

# **Graph2D**

Kai Nickel

Copyright © CopyrightÂ©1994-95 Kai Nickel

**COLLABORATORS**

	<i>TITLE :</i> Graph2D		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Kai Nickel	July 19, 2024	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Graph2D</b>	<b>1</b>
1.1	Inhaltsverzeichnis Graph2D	1
1.2	Kurzbeschreibung	2
1.3	Autor	2
1.4	Copyright	3
1.5	Registrierung von Graph2D	4
1.6	Installation	5
1.7	Fragen & Antworten	7
1.8	Bedienung	7
1.9	Projekt-Menü	8
1.10	Funktionseditor	8
1.11	Notation	11
1.12	Diskussion	11
1.13	num. Integration	13
1.14	Wertetabelle	13
1.15	Tangente anlegen	13
1.16	Taylorreihe	14
1.17	Graph-Fenster	14
1.18	2D-Graph-Fenster	15
1.19	2D-Graph-Einsteller	17
1.20	3D-Graph-Fenster	19
1.21	3D-Graph-Einsteller	20
1.22	SIRDS-Graph-Fenster	21
1.23	SIRDS-Graph-Einsteller	23
1.24	Textanzeigefenster	24
1.25	Voreinstellungseeditor	24
1.26	Drucken	25
1.27	ARexx	25
1.28	Funktion	26
1.29	Interne Funktionen und Operatoren	28

---

---

1.30 Einbindung . . . . .	28
1.31 EBNF . . . . .	29
1.32 Benutzeroberfläche Mui . . . . .	30
1.33 MATHSCRIPT . . . . .	31
1.34 SIRDS . . . . .	31
1.35 Index . . . . .	32

---

# Chapter 1

## Graph2D

### 1.1 Inhaltsverzeichnis Graph2D

G r a p h 2 D

Funktionsplotter mit Analysefunktionen  
Version 3.10 (26.09.1995)  
Autor: Kai Nickel

#### 1. Allgemeines

1.1	Kurzbeschreibung	-	Was ist Graph2D?
1.2	Autor	-	Wer bin ich? Wo gibts Updates?
1.3	Copyright	-	Wen interessiert's?
1.4	Registrierung	-	Um Graph2D dauerhaft zu benutzen
1.5	Installation	-	So kommt Graph2D auf die Platte
1.6	Fragen und Antworten	-	Beantwortung häufiger Fragen und Probleme

#### 2. Bedienung

2.1	Allgemeines	-	Allgemeines zur Bedienung
2.2	Funktionseditor	-	Funktionseditor mit math. Optionen
2.3	Graphen	-	Die Graph-Fenster
2.3.1	2D-Graph	-	2D-Graph-Fenster
2.3.3	3D-Graph	-	3D-Graph-Fenster
2.3.5	SIRDS-Graph	-	SIRDS-Graph-Fenster
2.4	Textanzeiger	-	Die Textanzeigefenster
2.5	Voreinstellungen	-	Der Voreinstellungseditor
2.6	Drucken	-	Der Druck-Requester
2.7	ARexx	-	Der ARexx Port

#### 3. Anhang

3.1	Funktionen	-	Was ist eine Funktion?
3.2	Geschichte	-	Änderungen zu älteren Versionen
3.3	MUI	-	Das Benutzeroberflächen-System
3.4	MathScript	-	Der Formel-Editor MathScript
3.5	SIRDS	-	Die verwendete Magic3D-Technik
3.6	Index	-	Kapitel von A bis Z

## 1.2 Kurzbeschreibung

### 1.1 Kurzbeschreibung

Graph2D ist ein Funktionsplotter, der in beliebig vielen Arbeitsfenstern Graphen von mathematischen Funktionen darstellen kann. Dies können "normale" zweidimensionale, dreidimensionale oder auch SIRDS (Magic3D) Graphen sein, die auch ausgedruckt oder als IFF-Bild abgespeichert werden können.

Es können Kurvendiskussionen durchgeführt werden, zu denen die Bestimmung von Nullstellen, Extrema, Wendepunkten, Monotoniebereichen und der Symmetrie gehört. Es ist möglich, Funktionen abzuleiten, zu vereinfachen, Wertetabellen auszugeben, numerische Integrale zu berechnen, Tangenten anzulegen und Taylor-Reihen zu errechnen.

Graph2D hat eine systemkonforme Oberfläche, einen ARexx-Port, enthält ein Installations-Skript, eine Online-Hilfe und liegt lokalisiert in Deutsch und Englisch vor.

Die Features in Stichpunkten:

- Ausgabe von Funktionen in stark konfigurierbaren zweidimensionalen und dreidimensionalen Koordinatensystemen
- Erzeugung von SIRDS-Graphen für dreidimensionale Funktionen
- Arbeiten mit vielen Funktionen in mehreren Fenstern gleichzeitig
- Erstellung von Kurvendiskussionen (Nullstellen, Extrema, Wendepunkte, Monotonie, Symmetrie)
- symbolisches Ableiten und numerisches Integrieren von Funktionen
- Taylor-Reihen Entwicklung
- Erzeugung von Wertetabellen und Anlegen von Tangenten
- gegenseitige Einbindung von Funktionen problemlos möglich
- Darstellung von Funktionen auch in "mathematischer Notation"
- Ausdruck von Graphen, Funktionen und Diskussionen in wählbarer Größe und Auflösung
- Graphen und Funktionen als IFF-Graphiken beliebiger Größe speicherbar
- systemkonforme und komfortable MUI-Oberfläche
- Online-Hilfe im AmigaGuide-Format
- Installation mit Commodore-Installer
- Lokalisierung Deutsch und Englisch
- ARexx Unterstützung
- Shareware mit vollem Funktionsumfang

Systemanforderungen:

- Amiga mit Kickstart2.0 oder höher
- installiertes MUI-System in der Version 2.3 oder höher

## 1.3 Autor

### 1.2 Autor

Graph2D und diese Anleitung wurden geschrieben von

---

Kai Nickel

Post : Herzogstraße 29  
67435 Neustadt  
Germany

eMail : un7x@rz.uni-karlsruhe.de  
kai@rpsbbs.rlp.org

Bankverbindung: Kontonummer 247 014  
BLZ 546 600 24 (Raiffeisenbank Neustadt/Wstr. Nord)

Über Fehlerberichte oder Verbesserungsvorschläge wäre ich sehr dankbar, ebenso über Menschen, die Graph2D oder die Anleitung in eine andere Sprache übersetzen möchten. Bitte setzt Euch aber vorher mit mir in Verbindung!

Bezugsquellen:

Updates von Graph2D werden von mir ins FRAS/Anwendungen und ins Aminet/misc/math gepostet und wenn möglich auch auf der SaarAG-Serie veröffentlicht. Außerdem gibt es sie insbesondere in der folgenden Mailbox, in der auch immer eine ausreichend aktuelle Version von MUI zu finden sein sollte:

Graph2D-Supportbox: RPSBBS  
(24h online) Modem: 06329-1624 (HST/V32b V42b)  
ISDN: 06329-990015 (X75)  
Login: graph2d  
Brett: /Lokal/Support/Amiga/Graph2D  
(ich selbst heiße dort "Kai")

## 1.4 Copyright

### 1.3 Copyright

Graph2D ist Copyright © 1994-95 Kai Nickel.

Graph2D ist Shareware. Der Autor behält sich das Copyright für das Programm vor. Wer Graph2D einigermaßen ernsthaft bzw. über längere Zeit hinweg benutzen möchte, muß sich registrieren lassen.

Die Weitergabe der unregistrierten Version bleibt erlaubt, soweit damit keine kommerziellen Interessen verbunden sind. Konkret darf der Preis einer Diskette, die Graph2D enthält, 5.- DM nicht übersteigen. Eine Aufnahme in PD-Serien ist hiermit unter diesen Bedingungen explizit erlaubt, ebenso die Verbreitung mittels DFÜ. Bei jeder Art der Weitergabe müssen alle zum Originalarchiv von Graph2D gehörenden Dateien unverändert und gemeinsam übernommen werden.

Der Autor gewährt keine Garantie für die Richtigkeit und Funktionsfähigkeit von Graph2D und übernimmt keinerlei Haftung für irgendwelche negativen

Folgen, die durch Graph2D entstehen könnten. Spätere Updates und Fehlerbeseitigungen können natürlich nicht garantiert werden.

Graph2D wurde mit dem Amiga-Oberon-Compiler V3.20 der A+L AG erstellt. Die Lizenzbedingungen des Compilers verbieten eine Nutzung des Programmes im militärischen Bereich.

Graph2D benutzt für seine Oberfläche das MUI-System von Stefan Stuntz.

MagicWB und somit auch einige der mitgelieferten Icons von Graph2D stehen unter Copyright von Martin Huttenloher.

Der Algorithmus zur 3D-Darstellung von Bildern stammt von Kilian Singer und wurde mit seiner Genehmigung in Graph2D angewandt.

## 1.5 Registrierung von Graph2D

### 1.4 Registrierung

Graph2D ist Shareware. Wer das Programm einigermaßen ernsthaft nutzt, ist aufgefordert, sich beim Autor registrieren zu lassen.

Die Registrierungsgebühr beträgt 20,- DM bzw. 25,- DM

Wer Graph2D über eine gewisse Testphase hinaus benutzt, ohne sich registrieren zu lassen, verstößt gegen die Nutzungsbedingungen. Die unregistrierte Version ist zwar uneingeschränkt funktionsfähig, jedoch erscheinen von Zeit zu Zeit "Nerv-Requester", die an die Registrierung erinnern sollen.

Durch die Registrierung bekommt man ein persönliches "Keyfile", daß einen zur dauerhaften Benutzung von Graph2D und eventueller zukünftiger Updates berechtigt. Ich kann aber natürlich nicht zusichern, ob und wie viele Updates es noch geben wird - in der Vergangenheit waren es aber schon etliche...

In aller Regel werden die Registrierungen übrigens gleich am Tag des Eingangs bearbeitet und abgeschickt.

Nach der Installation des Keyfiles verschwinden natürlich die Info-Fenster und Nerv-Requester!

Bedenkt bitte, daß in Graph2D viele Monate Arbeit stecken und daß gerade auch Rückmeldungen der User in Form von Registrierungen über mögliche Weiterentwicklungen und Fehlerbeseitigungen von Graph2D entscheiden.

Und so geht die Registrierung:

Ihr überweist die DM 20,- auf mein Konto und schreibt mir dann eine eMail oder einen Brief, daß Ihr Euch für Graph2D registrieren lassen möchtet. Dazu benutzt Ihr am besten das mitgelieferte Formular Graph2D.RegForm. Wer einen Brief abschickt, kann diesem natürlich auch das Geld in bar oder Euroscheck beilegen. Wie auch immer werde ich Euch dann das Keyfile per eMail (falls angegeben) senden. Wer keine eMail-Adresse hat, dem werde ich das Keyfile auf einer Disk mit der Post zuschicken. In diesem Fall

beträgt die Registrierungsgebühr allerdings DM 25,- da Diskette und Porto für mich leider nicht so billig wie eine eMail sind.

Probleme?

Da bekanntermaßen durchaus einmal eine eMail verloren gehen kann, schreibt mir ruhig noch einmal, wenn meine Antwort länger als z.B. zwei Wochen aussteht. Ich werde dann natürlich noch einmal das Keyfile absenden. Sollte es so gar nicht funktionieren, dann schickt mir einfach einen Brief mit der Post und ich werde dann genauso antworten. Um Pannen zu vermeiden ist es ganz wichtig, daß Ihr im "Verwendungszweck" des Überweisungsdruckes Euere e-Mail und am besten auch reale Adresse angebt - dann kann eigentlich nichts mehr passieren!

Nach der Registrierung:

Das empfangene Keyfile sollte am besten in das Verzeichnis kopiert werden, in dem sich das Hauptprogramm "Graph2D" befindet. Falls es dort aus unerfindlichen Gründen nicht erkannt werden sollte - oder gar bei jedem Start eine Meldung "Please insert Volume PROGDIR:" erscheint - dann kopiert man das Keyfile in das 'S:'-Verzeichnis des Systems. Dort wird es sicher erkannt. Vielen Dank für die Unterstützung des Shareware-Gedankens!

An andere Shareware-Autoren:

Falls Ihr ein Programm geschrieben habt, von dem Ihr denkt, daß ich damit etwas anfangen kann: Schreibt mich doch einfach mal zwecks Keyfile-Tausch an!

## 1.6 Installation

### 1.5 Installation

Automatisch:

Man installiert Graph2D am besten, indem man das mitgelieferten Skript "Graph2D-Install" anklickt. Alles weitere läuft dann im Dialog mehr oder weniger von selbst ab. Graph2D wird danach einfach durch Anklicken von der Workbench aus gestartet.

Für die Ängstlichen: In der Einstellung "Expert User" darf man wirklich jede Aktion selbst bestätigen!

Manuell:

Wer den Installer von Commodore nicht hat, kann Graph2D auch folgendermaßen von Hand installieren (genau das macht auch das Skript):

- Kopieren von Hauptprogramm, Guide und Registrierungsformular in ein selbstgewähltes Verzeichnis. Die Guide muß sich im selben Verzeichnis wie Graph2D befinden, da sie sonst nicht über die "HELP"-Taste geöffnet

werden kann.

- Kopieren der "GarbageCollector.library" in das "LIBS:" Verzeichnis des Systems oder nach PROGDIR:Libs/. Diese Library wird zum Ablauf unbedingt benötigt!
- optional: Unterverzeichnis "Functions" ins Graph2D-Verzeichnis kopieren
- optional: Unterverzeichnis "Graphs" ins Graph2D-Verzeichnis kopieren
- optional: Unterverzeichnis "Rexx" ins Graph2D-Verzeichnis kopieren
- optional: Kopieren des engl. Katalogs nach Locale:Catalogs/english/ oder PROGDIR:Catalogs/english/
- optional: Ersetzen der Icons durch die aus dem Verzeichnis "MWBIcons"

#### Originalarchiv:

Zum original Graph2D-Paket gehören folgende Dateien:

Graph2D	- Das Hauptprogramm
Graph2D-Install	- Das Installationskript
Docs/Deutsch/Graph2D.guide	- Dokumentation/Onlinehilfe
Docs/Deutsch/Graph2DHistory.guide	- Entwicklungsgeschichte
Docs/Deutsch/Graph2D.RegForm	- Registrierungsformular
Docs/Deutsch/Graph2D.LiesMich	- Kurzbeschreibung
Docs/English/Graph2D.guide	- engl. Dokumentation/Onlinehilfe
Docs/English/Graph2DHistory.guide	- engl. Entwicklungsgeschichte
Docs/English/Graph2D.RegForm	- engl. Registrierungsformular
Docs/English/Graph2D.ReadMe	- engl. Kurzbeschreibung
Docs/English/Product-Info	- Beschreibung im "Fish-Format"
Catalogs/english/Graph2D.catalog	- engl. Katalog
Libs/garbagecollector.library	- benötigte Library
Libs/GarbageCollector.LiesMich	- Info zur Library
Graphs/Standard.2D	- 2D-Standardkoordinatensystem
Graphs/Standard.3D	- 3D-Standardkoordinatensystem
Graphs/Standard.SIRDS	- SIRDS-Standardkoordinatensystem
Graphs/Trigonometrie.2D	- 2D-Systembeispiel für Trigonometrie
Functions/Bsp2D.fkt	- Beispielfunktionen
Functions/Bsp3D.fkt	- Beispielfunktionen für 3D-Ansicht
Rexx/CloseGraph2D	- ARexx Skript
Rexx/ExportEPS	- ARexx Skript
Rexx/GetInfos	- ARexx Skript
Rexx/SendToMathScript	- ARexx Skript
MWBIcons/ClickForColors	- MagicWB-Palette (zum Ausprobieren!)
MWBIcons/Graph2D.info	- Icon im MagicWB-Stil
MWBIcons/Graph2D.guide.info	- Icon im MagicWB-Stil
MWBIcons/Graph2DHistory.guide.info	- Icon im MagicWB-Stil
MWBIcons/Graph2D.RegFrom.info	- Icon im MagicWB-Stil
MWBIcons/Drawer.info	- Icon im MagicWB-Stil
MWBIcons/Graphs.info	- Icon im MagicWB-Stil
MWBIcons/Functions.info	- Icon im MagicWB-Stil
MWBIcons/Rexx.info	- Icon im MagicWB-Stil
MWBIcons/Catalogs.info	- Icon im MagicWB-Stil

MWBIcons/Libs.info

- Icon im MagicWB-Stil

## 1.7 Fragen & Antworten

### 1.6 Fragen und Antworten

Wie stelle ich jetzt eigentlich eine Funktion als Graph dar?

Zuerst erzeugt man eine neue Funktion mittels "Erzeugen", gibt ihren Text ein (z.B. " $f(x)=\sin(x)^2$ ") und öffnet einen neuen Graphen mittels "Neuer Graph". Um die neue Funktion im neuen Graphen erscheinen zu lassen, markiert man sie als "Gezeichnet" und bringt den Graphen mittels "Aktualisieren" dazu, sich neu aufzubauen. Die Funktion sollte nun zu erkennen sein, falls ihre Werte in den im Graphen sichtbaren Bereich fallen. Gegebenenfalls muß man den Graph noch "Einstellen" um ein angemessenes Darstellungsintervall zu wählen. Die genaue Art und Weise (z.B. die Farbe), mit der die Funktion im Graphen erscheint, muß auch im Graph-Einsteller eingestellt werden.

Warum sind nach dem Laden einer Funktionliste manchmal korrekte Funktionen als "falsch" dargestellt?

Nach dem Laden werden alle Funktionen in einer bestimmten Reihenfolge interpretiert. Bindet nun eine Funktion eine andere ein, die erst später interpretiert werden wird, so ist sie zum Interpretationszeitpunkt noch ein unbekannter Bezeichner, der zur Fehleranzeige führt. Abhilfe: Einfach die Funktion anklicken und mit <Return> noch einmal interpretieren. Da die eingebundene Funktion ja inzwischen bekannt sein sollte, dürfte es jetzt zu keinem Fehler mehr führen.

Wenn ich ein ARExx-Skript ausführen will, passiert gar nichts!

Damit ARExx-Skripts überhaupt laufen können, muß das Programm "rexxmast" laufen. Wie man dies am besten erledigt, kann man in der ARExx-Dokumentation der Amiga-Handbücher nachlesen. Die meisten beigelegten Skripts funktionieren nur in Verbindung mit dem Programm MathScript. Wer dieses Programm nicht hat, kann auch mit diesen Skripts nichts anfangen.

Graph2D kann dreidimensionale und SIRDS-Graphen darstellen - warum heißt es dann nicht schon längst Graph3D???

Ich weiß es auch nicht so genau. Wenn man sich einmal an einen Namen gewöhnt hat... ;-)

## 1.8 Bedienung

---

## 2.1 Bedienung

Nach dem Start von Graph2D öffnet sich zunächst das Hauptfenster, der sog. Funktionseditor, in dem die Funktionen, mit denen gearbeitet werden soll, eingegeben und bearbeitet werden können. Sind Funktionen vorhanden, können sie in Graph-Fenstern gezeichnet oder mathematisch behandelt werden.

Das Projekt-Menü ist in den meisten Fenstern von Graph2D verfügbar.

Graph2D bietet diese Datei hier auch als kontextsensitive Onlinehilfe an. Durch Drücken der Help-Taste auf der Tastatur kann sie jederzeit geöffnet werden. Dabei wird automatisch in das zum aktiven Fenster passende Kapitel gesprungen.

Da Graph2D eine Anwendung ist, die es erlaubt, mit mehreren Arbeitsfenstern gleichzeitig zu arbeiten, ist es oft sinnvoll, Graph2D mittels der MUI-Prefs einen eigenen Screen zuzuteilen.

Wenn das letzte Fenster geschlossen wird, hat man die Möglichkeit, Graph2D zu beenden oder zu ikonifizieren (ein Piktogramm auf der Workbench, erzeugen, mit dem Graph2D jederzeit wieder zum Leben erweckt werden kann).

## 1.9 Projekt-Menü

### 2.1.1 Das Projekt-Menü

Das Projekt-Menü steht im Funktionseditor, den Graph-Fenstern und dem Notationsfenster zur Verfügung. Es bietet folgende Funktionen:

Projekt

- Funktionseditor... - Öffnet den Funktionseditor, falls er nicht ohnehin schon geöffnet ist.
- Neuer Graph... - Öffnet ein neues Graph-Fenster. Der Typ (2D, 3D oder SIRDS) muß nun noch in einem kleinen Requester ausgewählt werden.
- Voreinstellungen... - Öffnet den Voreinstellungseditor
- Hilfe... - Zeigt die Online-Hilfe an, identisch mit der Funktion der HELP-Taste
- Über... - Öffnet ein Info-Fenster
- Ikonifizieren - Schließt alle Fenster und erzeugt ein Icon auf der Workbench, mit dem Graph2D jederzeit wieder aktiviert werden kann
- Ende - Beendet Graph2D nach Sicherheitsabfrage

## 1.10 Funktionseditor

### 2.2 Funktionseditor

---

Der Funktionseditor ist ein Fenster, in dem alle im Speicher befindlichen Funktionen aufgelistet und bearbeitet werden können.

Von hier aus sind folgende Programmfunktionen erreichbar:

- |       |                  |   |                                       |
|-------|------------------|---|---------------------------------------|
| 2.2.1 | math. Notation   | - | Die Notations-Fenster                 |
| 2.2.2 | Kurvendiskussion | - | Das Kurvendiskussionsfenster          |
| 2.2.3 | num. Integral    | - | Berechnung des Simpsonintegrals       |
| 2.2.4 | Wertetabelle     | - | Erstellen einer Wertetabelle          |
| 2.2.5 | Tangente         | - | Anlegen einer Tangente                |
| 2.2.6 | Taylorreihe      | - | Approximationen mit dem Taylorpolynom |

Die Hilfezeile am unteren Fensterrand zeigt einen kurzen Hilfstext über den Fensterbestandteil an, über dem sich der Mauszeiger gerade befindet.

Eine Funktion aus der Funktionsliste erscheint durch Anklicken im Funktionen-Gadget und kann nun editiert werden. Wenn die Eingabe bzw. Änderung der Funktion durch <RETURN> beendet ist, wird sie automatisch interpretiert. (Der Auswahlbalken kann übrigens auch aus dem Funktionen-Gadget heraus mit den Cursortasten bewegt werden.)

Tritt beim Interpretieren ein Fehler auf, wird dieser in der "Hilfezeile" angezeigt, wenn sich der Mauszeiger über dem Funktions-Gadget befindet. Eine fehlerhafte Funktion wird in der Funktionsliste kursiv dargestellt und kann natürlich nicht gezeichnet oder analysiert werden!

Die HELP-Taste über dem Funktions-Gadget gedrückt bewirkt, daß direkt das "Funktionen"-Kapitel der Anleitung angezeigt wird, mit dessen Hilfe man schnell eventuelle Fehler in der Syntax erkennen kann.

Im Graph-Gadget mit beigeordneter Popup-Liste werden die existierenden Graphen mit ihrem Namen angezeigt. Jedem Graphen kann ein individueller Name gegeben werden, unter dem er in der Liste erscheinen wird. Ein Doppelklick auf einen Graphen in der Liste bringt diesen sofort in den Vordergrund.

Von jeder augenblicklich ausgewählten Funktion kann eingestellt werden, ob sie im ausgewählten Graphen eingezeichnet werden soll. Ein Doppelklick auf eine Funktion schaltet ihren Zustand zwischen gezeichnet und nicht-gezeichnet hin und her. Dies ist identisch mit einem Klick auf das "Gezeichnet"-Gadget. Im aktiven Graphen gezeichnete Funktionen werden in der Liste durch ein vorangestelltes ">" gekennzeichnet.

An einer eingezeichneten Funktion gemachte Änderungen werden normalerweise erst dann sichtbar, wenn im Graphen "aktualisieren" angeklickt wird. Auf diese Weise wird nicht nach jeder einzelnen Änderung der ganze Graph aufwendig neu berechnet, sondern erst, wenn dies gewünscht wird. Durch den "Auto-Aktualisieren"-Modus, der im Voreinsteller eingestellt werden kann, kann man Graph2D dazu bringen, einen Graphen nach jeder Änderung (auch Entfernen/Hinzufügen) einer in ihm gezeichneten Funktion automatisch neu gezeichnet wird.

Im Funktionseditor stehen folgende Bild-Buttons zur Verfügung (zu jedem Button existiert aber auch ein entsprechender Menüpunkt):

- Neuer Graph - Brint einen Requester zum Vorschein in dem ausgewählt werden kann, ob ein neuer 2D, 3D oder SIRDS-Graph geschaffen werden soll.
- Voreinstellungen - Öffnet das Programmeinstellungs-Fenster
  
- Vereinfachen - Eine äquivalente Vereinfachung wird erzeugt, d.h. die Funktion wird möglichst weit "ausgerechnet".
- Ableitung - Erzeugt die (erste) Ableitung
- Diskussion - Unternimmt eine konfigurierbare Kurvendiskussion
- num. Integral - Berechnet das bestimmte Integral
- Wertetabelle - Erzeugt eine Wertetabelle
- Tangente - Erzeugt eine Tangente
- Taylorpolynom - Erzeugt eine Taylor-Approximation
- math. Notation - Öffnet das Notations-Fenster
  
- ARexx-Skript - Öffnet einen File-Requester, in dem ein ARexx-Skript ausgewählt werden kann, das ausgeführt wird

Solange der Funktionseditor das aktive Fenster ist, stehen folgende Menüs zur Verfügung:

- Projekt - Das Projekt-Menu

#### Funktionen

- Alle löschen - Alle Funktionen werden gelöscht
- Laden ... - Eine Funktionsliste wird geladen
- Laden dazu... - Eine Funktionsliste wird geladen und an die im Speicher vorhandene angehängt
- Speichern als... - Die Funktionsliste im Speicher wird unter wählbarem Namen abgespeichert

#### Edit

- Erzeugen - Eine neue, "leere" Funktion wird an ausgewählter Stelle in die Funktionsliste eingesetzt und kann sofort editiert werden.
- Ausschneiden - Die ausgewählte Funktion wird aus der Liste ausgeschnitten.
- Einfügen - Eine zuvor ausgeschnittene Funktion wird an ausgewählter Stelle wieder in die Funktionsliste eingefügt.

#### Sonstiges

- Vereinfachen - wie der entsprechende Bildbutton (s.o.)
- Ableitung - s.o.
- Diskussion - s.o.
- num. Integral... - s.o.
- Wertetabelle... - s.o.
- Tangente ... - s.o.
- Taylorreihe... - s.o.
- math. Notation... - s.o.

- Rexx - siehe Kapitel ARexx

## 1.11 Notation

### 2.2.1 Notation

In einem Fenster wird die ausgewählte Funktion in einer etwas lesbareren, "normaleren" Form gezeigt. Divisionen werden hierbei mit Bruchstrichen dargestellt und Exponenten sind richtig hochgestellt. Die ganze Sache heißt (vielleicht etwas unglücklich) "mathematische Notation"...

Ist die math. Notation einer Funktion größer als das Fenster, kann der sichtbare Ausschnitt mit den Verschieberegulern bewegt werden. Geschlossen wird das Fenster durch das Schließ-Gadget.

Von den Notations-Fenstern können mehrere gleichzeitig geöffnet sein. Auch während sie offen sind, kann anderweitig weitergearbeitet werden.

Folgende Menüs können vom Notations-Editor angewählt werden:

Projekt	- Das Projekt-Menü
Notation	
Drucken	- Druckt die sichtbare Notation aus. Die Druckdichte kann durch die Workbench-Prefs eingestellt werden.
Speichern IFF...	- Speichert die Notation als IFF-Graphik ab. Sie kann so von allen Programmen verwendet werden, die das IFF-ILBM-Format unterstützen. Die Größe des IFF-Bildes hängt von der verwendeten Druck-Schrift ab.

## 1.12 Diskussion

### 2.2.2 Kurvendiskussion

Graph2D ist in der Lage, Funktionen einer Kurvendiskussion zu unterziehen. Voraussetzung dafür ist eine ganzrationale Funktion mit einer Laufvariablen.

In diesem Fenster muß zunächst das sogenannte Untersuchungsintervall eingegeben werden, in dessen Grenzen die Diskussion durchgeführt werden soll. Das Untersuchungsintervall sollte stets so klein wie möglich gehalten werden - je größer es gewählt wird, desto schlechter findet Graph2D die gesuchten Stellen der Funktion. Das Ergebnis einer jeden Diskussion wird in Klartext in einem Textanzeigefenster ausgegeben und kann von dort aus auch ausgedruckt werden.

Eine Kurvendiskussion wird mit Start begonnen und gegebenenfalls mittels Abbruch vorzeitig beendet werden.

Die Diskussion einer Funktion kann (besonders auf 68000er Amigas) mitunter recht lange dauern, auch sind die Ergebnisse meist mit Vorsicht zu genießen. Es empfiehlt sich immer, sich zur Kontrolle den Graphen der betreffenden Funktion zeichnen zu lassen und die Ergebnisse der Diskussion

mit seiner Hilfe zu verifizieren! Man wird schnell feststellen, daß bspw. bestimmte Nullstellen leider einfach nicht gefunden werden.

Eine komplette Diskussion besteht aus folgenden Teilen, die alle einzeln zuschaltbar sind:

Nullstellen:  $f(x) = 0$

Die Nullstellen der ausgewählten Funktion im Untersuchungsintervall werden bestimmt. Zu jeder Nullstelle bekommt man auch den Typ der Nullstelle angezeigt. Mögliche Typen sind:

```
-+      (von unten nach oben)   :  f'(x) > 0
+-      (von oben nach unten)   :  f'(x) < 0
++      (Berührstelle von oben) :  f'(x) = 0
--      (Berührstelle von unten):  f'(x) = 0
```

Auf die Nullstellenbestimmung beruhen auch alle anderen Diskussionsteile. Momentan arbeitet der verwendete Algorithmus leider jedoch nicht ganz zufriedenstellend, einige Nullstellen, besonders bei einer größeren Anzahl im Untersuchungsintervall, werden nicht erkannt - das fälschliche Erkennen einer Nullstelle allerdings kommt so gut wie nicht vor.

Extrema:  $f'(x) = 0$

Hierbei werden die lokalen Minima (Tiefpunkte) und Maxima (Hochpunkte) der Funktion im Untersuchungsintervall bestimmt.

```
min (Minimum):  f''(x) > 0
max (Maximum):  f''(x) < 0
```

Wendepunkte:  $f''(x) = 0$

Die Wendepunkte der ausgewählten Funktion werden ausgegeben, dazu erhält man noch die eine der folgende zusätzlichen Information:

```
links->rechts:  f'''(x) < 0
rechts->links:  f'''(x) > 0
```

Monotonie:

Das Untersuchungsintervall wird in Bereiche unterteilt, in denen die ausgewählte Funktion folgende Monotonieeigenschaften zeigt:

```
monoton steigend      :  f'(x) >= 0
monoton fallend       :  f'(x) <= 0
streng monoton steigend:  f'(x) > 0
streng monoton fallend :  f'(x) < 0
```

Symmetrie:

Hierbei wird die ausgewählte Funktion geprüft auf

```
Punktsymmetrie (zum Ursprung): f(x) = - f(-x)
Symmetrie zur y-Achse         : f(x) =  f(-x)
```

Trifft beides nicht zu, wird "keine Symmetrie" erkannt.

Die Berechnung der Symmetrie erfolgt durch Einsetzen einer größeren Anzahl von Testwerten.

## 1.13 num. Integration

### 2.2.3 Numerische Integration

Funktionen mit einer Laufvariablen können mittels der Simpsonformel numerisch integriert werden. Dazu muß mit "von" und "bis" ein Integrationsintervall angegeben und durch "Zerlegungen" die Genauigkeit der Integration eingestellt werden. Ist dies geschehen, wird in einem Textanzeigefenster ein Report erstellt, der das Resultat der Integration bekannt gibt. Es ist zu beachten, daß "Flächen" die unterhalb der x-Achse liegen mit negativem Vorzeichen bedacht werden, so daß z.B. das Integral von "sin(x)" im Bereich [0, 2\*Pi] genau 0 beträgt, da sich die positive und die negative "Fläche" gegenseitig aufheben. Ist man allerdings nur an der Maßzahl dieser Flächen interessiert (also am nichtorientierten Flächeninhalt), so könnte man das in diesem Beispiel durch die Betrachtung von "abs(sin(x))" erreichen.

Simpsonsche Regel:

a, b: Integrationsintervallgrenzen

k : Anzahl der Zerlegungen

$$\int_a^b f(x) dx = \frac{b-a}{k*3} * ( f(x_0) + 4*f(x_1) + 2*f(x_2) + \dots + 2*f(x_{k-2}) + 4*f(x_{k-1}) + f(x_k) )$$

## 1.14 Wertetabelle

### 2.2.4 Wertetabelle

Von Funktionen mit max. einer Laufvariablen können Wertetabellen errechnet werden. Die Grenzen der Tabelle werden mit "von" und "bis" eingestellt, in "Schrittweite"-breiten Abständen erfolgt die Berechnung der Werte. Die Ausgabe erfolgt ein einem Textanzeigefenster.

## 1.15 Tangente anlegen

### 2.2.5 Tangente anlegen

An Funktionen mit max. einer Laufvariablen können Tangenten an beliebigen

Punkten "x" angelegt werden. Dabei wird eine neue Funktion erzeugt, die die Gleichung der gewünschten Tangente besitzt. Diese Funktion besitzt den Anhang "tan" in ihrem Namen.

## 1.16 Taylorreihe

### 2.2.6 Taylorreihe

Funktionen mit max. einer Laufvariablen können durch ein Taylorpolynom in einer Umgebung um einen wählbaren Punkt  $x_0$  angenähert werden. Je höher der Grad gewählt wird, desto genauer wird die Approximation. Im Normalfall sollte man den Grad jedoch nicht unbedingt  $> 10$  wählen, da sonst der Speicher- und Zeitbedarf sprunghaft gegen unendlich strebt... Die erzeugte Polynom-Funktion wird in die Funktionsliste aufgenommen und bekommt ein vorgestelltes "T" in ihrem Namen.

Taylorreihe:  $x_0$  = gewünschter Antragspunkt

$$\sum_{n=0}^{\infty} \frac{f^{(n)}(x_0) * x_0^n}{n!} * (x - x_0)^n$$

Graph2D summiert natürlich nicht unendlich ( $\infty$ ) und bringt die Ausgabe in die gewohnte Polynom-Form. Deshalb wird folgende, angepasste Berechnung durchgeführt ( $n$  = gewünschter Grad):

$$\sum_{k=0}^n \frac{f^{(k)}(x_0) * (-x_0)^{n-k}}{k! * (n-k)!} * x^k$$

## 1.17 Graph-Fenster

### 2.3 Graph-Fenster

Graph-Fenster können mit "Neuer Graph" in beliebiger Anzahl vom Benutzer geöffnet werden. Es erscheint dann ein Auswahl-Requester, in dem man wählen kann, was für einen Graph-Typen man öffnen möchte. Es gibt folgende Graph-Typen:

2D  
3D  
SIRDS

Es ist auch möglich, einen Graphen beim Öffnen mit "Graph als Hintergrund" formatfüllend als Hintergrund-Fenster zu verwenden. Den Funktionseditor kann man ja bei Bedarf auch schließen, um den Graphen besser betrachten zu können. Da so ein Hintergrund-Graph kein Schließ-Symbol hat, wurde in das "Graph"-Menü der Eintrag "Schließen" aufgenommen.

Die Graph-Fenster und der Funktionseditor laufen parallel, so daß zwischen ihnen hin- und hergeschaltet und "gleichzeitig" in ihnen gearbeitet werden kann. Graph-Fenster sind in der Größe veränderbar. Das momentane Darstellungsintervall und der Name jedes Graphen werden in der Titelleiste des Graph-Fensters angezeigt. Zu jedem Graphen gehört noch ein Einstellungs-Fenster, in dem die Art der Darstellung den eigenen Wünschen angepaßt werden kann.

In der Hilfezeile am unteren Fensterrand eines jeden Graph-Fensters wird ständig ein kleiner Hilfstext über das Objekt unter dem Mauszeiger angezeigt.

Welche Funktionen in einem Graph-Fenster angezeigt werden, läßt sich im Funktionseditor einstellen, indem man den gewünschten Graphen aus der Popup-Liste auswählt und die Funktion dann für diesen Graphen als "Gezeichnet" markiert.

## 1.18 2D-Graph-Fenster

### 2.3.1 2D-Graph-Fenster

In 2D-Graph-Fenstern können Funktionen (auch mehrere gleichzeitig) mit maximal einer Laufvariablen in einem zweidimensionalen Koordinatensystem betrachtet werden. Welche Funktionen in einem Graph-Fenster angezeigt werden, läßt sich im Funktionseditor einstellen.

Ein 2D-Graph läßt sich mit dem 2D-Einsteller konfigurieren.

Die optionale Bildbuttonleiste ganz oben im Fenster bietet folgende Funktionen an, die auch durch die Menus und deren Shortcuts erreichbar sind:

- Neuer Graph - Brint einen Requester zum Vorschein in dem ausgewählt werden kann, ob ein neuer 2D, 3D oder SIRDS-Graph geschaffen werden soll.
- Voreinstellungen - Öffnet das Programmeinstellungs-Fenster
- Aktualisieren - der Graph wird mit allen eventuell in der Zwischenzeit erfolgten Änderungen neu gezeichnet. An einer eingezeichneten Funktion gemachte Änderungen werden also erst dann sichtbar, wenn der Graph aktualisiert wird! Auf diese Weise wird nicht nach jeder einzelnen Änderung der ganze Graph aufwendig neu berechnet, sondern erst, wenn dies auch gewünscht wird. Auch das Vergrößern und Verkleinern eines Graphfensters geht relativ schnell vonstatten, da alle Funktionswerte gepuffert sind und erst durch "Aktualisieren" neu berechnet werden.

Aktualisiert wird ein Graph automatisch von alleine nach Verlassen seines Einstellers und im Auto-Aktualisieren-Modus auch nach jeder Änderung der gezeichneten Funktionen.

- Einstellen - Öffnet den 2D-Graph Einsteller, von dem aus alle Einstellungen des Koordinatensystems festgelegt werden können.
- Zoom in - Mit der Maus kann nach Anwahl dieses Punktes im Graphen ein Bereich markiert werden. Nach Bestätigung öffnet sich dann ein neues Graph-Fenster, das genau diesen Bereich anzeigt.
- Zoom out - Ein neuer Graph öffnet sich, der den vierfachen Darstellungsbereich des Ursprungsgraphen umfaßt.
- Drucken - Druckt den Graphen aus
- Speichern IFF - Speichert den Graphen als IFF-Bild, so daß er von allen Programmen, die das IFF-ILBM-Format unterstützen verwendet werden kann. Die Pixel-Größe in der der Graph, abgespeichert werden soll kann in den Voreinstellungen festgelegt werden. Der Graph wird fürs Speichern intern in der vollen Pixelgröße berechnet und aufgebaut.

Das momentane Darstellungsintervall und der Name des Graphen werden in der Titelleiste des Graph-Fensters angezeigt. Solange man die linke Maustaste gedrückt hält, wird die Position des Mauszeigers in Graph-Koordinaten umgerechnet in der Textzeile oberhalb des Graphen angezeigt.

Der sichtbare Graph-Ausschnitt kann mittels den Cursorstasten in kleinen Schritten verschoben werden.

Solange ein 2D-Graph-Fenster aktiv ist, stehen folgende Menüs zur Verfügung:

Projekt - Das Projekt-Menü

#### Graph

- Aktualisieren - Wie entsprechender Bildbutton (s.o.)
- Zoom in - s.o.
- Zoom out - s.o.
- Drucken... - s.o.
- Speichern IFF... - s.o.
- Schliessen - Schließt das Graph-Fenster. Das gleiche passiert natürlich auch beim Anklicken das normalen Fenster-Schließ-Symbols.

#### Einstellungen

- Achsen... - Öffnet den 2D-Graph Einsteller mit aktiver "Achsen"-Seite
- Funktionen... - Öffnet den 2D-Graph Einsteller mit aktiver "Funktionen"-Seite
- Sonstiges... - Öffnet den 2D-Graph Einsteller mit aktiver "Sonstiges"-Seite

- Fenster... - Öffnet den 2D-Graph Einsteller mit aktiver "Fenster"-Seite
- Speichern... - Speichert die Einstellungen des Graphen zur späteren Verwendung unter wählbarem Namen ab.
- Laden... - Lädt eine Graph-Einstellungsdatei und wendet sie auf den Graphen an.

## 1.19 2D-Graph-Einsteller

### 2.3.2 2D-Graph Einsteller

In diesem Einsteller wird der Darstellungsbereich, die Achsenskalierung und einige weitere Merkmale eines Graphen verändert. Der Einsteller ist in vier verschiedene Seiten unterteilt:

Seite: Achsen

Für die x- und die y-Achse können jeweils unabhängig voneinander folgende Angaben gemacht werden:

von ... bis - Das Darstellungsintervall für die betreffende Achse. Die gemachte Angabe ist abhängig von der gewählten Einheit!

Einheit - Die Basiseinheit der Achse, mit der alle anderen Angaben skaliert werden. Üblich ist eine Einheit von "1". Wenn man aber z.B. ein Koordinatensystem speziell für trigonometrische Funktionen erstellen will, so würde als Einheit sicher auch "3.141" sinnvoll sein. Ein Darstellungsintervall von z.B. [-2; 4] würde in diesem Fall bedeuten, daß das tatsächliche Intervall von  $-2 * 3.141$  bis  $4 * 3.141$  reichen würde, also effektiv x-Werte aus [-6.282;12.564] umfaßt. Die Einheit ist frei wählbar, solange sie  $>0$  ist.

Achse zeichnen - Legt fest, ob die Achse überhaupt sichtbar sein soll. Auch wenn sie nicht gezeichnet wird, sind die Angaben über den Darstellungsbereich natürlich trotzdem nötig.

Titel - Der Anwender kann einer Achse einen Titel geben, der neben der Achse angezeigt wird. Wählt man z.B. für die x-Achse eine Einheit von "3.141" so ist es sicher wichtig, dies auch im Titel kenntlich zu machen, in dem man sie z.B. "x in pi" nennt.

Striche - Die Achse kann optional durch kleine Striche unterteilt werden. Hier kann man nun festlegen, nach wieviel Einheiten jeweils ein Strich gezeichnet werden soll. Außerdem können zwischen jeweils zwei dieser Striche eine beliebige Anzahl an Zwischenstrichen gezeichnet werden, die dann etwas kleiner als die eigentlichen Striche sind. Eine Zwischenstrichanzahl von 0 bedeutet keine Zwischenstriche. Folgen die Striche zu eng aufeinander, daß man sie nicht mehr unterscheiden könnte, so werden sie automatisch einfach nicht dargestellt.

Beschriftung - Die Striche können optional mit Zahlen beschriftet werden. Es kann festgelegt werden, nach wievielen Strichen wieder eine Zahl geschrieben werden soll. Die geschriebene Zahl ist auch wiederum abhängig

von der gewählten Einheit. Falls die Beschriftung so eng gewählt wird, daß sich Zahlen überschneiden würden, so wird das automatisch verhindert, indem manche einfach ausgelassen werden.

Seite: Funktionen

Hier können für jede Funktion, die im Graphen eingezeichnet ist, bestimmte Darstellungs-Einstellungen gemacht werden. Die Funktion, deren Darstellungs-Parameter man verändern möchte, klickt man einfach in der Liste an. Der Listeneintrag "< default settings >" hat eine besondere Bedeutung: Die für ihn eingestellten Werte gelten für alle Funktionen, die eventuell später noch dem Graphen zum Einzeichnen übergeben werden. Diese "default settings" werden auch im Gegensatz zu den einzelnen Funktionen mit den anderen Grapheneinstellungen abgespeichert. Hier nun die einzelnen Darstellungsparameter einer Funktion:

**Farbe** - In dieser Farbe erscheint die Funktion. Es sollten alle Bildschirmfarben zur Verfügung stehen.

**Strichstärke** - gibt an, ob Funktionen einen Punkt (normal) oder zwei Punkte breit (fett) gezeichnet werden sollen. Fettdruck ist zwar deutlich langsamer, aber z.B. zum Drucken gut geeignet.

**Genauigkeit** - Hiermit wird festgelegt, wie viele Funktionswerte für die Ausgabe der Funktion tatsächlich berechnet werden sollen. Bei geringerer Genauigkeit werden weniger Funktionswerte berechnet und die Ausgabe erfolgt in höherer Geschwindigkeit - jedoch wird der Funktionsgraph etwas "eckiger" und unpräziser im Verlauf.

**Punkte verbinden** - Wenn dieser Modus eingeschaltet ist, werden die einzelnen berechneten Funktionswerte durch Linien miteinander verbunden. Wenn ein Funktionsgraph größere Sprünge macht (z.B. bei geringer Genauigkeit), werden so die entstehenden Lücken verhindert. Andererseits können so aber auch Verbindungen entstehen, die in Wirklichkeit gar nicht existieren (Beispiel:  $\tan(x)$ ).

**Pattern** - Gibt das Muster (die "Strichelung") an, mit dem die Funktion gezeichnet werden soll. Es stehen mehrere vordefinierte Muster zur Auswahl. Die Angabe eines Patterns ist natürlich nur dann sinnvoll, wenn "Punkte verbinden" eingeschaltet ist, da der Graph sonst ja ohnehin nur aus einzelnen Punkten besteht.

Seite: Sonstiges

Für beide Achsen kann angegeben werden, ob ein "Raster" in Form von Punkten oder Linien gezeichnet werden soll und welchen Abstand es in x- und y-Richtung (Wert wieder in Einheiten) haben soll. Der Typ legt fest, ob das Raster nur aus einzelnen Punkten, aus durchgezogenen Linien oder aus gestrichelten Linien bestehen soll.

Die "Strichstärke" gibt an, ob Achsen einen Punkt (normal) oder zwei Punkte breit (fett) gezeichnet werden sollen. Fettdruck ist zwar deutlich langsamer, aber z.B. zum Drucken gut geeignet.

"Schrift" erlaubt die Wahl des Zeichensatzes für den Graphen, in der die

Einheitszahlen und die Achsenbeschriftung dargestellt werden.

Seite: Fenster

Hier können die "Bildbuttons" am oberen und die "Statuszeile" am unteren Rand eines Graph-Fensters an- und ausgeschaltet werden.

Nach Verlassen des 2D-Graph-Einstellers mit "Ok" wird der aktive Graph aktualisiert, um die evtl. gemachten Änderungen sichtbar zu machen. Beim Verlassen durch "Abbruch" oder Schließen des Fensters bleibt es natürlich bei den vorherigen Werten. "Speichern" und "Laden" erlauben das Abspeichern und Einladen von Graph-Einstellungen unter beliebigen Namen.

## 1.20 3D-Graph-Fenster

### 2.3.3 3D-Graph-Fenster

In einem 3D-Graph-Fenster kann eine Funktion mit maximal zwei Laufvariablen betrachtet werden. Die Darstellung erfolgt als ausgefülltes "Drahtgittermodell", in dem unsichtbare Linien auch wirklich nicht sichtbar sind. Welche Funktion in einem 3D-Graph-Fenster angezeigt wird, läßt sich im Funktionseditor einstellen.

Ein 3D-Graph läßt sich mit dem 3D-Einsteller konfigurieren.

Das momentane Darstellungsintervall und der Name des Graphen werden in der Titelleiste des Graph-Fensters angezeigt.

Die zwei Schieberegler am rechten und unteren Fensterrand erlauben es, die Perspektive etwas zu verändern, mit dem man auf den Graphen schaut. Die Wirkung probiert man am besten erst einmal mit geringer Genauigkeit aus, da nach jeder Änderung der Graph sofort neu aufgebaut wird.

Die drei Bild- bzw. Textbuttons haben folgende Bedeutung (v.l.n.r.):

- |                  |  |
|------------------|--|
| Neuer Graph      | - Bricht einen Requester zum Vorschein in dem ausgewählt werden kann, ob ein neuer 2D, 3D oder SIRDS-Graph geschaffen werden soll.   |
| Voreinstellungen | - Öffnet das Programmeinstellungs-Fenster  |
| Aktualisieren    | - der Graph wird mit allen eventuell in der Zwischenzeit erfolgten Änderungen neu gezeichnet. An der eingezeichneten Funktion gemachte Änderungen werden also erst dann sichtbar, wenn der Graph aktualisiert wird! Auf diese Weise wird nicht nach jeder einzelnen Änderung der ganze Graph aufwendig neu berechnet, sondern erst, wenn dies auch gewünscht wird. Auch das Vergrößern und Verkleinern eines Graphfensters geht so relativ schnell vonstatten, da alle Funktionswerte gepuffert sind und erst durch "Aktualisieren" neu berechnet werden müssen. Aktualisiert wird ein Graph automatisch von alleine nach Verlassen seines Einstellers und im Auto-Aktualisieren-Modus auch nach jeder Änderung der gezeichneten Funk- |

- tionen.
- Einstellen - Öffnet den 3D-Graph-Einsteller
  - Drucken - Druckt den Graphen aus
  - Speichern IFF - Speichert den Graphen als IFF-Bild, so daß er von allen Programmen, die das IFF-ILBM-Format unterstützen verwendet werden kann. Die Pixel-Größe in der der Graph, abgespeichert werden soll kann in den Voreinstellungen festgelegt werden. Der Graph wird fürs Speichern intern in der vollen Pixelgröße berechnet und aufgebaut.

Solange ein 3D-Graph-Fenster aktiv ist, stehen folgende Menüs zur Verfügung:

- Projekt - Das Projekt-Menü
- Graph
- Aktualisieren - wie der entsprechende Bild-Button (s.o.)
  - Drucken - s.o.
  - Speichern IFF... - Speichert den Graphen als IFF-Bild, so daß er von allen Programmen, die das IFF-ILBM-Format unterstützen verwendet werden kann. Die Pixel-Größe in der der Graph, abgespeichert werden soll kann in den Voreinstellungen festgelegt werden. Der Graph wird dafür intern in der vollen Pixelgröße berechnet und aufgebaut.
  - Schliessen - Schließt das Graph-Fenster. Das gleiche passiert natürlich auch beim Anklicken das normalen Fenster-Schließ-Symbols.
- Einstellungen
- Achsen... - Öffnet den 3D-Graph-Einsteller mit aktiver Einstellungs-Seite "Achsen".
  - Farben... - Öffnet den 3D-Graph-Einsteller mit aktiver Einstellungs-Seite "Farben".
  - Sonstiges... - Öffnet den 3D-Graph-Einsteller mit aktiver Einstellungs-Seite "Sonstiges".
  - Fenster... - Öffnet den 3D-Graph-Einsteller mit aktiver Einstellungs-Seite "Fenster".
  - Seichern... - Speichert die Einstellungen des Graphen zur späteren Verwendung unter wählbarem Namen ab.
  - Laden... - Lädt eine Graph-Einstellungsdatei und wendet sie auf den Graphen an.

## 1.21 3D-Graph-Einsteller

### 2.3.4 3D-Graph-Einsteller

In diesem Einsteller wird der Darstellungsbereich und einige weitere Merkmale eines 3D-Graphen verändert. Dies Einstellungsmöglichkeiten sind auf vier Seiten verteilt:

---

Seite: Achsen

Für die x-, y- und z-Achse kann jeweils ein Darstellungsintervall festgelegt werden. x- und y-Achse verlaufen später im Graphen auf einer waagrechten Ebene, wohingegen die z-Achse senkrecht nach oben ragt.

Seite: Farben

Der dreidimensionale Graph einer Funktion besteht aus einem "Flickenteppich" von Flächen, die immer abwechselnd mit einer von zwei Farben ausgefüllt sind. Mit Funktion1 und Funktion2 können diese beiden Farben festgelegt werden. Es sollten alle Bildschirmfarben zur Verfügung stehen. Man kann selbstverständlich auch für beide die gleiche Farbe einstellen.

Die Farbe des Achsenkreuzes und der Flächenumrandungen im Graphen wird mit Achsen eingestellt.

Seite: Sonstiges

Es kann eine Genauigkeit bestimmt werden, mit der ein Graph gezeichnet werden soll. Bei geringerer Genauigkeit werden weniger Funktionswerte berechnet und die Ausgabe erfolgt in höherer Geschwindigkeit - jedoch wird der Funktionsgraph "eckiger" und die Flächen werden immer größer.

Mit Achsen bestimmt man, ob als "Koordinatensystem" ein Kasten ("Box"), ein konventionelles Achsenkreuz oder gar nichts gezeichnet werden soll.

Verläßt eine Funktion den Darstellungsbereich der z-Achse, so kann mit Grenzflächen eingestellt werden, ob an diesen Stellen eine waagrechte Fläche am oberen bzw. unteren Rand des Darstellungsbereiches gezeichnet werden soll. Dies sieht oft besser aus, als wenn die Funktion einfach verschwindet - allerdings muß man sich darüber klar sein, daß an diesen Stellen die Funktion nicht wirklich plan verläuft.

Seite: Fenster

Hier können die "Bildbuttons" am oberen und die "Statuszeile" am unteren Rand eines Graph-Fensters an- und ausgeschaltet werden.

Graph-Einstellungen können mit Laden... und Speichern... unter beliebigen Namen gespeichert und später wieder geladen werden.

Nach Verlassen des 3D-Einstellers mit Ok wird der aktive Graph aktualisiert, um die evtl. gemachten Änderungen sichtbar zu machen. Beim Verlassen durch Abbruch oder Schließen des Fensters bleibt es natürlich bei den vorherigen Werten.

## 1.22 SIRDS-Graph-Fenster

### 2.3.5 SIRDS-Graph-Fenster

---

In einem SIRDS-Graph-Fenster kann eine Funktionen mit maximal zwei Laufvariablen betrachtet werden. Zur Art der Darstellung siehe Kapitel SIRDS.

Welche Funktion in einem SIRDS-Graph-Fenster angezeigt wird, läßt sich im Funktionseditor einstellen. Das momentane Darstellungsintervall und der Name des Graphen werden in der Titelleiste des Graph-Fensters angezeigt.

Ein SIRDS-Graph läßt sich mit dem SIRDS-Einsteller konfigurieren.

Die drei Bild- bzw. TextButtons haben folgende Bedeutung (v.l.n.r):

- |                  |  |
|------------------|--|
| Neuer Graph      | - Brint einen Requester zum Vorschein in dem ausgewählt werden kann, ob ein neuer 2D, 3D oder SIRDS-Graph geschaffen werden soll.  |
| Voreinstellungen | - Öffnet das Programmeinstellungs-Fenster  |
| Aktualisieren    | - der Graph wird mit allen eventuell in der Zwischenzeit erfolgten Änderungen neu gezeichnet. An der eingezeichneten Funktion gemachte Änderungen werden also erst dann sichtbar, wenn der Graph aktualisiert wird! Auf diese Weise wird nicht nach jeder einzelnen Änderung der ganze Graph aufwendig neu berechnet, sondern erst, wenn dies auch gewünscht wird. Auch das Vergrößern und Verkleinern eines Graphfensters geht so relativ schnell vonstatten, da alle Funktionswerte gepuffert sind und erst durch "Aktualisieren" neu berechnet werden müssen. Aktualisiert wird ein Graph automatisch von alleine nach Verlassen seines Einstellers und im Auto-Aktualisieren-Modus auch nach jeder Änderung der gezeichneten Funktionen. |
| Einstellen       | - Öffnet den SIRDS-Graph-Einsteller  |
| Drucken          | - Druckt den Graphen aus   |
| Speichern IFF    | - Speichert den Graphen als IFF-Bild, so daß er von allen Programmen, die das IFF-ILBM-Format unterstützen verwendet werden kann. Die Pixel-Größe in der der Graph, abgespeichert werden soll kann in den Voreinstellungen festgelegt werden. Der Graph wird fürs Speichern intern in der vollen Pixelgröße berechnet und aufgebaut.   |

Solange ein SIRDS-Graph-Fenster aktiv ist, stehen folgende Menüs zur Verfügung:

Projekt - Das Projekt-Menü

Graph

- |                  |   |
|------------------|---|
| Aktualisieren    | - wie der entsprechende Bild-Button (s.o.)  |
| Drucken          | - s.o.  |
| Speichern IFF... | - Speichert den Graphen als IFF-Bild, so daß er von allen Programmen, die das IFF-ILBM-Format unterstützen verwendet werden kann. Die Pixel-Größe in der der Graph, abgespeichert werden soll kann in den Voreinstellungen festgelegt werden. Der Graph wird dafür intern in der vollen Pixelgröße berechnet und aufgebaut. |
| Schliessen       | - Schließt das Graph-Fenster. Das gleiche passiert natürlich auch beim Anklicken das normalen Fenster-  |

Schließ-Symbols.

Einstellungen

- |              |   |
|--------------|---|
| Achsen...    | - Öffnet den SIRDS-Graph-Einsteller mit aktiver Seite "Achsen"                              |
| Sonstiges... | - Öffnet den SIRDS-Graph-Einsteller mit aktiver Seite "Sonstiges"                           |
| Fenster...   | - Öffnet den SIRDS-Graph-Einsteller mit aktiver Seite "Fenster"                             |
| Speichern... | - Speichert die Einstellungen des Graphen zur späteren Verwendung unter wählbarem Namen ab. |
| Laden...     | - Lädt eine Graph-Einstellungsdatei und wendet sie auf den Graphen an.                      |

## 1.23 SIRDS-Graph-Einsteller

### 2.3.6 SIRDS-Graph-Einsteller

In diesem Einsteller wird der Darstellungsbereich, die Achsenskalierung und einige weitere Merkmale eines SIRDS-Graphen verändert. Der Einsteller hat drei Seiten:

Seite: Achsen

Für die x-, y- und z-Achse kann jeweils ein Darstellungsintervall festgelegt werden. x- und y-Achse verlaufen später im Graphen auf einer imaginären Ebene, wohingegen die z-Achse dem Betrachter "auf magische Weise" entgegentzukommen scheint.

Seite: Sonstiges

Mit Farben kann festgelegt werden, mit wievielen der verfügbaren Farben der Graph aufgebaut werden soll. Viele empfinden mehrere Farben angenehmer als einen monochromen Graphen (Farben=2), der seinerseits zum Ausdrucken besser geeignet wäre. Auf die Geschwindigkeit hat die Farbanzahl keinen Einfluß.

Die Transparenz bestimmt, wieviele Weiß-Anteile der Graph enthalten soll. Je höher der Wert, desto lichter und heller erscheint der Graph, da einfach weniger Punkte gezeichnet werden, was besonders beim Ausdrucken wichtig sein könnte.

Außerdem kann eine Genauigkeit ausgewählt werden, mit der ein Graph gezeichnet werden soll. Bei geringerer Genauigkeit werden weniger Funktionswerte berechnet und die Berechnung erfolgt in deutlich höherer Geschwindigkeit - jedoch wird der Funktionsgraph zuweilen etwas "eckiger" und der dreidimensionale Effekt erscheint "stufig". Trotzdem sind die geringen Genauigkeiten von großer Bedeutung, da das Berechnen eines SIRDS mitunter recht lange dauern kann...

Seite: Fenster

Hier können die "Bildbuttons" am oberen und die "Statuszeile" am unteren Rand eines Graph-Fensters an- und ausgeschaltet werden.

Graph-Einstellungen können mit Laden... und Speichern... unter beliebigen Namen gespeichert und später wieder geladen werden.

Nach Verlassen des SIRDS-Einstellers mit Ok wird der aktive Graph aktualisiert, um die evtl. gemachten Änderungen sichtbar zu machen. Beim Verlassen durch Abbruch oder Schließen des Fensters bleibt es natürlich bei den vorherigen Werten.

## 1.24 Textanzeigefenster

### 2.4 Textanzeigefenster

Das Textanzeigefenster zeigt in Graph2D alle Arten von Texten in einem Listen-Gadget an. Mit dem Rollbalken kann man sich im Text bewegen und diesen mittels "Drucken" auch ausdrucken. "Speichern" öffnet einen Dateiauswahl-Requester, mit dem man den Text als ganz normalen ASCII-Text abspeichern kann, um ihn dann z.B. von einem Textverarbeitungsprogramm aus einladen zu können. Geschlossen wird das Fenster durch "Ok" oder das Schließ-Symbol.

Ein Textanzeigefenster muß nicht unbedingt geschlossen werden, um mit Graph2D weiterzuarbeiten.

## 1.25 Voreinstellungseditor

### 2.5 Voreinstellungseditor

Hier können folgende Voreinstellungen für Graph2D festgelegt werden:

- |             |  |
|-------------|--|
| Verzeichnis | - Dieses Verzeichnis wird von den Laden/Speichern-Requestern in den Grapheditoren vorgegeben   |
| 2D-Graph    | - Diese 2D-Grapheinstellungsdatei wird immer verwendet, wenn ein neuer 2D-Graph geöffnet wird  |
| 3D-Graph    | - Diese 3D-Grapheinstellungsdatei wird immer verwendet, wenn ein neuer 3D-Graph geöffnet wird  |
| SIRDS-Graph | - Diese SIRDS-Grapheinstellungsdatei wird immer verwendet, wenn ein neuer SIRDS-Graph geöffnet wird  |
| IFF-Größe   | - Ausmaße (in Pixel), die ein Graph bekommt, wenn er als IFF-Graphik abgespeichert wird  |
| Auto-Redraw | - Normalerweise wird ein Graph erst dann aktualisiert, wenn man auf Aktualisieren klickt. Ist "Auto-Redraw" eingeschaltet, so werden die Graphen automatisch bei jeder Änderung der in ihnen gezeichneten Funktionen |
-

aktualisiert.

- |               |  |
|---------------|--|
| Schrift       | - Diese Schrift wird in der math. Notation verwendet   |
| Druck-Schrift | - Schrift, in der die Notation ausgedruckt und als IFF gespeichert wird  |
| Buttonleiste  | - Da die Bildbuttons im Funktionseditor nicht jedermenschs Sache sind, können sie hier abgestellt oder durch Text-Buttons ersetzt werden |

Die Voreinstellungen können mit "Benutzen" bis zum Programmende verwendet werden. Mit "Speichern" werden sie dauerhaft gemacht. Bei "Abbruch" passiert natürlich gar nichts.

## 1.26 Drucken

### 2.6 Druck-Requester

Mit diesem Requester kann ein Graph über die in den Workbench-Printer-Settings getätigten Einstellungen ausgedruckt werden.

Selbst wählen darf man hier die "Breite" und "Höhe" des auf dem Papier entstehenden Bildes in Millimetern (mm).

Der Graph wird (abhängig von der in der Workbench gewählten Dichte) mehr oder weniger in der vollen Drucker-Auflösung intern erstellt und ausgedruckt. Bei hohen Auflösungen werden dabei die einzelnen Pixel mitunter sehr klein, so daß man z.B. für die Beschriftung auch schon mal einen sehr großen Font verwenden muß, um das ganze letztendlich auch noch lesen zu können.

## 1.27 ARexx

### 2.7 Der ARexx Port

ARexx ist die Amiga-Umsetzung der Skriptsprache Rexx, die zur Kommunikation zwischen Programmen und zur Automatisierung bestimmter Programmabläufe benutzt werden kann. Dazu muß ein ARexx-Programm (Skript) geschrieben werden.

Im Funktionseditor befindet sich ein Menu, das es erlaubt, beliebige ARexx-Skripts auszuführen. Einige der im Verzeichnis "Rexx" mitgelieferten Skripts sind direkt über Menupunkte anwählbar, andere oder eigene Skripts können mit "Skript ausführen..." in einem Dateirequester ausgewählt und gestartet werden.

Funktion der beigelegten Skripts

- SendToMathScript - Sendet die im Funktionseditor ausgewählte Funktion an das Programm MathScript. Falls MathScript noch nicht läuft, wird es gestartet. Die Funktion wird automatisch in das von MathScript erwartete Format umgewandelt.
- ExportEPS - Exportiert den Text der ausgewählten Funktion als EPS-Vektor-Graphik. Dies funktioniert über den Umweg "MathScript". Wer MathScript nicht installiert hat, kann deshalb auch keine EPS-Exporte vornehmen.
- GetInfos - Dieses unspektakuläre Skript demonstriert, wie man einem laufenden Graph2D ein paar Informationen entlocken kann.
- CloseGraph2D - Beendet Graph2D ohne Sicherheitsabfrage. Ziemlich sinnlos.

### Eigene ARexx-Befehle

Wie bei anderen MUI-Programmen auch ist der Name des ARexx-Ports von Graph2D "GRAPH2D.1". Laufen mehrere Graph2D's gleichzeitig, so lassen diese sich der Reihe nach mit "GRAPH2D.2", "GRAPH2D.3", u.s.w. ansprechen.

Neben grundlegenden ARexx-Kommandos wie 'QUIT', die jede MUI-Applikation besitzt und die in der MUI-Dokumentation erläutert sind, besitzt Graph2D zusätzlich noch folgende eigene Befehle:

- GETFUNCTION PLAIN - Liefert den Funktionstext der im Funktionseditor ausgewählten Funktion. Ist keine Funktion ausgewählt, wird ein leerer String "" retourniert.
- GETFUNCTION MS20 - Diese Programmfunktion von Graph2D ist nur über ARexx erreichbar. Wie bei der Option PLAIN liefert sie den Text der im Funktionseditor ausgewählten Funktion. Dieser Text liegt allerdings jetzt in einer Form vor, die die Formatierungsmöglichkeiten von MathScript nutzt, d.h. Wurzeln, Betragsstriche, Bruchstriche, Hochstellungen und der Buchstabe 'pi' werden in MathScript automatisch richtig gezeigt.

## 1.28 Funktion

### 3.1 Funktionen

Zum Thema Funktionen gibt es noch die folgenden Kapitel

- 3.1.1 Interne Operatoren & Funktionen
- 3.1.2 Gegenseitige Einbindung
- 3.1.3 EBNF-Syntax

Eine Funktion muß in Graph2D aus diesen Teilen bestehen:

### Name

Eine Funktion besteht zunächst einmal aus ihrem Namen. Dieser darf keine führenden Zahlen oder Leerzeichen enthalten und eine Größe von 30 Zeichen nicht überschreiten. Erlaubt sind Buchstaben (keine Umlaute), Zahlen (ab zweiter Stelle) und das einzelne Hochkomma (').

### Argumentenliste

Direkt an den Namen schließt sich die Argumentenliste an. Sie besteht aus mehreren durch Kommata getrennten Variablennamen. Für Variablennamen gilt das gleiche wie für Funktionsnamen, in einer Argumentenliste dürfen sie nicht doppelt vorkommen! Die Argumentenliste kann übrigens auch leer sein, wobei man sich dann sogar die Klammern sparen kann. Name und Argumentenliste bilden zusammen den sogenannten "Funktionsheader".

### Das Zeichen "="

Zwischen Funktionsheader und Funktionstext steht das Zeichen "=". Wer hätte das gedacht...

### Funktionstext

Die Definition des Funktionstextes erfolgt in der "üblichen" Notation für mathematische Terme, zwischen Groß- und Kleinschreibung wird (übrigens auch bei Variablen- und Funktionsnamen) unterschieden! Der Funktionstext besteht aus:

- Operatoren siehe Liste der internen Operatoren
- Funktionen siehe Liste der internen Funktionen

Außerdem können eigene Funktionen aus der Funktionsliste eingebunden werden.

- Konstanten

Eine Konstanten eine reelle Zahlen im Bereich  $\pm 9.22337177E18$

- Variablen

Natürlich dürfen in einer Funktion nur die Variablen verwendet werden, die in der Argumentenliste namentlich aufgeführt sind.

### Beispiele für richtige Funktionen:

```
"p(x)=x^2+98",
"Summe(a,b)=a+b",
"g=9.81",
"f(x)=sin(x^2)+e^(1/2*x)",
"funktion23a(iks, yps, zett)=iks^yps*zett+(-1.23+1.0356E-5)"
```

## 1.29 Interne Funktionen und Operatoren

### 3.1.1 Interne Funktionen und Operatoren

Bekannt sind folgende Funktionen und Operatoren:

$\sin(x)$ , $\arcsin(x)$ , $\sinh(x)$ , $\cos(x)$ , $\arccos(x)$ , $\cosh(x)$ , $\tan(x)$ , $\arctan(x)$ , $\tanh(x)$ , $\operatorname{arctanh}(x)$	- Dies sind die trigonometrischen Funktionen und ihre Umkehrfunktionen
$\operatorname{abs}(x)$	- Der Absolutwert des Arguments, auch Betrag $ x $ genannt
$\operatorname{int}(x)$	- Der ganzzahlige Teil des Arguments
$\ln(x)$	- natürlicher Logarithmus (zur Basis 2)
$\operatorname{sqrt}(x)$	- Die Quadratwurzel, also $x^{1/2}$
$\operatorname{sgn}(x)$	- Die "Vorzeichenfunktion", liefert je nach $x > 0$ , $x = 0$ oder $x < 0$ die Werte 1, 0 oder -1
$\operatorname{fak}(x)$	- Die Fakultät ( $x!$ ) z.B. $\operatorname{fak}(4) = 4 \cdot 3 \cdot 2 \cdot 1 = 24$ (nur sinnvoll bei positiven, ganzzahligen Werten)
$\operatorname{if}(a, b)$	- liefert $b$ zurück, wenn Argument $a$ WAHR (=1) ist, ansonsten ist der Funktionswert 0. (nützlich für partiell definierte Funktionen)
$\operatorname{binom}(n, k)$	- Binomialkoeffizient "n über k"
$\operatorname{exp}(x)$	- Exponentialfunktion $e^x$
$e$	- Euler'sche Zahl 2.71...
$\pi$	- 3.141...
$+$ , $-$	- Addieren und subtrahieren wie gewohnt
$*$ , $/$	- Multiplikation und Division
$^$	- Potenz-Operator (Vorsicht: $a^b^c = (a^b)^c$ )
$=$ , $>$ , $<$ , $>=$ , $<=$ , $<>$	- Vergleichsoperatoren, z.B. gilt $a > b = 1$ falls $a > b$ , ansonsten gilt $a > b = 0$
AND, OR, NOT	- Bool'sche Operatoren, momentan nur sinnvoll auf die booleschen Werte 1 (=WAHR) und 0 (=FALSCH) anwendbar, nicht bitweise.

## 1.30 Einbindung

### 3.1.2 Einbindung

Funktionen können sich gegenseitig einbinden, d.h. zu ihrer Definition andere Funktionen heranziehen. Dabei ist folgendes zu beachten:

Spätere Änderungen in der eingebundenen Funktion haben keine Auswirkungen mehr auf die einbindende Funktion, da sie zum Zeitpunkt des Interpretierens in die einbindende Funktion (intern) hineinkopiert wird. Um Änderungen auch in der einbindenden Funktion wirksam zu machen, muß diese lediglich neu interpretiert werden. (Im Textgadget einfach Return drücken.) Auf diese Weise wird ausgeschlossen, daß versehentlich "endlose" Rekursionen eingegeben werden.

Existieren mehrere Funktionen gleichen Namens, so ist eine eindeutige Zuweisung nicht mehr gewährleistet - es wird dann die erste Funktion aus der Liste der möglichen angenommen.

Beispiele:

```
"Summe(a,b)=a+b"
"f(x)=2^Summe(x,5)+x^2"           richtig

"k=5.6"
"Blabla(x,y)=k*x+y+1"           richtig

"f(x)=g(x)/6"
"g(x)=2^f(x)"                   falsch! (Zirkel!)

"u(x)=u(1/x)+3.14"              falsch! (Zirkel!)
```

## 1.31 EBNF

### 3.1.3 EBNF-Syntax

Leerzeichen sind (außer innerhalb von Bezeichnern) im gesamten Funktionstext erlaubt und können auch der Übersichtlichkeit dienen.

Zur Erläuterung: [ x ] : kein- oder einmaliges Vorkommen von x  
 { x } : kein- ein- und mehrmaliges Vorkommen von x  
 "xyz" : explizites Vorkommen der Zeichenkette xyz

```
Funktion = Bezeichner [ "(" [ Bezeichner { "," Bezeichner } ] ")" ]
          "=" Ausdruck .
```

```
Ausdruck = Priol [ ( "=" | ">" | "<" | "<=" | ">=" | "<>" )
              Priol ] .
```

```
Priol = [ "+" | "-" ] Prio2 { ( "+" | "-" | "OR" ) Prio2 } .
```

```
Prio2 = Prio3 { ( "/" | "*" | "AND" ) Prio3 } .
```

```
Prio3 = [ "NOT" ] Prio4 { "^" Prio4 } .
```

```
Prio4 = Zahl | "(" Ausdruck ")" | Bezeichner | Aufruf .
```

```
Aufruf = Bezeichner [ "(" Ausdruck { "," Ausdruck } ")" ] .
```

```
Zahl = Ziffer { Ziffer } [ "." Ziffer { Ziffer } ]
      [ "E" ["-"] Ziffer {Ziffer} ] .
```

```
Bezeichner = Buchstabe { Buchstabe | Ziffer } .
```

```
Ziffer = "0" | .. | "9" .
```

```
Buchstabe = "A" | .. | "Z" | "a" | .. | "z" | "'".
```

## 1.32 Benutzeroberfläche Mui

### 3.3 Benutzeroberfläche MUI

MUI-Copyright:

This application uses  
MUI - MagicUserInterface  
(c) Copyright 1993/94 by Stefan Stuntz

MUI is a system to generate and maintain graphical user interfaces. With the aid of a preferences program, the user of an application has the ability to customize the outfit according to his personal taste.

MUI is distributed as shareware. To obtain a complete package containing lots of examples and more information about registration please look for a file called "muiXXusr.lha" (XX means the latest version number) on your local bulletin boards or on public domain disks.

If you want to register directly, feel free to send

DM 30.- or US\$ 20.-

to

Stefan Stuntz  
Eduard-Spranger-Straße 7  
80935 München  
GERMANY

Anmerkungen zu MUI in Graph2D:

Um eine (hoffentlich) optisch ansprechende und komfortable Oberfläche zu bieten, benutzt Graph2D das MUI-System von Stefan Stuntz. Für Graph2D muß deshalb MUI in der Version 2.3 oder höher installiert sein.

Man darf MUI benutzen, ohne sich dafür registrieren zu lassen. In der registrierten Version hat man allerdings den Vorteil, daß alle Einstellungen in dem zu MUI gehörende Preferences Programm abgespeichert werden können und so jede MUI-Applikation (also auch Graph2D ;-)) in ihrem Aussehen dauerhaft den eigenen Wünschen angepaßt werden kann! Die wichtigsten Einstellungen lassen sich allerdings auch in der nicht registrierten Version von MUI tätigen.

Möchte man beispielsweise, daß sich Graph2D nicht mehr auf dem Workbench-Screen öffnet, so kann mit den MUI-Preferences leicht ein anderer Public-Screen oder auch ein eigener Bildschirm eingestellt werden, auf dem Graph2D dann in Zukunft erscheinen wird.

Es empfiehlt sich also auf jeden Fall, auch die Anleitung(en) des MUI-Systems - insbesondere der MUI-Preferences - gründlich zu lesen. Besonders möchte ich an dieser Stelle noch auf ein paar Eigenarten aller MUI-Applikationen hinweisen, die natürlich auch für Graph2D zutreffen:

Fenster von MUI-Applikationen sind in der Größe veränderbar und vollständig fontsensitiv, d.h. sehen mit jedem eingestellten Zeichensatz gut aus!

MUI-Applikationen lassen sich durch ein Extra-Gadget in der Fenster-Titelleiste jederzeit ikonifizieren, d.h. alle zum Programm gehörenden Fenster schließen sich sofort, und auf der Workbench erscheint das Programmicon, durch dessen Aktivierung man jederzeit die Applikation wieder zum Leben erwecken kann...

MUI-Applikationen sind auch vollständig über die Tastatur bedienbar. Durch Tab-Cycling und Shortcuts lassen sich alle Gadgets aktivieren und einstellen, ohne die Maus zur Hand nehmen zu müssen. Das Gadget, das gerade Tastatureingaben empfängt, ist durch Umrahmung o.ä. markiert. Fenster lassen sich mittels Tastatur schließen (normalerw. durch ESC).

MUI-Applikationen sind als Commodities ins System eingebunden und lassen sich deshalb mit dem CommodityExchange-Programm steuern.

Vieles mehr, das in der MUI-Anleitung ausführlich beschrieben ist...

## 1.33 MATHSCRIPT

### 3.4 MathScript

MathScript ist ein Formel-Editor von Simon Ihmig. Er steht unter dem Copyright seines Autors und ist Shareware. Es handelt sich dabei um ein eigenständiges Programm und liegt nicht Graph2D bei.

Mit MathScript ist es nun (unter anderem) möglich, den Funktionstext mathematischer Formeln in höchster Qualität im EPS-Format abzuspeichern, was von den gängigen Textverarbeitungsprogrammen importiert und weiterverwendet werden kann. Die mit MathScript erreichbaren Ergebnisse sind hervorragend und übersteigen die der "math. Notation" von Graph2D.

Aus diesem Grund liegt Graph2D ein ARexx-Skript bei, mit dem es ganz einfach möglich ist, die im Funktionseditor ausgewählte Funktion dem Programm MathScript zur weiteren Bearbeitung zu überstellen. Die Kommunikationsmöglichkeit mit MathScript wurde für die MathScript Versionen 2.0 und 2.1 entwickelt. Daß sie mit neueren Versionen funktioniert kann deshalb nicht garantiert werden, ist aber wahrscheinlich.

## 1.34 SIRDS

### 3.5 SIRDS

---

Graph2D verwendet zur 3D-Darstellung von Funktionen eine Technik, die üblicherweise als SIRDS (= Single Image Random Dot Stereogramm) bezeichnet wird. Man braucht keine Spezialbrille oder sonstige Hilfsmittel, um den 3D-Effekt zu erreichen, auch kann man diese Bilder ausdrucken oder kopieren, ohne daß sie an Wirkung verlieren.

Ich möchte an dieser Stelle nicht die Funktionsweise der Einzelbildstereogramme erläutern - wichtig erscheint mir für den Anwender von Graph2D nur das Erreichen des 3D-Eindrucks zu sein:

Das errechnete Bild besteht auf den ersten Blick nur aus einem wirren, mehr oder weniger farbigen Punkt-Muster. Man muß nun versuchen, den Blick ganz entspannt hinter die Bildebene zu lenken, z.B. auf die Wand hinter dem Monitor oder vielleicht auch auf das eigene Spiegelbild auf dem Bildröhrenglas. Natürlich sieht man dann die Bildpunkte, die gesamte Bildebene nicht mehr scharf, sie verschwimmt (auch sollte ein fester Punkt, z.B. der Mauszeiger doppelt erscheinen, doch das nur nebenbei). Beginnt dann bei richtigem Blick eine neue, dreidimensionale Ebene "im" Monitor zu erscheinen, darf man nicht den Fehler machen, sich sofort darauf zu konzentrieren, denn dann verschwindet sie sofort wieder. Vielmehr muß man sich bemühen, weiterhin die momentane Augeneinstellung beizubehalten. Erst mit der Zeit ist man in der Lage, im Bild etwas "herumzuschauen" oder auch einmal zu zwinkern...

Verzweifelt nicht, wenn sich der Effekt nicht so schnell einstellen will! Das erste Mal - ich weiß es aus eigener Erfahrung - kann sehr lange dauernd und recht frustrierend sein. Doch wenn man es einmal geschafft hat, lohnt es sich wirklich, und der Effekt wird von nun ab auch immer leichter und schneller erreichbar sein. Die Augen sind bei der beschriebenen Methode im Grunde sehr entspannt, es strengt also auch bei längerem Betrachten nicht übermäßig an. Es sollte aber auch erwähnt werden, daß man auch durch Schielen den beschriebenen Effekt erzielen kann. Davon rate ich allerdings ab, da dies in der Regel sehr anstrengend ist, und man bedingt durch die Technik die Bilder invertiert sieht, also "hoch" und "tief" vertauscht wird.

Ich möchte mich an dieser Stelle bei Kilian Singer bedanken, dessen kleine, aber wirkungsvolle Routine diese Bilder in Graph2D ermöglicht!

## 1.35 Index

### 3.6 Index

- 2D-Graph
- 2D-Graph Einsteller
- 3D-Darstellung
- 3D-Graph
- 3D-Graph-Einsteller

- A ARexx
- Autor
- Allgemeines

---

---

B bestimmte Integral

C Copyright

D Drucken

E EBNF-Syntax  
eingebunden

F Fragen & Antworten  
Funktionen  
Funktionseditor

G Gegenseitige Einbindung  
Geschichte  
Graph2D.RegForm

I internen Funktionen  
internen Operatoren  
Installation

K Konto  
Kurvendiskussion  
Kurzbeschreibung

M MathScript  
math. Notation  
MUI-System

N Notations-Fenster  
num. Integral  
Nutzungsbedingungen

O Originalarchiv

P Projekt-Menu

R registrieren  
Rexx

S SIRDS  
SIRDS-Graph  
SIRDS-Graph-Einsteller

T Taylorreihe  
Tangente  
Textanzeigefenster

V Voreinstellungen  
Voreinstellungseditor

W Wertetabelle

---