

# **Dokumentation zum KSWS Projekt**

von Marcus Timm

Rostock, 26.02.2006

# Inhaltsverzeichnis

1. Einleitung.....	S. 4
1.1 Über das Projekt.....	S. 4
1.2 Inhalt des KSWS.....	S. 4
1.3 Weitere Informationen.....	S. 4
2. Benutzerhandbuch zu BendManager.....	S. 5
2.1 Installations-Hinweise.....	S. 5
2.2 Allgemeines zur Benutzung von BendManager.....	S. 5
2.3 Menüpunkt „Datei“.....	S. 6
2.4 Menüpunkt „Bearbeiten“.....	S. 7
2.5 Menüpunkt „Einstellungen“.....	S. 7
2.6 Menüpunkt „Ansicht“.....	S. 7
2.7 Menüpunkt „Hilfe“.....	S. 8
2.8 Der DataBrowser.....	S. 8
2.9 Das interaktive Diagramm.....	S. 9
3. Benutzerhandbuch zu BendWorker.....	S. 11
3.1 Installations-Hinweise.....	S. 11
3.2 Allgemeines zur Benutzung von BendWorker.....	S. 11
3.3 Die Startseite.....	S. 12
3.4 Auswahl „Biegung durchführen“.....	S. 12
3.4.1 Die Material-Auswahl.....	S. 13
3.4.2 Die Test-Biegung.....	S. 14
3.4.3 Die Modus-Auswahl.....	S. 14
3.4.4 Der Prognose-Modus.....	S. 15
3.4.5 Der CNC-Modus.....	S. 15
3.5 Auswahl „Benutzerhandbuch“.....	S. 15
3.6 Auswahl „Hilfe anfordern“.....	S. 16
4. Technische Dokumentation.....	S. 17
4.1 Allgemein.....	S. 17
4.2 BendLibrary.dll.....	S. 18
4.2.1 Auflistung aller Pakete und Kurzbeschreibung.....	S. 18
4.2.2 Auflistung aller Klassen und Kurzbeschreibung.....	S. 18
4.3 BendManager.exe.....	S. 20
4.3.1 Auflistung aller Pakete und Kurzbeschreibung.....	S. 20
4.3.2 Auflistung aller Klassen und Kurzbeschreibung.....	S. 20
4.4 BendWorker.exe.....	S. 20
4.4.1 Auflistung aller Pakete und Kurzbeschreibung.....	S. 20
4.4.2 Auflistung aller Klassen und Kurzbeschreibung.....	S. 21

4.5 Berechnung der Rückverformung.....	S. 21
5. Empfehlungen für Weiterentwicklung.....	S. 22
5.1 Ein 3D-Diagramm für BendManager.....	S. 22
5.2 Ein kompletter Neuanfang.....	S. 22
6. Anhang A – Abbildungsverzeichnis.....	S. 23
7. Anhang B – FORTRAN-Compiler für .NET.....	S. 24

# 1. Einleitung

## 1.1 Über das Projekt

Das Projekt entstand in einer Zusammenarbeit des Lehrstuhls für Computergraphik der Fakultät für Informatik und Elektrotechnik der Universität Rostock unter Leitung von Prof. Dr.-Ing. habil. Heidrun Schumann und des Lehrstuhls für Strukturmechanik der Fakultät für Maschinenbau und Schiffstechnik der Universität Rostock unter Leitung von Prof. Dr.-Ing. habil. U. Röhr.

Ein erstes Treffen der beteiligten Parteien erfolgte am 4.11.2005. Selbstgewählter Abgabetermin der vorliegenden Arbeit ist der 1.3.2006.

## 1.2 Inhalt des KSWS

Worum es in diesem KSWS geht:

Anfängliche Aufgabe des KSWS war es, eine grafische Benutzeroberfläche (im folgenden kurz als „GUI“ bezeichnet) für eine bereits vorhandene Software zu entwickeln. Die bereits vorhandene Software sollte eine Berechnung und eine Datenbank beinhalten, welche in FORTRAN (genauer: FORTRAN77) geschrieben wurden.

Bei der Entwicklung des Programms sollte auf gewisse Randbedingungen geachtet werden, welche maßgebend für den Erfolg des Projektes sein würden. Diese waren die einzigen konkreten Vorgaben, die zu Beginn feststanden.

Die zu erfüllenden Randbedingungen waren folgende:

- Es sollte zwei Sichten auf das Programm geben: Eine Sicht sollte einem Ingenieur gerecht werden, der die Datenbank zu verwalten hat. Eine andere Sicht sollte speziell auf weniger hoch qualifizierte Arbeitskräfte ausgerichtet sein, welche die Software im Alltag verwenden sollen. Besonderes Augenmerk sollte auf die intuitive Bedienbarkeit bei der zweiten Sicht gelegt werden.
- Die Software sollte vor allem auf der Plattform „Windows PC“ eingesetzt werden.
- Auf ein bekanntes Framework zur Entwicklung von GUIs mittels C++ sollte aus Kosten- und Urheberrechtsgründen verzichtet werden.

Der Studierende wählte als Lösung der Randbedingungen folgenden Ansatz:

- Es wurden zwei Programme entwickelt: „BendManager“ und „BendWorker“, welche den Ansprüchen der beiden Sichtweisen auf das Programm genügen sollten.
- Die Programmierung erfolgte mit dem .NET Framework, Version 1.1, in der Sprache C#.

## 1.3 Weitere Informationen

Wie man der technischen Dokumentation entnehmen kann, war eine Hauptschwierigkeit bei der Durchführung des Projektes die Kopplung einer modernen Software mit dem FORTRAN-Programm und ganz allgemein die zahlreichen Unzulänglichkeiten dieses Programmes, was die Struktur, aber auch die Funktionalität der Software anbelangt. Ebenfalls beeinträchtigend, aber auch sehr lehrreich, war der Zwang zur Einarbeitung in eine neue Programmiersprache.

## 2. Benutzerhandbuch zu BendManager

### 2.1 Installations-Hinweise

BendManager erfordert das .NET Framework, Version 1.1 oder höher.

Es kann demzufolge unter Windows (bis inklusive Windows 98) betrieben werden, solange das .NET Framework installiert wurde. Außerdem ist eine Version von .NET für Linux erhältlich, die den Namen „Mono“ trägt.

Falls das .NET Framework noch nicht installiert wurde, bestehen folgende Möglichkeiten:

- Es besteht zum einen die Möglichkeit, .NET bei Bedarf selbst von den Webseiten der Firma Microsoft herunterzuladen und zu installieren.
- Oder aber die beiliegende Datei „**dotnetfx.exe**“ zu starten. Diese beinhaltet eine Version des .NET Framework in deutsch, Version 1.1.
- Eine Installations-Software (kurz „Installer“), welche das .NET Framework und BendManager in einem ganzheitlichen Vorgang installiert, ist bereits in Arbeit und wird mit einer zukünftigen Version des Programms erscheinen.

Wenn die zuvor genannte Voraussetzung erfüllt ist, kann jetzt BendManager installiert werden. Dies kann wieder auf zwei Arten erfolgen:

- Eigenhändiges Kopieren der Dateien vom Datenträger auf die Festplatte des Zielrechners und eigenhändiges Erstellen und Kopieren von Verknüpfungen an die bevorzugte Stelle.
- Benutzung der Installations-Software (kurz: „Installer“), welche unter dem Namen „**setup.exe**“ auf dem Datenträger vorliegt.

### 2.2 Allgemeines zur Benutzung von BendManager

Der Zweck und die Fähigkeiten des Programms „BendManager“ lassen sich wie folgt beschreiben:

Das Programm ermöglicht die Bearbeitung und Verwaltung einer Datenbank, die Messdaten nach einem bestimmten Schema speichert, welches auf das Problem (Biegung von Materialien) zugeschnitten ist. Außerdem erleichtert das Programm die visuelle Auswertung und Beurteilung der Messdaten durch den Benutzer.

Das Programm soll dabei helfen, neue Messdaten auf bequeme Art in die Datenbank einzufügen und stellt dann die Kopplung zu dem FORTRAN-Programm her, welches die Berechnung einiger Metainformationen und die Errechnung von Parametern für Regressionskurven durch die Messpunkte vornimmt.

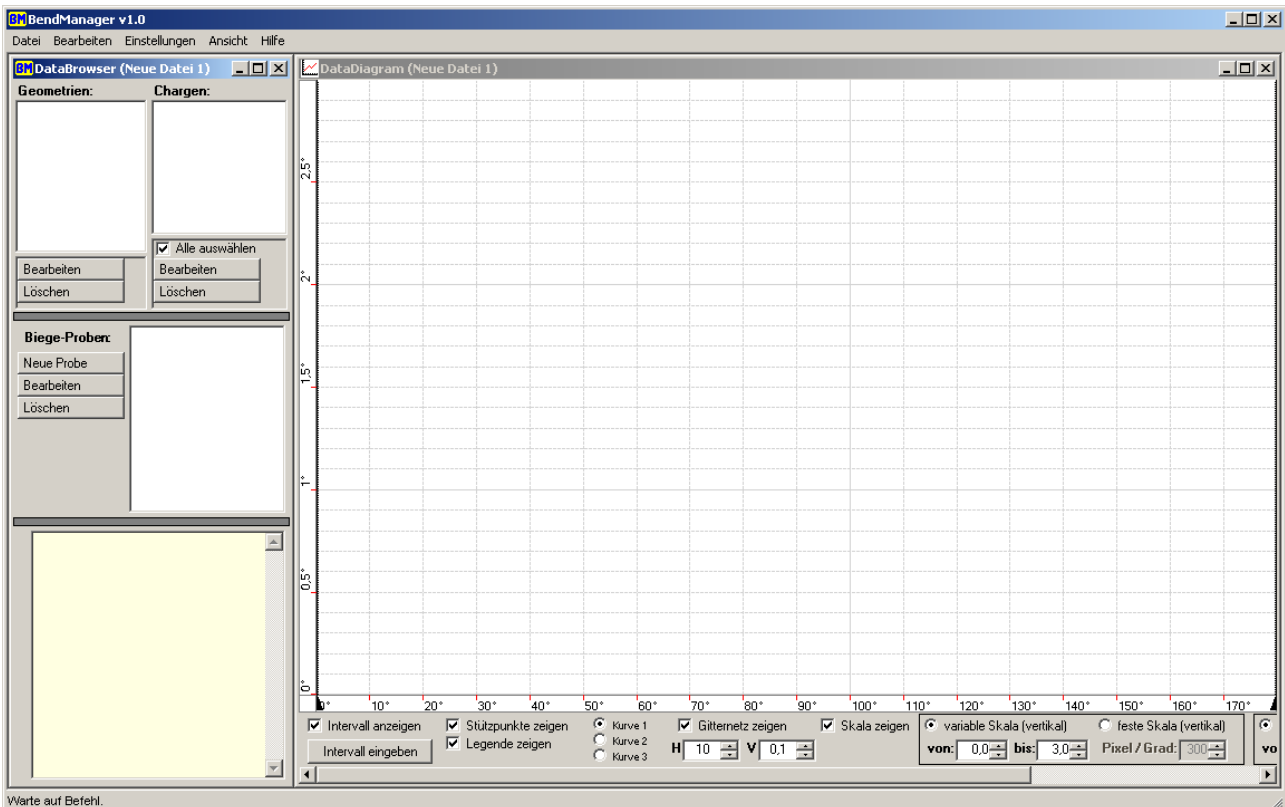
#### Start von BendManager:

Das Programm wird gestartet, indem entweder die Datei „BendManager.exe“ ausgeführt wird, oder indem eine entsprechend angelegte Verknüpfung gestartet wird. Das Programm zeigt ein Fenster mit einer Menüleiste, in der es auch einen Menüpunkt „Datei“ gibt. Diesen kann man nun benutzen, um eine vorhandene Datei zu öffnen oder eine neue Datei zu erstellen.

### Typischer Arbeitsablauf:

Anmerkung: Zum Verständnis der folgenden Ausführungen kann man die unten gezeigte Abbildung, sowie die folgenden Kapitel in dieser Dokumentation benutzen.

Üblicherweise wird man wie folgt beginnen: Im Menüpunkt „Datei“ wird eine Datei geöffnet oder aus einer Quelle importiert. Dann können die Daten mit Hilfe des „DataBrowser“ und eines interaktiven Diagramms untersucht werden. Falls erforderlich, kann man durch den Menüpunkt „Bearbeiten“ oder den „DataBrowser“ die Daten der Datenbank ändern oder neue Daten eingeben.



## 2.3 Menüpunkt „Datei“

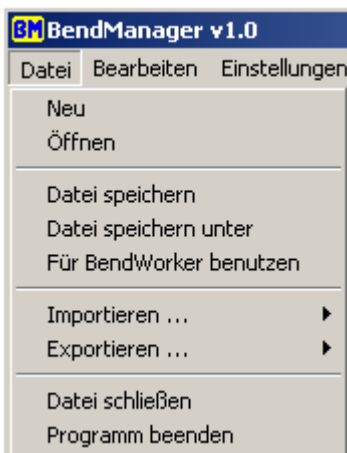


Abbildung 1: Menüpunkt "Datei" in BendManager

Wie in der Abbildung zu sehen ist, gibt es eine Reihe Unterpunkten, die da wären:

**Neu** – erstellt eine neue, leere Datenbank.

**Öffnen** – öffnet eine BendManager-Datei (Typ: „bm1“)

**Datei speichern** – speichert die geöffnete Datenbank in einer BendManager-Datei (Typ: „bm1“) ab.

**Datei speichern unter** – speichert die Datei, aber mit neuem Namen.

**Für BendWorker benutzen** – kopiert die Datei in das Verzeichnis „workfiles“ unter dem Namen „Arbeitsdatei.bm1“ (beziehungsweise dem Standard-Namen für Arbeitsdateien, siehe Dokumentation zu BendWorker).

**Importieren** – lädt eine Datenbank aus anderen Dateiformaten. Momentan wird jedoch nur das Format „basis.dat“ von Dr. Möller unterstützt.

**Exportieren** – speichert die Datenbank in einem anderen

Dateiformat. Momentan wird jedoch nur das Format „basis.dat“ von Dr. Möller unterstützt.

**Datei schließen** – schließt die geöffnete Datei (gleicher Effekt wie „Programm beenden“)

**Programm beenden** – beendet das Programm (mit Sicherheits-Abfrage zum Speichern der Datei).

## 2.4 Menüpunkt „Bearbeiten“

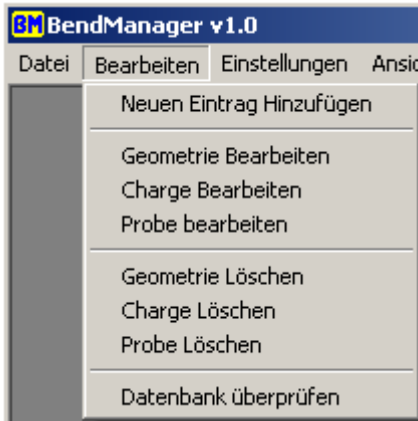


Abbildung 2: Menüpunkt "Bearbeiten" in BendManager

Achtung:

Alle folgenden Programm-Befehle sind nur dann sinnvoll, wenn eine BendManager-Datei geöffnet wurde. Dazu wird üblicherweise einer der Befehle aus dem Menüpunkt „Datei“ verwendet.

Einige der Befehle sind auch nur sinnvoll, wenn ein DataBrowser geöffnet wurde. Nähere Informationen dazu findet man im Kapitel „Der DataBrowser“.

**Neuen Eintrag Hinzufügen** – es wird ein Dialog zum Eingeben eines neuen Datenbank-Eintrages geöffnet.

**Geometrie Bearbeiten** – Die aktuell ausgewählte Geometrieklasse kann bearbeitet werden. Man kann einen neuen Namen, sowie einen Biegeradius eingeben.

**Charge Bearbeiten** – Die aktuell ausgewählte Chargenklasse kann bearbeitet werden. Man kann eine neue Nummer für die

Material-Charge eingeben.

**Probe Bearbeiten** – Hier ist die Änderung der konkret erfassten Messwerte möglich, sowie die Änderung der Meta-Informationen (Probe-Name).

**Geometrie Löschen, Charge Löschen, Probe Löschen** – Diese Programm-Befehle führen dazu, dass ein oder mehrere Einträge aus der Datenbank entfernt werden.

**Datenbank überprüfen** – Veranlasst das Programm von Dr.Möller, einen kompletten Datenbank-Check hinsichtlich seiner Kriterien zur Korrektheit durchzuführen.

## 2.5 Menüpunkt „Einstellungen“



Abbildung 3: Menüpunkt "Einstellungen" in BendManager

Hier gibt es bisher nur eine einzige Option, mit der man kleinere Einstellungen vornehmen kann, zum Beispiel ist es möglich, den bevorzugten Typ der Ausgleichskurve festzulegen oder die Diagramm-Farben den eigenen Wünschen anzupassen.

## 2.6 Menüpunkt „Ansicht“

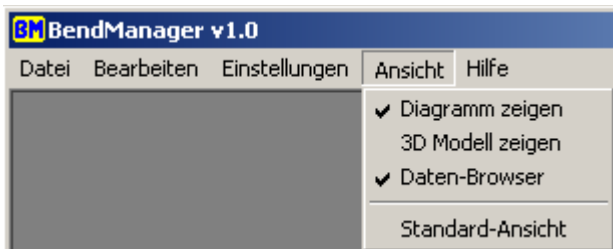


Abbildung 4: Menüpunkt "Ansicht" in BendManager

Es kann ausgewählt werden, welche Diagramme gezeigt werden.

Es wird empfohlen, die Option

**Data-Browser** immer ausgewählt zu lassen, da der DataBrowser das wichtigste Instrument darstellt.

Mit **Standard-Ansicht** kann man die gewohnte Ansicht jederzeit wieder herstellen.

## 2.7 Menüpunkt „Hilfe“



Abbildung 5: Menüpunkt "Hilfe" in BendManager

Der Punkt **Hilfe zeigen** bietet im jetzigen Stadium nur eine rudimentäre Hilfe, die von dieser Dokumentation in der Nützlichkeit bei weitem übertroffen wird.

Der Punkt **Online-Hilfe**, der einem die Möglichkeit bietet, Kontaktadressen von Zuständigen und Verantwortlichen zu

bekommen ist möglicherweise der nützlichste Punkt.

Der Punkt **Info (Impressum)** entspricht dem bei Software üblichen „About“, mit dem sich die Programm-Version und Informationen über den oder die Programmierer abfragen lassen.

## 2.8 Der DataBrowser

In der Abbildung ist der geöffnete DataBrowser zu sehen (siehe auch Menüpunkt „Ansicht“). Dies ist das Standard-Werkzeug, bei der Untersuchung der Daten. Er bietet folgende Funktionen:

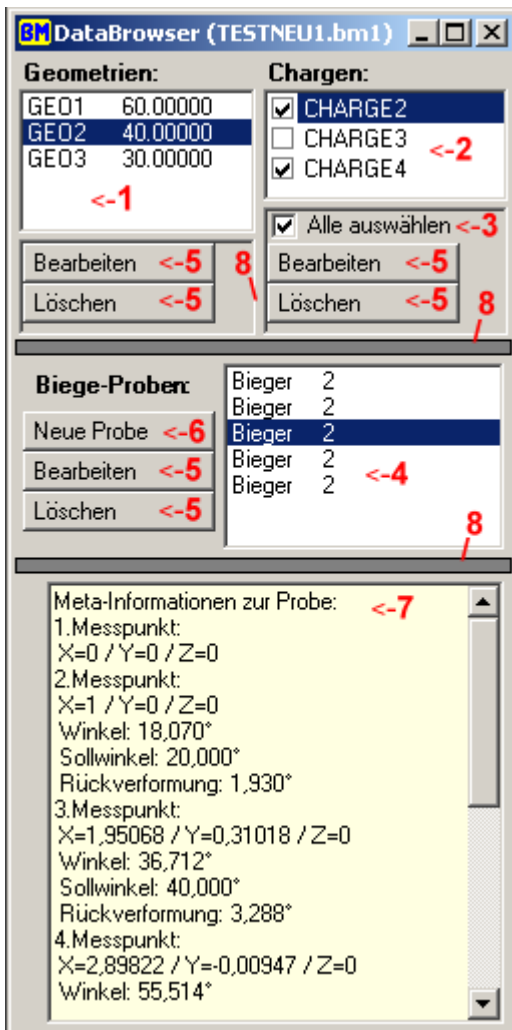


Abbildung 6: Der DataBrowser in BendManager

- 1) Auswahlfeld für die Geometrie-Klasse.
- 2) Auswahlfeld für die Charge – die Chargen können zusätzlich mit einem Häkchen markiert werden, was dazu führt, dass sie in einem interaktiven Diagramm angezeigt werden (siehe nächstes Kapitel).
- 3) Markiert alle Chargen mit einem Häkchen.
- 4) Auswahlfeld für die Biege-Proben.
- 5) Bearbeiten / Löschen der jeweils ausgewählten Geometrieklasse / Charge / Biege-Probe, genau wie mit dem Menüpunkt „Bearbeiten“ im Hauptfenster.
- 6) Dieser Knopf entspricht dem Menüpunkt „Bearbeiten -> Neuen Eintrag Hinzufügen“ aus dem Hauptfenster der Anwendung.
- 7) Dieses Textfeld liefert ein paar Informationen zu den Daten, die in den Auswahlfeldern ausgewählt wurden.
- 8) Mit diesen Schiebepalken kann das Aussehen des gesamten DataBrowser den Erfordernissen angepasst werden.



## 2.9 Das interaktive Diagramm

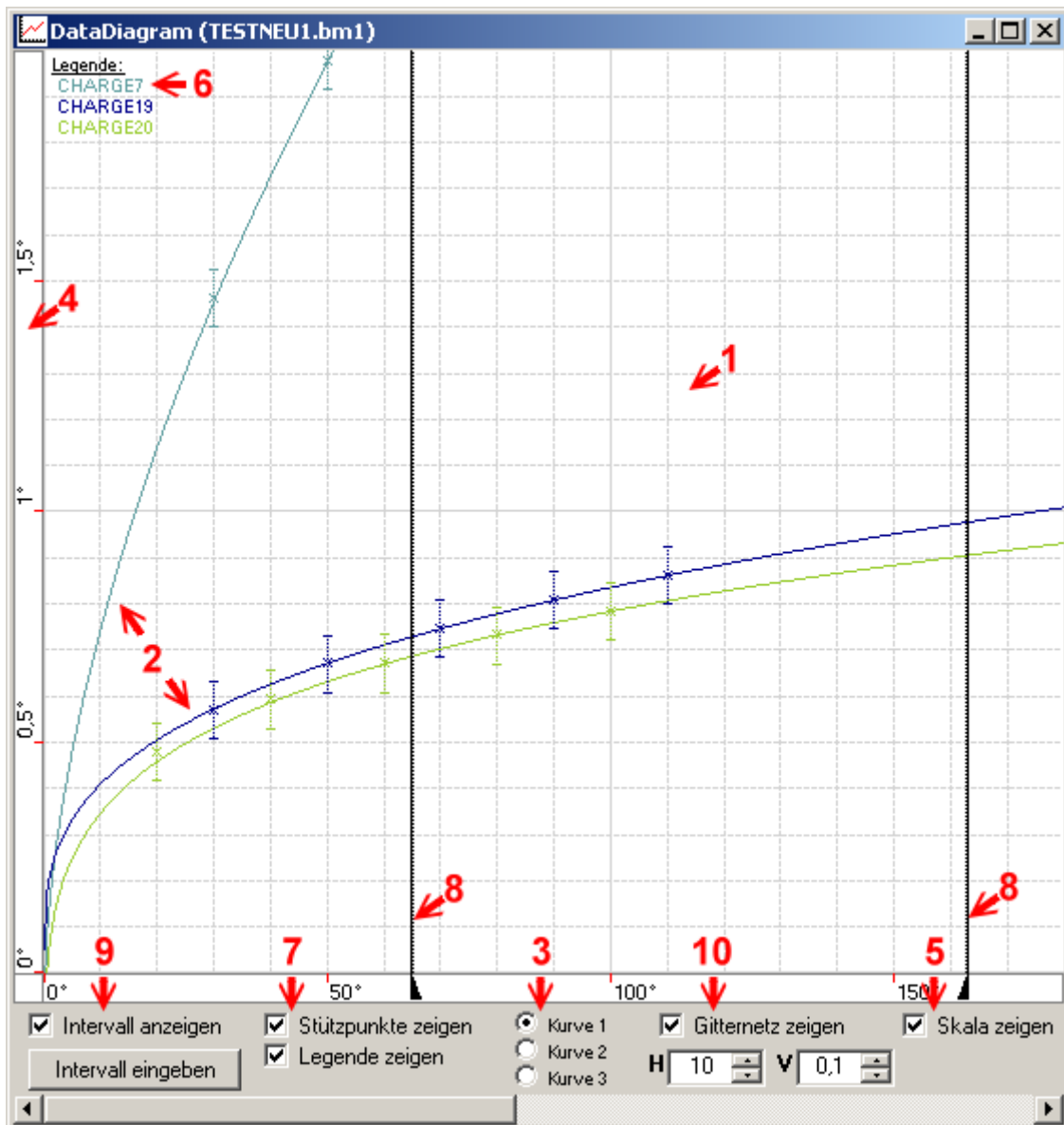


Abbildung 7: Das interaktive Diagramm von BendManager

Folgende Funktionen bietet das Diagramm an:

- 1) Dies ist das eigentliche Diagramm. Mit einem doppelten Mausklick kann man es auf die maximale Größe vergrößern (der untere, graue Teil wird ausgeblendet / eingeblendet).
- 2) Dies sind die Kurven. Es werden Kurven zu allen Material-Chargen angezeigt, die im DataBrowser geöffnet wurden und die durch das FORTRAN-Programm berechnet werden konnten. Wenn nicht genug Datenwerte für eine Berechnung vorliegen und keine Kurve berechnet werden kann, wird sie auch nicht angezeigt, nur die Stützpunkte (siehe 7).
- 3) Es stehen drei Kurventypen zur Auswahl, die das Programm kennt. „Kurve3“ steht z.B. für den dritten Kurventyp, der eine lineare Regressions-Gerade darstellt.
- 4) Diese Skale zeigt den Definitions- und Wertebereich an und passt sich an die Größe an.

- 5) Hiermit wird die Skala eingeblendet / ausgeblendet.
- 6) Diese Legende zeigt die Namen der ausgewählten Chargen in den jeweiligen Farben der zugehörigen Kurve an, damit eine Zuordnung bei mehr als einer Kurve möglich ist.
- 7) Mit diesen beiden Schaltern kann man die Legende ein- und ausblenden beziehungsweise die Stützpunkte der Kurven anzeigen / weglassen. Wenn man die Stützpunkte anzeigen lässt, wird der Mittelwert dargestellt, sowie das obere und das untere Streuungsintervall.
- 8) Diese beiden „Kämme“ stellen die Grenzen des Intervalls dar, aus dem Punkte für die Berechnung herangezogen werden. Das FORTRAN-Programm ist nämlich in der Lage, auf diese Weise den Bereich für die Berechnung festzulegen. Man kann die „Kämme“ mit gedrückter linker Maustaste per Drag&Drop an die passende Stelle ziehen oder mit einem Rechtsklick auf einen der beiden Kämme einen Dialog zur Eingabe von exakten Werten öffnen.
- 9) Mit diesen Schaltern kann man sich entscheiden, ob man die Intervallgrenzen sehen möchte, beziehungsweise einen Dialog zur Eingabe von Werten für das Intervall öffnen.

Wenn man den Laufbalken am unteren Rand des Diagramms bewegt, sieht man noch weitere Optionen, die das Diagramm hat:

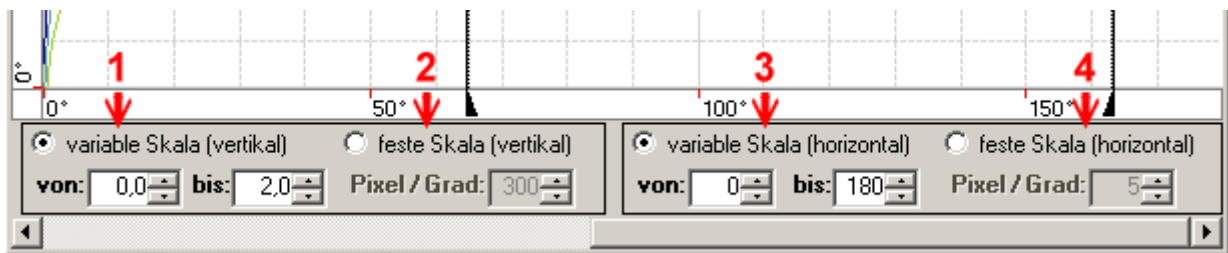


Abbildung 8: Weitere Optionen im Diagramm von BendManager

- 1) Falls man sich für diese Option entscheidet, wird die Y-Achse immer so skaliert, dass sie von einer festgelegten Untergrenze (z.B. 0°) bis zu einer festgelegten Obergrenze (z.B. 2°) reicht. Es ist egal, ob man das Diagramm vergrößert oder verkleinert, die Y-Achse wird sofort neu skaliert und angepasst.
- 2) Falls man sich für diese Option entscheidet, gibt es einen festen Maßstab für die Y-Achse. Zum Beispiel würden bei einer Eingabe von 300 immer genau 300 Pixel einen Abstand von einer Einheit (also 1°) bilden. Das ist in gewissen Situationen, z.B. beim Vergleich von zwei Diagrammen bisweilen sinnvoll und nützlich.
- 3) Falls man sich für diese Option entscheidet, wird die Y-Achse immer so skaliert, dass sie von einer festgelegten Untergrenze (z.B. 0°) bis zu einer festgelegten Obergrenze (z.B. 180°) reicht. Es ist egal, ob man das Diagramm vergrößert oder verkleinert, die X-Achse wird sofort neu skaliert und angepasst.
- 4) Falls man sich für diese Option entscheidet, gibt es einen festen Maßstab für die X-Achse. Zum Beispiel würden bei einer Eingabe von 5 immer genau 5 Pixel einen Abstand von einer Einheit (also 1°) bilden. Das ist in gewissen Situationen, z.B. beim Vergleich von zwei Diagrammen bisweilen sinnvoll und nützlich.

## 3. Benutzerhandbuch zu BendWorker

### 3.1 Installations-Hinweise

BendWorker erfordert das .NET Framework, Version 1.1 oder höher.

Es kann demzufolge unter Windows (bis inklusive Windows 98) betrieben werden, solange das .NET Framework installiert wurde. Außerdem ist eine Version von .NET für Linux erhältlich, die den Namen „Mono“ trägt.

Falls das .NET Framework noch nicht installiert wurde, bestehen folgende Möglichkeiten:

- Es besteht zum einen die Möglichkeit, .NET bei Bedarf selbst von den Webseiten der Firma Microsoft herunterzuladen und zu installieren.
- Oder aber die beiliegende Datei „**dotnetfx.exe**“ zu starten. Diese beinhaltet eine Version des .NET Framework in deutsch, Version 1.1.
- Eine Installations-Software (kurz „Installer“), welche das .NET Framework und BendWorker in einem ganzheitlichen Vorgang installiert, ist bereits in Arbeit und wird mit einer zukünftigen Version des Programms erscheinen.

Wenn die zuvor genannte Voraussetzung erfüllt ist, kann jetzt BendWorker installiert werden. Dies kann wieder auf zwei Arten erfolgen:

- Eigenhändiges Kopieren der Dateien vom Datenträger auf die Festplatte des Zielrechners und eigenhändiges Erstellen und Kopieren von Verknüpfungen an die bevorzugte Stelle.
- Benutzung der Installations-Software (kurz: „Installer“), welche unter dem Namen „**setup.exe**“ auf dem Datenträger vorliegt.

### 3.2 Allgemeines zur Benutzung von BendWorker

Der Zweck und die Fähigkeiten des Programms „BendWorker“ lassen sich wie folgt beschreiben:

Das Programm ist für den praktischen Einsatz in einer Werkhalle konzipiert, in welcher an einer speziellen Maschine in einem semi-automatischen Verfahren Metallgegenstände gebogen werden.

Diese Software soll den Werksarbeiter dabei unterstützen, schneller und genauer zum Ziel zu kommen, als das bisher der Fall war.

Um dieses Ziel zu erreichen, greift das Programm auf eine Datenbank zurück, in welcher die Eigenschaften von verschiedenen Materialarten gespeichert sind. Es sagt dann, basierend auf dieser Datenbank voraus, welche genauen Einstellungen (Winkel bzw. CNC-Korrektur) man an der Maschine vornehmen müsste, um den materialabhängigen Eigen-Rückformungen gezielt entgegenzuwirken.

#### Start von BendWorker:

Das Programm wird gestartet, indem entweder die Datei „BendWorker.exe“ ausgeführt wird, oder indem eine entsprechend angelegte Verknüpfung gestartet wird. Das Programm zeigt ein Fenster mit mehreren Auswahlmöglichkeiten. Dieses Fenster wird als „die Startseite“ bezeichnet.

### 3.3 Die Startseite

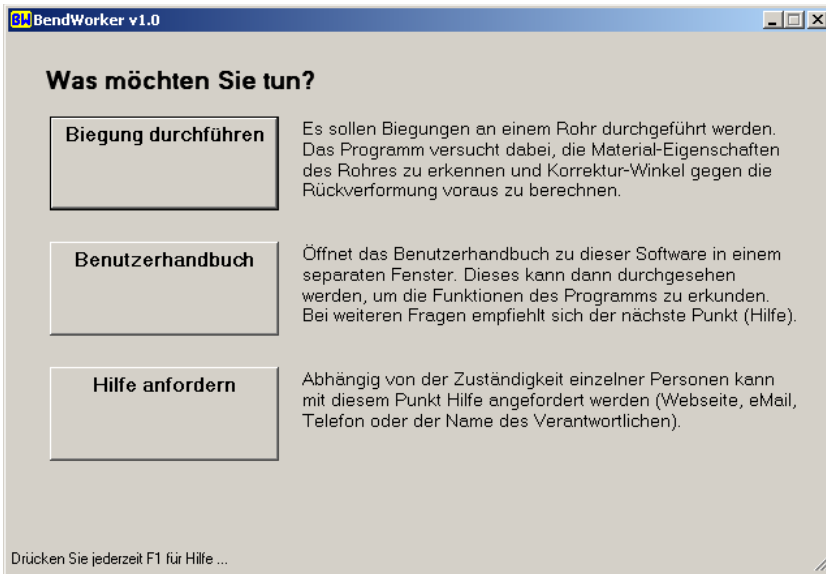


Abbildung 9: Die Startseite von BendWorker

Die Startseite bietet die Möglichkeit, zwischen 3 verschiedenen Optionen zu wählen:

- **Biegung durchführen** – normalerweise sollte dieser Punkt gewählt werden. Eine Reihe von weiteren Dialogen begleitet den Nutzer der Software durch den Prozess der Biege-Korrektur.
- **Benutzerhandbuch** – es wird eine HTML-Version dieser Dokumentation geladen.
- **Hilfe anfordern** – Webseite, eMail, Telefonnummer oder Name von Auskunfts-Personen werden angezeigt. Dies ist notwendig für den Fall einer Störung im Programm oder für den Fall, dass die Datenbank erweitert werden muss, weil ein vollkommen neues Material mit gänzlich eigenen Eigenschaften verwendet werden soll.

### 3.4 Auswahl „Biegung durchführen“

Nachdem die Option „Biegung durchführen“ gewählt wurde, wird ein Arbeits-Prozess gestartet, der durch eine Reihe von Dialogfenstern begleitet wird. Im großen und ganzen lässt sich der gesamte Prozess wie folgt skizzieren:

1. Zuerst gibt der Arbeiter an, welches Material für die Biegung verwendet werden soll. Falls das Material das selbe wie bei der letzten Biegung ist, muss hier nichts eingestellt werden.
2. Falls das Material dem Programm noch unbekannt ist, wird ein Hinweis ausgegeben, dass die Datenbank erweiterungsbedürftig ist. Es kann dennoch versucht werden, ein Material zu finden, das zumindest ähnliche Eigenschaften besitzt wie das neue Material. Für diese Erkennung müssen einige wenige Test-Biegungen durchgeführt werden. Falls der Vorgang erfolgreich war, kann normal weitergearbeitet werden. Das Programm merkt sich die Ergebnisse dieser Test-Biegung und kennt fortan das neue Material.
3. Zum Schluss gibt das Programm je nach Wunsch entweder 4 Werte an, die in das CNC-Programm der Biege-Maschine eingegeben werden müssen oder es sagt selbst voraus, welchen Winkel man einstellen müsste, um einen gewünschten Zielwert zu erreichen.

### 3.4.1 Die Material-Auswahl

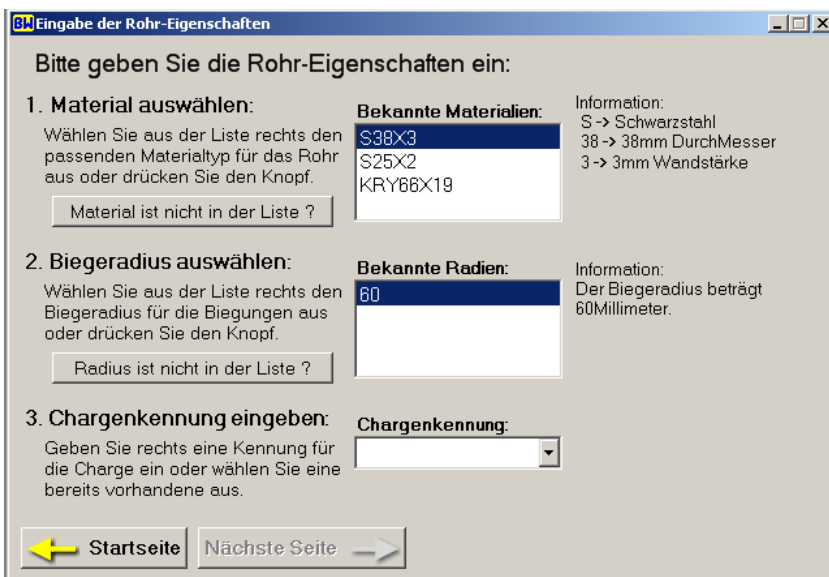


Abbildung 10: Dialogfenster "Material-Auswahl" in BendWorker

Aus einer Liste von bekannten Materialarten kann in diesem Dialog ausgewählt werden. Dabei gibt es 3 grundsätzliche Komponenten, welche nur gemeinsam, als Einheit, eine Aussage machen können über das Verhalten des Materials:

- Das „Material“ - wiederum eine Kombination verschiedener Werte, gibt an, ob es sich zum Beispiel um einen Rohling aus Stahl oder Kupfer handelt und welche Dicke oder Durchmesser der Rohling hat.
- Der „Biegeradius“ - ein Radius, der die Krümmung der Biegung bestimmt. Bei einem kleinen Radius ist die Biegung ein Knick, bei einem großen Radius eher eine Kurve.
- Die „Chargenkennung“ - steht für die Materialcharge, also eine Menge von Rohlingen, die aus der gleichen Grundproduktion stammen und gemeinsam angeliefert wurden.

Bei der Eingabe der Charge kann es zu einem von drei Fällen kommen:

**Chargenkennung:** 528986

Information: Diese Charge-Nummer ist dem Programm bekannt. Sie liegt in der Datenbank vor (und wurde gründlich getestet).

Im ersten Fall ist alles in Ordnung – das bedeutet, die Datenbank des Programms kennt das Material und seine Eigenschaften ganz genau.

**Chargenkennung:** 1111111

ACHTUNG: Diese Charge-Nummer ist zwar dem Programm bekannt. Sie stammt jedoch aus einer schnellen Testbiegung. Sie sollten die Datenbank bei Gelegenheit erweitern!

In einem anderen Fall kennt das Programm zwar das Material, jedoch nur aus einer flüchtigen Testbiegung, die von einem Arbeiter vor Ort durchgeführt wurde, was die Präzision senkt.

**Chargenkennung:** 999999

ACHTUNG: Die Charge-Nummer ist dem Programm noch unbekannt. Es müssen gleich ein paar Testbiegungen durchgeführt werden.

Bei Eingabe eines vollkommen neuen Materials wurde weder eine erfolgreiche Test-Biegung durchgeführt, noch ist das Material irgendwo in der Datenbank abgelegt. Es ist halt unbekannt.

Im letzten der drei Fälle (unbekanntes Material) muss entweder die Datenbank erweitert werden oder eine schnelle Test-Biegung wird durchgeführt (siehe nächstes Kapitel).

### 3.4.2 Die Test-Biegung

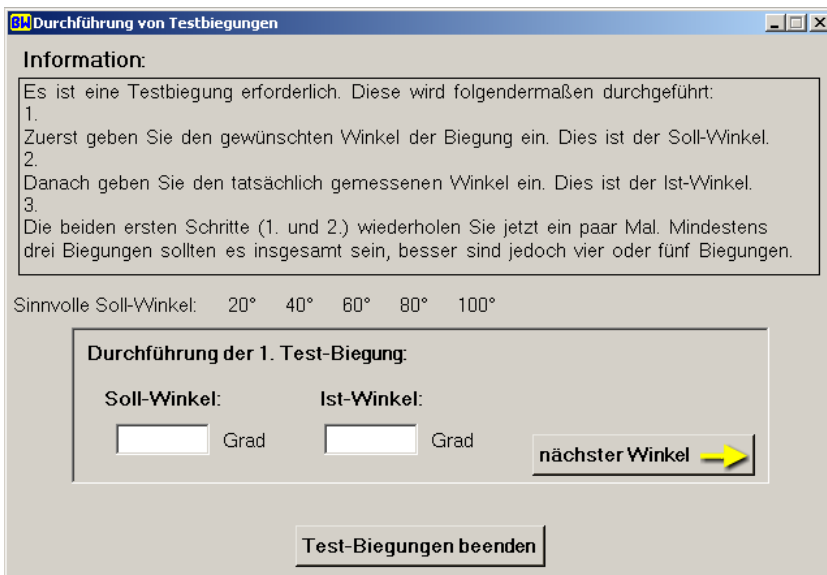


Abbildung 11: Dialogfenster "Test-Biegung" in BendWorker

Für die Erkennung eines Materials, das nicht in der Datenbank liegt, kann eine Testbiegung durchgeführt werden.

Der Nutzer muss dazu angeben, welchen Winkel er erwarten würde und danach, welchen Winkel er nach einer einzigen Biegung festgestellt hat. Dies wird für ein paar Winkel durchgeführt, es müssen also ein paar Testbiegungen durchgeführt werden.

Dadurch erhöht sich die Chance auf eine erfolgreiche Erkennung.

### 3.4.3 Die Modus-Auswahl

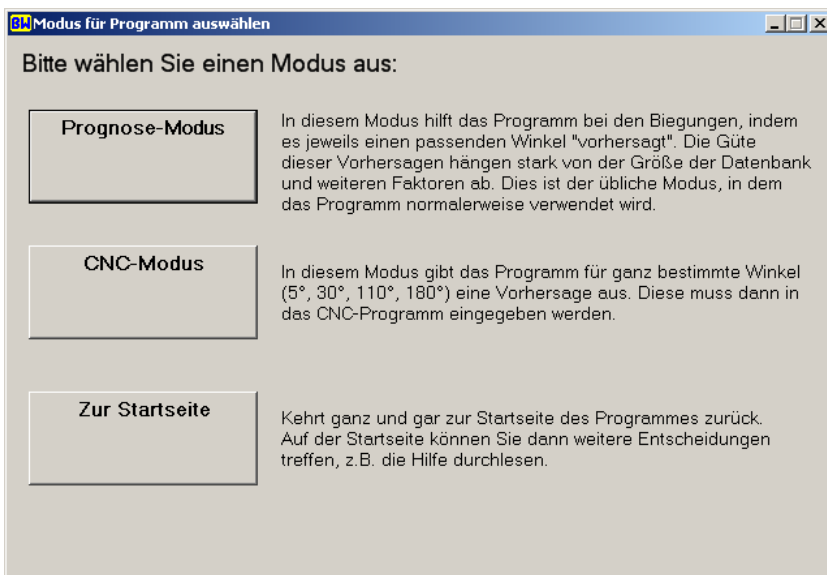


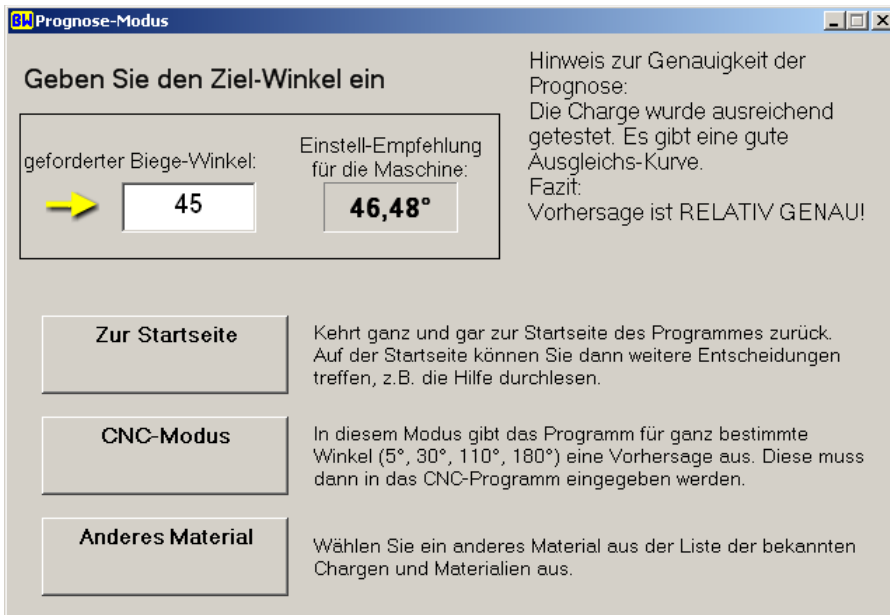
Abbildung 12: Dialogfenster "Modus-Auswahl" in BendWorker

Es kann zwischen zwei verschiedenen Modi gewählt werden:

Der **Prognose-Modus** spielt die gesamten Fähigkeiten des Programmes aus. Dies ist der normale Modus (siehe nächstes Kapitel).

Es gibt aber noch den sogenannten **CNC-Modus**, der ein paar Werte zur Justierung der Maschine angibt. Dadurch werden eventuell 4 Biegungen eingespart und die Präzision der Einstellung erhöht.

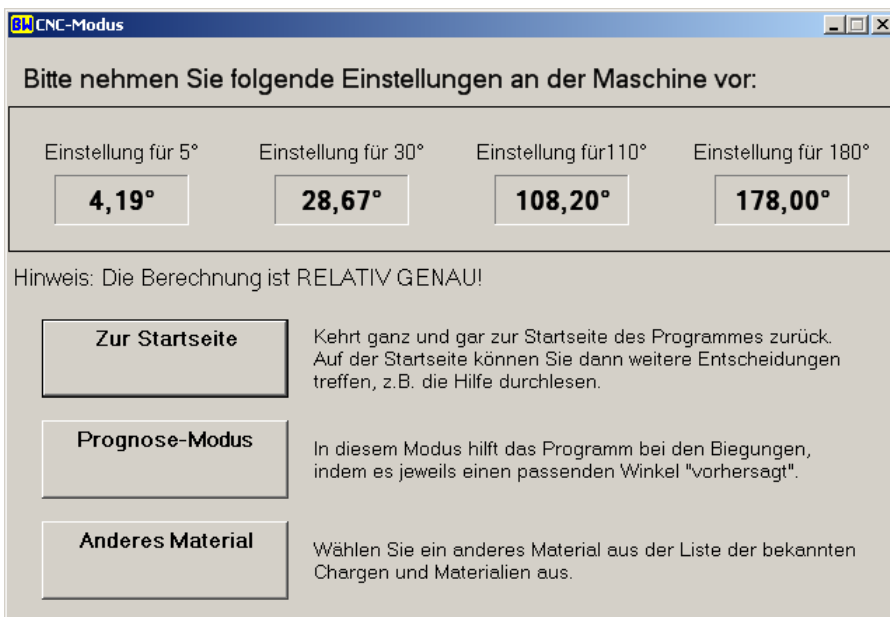
### 3.4.4 Der Prognose-Modus



Es muss ein Ziel-Winkel für die Biegung eingegeben werden. Das Programm findet dann in Echtzeit den Winkel, den man einstellen muss, damit (theoretisch) am Ende der gewünschte Zielwinkel erreicht wird. Dazu benutzt BendWorker die mathematischen Kurven, die in der Datenbank gespeichert wurden. Rechts oben im Dialogfenster wird ein Hinweis zur Genauigkeit der Berechnung angezeigt.

Abbildung 13: Dialogfenster "Prognose-Modus" in BendWorker

### 3.4.5 Der CNC-Modus



Der CNC-Modus liefert diejenigen Werte, welche nach Meinung des Programms eingestellt werden müssten, damit man die gleichen Ergebnisse erhält, als würde man die 4 notwendigen Biegungen jetzt durchführen und die gemessenen Ergebnisse experimentell bestimmen.

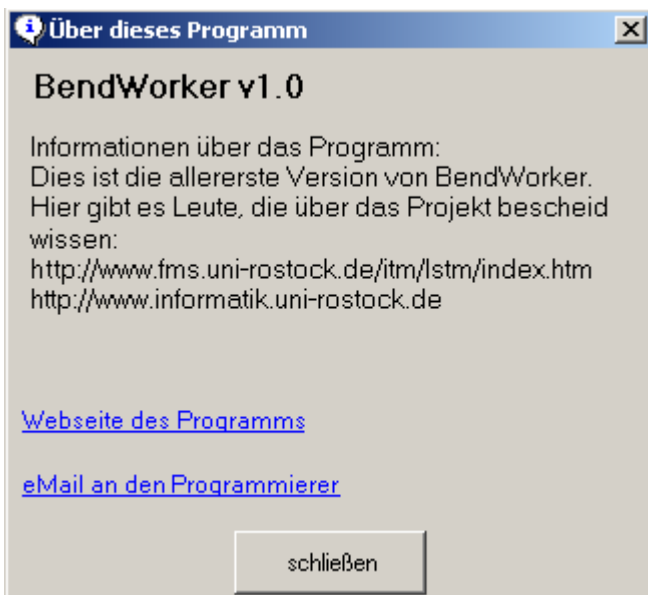
Abbildung 14: Dialogfenster "CNC-Modus" in BendWorker

## 3.5 Auswahl „Benutzerhandbuch“

Öffnet eine gekürzte Fassung dieser PDF-Datei.

### 3.6 Auswahl „Hilfe anfordern“

Öffnet momentan nur ein kleines Impressum mit möglichen Adressen für Ansprechpartner zu diesem Programm:





## 4. Technische Dokumentation

### 4.1 Allgemein

Momentan sieht die Struktur der Software folgendermaßen aus:

Das FORTRAN-Programm schreibt Textdateien auf die Festplatte, welche dann vom C#-Programm geparkt und importiert werden. Die Hauptfunktion des FORTRAN-Programms ist es, Berechnungen anzustellen (Ausgleichskurvenberechnung) und neue Einträge in die Datenbank aufzunehmen. Außerdem besitzt es noch einen Befehl für die Klassifikation von Test-Biegungen, die mit BendWorker durchgeführt wurden.

Auch die Kommandos an das FORTRAN-Programm erfolgen über Textdateien, die RPARAM.DAT heißen (siehe Dokumentation zum Rohrbiegen-Programm von Dr.Möller).

Die Datenbank-Datei BASIS.DAT wird dabei jedes Mal vom C#-Programm erzeugt, da dieses über weitreichende Datenbank-Kommandos wie das Entfernen von Datenbank-Einträgen verfügt, die das FORTRAN-Programm nicht bietet.

Es ist also anzumerken, dass die Aufgabenstellung des KSWS um die Entwicklung eines Text-Parsers und einer Datenbank erweitert werden musste, um die Zielstellung zu erreichen.

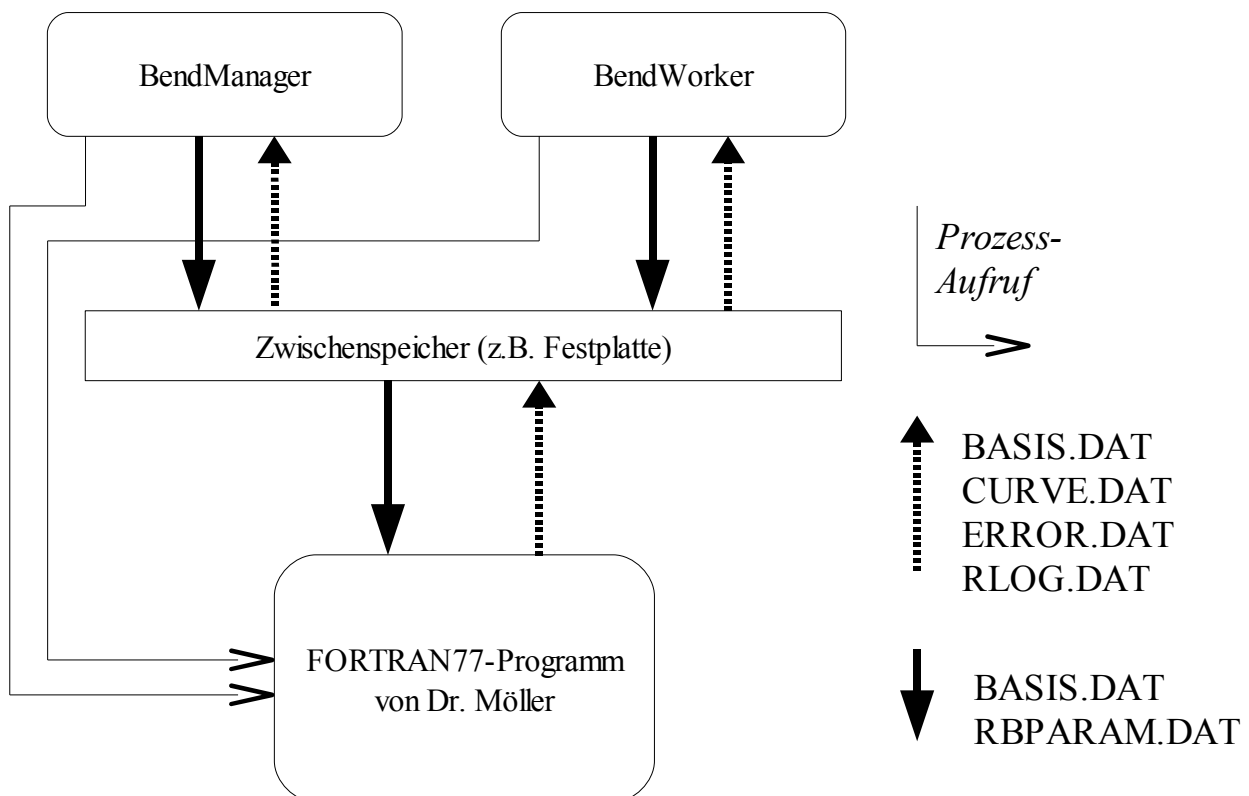


Abbildung 15: Struktur der gesamten Software

Die Nachteile dieser Vorgehensweise sind zwar offensichtlich, jedoch wurden die Vorschläge, auf eine neuere FORTRAN-Version umzusteigen oder eine der drei für .NET verfügbaren Compiler zu nehmen (siehe Anhang B - „FORTRAN Compiler für .NET“) abgelehnt. Eine ordentliche Fehlerbehandlung ist zum Beispiel dadurch wesentlich erschwert worden.

Das C#-Programm selbst ist folgendermaßen aufgebaut: Es gibt drei sogenannte Assemblies (das .NET-Äquivalent zu Bytecode-Sammlungen in class- und jar-Dateien), von denen zwei ausführbare Programme sind (BendManager.exe und BendWorker.exe).

Das dritte Assembly (BendLibrary.dll) ist (wie der Name schon andeutet) eine gemeinsam genutzte, dynamisch gelinkte Bibliothek von Klassen, die sowohl in BendManager als auch in BendWorker verwendet werden können.

Ursprünglich war lediglich ein Programm geplant, welches über einen Experten-Modus verfügen sollte, daher wurde eine klare Auftrennung des Designs erst vor ein paar Wochen durchgeführt.

Im folgenden wird eine Auflistung aller verwendeten Klassen und ihres Zweckes durchgeführt.

## 4.2 BendLibrary.dll

### 4.2.1 Auflistung aller Pakete und Kurzbeschreibung

Eine exakte Auflistung aller Pakete findet man auch in der Datei „Solution\_BendManager.HTM“ im Verzeichnis „CodeCommentReport“.

BendLibrary.App	<i>Hauptklassen, wie Hauptfenster und Datei</i>
BendLibrary.Config	<i>Konfiguration des Programms</i>
BendLibrary.Controls	<i>Eigene Benutzersteuerelemente</i>
BendLibrary.Data	<i>Alles, was sich näher mit Datenbank befasst</i>
BendLibrary.Dialogs	<i>Dialogfenster für diverse Aufgaben</i>
BendLibrary.Enums	<i>Aufzählungen</i>
BendLibrary.Events	<i>Ereignisbehandlung</i>
BendLibrary.Exceptions	<i>Ausnahmebehandlung</i>
BendLibrary.Helpers	<i>Kleinere Hilfsklassen</i>
BendLibrary.IO	<i>Daten-Eingabe und Daten-Ausgabe (für Text-Parser)</i>

### 4.2.2 Auflistung aller Klassen und Kurzbeschreibung

Eine exakte Auflistung aller Klassen findet man auch in der Datei „Solution\_BendManager.HTM“ im Verzeichnis „CodeCommentReport“.

App.AbstractMainWindow	<i>Abstrakte Oberklasse für Hauptfenster einer Anwendung</i>
App.BMFile	<i>Oberklasse für bw1-Dateien (IO, Kapselung Datenbank)</i>
App.TempChargesFile	<i>basis.neu-Datei (IO, Kapselung Daten aus Test-Biegungen)</i>
Config.Constants	<i>Konstanten, Standard-Namen, Globale Objekt-Vorlagen</i>
Config.ProgramConfiguration	<i>Singleton zur Kapselung allgemeiner Programm-Einstellungen</i>
Controls.DiagramCanvas	<i>Ein Diagramm zur Darstellung mathematischer Kurven</i>
Controls.WebLabel	<i>Eine komfortablere Ableitung von System.Controls.LinkLabel</i>

Data.CurveParameters	<i>Kapselung von Kurvenparametern (struct-ähnliche Klasse)</i>
Data.DataBase	<i>Die selbstgeschriebene Datenbank für das Problem</i>
Data.PipeCharge	<i>Kapselt Werte und Referenzen einer Charge</i>
Data.PipeGeometry	<i>Kapselt Werte und Referenzen einer Geometrie-Klasse</i>
Data.PipeSample	<i>Kapselt Werte und Referenzen einer Biege-Messung</i>
Data.TempCharges	<i>Kleine Mini-Datenbank, für Resultate von Test-Biegungen</i>
Data.TempEntry	<i>Eintrag in der Mini-Datenbank</i>
Dialogs.AboutDialog	<i>Dialog für Programmhinweise (wie bekanntes „About ...“)</i>
Dialogs.EditChargeDialog	<i>Dialog zum Ändern der Chargenkennung eines Eintrags</i>
Dialogs.EditGeometryDialog	<i>Dialog zum Ändern von Geometrienamen und Geometrieradien</i>
Dialogs.EditSampleDialog	<i>Dialog zum Ändern von Werten einer Biege-Messung</i>
Dialogs.ErrorDialog	<i>Dialog zum Anzeigen einer schwerwiegenden Fehlermeldung</i>
Dialogs.InfoDialog	<i>Dialog für Hinweise</i>
Dialogs.IntervalBordersDialog	<i>Dialog zur Eingabe von neuen Intervall-Grenzen</i>
Dialogs.NewSampleInput	<i>1. Teildialog bei Eingabe von neuen Biege-Messungen</i>
Dialogs.NewSampleInput2	<i>2. Teildialog bei Eingabe von neuen Biege-Messungen</i>
Dialogs.NewSampleInput3	<i>3. Teildialog bei Eingabe von neuen Biege-Messungen</i>
Dialogs.NewSampleInput4	<i>4. Teildialog bei Eingabe von neuen Biege-Messungen</i>
Dialogs.NewSampleInputDialog	<i>Fasst die Teildialoge 1 bis 4 zu einer Einheit zusammen</i>
Dialogs.PatienceDialog	<i>Dialog für Fortschrittsbalken</i>
Dialogs.SaveChangesDialog	<i>Dialog zur Sicherheitsabfrage („Änderung Speichern?“)</i>
Dialogs.WarningDialog	<i>Aggressiver Dialog mit deutlichem Warnhinweis</i>
Dialogs.YesNoDialog	<i>Dialog für JA / NEIN Entscheidungen</i>
Enums.DBAction	<i>Aufzählung möglicher Datenbank-Aktionen</i>
Events.DialogClosedEventArgs	<i>Ereignis für spezielle Situationen des Beendens von Fenstern</i>
Events.IntervalEventArgs	<i>Ereignis, wenn sich Intervallgrenzen geändert haben</i>
Exceptions.ParseFileException	<i>falls Parser Datei nicht erkennen kann (mit Zeilenangabe)</i>
Exceptions.UnknownStringException	<i>falls der angegebene String nicht bekannt ist</i>
Helpers.curveCND	<i>Struktur zur Kapselung von Kurvenparametern</i>
Helpers.FloatFormatter	<i>Stellt Format-Strings her, für formatierte Zahlenstrings</i>
Helpers.NewSample	<i>Kapselt Eingaben neuer Biege-Messung / Charge / Geometrie</i>
Helpers.NewSampleArray	<i>Sammelt und überwacht eine Reihe von Helpers.NewSample</i>
Helpers.xyz	<i>Primitive Vektorklasse, als Ersatz für fehlende Vec3-Klasse</i>
IO.StringExporter	<i>Schreibt streng formatierte Ausgaben (Zahlen etc. als Text)</i>

IO.PaddingStyle	<i>Lokale Aufzählung, für Art der Formatierung</i>
IO.StringFormater	<i>Stellt ähnlich wie Helpers.FloatFormatter Formatierung her</i>
IO.StringImporter	<i>Globale Funktionen zum Einlesen von formatierten Textdaten</i>

## 4.3 BendManager.exe

### 4.3.1 Auflistung aller Pakete und Kurzbeschreibung

Eine exakte Auflistung aller Pakete findet man auch in der Datei „Solution\_BendManager.HTM“ im Verzeichnis „CodeCommentReport“.

BendManager.App	<i>Hauptanwendung (Hauptfenster, Dateiklasse)</i>
BendManager.Config	<i>BendManager-spezifische Einstellungen</i>
BendManager.Dialogs	<i>Sammlung von Dialogfensterklassen</i>
BendManager.Forms	<i>Sammlung von Nicht-Dialog-Fenstern</i>

### 4.3.2 Auflistung aller Klassen und Kurzbeschreibung

Eine exakte Auflistung aller Klassen findet man auch in der Datei „Solution\_BendManager.HTM“ im Verzeichnis „CodeCommentReport“.

App.BendManagerFile	<i>Spezialisiert BendLibrary.App.BMFile für BendManager</i>
App.MainWindow	<i>Spezialisiert BendLibrary.App.AbstractMainWindow</i>
Config.BendManagerConfiguration	<i>Eine Ableitung des Singleton BendLibrary.Config.ProgramConfiguration</i>
Dialogs.ProgramConfigDialog	<i>Dialog für Änderung von Programmeinstellungen</i>
Forms.DataBrowser	<i>Werkzeug zum Auswählen und Bearbeiten von Daten</i>
Forms.DataDiagram	<i>Interaktives 2D-Diagramm (benutzt einen BendLibrary.Controls.DiagramCanvas zur Anzeige)</i>
Forms.IntroScreen	<i>Splash-Screen mit Copyright-Angabe und Grafik</i>

## 4.4 BendWorker.exe

### 4.4.1 Auflistung aller Pakete und Kurzbeschreibung

Eine exakte Auflistung aller Pakete findet man auch in der Datei „Solution\_BendManager.HTM“ im Verzeichnis „CodeCommentReport“.

BendWorker.App	<i>Hauptanwendung (Hauptfenster)</i>
BendWorker.Config	<i>Spezielle Programmeinstellungen für BendWorker</i>

#### 4.4.2 Auflistung aller Klassen und Kurzbeschreibung

Eine exakte Auflistung aller Klassen findet man auch in der Datei „Solution\_BendManager.HTM“ im Verzeichnis „CodeCommentReport“.

App.MainWindow	<i>Spezialisierung von BendLibrary.App.AbstractMainWindow</i>
Config.BendWorkerConfiguration	<i>Spezialisiert BendLibrary.Config.ProgramConfiguration für BendWorker-spezifische Einstellungen</i>
Enums.BendWorkerModus	<i>Aufzählung aller bekannter Modi (+1 Zustand „kein Modus“)</i>
Enums.BendWorkerDialogs	<i>Aufzählung aller im Programm möglichen Dialog-Zustände</i>
Forms.AnglePrognosisDialog	<i>Dialogfenster, das Ergebnisse einer Winkel-Vorhersage zeigt</i>
Forms.BendTestDialog	<i>Dialogfenster für Durchführung von Testbiegungen</i>
Forms.ChoseMaterialDialog	<i>Dialogfenster für Auswahl aus einer großen Liste</i>
Forms.ConfigureCNCDialog	<i>Dialogfenster zur Anzeige von Einstellungen für das CNC-Programm der Biegemaschine</i>
Forms.SelectModusDialog	<i>Dialog für Auswahl eines der Modi von BendWorker</i>
Forms.IntroScreen	<i>Splash-Screen mit Copyright-Angabe und Grafik</i>

#### 4.5 Berechnung der Rückverformung

Die Berechnung der Rückform-Winkel in BendWorker wurde nach einem Prinzip ähnlich der Intervallschachtelung für die Nullstellenberechnung durchgeführt. Dabei war vor allem hilfreich, dass die Funktionen der Ausgleichskurven ein festgelegtes Monotonieverhalten und wenige Extremwertstellen besaßen. Außerdem lag der erwartete Zielwert immer in der Nähe eines bekannten Ausgangswertes. Des Weiteren war die Genauigkeitsanforderung an die Näherung der Nullstellen äußerst gering. Für die Berechnung galt folgendes:

$$Y = \phi - d\phi$$

wobei „Y“ der gesuchte End-Winkel, „phi“ der zu biegende Winkel und „dphi“ die Rückverformung darstellen. Die Rückverformung ergibt sich dabei im allgemeinen Fall aus folgender Formel:

$$d\phi = C * \phi^N + D$$

Es müssen somit bei bekanntem Y, C, N, D gute Näherungen für phi gefunden werden. Die Suche beginnt in der Nähe von phi = Y und benötigt 2 Nachkommastellen Genauigkeit für phi.

## 5. Empfehlungen für Weiterentwicklung

### 5.1 Ein 3D-Diagramm für BendManager

Formeln für die Berechnung einer 3-dimensionalen Form, die ein Rohr mit Biegungen darstellt wurden bereits aufgestellt.

Zur praktischen Durchführung (Implementation) ist es leider nicht mehr gekommen, da sich auf Grund der Schwierigkeiten durch die fehlende Datenbank eine Verzögerung ergab.

Jemand mit guten Kenntnissen in DirectX sollte in der Lage sein, die Implementation zu übernehmen, da die Formeln zum Aufbau der geometrischen Form bereits vorliegen.

Jemand mit eher guten Kenntnissen in OpenGL (wie der Studierende, der dieses KSWs bearbeitet hat), würde eher auf .NET-OpenGL, wie das TAO-Framework zurückgreifen.

### 5.2 Ein kompletter Neuanfang

Obwohl einige Komponenten der Software wiederverwendet werden können, ist jedoch deutlich erkennbar, dass sie das erste Programm sind, das der Autor mit dieser Programmiersprache und Programmierumgebung entwickelt hat. Des weiteren ist die Schnittstelle zum FORTRAN-Programm eher eine Kuriosität, als effizientes und wartbares Design.

Auch die Datenbank, die der Autor entwickelt hat, sollte durch etwas ausgeklügeltes ersetzt werden, wie etwa die Anbindung einer relationalen oder einer objektorientierten Datenbank. Dazu ist eine Überarbeitung des vorliegenden Datenbankschemas unausweichlich.

Auch die Berechnung und Klassifikation sollte einer Prüfung unterzogen werden. Gemeinsam mit dem Autor des FORTRAN-Programms, Dr. Möller, sollte der bestehende Ansatz zur Klassifikation untersucht und, falls nötig, verbessert werden.

## 6. Anhang A – Abbildungsverzeichnis

### Abbildungsverzeichnis

Abbildung 1: Menüpunkt "Datei" in BendManager.....	S. 6
Abbildung 2: Menüpunkt "Bearbeiten" in BendManager.....	S. 7
Abbildung 3: Menüpunkt "Einstellungen" in BendManager.....	S. 7
Abbildung 4: Menüpunkt "Ansicht" in BendManager.....	S. 7
Abbildung 5: Menüpunkt "Hilfe" in BendManager.....	S. 8
Abbildung 6: Der DataBrowser in BendManager.....	S. 8
Abbildung 7: Das interaktive Diagramm von BendManager.....	S. 9
Abbildung 8: Weitere Optionen im Diagramm von BendManager.....	S. 10
Abbildung 9: Die Startseite von BendWorker.....	S. 12
Abbildung 10: Dialogfenster "Material-Auswahl" in BendWorker.....	S. 13
Abbildung 11: Dialogfenster "Test-Biegung" in BendWorker.....	S. 14
Abbildung 12: Dialogfenster "Modus-Auswahl" in BendWorker.....	S. 14
Abbildung 13: Dialogfenster "Prognose-Modus" in BendWorker.....	S. 15
Abbildung 14: Dialogfenster "CNC-Modus" in BendWorker.....	S. 15
Abbildung 15: Struktur der gesamten Software.....	S. 17

## 7. Anhang B – FORTRAN-Compiler für .NET

Die folgenden Compiler sollen in der Lage sein, FORTRAN95 mit .NET zu koppeln.  
Es gibt angeblich sogar schon ein FORTRAN2003.

Lahey/Fujitsu Fortran Compiler:

<http://www.lahey.com/>

[http://www.lahey.com/lf71/net\\_down.htm](http://www.lahey.com/lf71/net_down.htm)

-

Trial version: 14 tage testversion

Salford Fortran Compiler:

<http://www.salfordsoftware.co.uk/software/downloads/compilers.html>

-

Personal edition: blendet hinweis-banner ein

Intel Fortran Compiler:

<http://support.intel.com/support/performance/tools/fortran/windows/>

<http://www.intel.com/support/performance/tools/fortran/sb/cs-007782.htm>

<https://registrationcenter.intel.com/EvalCenter/EvalForm.aspx?ProductID=410>

-

Evaluation version: 30 tage testversion