

## Installation

MainActor needs some additional software packages to run correctly:

- o Software required by the Windows version.
- o Software required by the OS/2 version.

## Windows Installation

MainActor needs **DirectX**, especially **DirectDraw** and **DirectSound**, for playback.

If you are running Windows NT 4.0 or Windows95 Release 2, DirectX is already installed, otherwise download the latest DirectX runtime release from the Microsoft Homepage at <http://www.microsoft.com>.

Also, your graphics card manufacturer should have DirectX enabled drivers available.

## **OS/2 Installation**

This archive contains versions of TIMER0.SYS and DIVE. They are not installed automatically and will have to be installed by hand.

Please note that these files are copyrighted by IBM and are included in the MainActor archive by agreement of IBM Germany.

## Installing DIVE

**DIVE** is used by MainActor for displaying video. The archive **dive.zip** contains the DIVE related files. Please copy the \*.LUT files into the directory and the dive.dll into the directory.

Make sure that you do not have a newer version installed, Warp v+4 has DIVE already installed and you dont need to replace it.

## Installing TIMER0.SYS

**TIMER0.SYS** is needed by MainActor for syncing audio and video streams. If TIMER0.SYS is not installed, video and audio streams will not be synced correctly.

To install TIMER0.SYS, extract the files from the **timer0.zip** archive and install them with the device driver installer (DDINSTAL) utility. Make sure to remove the readonly attributes of the \*.sys files first with: **attrib -r \*.sys**.

Make sure that you do not have a newer version installed, Warp v4 has TIMER0.SYS already installed and you don't need to replace it.

## Introduction

MainActor is a multimedia processing package. It allows you to load, edit, play and save / convert animations, pictures and sounds of various formats and of any size.

MainActor features animation processing and editing functions and is the most powerful part of this package.

MainView is the external player of MainActor. It is useful when you only want to play an animation and don't want to load MainActor just for that. MainView can also be associated to other programs.

Both MainActor and MainView use the same modules for processing the various formats and are therefore quite flexible and expandable.

A sequencer add-on for MainActor is in work.

## Available Platforms

MainActor is currently available for Windows 95 / NT 4.0 and OS/2.

We hope to support other platforms as well one day. All systems for which IBM's excellent VisualAge C++ development platform is available, is a possible and easy target platform, like AIX, Solaris, HP-Unix etc.

MainActor always uses the fastest video and audio playback system available on each platform, like **DIVE** under OS/2 and **DirectX** under Windows 95 / NT 4.0.

**If you decide to register MainActor, you automatically get a license for all platforms MainActor is or will be running on.**

## **Additional Products**

Additional animation related products :

- o mainCODEC/2 for OS/2



## **mainCODEC/2**

**mainCODEC/2** is part of MainActor but is distributed as a separate archive.

It contains a subset of the MainActor modules as MMPM/2 codecs. These codecs allow the supported formats to be used within the native OS/2 multimedia system, for example inside the player or VideoIN.

mainCODEC/2 can be downloaded from our Web Pages (<http://www.mainconcept.de>) and our BBS (+49-(0)241-4090446).

Also, it should be available per ftp at [hobbes.nmsu.edu](ftp://hobbes.nmsu.edu).

## MainActor

MainActor is the animation processing part of the MainActor package.

You can load, edit and play/show every animation, picture and sound format for which MainActor has the proper loader module. You can save any animation, picture and sound format for which MainActor has the proper saver module.

The currently supplied modules are listed by selecting the **Loader Modules...** and **Saver Modules...** items of the **Help** menu. For a list of feature of these modules, see [The Modules](#).

MainActor features detailed online help. For additional information, please have a look at the [User Guide](#).

## Start Options

You can pass MainActor the names of projects to load.

For example **mactr.exe gfx.mov pix.pcx** would load the animation test.mov and the picture art.pcx right after MainActor has started.

## FAQ

The **frequently asked questions** document for MainActor can be requested by sending email to [mafaq@mainconcept.de](mailto:mafaq@mainconcept.de).

## **User Guide**

This guide explains the functionality of MainActor as well as the vocabulary used by the online help and this documentation.

## Working with Projects

A project is either an animation, one or several pictures or a sound. An animation or sound is always a project of its own, the name of the project will be the name of the animation/sound.

Pictures are handled differently. All pictures of a given format are grouped together in one project, the name of the project will be the name of the format of the pictures, like IFF or JPEG. This way you can group any number of pictures together and convert them into an animation.

Every project has its own pop-up menu which is identical to the **Project** menu. If you press the right mouse button in the containers whitespace, the **File** menu will be shown as a pop-up menu.

## Loading Projects

There are several ways how to load a project.

Under **OS/2** we included our own file requester with MainActor, this way you can select any number of files very easily, for example load a whole directory or a range of files. If you choose to use the system file requester you will have to select any single file per hand.

However, under **Windows** we simply use the system file requester, as it is powerful enough to simply select any number of files directly.

A different way is to drag and drop the file icons over the project list. This way you can also load any number of files.

Picture lists can also be loaded by simply selecting a picture inside the list. MainActor will automatically recognize that the picture is part of a list and will optionally load in the rest of the pictures.

## Playing Projects

Simply double click on the project or select the **Play Project...** item of the **Project** menu.

For additional help for the functionality available during playback, consult the online help of the display window.

Under **Windows**, you also have full screen functionality, for a description of the possible key commands please have a look at the online help of the display window (simply press F1 in window playback mode).



## Removing Projects

Use the **Remove Project** item of the **Project** menu if you want to remove a specific project. Also, if you want to remove all currently loaded frames, you can use the **Remove All** item of the **File** menu.

Under **OS/2** you can also drag and drop the project icon over a shredder object.

## Working with Frames

The frames of the current project are displayed in the frame list. A frame is an image which is either part of an animation or a picture in a picture list.

If the frame is part of a picture list, the frames can be sorted, single or various pictures can be removed or their locations can be changed by drag and drop. This is not possible for animation frames, as these are mostly dependent on a each other in a linear fashion.

The icons of frames in an animation indicate if the frame is a key frame or a delta frame.

Frames have the **Frames** menu available as a pop-up menu.

## Key/Delta Frames

Frames inside an animation can be stand alone frames, these frames do not need another frame to decompress correctly. These frames are called **key frames**.

**Delta frames** need other frames (mostly the previous frames) to decompress. This is done to reduce the size of the frames inside an animation, as only the differences from the previous to the next frame have to be processed.

First generation animation formats (like FLI, FLC, IFF-Anim) mostly have a key frame as the first frame and all other frames are delta frames. Second generation animation formats, like AVI and Quicktime mostly have key frames every few seconds. This enables the playback software to quickly find a specific position inside the stream and to skip frames when playback gets too slow.

Inside MainActor, **gray frame icons** in the frame list indicate key frames, **transparent icons** indicate delta frames.

## Selecting Frames

Frames can be simply selected with the mouse or by selecting one of the **Select** or **Deselect** sub menu items of the **Edit** menu.

The items of the **Frames** menu often work on all selected frames, like the items for removing and saving.

## Showing Frames

A frame can be displayed by simply double clicking on it or by selecting the **Show Frames...** item of the **Frames** menu.

## Sorting Frames

Frames can be sorted by hand or automatically by their size or filename.

To sort the frames by hand, simply drag and drop the frames to their new location inside the frame list (OS/2 only).

The sub menu items of the **Sort Frames** item of the **Frames** menu sort the frames by their size or filename.

Sorting frames is only possible if the frames are part of a picture list.

## Removing Frames

You can remove the currently selected frames by selecting the **Remove Frames** item of the **Frames** menu or by dragging the frames to a shredder object (OS/2 only).

Removing frames is only possible if the frames are part of a picture list.

## Working with Timecodes

Every frame of an animation or picture list has a timecode associated to it. The timecode controls the time the frame is shown during playback. MainActor displays the timecode in milliseconds, therefore a timecode of 1000 would mean that the frame would be displayed exactly one second.

Projects can have two different kinds of timecodes: Global or local timecodes.

Pictures have always a default local timecode of 1 second.

Converting projects with different timecode methods can be tricky, have a look at the Timecode Translation.



## Global Timecodes

Global timecodes only support one timecode per animation, meaning that every frame has to have the same timecode.

These animation formats, for example AVI/FLI/FLC, often insert additional frames into the animation for timing purposes. These frame are dummy frames and have a size of zero, they just have the purpose to slow down playback for the last 'real' frame.

You can set the global timecode of an animation by selecting the **Global Timecode...** item of the **Project** menu.

## Local Timecodes

Local timecodes can be set for every frame of an animation and allow a different timecode for each frame. Local timecodes can be found in Quicktime and IFF-AnimX animation formats. You can however not set the local timecodes of Quicktime animations inside MainActor.

Additionally, MainActor treats picture lists as if they would have a local timecode feature. You can therefore set the timecodes of pictures. This is useful for quickly making a slide show as a preview of an animation or to set the timecodes prior to saving, so that the resulting animation automatically has the right timecodes associated to it.

You can change the local timecodes of the currently selected frame by choosing the **Local Timecodes...** item of the **Frames** menu.

## **Working with Audio**

MainActor is able to save out sound from animations directly into a sound saver module, like MPEG-Audio or WAV.

You can also convert sound formats, like WAV to MPEG-Audio, but that is just a side effect. The sound loader modules will be very important when the sequencer is ready.

## Notes on Saving

Saving new pictures or animations is quite easy, you can save all frames or just the selected frames of the frame list to the new format.

When saving pictures, MainActor will automatically enumerate the new pictures, like "test0003.bmp". The start index can be set within MainActor.

There are however some things you need to know, like how MainActor deals with Timecode Translations or handles 8Bit Palettes.

## 8Bit Palettes

When you save a new 8Bit (256 color) animation, MainActor will use a customized palette for the whole animation. That means that MainActor will not save a new palette once the source palette changes but tries to create an optimized palette for the new animation which will then be used for all frames.

We think this approach results in smoother animations, especially on platforms which write directly to the color space, like low level DOS, Amiga and MAC viewers.

The generation of the animation is, however, slower, as MainActor has to process all frames to generate the source palette.

It is possible to edit the palette prior to saving.

## Editing 8bit Palettes

You can edit 8bit palettes prior to saving with the palette editor. Useful if you want to optimize the palette, but as MainActor changes the color indices while optimizing the palette it becomes a must when you want to set a specific color index as color 0, for example if you want to save the color as transparent with the **GIF** saver.

With the color editor you always have full control over the palettes you save in your 8bit animations or pictures.

The palette editor is able to load/save your palettes, this way you can load in any custom palette you created with other programs (or MainActor) for saving.

## Timecode Translation

There can be some problems while converting local timecode animations to global timecode animation formats.

Converting global timecode animation formats to local timecode animation formats is easy, because MainActor can set every local timecode to the global timecode of the source project.

This does not work the other way round, as you can set the global timecode of the animation to only one of the local timecodes of the source project. MainActor tries to calculate the best global timecode for the new project and inserts dummy frames behind frames which have higher timecodes than the new global one.

This all sounds a little complicated but we hope MainActor does the job without you even noticing it.

If you are not satisfied with the timing of a certain animation, you can always split it up into pictures, load them as a picture lists and can set the timecodes of the pictures yourself. If you want to add dummy frames in the resulting animation, simply add the same picture various times into the picture list. Of course, a later version of MainActor will have a sequencer which will make this kind of work very easy indeed.

## **MainView**

MainView is the player only part of MainActor. Useful if you just want to play an animation or want to run an animation from within another program.

MainView can play every format for which MainActor has the proper loader module. Saving is not supported from within MainView.



## Start Options

MainView features different options for the different platforms :

- o Start Options for the Windows version.
- o Start Options for the OS/2 version.

## Windows Start Options

Starting MainView by clicking on its icon opens a file requester. Otherwise you can pass the name on the command line.

The syntax for MainView is: **mainview.exe [options] filename.**

If you do not pass any options on the command line, MainView will use the global settings of MainActor.

MainView supports the following options in this release:

### **-(c|cache)**

Cache Data. This option will load all data of the animation into a ram cache before playing. This will speed up animation playback. If not enough free ram is available, MainView will free the already allocated memory and play directly from harddisk/cdrom. This option is off by default.

### **-(v|volume) number**

Set the sound volume, the range for the number is 0-100. Default is 100.

### **-(das|disableautostart)**

This option will stop MainView to start animations automatically.

### **-(r|repeat) number**

The number indicates how many times the project shall be repeated while playing. Default is 1. Note, animation with sound will always only play once.

### **-(fs|fullscreen)**

Enables full screen playback.

### **-(sts|scaletoscreen)**

Only available in full screen mode. The images of the project will always be scaled to the full size of the screen.

### **-(ubm|usebestmode)**

Only available in full screen mode. Normally, MainView takes the screen mode of the windows user interface for playback. This option enables MainView to search for the best mode available for the project on your system.

Example: **mainview -volume 80 -tbl bottom dino.mov** would play the dino.mov animation with a sound volume of 80 and the toolbar would be located at the bottom of the window.

## OS/2 Start Options

Starting MainView by clicking on its icon opens a file requester. Otherwise you can pass the name on the command line.

Note that you can also drop the icon of the animation or picture on the MainView icon.

The syntax for MainView is: **mainview.exe [options] filename.**

If you do not pass any options on the command line, MainView will use the global settings of MainActor.

MainView supports the following options in this release:

### **-(c|cache)**

Cache Data. This option will load all data of the animation into a ram cache before playing. This will speed up animation playback. If not enough free ram is available, MainView will free the already allocated memory and play directly from harddisk/cdrom. This option is off by default.

### **-(v|volume) number**

Set the sound volume, the range for the number is 0-100. Default is 80.

### **-(das|disableautostart)**

This option will stop MainView to start animations automatically.

### **-(r|repeat) number**

The number indicates how many times the project shall be repeated while playing. Default is 1.

Example: **mainview -volume 80 -tbl bottom dino.mov** would play the dino.mov animation with a sound volume of 80 and the toolbar would be located at the bottom of the window.

## The Modules

MainActor supports various modules and codecs. The modules are updated frequently and more will be added in future releases.

- o Loader Modules
- o Saver Modules

## Loader Modules

Currently supplied loader modules:

- o AVI
- o BMP
- o DL
- o FLC/FLI
- o GIF
- o GIF-Anim
- o IFF
- o IFF-Anim3/5/7/8/J
- o JPEG
- o MPEG
- o MPEG-Audio
- o PCX
- o Quicktime
- o WAV

# AVI

## Type:

Animation (Supports Sound)

## Codecs:

- o Cinepak
- o Intel Indeo v2.1 / v3.1 / v3.2 / v4.1 (Windows Only)
- o Microsoft RLE
- o Microsoft Video 1
- o Motion JPEG
- o Ultimotion
- o Uncompressed

## Notes:

- o AVI uses global timecodes.
- o If you encounter an AVI animation which MainActor does not play correctly or uses an codec unknown by MainActor, please contact us.

# BMP

## Type:

Picture

## Codecs:

- o RunLength Encoding
- o Uncompressed

## Notes:

None

## **DL**

### **Type:**

Animation

### **Codecs:**

- o Type 1 (Untested)
- o Type 2

### **Notes:**

- o Type 1 and 2 DL animations are always uncompressed.
- o Type 3 animations are not supported, in fact we did never even see one.



## **FLI/FLC**

### **Type:**

Animation

### **Codecs:**

- o ByteRun
- o Byte LineCoding
- o Clear Screen
- o Uncompressed
- o Word LineCoding (FLC only)

### **Notes:**

- o FLI / FLC animations are limited to 256 colors.
- o FLI only supports resolutions of 320 \* 200.
- o FLI / FLC use global timecodes.

# GIF

## Type:

Picture

## Codecs:

- o GIF-LZW

## Notes:

- o GIF supports 256 colors only.
- o The LZW decoding/encoding is officially licensed from UniSys.

## **GIF-Anim**

### **Type:**

Animation

### **Codecs:**

- o GIF-LZW

### **Notes:**

- o GIF-Anim supports local timecodes.
- o GIF supports 256 colors only.
- o The LZW decoding/encoding is officially licensed from UniSys.

## IFF

### Type:

Picture

### Codecs:

- o ByteRun Encoding

### Notes:

Supports HAM6/8 and EHB.

## **IFF-Anim3/5/7/8/J**

### **Type:**

Animation

### **Codecs:**

- o ByteRun Encoding
- o Various LONG and WORD delta codecs

### **Notes:**

- o Supports HAM6/8 and EHB.
- o Supports local timecodes.

# JPEG

**Type:**

Picture

**Codecs:**

- o JPEG

**Notes:**

Based on the source code of the Independent JPEG group.

# MPEG

**Type:**

Animation

**Codecs:**

- o MPEG-I
- o MPEG-II
- o Audio Layer I/II/III

**Notes:**

- o MainActor should be able to load in every MPEG. If you experience any problems with an MPEG file, please contact us.

## **MPEG-Audio**

### **Type:**

Sound

### **Codecs:**

- o MPEG Audio Layer I/II/III

### **Notes:**

The audio file is splitted into 1 second parts when loaded into MainActor. This makes it easier to access specific parts of the audio file.



## PCX

### Type:

Picture

### Codecs:

- o RunLength Encoding

### Notes:

None

## Quicktime

### Type:

Animation (Supports Sound)

### Codecs:

- o Apple Animation
- o Apple Graphics
- o Cinepak
- o Intel Indeo v2.1 / v3.1 / v3.2
- o Motion JPEG

### Notes:

- o Quicktime supports local timecodes.
- o If you encounter an Quicktime animation which MainActor does not play correctly or uses an codec unknown by MainActor, please inform us.

## WAV

### Type:

Sound

### Codecs:

- o Pulse Code Modulation

### Notes:

None

## Saver Modules

Currently supplied saver modules:

- o AVI
- o BMP
- o FLC/FLI
- o GIF / GIF-Anim
- o JPEG
- o MPEG
- o MPEG-Audio
- o Video Data
- o Quicktime
- o WAV

# AVI

## Type:

Animation (Supports Sound)

## Codecs:

- o Cinepak (Only Windows)
- o Intel Indeo v3.1 (OS/2) / v3.2 (Windows) / v4.1 (Windows)
- o Microsoft RLE
- o Motion JPEG
- o Ultimotion (Only OS/2)
- o Uncompressed

## Options:

### Motion JPEG Quality Compression Quality

Defines the JPEG compression quality for the Motion JPEG codec.

### MMPM/2 or ICM Compression Quality

The compression quality for the multimedia codec subsystem (**MMPM/2 for OS/2** and **ICM for Windows**). -1 is the default quality of the codec.

### MMPM/2 or ICM Key Frame Rate

The Key frame rate of the system codec. 1 means every frame is a key frame, 2 means every second frame is a key frame etc. -1 is the default rate of the codec.

## Notes:

- o MainActor generated AVIs have been compatibility tested on nearly all platforms, including Windows 3.1 / W95 / NT, OS/2 and ActiveMOVIE.

# BMP

## Type:

Picture

## Codecs:

- o Uncompressed
- o RunLength Encoded

## Options:

None

## Notes:

None

## **FLI / FLC**

### **Type:**

Animation

### **Codecs:**

- o ByteRun
- o Byte LineCoding (FLI Only)
- o Word LineCoding (FLC Only)

### **Options:**

None

### **Notes:**

- o FLI / FLC animations are limited to 256 colors.
- o FLI only supports resolutions of 320 \* 200.
- o FLI / FLC use global timecodes.

## **GIF / GIF-Anim**

### **Type:**

Picture / Animation

### **Codecs:**

- o GIF-LZW

### **Options:**

#### **Interlace**

Enables interlace compression.

#### **Background Index**

Identifies the color index which will be used as the background color of the image.

#### **Transparent Index**

Identifies the color index which will be set to transparent, i.e. the background shines through.

#### **Netscape Loop**

**(GIF-Anim Only)** Specifies how many times the animation will be repeated inside the Netscape browser, Microsoft support this option too.

### **Notes:**

- o GIF supports 256 colors only.
- o The GIF-Anim saver supports local timecodes, it is therefore possible to convert AVI or Quicktime animations to GIF-Anim without loss of timing information.
- o The LZW decoding/encoding is officially licensed from UniSys.



# JPEG

## Type:

Picture

## Codecs:

- o JPEG

## Options:

## Compression Quality

The quality used for compressing the images.

## Notes:

Based on the source code of the Independent JPEG group.

# MPEG

## Type:

Animation

## Codecs:

- o MPEG-I NTSC (30 fps)
- o MPEG-I PAL (25 fps)
- o MPEG-II NTSC (30 fps)
- o MPEG-II PAL (25 fps)

## Options:

### Ignore Timecodes

MPEG streams always have a fixed frames per second rate. When this option is on (which is the default) then MainActor simply inserts the frames into the stream and ignores the timecodes of the source project. If this option is disabled, MainActor inserts additional frames into the stream to preserve the timing of the source project, but this results in bigger MPEG files. This option will always be disabled when saving MPEGs with audio information.

### MPEG-I Bits per Second

Defines the bit rate of the generated MPEG-I stream.

### MPEG-II Bits per Second

Defines the bit rate of the generated MPEG-II stream.

### Interlaced

If enabled, this option will activate the interlace mode. Animation will then consist of fields rather than frames.

### Top Field

This field can either be one or zero and is only valid if the Interlaced option is on. It defines the field which is first encoded, either the first (0) or second (1) line of each frame.

### Audio Sampling Rate

Select the sampling rate of the MPEG-Audio. If the source sampling rate is different, the audio data will be automatically converted to the chosen frequency.

### Audio Checksums (error protection)

If this option is enabled, MPEG-Audio will contain checksums to prevent errors while playback.

## Notes:

- o None.

## **MPEG-Audio**

### **Type:**

Sound

### **Codecs:**

- o MPEG Audio Layer II

### **Options:**

#### **Audio Bitrate**

Select the bitrate of the resulting audio stream.

#### **Audio Sampling Rate**

Select the sampling rate of the MPEG-Audio. If the source sampling rate is different, the audio data will be automatically converted to the chosen frequency.

#### **Audio Checksums (error protection)**

If this option is enabled, MPEG-Audio will contain checksums to prevent errors while playback.

### **Notes:**

- o None.

## Video Data

### Type:

Picture

### Codecs:

- o Video binary

### Options:

None

### Notes:

Splits the compressed video data from animations. Very useful to save standalone image data from animation formats. For example motion JPEG AVIs and Quicktimes contain standalone JPEG image data for each frame. Using this saver module you can save them into single files without having to decompress and then compress them again.

## Quicktime

### Type:

Animation (Supports Sound)

### Codecs:

- o Cinepak (Windows only)
- o Intel Indeo v3.1 (OS/2) / v3.2 (Windows) / v4.1 (Windows)
- o Motion JPEG
- o Uncompressed

### Options:

#### **Motion JPEG Quality Compression Quality**

Defines the JPEG compression quality for the Motion JPEG codec.

#### **MMPM/2 or ICM Compression Quality**

The compression quality for the multimedia codec subsystem (**MMPM/2 for OS/2** and **ICM for Windows**). -1 is the default quality of the codec.

#### **MMPM/2 or ICM Key Frame Rate**

The Key frame rate of the system codec. 1 means every frame is a key frame, 2 means every second frame is a key frame etc. -1 is the default rate of the codec.

### Notes:

- o MainActor generated Quicktimes have been compatibility tested on many platforms, including Quicktime for Windows, ActiveMOVIE and MacOS.

## WAV

### Type:

Sound

### Codecs:

- o Pulse Code Modulation

### Options:

None

### Notes:

Use this module to split sound from animations.

## Miscellaneous

Miscellaneous Topics:

- o Credits
- o Registration
- o Mailing Lists

## Credits

We have to thank the following people:

- o Mark Podlipec. Some of the AVI and Quicktime decoders included in MainActor are based upon his work.
- o The Independent JPEG group for their JPEG source.
- o The folks at IBM for their great VisualAge C++ environment.
- o The MPEG Software Simulation Group for their MPEG source.
- o Norbert Bogenrieder for the artwork and the icons.



## Registration

If you decide to register **MainActor**, we will send you a personal serial number which will be able to unlock all future versions of this software **on all available platforms**.

MainActor is currently running on OS/2 and Windows 95 / NT 4.0.

The fee for **MainActor** is 99,- DM (appr. \$60) or any equivalent sum in your local currency. Please no foreign checks (except EuroChecks). We do take Master/EuroCard and VISA.

You can contact us at :

MainConcept, GbR Moenig/Zabel  
Hermann-Heusch-Platz 3  
D-52062 Aachen  
Germany

Tel: +49-(0)241-4090444  
FAX: +49-(0)241-4090445  
BBS: +49-(0)241-4090446

Email: [info@mainconcept.de](mailto:info@mainconcept.de)  
<http://www.mainconcept.de>

You can also order at BMT Micro (<http://www.bmtmicro.com>). Please have a look at the file **bmtmicro.txt** included in this package.

Online registration for the Windows version is also available at <http://www.buydirect.com>.

## Mailing Lists

We are running mailing lists for every MainActor version. If you are interested in the product we recommend to join the list to get the latest information on MainActor.

- o [Windows Mailing List](#).
- o [OS/2 Mailing List](#).

You can get the frequently asked question (FAQ) text file for MainActor by sending email to **mafaq@mainconcept.de**.

## Windows Mailing List

To join to the mailing list send email to **steward@mainconcept.de**.

### **SUBSCRIBE win-MainActor <your email address>**

Adds your email address to the mailing list.

### **UNSUBSCRIBE win-MainActor <your email address>**

Deletes your email address from the mailing list.

### **LISTS**

Will send a list of all available mailing lists on our server.

### **HELP**

Will send the help page for all available commands on our server.

## **OS/2 Mailing List**

To join to the mailing list send email to **steward@mainconcept.de**.

**SUBSCRIBE os2-MainActor <your email address>**

Adds your email address to the mailing list.

**UNSUBSCRIBE os2-MainActor <your email address>**

Deletes your email address from the mailing list.

**LISTS**

Will send a list of all available mailing lists on our server.

**HELP**

Will send the help page for all available commands on our server.

## History

The history lists the last three revisions of MainActor, together with the bug fixes, improvements and new features of every revision.

- o [MainActor v1.00](#)
- o [MainActor v1.01](#)
- o [MainActor v1.10](#)

## **Version 1.00**

### **General improvements :**

- o Lots of bugfixes, both in the modules and in MainActor itself. All known/reported bugs of v0.80 were fixed.
- o New WAV loader module.
- o New MPEG-I/II stream loader and MPEG-I/II animation saver modules.
- o New Multi-language support. Currently supported languages are english and german.
- o Some improvements to the interface, i.e. toolbar, logo etc.
- o Much faster JPEG support, both for still and motion JPEG.
- o Support for JPEG frames with missing huffman tables, these are mostly generated by video capture hardware. We tested this feature with animations generated by FAST and Miro hardware.
- o Improvements to the Palette Editor, i.e. you can now load / save palettes.
- o New Save Options window and improvements to the Save Window. New options include support for interpolation while scaling and error diffusion when dithering.

### **OS/2 specific improvements :**

- o Video/audio synchronization would loose sync of about 3/4 tenths of a second for every 15 minutes of video. This has been fixed but was only noticeably when playing animations with at least five minutes playback time.

### **Windows specific improvements :**

- o New Video Capture Interface.

## **Version 1.01**

### **General improvements :**

- o Some bugs crept into the v1.00 release. There were bugfixes for Motion JPEG AVIs, QuickTime to AVI conversion, the GIF-Anim loader and some more.
- o New picture list feature. You can now simply select a single picture from a picture list. MainActor will recognize that it is part of a list and asks if you want to load the following pictures in the list.
- o The BMP loader module now also loads OS/2 bitmaps.
- o The BMP saver module now supports compression for 8bit bitmaps.

### **OS/2 specific improvements :**

- o None.

### **Windows specific improvements :**

- o None.

## **Version 1.10**

### **General improvements :**

- o Full MPEG support. Audio Layer I/II/III is now supported for loading and Audio Layer II for saving. MPEGs can now be de/encoded with full audio support.
- o Easier loading of picture lists. Simply select a picture in the list and MainActor will ask you if you want to load the rest of the picture in the list.
- o Picture can now also be sorted in inverse mode.
- o New user interface for the player.
- o Support for interlaced Motion JPEGs from Miro.
- o Faster loading of AVIs and FLI/FLC animations. FLI/FLC animations will now be streamed directly by the player.
- o Lots of smaller (speed) improvements and bug fixes.

### **OS/2 specific improvements :**

- o None.

### **Windows specific improvements :**

- o Support for Drag and Drop of external files. You can now drag and drop file icons into the Project List of MainActor or into MainView's window.





