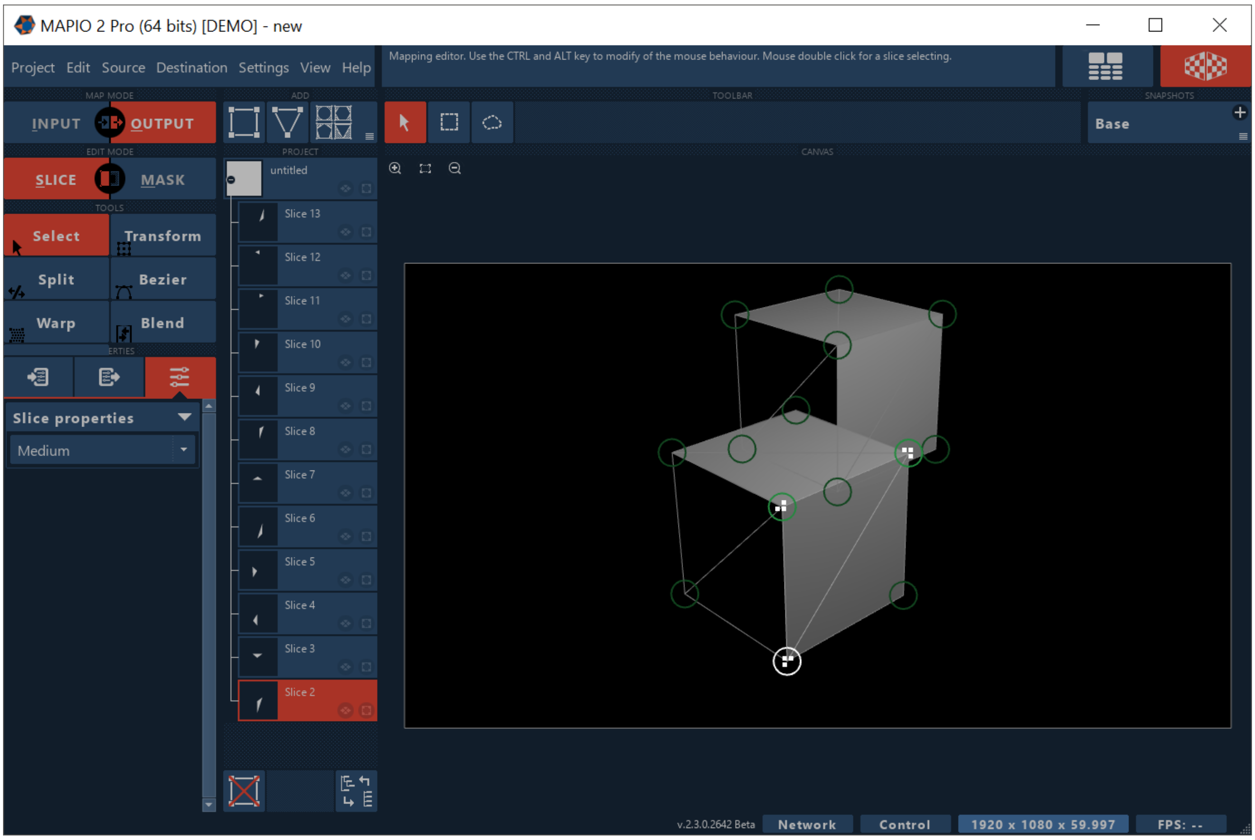
Import results, all slices are joint in a group.



MAPIO 2 Pro (64 bits) [DEMO] - new

Mapping editor. Use the CTRL and ALT key to modify of the mouse behaviour. Mouse double click for a slice selecting.

You only need to set the right project resolution and select render file as a source for all slices.



Mapping editor. Use the CTRL and ALT key to modify of the mouse behaviour. Mouse double click for a slice selecting.

When performing projection mapping, this import result allows to quickly adjust user's content to a physical object without slicing, which is time-consuming.

16 QML sketches

16.1 What is QML?

QML is a declarative user interface language. It allows developers and designers to create high-performance, visually attractive and richly animated applications. QML ensures easy-to-read, declarative JSON-like syntax with support for imperative JavaScript expressions combined with dynamic property bindings.

Link to QML Reference: http://doc.qt.io/qt-5/qmlreference.html

MAPIO uses this language to generate different kinds of static, dynamic and interactive content. Within MAPIO, these programs are called “QML sketches” or simply “sketches”.

Tool’s assortment of features is very wide and almost doesn’t differ from the features of QML language itself. However, there are some restrictions (for more details, see below).

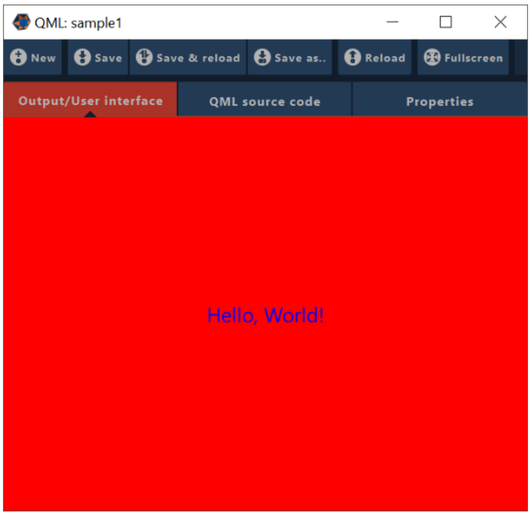
MAPIO is supplied with a certain number of ready-made sketches with open source codes that can serve as a base for user sketches.

Source code example for a simplest sketch:

```
import QtQuick 2.9

Rectangle {
    color: "red"

    Text {
        anchors.centerIn: parent
        text: "Hello, World!"
        color: "blue"
        font.pixelSize: 40
    }
}
```



Code's execution result is represented on the above figure. For more introductory information on QML programming, see: http://doc.qt.io/qt-5/qtqml-syntax-basics.html.

For more information on how to create your own sketches, see Section 16.4.

16.2 QML source general features

16.2.1 Loading

Bundled sketches are files containing source code with “qml” extension located in the “QML sketches” folder of the program directory. You can load these sketches via MAPIO “Sources/QML/…” menu.

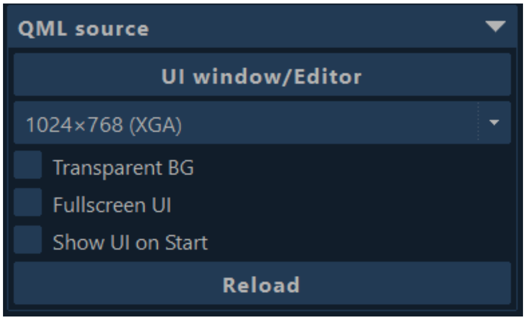
This menu also automatically contains sketches saved by the user in “MAPIO QML sketches” folder in standard “Documents” directory of the current user. Such sketches will have “*” symbol at the end of the name.

With the help of the “Sources/QML/Open file…” menu option you can open sketches from any folder.

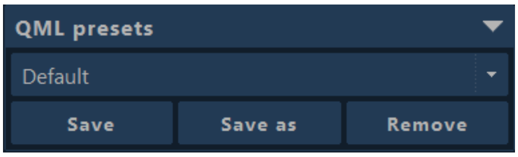
16.2.2 General parameters

As media sources, all sketches have four groups of parameters:

1) Playback — A standard group of all deck elements. For more information, see Section 8.3.

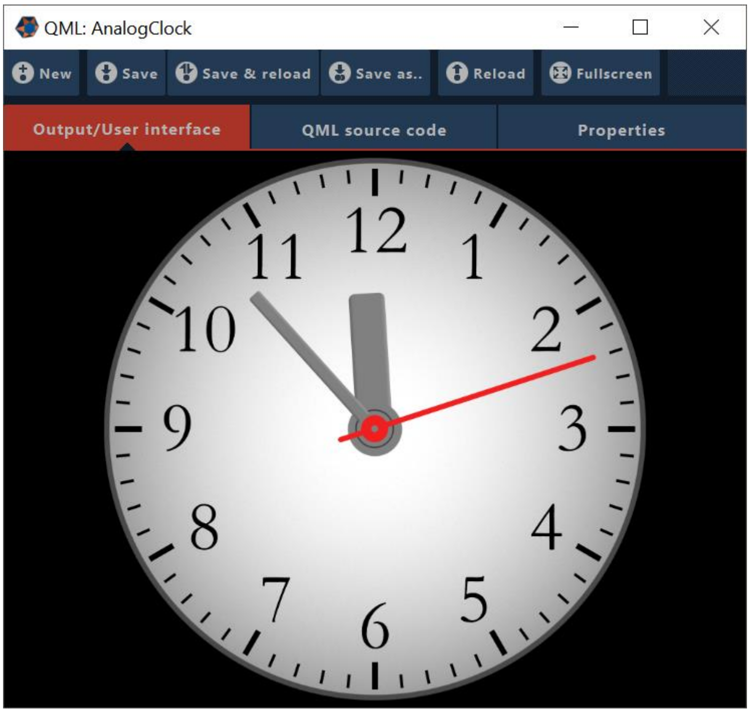


2) QML source — Parameters of how MAPIO handles a sketch. It determines sketch render resolution, background transparency and other parameters. “UI window/Editor” button opens QML window, its output and other settings (see Section 16.2.3). “Reload” button — Restarts sketches; it can be helpful when you need to apply changes after having corrected sketch source code (e.g. in a third-party editor).



3) QML presets — A group containing control elements for sets of saved sketch parameters to save certain settings and quickly shift between them.
4) QML parameters — A group of sketch parameters defined and used in a sketch. This group can be absent if sketch parameters are not defined.

16.2.3 Editor



If you press "UI window/Editor" button in the source QML properties section, a window will open (see Fig. above). This window is used for enhanced interaction with sketch program, as well as for editing its source code.

At the top of the window, there is a panel with the following options:

1) **New** — Create a new sketch.
2) **Save** — Save source code of the current sketch. If you have already uploaded a sketch from MAPIO set, this option will offer you to save this sketch under a new name in the current user "Documents" directory.
3) **Save & Reload** — Save your current sketch and apply changes for current source.
4) **Save as...** — Save current source code in a new file.
5) **Reload** — Reset source code changes and reload sketch.
6) **Fullscreen** — Maximize QML program UI window in full screen mode. You can close the window by pressing ALT+F4. This status, including the monitor where UI is deployed, is saved in the project. At the next source startup, the status will restore.

The window below contains three pages:

Output / User interface — An area where sketch execution result is displayed. Depending on sketch contents, full operational area or only a certain area is displayed. User–sketch interaction with the help of mouse and keyboard is supported.

QML source code — This page contains a simple source code editor for the current sketch.

Properties — Current sketch properties. For convenience, the properties of QML as a source are repeated here.

16.3 Sketch list

AnalogClock / DigitalClock — Clock and countdown generator.

ColorsGrid — Free color block grid generator.

ColorSolid — Solid color or different types of gradients allowing HUE animation.

GLSLSandboxSketch / GLSLShaderToy — GLSL shader generators optimized for Internet resources: http://glslsandbox.com and https://www.shadertoy.com, respectively. To access to the shader code, right-click on the UI window and press “Edit” button.

ShadedSwirlGLSL — Simple animated GLSL shader example.

InstaFeed — A sketch used for image flow generation from Instagram. Currently, API is obsolete and restricted by Instagram Terms of Use. Held up as example.

OSCRecieveDemo / OSCSendDemo — An example of use for reception and sending of OSC commands.

Outline / OutlineAnimation / Shapes — Generation and animation of different types of geometry.

PatternGenerator — Setting-pattern generator for edge blending and warping tasks.

RunningText — Crawling text generator.

SlideShow — Slideshow generator for images from a selected directory.

Text — Text block generator with enhanced formatting capabilities.

TweetFeed — Twitter message flow generator by a keyword.

WebBrowser — Full-featured browser based on the Chromium project.

YoutubeView — Search and view Youtube videos.

16.4 Creating your own sketch

To be able to create user sketches, some experience in such technologies as JavaScript / JSON / HTML / CSS is needed. QML is a mixture of these technologies. Its code structure reminds JSON code, which consists of elements that are hierarchically put one into another and contain properties in “name: value” pairs, interspersed with standard and full-featured JavaScript code.

The simplest way to create your own sketch is to open one of the supplied sketches, save it in a new file, and then you can proceed with code editing. Either you can press “New” button in the UI window of any sketch and start from a blank slate. You can use embedded editor or any other text editor, but for ease of use, I would recommend you “Qt Creator” (this software is free). To apply changes in the source code, press “Reload” button or “Save & Reload” button, if you have used embedded editor.

A minimal operating sketch looks as follows:

```
import QtQuick 2.9
Item{
}
```

Root element should not contain ”x”, “y”, “width”, “height” and other geometry-determining properties. Root element geometry is set by MAPIO. The entire sketch code should be placed in the single root element.

“ApplicationWindow” or “Window” **MAY NOT** be root elements. To adapt existing source code, you can try to rename these elements as “Item” or “Rectangle”.

To support specific functions and possibilities related to MAPIO operation, add the following string at the beginning of the source code:

```
import mapio.support 1.0
```

List of the provided extensions:

1) SourceProperty element — Sketch properties description that will be available in MAPIO interface and external control system for OSC, MIDI, etc. It can be described anywhere in the sketch source code.
2) MAPIO object with ContentForMapio, ContentForMapioUI, pixelRatio constants.
3) MapioStyle object — Includes constant values for the current MAPIO color palette and style. It includes the following colors — Control, Hovered, Selected, Activepoint, UIText, and the current interface Font.
4) OSCReciever, OSCSender elements and OSCMsg class necessary for receiving and sending OSC messages. Usage example can be found in OSCRecieveDemo / OSCSendDemo sketches.
5) DocumentHandler element — enhance text editor capabilities; it is used in the "Text" sketch.

SourceProperty element description

"SourceProperty" element is inherited from the "Item" element.

Properties:

```
value: var
valuetype: PropertyValueType
name: string
description: string
oscName: string
valueMax: var
valueMin: var
showInMapio: bool
valuesList: var
```

"PropertyValueType" enumeration contains a list of supported data types:

```
SourceProperty.Real,
SourceProperty.Number,
SourceProperty.Boolean,
SourceProperty.String,
SourceProperty.Button,
SourceProperty.Color,
SourceProperty.Selector
```

Example 1 from "AnalogClock" sketch:

```
    SourceProperty{
        id: secondHandColorPty
        name: "Second hand color"
        valuetype: SourceProperty.Color
        value: "#FFF02020"
    }
```

Example 2 from "AnalogClock" sketch:

```
    SourceProperty{
        id: shiftValuePty
        name: "Time shift"
        valuetype: SourceProperty.Number
        value: 0
        valueMax: 12
        valueMin: -12
    }
```

Example 3 from "ColorsGrid" sketch:

```
    SourceProperty{
        id: colorAminationModePty
        name: "Animation mode"
        value: 0
        valuetype: SourceProperty.Selector
        valuesList: ["No animation", "HUE rotate", "Random flashing"]
        onValueChanged: {
            resetAnimation();
        }
    }
```

MAPIO object description

By default, sketch execution result is entirely used as media source, in the same way UI window displays the entire sketch. However, in some cases it is necessary to specify what area will be output as slice contents and what area is either a controlling one along with UI or auxiliary. In these cases, two key words are used for specifying areas to MAPIO. These key words are assigned to the “objectName” property of any element:

Mapio.ContentForMapio — Marks an element whose geometry is used as a source for MAPIO.

Mapio.ContentForMapioUI — Marks an element whose geometry is used as a UI window.

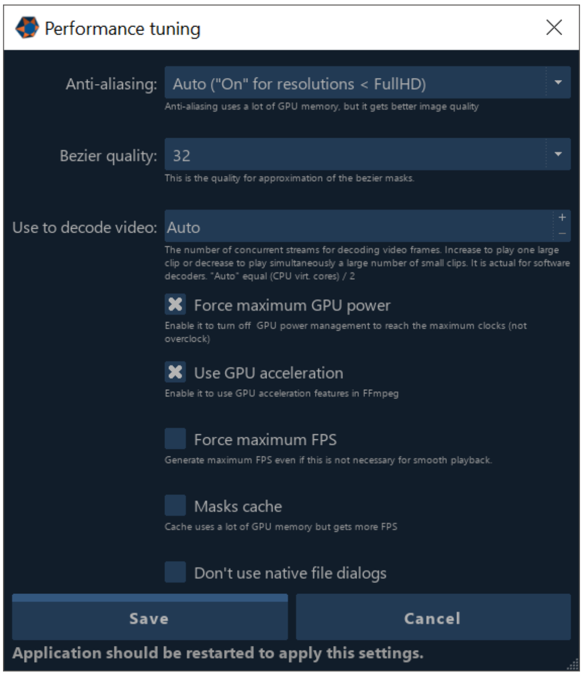
In a sketch you can define only one element of each type.

Example: “GLSLShaderToy” sketch extract

```
…
Item{
    property bool inEditMode: false

    Rectangle {
        id: renderZone
        anchors.fill: parent
        color: colorBGPty.value
        objectName: Mapio.ContentForMapio
…
```

17 Performance settings



Anti-aliasing: Auto ("On" for resolutions < FullHD)

Anti-aliasing uses a lot of GPU memory, but it gets better image quality

Bezier quality: 32

This is the quality for approximation of the bezier masks.

Use to decode video: Auto

The number of concurrent streams for decoding video frames. Increase to play one large clip or decrease to play simultaneously a large number of small clips. It is actual for software decoders. "Auto" equal (CPU virt. cores) / 2

☒ Force maximum GPU power

Enable it to turn off GPU power management to reach the maximum clocks (not overclock)

☒ Use GPU acceleration

Enable it to use GPU acceleration features in FFmpeg

□ Force maximum FPS

Generate maximum FPS even if this is not necessary for smooth playback.

□ Masks cache

Cache uses a lot of GPU memory but gets more FPS

□ Don't use native file dialogs

Application should be restarted to apply this settings.

This setting window contains several application parameters that affect the nature and performance of operation. Changes can be saved by pressing “Save” button, though to apply them you will need to restart the application.

Anti-aliasing — For pixel smoothing. This improvement is particularly necessary for low resolutions, put it requires additional GPU computing resources.

Bezier quality — Approximation quality for smooth Bezier curves used in vector mask rendering. The higher the number, the better the quality. However, it requires significant computing resources.

Use to decode video — Preferential quantity of parallel streams for decoding file process. It is important to remember that not all decoders support multi-stream decoding. “Auto” mode is suitable for most situations.

Force maximum GPU power — An option that includes GPU driver profile setting to force maximum performance mode regardless of GPU load. Nvidia and AMD video card compatible.

Use GPU acceleration — An option for enabling / disabling GPU usage to decode formats supported by your GPU. It does not apply to HAP format since this decoder in any case operates with GPU acceleration.

Force maximum FPS — Change rendering method. When your frame is already output on the monitors it allows you not to wait for project update (for example, for a new frame to come from the decoder) but immediately proceed with the repeated project rendering. As a rule, FPS and GPU load are higher in this mode but sometimes you can archive better smoothness. However, FPS is limited from the above by the monitor / projector vertical scanning.

Masks cache — Use cache for vertical mask render. Thus, at the expense of greater memory consumption you can slightly increase performance, if you have multiple masks and / or multiple slices with masks.

Don’t use native dialogs — This option allows not to use system standard dialog windows either to open or save files. In some cases, the use of these dialogs can lead to instability, particularly if you have installed different extensions, improvements and Explorer plugins.

18 Command-line parameters

MAPIO parameters

Usage: C:\Program Files\Visution\MAPIO2 Pro\Mapio2Pro.exe [options] [file]

Where [file] is a project that should upload.

[options]:

- **--init** — Initialize MAPIO in the system; used only at application installation.
- **--agent** — Link to an external agent (FFGL plugin); used at FFGL-mode startup.
- **--agent-load** — Initialize an external agent (FFGL plugin) and output; used at FFGL-mode startup.
- **--disablerestore** — Disable recovery system. This key is needed in case of autonomous operation. When running from MAPIO Observer, this key is added by default.
- **--opendisplay** — Open a simple screen window in full-screen mode after your project has been uploaded.
- **--displaynumber** — Screen number for the '--opendisplay' option.
- **--vcodecThreads** — Stream quantity for video decoder. When this value is set, a similar parameter in "Performance" settings will be ignored.
- **--network** — Run MAPIO in network mode. The value can be either 'master' or 'slave'.
- **--openmdisplay** — Run "Multi-display" mode after your project has been uploaded.
- **--ipgap-mult** — A multiplier that corrects "Inter-packet gap" value to handle a PixelPusher device.
- **--debug** — Run debug mode to log more detailed information.
- **--minimized** — Run MAPIO with the window minimized to the taskbar.
- --**noQuitConfirm** — Disable the confirmation dialog box when exiting.

MAPIO Observer parameters

MAPIO Observer has several parameters or commands to control its operation:

- **start** — Start (resume) application control process.
- **stop** — Stop control process and terminate MAPIO. MAPIO Observer should be running to be able to transmit these commands. It is not possible to run several MAPIO Observer instances.
- **quit** — Terminate MAPIO Observer.

Usage: C:\Program Files\Visution\MAPIO2 Pro\MapioObserver.exe [command]

19 Recommendations

HAP-codec recommended: https://github.com/vidvox/hap-qt-codec/releases/

Hap / Hap Q / Hap Alpha formats are supported. Please note that Hap Q Alpha(HapM) format is **NOT** supported.

HAP (HAP Q) codec requires significant disk subsystem capacity therefore it is recommended to place similar files on Solid State Drive (SSD).

If disk subsystem resources are limited, it is recommended to use MJPEG/PhotoJPEG codec. Or if your GPU supports h264/h265 acceleration, a medium bitrate file with frequent keyframes is a good option too.

To archive optimal playback smoothness, video file frame frequency must be multiple to projector / monitor scanning frequency, e.g. for a projector operating at 60Hz the perfect file FPS will be either of 30 or 60 frames per second, while a file with a frequency of 25 frames per second will ideally play at a scanning frequency either of 50Hz or 75Hz.

20 Licenses

20.1 Mapio EULA

IMPORTANT, PLEASE READ CAREFULLY. THIS IS A LICENSE AGREEMENT

This SOFTWARE PRODUCT «MAPIO 2»is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. This SOFTWARE PRODUCT «MAPIO 2»is licensed, not sold.

End User License Agreement

This End User License Agreement ("EULA") is a legal agreement between you (either an individual or a single entity) and Ivan Y. Ryabov with regard to the copyrighted Software (herein referred to as "SOFTWARE PRODUCT" or "SOFTWARE") provided with this EULA. The SOFTWARE PRODUCT «MAPIO 2» includes computer software, the associated media, any printed materials, and any "online" or electronic documentation. Use of any software and related documentation ("Software") provided to you by Ivan Y. Ryabov in whatever form or media, will constitute your acceptance of these terms, unless separate terms are provided by the software supplier, in which case certain additional or different terms may apply. If you do not agree with the terms of this EULA, do not download, install, copy or use the Software. By installing, copying or otherwise using the SOFTWARE PRODUCT, you agree to be bound by the terms of this EULA. If you do not agree to the terms of this EULA, Ivan Y. Ryabov is unwilling to license the SOFTWARE PRODUCT «MAPIO 2»to you.

1. Eligible Licensees. This Software is available for license solely to SOFTWARE owners, with no right of duplication or further distribution, licensing, or sub-licensing. IF YOU DO NOT OWN THE SOFTWARE, THEN DO NOT DOWNLOAD, INSTALL, COPY OR USE THE SOFTWARE.

2. License Grant. Ivan Y. Ryabov grants to you a personal, non-transferable and non-exclusive right to use the copy of the Software provided with this EULA. You agree you will not copy the Software except as necessary to use it on a single computer. You agree that you may not copy the written materials accompanying the Software. Modifying, translating, renting, copying, transferring or assigning all or part of the Software, or any rights granted hereunder, to any other persons and removing any proprietary notices, labels or marks from the Software is strictly prohibited. Furthermore, you hereby agree not to create derivative works based on the Software. You may not transfer this Software.

3. Copyright. The Software is licensed, not sold. You acknowledge that no title to the intellectual property in the Software is transferred to you. You further acknowledge that title and full ownership rights to the Software will remain the exclusive property of Ivan Y. Ryabov and/or its suppliers, and you will not acquire any rights to the Software, except as expressly set forth above. All copies of the Software will contain the same proprietary notices as contained in or on the Software. All title and copyrights in and to the SOFTWARE PRODUCT «MAPIO 2»(including but not limited to any images, photographs, animations, video, audio, music, text and "applets," incorporated into the SOFTWARE PRODUCT), the accompanying printed materials, and any copies of the SOFTWARE PRODUCT, are owned by Ivan Y. Ryabov or its suppliers. The SOFTWARE PRODUCT «MAPIO 2»is protected by copyright laws and international treaty provisions. You may not copy the printed materials accompanying the SOFTWARE PRODUCT.

4. Reverse Engineering. You agree that you will not attempt, and if you are a corporation, you will use your best efforts to prevent your employees and contractors from attempting to reverse compile, modify, translate or disassemble the Software in whole or in part. Any failure to comply with the above or any other terms and conditions contained herein will result in the automatic termination of this license and the reversion of the rights granted hereunder to Ivan Y. Ryabov.

5. Disclaimer of Warranty. The Software is provided "AS IS" without warranty of any kind. Ivan Y. Ryabov and its suppliers disclaim and make no express or implied warranties and specifically disclaim the warranties of merchantability, fitness for a particular purpose and non-infringement of third-party rights. The entire risk as to the quality and performance of the Software is with you. Neither Ivan Y. Ryabov nor its suppliers warrant that the functions contained in the Software will meet your requirements or that the operation of the Software will be uninterrupted or error-free. Ivan Y. Ryabov IS NOT OBLIGATED TO PROVIDE ANY UPDATES TO THE SOFTWARE.

6. Limitation of Liability. Ivan Y. Ryabov's entire liability and your exclusive remedy under this EULA shall not exceed the price paid for the Software, if any. In no event shall Ivan Y. Ryabov or its suppliers be liable to you for any consequential, special, incidental or indirect damages of any kind arising out of the use or inability to use the software, even if Ivan Y. Ryabov or its supplier has been advised of the possibility of such damages, or any claim by a third party.

7. Rental. You may not loan, rent, or lease the SOFTWARE.

```
 8.  Upgrades.  If the SOFTWARE is an upgrade from an earlier release or previously released
version, you now may use that upgraded product only in accordance with this EULA.  If the
SOFTWARE PRODUCT «MAPIO 2»is an upgrade of a software program which you licensed as a single
product, the SOFTWARE PRODUCT «MAPIO 2»may be used only as part of that single product package
and may not be separated for use on more than one computer.

 9.  OEM Product Support. Product support for the SOFTWARE PRODUCT «MAPIO 2»IS provided by Ivan
Y. Ryabov.  For product support, please call Ivan Y. Ryabov.  Should you have any questions
concerning this, please refer to the address provided in the documentation.

 10.  No Liability for Consequential Damages.  In no event shall Ivan Y. Ryabov or its suppliers
be liable for any damages whatsoever (including, without limitation, incidental, direct,
indirect special and consequential damages, damages for loss of business profits, business
interruption, loss of business information, or other pecuniary loss) arising out of the use or
inability to use this "Your Company" product, even if Ivan Y. Ryabov has been advised of the
possibility of such damages.  Because some states/countries do not allow the exclusion or
limitation of liability for consequential or incidental damages, the above limitation may not
apply to you.

 11.  Indemnification By You.  If you distribute the Software in violation of this Agreement,
you agree to indemnify, hold harmless and defend Ivan Y. Ryabov and its suppliers from and
against any claims or lawsuits, including attorney's fees that arise or result from the use or
distribution of the Software in violation of this Agreement.
Third-party licenses
```

20.2 Third-party licenses

Qt

Qt [site: qt.io] is licensed under the GNU Lesser General Public License version 3. Qt is a trademark of Qt Company Ltd. and other contributors: http://doc.qt.io/qt-5/lgpl.html

FFmpeg

FFmpeg [site: ffmpeg.org] is licensed under the GNU Lesser General Public License (LGPL) version 2.1 or later.

FTD2

FTD2xx - Copyright © 2001-2011 Future Technology Devices International Limited

```
THIS SOFTWARE IS PROVIDED BY FUTURE TECHNOLOGY DEVICES INTERNATIONAL LIMITED "AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL FUTURE
TECHNOLOGY DEVICES INTERNATIONAL LIMITED BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES LOSS OF USE, DATA, OR PROFITS OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.
FTDI DRIVERS MAY BE USED ONLY IN CONJUNCTION WITH PRODUCTS BASED ON FTDI PARTS.
FTDI DRIVERS MAY BE DISTRIBUTED IN ANY FORM AS LONG AS LICENSE INFORMATION IS NOT MODIFIED.
IF A CUSTOM VENDOR ID AND/OR PRODUCT ID OR DESCRIPTION STRING ARE USED, IT IS THE RESPONSIBILITY
OF THE PRODUCT MANUFACTURER TO MAINTAIN ANY CHANGES AND SUBSEQUENT WHQL RE-CERTIFICATION AS A
RESULT OF MAKING THESE CHANGES.
```

QtAV

QtAV [site: qtav.org] library under the GNU LGPLv2.1

libUSB

[site: libusb.info]

```
Copyright © 2001 Johannes Erdfelt johannes@erdfelt.com
Copyright © 2007-2008 Daniel Drake dsd@gentoo.org
Copyright © 2012 Pete Batard pete@akeo.ie
Copyright © 2012 Nathan Hjelm hjelmn@cs.unm.edu
For more information, please visit: http://libusb.info
This library is free software; you can redistribute it and/or modify it under the terms of the
GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of
the License, or (at your option) any later version.
This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY;
without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See
the GNU Lesser General Public License for more details.
You should have received a copy of the GNU Lesser General Public License along with this
library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor,
Boston, MA 02110-1301 USA
```

MediaInfo

MediaInfo under BSD 2-Clause License

```
Copyright (c) 2002-2016, MediaArea.net SARL. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted
provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions
and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of
conditions and the following disclaimer in the documentation and/or other materials provided
with the distribution.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
```

NewTek NDI

NewTek NDI (“Network Device Interface”) is a trademark of NewTek, inc.

USBDMX

```
Copyright (c) 2010, Alan Ott, Signal 11 Software. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted
provided that the following conditions are met:
    * Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above copyright notice, this list of
conditions and the following disclaimer in the documentation and/or other materials provided
with the distribution.
    * Neither the name of Signal 11 Software nor the names of its contributors may be used to
endorse or promote products derived from this software without specific prior written
permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.
```

Autodesk FBX

Autodesk® FBX® SDK is developed by Autodesk, Inc.

```
Copyright Autodesk, Inc. All rights, reserved. Such code is provided “as is” and Autodesk, Inc.
disclaims any and all warranties, whether express or implied, including without limitation the
implied warranties of merchantability, fitness for a particular purpose or non-infringement of
third party rights. In no event shall Autodesk, Inc. be liable for any direct, indirect,
incidental, special, exemplary, or consequential damages (including, but not limited to,
procurement of substitute goods or services; loss of use, data, or profits; or business
interruption) however caused and on any theory of liability, whether in contract, strict
liability, or tort (including negligence or otherwise) arising in any way out of such code.
```

Crypto++

The Crypto++ Library under Boost Software License v1.0

Spout

Spout SDK is licensed under the Simplified BSD license.

RtMidi

RtMidi: real-time MIDI i/o C++ classes

```
Copyright (c) 2003-2016 Gary P. Scavone
Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:
  *  The above copyright notice and this permission notice shall be included in all copies or
substantial portions of the Software.
  *  Any person wishing to distribute modifications to the Software is asked to send the
modifications to the original developer so that they can be incorporated into the canonical
version.  This is, however, not a binding provision of this license.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING
BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
```