

CeeD

COLLABORATORS

	<i>TITLE :</i> CeeD		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 19, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	CeeD	1
1.1	Documentation for CeeD!	1
1.2	Just which version is this documentation for?	2
1.3	Who wrote this thing?	2
1.4	What do I need to do to get it to go?	2
1.5	What's this stuff in the icons?	3
1.6	What does this button do?	3
1.7	How do I teach it about my music?	4
1.8	How do I pick a track?	4
1.9	What else does it do?	5
1.10	Why doesn't it work?	5
1.11	What could be done to make it better?	6
1.12	Who can have a copy?	6
1.13	So who's fault is it?	7
1.14	What the hell is easyrex anyway?	7
1.15	What! It supports AREXX as well?	7
1.16	arexx_stop	9
1.17	arexx_play	9
1.18	arexx_pause	9
1.19	arexx_resume	9
1.20	arexx_track_back	10
1.21	arexx_track_forth	10
1.22	arexx_index_back	10
1.23	arexx_index_forth	10
1.24	arexx_rewind	11
1.25	arexx_fast_forward	11
1.26	arexx_scan_forward	11
1.27	arexx_scan_backward	11
1.28	arexx_scan_stop	11
1.29	arexx_eject	12

1.30	arexx_get_cur_track	12
1.31	arexx_get_cur_index	12
1.32	arexx_get_cur_time	12
1.33	arexx_get_num_tracks	13
1.34	arexx_get_track_time	13
1.35	arexx_get_artist_name	13
1.36	arexx_get_disc_name	14
1.37	arexx_get_track_name	14
1.38	arexx_quit	14
1.39	arexx_version	14

Chapter 1

CeeD

1.1 Documentation for CeeD!

Documentation for CeeD - an audio CD player for SCSI-2 compliant CD-ROM drives, with enhancements for SONY drives.

Author

Installation

ToolTypes

Gadgets

Keyboard

AREXX

Bugs

Future

Copyright

Credits

This Version

CeeD is Copyright ©1994-1995 Peter Urbanec

If you teach CeeD about new CDs that it doesn't know yet, send your catalogue files to the author: amiga@cse.unsw.edu.au
If that doesn't seem to work, try Stuart MacKinnon: stuartm@zip.com.au

Please use LHA to archive your CD catalogues and make sure that the file comments get included.

1.2 Just which version is this documentation for?

Rather than typing in manually what the version information for this is I get the excellent HWGRCS from Heinz Wrobel to tell you:

```
$Author: peteru $
$Date: 1995/12/22 14:44:10 $
$Revision: 1.270 $
```

This version introduces:

- A track chooser.
- Support for all SCSI-2 compliant drives.
- Info in title bar while CeeD is shrunk.
- A NO-GUI mode.
- Auto detection of SONY drives.
- Skips any initial non-audio tracks.
- More CDs are now recognised
- Title list editor can now be resized.
- AREXX QUIT function that actually quits CeeD :-)

1.3 Who wrote this thing?

The author can be reached via the following means:

E-Mail: amiga@cse.unsw.edu.au

Snail mail: Peter Urbanec
Union Box 12
PO Box 173
Kingsford NSW 2032
AUSTRALIA

Telephone: + 61 2 398 3456 (Home)
+ 61 2 693 0586 (Work)

N.B. The 2 in the phone numbers above will be replaced by 29 in early 1996.

1.4 What do I need to do to get it to go?

You have already performed most of the installation by ↔
decompressing
the LHA archive. CeeD likes to live in it's own directory. It also
likes a subdirectory called "Disks" where it puts the catalogues of
the CDs it knows about. CeeD also needs
(easyrexx.library)
which
is used to implement the
AREXX port

of CeeD.

1.5 What's this stuff in the icons?

Tooltypes supported:

- UNIT - The SCSI unit number of your CD-ROM drive.
Default: 3
- DEVICE - The SCSI name of device to which the CD-ROM drive is attached. Device driver must support SCSI-Direct protocol.
Default: scsi.device
- CATALOG - The directory in which the CD catalogues are stored.
See
title editor
for more info.
Default: PROGDIR:Disks/
- PUBSCREEN - The public screen to open CeeD user interface on. If this tooltype is omitted, or specifies a public screen which is not currently open, CeeD will open on the default public screen. This is usually the Workbench screen.
- WINDOW_POS_X - The initial window X coordinate.
Default: 0
- WINDOW_POS_Y - The initial window Y coordinate.
Default: 16
- NOGUI - Do not display a graphical user interface. The player can still be controlled using keyboard and Arexx.
Default: Open a GUI

1.6 What does this button do?

Main window gadgets:

Black area of main window is Edit List gadget. A left mouse button click will bring up the

title editor.

A right mouse button click will bring up

the

title chooser.

Play - Plays CD from track 1. If paused, resumes play.

Pause - Pauses the CD. If paused, resumes play.

Stop - Stop playing.

- Track Back - Go to beginning of previous track.
- Track Forward - Go to beginning of next track.
- Index Back - Go to previous index.
- Index Forward - Go to next index.
- Rewind - Start scanning backwards. SONY drives will use the built in scanning mode, other brands will use CeeD scanning routines.
- Fast Forward - Start scanning forwards. SONY drives will use the built in scanning mode, other brands will use CeeD scanning routines.
- Eject - Eject CD. If drawer is open and drive supports it, close drawer. SONY drives can figure out whether the door is currently open or closed and always keep in sync. Other brands do not have a reliable way of telling, and so, this control will work as a toggle.

1.7 How do I teach it about my music?

The title editor will allow you to teach CeeD about the compact discs that you feed it. You can enter the Artist, the Name of the CD and also the names of titles. You can choose to Save the catalogue for that particular disc, Use it just this once or Cancel the operation and use the previous definitions for these settings.

The catalogue is saved in a directory specified by the CATALOG tooltype. The filename of the catalogue file is derived by examining the time characteristics of the disc and composing them to give a hopefully unique ID. The catalogue file itself contains The artist, newline, the CD title, newline and then the names individual titles separated by newlines.

CeeD will also put the name of the CD in the file comment, this way you can identify the catalogues easily.

1.8 How do I pick a track?

The track chooser will allow you to select a track to play. You can also use it to just view the details of the entire CD. You can click on a particular track to get CeeD to start playing it, or just close the window if you don't want to make a selection.

1.9 What else does it do?

Keyboard shortcuts:

All keyboard shortcuts are accessible from any operating system compliant application.

NOTE: LAMIGA stands for Left Amiga key, all keys quoted bellow are numeric keypad keys.

- LAMIGA 0-9 - The digit keys are used to numerically enter the number of track to play. You can enter the track number as a single digit number, or as a two digit number. Only the last two digits will be taken into consideration if you enter more than two digits in sequence. Use the next keyboard shortcut to activate play.
- LAMIGA Enter - Play numerically entered track as described above.
- LAMIGA (- Scan backwards.
- LAMIGA) - Scan forward.
- LAMIGA SHIFT (- Skip to previous index.
- LAMIGA SHIFT) - Skip to next index.
- LAMIGA / - Stop playing.
- LAMIGA SHIFT / - Eject CD. If drawer is open and drive supports it, close drawer.
- LAMIGA * - If stopped start playing, if paused resume play, if playing pause.
- LAMIGA - - Skip to previous track.
- LAMIGA + - Skip to next track.

1.10 Why doesn't it work?

CeeD has been developed with SegTracker, MungWall and Enforcer running all the time, so hopefully most serious bugs have been eliminated before release. However, it is possible that there are some hidden "features" which I have not found. If you do run into problems, please report them to

me

, together with any steps that are needed to reproduce these problems. After all, if I don't know about your problems I can't fix them :-)

CeeD only claims to support SCSI-2 compatible drives. It also has

enhancements for SONY CDU-xxx drives. Currently CeeD has been tested and confirmed to work on CDU-55S and CDU-76S drives. No doubt it will work on any drive that supports the SCSI-2 set of features. Certain features such as scanning forwards and backwards use commands that SONY have proposed for the next revision of the SCSI standard. These are only enabled when CeeD sees a SONY drive. Different routines are used for CD-ROM drives from other vendors.

As an aside, CeeD was originally written to comply strictly with the SCSI-2 spec, due to lack of documentation from SONY. When I managed to obtain the relevant documentation the number of features that the SONY drive offered convinced me to implement alternative routines in order to make CeeD easier to use. If you have documentation available for other manufacturers drives, send it me and I will see if I can implement conditional routines to handle your drive.

1.11 What could be done to make it better?

There are quite a few things that CeeD could or should support.

- Better font sensitivity, not just Y-sensitive like now.
- A much better icon.
- Redefinable keyboard shortcuts.
- Saveable settings.
- Context sensitive, online AmigaGuide help.
- Your suggestion here.

These might get implemented or maybe not, it all depends on how much time I have and if there is demand for another release. You can drop me a line or two to any of my addresses.

1.12 Who can have a copy?

CeeD and documentation is freely distributable copyrighted piece of software. This means that you shouldn't have to pay anything for the software, except perhaps for a small duplication fee. It must be distributed with it's documentation - CeeD.guide

If you like CeeD enough to consider sending me a donation you can send money, beer, chocolate, music CDs or at least e-mail. If I don't get any response to CeeD I won't bother releasing subsequent versions and just get it to do what I want it to do and keep it to myself.

NOTE WELL: CeeD might not be included as a part of a commercial distribution without prior written agreement from the author. This includes magazine cover disks, compilation CD-ROMs, as well as other commercial packages.

This is a notice to explicitly allow the inclusion of CeeD on AMINET, FreshFish, GoldFish and FrozenFish CDs.

1.13 So who's fault is it?

Big thanks for beta testing, suggestions, GUI improvements and some code must go to Stuart MacKinnon (stuartm@zip.com.au)

Also, more thanks to Stuart, for adding a lot of little features to this version of CeeD.

1.14 What the hell is easyrexx anyway?

CeeD supports an ARexx interface. This interface uses EasyRexx by Ketil Hunn. easyrexx.library is ©1994,1995 Ketil Hunn.

CeeD only needs version 1 of this library, so if you haven't got the latest version you will still be OK.

1.15 What! It supports AREXX as well?

CeeD has an ARexx interface. This interface uses EasyRexx.

CeeD will

not start without EasyRexx available, but don't worry, the library is included with this distribution.

CeeD supports 24 ARexx commands divided into 2 groups:

Operations:

STOP

Stops play

PLAY

Plays a track

PAUSE

Pauses play

RESUME

Resumes play

TRACK_BACK

Goes to previous track

TRACK_FORTH

Goes to next track

INDEX_BACK
Goes to previous index

INDEX_FORTH
Goes to next index

REWIND
Skips 5 seconds back

FAST_FORWARD
Skips 5 seconds forward

SCAN_FORWARD
Starts scanning forward

SCAN_BACKWARD
Starts scanning backward

SCAN_STOP
Stops scanning

EJECT
Opens/Closes the drawer

QUIT
Quits CeeD

Inquiries:

GET_CUR_TRACK
Returns current track

GET_CUR_INDEX
Returns current index

GET_CUR_TIME
Returns current time into disc

GET_NUM_TRACKS
Returns total number of tracks on CD

GET_TRACK_TIME
Returns absolute starting time of track

GET_ARTIST_NAME
Returns name of the artist

GET_DISC_NAME
Returns name of CD

GET_TRACK_NAME
Returns name of CD

VERSION
Returns version of CeeD

1.16 arexx_stop

Syntax: STOP

Arguments: None

Purpose: Stops the current play operation. No error checking is done.

Results: RC_OK

1.17 arexx_play

Syntax: PLAY TRACK/N

Arguments: TRACK/N - Optional number. Specifies which track to play.
Defaults to 1.

Purpose: Plays a specified audio track. If TRACK is not specified or TRACK is not in a valid range track 1 is played.

Results: RC_OK

1.18 arexx_pause

Syntax: PAUSE

Arguments: None

Purpose: Pauses the current play operation. No error checking is performed.

Results: RC_OK

1.19 arexx_resume

Syntax: RESUME

Arguments: None

Purpose: Resumes a paused play operation. No error checking is done.

Results: RC_OK

1.20 arexx_track_back

Syntax: TRACK_BACK

Arguments: None

Purpose: Skips to previous track. Will only skip if the CD is being played and current track is not the first track on the disc.

Results: RC_OK

1.21 arexx_track_forth

Syntax: TRACK_FORTH

Arguments: None

Purpose: Skips to the next track. Will only skip if the CD is being played and the current track is not the last track on the disc.

Results: RC_OK

1.22 arexx_index_back

Syntax: INDEX_BACK

Arguments: None

Purpose: Skips to previous index. Will only skip if the CD is being played and the current index is not the first index on the track.

Results: RC_OK

1.23 arexx_index_forth

Syntax: INDEX_FORTH

Arguments: None

Purpose: Skips to the next index. Will only skip if the CD is being played and the current index is less than 99. A common side effect for many CD-ROM drives is to skip to the start of the next track if asked to skip to an index which is larger than the highest index on the current track.

Results: RC_OK

1.24 arexx_rewind

Syntax: REWIND

Arguments: None

Purpose: Jumps five seconds back, but only if CD is playing.

Results: RC_OK

1.25 arexx_fast_forward

Syntax: FAST_FORWARD

Arguments: None

Purpose: Jumps five seconds forwards, but only if CD is playing.

Results: RC_OK

1.26 arexx_scan_forward

Syntax: SCAN_FORWARD

Arguments: None

Purpose: Starts scanning forwards, but only if CD is playing.

Results: RC_OK

1.27 arexx_scan_backward

Syntax: SCAN_BACKWARD

Arguments: None

Purpose: Starts scanning backwards, but only if CD is playing.

Results: RC_OK

1.28 arexx_scan_stop

Syntax: SCAN_STOP

Arguments: None

Purpose: Stops the scanning

Results: RC_OK

1.29 arexx_eject

Syntax: EJECT

Arguments: None

Purpose: To open or close the CD-ROM drawer. This command is a toggle, it opens a closed drawer and closes an open drawer. CD-ROM drives which use a caddy will probably only eject discs.

Results: RC_OK

1.30 arexx_get_cur_track

Syntax: GET_CUR_TRACK

Arguments: None

Purpose: To report the current track on the CD. No error checking is done. Calling this function with no CD in drive has undefined result. Use GET_NUM_TRACKS to find out if there is a CD in the drive.

Results: NUMBER (Current Track)

1.31 arexx_get_cur_index

Syntax: GET_CUR_INDEX

Arguments: None

Purpose: To report the current index within a track. No checking is done. Calling this function with no CD in drive has undefined result. Use GET_NUM_TRACKS to find out if there is a CD in the drive.

Results: NUMBER (Current Index)

1.32 arexx_get_cur_time

Syntax: GET_CUR_TIME

Arguments: None

Purpose: To report the current absolute time on disc. No checking is done. Calling this function with no CD in drive has undefined result.

Use GET_NUM_TRACKS to find out if there is a CD in the drive.

The result has the form "MM:SS:FF" where MM is 2 digits for minutes, SS is 2 digits for seconds and FF is 2 digits for frames. Frames range from 00 to 74. All numbers have leading zeros.

Results: STRING (Current Absolute Time)

1.33 arexx_get_num_tracks

Syntax: GET_NUM_TRACKS

Arguments: None

Purpose: To determine if a CD is present and if so how many tracks are on it. If no disk is inserted the result will be 0, otherwise the number of tracks on the disc will be returned.

Results: NUMBER (Number of tracks on disc or 0)

1.34 arexx_get_track_time

Syntax: GET_TRACK_TIME TRACK/N

Arguments: TRACK/N - Optional track number.
Defaults to 1.

Purpose: To determine the absolute starting time of a particular track. The TRACK number must have a valid value, otherwise 1 will be used. Calling this function with no CD in drive has undefined result. Use GET_NUM_TRACKS to find out if there is a CD in the drive.

The result has the form "MM:SS:FF" where MM is 2 digits for minutes, SS is 2 digits for seconds and FF is 2 digits for frames. Frames range from 00 to 74. All numbers have leading zeros. To determine the total duration of the CD you can read the LEAD OUT time which has a TRACK number equal to GET_NUM_TRACKS+1.

Results: STRING (Track Start Absolute Time)

1.35 arexx_get_artist_name

Syntax: GET_ARTIST_NAME

Arguments: None

Purpose: To determine if a CD is in drive, and if so, who the author is. This function will return "NO DISC" if there is no CD in drive; the name of the author of the disc otherwise.

Results: STRING (Artist)

1.36 arexx_get_disc_name

Syntax: GET_DISC_NAME

Arguments: None

Purpose: To determine if a CD is in drive and if so, what it's title is. This function will return "NO DISC" if there is no CD in drive or the name of the disc.

Results: STRING (CD Name)

1.37 arexx_get_track_name

Syntax: GET_TRACK_NAME TRACK/N

Arguments: TRACK/N - Optional track number.
Defaults to 1.

Purpose: To determine the name of a CD track. The TRACK number must have a valid value, otherwise track 1 will be used. Calling this function with no CD in drive will return "NO DISC". The special value GET_NUM_TRACKS+1 will result in title "LEAD OUT".

Results: STRING (Track Name)

1.38 arexx_quit

Syntax: QUIT

Arguments: None

Purpose: To quit CeeD. No more commands should be sent to CeeD after sending a QUIT command. The program will clean up and terminate.

Results: RC_OK

1.39 arexx_version

Syntax: VERSION FULL/S

Arguments: FULL/S - A switch to choose full version string.

Purpose: To determine the version number of CeeD. Without the FULL switch CeeD will return the version string in "VERSION.REVISION" form.

eg. "1.2"

When the FULL switch is included the complete version string will be returned. eg. "CeeD 1.2 (23.3.95) ©1994-1995 Peter Urbanec"

Results: STRING (Version String)
