

Grace Benutzerhandbuch

für Grace-5.1.19

vom Grace Team
Deutsche Übersetzung durch Tobias Krähling
eMail: <webmaster@semibyte.de>
Homepage: <www.semibyte.de>

Original vom 19.05.2006
Übersetzung vom 17.05.2006
Version: 5.1.19v1.0.0de

Zusammenfassung

Dieses Dokument beschreibt die Verwendung von **Grace**, einem WYSIWYG 2D Plot-Programm für numerische Daten. Das englischsprachige Originalbenutzerhandbuch ist unter <http://plasma-gate.weizmann.ac.il/Grace/doc/UsersGuide.html> zu finden. Diese Übersetzung bezieht sich auf das o. g. englischsprachige Originalbenutzerhandbuch.

Inhaltsverzeichnis

1.	Einführung	1
1.1	Was ist Grace?	1
1.2	Copyright-Hinweis	1
2.	Installationsanweisung	2
2.1	Installation von den Quellen	2
2.2	Binäre Installation	3
2.3	Alternative Paketschemata (RPM,...)	3
3.	Erster Start	3
3.1	Generelles Konzept	4
3.1.1	Projektdateien	4
3.1.2	Parameterdateien	4
3.1.3	Eingangsdateiformate	4
3.1.4	Graphen	4
3.1.5	Datensätze	5
3.1.6	Sätze	5
3.1.7	Regionen	5
3.1.8	Echtzeiteingabe	6
3.1.9	Hotlinks	6
3.1.10	Geräte	6
3.1.11	Magic Pfade	7
3.1.12	Dynamische Module	7
3.1.13	Koordinatenrahmen	7
3.2	Aufruf	7
3.2.1	Betriebsweise	7
3.2.2	Kommandozeilenoptionen	8
3.3	Anpassung	9
3.3.1	Umgebungsvariablen	9
3.3.2	Init-Datei	9
3.3.3	Standardtemplate	9
3.3.4	X-Ressourcen	9
4.	Handbuch für das graphische Benutzerinterface (GUI)	10
4.1	GUI-Steuerung	10
4.1.1	Dateiauswahl-Dialog	10
4.1.2	Listenauswahl	11
4.1.3	Graphenauswahl	11
4.1.4	Satzauswahl	11
4.2	Das Hauptfenster	12
4.2.1	Die Zeichenfläche	12
4.3	Dateimenü (File)	13
4.3.1	Neu (New)	13

4.3.2	Öffnen (Open)	13
4.3.3	Speichern (Save)	14
4.3.4	Speichern unter (Save as)	14
4.3.5	Zurücksetzen auf letzte Speicherung (Revert to saved)	14
4.3.6	Druckeinstellungen (Print setup)	14
4.3.7	Drucken (Print)	14
4.3.8	Beenden (Exit)	14
4.4	Bearbeitenmenü (Edit)	14
4.4.1	Datensätze (Data Sets...)	14
4.4.2	Tabellendatensatz-Editor	15
4.4.3	Satzoperationen (Set Operations...)	15
4.4.4	Graphen ausrichten (Arrange graphs...)	15
4.4.5	Graphen überdecken (Overlay graphs...)	15
4.4.6	Autoskalierung (Autoscale graphs...)	16
4.4.7	Region-Menü (Regions)	16
4.4.8	Hotlinks (Hotlinks...)	16
4.4.9	Setze festen Punkt für Lokalisierung (Set locator fixed point)	16
4.4.10	Lösche festen Punkt für Lokalisierung (Clear locator fixed point)	17
4.4.11	Lokalisierungseinstellungen (Locator props)	17
4.4.12	Einstellungen (Preferences)	17
4.5	Datenmenü (Data)	17
4.5.1	Datensatzoperationen (Data set operations)	17
4.5.2	Transformationsmenü (Transformations)	17
4.5.3	Eigenschaftsermittlung (Feature extraction)	20
4.5.4	Import-Menü (Import)	21
4.5.5	Export-Menü (Export)	22
4.6	Plotmenü (Plot)	22
4.6.1	Plotaussehen (Plot appearance)	22
4.6.2	Graphaussehen (Graph appearance)	22
4.6.3	Satzaussehen (Set appearance)	22
4.6.4	Achsen-Einstellungen (Axis properties)	23
4.7	Ansichtmenü (View)	23
4.7.1	Anzeige der Lokalisierungsleiste (Show locator bar)	23
4.7.2	Anzeige der Statusleiste (Show status bar)	24
4.7.3	Anzeige der Toolleiste (Show tool bar)	24
4.7.4	Seiteneinstellungen (Page setup)	24
4.7.5	Neuzeichnen (Redraw)	24
4.7.6	Alles aktualisieren (Update all)	24
4.8	Fenstermenü	24
4.8.1	Kommandos (Commands)	24
4.8.2	Punktexplorer (Point explorer)	24
4.8.3	Objekte zeichnen (Drawing objects)	24

4.8.4	Schrifttool (Font tool)	24
4.8.5	Konsole (Console)	24
4.9	Hilfemenü (Help)	25
4.9.1	Kontexthilfe (On context)	25
4.9.2	Benutzerhandbuch (User's guide)	25
4.9.3	Tutorial (Tutorial)	25
4.9.4	FAQ (FAQ)	25
4.9.5	Änderungen (Changes)	25
4.9.6	Beispiele (Examples)	25
4.9.7	Kommentare (Comments)	25
4.9.8	Lizenzvereinbarung (License terms)	25
4.9.9	Über (About)	25
5.	Kommandointerpreter	25
5.1	Generelle Hinweise	25
5.2	Definitionen	26
5.3	Variablen	27
5.4	Numerische Operatoren und Funktionen	27
5.4.1	Operatoren	27
5.4.2	Funktionen	28
5.5	Prozeduren	30
5.6	Geräteparameter	32
5.7	Ablaufkontrolle	32
5.8	Deklarationen	33
5.9	Eigenschaften von Graphen	33
5.9.1	Kommandooperationen	33
5.9.2	Parametereinstellungen	34
5.10	Eigenschaften von Datensätzen	36
5.10.1	Kommandos	36
5.10.2	Parametereinstellungen	37
6.	Weiterführende Themen	37
6.1	Schriften	37
6.1.1	Schriftartenkonfiguration	37
6.1.2	Schriftarten-Datendateien	38
6.1.3	Benutzerspezifische Schriftarten	38
6.2	Interaktion mit anderen Anwendungen	39
6.2.1	Verwendung von Pipes	39
6.2.2	Verwendung der <code>grace_np</code> -Bibliothek	39
6.3	FFTW Einstellungen	42
6.4	DL-Module	42
6.4.1	Funktionstypen	43
6.4.2	Beispiele	43
6.4.3	Betriebssystem-Probleme	45

7. Referenzen	46
7.1 Schriftsatz	46
7.2 Gerätespezifische Einschränkungen	47
7.3 Gerätespezifische Einstellungen	48
7.3.1 Optionen des PostScript-Treibers	48
7.3.2 Optionen des EPS-Treibers	48
7.3.3 Optionen des PDF-Treibers	49
7.3.4 Optionen des PNM-Treibers	49
7.3.5 Optionen des JPEG-Treibers	49
7.3.6 Optionen des PNG-Treibers	49
7.4 Datum in Grace	50
7.5 Xmgr nach Grace Migrationshandbuch	51
Anhang	53
Tabellenverzeichnis	53

1. Einführung

1.1 Was ist Grace?

Grace ist ein WYSIWYG-Programm, um zweidimensionale Plots von numerischen Daten zu erstellen. Es läuft unter verschiedenen (wenn nicht allen) Arten von UNIX mit X11 und M*tif (LessTif oder Motif). Zusätzlich läuft es auch unter VMS, OS/2 und Windows (95/98/NT/2000/XP). Seine Fähigkeiten sind annähernd ähnlich wie bei GUI-basierten Programmen wie Sigmaplot oder Microcal Origin und scriptbasierten Tools wie GnuPlot oder Genplot.

Grace ist von Xmgr (a.k.a. ACE/gr) abgeleitet, im Original von Paul Turner geschrieben.

Seit der Versionsnummer 4.00 ist die Entwicklung auf ein Team von Freiwilligen übergegangen, das von Evgeny Stambulchik koordiniert wird. Neuesten Information über Grace und Download der letzten Version sind auf der [Grace Homepage](#) verfügbar.

Als das Copyright auf GPL geändert wurde, wurde auch der Name des Programms auf Grace geändert, das für »G Raphing, Advanced Computation and Exploration of data« oder »Grace Revamps ACE/gr« steht. Die erste Version von Grace trägt die Versionsnummer 5.0.0, da die letzte öffentliche Version von Xmgr die Versionsnummer 4.1.2 besaß.

Paul hält eine nicht-öffentliche Version von Xmgr für die interne Verwendung aufrecht, die er auch weiter entwickelt.

1.2 Copyright-Hinweis

Copyright (©) 1991-1995 Paul J Turner, Portland, OR
Copyright (©) 1996-2005 Grace Development Team

Maintained by Evgeny Stambulchik

All Rights Reserved

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Für bestimmte Bibliotheken, die für die Erstellung von Grace notwendig sind (diese werden in einer passenden Form eingebunden), können andere Copyright/Lizenzen bestehen.

Paket	Lizenz
cephes library	Free
T1lib	LGPL
Xbae	BSD-like
Tab Widget	BSD-like

Tabelle 1: Bibliothekslizenzen

2. Installationsanweisung

2.1 Installation von den Quellen

1. Konfiguration

- Anforderungen: Grace kompiliert normalerweise »out of the box« in einer regulären, UNIX-artigen Umgebung. Ein ANSI C Compiler (z. B. `gcc`), die Bibliotheken und Header-Dateien für X11R5 oder höher sowie eine Implementierung der M*tif API, Version 1.2 oder höher, werden benötigt. Wenn Sie eigene Änderungen zu verschiedenen Teilen von Grace kompilieren möchten, wird ein Parser-Generator (`yacc` oder besser `bison`) benötigt.
- Extra Bibliotheken: Einige Features sind nur verfügbar, wenn zusätzliche Bibliotheken installiert sind. Diese sind:
 - Das JPEG Backend benötigt die IJG's (JPEG-Bibliothek), Version 6.x.
 - Das PNG Backend benötigt die (libpng-Bibliothek), Version 0.96 oder höher.
 - Der PDF-Treiber benötigt die installierte PDFlib-Bibliothek von Thomas Merz in der Verion 4.0.3 oder höher, die [hier](#) verfügbar ist.
 - Wenn auf Ihrem Computer bei der Kompilierung von Grace die FFTW-Bibliothek installiert ist, wird Grace selbstständig diese Bibliothek verwenden und verwirft alle konventionelle FFT's und DFT's. Alle Transformationen werden über diese Bibliothek durchgeführt. Beachten Sie, das es dann keine Unterschiede zwischen der Betätigung des »FFT«-Button und dem »DFT«-Button mehr gibt, außer das FFT sich beschwert, wenn die Länge nicht eine Potenz von 2 ist und DFT nicht funktioniert. Für mehr Informationen zu diesem Paket sehen Sie bitte auf der [FFTW Homepage](#) nach. Kurzgesagt, dieses Paket erlaubt FFTs, die nicht eine Länge von einer Potenz von 2 haben, entlang der normalen. Es scheint sehr effizient für jede gesetzte Länge zu arbeiten, bei denen die Faktoren innerhalb 2^a , 3^b , 5^c , 7^d für die Integer a, b, c, d liegen. Das größte Feature hier ist, daß eine gesetzte Länge, die eine Potenz von 10 ist (z.B. 1000, 100000) und Integers Vielfaches von diesem sind (500, 2000, 2500, 5000, usw.), mit einer nicht-signifikaten Abweichung (vielleicht 20%) gegenüber einer 2ten-Potenz Transformationen berechnet werden können. Sehr oft kommen reale Datensätze in dieser Größe vor und nicht ein einer 2ten-Potenz.
 - Sollen Datensätze im NetCDF Datenformat geschrieben/gelesen werden, wird zusätzlich die [NetCDF-Bibliothek](#) benötigt.
- Entscheiden Sie, ob Sie an einem separaten Ort kompilieren wollen (dies läßt den Quellbaum unberührt). Dies ist häufig gewünscht, wenn Grace für mehrere Betriebssysteme kompiliert werden soll und die Quellen zentral Verfügbar gemacht werden (z.B. über ein NFS-Share). Wenn dies nicht gewünscht ist, kann der Rest dieses Paragraphen übersprungen werden und zu zum nächsten Schritt übergegangen werden. Andernfalls, in der Annahme das die Quellen in `/usr/local/src/grace-x.y.z` liegen und die Kompilierung in `/tmp/grace-obj` durchgeführt werden soll, sind die folgenden Schritte durchzuführen:


```
% mkdir /tmp/grace-obj
% cd /tmp/grace-obj
% /usr/local/src/grace-x.y.z/ac-tools/shtool mkshadow \
  /usr/local/src/grace-x.y.z .
```
- Das `configure` Shell-Script versucht, korrekte Werte für verschiedene systemabhängige Variablen zu ermitteln, die während der Kompilierung verwendet werden. Es verwendet diese Variablen, um die `Make.conf` im obersten Verzeichnis des Paketes zu erstellen. Es erzeugt ebenfalls die Datei `config.h`, die systemabhängige Definitionen enthält. Abschließend wird ein Shell-Script `config.status` erstellt, das zukünftig aufgerufen werden kann, um die aktuelle Konfiguration erneut zu erstellen; die Datei `config.cache` enthält Ergebnisse, die nicht behalten werden müssen – sie kann gelöscht oder bearbeitet werden.
- Mit `./configure -help` kann eine Liste von zusätzlichen Schaltern für Grace ermittelt werden.
- Aufruf von `./configure <options>`. Hier ein Beispiel:


```
% ./configure --enable-grace-home /opt/grace \
  --with-extra-incpath=/usr/local/include:/opt/include \
  --with-extra-ldpath=/usr/local/lib:/opt/lib --prefix=/usr
```

Dieses Beispiel verwendet `/usr/local/include` und `/opt/include` zusätzlich zum Standard-Include-Pfad und `/usr/local/lib` und `/opt/lib` zusätzlich zum Standard-LD-Pfad. Alle Dateien werden unter `/opt/grace` angelegt und Softlinks nach `/usr/bin`, `/usr/lib` und `/usr/include` angelegt.

Hinweis: Werden die Optionen `-with-extra-incpath` oder `-with-extra-ldpath` von einem `configure`-Lauf zu dem nächsten geändert, so muß die Datei `config.cache` gelöscht werden.!!

2. Kompilierung

- Aufruf von `make`

Wenn irgendetwas verkehrt laufen soll, schauen Sie in the **Grace FAQ** (im doc-Verzeichnis), ob dieses Problem bereits dort beschrieben wurde.

3. Testen

- `make tests`

Dies erzeugt eine Slide-Show, die einige interessante Features von Grace demonstriert.

4. Installation

- `make install`
- `make links`

Dieser optionale Schritt erstellt Softlinks von einigen Dateien im Grace-Verzeichnis zu den systemweiten Standardorten (dieser Ort kann über die `-prefix` Option während der Konfiguration eingestellt werden, siehe oben).

2.2 Binäre Installation

1. Download des vorgefertigten Paketes
2. Installation
3. Tests ausführen

2.3 Alternative Paketschemata (RPM,...)

Derzeit noch nicht geschrieben...

3. Erster Start

Für einen Einsprung in Grace können die Demos verwendet werden (über »Hilfe/Beispiel« (Help/Examples) im Menü). Dies sind richtige Grace-Projekte, so daß mit diesen experimentiert und die Änderungen gespeichert werden können. Zusätzlich kann das [Tutorial](#) weiterhelfen.

OK, hier folgt nun eine *sehr* schnelle Einführung:

1. Starten der GUI Version: `xmgrace` (return).
2. Auswahl/Überprüfung des Ausgabemediums und Zeichenblattgröße in *Datei/Druckeinstellungen* (File/Print Setup), S. 14.
3. Die Graphengröße wenn benötigt setzen (›Viewport‹ in *Plot/Graphaussehen* (Plot/Graph Appearance), S. 22).
4. Importieren der Daten über *Daten/Import/ASCII*, S. 21. ›Load as:‹ ›Single set‹ für zweispaltige ASCII-Daten, ›Block data‹ für mehrspaltige ASCII-Daten.
5. Anpassen der Skala, Achsenbeschriftung und Achsenmarkierungen in *Plot/Achsen-Einstellungen* (Plot/Axis properties), S. 23. Bestätigung aller Änderungen mit ›Apply‹.
6. Anpassen der Linie, Symbole, Legende in *Plot/Satzaussehen* (Plot/Set Appearance), S. 22.
7. Anpassen des Titels, Plotumrandung und Legende in *Plot/Graphaussehen* (Plot/Graph Appearance), S. 22.

8. Die Manipulation von Daten kann über *Daten/Transformation* (Data/Transformations), S. 17 durchgeführt werden. Um z. B. einen Datensatz um 20 nach links zu verschieben, kann in *Ausdruck auswerten* (Evaluate Expression), S. 17, der gleiche Datensatz auf der linken und rechten Seite ausgewählt werden und die Formel: $y=y-20$ eingetragen werden. Wie Sie sicherlich errahnen können, kann Grace noch sehr viel mehr. Erforschen Sie es, wenn Sie die Zeit dazu finden.
9. Wenn der Graph ausgedruckt werden soll, wählen Sie *Datei/Drucken* (File/Print), S. 14, aus. Das ist alles!

3.1 Generelles Konzept

3.1.1 Projektdateien

Eine Projektdatei enthält alle Informationen, um Plots, die von Grace erstellt wurden, wieder herzustellen und diese zu modifizieren. Jeder Plot wird auf einer eigenen Seite dargestellt, jedoch kann dieser eine unlimitierte Anzahl von Graphen enthalten. Eine Projektdatei des aktuellen Graphen kann über *Datei/Speichern* (File/Save), S. 14, bzw. *Datei/Speichern unter* (File/Save as), S. 14, erstellt werden.

3.1.2 Parameterdateien

Eine Parameterdatei enthält die ausführlichen Einstellungen Ihres Projektes. Diese kann verwendet werden, um Einstellungen zu einem anderen Plot/Projekt zu übertragen. Sie können eine Parameterdatei erzeugen, indem der *Datei/Speichern* (File/Save) Menüeintrag im *Plot/Graphaussehen* (Plot/Graph appearance) Dialog, S. 22, ausgewählt wird. Diese Parameterdatei kann dann über *Datei/Öffnen* (File/Open) eingelesen werden.

3.1.3 Eingangsdateiformate

Grace versteht verschiedene Eingangsdateiformate. Der grundlegendste ist die ASCII Textdatei, die die Daten als Leerzeichen und kommaseparierte Spalten enthält. Die Datenfelder können in numerischer Form (Fortran $\rangle d \langle$ und $\rangle D \langle$ Exponenten werden unterstützt) oder in alphanumerischer Form (mit oder ohne Quotierung) vorliegen. Verschiedene Kalender-Datum-Formate werden automatisch erkannt und Sie können Ihre eigenen Datumsformate festlegen. Zeilen, die mit $\rangle \# \langle$ beginnen werden als Kommentar gewertet und ignoriert. Leere Zeilen zeigen einen neuen Datensatz an. Grace besitzt auch eine Befehlssprache (siehe *Kommandointerpreter*, S. 25, Sie können ebenfalls Befehle für den Kommandointerpreter in die Eingangsdatei schreiben. Dies wird durch ein $\rangle @ \langle$ als erstes Zeichen, das nicht ein Leerzeichen ist (Space, Tab. . .), obwohl dieses nicht empfohlen wird. Abhängig von der Konfiguration kann Grace auch NetCDF Dateien einlesen (siehe *Konfiguration*, S. 2).

3.1.4 Graphen

Ein Graph besteht aus den optionalen Elementen: ein Graph-Rahmen, Achsen, einem Titel und einem Untertitel, mehreren Sätzen und weiteren Objekten (Zeitmarken, Texten, Linien, Kästen und Ellipsen).

Der Graph-Typ kann einer der folgenden Typen haben:

- XY-Graph
- XY-Chart
- Graph in Polardarstellung
- fester Graph
- Tortendiagramm

Die Idee des XY-Chart ist, mehrer Datensätze als Balken (oder im Allgemeinen Symbole) nebeneinander anzuordnen, in der Annahme, die Abszissen aller Datensätze sind gleich (oder Teilmengen des längsten Datensatzes).

3.1.5 Datensätze

Ein Datensatz ist eine Sammlung von Punkten mit x- und y-Koordinaten, die bis zu 4 optionale Datenwerte enthalten können (abhängig vom Datensatztyp können diese u.a. als Fehlerbalken dargestellt werden) und einer optionalen Zeichenkette.

3.1.6 Sätze

Ein Satz ist ein Darstellungsweg eines Datensatzes. Es besteht aus einem Zeiger auf einen Datensatz sowie eine Sammlung von Parametern, die die Darstellung der Daten steuern (wie Farbe, Linienart, usw.).

Der Satztyp kann einer der in Tabelle 2 aufgelisteten sein.

Satztyp	Anzahl der numerischen Zeilen	Beschreibung
XY	2	Ein X-Y Punkt- und/oder Liniendiagramm, mit (optional) Wertkennzeichnung
XYDX	3	Wie XY, aber mit Fehlerbalken (sowohl ein- wie auch zweiseitig) an der X-Achse
XYDY	3	Wie XYDX, aber mit Fehlerbalken an der Y-Achse
XYDXDX	4	Wie XYDX, aber linker und rechter Fehlerbalken sind einzeln definiert
XYDYDY	4	Wie XYDXDX, aber Fehlerbalken entlang der Y-Achse
XYDXDY	4	Wie XY, aber mit X- und Y-Fehlerbalken (sowohl ein- wie auch zweiseitig)
XYDXDXDYDY	6	Wie XYDXDY, aber linker/rechter sowie oberer/unterer Fehlerbalken sind einzeln definiert
BAR	2	Wie XY, jedoch werden vertikale Balken anstelle von Symbolen verwendet
BARDY	3	Wie BAR, aber mit Fehlerbalken (sowohl ein- wie auch zweiseitig) an der Y-Achse
BARDYDY	4	Wie BARDY, aber oberer/unterer Fehlerbalken sind einzeln definiert
XYHILO	5	Oberer/Unterer/Offen/Geschlossen-Diagramm
XYZ	3	Wie XY; macht keinen Sinn, wenn die Wertkennzeichnung nicht Z ist
XYR	3	X, Y, Radius. Nur in festen Graphen erlaubt
XYSIZE	3	Wie XY, jedoch ist die Symbolgröße variabel
XYCOLOR	3	X, Y, Farbindex (der Symbolfüllung)
XYCOLPAT	4	X, Y, Farbindex, Füllungsindex (derzeit nur in Tortendiagramme verwendet)
XYVMAP	4	Vektorgraph
XYBOXPLOT	6	Kastengraph (X, Mittellinie, oberes/unteres Limit, obere/untere Linie)

Tabelle 2: Liste der verfügbaren Satztypen

Nicht alle Satztypen können durch alle Graphentypen dargestellt werden. Eine Übersicht hierüber ist in Tabelle 3 zu finden.

3.1.7 Regionen

Regionen sind Bereiche des Graphen, die über ein inneres oder äußeres Vieleck (Polygon) oder einer halben Ebene, definiert über eine Linie, beschrieben werden. Diese Regionen werden bei Datentransformationen verwendet, um Manipulationen nur in der definierten Region durchführen zu können.

Satztyp	XY-Graph	XY-Chart	fest	Polar- diagramm	Torten- diagramm
XY	+	+	+	+	+
XYDX	+	-	+	-	-
XYDY	+	+	+	-	-
XYDXDX	+	-	+	-	-
XYDYDY	+	+	+	-	-
XYDXDY	+	-	+	-	-
XYDXDXDYDY	+	-	+	-	-
BAR	+	+	+	-	-
BARDY	+	+	-	-	-
BARDYDY	+	+	-	-	-
XYHILO	+	-	-	-	-
XYZ	+	-	+	+	-
XYR	-	-	+	-	-
XYSIZE	+	+	+	+	-
XYCOLOR	+	+	+	+	+
XYCOLPAT	-	-	-	-	+
XYVMAP	+	-	+	-	-
XYBOXPLOT	+	-	-	-	-

Tabelle 3: Übersicht, welche Satztypen durch welchen Graphyp dargestellt werden können.

3.1.8 Echtzeiteingabe

Mit der Echtzeiteingabe besteht in Grace die Möglichkeit, Daten in Echtzeit von einem externen Programm zur Verfügung zu stellen und diese durch Grace verarbeiten zu lassen. Der Grace-Prozeß, der vom Treiberprogramm erzeugt wird, ist ein vollständiger Grace-Prozeß: der Benutzer kann über die GUI mit diesem Grace-Prozeß arbeiten, während zur gleichen Zeit das Programm Daten und Kommandos an Grace sendet. Eine Anpassung an die ankommende Datenrate seitens Grace erfolgt automatisch.

3.1.9 Hotlinks

Hotlinks sind Quellen, die unterschiedliche Daten enthalten. Grace kann angewiesen werden, eine Datei oder eine Pipe als Hotlink zu betrachten, wobei in diesem Fall spezifische Befehle zur Verfügung gestellt werden, um die Daten über einen Maus-Klick zu aktualisieren (eine spätere Version wird wahrscheinlich die automatische Aktualisierung erlauben).

3.1.10 Geräte

Grace ermöglicht dem Anwender, zwischen verschiedenen Ausgabegeräten für die Darstellung der Graphen zu wählen. Die derzeitige Liste von unterstützten Geräten ist:

- X11
- PostScript (Level 1 und Level 2)
- EPS (Encapsulated PostScript)
- Metafile (dies ist ein Grace-Format und wird momentan meistens für Debuggingzwecke verwendet)
- MIF (Make Interchange Format, verwendet von FrameMaker)
- SVG (Scalable Vector Graphics, eine Sprache für die Beschreibung zweidimensionaler Vektoren und gemischten Vektor/Raster Graphiken in XML)

- PDF (abhängig von externen Bibliotheken, siehe *Konfiguration*, S. 2
- PNM (Portable anymap Dateiformat)
- JPEG (abhängig von externen Bibliotheken, siehe *Konfiguration*, S. 2
- PNG (abhängig von externen Bibliotheken, siehe *Konfiguration*, S. 2

Beachten Sie, daß GIF aufgrund der Urheberrechtspolitik von Unisys nicht länger mehr unterstützt wird. Grace kann zusätzlich externe Umwandlungsprogramme in Abhängigkeit des Dateinames aufrufen. Als ein Beispiel können MIF (FrameMaker Interchange Format) oder Java-Applets über `ps-to-edi` erzeugt werden. Ebenfalls kann fast jedes Bildformat erzeugt werden, welche von der *netpbm*-Suite unterstützt wird (siehe auch [FAQ](#)).

3.1.11 Magic Pfade

Muß Grace Zugriff auf eine Datei erhalten, die mit relativen Pfadnamen angegeben wurde, so sucht es in vielen Fällen nach dieser Datei in den folgenden Pfaden:

```
./pathname:./.grace/pathname:~/grace/pathname:$GRACE_HOME/pathname
```

3.1.12 Dynamische Module

Grace kann auf externe Funktionen zugreifen, die in Systembibliotheken oder in Bibliotheken von Drittanbietern vorhanden sind (Shared Libraries). Ebenfalls besteht die Möglichkeit, Funktionen aus Modulen zu verwenden, die speziell für Grace kompiliert wurden. Der Ausdruck mit der zusätzlichen Funktionalität referenziert dynamisch auf die Möglichkeit von Grace, Bibliotheken zur Laufzeit zu öffnen und den Code der externen Funktion hierdrin zu finden. Hierzu muß Grace nicht neu kompiliert werden (die Funktionen, die in Grace bereits verfügbar sind, sind statisch eingebunden).

3.1.13 Koordinatenrahmen

Innerhalb Grace gibt es zwei Typen von Koordinatensystemen: die **globalen Koordinaten** (world coordinates) und die **Viewport-Koordinaten**. Punkte von Datensätzen werden in den globalen Koordinaten definiert. Die Viewport-Koordinaten entsprechen dem Bild des Plots, das auf der Zeichenfläche gezeichnet ist (oder gedruckt auf, sagen wir, der PostScript Ausgabeseite). Die Transformation, die die globalen Koordinaten in die Viewport-Koordinaten umwandelt, wird sowohl durch den Graph-Typ wie auch die Achsenskalierung beeinflusst.

In Wirklichkeit gibt es noch ein weiteres, zusätzliches Niveau in der Koordinatenhierarchie – die **Geräte-Koordinaten**. Jedoch brauchen Sie sich (als ein Anwender von Grace) über diese keine Gedanken zu machen. Die Umwandlung der Viewport-Koordinaten in die Geräte-Koordinaten wird immer in der Weise durchgeführt, daß der Ursprung des Viewport der linken, untersten Ecke der Geräteseite entspricht. Die kleinste Dimension des Gerätes entspricht dabei einer Einheit der Viewport-Koordinaten. Als wichtigste Eigenschaft der Transformation zwischen Viewport und Gerät ist die Homogenität, z. B. ist garantiert, daß ein Quadrat ein Quadrat bleibt und nicht zu einem Rechteck wird, ebenso bleibt ein Kreis ein Kreis (und wird nicht zu einer Ellipse) usw.

3.2 Aufruf

3.2.1 Betriebsweise

In Bezug auf die Benutzerschnittstelle gibt es drei Verfahrensweisen, in denen Grace aufgerufen werden kann. Die vollständige GUI-basierende Version wird `xmgrace` genannt. Eine Version, mit der man im Batchbetrieb drucken kann, wird `gracebat` genannt. Die Kommandozeilenversion wird `grace` genannt. Gewöhnlich wird eine einzelne ausführbare Datei in verschiedenen Weisen aufgerufen, wobei zwei oder drei Dateien (symbolische) Links sind und auf die »echte« verweisen.

3.2.2 Kommandozeilenoptionen

- autoscale <x/y/xy> Überschreibt die Einstellungen in der Parameterdatei.
- barebones Schaltet alle Toolbars aus.
- batch <batch_file> Beim Starten wird die angegebene Batchdatei ausgeführt.
- block <block_data> Nimmt an, daß die Datendatei Block-Daten enthält.
- bxy <x:y:etc.> Bildet einen Satz aus dem gegenwärtigen Block-Datensatz, der den aktuellen Satz-Typ für Spalten verwendet, der als Argument angegeben wurde.
- datehint <iso|european|us|days|seconds|nohint> Setzt einen Hinweis für die Datumsanalyse.
- dpipe <descriptor> Einlesen von Daten bei Starten über den angegebenen Descriptor (anonyme Pipe).
- fixed <width> <height> Setzt Zeichenblattgröße fest auf <width>*<height>.
- free Verwendet freies Seitenlayout.
- graph <graph_number> Setzt die aktuelle Graphnummer.
- graphtype <graph_type> Setzt den Type des aktuellen Graphen.
- hardcopy Keine interaktive Sitzung, nur Ausdruck und anschließend Grace beenden.
- hdevice <hardcopy_device_name> Setzt das standardmäßige Ausgabegerät für eine Hardcopy.
- install Installation einer privaten Farbpalette.
- legend <load> Schaltet die Graphenlegende an.
- log <x|y|xy> Setzt die Achsenskalierung des aktuellen Graphen auf logarithmisch.
- mono Läßt Grace im Monochrome-Modus laufen (bezieht sich nur auf die Darstellung).
- netcdf <file> Nimmt an, daß <file> im netCDF-Format vorliegt. Diese Option ist nur Verfügbar, wenn netCDF-Unterstützung in Grace einkompiliert wurde.
- netcdfxy <X_var> <Y_var> Wenn -netcdf vorher verwendet wurde, einlesen der *X_var* und *Y_var* Variablen aus der netCDF-Datei und erzeugt hieraus einen Satz. Ist der *X_var*-Name »null«, dann wird der Index von Y nach X eingelesen. Diese Option ist nur Verfügbar, wenn netCDF-Unterstützung in Grace einkompiliert wurde.
- noask Nimmt an, das auf alle Anfragen mit *Ja* geantwortet wurde – wenn die Operation eine Datei überschreiben würde, Grace würde dies bei der Verwendung dieser Option ohne jegliche Rückfrage durchführen.
- noinstall Keine private Farbpalette verwenden.
- noprint Im Batch-Modus wird nicht ausgedruckt.
- nosafe Sicherer Modus ausschalten.
- nosigcatch Keine Signale einfangen.
- npipe <file> Einlesen der Daten beim Starten über eine benannte Pipe.
- nxy <nxy_file> Nimmt an, daß die angegebene Datendatei im *X Y1 Y2 Y3...-Format* vorliegt.
- param <parameter_file> Parameter werden von der angegebenen Parameterdatei *parameter_file* eingelesen und dem aktuellen Graphen zugeordnet.
- pipe Einlesen der Daten beim Starten über die Standardeingabe (stdin).
- printfile <file> Sichert Druckausgabe in *file*.
- remove Entfernt die Datendatei nach dem Einlesen.
- result <results_file> Ergebnisse von einigen Datenmanipulationen werden in *results_file* geschrieben.
- rvideo Tascht die Farbindizes für schwarz und weiß aus.
- save Ausführung im sicheren Modus (Standard) - keine Dateisystemmodifikationen sind durch die Batchsprache erlaubt.
- saveall <save_file> Sichere alle Graphen in *save_file*.
- seed <seed_value> »Ganzhalsamen« für den Zufallszahlengenerator.
- source <disk|pipe> Quellentyp der nächsten Datendatei.

- timer** <delay> Setzte die erlaubte Zeitscheibe für Echtzeiteingaben auf *delay* ms.
- timestamp** Zeitstempel zum Plot hinzugefügen.
- settype** <xy|xydx|...> Setzte den Typ der nächsten Datendatei.
- version** Anzeige der Programmversion.
- viewport** <xmin> <ymin> <xmax> <ymax> Setzen des Viewports für den aktuellen Graphen.
- wd** <directory> Setzen des aktuellen Arbeitsverzeichnisses auf *directory*.
- world** <xmin> <ymin> <xmax> <ymax> Setzt die globalen Koordinaten für den aktuellen Graphen.
- usage|-help** Ausgabe der Kommandozeilenhilfe.

3.3 Anpassung

3.3.1 Umgebungsvariablen

- **GRACE_HOME**
Setzt die Position von Grace. In dieser Position sind die Hilfedateien, Hilfsprogramme und Beispiele enthalten. Wenn Sie dieses Verzeichnis nicht finden können, setzen Sie sich mit Ihrem Systemverwalter in Verbindung.
- **GRACE_PRINT_CMD**
Das Druckcommando. Wenn diese Variable definiert worden ist, jedoch einen leeren String enthält, wird der Ausdruck in eine Datei (»Print to file«) standardmäßig ausgewählt.
- **GRACE_EDITOR**
Der Editor, der für manuelles Bearbeiten von Datensatzwerten verwendet wird.
- **GRACE_HELPVIEWER**
Das Shell-Kommando, um einen HTML-Betrachter für die Online-Hilfe von Grace zu starten. Innerhalb Kommandos muß mindestes einmal »%s« vorkommen, wobei Grace diesen Platzhalter durch die aktuelle URL ersetzt.
- **GRACE_FFTW_WISDOM_FILE** und **GRACE_FFTW_RAM_WISDOM**
Diese Flags kontrollieren Eigenschaften des FFTW-Planers (siehe FFTW Tuning, S. 42, für ausführliche Informationen).

3.3.2 Init-Datei

Während des Starten lädt Grace seine Init-Datei, *gracerc*. Diese Datei wird in den Magic-Pfaden (siehe Magic Pfade, S. 7) gesucht; wird eine gefunden, so wird der Rest dieses Pfades ignoriert. Es wird empfohlen, daß in der *gracerc*-Datei keine Einstellungen verwendet werden, die Teil einer Projektdatei sind – solche Standardwerte sollten im Standardtemplate (siehe Standardtemplate, S. 9) definiert werden.

3.3.3 Standardtemplate

Wird ein neues Projekt angefangen, so lädt Grace das Standardtemplate, *templates/Default.agr*. Diese Datei wird ebenfalls über die Magic-Pfade (siehe Magic Pfade, S. 7) gesucht; wird eine gefunden, so wird der Rest dieses Pfades ignoriert. Es wird empfohlen, daß in dem Standardtemplate keine Einstellungen vorgenommen werden, die nicht ein Teil der Projektdatei sind – solche Standardwerte sollten in der Init-Datei *gracerc* vorgenommen werden (siehe Init-Datei, S. 9).

3.3.4 X-Ressourcen

Die folgenden Grace-spezifische X-Ressourcen Einstellungen werden unterstützt:

- **XMgrace.invertDraw**
Verwendet *GXinvert* anstellen von *GXxor* für dynamisch angepaßte Linien. Erscheinen diese dynamisch angepaßte Linien nicht bei Zooms und Linien etc. nicht auf der Zeichenfläche, sollte diese Resource auf *yes* gesetzt werden.

- `XMgrace.allowDoubleClick`
Wenn auf `yes` gesetzt, werden bei einem Doppelklick verschiedene Popups in Abhängigkeit von der Position des Mauszeigers angezeigt.
- `XMgrace.toolBar`
Aktiviert die Button Tooleiste.
- `XMgrace.statusBar`
Aktiviert die Statusleiste.
- `XMgrace.locatorBar`
Aktiviert die Lokalisierungsleiste.

Es ist ebenso möglich, das Menü durch Zuweisung von Tastaturkürzel (Shortcuts) für jeden Eintrag benutzerspezifisch einzustellen.

Sie müssen hierzu den X Resource-Namen des Menüeintrages vom entsprechenden Menünamen ableiten müssen, wobei der X Resource-Name über die folgenden Regeln ermittelt werden kann:

- Alle nicht-alphanumerische Zeichen werden übersprungen;
- Begonnen wird mit Kleinbuchstaben, jedes neue Wort (wenn es eins gibt) folgt mit einem großen Anfangsbuchstaben;
- Der Eintragstyp wird am Ende angehängt - »Menu« für Pulldown-Menüs, »Button« für Menübuttons.

Beispiel: Der Grace Dialog für nichtlineare Kurvenanpassung (non-linear curve fitting) soll über CTRL-F erreichbar sein. Hierfür müssen die folgenden zwei Zeilen:

```
XMgrace*transformationsMenu.nonLinearCurveFittingButton.acceleratorText: Ctrl+F
XMgrace*transformationsMenu.nonLinearCurveFittingButton.accelerator: CTRL<Key>f
```

in Ihre `.Xresources`-Datei eingetragen werden (die Datei, die beim Starten der X-Session eingelesen wird; dies kann `.Xdefaults`, `.Xsession` oder irgendeine andere sein – fragen Sie Ihren Systemadministrator, wenn Sie unsicher sind).

In ähnlicher Weise kann es wünschenswert sein, die standardmäßigen Dateinamensmuster von Dateiauswahldialogen anzupassen. Das Verfahren, um den Dialognamen zu erhalten, ist wie bei den Menüeinträgen, was oben beschrieben wurde. Dabei wird »Button« durch »FSB« ersetzt. Sollen z.B. im Dialog »Projekt öffnen« ("File/Open...") alle Dateien standardmäßig angezeigt werden, so muß die Resource auf:

```
XMgrace*openProjectFSB.pattern: *
```

gesetzt werden.

4. Handbuch für das graphische Benutzerinterface (GUI)

4.1 GUI-Steuerung

Dieser Abschnitt beschreibt die Schnittstellen-Steuerung – grundlegende Bausteine, die in vielen Popups verwendet werden.

4.1.1 Dateiauswahl-Dialog

Wird von dem Anwender ein Dateiname erwartet, sei es zum Lesen oder zum Schreiben von Daten, so wird dieser durch einen Dateiauswahl-Dialog unterstützt. Zusätzlich zu den Standardeinträgen (die Verzeichnis- und Dateilisten sowie den Filtern) gibt es ein Pulldown-Menü, um schnell ein Wechsel in Verzeichnisse mit vordefinierten Positionen durchzuführen (das aktuelle Arbeitsverzeichnis, das Benutzerheimatverzeichnis sowie zur Dateisystemwurzel). Außerdem ist ein Button »Setze als aktuelles Arbeitsverzeichnis« (Set as cwd) vorhanden, um dem Anwender die Möglichkeit zu geben, jedes Verzeichnis, durch das man im Dateisystem navigiert, als aktuelles Arbeitsverzeichnis (cwd) zu setzen. Einmal definiert, kann es in allen anderen Dateiauswahl-Dialogen verwendet werden, um schnell in dieses Verzeichnis zu wechseln.

4.1.2 Listenauswahl

Verschiedene Auswahlen sind in verschiedenen Popups verfügbar. Sie alle stellen eine Liste von Objekten (Graphen, Sätze. . .) dar und können dazu verwendet werden, einfache Operationen auf diese Objekte (kopieren, löschen. . .) durchzuführen. Diese Operationen sind über ein Popup-Menü zugänglich, das über die dritte Maustaste aufgerufen werden kann. Abhängig von der erforderlichen Funktionalität, können mehrere Möglichkeiten bestehen oder auch nicht. Die folgenden Tastaturkürzel (Shortcuts) sind verfügbar (wenn das Ergebnis einer Aktionen der Listenauswahlregel widersprechen würde, wird dieses ignoriert):

- CTRL-a Alle auswählen
- CTRL-u Alle abwählen
- CTRL-i Auswahl umkehren (invertieren)

4.1.3 Graphenauswahl

Die Operationen, die auf Graphen über das Graphenauswahl-Popupmenü ausgeführt werden können, sind:

- setze Fokus
- verstecken
- anzeigen
- duplizieren
- löschen
- vertauschen
- neu erstellen

Nicht in jeder Instanz der Auswahl sind alle Operationen verfügbar. Zum Beispiel kann im »Lese Satz« (read sets) Popup nur ein Graph zu einer Zeit ausgewählt werden, und die Vertauschen-Operation ist deaktiviert.

Ein Doppelklick auf einen Listeneintrag wechselt den Fokus auf diesen Graphen.

4.1.4 Satzauswahl

Die Operationen, die auf Sätze über das Satzauswahl-Popupmenü ausgeführt werden können, sind:

- verstecken
- anzeigen
- nach vorne bringen
- nach hinten verschieben
- duplizieren
- löschen
- Daten löschen
- vertauschen
- bearbeiten
- in Tabelle (siehe Tabellendatensatz-Editor, S. 15)
- im Text-Editor
- neu erstellen
- über Formel
- in Tabelle (siehe Tabellendatensatz-Editor, S. 15)
- im Text-Editor

- aus Blockdaten
- Packen aller Sätze
- Auswahl-Operationen
- Anzeige des Satz-Kommentars
- Anzeige ohne Daten
- Zeige versteckte
- alle auswählen
- alle abwählen
- Auswahl umkehren (invertieren)
- aktualisieren

Ein Doppel-Klick auf einen Listeneintrag öffnet den Tabellen-Editor mit den Daten des Satzes (siehe Tabellendatensatz-Editor, S. 15).

4.2 Das Hauptfenster

4.2.1 Die Zeichenfläche

Zeichenfläche-Hotkeys

Ist der Zeigerfokus auf der Zeichenfläche (wo der Graph gezeichnet wird), so können verschiedene Aktionen über Tastaturkürzel (shortcuts) aktiviert werden. Diese sind:

- CTRL-A: Aktuellen Graphen autoskalieren (**autoscale**);
- CTRL-D: Löschen eines Objektes (**delete**);
- CTRL-L: Verschieben der aktuellen Graphenlegende (**move legend**);
- CTRL-M: Objekt verschieben (**move**);
- CTRL-T: Zeitstempel plazieren (**timestamp**);
- CTRL-U: Aktualisierung der Hotlinks (**update**);
- CTRL-V: Viewport mit der Maus setzen (**viewport**);
- CTRL-Z: Zoom (**zoom**);
- CTRL-ALT-L: Linie zeichnen (**line**);
- CTRL-ALT-B: Box zeichnen (**box**);
- CTRL-ALT-E: Elipse zeichnen (**ellipse**);
- CTRL-ALT-T: Zeichenkette schreiben (**text**).

Klicks und Doppelklicks

Ein einfacher Klick innerhalb eines Graphen wechselt den Fokus auf diesen Graphen. Dieses Standardverhalten kann jedoch über das Einstellungsmenü (»Edit/Preferences«) geändert werden.

Ein Doppelklick auf einen Teil der Zeichenfläche werden bestimmte Aktionen aufrufen oder ein Popup wird angezeigt:

- auf eine Fokus-Makierung: verschieben der markierten Viewport-Ecke;
- auf eine Achse: Achseneinstellungsdialog wird angezeigt (»Plot/Axis properties«);
- auf einen Satz: Satzeinstellungsdialog wird angezeigt (»Plot/Set appearance«);
- auf eine Legende: Grapheinstellungsdialog wird angezeigt (»Plot/Graph appearance«);
- auf einen (Unter-)Titel: Grapheinstellungsdialog wird angezeigt (»Plot/Graph appearance«);
- auf ein Objekt (Box, Linie. . .): Einstellungsdialog für dieses Objekt wird angezeigt.

Die Doppelklick-Aktionen können über den Einstellungsdialog (»Edit/Preferences«) aktiviert bzw. deaktiviert werden.

Werkzengleisten-Buttons

Entlang der linken Seite der Zeichenfläche befindet sich die Werkzengleiste (wenn sie angezeigt wird). Sie enthält mehrere Buttons, um schnellen und leichten Zugang zu den eher allgemeinen Grace-Funktionen zu erhalten.

- **Draw:** Zeichnet die Zeichenfläche und Sätze neu. Hilfreich wenn »Automatisches Neuzeichnen« (»Auto Redraw«) im Einstellungsdialog (»Edit/Preferences«) deaktiviert wurde oder nach der Ausführung von Kommandos direkt über den Kommandozeileninterpreter (»Window/Commands«).
- **Lens:** Eine Lupe. Wird die Lupe aktiviert (angeklickt), so kann über die Auswahl eines interessanten Bereiches innerhalb des Graphen dieser vergrößert werden. Dieser Bereich füllt dann den gesamten Graphen aus.
- **AS:** Autoskalierung. Skaliert den Graphen automatisch, so daß alle Datenpunkte der sichtbaren Sätze im Graphen enthalten sind. Die als versteckt-markierten Sätze werden dabei ignoriert.
- **Z/z:** Hinaus/Hinein-Zoom mit einem Zoom-Faktor von 5%. Der Zoom-Faktor kann über den Einstellungsdialog (»Edit/Preferences«) eingestellt werden.
- **Arrows:** Verschiebt den aktiven Graphen um einen Verschiebungsfaktor von 5% in Pfeilrichtung. Der Verschiebungsfaktor kann über den Einstellungsdialog (»Edit/Preferences«) eingestellt werden.
- **AutoT:** Achsen mit automatischen Markierungen. Dies findet die optimale Anzahl an primären und sekundären Marken für beide Achsen.
- **Auto0:** Aktiviert die automatische Skalierung. Nach dem anklicken des Auto0-Buttons können Sie innerhalb des Graphen in die Nähe des Satzes klicken, der für die automatische Skalierungsgrenzen des Graphen verwendet werden soll.
- **ZX, ZY:** Zoom entlang einer Achse. Diese Buttons arbeiten wie die Lupe (s. o.), aber beschränken sich nur auf eine Achse.
- **AX, AY:** Autoskalierung nur auf einer Achse. Diese Buttons arbeiten wie Autoskalierung (AS), aber beschränken sich nur auf eine Achse.

Die nachfolgenden Buttons arbeiten mit den Graph-Stack, ein gutes Beispiel hierfür ist unter »Help/Examples/General Intro/World Stack« zu finden.

- **Pu/Po:** Globale Einstellung auf den Stack des Graphen aufsetzen bzw. entnehmen. Wenn diese entnommen werde, wird der neue oberste Stack der aktuelle.
- **PZ:** Aufsetzen vor dem zoomen. Funktioniert wie die Lupe, nur daß die aktuellen globalen Einstellungen vorher auf den Stack gelegt werden.
- **Cy:** Zirkuliert über die Stack-Einstellungen des aktuellen Graphen. Jeder Graph kann bis zu 20 Ebenen auf dem Stack besitzen.
- **Exit:** Ziemlich offensichtlich, oder? Beendet XmGrace.

4.3 Dateimenü (File)

Das Dateimenü enthält alle Einträge, die sich auf die Eingabe/Ausgabe-Fähigkeiten von Grace beziehen.

4.3.1 Neu (New)

Setzt den Status von Grace zurück, als wenn es gerade gestartet wäre (einen leeren Graphen mit dem Bereich 0 bis 1 auf beiden Achsen). Wenn bereits Arbeiten mit Grace verrichtet wurden, die noch nicht gespeichert worden sind, wird ein Dialog angezeigt, das die Möglichkeit bietet, um die Operation abzubrechen und zu der aktuellen Einstellung zurückzukehren.

4.3.2 Öffnen (Open)

Öffnet ein bestehendes Projekt (siehe Projektdateien, S. 4). Ein Popup-Dialog wird angezeigt, um durch das Dateisystem zu wandern und ein Projekt auszuwählen.

4.3.3 Speichern (Save)

Speichert die aktuelle Arbeit in einer Projektdateien, siehe S. 4. Der verwendete Name entspricht dem letzten, der für öffnen oder speichern verwendet wurde. Wenn noch kein Name gesetzt wurde (z. B. da das Projekt neu ist), verhält sich dieser Menüpunkt wie Speichern unter, siehe S. 14.

4.3.4 Speichern unter (Save as)

Speichert die aktuelle Arbeit in einer Projektdatei, siehe S. 4, mit einem neuen Namen. Ein Pop-up-Dialog gibt die Möglichkeit, im Dateisystem zu wandern und einen Namen anzugeben, das Format für die Speicherung von Datenpunkten zu spezifizieren (der Standardwert ist »%16.8g«) sowie eine Beschreibung des Projektes anzugeben. Existiert bereits eine Datei mit gleichem Namen, so wird eine Warnung angezeigt.

4.3.5 Zurücksetzen auf letzte Speicherung (Revert to saved)

Ignoriert alle Änderungen, die seit dem letzten Speichern an der Projekt-Datei vorgenommen wurden und setzt die aktuelle Arbeitsversion auf die zuletzt gespeicherte. Ein Bestätigungsdialog wird angezeigt, um den Benutzer die Möglichkeit zu geben, diese Operation abzubrechen.

4.3.6 Druckeinstellungen (Print setup)

Setzt die Einstellungen für das Druckgerät. Jedes Gerät hat seinen eigenen Satz von spezifischen Optionen (siehe auch Gerätespezifische Einstellungen, S. 48). Entsprechend dem Gerät kann die Ausgabe direkt zu einem Drucker geschickt werden oder in einer Datei gespeichert werden. Die globalen Einstellungen, die für alle Geräte verfügbar sind, sind die Größenparameter. Die Größe des Graphen ist fest. Änderung der ›Seiten«-Einstellungen ändert die Größe der Zeichenfläche (Canvas) unterhalb des Graphen. Der Wechsel zwischen Hochformat (portrait) und Querformat (landscape) rotiert die Zeichenfläche (Canvas). Stellen Sie sicher, daß die Zeichenfläche groß genug ist, um den Graphen aufzunehmen. Andernfalls erhalten sie eine Warnung, daß der Ausdruck abgeschnitten wurde (Printout truncated). Wenn Ihre Zeichenfläche nicht einfach geändert werden kann, weil z.B. ein Ausdruck im Letter Papierformat erfolgen soll, können Sie die Größe des Graphen anpassen (über ›Viewport« in ›Plot/Graph Appearance«).

4.3.7 Drucken (Print)

Druckt das Projekt mit den aktuellen Druckereinstellungen.

4.3.8 Beenden (Exit)

Beendet Grace. Wenn noch ungespeicherte Arbeiten offen stehen, wird ein Warnungsdialog angezeigt und dem Benutzer die Möglichkeit gegeben, diese Operation abzubrechen.

4.4 Bearbeitenmenü (Edit)

4.4.1 Datensätze (Data Sets...)

Eigenschaften von Datensätzen können über das Datensatz-Popup angezeigt werden. Dies schließt seinen Typ, Länge, zugeordnete Kommentare und etwas Statistik (Minimum, Maximum, Mittelwert, Standardabweichung) mit ein. Der horizontale Scrollbalken unterhalb der Statistiktafel erlaubt, die zwei letzten Eigenschaften zu betrachten – diese werden standardmäßig nicht mit angezeigt. Beachten Sie, das die Spalten bei Bedarf vergrößert werden können, wenn nicht alle Ziffern angezeigt werden. Die Spaltenbegrenzungen können über Shift+Maustaste 2 verschoben werden.

Mit dem oberen Menü in diesem Dialog können bereits bestehende Sätze manipuliert oder neue hinzugefügt werden. Unter den wichtigsten Einträgen im Menü sind die Optionen zum Erzeugen oder zum Manipulieren von Sätzen im Tabellendatensatz-Editor (siehe Tabellendatensatz-Editor, S. 15).

4.4.2 Tabellendatensatz-Editor

Dieser Dialog präsentiert eine Matrix von Zahlen (Tabelle), die dem Datensatz entsprechen. Der Satz-Typ (und folglich, die Zahl von Datenspalten) kann über die Auswahl »Typ verwenden« (»Type«) geändert werden. Wird ein Spaltenlabel angeklickt, so kann über den angezeigten Dialog die Spaltenformatierung eingestellt werden. Klicken auf ein Zeilenlabel ändert den jeweiligen Reihenzustand (ausgewählt/unausgewählt). Die ausgewählten Reihen können über das Bearbeiten-Menü (»Edit«) des Dialoges gelöscht werden. Ein anderer Eintrag in diesem Menü gibt Ihnen die Möglichkeit, eine neue Zeile hinzuzufügen; der Ort der neuen Zeile wird über die Zeile bestimmt, die eine Zelle mit dem Tastaturfokus enthält. Eine Eingabe in einer leeren Zelle fügt diese oder mehrere Zeilen hinzu (die dazwischenliegenden Zeilen werden mit 0 aufgefüllt).

Die Spaltenbegrenzungen können über *Shift+Maustaste 2* verschoben werden.

4.4.3 Satzoperationen (Set Operations...)

Der Satzoperations-Popup erlaubt Ihnen, Sätze als Ganzes zu betrachten und mit diesen zu arbeiten. Wenn Sie auf den Daten, die zu dem Satz zugeordnet sind, arbeiten möchten, sollten Sie das Datensatz-Popup, siehe S. 14, aus dem Datenmenü verwenden. Dieses Popup erlaubt ihnen, eine Quelle (ein Satz innerhalb eines Graphen) sowie ein Ziel auszuwählen und einige Aktionen auf diesen auszuführen (kopieren, verschieben, vertauschen). Ebenso gibt Ihnen dieses Popup einen schnellen Zugriff auf verschiedene Graphen und Satzauswahlen, wenn Sie einige andere Operationen ausführen möchten, wie z. B. das Verstecken eines Graphen oder die Erstellung eines neuen Satzes aus Blockdaten.

4.4.4 Graphen ausrichten (Arrange graphs...)

Dieser Eintrag zeigt ein Popup an, um mehrere Graphen in eine Gitter aus **M** Reihen und **N** Spalten auszurichten.

Die Graphenauswahl oben erlaubt, mehrere Graphen auszuwählen, die ausgerichtet werden sollen. Ist die Anzahl der ausgewählten Graphen nicht gleich der Gitterknoten (**M** mal **N**), werden neue Graphen erzeugt oder zusätzlich Graphen bei Bedarf entfernt, so daß die Zahl der Graphen gleich dem der Gitterknoten ist. Diese Optionen werden über die jeweiligen Checkboxen unter der Graphenauswahl gesteuert.

Die Methode, wie die Matrix wird mit den Graphen aufgefüllt werden soll, kann über ein Dropdown-Element ausgewählt werden (zuerst horizontal und dann vertikal, oder andersherum; beides kann noch invertiert werden). Zusätzlich kann man angeben, daß die Matrix auf schlangenmäßige Weise befüllt werden soll (angrenzende »Schläge« sind antiparallel).

Der Rest der Steuerelemente in diesem Dialog stellt den Matrixabstand ein: linker/rechter/oberer/unterer Seitenabstand (in Viewport-Koordinaten) und *relative* Abstände zwischen den Zellen, sowohl vertikal wie auch horizontal. Neben jedem Eintrag für die vertikalen und horizontalen Abstände befindet sich eine »GepacktCheckbox (»Pack«). Wird diese aktiviert, so werden die effektiven Abstände zwischen den Zellen auf Null gesetzt und die Achsen-Tickmarkierungen so geändert, das nur die unteren bzw. linken Tickmarkierungen angezeigt werden.

Wünschen Sie kein regelmäßige Layout der Graphen, so wie dies über diesen Dialog eingestellt werden kann, können Sie es später mithilfe der Maus ändern (einen Graphen mit der Maus markieren und auf die Fokuserkennung doppelklicken, siehe 4.2.1 Klicks und Doppelklicks, S. 12).

4.4.5 Graphen überdecken (Overlay graphs...)

Sie können einen Graphen mit einem anderen überdecken. Hauptsächlich wird diese Eigenschaft verwendet, wenn mehrere Kurven gezeichnet werden sollen, die unterschiedliche Skalen auf (anscheinend) demselben Graphen verwenden. Die Hauptschwierigkeit bei Verwendung dieser Funktion besteht darin, sicher zu sein, das man auf dem richtigen Graphen operiert (zur Vereinfachung kann ein Graph auch temporär versteckt werden).

4.4.6 Autoskalierung (Autoscale graphs...)

Über diesen Menüeintrag können Sie einen oder alle Graphen automatisch auf den spezifizierten Satz skalieren. Dies ist hilfreich, wenn Sie vergleichbare Graphen benötigen, die Daten mit verschiedenen Bereichen beinhalten, oder wenn Sie den Fokus auf einen Satz setzen wollen, jedoch weitere Daten in einem komplexen Graphen anzeigen wollen.

4.4.7 Region-Menü (Regions)

Status (Status)

Dieses kleine Popup zeigt nur den aktuellen Status von bestehenden Regionen (Typ und ob er aktiv ist oder nicht).

Definition (Define)

Hier können Sie eine neue Region definieren (oder eine bestehende ändern). Die erlaubten Typen der Region sind:

- Innerhalb eines Polygons (Inside polygon)
- Außerhalb eines Polygons (Outside polygon)
- Über der Linie (Above line)
- Unterhalb der Linie (Below line)
- Links von der Linie (Left of line)
- Rechts von der Linie (Right of line)
- Innerhalb des horizontalen Bereich (In horizontal range)
- Innerhalb des vertikalen Bereich (In vertical range)
- Außerhalb des horizontalen Bereich (Out of horizontal range)
- Außerhalb des vertikalen Bereich (Out of vertical range)

Eine Region kann sich auf den aktuellen Graphen oder auf alle Graphen beziehen.

Löschen (Clear)

Löscht eine Region.

Report on

Dieses Popup zeigt Ihnen, welche Sätze oder Punkte sich innerhalb oder außerhalb einer Region befinden.

4.4.8 Hotlinks (Hotlinks...)

Sie können einen Satz zu einer Datei oder Pipe über diese Funktion verlinken. Wenn ein Link eingerichtet wurde, können Sie diesen aktualisieren (z. B. Daten neu einlesen), indem Sie den Aktualisierungsbutton (update button) verwenden.

Derzeit können nur einfache XY-Sätze für Hotlinks verwendet werden.

4.4.9 Setze festen Punkt für Lokalisierung (Set locator fixed point)

Nachdem Sie diesen Menüeintrag ausgewählt haben, können Sie einen Punkt in einem Graphen auswählen, der als Ursprung für die Lokalisierungsleiste (unterhalb des Menüs) verwendet wird. Dieser feste Punkt wird nur verwendet, wenn der Anzeigetyp der Lokalisierung auf [DX,DY] gesetzt wurde.

4.4.10 Lösche festen Punkt für Lokalisierung (Clear locator fixed point)

Dieser Eintrag löscht einen festen Punkt, der vorher gesetzt wurde, und verwendet anschließend den Standardursprung von [0,0].

4.4.11 Lokalisierungseinstellungen (Locator props)

Die Lokalisierungseinstellungen erlauben Ihnen, die Anzeige der Lokalisierung einzustellen, hauptsächlich den Typ, das Format und die Genauigkeit der Anzeige. Es können hierfür alle Formate verwendet werden, die auch in der Graphenskalierung möglich sind.

4.4.12 Einstellungen (Preferences)

Der Einstellungsdialog erlaubt Ihnen, verschiedene Einstellung der Grace-Session einzustellen, wie das GUI-Verhalten, Cursor-Typ, Hinweis für die Datumsverarbeitung und das Bezugsdatum für die Kalender-Konvertierung.

4.5 Datenmenü (Data)

4.5.1 Datensatzoperationen (Data set operations)

Dieses Popup enthält eine Sammlung von Operationen, die sich auf die Sortierung/Ordnung der Datenpunkte innerhalb eines Satzes oder zwischen Sätzen beziehen. Wollen Sie auf einen Satz als Ganzes operieren, müssen Sie den Menüpunkt 4.4.3 Satzoperation (Set Operations. . .), S. 15, aus dem Bearbeiten-Menü (»Edit«) verwenden. Sie können nach jeder Koordinate (X, Y, DX, . . .) in aufsteigender und absteigender Ordnung sortieren, die Richtung einiger Punkte umkehren, mehrere Sätze in einen zusammenfügen, aufteilen eines Satzes in mehrere andere mit gleicher Länge, oder von einem Satz einen Bereich von Punkten verwerfen. Die 4.1.4 Satzauswahl, S. 11, des Popup zeigt die Anzahl der Punkte jeden Satzes in eckigen Klammern (Beispiel: G0.S0[63]), die Punkte werden von 0 bis n-1 durchnummeriert.

4.5.2 Transformationsmenü (Transformations)

Das Transformationsuntermenü gibt Ihnen den Zugriff auf alle Funktionen der Datenanalyse von Grace.

Ausdruck auswerten (Evaluate expression)

Mit »Ausdruck auswerten« können Sie einen neuen Satz erstellen, indem eine Formel auf einen anderen Satz, oder eine Teil eines anderen Satzes bei der Verwendung von Regionen, angewendet wird.

Die klassischen mathematischen Funktionen sind hierfür verfügbar (cos, sin, aber auch lgamma, j1, erf, . . .). Wie gewöhnlich verwenden alle trigonometrischen Funktionen standardmäßig Radian als Winkelmaß, jedoch können Sie eine andere Einheit angeben, wenn Sie dies benötigen (z.B. cos(x rad) oder sin(3 * y deg). Eine vollständige Liste der verfügbaren numerischen Funktionen und Operatoren ist im Abschnitt 5.4 Numerische Operatoren und Funktionen, S. 27, verfügbar.

Innerhalb einer Formel können Sie X, Y, Y1, . . . , Y4 verwenden, um jede Koordinate aus dem Quellensatz anzusprechen. Eine implizite Schleife wird um die Formel gelegt, daß heißt, wenn Sie angeben:

$$x = x - 4966.5$$

werden alle Punkte in Ihrem Satz um 4966.5 Einheiten nach links verschoben.

Sie können auch mehr als einen Satz in einer Formel verwenden, z. B. wie in der nachfolgenden Formel:

$$y = y - 0.653 * \sin(x \text{ deg}) + s2.y$$

Diese Formel verwendet sowohl X und Y vom Quellensatz, jedoch auch die Y-Koordinate aus dem Satz 2. Beachten Sie, daß die Schleife eine einfache Schleife über die Indizes ist. Alle Sätze in einem solchen hybriden Ausdruck sollte deshalb die gleiche Anzahl an Punkte besitzen und Punkt i des einen Satzes wirklich im Zusammenhang mit Punkt i des anderen Satzes stehen. Wenn Ihre Sätze nicht diesen Voraussetzungen entsprechen, sollten Sie sie zuerst über 4.5.2 Interpolation, S. 18, homogenisieren.

Histogramme (Histograms)

Das Histogramm-Popup gibt Ihnen die Möglichkeit, von den Y-Koordinaten Ihrer Daten normale oder kumulative Histogramme zu berechnen. Optional können diese Histogramme auch auf 1 normalisiert werden (um eine Wahrscheinlichkeitsverteilungsfunktion zu erstellen (PDF = Probability Distribution Function)).

Die Behälter können entweder eine lineare Verzahnung, definiert über die min., max. oder Längenwerte, oder eine Verzahnung über eine Abszisse eines anderen Satzes sein (in diesem Fall müssen die Abszisse des Satzes eine ausschließlich monotone Reihe bilden).

Fourier-Transformationen (Fourier transforms)

Mithilfe dieses Popups können Sie direkte und invertierte Fourier-Transformationen durchführen (aktuell werden Kraftspektren berechnet). Standardmäßig werden direkte Transformationen auf ungefilterte Daten durchgeführt und produziert einen Satz mit dem Index als Abszisse und der Größenordnung als Ordinate. Die Filterung der Eingabedaten kann über Dreieck-, Hanning-, Welch-, Hamming-, Blackman- und Parzen-Filter erfolgen. Sie können die Größenordnung, Phase und Koeffizient laden und Index, Frequenz oder die Periode als Abszisse verwenden. Zwischen direkter und invertierter Fourier-Transformation können Sie wählen. Wenn Sie reale Eingabedaten spezifizieren, wird X als leer angenommen und ignoriert; geben die komplexe Eingabedaten an, so wird X als reeller Teil und Y als imaginärer Teil verwendet.

Wenn Grace mit der FFTW-Bibliothek (siehe 2.1 Konfiguration, S. 2) konfiguriert wurde, dann üben die beiden Buttons DFT und FFT die selbe Transformation aus (also gibt es in diesem Fall kein Geschwindigkeitsvorteil in der Verwendung von FFT). Wenn Sie innerhalb Grace FFTW-Dateien verwenden wollen, die die Methode spezifizieren sollen, müssen Sie verschiedene 3.3.1 Umgebungsvariablen, S. 9 setzen, um diese zu benennen.

Mittelwerte (Running averages)

Dieses Popup erlaubt Ihnen, einige Werte in einem gleitenden Fenster über Ihre Daten zu berechnen. Sie wählen beides um die Operation auszuführen, den Wert den Sie benötigen (Mittelwert, Median, Minimum, Maximum, Standardabweichung) sowie die Länge des Fensters. Sie können die Operation auf Punkte innerhalb (bzw. außerhalb) eine Region beschränken.

Differenzen (Differences)

Das Differenzen-Popup wird verwendet, um die Approximation der ersten Ableitung einer Funktion mit begrenzten Abweichungen zu berechnen. Als einzige Einstellmöglichkeit (neben der Wahl des Quellsatzes natürlich) ist der Typ von Abweichungen, die verwendet werden sollen, angebar: vorwärts, rückwärts oder zentriert.

Zeitliche Differenzen (Seasonal differences)

Die zeitliche Differenz wird verwendet, um Daten einer Periode von einer vorherigen Periode (nämlich $y[i] - y[i + \text{Periode}]$) zu subtrahieren. Beachten Sie, daß die Periode als Term mit Bezug auf den Index innerhalb eines Satzes und nicht als Term mit Bezug auf die Abszisse angegeben wird!

Integration (Integration)

Über das Integration-Popup kann das Integral über einen Satz berechnet und optional geladen werden. Der numerische Wert des Integrals wird nach der Berechnung in einem Textfeld angezeigt. Die Auswahl von »kumulative Summe« (cumulative sum) in der Auswahlbox erzeugt und lädt einen neuen Satz mit dem Integral und berechnet den Endwert; die Auswahl von »nur Summe« (sum only) berechnet nur den Endwert.

Interpolation/Splines (Interpolation/Splines)

Dieses Popup wird verwendet, um einen Satz auf einer Reihe von alternativen X-Koordinaten zu interpolieren. Dies wird hauptsächlich verwendet, bevor eine komplexe Operationen zwischen zwei Sätzen über das 4.5.2 Ausdruck auswerten (Evaluate expression)-Popup, S. 17, durchgeführt wird.

Die Reihe kann entweder eine lineare Verzahnung, definiert über seine Minimum-, Maximum- und Längenwerte, oder eine Verzahnung, gebildet durch die Abszisse eines anderen Satzes, sein.

Verschiedene Interpolationsmethoden können verwendet werden: linear, Spline oder Akima Spline.

Beachten Sie, wenn die gesammelten Maschen nicht vollständig in den von der Quelle gesetzten X-Grenzen sich befindet, wird die Bewertung der Punkte hinter dieser Grenze über die Interpolationsparameter vom ersten (oder letzten) Segment des Quellsatzes durchgeführt, was als eine primitive Extrapolation angesehen werden kann. Dieses Verhalten kann über die »StrictOption innerhalb des Popups deaktiviert werden.

Die Abszissen des Satzes, der interpoliert wird, müssen eine ausschließlich monotone Reihe bilden.

Regression (Regression)

Das Regressions-Popup kann dazu verwendet werden, um einen Satz gegen Polynome oder einige spezifische Funktionen ($y = A \cdot x^B$, $y = A \cdot \exp(B \cdot x)$, $y = A + B \cdot \ln(x)$ und $y = \frac{1}{(A+Bx)}$) anzupassen, für den eine einfache Transformationen der Eingabedaten verwendet werden kann, um lineare Regressionsformeln anzuwenden.

Sie können entweder die angepassten Werte, das Residuum oder die Funktion selbst laden. Die Auswahl, angepassten Werte oder das Residuum zu laden, führt zu einem Satz mit der selben Länge und Abszisse wie der ursprüngliche Satz. Die Auswahl um die Funktion zu laden ist fast ähnlich wie das Laden der angepassten Werte, außer daß Sie selbst die Grenzen der Punktzahl wählen. Dies kann zum Beispiel dazu verwendet werden, eine Kurve außerhalb der Datenbeispielreihe zu zeichnen oder einen gleichmäßig verteilten Satz aus einem unregelmäßigen zu erzeugen.

Nichtlineare Anpassung (Non-linear fit)

Das Popup für die nichtlineare Anpassung kann für Funktionen verwendet werden, die außerhalb des Bereiches für die einfache Regressionsmethode liegen. Mithilfe dieses Popups können Sie den Ausdruck selber angeben, mit der Angabe von a_0, a_1, \dots, a_9 für die Anpassungsparameter (zum Beispiel können Sie angeben: $y = a_0 \cdot \cos(a_1 \cdot x + a_2)$). Sie geben eine Toleranz an, die Startwerte, optional die Grenzen und können mehrere Schritte vor dem Laden der Ergebnisse ausführen.

Die Anpassungs-Charakteristica (Anzahl an Parametern, Formel, . . .) können in einer Datei gespeichert werden und wenn erforderlich, über die Verwendung des Dateimenüs (File menu) geladen werden.

Im Erweitert-Tabulator (»Advanced«) können Sie zusätzlich eine Beschränkung auf den Satz/Sätzen einrichten, die angepaßt werden sollen (dies ignoriert Punkte, die das Kriterium nicht erfüllen), eines von den voreingestellten Gewichtungsschematas verwenden oder die Definition von Ihren eigenen durchführen (beachten Sie, daß sich »dY« im voreingestellten »1/dY²« derzeit auf die dritte Spalte des Datensatzes bezieht; verwenden Sie die Benutzerdefiniert-Funktion (»Custom«), wenn dies keinen Sinn für Ihren Datensatz macht), und wählen Sie aus, ob die angepassten Werte, die Differenzen oder die Funktion an sich geladen werden soll. Wählen Sie aus, ob die angepassten Werte oder die Differenzen eines Satzes die selbe Länge und Abszisse besitzen sollen, wie der ursprüngliche Satz. Die Auswahl des Ladens einer Funktion ist ähnlich dem, die angepassten Werte zu laden, mit der Ausnahme, daß Sie selber die Grenzen und die Anzahl der Punkte bestimmen. Dies kann beispielsweise dazu verwendet werden, eine Kurve außerhalb des Bereiches der gesammelten Daten zu zeichnen oder eine gleichmäßigen Satz aus einem ungleichmäßigen zu erzeugen.

Korrelation/kumulative Varianz (Correlation/covariance)

Dieses Popup kann dazu verwendet werden, um Autokorrelationen eines Satzes oder eine Querkorrelation zwischen zwei Sätzen zu berechnen. Sie können nur den Satz (oder die Sätze) auswählen und den maximalen Zeitabstand spezifizieren. Eine Check-Box erlaubt Ihnen, Kovarianzen anstelle von Korrelationen zu bestimmen. Das Ergebnis wird dann zu $\text{abs}(C(0)) = 1$ normalisiert.

Digitale Filter (Digital filter)

Sie können einen Satz als Gewichtung verwenden, um einen anderen Satz zu filtern. Einzig der Y-Teil und die Länge des Gewichtung-Satzes sind relevant, der X-Teil wird ignoriert.

Lineare Faltung (Linear convolution)

Das »Lineare Faltung«-Popup wird verwendet. . . um zwei Sätze zusammenzufalten. Sie können nur die Sätze auswählen und anschließend bestätigen.

Geometrische Transformationen (Geometric transforms)

Sie können Sätze mithilfe des Popup »Geometrische Transformation« rotieren lassen, diese skalieren oder transformieren. Spezifiziert werden muß von Ihnen die Charakteristik jeder Transformation und Reihenfolge der Ausführung.

Beispielpunkte (Sample points)

Dieses Popup stellt zwei Sammelmethode zur Verfügung. Die erste ist die Auswahl eines Startpunktes und einer Schrittweite, die zweite ist die Auswahl von Punkten, die einen von Ihnen spezifizierten booleschen Ausdruck genügen.

Daten beschneiden (Prune data)

Dieses Popup wird zur Reduzierung von riesigen Sätzen verwendet (und anschließend der Reduzierung von Berechnungszeit wie auch Speicherplatz).

Die Interpolation-Methode kann nur auf geordneten Sätzen angewendet werden: sie beruht auf der Annahme, daß ein Punkt überflüssig ist und entfernt werden kann, wenn ein echter Punkt und eine auf benachbarten Punkten beruhende Interpolation näher liegen als eine angegebene Schwelle.

Die geometrischen Methoden (Kreis, Ellipse, Rechteck) können auf jeden Satz angewendet werden, sie prüfen jeden Punkt der Reihe und behalten nur diejenigen, die nicht in der Nachbarschaft von vorherigen Punkten liegen.

4.5.3 Eigenschaftsermittlung (Feature extraction)

In Anbetracht einer Reihe von Kurven innerhalb eines Graphen können Sie eine Eigenschaft aus jeder Kurve herausziehen und die Werte der Eigenschaft dazu verwenden, um die Y-Werte für eine neue Kurve zur Verfügung zu stellen.

Eigenschaft	Beschreibung
Y-Minimum (Y minimum)	Minimaler Y-Wert eines Satzes
Y-Maximum (Y maximum)	Maximaler Y-Wert eines Satzes
Y-Mittel (Y average)	Mittelwert der Y-Werte eines Satzes
Y Standardabweichung (Y std. dev.)	Standardabweichung der Y-Werte eines Satzes
Y-Median (Y median)	Median der Y-Werte eines Satzes
X-Minimum (X minimum)	Minimaler X-Wert eines Satzes
X-Maximum (X maximum)	Maximaler X-Wert eines Satzes
X-Mittel (X average)	Mittelwert der X-Werte eines Satzes
X-Standardabweichung (X std. dev.)	Standardabweichung der X-Werte eines Satzes
X-Median (X median)	Median der X-Werte eines Satzes
Frequenz (Frequency)	Führt eine DFT (FFT wenn die Satzlänge eine Potenz von 2 ist) durch und sucht die längste Frequenz-Komponente
Periode (Period)	Invertierung von Frequenz
Nulldurchgang (Zero crossing)	Zeit des ersten Nulldurchgangs, im + oder - Bereich
Anstiegszeit (Rise time)	Annahme, daß die Kurve am Minimum beginnt und bis zum Maximum ansteigt; die Zeit, die benötigt wird, um von 10% auf 90% des Anstieges zu gelangen. Für einzelne Exponentialkurven liegt dies konstant bei 2.2*Zeit.
Abfallzeit (Fall time)	Annahme, daß die Kurve beim Maximum beginnt und zum Minimum fällt; die Zeit, die benötigt wird, um von 90% auf 10% des Abstiegs zu gelangen.

Eigenschaft	Beschreibung
Steigung (Slope)	Führt eine lineare Regression durch, um die Steigung zu ermitteln
Y-Achsenabschnitt (Y intercept)	Führt eine lineare Regression durch, um den Achsenabschnitt zu ermitteln
Satzlänge (Set length) maximale Breite bei halber Höhe (Half maximal width)	Anzahl der Datenpunkte innerhalb des Satzes
X-Schwerpunkt (Barycenter X)	Schwerpunkt entlang der X-Achse
Y-Schwerpunkt (Barycenter Y)	Schwerpunkt entlang der Y-Achse
X(Y max)	X-Wert bei maximalen Y-Wert
Y(X max)	Y-Wert bei maximalen X-Wert
Integral (integral)	Integral

Tabelle 5: Übersicht der Eigenschaften

4.5.4 Import-Menü (Import)

ASCII

Einlesen neuer Sätze von Daten in einen Graphen. Die 4.1.3 Graphenauswahl, S. 11, wird verwendet, um den Graphen anzugeben, zu dem die Daten hinzugefügt werden sollen (außer wenn Blockdaten eingelesen werden, die zum Graphen später kopiert werden).

Einlesen als »Einzelner Satz« (Single set) bedeutet, daß, wenn die Quelle nur eine Spalte mit numerischen Daten enthält, ein Satz hieraus erzeugt wird, bei dem der Index (von 1 bis zur absoluten Anzahl von Punkten) als Abszisse verwendet wird und die gelesenen Werte als Ordinate. Wenn die Quelle mehr als eine Datenspalte enthält, so werden die ersten zwei numerischen Spalten verwendet. Einlesen als »NXY« bedeutet, daß die erste numerische Spalte die Abszisse darstellt und alle nachfolgenden Spalten die Ordinaten von verschiedenen Sätzen bereitstellen. Einlesen als »Blockdaten« (Block data) bedeutet, alle Spalten werden eingelesen und gespeichert – ein anderes Popup erlaubt dann, die Abszissen und Ordinaten nach Wunsch auszuwählen. Anzumerken sei hier, daß Blockdaten nur so lange gespeichert werden, wie sie nicht von einem neuen Einlesen überschrieben werden. Sie können noch Daten von einem Block wiederbekommen, nachdem die gesamten Popups geschlossen wurden, über die 4.1.4 Satzauswahl, S. 11.

Der Satztyp kann einer der vorgegebenen Satzdarstellungstypen sein (siehe 3.1.6 Sätze, S. 5).

Die Quelle der Daten kann als »Platte« (Disk) oder »Pipe« ausgewählt werden. Im ersten Falle wird der Text im Auswahlfeld (Selection) als Dateiname aufgefaßt (es kann automatisch über die Dateiauswahl im oberen Teil des Popup gesetzt werden). Im zweiten Fall wird der Text im Auswahlfeld als ein Kommando aufgefaßt, der ausgeführt wird und die Daten auf der Standardausgabe ausgibt. Auf System, die dies erlauben, kann das Kommando eine komplette Sequenz von Programmen sein, die über Pipes miteinander verbunden sind.

Wenn die Quelle Datumsfelder enthält, sollten diese automatisch erkannt werden. Verschiedene Formate werden dabei erkannt (siehe Anhang 7.4 Datum in Grace, S. 50). Kalendarische Datums werden beim Einlesen in numerische Datums konvertiert.

Das Popup-Pulldownmenu »Autoskalierung beim Lesen« (Autoscale on read) spezifiziert, welche Achsen eines Graphen beim Einlesen eines neuen Satzes autoskaliert werden sollen.

NetCDF

Dieser Eintrag existiert nur, wenn Grace mit der Unterstützung für das NetCDF Datenformat kompiliert wurde (siehe 2.1 Konfiguration, S. 2).

4.5.5 Export-Menü (Export)

ASCII

Speichert Datensätze in eine Datei. Eine [4.1.4 Satzauswahl](#), S. 11, wird dazu verwendet, die zu exportierenden Sätze auszuwählen. Spezifiziert werden kann das Format, in dem die Datenpunkte gespeichert werden (der Standardwert ist »%16.8g«). Existiert bereits eine Datei mit gleichem Namen, so wird eine Warnung angezeigt.

4.6 Plotmenü (Plot)

4.6.1 Plotaussehen (Plot appearance)

Das Popup für das Plotaussehen (Plot appearance) ermöglicht Ihnen, die Zeitstempелеigenschaften und die Hintergrundfarbe der Seite einzustellen. Die Farbe wird außerhalb des Graphen verwendet und, falls keine spezifische Hintergrundfarbe gesetzt wird, auch auf den Graphen angewendet. Die Zeitmarke wird jedes Mal aktualisiert, wenn das Projekt modifiziert wird.

4.6.2 Graphaussehen (Graph appearance)

Das Popup für das Graphaussehen kann über das Plotmenü oder über einen Doppelklick auf die Legende, Titel oder Untertitel eines Graphen erreicht werden (siehe [4.2.1 Klicks und Doppelklicks](#), S. 12). Die Graphenauswahl im oberen Teil erlaubt Ihnen, den Graphen auszuwählen, auf den sich die Änderungen beziehen sollen, dieser erlaubt auch, verschiedene allgemeine Aktionen über dessen Pop-upmenü durchzuführen (siehe [4.1.3 Graphenauswahl](#), S. 11). Viele dieser Aktionen können ebenfalls über das Bearbeitenmenü (Edit) in der Popup Menüleiste ausgeführt werden. Der Haupttab enthält die Eigenschaften, die Sie sehr häufig benötigen werden (den Titel zum Beispiel), sowie weitere Tabs für die Feineinstellungen, die nicht so häufig benötigt werden (Schriften, Größen, Farben, Ausrichtungen).

Sollten Sie spezielle Zeichen oder eine spezielle Formatierung in Ihrem Titel oder Untertitel benötigen, können Sie die Grace Escape-Sequenzen verwenden (die Sequenz wird im Textfeld wie eingegeben angezeigt, jedoch auf dem Graphen entsprechend interpretiert und dargestellt), siehe [7.1 Schriftsatz](#), S. 46. Wenn Sie sich nicht an das Mapping zwischen einem alphabetischen Zeichen und dessen Glyph in einem spezifischen Zeichensatz erinnern können (hauptsächlich Symbole und Zapfdingbats), können Sie das Schriftarten-Werkzeug über die Tastaturkombination STRG-e vom Textfeld aus aufrufen. In diesem können sie die Schrift ändern und Zeichen aus diesem Auswählen und über den »Akzeptieren-Button« (Accept) die ausgewählten Zeichen in das Textfeld kopieren. Achten Sie dabei auf die Position des Cursors, wenn Sie Text eingeben oder Änderungen über das Schriftarten-Werkzeug durchführen – die Zeichen oder Kommandos werden an der Cursor-Position eingefügt, nicht am Ende der Zeichenkette!

Sie können die Parameter für das Graphenaussehen speichern oder Einstellungen, die vorher gesichert wurden, über das Dateimenü (File) in diesem Popup wieder einlesen. Im Dialog »Sicherung der Einstellungen« (Save parameters) können Sie wählen, ob Sie die Einstellungen für den aktuellen Graphen oder für alle Graphen speichern möchten.

4.6.3 Satzaussehen (Set appearance)

Das Popup für das Satzaussehen kann entweder über das Plotmenü oder über einen Doppelklick irgendwo innerhalb eines Graphen aufgerufen werden (siehe [4.2.1 Klicks und Doppelklicks](#), S. 12). Die Satzauswahl im oberen Teil erlaubt Ihnen, einen Satz auszuwählen, den Sie bearbeiten wollen – dieser erlaubt auch, verschiedene allgemeine Aktionen über dessen Pop-upmenü durchzuführen (siehe [4.1.4 Satzauswahl](#), S. 11). Der Haupttab faßt die Eigenschaften zusammen, die Sie am häufigsten benötigen werden (Linien- und Symboleinstellungen oder den Legendentext zum Beispiel). Die anderen Tabs werden zur Feineinstellung von nicht so häufig benötigten Optionen benötigt (zum Beispiel Linien entfernen, Füllungseigenschaften, kommentierte Werte und Fehlerbalkeneinstellungen).

Beachten Sie, daß trotz der Angabe einer Legendenzeichenfolge für *einen* Satz im Satzaussehen-Popup, dieser noch nicht vollständig genug ist, um diesen anzuzeigen. Die Anzeige von *allen* Legenden ist

eine Entscheidung auf Graphenniveau, so das die Anzeige über den Haupttab des Popup für das 4.6.2 Graphenaussehen, S. 22, eingestellt werden muß.

Sollten Sie spezielle Zeichen oder eine spezielle Formatierung in Ihrem Titel oder Untertitel benötigen, können Sie die Grace Escape-Sequenzen verwenden (die Sequenz wird im Textfeld wie eingegeben angezeigt, jedoch auf dem Graphen entsprechend interpretiert und dargestellt), siehe 7.1 Schriftsatz, S. 46. Wenn Sie sich nicht an das Mapping zwischen einem alphabetischen Zeichen und dessen Glyph in einem spezifischen Zeichensatz erinnern können (hauptsächlich Symbole und Zapfdingbats), können Sie das Schriftarten-Werkzeug über die Tastaturkombination STRG-e vom Textfeld aus aufrufen. In diesem können sie die Schrift ändern und Zeichen aus diesem Auswählen und über den »Akzeptieren-Button« (Accept) die ausgewählten Zeichen in das Textfeld kopieren. Achten Sie dabei auf die Position des Cursors, wenn Sie Text eingeben oder Änderungen über das Schriftarten-Werkzeug durchführen – die Zeichen oder Kommandos werden an der Cursor-Position eingefügt, nicht am Ende der Zeichenkette!

4.6.4 Achsen-Einstellungen (Axis properties)

Das Popup für die Achsen-Einstellungen kann sowohl vom Plotmenü (Plot) wie auch durch einen Doppelklick auf eine Achse (siehe 4.2.1 Klicks und Doppelklicks, S. 12) aufgerufen werden. Das Pulldown-Menü im oberen Teil erlaubt Ihnen, die Achse auszuwählen, die Sie bearbeiten möchten. Der »Aktive«-Knopf (Active) aktiviert oder deaktiviert global die Achse (alle GUI-Elemente werden bei ausgeschalteter Achse deaktiviert). Die Start- und Stop-Felder zeigen den Darstellungsbereich. Drei Skalentypen sind verfügbar: linear, logarithmisch und reziprog – ebenso können Sie die Achsen invertieren (welche normalerweise von links nach rechts und von unten nach oben zunimmt). Der Haupttab enthält die Eigenschaften, die Sie häufiger verwenden werden (zum Beispiel Achsenbeschriftung, Tick-Abstand und Formate) – weitere Tabs werden für die Feineinstellung bereitgestellt, die nicht so häufig verwendet werden (Schriften, Größen, Farben, Plazierungen, Staffellungen, Rasterlinien, spezielle Ticks, ...).

Sollten Sie spezielle Zeichen oder eine spezielle Formatierung in Ihrem Titel oder Untertitel benötigen, können Sie die Grace Escape-Sequenzen verwenden (die Sequenz wird im Textfeld wie eingegeben angezeigt, jedoch auf dem Graphen entsprechend interpretiert und dargestellt), siehe 7.1 Schriftsatz, S. 46. Wenn Sie sich nicht an das Mapping zwischen einem alphabetischen Zeichen und dessen Glyph in einem spezifischen Zeichensatz erinnern können (hauptsächlich Symbole und Zapfdingbats), können Sie das Schriftarten-Werkzeug über die Tastaturkombination STRG-e vom Textfeld aus aufrufen. In diesem können sie die Schrift ändern und Zeichen aus diesem Auswählen und über den »Akzeptieren-Button« (Accept) die ausgewählten Zeichen in das Textfeld kopieren. Achten Sie dabei auf die Position des Cursors, wenn Sie Text eingeben oder Änderungen über das Schriftarten-Werkzeug durchführen - die Zeichen oder Kommandos werden an der Cursor-Position eingefügt, nicht am Ende der Zeichenkette!

Die meisten Steuerungselemente im Dialog sollten selbsterklärend sein. Einer der es nicht ist (und oft vermißt wird) ist das Achsentransformations-Eingabefeld (Axis transform) im Tickbeschriftungs-Tab (Tick labels). Wird hier z. B. »-\$t« eingetragen, so werden die Tick-Bezeichnungen der reellen Koordinaten negativ angezeigt. Sie können jeden Ausdruck hierfür angeben, der vom Interpreter (siehe 5 Kommandointerpreter, S. 25) verstanden wird.

Sobald Sie die von Ihnen gewünschten Optionen gesetzt haben, können Sie sie anwenden. Eine nützliche Eigenschaft ist, daß Sie mehrere Achsen mit dem Pulldown-Menü im unteren Teil auf einmal setzen können (aktuelle Achse, alle Achsen des aktuellen Graphen, aktuelle Achse aller Graphen, alle Achsen in allen Graphen). Beachten Sie, daß Sie jeweils die Eigenschaften in allen Tabs anwenden, nicht nur die des derzeit aktivierten Tabs.

4.7 Ansichtmenü (View)

4.7.1 Anzeige der Lokalisierungsleiste (Show locator bar)

Dieser Umschaltpunkt zeigt oder versteckt die Lokalisierungsleiste, die sich unter der Menüleiste befindet.

4.7.2 Anzeige der Statusleiste (Show status bar)

Dieser Umschaltpunkt zeigt oder versteckt die Statusleiste, die sich unterhalb der Zeichenfläche befindet.

4.7.3 Anzeige der Tooleiste (Show tool bar)

Dieser Umschaltpunkt zeigt oder versteckt die Tooleiste, die sich an der linken Seite der Zeichenfläche befindet.

4.7.4 Seiteneinstellungen (Page setup)

Setzt die Eigenschaften des Anzeigegerätes. Es ist der gleiche Dialog wie in [4.3.6 Druckeinstellungen](#), S. 14.

4.7.5 Neuzeichnen (Redraw)

Dieser Menüpunkt läßt die Zeichenfläche neu erzeugen.

4.7.6 Alles aktualisieren (Update all)

Dieser Menüpunkt läßt alle GUI-Elemente aktualisieren. Normalerweise wird alles automatisch aktualisiert, es sei denn, Änderungen werden über den [5 Kommandointerpreter](#), S. 25, durchgeführt.

4.8 Fenstermenü

4.8.1 Kommandos (Commands)

Kommandozeilenversion der Schnittstelle zu Grace. Hier können Kommandos in das untere Textfeld eingegeben werden und durch Betätigung der Return-Taste ausgeführt werden. Das Kommando wird ausgewertet und ausgeführt und das Kommando wird in die History-Liste eingefügt. Einträge in der History-Liste können durch anklicken mit der linken Maustaste einfach ein weiteres Mal ausgeführt werden. Für eine Referenz des Grace Kommandozeileninterpreters siehe [5 Kommandointerpreter](#), S. 25.

4.8.2 Punktexplorer (Point explorer)

Noch nicht geschrieben. . .

4.8.3 Objekte zeichnen (Drawing objects)

Noch nicht geschrieben. . .

4.8.4 Schrifttool (Font tool)

Noch nicht geschrieben. . .

4.8.5 Konsole (Console)

Das Konsolenfenster zeigt Fehler und Ergebnisse einiger numerischer Operationen an, z.B. Nichtlineare Anpassungen (siehe [4.5.2 nichtlineare Anpassungen](#), S. 19). Das Fenster wird automatisch angezeigt, wenn ein Fehler aufgetreten ist oder eine neue Ergebnismeldung eingetroffen ist. Dies Verhalten kann über den Menüpunkt »Options/Popup only on errors« geändert werden, der in diesem Konsolenfenster verfügbar ist.

4.9 Hilfemenü (Help)

4.9.1 Kontexthilfe (On context)

Nach der Aktivierung der Kontexthilfe kann über Anklicken eines Elements des Interface eine Kontexthilfe erhalten werden. Derzeit nur teilweise implementiert.

4.9.2 Benutzerhandbuch (User's guide)

Anzeige des Grace Benutzerhandbuchs.

4.9.3 Tutorial (Tutorial)

Anzeige des Grace Tutorials.

4.9.4 FAQ (FAQ)

Anzeige der häufig gestellten Fragen (FAQ) mit Antworten.

4.9.5 Änderungen (Changes)

Anzeige der Liste mit Änderungen im Laufe der Graceentwicklung.

4.9.6 Beispiele (Examples)

Der gesamte Zweig von Untermenüs, wobei jeder Menüeintrag ein Beispielplot anzeigt.

4.9.7 Kommentare (Comments)

Dieser Menüpunkt kann dazu verwendet werden, den Entwicklern Programmwünsche und Fehlerberichte zuzusenden.

4.9.8 Lizenzvereinbarung (License terms)

Anzeige der Grace Lizenzvereinbarung (GPL Version 2).

4.9.9 Über (About)

Ein Popup-Dialog mit grundsätzlichen Informationen zu der Software, enthält auch einige Konfigurationseinzelheiten. Mehr Einzelheiten können über den Kommandozeilenparameter »-version« beim Starten von Grace ermittelt werden.

5. Kommandointerpreter

5.1 Generelle Hinweise

Der Interpreter bearbeitet seinen Eingang zeilenbasiert. Es können mehrere Statements innerhalb einer Zeile vorkommen, diese müssen das über ein Semikolon (;) getrennt werden. Die maximale Länge einer Zeile beträgt 4kB (fest codiert). Der Parser arbeitet unabhängig von Groß- und Kleinschreibung und alle Zeilen, die mit einer Raute (#) beginnen, werden ignoriert.

5.2 Definitionen

Name	Beschreibung	Beispiel
expr	jeder numerischer Ausdruck	1.5 + sin(2)
iexpr	jeder Ausdruck, der eine Ganzzahl (Integer) berechnet	25, 0.1 + 1.9, PI/asin(1)
nexpr	nichtnegative iexp	2-1
indx	nichtnegative iexp	
sexpr	Zeichenausdruck	»ein String«, »ein «. »String«, »Quadratwurzel von 4 = «. sqrt(4)
vexpr	Vektorausdruck	»2*x«

Tabelle 6: Basis-Typen

Ausdruck	Beschreibung	Typen	Beispiel
GRAPH[id]	Graph <i>id</i>	indx <i>id</i>	GRAPH[0]
Gnn	Graph <i>nn</i>	nn: 0-99	G0

Tabelle 7: Graphenauswahl

Ausdruck	Beschreibung	Typen	Beispiel
graph.SETS[id]	setze <i>id</i> im Graphen <i>graph</i>	indx <i>id</i> , graphsel <i>graph</i>	GRAPH[0].SETS[1]
graph.Snn	setze <i>nn</i> im Graphen <i>graph</i>	nn: 0-99, graphsel <i>graph</i>	G0.S1
SET[id]]	setze <i>v</i> im aktuellen Graphen	indx <i>id</i>	SET[1]
Snn	setze <i>nn</i> im aktuellen Graphen	nn: 0-99	S1
S_	der letzte implizit (d.h. infolge einer Datentransformation) zugeteilte Satz im aktuellen Graphen	-	S_
S\$	der aktive Satz im aktuellen Graphen	-	S\$

Tabelle 8: Satzauswahl

Ausdruck	Beschreibung	Typen	Beispiel
Rn	Region <i>n</i>	n: 0-4	R0

Tabelle 9: Regionauswahl

Ausdruck	Beschreibung	Typen	Beispiel
COLOR »colorname«	eine gemappte Farbe <i>colorname</i>	-	COLOR »red«
COLOR <i>id</i>	eine gemappte Farbe mit ID <i>id</i>	nexpr <i>id</i>	COLOR 2

Tabelle 10: Farbauswahl

Ausdruck	Beschreibung	Typen	Beispiel
PATTERN <i>id</i>	Pattern mit ID <i>id</i>	nexpr <i>id</i>	PATTERN 1

Tabelle 11: Patternauswahl

Ausdruck	Beschreibung	Typen	Beispiel
X	die erste Spalte	-	X
Y	die zweite Spalte	-	Y
Yn	(<i>n</i> +2)te Spalte	n = 0-4	Y3

Tabelle 12: Datenspaltenauswahl

Derzeit noch nicht vollständig. . .

5.3 Variablen

Variable	Beschreibung
datacolumn	Datenspalte des aktuellen (»aktiven«) Satzes
set.datacolumn	Datenspalte des Satzes
vvar	Benutzerdefiniertes Feld (Array)
vvariable [i:j]	Segment einer Vektorvariablen (Elemente vom i-ten bis inklusive zum j-ten, $i \leq j$)

Tabelle 13: Vektorvariablen

Variable	Beschreibung
vvariable[i]	i-tes Element einer Vektorvariablen
var	Benutzerdefinierte Variable

Tabelle 14: Skalarvariablen

5.4 Numerische Operatoren und Funktionen

In numerischen Ausdrücken wird das Infix-Format verwendet. Argumente von sowohl Operatoren als auch Funktionen können entweder Skalare oder Vektor-Felder sein. Arithmetische, logische und Vergleichsoperatoren sind in den nachfolgenden Tabellen wiedergegeben.

5.4.1 Operatoren

Operator	Beschreibung
+	Addition
-	Subtraktion
★	Multiplikation
/	Division
%	Modulo
^	Potenzen

Tabelle 15: Arithmetische Operatoren

Operator	Beschreibung
AND oder &&	Logisches UND
OR oder	logisches ODER
NOT oder !	logisches NICHT

Tabelle 16: Logische Operatoren

Operator	Beschreibung
EQ oder ==	Gleichheit
NE oder !=	Ungleichheit
LT oder <	Kleiner als
LE oder <=	Kleiner als/Gleich
GT oder >	Größer als
GE oder >=	Größer als/Gleich

Tabelle 17: Vergleichsoperatoren

Ein weiterer Bedingungsoperator ist der »?:« (oder ternäre) Operator, der wie in C oder vielen anderen Sprachen ausgeführt wird.

`(expr1) ? (expr2) : (expr3);`

Dieser Ausdruck evaluiert zu `expr2`, wenn `expr1` zu TRUE evaluiert, und zu `expr3`, wenn `expr1` zu FALSE evaluiert.

5.4.2 Funktionen

Funktion	Beschreibung
<code>abs(x)</code>	Absolutwert
<code>acos(x)</code>	Arcuscosinus
<code>acosh(x)</code>	hyperbolischer Arcuscosinus
<code>asin(x)</code>	Arcussinus
<code>asinh(x)</code>	hyperbolischer Arcuscosinus
<code>atan(x)</code>	Arcustangens
<code>atan2(y,x)</code>	Arcustanges von zwei Variablen
<code>atanh(x)</code>	hyperbolischer Arcustangens
<code>ceil(x)</code>	größte ganze Zahl
<code>cos(x)</code>	Cosinus
<code>cosh(x)</code>	hyperbolischer Cosinus
<code>exp(x)</code>	Exponentialfunktion (e^x)
<code>fac(n)</code>	Fakultät, $n!$
<code>floor(n)</code>	kleinste ganze Zahl
<code>irand(n)</code>	Zufallszahl als Integer, die Zahl ist kleiner als n
<code>ln(x)</code>	Natürlicher Logarithmus
<code>log10(x)</code>	Logarithmus zur Basis 10
<code>log2(x)</code>	Logarithmus zur Basis 2
<code>maxof(x,y)</code>	Gibt den größeren Wert der beiden übergebenen Zahlen zurück
<code>mesh(n)</code>	Maschenreihe (0 . . . n - 1)
<code>mesh(x1, x2, n)</code>	Maschenreihe von n Punkten mit gleichem Abstand zwischen x1 und x2 (inklusive)
<code>minof(x,y)</code>	Gibt den kleineren Wert der beiden übergebenen Zahlen zurück
<code>mod(x,y)</code>	Modulo-Funktion (also $x\%y$)
<code>pi</code>	die PI-Konstante
<code>rand</code>	Pseudo-Zufallszahl zwischen (0,0,1,0)
<code>rand(n)</code>	Feld von n Zufallszahlen
<code>rint(x)</code>	Rundung auf nächste ganze Zahl
<code>rsum(x)</code>	Laufende Summe von x
<code>sin(x)</code>	Sinus
<code>sinh(x)</code>	hyperbolischer Sinus
<code>sqr(x)</code>	x^2
<code>sqrt(x)</code>	Quadratwurzel (\sqrt{x})
<code>tan(x)</code>	Tangens
<code>tanh(x)</code>	hyperbolischer Tangens

Tabelle 18: Funktionen

Funktion	Beschreibung
chdtr(df, x)	Chi-Quadrat Verteilung
chdtrc(v, x)	komplementäre Chi-Quadrat Verteilung
chdtri(df, y)	invertierte, komplementäre Chi-Quadrat Verteilung
erf(x)	Fehlerfunktion
erfc(x)	komplementäre Fehlerfunktion
fdtr(df1, df2, x)	F-Verteilungsfunktion
fdtrc(x)	komplementäre F-Verteilungsfunktion
fdtri(x)	invertierte, komplementäre F-Verteilungsfunktion
gdtr(a, b, x)	Gamma-Verteilungsfunktion
gdtrc(a, b, x)	komplementäre Gamma-Verteilungsfunktion
ndtr(x)	Normalverteilungsfunktion
ndtri(x)	invertierte Normalverteilungsfunktion
norm(x)	Gaußsche Dichteverteilung
pdtr(k, m)	Poisson-Verteilung
pdtrc(k, m)	komplementäre Poisson-Verteilung
pdtri(k, y)	invertierte Poisson-Verteilung
rnorm(xbar, s)	Pseudozufallszahlenverteilung $N(xbar, s)$
stdtr(k, t)	Student's t-Verteilung
stdtri(k, p)	funktionell invertierte Student's t-Verteilung

Tabelle 19: Statistische Funktionen

Funktion	Beschreibung
MIN(x)	Minimalwert des Feldes x
MAX(x)	Maximalwert des Feldes x
AVG(x)	Mittelwert des Feldes x
SD(x)	Standardabweichung des Feldes x
SUM(x)	Summe aller Elemente des Feldes x
INT(x,y)	Integral von $y dx$
IMIN(x)	Index des Minimalwertes des Feldes x
IMAX(x)	Index des Maximalwertes des Feldes x

Tabelle 20: Aggregatfunktionen

Funktion	Beschreibung
ai(x), bi(x)	Airy Funktionen (zwei unabhängige Lösungen für die differentielle Gleichung $y''(x) = xy$)
beta(x)	Beta-Funktion
chi(x)	hyperbolisches Cosinus-Integral
ci(x)	Cosinus-Integral
dawsn(x)	Dowson's Integral
ellie(phi, m)	unvollständiges Ellipsen-Integral der zweiten Art
ellik(phi, m)	unvollständiges Ellipsen-Integral der ersten Art
ellpe(m)	vollständiges Ellipsen-Integral der zweiten Art
ellpk(m)	vollständiges Ellipsen-Integral der ersten Art
expn(n, x)	Exponential-Integral
fresnlc(x)	Cosinus Fresnel-Integral
fresnls(x)	Sinus Fresnel-Integral
gamma(x)	Gamma-Funktion
hyp2fl(a, b, c, x)	Gauß'sche hypergeometrische Funktion
hyperg(a, b, x)	zusammenfließende hypergeometrische Funktion
i0e(x)	modifizierte Bessel-Funktion der Ordnung 0, exponential skaliert
i1e(x)	modifizierte Bessel-Funktion der Ordnung 1, exponential skaliert
igam(a, x)	unvollständiges Gamma-Integral
igamc(a, x)	komplementäres, unvollständiges Gamma-Integral
igami(a, p)	Invertierung des komplementären, unvollständigen Gamma-Integrals
incbet(a, b, x)	unvollständiges Beta-Integral
incbi(a, b, y)	Invertierung des unvollständigen Beta-Integrals
iv(v, x)	Veränderte Bessel-Funktion der Ordnung v
jv(v, x)	Bessel-Funktion der Ordnung v
k0e(x)	veränderte Bessel-Funktion, dritte Art, 0. Ordnung, exponential skaliert
k1e(x)	veränderte Bessel-Funktion, dritte Art, 1. Ordnung, exponential skaliert
kn(n, x)	veränderte Bessel-Funktion, dritte Art, Ganzzahl-Ordnung
lbeta(x)	natürlicher Logarithmus von beta(x)
lgamma(x)	Logarithmus der Gamma-Funktion
psi(x)	PSI (Digamma) Funktion
rgamma(x)	Reziproke Gamma-Funktion
shi(x)	Hyperbolisches Sinus-Integral
si(x)	Sinus-Integral
spence(x)	Dilogarithmus
struve(v, x)	Struve-Funktion
yv(v, x)	Bessel-Funktion der Ordnung v
zeta(x, q)	Riemann Zeta-Funktion von zwei Argumenten
zetac(x)	Riemann Zeta-Funktion

Tabelle 21: Spezielle mathematische Funktionen

5.5 Prozeduren

Methoden zur direkten Manipulation von Daten, die mit dem Daten|Transformationen-Menü (Data|Transformation) korrespondieren, sind in der Tabelle 22 (Transformationen) beschrieben. Um einen Ausdruck zu evaluieren, können Sie diesen direkt dem Kommandozeileninterpreter übergeben, in der Form, wie Sie es auch im Ausdruck-Auswerten Fenster (evaluate expression window) durchführen würden. Als ein Beispiel, $S1.X = S1.X * 0.000256$ skaliert die Koordinaten der X-Achse. Die Funktionalität des Beispielpunkte (Sample points)-Menüeintrages, Abschnitt 4.5.2, S. 20, kann durch *RESTRICT*

erhalten werden.

Statement	Beschreibung	Typen	Beispiel
RUNAVG (set, npoints)	Mittelwert von <i>set</i> unter Verwendung von <i>npoints</i> Punkten berechnen	nexpr <i>npoints</i>	RUNAVG (S0, 100)
RUNMED (set, npoints)	Median von <i>set</i> unter Verwendung von <i>npoints</i> Punkten berechnen	nexpr <i>npoints</i>	RUNMED (S0, 100)
RUNMIN (set, npoints)	Minimum von <i>set</i> unter Verwendung von <i>npoints</i> Punkten berechnen	nexpr <i>npoints</i>	RUNMIN (S0, 100)
RUNMAX (set, npoints)	Maximum von <i>set</i> unter Verwendung von <i>npoints</i> Punkten berechnen	nexpr <i>npoints</i>	RUNMAX (S0, 100)
RUNSTD (set, npoints)	Standardabweichung von <i>set</i> unter Verwendung von <i>npoints</i> Punkten berechnen	nexpr <i>npoints</i>	RUNSTD (S0, 100)
INTERPOLATE (set, mesh, method, strict)	Interpoliere <i>set</i> auf ein gesamtes <i>npoints</i> unter Verwendung der Methode <i>method</i> . Das <i>strict</i> -Flag bestimmt, ob das Ergebnis innerhalb des Quellsatzes gebunden werden soll.	vexpr <i>mesh</i> , <i>method</i> : eine von LINEAR, SPLINE und ASPLINE, onoff <i>strict</i>	INTERPOLATE (S0,S1.X,ASPLINE,OFF)
HISTOGRAM (set, bins, cumulative, normalize)	Berechnet das Histogramm von <i>set</i> auf definierten <i>bins</i> . Die <i>cumulative</i> und <i>normalize</i> Flags bestimmen, ob kumulative und normalisierte (aka PDF) Histogramme berechnet werden solle. Datenpunkte werden als oberes Limit von <i>bin</i> plaziert.	vexpr <i>bins</i> , onoff <i>cumulative</i> , onoff <i>normalize</i>	HISTOGRAM (S0,MESH(0,1,11),OFF,ON)
XCOR (set1, set2, maxlag, covar)	Bestimmt die Kreuzkorrelation (oder -kovarianz, wenn das <i>covar</i> -Flag gesetzt ist) von <i>set1</i> mit <i>set2</i> mit dem maximalen Abstand <i>maxlag</i> .	nexpr <i>maxlag</i> , onoff <i>covar</i>	XCOR (S0, S0, 50, OFF)
RESTRICT (set, restriction)	Filter <i>set</i> gemäß der logischen Beschränkung <i>restriction</i> /. Der originale Satz wird überschrieben.	vexpr <i>restriction</i>	RESTRICT (S0, S0.X < 0)
RESTRICT (set, region, negate)	Filter <i>set</i> , bei dem nur Punkte, die innerhalb/außerhalb der Region <i>region</i> liegen, behalten werden. Der originale Satz wird überschrieben.	onoff <i>negate</i>	RESTRICT (S0, R1, OFF)

Tabelle 22: Transformationen

Derzeit noch nicht fertiggestellt. . .

5.6 Geräteparameter

Für die Erstellung von Ausdrucken können verschiedene Parameter über den Kommandointerpreter gesetzt werden. Diese sind in der Tabelle 23 (Geräteparameter) zusammengefaßt.

Kommando	Beschreibung
PAGE SIZE xdim, ydim PAGE RESIZE xdim, ydim	setzte Seitenabmessungen (in pp) für alle Geräte wie oben, nur das zusätzlich der aktuelle Plot entsprechend reskaliert wird
DEVICE »devname" PAGE SIZE xdim, ydim DEVICE »devname" DPI dpi	setzte Seitenabmessungen (in pp) für Gerät <i>devname</i> setzte die Auflösung in dpi (dots per pixel) für Gerät <i>devname</i>
DEVICE »devname" FONT onoff	aktiviere/deaktiviere die Verwendung von eingebauten Schriftarten für Gerät <i>devname</i>
DEVICE »devname" FONT ANTIALIASING onoff	aktiviere/deaktiviere die Verwendung von Font-Aliasing für Gerät <i>devname</i>
DEVICE »devname" OP »options«	setzte gerätespezifische Optionen (siehe 7.3 Gerätespezifische Einstellungen, S. 48)
HARDCOPY DEVICE »devname« PRINT TO »filename« PRINT TO DEVICE	setzte Gerät <i>devname</i> als aktuelles Ausdruck-Gerät setzte Druckausgabe auf <i>filename</i> (aber drucke nicht) setzte Druckausgabe auf Ausgabegerät (aber drucke nicht)

Tabelle 23: Geräteparameter

5.7 Ablaufkontrolle

Statement	Beschreibung	Typen	Beispiel
EXIT(<i>status</i>)	Normale Beendigung des Programms mit Exit-Status <i>status</i>	iexpr <i>status</i>	EXIT(0)
EXIT	Normale Beendigung des Programms; wie EXIT(0)		EXIT
HELP <i>uri</i> HELP	Öffnet das HTML-Dokument <i>url</i> Öffnet das Benutzerhandbuch	sexpr <i>url</i>	HELP »doc/FAQ.html« HELP
PRINT	Führt den Druckjob aus		PRINT
AUTOSCALE AUTOSCALE XAXES AUTOSCALE YAXES	Skaliert den Graphen Skaliert den Graphen nur auf der X-Achse Skaliert den Graphen nur auf der Y-Achse		AUTOSCALE AUTOSCALE XAXES AUTOSCALE YAXES
AUTOSCALE <i>set</i> AUTOTICKS	Skaliert den angegebenen Satz <i>set</i> Automatische Erstellung von Ticks auf allen Achsen		AUTOSCALE S0 AUTOTICKS
REDRAW	Aktualisierung der Zeichenfläche um den aktuellen Projekt-Status abzubilden		REDRAW
SLEEP <i>n</i> UPDATEALL	Schlafe für <i>n</i> Sekunden Aktualisiere die GUI (Graphen, Satzauswahlen, etc) um den aktuellen Projekt-Status abzubilden	expr <i>n</i>	SLEEP(3) UPDATEALL
SAVEALL »file« LOAD »file«	Speichere Projekt in Datei <i>file</i> Lade Projekt aus Datei <i>file</i>	sexpr <i>file</i> sexpr <i>file</i>	SAVEALL »foo.agr« LOAD »foo.agr«

Tabelle 24: Ablaufkontrolle

5.8 Deklarationen

Benutzerdefinierte Variablen werden gesetzt und verwendet entsprechend der Syntax, die in der nachfolgenden Tabelle beschrieben wird.

Statement	Beschreibung	Typen	Beispiel
DEFINE <i>var</i>	Definiert eine neue skalare Variable <i>var</i>		DEFINE myvar
DEFINE <i>vvar</i> []	Definiert eine neue vektorielle Variable der Länge 0		DEFINE myvvar[]
DEFINE <i>vvar</i> [<i>n</i>]	Definiert eine neue vektorielle Variable der Länge <i>n</i>	nexpr <i>n</i>	DEFINE myvvar[10]
CLEAR <i>var</i>	Löscht neue Variable <i>var</i> und gibt den assoziierten Speicher frei		CLEAR myvar
<i>vvar</i> LENGTH <i>n</i>	Ändert die Länge der vektoriellen Variable <i>vvar</i> auf <i>n</i>	nexpr <i>n</i>	myvvar LENGTH 25

Tabelle 25: Deklarationen

Derzeit noch nicht fertiggestellt...

5.9 Eigenschaften von Graphen

Wir teilen die Kommandos, die die Eigenschaften und das Aussehen bestimmen, in die auf, welche die Graphen direkt manipulieren und die, die das Aussehen von Graphenelementen verändern – die Parameter, die in einer Projektdatei von Grace auftauchen können.

5.9.1 Kommandooperationen

Generelle Grapherstellung und -vernichtung sowie Steuerungskommandos sind in der Tabelle 26 Graphoperationen, S. 33, aufgelistet.

Statement	Beschreibung	Typen	Beispiel
FOCUS <i>graph</i>	Setzt <i>graph</i> als aktuellen und mache ihn, falls nötig, sichtbar	graphsel <i>graph</i>	FOCUS G0
KILL <i>graph</i>	Löscht <i>graph</i>	graphsel <i>graph</i>	KILL G0
ARRANGE(<i>nrows</i> , <i>ncols</i> , <i>offset</i> , <i>hgap</i> , <i>vgap</i>)	Richtet existierende Graphen (oder füge wenn nötig weitere hinzu) so aus, das eine <i>nrows</i> mal <i>ncols</i> Matrix entsteht, mit einem Freiraum <i>offset</i> auf jeder Seite und <i>hgap</i> und <i>vgap</i> als relativen horizontalen und vertikalen Freiraum	nexpr <i>nrows</i> , <i>ncols</i> , expr <i>offset</i> , <i>hgap</i> , <i>vgap</i>	ARRANGE(2, 2, 0.1, 0.15, 0.2)
ARRANGE(<i>nrows</i> , <i>ncols</i> , <i>offset</i> , <i>hgap</i> , <i>vgap</i> , <i>hwinv</i> , <i>hinw</i> , <i>vinv</i>)	wie zuvor, mit zusätzlicher Angabe der Flags <i>hwinv</i> , <i>hinw</i> und <i>vinv</i> , um die Reihenfolge der Befüllung der Matrix zu ändern	nexpr <i>nrows</i> , <i>ncols</i> , expr <i>offset</i> , <i>hgap</i> , <i>vgap</i> , onoff <i>hwinv</i> , <i>hinw</i> , <i>vinv</i>	ARRANGE(2, 2, 0.1, 0.15, 0.2, ON, OFF, ON)
ARRANGE(<i>nrows</i> , <i>ncols</i> , <i>offset</i> , <i>hgap</i> , <i>vgap</i> , <i>hwinv</i> , <i>hinw</i> , <i>vinv</i> , <i>snake</i>)	wie zuvor, mit zusätzlichem <i>snake</i> -Flag, um die Matrix in Schlangenform zu füllen	nexpr <i>nrows</i> , <i>ncols</i> , expr <i>offset</i> , <i>hgap</i> , <i>vgap</i> , onoff <i>hwinv</i> , <i>hinw</i> , <i>vinv</i> , <i>snake</i>	ARRANGE(2, 2, 0.1, 0.15, 0.2, ON, OFF, ON, ON)

Tabelle 26: Graphenoperationen

5.9.2 Parametereinstellungen

Einstellungen des aktiven Graphen und seines Typs können über die Kommandos durchgeführt werden, die in der Tabelle 27 Parameter der Graphenauswahl, S. 34, wiedergegeben sind.

Statement	Beschreibung	Typen	Beispiel
WITH <i>graph</i> TYPE <i>type</i>	Setzte <i>graph</i> als aktiven Setzte <i>type</i> des aktiven Graphen	graphsel <i>graph</i> graphtype <i>type</i>	WITH G0 TYPE XY
<i>graph</i> onoff	(De-)Aktiviere den Graphen <i>graph</i>	graphsel <i>graph</i> , onoff	G0 ON
<i>graph</i> HIDDEN onoff	Verstecke den Graphen <i>graph</i>	graphsel <i>graph</i> , onoff	G1 HIDDEN TRUE
<i>graph</i> TYPE <i>type</i>	Setzte <i>type</i> des Graphen <i>graph</i>	graphsel <i>graph</i> , graphtype <i>type</i>	G0 TYPE XYDY

Tabelle 27: Parameter der Graphenauswahl

Der Achsen-Bereich und die Skalierung des aktuellen Graphen sowie die Platzierung innerhalb des Plotbereiches können über die Kommandos in der Tabelle 29 Achsen-Parameter, S. 35, eingestellt werden.

Statement	Beschreibung	Typen	Beispiel
WORLD XMIN <i>xmin</i>	Setzt den minimalen Wert der X-Achse des aktuellen Graphen auf <i>xmin</i>	expr <i>xmin</i>	WORLD XMIN -10
WORLD XMAX <i>xmax</i>	Setzt den maximalen Wert der X-Achse des aktuellen Graphen auf <i>xmax</i>	expr <i>xmax</i>	WORLD XMAX 22.5
WORLD YMIN <i>ymin</i>	Setzt den minimalen Wert der Y-Achse des aktuellen Graphen auf <i>ymin</i>	expr <i>ymin</i>	WORLD YMIN 0
WORLD YMAX <i>ymax</i>	Setzt den maximalen Wert der Y-Achse des aktuellen Graphen auf <i>ymax</i>	expr <i>ymax</i>	WORLD YMAX 0
VIEW XMIN <i>xmin</i>	Setzt die linke Seite der X-Achse des aktuellen Graphen auf $x=xmin$ innerhalb des Viewports	expr <i>xmin</i>	VIEW XMIN .2
VIEW XMAX <i>xmax</i>	Setzt die rechte Seite der X-Achse des aktuellen Graphen auf $x=xmax$ innerhalb des Viewports	expr <i>xmax</i>	VIEW XMAX 1.0
VIEW YMIN <i>ymin</i>	Setzt die untere Seite der Y-Achse des aktuellen Graphen auf $y=ymin$ innerhalb des Viewports	expr <i>ymin</i>	VIEW YMIN .25
VIEW YMAX <i>ymax</i>	Setzt die obere Seite der Y-Achse des aktuellen Graphen auf $y=yymax$ innerhalb des Viewports	expr <i>yymax</i>	VIEW YMAX .75

Fortsetzung auf der nächsten Seite...

Statement	Beschreibung	Typen	Beispiel
VIEW <i>xmin, ymin, xmax, ymax</i> XAXES SCALE <i>type</i>	Setzte den Viewport des Graphen Setzte die Skalierung der X-Achse auf <i>type</i>	<i>expr xmin, ymin, xmax, ymax</i> NORMAL LOGARITHMIC RECIPROCAL	VIEW 0.15, 0.15, 1.15, 0.85 XAXES SCALE NORMAL
YAXES SCALE <i>type</i>	Setzte die Skalierung der Y-Achse auf <i>type</i>	NORMAL LOGARITHMIC RECIPROCAL	YAXES SCALE LOGARITHMIC
XAXES INVERT onoff	Wenn ON, so werden <i>xmin</i> nach <i>xmax</i> von rechts nach links gezeichnet	onoff	XAXES INVERT OFF
YAXES INVERT onoff	Wenn ON, so werden <i>ymin</i> nach <i>ymax</i> von oben nach unten gezeichnet	onoff	YAXES INVERT OFF
AUTOSCALE ON- READ <i>type</i>	Setzte automatische Skalierung beim Einlesen auf <i>type</i>	<i>type</i> : einer von NONE, XAXES, YAXES, XYAXES	AUTOSCALE ONREAD NONE

Tabelle 29: Achsen-Parameter

Die Befehle, die das Erscheinungsbild und den Textinhalt von Titeln und Legenden beeinflussen, sind in der Tabelle 31 Titel und Legenden, S. 36, wiedergegeben.

Statement	Beschreibung	Typen	Beispiel
TITLE <i>title</i>	Setzt den Titel des aktuellen Graphen	<i>sexpr title</i>	TITLE »Foo«
TITLE FONT <i>font</i>	Auswahl der Schriftart für den Titel	<i>fontsel font</i>	TITLE FONT 1
TITLE SIZE <i>size</i>	Auswahl der Schriftgröße für den Titel	<i>expr size</i>	TITLE SIZE 1.5
TITLE COLOR <i>color</i>	Auswahl der Schriftfarbe für den Titel	<i>colorsel color</i>	TITLE COLOR 1
SUBTITLE <i>subtitle</i>	Setzt den Untertitel des aktuellen Graphen	<i>sexpr subtitle</i>	SUBTITLE »Bar«
SUBTITLE FONT <i>font</i>	Auswahl der Schriftart für den Untertitel	<i>fontsel font</i>	SUBTITLE FONT »Times-Italic«
SUBTITLE SIZE <i>size</i>	Auswahl der Schriftgröße für den Untertitel	<i>expr size</i>	SUBTITLE SIZE .60
SUBTITLE COLOR <i>color</i>	Auswahl der Schriftfarbe für den Untertitel	<i>colorsel color</i>	SUBTITLE COLOR »blue«
LEGEND onoff	Aktivierung und Deaktivierung der Legende	onoff	LEGEND ON
LEGEND LOCTYPE <i>type</i>	Positionierung der Legende in <i>type</i> -Koordinaten	WORLD VIEW	LEGEND LOCTYPE WORLD
LEGEND <i>xloc, yloc</i>	Position der Legendendbox (obere, linke Ecke)	<i>expr xloc, yloc</i>	LEGEND .5,.75
LEGEND FONT <i>font</i>	Auswahl der Schriftart für die Legende	<i>fontsel font</i>	LEGEND FONT »Helvetica«

Fortsetzung auf der nächsten Seite. . .

Statement	Beschreibung	Typen	Beispiel
LEGEND CHAR SIZE <i>size</i>	Auswahl der Schriftgröße für die Legende (1 ist normal)	<i>expr size</i>	LEGEND CHAR SIZE .30
LEGEND COLOR <i>color</i>	Auswahl der Schriftfarbe für die Legende	<i>colorsel color</i>	LEGEND COLOR 1
LEGEND VGAP <i>gap</i>	Setzt den vertikalen Zwischenraum zwischen Legendeneinträgen	<i>nexpr gap</i>	LEGEND VGAP 1
LEGEND HGAP <i>gap</i>	Setzt den horizontalen Zwischenraum zwischen Symbol und Beschreibung	<i>nexpr gap</i>	LEGEND HGAP 4
LEGEND LENGTH <i>length</i>	Setzt die Länge der Legende	<i>nexpr length</i>	LEGEND LENGTH 5
LEGEND INVERT <i>onoff</i>	Bestimmt den Zusammenhang zwischen der Satzreihenfolge und der Reihenfolge der Legendenbezeichner	<i>onoff</i>	LEGEND INVERT true
LEGEND BOX <i>onoff</i>	Bestimmt, ob ein Kasten um die Legende gezeichnet werden soll	<i>onoff</i>	LEGEND BOX off
LEGEND BOX COLOR <i>color</i>	Setzt die Farbe des Legendenkastens	<i>colorsel color</i>	LEGEND BOX COLOR 1
LEGEND BOX PATTERN <i>pattern</i>	Setzt das Muster des Legendenkastens	<i>patternsel pattern</i>	LEGEND BOX PATTERN 2
LEGEND BOX LIFESTYLE <i>style</i>	Setzt den Linienstyle des Legendenkastens	<i>nexpr style</i>	LEGEND BOX LIFESTYLE 1
LEGEND BOX LINEWIDTH <i>width</i>	Setzt die Linienbreite des Legendenkastens	<i>nexpr width</i>	LEGEND BOX LINEWIDTH 2
LEGEND BOX FILL <i>onoff</i>	Bestimmt, ob der Legendenkasten eine Füllung erhalten soll	<i>onoff</i>	LEGEND BOX FILL false
LEGEND BOX FILL COLOR <i>color</i>	Setzt die Farbe der Füllung des Legendenkastens	<i>colorsel color</i>	LEGEND BOX FILL COLOR 3
LEGEND BOX FILL PATTERN <i>pattern</i>	Setzt das Muster der Füllung des Legendenkastens	<i>patternsel pattern</i>	LEGEND BOX FILL PATTERN 1

Tabelle 31: Titel und Legenden

Derzeit noch nicht fertiggestellt. . .

5.10 Eigenschaften von Datensätzen

Erneut, wie bei den Graphen, trennen wir die Parser-Kommandos, die die Daten innerhalb eines Satzes manipulieren von denen, die Parameter beschreiben – Elemente, die in einer Projektdatei gespeichert werden können.

5.10.1 Kommandos

Operationen für die Satz-IO sind in der Tabelle 32 zusammengefaßt. (Beachten Sie, daß diese derzeit nicht komplett ist und derzeit nur die Eingabekommandos auflistet.)

Statement	Beschreibung	Typen	Beispiel
READ »file«	Einlesen von <i>file</i> als ein einzelner Satz	sexpr <i>file</i>	READ »foo.dat«
READ <i>settype</i> »file«	Einlesen von <i>file</i> als ein einzelner Satz vom Typ <i>settype</i>	xytype <i>settype</i> , sexpr <i>file</i>	READ xydy »bar.dat«
READ NXY »file«	Einlesen von <i>file</i> als NXY-Daten	sexpr <i>file</i>	READ NXY »gad.dat«
READ BLOCK »file«	Einlesen von <i>file</i> als Block-Daten	sexpr <i>file</i>	READ BLOCK »zooks.dat«
BLOCK <i>settype columns</i>	Erzeugt einen Datensatz vom Typ <i>settype</i> unter der Verwendung von Spalten <i>columns</i> aus der aktuellen Block-Daten Datei	xytype <i>settype</i> , sexpr <i>columns</i>	BLOCK xydxdy »0:2:1:3«
WRITE <i>set</i>	Schreibe <i>set</i> auf stdout	setsel <i>set</i>	WRITE G0.S1
WRITE <i>set</i> FORMAT »formatstring«	Schreibe <i>set</i> auf stdout unter Verwendung der Formatspezifikation <i>formatstring</i>	setsel <i>set</i> , sexpr <i>formatstring</i>	WRITE G0.S1 FORMAT »%18.8g«
WRITE <i>set</i> FILE »file«	Schreibe <i>set</i> in Datei <i>file</i>	setsel <i>set</i> , sexpr <i>file</i>	WRITE G0.S1 FILE »data.dat«
WRITE <i>set</i> FILE »file« FORMAT »formatstring«	Schreibe <i>set</i> in Datei <i>file</i> unter Verwendung der Formatspezifikation <i>formatstring</i>	setsel <i>set</i> , sexpr <i>file</i> , sexpr <i>formatstring</i>	WRITE G0.S1 FILE »data.dat« FORMAT »%18.8g«

Tabelle 32: Satzeingabe, -ausgabe und -erstellung

Die Parser-Kommandos, die analog zu denen im *Daten|Datensatzoperationen*-Dialog (Data|Data set operations) sind, sind in der Tabelle 33 wiedergegeben.

Statement	Beschreibung	Typen	Beispiel
COPY <i>src</i> TO <i>dest</i>	Kopiert <i>src</i> nach <i>dest</i>	setsel <i>src,dest</i>	COPY S0 TO S1
MOVE <i>src</i> TO <i>dest</i>	Verschiebt <i>src</i> nach <i>dest</i>	setsel <i>src,dest</i>	MOVE G0.S0 TO G1.S0
SWAP <i>src</i> AND <i>dest</i>	Vertauscht <i>src</i> und <i>dest</i>	setsel <i>src,dest</i>	SWAP G0.S0 AND G0.S1
KILL <i>set</i>	Löscht <i>set</i>	setsel <i>set</i>	KILL G0.S0

Tabelle 33: Satz-Operationen

Derzeit noch nicht fertiggestellt...

5.10.2 Parametereinstellungen

Derzeit noch nicht geschrieben...

6. Weiterführende Themen

6.1 Schriften

Für alle Geräte verwendet Grace Type1-Schriftarten. Sowohl PFA (ASCII) und PFB (Binäre) Formate können dabei verwendet werden.

6.1.1 Schriftartenkonfiguration

In Grace wird die Schriftartenkonfiguration über die Datei *fonts/FontDataBase* vorgenommen. Die erste Zeile enthält eine positive Ganzzahl, die die Anzahl der deklarierten Schriften in dieser Datei

spezifiziert. Alle weiteren Zeilen enthalten die Deklaration von jeweils einer Schrift, bestehend aus drei Feldern:

- Der Schriftartenname. Dieser Name wird in den Steuerelementen für die Schriftartauswahl angezeigt. Außerdem wird Backend-Geräten, die eingebaute Schriftarten haben, dieser Name als Bezeichner für die Schriftart übergeben.
- Ersetzungsschriftart. Grace wird versuchen, diese Schrift zu verwenden, wenn die primäre Schriftart nicht gefunden wurde.
- Schriftarten-Dateiname. Die Datei mit den Schriftart Outline-Daten.

Hier folgt die standardmäßig verwendete FontDataBase-Datei:

```

14
Times-Roman          Times-Roman          n0210031.pfb
Times-Italic         Times-Italic         n0210231.pfb
Times-Bold           Times-Bold           n0210041.pfb
Times-BoldItalic     Times-BoldItalic     n0210241.pfb
Helvetica            Helvetica            n0190031.pfb
Helvetica-Oblique   Helvetica-Oblique   n0190231.pfb
Helvetica-Bold       Helvetica-Bold       n0190041.pfb
Helvetica-BoldOblique Helvetica-BoldOblique n0190241.pfb
Courier              Courier              n0220031.pfb
Courier-Oblique      Courier-Oblique      n0220231.pfb
Courier-Bold         Courier-Bold         n0220041.pfb
Courier-BoldOblique  Courier-BoldOblique  n0220241.pfb
Symbol              Symbol              s0500001.pfb
ZapfDingbats        ZapfDingbats        d0500001.pfb

```

6.1.2 Schriftarten-Datendateien

Für die Text rasterung werden drei Dateitypen verwendet.

- `.pfa-`/`.pfb`-Dateien: Diese beinhalten die Outline-Beschreibungen für die Zeichen. Es wird angenommen, daß sich diese im `fonts/type1`-Verzeichnis befinden; dies sind die in der `FontDataBase`-Konfigurationsdatei angegebenen Namen.
- `.afm`-Dateien: Diese beinhalten hochpräzise Beschreibungen der Schriftmetriken sowie einige weiterführende Informationen, wie Kerning- und Ligaturen-Informationen für eine bestimmte Schriftart. Es wird angenommen, daß der Dateiname einer Schriftmetriken-Datei den gleichen Basisnamen wie die entsprechende Datei für die Outline-Beschreibung hat, jedoch mit der `.afm`-Erweiterung; für die Metrik-Dateien wird ebenfalls erwartet, daß diese im `fonts/type1`-Verzeichnis gefunden werden können.
- `.enc`-Dateien: Diese beinhalten Encoding-Felder in einer speziellen, jedoch einfachen Form. Diese werden nur verwendet, wenn jemand ein spezielles Encoding laden möchte, um eine Schriftart zu Re-Encoden. Ihr Platz ist in `fonts/enc`.

6.1.3 Benutzerspezifische Schriftarten

Es ist möglich, in Grace benutzerspezifische Schriftarten zu verwenden. Häufig muß man Extraschriftarten zum Zwecke der Lokalisierung verwenden. Für viele europäischen Sprachen sollten die mit Grace gelieferten Standardschriftarten alle erforderlichen Zeichen enthalten, jedoch das Encoding müßte gegebenenfalls angepaßt werden. Dies kann durchgeführt werden, indem eine `Default.enc`-Datei mit dem richtigen Encoding-Schema in das `fonts/enc`-Verzeichnis abgelegt wird. Grace wird mit einigen Encoding-Dateien in diesem Verzeichnis ausgeliefert; weitere können einfach im Internet gefunden werden (Existiert die `Default.enc`-Datei nicht, so wird *isoLatin1*-Encoding verwendet). Beachten Sie, daß für Schriftarten, die ein Encoding-Schema in sich selbst mitbringen (wie u. a. die Symbol-Schriftart und viele nationalisierten Schriftarten), das Standard-Encoding ignoriert wird.

Wenn Sie wirklich Extraschriftarten verwenden müssen, sollten Sie die `FontDataBase`-Datei entsprechend modifizieren, unter Berücksichtigung des o. g. Formates. Jedoch, wenn Sie Grace-Projektdateien

mit anderen Benutzern austauschen, die nicht diese Extraschriftarten konfiguriert haben, ist es wichtig, vernünftige Fall-Back-Schriftarten zu definieren.

Zum Beispiel, lassen Sie uns annehmen, ich möchte hebräische Schriftarten verwenden, und die Konfigurationsdatei enthält Zeilen wie diese:

```
...
Courier-Hebrew          Courier          courh____.pfa
Courier-Hebrew-Oblique  Courier-Oblique  courho____.pfa
...
```

Mein Kollege, der in Rußland leben, verwendet kyrillische Schriftarten in Grace, die wie folgt konfiguriert sind:

```
...
Cronix-Courier          Courier          croxc.pfb
Cronix-Courier-Oblique  Courier-Oblique  croxco.pfb
...
```

Die Informationen zum Schriftarten-Mapping (Schriftartenname ↔ Ersetzungsschrift) wird in Grace Projekt-Datei gespeichert. Vorausgesetzt, daß alle lokalisierten Schriftarten die englischen Zeichen im unteren Teil der ASCII-Tabelle nicht modifiziert haben, kann ich meinen Freunden Dateien senden (ohne die hebräischen Zeichen natürlich) und sicher sein, daß sie richtig auf seinem Computer angezeigt werden.

So, mit richtig konfigurierten nationalen Schriftarten, können Sie lokalisierte Anmerkungen für Plot erstellen, die nur für den internen Gebrauch in Ihrer Institution gedacht sind, während Sie die Möglichkeit haben, Dateien mit Kollegen aus dem Ausland auszutauschen. Leute, die dies jemals mit Anwendungen wie MS Office probiert haben, lernen diese Flexibilität zu schätzen :-).

6.2 Interaktion mit anderen Anwendungen

6.2.1 Verwendung von Pipes

6.2.2 Verwendung der `grace_np`-Bibliothek

Die `grace_np`-Bibliothek ist eine Reihe von kompilierten Funktionen, die es erlauben, einen Subprozeß von Grace aus Ihrer C- oder Fortran-Anwendung heraus aufzurufen und zu steuern. Zur Verfügung werden Funktionen gestellt, um einen Subprozeß zu starten, Ihm Kommandos zu senden, ihn zu stoppen und sich davon zu lösen. Eine Liste der Funktionen ist in den Tabelle 34 und 35 zu finden.

Es gibt keine Fortran-Entsprechung für die `GracePrintf`-Funktion, hier müssen Sie alle Daten und Kommandos selbst formatieren, bevor diese mittels `GraceCommandF` gesendet werden.

Der Grace Subprozeß wartet auf von Ihnen gesendete Kommandos und interpretiert diese, als wenn sie in einer Batch-Datei angegeben wurde. Sie können jedes beliebige (gültige) Kommando senden (Neuzeichnen, Autoskalierung, ...). Wenn Sie Daten senden wollen, so sollten Sie diese in einem Kommando wie »g0.s0 point 3.5, 4.2« angeben.

Abgesehen von der Tatsache, daß die Daten, die über eine anonyme Pipe gesendet werden, überwacht werden, ist der Grace Subprozeß ein normaler Prozeß. Sie können mit ihm über die GUI interagieren. Beachten Sie, daß kein Fehler zum Eltern-Prozeß zurückgesendet werden kann. Wenn Ihre Anwendung fehlerhafte Kommandos sendet, so wird ein Fehler-Popup durch den Subprozeß angezeigt.

Wenn Sie den Subprozeß beenden, während der Eltern-Prozeß ihn noch weiterhin benutzt, so wird die abgebrochene Pipe erkannt. Ein Fehlerwert wird bei jedem nachfolgenden Aufruf der Bibliothek zurückgegeben (aber Sie können einen neuen Prozeß starten, wenn Sie diese Situation abfangen möchten).

Ein Beispiel für die Verwendung der Bibliothek ist im Beispiel 1, S. 41 wiedergegeben, Sie finden dieses Programm in der Distribution.

Funktion	Beschreibung
<code>int GraceOpenVA(char* exe, int buf_size, ...)</code>	Ausführung der Grace-Anwendung <i>exe</i> , ein Kommunikationskanal wird mit einer Buffer-Größe von <i>buf_size</i> Bytes für die Datenpufferung verwendet. Die nachfolgende NULL-terminierte Liste von Optionen sind Kommandozeilenargumente, die an den Grace-Prozeß übergeben werden.
<code>int GraceOpen(int buf_size)</code>	Equivalent zu <code>GraceOpenVA(»xmgrace«, buf_size, »-nosafe«, »-noask«, NULL)</code>
<code>int GraceIsOpen(void)</code>	Prüft, ob der Grace Subprozeß derzeit verbunden ist
<code>int GraceClose(void)</code>	Schließt den Kommunikationskanal und beendet den Grace Subprozeß
<code>int GraceClosePipe(void)</code>	Schließt den Kommunikationskanal und läßt den Grace Subprozeß alleine
<code>int GraceFlush(void)</code>	Alle verbleibenden Daten im Puffer werden entfernt
<code>int GracePrintf(const char* format,...)</code>	Formatiert ein Kommando und sendet ihn zum Grace Subprozeß
<code>int GraceCommand(const char* cmd)</code>	Sendet ein bereits formiertes Kommando zum Grace Subprozeß
<code>GraceErrorFunktionType</code> <code>GraceRegisterErrorFunction(GraceErrorFunktionType f)</code>	Registrierung einer benutzerdefinierten Funktion <i>f</i> , um Bibliotheksfehler anzuzeigen

Tabelle 34: grace_np-Bibliothek: C-Funktionen

Funktion	Beschreibung
<code>integer GraceOpenF(integer buf_size)</code>	Ausführung eines Grace Subprozesses und öffnen eines Kommunikationskanals zu ihm
<code>integer GraceIsOpenF(void)</code>	Prüft, ob der Grace Subprozeß derzeit verbunden ist
<code>integer GraceCloseF(void)</code>	Schließt den Kommunikationskanal und beendet den Grace Subprozeß
<code>integer GraceClosePipeF(void)</code>	Schließt den Kommunikationskanal und läßt den Grace Subprozeß alleine
<code>integer GraceFlushF(void)</code>	Alle verbleibenden Daten im Puffer werden entfernt
<code>integer GraceCommandF(character*(*) cmd)</code>	Sendet ein bereits formiertes Kommando zum Grace Subprozeß
<code>GraceFortranFunktionType</code> <code>GraceRegisterErrorFunctionF(GraceFortranFunktionType f)</code>	Registrierung einer benutzerdefinierten Funktion <i>f</i> , um Bibliotheksfehler anzuzeigen

Tabelle 35: grace_np-Bibliothek: Fortran F77-Funktionen

Beispiel 1: *grace_np* Beispiel

```
<code c|grace_np Beispiel>
#include <stdlib.h>
#include <stdio.h>
#include <unistd.h>
#include <grace_np.h>

#ifdef EXIT_SUCCESS
# define EXIT_SUCCESS 0
#endif

#ifdef EXIT_FAILURE
# define EXIT_FAILURE -1
#endif

void my_error_function(const char *msg)
{
    fprintf(stderr, ">library message: \"%s\\n\"", msg);
}

int
main(int argc, char* argv[])
{
    int i;

    GraceRegisterErrorFunction(my_error_function);

    /* Start Grace with a buffer size of 2048 and open the pipe */
    if (GraceOpen(2048) == -1) {
        fprintf(stderr, ">Can't run Grace. \\n");
        exit(EXIT_FAILURE);
    }

    /* Send some initialization commands to Grace */
    GracePrintf("world xmax 100");
    GracePrintf("world ymax 10000");
    GracePrintf("xaxis tick major 20");
    GracePrintf("xaxis tick minor 10");
    GracePrintf("yaxis tick major 2000");
    GracePrintf("yaxis tick minor 1000");
    GracePrintf("s0 on");
    GracePrintf("s0 symbol 1");
    GracePrintf("s0 symbol size 0.3");
    GracePrintf("s0 symbol fill pattern 1");
    GracePrintf("s1 on");
    GracePrintf("s1 symbol 1");
    GracePrintf("s1 symbol size 0.3");
    GracePrintf("s1 symbol fill pattern 1");

    /* Display sample data */
    for (i = 1; i <= 100 && GraceIsOpen(); i++) {
        GracePrintf("g0.s0 point %d, %d", i, i);
        GracePrintf("g0.s1 point %d, %d", i, i * i);
        /* Update the Grace display after every ten steps */
        if (i % 10 == 0) {
            GracePrintf("redraw");
            /* Wait a second, just to simulate some time needed for
               calculations. Your real application shouldn't wait. */
            sleep(1);
        }
    }

    if (GraceIsOpen()) {
```

```
/* Tell Grace to save the data */
GracePrintf("saveall \"sample.agr\"");

/* Flush the output buffer and close Grace */
GraceClose();

/* We are done */
exit(EXIT_SUCCESS);
} else {
    exit(EXIT_FAILURE);
}
}
```

Um dieses Programm zu kompilieren, geben Sie

```
cc example.c -lgrace_np
```

ein. Wenn Grace nicht richtig installiert wurde, kann es notwendig sein, dem Kompiler die Include- und Bibliothekspfade explizit angeben zu müssen, z. B.

```
cc -I/usr/local/grace/include example.c -L/usr/local/grace/lib -lgrace_np
```

6.3 FFTW Einstellungen

Wenn die FFTW-Unterstützung einkompiliert wurde, prüft Grace zwei Umgebungsvariablen, um zu entscheiden, wie mit der FFTW 'wisdom'-Unterstützung umgegangen werden soll. Zuerst einer schnelle Zusammenfassung, was dies überhaupt ist. Das FFTW-Paket besitzt die Fähigkeit, die am effizienteste Faktorisierung eines Satzes zu bestimmen – dies ist anpassungsfähig –, um die schnellste Berechnung durchzuführen. Es kann diese Faktorisierung als 'wisdom' speichern, so daß eine Transformationen mit einer gegebenen Größe wiederholt werden kann, ohne das eine Neuanpassung notwendig ist. Die guten Nachrichten sind, daß dies anscheinend sehr gut arbeitet. Die schlechten Nachrichten sind, wenn das erste Mal eine Transformation zu eine gegebenen Größe berechnet werden soll, die nicht ein Subvielfaches einem bereits bekannten ist, wird hierfür eine *lange* Zeit benötigt (Sekunden bis zu Minuten).

Die erste Umgebungsvariable ist `GRACE_FFTW_WISDOM_FILE`. Ist diese auf einen Dateinamen gesetzt, wobei die Datei gelesen und geschrieben werden kann (z. B. `$HOME/.grace_fftw_wisdom`), dann erstellt Grace (falls benötigt) die Datei automatisch und wird von Grace aufrechterhalten. Ist diese Datei nur lesbar (read-only), so wird sie eingelesen, jedoch werden keine Aktualisierungen mit neuen 'wisdoms' vorgenommen. Wenn das Symbol `GRACE_FFTW_WISDOM_FILE` entweder nicht existiert oder zu einer leeren Zeichenkette ausgewertet wird, wird Grace keine 'wisdoms' verwenden und anstelle den FFTW-Kalkulator (das `FFTW_ESTIMATE`-Flag wird zum Planer gesendet) verwenden, um eine gute Faktorisierung zu erraten, anstatt diese anpassungsfähig zu bestimmen.

Die zweite Variable ist `GRACE_FFTW_RAM_WISDOM`. Ist diese Variable definiert als nicht-null und die `GRACE_FFTW_WISDOM_FILE`-Variable ist nicht definiert (oder enthält eine leere Zeichenkette), so wird Grace 'wisdoms' intern verwenden, jedoch verwaltet keinen persistenten Zwischenspeicher für diese. Dies wird sich in sehr langsamen Ausführungszeiten bemerkbar machen, wenn zum ersten Mal nach dem Starten von Grace eine Transformation ausgeführt wird, aber in sehr schnellen Wiederholungen. Ich bin mir nicht sicher, warum jemand 'wisdoms' verwenden möchte, ohne sie auf die Festplatte zu schreiben, aber wenn Sie es möchten, können Sie dieses Flag dazu verwenden, um dies zu aktivieren.

6.4 DL-Module

Grace kann auf externe Funktionen zugreifen, die sowohl in System- als auch in Third-Party-Bibliotheken (sogenannte shared libraries) verfügbar sind sowie in Modulen, die speziell für die Verwendung mit Grace kompiliert wurden.

6.4.1 Funktionstypen

Man muß jedoch sicherstellen, daß die externe Funktion von einem Typ ist, der durch die Grace-Typen unterstützt wird:

Grace Type	Beschreibung
f_of_i	eine Funktion mit 1 <i>int</i> Variable
f_of_d	eine Funktion mit 1 <i>double</i> Variable
f_of_nn	eine Funktion mit 2 <i>int</i> Parametern
f_of_nd	eine Funktion mit 1 <i>int</i> Parameter und 1 <i>double</i> Variablen
f_of_dd	eine Funktion mit 2 <i>double</i> Variablen
f_of_nnd	eine Funktion mit 2 <i>int</i> Parametern und 1 <i>double</i> Variablen
f_of_ppd	eine Funktion mit 2 <i>double</i> Parametern und 1 <i>double</i> Variablen
f_of_pppd	eine Funktion mit 3 <i>double</i> Parametern und 1 <i>double</i> Variablen
f_of_ppppd	eine Funktion mit 4 <i>double</i> Parametern und 1 <i>double</i> Variablen
f_of_pppppd	eine Funktion mit 5 <i>double</i> Parametern und 1 <i>double</i> Variablen

Tabelle 36: Funktionstypen

Für den Rückgabewert von Funktionen wird angenommen, daß diese vom Type *double* sind.

Beachten Sie, daß es keinen Unterschied aus der Sicht von Funktions-Prototypen zwischen Parametern und Variablen gibt; der Unterschied wird durch die unterschiedliche Verarbeitungsweise von Grace bestimmt – ein Versucht, einen Vektor-Ausdruck als Parameter-Argument zu verwenden wird in einen Syntaxfehler hinauslaufen.

Lassen Sie uns über einige Beispiele nachdenken.

6.4.2 Beispiele

Warnung: die Beispiele, die unten zur Verfügung gestellt sind (Pfade und Compiler-Flags) sind für Linux/ELF mit gcc gültig. Auf anderen Betriebssystemen oder Compilerumgebungen könnte Sie sich auf die Compiler/Linker Handbücher beziehen oder Fragen einen Guru nach der richtigen Syntax.

Beispiel 2

Nehmen wir an, daß die Funktion $\text{pow}(x, y)$ aus der `Un*x` Mathematikbibliothek (`libm`) verwendet werden soll. Natürlich können Sie auch den `»^«` Operator, der in der Grace-Sprache definiert ist, verwenden, aber hier wollen wir aufgrund der Beispiels die Funktion direkt verwenden.

Der Befehl, um diesen in Grace zugänglich zu machen ist:

```
USE "pow" TYPE f_of_dd FROM "/usr/lib/libm.so"
```

Versuchen Sie, $y = \text{pow}(x, 2)$ und $y = x^2$ in einem Graph zu plotten (z. B. über die Verwendung von `»create new -> Formula«` aus jeder Satzauswahl (siehe 4.1.4, S. 11) (set selector)) und vergleichen Sie die Ergebnisse.

Beispiel 3

Nun lassen Sie uns versuchen, eine Funktion selber zu schreiben. Wir definieren eine Funktion `my_function`, die einfach ihr (zweites) Argument mit dem ersten Parameter, einer Ganzzahl (`integer`), multipliziert.

In einem Texteditor wird der nachfolgende C-Code eingegeben und als `my_func.c` gespeichert:

```
double my_function (int n, double x)
{
    double retval;
    retval = (double) n * x;
    return (retval);
}
```

OK, nun kann dies kompiliert werden:

```
$gcc -c -fPIC my_func.c
$gcc -shared my_func.o -o /tmp/my_func.so
```

(Sie können die Bibliothek auch `strip`, um etwas Speicherplatz auf der Festplatte zu sparen):

```
$strip /tmp/my_func.so
```

Das ist alles! Nun sind wir bereit, diese Funktion als »myf« in Grace sichtbar zu machen – wir sind zu faul, die sehr lange Zeichenkette »my_function« häufig zu tippen.

```
USE "my_function" TYPE f_of_nd FROM "/tmp/my_func.so" ALIAS "myf"
```

Beispiel 4

Ein ersteres Beispiel. Es gibt eine spezielle Bibliothek eines Drittanbieters auf Ihrem System, die eine sehr wichtige Funktion beinhaltet, die Sie in Grace verwenden möchten und schwierig ist, selbst zu implementieren. Aber der Funktionsprototyp ist *nicht* in der Form, wie die in Grace vordefinierten. Die Lösung hierfür, einfach einen Funktionswrapper zu schreiben. Nachfolgend wird gezeigt, wie dies durchgeführt werden kann.

Angenommen, der Name der Bibliothek ist »special_lib« und die Funktion, die Sie verwenden möchten, heißt »special_func«. Nach dem Handbuch der Bibliothek wird diese Funktion über `void special_func(double *input, double *output, int parameter)` angesprochen. Der Wrapper könnte dann wie folgt aussehen:

```
double my_wrapper(int n, double x)
{
    extern void special_func(double *x, double*y, int n);
    double retval;
    (void) special_func(&x, &retval, n);
    return retval;
}
```

Kompilieren Sie es:

```
$gcc -c -fPIC my_wrap.c
$gcc -shared my_wrap.o -o /tmp/my_wrap.so -lspecial_lib -lblas
$strip /tmp/my_wrap.so
```

Beachten Sie, das Ich `-lblas` in der Annahme hinzugefügt habe, daß die `special_lib`-Bibliothek einige Funktionen aus BLAS verwendet. Generell müssen *alle* Bibliotheken hinzugefügt werden, von denen Ihr Modul abhängt (und alle Bibliotheken, die von den Bibliotheken verwendet werden usw.), als ob Sie eine ausführbare Datei kompilieren möchten.

Fein, machen Sie die neue Funktion in Grace bekannt:

```
USE "my_wrapper" TYPE f_of_nd FROM "/tmp/my_wrap.so" ALIAS "special_func"
```

Mit dieser Sichtbarmachung können wir in Grace auf die Funktion mit ihrem originalen Namen zugreifen (aufgrund der Alias-Spezifikation).

Beispiel 5

Ein Beispiel für die Verwendung von Fortran-Modulen.

Hier möchten wir die gleiche Funktionalität wie in Beispiel 2 erreichen, jedoch mit der Hilfe von F77.

```
DOUBLE PRECISION FUNCTION MYFUNC (N, X)
IMPLICIT NONE
INTEGER N
DOUBLE PRECISION X
C
MYFUNC = N * X
C
RETURN
END
```

Im Unterschied zu C besteht hier kein Weg, eine solche Funktion direkt aus Grace aus aufzurufen – das Problem besteht darin, das Fortran alle Argumente einer Funktion (oder Subroutine) über Referenzen übergeben werden. Also brauchen wir hierfür einen Wrapper:

```
double myfunc_wrapper(int n, double x)
{
    extern double myfunc_(int *, double *);
    double retval;
    retval = myfunc_(&n, &x);
    return (retval);
}
```

Beachten Sie, daß viele der F77-Compiler standardmäßig Unterstriche zum Funktionsnamen hinzufügen und alle Namen in Kleinbuchstaben konvertieren, folglich verweise ich auf die Fortran-Funktion MYFUNC in meinem C-Wrapper über `myfunc_`, jedoch kann das in Ihrem Fall unterschiedlich sein!

Lassen Sie uns alles Kompilieren:

```
$g77 -c -fPIC myfunc.f
$gcc -c -fPIC myfunc_wrap.c
$gcc -shared myfunc.o myfunc_wrap.o -o /tmp/myfunc.so -lf2c -lm
$strip /tmp/myfunc.so
```

Zum Schluß, informieren wir Grace über diese neue Funktion

```
USE "myfunc_wrapper" TYPE f_of_nd FROM "/tmp/myfunc.so" ALIAS "myfunc"
```

6.4.3 Betriebssystem-Probleme

OS/2

Im Allgemeinen kann die Methode, wie sie oben in den Beispielen gezeigt wurde, auch auf OS/2 verwendet werden. Jedoch müssen Sie eine DLL erstellen (Dynamic Link Library), die auf OS/2 ein wenig schwieriger zu erstellen ist wie auf Un*x Systemen. Seitem Grace über die Verwendung von EMX portiert wurde, verwenden wir es auch für die Erstellung der Beispiele; jedoch sollten andere Entwicklungsumgebungen ebenso arbeiten (stellen Sie sicher, daß die `_System` Aufrufkonvention verwendet wird!). Wir beziehen uns nur auf Beispiel 3. Beispiel 2 könnte demonstrieren, daß die Eingangspunkte (z. B. exportierte Funktionen) einer DLL nur über Ordnungszahlen aufgerufen werden können, so könnten Sie nicht wissen, wie auf eine spezifische Funktion zugegriffen wird, ohne etwas Nachforschung zu betreiben. Zuerst kompilieren wir die Quelle aus Beispiel 2 nach `my_func.obj`:

Beispiel 6: *Kompilierung von Beispiel 3 unter OS/2*

```
gcc -Zomf -Zmt -c my_func.c -o my_func.obj
```

Anschließend müssen eine Linker-Definitionsdatei `my_func.def` erstellen, die einige grundlegende Information über die DLL enthält sowie Informationen über die exportierten Funktionen.

Beispiel: *Linker-Definitionsdatei*

```
LIBRARY my_func INITINSTANCE TERMINSTANCE
CODE LOADONCALL
DATA LOADONCALL MULTIPLE NONSHARED
DESCRIPTION 'This is a test DLL: my_func.dll'
EXPORTS
my_function
```

(Vergessen Sie nicht das Limit von 8 Zeichen für einen DLL-Namen!). Schlußendlich kann die DLL gelinkt werden:

Beispiel: Linken der DLL

```
gcc my_func.obj my_func.def -o my_func.dll -Zdll -Zno-rte -Zmt -Zomf
```

(für die verwendeten Compiler/Linker-Flags schauen Sie bitte in der EMX-Dokumentation nach!) Um diese neue Bibliotheksfunktion in Grace zu verwenden, müssen Sie die DLL entweder in den LIBPATH verschieben und können dann die Kurzform:

Beispiel: Sichtbarmachung in Grace über die Kurzform

```
USE "my_function" TYPE f_of_nd FROM "my_func" ALIAS "myf"
```

oder verschieben die DLL in einen beliebigen Pfad, den Sie dann explizit angeben müssen:

Beispiel: Sichtbarmachung in Grace mit expliziter Pfadangabe

```
USE "my_function" TYPE f_of_nd FROM "e:/foo/my_func.dll" ALIAS "myf"
```

(wie für die meisten System-APIs können Sie den Un*x-artigen Schrägstrich in Pfaden verwenden!)

7. Referenzen

7.1 Schriftsatz

Grace erlaubt einen ziemlich komplexen Schriftsatz auf Zeichenkettenbasis. Jede angezeigte Zeichenkette (Titel, Legenden, Tick-Markierungen, . . .) kann spezielle Kontrollcodes enthalten, um innerhalb der Zeichenkette den Text hochzustellen, die Schriftart zu ändern etc.

Kontrollcode	Beschreibung
<code>\f{x}</code>	Wechsel zu Schrift mit Name »x«
<code>\f{n}</code>	Wechsel zu Schrift mit Nummer »n«
<code>\f{}</code>	Wechsel zur Standardschrift
<code>\R{x}</code>	Wechsel zur Farbe mit Name »x«
<code>\R{n}</code>	Wechsel zur Farbe mit Nummer »n«
<code>\R{}</code>	Wechsel zur Standardfarbe
<code>\#{x}</code>	behandle »x« (muß eine grade Länge haben) als eine Liste von hexadezimalen Zeichencodes
<code>\t{xx xy yx yy}</code>	wende Transformationsmatrix an
<code>\t{}</code>	setze Transformationsmatrix zurück
<code>\z{x}</code>	Zoom x-mal
<code>\z{}</code>	Wechsel zum Standardzoom
<code>\r{x}</code>	Rotiere um x Grad
<code>\l{x}</code>	setzte Neigungsfaktor auf x
<code>\v{x}</code>	Hochstellung um x
<code>\V{x}</code>	Verschiebe die Basislinie um x
<code>\V{}</code>	Setze Basislinie auf Standardwert
<code>\h{x}</code>	Horizontale Verschiebung um x
<code>\n</code>	Neue Zeile
<code>\u</code>	Starte Unterstreichen
<code>\U</code>	Beende Unterstreichen

- Raster-Treiber (PNM/JPEG/PNG):
 - nur gerade-ungerade Füllregeln werden unterstützt
 - gemusterte Linien sind nicht implementiert
- PDF-Treiber:
 - Texte in Punktgraphik sind nicht transparent
- MIF-Treiber:
 - einige der gemusterten Füllungen sind nicht implementiert
 - Texte in Punktgraphik sind nicht implementiert
- SVG-Treiber:
 - Texte in Punktgraphik sind nicht implementiert

7.3 Gerätespezifische Einstellungen

Einige der Ausgabegeräte akzeptieren verschiedene Optionen für die Konfiguration. Sie können die Optionen setzen, indem Sie eine jeweilige Zeichenfolge dem Interpreter übergeben. Hierfür wird das Kommando »`DEVICE "devname" OP "options"`« verwendet (siehe Abschnitt 5.6 Geräteparameter, S. 32). Ein paar Optionen können innerhalb eines Kommandos übergeben werden, indem diese mittels Kommatas getrennt werden.

7.3.1 Optionen des PostScript-Treibers

Kommando	Beschreibung
<code>grayscale</code>	setze Ausgabe auf Graustufen
<code>color</code>	setze Ausgabe auf Farbe
<code>level 1</code>	verwende nur PS Level 1 Kommandos
<code>level 2</code>	verwende auch PS Level 2 Kommandos wenn benötigt
<code>docdata:7bit</code>	die Dokumentendaten sind in 7bit
<code>docdata:8bit</code>	die Dokumentendaten sind in 8bit
<code>docdata:binary</code>	die Dokumentendaten können Binär sein
<code>xoffset:x</code>	setze das Seitenoffset in X-Richtung auf x pp
<code>yoffset:y</code>	setze das Seitenoffset in Y-Richtung auf y pp
<code>mediafeed:auto</code>	Standardeingabeschacht
<code>mediafeed:match</code>	wähle Eingabe über die Medienseitendimensionen
<code>mediafeed>manual</code>	manueller Medienschacht
<code>hwresolution:on</code>	aktiviere Hardwareauflösung
<code>hwresolution:off</code>	deaktiviere Hardwareauflösung

Tabelle 39: Optionen des PostScript-Treibers

7.3.2 Optionen des EPS-Treibers

Kommando	Beschreibung
<code>grayscale</code>	setze Ausgabe auf Graustufen
<code>color</code>	setze Ausgabe auf Farbe
<code>level 1</code>	verwende nur PS Level 1 Kommandos
<code>level 2</code>	verwende auch PS Level 2 Kommandos wenn benötigt
<code>bbox:tight</code>	aktiviere enge Bounding Box
<code>bbox:page</code>	Bounding Box stimmt mit Seitendimensionen überein

Tabelle 40: Optionen des EPS-Treibers

7.3.3 Optionen des PDF-Treibers

Kommando	Beschreibung
PDF1.3	setze Kompatibilitätsmodus auf PDF-1.3
PDF1.4	setze Kompatibilitätsmodus auf PDF-1.4
compression:value	setze Komprimierungslevel (0-9)
patterns:on	aktiviere die Verwendung von Mustern
patterns:off	deaktiviere die Verwendung von Mustern

Tabelle 41: Optionen des PDF-Treibers

7.3.4 Optionen des PNM-Treibers

Kommando	Beschreibung
format:pbm	Ausgabe im PBM-Format
format:pgm	Ausgabe im PGM-Format
format:ppm	Ausgabe im PPM-Format
rawbits:on	»rawbits« (binäre) Ausgabe
rawbits:off	ASCII-Ausgabe

Tabelle 42: Optionen des PNM-Treibers

7.3.5 Optionen des JPEG-Treibers

Kommando	Beschreibung
grayscale	setze Ausgabe auf Graustufen
color	setze Ausgabe auf Farbe
optimize:on/off	aktiviere/deaktiviere Optimierung
quality:value	setzte Komprimierungsqualität (0-100)
smoothing:value	setze Glättung (0-100)
baseline:on/off	erzwinge/erzwingt nicht Basislinienausgabe
progressive:on/off	gebe/gebe nicht im Progressiv-Format aus
dct:ifast	verwende schnelle Ganzzahl DCT-Methode
dct:islow	verwende langsame Ganzzahl DCT-Methode
dct:float	verwende Fließkomma-DCT-Methode

Tabelle 43: Optionen des JPEG-Treibers

7.3.6 Optionen des PNG-Treibers

Kommando	Beschreibung
interlaced:on	erzeuge Interlaced-Graphik
interlaced:off	erzeuge keine Interlaced-Graphik
transparent:on	erzeuge transparente Graphik
transparent:off	erzeuge keine transparent Graphik
compression:value	setze Komprimierungslevel (0-9)

Tabelle 44: Optionen des PNG-Treibers

7.4 Datum in Grace

In Grace verwenden wir zwei Kalender: derjenige, der 532 von Denys gegründet wurde und bis 1582 verwendet wurde und derjenige, der von Luigi Lilio (Alyosius Lilius) und Christoph Klau (Christophorus Clavius) für Papst Gregorius XIII. erstellt wurde. Beide verwenden dieselben Monate (sie wurden unter dem Kaiser Augustus ein paar Jahre nach der Kalender-Einführung von Julian eingeführt; sowohl Julius als auch Augustus wurden durch die Benennung eines Monats nach ihnen geehrt).

Die Schaltjahre kamen regelmäßig im Kalendar von Denys vor: einmal alle vier Jahre, es gibt kein Jahr 0 in diesem Kalender (das Schaltjahr -1 war kurz vor dem Jahr 1). Dieser Kalender war mit der Erdbewegung nicht konform und das Datum bewegte sich hinsichtlich astronomischer Ereignisse langsamer.

Dies wurde 1582 korrigiert, indem der Gregorianische Kalender eingeführt wurde. Zuerst wurde eine Verschiebung von zehn Tagen eingeführt, um ein korrektes Datum zu erreichen (der Nachfolgetag vom Donnerstag, den 4. Oktober war der Freitag, 15. Oktober). Die Regeln für die Schaltjahre wurde ebenfalls geändert: 3 Schaltjahre werden innerhalb 4 Jahrhunderten entfernt. Das sind die Jahre, indem die Jahreszahl ein Vielfaches von 100 jedoch nicht von 400 ist: 1700, 1800 und 1900 waren keine Schaltjahre, jedoch 1600 und 2000 waren Schaltjahre.

Wir verwenden den Gregorianischen Kalender noch heute, jedoch haben wir jetzt verschiedene Zeitskalen für eine größere Genauigkeit. Die Internationale Atomzeit (International Atomic Time, TAI) ist eine lineare Skala: die beste Skala, die im wissenschaftlichen Bereich verwendet werden kann. Die koordinierte, universelle Zeit (UTC, häufig mit der Greenwich Mean Time GMT verwechselt) ist eine gesetzliche Zeit, die mit der Erdbewegung synchronisiert wird. Jedoch, da sich die Erdbewegung etwas verlangsamt, werden Sprung-Sekunden von Zeit zu Zeit in der UTC eingeführt (ungefähr eine Sekunde alle 18 Monate). UTC ist also eine nicht durchgängige Skala! Wenn eine Sprung-Sekunde durch den Internationalen Erdrotations-Dienst (International Earth Rotation Service) eingefügt wird, wird dies im Voraus veröffentlicht, und die gesetzliche Zeitfolge ist wie folgt: nach 23:59:59 folgt eine Sekunde später 23:59:60 und eine Sekunde später 00:00:00. Zur Zeit, in dem dies geschrieben wurde (05.01.1999) war der Unterschied zwischen TAI und UTC 32 Sekunden und die letzte Sprung-Sekunde wurde am 31.12.1998 eingeführt.

Diese Kalendar erlauben, jedes Datum von der Vergangenheit bis in die Zukunft zu beschreiben, aber sie sind für die Berechnung nicht günstig. Eine weitere Zeitskala ist möglich: das Zählen nur der Tage von einem Referenztag aus. Solch eine Zeitskala wurde von Joseph-Juste Scaliger (Josephus Justus Scaliger) 1583 eingeführt. Er entschied sich dafür, »-4713-01-01T12:00:00« als ein Referenzdatum zu verwenden, da es gleichzeitig ein Montag, 1. Janur eines Schaltjahres war, es war eine exakte Anzahl von 19 Jahren des Meton Zyklus zwischen diesem Datum und Jahr 1 (für die Osterberechnung), sowie war es der Beginn eines *Roman indiction* 15-Jahres-Zyklus. Die von diesem Referenzdatum aus gezählten Tageszahlen wird traditionell *Julianischer Tag* genannt, jedoch hat wirklich nicht mit dem Julianischen Kalender zu tun.

Grace speichert Datums intern als reelle Zahlen, gezählt von einem Referenzdatum aus. Das standardmäßige Referenzdatum ist dasjenige, das von Scaliger gewählt wurde; es ist eine klassische Referenz für astronomische Ereignisse. Es kann für eine einzelne Session unter Verwendung des *Bearbeiten>Einstellungen-Popup*, siehe Abschnitt 4.4.12 Einstellungen (Preferences), S. 17, der GUI geändert werden. Sollten Sie häufig mit einem spezifischen Referenzdatum arbeiten, können Sie dies für jede Session in der Konfigurationsdatei über das Kommando *REFERENCE DATE* einstellen (siehe auch Standard-template, Abschnitt 3.3.3, S. 9).

Die folgenden Datumsformate werden unterstützt (Stunden, Minuten und Sekunden sind immer optional):

- iso8601: 1999-12-31T23:59:59.999
- european: 31/12/1999 23:59:59.999 oder 31/12/99 23:59:59.999
- us: 12/31/1999 23:59:59.999 oder 12/31/99 23:59:59.999
- Julian: 123456.789

Man sollte sich bewußt sein, daß es in Grace nicht erlaubt ist, ein Leerzeichen in einer Datenspalte zu verwenden, wie Leerzeichen dazu verwendet werden, Felder zu separieren. Sie sollten immer einen

anderen Separator(/.- oder besser T) zwischen Datum und Zeit in Datendateien verwenden. Die GUI, die Batch-Sprache und die Kommandozeilen-Flags haben nicht diese Limitierung, Sie können Leerzeichen hier ohne weitere Probleme verwenden. Der T-Separator kommt aus dem ISO8601-Standard. Grace unterstützt die Verwendung auch in europäischen und US-amerikanischen Formaten.

Sie können auch einen Hinweis über das Format zur Verfügung stellen (»ISO8601«, »european«, »us«) über die Verwendung des *-datehint* Kommandozeilen-Flag oder des Referenz-Namens = "Edi t->Preferences" id="preferences"> Popup der GUI. Die Formate verwenden in der folgenden Reihenfolge durchprobiert: zuster der Hinweis, der vom Anwender gegeben wurde, dann ISO, europäische und US-amerikanisch (es gibt keine Zweideutigkeit zwischen Kalenderformaten und numerischen formaten und somit wird keine Reihenfolge für diese Angegeben). Die Separatoren zwischen verschiedenen Feldern können jedes Zeichen aus dem Satz »:./-T« enthalten (ein oder mehrere Leerzeichen werden als ein Trennzeichen aufgefaßt, andere Zeichen können nicht wiederholt werden, der T-Separator ist nur zwischen Datum und Zeit erlaubt, hauptsächlich für ISO8601), so das die Zeichenfolge »1999-12 31:23:59« erlaubt ist (jedoch nicht empfohlen wird). Das '-'-Zeichen wird sowohl als Separator (es wird traditionell im ISO8601-Format verwendet) als auch als unäres Minus verwendet (für Datums in der weiten Vergangenheit oder für numerische Daten). Standardmäßig werden Jahre nicht verändert, so das 99 ein Datum weit in der Vergangenheit kennzeichnet. Dieses Verhalten kann sowohl über das *Bearbeiten->Einstellungen-Popup*, siehe Abschnitt 4.4.12 Einstellungen (Preferences), S. 17, wie auch über die Kommandos `DATE WRAP on` und `DATE WRAP YEAR year` geändert werden. Nehmen Sie zum Beispiel an, daß das Basisjahr als 1950 definiert wurde, so werden die Jahr zwischen 0 und 99, die mit zwei oder weniger Zahlen geschrieben wurde, in der folgenden Art und Weise auf ein 4-stellige Jahresangabe umgeschrieben:

- Bereich [00;49] wird zu [2000;2049]
- Bereich [50;99] wird zu [1950;1999]

Wird das Basisjahr auf 1970 gesetzt, so werden die Jahresangaben wie folgt interpretiert:

- Bereich [00;69] wird zu [2000;2069]
- Bereich [70;99] wird zu [1970;1999]

Dies ist vollständig Jahr-2000-Konform und stimmt mit der derzeitigen Verwendung überein. Die Spezifizierung von Jahr 1 ist möglich, indem mehr als 2 Ziffern wie folgt angegeben werden: »0001-03-04« ist eindeutig der 4. März im Jahre 1. Die entgegengesetzt Transformation wird für Datums angewendet, die von Grace geschrieben werden, z.B. als Tick-Beschriftungen. Die Verwendung von Jahresangaben mit nur 2-Ziffern wird nicht empfohlen, wir führen hier einen *wrap year + 100* Bug ein, wenn diese Eigenschaft zu irgendeinem Zeitpunkt in der Zukunft mal entfernt werden sollte. . .

Der Datum-Scanner kann sowohl für Denys's und Gregorianische Kalender verwendet werden. Nichtexistierende Datums werden erkannt, dies schließt das Jahr 0, Datums zwischen 05.10.1582 und 14.10.1582, den 29. Februar einen Nicht-Schaltjahres, Monate unterhalb 1 und oberhalb 12, . . . ein. Der Scanner zieht jedoch Schaltsekunden nicht in Betracht: Sie können sich denken, daß er nur auf der Internationalen Atomzeit (International Atomic Time, TAI) und nicht auf der koordinierten, universellen Zeit (Coordinated Unified Time, UTC) arbeitet. Wenn Sie sich jedoch in einer Situation finden, in der Sie die UTC brauche, eine sehr genaue Skale, und müssen die Schaltsekunden in Betracht ziehen. . . so sollten Sie die Daten selbst umwandeln (zum Beispiel unter Verwendung der TAI). Aber wenn Sie sich darum kümmern müssen, wissen Sie wahrscheinlich bereits, was zu tun ist.

7.5 Xmgr nach Grace Migrationshandbuch

Dies ist eine sehr kurzes Handbuch, das Probleme und Workaround beschreibt, um Projektdateien, die mit Xmgr gespeichert wurden, einzulesen. Sie sollten die Dokumentationen lesen oder entwas mit Grace herumspielen, um die neuen Features und Steuerungen zu prüfen.

1. Grace muß explizit die Versionsnummer der Software mitgeteilt werden, mit der die Datei erzeugt wurde. Sie können manuell die Zeichenfolge `@version VERSIONID` zu Beginn der Datei eintragen. Die `VERSIONID` wird gebildet aus `MAJOR_REV*10000 + MINOR_REV*100 + PATCHLEVEL`; dementsprechend entspricht die Versionskennung 40101 der Version `xmgr-4.1.1`. Projekte, die mit

- Xmgr-4.1.2 gespeichert wurde, brauchen das o. g. Verfahren *nicht*, die Versionskennung wird automatisch von Xmgr-4.1.2 eingefügt. Haben Sie keine Idee, mit welcher Version von Xmgr Ihre Datei erzeugt wurde, versuchen Sie verschiedene Versionskennungen. In vielen Fällen ist es 40102.
2. Das oben genannte bezieht sich nur auf ASCII-Projekte. Die alten binären Projekte (gespeichert mit xmgr-4.0.*) werden nicht mehr automatisch umgewandelt. Hier muß ein Eingangsfiler definiert werden, um die Konvertierung »on-the-fly« durchzuführen. Fügen Sie hierzu die folgende Zeile in `./gracerc` oder der systemweiten `$GRACE_HOME/gracerc` Ressourcen-Datei hinzu: `DEFINE IFILTER "grconvert %s -" MAGIC "00000031"`. Für weitere Information zu den I/O-Filtern sehen Sie bitte in den Dokumentationen nach.
 3. Die Dokumentation der Scriptsprache fehlt noch komplett.
 4. Grace ist WYSIWYG. Xmgr war es nicht. Viele Änderungen, die erforderlich waren, um ein WYSIWYG-Verhalten zu erreichen, führten zu Situationen, in denen Objekte in Graphen, die mit Xmgr ausgerichtet wurden, in dieser Form nicht unter Grace betrachtet und angezeigt werden können. Grace versucht sein Bestes, diese Unterschiede zu kompensieren, jedoch kann es vorkommen, daß Sie solche Graphen manuell neu ausrichten müssen.
 5. Viele Symbol-Typen (alle außer *realen* Symbolen) werden entfernt. '>Location *'<-Typen können (mit sehr viel mehr Komfort) durch kommentierte Werte ersetzt werden. '>Impulse *'<, '>Histogram *'< und '>Stair steps*'< Effekte können über die Verwendung der Verbindungslinien-Parameter (Typ, Linien entfernen) erreicht werden. Das '>Punkt'<-Symbol ist ebenfalls entfernt worden; verwenden Sie das gefüllte Kreissymbol mit der Größe 0 und ohne Umriß, um dieselbe Wirkung zu erhalten.
 6. Das Standard-Seitenlayout ist von frei (erlaubt die Änderung der Zeichenfläche mit der Maus) auf fest geändert worden. Für das alte Verhalten, fügen sie `PAGE LAYOUT FREE` in die Grace-Ressourcendatei ein oder verwenden Sie den `-free` Kommandozeilenschalter. **Die Verwendung des '>freien'< Seitenlayout ist im Allgemeinen veraltet.**
 7. Die System(-Shell)-Variablen `GR_*` wurden zu `GRACE_*` umbenannt.
 8. Smith-Plots funktionieren derzeit nicht. Sie werden bald zurückkommen.

Anhang

Tabellenverzeichnis

Tabelle 1	Bibliothekslizenzen	1
Tabelle 2	Liste der verfügbaren Satztypen	5
Tabelle 3	Übersicht, welche Satztypen durch welchen Graphyp dargestellt werden können...	6
Tabelle 5	Übersicht der Eigenschaften	21
Tabelle 6	Basis-Typen	26
Tabelle 7	Graphenauswahl	26
Tabelle 8	Satzauswahl	26
Tabelle 9	Regionauswahl	26
Tabelle 10	Farbauswahl	26
Tabelle 11	Patternauswahl	26
Tabelle 12	Datenspaltenauswahl	26
Tabelle 13	Vektorvariablen	27
Tabelle 14	Skalarvariablen	27
Tabelle 15	Arithmetische Operatoren	27
Tabelle 16	Logische Operatoren	27
Tabelle 17	Vergleichsoperatoren	27
Tabelle 18	Funktionen	28
Tabelle 19	Statistische Funktionen	29
Tabelle 20	Aggregatfunktionen	29
Tabelle 21	Spezielle mathematische Funktionen	30
Tabelle 22	Transformationen	31
Tabelle 23	Geräteparameter	32
Tabelle 24	Ablaufkontrolle	32
Tabelle 25	Deklarationen	33
Tabelle 26	Graphenoperationen	33
Tabelle 27	Parameter der Graphenauswahl	34
Tabelle 29	Achsen-Parameter	35
Tabelle 31	Titel und Legenden	36
Tabelle 32	Satzeingabe, -ausgabe und -erstellung	37
Tabelle 33	Satz-Operationen	37

Tabelle 34	grace_np-Bibliothek: C-Funktionen	40
Tabelle 35	grace_np-Bibliothek: Fortran F77-Funktionen	40
Tabelle 36	Funktionstypen	43
Tabelle 38	Kontrollcodes für Schriftsatz	47
Tabelle 39	Optionen des PostScript-Treibers	48
Tabelle 40	Optionen des EPS-Treibers	48
Tabelle 41	Optionen des PDF-Treibers	49
Tabelle 42	Optionen des PNM-Treibers	49
Tabelle 43	Optionen des JPEG-Treibers	49
Tabelle 44	Optionen des PNG-Treibers	49

Copyright-Hinweis

Dieses Dokument, die Übersetzung der Grace FAQ, steht ebenfalls wie Grace unter der GPL-Lizenz.

Copyright (©) 1991-1995 Paul J Turner, Portland, OR
 Copyright (©) 1996-2005 Grace Development Team

Maintained by Evgeny Stambulchik

All Rights Reserved

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

Liste der Versionen

Version	Datum	Bearbeiter	Bemerkung
	19.05.2005		Englischsprachiges Original
	17.05.2006	Bri	Deutsche Übersetzung
	14.11.2006	Krä	Erstellung PDF-Dokument