Negatron

The fun side of electrons

Negatron

Le côté amusant des électrons

负电子

电之有趣的一面

Contents

If you just need to know how to use Negatron, please go directly to the Emulation Management *chapter.*

Overview

After Alan Turing theorised the principles of modern computers in his 1936 paper *On Computable Numbers*, the idea of on-screen games quickly made its way as soon as the very first digital computers were made available to scientists and engineers. But it's not until Nolan Bushnell and Ted Dabney released the coin-operated video game *Computer Space* in 1971 and then *Pong* in 1972 that video gaming became a widespread activity.

Since the mid-1990s, as computers became powerful enough and with the advent of the Internet, more and more emulators have been produced to simulate video games of other platforms on personal computers. Often released for free, most emulators reproduce the behaviour of a single machine, e.g. an arcade game or a console. *MAME* is one of those exceptions, which allows to play most of the video games ever released since 1971 on a single machine.

After reviewing how *MAME* came to be and why *Negatron* was made, the last part of this document explains how to use *Negatron* together with *MAME*.

About MAME

THE ORIGINS

MAME is a multiple arcade machine emulator, originally for MS-DOS, on which Nicola Salmoria began working starting December 24th, 1996, in order to merge several emulators running arcade games based on the same kind of machine units as the original Pac-Man. It was first released on February 5th, 1997 with v0.1. Except during 6 months in 1997, by handing matters to Mirko Buffoni while honouring his military duties, Salmoria successfully led the project until 2003 to become the most versatile emulator, being able to run nearly any kinds of arcade games, helped by a growing team of enthusiasts.

FROM AN ARCADE-ONLY TO A UNIVERSAL EMULATOR

As *MAME* remained focused on arcade machines, some maintainers of the Mac version wanted to use its versatile core to emulate other kinds of machines, them being consoles, computers, calculators, and so on. So *MESS* (Multi Emulator Super System) came into inception with Brad Oliver at its helm, and was first released on June 3rd, 1998. But it's only when the technical hurdle of source code synchronisation with *MAME* was resolved in 1999, with Ben Bruscella as the main coordinator, that *MESS* became a huge side project contributing new features and bug fixes back into *MAME*.

As it made no sense to let those 2 projects go on separately, David Haywood rightly proposed their merging, notably with his *Ultimate MAME* / *UME* (Universal Machine Emulator) initiative starting November 15th, 2011, with v0.144. This same year, Miodrag Milanovic took over the coordinator role of the *MESS* project from Nathan Woods and, the following year, also succeeded Angelo Salese for the same role on the *MAME* side. It led the path to the actual merging of the source code repositories of both projects on August 21st, 2012, despite still being distributed separately. Eventually on May 27th, 2015, both entities had completely merged with v0.162, simply distributed as *MAME*, the all-encompassing machine emulator.

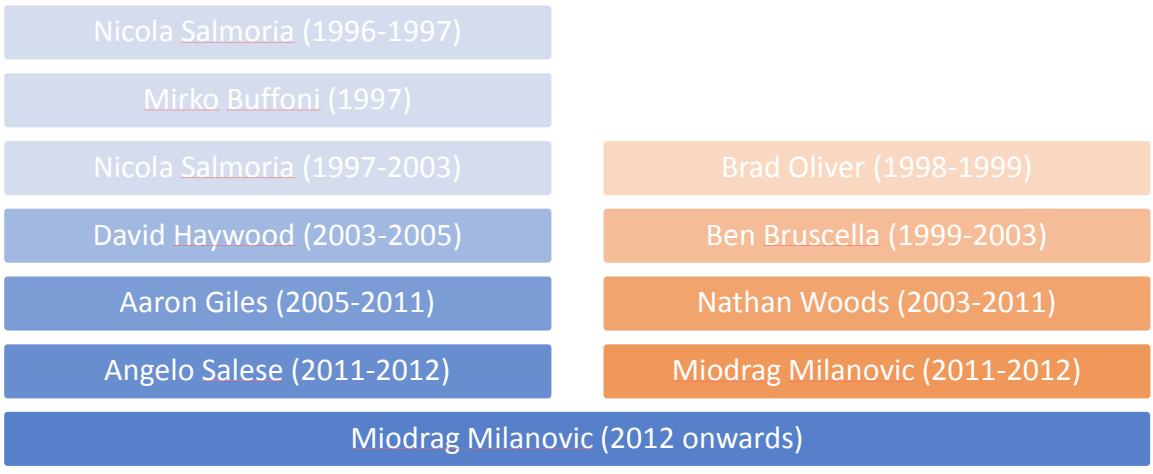

Figure 1 Team coordinators, MAME *and* MESS *sides respectively*

FROM A MS-DOS-ONLY TO A MULTI-PLATFORM EMULATOR

MAME was originally a MS-DOS application. On May 24th, 2001, Windows became the primary development target with v0.37b15. In parallel of the primary target, versions for other OSes were also made available. It was first ported to MacOS as *MacMAME* by John Butler, Aaron Giles and Brad Oliver on February 18th, 1997, and then ported to the various Unix / Linux distributions as *XMAME* by Allard Van Der Bas, Dick the Ridder, Juan Antonio Martinez and Hans de Goede on April 20th, 1997. But as *MAME* gained in complexity, it became increasingly difficult to synchronise those other OS targets to the Windows version. This issue got solved by Olivier Galibert and R. Belmont (not his real name) by relying on the SDL graphic library. The first public release of *SDLMAME* was made available on June 3rd, 2006 with v0.106u1, and this solution has been integrated into *MAME* since May 15th, 2010 with v0.138, officially making *MAME* a multi-platform application.

A GRAPHICAL APPLICATION?

MAME was originally a command line application until July 19th, 2007, when Aaron Giles integrated a minimal game selection user interface to v0.117u1, completing the already existing internal UI. But few efforts is dedicated to its maintenance as emulation related options can change dramatically from a version to another, and as the focus clearly remains on improving the overall emulation of machines. So, *MAME* remains best used in association with a separate dedicated front-end to easily manage everything it has to offer.

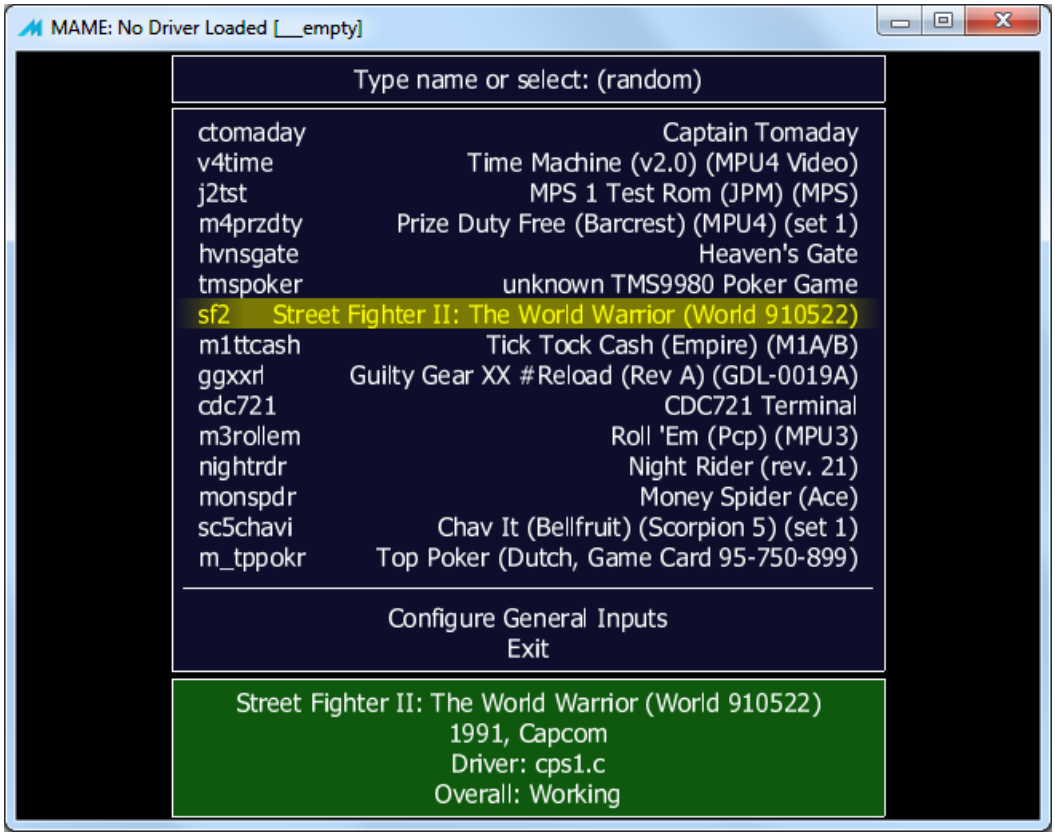

Figure 2 MAME *internal user interface*

About Negatron

Negatron is yet another front-end for *MAME*, the well-known multi-system emulator.

The production of *Negatron* began on July 7th, 2015, because of the inability of any existing front-ends at the time to entirely unlock all the features *MAME* has to offer, especially since it has completely merged with its sister project – *MESS* – in v0.162, which features a dynamic system of contextual configuration of the emulated machines: the **slot expansion** system.

FRONT-END CATEGORY

Front-ends can be classified into 2 broad categories:

- Desktop front-ends, geared to easily handle the huge machine database of *MAME*,
- Multimedia-styled front-ends, trying to propose a practical navigation system for computers (or TV sets) interfaced through remote controllers or arcade-styled control panels.

Negatron pertains to the first category as the huge number of emulated machines by *MAME* cannot be efficiently browsed through without a desktop-styled user interface and a keyboard-mouse set. Still, multimedia-styled front-ends are often favoured for their eye-candy and fully animated user interfaces. Nevertheless, *Negatron* doesn't entirely ignore this eye-candy aspect and compensate its less impressive interface with an intuitive and largely more ergonomic one.

Moreover, only desktop-typed applications can easily manage the concept of **slot expansion** (further explanations on the next section). Its intricate complexity when having to handle machines with a lot of parameters isn't really compatible with the seamless interacting motto sought after by multimedia-styled front-ends.

MAME MANAGEMENT SYSTEM

When surveying the available front-ends, most of them can only manage the emulated arcade machines, which obey the typical "one emulator to multiple game images" association rule. But since v0.162, *MAME* officially supports computer and console-like machines.

While arcade machines were mostly fixed packaged hardware sets, computers and consoles came in different hardware flavours and could actually be only used associated with a variety of software extensions, them being available through cassettes, cartridges, optical disks or even flash drives. To perfectly emulate all those features, *MESS* developed the concepts of software list (introduced by Wilbert Pol on May 22th, 2010, with v0.138) and slot expansion (introduced by Fabio Priuli on February 11th, 2013, with v0.148u1). Their implications can be summed up into 4 points:

1. For machines accepting cassettes/cartridges/optical discs, *MAME* adds a third indirection level to the above-mentioned association rule through **software lists**, i.e. instead of having an emulator managing a game library, *MAME* is an emulator managing a machine library, a machine being possibly bound to a software library.
2. A particular emulated machine can actually be bound to several software lists as its real counterpart could have several input **devices**, e.g. a cartridge slot and a CD-ROM drive.
3. Some machines could have optional or replaceable hardware parts, connected to the machine through specific **slots** and possibly modifying its behaviour.

4. Lastly, any changes to a machine parameter can trigger the appearance of **additional parameters**, them being new devices or new slots.

As of November 2015, only a handful of desktop-styled front-ends could handle the software lists and the slot expansion system, but none could totally support the above mentioned 4 points:

- Software lists were supported by *MEWUI*, *MESSUI*, *Emu Loader* and *QMC2*,
- Devices were supported by *MESSUI* and *QMC2*,
- Slots were only supported by *QMC2*,
- *QMC2* could partially manage the dynamic appearance of additional parameters.

So *Negatron* was engineered to entirely cover all those points, and assist users in managing them as intuitively as possible. It certainly does not cover every functionality *MAME* offers, it simply focuses on the most useful ones.

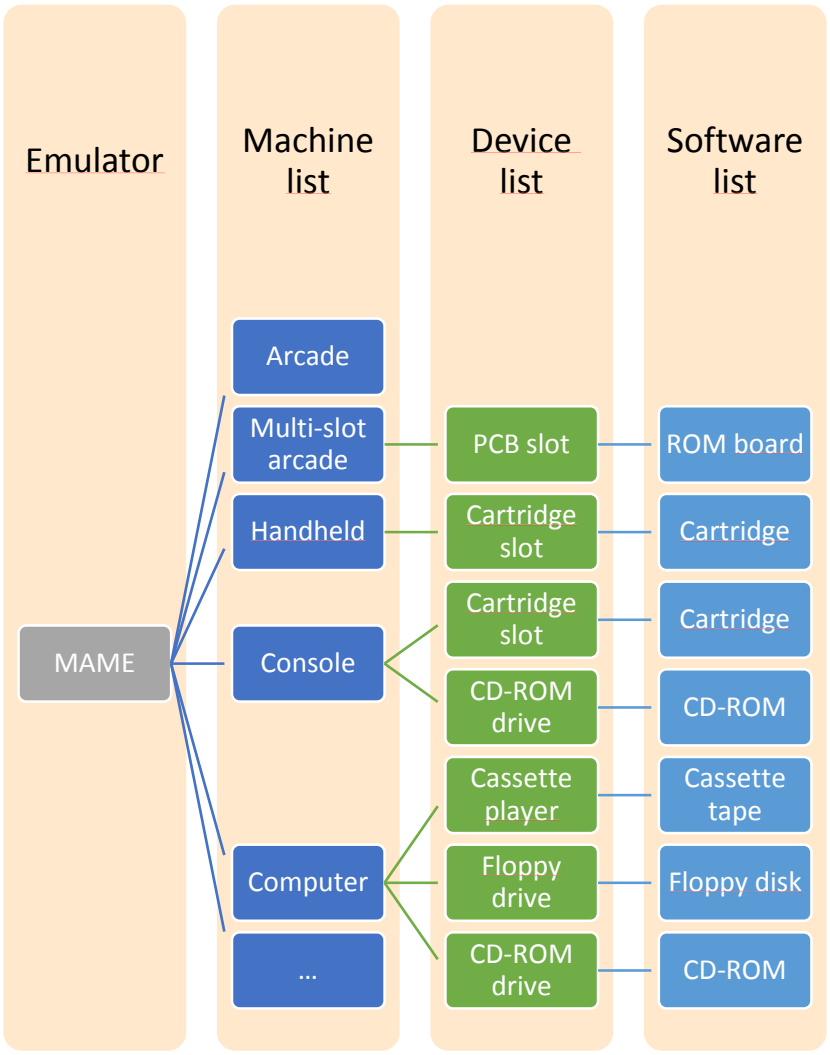

Figure 3 MAME *association rule: one emulator to several machines to multiple pieces of software*

VIDEO GAME PRESERVATION

Video games, also known as interactive media, are more and more recognised as a form of art worthy of physical preservation. Hence, museums are being built and opened all over the world, e.g. the *Computerspielemuseum* at Berlin in Germany, the *Nexon Computer Museum* on Jeju Island in South Korea, or the *Videogame History Museum* at Frisco in the state of Texas, USA.

Some other curated initiatives work at preserving those media mainly in digital form, e.g. the *Preserving Virtual Worlds* project led by the University of Illinois, the European *KEEP* (Keeping Emulation Environments Portable), or the *Game Archive Project* of the Ritsumeikan University. Nevertheless, because of the legal complications, video game preservation is still mainly going through personal projects maintained by individuals on private funds.

Figure 4 The logos of the aforementioned museums

One of those early personal projects is *MAME*, which main goal is to preserve video game history by documenting the original hardware using application code that can emulate their original behaviour as accurately as possible, i.e. the documentation is the application code itself. As hardware hardly works without software, *MAME* also maintains a database of the accompanying software, describing how they should be in pure digital form.

Yet, video games weren't just sold as raw pieces of hardware and software, they were also sold in neat packaging along marketing campaigns. Despite the gaming industry is moving towards pure digital form, some other private projects strive to archive complete digital sets representing in-game screenshots, raw hardware, box arts, covers and manuals that went along video games sold physically. One of those projects is called *MAME EXTRAs*, a massive attempt to build a complete information set around *MAME* that mainly draws from the resources made available by AntoPISA from *Progetto-Snaps*. Another one is *Emumovies*. Originally focused on capturing video sequences representative of the related games, it has become a paid service proposing a complete set of resources as well.

Figure 5 The logos of those allaround projects

Some others are more focused on a single kind of materials. For instance, the *Cover Collections* project hosted within the *GBAtemp* forums gathers a whole list of covers for a selected set of consoles. Some of its covers were drawn from the *Cover Project*, which acquires covers and inserts of jewel and keep cases. Besides, the project hosted at *NintAndBox* amazingly collects game boxes related to *Nintendo* consoles in the form of folding carton templates, i.e. as printable paperboard blanks that can be folded and sealed.

In order to ease the execution of this kind of projects around *MAME*, *Negatron* is thus also engineered as a video game information management system that can automatically organise those resources in a pre-defined way through drag and drop operations, and display them rightly associated with the related machines or pieces of software, this way helping to create a complete **information set** about video game history.

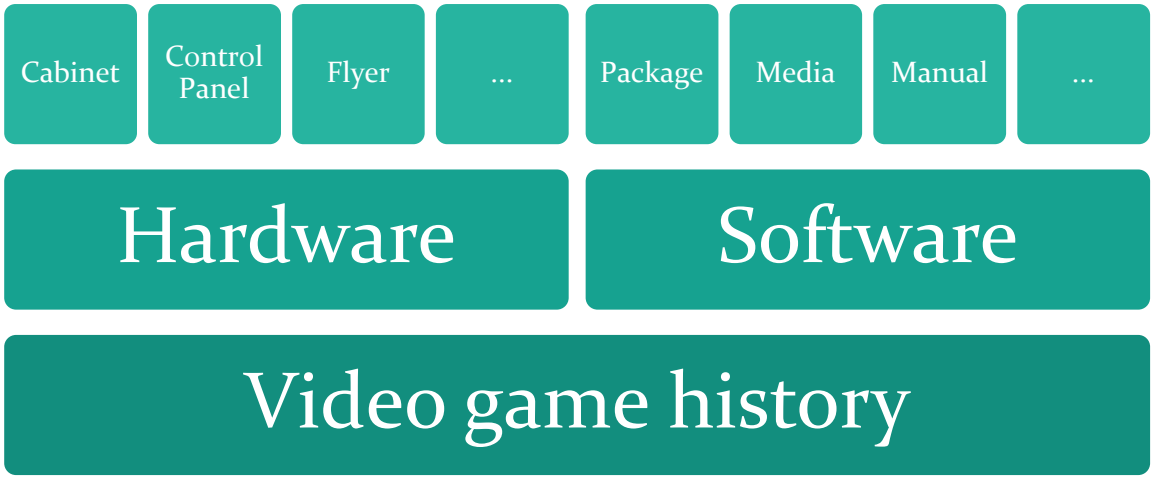

Figure 6 Video game history, made through games and a whole wealth of rich cultural material

Emulation management

Negatron is a Java application. So as most Java applications, it can work on any desktop computers under Windows, MacOS X or Linux. Negatron hasn't been designed to run under small screen portable devices, and any attempts to do so is discouraged as it may lead to a poor user experience. Indeed, as of November 2015, the official flavour of Java on Android is still based on Oracle's Java 7 and iOS doesn't officially support Java.

PREREQUISITES

Negatron requires a Java runtime environment (JRE) to work: ensure that your computer has at least Java version 8 update 40, or newer. If you have any doubt, you can simply download the latest Java version at http://www.java.com and install it on your computer.

Negatron also requires MAME v0.162 or newer. You can get the latest version for your specific OS at http://www.mamedev.org.

COMMON USE CASE

Download Negatron and launch it by double-clicking on Negatron.jar.

It will first prompt for the locations of the MAME executable and the MAME EXTRAs root folder:

- The MAME executable path is mandatory,
- The MAME EXTRAs root folder path is optional.

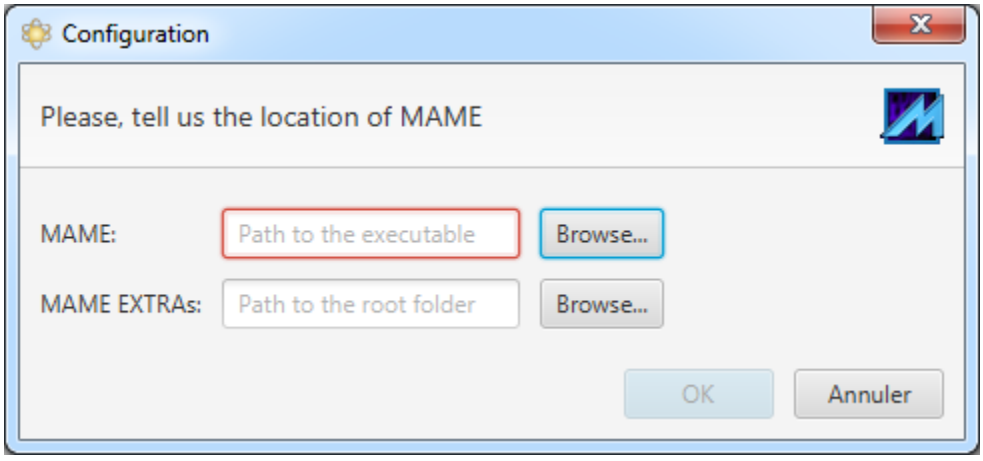

Figure 7 The preliminary configuration dialog box

MAME EXTRAs is a standard set of folders and files that greatly enhances navigation in Negatron by providing actual pictures of arcade cabinets, panel controls and in-game screenshots, among other kinds of information on machines and pieces of software. It is regularly updated along any new versions of MAME.

After this preliminary prompt, Negatron will load the MAME machine database, cache it and eventually display the main navigation window.

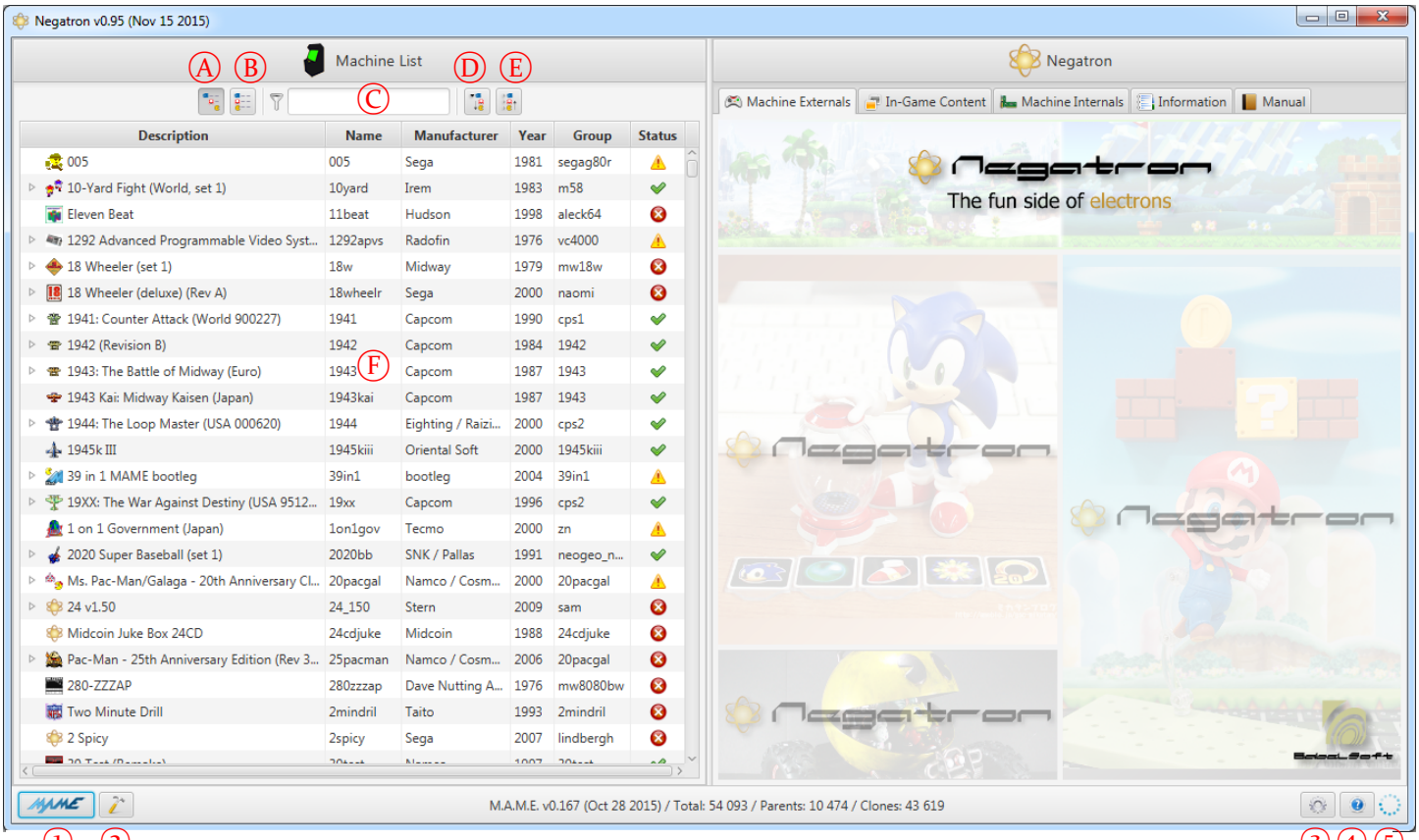

Figure 8 The main navigation window

The main navigation window is composed of the machine list on the left and the machine information pane on the right. The machine list is organised as a tree representing the typical parent-clone hierarchy used in MAME. If MAME EXTRAs paths are properly configured and your computer does hold the related resource files, the machine information pane displays the resources associated with the selected machine in the machine list, otherwise it simply goes with information related to Negatron by default. For further information on the information pane, please refer to the *Information Management* chapter.

On the bottom of the main navigation window lies the status bar, which contains a series of action buttons:

1. The MAME launch button launches the emulator with the currently selected machine. The same action can be done by double-clicking on a machine name in the machine list with the mouse or by typing on the Enter/Return key of the keyboard after machine selection.
2. The advanced parametrisation button opens a dialog box translating the current machine selection into a MAME native parametrised command line and allows to edit it further before launch.
3. The global configuration button opens a form where most of the paths related to MAME and MAME EXTRAs can be changed on the fly.
4. The about button directly shows some basic information about Negatron.
5. The cache loading animation is displayed while Negatron is checking and possibly updating its cache.

On the top of the machine list lies a series of controls focusing on machine browsing:

A. The tree view button allows to organise the content of the machine list following the default parent-clone hierarchy used in MAME.
B. The table view button allows to display its content following a flat list of machines.
C. The filtering text box allows to filter out machine entries whose description doesn't match the content of the text box.
D. The expand all button forces the expansion of every machine node in the tree view mode.
E. The collapse all button forces the collapsing of every node in the tree view mode.
F. While the focus is on the machine list, you can also quickly type the name of a machine on your keyboard to directly navigate towards its entry.

So browse through the machine list, select a machine, launch it with MAME, and enjoy several decades of gaming.

MACHINES WITH SPECIFIC PARAMETERS

Some machines can take optional parameters, which can dramatically change their behaviour. Some others require specific parameters in order to launch in a working state, e.g. a console which needs a game cartridge to be put on top of it in the real world will need a ROM image of this cartridge to be passed as a parameter to MAME.

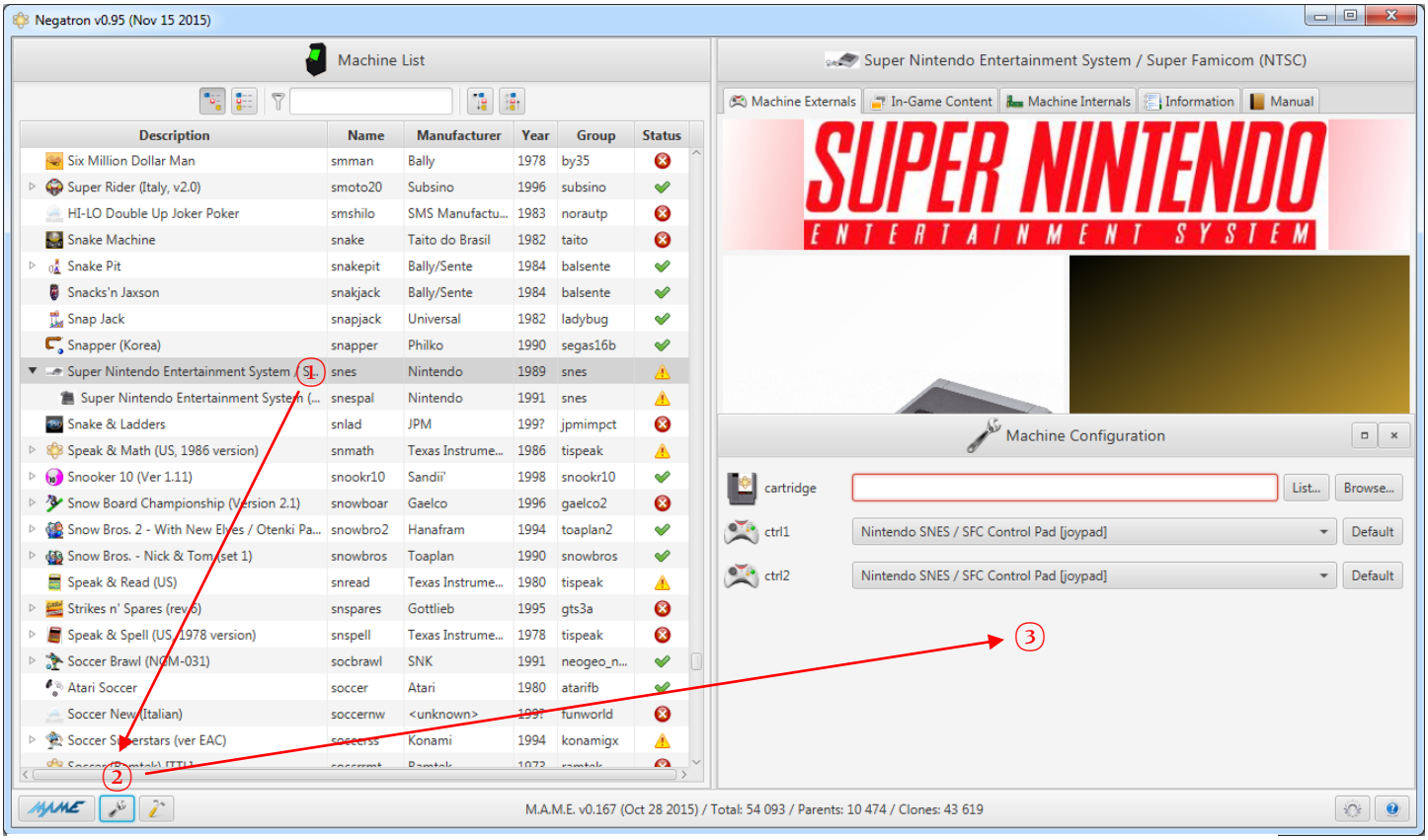

Figure 9 The procedure to get to the machine configuration form

Through the **slot expansion** system featured in MAME, Negatron simplifies the task to know what is optional or mandatory for a particular machine by showing a form containing all the parameters available to a machine: the machine configuration form.

To get to the machine configuration form, follow the below procedure:

1. Select a machine that take specific parameters, e.g. the Sega Game Gear.
2. If the selected machine does take parameters, the machine configuration button should appear between the MAME launch button and the advanced parametrisation button.
3. Click on it to open the machine configuration form, which then covers the lower part of the machine information pane.

To get access to the machine configuration form, you can alternatively right-click on the machine entry in the list with your mouse or type on the *Space* bar on your keyboard after machine selection.

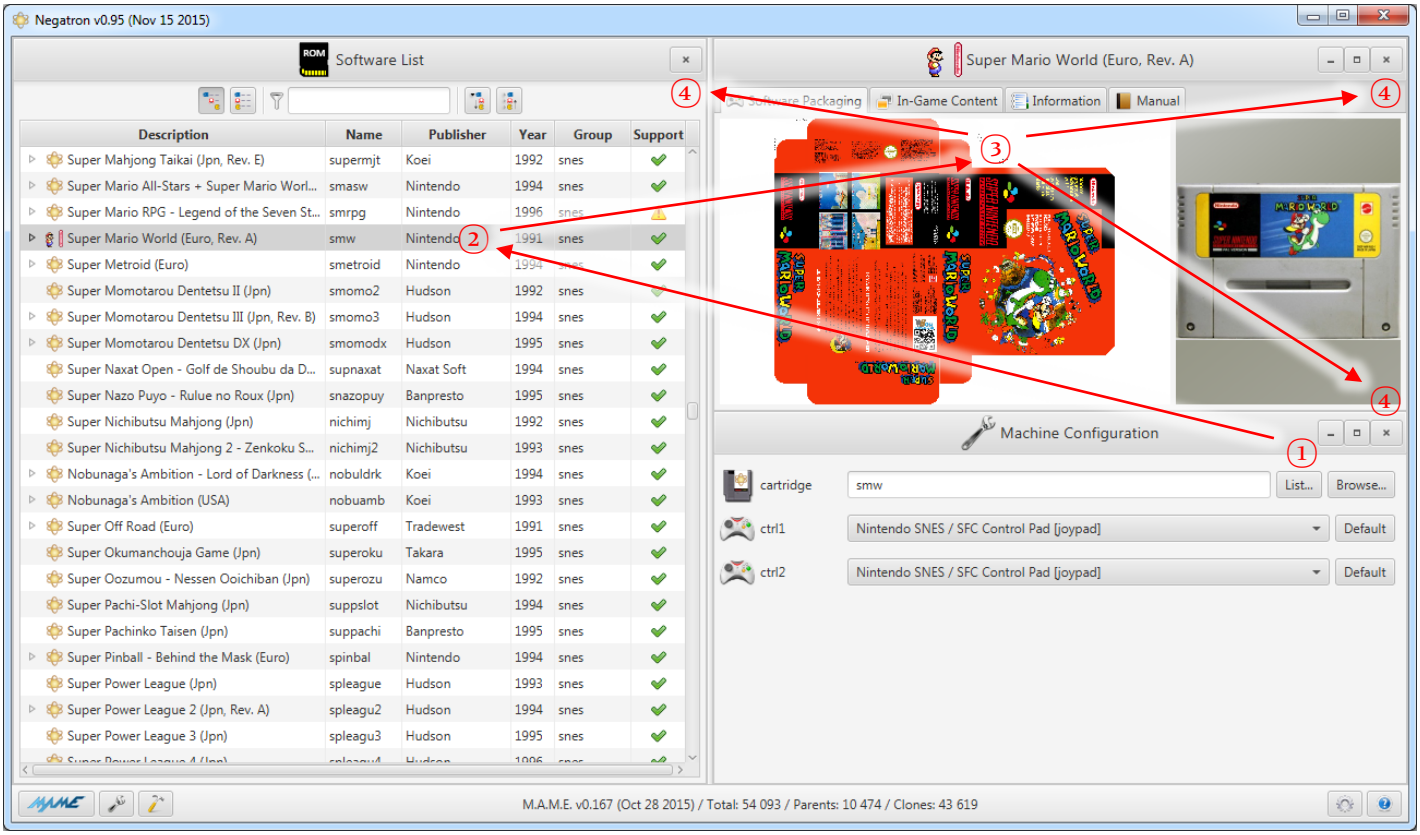

Figure 10 The software dedicated panes

Some parameters propose a selection of items to choose from, others are meant to take a file path, which is hinted at with the occurrence of a browse button, others again can additionally take an item coming from a **software list**, which is hinted at with the occurrence of a list button. In this case, additional panes dedicated to software show one by one, eventually covering all the machine dedicated panes:

1. By clicking on the list button, the software list appears by covering the machine list.

2. When selecting a software entry from the software list, the software information pane appears by covering what was still visible of the machine information pane.
3. The software information pane works the same way as the machine information pane.
4. To return to the machine dedicated panes, simply click on one of the close buttons. Alternatively, type on the *Escape* key on your keyboard for the same result.

In very complex machine configuration, some parameter editions can trigger the modification, addition or deletion of some other parameters. In the two former cases, a badge appears next to the icon of the related parameters so that you can easily spot all those changes.

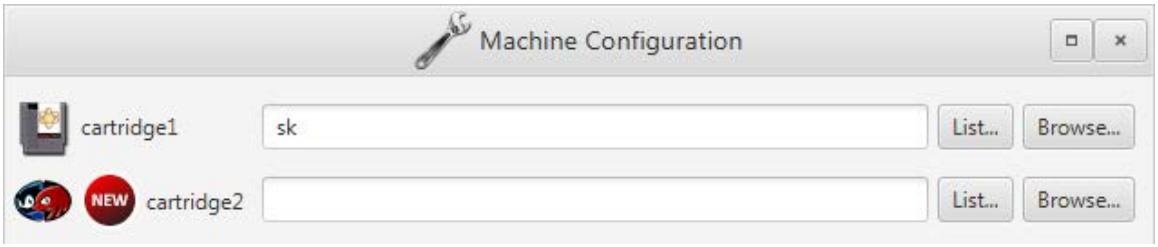

Figure 11 After selecting Sonic&Knuckles (Mega Drive) appears the corresponding lock-on cartridge slot

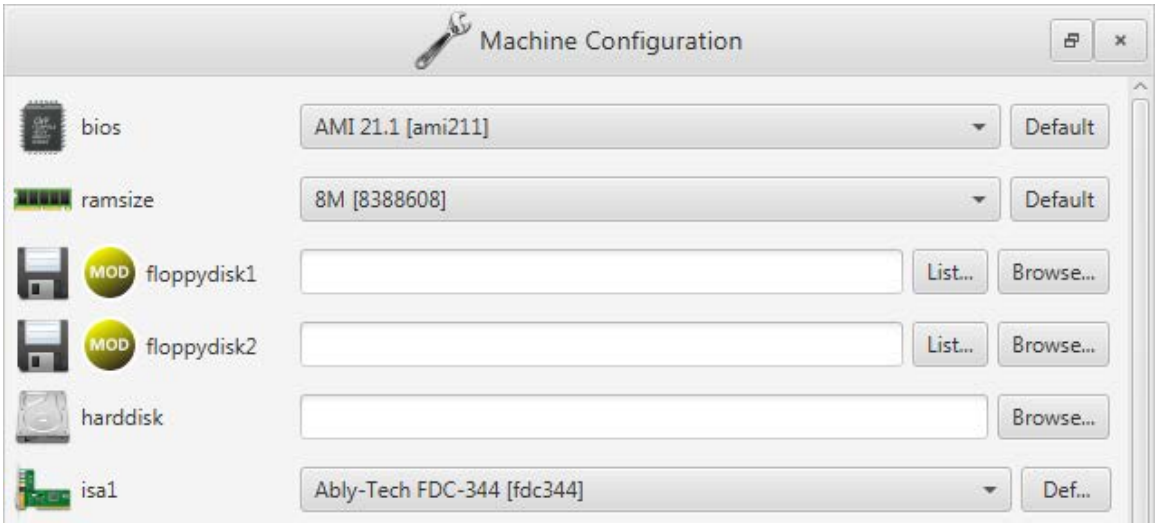

Figure 12 Editing any parameters of a PC/AT computer usually triggers a whole chain of modifications

Information management

As explained in the *Emulation Management* chapter, the preliminary configuration dialog box allows to quickly configure Negatron with the strict minimum to use it out of the box. If you fill in the optional MAME EXTRAs root folder field, Negatron will automatically determine a default configuration suitable for a typical **information set** following the MAME EXTRAs guidelines, extended with a few added entries to more properly take into account the information subset pertaining to software.

GLOBAL CONFIGURATION

The complete information set is structured as shown in the below figure, with some entries specific to the machine subset, others to the software subset, others again common to both subsets (items marked with an asterisk * are not part of MAME EXTRAs as of November 2015):

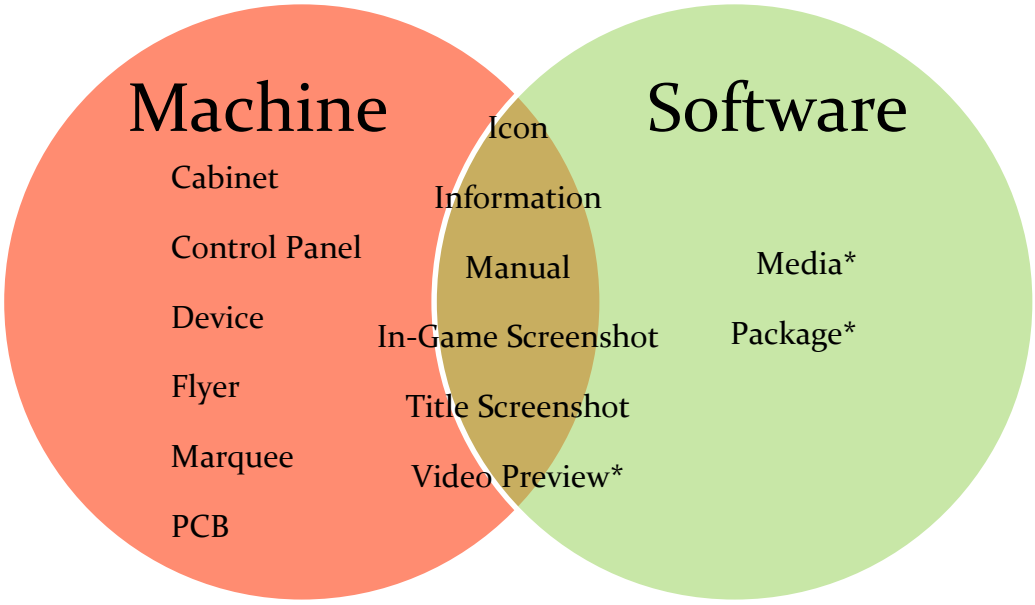

Figure 13 The structure of an information set following the MAME EXTRAs guidelines

If the default configuration doesn't suit your own information set, every entry path can be changed through the global configuration form:

1. Icon holds the icon resources representing each item (machine or software) as a 32x32 ICO file. It does not represent actual objects linked to an item but it is simply used as a quick visual cue within Negatron's interface.
2. Information takes the paths to DAT files giving a descriptive encyclopaedic view on all machines and software.
3. Manual represents the actual manuals or notice booklets that were provided within a packaging. They should be formatted as PDF files.
4. In-game screenshot caters for raster graphics formatted as JPG or PNG files representative of the gameplay of an item (machine or software). They may also be called snapshots.
5. Title screenshot pertains to raster graphics representative of the title sequence of an item (machine or software).

6. Video preview* stores video sequences formatted as MP4 or FLV files representative of an item (machine or software). They may also be called video snaps. A MP4 file must have its video stream encoded with the H.264/AVC codec and its audio stream with AAC while a FLV file must have its video stream as VP6 and its audio stream as MP3.

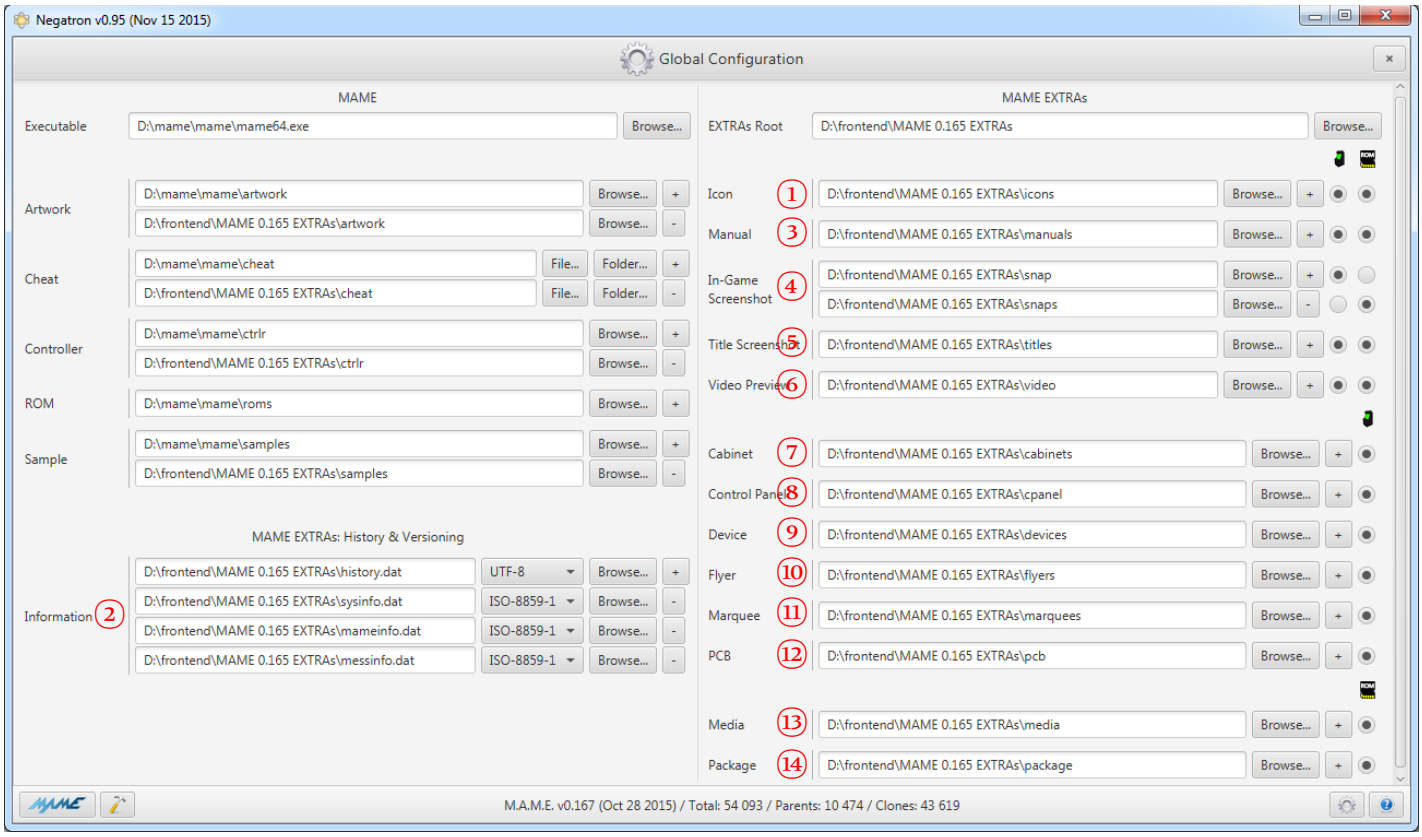

Figure 14 The global configuration form, an example of a default configuration

7. Cabinet refers to raster graphics representing the outside shell of a machine in its entirety.
8. Control panel relates to raster graphics focusing on the input devices coming along a machine, them being joypads, joysticks or arcade control panels.
9. Device supplies raster graphics for every component embedded within the emulated machines. They can illustrate their CPUs, memory chips, connectors, etc.
10. Flyer deals with raster graphics representing typical advertisement pamphlets associated with a machine.
11. Marquee is about raster graphics representing the typical signs placed at the top of arcade cabinets. For non-arcade machines, it usually represents their brands as logos.
12. PCB stands for printed circuit boards and holds the raster graphics representative of the main electronic board of a machine.

13. Media* illustrates pieces of software in their original physical forms as raster graphics, them being floppy disks, cartridges, optical discs, etc.
14. Package* provides raster graphics corresponding to the art representations of the external packages holding software media, manuals and other informative leaflets together, e.g.

covers for keep cases or folding carton blanks for cardboard boxes. Those graphics can then be used as textures for packages modelled in 3D.

Some entries can take several paths. This is indicated with the occurrence of a "+" button on the first field associated with the entry. Some others are also associated with radio boxes placed on the rightmost side of the fields. This is to indicate what paths should be the preferred folders to contain new resources through drag and drop operations. If an entry makes sense for both subsets, related fields will occur with two radio boxes, one for the machine subset and another for the software subset, otherwise it will occur with a single radio box.

Figure 15 Specific user interface controls for specific configuration settings

If you edit the EXTRAs root field, all the MAME EXTRAs related fields will be synchronized accordingly on the fly, saving you from the hassle to update every single fields manually.

The other entries directly apply to MAME:

- Artwork contains sets of artwork files that are applied while in-game.
- Cheat emulates a cheating device equivalent to the infamous Game Genie by providing cheat codes through XML files that can modify game behaviour. Those cheat codes can then be selected through the internal user interface of MAME.
- Controller gives the button mapping between your own input device and the emulated ones.
- ROM includes every software images needed by machines emulated in MAME, CHDs included.
- Sample covers the placeholders of sound effects that can't be easily emulated as being originally emitted by discrete circuitries.

Please refer to the MAME documentation for further information.

INFORMATION PANES

If Negatron is rightly configured, the complete **information set** available on your computer can be displayed by the machine and software information panes. But instead of bringing in an already built information set, you can on the contrary build a new one through drag and drop operations on both information panes, or even complete an already existing one in the same way.

While dragging a resource on a pane, Negatron will give you visual clues indicating whether the resource file is in an acceptable format and can be dropped. Negatron checks for resource compatibility only based on the file extension, and doesn't check for malformed files or files with wrong extensions (indeed, JPG files are often renamed as PNG files). After dropped, the resource is copied into the configured preferred folder of the corresponding entry.

The figures of the two next subsections indicates which user interface controls is related to which information set entry.

Machine information pane

Figure 19 Machine externals tab

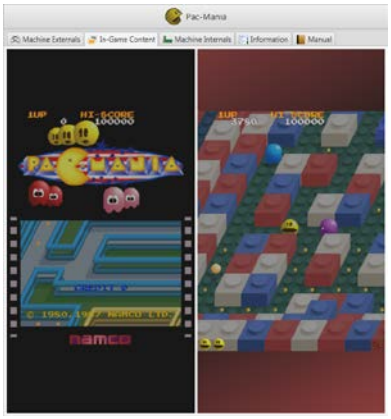

Figure 18 In-game content tab

Figure 17 Machine internals tab

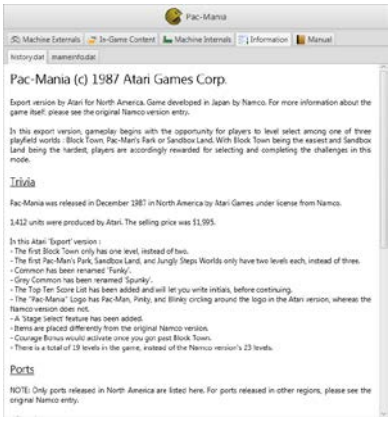

Figure 16 Information tab

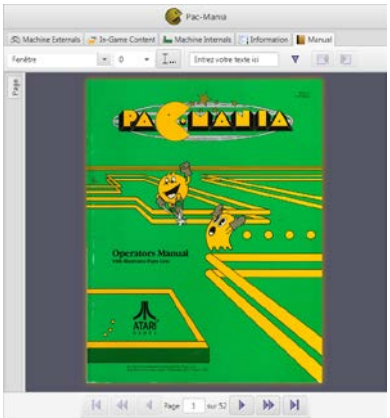

Figure 20 Manual tab

Software information pane

Figure 21 Software packaging tab, normal state

Figure 22 Software packaging tab, maximised

Figure 24 In-game content tab

Figure 23 Information tab

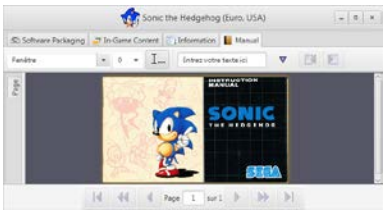

Figure 25 Manual tab

Acknowledgements

The term *negatron* has been first proposed to design regular electrons by Carl D. Anderson in 1932 after discovering the positron, following the predictions of Paul Dirac.

The logo of Negatron is based on the old planetary model of atoms as introduced by Ernest Rutherford in 1911 and Niels Bohr in 1913. It has been since superseded by the electron cloud model as first devised by Werner Heisenberg and Erwin Schrödinger in 1925.

MAME is a registered trademark of Nicola Salmoria.

Most icons used in Negatron are part of the Oxygen icon library, a LGPL project developed by KDE. The exceptions are:

- MAME logo, designed by chemical and cleaned up by Exodus3D,
- The arcade machine icon, by Bruce The Deus at the Italian Wikipedia,
- The default cartridge icon is based on the NES cartridge template by VoidSentinel,
- The default media picture is based on the Neo Geo cartridge template by blinkybill,
- JavaFX and NetBeans are trademarks or copyrighted properties of Oracle America, Inc.,
- IDR Solutions Ltd is a UK company specialised in Java, PDF and HTML5 solutions,
- The remaining icons not part of *Oxygen* have been designed by BabelSoft.

Most of Negatron code base has been developed by BabelSoft. The exceptions are:

- OpenViewerFX, the integrated LGPL PDF viewer by IDR Solutions,
- The HiddenSidesPane component of ControlsFX, under the BSD 3-Clause license, which has been heavily modified to suit Negatron's need,
- The JavaFX TreeView filtering algorithm by Christoph Keimel, now part of the e(fx)clipse open source framework, under the Eclipse Public License 1.0,
- Part of DirectoryWatchService by Hindol Adhya, under the MIT license.

Negatron uses a graphical design theming around the following video game characters:

- Sonic the Hedgehog, a trademark of Sega Games Co., Ltd.,
- Mario and Super Mario Bros., trademarks of Nintendo Co., Ltd.,
- Pac-Man, a registered trademark of Bandai Namco Entertainment Inc..

The use of some graphical elements directly referring to those characters have not been endorsed by any of the trademark holders. This has not been done in pursuit of Negatron's advertisement or commercial benefice, but any claims can be emailed to BabelSoft at cyrilapan@babelsoft.net and will be promptly complied with. Those elements are:

- Nendoroid Sonic the Hedgehog, produced by Good Smile Company,
- Nendoroid Mario, produced by Good Smile Company,
- Pac-Man Monster Truck, a custom product by KodykoalaToys,

- Sonic the Hedgehog BADASS, a fan-art by Sylvain Sarrailh,
- Super Smash Bros: Remixed – Mario, a digital art by Andrew Domachowski,
- Pac-Man Movie, a short movie produced by Steelehouse Productions,
- Sonic CD, a Mega CD game developed by Sega,
- Super Mario World, a Super NES game developed by Nintendo,
- Pac-Mania, an arcade game developed by Namco.