

Übungsaufgaben-Finder

Hilfe zur Arbeit mit dem Programm

Stefan Ackermann

Mathematisches Institut
der Universität Leipzig

16. September 2016

Inhaltsverzeichnis

1	Motivation	2
2	Prinzipien	2
3	Voraussetzungen	3
4	Bedienung	4
5	Das Aufgabenverzeichnis	6
6	Das Hauptverzeichnis	7
6.1	Die Datei UeAFinder.ini	7
6.2	Die Datei mkpdf.bat	9
7	Editieren der Daten außerhalb des Programms UEAFINDER	9
7.1	Eingabe von Aufgaben-Daten	10

1 Motivation

In meiner Tätigkeit als Seminarleiter zu den unterschiedlichsten Vorlesungen, die von verschiedenen Professoren teilweise zu gleichen Themen gehalten wurden, habe ich eine Menge von Übungsaufgaben und dazugehörigen Lösungen gesammelt. Mit der Zeit wurde es zunehmend schwieriger, aus dieser Sammlung eine zum gerade im Seminar behandelten Stoff passende Aufgabe oder die Lösung einer erneut gestellten Übungsaufgabe zu finden. Ich habe mir deshalb ein hierarchisch gegliedertes Themensystem ausgedacht und ein kleines Programm dazu geschrieben, das die Suche nach einer passenden Aufgabe wesentlich erleichtert.

Da es gut sein kann, dass andere vor denselben Problemen stehen, habe ich mich entschlossen, das Programm zu veröffentlichen; auf Wunsch gebe ich auch gern den Delphi-Quelltext heraus.

In seiner einfachsten Art ist das einfach ein Betrachter von in einer bestimmten Verzeichnisstruktur hierarchisch abgelegten (PDF)-Dateien. Da diese in der Mathematik üblicherweise aus LaTeX-Quellen erzeugt werden, ist das Programm auch in der Lage, aus für die Aufgabe absolut notwendigen „LaTeX-Schnipseln“, diese PDF-Dateien zu erzeugen bzw. zu verändern. Diese Möglichkeiten muss man aber nicht nutzen.

Die Aufgabensammlung*) (mit Lösungen) selbst gebe ich allerdings höchstens an Kollegen des Mathematischen Institutes weiter. Interessenten können sich an mich wenden.

2 Prinzipien

Der Computer bietet durch die Datenspeicherung in Verzeichnissen bereits alle Möglichkeiten, Informationen strukturiert abzulegen. Jedoch ist das Suchen in diesen Strukturen nicht immer ganz einfach und vor allem zeitaufwendig, wenn man - um die gewünschten Informationen zu bekommen - erst in die Unterverzeichnisse wechseln und dort die Dateien, die diese Informationen enthalten könnten, mit einem geeigneten Programm öffnen muss. Nur selten gelingt es ja, nur aus dem Dateinamen auf die in der Datei abgelegten Inhalte zu schließen. Bei mathematischen Aufgaben, die zweckmäßigerweise als LaTeX-Dateien gespeichert sind, kommt erschwerend hinzu, dass diese selbst schlecht zu lesen sind und man zunächst daraus eine .dvi- oder .pdf-Datei herstellen oder - sofern diese bereits vorhanden ist - ein weiteres Programm öffnen muss.

Es fehlt also vor allem an einem Browser, der die gesammelten Informationen übersichtlich und „human readable“ darstellt. Dies soll der Übungsaufgaben-Finder leisten, wobei noch folgende Prinzipien eingehalten werden sollten:

- Die gewünschte Verzeichnisstruktur soll sich der Nutzer fast ohne Restriktionen selbst vorgeben und anlegen können.
- Pro Aufgabe soll ein Verzeichnis angelegt werden, das Aufgabentext und Lösung als zwei verschiedene LaTeX-Dateien getrennt und platzsparend enthält, d. h. die gemeinsamen Header und Rahmen sollen nur einmal getrennt von den Inhalten gespeichert werden und die Inhalte als \input-Dateien aufnehmen. Alle anderen notwendigen Dinge wie z.B. Bilder u.ä. sollen ebenfalls in diesem Verzeichnis enthalten sein.

*)Im Moment ist diese noch im Aufbau. Nach und nach werde ich meine gesammelten Aufgaben in diese Nomenklatur einarbeiten.

- Da diese \input-Dateien nun nicht mehr direkt in eine lesbare Form gebracht werden können, muss die „human readable“ Version entweder zur Laufzeit erzeugt oder mit im selben Verzeichnis vorgehalten werden. Ich habe mich aus Performance-Gründen für letzteres entschieden; möglicherweise muss man bei steigenden Datenmengen und genügender Rechenleistung später umdenken.
- Als lesbare Form habe ich mich für PDF-Dateien entschieden, da es für diese ein Acrobat-Reader-Plugin gibt, mit dessen Hilfe man sich die Dateien innerhalb von Delphi-Programmen ansehen kann. Damit das funktioniert, muss auf dem Rechner der frei erhältliche Acrobat-Reader oder der ebenfalls frei erhältliche, aber wesentlich schnellere Foxit-Reader installiert sein.
- Das Programm soll sich relativ leicht an die Gegebenheiten des jeweiligen Rechners anpassen lassen. Im wesentlichen handelt es sich dabei um die Angabe von Pfaden in einer ini-Datei.
Einem ungeübten Nutzer hilft das Programm beim ersten Programmstart, diese Datei anzulegen.
- Ursprünglich sollte das Programm auch auf verschiedenen Plattformen (Windows, Linux) laufen. Da die Portierung von Delphi nach Linux (Kylix) aber nicht mehr unterstützt wird und es insbesondere kein Plugin zur Anzeige von PDF-Dateien aus dem Programm heraus gibt, habe ich diesen Anspruch zunächst erst einmal wieder aufgegeben. Inzwischen gibt es in Lazarus eine Delphi-ähnliche Programmierungsumgebung, mit der man auch unter bzw. für Linux programmieren kann. Eine Komponente, die PDF-Dateien anzeigt, vielleicht sogar ohne dass Acrobat-Reader installiert sein muss, gibt es dafür aber leider noch nicht.

Nach einer längeren Arbeitsphase des Programms wurde der Wunsch, eine Aufgabe verschiedenen Unterpunkten zuzuordnen, immer stärker. Natürlich kann man einfach das ganze Verzeichnis mit allen notwendigen Dateien noch an eine weitere Stelle der Hierarchie kopieren. Aus Platzgründen habe ich aber eine zweite Möglichkeit entwickelt, bei der das neue Unterverzeichnis nur zwei einzeilige Textdateien enthält, von denen eine den Pfad auf das Originalverzeichnis enthält. Solche Aufgaben sind also nichts weiter als *Verweise* auf andere Aufgaben.

3 Voraussetzungen

Damit das Programm auf einem (Windows-)Computer wenigsten seine Grundfunktion, Übungsaufgaben zu suchen, erfüllen kann, muss auf diesem Computer ein Acrobat-Reader (Version 6 oder höher) oder ein Foxit-Reader installiert sein.

Um das Programm in vollem Umfang nutzen zu können, muss eine vollständige T_EX-Installation einschließlich Ghostscript usw. vorliegen. Ohne diese kann man das Programm aber immer noch als Anzeiger für PDF-Dateien nutzen.

Außerdem sollte dann der Lieblings-L^AT_EX-Editor so installiert sein, dass er sich automatisch öffnet, wenn man im Explorer auf einer `.tex`-Datei doppelklickt

Ich gehe davon aus, dass diese Voraussetzungen auf dem PC eines Mathematikers gegeben sind.

4 Bedienung

Beim Start des Programms erscheint das zweigeteilte Hauptfenster des Programms (Abbildung 1).

Im linken Teil ist die baumartige Struktur zu erkennen, in der die Inhalte abgelegt sind. Diese Struktur genügt bereits, um der Hauptaufgabe, gezielt nach Übungsaufgaben zu einem Thema zu finden, gerecht zu werden. Die einzelnen Themen und Unterthemen sind mit den Verzeichnis-Symbol versehen, die Aufgaben bzw. Lösungen tragen das PDF-Symbol.

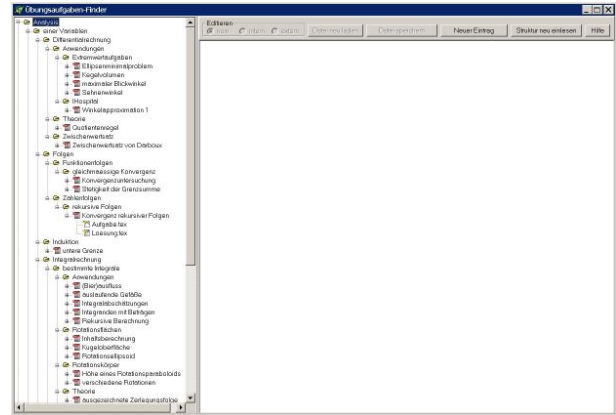


Abbildung 1: Hauptfenster des ÜA-Finders

Klickt man auf einen Eintrag mit dem PDF-Symbol, dann wird im rechten Fenster die Aufgabe samt Lösung - falls vorhanden - angezeigt (Abbildung 2). Diese Funktionalität genügt einem Nutzer bald nicht mehr, deshalb wurde im rechten Fenster eine Buttonleiste angefügt, in der - je nach Situation - verschiedene Schalter zur Erleichterung und Verbesserung der Arbeit mit diesem Programm zur Verfügung stehen (Abbildung 3).

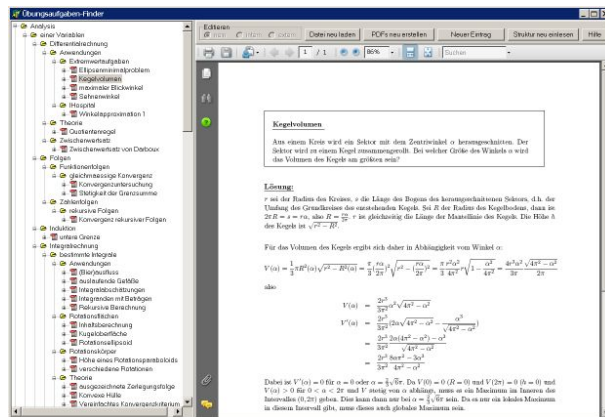


Abbildung 2: Anzeige einer Aufgabe

Stets vorhanden ist der Button „Hilfe“, der diesen Text anzeigt, sowie der Button „Struktur neu einlesen“. Dieser Button gewährleistet einem Nutzer, der zur Laufzeit des Programms, aber außerhalb des Programms, d. h. mit den vom Betriebssystem bereitgestellten Mitteln, Inhalte ändert oder hinzufügt, Verzeichnisse anlegt usw., dass diese Änderungen vom Programm übernommen werden, ohne dies beenden und neu starten zu müssen.

Immer dann, wenn im linken Baumdiagramm ein Verzeichnis angeählt ist, steht der Button „Neuer Eintrag“ zur Verfügung, der es nun auch innerhalb des Programmes erlaubt, Inhalte hinzuzufügen, neue Unterverzeichnisse und Aufgabenverzeichnisse anzulegen. Klickt

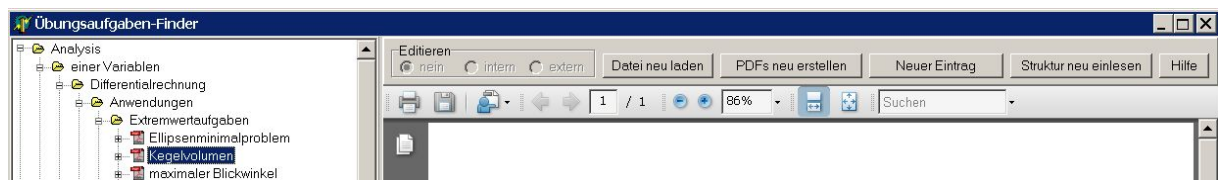


Abbildung 3: Buttonleiste

man diesen Button an, so erscheint der in Abbildung 4 gezeigte, selbsterklärende Dialog.

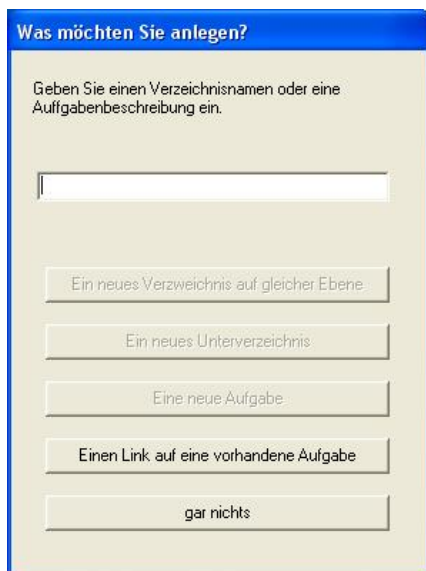


Abbildung 4: Auswahldialog

Solange im Editfeld keine Beschreibung eines neuen Verzeichnisses oder einer neuen Aufgabe eingetragen ist, hat man nur die Möglichkeit einen Verweis auf eine bereits vorhandene Aufgabe anzulegen. Deren Beschreibung wird dann übernommen.

Hat man in das Editfeld eine Beschreibung eingetragen, denn stehen alle Möglichkeiten zur Verfügung.

Während man beim Anlegen von Verzeichnissen wählen kann, ob man ein echtes Unterverzeichnis oder ein Verzeichnis auf gleicher Ebene anlegen möchte, werden Aufgaben und Verweise immer in das Verzeichnis abgelegt, das im Hauptfenster ausgewählt wurde.

Beim Anlegen eines Verweises auf eine schon vorhandene Aufgabe hat man die Möglichkeit, in einem weiteren Fenster aus einer Baumstruktur wie im Hauptfenster, die Aufgabe, auf die verwiesen werden soll, auszuwählen. Ist das Editfenster gefüllt, so wird bei Verweisen die dort gemachte Angabe (und nicht die Beschreibung der ursprünglichen Aufgabe) verwendet.

Jeder Aufgabeneintrag im linken Fenster des Hauptformulars enthält zwei zunächst nicht expandierte Untereinträge, die auf die L^AT_EX-Fragmente der Aufgabenstellung und Lösung zeigen.

Ist im linken Fenster einer dieser Einträge oder der übergeordnete für die PDF-Datei angewählt, dann ist auch der Button „Datei neu laden“ aktiv. Sollte ein Nutzer die Inhalte außerhalb des Programmes UEAFINDER ändern, also z. B. die L^AT_EX-Dateien editieren oder die PDFs neu erzeugen, so können diese Änderungen ins Programm übernommen werden, ohne dies beenden und neu starten zu müssen.

Die L^AT_EX-Quelltexte lassen sich auch innerhalb des Programmes UEAFINDER editieren. Dazu muss man, wenn im linken Feld der entsprechende Eintrag ausgewählt ist, in der Radiogruppe „Editieren“ auf den Eintrag „intern“ schalten. Nun sind Änderungen im angezeigten Memofeld möglich. Sobald man etwas ändert, wird ein Button „Datei speichern“ aktiv. Verlässt man diese Ebene, ohne zu speichern, wird man explizit darauf hingewiesen.

Hat man die Datei gespeichert, möchte man die Änderungen natürlich auch in der PDF-Datei sehen. Dazu muss man nur im linken Fenster wieder auf die PDF-Datei stellen. In der Buttonleiste hat man dann den Schalter „PDFs neu erstellen“ zur Verfügung, der allerdings nur funktioniert, wenn auf dem Computer ein T_EX-System vollständig installiert ist und die

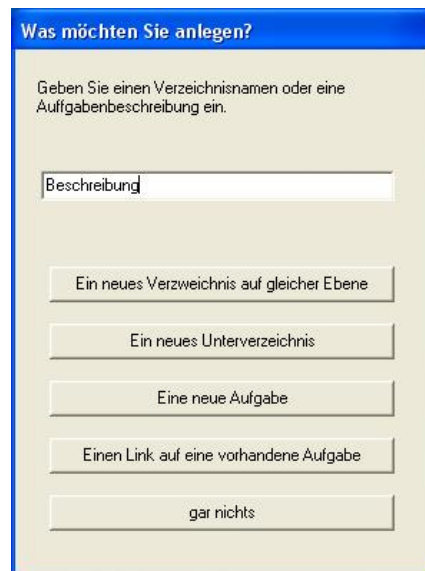


Abbildung 5: Auswahldialog nach Eingabe einer Beschreibung

Pfadangaben in der Datei `UeAFinder.ini` korrekt sind. Hat man einen Lieblings-Editor bzw. eine \LaTeX -Entwicklungsumgebung korrekt installiert, so dass diese aufgerufen wird, wenn man im Explorer auf eine `.tex`-Datei klickt, dann öffnet sich diese auch, wenn man in der Radiogruppe auf „extern“ schaltet, und man kann die aktuelle `.tex`-Datei dort editieren.

Hat man im linken Fenster einen \LaTeX -Quelltext ausgewählt, dann gibt es in der Buttonleiste einen Schalter, mit dem man den Inhalt der `.tex`-Datei in die Zwischenablage kopieren kann. Dies ist bequem, wenn man aus seinen gesammelten Werken neue Übungsblätter oder Klausuren zusammenstellen möchte.

5 Das Aufgabenverzeichnis

Während ein Nutzer sich die Hierarchie der Aufgabenablage durch das Anlegen und Benennen von Unterverzeichnissen im Rahmen der Möglichkeiten seines Betriebssystems völlig frei gestalten kann, ist für die Ablage der Aufgaben eine bestimmte Struktur vorgegeben.

Da Aufgabe und Lösung getrennt gehalten werden sollten und außerdem eventuell weitere Dateien wie Bilder und Grafiken und/oder Dateien, aus denen diese Grafiken entstanden sind (Maple-Dateien, `*.fig`-Dateien u. ä.) aufgehoben werden sollten, ist pro Aufgabe ein Verzeichnis notwendig.

Da es außerdem schwierig sein könnte, ähnliche Aufgaben unterschiedlich zu benennen, habe ich die Verzeichnisnamen nummeriert (`_001`, `_002`, usw.). Die Bezeichnung des im linken Programmfenster zu der Aufgabe gehörigen Eintrages muss also nun irgendwo mit im Verzeichnis stehen. Deswegen gibt es in jedem Aufgabenverzeichnis wenigsten fünf^{†)} Dateien:

Dateiname	Bedeutung
<code>Aufgabe.tex</code>	\LaTeX -Fragment des Aufgabentextes
<code>Loesung.tex</code>	\LaTeX -Fragment des Lösungstextes (kann leer sein)
<code>Aufgabe.pdf</code>	Aufgabe als PDF-Datei
<code>Loesung.pdf</code>	Aufgabe und Lösung als PDF-Datei (= <code>Aufgabe.pdf</code> , wenn <code>Loesung.tex</code> leer ist)
<code>Aufgabe.diz</code>	einzeilige Datei mit dem Text des Eintrages im linken Programmfenster

Daneben kann es - wie gesagt - noch weitere Dateien geben, die zur Erstellung der PDF-Dateien notwendig oder nützlich sind.

Eine Ausnahme stellen Aufgabenverzeichnisse dar, die nur einen Verweis auf eine andere Aufgabe enthalten. In diesen befinden sich nur zwei Dateien, nämlich

Dateiname	Bedeutung
<code>Aufgabe.dir</code>	Pfad zur Ursprungsaufgabe, relativ zum Hauptverzeichnis (siehe 6.)
<code>Aufgabe.diz</code>	einzeilige Datei mit dem Text des Eintrages im linken Programmfenster

Nutzt man das Programm ständig nur auf einem Rechner, dann sollten diese Einschränkungen leicht zu verschmerzen sein, insbesondere dann, wenn man zum Anlegen neuer Aufgaben die im

^{†)}In der reinen Browser-Funktion nur zwei oder drei, da die `*.tex`-Dateien und (im Studenten-Modus) auch die Lösungsdateien entfallen.

Programm dafür vorgesehenen Schalter nutzt. Man muss sich nämlich dann um die Nummerierung der Unterverzeichnisse und die Generierung der \LaTeX -Dateien `Aufgabe.tex` und `Loesung.tex` nicht selbst kümmern, alles passiert automatisch. Wie man es selbst von Hand und außerhalb des Programmes am Besten macht, steht im übernächsten Abschnitt.

Etwas komplizierter ist die Sache schon, wenn man zwei Datenmengen zusammenführen will, die unabhängig voneinander auf zwei verschiedenen PCs entstanden sind. Dort muss man sehr genau darauf achten, dass nicht unterschiedliche Inhalte in gleichnummerierten Aufgabenverzeichnissen stehen, von denen man sich einen beim Kopieren überschreibt.

6 Das Hauptverzeichnis

Das Hauptverzeichnis ist das Verzeichnis, das die oberste Ebene aller Unterverzeichnisse mit Aufgabendaten enthält. In ihm sollten sich außerdem mindestens folgende Dateien^{‡)} befinden:

Dateiname	Bedeutung
<code>UeAFinder.exe</code>	Das eigentliche Programm
<code>UeAFinder.ini</code>	Textdatei, die Pfadangaben des Systems enthält
<code>Hilfe.pdf</code>	diese PDF-Datei
<code>Aufgabe.diz</code>	Vorlage für die einzeilige Aufgabenbeschreibung
<code>Aufgabe.tex</code>	\LaTeX -Vorlage für das Fragment des Aufgabentextes
<code>Loesung.tex</code>	\LaTeX -Vorlage für das Fragment des Lösungstextes
<code>UebAufg.tex</code>	\LaTeX -Vorlage zum Erzeugen von <code>Aufgabe.pdf</code>
<code>UebAufgLoes.tex</code>	\LaTeX -Vorlage zum Erzeugen von <code>Loesung.pdf</code>

Während der Nutzung des Programms entstehen hier weitere Dateien:

Dateiname	Bedeutung
<code>mkpdf.bat</code>	Kommandoskript, der aus einem Aufgabenverzeichnis heraus aufgerufen werden kann und dort aus <code>Aufgabe.tex</code> und <code>Loesung.tex</code> die Dateien <code>Aufgabe.pdf</code> bzw. <code>Loesung.pdf</code> erzeugt.
<code>UebAufg.*</code> <code>UebAufgLoes.*</code>	Zwischenprodukte beim Erzeugen der PDF-Dateien die <code>*.log</code> Dateien sind nützlich, wenn Fehler auftreten.

Zwei dieser Dateien sollen im folgenden näher erläutert werden.

6.1 Die Datei `UeAFinder.ini`

Diese Datei muss sich im selben Unterverzeichnis wie die Programmdatei `UeAFinder.exe` befinden; sie enthält abschnittsweise alle notwendigen Pfadangaben zu Programmen und Daten, die `UEAFINDER` auf verschiedenen Rechnern benötigt. Da `UEAFINDER` ein portables Programm ist, das ohne Installation auf jedem Windows-Rechner - z. B. vom USB-Stick läuft, muss für den aktuellen Rechner ein entsprechender Abschnitt vorhanden sein. Ist er das nicht, dann wird er vom Programm angelegt. Die Datei `UeAFinder.ini` könnte beispielsweise so aussehen:

^{‡)}Die `*.tex`-Dateien sind in der reinen Viewer-Version nicht nötig.

```
[MIPC66]
Root=c:\users\ackermann\ownCloud\UeAFinder\
MikTeX=c:\program files\miktex 2.9\miktex\bin\x64\
GhostScript=C:\Program Files (x86)\gs\gs9.14\bin\
Include=Z:\Win\tex\styles\
nurPDF=0
```

```
[STEFANS-NETBOOK]
Root=C:\Users\ackermann\ownCloud\UeAFinder
MikTeX=c:\program files\miktex 2.9\miktex\bin\
GhostScript=C:\Program Files\gs\gs9.14\bin\
Include=c:\users\ackermann\Documents\tex\styles\
nurPDF=0
```

Der Eintrag `Root` ist nur erforderlich, wenn die Daten, d. h. die Aufgabenverzeichnisse, sich nicht ebenfalls im Verzeichnis der Programmdatei befinden.

Die nächsten beiden Einträge zeigen auf die Pfade der Programme, die UEA-FINDER benutzt, wenn man den Button „PDFs neu erstellen“ anklickt. Fehlen diese beiden Angaben, dann funktioniert diese Option im Programm eventuell nicht. Das ist natürlich insbesondere dann der Fall, wenn die beiden Programme auf Ihrem Rechner gar nicht installiert sind.

Sind *MikTeX* und *GhostScript* korrekt installiert, kann das Programm UEA-FINDER auch dann funktionieren, wenn diese Pfadangaben nicht in der Datei `UeAFinder.ini` stehen, da die Programme dann schon über die Registry oder die Umgebungsvariable `%PATH%` gefunden werden.

Die letzte Eintragung im Abschnitt `[Paths]` ist für Nutzer vorgesehen, die ständig eigene Macrodateien in Ihre \LaTeX -Quelltexte einbinden. Hier muss der Pfad zu diesen Dateien stehen.

Der Eintrag `nurPDF=0` gewährleistet die volle Funktion des Programms. Ist `nurPDF=1`, dann arbeitet das Programm nur als Browser für PDF-Dateien.

Immer dann, wenn die Datei `UeAFinder.ini` fehlt oder keine Pfadangaben für *MikTeX* und *GhostScript* enthält und `nurPDF=0` ist, versucht das Programm UeAFinder die Programme auf Ihrem Rechner zu finden und einen für Ihr System funktionierenden Abschnitt in die `ini`-Datei zu schreiben. Dies sollten Sie gestatten, damit bei folgenden Starts diese Suche entfallen kann. Eventuell müssen Sie die Einträge danach von Hand editieren. Dies betrifft insbesondere die Variable `Include`.

Sollten Sie mit einer anderen \LaTeX -Distribution und eventuell sogar mit einem anderen PostScript-Interpreter arbeiten, dann müssen Sie die richtigen Pfade zu den ausführbaren Dateien von Hand eintragen. Auch der Pfad zu eventuell benutzten Include-Verzeichnissen kann nicht automatisch ermittelt werden, sondern muss von Ihnen in der `ini`-Datei entsprechend eingetragen werden, da sonst bei der Erzeugung der PDF-Dateien Fehler auftreten.

Sollten Sie beide Programme gar nicht installieren wollen und UEA-FINDER nur als Aufgabenbrowser benutzen wollen, dann schreiben Sie den Abschnitt

```
[Rechnername]
Root=z:\tex\UeA\
MikTeX=-
GhostScript=-
```

```
Include=  
[Options]  
nurPDF=1
```

in die ini-Datei. Sie werden dann nicht mehr durch ständiges Nachfragen genervt, können aber auch den Button „PDFs neu erstellen“ nicht benutzen.

6.2 Die Datei mkpdf.bat

Aus den Angaben in der Datei UeAFinder.ini erzeugt UEAFINDER bei jedem Programmstart die für Ihr System angepasste Datei mkpdf.bat, mit deren Hilfe Sie auch außerhalb des Programms aus den L^AT_EX-Fragmenten in den Aufgabenverzeichnissen die vollständigen PDF-Dateien erzeugen können. Alles was Sie dazu tun müssen ist, diese Datei (mit vollem Pfadnamen) aus dem entsprechenden Aufgaben-Unterverzeichnis heraus aufzurufen.

Die zur oben angegebenen ini-Datei gehörige Kommando-Datei sieht z. B. folgendermaßen aus:

```
@echo off  
  
rem mkpdf.bat - erzeugt PDF-Dateien Aufgabe.pdf und Loesung.pdf  
rem aus den Fragmenten Aufgabe.tex und Loesung.tex und den Dateien  
rem z:\tex\UeA\UebAufg.tex und z:\tex\UeA\UebAufgLoes.tex  
  
rem Verzeichnis einstellen  
cd %1  
rem Aufgabe.pdf erzeugen  
echo Aufgabe.pdf wird erzeugt  
"C:\texmf\MiKTeX\bin\latex.exe" -interaction=nonstopmode  
    -include-directory="z:\tex\styles"  
    -output-directory="z:\tex\UeA" "z:\tex\UeA\UebAufg.tex"  
"C:\texmf\MiKTeX\bin\dvips.exe" -P pdf -o "z:\tex\UeA\UebAufg.ps"  
    "z:\tex\UeA\UebAufg.dvi"  
"C:\programme\gs\gs8.14\bin\gswin32c.exe" -sPAPERSIZE=a4 -dSAFER  
    -dBATCH -dNOPAUSE -sDEVICE=pdfwrite -sOutputFile="Aufgabe.pdf"  
    -c save pop -f "z:\tex\UeA\UebAufg.ps"  
Loesung.pdf erzeugen  
echo Loesung.pdf wird erzeugt  
"C:\texmf\MiKTeX\bin\latex.exe" -interaction=nonstopmode  
    -include-directory="z:\tex\styles"  
    -output-directory="z:\tex\UeA" "z:\tex\UeA\UebAufgLoes.tex"  
"C:\texmf\MiKTeX\bin\dvips.exe" -P pdf -o "z:\tex\UeA\UebAufgLoes.ps"  
    "z:\tex\UeA\UebAufgLoes.dvi"  
"C:\programme\gs\gs8.14\bin\gswin32c.exe" -sPAPERSIZE=a4 -dSAFER  
    -dBATCH -dNOPAUSE -sDEVICE=pdfwrite -sOutputFile="Loesung.pdf"  
    -c save pop -f "z:\tex\UeA\UebAufgLoes.ps"
```

Aus Platzgründen mussten die Zeilen umgebrochen werden. Alles, was nicht ganz links beginnt, gehört - nach einem Leerzeichen - mit zur vorhergehenden Zeile.

7 Editieren der Daten außerhalb des Programms UeAFINDER

Der Einfachheit halber gehen wir davon aus, dass Sie sich in einem Verzeichnis Ihrer Wahl bereits eine thematische Hierarchie von Unterverzeichnissen nach Ihren Vorstellungen geschaffen haben, wie sie z. B. im rechten Bild deutlich wird.

Es sei hier nochmal darauf hingewiesen, dass sich diese Unterverzeichnisse auch ganz woanders befinden können, wenn dies nur in der Datei `UeAFinder.ini` in der Variablen `Roots=` angegeben wird.

Um dies nicht immer umschreiben zu müssen, nennen wir dieses Verzeichnis - passend zu den obigen Beispielen - z.: `\tex\UeA\`. In allen folgenden Angaben ersetzen Sie einfach diese Zeichenkette durch Ihren Verzeichnisnamen.

Weiter nehmen wir an, dass sich in diesem Verzeichnis bereits die acht im Abschnitt 6 genannten Dateien befinden.



Die beiden Dateien `UebAufg.tex` und `UebAufgLoes.tex` müssen Sie eventuell nun noch etwas editieren, d. h. von Ihnen gewünschte `\package-` und `\input-` bzw. `\include-`Anweisungen hinzufügen und Dinge, die Sie nicht benötigen, löschen.

7.1 Eingabe von Aufgaben-Daten

1. Wechseln Sie nun in eines der Unterverzeichnisse, in dem Sie eine Aufgabe unterbringen wollen.
2. Erstellen Sie ein Unterverzeichnis namens „_??“, wobei die drei Fragezeichen für eine dreistellige Zahl mit führenden Nullen steht. Gibt es noch kein solches Unterverzeichnis, dann nehmen Sie einfach „_001“, anderenfalls zählen Sie einfach weiter und nehmen die erste noch nicht vergebene Nummer.
3. Wechseln Sie in dieses Unterverzeichnis und kopieren Sie die fünf Dateien^{s)} `Aufgabe.diz`, `Aufgabe.tex`, `Loesung.tex`, `Aufgabe.pdf` und `Loesung.pdf` aus dem Hauptverzeichnis

^{s)}Die *.tex-Dateien können im Browser-Modus wieder entfallen.

hierher.

4. Editieren Sie die Datei **Aufgabe.diz** mit einem einfachen Texteditor (notepad. o. ä., also nicht mit MS-Word) und geben Sie dort eine möglichst kurze, aber auf jeden Fall einzeilige Beschreibung der Aufgabe an.
5. Tragen Sie in der Datei **Aufgabe.tex** mit Ihrem Lieblings-L^AT_EX-Editor an den vorgesehen Stellen die Aufgabenüberschrift, also z. B. den Text aus der diz-Datei, und den Aufgabentext ein. Sie können auch noch ein Bild hinzufügen, dann sollten Sie allerdings die Spaltenbreiten der bereits angelegten Tabelle anpassen.
6. Tragen Sie in die Datei **Loesung.tex** die Lösung der Aufgabe ein. Alles, was dafür an externem Material benötigt wird, also z. B. Bilder, Grafiken oder andere Dinge, die durch `\include`-Anweisungen eingebunden werden sollen, muss nun noch in dieses Verzeichnis kopiert werden.
Allerdings sollten Dinge, die Sie häufig an verschiedenen Stellen benutzen wollen, besser in einem extra Verzeichnis gesammelt werden, dessen Name in der Pfadanweisung **Include=** der Datei **UeAFinder.ini** steht.
7. Sollten Sie UeAFINDER bereits wenigsten einmal gestartet haben, dann finden Sie im Hauptverzeichnis die Datei **z:\tex\UeA\mkpdf.bat**, die Sie nun einmal aus ihrem Aufgabenverzeichnis heraus aufrufen müssen und damit die Dateien **Aufgabe.pdf** und **Loesung.pdf** (neu) erzeugen.
8. Starten Sie nun UeAFINDER und sie sollten Ihre gerade eingegebene Aufgabe an der entsprechenden Stelle ansehen können.
Ist UeAFINDER bereits gestartet, genügt es, den Button „Struktur neu einlesen“ einmal zu betätigen.