**pfeMAME Changelog**

**v3.3**

* Fixed a bug that was causing clicks on some child window button from firing the ribbon bar button that opened them, thus opening a duplicate and breaking the child window. Turns out this was a major deal-breaker. Sorry I missed it.

**v3.2**

* Fixed a bug introduced in the first version 3 release whereby in the fullscreen video player, the video would not automatically proceed to the next one once the currently playing video had ended. Needed to add the is\_playing flag to the while loop.
* Fixed a bug that would cause the ROM image & video to be positioned incorrectly after resizing the main window and then selecting a new ROM.
* Updated the software video decode fallback to improve performance on slower systems.
* Fixed the white-flash issue when playing back video using the GL hardware accelerated decode.
* Changed the audio playback to use snddevice which natively plays the audio. This vastly improves synchronization issues between video and audio.
* Fixed a bug that would allow infinite multiple ‘About’ windows to be opened.
* Fixed a bug that would keep a ribbon bar button highlighted after a child window opens and is then closed.
* Fixed an issue where the ribbon bar button highlight area was not all detecting click events – just the center portion with the actual button image.
* Removed the Check for Wayland on Startup option from pfeMAME preferences as the video player is no longer impacted by this.
* Fixed the column headers no longer allowing a click to filter by column. This was broken around the time of the V3xx release.
* Added caching to the extracted audio for video file playback to speed things up a little bit on slower systems
* Fixed a bug that was causing the view ‘controls’, ‘logfile’, and ‘changelog’ to generate issues after closing the popup window.
* Fixed a bug that would allow a failure to run an alternate emulator engine to fail silently with no feedback if the path to the emulator cannot be found
* Removed some unnecessary old futures / annotations imports
* All support files not part of the application itself have now been moved into a ‘tools’ subfolder. This is where you will find, for example, the setup file for building compiled distributions.

**v3.1**

* Added support for merged romsets
* Fixed a bug from the previous release that prevented the ROMs list filter from being applied on startup.
* Fixed my log file viewer. Had to limit to 2000 lines due to the control size limitations.
* Updated Filepaths so that it will always try to open the root MAME folders when you double-click on a item to manually set / change it.
* Fixed a bug in the main messages window where multi-line error outputs were not being shown. At the moment the fix is a bit of a hack as the issue is rather complex and I will have to come back to it later to work it out.
* Fixed the ‘Click to stop video’ button that I broke in the previous release.
* Fixed no audio with macOS video playback.

**v3.0**

* Fixed the MacOS localization issue that would prevent alternative languages from working on packaged builds.
* Added an auto updater. If a new software version is found online, you will be given the option to download and update automatically. This preserves your INI files, backups, database etc.
* Modified Filepaths to allow the systems / software folder to be software, systems, or ROMs. I believe MAME prefers this to be ‘software’. If its not defined then the fallback is roms.
* Improvement to Filepaths code – the lookup list used to define allowable folders for different storage types has been simplified and the lookup code improved to manage different case types in the folders.
* The fullscreen video player has been updated to use the new video class. This will allow me to better handle future video backend changes when I get around to sorting out the very annoying Linux / Wayland support issue that WXPython is having.
* The separate file and tool bars have been combined into one ribbon style menu bar. Trying to make things a bit more modern.
* Fixed a bug where if a color lookup wasn’t found, it would return 0,0,0,0 (Black) instead of the default I have set.
* Fixed some notebook tab colors.
* Fixed a bug that would cause an error if there is no catver.ini file and you try to build up a database anyway on first time use. This is due to there being no categories and thus no list present.
* When the database is refreshed, if no catver.ini file is present then the categories list will at least now include ‘Not Classified’ and all ROMs will default to that classification. This way the application can still be used without the catver.ini file. You will still get a warning on each startup that the file is missing as you really should have it.
* When the window is now drawn on startup, the screen fades into view in a nice way so that you don’t see individual controls being created.
* Redesigned the entire menu bar and implemented a ribbon ‘style’ menu interface.
* Started replacing multiline text controls and scrolled panels with a custom control that gives me full control of the colors. This allows me to make the scroll bar look much better in dark mode. The scrollbars are also now hidden by default and a mouse-over will make the appear.
* Added a ‘Toggle All’ button to the top of the ROM categories list and used that rather than double-clicking the list to toggle all SET and all CLEAR.
* Fixed a bug introduced in the last release where actively changing ROM category selections didn’t make a change until the application was restarted. The cache wasn’t being flushed on change.
* Removed the -forcevideo command line option as video is no longer disabled due to an OS type.
* Removed the command line -novlc as there is no longer VLC playback support.
* Made some adjustments to the default ‘dark mode’ window background color
* Created a custom drop down list and replaced all built-ins and previous attempts at a sudo custom drop down.
* Created a Class helper for storing the systems configuration data after change. This will make it easier to manage in the future.
* Fixed a bug where if you have resized the main window, the progress bar may be drawn in the wrong location.
* Fixed the ROMs list column sizes to work better with the new layout and removed the column adjusting when resizing the window as it never worked right and just isn’t needed.
* Fixed a bug where if you defined multiple folders for a path (e.g. ROMS, ROMS2…), the Filepaths window would only show the 2nd one you had set when you leave and go back to it.
* Have removed the option for a SOFTWARE folder as this is pointless and not needed. Systems programs software MUST be in the ROMs folder for MAME to know where they are.
* Fixed an issue that caused systems programs to get out of order in the list when refreshing the database if they had system short names.
* Improvements to the database build code. Bug fixes within multi-processor & full-systems versions. General code improvements.
* Large refactor of the panelSlots code that controls the MAME systems slots configurations. Improved the code performance, layout and readability and fixed some bugs that caused issues. Also got rid of the annoying popup every time you create a new slot entry – totally unnecessary.

**v2.34**

* Fixed a bug that would re-enable video if you had previously disabled it after a restart. The command prompt parsing was overriding it.
* Fixed a very uncommon bug where on closing I could try popping the logging handlers with zero length.
* Fixed a problem with the text in the ‘views’ dropdown not being displayed nicely in Linux. The optimum height is now calculated based on platform and font size.
* Optimised code in self.populate\_list to make searching faster
* Tidied up some poorly written code across the codebase (Quite a lot actually. Well, massive amounts to be sure. I’m sure I have probably broken something in this process as I don’t have the resources to test everything across each platform).
* Changed to using PIL for image rescaling rather than img.rescale. Not necessarily faster for my use case but it provides more flexibility later.
* Added some error detection to the code that checks if you have a valid videos to play for the fullscreen video player
* Improved the code that checks for newer versions online so that it can look for 7z and zip file extensions just in case I upload it differently
* Improved performance of unescape function used during database build
* Significant speed improvement in standard database method by implementing Thread Pool.
* Fixed a bug that would cause an application launch crash if more than 2 joysticks / gamepads are detected. The library can only handle 2. If more are found, a warning will be logged with the details and only the first 2 will be available.
* Added logging of joystick information
* Changed the vertical size of the Bios dropdown list in the Slots configuration panel to better fit the lists available
* Displaying the ROM-specific preferences pages takes a while. To make the visuals a bit better I have frozen the GUI update until it is ready to show. This prevents the visual corruption in the corner as its being drawn before its ready to be shown. I still need to work on why it takes so long to dynamically add this tab to the notebook.
* Moved some more of the common VLC code into the functions helper
* Updated scrolledwindow.py to be a dialog control and to be modal
* Significant refactoring of Tileview including a lot of code improvements
* Increased the size of the busywindow a bit to make it more obvious
* Significant speedup in checkall by using threadpool
* Changed the method for opening the user manual under linux. Now defaulting to webbrowser which under linux has in-built fallbacks to manage different methods.
* Changed the background flash behind the filepaths icon to be a prettier ramping color rather than a harsh flash.
* Added some better compatibility checks to the image copy from clipboard.
* Improved the command line interpreter a bit
* Reduced INI file writes but combining into one write process where possible. Also check for changes before writing. This was updated in pfemame.py and filepaths.py files. This greatly reduces drive writes.
* Fixed some errors writing to the build\_errors.txt file. I needed to explicitly set the encoding to utf-8.
* Improved the mouse binding in the main window to better rest the inactivity timer with mouse movement
* Modified the populate\_list function within pfeMAME.py to use a search cache for faster repeated searches. This dramatically improves the speed that results are provided for repeated searches. Doesn’t really happen that often in real-world scenarios but my provide improvements on poorly performing systems.
* Rewrite of the AboutBox. License also split out to a file.
* Updated the pfeMAME logo for a more modern look
* The romRating window has been replaced with starRating and changed to a transient popup that will appear wherever the mouse is. This works really well when you enable the doubleclick to set favorites & ratings option.
* Fixed a unicode decode error in the database build full systems support version.
* Fixed some configparser issues reading from the pfeMAME.ini file with different languages.
* Fixed more language translation issues in my (very poor) automatic language translater
* Fixed a bug where the selected ROM foreground and background colors were not set to default for a clean install.
* Improved the cross-platform appearance of the highlight around the language image on the language picker.
* Finally got rid of the issue where on Linux in Tileview, the first tile has a focus box. Have also masked the event created by left click to prevent the control from taking focus. At the same time I fixed the issue where the brightness would be decreased by 1 step every time you clicked.

**v2.33**

* Altered some startup sequencing to support some command line changes
* Altered the -v / --version command line to display more version information (pfeMAME, MAME, compiled platform, Python version, and wxPython version
* Added a command line option -c / --count that will give the ROM count based on the stored database file. If there is no current database then it will return 0.
* Changed some command line exit() commands to sys.exit() as cx\_freeze doesn’t support exit().
* Altered the command line options that generate an output (e.g. -v and -c) so that under windows, if the application has been frozen for distribution, a new command prompt window is opened in order to display the output. This has to be done as under Windows when frozen you cannot send an output to the command prompt window if that’s where you ran the application.
* Fixed a bug that was stopping the mameinfo.dat file from being read. Also cleaned up the code and fixed some incorrect types.
* Fixed a bug that would throw an error when trying to use the inbuilt logfile reader. I was not specifying utf-8 as the file type and recent python builds no longer make that assumption for you.
* The setup.py file has been altered to use cx\_freeze under macOS now as I was just having too many issues with py2app. It means that the structure doesn’t look macOS-ish and is instead the same structure and Linux & Windows but at least it works now.
* Fixed a bug that would probably cause a crash when using TileView and the full screen video player kicks in after timeout. I hadn’t updated the TileView code based on changes I had made to the logger.
* Removed the alternate gamepad code that used the ‘inputs’ library. It was terrible.
* Added a new feature that allows you to save an image stored in the clipboard as a ROM snapshot for the currently selected ROM. This can be accessed from the ‘Edit’ menu.
* Alternate Emulators window – Changed controls to use smaller ‘…’ button to select file / folder rather than the older / larger ones. Changed the layout to a vertical style for the controls & descriptions on the right hand side in order to layout better for smaller displays. Changed some control sizes for better fitting. Made the control descriptors BOLD to make them easier to follow.

**v2.32**

* Added a lot more logging of the MAME executable location process in FilePaths. I have done this in order to assist with major problems in the MacOS distributable version of pfeMAME no longer getting the right responses from subprocess when trying to run MAME from the system path rather than an absolute path.
* Modified the method used to find the MAME executable when auto-finding to try and be a bit more robust for MacOS.
* Simplified the icon code for the main menu
* Simplified the on\_control\_changed code in alternate emulator engines
* Fixed bug in panelslots.py that was generating bios names for all machine types for the selected system, rather that the specific system machine type (i.e. list was way too long and selections could be incorrect for the actual system)
* Simplified and made more pythonic a number of other sections of code in panelslots.py
* Fixed bugs in panelslots
  + Fixed bug that would cause a crash for systems that don’t have a bios option
  + Added a first bios option to be a blank field which is possible for some systems
  + Fixed bug that would prevent you from being able to default or empty slots for systems that you are configuring for the first time
  + Fixed bug that would stop you from clearing a single line in slot options
* After closing the alternate\_emulators window, the XML file is reloaded to update the alternate emulator data in memory. This way you shouldn’t have to restart pfeMAME but you will have to refresh the database to see the changes.
* The Tile View filter (All of Favorites) is now set based on the value from the Main View. You can still toggle it when in Tile View.
* Updated the non mame defaults XML file to include additional command line options for visual pinball that send it to the background and fully close when the table exits. Much better for non-keyboard implementations.
* Added a feature to allow all ROMs tagged as a favorite to be exported to a selected folder. Can be found in the ‘File’ menu.
* Added a new preference setting to set the main window Fullscreen on startup
* Added a new preference setting to recall the main view ROMs list columns widths on startup. Whatever they were when the application was last closed will be recalled.
* Updated the preferences window to expand vertically to use available space.
* Fixed issue with the joystick code causing high CPU usage. By default the polling is set to maximum speed (0 delay) which chews up to 20% CPU which is ridiculous. Have set to 50 which is just fine.
* Added a set-focus event so that when changing the filtered view for the main list, the list is selected for focus so you can start scrolling without having to first click into the window. At the same time I force the visible row back to ZERO to ensure that a row is aways selected.
* Had to change the get checkbox controlls code as for some reason I don’t currently understand, its causing errors in python 3.13. Also moved it to a function as its used in 3 different places.
* Fixed a bug in the code that recalled the last ROM selected. For systems this would work if the first system ROM was the top level system (e.g. ZX Spectrum) and then the next ROM after that was the program you wanted to run. This would not work if the first system was the system and also the ROM (e.g. pinball tables).
* Significant improvement to how the mame history file was loaded into memory and then searched on the fly. Previously done in a list which was then iterated through for searching using pure python. Have moved to keeping it in one massive string and using the inbuilt ‘find’ which is in C so a heck of a lot faster. We’re talking x10 speed improvement on a fast system. I would expect slower systems to be even better. The net result is that scrolling through ROMs will be just that little bit faster now. IMPORTANT: At the same time I have dropped the legacy support for the older history.dat file. Only the XML file is now supported.
* Updated the manual – please read it if this is your first time using this application.
* Fixed a bug in alternate emulator engines when using the set defaults feature that would cause an error if you didn’t select any engine types. Also fixed a bug in the same custom dialog that would return an wx.ID\_OK event even if you pressed ESC or closed the dialog without selecting anything in the drop-down.

**v2.31**

* Added some additional debug logging if developer mode is active
* Enabled VLC to be used for MacOS. Have tested on Intel silicon only. Seems to work ok ish.
* Fixed an issue with the selected row text color not taking user changes
* Added the ability to change the selected row background color. This was locked to the system default. Have also changed the default to something more pleasant (imho).
* Fixed a MacOS console warning relating to the vertical size of the joystick selection combobox in preferences. Just changed the vertical size to system default.
* Changed the tabs background area color to the control bar color for all tabbed controls. This allows for better contrast and ‘view ability’, especially in dark mode.
* Changed the way the MAME executable is looked for in Filapaths. I was previously incorrectly assuming that only Linux required it to be located in the system path (So that you can just run MAME from the command line and it works regardless of what the current folder is). As I have found out, some installs under MacOS install it in the system path rather than just in a folder (e.g. MacPorts). The new change looks for the executable firstly as a file in the folder you have set as the MAME folder. If it can’t find it there then it checks to see if it’s in the system path. If it still can’t find it, then you will just need to find it yourself.
* Made the RED background highlight around the filepaths icon flash to make it more obvious for first time setup that the filepaths are where you need to start.
* Added by build\_source\_distribution.py file to the source distributions
* The python modules ffmpeg-python and python-vlc are now compulsory if you are building a distribution executable. This ensures that the required include files are added to the build.
* Fixed duplicate message logging after auto-restart
* Updated the manual to include detail in the Full Screen Video Player section relating to displaying metadata needing ffprobe (of ffmpeg for Linux)
* Fixed an issue where the buttons at the bottom of the Filepaths window were greyed out in some situations (They were still useable, but the GUI needed to be refreshed to redraw them correctly)
* Fixed a bug that would cause Filepaths to crash out if you had selected ‘Clear All’ and then re-run it without setting the INI path to a valid path.
* In Filepaths, any invalid or missing paths are highlighted RED. This doesn’t mean it’s a critical issue, just that this particular path is not set.
* Improved MacOS ability to find the ffprobe application that you should copy to the pfeMAME folder.
* Fixed a bug that could cause a KeyError on some video files when trying to get matadata that isn’t present or the file is not formatted correctly.
* Fixed a crash that would occur when using the Fullscreen video player under MacOS. Not a Mac bug, but I wasn’t handling a send to the main screen messages window correctly. Doing it from a thread in a non-thread-safe manner was a bad idea. Have used wx.CallAfter now to correct this. Pretty big oversight.
* Updated the manual based on recent changes and added improved detail around the Full Screen Video Player ffprobe support.
* Fixed a major bug that was causing Linux installs to save the pfeMAME.ini file in the wrong path when setting up file paths. This then stopped the file paths process from working. Seems it was specific to certain builds and configuration types as I had not seen it previously.

**v2.30**

* Fixed a bug that was preventing the Check All Roms module from running. I hadn’t updated its code to follow the new logging structure.
* Added a refresh after showing the busywindow when calling mame to get the -listxml as under Linux the busywindow would not always show.
* Fixed an issue under Linux where the auto-restart (if it works) would throw an error relating to an unrecognized command line variable. This was due to the change I introduced previously on how the CLI is managed.
* Improved the showing of the media player in the main screen under Linux (if you’re using X.Org, not Wayland as that’s broken still).
* The manual has been updated with more modern instructions for installing wxPython under Linux (Only tested on Ubuntu and it will differ for non-Debian OS’s.
* In TileView I have noved the keypress and mouse click binding from the tiles to the mainframe. This better manages problems where the tiles may loose focus and not respond.
* The inactivity timer was not being reset during searching. This means that if you were just searching for a long time without then clicking on anything, the fullscreen video player could kick in automatically.
* Fixed a bug that was causing the search function to not pick up on the first character until the 2nd character was being entered. Had to switch to using the wx.searchctrl which just handles things a lot better.
* Fixed a bug where when searching, each character entry would cause the ROM list to be repopulated twice.
* Greatly reduced the number of times the ROM list gets updated on startup.
* Finally fixed the annoying linter issue in functions.py. It wasn’t happy about not seeing the \_ declaration for the language settings.
* Did a better job of passing back variables from the joystick setup function
* Added a feature in the setup.py file for Linux builds where it will automatically search for and copy some dependency Linux Libraries that are not always present on Linux builds (e.g. libsdl and libtiff).
* Tidied up the setup file a bit
* Simplified some unnecessary code in the alternateemulators.py file
* Moved the UltimateHeaderRenderer code to functions.py as it was used across three different modules
* Patched ultimatelistctrl to allow the currently selected row foreground color to be adjusted. This is not possible by default and causes for some horrible contrast issues.
* Added the color control to pfeMAME preferences window to allow the selected row color to be changed.
* Fixed tooltips not being displayed for rows in the ROMs list that have text too long to be displayed in the cell. Now when you hover over the cell a tooltip will pop up showing the full text. Needed to add an event.skip() to the mouse motion event as this was blocking the propagation of this event for the ultimatelistctrl hover detection.
* Added error / info logging to the fullscreen video player
* Created a non\_mame\_defaults xml file and added functionality in the alternate emulators window to select defaults which helps with making the right detault settings for various emulators. The XML file can be added to over time based on others that I test or feedback from other users.
* Fixed a problem that was causing a crash in Linux with the fullscreenvideo player with Wayland when using the wx.mediactrl. The Gstreamer buffer / stream (whatever it is) wasn’t getting flushed after each video and everything would fall apart. I think this is an issue with wx.mediactrl. For now I am destroying the control and re-creating it each time a new video plays.
* Implemented the same as above in pfeMAME.py
* Implemented the same as above in tileview.py
* Added a ‘Check for Wayland’ checkbox to the preferences window. On startup under linux this will be checked. You can optionally disable this after the first time at the prompt, or from the preferences window.
* Added a new preferences option to ‘Disable artwork for all ROMs’. Does pretty much as it says.
* Fixed a bug that would stop the ROMs list from regaining focus when switching to a different tab and then back again.
* Added a check when opening the ‘CheckAll’ window that the File Path for the ROMs folder is valid.
* Updated rungame to log the correct engine being used (was always logging as MAME even if you were not using it)
* Tidied up how the delimiter is shown in the messages window after running a ROM
* Fixed an issue that caused preferences to be written when opening any of the preferences windows when they should only be written if a control is actually changed
* Refactored some of the preferences control update code to tidy it up and make a lot easier to maintain.
* Fixed a bug that stopped manual configuration data in SLOT configs from ever being written to file as it didn’t realise the control had changed.
* Database age was being shown as 0 days if the preference to not check on startup was checked. Changed this so that preference only effects if you are notified of the age. Now the age is always shown as the correct value.
* Added a Unbind in on\_close to the on\_paint event in TileView to prevent Runtime errors being generated when the on\_paint tried to paint to a device that had been destroyed during the closing process. Only happened under some situations but better to prevent it.
* When using the language picker, the currently selected language now has a border drawn around it so you know what is already selected.

**v2.29**

* Fixed a bug that was causing a Thread not closed warning under linux. Seems you need to force destroy the instance of wx.adv.joystick as it keeps a thread open.
* Forced stop the joystick polling timer as having that running when the application is closed can cause errors sometimes.
* Fixed a bug that would prevent snapshots from being displayed if the folder for Ini files was not defined – even though that folder is not required for snapshots (it is for the rest of the image types)
* Force stop added to all timer events in Tileview to ensure a timer event doesn’t try to occur after the application is closed
* Fixed a bug that would cause the cursor up event to be triggered with the POV/HAT timer event if a joystick/gamepad was plugged in that didn’t actually have a POV/HAT.
* Have re-introduced the minimize / maximise buttons on the main window
* Fixed a bug that would cause the ROM image window to resize to the same size as the ROM image. For smaller images this meant it would not then resize back for larger images.
* Switched to using argparse for command line parsing as it allows for much cleaner code and simplifies things. At the same time fixed some bugs (and probably introduced others).
* Stop video playback if window is resized – this ensures we don’t have floating remnants of the video player windows in the wrong place.
* Added a -novlc command line option to disable VLC playback even if its available and force the inbuilt mediactrl (which is horribly terrible and lacks good support). This is passed through to the fullscreenplayer and tileview.
* Added a video title / file name display above the video in the full screen player and removed the overlay option for VLC that I hated.
* Added support for ffprobe to produce the video ‘Title’ for the fullscreen video player. You need to have ffprobe.exe in the system path for this to work. If not, then it falls back to just displaying the video file name.
* Made the fullscreen video player background black to make transitions between videos look nicer.
* Changed the color for the rominfo window. I think it looks cleaner this way.
* Improved the look of the XML Info window (A bit).
* Added a ‘Test’ feature to the alternate emulators setup page. This allows you to check that the selected ROM for the system you are setting up can be run successfully. It uses the same run environment as the normal ROM running process.
* Fixed a bug that would cause an indexerror in the Fullscreen video player if there was only 1 video file and it tried switching to the next video.
* Added a check in the main page when switching to the fullscreen video player if a valid folder is not set up or there are no files in the folder. A prompt is shown if you manually selected to show the player, but this is suppressed for automatically running it as a screensaver.
* Fixed a bug in the alternate emulation engines page that would not allow you to ‘Default’ the ROMs folder if you had already set it to something.
* If running Linux and the wayland screen manager is found running (rather than X), video playback will be disabled. I have had to do this as wxWidgets doesn’t seem to currently work well with wayland and quite specifically the VLC and wx.MediaCtrl players. Wayland wont expose a hook to a window so the player windows cant be assigned to a wxPython window and all heck breaks loose.
* Fixed some bugs where the Full Screen Video player was still available if video playback was disabled. Have also greyed out the menu item for this option.
* Added a command line option -forcevid. This will override the check for the X screen manager. Then run the application using the methods below (first one is for development and 2nd one is for distribution.
  + GDK\_BACKEND=x11 python pfeMAME.py -forcevid
  + GDK\_BACKEND=x11 ./pfemame

**v2.28**

* Fixed a bug in message formatting for the checking of software versions
* Added a note in the manual regarding merged rom sets – pfeMAME does not support them.
* Altered the ’Systems’ title in the manual to better reflect that I am referring to software lists support.
* Added pfeMAME preferences to force all ROMs to be run either in a window for full screen. This will override any ROM specific preferences. Under Tile View it will still always be run full screen.
* Altered panelslots.py code so it can be run locally for testing
* Removed some unused variables passed to panelslots.py
* Added a preference setting to allow the checking of software version on startup to be disabled
* Added a preference setting to allow the flashing of the messages window when an error is detected to be disabled
* Fixed some rubbish code with the inactivity timer where I was defaulting a text value to an integer if it wasn’t set. I have no idea how this didn’t break things.
* Fixed a problem where the ‘Bios’ value for systems programs was not showing correctly when first viewed for a specific program. When you viewed it a 2nd time it would look ok. This was due to the 2nd time viewed it was reading the INI file that was created automatically the first time it was viewed. Since I switched to the new controls for better color options, the new control draws slower so if I write the value too early its not shown. Added a wx.Callafter to work around it.
* Fixed an issue where after loading the preferences tabs that updateini function would be called automatically even though we haven’t made any changes yet. Added a block flag so that it cant call that function until after the pages are loaded and the controls set.
* Fixed some very poorly written code for writing mame.ini file values. For all of the variables that I write, I was writing to the file for each variable to update, rather then just creating a large list of data to write and then bulk write it. Not good at all. All nice now.
* Added pubsub listeners to the panelpfemame.py module so that the checkbox’s for checking the catver.ini file and software versions can be disabled remotely. This way the controls look right if you disable that feature from the message popup and then go to the settings page to look at those checkbox’s.
* Added a ‘Finder’ to TileView. Press ‘F’ and a search box will appear. Enter the ROM name or partial to filter the main ROMs list. Obviously, you’ll need a keyboard for this.
* Changed the order of the main group of checkboxs in pfeMAME preferences window. It makes a little more sense now.
* Fixed some encoding issues in create\_language\_files.py tool.
* Removed English translation as that’s the default. Added fallback=True so that when it cant find the English translation files, that becomes the fallback.
* Added Japanese, Chinese (Mandarin) and Russian to language options.
* Improved some translation encapsulation issues
* Hovering over a country’s flag in languagepicker now shows the country as a tool-tip
* Fixed incorrect string encoding in create\_language\_files. Forced default to utf-8 so that the Asian character sets are recognized.
* Improved languagepicker code to allow additional languages to be more easily added
* Updated the logging handler to specify utf-8 encoding.
* Placed the pfeMAME preferences checkboxs into a scrollable panel. This was required due to different languages text causing the window to size too large.

**v2.27**

* Added a cache file (xmlcache.dat) that is written to the pfeMAME folder after a full database refresh. This is essentially a copy of the output from the mame.exe -list xml command. Then on the next database refresh I read that file and compare its MAME version info with the MAME version info currently running. If they match, then I use that data. If not then I have to perform the full -listxml which takes longer. This has reduced a full database refresh on my machine from 36 to 24 seconds.
* Fixed a but that stopped the visual pinball roms from working. I hadn’t updated that code in quite a long time and found that the list format was no longer correct. I need to find a better way of doing this moving forward.
* Removed the right click context menu items for playing ROMs with different emulation engines.
* Removed enginetouse.dat file and related code.
* Added a new non\_mame\_exulators.xml file.
* Rewrote the playgame code to allow for reading different emulator engine options from the XML file.
* Added additional code to the database build to support the new alternate emulation engine process.
* Added a new menu item under the Edit Menu -> Setup Alternate Emulators.
* Added a new module -> Alternate Emulators
* Fixed a bug that caused the filepaths buttons to be greyed out on first-time-run even though they were enabled. Just needed a refresh after the first time run popup message.
* Removed any code and manual references relating to the psutil and keyboard imports and ability to shut down MAME by holding a key. This never worked properly as various platforms think you are trying to run a key logger and block it or require admin rights which is bonkers.
* Added a button hover highlight effect
* Added a bold font and underline to the slot headings
* Other graphical improvements to the slot controls page
* Additional changes to controls in preferences window to support color modes
* Improved the refresh of the quantities section of the statusbar when searching. Still not perfect but at least better.
* Added a startup window to select the application language
* Added application language selection to the preferences window and command line
* Added a framework for the application languages (still needs a lot of work)
* More changes to preferences window layout to sizer better for different languages
* Moved logging to functions.py module and got rid of write\_log\_remote. Any module that now needs to write to the log file can set up local logging or receive the logging object through init.
* Added line number and module name to the logger to assist with debug
* Fixed setup.py to handle multiple folder imports with setuptools
* Updated build\_source\_distribution.py so that for folder creation with all sub folders and files you just have to include the top level folder and refer to it as ‘folder\_all’
* Updated the manual to include language selection and the new alternate emulation engine support.
* Command line errors will now generate a message but not then start the pfeMAME application. This way you can see they you had an error.
* Updated the search box so that when it looses focus, the search text gets cleared. Fixes an issue where the box locks up when it looses focus after some text is entered.
* Updated the setup.py file for building distributions. Under macOS you need to put the ‘locales’ language folder within the ‘Resources’ folder. To do this just define a second insertion point for the data\_files setup variable.

**v2.26**

* I have dropped the support for force shutting down the MAME executable. The support for the keyboard monitoring was flaky and sometimes just crashed everything. Also support under Linux and MacOS is almost non-existent due to those operating systems being more secure.
* Fixed the bug that stopped the log file viewer from scrolling to the bottom of the log on MacOS.
* Fixed an issue I started getting with SSL certificates on some platforms when checking for the software update. This was throwing an exception and so I couldn’t find the current software version from the repo.
* Fixed the toolbar button sizes so that the hover text on all platforms aligns correctly.
* Changed the view and search box’s to be default size. MacOS was complaining about my trying to make the control too large.
* Created a custom search control so that I can control the text and background colours as well as the hint text color. The default control was too strict in MacOS. Doing this has lost the pretty magnifying glass but maybe later I will work out how to add that back in.
* Changed the ROM filter to a wx.ComboCtrl with custom list control as the drop down. This allows me full control over the text and background colors across all platforms.
* Finally ditched the generic status bar and created my own custom status bar. This gives me full control over the colours across all platforms (Yes I’m pointing at you MacOS).
* Switched to using pygauge rather than the inbuilt gauge for progress bar display. This allows me to set the color and I just like it more.
* Moved the main screen progress bar to be an overlay rather than the small patch it used to be contained in. Looks nicer and more modern to me.
* Made the quantities info on the statusbar readonly (Yes you could actually delete it).
* Stopped the statusbar quantities field from being able to be given focus at all and no left click highlighting.
* Pop-up image viewer was not closing with the ESC key
* Altered the pop-up image viewer to use the full vertical space available.
* Added keyboard shortcut SHIFT-V to launch the fullscreen video player
* When the full screen vide player is running, pressing the space bar will skip to the next video. Any other key will exit.
* Changed to using a callafter to set the ROM tiles focus back after running a ROM in TileView. Have done this to try and fix an issue whereby sometimes after running a ROM, the tiles don’t get back focus and you loose keyboard navigation without clicking the left mouse button (Which forces focus on the first tile). The callafter should allow all controls to be redrawn before trying to give focus.
* Removed an annoying underscore form the output in the RomInfo screen.
* Changed the font\_info function to return a named tuple. This makes my code cleaner in each module that I call that function to pull out the font information and set local font variables.
* Added a Bios drop down list to the Slots preferences page. This is populated from the xml file. This allows you to set ROM specific Bios configurations (Like if you want to always have the spectrum ROM using the GOSH bios, or Amiga 500 using the 3.1 Kickstart ROM). If nothing is found in the ROM specific INI file then it will default to the first BIOS in the list.
* Dropped the auto restart after certain preferences are changed. It never worked too well across different platforms, and even on Windows after restarting I was finding control bindings stopped working.
* Fixed the button colors not working properly on Linux. Needed to be specific in the colors.
* Updated the slot control buttons to use the global color theme.
* Fixed a bug where the random ROM runner could select a ROM that is one too far off the list which would cause an unmanaged exception.
* Fixed a bug in the backup code. If a file that was listed to backup didn’t exist (like if you have no favourites) then the process would fail badly. Now it will tell you that the missing file has been skipped.
* Fixed similar bug in the restore code.
* Fixed a bug where if a video play is not available, the playback timer would keep counting down in a loop.
* Fixed a bug in the code I use to check for valid values for Audio volume. It seems that using isnumeric() for negative numbers thinks they are not a number. Have to strip the leading ‘-‘ off the string.
* Major change to preferences. I have moved the preferences window into the main window as a tab shared with the ROMs list. Also done the same with the ROM specific preferences. They are now side-by-side.
* Improved the slot configuration’s view – it now expands to take up the horizontal space available which makes systems with no slots look better in the view.
* Changed the size of the spin controls on the Sound tab in preferences to stop annoying GTK complaints about the size being too small for the native control.
* Slot option heading and buttons will no longer show for System ROMs that don’t actually have slot options.
* Changed to using event.m\_col to determine what column has been clicked in the Ultimatelistctrl. Was previously using a calculation based on pixels that doesn’t work well enough. As Ultimatelistctrl supports m\_col where the standard listctrl doesn’t across all platforms, this seems a much better idea. This fixes an issue with ordering by Manufacturer as the column wasn’t being detected correctly.
* Replaced my custom code for replacing escaped ASCII characters with html.unescape. My code took 2 seconds while the inbuilt takes 0.07 so I think its written in C. Speeds up the -listxml read from MAME.
* Lots of general code cleanup and refactoring.
* Extra error checking in xmlinfo.py to check if the softwarelist ‘hash’ folder and relevant file doesn’t actually exist.
* Layout improvements in xmlinfo.py.
* Corrected some misleading naming in tileview.py (self.rom\_name that was actually a version of self.rom\_file\_name but not really for example.
* It seems I have been using the shelve module incorrectly. It returns an object, not a list, but still seems to work. I have rewritten the code a bit to ensure a list is returned.
* Created docstrings for functions in functions.py. Still a bit of a WIP but a step in the right direction.
* Some slight improvements to dark mode coloring. Notebook tab background is now set to the window background color and inactive tab text color is the same as window text color.
* Fixed a bug where if the color string is not set in the pfemame.ini file, it would throw an unmanaged exception as the default type was a colour object, not a color string.
* Added some more defaults to the pfemame inifile creation for first-time-run.

**v2.25**

* Changed the Unsupported format popup message for video playback to a message in the messages window. The continuous popups that had to be acknowledged were quite annoying.
* Removed the ‘Clear’ , ‘Copy to Clipboard’, and ‘Reset Spacing’ buttons added new menu items for this functionality to the EDIT menu. This gets rid of one sizer and also makes the interface cleaner.
* Expanded the ‘Reset Spacing’ menu item to reset all column spacing.
* Moved default column spacing values out to a constant section
* Moved a whole heap of constants to the place where they should be (Just before the main class.
* Moved the wx.MediaCtrl object creation into a function as the same code was used in three separate modules.
* Moved code that determines which media player to use for video files into a function as it was duplicated across multiple modules.
* Updated the version checking code to look for updates based on the operating system type you are using. If there isn’t a build update available for your OS then you don’t get the notification (Even if there is a newer source file).
* Added elapsed time to the application logfile.
* Added -? to show command line options and -v to show the pfemame version
* Added a pop-up image viewer. When you click on the ROM image a larger version pops up onto the screen. Also added a magnifier icon to the bottom right hand corner of the displayed ROM image.
* Fixed a bug that was preventing the ROM image from being displayed after the ‘Playing Video – Click to stop’ button was pressed.
* Updated the fullscreen video player to check for an allowable list of video file extensions for playback. This better protects it from incorrect file types.
* Stopped the fullscreen player from throwing an exception of there were no video files to play
* Removed the ‘MainFrame’ inheritance from the virtuallist class as it actually wanted needed and was preventing the virtual list control from being added to the panel under testing for python3.10 & wxPython 4.2.0 release (Which is nowhere ready yet due to other rending issues being caused by the ultimatelistctrl).
* Moved to using python 3.10 and wxPython 2.2 – this also moves the build to 64bit only. NOTE: You must have a 64bit version of VLC player installed in order to use that as the video backend. The 32 bit version is no longer supported for this application.
* Added the note to OSError exception when the vlclib.dll file cannot be found that you need VLC 64 bit.
* Found a bug in wx.lib.agw.ultimatelistctrl. The OnPaint event was throwing an error due to the vars within the Rect() being float rather than integers. Have raised this as a bug on wx. Until this bug is resolved, only bundled versions of the app will work unless you manually edit the wx.lib.agw.ultimatelistctrl entry yourself.
* Fixed a bug where on the main view when a VLC video ended, the ROM image was not displayed.
* Removed unused entries in busywindow.py
* Added a local copy of ultimatelistctrl.py until a python3.10 related bug is fixed by the developers.
* Refactoring of the colour control code in preferences
* Added color HEX value to the color controls in preferences
* Added the MAME preference ‘artwork\_crop’ to the Video tab of preferences. This can work globally or be ROM specific.
* Refactoring of valid numeric field values code in preferences
* Added an option to globally disable ROM artwork from being displayed. This is in the Video tab in preferences. It is global in that at present you cannot allow to only disable for specific ROMs.
* Some code refactoring in Tileview
* Converted a number of bulky for loops into list comprehensions
* Moved the Joystick setup code in pfemame.py and tileview.py into a common function
* Main screen toolbar objects background color now matches the window background color so that dark mode looks right.
* Added a keyboard monitoring event to rungame so that the MAME executable can be shut down by holding down the Coin button (5) for 5 seconds. This is currently only supported under Windows.
* Fixed a bug in preferences (same bug I fixed in Filepaths) that under Linux & MacOS would cause the restart notification popup to appear under the window so you could not close it.
* Added a border to the ‘please wait’ window to make it more obvious.
* Fixed the ‘please wait’ window text not following the application colors. In dark mode you almost couldn’t see the text.
* Added a confirmation to the ‘remove from favorites’ to prevent accidental deletion.
* Added a check to ensure the ROM was actually flagged as a favorite before trying to remove it.
* Fixed a few old path concatenations that should be using os.path.join rather than +.
* Added explicit utf-8 encoding to the subprocess call in xmlinfo as Python10 was causing a Unicode error.
* Fixed the issue with ROM names being displayed incorrectly due to getting the data from the -lisxml output but not correcting the & escape sequences.
* Fixed a bug that was stopping a number of Systems / Systems programs being added to the ROMs list with FullSystemsSupport selected. This was due to output encoding not being specified as utf-8.
* Made the Rom Categories double click function a lot more pythonic.
* Added error capture to the database\_build module so that when using multiprocessor support any errors generated in the multiprocessor pool are reported back to the main loop. A message will be displayed if exception errors were caught in the process and they will be dumped to a build\_errors.txt file.
* Force a refresh after some of the info message pop-ups like catver in ini file mismatch or new version. This fixes the issue of the ROM categories not being drawn correctly in Windows.

**v2.24**

* Added additional exceptions checks to the import vlc in pfemame.py, fullscreenplayer and tileview. If there was a library issue the application would throw and exception and refuse to start. Also if you just are not interested in VLC or its not supported on your system then it just warns that its unavailable and drops back to mediactrl.
* Added additional file validity checks into the fullscreenvideo player to try and prevent it from trying to play files that are not actually video files. This is necessary as on MacOS there can be hidden files in folders.
* VLC video play code in pfemame.py moved to a thread. This is mostly to support some hacks I am trying to use to get VLC support to work right in MacOS. Right now it doesn’t scale the video to the window – apparently a known issue that there doesn’t seem to be a good work around right now. As such VLC playback is still only supported for Windows (Yes Linux is a problem due to distributions moving to Wayland and the VLC plugin only working nice with X servers).

**v2.23**

* Removed the check for valid file paths from FilePaths. It was a major headache and caused issues when starting a fresh instal under Linux as Mame sets some defaults that may or may not exist anyway.
* Changed Filepaths to check internally if a manual or automatic restart can be performed based on OS type. This stops the issue of the prompt appearing behind that window and locking the application.
* The Filepaths auto find under Linux no longer tries to look in paths that you didn’t specify. That is, if you specify your Mame folder in your Home folder, autofind wont try looking in places like usr/games etc for any of the various folders. This fixes huge problems for new installs. You need to manage your folder locations yourself and just point to where they are for auto-find to find them.
* Filepaths autofind for Linux will now handle ‘SNAP’. The MAME executable is not longer specified explicitly but rather just listed as ‘mame’ as it should be in the operating system $Path variable. I no longer try to find the exact location of the executable file**.** This was needed as you can’t do that under SNAP installs. So you MUST use Auto-find in Linux for the executable to be defined correctly for SNAP installs.
* Fixed a bug that would prevent the checkallroms feature from working if you had more than one ROM path defined in filepaths. It will now only look in the first path defined.

**v2.22**

* Replaced the wx.busyinfo with my own pop-up window. Had to do this as busyinfo doesn’t seem to work on Linux or MacOS anymore. I like my version better as I can customise it.
* Added a SetFocus for the first ROM button in TileView after starting. After recent changes this was preventing navigation unless you used the mouse to click in the window (Under Linux and MacOS).
* Added further error trapping for systems that cannot play media files.
* Have tested pfeMAME under Windows Subsystem for Linux with Ubuntu (WSL2). Had to add a few extra error traps for video playback as I haven’t worked out yet how to get that to work in WSL2.
* Added additional command line switch -novid. This disables video playback in the main and tile views.
* Added a pfeMAME preference ‘Disable All Video Playback’
* Added python-vlc as a required external library. This is for a feature I am working on.
* Added VLC as the primary option for video playback under MS Windows. This gives much better playback file support, but it does require you to have VLC installed on your machine. Unfortunately, on Linux and Mac OS this doesn’t currently work as I am having problems getting VLC playback to work correctly. On those platforms it will default to wx MediaCtrl.
* Added a video screensaver. The inactivity time is set in Preferences (Set to Zero to disable). After the pre-set amount of time inactive a fullscreen video player will launch (if video playback is supported). The player plays a random selection of videos from the Video folder (if you have one defined). Press any key to exit. NOTE: If not using VLC as the video player you will have the same limitations above in regard to video file support. If the file is not supported, you may just get a black screen.
* Added the right & left gamepad bumper controls to navigate page up / down in the ROMs list.
* Added support for the gamepad D-Pad up / down to navigate the page up / down.
* Fixed joystick navigation in Tileview from not stopping the ROM video causing the currently playing video to move with the navigation.
* Added preferences to selectively allow the joystick navigation to be enabled in the main view and tile view. By default, they are enabled. In Tile View, if you are using a cabinet with a USB joystick device mapped to the cursor keys, then you will likely want to disable the joystick navigation support in preferences if this causes any issues.
* I have updated the layout of the preferences window again as there are a lot more checkbox options and I was running out of vertical space.
* On first time run the File Paths toolbar icon will be highlighted with a RED background to indicate this is where to go first.
* Added an option under the ‘View’ menu item to launch the fullscreen video player (AKA screensaver) manually.
* Added a button to allow you to reset the ROMs list horizontal spacing back to its default startup size. A nice easy way to get the view back to normal after expanding columns.
* Modified the build setup file to import pfemame and use variable declarations from that file for various attributes (e.g. \_\_version\_\_). This way I don’t have to up-rev both the main file and the setup file every time I release a new version.
* Updated the image display type on main screen to use the custom colors so that it can support dark mode.

**v2.21**

* MacOS when restarting prompts the user to restart manually. It would then still try the non-automatic approach (which it’s not supposed to) and issue a crash report. Fixed that.
* Fixed the manual not opening on MacOS (I was using os.system rather than self.system).
* Fixed an issue on MacOS where in Tileview the scrolling rom info window would show its scrollbar first time until you moved to a different ROM.
* Fixed MacOS building where the data files were being put in the wrong location by the build script and I had to manually add them to the right location. I was missing the additional part of the data\_files tuple that specified location. The default location for MacOS is not as you would expect. Updated the manual also to reflect this.
* Moved the systems slots preferences to their own tab in ROM specific preferences
* Changed layout of the preferences window – hopefully sizes correctly now.
* Added a dialog if the MAME executable cannot be auto-found due to permission error which is usually because I am looking in the wrong place or because you have not installed MAME yet.
* Redesigned the preferences window to put the system ‘slots’ configuration into their own tab rather than sitting under the other tab pages.
* Added management of systems that don’t have media playback capability – will no longer try to call any media control functions (Rather than just locking up, crashing, etc).
* Combined the MacOS setup file into the same file as Windows / Linux so that there is just one setup.py file.
* Added a VeryLarge option for tile sizes in TileView
* Made the size of the image / video in the main view just a little bit larger
* Added ability for filepaths to handle ui.ini paths (These are needed for different artwork types like pcb, marquee, etc).
* Added image option selection to the main screen. You can now select what image you want displayed (PCB, Snapthot, Marquee etc).
* Mainscreen images now resize while maintaining their aspect ratio. They are also centered in the view.
* Removed ability for the main window to be resized or maximized. Given the layout design there is currently no real practical need for this ability and when it was implemented it didn’t work that well anyway.
* You can now sort the displayed ROMs list by column by clicking on the column header.
* Changed to using -listxml rather than -ll when building the MAME roms list. Its slower (takes about 6s longer on a fast system) but it’s the only way to get additional info about the ROM (year, manufacturer etc).
* Added a ‘busy box’ popup while the MAME -listxml is being run just to let you know that things are ok. There is no progress bar during this phase as its all internal to MAME and I have zero control over it.
* Added Year & Manufacturer to the roms list
* Adjusted ROMs list column widths for better fit and reduced font size from 10 to 8 point
* Force a full database rebuild if the loaded database on startup doesn’t match the defined size (Situations when I add new columns to the display will cause this).
* Added a few extra icons to the menubar menu.
* Improved tileview code more to not require setfocus at all. Also improved some math.
* Reduced most of the flickering in TileView when moving over a ROM that has a video to play. I wasn’t stopping the video playback when moving.

**v2.20**

* Fixed an issue with packaged file size. It was adding pygame under windows and also tkinter, neither of which are used. It’s a problem with cx\_freeze crawling my source folder, I think.
* Refactored the toolbar creation code. Much easier to read now.
* Clean-up of numerous PEP violations.
* ROM Sample checks were not working correctly. Needed to change to the MAME path. Was also not checking if there was an error before saying that everything was actually ok. Also added a message to the output window if the samples folder has not been defined as that is kind of necessary for this to work.
* Added the ability for the pfeMAME main application to restart itself. An additional developer button has been added to force this. On restart a message will be added to the messages window to indicate that a restart was forced. The plan is to use this to allow preferences and first time setup changes to force a restart rather than you having to do this manually.
* Changes in preferences settings will cause the application to try and restart. At the moment is will restart minimised so you need to click on the icon in the taskbar.
* After setting up the file paths for the first time the application will auto-restart.
* On initial startup when the database has to be refreshed for the first time I have moved a chuck of code into a callafter to allow the display to be completely drawn before calling these long running tasks. This makes the display actually look as it should while the database is being refreshed.
* Added a check if there was any quantity of joysticks added in panelpfemame. If not don’t try to read their info as that would crash on MAC.
* Added an IndexError check to play\_game in tileview (only found this issue with MacOS but it should have been there anyway)
* Added os.getcwd() to the list of paths to check for file/folder matches in auto\_find for file paths. This makes sure it can find catver.ini file if you’ve not included it in the MAME folder.
* Added utf-8 encoding when reading from the mame.ini file just in case there are some non-standard characters in there.
* The preferences window was not clearing the slots header text when slots were reconfigured. Until recently I didn’t realise that it was actually being hidden behind controls and then being drawn over.
* Rewrote some version checking code and moved the now common process to functions
* Started testing on MAC OS. Changes required so far;
  + Move the menubar color definitions to after the menubar is set
  + Added specific platform detection for ‘mac’ as Darwin.
  + Displaying pfeMAME manual
  + Filepaths & inifile code need to treat sections where I check if Windows or Linux as Windows like (root paths etc are same for MAC as Windows when looking at the MAME folder)
  + Added a .icns icon file for the application
  + Created a MAC specific file setup\_mac.py for building a MAC distribution
  + Added self.Show() to tileview as the show full screen wouldn’t actually show the window on MacOS
  + Added mc.Raise() to tileview to ensure the media control is visible in front of the static bitmap button.
  + Changed tileview to use an index to remember the currently selected button (was previously using setfocus and getfocus methods but they don’t work on MacOS.
  + Added a lot of utf-8 encodings to file read & writes as MacOS app bundling has an issue if you don’t.
  + Have changed the preferences window to use a fixed size rather than auto-sizing based on the requirements. Had to do this as MacOS was not handling the auto-size very well at all.
  + Changed to using a generic button rather than the application based wx.Button. This is needed as on MacOS you are limited by button sizing and you cannot change the button background color.
  + Changed the scrolling rominfo control in TileView from textctrl to richtextctrl so that I could actually change the text foreground color as MacOS has further limitations to foreground colors for native text controls.
  + Changed the onscreendisplay toggle key for systems from FD-Delete to DEL which is correct for the MacOS
  + Updated the manual to include MacOS support

**v2.19**

* Fixed a bug introduced in the previous release that was corrupting the check all ROMs saved text output.
* On first time run, all ROM categories will now be selected, rather than none. This helps the user to actually see ROMs in the main list after the first refresh or restart.
* Increased the toolbar image button sizes to make them a bit easier to see and to also fix the issue of Linux cropping the images too much.
* Removed the border around the toolbar images.
* Fixed the toolbar ‘hover/focus’ image not being displayed in Linux. No idea why it doesn't work as intended – have coded a fix to force the alternate images to be displayed and then restore the default image once done.
* Fixed the about box animation not being displayed right in Linux – apparently in Linux you need to specify the size to ensure it displays right – go figure.
* Default sound options were not displayed right under Windows (not sure about Linux). This is due to the ‘default’ sound device being changed to ‘Auto’ when it used to be ‘dsound’. Latest MAME docs don't seem to reflect this.
* Minor changes to the text style for the ROM quantities in bottom right of main view. Make it easier to read.
* On first time startup, the first ROM in the list will now be selected. This means that if the first thing you select is Random Play, it will work.
* ROM info window was not centered on the screen
* Have added a StaticLine (horizontal line) on some windows as a delimiter between the main stuff and the buttons row. Just makes the window look better.
* Excluded wx.StaticLine from the set\_colors function as under Linux it gets screwed up so you cant see it.
* Changed to using double click to clear the Alert Flashing background – was previously single click but this overrode the ability to highlight text for copying from the window.
* Changed the rotating error flashing color list to an algorithm
* Changed the default view in TileView to All ROMS rather than Favorites. For first time users opening to a potentially blank screen because you haven’t set any favorites yet probably makes them think they have done something wrong.
* Selecting ‘Refresh’ on first time run before you have defined any folders would cause all ROMs in the categories view to be deselected.
* Prevent the checkall window from being opened on a clean install as there won’t be any ROMs available in a path to be checked.
* Allowed the scrolledwindow to be resized. In Linux the default font sizes are different from Windows and some more wordwrap is occurring. Also, so much info exists that it should be ok for the user to resize the window for ease of use.
* Fixed issue where video mode ‘Auto’ was not displaying in the Video preferences. I had fixed this for the BGFX backend but not for the main VIDEO.
* Removed references to Windows 7 support from the About box and manual. Recent versions of python are not supporting Windows 7 (I believe).
* Added a note to supporting Windows 11 as I have now tested this.
* Combined the code used in pfeMAME.py and TileView.py for loading the history dat file into a common function.
* Combined the code used in pfeMAME.py and TileView.py for parsing the ROM information into a common function.
* Added logging function to TileView
* Fixed vertical size of the instructions window in TileView – bottom rows were being cut off.
* Added media playback to TileView. Pretty cool. The video, if available, will play in a loop.
* Force the video timer to stop when the video playback is stopped. This ensures a video won’t start playing when it should have been stopped. TileView could actually start when a video was due to play in the background which is just wrong.
* Added joystick navigation to the mainview. Up and down to select and button 1 to run the ROM.
* Fixed a bug that may have been around quite some time. Joystick events in TileView were bring triggered when a ROM was run if using a joystick within the ROM. This caused a total mess and made the ROMs unusable.
* Added freeze/thaw to the game image update when scrolling down a line in TileView. Makes things a little more fluid.
* Added selection of which joystick to use for pfeMAME navigation. Was defaulting to Joystick 1 which was a problem in VMs where often J1 was the mousepad. Probably also caused all sorts of issues on various setups. The selection is in the main pfeMAME preferences window. Lookups for joystick hardware name under Linux & Windows have been managed as well (Yes they don’t work the same like a number of things).
* Added management for when a media playback system is not available on the operating system, or if wxPython has been built without this support. The media control playback is now disabled rather than raising a notimplemented exception and exiting.
* Have tested operation under WSL2. The application works (although some manual playing with file paths may be needed). It can also run MAME but getting sound to work is almost impossible with WSL at present.
* Added page down and page up button control to the main ROMs list
* Fixed (mostly) a very longstanding issue I had where with some large page jumps the main ROMs list would not redraw correctly and leave blank lines. This is an issue with the ultimatelistctrl that never seems to have been fixed completely.
* Added some expanding in the TileView information panel to better utilize the space available.
* Fixed an issue whereby the video timer was not reset when moving between ROMs. This could cause it to try and play a video for something that no longer exist. Under Linux this would throw a major background error (but not stop things working).
* If set for manually clicking to play video, after clicking the ROMs list is given back focus.
* Main window checkallroms progress bars are now echoed in the application icon under Windows (And MAC however work needs to be done to get pfeMAME working under MAC as I don’t have a development platform for that OS).
* Added auto scrolling of the ROM information in TileView as you can’t scroll it manually.
* Added support for displaying the mameinfo.dat file information for the selected ROM in the main view. The Rom Information window has been changed to a notebook with two tabs (One for ROM Info and the other for MAME info). The filepaths window has been updated to include support for selecting the mameinfo.dat file.
* The changelog can now be viewed from the About menu on the main screen. NOTE, this does require python-docx to be installed using pip if you are using the source code.
* When selecting a color in preferences, the color picker is passed the current color you want to change rather than defaulting to black.
* Changed the Dark Mode background color to Charcoal rather than black. I think the contrast is easier on the eyes.
* Changed to using FlatNotebook rather than the standard Notebook. This allows me some level of control over the tab coloring so that I can better support the dark mode.
* Added control bar color to preferences.
* Changed some controls to use the control bar color rather than window background color. This way I can add a bit more contrast to the display.
* Added a custom renderer for the UltimateListCtrl header so that I can implement the control bar color.
* Quite a big change made to Filepaths code as I have moved from using a ListCtrl to UltimateListCtrl pretty much just so I can use the custom renderer to get the dark mode header support. While I was doing it I also found a few bugs. This change will also help me better support color themes moving forward.
* Fixed incorrect coloring of text in header rows for ROM Information.
* Altered Dark Theme colors so that the white part of alternating line colors is now light grey.
* Fixed a bug that would cause an error if the history file path was not set
* Fixed the restart required text on the preferences window which was getting cut off at the bottom

**v2.18**

* Added logging of which history file (DAT or XML) is being loaded
* Changed the default Font size for the game categories list to the same as the message box’s
* Used inherency to apply Font types to all controls on the main window
* Added check if game video is actually playing before processing all of the ‘mc\_forced\_stop’ code. This prevents unnecessary text flickering on the video play control when scrolling using the cursor keys.
* The changelog file is now docx format (Its smaller this way)
* Added a new menu item Help > pfeMAME web page. This opens up the sourceforge web page for the application.
* Added a checkbox to pfeMAME preferences window to check for the roms database age on startup.
* Added a checkbox to the popup displayed when the roms database is too old. The checkbox allows you to prevent the message from displaying again. This can be re-enabled from the preferences menu.
* Added a checkbox to the popup displayed for checking catver.ini file version mismatch. The checkbox allows you to prevent the message from displaying again. This can be re-enabled from the preferences menu.
* Updated the distribution source builder to include the version in the folder name.
* Global fonts are now defined in the INI file. Maybe in future I will allow some control over the font type used.
* Fixed bug from rungame envoked from TileView. When in TileView mode the output from p.communicate is set to an empty string as I have had issues in the past with the process hanging waiting for an output that never comes. Then the code tries converting the output to ascii which you can’t do to a string. Have fixed that.
* Fixed most windows that were not fully using the global color preferences (Still need to sort out lists, headers, and some buttons)
* Added a ‘Dark Mode’ button to the preferences window (Requires application restart). This works quite nice in Linux but quite poorly in Windows. In Windows a number of the controls don’t allow you to change the system color. Adding this to my to-list to have a look at what work-arounds are possible.
* Created a modal window for RomRating (was previously using a single choice dialog). I did this to then ensure that this window would follow the system wide color scheme (The singlechoicedialog was not.
* Changing any of the colors from the preferences window will prompt you to restart the application.
* Fixed a bug in the game ratings update code that was causing duplication in the ratings.dat file
* Created a separate setup\_msi.bat file that will build Windows MSI installers.
* Fixed incorrect row color in Rom Information window for rom file with RED ‘error’ text.
* Added a ‘Copy to Clipboard’ button to the messages window.
* Improved the method used to determine what the correct path name to the system program should be when you are trying to run it. This is the path that MAME expects to see. If MAME cant find it when you are running a system program using the short name method then I try to prompt you as to what the correct path should be.
* The scrolledwindow used for the logfile and controls views was not correctly setting the text color under Linux. This became apparent when I implemented Dark Mode as I could no longer see the text.
* Updated the XMLinfo.py file to use the newer ‘correct’ path when looking for the system program. I have still kept the old path so that you can at least check a file for correctness against the XMl file but if its in the wrong path MAME wont run it using the short name.
* If XMLinfo cant find the file, it now reports the full paths that were checked – this makes it easier for you to go and have a look.
* Fixed incorrect location of the OK button on the XMLinfo page.
* Replaced a few remaining subprocess.Popen with the newer subprocess.run. Added universal\_newlines=True and removed the now unnecessary code that I had put in place to handle different newline characters from different operating systems.
* Added overlay statictext controls over the left & right sections of the status bar. This allows this area to use the application colors (The wx.statusbar doesn’t allow you to change the text foreground color in Windows)
* Modified the RH statusbar size as it was truncating the text a bit.
* Replaced the built-in wx.toolbar with my own custom toolbar. This allows me better cross platform control of the toolbar items, colors, etc.
* Have changed the toolbar icons to what is hopefully a more consistent look. Added some mouseover actions. Working more how I want it now that it’s a custom design rather than locked into the OS which really limits what you can do. Still needs a bit more work.
* Updated the manual a bit.
* Added a ‘Reset INI file to Default’ in the Edit menu.
* Changed the catver.ini file version checking to convert the search text to lowercase as latest catver file checks were not finding the version (Someone changed the case in the file).
* Found some child windows were not checking for their parent correctly which in turn caused their window close event cleanups to not be functioning correctly (self.GetParent() was never seeing the parent as I had defined it as none rather than wx.GetApp.TopWindow().
* Added animation to about box window
* TileView changes:
  + Removed the scrolling ROMs list – was pretty pointless as you can see the ROM name under the image
  + Added a new toggle key (ALT-1) to toggle the ROM menu into view. ALT-1 should be equal to the 2nd fire button and 1-Player button being held down at the same time.
  + Some color changes and minor layout changes.
  + Added the ROM information panel from the main view
* Fixed a bug that was stopping the ‘Random Play’ feature from working on build distributions. No idea how long this has been an issue. Needed to add an event.skip() to prevent an event retrigger loop if no ROM was selected.

**v2.17**

* Added a setfocus method to all on\_close events for child windows to ensure that the list control has focus on the main window when child windows close. This was needed under Linux to ensure that when a child window closes, keyboard events are captured by the list control again - so that you can press ENTER to run a ROM.
* Fixed missing distribution file for MISC preferences panel. Could not build from source without it.
* Added a check for the MAME executable file to be where File Paths say it is. Mostly due to recent changes to MAME 64 bit no longer having the 64 in the file name, but a good check to have regardless.
* Stopped the catver version check message from being displayed the first time pfeMAME is run as there will not be a MAME executable defined so its impossible to get the version.
* I have finally found out the correct way of handling systems programs with the ‘short’ name defined in an XML HASH file. This is just too obscure and not documented well at all. It seems that for systems programs, you need to put the program in its own subfolder. The subfolder must be named as per the short name in the XML HASH file. As an example;

For the ZX spectrum program ‘The Birds and The Bees’; The program file name is ‘the birds and the bees.tzx’, while the short name in the XML file is ‘birdsbee’. Your ROMs folder structure needs to look like this;

* + - ROMS
      * spectrum\_cass
        + birdsbee

the birds and the bees.tzx

If you were to structure as per the above, and from the command line run;

Mame.exe spectrum -cass birdsbee

then MAME will execute the program using the settings stored in the XML file. When you refresh the pfeMAME database it will assume the program is stored as per above, but if you have not stored it correctly it will still try to find it for you but executing it with XML HASH file support likely won’t work.

* Also with the above, I was not strictly specifying the media type (e.g. cass or flop1) when running the systems programs with XML short names. This worked for some but not for others (e.g. for Spectrum it worked ok for cassette software, but for specpls3 it did not work for disk software).
* Corrected how I was handling the ‘software’ folder. I was treating this as a pointer to systems programs, but it is not. I have now linked the systems programs folder to the same path as the roms folder. The ‘software’ folder is stored in the mame.ini file as it should be for swpath. This is essentially just a scratchpad software path for MAME to use when you us its built in file manager.
* Had to improve how I look for the correct index match when adding systems programs to the favorites file. I had assumed that the system program file name would be unique across systems – bad idea as system clones will use the same program file name, but their index location in the database is different.
* Increased the frequency at which the mainloop is refreshed (yielded) when running the database build under multi-threading. The GUI was locking up periodically as I was allowing a yield every 500 iterations. Have reduced this to 100 as some systems path lookups are taking a while in the threads.
* If not using the ‘full’ systems build method for database refresh (for systems specifically), there is a substantial speed increase in the database refresh. This is still not the most accurate method for finding systems programs but it works quite well for basic systems support.
* Reinstated some ‘Restart Required’ flags that I had removed a while back but are still needed for pfeMAME preferences.
* Added a console output message to display an error if pypubsub is not installed. Only useful for source file work.
* Have added a list of likely paths to the filepaths auto-find process so that it will now look automatically in a number of logical locations for the folders & files. These locations are based on the MAME folder you select being the highest priority, then the mame-extras folder (very important for Linux as this is not the same location) and the snap folder location (requires a lot more testing for snap). This should reduce what you have to manually find for linux.
* Fixed a bug in Linux that would throw a PermissionError exception if you cleared all file paths and then clicked on Ok. Linux throws PermissionError when you try to use subprocess to run an executable file that doesn’t exist. By contrast, Windows throws an IndexError.
* In certain situations, the Auto Find file paths feature was not working in Windows.
* Major refactoring of the database\_build.py code. There were just too many nested if branches and things mixed up too much after so many changes. Much easier to read now.
* Added some additional exception captures to the inifile functions. Have added PermissionError to stop the app from crashing if an unacceptable file path is defined.
* In filepaths, when clearing a path or clearing all paths, they are now set to ‘.’ Rather than empty. Setting them empty was having some strange behaviour in mame.exe -createconfig. I found that specifically it was the plugins file path that causes the issue. I believe MAME is possibly not performing adequate checks to the file paths.
* Significant change to how the database\_build is passed a list of folders to search for systems programs. This is now a list of files & folders. This way the database build loop is not calling os.listdir for each folder iteration – even with caching that was never a great idea. I think its was probably bad for mechanical hard drives as well. I now pass a dictionary object with the files and folders already determined. This is a small amount slower due to overheads in calls but overall a better way of doing things.
* When running a system program, if the application detects that MAME was unable to run the program with the ‘File not found’ error, it is assumed that the XML short name method had been tried and the application will now re-try using the full path name. This disables the XML benefits but at least the program can run. The user is notified of this in the messages window.
* Fixed a bug in cheakall.py that I probably introduced in the last release. A pubsub subscriber was not working causing a crash. I had renamed an unused variable with underscore but had not changed the name in the subscriber.
* If a critical error is displayed in the messages window that its rather important for you to read, the window pulses red until you click in it to acknowledge the error. This feature works well in Windows but for some reason under Linux the color highlighting is quite odd. Need a bit more research into this.

**v2.16**

* In slots configs, improved code detection of trying to add slot rows beyond what was allowed. Still have exception handling for the index error should it still occur in unknown circumstances but hopefully this has been fixed.
* Fixed a bug in slot configs that would cause an error prompt suggesting a problem with your config file if you added a cascaded slot that has a Slot Name, but no device name (can happen, especially in early XML hash files or poorly written ones). Now if there is no device name it just allows you to rese the row to empty.
* Fixed a minor bug in slot configs. When selecting a previously selected Slot Name to ‘Empty / Blank / Nothing’, the Slot Parent cell would still show the previous parent name (for sub slots). Nothing was actually done with it but it didn’t look right so now I clear the cell.
* Copyright information from main page (pfeMAME.py) is passed through to tileview so that I don’t have to updated the year info separately. Keeps everything in the one place.
* Started converting all unused ‘event’ in callbacks to ‘\_event’ to get rid of the event unused alert in pylint.
* Updated filepaths to allow \*.dat and \*.xml wild cards for the history file. Changed reference from history.dat to history.
* Added support for the new history.xml file. The history.dat file will still work. Parsing the XML version is a bit slower but that seems to be down to it being a lot larger.
* Added command line argument -d to invoke developer mode. This will dump various additional startup timing to the messages window. At the same time I have removed the old developer mode that printed the same information to the console.
* Added a command line option -r to allow you to force a specific rom to be run once startup is complete. E.g. pfemame -r pacman will cause the rom for pacman to be run once startup is complete. The currently selected rom in the roms list will be set based on the rom name specified and game info / image are also updated. This won’t really be of any use for most people and more something for testing purposes. This probably shouldn’t be used in conjunction with Tile View.
* Fixed TileView rungame function. Wouldn’t work since including controller mapping support in previous version.

**v2.15**

* Added ‘Ctrl-P’ text to the main Edit menu for ‘Preferences’
* Added ‘ROM Preferences’ to the main Edit menu
* Major update to the manual. Lots of corrections, better layout etc.
* Right aligned the MAME version and Database Age information in the window
* Right aligned the statusbar text relating to number of ROMs etc
* Removed the ‘v’ from the MAME version information and removed a trailing space
* Added a check for the catver.ini file version. If it doesn’t match the MAME version then the user is informed with a pop-up message. This is selectable from a new preference in the Preferences window.
* Added logging of catver.ini file version
* Added detection of numpad ENTER key to start a ROM
* Fixed a long standing bug that would cause a ROM to be run if you pressed ENTER on a pop-up message. The ON\_KEY\_UP event would essentially propagate through to the window after the pop-up was closed. Changed to using ON\_CHAR\_HOOK and manually handling the up and down arrows.
* When closing the right-click context menu, reset the status bar text to ‘Ready’ rather than leaving the last selected items help text.
* Reduced the size of the category column. Most categories are not that long and for the few that are, truncating them isn’t a major issue.
* Fixed some column layout in ROM Information. BAD ROMs were not aligning correctly.
* Corrected how background panel colours were being set in Checkall ROMs.
* Bumped up the minimum python version to 3.8 in order to support new features (e.g. the walrus operators)
* Fixed a bug introduced some time ago where ROM specific preferences would no longer work. This was introduced when we added the -inipath option to the command line so that we could tell MAME where to find its INI file for Linux support. Problem was this also overrode where it would look for the ROM specific INI files. Have fixed it by providing both paths as you can specify more than one.
* Redesign of the Preferences window. Options are now grouped in a TAB view to allow for room to add additional options moving forward. This was quite a significant re-write additional bugs may have been introduced. Each of the notebook TABs have been split out into separate include files which makes support moving forward a lot easier.
* Added ‘Sound’ preferences to MAME preferences.
* Removed duplicate code from preferences.py by combining the variable definitions methods used for mame.ini and rom specific ini files.
* Added support for BGFX visual effects via the Video tab in preferences. You must have the Video System set to BGFX and you must have the MAME support folders (BGFX etc) in your MAME folder. I find the best settings as follows;
  + BGFX Backend: Auto
  + BGFX Screen Chains: ctg-geom
  + BGFX Shadow Mask: aperature-grille.png
* Added new File path ‘Mapping folder’. This is for the per-rom input mapping. Works with the new addition below.
* Added a right-click menu item ‘Copy controller mapping to this ROM’. This essentially allows for per-rom input mapping which is not supported natively in MAME. Run a ROM, define the input mapping as per normal. Then right click it and any other ROM you want to have the same permanent mapping and select the Copy option. Next time you run that ROM it will have that mapping. Please see the manual on ‘Controller Input Mapping’ for the full details.

**v2.14**

* Moved the main window Refresh to the end of the setup process. Found on a new install when the initial pop ups are complete, the game categories view is not displayed correctly. Must have forgotten to move it after adding extra code to the end of the process some time ago.
* Added Python and WX versions to the about box
* Converted the mylist.db read from relative to absolute (as was done with the write in 2.13). Don’t think it was needed but good for consistency.
* Fixed a longstanding but that would look for systems programs even if systems support was not selected. This lead to duplicates being shown in the roms list.
* Made the background color of the messages window ‘Clear’ button the same as the window panel. Fits in better.
* Added a new pfeMAME preference to suppress the detection of system program short names. These are usually looked up in the XML file, and if they exist then MAME is told to run the program as its short name. This forces MAME to pull in the settings from the XML file (things like cheats). The problem is that not all systems programs have been set up right or you may have a ROM that doesn’t match the detail in the XML file. Under that situation the ROM wont run and will report an error. You can use this new setting to ignore ALL program short names rather than using the right-click menu just to ignore the short name for the particular program you are running.
* Fixed a bug that would stop the pfeMAME source code from being run under Linux using ‘python3 pfemame.py’. The code that determines the directory holding the pfeMAME.py file wasn’t working right when run from the command line.
* Changed most of the text showing the word ‘Game’ with ‘ROM’ as it more accurately reflects the modern state of MAME – its not just for games anymore.
* Adjusted some text spacing in Tile View
* Improvement to the code that reads / updates various DAT files
* Added CTRL-P keyboard shortcut to open default preferences
* Added CTRL-SHIFT-P to open rom specific preferences
* Made the dynamic search code slightly more pythonic
* Added a prompt to check the Inputs file path setting if one has not been set for recording / playback of program inputs.
* Improved method of checking for a video file existing for the selected rom – faster now as well.
* Make the code for checking for program images more pythonic.
* Fixed the right click context menu not showing help string in Windows. Still not working in the submenu at this stage – working on it.

**v2.13**

* In filepaths.py, when adding a new file path (for items that can have multiple file paths), when you click on ‘Add’, the new path is added underneath now with the text ‘Double Click To Set’. Before there was no text until you saved and reloaded.
* Fixed a bug that only became evident with the latest version of wxPython (4.1.0). When adding buttons to horizontal sizers I was including an align right flag. This is not allowed and now causes c++ core errors.
* Fixed a program that has only appeared with the latest version of python (3.8.2) on my system, but may have been present on other systems in different locations. Python stopped recognising my locale when calling datetime. I have now imported locale and forced python to set to the system locale. Has fixed on my system so hopefully doesn’t cause issues on others.
* Changed the vertical size of the video play / caption button, and the messages ‘Clear’ button. Was getting cropped slightly.
* Fixed a bug causing serious issues under Linux and sometimes glitches under Windows. I had moved the software update check to a separate thread so that it didn’t lock up the main thread. Within the new thread it was updating graphical elements in a way that was not thread safe. I have added wx.CallAfter to all element updates. Seems to have fixed the issues. Surprised nobody raised this as an issue under Linux.
* Have changed some test references from ‘Game’ to ‘Romset’ as its more appropriate given that systems roms are not all games. Still need to do more in this area.
* Replaced the built in aboutbox with my own custom window. The built-in didn’t allow me to change the background color to the application wide background color. Also didn’t handle links in a way I wanted.
* Added PEP compliant docstring to pfeMAME.py. Also added \_\_xx\_\_ header information.
* Software update check status is now displayed in the messages window.
* Fixed a bug that would cause the incorrect ‘assumption’ of the current working directory when first loading and checking for the games database. I found if I ran from Visual Studio Code and the development folder was not the same as the source code folder, then the current working directory was incorrect. Had to fix it for console as well as frozen builds as the method to find the path to the executed file is different.
* Further improvements to how folders are detected and also forced the ini path to the same folder as the rest of the MAME folders (roms etc) – this was required for Linux as I have to override the default /etc/mame/mame.ini settings otherwise everything basically has to be in $HOME/.mame otherwise things just don’t work right. This was likely stopping some Linux installs from working without a bit of manual tweaking.
* Fixed bug where I was using relative instead of absolute path to write to the temp games database. Caused issues in Tileview under linux.

**v2.12**

* Moved the software update checking code into a separate thread so that it doesn’t block the main GUI thread.
* Added an automatic check for updates on startup.
* Added a capture for exceptions form URLLIB when checking for software updates. This may happen if there is no internet connection. This now prevents a crash.
* Major re-write of the history.dat file parser. Generally, a lot faster, especially if no match is found.
* Found that I was calling build\_categories twice on startup if a full database refresh was needed as I wasn’t using a global to store the game categories dictionary.
* Re-write of the game category parsing function – much more pythonic now.
* Further speed-ups to the search string processing
* Moved the Messages window ‘Clear’ button to be adjacent to the ‘Messages:’ text instead of the messages window itself. The layout looks better this way.
* Amended the build\_one\_button function to include a size component
* Tileview speed improvement; Toggling between favorites and all games is now faster. I have filtered the programs list on startup in tileview so that only programs that actually exist are in the list. Then I have improved the parsing code speed quite substantially (from average of 0.16s per toggle to 0.006s).
* Tileview was reporting the wrong number of programs in the current view as it was calculated based on the padded size, not the real size (the last row is padded to ensure the whole row is populated).
* When building the database, I no longer check for valid file extensions for systems, and instead just add any files that are found in that systems folder. The reason is that for systems with expansion card options that add additional media types (e.g. external floppy drive), I am unable to determine if the file extension is valid or not without going through extreme hoops that would slow everything down to a crawl.
* When manually selecting paths for files (ini, exe, dat) in Filepaths, the wildcard is automatically set to only allow files of that specific type to be seen. Under Linux this isn’t done for executables as they don’t have a file extension.
* Fixed bug in XML info window that would throw an error if you looked for XML info for the top level system program.
* XML info now informs you when a system program has no XML info.
* Layout improvements to the ROM info window – added some cell padding and some underlining
* Layout improvements to XML info window
* Converted a number of function calls to return dictionaries rather than multiple values. This has reduced some multiple calls and overheads and is also a more pythonic method of returning an object rather than multiple values.
* Added a preferences option to allow the OSD UI toggle key to be changed for systems. Most common for Windows PCs is SCRLOCK or INSERT. I cant test the other two options are they are specific to MAC computers and I don’t have one.
* Fixed a bug in Check All that would cause the ‘This could take a while’ message to disappear if you changed filter settings before running the scan.
* Shaved about 0.5s off the application startup time (on a fairly fast machine) by using freeze / thaw when building the game categories for display. Seems dynamically adding graphical components in a loop has overheads due to having to update the window each time.

**v2.11**

* Fixed a bug whereby hand held games consoles were not showing their programs. I needed to add a ‘Heldheld’ category to the systems list.
* Updated the systems list toinclude a few extra device types
* Fixed a bug introduced in 2.10 that prevented the search function from working after a database refresh.
* Added a menu item and right click menu item to allow INI files to be reset. If an INI is found, it will be deleted. This is useful for situations where INI files have been corrupted.
* Fixed a bug where an error would be thrown if the history.dat file was not found where it was expected.
* Added test in the Game History window telling you if the history.dat file wasn’t found.
* Added better error reporting for the catver.ini file. Now has separate message depending upon issue with file, file not found, or file permission issues. I was seeing a parsing error message when in fact the file was not where it was expected. Sent me on the wrong path for debug.
* Minor code improvement to history.dat file parsing.
* Improved the check in preferences for the games cache days check – make sure it’s a number between 1 and 365.
* In slot configs, have made sure that you don’t have empty columns where they are not allowed.
* Slight speed up to the slot configs set to empty function. It was re-reading slot information from XML file when this had already been done when the preferences window opens so doesn’t need to be done again.
* Converted some duplicated code in filepaths to a function.
* Image scaling in TileView has been converted to integer values. Previously they were floats but this has been depreciated as scaling must muse an integer.
* Changed the operation of the video player. When a video is playing you can now stop it by clicking on the ‘playing Video – Click to Stop’ button under the video. You can then click on the same button to start it again.
* Fixed a bug that I found when testing MAME 0.216. Slot configurations for plusD and Disciple interfaces were not using the right sub slot controls. I wasn’t indexing based on the parent slot name. This like affected many other devices.
* Removed unwanted extra linefeeds between the saved results for checkall roms and increased the delimiter length
* Added a work-around to how MAME is behaving with slot devices. If slot devices are defined in the program INI file (e.g. spectrum.ini), if you try running the spectrum program from the command line, MAME is ignoring the slot devices in the INI file. This then causes MAME to generate errors about unknown media types for slot devices that add new media types (e.g. an expansion floppy disk for a base system that only has cassette). I have raised this as an issue as I think it’s the INI file not being fully parsed before the media types are checked. Not sure if I will be listened to. Issue exists as of MAME 0.216.
* Removed the check for valid media types when building the games database as this will miss valid media types for systems that have expansion cards as then the database is built, pfeMAME doesn’t know that the optional expansion card is fitted.
* Forced a delete of the backup top level system INI file before it tries to rename it to ‘\_backup’ when running system program files. If anything had previously gone wrong and the old backup file had not been removed, a fileoverwrite error would occur.
* Fixed a bug I probably introduced quite a while back when I changed the order of items in the games list. It stopped me from being able to run alternate game engines for the SEGA roms.
* Fixed a bug that would cause the loss of a top-level system INI file when running one of its programs that didn’t have their own specific INI file.
* Have added OS X under platform check. If its detected as OS X, the system default behaviour will be as per Linux. This is just my best guess. I don’t have an OS X install to try pfeMAME under so I have no idea how it will behave. I’m sure there will be some platform specific issues.

**v2.10**

* Fixed a Windows issue with frozen (distribution) builds. cx\_Freeze isn’t bundling necessary DLLs that are required by the urllib for reading https URLs. This caused the ‘check for updates’ feature to not work. In the background it was actually throwing an error that you couldn’t see. The issue doesn’t exist when running pfeMAME directly from python or within an IDE as the DLLs exist in the native python distribution. I have updated the setup.py builder to search for and include these files.
* Added a note in the manual regarding the need to manually install pypubsub as the build-in has been depreciated. I have also removed my hooks to support the built-in.
* Fixed bug in database build – under Windows if the full system support method was selected, the process would spawn command prompt windows for each subprocess. Very ugly. Had hardcoded the shell type in the subprocess calls when it should be the shell\_to\_use variable depending upon operating system type.
* Found problems sometimes if I close the application when a video is being played. Have added a video\_stop() to the on\_close just to try and prevent this. Will still have to keep an eye on it though.
* When the preference to auto-play videos was not set, manually playing a video by clicking on the ‘Video Available’ button didn’t start the video.
* Changed the caption ‘Video Available’ to ‘Video Available – Click to Play’ just to make it more obvious what you’re actually supposed to do.
* Fixed a bug that would prevent a game vide from being found if any of its file name characters were upper-case.
* Changed the search control to an actual wx.searchctrl. Looks nicer and makes it more obvious that its for searching. Also removed the ‘search’ text in front of the control as it has its own ‘search’ text.
* As of MAME 0.212, the command line options to enable / disable all of the artwork types have been removed – by default they were ON anyway. I have removed these options from the Preferences window. Now, if you have an artwork folder defined, and valid files are there for the rom you are running, the artwork will be displayed.
* If information for system programs is available in the history.dat file, it will be displayed rather than the information for the system itself.
* Changed the history viewing code to use a list rather than string. This is required in order to ensure we can find all of the different match variants, especially with the inclusion of systems in the history.dat file. The downside is that its quite a bit slower. Will try to optimize it using multi-core maybe.
* Have given the games list the focus after loading. This means that the last selected game is highlighted correctly and much more obvious. Also means that the cursor key navigation works without first having to click in the games list.
* Have given the games list focus after a database refresh
* Added an option to the file paths window to allow a single path to be cleared (made empty).
* Have added filters for the ‘check all’ window.
* Added some additional error checking and feedback in the slots configuration in preferences. Found that some INI files created with an earlier version of preferences could cause it to lock up.
* Updated the manual – mostly just updated screenshots
* Removed a pile of unnecessary local variables in preferences
* Fixed some syntax problems with puthon3.8. You can no longer use ‘in not’ and must instead use ‘!=’.
* Updated the manual to include more detail on how to install cx\_Freeze as it can be quite tricky.

**v2.9**

* PLEASE CHECK YOUR FILE PATHS WINDOW – have added systems programs which will be empty by default. Point this to where you keep systems programs (should be the MAME roms folder by default anyway)
* As of this version you need a minimum of MAME 0.206 – about time you updated anyway
* Found that on startup, on\_filter\_game\_list was being called twice. Once after setting the games view filter combobox based on preferences, and then once again after loading the last saved state of the last selected game (if that’s an option). Have combined the two so it only gets read once.
* Added the LRU cache lookup to the functions.py file get\_system\_media\_types function. If this function is called multiple times for systems – say you are playing a number of games from the same system type, then the output of the function each time will be the same so the result can just be pulled from the cache. This creates a small speed improvement but is more as a beginning for me to look into better optimisations elsewhere.
* Added the application startup time to the startup complete log file message. Also added it to the application messages window.
* Found that on startup on\_filter\_game\_list was being called twice. This was due to the build database calling it and then it being called again after setting the filter view. Have modified the build database code to not call it during startup. Also shaves a few hundred milliseconds off the startup time although you will probably only notice it on slower machines.
* A big change to the Check All Games feature. The main checking code has been moved to a separate thread. This means it no longer slows down the main window operation and can be safely run in the background – even while running games.
* For game specific preferences for systems, the system program name will be displayed in the title bar if one exists.
* Fixed a bug in game specific preferences – it was trying to detect and save slots information for roms that are not systems – I don’t think they have slots and it was causing corruption and some GTK error messages under Linux. Have added a detection ‘is\_a\_system’ as a variable.
* Fixed a bug in game specific preferences that would cause a GTK error and some on-screen corruption if no slots information existed in the game specific INI file for systems.
* Added a preference ‘use multi-processor’. This uses a multi-processor / multi-core enabled method when building the games database. Each MAME rom name is sent to a separate process to be acted upon. The maximum number of processes is limited to the number of cores – 1 (So that we leave something left for the GUI). This is hugely faster when you are using the ‘Build database with full systems support method’ below. Other times it is at least as fast, but maybe not if you only have one core due the multi-processor code overheads.
* Added a preference ‘Build database with full systems support method’. This expands upon the current system support method in that it looks at the XML data for EVERY system type that you have – it looks up the folder name that the system files should be stored in based on the XML file (from the MAME HASH folder) and then adds them to the database list. This is the most thorough way of doing this but it is slower, even with multi-processor support. On my machine it takes 95 seconds versus 20 seconds if multi-process is enabled (470 seconds if you don’t enable multi core so please don’t do this).
* Systems programs will only be added to the roms list if the rom for the system exists. No point showing programs that you can’t run. This makes the full version of the systems support magnitudes faster than it would be otherwise.
* Re-arranged some items in the preferences window – have grouped them in a more logical order. Probably still needs a bit of work though.
* Added a right-click menu item ‘Analyse XML info. For systems programs with a short name (i.e. the system program exists in that systems software list XML data) – this allows you to look at the expected file size as to what it actually is. For short names, all of these must match so this is an easy way to see what’s wrong with your file.
* Fixed the scrolledwindow background colour. It now follows the system wide background preference setting.
* Added the pfeMAME icon to the scrolled window (log file and controls viewing)
* Removed the separate colours window and merged the controls into the main preferences window.
* Updated the manual for the latest changes.
* Fixed a bug that was preventing Spectrum +3 (specpls3) disk images from loading using the +3 built in Loader feature after running MAME with the program. The media type helper function was missing the first floppy drive (flop1) and only finding the second one. In the case of the spectrum, the Loader feature will only work with Floppy drive 1. This will likely have affected other systems also where the program would be associated with the second drive instead of the first one.
* Have added a Software Programs folder path back into the file paths window. Using auto find it will default to the same as the roms folder. This is because MME seems to have it hard coded that the systems programs have to be in there. Don’t know why they force that upon us as its bad housekeeping. I’m hoping for a future change.
* Added the -nodtd switch to the listxml and listsoftware commands. Helps speed up further the games database build, especially with multiprocessor enabled.
* Removed the hardcoded -offscreen\_reload and moved it to the preferences window
* Complete redesign of the file paths window. Have moved to using a table to store the file path information. Looks a lot nicer and easier to manage. The new version supports multiple paths for some items (e.g. ROMS, CHEATS, ARTWORK). Additional paths can be added or deleted from the list provided.
* Updated pfemame.py to manage the addition of multiple paths for ROMS folders.
* Fixed some UI differences in the border spacing in windows, case of the buttons, and spacing around buttons.
* Updated the pfemame.ini file read and write code to handle corrupted INI files. If one is detected on the read it will ignore it. If detected on the write it will reset the INI file to default and then write to it. This prevents the application from crashing to the console on startup if the INI file gets corrupted (Which I managed to do).
* In Filepaths window, when you select OK, if the mame.ini file doesn’t exist where it should, it tries to create it using -createconfig. It then checks again. If it still doesn’t exist, then the operator is prompted to go look for it.
* Found that in Linux I can’t manually find the mame executable as the wildcard in the file finder is set to \*.\*. This is a major stuff up.
* Added a MAKE INI button to the filepaths window – this will attempt to create the mame.ini file in the same folder defined by the MAME executable file path. This should only be used for Windows as this folder will likely be read-only under Linux, however if you change the folder permissions then it should work. Not really the right place under Linux for the mame.ini file through.

**v2.8**

* Fixed a bug introduced back in the 2.6 version. When I cleaned up the column order in the games list, I forgot to swap the ‘sort by’ field. From then it was sorting by the rom file name, not the rom name. This kind of looked odd and made it hard to find things. Fixed it.
* Slot config headings were being drawn in 3 different locations. Moved to a function.
* Moved mylist\_filtered into a class variable. PyCharm doesn't complain any more.
* When slot controls are reloaded in preferences, the slot option drop-downs are now fully populated with permissible options for that slot name – now you don't have to toggle the slot name to get options available.
* Under slot configurations I have added a ‘Reset Slots To Default’ button and made a clear difference to resetting to empty versus resetting to default.
* Slot configurations if not found in the INI file will try to set to a default state as listed in the systems XML file. It will do its best but won’t be right for all configurations at this stage as the XML formatting for slot parents versus children (subs) is not consistent and shortcuts have been taken.
* The slot configuration heading will now tell you if the slots are based on the INI file, XML defaults, or empty. Gives you a better idea of where you’re at.
* Code improvements and speedups to slot devices in preferences. Also improved some of the display refreshing / redrawing – some were nested which was slow and ugly. Removed unnecessary variables / lists.
* Slot device names in the drop-down list are now sorted alphabetically – finally.
* Updated the manual to include the following;
  1. A more up-to-date screenshot of the game specific preferences screen
  2. Better instructions for the slot configurations
  3. Additional instructions for installing wxPython under Linux
* Fixed a bug in slot configs where some cascaded slot devices would pull out the wrong device name when looking up the XML file and fail to get the correct match. I wasn’t setting the start pointer for lookup to the line following the start of that slot option section. Usually wasn’t an issue but things like multiple devices with the same short name (e.g. cdrom) caused problems.
* Added freeze / thaw to a number of slot controls functions to stop you from seeing individual controls from being added / removed. Just happens in one quick flash now.
* When opening the checkall roms window, it no longer blocks the main window so you can run the check all roms in the background – it slows things down a bit as I have not yet implemented it in a separate thread – maybe a to-do for the future.
* Fixed a bug that I must have introduced a while ago when I moved the catlist to a dictionary – Visual pinball tables were overwriting Not Classified roms as they both had an empty key in the dictionary and you can’t have duplicates.

**v2.7**

* Greatly simplified the software version checking code -never know that you could perform tuple comparisons like that.
* Fixed a bug in check all roms that caused it to not work at all. Made a bit of an error with the last updated and never checked it.
* Major improvement in the time it takes to run the check all games test. On my test system it has gone from 235 to 34 seconds.
* A number of changes to the slot configurations in preferences as follows;
  1. Have fixed the issue in preferences where I could not use a slot name that had colons in it (e.g. isa2:fdc:fdc:0).
  2. Changed the ‘Reset Slots to Default’ to ‘Reset Slots to Empty’ in the game specific preferences. The feature doesn’t read the XML file and set the ‘default’ slot configs. I’ll try to add this feature later.
  3. Added a capture for incorrect slot configuration – at the moment it is possible to define slot configs that are not compatible – when the preferences window then tries to reload them the controls may get out of order and raise an exception. The only way forward is to delete them and try again.
  4. Added a ‘Slot Parent’ column to the slots configuration in preferences. This column will hold the value of the parent slot for cascaded (sub) slots.
  5. Cascaded slots (sub slots) now appear under their parent, rather then at the end of the list. This makes them much easier to read and understand.
  6. Fixed a bug in the slot options that would not correctly read some saved slots with cascaded slot names with more than one ‘:’ in them.
  7. Slots configurations can be changed when reloaded
  8. When slots configurations are reloaded, all available slot rows are now shown. Previously if you only used 2 slots when saving, after reloading you would only see those two slot rows. Now you can see all of them so you can make changes easier.
  9. Updated the preferences slot options to only allow options for a slot that that slot type can have. This will help to prevent setting invalid slot options.
  10. Fixed the problem with the slot options list being reversed every time you saved and reloaded it.
  11. Code refactoring in preferences (slot control creation was repeated 4 times)
  12. Made the slot option names text controls in preferences read only as they can only work that way.
  13. Fixed a bug due to the latest versions of Python returning a NoneType when I never used to get it when looking for the games categories from the catver.ini. This broke the code.
  14. Fixed a bug that would cause preferences to crash if you tried setting game specific preferences for a rom that has previously not had any slots configured. This may have only been introduced since the 2.7 branch, however I need to record it just in case it slipped in earlier.

**v2.6**

* Improved how tileview uses the onPaint event to draw the red box around the currently selected game. It now checks that valid coordinates are received before painting.
* Extended the checks that the games database has not been corrupted. Have had to use a broad exception clause as there isn't a defined exception type for dbm.err.
* When the games database is written, always create a new database (add the ‘n’ flag). This is needed to support the change above.
* Have added a feature to Tile View to allow you to change the brightness of the unselected game tiles (only the currently selected one will be bright). Pressing the ‘5’ key will scroll through different brightness levels. The brightness value is remembered and stored in the INI file. Obviously if you have selected in Preferences to not dim unselected game tiles then this will do nothing.
* Have reduced the vertical size of the view list box and the search field, as well as similar controls on the preferences window. This makes them consistent in Linux and Windows (previously Linux would default to some ugly vertical height – maybe a GTK3 thing).
* Adjusted the control height and open dialog button height in the file paths window. Now more consistent across platform and Linux no longer looks ghastly.
* Fixed a bug introduced some time ago that was incorrectly setting the width of some controls – this causes the game image to be narrower than the game information text and video playing button (on some platforms).
* Changed the default height of the video playing control – under Linux it looked ghastly.
* Added the rom name to the window title for the rom information window.
* Have added some colour to the rominfo window. Alternate row colours now follow the system wide colour settings.
* Rominfo window is now only shown after its correct size is determined. This stops the annoying drawing the wrong size then re-sizing after being displayed (only took a fraction of a second but visually distracting).
* Increased the size of the first field in the statusbar. This allows for longer tooltip messages to be displayed.
* Added a right click option to view the media types allowed for the selected system. Also added to the File menu.
* Included a script into the source distribution called ‘build\_source\_distribution.py’. This creates a distribution folder and copies all necessary source files that are required for a distribution build. You need to amend the payload section in it if you create new files or make changes.
* Have added bgfx and accel as video options to the preferences
* Moved the Click to set Favorite + Rating checkbox from MAME preferences to pfeMAME Application preferences section of the preferences window.
* Major change to how the preferences are transferred to the MAME executable. Prior to this release it has all been via command line, but this has grown so long that it is very difficult to manage. Also, any issues with the command line cause MAME to not run and no useful feedback. Have moved to storing all preferences (specific to MAME, not the frontend) in the mame.ini file. This is a much more elegant way of doing things and makes more sense. The program will now look for the mame.ini file in the default locations where it should be, and if it doesnt find one then it will create a blank one and fill it out. If you already have one, it will only amend the variables that it is using. Anything that pfeMAME frontend doesnt cover (e.g. bgfx) will be left as it is already in the mame.ini file. The first time you run this new version, you must go into file paths and preferences and check everything is correct – make any changes needed and save. Any old MAME specific preferences in the pfemame.ini file will be ignored moving forward.
* Fixed a bug that would cause an exception if you tried to run a random game on first time startup if file paths were not yet set up.
* Added file menu options to play the rom and save the control input for replaying later, or to play the rom using previously recorded inputs.
* When performing a -verifyroms, the application will also check to see if this romset should have samples and checks them out using -verifysamples.
* Added a catch for exceptions when the systems rom folder is incorrectly defined or somehow empty (which shouldn’t really happen but could in some situations)
* Moved the media types checking code into a function. This will be used in three different places.
* When building the games database, only systems programs with valid file extensions will be added to the list.
* When trying to save the file paths (clicking ok in the file paths window), it will check that the right output is obtained from mame.exe. This checks that the executable is defined correctly, but also performs a kind of ‘first time run’ on the MAME executable under Linux – this is needed to ensure that the ‘.mame’ folder is created.
* File extension checking for systems wasn’t handling extensions more than 3 characters long – that would cause two extensions to merge, thus making both unusable.
* Slot configurations in the game specific preferences for systems now support cascaded slots (only to one additional level at this stage). This allows for adding an expansion card that itself has slots (e.g. an IDE control card that has 2 drive slots that you can also add drives to).
* Added a reset button to the slot configurations – this takes them back to the default options for that system type. This is needed as you can’t undo cascaded slots.
* Added column headers to the slot configs on game specific preferences.
* Cascaded (sub) slots have the slot name indented slightly to make it obvious that this row is a sub slot.
* Simplified the play video code a bit.
* Have converted the game category list to a dictionary after loading it and then used that dictionary when building the games database. More efficient and slightly faster.
* Finally got around to correcting the column order in the games database. From day 1 the first two columns have been reversed, and my code has compensated for this – was easy when I only had two columns but now it was getting very confusing and difficult to manage. You will have to perform a full games database refresh to get everything looking right the first time you run this release if upgrading from a previous version.
* Game video is now stopped during database refresh and when deleting games from the favorites list
* Moved the game video stopping to a function as it is used in a number of places. Made it also check if a video was actually available and display the right type of caption.
* Improvements to how the number of runs is updated from tileview

**v2.5**

* Fixed a bug introduced around the time of 2.4 (due to versions in python I think) that caused the ‘Check for Updates’ process to fail. Was not expecting a byte formatted string.
* Added the rom name to the game specific preferences window title bar.
* Changed a few more string concatenations to ‘f’ strings
* Fixed a bug introduced a while back that padded text in columns in the games list with blank icons. This should only happen in the first column. The result looked odd but took ages for me to spot.
* Have padded out the tick & cross images so that they take up the empty space in the columns they are used in. This is the only way I have been able to center the images – trying to do it in code fails even though it’s supposed to work.
* Have changed the font size of the game categories check boxes. They were a bit large and not matched with the rest of the windows font sizes.
* Moved some font definitions into variables – will make it easier to make global changes in future.
* Simplified the MAME version finding code
* Fixed a bug whereby the Normal View and Tile View Snapshot image would not correctly handle the existence of a system program with a short name. They now look for the short name first, then the long name if the short doesn't exist.
* Made some improvements to the Snapshot image handling in Tileview – code cleanups, removed redundant code.
* Fixed an annoying problem where the logfile viewer wouldn't correctly scroll to the bottom to show the last log entry – worked ok on Windows but not Linux. Switched from using wx.textctrl to richtextctrl
* Changed the scrolled window viewer (logfile & controls) so that for log files it scrolls to the bottom but for controls it stays at the top
* Tidied up the pfeMAME manual opening code
* Re-wrote some of the drop-down menu code – made it easier to read and easier to change / expand later
* Fixed the annoying screen flash when running games in TileView in Linux. I had previously put in a turn off maximise then turn on maximise to fix an issue I had quite a while back where it wouldn't be fullscreen after playing a game. This doesn't seem to be necessary now.
* Massive reduction in database building time – as catlist is used to build the games list an the catlist, when I find a match with mame -ll, I remove that entry from the catlist temporary search list so that next time the list is one line shorter.
* Further general code improvements
* Substantial speed improvement to the get\_exists part of the database building.
* Fixed a bug that had broken the saving of the number of runs.
* Some code improvements in the checkall module – just better ways of doing things.
* Fixed a but that would prevent systems programs from running in tile mode when you toggled the view mode.
* Removed the debug window from Tile View – things work well enough now that I dont need that additional code overhead.
* In Tile View, moved the categories file loading to the start of code – it was meing called every time the games list was rebuilt which happens once when first run, then every time you toggle the view mode.
* Simplifications to toolbar menu code
* Simplifications to toolbar icons code
* Simplified context menu helpstring helper code
* Don’t allow the right click menu from appearing when clicking over empty space in the games list
* Have moved the right click menu into its own class. More Pythonic way of doing things.
* Rerwrite of some of the sizer and panels position code in TileView. Have increased the size of the scrolling games list for better readability and fixed an issue where the game name length wasn’t correct the first time it was called, then too long after the first refresh. I wasn’t really doing things correctly with only using 1 panel and having everything else drawn to the frame. Now uses two child panels with a parent panel underneath.

**v2.4**

* Fixed the formatting of the first time run message box. The text is now laid out properly.
* Corrected the status bar message when the first time run is detected.
* Changed the way the last game selected recall is done. It will now recall to any last game selected, even if it was found during a search. Previously it would not recall a selected game found using search.
* Have stopped the video player from trying to play a video when the type ahead search is used. When you start searching, the current rom name is lost and the video player code will generate an iteration error.
* Added a menu option under the Help menu to check for pfeMAME software updates.
* Added a blank (empty) selection as the first item in the game specific preferences slots options so that you can remove a slot configuration.
* Implemented a faster search method for last rom selected on startup – the previous version introduced in v2.3 was a quick hack but slow.
* Started replacing the discrete lines for 2-dimensional list searching to a def in functions. I don’t yet understand why, but calling a def with this search function is faster, especially if its in a nested loop (heaps faster it seems).
* Substantial speed improvement in refreshing games database if you have a lot of systems (can be up to twice as fast). My home development system wen to from 38 to 21 seconds total refresh time. A slower machine went from 23 to 13 seconds (It had a lot less roms & systems on it).
* Fixed a bug that was introduced some time ago that stopped missing games from being highlighted (If that option is selected).
* Added a new child window under the ‘Edit’ menu called Application Colors. This allows you to adjust some of the application object colors. This is an early work in progress as I have to balance the ability to customise the application against the applications ability to adjust to the operating systems theme. To make the background and text colors the same as the operating system theme, select ‘Default’.
* Capture the click on the URL at the end of the game information. The URL is loaded when left clicked in the systems default web browser.
* Changed the encoding when reading the history.dat file into memory. The problem is that this file doesn’t have just one encoding type and python requires you to be very specific. I have reverted to encoding to uft-8 and have told it to ignore any characters that don’t comply. This means anything that’s not utf-8 will just be omitted. This fixes some very odd characters from being displayed.
* More output formatting fixes in checkallgames. Display improved. Save option now strips out any HTML tag characters as well.
* Added a Stop button to the Check All Games window. As this process takes quite a long time, you can now stop it at any time and save what you have so far or just exit. Before, if you tried exiting before the process was complete, the application could lock up.
* Fixed some layout issues in the preferences window (extra blank line above the Video option due to incorrect number of cells defined for gridsizer).
* Fixed some minor text alignment issues in the rom information window
* Fixed issue in rungame where various message for the log file were never getting saved or sent to the messages window
* Removed a lot of unnecessary globals in tileview
* Cleaned up a bunch of tileview code
* Moved string concatenation formatting to f-strings
* Got rid of the splash screen – unnecessary and just downright annoying.
* Updated the rungame module to run in a separate thread. This means that a running MAME game no longer blocks the mainloop, i.e. you can continue using pfeMAME while the MAME game runs in the background. You can even open multiple MAME games. Each one will send back output to the messages window when completed (or if it’s trying to issue messages to the console).
* Fixed incorrect layout of tileview – mostly the scrolling games list in the RH panel. Sizes were being calculated before the layout was complete. Added wx.CallAfter to ensure the \_\_init\_\_ is completed before determining sizes.
* Fixed a long-standing bug that would cause the scrolling game list in TileView to get out of sync if you scrolled down to a line that was only partially populated by games. Have padded out the list to fix this and have padded out partial lines with blank images. You can’t actually scroll over the blank images though.
* Added a preference to enable / disable the ability to click in the favorites & ratings columns to set those values rather than using the right click menu. I think some people will like it, and others will hate it.
* Significant code refactoring in the rungame file. This has simplified the code and, in some places, will make it easier to extend the features in the future.
* Fixed some bogus code in the INI file parsing code in favorites.py
* Fixed a bug that would throw an error if you clicked on the games list during database refresh. Have disabled the control until a refresh is completed.
* Fixed a bug that caused incorrect linefeed characters to be inserted in messages received from Visual Pinball.
* Rewrote the code stripping and correctly formatting linefeeds from the subprocess.Popen – it wasn’t working 100% and was messy.
* Fixed an issue where a visual pinball table wouldn’t be shown if the file extension was not lower case.
* Updated the manual to use cx\_Freeze for Linux deployments. This now works for both Windows and Linux.

**v2.3**

* The game categories selection has been moved to the main window so you don't have to open the separate window to select / change them. When you change a game category selection the games list is updated automatically. Double-clicking on the category list window will toggle all / none checked.
* Fixed a bug introduced in v2.2 that broke the rom information window. Shelltouse variable was missing.
* A few more generic code improvements.
* Fixed a problem that arose with a new build of Linux (Ubuntu). The loadhistorydat code was crashing with a unicode error. Had to change utf-8 encoding to ISO-8859-1. I’m not really sure why as it had worked ok old slightly older builds and on Windows.
* Fixed an issue created by v0.195 mame changing the output of the -ll (listfull). Seems they have decided to include all devices as well as roms.
* Fixed a bug in the Rom Information code. Under certain versions of linux it was crashing due to the differences in CRLF between platforms.
* Removed a number of unnecessary try/excepts
* Improved the add favorites code. Got rid of the unnecessary warning that you are tying to add a favorite that already exists and simplified the method for preventing duplicates.
* Double clicking on a games favorites flag will add that game to the favorites list. If it already exists as a favorite, it will delete from the favorites list.
* Double clicking on a games rating box will allow you to set that games rating
* Removed the need to be showing the favorites ‘view’ in order to delete a game from the favorite list
* Adjusted the size of the View drop down box as in Linux it wasn’t showing all of the text for the options.
* Removed unnecessary line feeds from the game specific preferences file creation – it was causing extra blank lines in the file.
* Updated the game specific ini file code. It no longer supports manual preferences to be put into these files. The file will now be overwritten if you select game specific preferences. This was done for consistency and to also set up for supporting slot settings which will be added into this code.
* Changed the combo box lengths in the preferences window as they were too short for the displayed text in some Linux builds (Note, Linux doesn’t always correctly auto size these critters).
* Added slot device configuration to game specific preferences. This works for systems (e.g. ZX Spectrum), but also the system programs. So you can have one program (e.g. firefly) set up with a kempston joystick and another one (e.g. jetpack) set up with the Interface 1. Its not 100% perfect due to the way MAME provides information about slots. For some systems the information is incomplete so pfeMAME can only go off what is available. Cascading slots seem to be the current issue. I have used the -listslots command. The necessary data is in the -listxml, but it is quite complex so I will come back to this later as an improvement.
* Added video auto load, show the last selected game, and systems support to the default preferences (first time run) and made them True (Selected by default)
* Added a manual preferences input box into the game preferences window. This allows you to add additional game specific preferences that are not explicitly covered by pfeMAME (There are a few, including bgfx, hlsl etc).
* Fixed a bug that raised an exception if you right clicked on a blank area of the games list (e.g. if you are in favorites view and you only have a few favorites so the area below is blank).

**v2.2**

* Added a size to the drop down options in the preferences window as under Linux some of them were not readable (Linux doesn’t size them automatically)
* Adjusted the size of the mame version and database age text as under Linux they were wrapping (need to investigate why I have to specify a size)
* Added a proper capture for the CTFL-F to search for roms. In some situations, the existing CTRL-F hook which was based upon the menubar just wouldn’t work.
* Made the 5 second video auto-play message actually count down (5-4-3-2-1)
* Removed the check that a game had been running for more than a second before it tried updating the number of runs. It never worked correctly in Windows and was also disabled in some other instances. Now regardless of holw long you let a game run, the number of runs count is incremented.
* Fixed a bug where if the mylist\_temp.cfg database was deleted the application would crash as it couldn’t handle a FileNotFoundError exception. Now if the file is not found it will rebuild from scratch.
* Sorted out (finally) the layout issues with the scrolled panels on the gametypes and file paths windows. They lay out the same across Windows & Linux now and the vertical scrollbar is in the right place under Linux and also there is no longer the ugly horizontal scroll bar that was not needed.
* Ensured that the game video timer restarts at 5 seconds if you click away and back again
* Changed where the game video timer is stopped in order to prevent a pop-up error message every 5 seconds if its unable to play a video.
* When game videos finish playing, the static game image will be displayed.
* Improvement to splash screen image to make it stand out more
* Added a border to the ‘pfe’ part of the icon so that on dark desktops you can still see it
* Removed the need for the blank icon – used the wxpython built in image creator
* Fixed an alignment issue with the game icons
* General corrections to comments in the source code
* Fixed a bug that was causing the ‘Unknown file extension’ message in the messages window to be displayed as ‘known file extension’. Was trimming the 2 leading characters in the same code that removed the b’ binary indicator from subprocess calls.
* Added a ‘Play Random Game’ function and an additional icon. This will randomly select a game that is listed as existing (however it won’t know if the game is flagged as actually working or not)
* Replaced io.open for the history.dat file with the built-in ‘open’ as under python3 one is an alias for the other. Also makes all my file opens the same. Also means I can get rid of the import io statement.
* Improvements in the buildgamelist code. Better laid out, reads better. Also, slight improvement in speed on slower systems when refreshing the database.
* Limited the game refresh time display in the messages window to 2 decimal places
* Fixed a problem where games that were not listed in the catver.ini file would not be displayed. There is now a ‘Not Classified’ game category that is hard coded in. As long as its ticked, you’ll be able to see those games (and search for Not Classified).
* Changed the numbering system again. Got rid of the last digit – my revision control doesn’t need to be that complex.
* Some fixes to the checkallgames save file
* Fixed / improved a number of file open r/w code sections – was not following best python practices
* Stopped double delimiters from being displayed in the messages window when there is no output from MAME
* Moved the code that displays the number of roms, available, and in view into a function as the same code was repeated 5 times
* Changed ‘Available’ to ‘Exists’ in the statusbar and also changed ‘In View’ to ‘In Current View’
* After changing game categories, the number of roms, available etc is updated
* Stopped the game video player when the tile view is called as well as when a game is run
* Merged the two lists that were being used to auto-locate all files/folders in the folderpaths window and cleaned up some unnecessary local variable passing and shadowing.
* Tileview code clean-ups, better definition of exception clauses, cleaned up some variable name shadowing
* Tileview Bug fix: System support was totally broken. Forgot to hardcode the system folder to roms folder (was still using old systems folder). Was likely causing crash on new installs.
* Rungame general code improvements and tidy ups
* Re-wrote the enginetouse code in rungame.py as the original version was a quick hack.
* Visual Pinball tables had disappeared from the games list due to the same problem that prevented Not Classified game types. Had to add ‘Pinball’ to the games list (you need to make sure its checked).
* Found that the pfeMAME.ico icon file was possibly not properly formed. While trying to get WindowsXP support working with wxpython phoenix I found that the current icon didn’t work. Comparing to a previous icon size they were different. Have reverted to the previous one (looks the same to me).
* Got pfeMAME working on WindowsXP with wxpython phoenix. You have to use python 3.4. Other than that, everything seems to work for now. This may change with future wxpython releases as I don’t know how long they will support older python versions.
* System short names were no longer being displayed. Seems that it happened way back when I moved to python 3.x but I never noticed it.
* Added additional game engine support (SM2/3 & Visual Pinball) to TileView. You will need a mouse!!
* When database refresh is performed and errors are found in the dat files, a single line for each dat file is now created with the quantity of errors found, rather than a separate error message in the messages window for each error found.
* Replaced the configparser based code in the gametypes file and the main pfemame.py with my own simpler version – is a lot faster to open the window now and the code is cleaner. Some memory use improvements also.
* Removed the global variables gamecatName and gamecatCat and passed them in the functions instead. This frees up memory and reduces unnecessary code.
* Replaced the ScrolledMessageDialog that was being used to display the logfile with my own window. Don't know why but the built in one was taking 6 seconds to get the data into the window and display it. Also used the same for the controls window as I like mine better.
* More code refactoring in filepath.py – man I did a few things the hard way originally.
* Some code cleanups (mostly nested Ifs) in preferences.py
* Fixed a problem that prevented systems roms from running if their folder name ends in \_cass, \_flop, \_cart (You could still run them if you right clicked and selected to override the HASH support).
* Lots of code clean-ups in Tileview
* Tileview now starts up a lot faster
* Fixed another issue with the way that systems short names work. Looking at alll HASH files that currently exist, there are many different folder suffixes that I would need to support (e.g. CASS, FLOP, etc). When building the games database, if a rom is identified as a system, the code now looks for any folder starting with the system name (e.g. spectrum) and then searches through that folder (e.g. spectrum\_cass) adding all files it finds. The problem with this method is that it doesn't solve the naming convention issue (or lack thereof) that exists in MAME at the moment – for example, in the case of the archimedes systems, there is just one hash file called archimedes.xml that covers all of the different system types. The ONLY way I have found to manage this would be so slow that it takes about 15 minutes to build the games database on a very fast PC – just not practical. For now you can get around it by creating an alias to the folder (rather than duplicating each different name that you need.
* Message window and log file will now record if a system has been run using the short name or long name. Will help with debug.
* Fixed a problem that I think crept in with the move to wxPython Phoenix whereby the tile view wasn’t exactly full screen on some systems.
* Seems the about box was never pointing to the correct icon file.
* Additional exception tracking if file paths are incorrect for various files that are being opened.
* The messages box timestamp was formatted to gmtime, not local. Oops.
* Added a check that a MAME rom actually exists before trying to run it.
* Tidied up the backup & restore code
* Re-wrote the right-click context menu for the games list. It also now displays the help text in the status bar.
* pfeMAME icon was garbage in Linux – top level image was compressed which is not supported.
* A large number of general code cleanups and code refactorings
* Have switched to using the built in function ‘shelve’ to store the cached version of the games list. I was previously storing as ASCII text using file read and write functions as well as the necessary code overhead. This is way more simple and I never have to touch it again.
* Big update to ‘check all games’. You can now see the scroll bar as each game is checked using -verifyroms. All games in your library with an ‘exists’ flag are checked so if you have a very large library then expect this process to take a very long time. Also added some color to the output window.

**v2.1.0**

* Reinstated the pfeMAME.png file in the root pfeMAME folder. While it’s not used by any part of the code, in Linux it can be used to create a desktop icon and in some distributions, they will automatically see that icon for the desktop.
* Updated the code that tries to open the pfeMAME manual. It needed to check first for the existence of the pfeMAME manual file itself before trying to open it.
* Fixed the problem with the manual not always opening under Linux builds – depended upon the folder name – if it included a space (like the Builds do) then it wouldn’t work. Contrary to what everyone says, I did have to encapsulate the path in quotes.
* Added a method to make a game a favorite by clicking in the favorite column for that game. Did the same with star rating. *UPDATE – Have actually commented this out in the source code as it was causing an unwanted problem when you use the cursor keys to scroll and the mouse pointer was hovering over the rating or favorite column (every game was then selected as a favorite). Will come back to this later.*
* Added a method to reset the game emulation engine back to the default MAME for any games that have been selected to use a different engine (e.g. sega model2/3)
* Adjusted the panel size for the MAME controls popup window so that it displays more detail and fits better on the screen
* Fixed a bug in the filepaths.py window sizing that was causing very different size issues across different platforms and even within different display managers under Linux.
* Suppressed the option to play games with different emulation engines under Linux as there is no support for the model2/3 engines that I am aware of.
* In tile view I have changed some of the textctrl objects to statictext as under newer versions of Linux I was having problems with borders being displayed
* Fixed a bug that was creating duplicate entries in the runs.dat file once the number of runs exceeded 9. The method I was using to locate an entry in the file was incorrect and didn't allow for numbers above 9 (Due to 10 being 2 digits not 1).
* Added some highlighting of the messages and game information headings to make them a bit prettier
* Removed the unnecessary print() commands in the open manual code – what were they doing there I don't know.
* Added gio open as a method to open the pfeMAME manual as under Ubuntu 17.10 the old gnome-open has been depreciated and should no longer be used
* Finally fixed the longstanding indexerror exception raised by the ultimatelistcontrol object in Linux only when trying to move or drag the pointer over the header. Never found out why so have just implemented an override for the GetItem function and added a capture for the exception so I no longer get an error.
* If systems support is not enabled, do not look for systems at all when building the database from scratch. This can speed things up on slower systems.
* If in preferences a setting has been changed that requires a restart of pfeMAME, you will now be notified by a pop-up-prompt when the preferences window is closed.
* Added a game video play function. A new folder ‘video’ is available in file paths. If a video is available for the currently selected game, a caption under the game image will say that a video is available. If in preferences you have selected to auto play game videos, then the message will say that the video is auto loading in 5 seconds, after which it will automatically play.
* Replaced the message window delimiter with a global one (also made it longer)
* Added date/time stamp to the messages window
* Removed the listverticalsize variable as it was static and not needed
* Removed the preference to scale game icons as if you dont select this and you have a very large icon library like me, the application will essentially stall on startup as it will take forever to read each icon to find its size in order to set the maximum row size.
* Implemented a preference for game icon scaling size as a fixed value of 16, 24, or 32 pixels. 16 is default but 24 kind of looks nice.

**v2.0.0 Build 1**

* Updated version control to something more standard MAJOR.MINOR.REVISION BLUILD
* Made the right click menu item ‘Play game without softlist hash support’ only enabled if the program selected actually has a short name.
* Added right click menu items to run Sega model 2/3 games using a different emulation engine. I’ve done this as I quite like some of those games (Daytona), but MAME currently doesn’t emulate them very well, if at all.
* Fixed bug in dynamic search – when setting current row to the first row (0), it needs to ensure there are actually results otherwise the ultimatelistctrl throws an error.
* Visual Pinball tables support has now been added in properly. When you refresh the database, if you have set a path to the visual pinball tables and the executable file, then the tables will be added to the games list under ‘Visual Pinball’ – just search for that. The number of runs, favorites, etc work as per normal. You will have to copy game snaps into the snapshots folder yourself though.
* Removed the old ‘temporary’ method for playing Visual Pinball tables
* The manual has been updated to include the new alternate emulation engines
* Removed some more unnecessary use of ‘global’
* Moved some repeated code into functions
* Updated the menubar code to be more consistent and make it easier to add icons or effects / colors at a later date
* Have moved some global variable declarations to the main module in the main application – now that I know how to use global variables properly. No change to function and personally I think it’s a bit harder to read, but hey, apparently, it’s the right way to do it. Have a lot more that will need to be corrected over time – no hurry.
* Improved output display of the rom information window – style now consistent with the rest of the application
* Added a copy to clipboard option for the rom information display window. Aligned the buttons to the right and made the text uppercase as per other windows
* When running a game using a non-MAME emulation engine (e.g. Sega Model 2/3), a note is made in a new DAT file called enginetouse.dat. After that when you select that same game it will look up what engine to use. For now, if you want to go back to using MAME you will have to delete this game entry from the enginetouse.dat file manually.
* Replaced the exists and favorites columns Yes / no text with ticks and crosses – this requires a database refresh to start working properly.
* Fixed a bug that picks up on a configparser error when reading the catver.ini file. If it finds an error you will now be told that it can’t parse the file and you should revert to a previous one. Had to do this with the catver.ini version 0.189 as it broke my parser. Change made in pfemame.py and gametypes.py.
* Added a ‘allow no value=true’ to the configparser reading the catver.ini file as 0.189 version contains some lines with no values and it broke my parser. Change made in pfemame.py and gametypes.py.
* Added ‘Computer’ as a system type. Prior to catver 0.189 it was called ‘Home Computer’
* YOU MUST OPEN GAME CATEGORIES AND SET UP YOUR GAME CATEGORIES AGAIN AND THEN REFRESH THE GAMES DATABASE WHEN UPDATING TO THIS VERSION OF PFEMAME DUE TO CATVER.INI CHANGES.
* Updated the project to work with Python 3 and wxpython phoenix. A lot of code changes to support the differences between 2.7 and 3.x. If you want to build the project yourself then you will need to move across to Python 3.x .
* Confirmed issue with Ultimatelistctrl.py library. As of wxpython phoenix a divide is treated as a float. This was causing exceptions in the right click context menu. Raised the issue on Github and a change will be made. If you are building from scratch you will need to find this line in the ultimatelistctrl.py library and add a second slash (divide) to instruct it to perform an integer divide (Note, as of 17 Sept I see that the download files are updated so you no longer have to do this manually);

current = y/self.GetLineHeight()

* Changed the Windows exe creation script and batch file to use cx\_Freeze (Was previously using py2exe) as py2exe is currently not supporting python 3.6. This change means you have to install cx\_Freeze (pip install cx\_Freeze) if building from source.
* Added a check in rungame to make sure that if we are trying to use an alternate emulation engine, that the paths to that engines executable file are actually valid.
* Removed wx.richtext import from checkall as its not been used in a long time.
* Merged icons and images folders into the images folder (removed icons folder). Moved the pfeMAME.ico file from root to the images folder.
* Replaced the old wait display with a proper splash screen – its displayed for 5 seconds on application startup
* Fixed bug in preferences.py. The code to write to game specific INI files used a binary write which worked ok in python 2.7 but is not valid in 3.x as its text I am writing.
* Removed the minimise button from sub windows (preferences etc) and set them to always on top so that you can’t have a confusing situation where the window gets minimised so you think you’re working with the main window but nothing is happening as the sub window has control.
* A whole lot of code improvements / corrections. Not bug fixes but things like removing unnecessary duplications, id’s, variables, complexity, etc)
* Added a verify rom toolbar icon
* Added an adjuster to the horizontal size of the filepaths frame and panels to try and may it lay out better in both Windows and Linix – its better but still needs some work.
* Removed the blank.png image used in Tileview with wx.Image(x,y) as this does the same thing and is built in.

**v1.008**

* Fixed a bug where if I have forgotten to delete the pfemame.ini when I am packaging up the source files, the file paths will likely be invalid – this causes a number of problems and a crash.
* Corrected the width of the game information control – it was about 2 pixels short of the RHS window border
* Made the default game image (the one displayed when there is no actual game image) transparent so that the top and bottom blend nicely into the window. The white top and bottom borders before were ugly. Had to change it from a JPG to GIF for the transparency.
* The window horizontal resize and maximize now works – the game list and associated columns grow to utilize the available space (vertical resize still does nothing – just empty space)
* When auto-finding file & folder paths, I now clear the visual pinball fields to prevent any issues when re-using old INI files.
* Fixed the annoying dead space under the buttons on the gametypes and filepaths windows under Linux. Had to put in a system specific hack as the sizers work a little differently between platforms (more likely a lack of understanding on my part on how to do this correctly).
* Fixed the message and game info box background colours in WindowsXP. XP panel colour is different that WindowsXP and Linux. The right way to make them the same is inherit from the panel colour.
* Added a Clear button next to the Messages box so that you can clear old messages – sometimes useful to know that the messages you are looking at relate to the rom just run rather than an old one.
* Implemented a vastly faster method for displaying the game information – turns out that indexing very large lists is way slower than using the find method on large strings. Now when you are scrolling through the list, the scroll speed is a lot faster and you don’t have those very slight pauses when a game doesn’t have any history info (it was having to iterate through the history.dat file all the way to the end in the previous code version before it would reaslise there is no match)
* Updated my implementation of the systems support. Now that I have a better understanding of how the softlists work in the HASH folder, I realised that my previous implementation would not let cheats work for systems. Also, MAME expects the systems programs to be in the roms folder, not another arbitrary folder. The code has been updated to now use the softlist ‘short’ name for calling the system. If it can’t find a short name match then it will revert to the previous brute-force method of calling the program. See the updated manual on how all of this works. You will notice that it takes longer to build the games database during a refresh as it has to try and find the short name for every systems program that you have.
* Force a database refresh on first time run in order to support the updated systems code
* Added an additional column for the systems short name
* Updated screen shots and manual
* Systems folder path has been hardcoded to be the same as the roms folder path (regardless of what you set it to). MAME requires that systems programs be put in the same folder as roms (its hard coded that way). If I don’t see them make this an option in the next few releases then I will remove all reference to the systems folder path.
* The manual now includes the instructions on how to build pfeMAME from source
* Moved the old startup wait dialog from the icons loading section to the main code section as it makes more sense to make it obvious to the user that the application is still loading and starting up so please wait.
* Good tidy up of the OnDatabaseRefresh code. There was code repetition that was unnecessary and ugly. Nice tidy up.
* During normal loading, after reading the temporary database into memory, it was then writing it out again when this is only necessary after a full database refresh. Fixed this.
* Main window default vertical size reduced to fit within a 768 pixel vertical screen size. This size is very common with laptops (like my one at work). You can still adjust the window size to suit.
* Major improvement to the code that is used to read/write various dat files – there was too much unnecessary code repetition and it was difficult to read.
* Changed the nesting of the sizers on the main screen. The game information box now goes to the bottom of the panel.
* Main window contents will grow vertically to take up available space when the window is resized.
* When doing a search, the first row of the results is forced to the top of the games list.
* Added the Game Category to the search field so that you can for example search for Maze games
* When right clicking a game, you can select to not use the softlist HASH file short name, and instead use the full path name. Useful as some systems short names just don’t work properly.
* Fixed a bug that would cause a crash if the path to MAME executable was invalid when refreshing the database.
* Have specified startup window location to be 0,0 regardless of running in source or as a compiled distribution. Was previously using wx.defaultposition but this didn’t have the safe result in all situations.

**v1.007**

* Fixed a problem that was causing a traceback error (non-fatal) at the console sometimes when the menubar helper was trying to display information about the menu function. Ages ago I created onHelp which was a helper function to display this information. This was when I was developing in Linux and had only just started testing in Windows. In Linux (Ubuntu actually) I couldn’t get the info text to display. The problem is that this helper was throwing console errors under windows. The original issue was actually caused by Ubuntu Unity putting the menubar at the top of the main window instead of the top of the program window. As this is really a Unity specific problem (or how wx widgets works with Unity), I am removing the helper code and going with the built-in function.
* Changed the default (1st run) video mode for Windows systems to GDI. Some very low-end systems (and emulated systems) don’t have a high enough version of D3D.
* Have reinstated the command output capture from running games – let’s see if it doesn’t lock up now.
* Fixed the column widths if systems support is not activated
* Added a few missing dlg.Destroy()s
* Further improvement to first time run message
* Implemented a very rudimentary and not well tested ability to play Visual Pinball tables. I love pinball and so this was just something I wanted to include in pfeMAME. It requires that you have already set up vPinMame and Visual Pinball to work with each other. At the moment, the table and executable locations are hard coded so this is really only for my internal purposes. This also only works under Windows installations as there is no Linux version of Visual Pinball (due to it using VB scripts) – I haven’t tried it under Wine. (!!!!! Fix this for final release of this version!!!!!!)
* Added an icon to the Quit option in the file menu
* In Tile View I have overridden the Display Debug Message preference (i.e. verbose). If this is enabled in Tile View the application locks up as soon as MAME is executed. As nothing can be displayed anyway there isn’t much point having this.
* Added a close to all subprocess.communicate methods as I think that’s the correct thing to do rather than leaving them open
* Changed the global variable mylist to a self.mylist – not for any specific reason other than I really need to get away from using global variables when I don’t need to. Over time I’ll get rid of some more as there are heaps
* Updated the pfeMAME build instructions. I have setup.bat file for windows builds and setup.sh file for Linux builds and things are a bit easier now for building pfeMAME.
* Added an ESC key event to all child windows so that you can exit them by pressing the ESC key
* When running a rom, the full rom name is displayed in the messages window before the verify is run on the rom – makes things easier to read.
* Removed the 24 pixel padding that was being added under the messages window – this created a 24 pixel sized blank void under the window which was a waste of space. Have also slightly increased the vertical size of the window to capture more information.
* Got rid of the annoying horizontal line above the toolbar icons. This was only happening in Windows and it seems that only Windows has a style option to remove this. Looks much better now.
* Changed the star ratings images – got rid of the ‘empty’ stars so you just see the gold ones. A zero star rating doesn’t show anything. A visually more appealing look.
* Added ‘Missing’ to the view combo box. This then only shows roms that are missing from your library. A nice way to spend some time looking for roms that you want to find.
* Added a preference setting that will Auto select the View and the currently selected game based upon what the View and selected game were when pfeMAME was last shut down. If this preference is selected, it overrides the default View setting.
* Event close wasn’t bound to the close application code for the main pfeMAME window.
* Changed the file paths window to be scrollable as there are too many inputs to fit into smaller resolution displays.
* Added Visual Pinball tables path and executable path to the file paths window.  These are not able to be ‘auto found’ as they sit outside the MAME ecosystem.
* Fixed a bug that would log an error in undeclared historydat variable during first time run as the path to the file doesn’t exist yet so the variable is not declared. Fixed by forcing the history.dat file to be loaded into memory after a database refresh. Also set the historydat variable to empty string if the file path is not set or found to prevent the same error if the dat file path is not specified.

**v1.006**

* In Linux (not Windows), using the ENTER key to run a game caused two event to be generated for reasons that I just could not find out. I have been able to work around this by using two different methods to capture the key press for Windows and Linux.
* Removed some empty linefeeds not needed when running verifyroms
* Have included the pfeMAME manual with the distributions now and have added a submenu under Help to view it
* General tidy up of some code – spacing, grammar, python-ish correctness etc
* Corrected some override code for the ultimatelistcontrol
* Used a better method for determining if a game has an icon or not – previous method was an ugly hack
* Have added a CLEAR ALL option to the file paths window. This is more useful for development than normal use
* Have added an AUTO FIND option to the file paths window. This is very handy as it just simplifies and speeds up the process of entering all of your folders. Under Linux only looks in \usr\games\mame – if it finds a match it adds it in, otherwise it is left empty so you know a match wasn’t found
* Changed the about dialog to use the wxpython inbuilt dialog
* Changed the singlechoice dialog to use the wxpython inbuilt dialog
* Removed my own coded singlechoicedialog and aboutbox include files from the distribution build as they are no longer used
* Changed the logfile file limit size to 512k before it creates an archive copy and then a fresh logfile.log. This was done to allow me to use the scrolledmessagedialog to display the logfile – this inbuilt control has a text size limit.
* Added a view option to display the logfile. Note, it will only display the currently active logfile (limited to 512k), not any archived log files (with a .1, .2, etc extension)
* Renamed the manual to pfeMAME\_manual (added the underscore). Linux couldn’t find the file due to the space causing issues with subprocess.popen.
* Fixed a bug that prevented the pfeMAME manual from being opened under Linux – the method to open a PDF file under Linux and Windows is different as under Linux you need to essentially tell the system to use a default opener.
* Updated the manual – brought it up to date and corrected spelling and punctuation etc.
* Removed all references to mameinfo.dat as it was not actually required or used anywhere but you still had to select it in the folder paths. Also removed as a prerequisite in the manual.
* When selecting a star rating for a rom, the pop-up will now set its default value based upon what the current star rating for the rom is
* Fixed a bug that caused some games to throw an error to the console when trying to get the game info from the mameinfo.dat file. This was due to non ascii characters being in the file. Have added encoding to utf8 when reading the file to solve the problem (this was done using io.open as the standard open in python 2.7 doesn’t support built in encoding on open).
* Fixed a bug whereby a slightly malformed game snapshot (PNG) would cause a long delay and then a core assertion error at console when the image was being resized for display. I have trapped the error and a message is displayed for the user so that they can track down the offending image file. The long pause still happens as it takes a while for python to realise there is an issue while it gets stuck trying to resize the image.
* Have improved the code for displaying game information – could still be faster.
* Adjustments to column widths in game list table
* Have improved the ‘first time use’ message a bit to make it more useful.

v1.005

* Snapshot (image) names created when running systems now use the program file name, not the top level system name for the snapshot. This now allows snapshots to be taken for system programs (Had to manually rename them up until now)
* Added file path definition for the softwarelist HASH files. This is needed for systems support. Up until now I have assumed that the default folder location was ok but this can cause issues in Linux.
* Added file path definition for the plugins files. This is needed to support LUA plugins. Up until now as I hadnt defined it you could not get the plugins to work. You must manually define a plugins.ini file and put it in the plugins folder. I may add some options for this in a later release but as the plugins can be anything you write yourself, it is pretty hard for me to define them.
* Removed unused imports
* Added ‘Find’ the edit menu and linked to CTRL-F event. This caused the search input box to take focus and it selects all text in the box at the same time. This allows for easier and faster searching.
* Carried out some improvements to code ‘correctness’
* Removed an incorrect write to the log file in the preferences.py file. The method I was using was not working.
* Corrected the method I was using to close child windows
* The issue of p.communicate() locking up after running a game in windows has come up again. I can see that other prople have similar issues and its to do with how windows closes the child when the game is executed but the p.communicate() method is still waiting for an output and locks up. This is only an issue if the game is run in fullscreen mode. If its run in a window then no issue. Until I can diagnose this further, if running in fullscreen I dont get the command output.
* Fixed bug that would cause tileview to lock up if there were no games in the favorites list. This bug was introduced in version 1.004 when I added systems to tileview. It was due to an un-initialised variable.
* Added output of platform type, python version, and wxpython version to the logfile. Can assist with debugging issues.
* Fixed a bug that prevented systems programs with a fullstop ‘.’ in the file name from being able to be run and also cant display an image snapshot
* After editing the game categories, the search input field is cleared as any searth test in the field is now unusable anyway
* Have further improved the performance of the check all roms code – output is piped to a temporary file and that file is used for parsing. This takes care of buffering issues causing the code to take far too long to execute. Have also gotten rid of the progress bar as it was almost worthless.
* Added a backup and restore function to the File menu. This will allow you to backup and restore the application preferences and DAT files and includes the following;
  + categories.dat
  + favorites.dat
  + ratings.dat
  + runs.dat
  + pfemame.ini
* Fixed a bug that caused incorrect game information from being displayed for the psa rom (sony playstation one)’ the psa entry in the history.dat file didnt exist so it was picking up a partial match from a different program with psa in the name. Had to add a check for the comma at the end of the info string ($info=psa,).

v1.004

* Redesigned the handling of system files. Initial implementation was a hack that merged the system file in with the rom file name. This was fine except that very long system names made it impossible to see what the file name was. It was also just a major hack to the code and quite ugly. System file names now have their own column. This has made the window a bit wider but its a better implementation method. When you first upgrade to this new version the code will force a database refresh as the cache file format is not compatible.
* Added notification to messages window when game ratings are updated
* Tile View now supports system program file images rather than just displaying the image for the top level system name
* Tile View number of runs is now fixed
* Corrected the way the games in the favourites list are deleted – it didn’t work with systems files
* Updated the manual to provide a caution regarding manual editing of any of the DAT files
* Updated some of the screenshots in the manual
* Miscellaneous minor changes and bug fixes
* Added F5 to refresh the games database
* The systems support preferences option has been changed. The program will now always look for and process systems into the game list, but if system support preference is disabled they will just not be displayed – the systems column will be hidden and the window size will shrink.
* Added a lookup list for game categories that are ‘systems’ - currently this is the only way that I can determine that a rom is for a system without implementing yet another dat file which would slow the game list build substantially. At the moment the lookup list is hard coded but I may later put it as an external file to make it easier to upkeep.
* Some code optimisations to populatelist – I’m doing this to try to speed up the search-as-you-type feature which is now quite slow given all of the fields it is searching through. Each keypress calls the populatelist code so I need to make some improvements
* Added some error trap to the stdout out & err processing. This was necessary due to testing with a pre-release of mame version 0.181 – it was outputting non ascii mapped characters that were throwing a unicode error.
* Fixed a bug in the gametypes.py code. When you selected OK it would hide the window before switching back to the parent. This would cause the parent to hide also meaning you had to select it from the taskbar to bring it back into view.

v1.003

* Have disabled number of runs update in TileView – have broken it with the including of the ability to run systems – need to try and fix it again later.
* Improved how the game info is parsed from history.dat. The previous method displayed a laggy update of the game information box which in turn slowed down the display and scrolling.
* Fixed bug under linux where you could not close the ‘About’ dialog box using the OK button. Pressing it actually threw an endmodal error. Windows didn’t seem to care.
* Changed pfeMAME to pfeMAME
* Added checks in the code for runs.dat, favorites.dat, and ratings.dat processing code that handles a name in one of these files not existing in the actual mame roms list causing a lockup. Not normally an issue but if you happened to have a rom that you have run and added as a favorite, then some time later deleted that rom, then pfeMAME would lock up when you performed a database refresh. Also added better checks for malformed dat files.
* Added -offscreen\_reload to support button 2 on lightguns as a reload button. Have hard coded for now.
* Errors in dat files will now display a message in the info screen on startup pointing you to review the log file for more information.
* Number of runs and game star rating now support system roms (e.g. ZX Spectrum roms).
* Re enabled the running of verifyroms before running system roms now that systems are better supported.
* The function checkall has been added back in now that I have finally gotten it to work. As it calls the -verify roms in a sequential process the window does freeze until it is complete
* TileView now supports systems – make sure you try to run them under the default ‘Favorites’ view – if you try running them under the ‘All’ view then it will only run the top level system ROM as it sees that one first.
* Tile view instructions was missing the last line telling you how to exit tile View by pressing ‘2’. Had to increase the Y dimension for the panel
* Have added a message on screen in the Messages box to confirm a game has been added to or removed from the favorites list
* When deleting a game from the favorites list, it would delete the correct one, and then the next one to it. I had a onkeydown and then an onkeyup trigger for the event. Have removed onkeydown. Also renamed the function from onKeyDown to onKeyPress
* Corrected some logfile messages relating to the startup sequence. On startup it tries to open the temporary games database cache. If it exists then all good but it was still logging that it was rebuilding the full games database when it wasnt.
* Have added an execution time to the full games database refresh – will help in the future to find code speed improvements
* Removed the border from around the Messages and Game Information boxes and merged the background color. Makes it blend in better and looks less blocky.

v1.002

* Removed all traces of the folderMemcard code as this was removed from MAME some time ago and I only put a quick fix in at the time just to get things going.
* Fixed rungame code that was preventing the number of runs in Linux from working. Forgot to set the error output strings to empty at the start of the code. Was ok if there was an error, otherwise a problem.
* Removed the ‘check all’ option – it just never worked properly and when it did it was so slow it was pretty much pointless
* Changed the help>controls window to a scrolled window so that the scrollbar works in both Linux and Windows.
* Fixed a problem in Linux in Tile View where any empty spaces (i.e. no game assigned to that space) showed a corrupted graphic in the space. The issue is a bug in EmptyBitmap in Linux. Have replaced this with a Black png image.
* Have added the ability to add a system(Home Computer, Game Console) to the favorites list
* Added support for system program snapshots. They MUST be names the same as the program file and especially for Linux they MUST have the same case. E.g. the snamshot image file for BRUCE LEE.sna must be BRUCE LEE.png

v1.001

* Fixed a major bug – some time ago I added an include for test2 which was just for testing purposes. The test2.py was never distributed but the include was still there. This I can only assume would mean anyone trying this for the first time would find a broken program. Very sorry.
* As of MAME v0.172 the command -multhreaded no longer exists. This stopped pfeMAME from being able to run any roms. Have removed this command from the rungame and other places.
* Removed sixaxis option as it seems to be no longer supported.
* Fixed a bug that was stopping the command string from being written to the log file in rungame.
* Switched back to compiling 32 bit target rather than 64 bit. Found no real advantage in 64 bit, and it caused some pose execution lock up issues – maybe some stability issues somewhere. Tested under Windows 10 and works ok as a 32 bit target.
* Type ahead search working much faster now its back as 32 bit.
* Added logging of pfeMAME version
* Added logging of Add / remove favourites
* Fixed a problem introduced in v0.173 MAME whereby getting the mame version with the -? switch now returns too much information. To maintain compatibility with earlier versions I have added a maximum string length as well as the old search criteria to build the software version.
* Added capturing the ENTER key to run a selected game (normal view, not tile view)
* Improved the speed of scrolling though games by pre-loading the history.dat file and parsing that file on the fly for game info. The old way I did it was parsing the file on the fly which had large overheads in speed and just not very efficient.
* Updated any code using subprocess.Popen to select the shell type depending upon if it is running under Windows or Linux. This is to solve issues relating to how the different operating system shell command interpreters work and that under Windows I have to use shell=True to prevent lockups (I still think this is something to do with 64 bit code execution under python as it never happened under XP)
* Prevented the rungame code from trying to extract the console output from the subprocess.popen call when running games under Tile View. This fixed further lockups that seemed to be happening under Windows 64 bit.
* Added better prompting on startup if there are no games categories set. It will now open the window for you to select some.
* Improved how systems are displayed so that all program files that are associated with a system (e.g. ZX Spectrum) are now grouped under the master system file name with a ‘-→’ between the system name and the program file name. Makes things look a lot nicer.

v1.000

* Fixed a bug in rungane for systems. Was performing a check to find a matching media type based upon the file extension. I had made an assumption that media types were 4 characters long when they can actually be variable length. Fixed it.
* Added a check in rungame for systems – if you try running a system program / rom with a file extension that doesn’t exist in the list of media types, a message popup will warn tell you that the extension type is unknown and the program will not be executed. This has fixed some annoying error popups under windows.
* Added a bit more information for first time run to try and help users that have never used pfeMAME before.
* After running a database refresh a check is performed on the number of games in the visible list. If its zero(0) then a message pops up advising you to check if you have any game categories selected

v0.999

* As of now, any pre-packaged Windows version is for 64 bit installs and will not run on 32 bit. If you need 32 bit then you will need to package it yourself as I cant manage the time involved in packaging different versions.
* **IMPORTANT:** As of this version, if you wish to have support for Home Computers (systems), you must use a version of catver.ini that has categories for home computers (e.g. ZX Spectrum). Try [http://www.progettosnaps.net](http://www.progettosnaps.net/)
  + The category description for the rom / program / game etc needs to be 'Home Computer', 'Game Console', etc..
* Games categories window modified to put the categories list into a scrollable panel. Had to do this as the latest catver.ini files have too many categories to fit onto the screen without overflow (Due to MAME and MESS being merged). The buttons are at the bottom of th escrolling panel so you have to scroll to the bottom to get to the buttons (I know this can be done better but that's for when I have more time)
* Started adding support for systems (e.g. ZX Spectrum) as follows;
  + As of MAME 0.164 MAME and MESS have been merged. This means that MAME can now run MESS systems like the ZX Spectrum. I have allowed these to be built into the main games list (after a refresh).
  + Added a Systems Support option to global preferences. Selecting this requires a manual refresh of the games database.
  + Added a Systems folder to the file paths – this is where your systems programs will be stored (in subfolders named after that particular system
  + Games list now also looks in a defined 'Systems' folder for the particular system (e.g. spectrum) and then media in that folder that fits the mame listmedia requirements for that system. This media or games / programs are then added to the main game database list
  + The category description for the rom / program / game etc needs to be 'Home Computer'.
* Have disabled game error / message output capture until I can solve a lock up issue found in Windows10 testing whereby if the game is exited very quickly after running (say 5 seconds), the code locks up at the p.communicate().
* Fixed the About window and Properties windows not closing properly without generating an error under windows 10 (Regression testing required for Windows XP and Linux)
* All child windows now have a specified pfeMAME icon to show in the title bar – was needed under windows (Linux obviously doesn’t use it)
* Removed the website reference from the 'About' window as it doesn’t exist anymore – just not enough time & money to pay for it.
* Fixed the code in add to favourite – was throwing an error if a game wasn’t currently selected when added to the favorites.
* Fixed bug in filepaths input that was throwing an error if the Systems folder was undefined.
* Fixed bug in filepaths input that didn’t check for an invalid Cheats folder
* Added the Backspace key to be able to be used to delete a game from the favorites list (Up until now you could only use the Delete key)
* Improved some of the code used to build the games database to streamline just a little
* Added a 'Controls' popup window under HELP. This displays a list of the in-game controls for MAME.
* Have fixed the annoying problem when running the stand alone executable version under Windows where a console window would open whenever running a game, verifying roms, etc.
* The manual has been updated. It also now contains a known problems section.
* Substantial improvement in speed when building / rebuilding the games database. As an example, a large library with over 4000 games used to take 50 seconds on a very fast machine. This then reduced to 19 seconds after the code opimisations.
* Fixed the embedded icon not displaying correctly in Windows VISTA/7/8/10 (Would only work in XP).
* Created and including in source code distribution a 'setup.bat' file to simplify the building process for Windows distribution builds.
* Updated the windows build setup.py file to include file name, description, and version. This information is then included in the built EXE file.
* Fixed an annoying windows related issue where any warnings / errors were written to another log file and also caused a popup window after closing the application telling you to review the log file. This was just plain annoying so good riddance.

v0.998

* When looking for game snapshot images to display, it now looks to see if a folder exists with that game/system name. If it does, it grabs the first image in that folder which should be named 0000.png. This adds support for system image snapshots which get put into a folder names as the system (e.g. spectrum).
* Tile view now updates the number of runs globally – meaning that when you return to the main window mode, the number of runs has been updated there as well as in the database. To do this I had to remove the feature that brought the played game to the top of the list.
* A few cosmetic changes to the games list header to make it stand out a bit better
* Fixed a bug in the type ahead filtering for search. It was all of a sudden broken with a newer install / version of mame (not sure exactly which one). Just decoded the strings into UTF-8 for the comparison and it fixed it.
* Added a text display on the screen to show the database cache age in days
* Added a check to make sure a game was actually selected before allowing the game to be run – just prevents odd output errors if you click the Play Game icon after first running the application but not selecting any games
* After performing a database cache manual refresh, the search input box is cleared. Had to be done as after a refresh the games list is rebuilt and reset to the start so any previous search value is useless.

v0.997

* When you select a different view from the drop down list, and search text in the search input box is deleted as the search is reset anyway
* Removed 'memcard' from the file paths window as it is no longer used by MAME (I had previously released a quickfix by dumping it from the command line but still had to delete it completely.
* Fixed the incorrect naming of variables in the [driver].ini files (INI path). This was causing INI file read errors from being reported in the 'Messages' window after running games. NOTE: YOU NEED TO DELETE YOUR INI FILES IN THE 'INI' FOLDER AND RESET ANY GAME SPECIFIC PREFERENCES FROM STOPPING THIS ERROR FROM POPPING UP IN FUTURE – SORRY FOR THE PAIN.
* Cosmetic changes in the Preferences window
  + Moved an asterisk to better location for readability
  + Removed a wayward ')'
  + Removed 'opengl16' and 'none' from the video rendering options. OpenGL16 for Linux worked on some linux platforms and not others. 'None' was an option to not draw any video at all and is really only there for MAME developers so I didn't see any practical need for it.
  + Arranged some options in a grid view to better utilise space and make room for the expanded set of artwork options
  + Added an expanded set of artwork options
* Added a full expanded set of artwork options to the system and game specific preferences including backdrops, overlays, bezels, control panels, and marquees
* Fixed some logging info when running games. It was always saying the game ran in Tile View even if it was normal view, and the message stating if using default or game specific preferences wasnt being displayed and logged
* Minor cosmetic changes in tile view;
  + 'From:' text changed to 'Available in this view:' as it makes more sense
  + Size of instructions and informative text reduced to fit things in a bit better – may still need some work though
* Added 'Number of runs:' into the Tile view information panel
* Fixed the way the search field is reset – now the previous search value is only reset when the view drop down box is altered.
* Added into preferences a field for the Games Cache number of days of age – now you can define how many days will pass before the program prompts you to update the games cache
* Some code improvements to the Check All Games function and window – slight speed improvements
* Moved the platform check (windows, linux, etc) into functions.py so that in the future additional platform checks can be centralised

v0.996

* Removed the setting of the memcard folder from the run game command line. This command seems to have been removed in the latest version of MAME for some reason. I cannot yet find a reason why.

v0.995

* Re-fixed the absolute paths for icons, images, files etc. The method I previously used was not 100% when it came to Linux / Windows and has now been tested to work under the following conditions;
  + Windows & Linux
  + Running application source(.py) from command line
  + Running from editor
  + Running compiled .exe directly
  + Running compiled .exe from a desktop shortcut
* Updated the pfeMAME.ico file to be more correct (multiple embedded resolutions). Design has also been updated
* Under windows, the EXE has an embedded icon
* Added to preferences the ability to highlight games that don't have a corresponding rom file in the ROMs folder path
* Fixed some graphical layout issues in the main view under Windows – the View combobox and search entry field top border were being overdrawn by the toolbar icons.
* Altered the way the games list is refreshed after changing; star ratings, number of runs, and adding to favorites. The list will now maintain whatever filter view it had previously prior to these actions.
* Fixed a bug that caused games with no category listing in catver.ini from being repeated may times in the game list as it tried to find a match through the categories.dat file.

v0.994

* Added a new preference to set the default filter view for the main game display so that the application will switch to your preferred view after startup (All, Favorites, Exists)
* Re-ordered some of the preferences in the preferences view
* Fixed relative path pointers so that if the application is run via a link (especially under Linux), the path to supporting files etc is correct and errors are prevented
* Added a check for the mame version and displays the version on the main screen
* Updated the formatting of the messages window to improve display
* Added the pfeMAME version number and mame executable version number display to the tile view

v0.993

* Fixed a bug in verify roms code, and check all roms code. I was not specifying the full path name to the roms. This would be ok if the mame.ini file contained the rom folder but as I am trying to get away from that this code needs to work correctly.
* You can now use .7z rom files - fixed a bug that prevented rom files with different extensions (e.g. .7z) from being shown in the exists list. I was filtering for .zip only.
* Fixed a bug that caused an error to be generated if after first time run you tried to set some game categories before the games database had been refreshed (or a restart performed)
* Fixed a 'feature' in tile view where if the mouse is used to click on a game image then the focus and location is screwed up. The idea is that the mouse is not used at all but if you do use it then the keyboard commands my stop responding and you're stuck unable to exit the view.
* Fixed the MAME logo in tile view – had accidentally made it very dull in previous release
* Some formatting changes to the rominfo window – looks nicer – fits better.
* 'Preferences' changed to 'Default Preferences'
* Enabled Game Specific Preferences. When a game is run, it will now check first if an INI folder is set up, and if so, if a valid game INI file exists. If it does it will read out any game specific preferences and use those rather than default preferences.
* Updated the preferences window to support either default or game preferences mode
* Game preferences changes write to the INI file in the INI folder path
* Added to the right click menu 'Game Preferences' so that game specific preferences can be set
* game command string and running code from main view and tile view combined into common class and imported from separate file. Code efficiency improvement.

v0.992

* Added TE\_RICH and TE\_AUTO\_URL to the game info text control – this should format the game URL as an actual URL.
* Added a busy info display for when building the game icon list. This is the only real way to tell the user what's going on as the icon list is created before the window is drawn as the control has to be created before it is inserted into the sizer etc etc. At the moment the icon list only takes ages if in preferences the option to not scale the icons is selected. Although counter intuitive, this is because scaling is done on the fly dynamically on only the icons displayed so its reasonably fast on modern systems (I don't have an older system to see how it goes on them). When you don't select the scaling, the icons will be displayed full size and nice and pretty. Only problem is that the program has to work out the maximum size of the largest icon so that it can set the row height up. The only way so far to do this is to scan each and every icon in the file path and check their sizes on startup, and then just set the row height to the largest size – hence this takes ages.
* Replaced the list control used in tile view for displaying the game list with a customized control built up with a series of text controls. The main reason for this is to have better control of how it all worked and looked, as well as solving the problem of displaying a scrollbar which I don't want (especially in Windows as it looks UGLY).
* Altered the horizontal spacing math in tile view as at some image sizes it was forcing the information panel off the RHS a bit.
* Updated the Quickstart & readme file. Also removed from the distribution files – just download it separately.
* Update the tileview code to include a panel around the information panel area and then applied a color to the panel. This will allow me to apply some color to this area to make it stand out a bit
* Tidied up the code in TileView and split the information panel out into a sub class
* When using the type ahead search, it will now also look for a match with the rom file name, not just the game name
* Removed the logging of the filtered list creation as it was firing too many logs when using the type ahead search
* The search is also more intelligent now. You can enter a series of search words that don't have to be in a specific order in order to get the right match – useful if you don't exactly know the right format of a long name (e.g. 'blah blah (bootleg of whatever)' – you could search for 'blah bootleg')
* Quickstart guide changed to pfeMAME Manual as there is more detail being added all the time so it's more than just a quickstart guide I think.
* In preferences the video option has been updated to be platform specific (For windows and Linux as I don't know the video options for OSX and don't have access to the operating system for testing). The INI file default writer has also been updated to include these differences when it creates a fresh pfeMAME.ini file. This change also fixes the error message in Windows that an invalid video option has been selected.
* Fixed a bug in the default INI file creator that wasn't setting default preferences correctly when creating a fresh pfeMAME.ini file.
* When running 'check all games' it checks the roms and also the samples now – while this takes a lot longer, it is more thorough.
* The information panel in tile view now expands to the right to fill up unused space. This makes much more efficient use of space and also allows for longer game names to be displayed.

v0.991

* Added a block around image drawing to prevent the system from reporting errors with malformed PNG files. In WindowsXP this was still causing crashes due to some odd reason.
* Added some padding around the border of the gametypes window to make it look nicer.
* Added some padding around the border of the filepaths window to make it look nicer.
* Added some padding around the border of the preferences window to make it look nicer.
* Added a check to gametypes for the existence of a path to catver.ini. If not present it warns the user and exits.
* In the main application I have added a check for catver.ini valid path when reading the preferences. If it is not set then it tries to fall back to local copy (out of date) distributed with the application. This is so that the game list is at least built up correctly.
* Added the writing of the results from checking all roms to a text file

v0.99

* Changed the icon list creation to a 'virtual' method. This greatly speeds up loading.
* Added an option to scale or not scale the icons. Scaling is faster to load.
* Added a 'cached' games database (written to csv file) which speeds up the startup time hugely. The system checks if the cache is older than 5 days and if it is it lets you know that it should be updated.
* Replaced the old blank icon with a transparent 16x16 pixel blank
* Changed 'defaultgamepreferences' to 'preferences'
* Fixed bug that would cause lockup if we tried to run a game without any game being selected. Also fixed the same for rom information
* Added a toolbar
* Changed add to favorites and star rating to only update the logfile if a game was actually selected prior to selecting the option
* Fixed bug introduced with ultimatelistctrl – delete keypress was not being decoded correctly as the event name changes with this control
* Updated the license in the 'About' box
* Added a 'check all' window
* Added a fallback check if the pfeMAME.ini file doesnt exist – it will be created from scratch
* Added a check if the path for the mame executable has not been set – the application will not try to build the games list and will prompt the user to set the path.
* Fixed a bug that caused lockups / crashes if the categories.dat file didnt exist. Major problem for first-time run.

v0.985

* Updated filepaths child to remember the last selected folder when defining folders. This is mostly just useful the first time the application is run when you go to set everything up – usually all of the files / folders are in the same path or very close. If you close the window then it forgets the path.
* Added a scrolling games list on RHS of the tileview
* In Tile view I have added a basic check for the operating system type ('win' or 'linux'). This is currently used just to sort out the difference in how the two platforms exit a game and get back into fullscreen mode. It may be useful for other rendering differences later.

v0.984

* An override class was added for wx.Log to redirect to the game messages box and the application log file. This takes care of the nast popup message 'tEXt CRC error' that happens when I try to display malformed png images.
* Hid the debug panel in text view by default. Pressing the '3' key will toggle it visible / hidden.
* Altered the 'getExists' code so that it first created a list of games in the folderRoms directory before performing the check if they exist. Previously I tested each one on the fly by trying to open it – this is fine if ti is local ona fast drive. If the drive is remote or visualized then this took AGES!!!!. much better now.
* Added the scrollbar to the 'checking if exists' stage of startup
* Removed the border from the bitmapbutton and added a horizontal gap of 2 pixels to the gridsizer. This makes the buttons look just like an image in Windows as they do in Linux.
* Fixed a bug I had introduced into TileView whereby the game would not run – this was introduced when I updated the command string generation – I was trying to write directly to the log file rather than using the remote function call designed for child windows.
* Corrected a bug in the alpha fade out / in of game images in tile view when the screen is scrolled up / down. When scrolled, position 0 would be highlited when it should be the game button with focus.
* In tile view I have adjusted the vertical and horizontal spacings a bit to better lay things out and prevent the highlite box and game text from being cut off at times.
* Fixed a bug that caused the screen not to redraw correctly when coming out of a game in tile view.
* Switched the list control to an ultimatelistctrl. This allows me to use different image sizes so that I can go back to using my awesome 5-star icons. I'm hoping it may also allow me to sort out the slow loading of the icons somehow.

v0.983

* Joystick navigation in TileView added
* Added preferences to use the game artwork
* Added preferences to show the game icons (can help speedup if the icon sizes are very large)
* Fixed a bug introduced earlier which prevented the output to the log file of the game options when running a game. This happened when I changed the way the 'cmd' is formatted for command line operations.
* Changed the way the game execution command line is built up after I learned a bit more about how the Popen command actually worked. I believe I am doing it correctly now and it also doesn't need the command prompt = True bit which looked odd in WindowsXP.

v0.982

* Changed the star rating icons to a 16x16 icon as under windows a larger icon would cause a python crash. This is due to Windows using a different control formatting than Linux. Linux allows any mix of different size images while windows does not.
* Created an application icon

v0.981

* Removed the disable instruction from the game name text controls as under windows XP this caused the background color to become light grey instead of black.
* Added -multithreading option to default game preferences and normal / tile views
* Changed name to pfeMAME as my previous name already existed in the Internet (should have checked)
* Added state and input folders settings
* Removed the background color setting that I was using in defaultgamepreferences module for the staticbox. In linux it just provided a nice colour for the outline, but in windows it made the entire background colour look horrible.
* Create my own single choice dialog as the built in one crashes under windows XP.
* In Tile View, removed the wx.Panel and have instead drawn all objects direct to the window. I had to do this in order to get the arrow keys to be detected without holding down the 'ALT' key under windows XP (Linux worked ok). It seems this issue is caused by the differences between Linux and Winodws in how they handle the navigation keys.
* During WindowxXP testing I found and fixed the following bugs;
  + FileDialog was crashing in the folderpaths module due to incorrect wildcard settings.
  + Font settings were failing due to incorrect setting of default font.
  + The command string for building the game list using mame -ll was incorrect. I had followed an example for passing multiple commands to the output by putting them in parenthesis, but If there is just one command then you need to omit the parenthesis.
  + Right click popup over the list control took two right clicks. I found out that I needed to change the event triggering to a button 'down' event, not up. This then ment that I had to manually use a hittest to determine the mouse pointer location and then select that row so that the event calls actually knew what was selected.
  + The search box height was huge whereas under linux it seemed find. This is because I was manually setting the height. I have changed this to default (-1) so that the system font size will now set the height. This way it looks ok under both operating systems.
  + Fixed a python crash if we hit delete on the favorites view when nothing is selected.
  + Created my own 'About' dialog box as the built in one crashed under windows XP