

**ddc**

**COLLABORATORS**

	<i>TITLE :</i> ddc		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 26, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>ddc</b>	<b>1</b>
1.1	ddc.guide	1
1.2	ddc.guide/About	2
1.3	ddc.guide/Copyright	2
1.4	ddc.guide/Thanks	3
1.5	ddc.guide/History	3
1.6	ddc.guide/Background	5
1.7	ddc.guide/Warning	5
1.8	ddc.guide/Requirements	5
1.9	ddc.guide/Usage	5
1.10	ddc.guide/FileFormat	6
1.11	ddc.guide/Future	6
1.12	ddc.guide/Contact	7
1.13	ddc.guide/Problems	7
1.14	ddc.guide/Attention	7

---

# Chapter 1

## ddc

### 1.1 ddc.guide

Driver Descriptor Creator Guide

Dies ist die AmigaGuide® Anleitung für Driver Descriptor Creator (ddc).

Inhalt:

Einführung

Wozu das ganze?.

Voraussetzungen

Was man für dieses Programm braucht.

Warnung

Haftungsausschluß.

Urheberrecht

Rechtliches.

Benutzung

Wie man dieses Programm benutzt.

Hintergrund

Wie dieses Programm entstanden ist.

Dateiformat

Wie sind diese "Driver Descriptors" aufgebaut?

Probleme

Probleme mit bestimmten Erweiterungen.

Geschichte

Was sich geändert hat.

Zukunft

Ein Blick in die Kristallkugel.

---

Kontakt  
Wie man den Autor erreicht.

Danksagungen  
Hall Of Fame.

## 1.2 ddc.guide/About

Über dieses Programm

ddc erzeugt sogenannte "Driver Descriptors". Das sind Dateien, die beschreiben welcher Treiber zu welcher Hardware gehört. Hauptziel dieses Programms ist es, eine Möglichst große Anzahl solcher "Driver Descriptors" zu sammeln, um sie in zusammen mit dem HardwareAssistent zu verwenden. Dem User, der die "Driver Descriptors" mit ddc erzeugt, bringt das natürlich zunächst wenig. Deshalb schickt mir die "Driver Descriptors" per email (walternn@studi.informatik.uni-stuttgart.de), damit ich sie sammeln und später veröffentlichen kann.  
Lesen Sie dazu auch

Voraussetzungen

.

## 1.3 ddc.guide/Copyright

Copyright:

Driver Descriptor Creator wurde geschrieben von  
Norman Walter

.

Dieses Programm darf frei verteilt werden.  
Bedingung ist, daß an dem Programm, der Dokumentation oder sonstigen Komponenten nichts verändert wurde und das Archiv vollständig bleibt.

Kommerzielle Nutzung dieses Programms ist nur mit vorheriger Genemigung des

Autors  
erlaubt.

Bitte lesen Sie auch unbedingt den  
Haftungsausschluß  
!

Dieses Programm genutzt folgende 3rd-Party Software:

- identify.library von Richard Körber (erhältlich in Aminet)

Zusätzliche Icons wurden erstellt von:

- Carsten Berger
- Thore Sittly

## 1.4 ddc.guide/Thanks

Danksagungen:

Ein großes Dankeschön geht an alle, die den "Driver Descriptor Creator" getestet und mir ihre "Driver Descriptors" und Bugreports zugeschickt haben.

Besonders hervorheben möchte ich hier Frank Cieslewicz, der den ddc durch sein komplettes "Amiga-Museum" gejagt hat.

Ein großes Lob geht an Carsten Berger und Thore Sittly, die Icons beigesteuert ↵ haben.

Hut ab für Richard Körber, dem Erfinder der identify.library. Ohne diese tolle Sache wäre ein solches Projekt nur mit wesentlich höherem Aufwand machbar.

Danke an Gerald Sittly für Hilfe beim Debuggen und den Hinweis, daß bei manchen Karten ein Schrägstrich im Produktnamen enthalten ist enthalten.

## 1.5 ddc.guide/History

Geschichte:

- 7.7.2003: ddc v0.1 - Erste interne Testversion
- 8.7.2003: ddc v0.2 - Erste öffentliche Testversion
  - Bug behoben: Lange Pfade werden jetzt richtig übernommen.
- 12.7.2003: ddc v0.3
  - Erweiterung der GUI für mehr Interaktion.
  - Sicherheitsabfrage beim Überschreiben bereits vorhandener Dateien
  - Icons
  - Erweiterung der Dokumentation
  - Skript für automatischen Download der identify.library
- 17.7.2003: ddc v0.4
  - Ernster Speicher-Bug behoben.
    - Für Insider: Ein Zeiger wurde benutzt, nachdem er bereits freigegeben wurde. Peinlich!
    - Das erklärt all die nicht reproduzierbaren Abstürze.
  - Workaround: Bei machen Karten enthält der Produktnamen einen Schrägstrich. Das ist natürlich Gift, wenn man daraus einen Dateinamen erzeugen will.

Der Schrägstrich wird nun durch ein anderes Zeichen ↔ ersetzt.

Danke an Gerald Sittly, der dieses Problem entdeckt und einen Vorschlag zum Workaround geliefert hat.

- Neu Sicherheitsabfrage, falls der User die Auswahl des Installerskripts abbricht.
- Die Probleme mit Pfaden sollten jetzt behoben sein.
- Die Anleitung wurde durch diesen Guide ersetzt.
- Neue Icons von Carsten Berger und Thore Sittly
- Viele neue Driver Descriptor Beispiele

19.7.2003: ddc v0.5

- Bug (hoffentlich) beseitigt: Beim Start von der Workbench kam es zu Gurus.
- Library-Handling verbessert
- Startskript zum starten von ddc über IconX mit erweiterten Debugging-Informationen.
- Neuer, effizienter Patternmatcher
- Neuer Requester: Warnt den User, falls der Name einer Karte einen Schrägstrich enthält.

26.7.2003: ddc v0.6

- Erkennt nun, von welchem Tool das Installationsprogramm aufgerufen werden will. Das war nötig, weil einige Install-Scripte Shell-Scripte sind.
- Setzt nun die Pfade im erzeugten Script in Anführungszeichen. Damit werden Konflikte bei Pfaden vermieden, die Leerzeichen enthalten.

18.12.2003: ddc v0.7

- Der Pfad, wo die DriverDescriptors gespeichert werden sollen, kann nun in den ToolTypes des driver\_descriptor\_creator angegeben werden.
- Warnt jetzt den User, falls er unsinnige Pfade wie z.B. "Ram Disk:" bei der Wahl des Installer-Scripts angibt.
- Das Start Script wurde wieder entfernt - es wird nicht mehr benötigt.
- Die DriverDescriptors befinden sich nun nicht mehr im Archiv, weil diese im Archiv des HardwareAssistent enthalten sind.

27.12.2003: ddc v0.71

- Bugfix: Anstatt C:Installer wurde C:C:Installer im Skript eingetragen.
- Bei Auswahl eines Installerskripts von Rad: wird der User jetzt auch gewarnt.

09.01.2004: ddc v0.71 - Bug entfernt, der zum Absturz führte.

Weitere Informationen zur Entstehung dieses Programms gibt es [hier](#)

.

## 1.6 ddc.guide/Background

Hintergrund:

Im amiga-news Forum (<http://www.amiga-news.de>) haben immer wieder Leute bemängelt, daß es auf dem Amiga so etwas wie MS Window's "Plug'N'Play" nicht gibt. Dieses Projekt ist ein erster Schritt diese Lücke zu füllen. Da ein Plug'N'Play-System (kurz PnP) wissen muß, welcher Treiber zu welcher Hardware gehört, muß eine entsprechende Datenbank angelegt werden. Genau dies besorgt ddc in Form der sogenannten "Driver Descriptors".

## 1.7 ddc.guide/Warning

Warnung:

Benutzung auf eigene Gefahr. Der

Autor

haftet nicht für Schäden

an Hard- und Software, die eventuell durch Benutzung dieser Software entstehen könnten.

## 1.8 ddc.guide/Requirements

Voraussetzungen

- Ein ECHTER Amiga. Draco, CD32 und CDTV müßten auch gehen (ungetestet). Auf einem Emulator (z.B. UAE) macht automatische Hardwareerkennung natürlich keinen Sinn. Da ddc auf der identify.library aufsetzt, funktioniert es natürlich nur auf Systemen, auf denen diese Library verfügbar ist.
- Es wird mindestens AmigaOS 2.0 vorausgesetzt. Empfohlen wird AmigaOS 3.5 oder besser.
- identify.library (gibt's im Aminet)

## 1.9 ddc.guide/Usage

Benutzung

Starte das Programm dcc per Doppelclick oder aus der Shell.

Nach wegklicken des About-Requesters prüft das Programm, welche Zorro Karten im System installiert sind.

Die erkannten Karten werden gelistet.

Für jede erkannte Karte wird der User zum Einlegen der (in der Regel ↔ mitgelieferten) Treiberdiskette aufgefordert.

Dannach wählt man mittels ASL-File-Requester das Installer-Skript auf der ↔ Treiberdiskette oder CD aus.

Falls die genannte Hardware keinen Treiber benötigt wie z.B. reinen bei Ram- ↔ Erweiterungen

drückt man im File-Requester einfach "Abbrechen". ddc fragt dann, ob man trotzdem ↵  
einen  
Driver Descriptor für die Hardware erstellen möchte. Wenn man auch "Yes" klickt ↵  
wird  
ein Driver Descriptor erzeugt, bei dem der Pfadname des Installerskripts auf einen ↵  
Dummy-Wert  
gesetzt ist.

Bitte schicke die erstellten Dateien dann per email an walternn@studi.informatik. ↵  
uni-stuttgart.de.  
Am besten packst Du zuvor die Driver Descriptors (Wichtig: samt .info-Dateien !)  
in ein lha File.

Was man bei der Benutzung beachten muß, steht  
hier

.

## 1.10 ddc.guide/FileFormat

Wie sind diese "Driver Descriptors" aufgebaut?

ddc erzeugt vollautomatisch ein Shell-Skript, welches das entsprechende  
Installer-Skript für die jeweilige Hardware startet. Damit fungieren  
die Driver Descriptors auch (per IconX) als Stand-Alone Programme.  
Weitere Informationen liegen als ToolTypes im Disk Object des Projekticons vor.  
Diese können auch nachträglich von Hand editiert werden.  
Beispielsweise kann das Feld "DRIVERURL=" ausgefüllt werden.  
Zukünftige Programme können diese Informationen auslesen und den Treiber  
beispielsweise direkt aus dem Internet laden.  
Die hohe Transparenz dieses Systems ermöglicht es auf einfachste Weise,  
solche Dateien für neue Hardware und neue Treiber zu erstellen.  
Ebenso einfach ist die Entwicklung von Tools, die auf dieses System aufsetzen.

## 1.11 ddc.guide/Future

Zukunft:

Im Moment werden nur Zorro Karten unterstützt. Dies kann ausgeweitet werden.  
Geplant ist außerdem ein Tool Namens "Hardware Assistent", daß in die  
WBStartup kommt und für neue Hardware automatisch Treiber installiert.  
Dies wird allerdings nur umgesetzt, wenn auch genügend Nachfrage besteht.  
Es steht jedem frei, Tools zu schreiben, die das "Driver Descriptors" System ↵  
benutzen.

Wie das Dateiformat der "Driver Descriptors" aufgebaut ist, wird in

Dateiformat  
beschrieben.

## 1.12 ddc.guide/Contact

Kontakt:

Autor: Norman Walter  
email: walternn@studi.informatik.uni-stuttgart.de  
www: <http://www.norman-interactive.com>

## 1.13 ddc.guide/Problems

Probleme:

Bestimmte Hardwareerweiterungen werden nicht oder falsch erkannt.  
Dies ist ein Problem der identify.library.

Es kann Probleme geben mit "Huckepack-Erweiterungen", die als Module auf andere Karten gesteckt werden. Berichte haben gezeigt, daß bei machen Turbokarten der CyberStorm Reihe das SCSI-Modul nicht erkannt wird. Ferner sind wahrscheinlich auch diverse Zorro-Karten mit Clockport-Erweiterungen problematisch.

Einige Karten werden falsch erkannt:

- Oktagon 2008 SCSI: Wird im Amiga 4000 als "Oktagon 4008" erkannt.
- CyberstormPPC/060: Wird als "CyberStorm\_'066\_MK-III" erkannt.

Bei einigen Karten wird nur der Hersteller, nicht aber das Produkt erkannt.

Beispiel: VarIO von Individual Computers

Dies beruht auf Unvollständigkeit der identify.library

## 1.14 ddc.guide/Attention

Bitte beachten:

Es ist unbedingt darauf zu achten, daß das Installationsskript von der Treiberdiskette/CD Rom ausgewählt wird.  
Pfade nach Ram:, Rad: oder Work: usw. sind sinnlos, da der im Driver Descriptor gespeicherte Pfad dann nicht mit dem Pfad auf dem System eines Benutzers, welcher später diesen Driver Descriptor verwenden möchte übereinstimmt.

Bitte NICHT die Dateinamen der "Driver Descriptors" ändern!

Wie man das Programm benutzt steht  
hier

.