

Anwendersoftwareklasse 2

DYNAMO – Compiler

Stand: 08.08.86

Alle Rechte vorbehalten.
Insbesondere ist die Überführung in maschinenlesbare Form, sowie das Speichern in Informationssystemen, auch auszugsweise, nur mit schriftlicher Genehmigung der GMD gestattet.

Herausgeber:

Gesellschaft für Mathematik und Datenverarbeitung mbH

Postfach 1240, Schloß Birlinghoven
D - 5205 Sankt Augustin 1
Telefon(02241) 14 - 1, Telex 8 89 469 gmd d
Telefax(02241) 14 28 89, BTX *43900#
Teletex 2627 - 224135 = GMDVV

Softwareklasse 2 (Anwendersoftware)

Regelmäßige Wartung

Autoren:

Diether Craemer, Robert Keil

Texterstellung:

Dieser Text wurde mit der EUMEL - Textverarbeitung erstellt und aufbereitet und mit dem Agfa Laserdrucksystem P400 gedruckt.

Umschlaggestaltung:

Hannelotte Wecken

Hinweis:

Diese Dokumentation wurde mit größtmöglicher Sorgfalt erstellt. Dennoch wird für die Korrektheit und Vollständigkeit der gemachten Angaben keine Gewähr übernommen. Bei vermuteten Fehlern der Software oder der Dokumentation bitten wir um baldige Meldung, damit eine Korrektur möglichst rasch erfolgen kann. Anregungen und Kritik sind jederzeit willkommen.

Inhaltsverzeichnis

1. Überblick	2
2. Einführung in die Benutzung des Compilers	4
3. Abweichungen gegenüber dem Sprachstandard	6
4. Das DYNAMO Runtime – System	8
5. Fehlermeldungen des DYNAMO – Compilers	9
6. Augenblicklich vorhandene Funktionen	14
7. Insertieren von Makros	16
8. Implementationshinweise	17
9. DYNAMO – EUMEL Programm – Tausch	19

1. Überblick

In diesem Kapitel des Handbuchs wird die Benutzung des DYNAMO – Compilers im EUMEL – System beschrieben und ferner die Einschränkungen oder Änderungen gegenüber dem DYNAMO – Sprachstandard angegeben. Eine Einführung in die Programmiersprache DYNAMO wird hier nicht gegeben.

Dafür verweisen wir auf :

Pugh, Alexander L. : "DYNAMO II User's Manual"
Cambridge, London 1973 : MIT – Press
ISBN 0 – 262 – 66018 – 0

Craemer, Diether : "Fluß und Zustand"
Arbeitspapiere der GMD Nr. 36
GMD – Informatik – Kolleg, Selbstverlag GMD Birlinghoven 1983

Der DYNAMO – Compiler wurde 1983 von Robert Keil und Torsten Fröhlich (Helmholtz – Gymnasium, Bonn) im Rahmen des MIKROS – Projektes am Informatik – Kolleg der GMD entwickelt.

Für Spezifikation und Betreuung der Entwicklung war Dr. Diether Craemer verantwortlich, software – technische Unterstützung kam von Prof. John Henize, Dr. Peter Heyderhoff, Rudolf Legde und Dipl. – Math. Lothar Oppor.

Die Programmiersprache DYNAMO

DYNAMO wurde von einer Gruppe um Jay FORRESTER am Massachusetts Institute of Technology (MIT) um 1960 entwickelt.

In DYNAMO können Systeme, in denen Veränderungen kontinuierlich geschehen, modelliert und simuliert werden.

Kontinuierliche Veränderung von Größen werden über deren Veränderungsrate im Wesentlichen nach folgender Gleichung berechnet

Größe jetzt = Größe früher + DT * Veränderungsrate,

dabei ist DT die Länge des Zeitintervalls von "früher" bis "jetzt".

Außer diesen Gleichungen für Größen braucht man Gleichungen für die Veränderungsrate, für Hilfsgrößen, zur Initialisierung von Größen, zur Definition von Konstanten und Tabellen, zu Angaben von Simulationsläufen und zur Wiedergabe von Ergebnissen in Zahlentabellen oder Diagrammen.

Alle diese Gleichungen können einfach in der Form, wie man sie aus dem Mathematik – Unterricht der Sekundarstufe kennt, hingeschrieben werden, ohne sich Gedanken über den Ablauf des Programms machen zu müssen.

DYNAMO ist also eine einfache funktional-applikative, nicht-prozedurale Sprache.

Das macht ihren Reiz und Leistungsfähigkeit aus, die zur Formulierung der bekannten Weltmodelle von FORRESTER, MEADOWS ("Die Grenzen des Wachstums"), PESTEL, MESAROVIC u.a. in dieser Sprache führten.

Anwendungsgebiete der Sprache sind ökologische, gesellschaftliche, wirtschaftliche und technische Systeme, deren dynamisches Verhalten der Modellbildner nachbilden und studieren möchte.

Im Allgemeinen verfolgt der Modellbildner mit seinem Modell einen Zweck (Verhaltensänderung des nachgebildeten Systems), sodaß auch neben gesicherten Fakten die Wertvorstellungen des Modellbildners in das Modell eingehen.

2. Einführung in die Benutzung des Compilers

Der im EUMEL – System implementierte DYNAMO – Compiler ist ein 2 – Pass Compiler, der die DYNAMO – Programme zunächst in ELAN übersetzt. Der Vorteil dieser Methode besteht darin, daß es möglich ist, die übersetzten Programme unabhängig vom DYNAMO – Compiler zur Ausführung bringen zu können.

Es bestehen mehrere Möglichkeiten, den Compiler aufzurufen :

dynamo (TEXT CONST quelledatei)

dynamo (TEXT CONST quelledatei, zieldatei, BOOL CONST pass2 ausfuehren)

dynamo

wobei im letzteren Fall als Quelledatei <last param> angenommen wird. Die Zieldatei erhält (wenn nicht anders angegeben) standardmäßig den Namen: <quelledatei> + ".elan".

Falls die Zieldatei nach der Übersetzung automatisch gelöscht werden soll, muß vorher die Prozedur

erase (BOOL CONST erase option)

aufgerufen werden.

Die Erstellung einer Fehlerdatei kann durch den Aufruf der Prozedur

error listing (TEXT CONST fehlerdatei)

erreicht werden.

Soll während der Laufzeit des Programms die gesamte Ausgabe nicht nur auf dem Bildschirm, sondern auch auf einer Datei mit Namen "dyn.out" protokolliert werden, so muß die Prozedur

protokoll (BOOL CONST protokoll option)

aufgerufen werden. Die Datei "dyn.out" enthält auch Seitenvorschubcodes und sollte nur mit einem EUMEL – Printer Programm ausgedruckt werden.

3. Abweichungen gegenüber dem Sprachstandard

- Die Länge der Namen ist nicht auf 7 Zeichen festgelegt, sondern praktisch beliebig (32000). Dies ist eine Erweiterung; wer seine Programme auch auf anderen DYNAMO – Compilern laufen lassen will, sollte sich aber auf 7 Zeichen beschränken.
- Zahlen werden intern mit einer Mantisse von 13 Stellen abgespeichert, von denen nur die ersten 7 bei der Ausgabe dargestellt werden. Die größte darstellbare Zahl ist daher 9.999999999999e126.
- Die maximale Anzahl der Gleichungen ist willkürlich auf 950 festgelegt, kann aber durch Ändern einer Konstante im Programmtext ("max tab size") beliebig nach oben verschoben werden.
- Der Compiler akzeptiert aus Gründen der besseren Lesbarkeit auch Programme, die in Kleinschrift geschrieben sind. Dabei ist es sinnvoll, die Quellprogramme konsistent zu halten (d.h. Groß- und Kleinschrift nicht zu vermischen). Man sollte grundsätzlich Kleinschrift vorziehen, da diese vom Compiler auch effizienter verarbeitet werden kann.
- Quellprogramme dürfen eine beliebige Zahl von Leerzeilen enthalten. X – Befehle (Fortschreibungszeilen) werden davon nicht beeinflusst.
- In der augenblicklichen Version 3.3 des Compilers gelten folgende Einschränkungen :
 1. Bei der Verarbeitung von Array – Gleichungen werden Compilerseitig keine Semantik – Überprüfungen auf eventuell unzureichende Initialisierung oder Überlappung (d.h. mehrfaches Setzen desselben Elements) durchgeführt. Defaultmäßig bekommen alle Elemente einer Array – Gleichung bei der Initialisierung den Wert '0.0' zugewiesen.
 2. Die maximale Größe von Tables und Array – Gleichungen ist durch Verwendung des Vector – Pakets auf 4000 Elemente festgelegt. Da pro Table – Element aber zur Zeit eine Zeile im Zielprogramm generiert wird, sollte man dies besser nicht ausnutzen.

3. Supplementary – Gleichungen werden aus Kompatibilitäts – Gründen korrekt übersetzt, aber sonst wie andere Gleichungen behandelt.
4. Print – und Plotperiode werden nur als Konstanten verarbeitet. Falls Gleichungen für 'prtper' oder 'pltper' angegeben werden, so bewirken diese keine Veränderung.
5. Array – Gleichungen dürfen nicht mehr als eine Dimension besitzen.
6. Für Gleichungen, die Makro – Aufrufe enthalten, sollten Initialisierungs (N) – Gleichungen angegeben werden.

4. Das DYNAMO Runtime – System

Nach erfolgreicher Übersetzung wird vom Zielprogramm das Runtime – System aufgerufen. In diesem Modus ist es möglich, Konstanten zu ändern und Dynamo – Programme zur Ausführung zu bringen.

Im DYNAMO Runtime – System stehen folgende Kommandos zur Verfügung :

run.....	Ausführen des übersetzten Programms
run <name>	Ausführen des übersetzten Programms und Retten des Konstantendatenraums in den Datenraum mit Namen " <name> .const"
run <name> .const.....	Übernehmen des Konstantendatenraums " <name> .const" und anschließende Ausführung des übersetzten Programms
c <Konstantenname> = Wert.....	Änderung einer oder mehrerer Konstanten
?.....	Anzeige der Konstanten und ihrer Werte
quit.....	Verlassen des Runtime – Systems
help.....	Zeigt diese Erklärungen

Bei PRINT oder PLOT – Ausgaben sind folgende Kommandos möglich :

+	Nächster Bildschirm
o.....	(Off), keine Unterbrechung der Ausgabe
e.....	(End), Zurück zum Runtime – System
<ESC>	Abbruch der Ausgabe

5. Fehlermeldungen des DYNAMO – Compilers

Falls der Compiler einen Fehler im DYNAMO – Programm entdeckt, gibt er eine Fehlermeldung nach dem folgenden Muster aus :

"FEHLER IN ZEILE " <zeilennummer> " BEI >>" <symbol> "<< : " <fehler-text> .

Im folgenden sind alle Fehlermeldungen und Möglichkeiten zur Abhilfe aufgelistet, sofern diese nicht klar ersichtlich sind:

- 1 GLEICHUNG DOPPELT DEFINIERT
- 2 DOPPELTE INITIALISIERUNG
- 3 FALSCHER ZEILENTYP
– > Erlaubt sind : a, c, l, n, r, s, print, plot, note, spec, *, x, macro, mend, for, noise, run.
- 4 VERSCHACHELTE MAKRO – DEFINITION
– > 'mend' – Befehl fehlt.
- 5 MAKRO – NAME ERWARTET
- 6 '(' ERWARTET
- 7 FORMALER PARAMETER ERWARTET
- 8 ')' NACH PARAMETERLISTE ERWARTET
- 9 BEI AUXILIARIES NUR SUBSKRIPTION MIT '.K' ERLAUBT
- 10 BEI KONSTANTEN – DEFINITION NAME ERWARTET
- 11 BEI LEVELS NUR SUBSKRIPTION MIT '.K' ERLAUBT
- 12 BEI RATES NUR SUBSKRIPTION MIT '.KL' ERLAUBT

- 13 BEI TABLE – DEFINITIONEN KEINE SUBSKRIPTION ERLAUBT
- 14 X – BEFEHL HIER NICHT ERLAUBT
- 15 BEI FOR – DEFINITION NAME ERWARTET
- 16 '=' NACH FOR – VARIABLE ERWARTET
- 17 BEREICHSANGABE ERWARTET
- 18 ';' ERWARTET
- 19 LOKALE GLEICHUNG NUR IN MAKRO ERLAUBT
- 20 BEI DEFINITION NAME ERWARTET
- 21 '=' ERWARTET
- 22 INDEX NICHT KORREKT
- > Als Index ist nur erlaubt : <for variable> !,
 <add op> <ganze zahl>!.
 <add op> ::= "+ "; "- ".
- 23 ')' NACH INDIZIERUNG ERWARTET
- 24 PRTPER NICHT DEFINIERT
- > Wenn das Programm einen Print – Befehl enthält, muß 'prtper' (Printperiode)
 als Konstante definiert werden.
- 25 PLTPER NICHT DEFINIERT
- > Wenn das Programm einen Plot – Befehl enthält, muß 'pltper' (Plotperiode)
 als Konstante definiert werden.
- 26 '/' ODER ';' BEI PLOT ERWARTET
- 27 NAME ALS PLOTPARAMETER ERWARTET
- 28 DOPPELTE SCALE – ANGABE IN EINER GRUPPE
- > Wenn mehrere Plotparameter mit ';' getrennt werden (Also die gleiche Skala-
 lierung erhalten), dürfen nicht mehrere feste Skalierungen angegeben wer-
 den.

- 29 ERSTE SCALE – ANGABE ERWARTET
- 30 ZWEITE SCALE – ANGABE ERWARTET
- 31 ')' NACH SCALE – ANGABE FEHLT
- 32 PRINTPARAMETER NICHT DEFINIERT
- 33 PRINTPARAMETER ERWARTET
- 34 TIME DARF NUR INITIALISIERT WERDEN
- 35 DT NICHT DEFINIERT
- 36 LENGTH NICHT DEFINIERT
- 37 BEI KONSTANTEN – DEFINITION ZAHL ERWARTET
- 38 BEI INITIALISIERUNG KONSTANTE ERWARTET
- 39 LEVELS MUESSEN INITIALISIERT WERDEN
- 40 KONSTANTE BEI TABLE ERWARTET
- 41 '/' ODER ", " ERWARTET
- 42 TABLE – DEFINITION OHNE BENUTZUNG
- 43 SIMULTANE GLEICHUNGEN
 - > Bei dem Versuch, A, R, oder N – Gleichungen zu sortieren, trat eine direkte oder indirekte Rekursion auf.
- 44 FAKTOR ERWARTET
 - > Erwartet : <zahl>;
<funktions aufruf>;
<macro aufruf>;
<gleichungs name>;
'(', <ausdruck>, ')';
<monadischer operator>, <faktor>.
<monadischer operator> ::= '+'; '- '.

- 45 TIME MUSS MIT '.J' ODER '.K' SUBSKRIBIERT WERDEN
- 46 SYMBOL NICHT DEFINIERT
- 47 FUNKTION NICHT DEFINIERT
- 48 UNZULAESSIGE INDIZIERUNG
–> Die Indices auf beiden Seiten der Gleichung müssen immer gleich sein.
- 49 FALSCHER PARAMETERANZAHL
- 50 FALSCHES TRENNSYMBOL ZWISCHEN PARAMETERN
- 51 ALS PARAMETER TABLE ERWARTET
- 52 FALSCHER PARAMETER IN TABLEFUNKTION
- 53 ZUVIELE AKTUELLE PARAMETER
- 54 ')' NACH MAKROAUFRUF FEHLT
- 55 REKURSIVER MAKROAUFRUF
- 56 BEI N – GLEICHUNG KEINE SUBSKRIPTION ERLAUBT
- 57 FALSCHER SUBSKRIPTION IN AUXILIARY – GLEICHUNG
- 58 ')' ERWARTET
- 59 FALSCHER SUBSKRIPTION IN LEVEL – GLEICHUNG
- 60 FALSCHER SUBSKRIPTION IN RATE – GLEICHUNG
- 61 FOR – VARIABLE NICHT DEFINIERT
–> Eine FOR – Variable muß vor der ersten Benutzung definiert werden.
- 62 KONSTANTE ERWARTET
- 63 FALSCHES REAL – FORMAT
–> Exponent fehlt

- 64 GLOBALE GLEICHUNG IN MACRO NICHT ERLAUBT
- 65 DOPPELTE DEFINITION BEI MEHRFACHEM MAKROAFRUF
- 66 ALS NOISE – PARAMETER ZAHL ERWARTET

6. Augenblicklich vorhandene Funktionen

In DYNAMO – Programmen können folgende Funktionen aufgerufen werden :
('t' steht für Table als Parameter, 'r' für Real)

clip	(r, r, r, r)
fifge	(r, r, r, r)
switch	(r, r, r)
fifze	(r, r, r)
noise	(r)
normn	(r, r)
pulse	(r, r, r)
ramp	(r, r)
sclprd	(t, r, r, t, r)
step	(r, r)
sumv	(t, r, r)
sum	(t)
table	(t, r, r, r, r)
tabhl	(t, r, r, r, r)
power	(r, r)

Dazu kommen noch weitere EUMEL – Standardprozeduren, (soweit in der Standard – Datei angegeben; siehe Abschnitt 8.):

abs, arctan, arctand, cos, cosd, exp, floor, frac, initializerandom, random, ln, log2, log10, max, min, sin, sind, sqrt, tan, tand.

Wer eigene Funktionen bereitstellen will, gehe folgendermaßen vor:

1. Schreiben Sie die Funktion, eine REAL PROC, in ELAN.
2. Schreiben Sie den Namen der Funktion, gefolgt von den Typen der Eingabeparameter in die bestehende Datei "dyn.std" (siehe 8.)
3. Insertieren Sie das Paket mit der Definition der Funktion.
4. Rufen Sie die Prozedur : init std ("dyn.std") auf.

Danach können Sie diese Funktion in DYNAMO – Programmen benutzen.

5. Schicken Sie bitte ausgetestete Funktionen zusammen mit einem DYNAMO – Programm, das die Funktion benutzt, an die

DYNAMO – EUMEL Programm – Tausch – Stelle
c/o Dr. Diether Craemer
GMD Institut für Technologie – Transfer
Postfach 1240

D – 5205 Sankt Augustin 1

(siehe auch unter Abschnitt 9.)

7. Insertieren von Makros

Zusätzlich zu der Möglichkeit, eigene Funktionen zu definieren, kann der Benutzer auch eigene Makros in eine eigene Makrobibliothek eintragen. Dies geschieht durch Aufruf der Prozedur :

```
insert macro (TEXT CONST datei name)
```

Die Datei sollte zweckmäßigerweise nur Makrodefinitionen enthalten. Es ist – im Gegensatz zu normalen Dynamo Programmen – nicht nötig, die Systemkonstanten zu definieren.

Falls in dieser Makrodatei dennoch Konstanten definiert werden, können diese anschließend in jedem Dynamo – Programm verwendet werden. Standardmäßig werden im DYNAMO – Compiler folgende Makros zur Verfügung gestellt (Nähere Beschreibung im DYNAMO II Manual).

```
macro delay1(in,del)
```

```
macro delay3(in,del)
```

```
macro delay3p(in,del,ppl)
```

```
macro dlinf3(in,del)
```

```
macro smooth(in,del)
```

Falls ein Makro in der Bibliothek nicht gefunden wird, kopieren Sie diesen an das Ende des Programms. Dann sollte das Programm mitsamt den Makro – Aufrufen laufen. Vorher versuchen Sie nochmals den Befehl:

```
insert macro ("dyn.mac")
```

8. Implementationshinweise

Der DYNAMO – Compiler wird als Quell – Datei auf einer Archiv – Floppy weitergegeben. Diese Datei trägt den Namen dyn.<n> (<n> ist die aktuelle Versionsnummer, z.Zt. 33) und muß nur insertiert werden. Um den Compiler dann noch betriebsfertig zu machen ist es nötig, die gewünschten Funktionen für DYNAMO in eine spezielle Datei mit dem Namen "dyn.std" einzutragen.

Beispiel für eine Standard – Datei :

```
cos r exp r max rr table trrr /*
```

Die auf der Archiv – Floppy mit dem Namen "dynamo" mitgelieferte Standard – Datei "dyn.std" hat folgenden Stand:

```
abs r arctan r arctand r cos r cosd r exp r floor r frac r
initializerandom r random r
ln r log2 r log10 r
max rr min rr
power rr
sin r sind r sqrt r tan r tand r
clip rrrr fifge rrrr switch rrr fifze rrr noise r normrn rr pulse rrr
ramp rr sclprd trrr step rr sumv trr sum t table trrrr tabhl trrrr /*
```

Die Buchstaben hinter den Funktionsnamen geben den Typ der Parameter an, ('t' für Table, 'r' für Real), das '/'* – Symbol bedeutet Datei – Ende. Es sollten natürlich nur Prozedurnamen angegeben werden, die auch wirklich vorher insertiert wurden. Damit diese Prozedurnamen in die Tabellen des DYNAMO – Compilers eingetragen werden, muß dann noch die Prozedur 'init std ("dyn.std")' aufgerufen werden. Dieser Aufruf ist nur nötig, falls man eine eigene Standard – Datei verwenden will.

Das auf der Archivdiskette "dynamo" ausgelieferte DYNAMO – System mit den vorgesehenen Standardfunktionen wird folgendermaßen implementiert:

- > archive ("dynamo")
- > fetch all (archive)
- > run ("dyn.gen")

Es werden dann mehrere Pakete insertiert. Das Ende des Generierungslaufs wird durch "dynamo system generiert" angezeigt.

Damit ein DYNAMO System erzeugt werden kann, sind auf dem Archiv folgende Dateien nötig:

"dyn.gen"
 "dyn.errors"
 "dyn.mac"
 "dyn.help"
 "dyn.tool"
 "dyn.33"
 "dyn.vec"
 "dyn.proc"
 "dyn.rts"
 "dyn.plot"
 "dyn.std"

Auf dem Archiv werden noch folgende Dateien ausgeliefert:

"dyn.workfluc"	siehe CRAEMER,a.a.O., S. 290
"dyn.cob"	siehe CRAEMER,a.a.O., S. 362
"dyn.oszillator"	siehe CRAEMER, a.a.O., S. 275
"dyn.steifedgl"	Literaturangabe im Programm
"dyn.natchez"	siehe CRAEMER, a.a.O., S. 354
"dyn.wachstum"	siehe CRAEMER, a.a.O., S. 179
"dyn.simon"	siehe CRAEMER, a.a.O., S. 290
"dyn.gekoppeltependel"	siehe CRAEMER, a.a.O., S. 280
"dyn.abnahme"	siehe CRAEMER, a.a.O., S. 210
"dyn.volterrabeginn"	siehe CRAEMER, a.a.O., S. 76
"dyn.volterra"	siehe CRAEMER, a.a.O., S. 86
"dyn.grashasenfuchs"	siehe CRAEMER, a.a.O., S. 94
"dyn.wasseröko"	Literaturangabe im Programm
"dyn.bev"	ein Bevölkerungsmodell der BRD
"dyn.delaytest"	Testprogramm für Makros
"dyn.mehreredelays"	Testprogramm für Makros
"dyn.wohnen"	Literaturangabe im Programm
"dyn.wurzel"	siehe CRAEMER, a.a.O., S. 342
"out.world"	siehe CRAEMER, a.a.O., S. 330
"dyn.quadrat"	
"dyn.forst7"	
"dyn.ruestungswettlauf"	Literaturangabe im Programm
"ruestungsgleichgewicht.const"	DATASPACE mit Konstanten für "dyn.ruestungswettlauf"
"stabileruestung.const"	DATASPACE mit Konstanten für "dyn.ruestungswettlauf"

9. DYNAMO – EUMEL Programm – Tausch

Wir möchten alle Benutzer des DYNAMO Compilers bitten, uns ihre Erfahrungen zugänglich zu machen.

Wer zum **** DYNAMO – EUMEL – Benutzerkreis **** gehört, wird in regelmäßigen Abständen über Neuerungen informiert und mit den inzwischen eingegangenen DYNAMO – Programmen und Funktionen versorgt.

Wie werden Sie Mitglied im DYNAMO – EUMEL – Benutzerkreis?

Schicken Sie uns ein dokumentiertes DYNAMO – Programm oder eine für DYNAMO nützliche ELAN – Funktion.

Sie erhalten dann eine Floppy mit der augenblicklich bestehenden Bibliothek von ausgetesteten DYNAMO – Programmen und Funktionen und werden als Benutzer geführt.

Bitte schreiben Sie an

DYNAMO – EUMEL Programm – Tausch – Stelle
c/o Dr. Diether Craemer
GMD Institut für Technologie – Transfer
Postfach 1240

D – 5205 Sankt Augustin 1

bitte hier ausschneiden

JA, ich möchte Mitglied im DYNAMO – EUMEL Benutzerkreis werden. Anbei übersende ich folgenden Beitrag:

Titel:

Kurzbeschreibung:

Meine Adresse:
