

# **Eagleplayer**

Eagleeye & DEFECT

**COLLABORATORS**

	<i>TITLE :</i> Eagleplayer		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Eagleeye & DEFECT	August 24, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1 Eagleplayer</b>	<b>1</b>
1.1 Eagleplayer 2.00 Dokumentation	1
1.2 Einleitung & Rechtliches	1
1.3 Tips & Tricks	2
1.4 Erste Schritte in der Benutzung	3
1.5 Probleme / Fehler im Programm	3
1.6 Copyright und Vertriebsbedingungen	4
1.7 Distribution	6
1.8 Was ist und kann der Eagleplayer ?	6
1.9 Entstehung des Eagleplayers	8
1.10 Future	9
1.11 Updates	9
1.12 Update-Service	10
1.13 Fehlerbeschreibung	10
1.14 Keyfilebenutzung	11
1.15 Systemanforderungen	11
1.16 Registrierung	12
1.17 Installation auf der Festplatte,...	13
1.18 Installation auf der Festplatte,...	13
1.19 Starten des Eagleplayers	14
1.20 Start von der Workbench aus	15
1.21 Das Konfigurationsfile	15
1.22 Die PlayerBatch-Datei	16
1.23 Benutzung, Steuerung und Konfiguration	16
1.24 Das Hauptfenster	17
1.25 Steuerung über die Pulldown-Menüs	18
1.26 Die AppWindow-Funktion	21
1.27 pubscreens	22
1.28 Commodity-Funktionen	22
1.29 Interne Depacker	23

---

1.30	Einführung und Beschreibung des Arexx-Portes . . . . .	24
1.31	Arexx-Kommandoübersicht . . . . .	25
1.32	Anpassung an Directory Opus 4.x . . . . .	28
1.33	Status . . . . .	29
1.34	aminetcd . . . . .	34
1.35	AboutEP . . . . .	35
1.36	AboutModule . . . . .	35
1.37	Addiere Listeneintrag . . . . .	35
1.38	Addiere Liste . . . . .	35
1.39	Allokiere Audiokanäle . . . . .	36
1.40	ASCEngine . . . . .	36
1.41	automaticsave . . . . .	36
1.42	autosubsong . . . . .	37
1.43	autopassword . . . . .	37
1.44	balance . . . . .	38
1.45	bigmodule . . . . .	39
1.46	calcduration . . . . .	39
1.47	ChipRAMAmplifier . . . . .	40
1.48	crunchmode . . . . .	40
1.49	Popup . . . . .	41
1.50	CX_Popkey . . . . .	41
1.51	CX_Priority . . . . .	42
1.52	config . . . . .	42
1.53	defaultspeed . . . . .	42
1.54	dbfloops . . . . .	42
1.55	dirjump . . . . .	43
1.56	DeleteFile . . . . .	44
1.57	documentation . . . . .	45
1.58	Eject . . . . .	45
1.59	epdir . . . . .	45
1.60	extractfirstmodule . . . . .	46
1.61	FadeIn . . . . .	47
1.62	FadeOut . . . . .	47
1.63	FastPlay . . . . .	48
1.64	fasterspeed . . . . .	48
1.65	filerequestermode . . . . .	48
1.66	Filter . . . . .	49
1.67	Help . . . . .	49
1.68	Hide . . . . .	50

---

---

1.69	hideall	50
1.70	Iconify	51
1.71	iconifymode	51
1.72	loadalways	52
1.73	loadfast	52
1.74	loadbefore	53
1.75	loaddir	54
1.76	Mastervolume	54
1.77	Menufont	55
1.78	mintimeout	55
1.79	modulefault	56
1.80	Modulesfont	56
1.81	NextPatt	57
1.82	NextSong	57
1.83	NextModule	58
1.84	NoEngine	58
1.85	nogui	59
1.86	notify	60
1.87	LoadModule	60
1.88	LoadPlayer	61
1.89	LoadPlayerDir	62
1.90	Enable	62
1.91	LoadEngine	63
1.92	overwrite	63
1.93	lockmodule	64
1.94	parentdir	64
1.95	password	64
1.96	Play	65
1.97	vorheriges Modul	65
1.98	Playerbatch	66
1.99	prefix	66
1.100	PrevSong	67
1.101	PrevPatt	67
1.102	programmode	68
1.103	pubscreen	69
1.104	printtext	69
1.105	protect	70
1.106	protectionbits	70
1.107	QuickStart	71

---

---

1.108quit . . . . .	71
1.109randomsong . . . . .	71
1.110randomstart . . . . .	72
1.111ReplaySong . . . . .	72
1.112rescandir . . . . .	73
1.113rasterlines . . . . .	73
1.114Save Configuration . . . . .	74
1.115savedir . . . . .	74
1.116safesave . . . . .	75
1.117samplemode . . . . .	75
1.118saveasprotracker . . . . .	76
1.119savemodule . . . . .	76
1.120scanalways . . . . .	77
1.121Scrollfont . . . . .	77
1.122ScrollInfos . . . . .	78
1.123slowerspeed . . . . .	78
1.124scrolltext . . . . .	78
1.125smallmodule . . . . .	79
1.126softint . . . . .	79
1.127songend . . . . .	80
1.128Songname . . . . .	80
1.129speed . . . . .	81
1.130Stop . . . . .	82
1.131Subsong . . . . .	82
1.132timingmode . . . . .	82
1.133timemode . . . . .	83
1.134timeout . . . . .	84
1.135togglefilter . . . . .	85
1.136Voice . . . . .	85
1.137volume . . . . .	86
1.138waitingmode . . . . .	86
1.139xfddec crunch . . . . .	88
1.140xfdloadseg . . . . .	88
1.141xpkpackmethod . . . . .	89

---

# Chapter 1

## Eagleplayer

### 1.1 Eagleplayer 2.00 Dokumentation

Anleitung

zum

Eagleplayer

Version 2.00

© 1993-97 Defect Software Productions

Inhaltsverzeichnis:

**Einleitung und Rechtliches** Systemanforderungen, Featureübersicht , Copyrightbedingungen

**Installation** Wie bekomme ich es auf die Festplatte ?

**Registrierung** Wie und wo kann ich mich registrieren, Anschrift der Autoren

**Benutzung des Eagleplayers** Schalter, Menüs, Arexx, Konfigurationshinweise

Die Player Alles zu den Abspielern

Die Engines Alles zu den Engines

**Tips & Tricks**

**Probleme** Probleme, die unter Umständen auftauchen können

Geschichte Entstehungsgeschichte

Danksagungen

### 1.2 Einleitung & Rechtliches

Einleitung & Rechtliches

**Copyright und Vertriebsbedingungen**

**Distribution**

**Was ist und kann der Eagleplayer ?**

**Systemanforderungen**

**Entstehung des Eagleplayers**

**Was fehlt in dieser Version noch ?**

Wann kommt der nächste Eagleplayer ?

Registrierung des Eagleplayers

Registrierkarte

Update-Service

Fehlerbeschreibung

Keyfilebenutzung

## 1.3 Tips & Tricks

Eagleplayer auf eigenem Pubscreen

Der Eagleplayer besitzt momentan nicht die Möglichkeit, einen eigenen Screen zu öffnen. Wollen Sie das Programm dennoch auf einem solchen betreiben, so bietet sich hierfür der beim MUI-Paket beiliegende "PSI" (public screen inspector) an.

Definieren Sie mit dem PSI einen Screen, geben Sie dem einen Namen (z.B. "EagleplayerScreen") und schalten Sie die Option "autom. Schließen" ein (dadurch wird der Bildschirm bei Verlassen des Eagleplayers automatisch geschlossen).

Dann erstellen Sie sich ein Script zum Starten des Eagleplayers mit einem geeigneten Texteditor in der folgenden Art und Weise

```
MUI:PSI EagleplayerScreen open
```

```
Eagleplayer2:Eagleplayer Pubscreen=EagleplayerScreen
```

und setzen die Protectionbits auf "script" (z.B. protect StartEP +s).

Dieses Script kann dann von der Shell oder von WB unter Zuhilfenahme von IconX aufgerufen werden.

Es ist genauso gut möglich die beiden Kommandozeilen über einen Toolstarter wie z.B. Toolmanager, Barnone, Screentab usw. direkt hintereinander ausführen zu lassen.

Eagleplayer & Aminet CD`s

Eagleplayer kann auch direkt über den AminetCD-Guidebrowser als Abspielprogramm verwendet werden. Näheres finden Sie [hier](#) .

Eagleplayer & DirectoryOpus

Informationen zur Arbeit mit DOPUS V4.x finden Sie [hier](#) .

DOPUS5...

Abspielen einer Musikdatei mittels Doppelklick vom DOpus5 aus

Sie müssen dazu lediglich den "soundmodule"-Dateityp aus der

```
`Eagleplayer2:Goodies/Dopus5/Filetypes`-Schublade nach Dopu5:filetypes
```

kopieren. Danach sollten Sie in der Lage sein, Ihre Modules lediglich

mit einem simplen Doppelklick auf deren Namen in den DOpus Listern

vom Eagleplayer abspielen zu lassen.

Es kann durchaus passieren, daß Sie für bestimmte Modultypen noch eigene Namens Kürzel (Suffixe oder Präfixe) zu dem Dateityp hinzufügen müssen. Schauen Sie bitte dazu im Handbuch des DOpus5 nach, wie Sie das erreichen können.

Start vom DOpus (oder anderen Programmen) aus ohne Hauptfenster

Es gibt hierfür mehrere Möglichkeiten. Als einfachster Weg ergibt sich, die "Iconify"-Option im Kommandoaufruf zu setzen.

Vorteil: Nach dem Deiconifizieren steht der Eagleplayer mit allen in der Konfiguration eingestellten Parametern/Engines usw.

zur Verfügung.

Nachteil: dadurch, daß alle in der Konfiguration eingestellten Engines geladen werden, dauert das Laden des Eagleplayers länger.

Eine andere Methode ist, alle sichtbaren Engines über die **Kommandozeile** auszuschalten (Scopes, Moduleinfo oder sonstwas) und Eagleplayer im **NoGUI**-Modus zu starten.

Vorteil: speichersparend und Eagleplayer wird schneller geladen

Nachteil: Eagleplayer kann im NoGUI-Modus nur über ARexx kontrolliert werden.

Automatisch einen bestimmten Subsong abspielen

Einige Musikprogramme erzeugen Modules mit mehreren Musikstücken, Geräuschen usw. Manchmal passiert es dann, daß das Hauptmusikstück nicht gerade das erste ist. (z.B. die Musiken von Beast-I)

Um den Eagleplayer automatisch beim gewünschten Song starten zu lassen, braucht man lediglich einen Dateikommentar mit SS=xxx setzen, wobei xxx die Nummer des abzuspielenden Songs darstellt.

Beispiel: Filenote Modules:Whittaker/WH.Beast-2 SS:2

(von der Shell aus, mit DOpus, DirWork usw. ist es selbstredend analog möglich)

## 1.4 Erste Schritte in der Benutzung

## 1.5 Probleme / Fehler im Programm

Executive

Bei der Verwendung von Executive kann es vorkommen, daß der Eagleplayer beim Abspielen "hoppelt" oder gar nichts passiert. Dies kann daran liegen, daß sowohl Eagleplayer als auch Executive denselben CIA-Interrupt verwenden wollen. Ändern Sie dies bitte in den Einstellungen von entweder

**Eagleplayer** und/oder Executive am sinnvollsten auf

"Timer.device". Damit sollte dieses Problem behoben sein.

Weiterhin wird empfohlen, in den Executive Prefs den Task "Eagleplayer" auf "noschedule" und "above" zu setzen.

Screennotify.library

So nützlich die Screennotify.library sein mag beim Umstellen bestimmter Einstellungen der Workbench, so lästig können auch deren prinzipbedingte Fehler und Nebenwirkungen sein. Sollte es auf Ihrem Rechner verstärkt zu System-Lockups kommen (nicht nur in Verbindung mit dem Eagleplayer), probieren Sie es bitte aus, ob sich diese Sorgen nach dem Löschen oder Umbenennen der Library abstellen. Der Eagleplayer ist für diese Art von entstehenden Problemen mit dieser Library aufgrund seiner - je nach Konfiguration - vieler Fenster leider prädestiniert (obwohl ich nochmals betonen möchte, daß dies auf die Lib zurückzuführen ist und nicht den EP).

A1200 & externe PCMCIA HD/CD-Rom Controller

Die Software einiger über PCMCIA angeschlossene Festplatten & CD-Roms für den A1200 verwendet den CIA-B Interrupt ohne sich um dessen momentane Benutzung durch andere Programme zu scheren. Dadurch kann es zu erheblichen Problemen führen, wenn diese Option im **Timingmodus** eingestellt ist.

Bitte ändern Sie in diesem Falle die Einstellung auf "Timer.device".

## 1.6 Copyright und Vertriebsbedingungen

Copyright und Vertriebsbedingungen

Der Eagleplayer, die externen Programme, die Dokumentation, sämtliche Interfaces und auch die Entwicklerbeispiele sind ©1993-1997 Henryk Richter und Jan Blumenthal. Wir behalten uns das Recht vor, juristische Schritte einzuleiten, besonders in dem Fall, daß Teile des Eagleplayerpaketes in anderen Paketen, Softwareprodukten usw. ohne unsere Zustimmung vertrieben werden. Der Eagleplayer ist in der vorliegenden Version als **Shareware** zu betrachten. Dies heißt, daß wir bei regelmäßiger Benutzung eine Gebühr von 20 DM (bzw. 20US\$ bei Registrierung von außerhalb von Deutschland) erwarten, die in BAR, als Eurocheque oder Postanweisung zu entrichten ist (andere Währungen oder Cheques können wir leider derzeit nicht akzeptieren). Als Gegenleistung erhalten Sie die neueste Version des Eagleplayers zurück, die einige, in der SharewareVersion gesperrte Funktionen (z.B. Save, EPDir, alle Engines) besitzt.

Schicken Sie bitte die ausgefüllte Registrierkarte (Orderform) an:

Buggs Eagleeye

Henryk Richter Jan Blumenthal

Stephan-Jantzen-Ring 47 oder Erich-Mühsam-Straße 34

18106 Rostock 18069 Rostock

Deutschland Deutschland

Telefon: 0381 / 1208867 Telefon/Fax: 0381 / 83979

E-Mail: henryk.richter@stud.uni-rostock.de

oder jan.blumenthal@stud.uni-rostock.de

Falls Sie irgendwelche Bemerkungen, Vorschläge, Bugreports, neue Soundformate, Drohbriefe, zu verschenkende Amigas, ... zu vermeiden haben oder Probleme in der Benutzung haben, können Sie sich auch an die obenstehenden Adressen wenden. E-Mail ist in jedem Fall bevorzugt, da wir über diesen Kanal am schnellsten antworten können.

Der Eagleplayer darf unter folgenden Bedingungen auf Public-Domain Serien, Shareware-Serien und Netzwerken vertrieben werden:

- das Eagleplayerpaket muß komplett vorliegen, die zu diesem Paket gehörenden Dateien sind in `Distribution` aufgelistet
- weder das Programm, noch die Dokumentation dürfen verändert werden, eine Ausnahme stellt das Packen dar, solange sich der Originalzustand wieder 100% herstellen läßt (z.B.LHA,LZX,ZIP)
- der Vertriebspreis für Eagleplayer selbst oder der Preis für Disketten, auf denen der Eagleplayer enthalten ist, darf nicht mehr als den Betrag von 5 DM oder den entsprechenden Betrag in einer anderen Währung ausmachen, gleiches gilt für die Übertragungsgebühren in Netzwerken

Ausgenommen sind Compilationen auf CDRom`s

- die Firma Mallander Software darf den Player nicht vertreiben !
- Kommerzielle Verbreitung, des Eagleplayers ist ohne SCHRIFTLICHE Genehmigung der Autoren ausdrücklich untersagt.

Wir, die Autoren des Programms, übernehmen keinerlei Haftung für die durch dieses Programm eventuell entstandenen Schäden an Hard- und Software. Die Nutzung des Programms erfolgt auf eigene Gefahr.

WIR, DIE AUTOREN, BEHALTEN UNS DAS RECHT VOR, DEN VERTRIEB DES EAGLEPLAYERS EINZELNEN ANBIETERN ZU UNTERSAGEN, FALLS DIESE SICH NICHT AN UNSERE COPYRIGHT- BEDINGUNGEN HALTEN. Der Eagleplayer wurde nicht geschrieben, damit irgendwelche Leute sich daran eine goldene Nase verdienen können. Er soll die Amigawelt um ein sinnvolles Musikabspielprogramm zum -fast- Nulltarif erweitern. Und wer nicht vorhat, sich registrieren zu lassen, wird auch weder geächtet noch verbannt. Der Eagleplayer ist KEIN SASG Produkt und kostet daher keine DM30! Updates sind in jedem Fall für registrierte User kostenlos !

xfdmaster.library (C) Georg Hörmann  
req.library (C) Colin Fox & Bruce Dawson  
reqtools.library,powerpacker.library (C) Nico Francois  
lh.library (C) Stephan Furmann  
LHA,LHX (C) Stefan Boberg  
Installer Programm (C) Commodore Büromaschinen GmbH  
Installer and Installer project icon  
(c) Copyright 1991-93 Commodore-Amiga, Inc. All Rights Reserved.  
Reproduced and distributed under license from Commodore.

## 1.7 Distribution

### Distribution

Der Eagleplayer ist ein Programmpaket, das auf 2 Disketten oder als LHA Archiv auf dem Aminet verbreitet wird.

Zur 2-Disk-Version: Nach Möglichkeit sollten alle Dateien auf den beiden Disketten verbleiben, da es ansonsten beim Booten oder Installieren Probleme geben kann. Bitte beachten Sie: Wenn der Name der Disketten geändert wird (Beispielsweise für eine PD-Serie), so muß diese Änderung auch in das "Install"-Script eingetragen werden, ebenso in den Scripts: "Deutsche\_Anleitung" und "English\_Documentation".

## 1.8 Was ist und kann der Eagleplayer ?

Was ist und kann der Eagleplayer ?

Der Eagleplayer ist ein Programmpaket zum komfortablen Abspielen von Musik verschiedenster Soundformate. Er besitzt bereits die Abspielprogramme für Sound/Noise/Protracker, Startrekker(AM) & AudioSculpture(AM) sowie weitere ProTracker-Packer und einen speziellen Customplay - Port intern und ist in der Lage, weiterhin benötigte, externe Player nachzuladen. Der Vorteil dieser Methode liegt auf der Hand: Es brauchen nur die jeweils benötigten Player mitgeladen werden, was Ladezeit und Speicher spart, außerdem steigert sich die Länge des Hauptprogramms nicht ins Unendliche. Alle internen Soundtrackerplayer unterstützen Songloading. Die Songs und Samples können gepackt sein. Der Instrumenten-Pfad kann über das Konfigurationfile eingestellt werden. Die Samples dürfen gepackt sein. Sie werden alle als RAW-Daten geladen.

Einige Fähigkeiten des Eagleplayers:

- mehrere verschiedene Benutzeroberflächen, frei einstellbar,
- Schriftarten für die Menüs frei wählbar, teilweise mit frei wählbarem Hintergrundbild und in der Größe ziehbar !

- Hardwareunabhängiges Abspielsystem, bei dem die Soundausgabe beliebig auf verschiedene Amplifiers umgeleitet werden kann, so daß z.B. digital delayed Surround oder 14 Bit Ausgabe mit Echo-Effekten möglich ist, Module lassen sich auch aus dem Fastram abspielen, 16 Bit Samples können in maximaler Qualität direkt und ohne vorherige Herunterrechnung nach 8 Bit in 14 Bit oder auch 16 Bit (über Soundkarte) abgespielt werden
  - Modularer Aufbau mit Über 150 !! unterstützten Modulformaten, externen Engines und speichersparendem System, bei dem die externen Plug-Ins erst geladen werden, wenn sie benötigt werden, so daß bis über 1 MB Speicher gespart werden kann.
  - Möglichkeit von bis zu 64 extern zu ladenden Engines (z.B. Amplifier, Benutzerinterfaces, ext. Entpacker, Analyzer, Patternscroller wie beim Protracker, Moduleinformationen, Moduleselektor uvm.)
  - Konfiguration einfach und komfortabel, die gesamten Einstellungen lassen sich mit einer einzigen Aktion **speichern** (inklusive aktuellem Musikstück, Zustand und Position der Engines, aktuellem Verzeichnis, Packmodus beim Speichern usw.)
  - Online Hilfe über Amigaguide
  - läßt sich auf jedem **öffentlichen Screen** öffnen
  - unterstützt **AppWindow**
  - Äußerst komfortable Bedienung: z.B. kann die Auswahl des zu spielenden Musikstückes direkt über das **PullDown-Menu** erfolgen
  - Verzeichniswechseln im Pulldown Menü
  - die Anzahl der nachzuladenden Player ist nur durch den vorhandenen Speicher begrenzt
  - alphabetische Ordnung der Player und Engines im Manager, das gesamte Verzeichnis wird angezeigt, manuelles nachladen der Player und Engines durch einfachen Doppelklick auf den Eintrag
  - **Geschwindigkeitsregelung** über Schieberegler, teilweise sogar mit Pitch up/down
  - Statusanzeige durch Scroller
  - sorgfältig und umfangreich angepaßte Playroutinen, fast alle lassen mindestens **Lautstärke** - und **Balanceeinstellung** zu
  - kann so gut wie alle Arten **gepackter Dateien** laden, so z.B. XPK, Powerpacker, FileImploder, Crunchmania, lh.library, Stonecracker, sowie unterstützt Archive mittels des Extractors (LHA,LZX,ZIP,RAR usw.) auf eine einmalige Art, indem diese wie normale Verzeichnisse gehandhabt werden können. Außerdem wird die XFDMaster.library von Georg Hörmann unterstützt, die eine Masse an weiteren Entpackern bietet. Bei XPK-gepackten Dateien erscheint während des Ladens eine Fortschrittsanzeige.
-

- umfangreiche Moduleinfos, wenn der Replayer entsprechend angepaßt wurde, inclusive Berechnung der **Spielzeit** für verschiedene Musikformate (Protracker, Oktalyzer, SonicArranger, Jamcracker uvm.)
  - mächtiger **ARexx-Port**
  - **DoubleBuffering** -Funktion zum ununterbrochenen Abspielen von Modulen durch sofortiges Nachladen des nächsten Moduls
  - verschiedene **Filerequester** , je nach Geschmack (ASL,Req,ReqTools)
  - läuft problemlos im Multitasking, die Audiokanäle werden ordnungsgemäß reserviert, alle rechenintensiven Programmteile (die Scopes) laufen mit geringer Priorität
  - CyberGraphX wird durch einige Scopes direkt unterstützt. (FFT-Analyzer)
  - ist ein Commodity (Hotkey-Aktivierung, Steuerung über Exchange)
  - **Ein** - und **Ausblenden** der Musik möglich
  - Abspielen eines zufällig ausgewählten Modules aus dem aktuellen Verzeichnis sofort beim Programmstart !
  - läßt sich sowohl über die **Voreinstellungsdatei** , als auch über **Tooltypes** und **Shell** -Argumente konfigurieren
  - Instrumente eines Soundmoduls können abgespielt und gespeichert werden
  - spielt sogar RAW-Dateien wie z.B. CDDA
  - kann sowohl die CIA-Timer, als auch VBlank und das Timer.device zur Synchronisation der Abspielroutinen verwenden
  - unterstützt Screennotify.library
  - kann in Dateien und ausführbaren Programmen nach Modulen suchen (Eagleexotic)
  - lädt Modulelisten aller wichtigen Abspielprogramme
  - verfügt über eine umfangreiche Listenverwaltung
  - .
  - .
  - .
  - .
- Durch seinen modularen Aufbau ist eine optimale Anpassung an das jeweilige System und die Vorlieben seines Nutzers möglich.

## 1.9 Entstehung des Eagleplayers

Entstehung des Eagleplayers

Alles fing mit dem Phenomena-Demo "Enigma" an. Dort wurde NoisePacker verwendet. Die Musik hatten wir gerippt, aber es gab keinen Player. Nun wurde schnell ein Tool zum Abspielen dieses wirklich guten Modules geschrieben. Irgendwann wurden dann auch andere

Playroutinen implementiert (NoiseTracker...). Daraus entstand dann der Megaplayer 1.0, der am 16.04.1992 released wurde. Er konnte zu dem Zeitpunkt 16 Playroutinen abspielen. Daraus wurden dann über 25 und das Hauptprogramm wurde immer unübersichtlicher. Im Juni 1992 entschied sich Jan, nachdem dauernd über seinen Noiseplayerähnlichen Megaplayer gelästert wurde, ein neues Soundplayerprogramm zu schreiben, das Maßstäbe setzen sollte. Er programmierte fleißig, erstellte eine ansprechende Oberfläche, versuchte möglichst noch mehr Playroutinen heranzuschaffen und diese in das Programm einzubauen. Es wurde ein Konzept für externe Player entworfen. Da das damalige Konzept von Delirium besser war, wurde dieser Port eingebaut. Später wurde unser Port für externe Player so stark erweitert und verbessert, daß die Nutzung unserer Player nur noch mit dem Eagleplayer möglich war. Da wir auf grafische Spielereien nicht verzichten wollten, entwickelten wir ein System für externe, asynchron laufende Analyzerprogramme, die wir in der frommen Absicht, diesen weiteren Port offen zu dokumentieren, anfangen zu programmieren. Mit der Zeit nam diese Arbeit allerdings Dimensionen an, die ein Außenstehender wohl kaum begreifen würde. Daher der seltsame Name. Aus diesem Analyzerport wurde ein universelles Interface für verschiedenste Anwendungen, beispielsweise Konverter (Noiseconverter), Amplifier, Listenverwaltung, Moduleinformationen, Pubscreenselector, Patternscroller, Benutzeroberflächen und vieles mehr.

## 1.10 Future

Was fehlt in dieser Version noch ?

Wir haben noch einen ganzen Haufen von Ideen, die aber zur Umsetzung vorraussichtlich eine Weile dauern werden.

ein paar Beispiele:

- weitere Benutzeroberflächen in extrem aufwendigem Look
- Fertigstellung der neuen Listenverwaltung
- ...

## 1.11 Updates

Wann und wo gibt es eine neue Versionen des Eagleplayers ?

Neue Versionen lassen sich auf jeden Fall aus dem Aminet unter 'mus/play' downloaden. Dort werden die neuen Versionen auf jeden Fall als erstes zu finden sein.

Weiterhin finden sich Updates und Betaversionen auf unserer Homepage, <http://home.pages.de/~buggs/eagleplayer2.html>

## 1.12 Update-Service

### Update-Service

Jeder registrierte User kann sich bei uns die neuste Version bestellen, ohne Gebühren zu bezahlen. Allerdings erwarten wird, daß er 2 Disketten und einen ausreichend frankierten Rückumschlag mitschickt. Auf min. einer der Disketten muß sich das Keyfile des registierten Users befinden. Weiterhin bitten darum, die Updatekarte auszufüllen und vor allem die Keyfile-Nummer (sie finden sie, wenn sie im "Project Menü" des Hauptfensters "About" anwählen und dann "Key" anklicken) anzugeben.

Wenn Sie Lust und Laune haben, können Sie ja ein paar Fragen beantworten:

- Sind Sie mit der Zustellung des Eagleplayer zufrieden ?
- Gibt es Probleme bei speziellen Rechnerkonfigurationen ?
- Haben Sie mehr bzw. weniger von Ihrem registrierten Eagleplayer erwartet ?
- Ist die Dokumentation ausreichend ?
- Welche Funktionen sollten unbedingt noch eingebaut werden ?
- Arbeitet der Eagleplayer mit allem Programmen zusammen ?
- Funktionieren die Playroutinen korrekt ?
- Benutzen Ihre Freunde auch den Eagleplayer ? Wenn nicht, wieso nicht ?

Wir werden dann auf diese Disketten die aktuelle Version des Eagleplayers kopieren und an Sie zurückschicken. Sollte noch derselbe Eagleplayer aktuell sein, wird auf die nächste Version gewartet.

## 1.13 Fehlerbeschreibung

### Fehlerbeschreibung

Wenn Sie uns einen Fehler im Eagleplayer beschreiben wollen, sollten Sie einige Sachen beachten und versuchen, den Fehler so rekonstruierbar wie möglich zu machen. Eine Aussage "Der Eagleplayer stürzt manchmal beim Laden ab", hilft uns nicht im geringsten. Es könnte sein, daß das Modul defekt ist oder der Depacker falsch entpackt, daß jemand den Speicher zersägt hat. Da gibt es viele Varianten. Hier nun einige Sachverhalte, die Sie nach Möglichkeit unbedingt beantworten sollten.

- \* Sind Sie registrierter User ?
  - \* Welche Eagleplayerversion nutzen Sie ?
  - \* Hardwarekonfiguration angeben !
  - \* Welche Patches haben Sie installiert ?
-

- \* Freier Speicher zur Zeit des Absturzes ?
- \* Welche Player/Engines waren geladen ? Ist es offensichtlich egal, ob Engines oder Player geladen wurden ?
- \* Tritt der Fehler auch bei anderen Rechnerkonfigurationen auf ?
- \* Konfigurationsfile angeben !
- \* Bei welchem Module gibt es Probleme ? Bitte immer mitschicken !!!
- \* Tritt der Fehler nur bei Rechnern mit Cache auf ?
- \* Wie waren die internen Einstellungen des Eagleplayers ? Ist es offensichtlich egal, wie diese sind ?
- \* Wurde das Module richtig erkannt ?
- \* Sind die Gadgets schon freigegeben worden (bei Moduleloading) ?
- \* War das Hauptwindow (noch) offen ?
- \* Bei welcher Kickstartversion trat der Fehler auf ? Ist es offensichtlich egal ?
- \* Welche Engines waren aktiv ?

## 1.14 Keyfilebenutzung

### Keyfilebenutzung

Registrierte User erhalten von uns ein Keyfile. Dieses muß entweder in das aktuelle Verzeichnis des Eagleplayers oder nach Devs: kopiert werden (kann auch vom Installscript erledigt werden) und ermöglicht ihnen den Zugang zu den bisher gesperrten Funktionen.

Das Keyfile ist personengebunden. Es enthält die volle Adresse des Users sowie die Registriernummer und einige weitere Daten. Für den Fall, daß ein Keyfile an Dritte weitergegeben wurde, behalten wir uns rechtliche Schritte vor. In der nächsten Version des Eagleplayers wird diese UserNr dann logischerweise zusätzlich gesperrt.

Alle Keyfiles können von nachfolgenden Eagleplayerversionen genutzt werden.

## 1.15 Systemanforderungen

### Systemanforderungen

#### Minimalanforderung:

- einen Amiga mit 68020 CPU
- OS 2.04
- 0.5 MB ChipRam und 1 MB Fastram
- Festplatte
- Monitor mit Lautsprechern (die auch angeschlossen sind)

Möglicherweise läuft der Eagleplayer auch mit einer 68000-CPU,  
allerdings wird hierfür keine Garantie übernommen

empfohlene Konfiguration:

- 68030/50 MHz oder besser (bei Verwendung einer Soundkarte 68040 oder besser)
- OS 3.0 / 3.1
- 4 MB oder mehr Fastram
- HiFi Anlage

Bitte beachten Sie, daß eine Menge Funktionen unter Kickstart 2.0 noch nicht verfügbar sind. Im Weiteren werden wir uns in dieser Anleitung auf OS 3.0 beziehen. Es ist darauf zu achten, daß nicht gleichzeitig ein anderes Soundprogramm arbeitet, das die Tonausgabekanäle belegt. In diesem Fall gibt der Player nach dem Laden eines Modules eine Meldung aus, siehe bei "Fehlermeldungen"

Ansonsten versucht der Eagleplayer zusätzlich zu den Standard-Systemlibraries die XPKMaster.library, die PowerPacker.library, CRM.library, XFDMaster.library, sowie die LH.Library zu öffnen, um dementsprechend gepackte Files zu laden. Diese Libraries sind NICHT UNBEDINGT zum Start notwendig. Zum Betrieb der weiteren optionalen Filerequester neben ASL werden außerdem die "req.library" bzw. die "reqtools.library" notwendig.

## 1.16 Registrierung

Registrierung des Eagleplayers

Die Sharewaregebühr für den Eagleplayer beträgt 20 DM (bzw. 20US\$ außerhalb von Deutschland), die in BAR oder als Eurocheque zu entrichten ist (andere Währungen oder Cheques können wir leider nicht akzeptieren). Als Gegenleistung erhalten Sie die neueste Version des Eagleplayers zurück, die einige, in der SharewareVersion gesperrte Funktionen (z.B. Save, Konfiguration) besitzt. Außerdem gibt's die verbesserten Versionen inclusive neuer Engines alle ständig gratis, so daß die Sharewaregebühr angesichts der Postpreise durchaus gerechtfertigt ist. Beachten Sie bitte, daß nur Privatadressen akzeptiert werden !

Sie können auch den Betrag auf folgendes Konto überweisen:

Ostseesparkasse Rostock

Konto: 1450017491

Blz: 130 500 00

(Kontoinhaber: Henryk Richter)

Bitte weisen Sie in dem Text darauf hin, daß es sich um die Registrierung

des Eagleplayers handelt und vergessen Sie bitte nicht, Ihren Namen anzugeben. Die Registrierkarte wird natürlich auch in diesem Fall benötigt. Sie können diese als Brief oder besser (weil schneller) über E-Mail versenden.

Bitte schicken Sie zur Registrierung des Eagleplayers die ausgefüllte

Registrierkarte an eine der

folgenden Adressen:

Bugs Eagleeye

Henryk Richter Jan Blumenthal

Stephan-Jantzen-Ring 47 oder Erich-Mühsam-Straße 34

18106 Rostock 18069 Rostock

Deutschland Deutschland

Telefon: 0381 / 1208867 Telefon/Fax: 0381 / 83979

E-Mail: [henryk.richter@stud.uni-rostock.de](mailto:henryk.richter@stud.uni-rostock.de)

[tfa652@cks1.rz.uni-rostock.de](mailto:tfa652@cks1.rz.uni-rostock.de)

[jan.blumenthal@stud.uni-rostock.de](mailto:jan.blumenthal@stud.uni-rostock.de)

[jb001@cks1.rz.uni-rostock.de](mailto:jb001@cks1.rz.uni-rostock.de)

Falls Sie irgendwelche Bemerkungen, Vorschläge, Bugreports, neue Soundformate, Drohbriefe, zu verschenkende Amigas, ... zu vermelden haben oder Probleme in der Benutzung haben, können Sie sich auch an die obenstehenden Adressen wenden.

Wenn Sie Fehler beschreiben, schauen Sie unbedingt in das Kapitel

**Fehlerbeschreibung**, um uns die Behebung zu vereinfachen !

## 1.17 Installation auf der Festplatte,...

Installation

[Installation auf Festplatte](#)

[Start vom CLI](#)

[Start von der Workbench](#)

[Konfigurationsdatei](#)

[Die Playerbatch-Datei](#)

## 1.18 Installation auf der Festplatte,...

Installation

Beim Eagleplayer liegt ein Installscript für den Commodore Installer bei, mit dem die Installation automatisch abläuft. Folgen Sie bitte den Anweisungen (es sind auch Hilfstexte zu den einzelnen Aktionen verfügbar) und dann sollte es keine Probleme geben. Bei der Installation wird dann

automatisch ein Verzeichnis für den Eagleplayer angelegt, aus welchem Sie das Programm dann starten können.

(Alle zum Eagleplayer gehörigen Dateien werden übrigens im Verzeichnis des Eagleplayers mit untergebracht und nicht wie bei anderen Paketen wild auf der Festplatte verstreut.

**ACHTUNG:** Mit Version 2.0 hat sich der Aufbau des Eagleplayer-Verzeichnisses grundlegend geändert. Versuchen Sie nicht, den Eagleplayer per Hand wie in Version 1.54 zu installieren, das geht 100%ig schief.

## 1.19 Starten des Eagleplayers

Starten des Eagleplayers von CLI/SHELL

Zum Start von Cli/Shell aus, genügt ein einfacher Aufruf des Players.

Vergessen Sie aber nicht, vorher in das Verzeichnis (DOS-Befehl 'CD'), in dem sich der Player befindet zu wechseln oder den korrekten Pfad mit anzugeben (z.B. Work:Music/Eagleplayer/Eagleplayer ).

Der Eagleplayer bietet eine Reihe von Argumenten beim Start von der Shell aus an, so daß Sie z.B. auch beim Start ein Modul angeben können, welches automatisch geladen wird. Weiterhin ist es möglich, den Eagleplayer auch ein zweites Mal aufzurufen, eventuell angegebene Module werden dann automatisch an den bereits laufenden Eagleplayer übergeben. Die angegebenen Shell-Argumente haben auf jeden Fall Priorität über die im Config-File gespeicherten Werte. Die entsprechenden Optionen finden Sie im Arexx-Kapitel, sowie bei allen Beschreibungen der angebotenen **Menüoptionen** .

Achtung! Seit der Version 2.0 hat sich das Verhalten beim Shellstart geändert, der Eagleplayer koppelt sich nicht mehr selbst von der Shell ab, sondern muß nun gegebenenfalls über `Run` gestartet werden. Dies wurde notwendig durch einige neue Features, wie z.B. die Unterstützung von Aminet-CD`s.

Beispiel:

```
Eagleplayer loadmodule="dh2:Protracker/Industrial/Mod.Bifat_Bunkerschliessung"
```

oder

```
Eagleplayer loadmodule "dh2:Protracker/Ravy/Mod.Ani_Soundcompo"
```

Sie haben bei den Shell-Optionen auch die Möglichkeit, bestimmte, im Konfigurationsfile gesetzte, Einstellungen ignorieren zu lassen. Dies geschieht über das Schlüsselwort: "Skip\_Config\_Item"

Beispiel:

```
Eagleplayer loadmodule Skip_Config_Item randomstart no
```

Im obigen Beispiel würde der Eagleplayer gestartet werden, ohne daß dabei ein eventuell in der Konfigurationsdatei eingestelltes Modul geladen wird, sehr praktisch z.B. um vom **DOPUS®** aus ein Modul anzuwählen.

## 1.20 Start von der Workbench aus

Start von der Workbench aus

Zum Start von der Workbench genügt ein einfacher Doppelklick mit der Maus auf das Piktogramm, wodurch der Eagleplayer geladen wird.

Seit Version 2.0 unterstützt der Eagleplayer auch Tooltypes, die zusätzlich zur **Konfigurationsdatei** nutzbar sind. Beachten Sie bitte, daß die Tooltypes in jedem Fall eine höhere Priorität als die Einstellungen der Konfigurationsdatei haben und auch nicht beim Speichern rückgesetzt werden. Das heißt im Klartext, daß Optionen, die über die Tooltypes manuell eingestellt werden, auch nur über diesen Weg geändert werden können. Die entsprechenden Optionen finden Sie im **Arexx-Kapitel**, sowie bei allen Beschreibungen der angebotenen **Menüoptionen**.

Beispiel für Tooltypes:

```
config="work:eagleplayer2/my_second_configs/"
```

(würde den Eagleplayer veranlassen, die Konfigurationen im alternativen Verzeichnis "my\_second\_configs" zu suchen)

## 1.21 Das Konfigurationsfile

Das Konfigurationsfile

Der Eagleplayer lädt ab Version V1.10 eine Konfigurationsdatei. Sie besteht aus einer Reihe von Kommandos, die als Strings eingegeben werden. In der Konfig kann u.a. angegeben werden, welches das aktuelle Playerverzeichnis ist oder ob Fadein ein/ausgeschaltet werden soll, usw. Die Datei befindet sich normalerweise im "configs/" Verzeichnis.

Optional kann der Pfad für die Konfiguration auf ein anderes Verzeichnis gelegt werden. Dies geschieht durch setzen der ENV-Variable "EP-Configpath". So läßt sich der Eagleplayer beispielsweise mit mehreren Konfigurationen nutzen.

```
Beispiel: setenv EP-Configpath Work:Music/Player/Eagleplayer/configs_2  
copy env:EP-Configpath envarc:
```

Beachten Sie bitte, wenn Sie eine Konfig **speichern**, wird der komplette aktuelle Stand des Eagleplayers abgespeichert, also Modulename, derzeitige Engineeinstellungen, sämtliche Pfade und alle Preferenceeinstellungen. Die Kommandos zum Konfigurationsfile finden Sie in der Config, die bei der Installation mitgeliefert wurde, oder in einer selbst abgespeicherten. Es müssen in einer Konfig-Datei nicht alle Kommandos vorkommen. Die Kommandos der Konfigdatei sind weitgehend kompatibel zu denen des ARexx-Ports.

## 1.22 Die PlayerBatch-Datei

Die PlayerBatch-Datei

Die PlayerBatchDatei ist eine Stapeldatei, die benötigt wird, wenn man die "Playerloading when needed"-Funktion nutzen will. Das bedeutet, der benötigte Player wird erst geladen, wenn er benötigt wird. Dazu ist es nötig, jedem Musikformat eine eigene Namenskennung als Präfix zu geben. (wie eigentlich auch üblich, also Protracker "MOD.", SonicArranger "SA." usw.). In dieser Datei wird die Namenskennung sowie der Pfad eines jeden Players festgelegt. Die Groß/Kleinschreibung braucht nicht beachtet zu werden.

Beispiel:

```
FC.*FC13.*="Eagleplayers/FC 1.3"
```

```
*.sdata ="Eagleplayers/MusicMaker8"
```

```
MDAT.*="Eagleplayers/TFMX 1.5"
```

```
MDAT.*="Eagleplayers/TFMX Pro"
```

```
MDAT.*="Eagleplayers/TFMX 7V"
```

Es sind, wie man sieht, auch mehrere Kennungen möglich! Weiterhin muß erwähnt werden, daß wenn Eject Players ausgeschaltet ist, jeder Player nur ein Mal eingeladen wird, egal ob er noch in der Liste ist oder nicht. Es ist auch möglich, verschiedene Player gleichzeitig über eine Kennung zu laden, wie es hier bei TFMX demonstriert wird.

Eine komplette Eagleplayer-BatchDatei liegt dem Originalpaket bei.

Um diese Datei nutzen zu können, muß die **Playerbatch** -Funktion aktiviert sein.

Seit Version 2.0 gibt es den Playerloader, der die Funktionen der Playerbatch praktisch überflüssig macht, indem die geladenen Dateien nach deren Typ analysiert werden und daraufhin automatisch der entsprechende Player geladen wird. Allerdings werden aus sicherheitsgründen nicht alle Player unterstützt (z.B. ActivisionPro).

## 1.23 Benutzung, Steuerung und Konfiguration

Benutzung des Eagleplayers

**Einführung** Erste Schritte in der Benutzung

**Hauptfenster** Die Buttons

**Menüs** Die Pulldown-Menüs

**Appwindow** Drag & Drop

**Pubscreens** Wo soll er denn hin ?

**Commodity** Wie kann ich ihn steuern ?

**Depacker** Was kann er alles entpacken ?

**Arexx** Einleitung zum Arexx-Port

**Kommandos** Liste der Arexx-Kommandos

**Arexx-Status** Auskunft über Arexx

**DirOpus 4** Wie steuere ich den Eagleplayer vom Opus 4 aus

**Aminet-CD`s** Eagleplayer und Aminet CD`s

## 1.24 Das Hauptfenster

Das Hauptfenster

Das Hauptfenster des Eagleplayers ist eine grafische Benutzeroberfläche (GUI), welche als Engine, sprich also ein externes Programm realisiert ist und somit nach Geschmack des Benutzers austauschbar ist. Ich habe einmal versucht, das Erscheinungsbild mit den bescheidenen Möglichkeiten von Amigaguide nachzubilden, sollte das Ganze durch Ihre eigenen Schriftarteneinstellungen von Multiview zur Unleserlichkeit verkommen, sei an dieser Stelle noch einmal auf die [Online-Hilfe](#) verwiesen.

Eagleplayer1.x-GUI

---

```

| * | Eagleplayer V2.00 Generic/Registered | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | Volume | | zz |
| | | I | P |-----| |
| 2 | | | Balance | | | |
| |-----|
| 3 | | | < | << | | > | >> | | >> | > | > | | ^ | 00 |
| |-----|
| 4 | Welcome to Eagleplayer |

```

---

BifatGUI

---

```

| * | Eagleplayer V2.00 Generic/Registered | | | > | | . | | |
|-----|
| O | | O | << |
|-----|
| \ | / | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | : | < | << | | > | >> | >> | > | > | ^ | 00 |
|-----|
| | / | Welcome to Eagleplayer | | | | |
|-----|
|-----|

```

---

## 1.25 Steuerung über die Pulldown-Menüs

Steuerung über die Pulldown-Menüs

Der Eagleplayer besitzt folgende Pulldown-Menüs, die wie gewohnt die Maus angewählt werden können.

Projekt

Modul laden

Modul speichern

Über das Modul

Player laden

Datei löschen

Hilfe

Dokumentation

Iconify

Über

Ende

Kommandos

Stimmen

Lautstärke

Balance

Geschwindigkeit

vorheriges Modul

vorheriger Song

vorheriges Pattern

Starte Song neu

Spielen

Schneller spielen

nächstes Pattern

nächster Song

nächstes Modul

Stop

Modul entfernen

Engines (siehe auch hier)

Engine Laden

\*\*\*\*-GUI Prefs <- Einstellungen zur aktuellen Benutzeroberfläche

14 Bit Amplifier

8 Bit Amplifier

Amplifiermanager

BifatGUI

---

---

Dirlistviewer  
Eagleexotic  
Extractor  
FFT-Analyzer  
Formatloader  
Levelgraph  
Levelmeter  
Manager  
Messagewindow  
Moduleinfo  
Noiseconverter  
Patternscroll  
Patternscroll++  
Playerloader  
Publicscreenselector  
Pysion  
Quadrascope  
Samplesaver  
SpaceScope+  
StereoScope  
Time  
Konfiguration  
Global  
Rolle Informationen  
Einblenden  
Ausblenden  
Filter  
Meistervolume  
Songname  
Vorsilbe  
Lade in Fastram  
Playerbatch  
Modulesprefs  
Directory neu laden  
Lade Directory  
EPDir  
Dosnachricht  
Lade Directory immer  
Verstecke alle

---

**Speichern**

ungepackt

Powerpacker

LH.Library

XPK.Library

Crunchmania

Automatisch speichern

Überschreiben

Sicher speichern

Instrumentenmodus

Speichern als Protracker

Modul schützen

Protectionbits

Setze Speicherdirectory

Ändere XPK-Packer

Ändere Passwort

**Programmierung**

kein neues Modul

nächstes Modul

vorheriges Modul

zufälliges Modul

Songende

Lade schon nächstes

Untersongs zufällig

Untersongs automatisch

Lade Immer

Directorysprünge

Zeitmodus

berechne Spielzeit

Start mit Zufallsmodul

Sofort starten

Min. Spielzeit

Max. Spielzeit

**Timing**

CIA-Timing

VBLANK-Timing

Timer.device

Automatisch warten

Auf Rasterzeilen warten

mit DBF-Schleifen warten  
reserviere Audiokanäle  
Softinterrupt  
Setze Rasterzeilen  
Setze DBFSchleifen  
Entpacken  
Passwort automatisch  
Entpacke erstes Modul  
Dateien mit XFD entpacken  
Executables mit XFD entpacken  
Warnen  
kleine Module  
große Module  
fehlerhafte Module  
Filerequester  
Iconify  
Aussehen  
wähle Menüfont  
wähle Modulfont  
wähle Rollfont  
PublicScreen ändern  
Filter  
Konfiguration speichern  
Modules-Directory, \*\*\* Einträge  
<<Mutterverzeichnis>>  
Mod.A  
Mod.B  
OK.C

## 1.26 Die AppWindow-Funktion

Die AppWindow-Funktion

Der Eagleplayer nutzt für das Hauptfenster die von Kickstart 2.0 an verfügbare AppWindow-Funktion. Was heißt das? Appwindow ist eine Art Drag&Drop, die Möglichkeit, z.B. Workbenchpiktogramme von Modulen anzuklicken, auf das Eagleplayerfenster zu ziehen und dort loszulassen, das Modul wird dann geladen und abgespielt. Sollten Sie durch durch Multiselect mehrere Icons auf das Eagleplayerfenster ablegen, werden diese in alphabetischer Reihenfolge ins PullDown-Menü übernommen. Die Icons können aus verschiedenen Verzeichnissen

stammen!

Besondere Bedeutung hat die Funktion seit DOpus5 erlangt, bei dem es möglich ist, per Drag&Drop Einträge aus den Listen auch in andere Fenster abzulegen, so sie das unterstützen. Somit können Sie auch Ihre Files aus den Listen auf dem Eagleplayer-Hauptfenster ablegen und der Player lädt diese dann automatisch ein.

## 1.27 pubscreens

Pubscreens

Der Eagleplayer kann ab Version V1.10 nun endlich auch auf Pubscreens geöffnet werden.

Die Pubscreenunterstützung betrifft alle Windows der zum Eagleplayer gehörenden Windows bzw. Programme, falls nicht anders angegeben, d.h. alle Engines öffnen sich auf demselben Screen. Hier nun eine kleine Liste von Publicscreennamen:

CygnusEdScreen1 (bei Rendering choices im Environment-Menu  
make screen public auf on setzen)

BootX

PowerPacker

DOPUS.1

TERM

EDGE

Seit der Version 1.52 existiert die Engine Publicscreenselector, über die man auf einen einfachen Mausklick hin den PubScreen, auf dem der Eagleplayer mit seinen Fenstern erscheinen soll, wechseln kann, einfacher geht`s nicht.

## 1.28 Commodity-Funktionen

Commodity-Funktionen

Der Eagleplayer besitzt seit der Version V1.10 einen Commodity-Port.

Dieser ist nur ab Kick2.0 aufwärts verfügbar. Der Eagleplayer kann über selbstdefinierte Hotkeys aufgerufen werden. Voreingestellt ist die Tastenkombination "CONTROL ALT E". Bei Aktivierung des Hotkey wird entweder das Hauptwindow mit Engines, Konfigurationsfenster etc. geöffnet und in den Vordergrund gebracht oder geschlossen.

In Zusammenarbeit mit Commodities-Exchange können weitere Commandos, wie u.a. Quit realisiert werden.

Die Optionen "CX\_Popup", "CX\_Popkey" und "CX\_Priority" können sowohl im Konfigurationsfile, als auch über Shell und Tooltypes angegeben werden.

Gegenüber den Anforderungen an kleine Commodities, die beim Schließen ihres Konfigurationsfensters im Speicher verbleiben, wird der Eagleplayer beim **Schließen** des Fensters beendet. Eine **Iconify** -Funktion ist extra verfügbar.

## 1.29 Interne Depacker

Interne Depacker

Der Eagleplayer in der vorliegenden Version kann Datenfiles entpacken, wenn diese mit folgenden Crunchern gepackt sind:

- CrunchMania ;nur mit Library, aber alle Modi
- Powerpacker (falls vorhanden, wird die Library benutzt)
- LH (PMC vom Powerplayer- doch was ist am Powerplayer POWER ???)
- Fimp
- Pack (Soundtracker-Songpacker)
- XPK (bei den meisten Packern [SQSH,NUKE] wird gleich während des Ladens entpackt !!!)
- StoneCracker V4.0

weitere Packer bietet die ebenfalls unterstützte XFDMaster.library, bitte schauen Sie in deren Dokumentation nach, welche Formate diese entpacken kann. Bei Fimp wird nur ein Memoryblock allokiert, in den das Module geladen wird und auch entpackt wird. Diese Methode funktioniert wunderbar, aber eben nur bei Fimp. Bei den anderen Packern muß erst ein Memoryblock zum Laden und einer zum Entpacken besorgt werden. Diese Methode verbraucht natürlich viel Speicher, und das ist eben weniger schön. Nach dem Entpacken wird der Speicher des geladene & gepackten Files wieder freigegeben.

Der EaglePlayer unterstützt die XPKMaster.Library, wenn die entsprechenden Libs auf der Festplatte installiert sind. Sollte eine Datei ein Passwort benötigen, kann dieses über ein StringGadget im Hauptwindow eingegeben werden. Der Vorteil von XPK ist, daß die Datei während des Ladens entpackt werden kann. Dies spart als Nebeneffekt mitunter eine Menge Speicher, je nach verwendeter XPK-Library. Für das Entpacken von PCM-Files, also mit der LH.Library gepackten Files, wird diese Library benötigt, ist diese nicht vorhanden, kann das Module nicht entpackt werden.

Bei Pack handelt es sich um den SongPacker der alten Tracker (NoiseTracker, Protracker, Startrekker).

Powerpacker-, Fimp-, Pack- und StoneCracker4.0-Files können ohne Hilfsmittel entpackt werden. Die entsprechenden Decrunchroutinen sind im EaglePlayer enthalten. Wird die Powerpacker.Library gefunden, wird diese zum Entpacken von PP-Dateien genutzt.

Der EaglePlayer kann weiterhin auch executable Files entpacken, wenn diese mit ImploderV4.0 (NormalMode ohne Library) gepackt sind. Diese Funktion wurde implementiert, weil das Programm zunehmend länger und länger wurde, vor allem

die Player und Engines. Um diese einem durchschnitts-User auch zugänglich zu machen, kann man sie eben alle packen. CustomModule können ebenfalls mit Imploder gepackt vorliegen. Ron-Klaren-Modules müssen ungepackt sein ! In späteren Versionen werden auf jeden Fall mehr File-Entpacker unterstützt, tut mir leid, das es gerade Imploder erwischt hat, aber besser als gar nicht packen, oder ?

Seit Version V1.52 wird die XFDMaster.Library unterstützt, d.h Sie können auch PowerPacker-Executables laden, wenn Sie den entsprechenden Slave besitzen. Sie können also auch Player und Engines mit dem Powerpacker oder CrunchMania oder StoneCracker packen, soweit dies durch die jeweilige Version der XFDMaster.library unterstützt wird. Achtung! Lesen Sie hierzu, bevor Sie dies einschalten, unbedingt den Dokumentationsteil zur entsprechenden **Funktion** !

Die XFDMaster.library können Sie bei

Georg Hörmann

Martinswinkelstraße 16c

82467 Garmisch-Partenkirchen

oder auf gutsortierten PD-Serien bzw. vom Aminet bekommen.

## 1.30 Einführung und Beschreibung des Arexx-Portes

Einführung und Beschreibung des Arexx-Portes

Der Eagleplayer unterstützt seit Version V1.10 Arexx. Dazu stellt der Eagleplayer eine Reihe von **Kommandos** zur Verfügung, durch die der Player fast komplett fernbedienbar wird. Sämtliche zu den entsprechenden **Menüfunktionen** gehörenden Arexx-Befehle sind auch in den dortigen Teilen der Anleitung mit aufgelistet.

Der Arexx-Port trägt den Namen rexx\_EP.

Eine Anpassung an Programme wie z.B. **DOpus** ist daher überhaupt kein Problem.

Achtung: Durch die Umstrukturierung des gesamten Eagleplayers zur V2.0 haben sich einige Kommandos geändert, weiterhin ist es in einigen Fällen sinnvoller geworden, auf die Shell-Optionen zurückzugreifen.

So z.B. erübrigt sich ein extra Arexx-Script, um vom DOpus aus eine Datei zu laden. Der Kommandoaufruf Eagleplayer Loadmodule="{f}" mit dem Kommandotyp "AmigaDOS" und der Option "run asynchronously" reicht für eine bescheidene Installation völlig aus.

## 1.31 Arexx-Kommandoübersicht

### Arexx-Kommandoübersicht

Im folgenden sehen sie die Arexx-Kommandos, die der Eagleplayer unterstützt. Ist kein Fehler aufgetreten, wird als Result der Scrolltext zurückgegeben, der im Statuswindow scrollen würde, außer wenn natürlich konkret Argumente als Returnwerte, z.B. beim Status, erwartet werden. Ist ein Fehler aufgetreten steht in rx\_Result der Fehlertext. Der Anwender sollte natürlich testen, ob die einzelnen Funktionen auch nutzbar sind, um sein Arexx-Programm ordentlich anzupassen. Wenn z.B ein Module gespeichert werden soll, aber dies der Replayer nicht erlaubt, dann funktioniert das eben nicht. Wenn im folgenden von Filenamen gesprochen wird, dann wird grundsätzlich der ganze Pfad mit angegeben (falls nicht anders erwähnt).

AboutEP

Aboutmodule

Addlistentry

Addlist

Allocchannels

ASCEngine

Automaticsave

Autopassword

AutosubSong

Balance

BigModule

CalcDuration

ChipRamAmplifier

CX\_Popup

CX\_Popkey

CX\_Priority

Config

Crunchmode

DBFLoops

Defaultspeed

DeleteAllEngines

DeleteAllPlayer

DeleteEngine

DeleteFile

DeleteGui

DeletePlayer

Dirjump

---

---

Enable  
Eject  
EjectPlayers  
EjectEngines  
EPDir  
ExtractFirstModule  
FadeIn  
FadeOut  
Fasterspeed  
FastRamAmplifier  
Fastplay  
Filerequestermode  
Filter  
Help  
Helplink  
Hide  
HideAll  
Iconify  
Iconifymode  
Instruments  
Loadalways  
Loadbefore  
LoadConfig  
LoadDir  
LoadEngine  
LoadEngineDir  
LoadFast  
LoadGui  
LoadModule  
LoadPlayer  
LoadPlayerConfig  
LoadPlayerDir  
LockModule  
Mastervolume  
Menufont  
Mintimeout  
Module  
ModuleFault  
Modulesfont

---

---

NextModule  
NextSong  
NextPatt  
NoEngine  
NoGui  
Notify  
Overwrite  
ParentDir  
Password  
Pause  
PBits  
Play  
PlayerBatch  
Playerconfig  
PrevModule  
PrevPatt  
PrevSong  
Prefix  
Printtext  
ProgramMode  
Protect  
PubScreen  
Quit  
QuitEagle  
QuickStart  
Randomsong  
Randomstart  
Rasterlines  
ReplaySong  
RescanDir  
SafeSave  
Samplemode  
SaveAsProtracker  
SaveConfig  
SaveDir  
SaveModule  
SaveModulePrefs  
SavePlayerConfig  
SaveT

---

Scanalways  
Scrollfont  
ScrollInfos  
Scrolltext  
Slowerspeed  
SoftInt  
SongEnd  
SongName  
SmallModule  
Speed  
SSDir  
Status  
Stop  
SubSong  
Timemode  
Timeout  
Timingmode  
ToggleFilter  
Voice  
Volume  
Waitingmode  
XFDDecrunch  
XFDLoadSeg  
XPKPackmethod

Das waren schon alle! :-)

## 1.32 Anpassung an Directory Opus 4.x

Die Arexxanpassung an Directory Opus 4.x

In diesem Kapitel soll beschrieben werden, wie Sie Arexxscrips, die zur Steuerung des Eagleplayers dienen an den Opus anpassen. Dazu wurden eine Reihe von Scripts mitgeliefert. Unsere Arexxscripts sind alle an DOpus4 angepaßt (ArexxPortname des ersten geöffneten Opuses: "DOPUS.1"). Beachten Sie bitte, daß der DOPUS3 einen anderen Arexx-Portnamen verwendet ! Zum Anpassen der ArexxScripts kopieren Sie diese bitte in Ihr assigntes REXX: Verzeichnis. Danach starten Sie das Configurationsprogramm des Opus. Wählen Sie bitte das BUTTON-Gadget an und klicken Sie dann auf eins der freien Gadgets die dann erscheinen. Am Besten ist, Sie öffnen einen eigene Bank (New Bank anklicken), denn Sie können mit den Arexx-Kommandos locker die komplette Bank belegen. So jetzt erscheint der Button-Edit-Screen. Geben Sie bitte den Namen des Buttons ein

("LoadModule"). Betätigen Sie Gadget "New Entry". Es werden einige Gadgets freigegeben und ein Cursor erscheint. Geben Sie bitte folgende Zeile ein "EP\_LoadModules.dopus" ein. Stellen Sie das linke Gadget auf "Arexx". Betätigen Sie nun "Okay" bis Sie wieder im Opus sind. Selektieren Sie mehrere Module und klicken Sie danach das LoadModule-Gadget.

Dieser Arexx-Script wertet alle angeklickten Namen auf und übergibt diese dem Eagleplayer, der wiederum das erste davon lädt und den Rest in sein Pull-Down-Menü übernimmt. Ist nur ein Module selektiert wird im Eagleplayer das ganze Directory geladen. Falls kein Eagleplayer geladen ist, erledigt dies der Arexx-Script, aber ändern Sie den Pfad des Eagleplayers im Script so, daß der Player auch gefunden wird. Möchten Sie ein Module per Doppelklick laden, müssen Sie einen Filetype definieren. Gehen Sie dazu bitte auf das Gadget "Filetypes". Nun wählen Sie aus dem PullDown-Menü den Punkt "New" an. Dort wo der Cursor erscheint geben Sie "SoundModule" und im nächsten Stringgadget "Music" ein. Die Eingaben sind zur Orientierung und haben sonst keinerlei Bedeutung. Wählen Sie nun "New Entry" an. Betätigen Sie die Lupe und klicken Sie "Matchname" an. Dort wo der Cursor erscheint geben Sie ein "MOD.\*". Drücken Sie Enter und gehen wieder auf "New". Diesmal klicken Sie "or" an und dann wieder auf "New Entry" gehen. Nochmals "Matchname" betätigen und folgendes eingeben: "CU.\*". Drücken Sie "Okay".

In dem diesem Window drücken Sie auf "New". Es erscheint ein Window in dem Sie das Wort, welches wir vorhin bei File\_Class eingegeben haben("SoundModule"). In dem oberen TeilWindow "Events..." klicken Sie "Double-Click" an. Bei Event Discription geben Sie den Text an, den der Opus bei Aktivation Ihrer Routine in die Titelleiste des Screens schreiben soll. Also geben Sie ein "Now loading selected Soundmodule ". Betätigen Sie "New Entry" und geben Sie ein "rx rexx:EP\_Loadmodule.dopus". Stellen Sie das linke Gadget auf AmigaDos und betätigen Sie "Okay". Wenn Sie Ihrem Filetype höchste Priorität geben wollen, müssen Sie den Namen im nun erschienene Window nach ganz oben "swappen". Sie haben jetzt eine FileClass geschrieben, d.h. alles was per Doppelklick angewählt wird und mit "MOD." oder "CU." anfängt, wird von Ihrer Class erkannt und an den Eagleplayer weitergeleitet, der falls es möglich ist, das Module lädt und abspielt.

### 1.33 Status

Status

Mit Hilfe des Statuskommandos können Sie die umfangreichen Einstellungs-möglichkeiten des Eagleplayers abfragen. Hier erhalten Sie eine Übersicht über die entsprechenden Argumente. Die sind auch bei den Beschreibungen der einzelnen Kommandos mitangegeben.

Arexx/Shellstart:

Status <Gruppe> <Argumente>

Gibt den aktuellen Status zurück. Dabei wird zwischen folgenden Gruppen

unterschieden: - Globale Einstellungen (g),

- Modulabhängige Einstellungen (m)

- Playerabhängige Einstellungen (p)

- SampleInfo (s) (nicht eingebaut !!!)

- Engines (e)

Je nach Gruppe müssen evtl. weitere Argumente übergeben werden. Im folgenden

ist eine Übersicht aller Status-Kommandos des Eagleplayers zu sehen.

Die globalen Einstellungen

-----  
status <Gruppe> <Argumente> ;<Gruppe> = g

;<Argumente> = <Kommando>

<Kommando> Bedeutung Returnwert  
-----

ver Eagleplayer Version Zahl

dve Eagleplayer Version Zahl

dir Playerverzeichnis String

udi Engineverzeichnis String

fon Derzeitigen Font String

hep HelpPfad String

ins Instrumentenverzeichnis String

sap SavePfad String

pbp PIBatchPfad String

ubp UPrgBatchPfad String

pke HotKey String

num Playeranzahl Zahl

ico Iconify on <Yesno>

mwi Main-Window offen <Yesno>

pli Playerwindow offen <Yesno>

vol derzeitige Lautstärke Zahl

bal derzeitige Balance Zahl

spe derzeitiger Speed Zahl

ply Song spielt <yesno>

hel HelpModus <yesno>

pub derzeitiger PubScreen String

mau Max. Engines Zahl

usp geladene Engines Zahl

snd aktuelle SongNummer Zahl  
-----

---

mil Anzahl der Module in der Liste Zahl  
epb Eagleplayer Globals Zahl  
vo1 Voice1 <yeslno>  
vo2 Voice2 <yeslno>  
vo3 Voice3 <yeslno>  
vo4 Voice4 <yeslno>  
eab EP-Batch <yeslno>  
ejp EjectPlayer <yeslno>  
flp FlashPointer <yeslno>  
msv MasterVolume <yeslno>  
sna Use Songname <yeslno>  
qsf Quit Surface <yeslno>  
que Quit Eagle <yeslno>  
sat SaveT <yeslno>  
sci Scroll-Infos <yeslno>  
led Filter <yeslno>  
fid Fadein <yeslno>  
fad Fadeout <yeslno>  
aps AutoPasswort <yeslno>  
ldd Load Dir <yeslno>  
aud Auto Dir <yeslno>  
sdn Show Dirnames <yeslno>  
hia HideAll <yeslno>  
uep Use EPDir <yeslno>  
not Notify <yeslno>  
alw Scan Always <yeslno>  
mme Größe des menuMems Zahl  
end Songend <yeslno>  
lbf Load Before <yeslno>  
law Load Always <yeslno>  
qst Quickstart <yeslno>  
rns RandomSong <yeslno>  
ass AutoSubSong <yeslno>  
tim TimeOut Zahl  
ovw Overwrite <yeslno>  
ats AutomaticSave <yeslno>  
ssd SS Dir <yeslno>  
prt Protect Module <yeslno>  
pbi Protection Bits <yeslno>

---

smo SampleMode <yesno>  
 spt Save As Protracker <yesno>  
 smo Small Module <yesno>  
 bmo Big Module <yesno>  
 mof Module fault <yesno>  
 icm aktueller IconifyMode String  
 frm aktuelles Filerequester String  
 Die moduleabhängige Einstellungen

-----  
 status <Gruppe> <Argumente> ;<Gruppe> = m  
 ;<Argumente> = <Kommando>  
 <Kommando> Bedeutung Returnwert

-----  
 pat ModulePath String  
 dir Moduleverzeichnis String  
 fil Modulename String  
 pnr Nr des aktiven Players Zahl  
 num Position in Modulliste Zahl  
 min max Anzahl SubSongs Zahl  
 max max Anzahl SubSongs Zahl  
 aut Authornamen String  
 son Songname String  
 len PatternLength Zahl  
 dpt verschiedene Pattern Zahl  
 stp Steps Zahl  
 sam Samples Zahl  
 chk CheckMod (mod ok?) <yesno>  
 pon PositionsNr Zahl  
 icm aktuelle Playernummer Zahl  
 dur berechnete Spielzeit Zahl  
 siz gesamte Länge des Moduls Zahl  
 Die playerabhängigen Einstellungen

-----  
 status <Gruppe> <Argumente> ;<Gruppe> = p  
 ;<Argumente> = <PlayerNr> <Kommando>  
 <Kommando> Bedeutung Returnwert

-----  
 nam Playernamen String  
 cre Creatornamen String

---

pin PlayerInfo String  
ver Versionsnummer Zahl  
vol Lautstärke veränderbar <yes/no>  
bal Balance veränderbar <yes/no>  
jmp Patternjump verfügbar <yes/no>  
pos GetPositionsNr <yes/no>  
spd Playspeed regelbar <yes/no>  
sub Subsong verfügbar <yes/no>  
cfg Player konfigurierbar <yes/no>  
pck Supports Packed Mods <yes/no>  
upr Engineunterst. <yes/no>  
csa Config savbar <yes/no>  
clo Config loadbar <yes/no>  
msa Module savebar <yes/no>  
mii ModuleInfo möglich <yes/no>  
sai SampleInfo möglich <yes/no>  
msm Max. Samples Zahl  
msy Max. SynthSamples Zahl  
mle Max. Length Zahl  
mpa Max. Pattern Zahl  
mst Max. Steps Zahl  
mso Max. SubSongs Zahl  
rst Player unterstützt Restartbar <yes/no>  
sge Player unterstützt Songende <yes/no>  
voi Player unterstützt Voices <yes/no>  
plf Player unterstützt PlayFaster <yes/no>  
mch Module change <yes/no>  
mrs Module restore <yes/no>  
dat Creator-Date String  
tgl Tagliste des Players Zahl  
Die engineabhängigen Einstellungen

-----  
status <Gruppe> <Argumente> ;<Gruppe> = e  
;<Argumente> = <EngineNr> <Kommando>  
<Kommando> Bedeutung Returnwert  
-----

nam Name des Engine String  
abo About Engine String  
cre Creator des Engine String

---

ver Version des Engine Zahl  
rev Revision des Engine Zahl  
eve min. EP-Version Zahl  
dat Creator-Date String  
kis Kickstart des Engine Zahl  
prz Prozessname des Engine String  
win Window offen <yesno>  
con Config des Engine Zahl  
kox X-Koordinate des Windows Zahl  
koy Y-Koordinate des Windows Zahl

### 1.34 aminetcd

Benutzung der Aminet-CD`s mit dem Eagleplayer

Was ist das Aminet überhaupt ? Aminet ist ein verteiltes Archiv, wo Autoren ihre PD- und FD-Programme veröffentlichen können. Getragen wird Aminet von etwa 15 Computern weltweit, die per Internet verbunden sind, einem Computer-Netzwerk das die meisten Universitäten und viele Firmen auf der ganzen Welt miteinander verbindet. Auch an vielen anderen Orten, so zum Beispiel in über 50 Mailboxen, kann man auf regelmässig aktualisierte Aminet- Dateien zugreifen. Was auch immer irgendwo ins Aminet eingespielen wird, steht innert nur 24 Stunden weltweit zur Verfügung - eine Geschwindigkeit, mit der kein anderes Medium konkurrieren kann. Es hat den Umfang mehrerer Gigabyte an Software. (= das momentan größte Verbundnetz für Software im Internet überhaupt) An dieser Stelle möchten wir Urban Müller unseren Dank nicht nur für XPK :-)) sondern auch für die Administration des Aminet aussprechen.

Zurück zum Thema: Die von den jeweiligen Neuigkeiten des Aminets gepreßten CD`s enthalten meist eine recht große Menge an Musikstücken, die direkt im Amigaguide-Index ausgewählt und über einen externen Modplayer abgespielt werden können. Um hierfür den Eagleplayer nutzen zu können, tragen Sie bitte nach Aufruf des auf der CD befindlichen "Prefs"-Programmes bei "Mod Player" folgende Zeile ein:

```
"<Pfad des Eagleplayers>/Eagleplayer loadmodule"
```

Als Pfad geben Sie die Position des Eagleplayers auf der Festplatte an (Sie können auch das über den "Set"-Schalter angebotene Dateiauswahlfenster verwenden) und nach dem Pfad und Namen muß dann noch das Wort loadmodule folgen.

Beispiel1: work:music/players/Eagleplayer loadmodule

Beispiel2: work:music/players/Eagleplayer iconify=on loadmodule

## 1.35 AboutEP

Über

Diese Funktion zeigt einige Informationen über den Eagleplayer, ob das Programm registriert ist und wenn ja, den Namen und die Adresse und sowie die Keyfilenummer des entsprechenden Nutzers.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

aboutEP

## 1.36 AboutModule

About Module

Scrollt einen Informationstext durch das Hauptfenster, der Angaben über das aktuelle Musikstück und den verwendeten Player macht. Der Umfang des Textes variiert je nach Player, im ungünstigsten Fall wird lediglich der Name der geladenen Datei und deren Länge ausgegeben.

Wenn kein Module geladen wird, wird das **Aboutwindow** des Eagleplayers geöffnet.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

Aboutmodule - diese beiden Funktionen liefern den selben Text zurück, welcher auch status m abo im Scroller angezeigt wird.

## 1.37 Addiere Listeneintrag

Addiere Listeneintrag

Nicht dokumentiert!

## 1.38 Addiere Liste

Addiere Liste

Nicht dokumentiert!

---

## 1.39 Allokiere Audiokanäle

Reserviere Audiokanäle

Da die Audio-Hardware und deren Softwaremäßiges Interface des Amigas nicht darauf ausgelegt ist, daß mehrere Programme sie gleichzeitig nutzen können (bzw. nur mit erheblichen Einschränkungen), ist der Eagleplayer darauf angewiesen, sie allein und ohne Wechselwirkungen mit anderen Programmen nutzen zu können. Um solche Probleme zu vermeiden, alloziert sich der Eagleplayer bei eingeschalteter Option die Audiokanäle zur alleinigen Nutzung für die Zeit, in der der Abspielvorgang läuft. Es wird empfohlen, diese Option nicht abzuschalten.

Seit Version 2.0 werden die Audiokanäle auch bei Benutzung der **Pause-**Funktion freigegeben und nur noch während des Abspielens belegt, so daß die aktuelle Musikdatei nicht zwingend aus dem Speicher **geworfen** werden muß.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Allocchannels Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g ach - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Allocchannels=Zustand - Zustand: siehe oben

## 1.40 ASCEngine

ASCEngine

Nicht dokumentiert!

## 1.41 automaticsave

Automatisch speichern

Bei eingeschalteter Funktion wird die Datei ohne weitere Abfrage des Namens unter dem aktuellen Dateinamen (oder dem **Songnamen + Dateierweiterung**, je nach Einstellung) in das **eingestellte Verzeichnis gespeichert**.

Tastatur:

-

---

Menü-Shortcut:

-

Arexx/Shellstart:

AutomaticSave Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g ats - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

AutomaticSave=Zustand - Zustand: siehe oben

## 1.42 autosong

Untersongs automatisch

Bei Soundsystemen, deren Musikdateien mehrere Musikstücke enthalten, werden erst alle Untersongs abgespielt, bevor abhängig von den Einstellungen bei **Programmode** eine neue Datei geladen wird. Voraussetzung dafür ist entweder die vom entsprechenden Abspieler unterstützte **Songendeerkennung** oder aber ein vom Benutzer eingestelltes **Timeout** . (siehe auch [hier](#) bzw. bei **Randomsong** ).

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Autosong Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g ass - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Autosong=Zustand - Zustand: siehe oben

## 1.43 autopassword

Passwort automatisch

Beim entpacken/entschlüsseln von mit Passwörtern geschützter Dateien kann über diese Option eingestellt werden, daß - falls bei allen das selbe Passwort verwendet wurde - ein über die **Ändere Passwort** - Funktion eingegebenes Passwort automatisch zum Entschlüsseln verwendet wird.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Autopassword Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g aps - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Autopassword=Zustand - Zustand: siehe oben

## 1.44 balance

Balance

Dieser Regler dient der Beeinflussung des Lautstärkeverhältnisses zwischen linkem und rechtem Kanal. Wird der Regler nach links bewegt, wird die Lautstärke des rechten Kanals verringert und umgekehrt. Ein Balancewert von "0" bedeutet, daß beide Seiten mit gleicher Lautstärke arbeiten. Sollte der entsprechende aktuelle Replayer keine Balanceunterstützung besitzen, so wird dieser Schieberegler (bzw. die entsprechenden Menüeinträge) gesperrt.

Das BifatGUI besitzt noch 2 Extraschalter für die Balance, der linke davon setzt beim ersten Anklicken die Balance nach ganz Links und bei wiederholter Anwahl auf die Mittenposition zurück. Der rechte Button verhält sich analog.

Die über das Menü aufrufbaren Möglichkeiten umfassen die Verschiebung des Lautstärkeverhältnisses um jeweils einen Schritt nach links/rechts, sowie die Rücksetzung auf Mittenstellung.

Tastatur:

-

Menü-Shortcut:

A-9 = Balance weiter nach links

A-0 = Balance weiter nach rechts

A-8 = Regler in die Mitte setzen, Gleichverteilung auf beiden Seiten

Arexx/Shellstart:

Balance Zahl - Zahl: -255...0...255 (-255=ganz links;0=Mitte, 255=ganz rechts)

Anmerkung: der Eagleplayer rechnet intern mit  $256*2$

Balancestufen, gibt aber dem Nutzer momentan einen Bereich von -64..64 als

Wert an, was der Amiga-Hardwarebeschränkung

Rechnung trägt.

---

Arexx:

status g bal - gibt den momentanen Wert zurück (-255...255)

Anmerkung: da Arexx Probleme mit negativen Zahlen

hat, werden für Werte <0 Zahlen von  
65535 (= -1) bis 65281 (= -255) zurück-  
gegeben, die vom Arexx-Programm  
entsprechend umgerechnet werden  
müssen.

Tooltype:

Balance=Zahl - Zahl: siehe oben

## 1.45 bigmodule

Warnen - große Module

Bei eingeschalteter Option öffnet der Eagleplayer bei zu langen Musikdateien einen Requester, der auf diesen Umstand hinweist. Zu große Musikdateien stellen zwar (normalerweise) kein Problem dar, aber so kann man sehr gut erkennen, wo sich auf der Festplatte noch etwas Platz sparen läßt.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Bigmodule Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g bmo - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Bigmodule=Zustand - Zustand: siehe oben

## 1.46 calcduration

Spielzeit berechnen

Wenn eingeschaltet und durch den Abspieler unterstützt, wird die Spielzeit der aktuellen Musikdatei berechnet und im Moduleinfo mit angezeigt. Welche Abspieler die Spielzeitberechnungsfunktion unterstützen, können Sie im Manager erfahren. (z.B. Protracker+Clones, SonicArranger, Jamcracker, Oktalyzer, AIFF, ADPCM)

Tastatur:

-

---

Menü-Shortcut:

-

Arexx/Shellstart:

CalcDuration Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g cad - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

CalcDuration=Zustand - Zustand: siehe oben

## 1.47 ChipRAMAmplifier

ChipRamAmplifier

Nicht dokumentiert!

## 1.48 crunchmode

Packmodus

Diese hier eingestellten Optionen verändern das Verhalten des Eagleplayers beim **Abspeichern** der aktuellen Datei(en). Sie haben hier die Wahl, die Datei(en) entweder ungepackt oder mit den unten ausgeführten Komprimierungsvarianten abzuspeichern. Es gibt weiterhin bei einigen Methoden (Powerpacker, Crunchmania und LH.Library) die Möglichkeit, einen speziell auf Soundmodules angepaßten, sogenannten **Delta-Modus** zu aktivieren, der in den meisten Fällen einen erheblichen Komprimierungsgewinn erbringen kann.

- ungepackt - die Datei wird nicht komprimiert
- Powerpacker - es wird die powerpacker.library von Nico Francois verwendet
- LH.Library - es wird die lh.library von Stefan Fuhrmann verwendet
- XPK.Library - es wird der momentan eingestellte **XPK-Packer** zum Speichern verwendet
- Crunchmania - es wird die crm.library von Thomas Schwarz zum Komprimieren herangezogen

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Crunchmode Zahl - Zahl: 0 = ungepackt

1 = Powerpacker

2 = LH.Library

3 = XPK.Library

4 = Crunchmania

Arexx:

status g cmo - gibt den momentanen Wert zurück (0..4)

Tooltpe:

Crunchmode=Zahl - Zahl: siehe oben

## 1.49 Popup

PopUp

Diese Funktion öffnet/schließt das HauptBedienungs Fenster des Eagleplayers je nach den übergebenen Argumenten.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

CX\_Popup yes - das Hauptfenster wird geöffnet

CX\_Popup no - das Hauptfenster wird geschlossen

status g mwi - yes/no als Resultat

Tooltpe:

CX\_Popup=yes - beim Start wird das Hauptfenster geöffnet

CX\_Popup=no - beim Start wird der Hauptfenster nicht geöffnet

## 1.50 CX\_Popkey

PopKey

Diese Funktion setzt das Hotkey mit der komplette Eagleplayer versteckt oder wieder geöffnet werden kann

Tastatur:

-

Menü-Shortcut:

-

Arexx:

CX\_Popkey <Hotkey> - Das Hotkey wird gesetzt

status g pke - Hotkeystringyes als Resultat

Tooltpe:

CX\_Popkey=<Hotkey> - Das Hotkey wird gesetzt

## 1.51 CX\_Priority

CX\_Priority

Diese Funktion setzt die Commoditypriorität.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

CX\_Pri <Zahl> - Priorität

Tooltipe:

CX\_Pri= <Zahl> - Priorität

## 1.52 config

Config

Nicht dokumentiert!

## 1.53 defaultspeed

Voreingestellte Geschwindigkeit

Die **Abspielgeschwindigkeit** wird auf den Normalwert zurückgesetzt.

Tastatur:

-

Menü-Shortcut:

A-#

Arexx:

Defaultspeed

## 1.54 dbfloops

setze DBF-Schleifen

In der Einstellung "mit DBF-Schleifen warten" des **Wartemodus** werden zum zeitgerechten Abspielen der Musikdateien mehrere Warteschleifen mit der CPU durchgezählt. Dieser Modus ist stark Prozessorabhängig und sollte im Normalfall zu Gunsten der anderen angebotenen Möglichkeiten nicht eingeschaltet werden. Ein auf dem 7.09 MHz 68000er Prozessor brauchbarer Wert ist in etwa 300. Wer Lust und Langeweile hat, kann ja einmal probieren, welcher Wert z.B. auf dem 68060 zu brauchbaren Resultaten führt, ich würde gern

einmal die Zahl sehen...

Einen zu geringen Wert an Warteschleifen erkennt man daran, daß ab und zu Noten der zu spielenden Musikdatei verschluckt werden. Erhöhen Sie in diesem Fall den Wert oder schalten Sie auf " **Automatisch warten** ".

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

DBFLoops Zahl - Zahl: Anzahl zu wartender Zyklen

Arexx:

status g dbl - gibt den momentanen Wert als Zahl zurück,

Tooltpe:

DBFLoops=Zahl - Zahl: siehe oben

## 1.55 dirjump

Directorysprünge

Wenn diese Funktion eingeschaltet ist, springt der Eagleplayer beim automatischen Laden einer nächsten Datei in Unterverzeichnisse des letzten durch den Benutzer ausgewählten Verzeichnisses hinein und sucht sich dort die nächste Musikdatei zum Abspielen heraus. Besonders dann, wenn Sie zum Beispiel Ihre Musikstücke nach Autor in verschiedene Verzeichnisse sortiert haben, wird diese Funktion auf jeden Fall für Abwechslung beim Musikabspielen sorgen.

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** und im **Programmierungsmodus** "zufälliges Modul".

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Dirjump Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g dij - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Dirjump=Zustand - Zustand: siehe oben

Beispiel:

Ihre Verzeichnisse auf der Festplatte mögen in etwa so organisiert sein:

Modules:

|

Protracker

||

| Audiomonster

||

| Dreamer

||

| Folxmusic

||

| Gabba

||

| Hardcore

.

.

.

Wenn Sie nun bei eingeschalteter Dirjump-Funktion per Dateirequester,Arexx etc. das Protracker-Verzeichnis auswählen und dann die NextModule-Funktion aktivieren, so wird der Eagleplayer sich die nächste Datei entweder aus "Audiomonster", "Dreamer" oder "Folxmusic" usw. holen und ins Protracker-Verzeichnis zurückkehren, bis eine neue Datei geladen werden soll.

## 1.56 DeleteFile

Datei löschen

Diese Funktion öffnet das Dateiauswahlfenster zur Angabe einer zu löschenden Datei.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

deletefile [pfad+name] - bei Angabe einer Datei wird der Dateirequester nicht geöffnet und die Datei sofort gelöscht

## 1.57 documentation

Dokumentation

Dieses Kommando veranlaßt den Eagleplayer, über die Amigaguide.library genau diese Datei, die Sie jetzt lesen (die Anleitung bzw. Online-Hilfe) zu öffnen und anzuzeigen

Anmerkung: Wenn Sie das Amigaguide-Fenster schließen, wird die Datei der Dok nicht automatisch freigegeben, sondern erst bei erneuter Anwahl des Menüeintrages.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

Documentation

## 1.58 Eject

Eject Module

Stoppt das aktuelle Module (falls eins gespielt wird) und entfernt es aus dem Speicher. Danach werden die Audiokanäle freigegeben, so daß auch andere Programme wieder auf die Audiofunktionen Ihres Amigas zugreifen können.

Tastatur:

-

Menü-Shortcut:

A-e

Arexx:

Eject

## 1.59 epdir

EPDir

EPDir ist eine Funktion, um das Laden der Verzeichnisse teilweise um ein erhebliches beschleunigen zu können. Dabei wird der gesamte Verzeichnisinhalt in einer kleinen Datei namens ".epdir" abgelegt und beim Laden der Verzeichnisse anstelle dieser geladen und im Menü angezeigt, sofern sich das Verzeichnis nach dem letzten Speichern des ".epdir"-Files nicht geändert hat.

Dies macht sich besonders bei großen Verzeichnissen sehr stark bemerkbar, wenn die Festplatte unaufgeräumt ist und mit dem normalen FFS-Dateisystem betrieben wird.

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** .

Achtung: Es kann in seltenen Fällen passieren, daß die **Notify** -Funktion mit der EPDir-Funktion kollidiert und dadurch ständig das Verzeichnis neu gelesen wird. Schalten Sie in diesem Fall entweder EPDir oder **Notify** aus.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

EPDir Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g epd - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

EPDir=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.60 extractfirstmodule

entpacke erstes Modul

Diese Option ändert das Verhalten des Eagleplayers in Bezug auf Archive. (LHA, LZX, ZIP usw.). Normalerweise agiert der Extractor in der Art und Weise, daß Archive wie normale Verzeichnisse gehandhabt werden und die einzelnen Module einzeln daraus ausgewählt werden können. Dies erweist sich allerdings bei den Aminet CD`s als recht unpraktisch, da bei den meisten dort zu findenden Archiven nur ein Modul enthalten ist und der Aufwand beim Durchforsten der Archive durch weitere nötige Mausklicks steigen würde.

Lange Rede - kurzer Sinn, bei eingeschalteter Option wird bei Anwahl eines Archives automatisch die erste darin gefundene Datei entpackt und - so es sich um ein Module handelt, abgespielt. Es wird dringend empfohlen, bei Verwendung von "extractfirstmodule" die **Hideall** -Funktion einzuschalten, da es sonst sehr leicht dazu kommen kann, daß der Eagleplayer versucht, das "Readme"-File als Modul abzuspielen.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Extractfirstmodule Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g efm - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Extractfirstmodule=Zustand - Zustand: siehe oben

## 1.61 Fadeln

Einblenden

Ist Fadein gesetzt, werden alle Musikstücke langsam auf die aktuell eingestellte Lautstärke eingeblendet.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Fadein Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g fai - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Fadein=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.62 FadeOut

Ausblenden

Ist FadeOut gesetzt, werden alle Musikstücke am Ende des Abspielens langsam ausgeblendet, bevor der Abspielvorgang stoppt.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

FadeOut Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g fao - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

FadeOut=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

---

## 1.63 FastPlay

Fast Play

"schneller Vorlauf", erleichtert die Suche nach einer bestimmten Passage im Titel.

Tastatur:

-

Menü-Shortcut:

A-o

## 1.64 fasterspeed

Schneller

Die **Abspielgeschwindigkeit** wird um einen Schritt erhöht.

Tastatur:

-

Menü-Shortcut:

A-+

Arexx:

Fasterspeed

## 1.65 filerequestermode

Filerequester

Mittels dieser 3 Optionen können Sie selbst nach Geschmack wählen, welche der angebotenen Dateiauswahlauswahlfenster vom Eagleplayer zu verwenden sind. Es besteht die Möglichkeit zwischen ASL (System Standard), Reqtools.library (Nico Francois) und der schon etwas älteren Req.library (Cygnum Software).

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Filerequestermode Zahl - Zahl: 0=req.library,1=reqtools.library,  
2=asl.library

Arexx:

status g frm - gibt den momentanen Zustand zurück (0...2)

Tooltype:

Filerequestermode=Zahl - Zahl: siehe oben

---

## 1.66 Filter

### Filter

Schaltet den Tiefpaßfilter des Amigas ein und aus. Diese Funktion hat keine Wirkung beim Amiga 1000 und Amiga 2000A. Bei eingeschaltetem Filter klingt die Ausgabe reichlich dumpf, da alle Frequenzen über 7 KHz durch den LowPass weggeschnitten werden. Gedacht ist die Funktion, um dafür zu sorgen, daß beim Start des Eagleplayers der Filter je nach Belieben automatisch ein-und ausgeschaltet wird, siehe auch den Kommentar zu [ToggleFilter](#) .

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Filter Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g led - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Filter=yes/no - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.67 Help

### Hilfe

Diese Funktion dient zur Aktivierung der Online-Hilfe des Eagleplayers, nach Aktivierung werden alle Schalter und Menüs anwählbar und nach Drücken eines Schalters oder Auswahl eines Menüpunktes wird ein AmigaGuide-Fenster geöffnet, in dem man dann den entsprechenden Teil dieser Dokumentation wiederfindet.

Tastatur:

HELP

Menü-Shortcut:

A-h

Arexx:

help yes/no - ein/ausschalten

status g hel - gibt als Resultat den Zustand der Hilfefunktion zurück (yes/no)

## 1.68 Hide

### Hide

Die Hide-Funktion schließt alle Fenster, die zum Eagleplayer gehören. Im Gegensatz zur "Iconify"-Funktion werden alle visuellen Teile des Eagleplayer versteckt. Man kann das Hauptfenster über verschiedene Möglichkeiten wieder aktivieren, einmal, indem man das Hotkey aktiviert (normal: CTRL-LALT-E), es über "Exchange" öffnet oder aber den Eagleplayer neu startet. (dieser sendet dem im Speicher befindlichen automatisch das Signal, sich zu öffnen)

## 1.69 hideall

### HideAll

HideAll sorgt dafür, daß beim Einladen des aktuellen Verzeichnisses alle vom Namen her als Nicht-Musikdateien bekannten Files nicht mit im Menü bzw. Dirlistviewer angezeigt werden.

Liste der versteckten Dateiendungen:

```
"info", ".s", ".m", ".i", ".o", ".h", ".c", ".a", ".z", ".hlp", ".nfo", ".lib"
".bat", ".bak", ".dmp", ".dms", ".doc", ".bbs", ".msd", ".dok", ".asc", ".diz"
".txt", ".ins", ".asm", ".rs", ".nt", ".fd", ".ss", ".as", ".nt.pp", ".catalog"
".guide", ".hyper", ".pic", ".gui", ".offs", ".help", ".rexx", ".prefs", ".instr"
".config", ".device", ".library", ".font", ".samp", ".ilbm", ".fastdir"
".backdrop", ".speeddir", ".isdir", ".sadir", ".okdir", ".epdir", ".display"
".displayme", ".readme", ".readme", ".liesmich", ".ip", ".i.n", ".ip.n", ".ip.1"
"-handler"
```

Liste der versteckten Dateivorsilben:

```
"info.", ".smp.", ".rsmp.", ".smp.", ".jpns.", ".index",
```

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** .

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

HideAll Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g hia - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

HideAll=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.70 Iconify

### Iconify

Schließt das Hauptfenster und alle anderen offenen Fenster des Eagleplayers und legt entweder ein sogenanntes "AppIcon" auf der Workbench ab oder legt einen Eintrag im "Tools"-Menü der Workbench an. Um die Fenster des Eagleplayers wieder zu öffnen, genügt ein Doppelclick auf das Appicon beziehungsweise eine Anwahl des Menüitems.

Tastatur:

i

Menü-Shortcut:

A-i

Arexx:

iconify yes

iconify no

status g ico - yes/no als Resultat

Tooltypes:

iconify=yes

iconify=no

## 1.71 iconifymode

### Iconifymodus

Hier haben Sie die Wahl zwischen den 2 Optionen, wie der Eagleplayer auf das **Iconify**

Möglichkeit, einen entsprechenden Menüeintrag in das Tools-Menü der Workbench einzufügen und andererseits ein Piktogramm auf der Workbench abzulegen, über das die Fenster des Eagleplayers reaktiviert werden können.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Iconifymode Zahl - Zahl: 0=Appicon,1=Appitem

Arexx:

status g icm - gibt den momentanen Zustand zurück (0...1)

Tooltype:

Iconifymode=Zahl - Zahl: siehe oben

## 1.72 loadalways

Lade immer

Hinter diesem Namen verbirgt sich die Möglichkeit, daß der Eagleplayer beim automatischen Laden der nächsten zu spielenden Datei nach **Songende** oder Timeout so lange die Liste der verfügbaren Dateien durchsucht, bis eine vom Eagleplayer als Musikstück erkannte Datei gefunden wird. Dies macht sich besonders dann vorteilhaft bemerkbar, wenn sich in dem aktuellen Verzeichnis auch beispielsweise noch Textdateien befinden. Bei ausgeschalteter Loadalways-Funktion würde der Eagleplayer ansonsten nach Ende der aktuellen Datei mit der Meldung "unbekanntes Format" stoppen.

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** und in einem **Programmierungsmodus** außer "kein neues Modul".

Anmerkung: Sollte der Eagleplayer keine abspielbare Datei finden, wird das Verzeichnis komplett immer wieder durchsucht. Um das Programm davon abzuhalten und diesen Vorgang zu stoppen, schalten Sie diese Option in diesem Fall am besten durch den zugehörigen Eintrag im Menü aus.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

LoadAlways Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g law - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Loadalways=Zustand - Zustand: siehe oben

## 1.73 loadfast

Lade in Fastram

Hiermit wird festgelegt, ob die Musikdateien ins Chipram oder (bei gesetztem Schalter) ins schnellere Fastram abgelegt werden.

Es wird dringend dazu geraten, diese Option auf jeden Fall einzuschalten, besonders wenn der 8 Bit oder 14 Bit-Amplifier verwendet wird, da das Chipram nun einmal besonders auf schnellen Maschinen um bis zu einem zehnfachen langsamer agiert als das Fastram !

Bei Musikformaten, die nicht mit den Amplifiern zusammenarbeiten, werden die Daten bei aktiviertem "LoadFast" automatisch ins Chipram umkopiert, wenn

notwendig.

Der einzige Grund, der mir momentan einfällt, diese Option auszuschalten, ist ein unaufgerüsteter Amiga500/2000 mit 68000-CPU, der einfach zu langsam wird, wenn die Module vom Fastmem aus abgespielt werden. In diesem Fall setzen Sie am besten durch den Amplifiermanager den Chipram-Amplifier an die erste Stelle der Amplifierliste, so daß durch das Abspielen aus dem Chipram auf die selbe Manier wie bei den 1.xx-Versionen des Eagleplayers ein Minimum an Rechenzeit beansprucht wird.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

LoadFast Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g lfa - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

LoadFast=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.74 loadbefore

lade schon nächstes

Durch diese Option wird der Eagleplayer veranlasst, sofort nach dem Einladen einer neuen Musikdatei die nächste aus der aktuellen Liste / dem aktuellen Verzeichnis abhängig von den Einstellungen des **Programmierungsmodus** zu laden, so daß es möglich wird, die Musikstücke pausenlos hintereinander abzuspielen.

Diese Funktion ist auch unter dem Namen "Doublebuffering" bekannt. Es ist hiermit vor allen Dingen auch möglich, Megamixe, die sich über mehrere Dateien erstrecken, korrekt abzuspielen. (z.B. "RAF Megamix2", "LSD-Total Kaos" oder "Max Mix"[Flash Prod.]

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** .

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Loadbefore Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g lbf - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

LoadBefore=Zustand - Zustand: siehe oben

Kleine Anmerkung: In der Doc zum Aplayer wurde behauptet, daß der Aplayer der erste Player mit solch einem Feature war. Interessanterweise gibt es diese Option im Eagleplayer schon länger als den Aplayer selbst...

## 1.75 loaddir

Lade Directory

Hiermit wird festgelegt, ob beim Laden einer Musikdatei das gesamte Verzeichnis eingelesen und im Pulldown-Menü (bzw. dem Dirlistviewer) zur Weiterverwendung angezeigt werden soll.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

LoadDir Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g ldd - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

LoadDir=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.76 Mastervolume

Mastervolume

Ist Mastervolume gesetzt, werden die Lautstärkewerte der Scopes verändert, das heißt in Abhängigkeit von der **Lautstärkeinstellung** des Eagleplayers ändert sich auch der Ausschlag von FFT-Analyzer, Levelmeter, Levelgraph, Quadrascope usw.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Mastervolume Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g msv - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Mastervolume=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

---

## 1.77 Menufont

wähle Menüfont

Öffnet das Schriftartenauswahlfenster zum Einstellen der Schriftart für die Pulldownmenüs außer dem **Modules-Menü** .

Tastatur:

-

Menü-Shortcut:

-

Arexx:

Menufont [Name.Größe] - bei Angabe eines Namens wird der entsprechende Font automatisch gesetzt, ansonsten der Requester aufgerufen

status g mfo - gibt den momentanen Font zurück

## 1.78 mintimeout

min. Spielzeit

Wenn Sie eine minimale Spielzeit eingeben, wird der Abspielvorgang durch die Einstellungen der **Songende** und/oder **max. Spielzeit** - Schalter erst nach Ablauf dieser eingegebenen Mindestzeit beendet. Dies macht sich vor allen Dingen Vorteilhaft bemerkbar, bei sehr kurzen Musikstücken, die man trotz allem ein paar Minuten hören will. Der Wert wird dabei in einem Requester in Sekunden angegeben, die Eingabe von "0" schaltet die Funktion aus.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

MinTimeout Zahl - Zahl: max. Spielzeit in Sekunden (0=aus)

Arexx:

status g ti2 - gibt den momentanen Wert zurück (in Sekunden)

Tooltpe:

MinTimeout=Zahl - Zahl: siehe oben

---

## 1.79 modulefault

Warnen - fehlerhafte Module

Bei eingeschalteter Option öffnet der Eagleplayer bei in sich fehlerhaften, zum Beispiel durch einen eventuellen Festplattencrash oder DFÜ-Übertragungsfehler zerstörten, Musikdateien einen Requester, der auf diesen Umstand hinweist.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

modulefault Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g mfa - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

modulefault=Zustand - Zustand: siehe oben

## 1.80 Modulesfont

wähle Modulesfont

Öffnet das Schriftartenauswahlfenster zum Einstellen der Schriftart für das Modules-Menü.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

Modulesfont [Name.Größe] - bei Angabe eines Namens wird der entsprechende Font automatisch gesetzt, ansonsten der Requester aufgerufen

status g mof - gibt den momentanen Font zurück

Beispiel:

```
/* kleines Arexx-Beispiel */
```

```
address `rexx_EP`
```

```
modulesfont `Helvetica.18`
```

---

## 1.81 NextPatt

next Pattern

Ein Pattern ist mit einem Notenblatt zu vergleichen, einige Soundsysteme bauen darauf auf, indem eine solche Liste von Befehlen nacheinander ausgeführt wird. Diese Befehlslisten (oder in Notenblätter, eben Patterns = Muster genannt) lassen sich meist beliebig kombinieren. Diese Funktion ist in der Lage, Patterns in dieser Abfolge sukzessive zu überspringen.

Ein Aufruf führt dazu, daß das nächste Pattern relativ zur aktuellen Position abgespielt wird, ist das letzte bereits erreicht, so wird das erste Pattern gespielt. Achtung: Bei aktivierter **Songend**-Funktion wird je nach Einstellung im **Program Mode** entweder eine neue Musikdatei geladen oder das Abspielen gestoppt, es sei denn, Sie haben **Lock Module** aktiviert.

Tastatur:

-

Menü-Shortcut:

A-x

Arexx:

NextPatt

## 1.82 NextSong

nächster Song

Es gibt Musikformate, die mehr als nur ein Musikstück enthalten können (z.B. Hippel,TFMX, OctaMED) Diese Musikstücke nennt man passenderweise "Subsongs", also Untersongs des Soundmodules. Diese Funktion schaltet auf den nächsten Subsong relativ zum gerade gespielten. Sollte der gespielte Subsong bereits der letzte sein, so hat diese Funktion keine Auswirkung. Wenn das Module nur einen Song besitzt, so wird der Button bzw. Menüeintrag gesperrt.

Tastatur:

-

Menü-Shortcut:

A-v

Arexx:

NextSong

---

## 1.83 NextModule

Next Module

Diese Funktion veranlaßt den Eagleplayer, das nächste Modul der aktuellen Liste zu laden, ist das aktuell gespielte Modul bereits das letzte, so wird wieder von vorn begonnen. Diese Funktion ist gesperrt, wenn sich in der Liste entweder weniger als 2 Module befinden oder die 'Load Dir'-Funktion ausgeschaltet ist.

Achtung: Sollte bei **Program Mode** "Random Module" eingestellt sein, so wird das nächste Modul zufällig aus der Liste ausgesucht, also nicht der Reihe nach gegangen.

Tastatur:

-

Menü-Shortcut:

A-n

Arexx:

NextModule

## 1.84 NoEngine

NoEngine

Diese Tooltype/Shellstart-Option sorgt dafür, daß die angegebenen Engines beim Start es Eagleplayers nicht mitgeladen werden. Sie wurde eingeführt, um in bestimmten Situationen sichtbare Engines temporär ausschalten zu können. Normalerweise sollten Sie diese Option nicht brauchen, da alle Zustände der zum Eagleplayer gehörenden Fenster automatisch mit dem Konfigurationsfile **gespeichert** werden.

Keyboard:

-

Menu-Shortcut:

-

Arexx:

-

Shellstart:

NoEngine=namen - namen: name(n) der engine(s), die nicht geladen werden soll(en), wenn Sie mehrere Engines angeben, so haben diese in Anführungszeichen zu stehen, durch SPACE getrennt.

Beispiel: Eagleplayer noengine="FFT-Analyzer Moduleinfo DirListViewer"

Tooltype:

Autosubsong=namen - namen: siehe oben

## 1.85 nogui

Eagleplayerbetrieb ohne Benutzeroberfläche

Diese, nur über Arexx bzw. Shell/Workbenchoptionen (oder manuellen Eintrag in die Konfiguration) aufrufbare Funktion ist für den Betrieb des Eagleplayers im Hintergrund bei minimalem Speicherverbrauch gedacht. Lesen Sie sich dieses Kapitel gründlich durch bevor Sie das ausprobieren, um die Verfahrensweise zu verstehen und unnötige Frustration zu vermeiden!

Bei eingeschalteter NoGUI-Option wird vom Eagleplayer keine Benutzeroberfläche mehr geladen, so daß das Programm nunmehr nur noch z.B. über Arexx ferngesteuert werden kann. Eine Anwendungsmöglichkeit dafür sind beispielsweise Custom-Buttons im Directoryopus oder eigens aus Arexx-Skripten zusammengestellte Toolmanager-Docks.

Zur Vorbereitung sollten Sie alle Fenster außer dem Hauptfenster schließen. Danach speichern Sie am besten die **Konfiguration** ab, beenden den Eagleplayer und starten ihn mit der **Shelloption** oder dem **Tooltpe** "nogui=yes" neu. Alternativ dazu können Sie dies auch von der Shell aus mit der Befehlssequenz `rx "address 'rexx_EP';nogui yes;saveconfig"` erledigen. In diesem Fall wird der NoGUI-Befehl auch in die Konfigurationsdatei mit aufgenommen. Bei Aufruf des Hotkeys (normal: CTRL-ALT-E) oder des "Show Interface"-Schalters im Commodities-Exchange wird ein Dateiauswahlfenster geöffnet, mit dem man ein oder mehrere Modules zum Abspielen anwählen kann. Beendet werden kann der Eagleplayer im NOGUI-Modus durch Commodities-Exchange oder das Arexx-Kommando **Quit** .

Ausschalten kann man die NOGUI-Option über die folgende Arexx-Sequenz einfach durch Kommandoeingabe in der Shell: `rx "address 'rexx_EP';nogui no"`. Danach wird das BifatGUI geöffnet und man kann wie gewohnt mit dem Eagleplayer arbeiten.

Wem das alles zu kompliziert erscheint und der aber trotzdem den Eagleplayer unsichtbar starten möchte, der sollte das Programm am besten mit der **Iconify=yes** Shelloption bzw. dem Tooltype starten.

Beispiel: `run >nil: Eagleplayer Iconify=yes`

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

NoGUI Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

`status g ngu` - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

NoGUI=Zustand - Zustand: siehe oben

## 1.86 notify

Dosnachricht

Bei aktiviertem Schalter überwacht der Eagleplayer das aktuelle Verzeichnis auf Änderungen und liest im Änderungsfall automatisch das neue Verzeichnis ein.

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** .

Achtung: Es kann in seltenen Fällen passieren, daß die Notify-Funktion mit der **EPDir** -Funktion kollidiert und dadurch ständig das Verzeichnis neu gelesen wird. Schalten Sie in diesem Fall entweder **EPDir** oder Notify aus.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Notify Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g not - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Notify=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.87 LoadModule

Load Module

Öffnet das Dateiauswahlfenster zur Anwahl eines anzuhörenden Musikstückes, es dürfen auch mehrere ausgewählt werden (Shift-Taste gedrückt halten); ist dies der Fall, wird die Liste der ausgewählten Dateien übernommen und kann über die Funktionen 'previous Module' und 'next Module' genutzt werden; weiterhin bietet sich beim Eagleplayer die Möglichkeit, die Musikstücke direkt über das Pulldownmenü aufzurufen (Modules-Directory). Wird nur ein Modul selektiert, so wird je nach Einstellung das gesamte Verzeichnis eingelesen, dabei werden die Dateien, die auf Grund Iheres Namens nicht als Musikdatei in Betracht kommen, wie z.B. '.readme', '.txt', '.c' oder '.asm' automatisch versteckt ( **HideAll** ), also nicht im Modules-Menü berücksichtigt

Tastatur:

-

Menü-Shortcut:

A-1

Arexx/Shellstart:

loadmodule name1 name2 ... - der Eagleplayer lädt das erste namentlich angegebene Modul und lädt bei nur einem Argument dessen Verzeichnis ein, bei mehreren angegebenen Namen werden nur diese ins Modules-Menü übernommen

Beispiel:

```
loadmodule "dh2:Protracker/Mod.Bifat_deadlocked EBM"
```

Tooltype:

loadmodule=name - analog zu obigem Beispiel, nur mit Gleichheitszeichen zur Trennung zwischen Kommando und Argument

Arexx Status Kommandos: [Siehe hier](#)

## 1.88 LoadPlayer

LoadPlayer / AddPlayer

Diese Funktion ermöglicht es, per Hand einen zusätzlichen Abspieler einzuladen.

Im Allgemeinen ist dies bei aktiviertem Playerloader, sowie **Playerbatch** nicht notwendig. Es wird bei Anwahl vom Menü aus ein Dateiauswahlfenster geöffnet, in welchem ein oder mehrere Abspieler angewählt werden können, die dann der Liste hinzugefügt werden.

Anmerkung: Der Manager zeigt bereits alle im "Eagleplayers"-Verzeichnis befindlichen Abspieler an, die über diesen durch einfachen Doppelklick auf den entsprechenden Eintrag auf einfachste Weise nachgeladen/entfernt werden können.

Tastatur:

a

Menü-Shortcut:

-

Arexx:

LoadPlayer [Pfad und Name] - bei Angabe des Dateipfades wird der Player sofort geladen, ansonsten öffnet sich analog

zur Menüfunktion der Dateirequester

siehe auch: [LoadPlayerDir](#)

---

## 1.89 LoadPlayerDir

LoadPlayerDir

Diese Funktion ermöglicht es, per Hand eine Anzahl zusätzlicher Abspieler einzuladen. Im Allgemeinen ist dies bei aktiviertem Playerloader, sowie **Playerbatch** nicht notwendig. Dabei werden alle im angegebenen Verzeichnis vorhandenen Abspieler eingeladen und der Playerliste hinzugefügt.

Anmerkung: Der Manager zeigt bereits alle im "Eagleplayers"-Verzeichnis befindlichen Abspieler an, die über diesen durch einfachen Doppelklick auf den entsprechenden Eintrag auf einfachste Weise nachgeladen/entfernt werden können.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

LoadPlayerDir [Pfad] - bei Angabe des Dateipfades wird der Player sofort geladen, ansonsten öffnet sich ein

Dateirequester

status g dir - liefert den aktuellen AbspielerPfad zurück

siehe auch: **LoadPlayer**

## 1.90 Enable

Enable

Diese Funktion ermöglicht es, über Arexx einzelne Player ein/auszuschalten. Dabei wird im ersten Argument die Nummer des Players und im zweiten Argument der Zustand angegeben.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

enable Num Status - "Num"=Nummer des Players, "Status" = 1 für erlaubt,

0 für gesperrt

status p ena - gibt zurück, ob der entsprechende Player erlaubt ist

## 1.91 LoadEngine

Engine laden

Öffnet das Dateiauswahlfenster zum Hinzufügen einer/mehrerer Engine(s) zu den momentan verfügbaren.

Anmerkung: Alle im Engines-Verzeichnis befindlichen Engines werden bereits im Pulldown-Menü, sowie im Manager angezeigt und können durch einfache Anwahl bzw. Doppelklick im Manager (de)aktiviert werden, so daß diese Funktion im Normalfall nicht in Anspruch genommen werden muß.

Tastatur:

e

Menü-Shortcut:

A-u

Arexx:

LoadEngine [Name] - bei Angabe eines Namens wird die entsprechende Engine geladen, ansonsten analog zur Menüfunktion der Dateirequester geöffnet

## 1.92 overwrite

Überschreiben

Bei eingeschalteter Funktion wird vor dem Überschreiben bestehender Dateien beim **Abspeichern** der aktuellen Musikdatei ein Abfragerequester geöffnet, um das versehentliche Löschen dieser gleichnamigen Dateien vermeiden zu können.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Overwrite Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g owv - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Overwrite=Zustand - Zustand: siehe oben

## 1.93 lockmodule

Modul behalten

Dieser Schalter sorgt dafür, daß das zur Zeit abgespielte Musikstück nicht automatisch angehalten bzw. durch das **nächste** der Liste ersetzt oder aus dem Speicher entfernt wird, unabhängig von den Einstellungen für **TimeOut** , **Songend** usw.

Die Funktion wird deaktiviert über die wiederholte Anwahl des Schalters oder das Anwählen des nächsten Musikstückes durch den Anwender.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

LockModule Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g mlo - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

LockModule=Zustand - Zustand: siehe oben

## 1.94 parentdir

ParentDir

Bei Aufruf wird das übergeordnete Verzeichnis relativ zum aktuellen eingelesen.

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** .

Tastatur:

-

Menü-Shortcut:

-

Arexx:

ParentDir

## 1.95 password

Ändere Passwort

Beim Laden einer durch ein Passwort geschützten Datei als erstes automatisch ein Entschlüsselungsversuch mit dem hier durch den Requester eingestellten Passwort gestartet. Falls der Entschlüsselungsvorgang durch ein nicht richtiges Passwort fehlschlagen sollte, wird wie im Normalfall auch, ein Requester

geöffnet, in dem dann nach einem manuell einzugebenden Passwort gefragt wird.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Password String - String: das neue Passwort als Zeichenkette

Arexx:

status g paw - gibt das momentane Passwort als String oder "RESULT" für

kein Passwort zurück

Tooltipe:

Password=String - String: siehe oben

## 1.96 Play

Spielen / Pause

Der Eagleplayer stoppt oder startet das aktuelle Musikstück, je nachdem ob gerade gespielt wurde oder nicht. Anmerkung: Nachdem der Eagleplayer aus dem "Pause"-Modus in den "Play"-Modus wechselt, wird von der aktuellen Position weitergespielt, nicht vom Anfang, siehe [Replaysong](#). Der entsprechende Schalter für die Play-Funktion wurde bei BifatGUI und Eagleplayer1.x-GUI dem eines Kassettendecks bzw. CD-Players nachempfunden und befindet sich in etwa in der Mitte des Bedienfensters.

Tastatur:

-

Menü-Shortcut:

A-p

Arexx:

Play

## 1.97 vorheriges Modul

vorheriges Modul

Diese Funktion veranlaßt den Eagleplayer, das vorherige Modul der aktuellen Liste zu laden, ist das aktuell gespielte Modul bereits das erste, so wird dann das letzte geladen. Diese Funktion ist gesperrt, wenn sich in der Liste entweder weniger als 2 Module befinden oder die 'Load Dir'-Funktion ausgeschaltet ist.

Achtung: Sollte bei [Program Mode](#) "Random Module" eingestellt sein, so wird das

nächste Modul zufällig aus der Liste ausgesucht, also nicht der Reihe nach gegangen.

Tastatur:

-

Menü-Shortcut:

A-b

Arexx:

PrevModule

## 1.98 Playerbatch

Playerbatch

Die Playerbatch-Funktion ermöglicht es, die Abspieler erst dann nachzuladen, wenn sie auch wirklich benötigt werden, wodurch Speicherbedarf und Ladezeit des Eagleplayers um einiges reduziert werden können. Weitere Infos finden Sie [hier](#).

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Playerbatch Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g eab - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Playerbatch=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.99 prefix

Vorsilbe

Beim **Abspeichern** der aktuellen Musikdatei wird ein Präfix vor den Namen des Moduls gesetzt, falls a) das Modul einen eigenen (im Modul gespeicherten) Namen besitzt und b) dies durch den benutzten Replayer unterstützt wird, sprich dieser auch ein entsprechendes Präfix parat hat.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Prefix Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g upr - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Prefix=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

---

## 1.100 PrevSong

vorheriger Song

Es gibt Musikformate, die mehr als nur ein Musikstück enthalten können (z.B. Hippel,TFMX,OctaMED) Diese Musikstücke nennt man passenderweise "Subsongs", also Untersongs des Soundmodules. Diese Funktion schaltet auf den vorherigen Subsong relativ zum gerade gespielten. Sollte der gespielte Subsong bereits der erste sein, so hat diese Funktion (logischerweise) keine Auswirkung. Wenn das Module nur einen Song besitzt, so wird der Button gesperrt.

Tastatur:

-

Menü-Shortcut:

A-c

Arexx:

PrevSong

## 1.101 PrevPatt

vorheriges Pattern

Ein Pattern ist mit einem Notenblatt zu vergleichen, einige Soundsysteme bauen darauf auf, indem eine solche Liste von Befehlen nacheinander ausgeführt wird. Diese Befehlslisten (oder Notenblätter, eben Patterns = Muster genannt) lassen sich meist beliebig kombinieren. Diese Funktion ist in der Lage, Patterns in dieser Abfolge zu überspringen und somit im Musikstück zurückzuspringen. Ein Aufruf führt dazu, daß das vorherige Pattern relativ zur aktuellen Position abgespielt wird, ist das erste bereits erreicht, so wird das letzte Pattern gespielt.

Tastatur:

-

Menü-Shortcut:

A-y

Arexx:

PrevPatt

## 1.102 programmode

### Programmierung

Die hier gemachten Einstellungen ändern das Verhalten des Eagleplayers in Bezug auf das Ende des Abspielens der aktuellen Musikdatei. Die folgenden 4 Einstellungen sind möglich:

kein neues Modul - Bei einem automatisch erkannten **Songende** wird kein neues Module geladen und das Abspielen stoppt.

vorheriges Modul - Wenn Songend oder **PlayTime** eingeschaltet sind, wird nach Abspielende das vorherige Module geladen.

nächstes Modul - Wenn Songend oder PlayTime eingeschaltet sind, wird das nächste Module geladen.

RandomPlay - Wenn Songend oder PlayTime eingeschaltet sind, wird ein zufällig gewähltes Module geladen.

Bei Betätigung von Next- oder PrevModule wird ebenfalls ein zufällig ausgewähltes Module abgespielt.

Dabei werden erst alle Module des geladenen Verzeichnisses gespielt, ehe ein Module doppelt geladen wird. Allerdings ist es möglich, ein Module über das Pull-Down Menü anzuwählen, ohne daß der interne Zähler sich das merkt, d.h. es kann per Randomplay auch nochmals gespielt werden. Eine weitere Möglichkeit des zufälligen Abspielens wird durch die **Directorysprünge** -Funktion gegeben

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** .

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Programmode Nummer - Nummer: 0 = no next module

1 = next module

2 = previous module

3 = random module

Arexx:

status g pmo - gibt den momentanen Zustand zurück (0...3)

Tooltype:

Programmode=Nummer - Nummer: siehe oben

## 1.103 pubscreen

Pubscreen

Diese Funktion erlaubt die Änderung, auf welchem Bildschirm der Eagleplayer sich zu öffnen hat. Bei Aufruf vom Pulldown-Menü aus wird ein Stringrequester geöffnet, in welchem der vollständige Name unter Beachtung der Groß-Kleinschreibung eingegeben werden kann. Alternativ dazu sei an dieser Stelle der Pubscreenselector empfohlen.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

pubscreen name - der Eagleplayer wird auf dem Pubscreen mit dem Namen `name` geöffnet

Beispiel: pubscreen "DOPUS.1"

Arexx:

status g pub - gibt den Namen des aktuellen PublicScreens zurück

Tooltype:

pubscreen=name - analog zu obigem Beispiel, nur mit Gleichheitszeichen zur Trennung

## 1.104 printtext

Printtext

Dieses Arexx-Kommando veranlaßt den Eagleplayer, im Hauptfenster (Bifat- oder Eagleplayer1.x-GUI) einen Text in der Statuszeile auszugeben.

Arexx:

Printtext String - String: beliebiger Text

Beispiel:

```
/* simple Example */
```

```
address `rexx_EP`
```

```
printtext `I I*ve Eagleplayer`
```

## 1.105 protect

Modul schützen

Es gibt bei einigen der vom Eagleplayer zum **Abspeichern** der aktuellen Datei angebotenen Komprimierungsmethoden (Powerpacker und Crunchmania) die Möglichkeit, die abgespeicherte Datei mit einem Passwort zu schützen. Diese Option schaltet die Abfrage für ein Passwort beim Speichern der Datei ein.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Protect Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g prt - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Protect=Zustand - Zustand: siehe oben

## 1.106 protectionbits

Protectionbits

Beim **Abspeichern** von Musikdateien durch den Eagleplayer werden diese auf "nur lesen" gesetzt, so daß sie nicht einfach versehentlich durch den Anwender gelöscht werden können.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Protectionbits Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g pbi - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Protectionbits=Zustand - Zustand: siehe oben

## 1.107 QuickStart

Sofort starten

Bei aktiviertem Menüeintrag wird die Musikdatei sofort nach dem Laden abgespielt, ansonsten muß das Abspielen manuell über die **Play** -Funktion aufgerufen werden.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Quickstart Zustand - [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g qst - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Quickstart=Zustand - [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.108 quit

Ende

Wollen Sie diese (unserer Meinung nach überflüssige :- ) Funktion wirklich benutzen ? Hiermit wird der Eagleplayer einfach schlicht und ergreifend beendet.

Achtung! Im Gegensatz zu den vorherigen Eagleplayerversion (<2.0) wird der Eagleplayer beim Betätigen des Schließbuttons des Fensters automatisch beendet, aus dem einfachen Grunde, daß es zumindestens beim BifatGUI einen eigenen **Iconify** -Button gibt.

Tastatur:

-

Menü-Shortcut:

A-q

Arexx:

quit

## 1.109 randomnessong

Untersongs zufällig

Bei Soundsystemen, deren Musikdateien mehrere Musikstücke enthalten, wird der erste abgespielte Untersong zufällig ausgewählt (siehe auch **hier** ).

---

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Randomsong Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g rms - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Randomsong=Zustand - Zustand: siehe oben

## 1.110 randomstart

Start mit Zufallsmodul

Wenn diese Funktion eingeschaltet ist, sucht sich der Eagleplayer beim Start nicht die mit der **Konfiguration** eingestellte Datei aus dem aktuellen Verzeichnis heraus, sondern wählt sich irgendeine andere statt dessen.

Der Vorteil dabei liegt darin, daß man beim Start des Eagleplayers nicht immer eine Datei selbst suchen muß bzw. immer dieselbe Musik beim Start hören muß.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

RandomStart Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g rst - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Randomstart=Zustand - Zustand: siehe oben

## 1.111 ReplaySong

Replay Song

Das geladene Musikstück wird noch einmal von vorn gespielt.

Tastatur:

-

Menü-Shortcut:

A-r

Arexx:

ReplaySong

---

## 1.112 rescandir

Directory neu laden

Beim Aktivieren dieser Funktion wird das aktuelle Verzeichnis unter Beachtung des Zustandes des **Hideall** -Schalters komplett neu eingelesen, wobei dabei die Einstellungen von **EPDir** ignoriert werden.

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** .

Tastatur:

r

Menü-Shortcut:

-

Arexx:

RescanDir

## 1.113 rasterlines

setze Rasterzeilen

In der Einstellung "auf Rasterzeilen warten" des **Wartemodus** werden zum zeitgerechten Abspielen der Musikdateien mehrere Bildschirmzeilen durchgezählt, um den Wartevorgang zeitlich korrekt auf allen Prozessoren gewährleisten zu können. Der vorgeschlagene Wert, der im Requester eingegeben werden sollte, ist 8. Allerdings ist die Zeitspanne, wie lange eine Rasterzeile dauert, auch nicht genormt und hängt vom aktuellen Bildschirmmodus des Amigas ab.

Einen zu geringen Wert an zu wartenden Rasterzeilen erkennen Sie daran, daß ab und zu Noten der zu spielenden Musikdatei verschluckt werden. Erhöhen Sie in diesem Fall den Wert oder schalten Sie auf " **Automatisch warten** ".

Beispielwerte bei entsprechendem Bildschirmmodus:

PAL/NTSC/EURO36 - 8...10

DBLPAL/DBLNTSC/MULTISCAN/EURO72 - 18...30

Bei Verwendung von Grafikkarten ist der letzte, vor dem Öffnen des Cybergraphics-Bildschirms aktive Amiga-Bildschirm relevant (meist PAL oder NTSC), siehe auch hier.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Rasterlines Zahl - Zahl: Anzahl zu wartender Zeilen

Arexx:

status g ral - gibt den momentanen Wert als Zahl zurück,

Tooltpe:

Rasterlines=Zahl - Zahl: siehe oben

## 1.114 Save Configuration

### Konfiguration Speichern

Diese Funktion öffnet das Dateiauswahlfenster zur Angabe des Namens für die Einstellungsdatei und speichert danach sämtliche Einstellungen des Eagleplayers ab. Dazu gehören alle Einstellungen von Positionen, Größen und Zuständen des Hauptfensters, der Engines, Abspieler, Sondereinstellungen von Abspielern/Engines, das aktuelle Modulesverzeichnis und das gerade abgespielte Musikstück.

Tip: Wenn Sie beim Start nicht unbedingt automatisch ein Musikstück abspielen wollen, wohl aber ein bestimmtes Modulverzeichnis zur Verfügung stehen haben möchten, wechseln Sie in dieses Verzeichnis, **werfen** das momentan abgespielte Musikstück raus, achten darauf, daß **Randomstart** ausgeschaltet ist und speichern dann die Konfiguration.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

SaveConfig [pfad+name] - bei Angabe eines Pfades/Namens wird die Konfiguration sofort ohne Dateirequester unter dem Namen abgespeichert

## 1.115 savedir

### setze Speicherdirectory

Mittels dieser Funktion kann das Verzeichnis gesetzt werden, welches vom Eagleplayer beim **Abspeichern** der aktuellen Datei im Dateiauswahlfenster automatisch vorgegeben wird. Der Hauptgrund für diese Funktion liegt allerdings im Vorhandensein der Automatisch Speichern - Option.

Anmerkung: wird beim manuellen Speichern einer Datei durch den Anwender ein anderer Pfad im Dateiauswahlfenster eingegeben, so wird dieser fortan verwendet.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

SaveDir [Pfad] - Pfad: entweder Verzeichnispfad oder ansonsten wird der Requester aufgerufen

Arexx:

status g sap - gibt den momentanen Pfad zurück

Tooltpe:

SaveDir=Pfad - Pfad: siehe oben

## 1.116 safesave

sicher Speichern

Wenn diese Funktion eingestellt wird, **speichert** der Eagleplayer die aktuelle Datei vorerst unter einem anderen Namen ab und benennt sie erst dann um, wenn die Datei wirklich komplett auf dem Datenträger untergebracht werden konnte.

Somit wird es vermieden, daß eine existierende Datei überschrieben wird und sich im Nachhinein herausstellt, daß ein Schreibvorgang durch Fehler des Datenträgers fehlschlug und die Daten verloren sind.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

SafeSave Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g sfs - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

SafeSave=Zustand - Zustand: siehe oben

## 1.117 samplemode

Instrumentenmodus

Es gibt bei einigen der vom Eagleplayer zum **Abspeichern** der aktuellen Datei angebotenen Komprimierungsmethoden (Powerpacker, Crunchmania und LH.Library) die Möglichkeit, einen speziell auf Soundmodules angepaßten, sogenannten Delta-Modus zu aktivieren, der in den meisten Fällen einen erheblichen Komprimierungsgewinn erbringen kann. Und genau dieses Feature aktiviert dieser Schalter.

Anmerkung: Der Samplemode für Powerpacker wurde von uns inoffiziell eingeführt und der Eagleplayer ist somit momentan das einzige Programm, welches solcherart gepackte Dateien laden kann. Ein Entpackslave für die xfd.library von Georg Hörmann ist in Vorbereitung

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Samplemode Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g smo - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Samplemode=Zustand - Zustand: siehe oben

## 1.118 saveasprotracker

als Protracker speichern

Diese Option wurde vor allen Dingen für die internen Abspieler des Eagleplayer vorgesehen (Player5.x/6.x, Trackerpacker, Noisepacker usw.). Bei Aktivierung werden diese Modules im normalen Protrackerformat **gespeichert**, welches durch die gängigen Trackerprogramme wie Protracker und OctaMED geladen und verarbeitet werden kann. Somit erübrigen sich Programme wie Noiseconverter oder Prowizard, besonders in Verbindung mit dem Eagleexotic.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

SaveasProtracker Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g spt - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

SaveasProtracker=Zustand - Zustand: siehe oben

## 1.119 savemodule

Modul speichern

Öffnet das Dateiauswahlfenster zur Anwahl eines Dateinamens, unter dem das aktuelle Musikstück abzuspeichern ist. Sollte es sich um ein Musikstück mit mehreren zugehörigen Dateien handeln (z.b. TFMX), so werden diese automatisch mit in dem ausgewählten Verzeichnis untergebracht.

Der Eagleplayer ist in der Lage, die Dateien auch automatisch in gepacktem Zustand zu speichern.

Diese Funktion erfordert die registrierte Version des Eagleplayers.

Tastatur:

-

Menü-Shortcut:

A-w

Arexx:

SaveModule [name] - der Eagleplayer speichert die Datei unter dem Namen

`name` bzw. öffnet den Filerequester, wenn kein Argument angegeben wird

Beispiel: savemodule "ram:Mod.DMR\_folx-vagen"

## 1.120 scanalways

Lade Directory immer

Bei aktiviertem Schalter wird jedes Mal, wenn über Dateiauswahlfenster, Arexx, oder eine externe Engine eine Musikdatei geladen wird, das Verzeichnis komplett neu gelesen.

Im Normalfall sollte diese Funktion ausgeschaltet bleiben. Wenn sie aber doch aktiviert wird, kann zumindestens **EPDir** abgeschaltet werden.

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** .

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

ScanAlways Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g alw - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

ScanAlways=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.121 Scrollfont

wähle Rollfont

Öffnet das Schriftartenauswahlfenster zum Einstellen der Schriftart für den Scroller der Benutzeroberflächen (BifatGUI und Eagleplayer1.x-GUI).

Tastatur:

-

Menü-Shortcut:

-

Arexx:

Scrollfont [Name.Größe] - bei Angabe eines Namens wird der entsprechende Font automatisch gesetzt, ansonsten der Requester aufgerufen

status g scf - gibt den momentanen Font zurück

Beispiel:

```
/* kleines Arexx-Beispiel */
```

```
address `rexx_EP`
```

```
scrollfont `Helvetica.18`
```

---

## 1.122 Scrollinfos

Rolle Informationen

Bei aktiviertem Menüeintrag werden alle Informationstexte in das Fenster hineingescrollt, ansonsten einfach hineingeschrieben.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Scrollinfos Zustand - [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g sci - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Scrollinfos=Zustand - [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.123 slowerspeed

Langsamer

Die **Abspielgeschwindigkeit** wird um einen Schritt verringert.

Tastatur:

-

Menü-Shortcut:

A--

Arexx:

Slowerspeed

## 1.124 scrolltext

Scrolltext

Dieses Arexx-Kommando veranlaßt den Eagleplayer, im Hauptfenster (Bifat- oder Eagleplayer1.x-GUI ) einen Text in der Statuszeile durchzurollen.

Allerdings muß dazu die Option **Scrollinfos** eingeschaltet sein.

Arexx:

Scrolltext String - String: beliebiger Text

Beispiel:

```
/* simple Example */
```

```
address 'rexx_EP'
```

```
scrollinfos yes
```

```
scrolltext 'I I*ve Eagleplayer'
```

## 1.125 smallmodule

Warnen - kleine Module

Bei eingeschalteter Option öffnet der Eagleplayer bei zu kurzen Musikdateien einen Requester, der auf diesen Umstand hinweist, da einige Soundsysteme (z.B. Deltamusic2) bei zu kurzen Modulen durchaus erhebliche Probleme verursachen können.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Smallmodule Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g smo - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Smallmodule=Zustand - Zustand: siehe oben

## 1.126 softint

Softinterrupt

Diese Option wurde eingebaut, um negative Wechselwirkungen mit anderen zeitkritischen Programmen (z.B. Modem-Betrieb) möglichst zu vermeiden.

Da der **CIA-Interrupt** eine höhere Priorität als der für den

Modembetrieb zuständige Interrupt hat, kann es normalerweise zu Übertragungsfehlern führen. Zur Minimierung dieser Gefahr wird nach dem Aufruf des CIA-Interrupts ein sogenannter Softinterrupt gestartet, welcher erst nach eventuellen Anfragen der seriellen Schnittstelle abgearbeitet wird. Sollten bei eingeschaltetem Softint jedoch immer noch Probleme auftauchen (z.B. bei langsamen Prozessoren), reduzieren Sie am besten die serielle Übertragungsrage in Ihrem Terminalprogramm.

Bei der Einstellung "Timer.device" als **Timingmodus**

Schalter.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Softint Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g sit - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Softint=Zustand - Zustand: siehe oben

## 1.127 songend

Songende

Wird durch den Abspieler ein Ende der aktuellen Musikdatei erkannt, so wird bei eingeschalteter Funktion das Abspielen gestoppt und je nach Einstellung im Programmmoder weitergearbeitet, sprich also eine neue Datei geladen oder es wird auf Benutzereingaben gewartet. Welche Abspieler die Songende-Funktion unterstützen, können Sie im Manager erfahren.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Songend Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g end - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Songend=Zustand - Zustand: siehe oben

## 1.128 Songname

Songname

Bei aktiviertem Menüeintrag wird in der Titelzeile, sowie im Scroller nicht der Name der geladenen Datei, sondern der tatsächliche, in der Datei selbst zu findende Name des Musikstückes angezeigt. Dies funktioniert allerdings nur bei Dateiformaten, die auch so etwas unterstützen (z.B. Protracker, (OctaMED, Screamtracker)

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Songname Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g sna - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

Songname=Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

## 1.129 speed

Geschwindigkeit

Dieser Regler ist von Belang für die Abspielgeschwindigkeit der spielenden Musikdatei, steht dieser in der Mitte, ist die Normalgeschwindigkeit eingestellt, eine Bewegung nach oben erhöht die Geschwindigkeit, eine Bewegung nach unten verlangsamt sie. Sollte der entsprechende aktuelle Replayer keine Geschwindigkeitsänderung unterstützen, so werden dieser Schieberegler und die entsprechenden Menüeinträge gesperrt.

Einige Abspieler besitzen zusätzlich zur Änderung der Abspielgeschwindigkeit die Möglichkeit, gleichzeitig die Tonhöhe zu variieren (englisch: Pitchup/down), so daß der Effekt entsteht, als wenn man einen Plattenspieler schneller/langsamer laufen lassen würde. (z.B. Protracker, AIFF, Oktalyzer, David Whittaker).

Die über das Menü aufrufbaren Möglichkeiten umfassen die Veränderung der Geschwindigkeit um jeweils einen Schritt schneller/langsamer, sowie die Rücksetzung auf Normalstellung.

Tastatur:

-

Menü-Shortcut:

A-+ - höhere Geschwindigkeit um einen Schritt

A-- - geringere Geschwindigkeit um einen Schritt

A-# - Normalgeschwindigkeit

Arexx/Shellstart:

Speed Zahl - Zahl: -25...0...25 (-25=langsam;0=normal, 255=schnell)

Arexx:

status g spe - gibt den momentanen Wert zurück (-255...255)

Anmerkung: da Arexx Probleme mit negativen Zahlen

hat, werden für Werte <0 Zahlen von 65535 (= -1) bis 65511 (= -25) zurückgegeben, die vom Arexx-Programm entsprechend umgerechnet werden müssen.

Tooltipe:

Speed=Zahl - Zahl: siehe oben

## 1.130 Stop

Stop

Diese Funktion hält das aktuelle Musikstück an. Wenn danach zu einem beliebigen Zeitpunkt die **Play/Pause** Taste gedrückt wird, beginnt das Stück wieder am Anfang zu spielen.

Tastatur:

-

Menü-Shortcut:

A-s

Arexx:

Stop

## 1.131 Subsong

Subsong

Bei Modulen mit mehreren **Subsongs** kann man durch diese Funktion angeben, welcher dieser Subsongs abgespielt werden soll. Als Argument wird eine Zahl erwartet.

Tastatur:

-

Menü-Shortcut:

-

Arexx:

Subsong Zahl - Angabe des Subsongs als Zahl

status g snd - gibt den aktuellen Subsong zurück

## 1.132 timingmode

Timing

Der Timingmodus gibt an, in welcher Art und Weise das Abspielen der Musikdatei zeitlich gesteuert wird. Im Moment gibt es zwei grundlegende Möglichkeiten des Timings: CIA und VBlank. Der CIA-Timer des Amigas arbeitet wesentlich genauer und in jeder Maschine mit der gleichen Geschwindigkeit, während der VBlank (= Rasterstrahl-Interrupt) von der momentanen Bildschirmfrequenz der OCS/ECS/AGA-Amigas abhängt. Normalerweise sollte CIA-Timing eingeschaltet sein. Es gibt nur sehr wenige (und meist alte) Soundmodules, die unbedingt VBlank verlangen (MOD.Face another day, MOD.Techdust). An dieser Stelle sei allerdings auf den Noiseconverter (Werbung in eigener Sache) hingewiesen,

der solche Module mittels der "Tempo berichtigen"-Funktion wieder in Ordnung bringen kann.

Sollte durch ein anderes, die CIA-Timer benutzendes Programm der benötigte Interrupt nicht frei sein, wird eine Fehlermeldung ausgegeben.

Seit der V2.0 des Eagleplayers kam zusätzlich die Möglichkeit der Nutzung des Timer.device hinzu. Dadurch, daß des Programm somit nicht mehr auf die direkte Verwendung der Hardware-Interrupts angewiesen ist, sollten sich so einige Probleme mit bestimmten anderen Programmen und Hardware-zusätzen erübrigen. Allerdings wird dabei etwas mehr Rechenzeit gebraucht, als wenn mit den Hardware-Timern gearbeitet wird.

Sollten Sie trotz der obigen Erklärungen mit diesen Optionen immer noch nichts anfangen können, so lassen Sie am besten die Finger davon, mit den Grundeinstellungen ab Eagleplayer V1.53 hat es bisher keine Probleme gegeben. (A500-A4000/60)

siehe auch [Waitingmode](#)

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Timingmode Zahl - Zahl: der zu verwendende Modus

0 = CIA

1 = VBlank

2 = Timer.device

Arexx:

status g tmd - gibt den momentanen Zustand als Zahl zurück,

0 = CIA

1 = VBlank

2 = Timer.device

Tooltype:

Timingmode=Zahl - Zahl: siehe oben

### 1.133 timemode

Zeitmodus

Diese Funktion sorgt dafür, daß bei Soundsystemen, bei denen ein **Ende** des Musikstückes automatisch erkannt wird, die Einstellungen für die maximale **Spielzeit** ignoriert werden. Somit ist es möglich, daß einerseits Musikstücke nicht ewig laufen - für den Fall, daß der Abspieler kein Ende des Stückes

unterstützt - und andererseits die restlichen Musikstücke nicht mitten im Abspielvorgang abgebrochen werden.

Diese Funktion macht nur Sinn in Verbindung mit **LoadDir** und bei eingeschaltetem **Songende** .

siehe auch min. Spielzeit...

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Timemode Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g tmo - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

Timemode=Zustand - Zustand: siehe oben

## 1.134 timeout

max. Spielzeit

Wenn Sie eine maximale Spielzeit eingeben, wird der Abspielvorgang nach genau dieser Zeitspanne gestoppt und je nach Einstellung des **Programmierungsmodus** weiter verfahren. Der Wert dafür wird in einem Requester in Sekunden angegeben, der Wert "0" schaltet die Funktion aus.

siehe auch **min. Spielzeit** und **Zeitmodus**

Tastatur:

-

Menü-Shortcut:

A-z

Arexx/Shellstart:

Timeout Zahl - Zahl: max. Spielzeit in Sekunden (0=aus)

Arexx:

status g tim - gibt den momentanen Wert zurück (in Sekunden)

Tooltype:

Timeout=Zahl - Zahl: siehe oben

## 1.135 togglefilter

Toggle Filter

Schaltet den Zustand des Tiefpaßfilters des Amigas um. Das heißt, war der Filter vorher aus, wird er eingeschaltet und umgekehrt.

Tastatur:

-

Menü-Shortcut:

A-f

Arexx:

togglefilter

## 1.136 Voice

Voices

Diese Funktion läßt die getrennte Ein/Ausschaltung der jeweiligen Stimmen der Musikdatei zu. Bei den Standard-Amigamodulen sind das 4 Stimmen, mehrstimmige Soundmodule können bis zu 32 (oder mehr) Stimmen besitzen, was derzeit (noch) nicht umschaltbar ist. Erscheint der Schalter "reingedrückt" (und unterscheidet sich in den Rändern somit von den anderen Schaltern) , ist die jeweilige Stimme ausgeschaltet, ansonsten an, bei den Menüs bedeutet ein gesetzter Haken eine eingeschaltete Stimme.

Tastatur:

-

Menü-Shortcut:

A-1

A-2 Ein-Ausschalten der Stimmen 1 - 4

A-3

A-4

A-d Normaleinstellung, sprich alle Stimmen eingeschaltet und auf maximaler Lautstärke

Arexx:

voice x y - x=Nummer (1...4),y=Zahl für Lautstärkewert von 0(=aus) bis 255(=max.)

status g voi x - x=Nummer (1...4), Resultat Zahl, 0 = aus, 255=max Lautstärke

## 1.137 volume

Lautstärke

Regelung der Lautstärke der abgespielten Datei. Sollte der entsprechende aktuelle Replayer keine Lautstärkeunterstützung besitzen, so wird dieser Schieberegler (bzw. die entsprechenden Menüeinträge) gesperrt.

Das BifatGUI besitzt noch 2 Extraschalter für die Lautstärke, der linke davon setzt beim ersten Anklicken die Lautstärke auf 0 (=Mute-Funktion) und bei wiederholter Anwahl auf den vorherigen Wert zurück. Der rechte Button macht dies analog, allerdings wird bei der ersten Anwahl die Lautstärke auf den Maximalwert gesetzt.

Die über das Menü aufrufbaren Möglichkeiten umfassen die Verringerung/Erhöhung der Lautstärke um jeweils eine Position, sowie die Rücksetzung auf Maximallautstärke.

Tastatur:

-

Menü-Shortcut:

A-6 = Lautstärke um 1 verringern

A-7 = Lautstärke um 1 erhöhen

A-5 = Lautstärke auf Voreinstellungswert setzen (=Maximum)

Arexx/Shellstart:

Volume Zahl - Zahl: 0...255 (0=aus, 255=maximum)

Anmerkung: der Eagleplayer rechnet intern mit 256

Lautstärkestufen, gibt aber dem Nutzer momentan einen Bereich von 0..64 als

Wert an, was der Amiga-Hardwarebeschränkung

Rechnung trägt.

Arexx:

status g vol - gibt den momentanen Wert zurück (0...255)

Tooltype:

Volume=Zahl - Zahl: siehe oben

## 1.138 waitingmode

Waitingmode

Aufgrund der Arbeitsweise der Amiga-Hardware ist es notwendig, beim Starten neuer, durch den Paula-Soundchip abgespielter Daten, eine kurze Zeitspanne zu warten. Der Eagleplayer bietet nun eine Reihe von Möglichkeiten an.

Im Normalfall wird die Option "Automatisch warten" empfohlen, die sich die günstigste, zu dem benötigten Zeitpunkt verfügbare, Methode aussucht. Bei Verwendung einiger Amplifier (8 Bit, 14 Bit Amplifier) sind diese Einstellungen für den Abspielvorgang relevant und daher ignoriert, so daß das Warten auf den abspielbereiten Zustand der Audiohardware dort entfällt.

Hier nun die Optionen

() Automatisch warten - benutzt wenn möglich, den CIA-Timer zum Warten, wenn nicht, wird eine Anzahl an **Rasterzeilen**

gewartet

() auf Rasterzeilen warten - es werden je nach Einstellung mehrere

**Bildschirmzeilen** abgewartet, bevor der Abspiel-

vorgang weiterläuft

() mit **DBF-Schleifen** warten - eigentlich wurde diese Funktion nur noch für

mit 68000-CPU's bestückte Amigas mit drin

behalten, bei dieser Einstellung wird einfach

eine Prozessor-Warteschleife durchlaufen, was

auf Prozessoren mit Cache an sich nicht

mehr haltbar sein dürfte

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

Waitingmode Zahl - Zahl: der zu verwendende Modus

0 = automatic

1 = Rasterlines

2 = DBF-Loops

Arexx:

status g wam - gibt den momentanen Zustand als Zahl zurück,

0 = automatic

1 = Rasterlines

2 = DBF-Loops

Tooltype:

Waitingmode=Zahl - Zahl: siehe oben

## 1.139 xfddecrunch

Dateien mit XFD entpacken

XFD ist eine Programmbibliothek von Georg Hörmann, welche in der Lage ist, die meisten auf dem Amiga existierenden Komprimierungsprogramme und ihre Daten erkennen und entpacken zu können. Obwohl der Eagleplayer von sich aus in der Lage ist, die gebräuchlichsten Packer erkennen und verarbeiten zu können, wurde die Möglichkeit eingebaut, über XFD auch die weiteren vorhandenen Packer unterstützen zu können (sehr praktisch z.B. bei Verwendung des Eagleexotic).

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

XFDDecrunch Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g xfd - gibt den momentanen Zustand zurück (yes/no)

Tooltpe:

XFDDecrunch=Zustand - Zustand: siehe oben

## 1.140 xfdloadseg

Executables mit XFD entpacken

Bei eingeschalteter Option wird der Eagleplayer in die Lage versetzt, auch ausführbare Dateien mittels XFD entpacken und nutzen zu können.

Somit ist es beispielsweise möglich, alle Abspieler und Engines auf der Festplatte in komprimierter (z.B. Powergepackter) Form unterzubringen, wenn dort nicht genügend Platz übrig ist, wobei dazu zu sagen ist, daß sich die Ladezeit des Eagleplayers notwendigerweise verlängert.

Achtung! Wenn Sie diese Funktion aktivieren wollen, stellen Sie vorher unbedingt sicher, daß sich die XFD-Slaves `XFD\_FIMP\_LHLib\_XPK.Slave` und `XFD\_ATN!\_LHLib\_XPK.Slave` keinesfalls in Ihrem Libs:xfd/ -Verzeichnis befinden! Diese XFD-Slaves waren Testversionen von uns, die aufgrund eines Mißverständnisses in Umlauf gerieten und bei Verwendung der XFD-Option für ausführbare Dateien erhebliche Probleme mit neueren XFD-Versionen verursachen. (Massenweise Enforcer-Hits und damit verbunden z.B. auch das Problem, das das Laden des Eagleplayers1.54 auf einigen Rechnern

einige Minuten dauern konnte) Daß dieses Problem nur in Verbindung mit dem Eagleplayer auftauchte, lag einzig und allein nur daran, daß der Eagleplayer anscheinend das einzige Programm ist, welches dieses Feature von XFD nutzt.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

XFDLoadseg Zustand - Zustand: [yes,on oder 1]=an, [no,off oder 0]=aus

Arexx:

status g xfl - gibt den momentanen Zustand zurück (yes/no)

Tooltype:

XFDLoadSeg=Zustand - Zustand: siehe oben

## 1.141 xpkpackmethod

Ändere XPK Packer

Bei Aufruf dieser Funktion wird über einen Requester der Name des beim **Speichern** einer mit XPK komprimierten Datei zu verwendenden Algorithmus abgefragt. Der Name hat genau 4 Buchstaben zu umfassen.

Beispiele: SQSH, CRMS, SMPL, GZIP, MASH, NUKE

Als besonders empfehlenswert für Musikdateien mit eingebundenen Samples (z.B. Protracker, OctaMED etc.) haben sich SQSH bzw. CRMS herausgestellt, die mit sehr guten Packraten brillieren, wobei rein subjektiv SQSH schneller und meistens besser agiert.

Tastatur:

-

Menü-Shortcut:

-

Arexx/Shellstart:

XPKPackmethod Name - Name: 4 Buchstabiger Name des XPK-Packers

Arexx:

status g xpk - gibt den momentanen Packernamen zurück,

String mit 4 Buchstaben

Tooltype:

XPKPackmethod=Name - Name: siehe oben