

mAirList

Kompakte Sendeautomation für Windows

Handbuch

Torben Weibert

7. Oktober 2005

Wichtiger Hinweis

Dies ist eine vorläufige Anleitung für mAirList. Sie bezieht sich auf die Version 1.3. Die Anleitung wird nach Bedarf und vor allem nach Zeit aktualisiert, ergänzt und erweitert. Da sich mAirList aber fortlaufend weiterentwickelt, und nicht immer genug Zeit da ist, auch die Anleitung zu aktualisieren, ist dieses Dokument nicht immer auf dem neuesten Stand. Darauf ist beim Lesen bitte Rücksicht zu nehmen.

Bei offenen Fragen steht das mAirList-Supportforum unter <http://forum.mAirList.de> jederzeit zur Verfügung.

Inhaltsverzeichnis

1. Einführung	5
1.1. Über mAirList	5
1.2. Systemvoraussetzungen	6
1.3. Lizenz	6
1.4. Haftungsausschluss	6
1.5. Über den Autor	6
1.6. In eigener Sache	6
2. Installation	7
3. Konzept	8
3.1. Playlists und Player	8
3.2. Die Cartwall	8
3.3. Die Browser	9
3.4. Faderstart	9
3.5. Die Datenbankanbindung eldoDB	9
4. Konfiguration	9
4.1. Playlists	9
4.2. Cartwall	12
4.3. Faderstart	13
4.3.1. Joystick/Gameport	14
4.3.2. Hotkeys	14
4.3.3. MIDI	14
4.4. HotDirs	14
4.5. Logging	14
4.6. eldoDB	15
4.7. Fernsteuerung	15
4.8. Optionen	16
5. mAirList benutzen	17
5.1. Das Hauptfenster	17
5.1.1. Die Playlist	18
5.1.2. Die Browser	18
5.1.3. Die Cartwall	18
5.2. Arbeiten mit der Playlist	18
5.2.1. Einfügen von Elementen	18
5.2.2. Verschieben von Elementen	18
5.2.3. Löschen von Elementen	18
5.2.4. Playlist-Optionen	19
5.3. Arbeiten mit Browsern	19
5.3.1. Der Archiv-Browser	20
5.3.2. Der Verzeichnis-Browser	20
5.3.3. Der Verzeichnisbaum-Browser	20
5.3.4. Der Playlist-Browser	20
5.3.5. Der Datenbank-Such-Browser	20
5.3.6. Der Datenbank-Playlist-Browser	20
5.3.7. Der Uhr-Browser	21

5.4. Arbeiten mit den Playern	21
5.4.1. Player laden	21
5.4.2. Spielzeit- und Backtiming-Anzeige	21
5.4.3. Starten und Stoppen von Playern	21
5.4.4. Der erweiterte PFL-Modus	21
5.4.5. Visualisierung der Ramp	22
5.5. Arbeiten mit der Cartwall	23
5.6. Öffnen und Speichern von Dateien	23
5.6.1. Vorlagen	23
5.6.2. Standard-Vorlage	23
5.6.3. Import von Fremdformaten	23
5.6.4. Anhängen von Dateien	23
5.7. Tastaturbedienung	24
5.8. Die Automation	24
A. Fenster-Layout anpassen	24
A.1. Aufbau von layout.ini	24
A.2. Beschreibung der Parameter	25
A.2.1. Gemeinsame Parameter aller Elemente	25
A.2.2. Zusätzliche Parameter von [Playlist <i>x</i>]	25
A.2.3. Zusätzliche Parameter von [Player <i>x_y</i>]	26
A.2.4. Zusätzliche Parameter von [CartWall]	26
B. Kommandozeilen-Parameter	27

1. Einführung

1.1. Über mAirList

mAirList ist eine kleine, schnelle, universell einsetzbare Sendeautomation für Windows 95/98/ME/2000/XP.

Zu den wesentlichen Features von mAirList gehören:

- Beliebige viele Playlists.
- Beliebige viele Player pro Playlist.
- Simultanes Ausspielen auf beliebig vielen Playern gleichzeitig.
- Zusätzlich integrierte Cartwall/Jingleplayer mit beliebig vielen Slots.
- Benutzt BASS.DLL, spielt die Formate MP3, MP2, WAV, WMA und OGG.
- "Intelligente Playlist" lädt die Player automatisch nach. Wahlweise können aber auch einzelne Stacks für jeden Player verwendet werden ("Split Mode").
- Dateien können sehr einfach durch Drag&Drop aus dem Windows-Explorer oder den eingebauten Browsern eingefügt werden.
- Zu jeder Datei können Cue-, Ramp/Intro-, Outro- und Fade-Out-Punkte definiert und wahlweise in der Playlist oder im ID3-Tag von MP3s gespeichert werden. Die Ramp/Intro-Zeit wird beim Abspielen vom Player graphisch dargestellt.
- Automations-Modus "fährt" die Sendung automatisch unter Beachtung von Cue- und Fade-Out-Punkten.
- Universelle Faderstart-Möglichkeiten per Hotkeys, MIDI oder Gameport. Über zweckentfremdete Joystick-Interfaces können so fast alle Mischpulte mit Faderstart-Ausgang zusammen mit mAirList betrieben werden.
- Anbindung an die "Schwester"-Musikdatenbank eldoDB zum schnellen Zugriff auf das Musik-Archiv und vordefinierte Playlisten.
- Arbeitsumgebung und Playlist kann in XML-basierendem Dateiformat gespeichert und geladen werden.
- Arbeitsoberfläche kann optisch frei gestaltet werden (Position, Farbe und Erscheinungsbild der graphischen Elemente).

mAirList wurde als Sendeautomation für kleine, nicht-kommerzielle Radiosender geschrieben, die sich eine "kommerzielle" Automation schon aus Kostengründen nicht leisten können, die aber auch den Verwaltungsaufwand scheuen, den die kommerziellen Produkte mit ihren komplexen Datenbanken nach sich ziehen.

1.2. Systemvoraussetzungen

mAirList stellt die folgenden (wenigen) Anforderungen an den Computer, auf dem es betrieben werden soll:

- Microsoft Windows 95, 98, ME, 2000 oder XP.
- Von Windows unterstützte Soundkarte. Für den reibungslosen Betrieb mit der BASS.DLL sollte der Soundkarten-Treiber im WDM-Format vorliegen (was aber bei den aktuellen Karten fast ausnahmslos der Fall ist).
- Aktuelle Version von DirectX, mindestens Version 8.1. Die jeweils neueste Version kann auf <http://www.microsoft.com> heruntergeladen werden.
- Für Faderstart-Anbindung an das Mischpult (optional): MIDI- oder Gameport bzw. umgebauter Joystick mit Treiber.
- Für die Datenbankanbindung (optional): Ein von mAirList unterstützter SQL-Server (PostgreSQL, MySQL bis Version 4.0, o.ä.).

Wie performant mAirList ist, hängt von der Performance des Rechners ab. Auf aktuellen Systemen ist das Ausspielen auf allen drei Kanälen gleichzeitig überhaupt kein Problem. Ansonsten gilt: ausprobieren (und evtl. die CPU-Auslastung im Auge behalten).

1.3. Lizenz

mAirList darf von nicht-kommerziellen Radiosendern und -projekten kostenlos eingesetzt werden. Gleiches gilt für Privatpersonen, sofern durch den Einsatz der Software keine Einnahmen erzielt werden.

Anfragen zum kommerziellen Einsatz bitte an den Autor. E-Mail: torben@weibert.de

1.4. Haftungsausschluss

mAirList wurde nach bestem Wissen und Gewissen entwickelt und getestet. Der Autor haftet nicht für Fehler in seinem Programm und eventuelle dadurch verursachte Schäden.

1.5. Über den Autor

Torben Weibert, Jahrgang 1977, ist Diplom-Informatiker und wissenschaftlicher Mitarbeiter der Arbeitsgruppe "Informationssysteme und Sicherheit am Lehrstuhl" am Lehrstuhl Informatik VI der Universität Dortmund. Seit November 2000 ist er beim Dortmunder Campusradio *eldorado** tätig – sowohl als Moderator und Redakteur als auch als "Herrscher" über die Technik.

1.6. In eigener Sache ...

Wenn mir vor drei Jahren jemand gesagt hätte, dass ich einmal ein Handbuch für mAirList schreiben würde, hätte ich nur ungläubig gelacht – denn dass das Programm einmal außerhalb von *eldorado** zum

Einsatz kommen würde, war weder geplant noch absehbar. Eigentlich sollte mAirList nicht einmal bei eldorado* zum allgemeinen Einsatz kommen, denn ich hatte mAirList vornehmlich für **mich** geschrieben, damit ich nicht mehr mit dem alten, umständlichen¹ "Airlist" meine Sendungen fahren musste.

Dass es da was gab, was MP3s spielen konnte und einfaches Drag & Drop in die Playlist beherrschte, sprach sich unter den Moderatoren bald herum, und plötzlich wollte jeder mAirList benutzen. Die Argumente sprachen wohl für mich ;) So wurde mAirList zum offiziellen eldorado-Jingle-Player (die Musik kommt bei uns noch immer von CD). Irgendwann kamen auch die ersten Anfragen von anderen Sendern. Radio Q aus Münster waren glaube ich die ersten. Ob sie mAirList jemals benutzt haben, weiß ich nicht, jedenfalls habe ich damals die alte Anleitung geschrieben.

Der Schritt, die Software auch für andere Sender zugänglich zu machen, lag dann eigentlich auf der Hand. Seitdem hat sich mAirList kontinuierlich weiterentwickelt, etliche Features sind hinzugekommen, Dinge, die bei eldorado* vermutlich nie zum Einsatz kommen werden, die aber für andere Sender wichtig waren, so dass es sich lohnte, sie einzubauen. Nicht zuletzt durch die vielen Ideen und Testberichte, die über die Support-Mailingliste kamen, wuchs mAirList zu einem brauchbaren "Produkt" heran. Allen, die sich daran beteiligt haben, sei hiermit herzlichst gedankt.

Wo und in welcher Anzahl mAirList heute benutzt wird, kann ich höchstens anhand der Web-Statistiken raten. Die Download-Zahlen liegen bei etwa 100 pro Monat, Tendenz steigend. Das soll aber nichts heißen. Natürlich bin ich neugierig. Wer also – wo er schon kein Geld für die Software bezahlen muss – mir jedenfalls mitteilen will, wo mAirList bei ihm im Einsatz ist, darf mir gerne eine kleine Mail schreiben: torben@weibert.de.

2. Installation

mAirList wird als .zip-Datei ausgeliefert. Diese enthält die folgenden Dateien:

- mAirList.exe – die Programmdatei
- mAirListConfig.exe - das Konfigurationsprogramm
- mAirList.ini – eine (Beispiel-)Konfigurationsdatei
- layout.ini – eine Beispiel-Layout-Konfigurationsdatei
- WHATSNEW.txt – eine Textdatei mit den Neuerungen seit der letzten Version
- mAirList.pdf – diese Anleitung
- BASS.DLL – Bibliothek für die Soundausgabe.
- BASSWMA.DLL – Erweiterung für das Windows Media Audio (WMA) Dateiformat.

mAirList kann ganz einfach durch Entpacken des Archivs in ein beliebiges Verzeichnis installiert werden. Eine Setup-Routine gibt es nicht. Ebenso wenig benutzt mAirList die Windows-Registry. Alle Konfigurationsdaten sind in der Datei mAirList.ini abgelegt. Dadurch kann mAirList auch im Zweifel einfach und ohne Rückstände wieder entfernt werden, indem das Programmverzeichnis wieder gelöscht wird.

¹Bei Airlist musste man für jedes Element erst lange Titel und Autor eintippen, bevor man es in die Playlist einfügen konnte; ein Vorgang, der so langwierig war, dass der Moderator der Morgensendung das meist schon am Vorabend erledigte oder vom Redakteur erledigen ließ – noch heute spricht man bei eldorado* noch manchmal vom "Airlist bauen". Trotzdem bin ich dem Autor des alten Airlist natürlich dankbar, denn mAirList hat sich viel von der alten Software abgeguckt.

Soll mAirList mit der eldoDB-Datenbank zusammen betrieben werden, muss zusätzlich die Datei `libmysql323.dll` aus dem eldoDB-Archiv in das mAirList-Verzeichnis bzw. ein anderes im Systempfad liegendes Verzeichnis kopiert werden.

Vor dem ersten Start muss mAirList mit dem Konfigurationsprogramm konfiguriert werden (siehe Abschnitt 4).

3. Konzept

In diesem Abschnitt wird das grundlegende Konzept von mAirList anhand seiner einzelnen Komponenten erläutert. In Abschnitt 4 wird dann beschrieben, wie die gewünschten Komponenten im Konfigurationsprogramm ausgewählt und eingerichtet werden.

3.1. Playlists und Player

Die wichtigste Komponente von mAirList sind eine oder mehrere Playlists, die auch den größten Platz auf der Benutzeroberfläche einnehmen. mAirList unterstützt beliebig viele Playlists gleichzeitig, wenngleich für die meisten Anwendungsszenarien eine oder zwei Playlists ausreichen sollten.

Die Playlist enthält die Elemente (z. B. Lieder, Jingles oder Beiträge), die in der laufenden Sendung ausgespielt werden sollen. Dazu verfügt jede Playlist über einen oder mehrere Player, die am unteren Bildschirmrand dargestellt werden. Wird mAirList an einem Mischpult eingesetzt, sollte jeder Player einem physikalischen Kanal auf dem Mischpult und damit einem Fader zugeordnet.

Die Elemente aus der Playlist werden nacheinander auf den ihr zugeordneten Playern ausgespielt, entweder automatisch (Automations-Betrieb) oder manuell durch den Moderator (Assist-Betrieb). Zusätzlich können die Elemente über die PFL-Funktion vorgehört werden. Die Ausgabe kann auf einer eigenen Soundkarte oder derselben Soundkarte erfolgen, auf der auch das eigentliche Ausspielen stattfindet. Über die Extra-PFL-Funktion können Elemente zusätzlich über einen vordefinierten Kanal vorgehört werden, auch dann, wenn der Player bereits läuft.

Im Assist-Betrieb kann die Playlist so konfiguriert werden, dass freie Player automatisch mit dem nächsten Element geladen werden. Die Arbeitsweise von mAirList entspricht dann der eines Cartplayers mit zugeordnetem Stack, über den mehrere Elemente nacheinander auf Knopfdruck abgefahren werden können.

3.2. Die Cartwall

Zusätzlich zu den Playern stellt mAirList eine Cartwall mit beliebig vielen Cartplayern (Slots) zur Verfügung. In jeden Player kann ein einzelnes Element geladen werden, zum Beispiel ein Jingle oder Drop. Die Belegung der Cartwall kann als Datei abgespeichert und später wieder geladen werden. Über eine Drop-Down-Menü kann auf häufig benötigte Cartwall-Belegungen schnell zugegriffen werden.

3.3. Die Browser

Um Elemente in die Playlists und die Cartwall einzufügen, stellt mAirList sogenannte Browser zur Verfügung, die sich am rechten Bildschirmrand befinden. Neue Elemente können dann mit der Maus in die gewünschte Playlist bzw. den Cartplayer gezogen werden.

Immer vorhanden ist der Archiv-Browser, in den alle bereits abgespielten Elemente aus den Playlists verschoben werden. Weiterhin lassen sich einblenden: Dateibrowser für Festplatten-Verzeichnisse, Dateibäume sowie vorhandene M3U-Playlisten. Außerdem stehen Browser für den Zugriff auf die eldoDB-Datenbank (siehe Abschnitt 3.5) zur Verfügung.

3.4. Faderstart

Alle Funktionen von mAirList (zum Beispiel die Steuerung der Player) werden über Faderstart-Befehle gesteuert, die mAirList auf verschiedenen Wegen generiert. Zum einen können für alle Funktionen Hotkeys definiert werden und diesen die gewünschten Befehle zugeordnet werden. Andererseits unterstützt mAirList die Kopplung an die Faderstart-Funktion von Mischpulten oder andere Fernsteuerungen über MIDI oder selbstgebaute Gameport-Interfaces.

3.5. Die Datenbankanbindung eldoDB

Im einfachsten Falle werden die Playlists und die Cartplayer über die Dateibrowser mit Elementen gefüllt. Für größere Musikarchive bietet mAirList eine Datenbankschnittstelle ("eldoDB") an. Die Daten werden auf einem SQL-Server (MySQL 4.0, PostgreSQL, ...) abgelegt. Die Verwaltung erfolgt mit dem sehr einfachen Programm `Musikliste.exe`, eine Nachfolge-Software ist zur Zeit in der Planung. Über die entsprechenden Browser kann dann nach Stücken aus der Datenbank gesucht oder mit der Musikliste erstellte Playlisten geladen werden.

4. Konfiguration

Alle Einstellungen von mAirList werden in der Datei `mAirList.ini` im Programmverzeichnis abgelegt. Zum Bearbeiten der Datei dient das Konfigurationsprogramm `mAirListConfig.exe`. Beim Start wird auf der linken Seite eine Reihe von Rubriken angezeigt, rechts können dann die Einstellungen verändert werden. Im folgenden werden die einzelnen Einstellungsmöglichkeiten erläutert.

4.1. Playlists

Das Herzstück von mAirList sind eine oder mehrere Playlists, in denen die zu spielenden Elemente geladen werden. Die Anzahl dieser Playlists ist prinzipiell nicht eingeschränkt, wenngleich die meisten Stationen je nach Anwendung mit einer bis drei Playlists auskommen sollten. Die Konfiguration der Playlists geschieht auf dieser Seite.

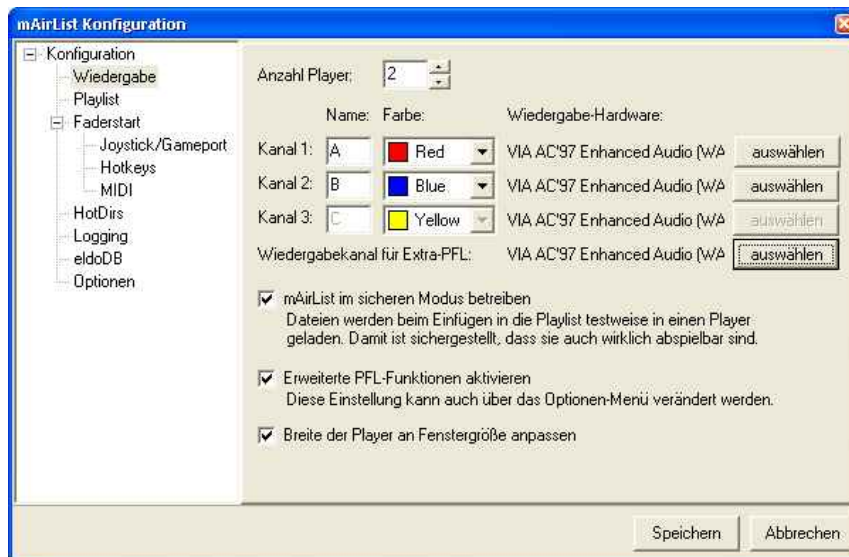


Abbildung 1: Das Konfigurationsprogramm mAirListConfig

Anzahl der Playlists Hier kann ausgewählt werden, wie viele getrennte Playlists auf dem Bildschirm erscheinen sollen. Standardwert ist eine Playlist. Zu jeder Playlist können dann unteren Bereich verschiedene Einstellungen vorgenommen werden.

Erweiterte Anzeige Ist diese Option aktiviert, werden die einzelnen Zeilen der Elemente höher dargestellt, so dass mehr Details angezeigt werden können.

Backtiming-Anzeige In diesem Modus zeigt mAirList automatisch die voraussichtlichen Startzeiten der Elemente in der Playlist an.

Fortschritt anzeigen Wenn diese Option aktiviert ist, zeigt mAirList unterhalb eines Elements, das gerade in einem Player abgespielt wird, einen weiteren Fortschrittsbalken an, wie er auch in den Player integriert ist. Dieser Balken ist üblicherweise größer als der des Players und deshalb besser abzulesen.

Automatisch laden mAirList verfügt über einen "intelligenten" Mechanismus, der die Player automatisch mit den Dateien aus der Playlist füllt. Wenn eine Datei gestoppt wird, wird sie automatisch aus dem Player herausgenommen und das entsprechende Element in den Archiv-Browser verschoben. Danach lädt mAirList automatisch das nächste (oberste freie) Elemente aus der Playlist in den Player. Um ein weiter unten stehendes Element in einen Player zu laden, zieht man es einfach mit der Maus nach oben.

Für Benutzer anderer Automationssoftware, wo die Player von Hand geladen werden müssen, mag dies auf den ersten Blick verwirrend erscheinen. Allerdings ist dies ein sehr praktisches Feature, insbesondere wenn man viele Dateien in kurzen Abständen abspielen will. Über die farbliche Markierung in der Playlist ist jederzeit klar, welches Element gerade in welchem Player geladen ist.

Will man auf diesen praktischen Mechanismus verzichten, kann man das automatische Laden der Player hier ausschalten. Ein Kompromiss könnte es sein, das automatische Laden aktiviert zu lassen und stattdessen den Split Mode mit je einer Playlist pro Player zu aktivieren.

Entladen bei STOP Ist diese Option aktiviert, wird jedes abgespielte Playlist-Element automatisch aus der Playlist entfernt, wenn der zugehörige Player ein STOP-Signal erhält.

Entladen bei EOF Ist diese Option aktiviert, wird jedes abgespielte Playlist-Element automatisch aus der Playlist entfernt, wenn der zugehörige Player zuende gelaufen ist.

Schriftart Hier kann die Schriftart eingestellt werden, die für die Playlist verwendet wird. Die Breite bzw. Höhe der Spalten und Zeilen passt mAirList automatisch an.

Anzahl Player für diese Playlist Hier kann eingestellt werden, wie viele Player dieser Playlist zugeordnet werden sollen. Die Gesamtzahl aller Player ergibt sich dann aus der Summe dieses Wertes aller Playlisten. Für jeden Player erscheint im Bereich unter dieser Option wieder eine Seite für die entsprechenden Einstellungen.

Name, Farbe Jedem Player kann ein Name (ein oder zwei Zeichen, z.B. "A", "B", "C", "1", "2", "3", ...) und eine Farbe zugewiesen werden. Die Farbe wird auch in der Playlist benutzt, um optisch zu kennzeichnen, in welchem Player ein bestimmtes Element geladen ist.

Soundkarte Durch Drücken des Buttons "Auswählen" erscheint ein Dialog, in dem die Soundkarte ausgewählt werden kann, auf der die Ausgabe dieses Players erfolgen soll. Außerdem kann die gewünschte Ausgaberoutine ausgewählt werden. mAirList stellt zur Zeit zwei verschiedene Ausgaberoutinen zur Verfügung:

- **BASS.DLL** – dies ist die empfohlene Einstellung. Bei BASS.DLL handelt es sich um eine Library zum Abspielen verschiedener Soundformate via DirectX. Unterstützt werden die Formate WAV, MP3, MP2 und OGG. Aller Voraussicht nach wird mAirList in Zukunft nur noch mit diesen Routinen arbeiten, daher ist die Benutzung der BASS.DLL empfohlen. Die jeweils aktuelle BASS.DLL ist in der mAirList-Distribution enthalten, kann aber auch unter <http://www.un4seen.com> heruntergeladen werden.

Mit BASS.DLL als Ausgaberoutine kann zusätzlich bestimmt werden, auf welchem Lautsprecherpaar die Ausgabe erfolgen soll. Moderne Soundkarten haben fast immer sechs (5.1) oder sogar acht (7.1) Ausspielkanäle. mAirList kann diese mit Hilfe der BASS.DLL einzeln ansprechen, so dass z.B. auf einer handelsüblichen 5.1-Soundkarte drei einzelne Stereopaare zur Verfügung stehen. Damit man die Lautsprecher auswählen kann, muss in der Systemsteuerung die entsprechende Lautsprecherkonfiguration (5.1 oder 7.1) ausgewählt werden. Sonst erscheint hier nur ein einziges Lautsprecherpaar.

- **DirectShow/MME (alt)** – dies sind die alten Routinen, die vor der Unterstützung der BASS.DLL verwendet wurden. Hier lassen sich alle Formate abspielen, die DirectShow unterstützt. Durch

Installation entsprechender Filter können also neue Formate hinzugefügt werden. Allerdings funktioniert die Ausgabe nur auf MME-konformen Soundkarten. Das bedeutet insbesondere, dass auf neuere Mehrkanal-Soundkarten, die keine einzelnen MME-Treiber für jeden Kanal mitbringen, nur ein Kanal zum Ausspielen verwendet werden kann.

PFL-Soundkarte Hier kann festgelegt werden, auf welcher Soundkarte und welchem Kanal die PFL-Ausgabe des Players erfolgen soll. Ist mAirList an ein echtes Mischpult mit Vorhörfunktion angeschlossen, sollte hier dieselbe Soundkarte wie oben ausgewählt werden. Soll man die PFL-Ausgabe auf einem eigenen Monitorkanal erfolgen, so ist die entsprechende Soundkarte hier auszuwählen.

EOF-Warnung Ist hier eine Zahl größer als null eingestellt, blinkt der Player die entsprechende Anzahl von Sekunden vor Ende des Elementes als Warnsignal.

4.2. Cartwall

mAirList beinhaltet eine Cartwall (Jingleplayer), die aus einer beliebigen Anzahl von Playern besteht. Diese können mit Elementen geladen und über Faderstart, Hotkeys oder Maus abgefahren werden.

Cartwall aktivieren Hier kann die Cartwall ein- oder ausgeschaltet werden.

Cartwall im eigenen Fenster anzeigen Ist diese Option aktiviert, so wird die Cartwall in einem eigenen Fenster angezeigt, das über den Button "Cartwall" im Hauptfenster sowie über Faderstart-Kommandos (s.u.) ein- bzw. ausgeblendet werden kann.

Anzahl Player Hier legt man fest, wie viele Player die Cartwall enthalten soll.

Auspielkanal, Vorhörkanal Genau wie bei den normalen Playern kann hier die zu verwendende Soundkarte (für das Ausspielen und für das Vorhören) ausgewählt werden.

EOF-Warnung Legt fest, ob und wie viele Sekunden vor Ende eines Elementes der Player als Warnung blinken soll.

Favoriten-Belegungen Die Belegung der Cartwall, also die Namen der gerade in den Playern geladenen Dateien, können in sogenannten "Belegungen" gespeichert werden. Hier können die Belegungen ausgewählt werden, die als Favoriten im Cartwall-Fenster erscheinen sollen.

Abgespielte Elemente in Logfiles schreiben Ist diese Option ausgewählt, werden auch die in der Cartwall abgespielten Elemente in die konfigurierten Logfiles (siehe Abschnitt 4.5) geschrieben.

4.3. Faderstart

mAirList unterstützt derzeit – neben der Mausbedienung – drei Möglichkeiten, die Player und weitere Funktionen zu steuern: Über Hotkeys, über den Gameport bzw. einen Joystick oder Gamepad und per MIDI. Alle diese Methoden können gleichzeitig und unabhängig voneinander betrieben werden. Zu jeder Methode gibt es eine Konfigurationsseite, auf der jedem Ereignis (Taste gedrückt/losgelassen, Joystick-Button gedrückt/losgelassen, MIDI-Event) ein Befehl zugeordnet werden kann. Zur Zeit stehen die folgenden Befehle zur Verfügung:

- `PLAYER x-y START` – startet den y-ten Player der x-ten Playlist.
- `PLAYER x-y PAUSE` – hält den y-ten Player der x-ten Playlist an. Der Player bleibt an der aktuellen Position stehen, mit `START` kann die Ausgabe fortgesetzt werden. Geht man in den PFL-Modus, springt der Player zurück an den eingestellten Cue-Punkt.
- `PLAYER x-y STOP` – stoppt den y-ten Player der x-ten Playlist. Je nach Konfiguration wird dann entweder das Element aus der Playlist entfernt oder der Player zurück an den Cue-Punkt gefahren.
- `PLAYER x-y START/STOP` – wenn Player y gerade nicht läuft, wird er gestartet, sonst gestoppt.
- `PLAYER x-y START/PAUSE` – wenn Player y gerade nicht läuft, wird er gestartet, sonst angehalten.
- `PLAYER x-y PAUSE/STOP` – wenn Player y gerade läuft, wird er angehalten. Ist er bereits zuende gelaufen (EOF), wird er gestoppt.
- `PLAYER x-y STOP` – wenn Player y gerade nicht läuft, wird er gestartet; wenn er läuft, wird er angehalten; ist er am EOF, wird er gestoppt.
- `PLAYER x-y PFL ON, OFF, ON/OFF` – startet oder stoppt die Vorhörfunktion, wie oben.
- `PFL PLAY, PAUSE, END MON` – entspricht den Buttons `PLAY/PAUSE/END MON` im erweiterten PFL-Dialog
- `PFL CUE PLUS, MINUS, SET, 0, TEST` – entspricht den Buttons `+/-/SET/0/TEST` der CUE-Einstellung im erweiterten PFL-Dialog.
- `PFL RAMP PLUS, MINUS, SET, 0, TEST; PFL OUTRO PLUS, MINUS, SET, 0, TEST; PFL FADEOUT PLUS, MINUS, SET, 0, TEST` – dasselbe für Ramp, Outro und Fadeout.
- `AUTOMATION x ON, OFF, ON/OFF` – aktiviert oder deaktiviert die Automation für Playlist x.
- `AUTOMATION x START, STOP, NEXT` – entspricht den Automations-Buttons von Playlist x.
- `CARTWALL SHOW, HIDE, SHOW/HIDE` – öffnet bzw. schließt das Cartwall-Fenster bzw. wechselt zwischen geöffnet und geschlossen (nur für den Fall, dass die Cartwall in einem eigenen Fenster ist).
- `CARTWALL NEXT SET, PREVIOUS SET` – lädt die nächste bzw. vorherige Belegung aus der Liste der definierten Favoriten.
- `CARTWALL STOP ALL` – stoppt alle Player in der Cartwall.
- `CARTWALL CLEAR ALL` – entlädt alle Player in der Cartwall.
- `CARTWALL x START` – startet Cartplayer x.
- `CARTWALL x STOP` – stoppt Cartplayer x.

- CARTWALL x START/STOP – startet Cartplayer x bzw. stoppt ihn, wenn er gerade läuft.
- CARTWALL x CLEAR – entlädt Cartplayer x.

4.3.1. Joystick/Gameport

mAirList kann die Buttons von Joysticks als Faderstart-Auslöser benutzen. Damit ist es möglich, einen Joystick als Interface zwischen Rechner und Mischpult zu benutzen, entweder durch Ausschlaten eines billigen (USB-)Gamepads (Faderstart-Ausgänge des Mischpultes werden an die ehemaligen Button-Kontakte angelötet) oder durch direktes Anschließen an den Gameport (wobei üblicherweise einige Widerstände eingebaut werden müssen, damit Windows die Schaltung als Joystick erkennt).

Das Konfigurationsprogramm zeigt hier alle in der Systemsteuerung installierten Joysticks an. Zu jedem Button dieser Joysticks kann festgelegt werden, welche Aktion jeweils beim Drücken bzw. beim Loslassen des Buttons ausgeführt werden soll.

4.3.2. Hotkeys

Auf dieser Seite können die o.g. Befehle beliebigen Tasten(kombinationen) zugewiesen werden. Dazu zunächst das Feld "Taste" aktivieren, die gewünschte Tastenkombination drücken, dann eine Aktion auswählen und auf "Hinzufügen" klicken.

Die Hotkeys sind systemweit gültig, reagieren also auch, wenn mAirList gerade nicht im Vordergrund sein sollte. Allerdings werden die Tasten dann nicht an das Programm im Vordergrund weitergereicht. Es sollten also nur Tasten(kombinationen) verwendet werden, die von anderen Anwendungen nicht benötigt werden. Die PFL-Hotkeys werden nur dann aktiviert, wenn tatsächlich gerade ein erweiterter PFL-Dialog angezeigt wird.

4.3.3. MIDI

Neben Joysticks und Hotkeys bietet mAirList zusätzlich die Fernsteuerung per MIDI an. Dazu kann für jedes MIDI-Event eine Aktion definiert werden. Das Konfigurationsprogramm zeichnet die MIDI-Events automatisch auf und zeigt die drei Parameter Status, Data1 und Data2 an. Für Data2 kann zusätzlich festgelegt werden, ob die Aktion ausgeführt werden soll, wenn Data2 kleiner, größer oder gleich dem angegebenen Wert ist. Damit lassen sich z.B. anschlagdynamische Keyboards zum Fernsteuern verwenden, die je nach Stärke des Anschlages einen anderen Data2-Wert schicken.

4.4. HotDirs

Hier können – Zeile für Zeile – Verzeichnisse angegeben werden, die im "Hinzufügen"-Menü über den Browsern (erreichbar durch Klicken auf den kleinen Pfeil) als Favoriten zur Verfügung stehen.

4.5. Logging

mAirList ist in der Lage, Log-Dateien über alle abgespielten Titel zu schreiben. Diese können dann von anderer Software z.B. für Online-Playlisten ausgewertet werden.

Dabei ist das Format sowohl des Dateinamens als auch der Logfile-Einträge frei wählbar. Zum Beispiel können durch Setzen des Dateinamens auf `C:\mAirList\log\%Y-%M-%D.LOG` tägliche rotierende Dateien geschrieben werden, in diesem Fall 2004-01-19.LOG, 2004-01-20.LOG usw.

Die möglichen Formatvariablen werden in diesem Dialogfeld angezeigt.

Außerdem kann ein zusätzliches Logfile am *Ende* eines jeden Titels geschrieben werden. Hier steht zusätzlich die Variable `%d` zur Verfügung, die die tatsächlich ausgespielte Länge des Elements (in Sekunden) angibt.

4.6. eldoDB

mAirList besitzt einer Schnittstelle zu eldoDB, einer SQL-basierten Musikdatenbank, die gleichzeitig mit mAirList entwickelt wird. Um die Schnittstelle zu aktivieren, müssen hier die Parameter zum Zugriff auf die Datenbank festgelegt werden (Host, User, Passwort und Name der Datenbank).

Datenbank-Typ Die Datenbank-Anbindung wurde über die Zeos-Library (www.zeoslib.net) realisiert. Dadurch arbeitet mAirList theoretisch mit allen von Zeos unterstützten Datenbanken zusammen. Getestet wurde mAirList aber nur mit MySQL.

Für den Betrieb mit einer Datenbank ist in den meisten Fällen eine zum Datenbank-Treiber passende DLL erforderlich (z.B. `libmysql323.dll` für `mysql-3.23`). Diese DLLs sind im "ZeosDBO"-Paket enthalten, das man auf der Zeos-Homepage herunterladen kann.

Aufgrund von Lizenzbeschränkungen darf mAirList nicht zusammen mit der MySQL-Client-Library (`libmysql.dll`) ab Version 4.0 betrieben werden.²

Logging in die Datenbank Ist eine Datenbank-Anbindung konfiguriert, so kann mAirList auch alle gestarteten Elemente in eine Datenbank-Tabelle protokollieren. Dazu muss in das entsprechende Feld ein SQL-INSERT-Befehl eingegeben werden, der die Zeile zu der gewünschten Tabelle hinzufügt, z.B.

```
INSERT INTO playlist VALUES(NOW(), "%a", "%b")
```

Es können alle Makros verwendet werden, die auch für das normale Logging in Dateien zur Verfügung stellen. Die eldoDB bringt in der Standardkonfiguration keine vorgefertigte Tabelle für das Logging mit, diese muss sich jeder selbst anlegen.

²Bis einschließlich Version 3.23 unterliegt die Client-Library von MySQL der GNU Lesser Public License (LGPL), die es erlaubt, sie mit Closed-Source-Projekten zusammen einzusetzen. Seit der Version 4.0 untersteht die Library der GPL, womit sie nur zusammen mit Open-Source-Projekten eingesetzt werden darf. Andernfalls ist eine kommerzielle Lizenz von MySQL erforderlich, die der Autor aber nicht besitzt.

4.7. Fernsteuerung

mAirList besitzt eine RPC-/Web-Services-Schnittstelle, über die eine Instanz über das Netzwerk oder von anderen Programmen auf demselben Rechner ferngesteuert werden kann. Auf dieser Konfigurationsseite kann die Schnittstelle aktiviert und der HTTP-Port (Standard: 9300) eingestellt werden, auf dem mit mAirList kommuniziert werden kann.

Die Schnittstelle wurde mit RemObjects SDK (siehe www.remobjects.com) realisiert. Neben dem RemObjects-eigenen BIN-Format (URL: `http://<host>:<port>/BIN`) unterstützt mAirList auch SOAP-Aufrufe (über die URL `http://<host>:<port>/SOAP`).

Die Fernsteuerung befindet sich zur Zeit noch in der Entwicklung. Erste Anwendung ist das Programm `mAirListComm.exe` (herunterzuladen auf der mAirList-Homepage), mit dem sich beliebige Faderstart-Befehle (s. Abschnitt 4.3) an eine laufende mAirList-Instanz schicken lassen, entweder über das Netzwerk oder auf dem lokalen Rechner.

4.8. Optionen

Automatisches Fenster-Layout Ist diese Option aktiv, ordnet mAirList die Playlists und Player automatisch im Programmfenster an: Unten nebeneinander die Player aller Playlists, darüber links übereinander alle Playlists sowie eventuell die Cartwall, rechts der Browser. Alternativ kann das Fenster-Layout auch von Hand festgelegt werden, über die Datei `layout.ini` (s. Anhang A).

Alte Browser-Darstellung Hier kann eingestellt werden, ob die Browser in der neuen ("Outlook"-ähnlichen) oder alten Optik erscheinen sollen.

Bei mehreren Monitoren: mAirList starten auf Bildschirm ... Wenn der Rechner mehrere Bildschirme hat, kann hier festgelegt werden, auf welchem der Monitore mAirList angezeigt werden soll. So kann man zum Beispiel zwei Kopien von mAirList (z.B. eine für Musik, eine für Jingles) auf demselben Rechner (mit DualHead-Grafikkarte) betreiben und von vornherein festlegen, welche Kopie auf welchem Bildschirm laufen soll.

Erweitertes PFL Ist diese Option aktiviert, wird während des PFL ein Fenster eingeblendet, über das die Cue/Ramp/Outro/Fadeout-Punkte verändert werden können. Der Erweiterte PFL-Modus kann auch später noch über das Options-Menü von mAirList (de-)aktiviert werden.

PFL OFF bei START Einige Mischpulte mit Faderstart-Funktion sind in der Lage, beim Aktivieren des Kanals gleichzeitig das PFL mit auszuschalten. Diese beiden Signale (PFL OFF, START) kommen in mAirList eventuell in der falschen Reihenfolge an (START vor PFL OFF), so dass mAirList den Player nicht startet, weil er ja noch im PFL-Modus ist. Wenn man diese Option aktiviert, wird jeder Player softwareseitig automatisch aus dem PFL-Modus genommen, wenn er ein START-Signal bekommt. Dies gilt auch für den Betrieb über Hotkeys und MIDI.

Vorhörkanal für Extra-PFL Die normale Vorhörfunktion (PFL) von mAirList spielt das Element immer auf der Soundkarte aus, die auch für die normale Wiedergabe des Players verwendet wird. Damit ist eine

nahtlose Kopplung an die PFL-Funktion von Mischpulten möglich. Zusätzlich kann hier ein weiterer Ausgabekanal definiert werden, auf dem Elemente im sogenannten Extra-PFL-Modus ausgegeben werden, unabhängig davon, ob sie gerade in einem Player geladen sind oder nicht. Diese Funktion erreicht man über das Kontextmenü, das man durch Klicken mit der rechten Maustaste auf ein Element erhält.

Pufferlänge (BASS.DLL) Hier kann die Länge des Ausgabepuffers eingestellt werden, den die BASS.DLL verwendet. Standardwert sind 500ms. Bei hoher Systemlast oder schlechten Netzwerkverbindungen kann man höhere Werte ausprobieren.

Überblendzeit (Automation) Dieser Wert legt fest, wie lange im Automations-Betrieb der letzte Titel ausfaden soll, während der neue bereits läuft.

Meldungen anzeigen in Sprache ... Wenn Dateien zur Übersetzung von mAirList in andere Sprachen installiert sind, können diese hier aktiviert werden.

5. mAirList benutzen

5.1. Das Hauptfenster

Das mAirList-Hauptfenster gliedert sich je nach Konfiguration in drei oder vier Teile:

- die Playlist(en),
- die Browser,
- die Player,
- ggf. die Cartwall.

5.1.1. Die Playlist

In der Playlist sind alle Dateien aufgelistet, die zum Abspielen vorgesehen sind. Im Split Mode ist nicht nur eine Playlist sondern mehrere sichtbar, je nach Konfiguration. Wie genau man mit der Playlist umgeht, steht in Abschnitt 5.2.

5.1.2. Die Browser

Im rechten Bereich des mAirList-Fensters können sogenannte "Browser" eingeblendet werden. Über diese können beispielsweise Verzeichnisse eingeblendet werden, aus denen dann neue Dateien in die Playlist gezogen werden können. Welche Browser zur Verfügung stehen und wie diese funktionieren, wird in Abschnitt 5.3 erklärt.

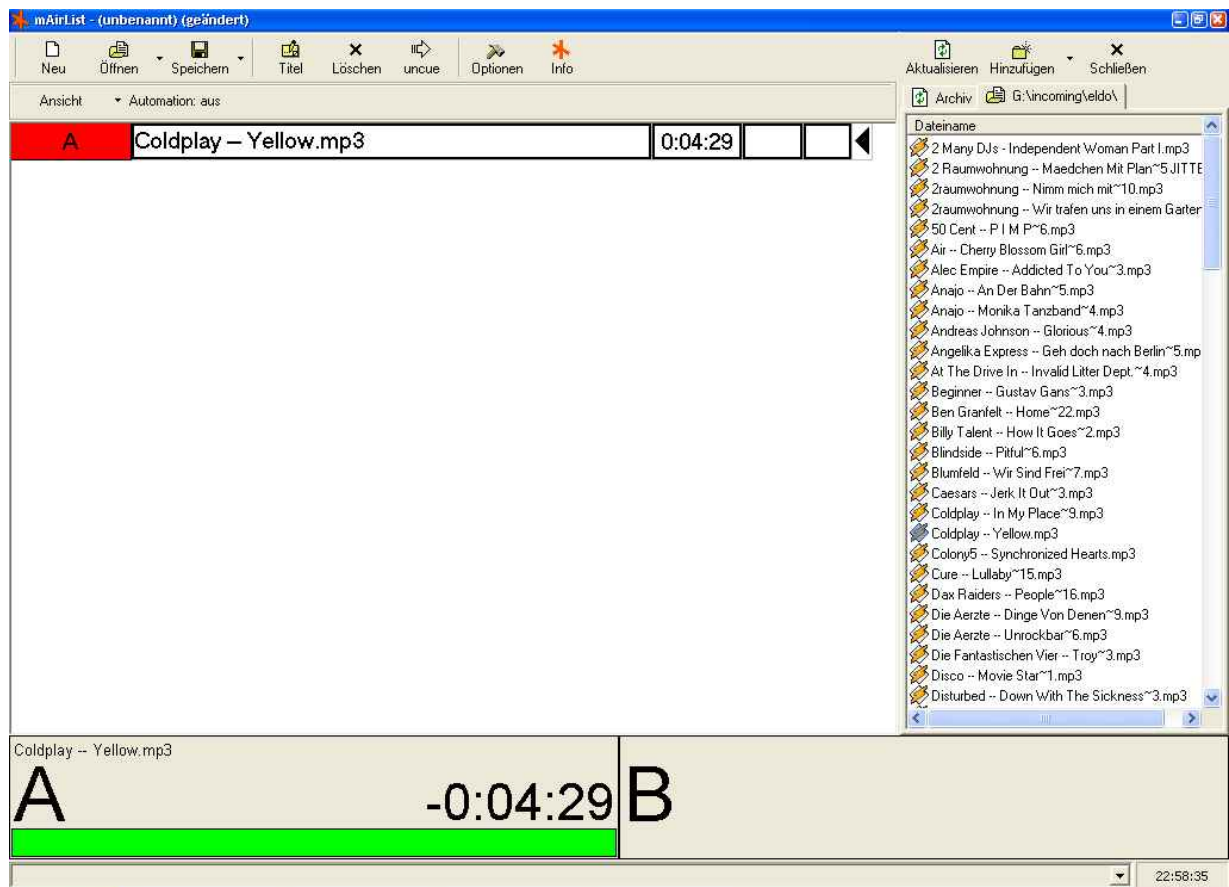


Abbildung 2: Das mAirList-Hauptfenster

5.1.3. Die Cartwall

Die Cartwall enthält die Cartplayer, über die beispielsweise Jingles abgefahren werden können. Die Cartwall ist entweder in das Hauptfenster integriert (zwischen den Playlisten und den Playern) oder wird in einem eigenen Fenster dargestellt, das über die konfigurierten Faderstart-Kommandos ein- und ausgeblendet werden kann. Die Bedienung der Cartwall ist in Abschnitt 5.5 beschrieben.

5.2. Arbeiten mit der Playlist

5.2.1. Einfügen von Elementen

Neue Elemente (Dateien) können ganz einfach per Drag & Drop in die Playlist gezogen werden, und zwar entweder aus dem Windows Explorer oder aus einem der Browser. Beim Einfügen in die Playlist wird die Abspielzeit der Datei ermittelt. Fehlerhafte Dateien werden erst beim Laden in einen Player erkannt.

5.2.2. Verschieben von Elementen

Elemente können einfach mit der Maus von oben nach unten oder umgekehrt verschoben werden. Dazu muss das Element in der ersten Spalte (ganz links) angeklickt werden und dann mit gedrückter Maustaste verschoben werden.

5.2.3. Löschen von Elementen

Es gibt zwei Arten, auf die ein Element aus der Playlist entfernt werden kann:

1. Verschieben in den Archiv-Browser (s. Abschnitt 5.3.1)
2. direktes Löschen

Dazu dienen die beiden Buttons "uncue" und "löschen" in der Symbolleiste.

5.2.4. Playlist-Optionen

Das Kontextmenü einer Playlist (erreichbar über die rechte Maustaste) enthält einige Optionen, die insbesondere die Darstellung der Playlist betreffen.

Eigenschaften Über diese Funktion wird ein Fenster geöffnet, in dem das aktuelle Element bearbeitet werden kann, also insbesondere Titel und Interpret. Außerdem können die eingestellten Cue-Punkte in die Datenbank zurückgeschrieben oder im ID3-Tag einer MP3-Datei gespeichert werden. Ferner kann hier eine Fix-Zeit eingestellt werden, die für die Berechnung des Backtimings herangezogen wird (aber für die Automation derzeit ohne Bedeutung ist).

PFL Hierüber wird die Extra-PFL-Funktion für das bearbeitete Element gestartet. Dazu öffnet sich ein Fenster mit einem zusätzlichen Player und denen aus dem erweiterten PFL-Modus (s. Abschnitt 5.4.4 bekannten Dialogelementen. Die Wiedergabe erfolgt auf der im Konfigurationsprogramm eingestellten Soundkarte (s. Abschnitt 4.8).

Erweiterte Anzeige Schaltet zwischen einfacher und erweiterter Anzeige um (s. Abschnitt 4.1).

Backtiming-Anzeige Diese Funktion aktiviert die Backtiming-Anzeige (s. Abschnitt 4.1).

Player-Fortschritt anzeigen Diese Option aktiviert den Fortschrittsbalken innerhalb der Playlist (s. Abschnitt 4.1).

Player automatisch laden Hiermit kann festgelegt werden, ob mAirList das Laden/Entladen von Playern automatisch vornehmen soll (s. Abschnitt 4.1).

Automation Dieser Menüpunkt aktiviert die Automation für diese Playlist. Mehr dazu steht in Abschnitt 5.8.

5.3. Arbeiten mit Browsern

Die Browser bilden das Herzstück von mAirList, wenn es darum geht, neue Dateien in die Playlist einzufügen. Über die Buttons "Hinzufügen" und "Schließen" können jederzeit neue Browser angezeigt oder ausgeblendet werden. Der Button "Aktualisieren" aktualisiert den Inhalt eines Browsers. Elemente aus den Browsern können per Drag & Drop in die Playlist gezogen werden.

Es gibt verschiedene Arten von Browsern, die im folgenden kurz erklärt werden sollen. Ein einfacher Klick auf den "Hinzufügen"-Button öffnet einen Verzeichnisbrowser, alle anderen Arten von Browsern (außer dem Archiv-Browser, der immer angezeigt wird) können über das Menü aufgerufen werden, das erscheint, wenn man auf den Pfeil neben "Hinzufügen" klickt.

5.3.1. Der Archiv-Browser

Der Archiv-Browser wird immer angezeigt (kann also nicht geschlossen werden). Wenn ein Element abgespielt und automatisch aus der Playlist entfernt wurde, landet es danach im Archiv. Das Archiv enthält also alle bislang abgespielten Dateien. Dadurch ist ein schneller Zugriff möglich, wenn eine Datei versehentlich aus der Playlist gelöscht wurde oder nicht.

5.3.2. Der Verzeichnis-Browser

Der Verzeichnis-Browser listet den Inhalt eines Verzeichnisses auf, wobei nur Sounddateien (MP3, MP2, WAV, OGG) angezeigt werden. Welches Verzeichnis der Browser darstellen soll, legt man beim Hinzufügen des Browsers fest.

5.3.3. Der Verzeichnisbaum-Browser

Dieser Browser arbeitet wie der Verzeichnis-Browser, wobei allerdings ein ganzer Verzeichnisbaum dargestellt wird, ähnlich wie in der linken Hälfte des Windows-Explorers. Verzeichnisse können über die kleinen "+"- und "-" -Buttons auf und zugeklappt werden. Es wird immer der beim Hinzufügen festgelegte Ordner und alle Unterverzeichnisse angezeigt.

5.3.4. Der Playlist-Browser

In diesem Browser können Playlisten im Winamp- (.m3u) bzw. Raudga- (.alb, .mpl) Format angezeigt werden.

5.3.5. Der Datenbank-Such-Browser

Falls aktiv, dient dieser Browser zum Suchen in der eldoDB. Es kann entweder nach Interpret/Titel gesucht oder eine komplette Rotation angezeigt werden.

5.3.6. Der Datenbank-Playlist-Browser

Dieser Browser dient zum Anzeigen einer in der eldoDB angelegten Playlist. Elemente der eldoDB-Playlist, zu denen kein MP3-Dateiname festgelegt wurde, werden automatisch ausgeblendet.

5.3.7. Der Uhr-Browser

Dieser Browser stellt eine große Uhr da. Vielleicht für den einen oder anderen ganz nützlich ;-)

5.4. Arbeiten mit den Playern

Am unteren Fensterrand befinden sich die (bis zu drei) Player, wie im Konfigurationsprogramm eingestellt.

5.4.1. Player laden

Normalerweise ist mAirList so konfiguriert, dass die Player automatisch mit den obersten Elementen in der Playlist geladen werden. Ist diese Option deaktiviert – entweder über das Konfigurationsprogramm (s. Abschnitt 4.1) oder über das Kontextmenü der Playlist – muss man die Elemente von Hand per Drag&Drop in den gewünschten Player ziehen.

5.4.2. Spielzeit- und Backtiming-Anzeige

Im Normalfall zeigt der Player die noch verbleibende Restzeit der Datei an. Durch einen Klick auf die Zeit kann zwischen Restzeit und aktueller Position umgestellt werden. Ein Doppelklick zeigt die Uhrzeit an, zu der die Datei zuende ist (bzw. zuende wäre, würde sie in diesem Moment gestartet).

5.4.3. Starten und Stoppen von Playern

Die Player werden normalerweise über die Hotkeys bzw. den Faderstart angesteuert, wie im Konfigurationsprogramm eingestellt. Die Bedienung mit der Maus ist zur Zeit nicht möglich.

5.4.4. Der erweiterte PFL-Modus

Wenn im Konfigurationsprogramm bzw. im Options-Menü so eingestellt, erscheint im PFL-Modus automatisch ein Fenster, über das die zu einem Element gehörigen Zeiten (Cue, Ramp, Outro, Fadeout) festgelegt werden können.

Die Cue-Zeit legt fest, ab welchem Zeitpunkt der Titel abgespielt werden soll. Damit lassen sich z.B. lange Intros von Liedern abschneiden.

Die Ramp kann auf den Zeitpunkt gesetzt werden, an dem der Gesang des Liedes anfängt. Der Moderator hat also zwischen Cue- und Ramp-Punkt Zeit zum Sprechen. Die Ramp wird dem Moderator auch graphisch angezeigt (s.u.).



Abbildung 3: Der erweiterte PFL-Modus

Outro ist zur Zeit noch ohne Funktion.

Fadeout legt – für die Automation – fest, an welchem Punkt der Titel ausgeblendet und der nächste gestartet werden soll.

PAUSE und PLAY Diese beiden Buttons halten die Wiedergabe an bzw. starten sie wieder.

END MON Dieser Button spielt die letzten 10 Sekunden des Titels ab.

SET Dieser Button setzt die zugehörige Zeit (Cue, Ramp, Outro oder Fadeout) auf den aktuellen Zeitpunkt. Damit kann z.B. ein Cue-Punkt zunächst grob eingestellt werden.

0 Dieser Button setzt die zugehörige Zeit wieder auf Null zurück. Bei Outro und Fadeout bedeutet 0, dass keine Zeit festgelegt wurde.

TEST Dieser Button testet die entsprechende Zeit, der Player springt also zu dem festgelegten Punkt.

+ und - Über diese Buttons kann die Zeit in 1/100-Sekunden-Schritten verstellt werden.

Bedienung mit der Maus Die Zeiten können auch mit dem Mousrad verändert werden, ein Schritt des Mousrades entspricht 1/100 Sekunden. Standardmäßig wird die Cue-Zeit verändert, ansonsten die Zeit, für die zuletzt ein Button gedrückt wurde.

5.4.5. Visualisierung der Ramp

Die Ramp ist im Radio-Jargon der (instrumentale) Beginn eines Liedes, auf den der Moderator noch sprechen kann, ehe der Gesang beginnt. mAirList unterstützt den Moderator dabei, indem die Ramp-Zeit – sofern sie über das erweiterte PFL eingestellt wurde – graphisch visualisiert wird. Innerhalb der Ramp teilt sich der sonst grüne Balken des Players. Die obere Hälfte wird rot und läuft zusammen mit der Ramp langsam runter. Außerdem wird die verbleibende Restzeit der Ramp im Player angezeigt.

5.5. Arbeiten mit der Cartwall

[...]

5.6. Öffnen und Speichern von Dateien

mAirList besitzt ein eigenes XML-basiertes Dateiformat, in dem der Zustand des Programms gespeichert und später wiederhergestellt werden kann. Dazu zählen

- Inhalt der Playlisten,
- Inhalt der Cartwall,
- Art und Inhalt der geöffneten Browser.

Die Endung der mAirList-Dateien lautet “.mpl”, wie “**m**AirList **P**laylist”³.

5.6.1. Vorlagen

Eine mAirList-Datei kann auch als Vorlage gespeichert werden. Das Format ist dasselbe, allerdings lautet die Endung dann “.mlt” (“mAirList Template”). Wird eine solche Datei später geöffnet, “vergisst” mAirList den Dateinamen, so dass der einfache “Speichern”-Button nicht funktioniert. Vorlagen sind so also gegen das versehentliche Überschreiben gesichert. Soll eine Vorlage überschrieben werden, muss das explizit über “Speichern unter ...” geschehen.

5.6.2. Standard-Vorlage

Beim Start sucht mAirList im Programmverzeichnis nach einer Datei `standard.mlt`. Ist diese vorhanden, wird sie geöffnet. So lassen sich Default-Einstellungen bei jedem Programmstart automatisch laden.

5.6.3. Import von Fremdformaten

mAirList kann folgende Dateiformate importieren:

- .xpl (mAirList 1.0)

³“.mpl” wäre auch denkbar gewesen, hätte mich aber zu sehr an nervende Vertreter erinnert.

- .m3u (Winamp)

Der Import geschieht ganz normal über den Öffnen-Dialog.

5.6.4. Anhängen von Dateien

Dieser Befehl verbirgt sich hinter dem kleinen Pfeil neben dem “Öffnen”-Button. Über diese Funktion können Dateien an die aktuelle angehängt werden, das heißt, die Elemente der Playlisten werden an die aktuellen Playlisten angehängt.

5.7. Tastaturbedienung

Die wichtigsten Funktionen von mAirList lassen sich auch über die Tastatur aufrufen:

Strg-O	Playlist öffnen
Strg-A	Playlist anhängen
Strg-S	Playlist speichern
Strg-T	Playlist unter anderem Namen speichern
Strg-E	Element bearbeiten
Entf	Element ins Archiv verschieben
Shift-Entf	Element aus Playlist löschen

Darüber hinaus können mit dem Konfigurationsprogramm Hotkeys für die Player-Funktionen frei definiert werden (s. Abschnitt 4.3).

5.8. Die Automation

[ToDo]

A. Fenster-Layout anpassen

Wenn im Konfigurationsprogramm die Option “Automatisches Layout” ausgeschaltet ist, können die graphischen Bestandteile von mAirList (Player, Playlists, Browser, ...) pixelgenau nach den eigenen Vorstellungen platziert sowie optisch angepasst werden. Dies geschieht über eine weitere Konfigurationsdatei, `layout.ini`. Durch Austausch dieser Datei kann mAirList schnell optisch an verschiedene Gegebenheiten angepasst werden.

A.1. Aufbau von `layout.ini`

Die Datei `layout.ini` ist wie eine herkömmliche Windows-INI-Datei aufgebaut. Zu jedem graphischen Element gibt es einen Abschnitt, der durch eckige Klammern und den Namen des Elements eingeleitet wird, z.B. `Playlist0`. Je nach im Konfigurationsprogramm gewählten Optionen stehen die folgenden Abschnitte zur Verfügung:

- [Toolbar] – die Toolbar mit den Buttons “Neu”, “Öffnen”, “Speichern” usw.
- [Browser] – der Datei-Browser, incl. Toolbar “Hinzufügen”, “Aktualisieren” und “Schließen”.
- [Cartwall] – die Cartwall bzw. das Cartwall-Fenster, sofern entsprechend konfiguriert.
- [Playlist x] – die x -te Playlist (wobei die erste Playlist die Nummer 0 hat, die zweite die Nummer 1 usw.)
- [Player x _ y] – der y -te Player der x -ten Playlist (wobei die erste Player wieder die Nummer 0 hat, der zweite die Nummer 1 usw.)

A.2. Beschreibung der Parameter

A.2.1. Gemeinsame Parameter aller Elemente

Jedes graphische Element verfügt über die folgenden Parameter:

- Left – X-Koordinate der linken oberen Ecke
- Top – Y-Koordinate der rechten oberen Ecke
- Width – Breite des Elements
- Height – Höhe des Elements

Ist die Cartwall so konfiguriert, dass sie in einem eigenen Fenster erscheint, lassen sich über diese Parameter Größe und Position dieses Cartwall-Fensters festlegen. Ansonsten beeinflussen sie die Position des Cartwall-Bereichs im Hauptfenster.

A.2.2. Zusätzliche Parameter von [Playlist x]

Für jede Playlist lassen sich weiterhin die folgenden Werte einstellen:

- FontName – zu verwendende Schriftart.
- FontSize – zu verwendende Schriftgröße.
- FontStyle – zu verwendender Schriftstil. Dieser ergibt sich durch Addition der folgenden Werte:
 - 1 fett
 - 2 kursiv
 - 4 unterstrichen
 - 8 durchgestrichen
- PlayerNameFontName, PlayerNameFontSize, PlayerNameFontStyle – wie oben, Schriftart für den Namen und Zustand des Players in der linken Spalte.
- BacktimingFontName, BacktimingFontSize, BacktimingFontStyle – wie oben, Schriftart für die Backtiming-Anzeige.
- BackgroundColor – Hintergrundfarbe.

- `RowColor` – Standardfarbe eines Playlist-Eintrages.
- `SelectedRowColor` – Farbe des ausgewählten Playlist-Eintrages.
- `LoadedRowColor`, `PlayingRowColor`, `PFLRowColor`, `PausedRowColor`, `EOFRowColor` – Farbe eines Playlist-Eintrages in Abhängigkeit vom Zustand: In Player geladend, wird gerade abgespielt, PFL, EOF, angehalten, Fehler.
- `SelectedLoadedRowColor`, `SelectedPlayingRowColor`, `SelectedPFLRowColor`, `SelectedPausedRowColor`, `SelectedEOFRowColor` – wie oben, für Einträge, die zusätzlich noch mit der Maus ausgewählt sind.
- `BorderColor` – Farbe des Rahmens um die Playlisteinträge herum.
- `BorderWidth` – Breite des Rahmens.
- `ProgressBarHeight` – Höhe des Fortschrittsbalkens.
- `ProgressBarBorderColor` – Farbe des Fortschrittsbalkens: Rahmen.
- `ProgressBarElapsedColor` – Farbe des Fortschrittsbalkens: abgelaufene Zeit.
- `ProgressBarRemainColor` – Farbe des Fortschrittsbalkens: verbleibende Zeit.
- `ProgressBarRampColor` – Farbe des Fortschrittsbalkens: Ramp.
- `ColWidths` – Spaltenbreiten in Pixel. Als Wert werden sechs, durch Komma getrennte Zahlen erwartet. Wird als Spaltenbreite 0 Pixel angegeben, wird die Spalte ausgeblendet. Beispiel: 150,400,0,0,0,30

Alle Farben können als HTML-Farbcodes (mit führender Raute) angegeben werden, z.B. #F0F0F0.

A.2.3. Zusätzliche Parameter von `[Playerx_y]`

Für jeden Player lassen sich weiterhin die folgenden Werte einstellen:

- `TitleFontName`, `TitleFontSize`, `TitleFontStyle` – wie oben. Definiert die Schriftart, die für den Namen des gerade geladenen Elements verwendet werden soll.
- `NameFontName`, `NameFontSize`, `NameFontStyle` – wie oben. Definiert die Schriftart, die für den Namen des Players verwendet wird.
- `StateFontName`, `StateFontSize`, `StateFontStyle` – wie oben. Definiert die Schriftart, die für den aktuellen Status des Players ("ON AIR", "PFL", ...) verwendet wird.
- `TimeFontName`, `TimeFontSize`, `TimeFontStyle` – wie oben. Definiert die Schriftart, die für die Zeitanzeige des Players verwendet wird.
- `ProgressBarHeight`, `ProgressBarBorderColor`, `ProgressBarElapsedColor`, `ProgressBarRemainColor`, `ProgressBarRampColor` – wie oben.
- `ProgressBarMargin` – definiert den Abstand zwischen Laufbalken und der darüberliegenden Schrift.

- EmptyColor, LoadedColor, PlayingColor, FadeColor, EOFColor, PFLColor, PausedColor, ErrorColor – legt die Hintergrundfarbe des Players in den verschiedenen Zuständen fest: Leer, geladen, spielend, ausblendend, EOF, PFL, angehalten, Fehler.
- FlashEOFColor – legt die Farbe fest, in der der Player (im Wechsel mit PlayingColor oder FadeColor) blinkt, um auf das Ende des Elements hinzuweisen (sofern die entsprechende Option aktiviert ist).

A.2.4. Zusätzliche Parameter von [CartWall]

Im Abschnitt [Cartwall] können dieselben Parameter benutzt werden wie im Abschnitt [Player x_y], um die Darstellung der Cartplayer zu beeinflussen. Die Einstellungen wirken sich auf alle Player gleichzeitig aus.

B. Kommandozeilen-Parameter

mAirList kennt zur Zeit die folgenden Kommandozeilen-Parameter:

- c <Dateiname> Benutzt <Dateiname> als Konfigurationsdatei anstelle von mAirList.ini
- l <Dateiname> Benutzt <Dateiname> als Layout-Konfigurationsdatei anstelle von layout.ini
- o <Dateiname> Öffnet bei Programmstart die Datei <Dateiname>