

fomsdt: Paket zur Erstellung wissenschaftlicher Arbeiten

Axel Kessler

<axel.kessler@gmx.net>

17. Februar 2008

Dieses Dokument wurde aus der Datei `fomsdt.dtx`
in der Version v1.0 vom 2007/10/29 erzeugt.

Zusammenfassung

Das Paket `fomsdt` ist das Ergebnis einer Diplomarbeit an der FOM – Fachhochschule für Oekonomie & Management in Essen und soll den Studierenden das Verfassen ihrer wissenschaftlichen Arbeiten unter Verwendung des Textsatzsystems \LaTeX erleichtern.

`fomsdt` basiert auf den an der FOM im Fachbereich Wirtschaftsinformatik anerkannten Regeln zur Erstellung wissenschaftlicher Arbeiten.

Inhaltsverzeichnis

1	Distribution	2
2	Dokumentation	3
3	Installation	3
3.1	Verzeichnisstruktur	4
3.2	Kopieren und Entpacken	4
3.3	Aktualisierung der Umgebung	5
3.4	Kurzanleitung	6
4	Paketübersicht	6
5	Dokumentenklasse	8
5.1	Optionen	8
5.2	Makros	10
6	Dokumentenbeginn	16
6.1	Optionen	17
6.2	Makros	18
7	Dokumentenende	20
7.1	Optionen	21
7.2	Makros	22
8	Absatzdefinition	30
8.1	Makros und Umgebungen	30
9	Gleitobjektdefinition	34
9.1	Optionen	34
9.2	Makros und Umgebungen	35

10 Vermischtes	44
10.1 Makros	44
Bekannte Probleme	47
Abhängigkeiten	49
Weiterführende Literatur	51
Index	52

1 Distribution

Zum Lieferumfang des Pakets `fomsdt` gehören die Dateien `fomsdt.ins`, `fomsdt.dtx` und `readme.txt`. Diese Installationsdateien beinhalten neben den eigentlichen Paketdateien, bestehend aus eine Klassendatei und mehreren Stildateien, auch die zum Paket gehörende Anleitung.

- `fomsdt.ins` Die Datei `fomsdt.ins` beinhaltet das zur Extraktion des Pakets `fomsdt` notwendige \LaTeX -Skript, dessen Hauptaufgabe darin besteht, die Aufteilung des Inhalts der Datei `fomsdt.dtx` vorzunehmen.
- `fomsdt.dtx` Der vollständige Paketcode ist in der Datei `fomsdt.dtx` enthalten. Ebenfalls in dieser Datei enthalten ist der Code zur Erstellung die hier vorliegende Paketdokumentation.
- `readme.txt` Die Datei `readme.txt` enthält neben allgemeinen Informationen auch ein Kurzanleitung zur Erstellung der hier vorliegenden Dokumentation.

Die hieran anschließenden Kapitel gehen sowohl auf die Erstellung der Dokumentation als auch auf die Extraktion und die Installation des Pakets `fomsdt` im \LaTeX -System ein.

2 Dokumentation

Zur Erstellung der Dokumentation wird zunächst die Datei `fomsdt.dtx` einmal kompiliert. Traten hierbei keine Fehler auf, wird die Datei `fomsdt.dtx` ein zweites mal kompiliert. Dies ist notwendig, da erst jetzt alle Indexeinträge auf die „richtigen“ Seiten verweisen.

Daran anschließend wird der Index generiert. Traten auch hierbei keine Fehler auf, ist die Datei `fomsdt.dtx` noch ein drittes und abschließendes mal zu kompilieren.

Der gesamte Prozess zur Erstellung der Paketdokumentation wird durch die Eingabe verschiedener \LaTeX -Kommandos am Prompt einer Shell beziehungsweise Konsole durchgeführt. Die nachfolgende Sequenz zeigt die einzelnen Befehle, mit deren Hilfe die Dokumentation generiert wird.

```
$ pdflatex fomsdt.dtx
$ pdflatex fomsdt.dtx
$ makeindex -s gind.ist -o fomsdt.ind fomsdt.idx
$ pdflatex fomsdt.dtx
```

Sollten nach dem dritten Kompiliervorgang eventuell noch Fehler oder Warnungen auftreten, dann empfiehlt es sich, das Kommando `pdflatex fomsdt.dtx` erneut anzuwenden, um die Dokumentation ein viertes mal zu kompilieren.

Nach Beendigung des Kompiliervorganges sollte im aktuellen Verzeichnis nun auch die Datei `fomsdt.pdf`, neben weiteren Ausgabedateien, vorhanden sein.

3 Installation

Die Installation des Pakets `fomsdt` wird in mehreren Einzelschritten durchgeführt. Dabei muss an einigen Stellen auf die Besonderheiten der unterschiedlichen Betriebssysteme Rücksicht genommen werden. Entsprechende Hinweise erfolgen an den jeweiligen Stellen.

3.1 Verzeichnisstruktur

Der erste Schritt der Installation des Pakets `fomsdt` ist die Erzeugung der notwendigen Verzeichnisstruktur. Hierfür werden innerhalb der \LaTeX -Umgebung insgesamt zwei Verzeichnisse, jedes mit dem Namen `fomsdt`, angelegt. Dabei ist zu beachten, dass diese Verzeichnisse als Unterverzeichnisse der Verzeichnisse $\$TEXROOT/tex/latex$ und $\$TEXROOT/makeindex$ erstellt werden müssen!

Der Bezeichner $\$TEXROOT$ dient an dieser Stelle nur als Platzhalter, da das entsprechende \LaTeX -Basisverzeichnis stark vom eingesetzten Betriebssystem abhängt.

Windows Unter Windows wird zumeist `MiKTeX` als \LaTeX -Umgebung eingesetzt. Daher ist für $\$TEXROOT$ das Verzeichnis der installierten `MiKTeX`-Distribution zu verwenden. Dies könnte zum Beispiel `C:\Programme\MiKTeX\` sein. Somit ergeben sich für dieses Beispiel und das Betriebssystem Windows die beiden Verzeichnisse `C:\Programme\MiKTeX\tex\latex\fomsdt` und `C:\Programme\MiKTeX\makeindex\fomsdt`.

Linux Unter Linux wird die \LaTeX -Distribution zumeist im Verzeichnis `/usr/share/texmf/` installiert, sodass als $\$TEXROOT$ genau dieser Pfad als Basis zu verwenden ist. Somit ergeben sich für dieses Beispiel und das Betriebssystem Linux die beiden vollständigen Pfade `/usr/share/texmf/tex/latex/fomsdt` und `/usr/share/texmf/makeindex/fomsdt`.

3.2 Kopieren und Entpacken

Nachdem zuvor die Zielverzeichnisse, wie in Abschnitt 3.1 beschrieben, erstellt wurden, werden die zum Paket gehörenden Dateien `fomsdt.ins`, `fomsdt.dtx` und `readme.txt` von ihrer Quelle, beispielsweise das Verzeichnis, in welchem die Dateien nach ihrem Download gespeichert wurden, in das Verzeichnis $\$TEXROOT/tex/latex/fomsdt$ kopiert.

Die unter Kapitel 2 generierte Datei `fomsdt.pdf` wird ebenfalls in dieses Verzeichnis kopiert. Dies weicht zwar von der \LaTeX -Empfehlung für den Dokumentenspeicherort ab, dient aber dem Zusammenhalt des gesamten Pakets.

Jetzt wird im Verzeichnis `$TEXROOT/tex/latex/fomsdt` eine Shell beziehungsweise Konsole geöffnet und darin der Befehl zum Entpacken des Pakets wie folgt eingegeben und ausgeführt:

```
$ pdflatex fomsdt.ins
```

Konnten bis hierhin alle Schritte fehlerfrei ausgeführt werden, dann sollten im Verzeichnis `$TEXROOT/tex/latex/fomsdt` mindestens die hier gezeigten Dateien vorhanden sein.

<code>fomsdt.dtx</code>	<code>fomsdt.cls</code>	<code>fomfod.sty</code>
<code>fomsdt.ins</code>	<code>fomdss.sty</code>	<code>fomisc.sty</code>
<code>readme.txt</code>	<code>fomdes.sty</code>	<code>fomidx.ist</code>
<code>fomsdt.pdf</code>	<code>fompar.sty</code>	

Als nächstes ist noch die Datei `fomidx.ist` in das Verzeichnis `$TEXROOT/makeindex/fomsdt/` zu verschieben!

3.3 Aktualisierung der Umgebung

Als dritter und letzter Schritt der Installation des Pakets `fomsdt` ist nun noch die Aktualisierung der \LaTeX -Umgebung durchzuführen. Hierfür wird am Prompt der noch geöffneten Eingabeaufforderung das Aktualisierungskommando wie folgt eingegeben:

```
$ texhash
```

Speziell für die Benutzer einer $\text{MiK}\TeX$ -Distribution gilt: Sollte das Kommando `texhash` *nicht* zur gewünschten Aktualisierung des \LaTeX -Systems führen, dann sollte das Aktualisierungskommando der $\text{MiK}\TeX$ -Distribution verwendet werden. In diesem Fall ist am Prompt stattdessen das Kommando `initexmf -u` einzugeben.

Mit der fehlerfreien Registrierung ist nun auch die Installation des Pakets erfolgreich beendet. Das Paket `fomsdt` kann von nun an zusammen mit \LaTeX verwendet werden.

3.4 Kurzanleitung

Für die Eiligen, die Kurzfassung der Anleitung zur Installation des Pakets `fomsdt` im \LaTeX -System.

- Jeweils ein Unterverzeichnis mit dem Namen `fomsdt`
 - im Verzeichnis `$TEXROOT/tex/latex` und
 - im Verzeichnis `$TEXROOT/makeindex`erstellen.
- Die Dateien `fomsdt.ins`, `fomsdt.dtx` und `readme.txt` in das Verzeichnis `$TEXROOT/tex/latex/fomsdt` kopieren.
- Die Paketdokumentation, also die Datei `fomsdt.pdf`, ebenfalls in das Verzeichnis `$TEXROOT/tex/latex/fomsdt` kopieren! Oder, falls noch nicht vorhanden, in diesem Verzeichnis erst erstellen.
- Im Verzeichnis `$TEXROOT/tex/latex/fomsdt` eine Shell öffnen und dort den Befehl `pdflatex fomsdt.ins` eingeben und ausführen.
- Die nun vorhandene Datei `fomidx.ist` in das Verzeichnis `$TEXROOT/makeindex/fomsdt` verschieben!
- Zur Aktualisierung des \LaTeX -Systems am Kommando-prompt noch den Befehl `texhash` eingeben.

Happy \TeX ing!

4 Paketübersicht

Dieser Abschnitt verschafft einen Überblick über die einzelnen Komponenten der \LaTeX -Nomenklatur *fomsdt zur Erstellung wissenschaftlicher Arbeiten* und geht kurz auf die Aufgaben sowie die Bedeutung der einzelnen Komponenten ein.

- `fomsdt.cls` Die Datei `fomsdt.cls` meint *FOM Scientific Document Template* und enthält den Code der Dokumentenklasse des Pakets. Weiterhin werden von dieser Paketklasse verschiedene Optionen zur Änderung des Dokumentenlayouts angeboten. Zusätzlich verfügt diese Dokumentenklasse auch über Alternativen zur Deckblattgestaltung.
- `fomdss.sty` Die Datei `fomdss.sty` meint *FOM Document Start Section* und enthält Makros, die es dem Benutzer erlauben, den einleitenden Teil einer wissenschaftlichen Arbeit mit leichter Hand zu erstellen. Beispielsweise beinhaltet der Einleitungsteil das Inhaltsverzeichnis und ein Abbildungsverzeichnis. Weitere Verzeichnisse werden ebenfalls unterstützt.
- `fomdes.sty` Die Datei `fomdes.sty` meint *FOM Document End Section* und enthält Makros für den abschließenden Teil der wissenschaftlichen Arbeit. Beispielsweise beinhaltet der Dokumentenabschluss den Index, das Glossar und ähnliches.
- `fompar.sty` Die Datei `fompar.sty` meint *FOM Paragraph Definition* und enthält spezielle aber auch nützliche Makros zur Formatierung von Absätzen. Beispielsweise können hierdurch beidseitig eingezogene Absätze oder auch einseitig eingezogene Absätze mit einem Betreffbezeichner erzeugt werden.
- `fomfod.sty` Die Datei `fomfod.sty` meint *FOM Floating Object Definition* und enthält Makros zur Einbindung von Abbildungen, zur Gestaltung von Tabellen und zur Erzeugung von Codelistings. Der Vorteil dieser Definitionsdatei besteht in der Einfachheit der Anwendung der bereitgestellten Makros.
- `fomisc.sty` Die Datei `fomisc.sty` meint *FOM Miscellaneous Definition* und dient als Sammelbecken für weitere nützliche Makros, die vom Sinnzusammenhang in keiner anderen Definitionsdatei untergebracht werden können.
- `fomidx.ist` Die Datei `fomidx.ist` meint *FOM Index Style Definition* und enthält verschiedene Konfigurationen für eine ansprechendere Gestaltung des Indexes. Sie wird durch das Programm `makeindex` verwendet und ist daher nur indirekt verwendbar. Zudem enthält diese Datei keine Makros im klassischen Sinne. Aus diesem Grund wird innerhalb dieser Dokumentation auf die weiterführende Beschreibung der Datei `fomidx.ist` verzichtet.

Mehrsprachigkeit Das Paket `fomsdt` unterstützt derzeit noch nicht die Verwendung unterschiedlicher Sprachen. Somit ist der Benutzer aufgefordert, bei einer von Deutsch verschiedenen Sprache, die Standardwerte der jeweiligen Makros zu überschreiben.

5 Dokumentenklasse

Die Dokumentenklasse `fomsdt`, implementiert in der Datei `fomsdt.cls`, basiert auf der L^AT_EX-eigenen Klasse `article`. Sie verfügt somit über alle Optionen ihrer Basisklasse. Weiterhin verfügt `fomsdt` über eigene Optionen, die in diesem Abschnitt einzeln beschrieben werden.

`\documentclass` Verwendet wird die Klasse `fomsdt` analog zur Verwendung aller anderen L^AT_EX-Dokumentenklassen. Daher ist die erste Zeile der Hauptdatei eines neuen Dokuments basierend auf der Klasse `fomsdt` die Zeile `\documentclass[Optionen]{fomsdt}`.

Die Angabe mehrerer Parameter im Optionsteil des Kommandos `\documentclass` ist möglich, wobei einzelne Optionen durch ein Komma voneinander getrennt werden.

An dieser Stelle noch ein Kurzbeispiel zur Illustration der zuvor beschriebenen Details zur Verwendung der Klasse `fomsdt`.

```
\documentclass[nochapter,noheadrule]{fomsdt}
\begin{document}
  Hallo Welt!
\end{document}
```

5.1 Optionen

Die Klasse `fomsdt` verfügt über Optionen, die vornehmlich zur Steuerung des Layouts eines Dokuments dienlich sind. Diese Optionen werden im Folgenden näher beschrieben.

`moderntitle` Die wissenschaftliche Arbeit soll über ein Titelblatt mit „modernem“ Aussehen verfügen. Als Standard für den Stil des Titels wird *kein Stil* verwendet.

- `classictitle` Die wissenschaftliche Arbeit soll über ein Titelblatt mit „klassischem“ Aussehen verfügen. Als Standard für den Stil des Titels wird **kein Stil** verwendet.
- Wird als Stil des Titelblatts weder die Option `moderntitle` noch die Option `classictitle` angegeben, dann können zur Erzeugung des Titels weiterhin die L^AT_EX-eigenen Makros verwendet werden.
- Werden hingegen beide Stiloptionen des Titelblatts gemeinsam verwendet, die Reihenfolge ist dabei unerheblich, dann wird als Standard der klassische Titelstil verwendet.
- `lockflag` Diese Option schaltet den Sperrvermerk auf der Seite direkt hinter der Titelseite ein. Der Standardwert für die Verwendung des Sperrvermerks ist **aus**.
- Die Verwendung des Sperrvermerks ist jedoch nur bei den wissenschaftlichen Arbeiten notwendig, die der Öffentlichkeit unzugänglich sein sollen. Dies ist beispielsweise häufig bei Diplomarbeiten der Fall, da sie eventuell firmeninterne Daten enthalten.
- `affidavit` Mit Hilfe dieser Option wird die eidesstattliche Versicherung des Autors auf der letzten Seite der wissenschaftlichen Arbeit aktiviert. Der Standardwert für die Verwendung der eidesstattlichen Versicherung ist **aus**.
- Verwendet wird die eidesstattliche Versicherung in der Regel nur bei Diplomarbeiten. Daher kann bei Studienarbeiten, Hausarbeiten oder Fallstudien grundsätzlich auf die eidesstattliche Versicherung verzichtet werden.
- `noheadrule` Diese Option schaltet die Linie zwischen der Kopfzeile und dem restlichen Text aus. Der Standardwert für die Verwendung dieser Trennlinie ist **an**.
- `nochapter` Diese Option schaltet die Anzeige des Kapitels in der Kopfzeile aus. Der Standardwert für die Verwendung des Kapitels in der Kopfzeile ist **an**.
- `coloredlinks` Die wissenschaftliche Arbeit zeigt Hyperlinks in PDF-Ausgabedateien in *blau*. Der Standardwert für die Hyperlinkfarbe ist **schwarz**.

5.2 Makros

Werden in den Klassenoptionen entweder `moderntitle` oder `classictitle` als Stil für die Titelseite angegeben, dann ist die Angabe weiterer Informationen unumgänglich. Hierfür werden von `fomsdt` entsprechende Makros bereitgestellt.

- `\General` Dieser Wert legt die Art der wissenschaftlichen Ausarbeitung fest. Es handelt sich hierbei um einen optionalen Parameter. Der Standardwert ist ***Hausarbeit***.
- `\Date` Dieser Wert gibt das Datum der wissenschaftlichen Ausarbeitung an. Es handelt sich hierbei um einen optionalen Parameter. Der Standardwert ist ***Essen, den \today***.
- `\Institute` Dieser Wert legt fest, an welcher Hochschule die wissenschaftliche Ausarbeitung geschrieben wurde. Es handelt sich hierbei um einen optionalen Parameter. Der Standardwert ist ***FOM – Fachhochschule für Oekonomie & Management Essen***.
- `\Course` Dieser Wert definiert den Studiengang, für den die wissenschaftliche Ausarbeitung geschrieben wurde. Es handelt sich hierbei um einen optionalen Parameter. Der Standardwert ist ***Berufsbegleitender Studiengang***.
- `\Semester` Dieser Wert gibt das Semester an, in dem die wissenschaftliche Ausarbeitung erstellt wurde. Es handelt sich hierbei um einen optionalen Parameter. Der Standardwert ist ***nichts***.
- `\Subject` Dieser Pflichtwert gibt entweder das Fach oder das Thema der wissenschaftlichen Ausarbeitung an.
- `\Title` Dieser Pflichtwert definiert den Titel der wissenschaftlichen Ausarbeitung.
- `\Professor` Dieser Pflichtwert legt den Betreuer des Studierenden fest. Das Makro verfügt sowohl über einen optionalen Parameter als auch über einen Pflichtparameter. Die Verwendung erfolgt als `\Professor[<Label>]{<Name>}`. Der Standardwert für den optionalen Parameter *<Label>* ist ***Betreuer:***.
- `\Student` Dieser Pflichtwert legt den Autor der wissenschaftlichen Ausarbeitung fest. Das Makro verfügt sowohl über einen optionalen Parameter als auch über einen Pflichtparameter. Die Verwendung erfolgt als `\Student[<Label>]{<Name>}`. Der Standardwert für den optionalen Parameter *<Label>* ist ***Autor:***.

An dieser Stelle zur Illustration ein Beispiel zur Verwendung der Parameter für die Titelseite.

```
\General{Hausarbeit}
\Date{Stadt, den \today}
\Institute{FOM}
\Course{Berufsbegleitendes Studium}
\Semester{6. Semester}
\Subject{Hausarbeit}
\Title{Betriebssysteme}
\Professor[Betreuer:]{Prof. Karl Muster}
\Student[Autor:]{Karin Muster (291067)}
\maketitle
```

Wird zur Erstellung der Arbeit das Programm `pdflatex` verwendet, dann besteht die Möglichkeit, zusätzliche Angaben über das erstellte Dokument in den PDF-Eigenschaften zu hinterlegen.

Wird hingegen das Programm `latex` anstelle von `pdflatex` zur Erstellung der Arbeit verwendet, dann werden eventuell gesetzte PDF-Eigenschaften ignoriert.

In jedem Fall ist bei der Verwendung der im Folgenden beschriebenen Makros zu beachten, dass sie in der Präambel, also noch vor der Zeile `\begin{document}`, zu verwenden sind!

- `\PDFTitle` Dieser Wert setzt die Eigenschaft *Titel* der PDF-Ausgabedatei. Der Standardwert ist `\leer`.
- `\PDFAuthor` Dieser Wert setzt die Eigenschaft *Autor* der PDF-Ausgabedatei. Der Standardwert ist `\leer`.
- `\PDFSubject` Dieser Wert setzt die Eigenschaft *Thema* der PDF-Ausgabedatei. Der Standardwert ist `\leer`.
- `\PDFProducer` Dieser Wert setzt die Eigenschaft *Erzeuger* der PDF-Ausgabedatei. Der Standardwert ist ***PDFLaTeX***.

Zur Illustration hier ein Beispiel zur Verwendung der zuvor genannten Makros.

```
\PDFTitle{Betriebssysteme}
\PDFAuthor{Karin Muster}
\PDFSubject{Hausarbeit}
\PDFProducer{MiKTeX}
```

Soll in einer wissenschaftlichen Ausarbeitung ein Sperrvermerk verwendet werden, dann ist die Angabe weiterer Informationen notwendig. Hierfür werden von `fomsdt` die im Folgenden beschriebenen Makros zur Verfügung gestellt.

`\LockflagContent` Mit Hilfe des Makros `\LockflagContent` [*Signatur*]{*Inhalt*} ist es möglich, den Inhalt des Sperrvermerks zu beeinflussen. Hierbei bestimmt der Pflichtparameter {*Inhalt*} den Text Sperrvermerks wie folgt:

`std` Wird `\LockflagContent`{`std`} verwendet, dann enthält der Sperrvermerk einen Standardtext, der bereits zum Zeitpunkt des Ladens der Dokumentenklasse `fomsdt` vor-eingestellt wird.

`fom` Wird `\LockflagContent`{`fom`} verwendet, dann enthält der Sperrvermerk einen FOM-spezifischen Text.

`...` Wird weder `std` noch `fom` als Parameter verwendet, dann enthält der Sperrvermerk den in {`...`} angegebenen und somit frei wählbaren Text.

Der optionale Parameter [*Signatur*] erlaubt das explizite Einschalten der Signatur des Sperrvermerks wie folgt:

`sig` Wird als optionaler Parameter `sig` verwendet, dann enthält der Sperrvermerk eine aus den Feldern (`Ort`, `Datum`) und (`Eigenhändige Unterschrift`) bestehende Signatur.

`\LockflagDate` Dieses Makro wird verwendet, um Ort und Datum der Signatur des Sperrvermerks festzulegen. Hierfür ist jedoch die Angabe des optionalen Parameters [*Signatur*] des Markos `\LockflagContent` notwendig. Anderenfalls hat die Verwendung von `\LockflagDate` keine Auswirkung. Der Standardwert für `\LockflagDate` ist `leer`.

`\LockflagCompany` Dieses Makro wird verwendet, um den Namen der Firma festzulegen, der innerhalb des Sperrvermerks zu verwenden ist. Der Standardwert ist `???`.

Achtung! Die Verwendung des Sperrvermerks verursacht Abhängigkeiten zu anderen `fom`-Makros, die an dieser Stelle kurz zusammengefasst werden:

- Wird `\LockflagContent{fom}` verwendet, dann werden durch den FOM-spezifischen Sperrvermerk die Inhalte der Makros `\General`, `\Title` und `\LockflagCompany` verwendet!
- Wird `\LockflagContent{std}` verwendet, dann werden durch den Standardsperrvermerk die Inhalte der Makros `\General` und `\LockflagCompany` verwendet!
- Wird `\LockflagContent{freier Text}` verwendet, dann besteht keine Abhängigkeit zu anderen Makros!
- Wird `\LockflagContent[sig]{Inhalt}` verwendet, dann wird durch die Signatur des Sperrvermerks der Inhalt des Makros `\LockflagDate` verwendet!

<code>\LockflagHeader</code>	Dieses Makro dient der Änderung der Überschrift des Sperrvermerks. Hierfür wird im Parameter <code>{Überschrift}</code> der geänderte Text der Überschrift angegeben. Der Standardwert für <code>\LockflagHeader</code> ist <code>(Sperrvermerk)</code> .
<code>\LockflagSigLeft</code>	Dieses Makro wird verwendet, um den linken Teil der Signatur des Sperrvermerks zu ändern. Hierfür wird im Parameter <code>{linker Signaturteil}</code> der geänderte Text des linken Signaturteils angegeben. Der Standardwert für <code>\LockflagSigLeft</code> ist <code>((Ort, Datum))</code> .
<code>\LockflagSigRight</code>	Dieses Makro wird verwendet, um den rechten Teil der Signatur des Sperrvermerks zu ändern. Hierfür wird im Parameter <code>{rechter Signaturteil}</code> der geänderte Text des rechten Signaturteils angegeben. Der Standardwert für <code>\LockflagSigRight</code> ist <code>((Eigenhändige Unterschrift))</code> .
<code>\AffidavitContent</code>	Das Makro <code>\AffidavitContent[Signatur]{Inhalt}</code> ermöglicht es, den Inhalt der eidesstattlichen Versicherung zu beeinflussen. Hierbei bestimmt der Pflichtparameter <code>{Inhalt}</code> den Text der eidesstattlichen Versicherung wie folgt:

`std` Wird `\AffidavitContent{std}` verwendet, dann enthält die eidesstattliche Versicherung einen Standardtext, der bereits zum Zeitpunkt des Ladens der Dokumentenklasse `fomsdt` voreingestellt wird.

... Wird der Parameter `std` nicht verwendet, dann enthält die eidesstattliche Versicherung den in `{...}` angegebenen und somit frei wählbaren Text.

Der optionale Parameter `[\langle Signatur \rangle]` erlaubt das explizite Einschalten der Signatur der eidesstattlichen Versicherung wie folgt:

`sig` Wird als optionaler Parameter `sig` verwendet, dann enthält die eidesstattliche Versicherung eine aus den Feldern (Ort, Datum) und (Eigenhändige Unterschrift) bestehende Signatur.

`\AffidavitDate` Dieses Makro wird verwendet, um Ort und Datum der Signatur der eidesstattlichen Versicherung festzulegen. Hierfür ist jedoch die Angabe des optionalen Parameters `[\langle Signatur \rangle]` des Markos `\AffidavitContent` notwendig. Anderenfalls hat die Verwendung von `\AffidavitDate` keine Auswirkung. Der Standardwert ist `\langle leer \rangle`.

Achtung! Die Verwendung der eidesstattlichen Versicherung verursacht Abhängigkeiten zu anderen `fomsdt`-Makros, die an dieser Stelle kurz zusammengefasst werden:

- Wird `\AffidavitContent{std}` benutzt, dann wird von der eidesstattlichen Versicherung der Inhalt des Makros `\General` verwendet!
- Wird `\AffidavitContent{\langle freier Text \rangle}` benutzt, dann besteht keine Abhängigkeit zu anderen Makros!
- Wird `\AffidavitContent[sig]{\langle Inhalt \rangle}` benutzt, dann wird durch die Signatur der eidesstattlichen Versicherung der Inhalt des Makros `\AffidavitDate` verwendet!

- `\AffidavitHeader` Dieses Makro dient der Änderung der Überschrift der eidesstattlichen Versicherung. Hierfür wird im Parameter `{\Überschrift}` der geänderte Text der Überschrift angegeben. Der Standardwert für `\AffidavitHeader` ist `\langleEhrenwörtliche Erklärung\rangle`.
- `\AffidavitSigLeft` Dieses Makro wird verwendet, um den linken Teil der Signatur der eidesstattlichen Versicherung zu ändern. Hierfür wird im Parameter `{\linker Signaturteil}` der geänderte Text des linken Signaturteils angegeben. Der Standardwert für `\AffidavitSigLeft` ist `\langle(Ort, Datum)\rangle`.
- `\AffidavitSigRight` Dieses Makro wird verwendet, um den rechten Teil der Signatur der eidesstattlichen Versicherung zu ändern. Hierfür wird im Parameter `{\rechter Signaturteil}` der geänderte Text des rechten Signaturteils angegeben. Der Standardwert für `\AffidavitSigRight` ist `\langle(Eigenhändige Unterschrift)\rangle`.

Zur Unterstützung der Verständlichkeit der zuvor beschriebenen Zusammenhänge hier noch einige Beispiele zur Verwendung des Sperrvermerks und der eidesstattlichen Versicherung.

```
% Ein Dokument mit Titelseite und Sperrvermerk für
% wissenschaftliche Arbeiten aber OHNE eidesstattliche
% Versicherung. Der Sperrvermerk 'fom', der Inhalt von
% \General und die Signatur inklusive Signaturdatum
% werden verwendet.
\documentclass[lockflag,classictitle]{fomsdt}
\begin{document}
\General{Fallstudie}
\Title{Betriebssysteme}
\Professor{Prof. Karl Muster}
\Student{Karin Muster (291067)}
\LockflagDate{Essen, den \today}
\LockflagCompany{Müller GmbH \& Co KG}
\LockflagContent[sig]{fom}
\maketitle% Sperrvermerk wird erzeugt!
Der Text des Dokuments ...
\end{document}
```

```

% Ein Dokument mit Titelseite und eidesstattlicher
% Versicherung für wissenschaftliche Arbeiten aber
% OHNE Sperrvermerk. Die eidesstattliche Versicherung
% verwendet einen freien Text ohne Signatur.
\documentclass[affidavit,classictitle]{fomsdt}
\AffidavitContent{Hiermit versichere ich ...}
\begin{document}
\General{Fallstudie}
\Title{Betriebssysteme}
\Professor{Prof. Karl Muster}
\Student{Karin Muster (291067)}
\maketitle
Der Text des Dokuments ...
\end{document}% Eidesst. Versicherung wird erzeugt!

```

6 Dokumentenbeginn

Jede wissenschaftliche Arbeit beginnt mit einer Reihe von Verzeichnissen, welche die Struktur der Arbeit widerspiegeln. Die Bereitstellung dieser Verzeichnisse ist Aufgabe von `fomdss`, welche in der Datei `fomdss.sty` implementiert ist.

`\usepackage` Die Verwendung des Stils `fomdss` erfolgt analog der Verwendung aller anderen \LaTeX -Stildateien. Somit wird in der Präambel der Hauptdatei eines neuen Dokuments die Zeile `\usepackage[(Optionen)]{fomdss}` eingetragen.

Die Angabe mehrerer Parameter im Optionsteil des Kommandos `\usepackage` ist möglich, wobei einzelne Optionen durch ein Komma voneinander getrennt werden.

Zur Illustration ein Beispieldokument, das ein Inhaltsverzeichnis mit einer Haupt- und zwei Unterüberschriften erzeugt. Der Bezeichner des Inhaltsverzeichnisses wird ebenfalls mit in das generierte Inhaltsverzeichnis eingetragen.

```

\documentclass{fomsdt}
\usepackage[toctotoc]{fomdss}
\begin{document}
\PrintContents
\section{Überschrift 1}

```

```

\subsection{Überschrift 1.1}
\subsection{Überschrift 1.2}
\end{document}

```

6.1 Optionen

Zur Steuerung des Layouts der verwendeten Verzeichnisse verfügt `fomdss` über die im Folgenden gezeigten Optionen. Bei unbekanntem oder nicht unterstützten Optionen wird eine Warnmeldung ausgegeben, die zur Information des Benutzers dient.

- `toctotoc` Diese Option fügt die Überschrift des Inhaltsverzeichnisses mit in das Inhaltsverzeichnis selbst ein. Der Standardwert ist ***nicht einfügen***.
- `nopagebreak` Diese Option verhindert den erzwungenen Seitenwechsel nach jeweils dem Abkürzungs-, dem Abbildungs- und dem Tabellenverzeichnis. Sie kann besonders bei kurzen Verzeichnissen nützlich sein. Dies gilt jedoch nicht für das Inhaltsverzeichnis! Der Standardwert ist ***Seitenumbruch***.
- `abbreviation` Diese Option legt fest, ob das Abkürzungsverzeichnis überhaupt erzeugt werden soll. Dies ist notwendig, da das Abkürzungsverzeichnis mit Hilfe des Pakets `nomencl` erstellt wird, das standardmäßig *nicht* installiert ist! Bei Verwendung dieser Option wird auf der Ausgabe eine Meldung angezeigt, wie unter Verwendung des Programms `makeindex` das Abkürzungsverzeichnis erstellt werden muss. Der Standardwert ist ***nicht verwenden***.
- `abbrevrefpage` Diese Option legt fest, ob im Abkürzungsverzeichnis eine Seitenreferenz für das jeweils erste Auftreten einer Abkürzung innerhalb des Textes enthalten sein soll. Sie ist aber nur dann wirksam, wenn `fomdss` ebenfalls mit der Option `abbreviation` geladen wird. Anderenfalls hat die Option `abbrevrefpage` keine Auswirkungen. Als Standardwert wird ***keine Seitenreferenz*** im Abkürzungsverzeichnis verwendet.

6.2 Makros

Ungeachtet der Ladeoptionen werden von `fomdss` zur Erstellung verschiedener Verzeichnisses die hier gezeigten Makros zur Verfügung gestellt.

`\PrintContents` Dieses Makro dient der Erzeugung des Inhaltsverzeichnisses. Es verfügt über einen optionalen Parameter zur Anpassung des Titels dieses Verzeichnisses. Als Standardwert wird ***Inhalt*** verwendet.

`\PrintFigures` Dieses Makro dient der Erzeugung des Abbildungsverzeichnisses. Es verfügt über einen optionalen Parameter zur Anpassung des Titels dieses Verzeichnisses. Als Standardwert wird ***Abbildungen*** verwendet.

`\PrintTables` Dieses Makro dient der Erzeugung des Tabellenverzeichnisses. Es verfügt über einen optionalen Parameter zur Anpassung des Titels dieses Verzeichnisses. Als Standardwert wird ***Tabellen*** verwendet.

Wurde `fomdss` zuvor mit der Option `abbreviation` geladen, dann stehen spezielle Makros zur Erzeugung des Abkürzungsverzeichnisses bereit.

`\PrintAbbrev` Dieses Makro dient der Erzeugung des Abkürzungsverzeichnisses. Es verfügt über einen optionalen Parameter zur Anpassung des Titels dieses Verzeichnisses. Als Standardwert wird ***Abkürzungen*** verwendet.

`\Abbrev` Mit Hilfe des Makros `\Abbrev{<Abkürzung>}{<Beschreibung>}` werden innerhalb des normalen Textes die Passagen markiert, die später in das Abkürzungsverzeichnis übernommen werden sollen. Dabei beinhaltet der erste Parameter die Abkürzung selbst und der zweite Parameter enthält die Bedeutung der im ersten Parameter angegebenen Abkürzung. Siehe hierzu nachfolgendes Beispiel.

`\Mark` Das Makro `\Mark{<Buchstabe>}` dient der Hervorhebung der relevanten Buchstaben innerhalb der Beschreibung einer Abkürzung. Alle hierdurch eingefassten Buchstaben werden später im Abkürzungsverzeichnis unterstrichen dargestellt. Siehe hierzu nachfolgendes Beispiel.

Zur Illustration und zur Verbesserung des Verständnisses der Verwendung der Makros `\Abbrev` und `\Mark` an dieser Stelle das bereits angekündigte Beispiel.

```
\documentclass{fomsdt}
\usepackage[abbreviation]{fomdss}
\begin{document}
\PrintAbbrev
\Abbrev{IC} {\Mark{I}ntegrated \Mark{C}ircuit}
\Abbrev{PIN}{\Mark{P}ersonal \Mark{I}dentification
             \Mark{N}umber}
\end{document}
```

Der dargestellte Codeschnipsel bewirkt, dass im Abkürzungsverzeichnis die beiden Einträge **IC** und **PIN** wie folgt hinzugefügt werden:

```
IC . . . . . Integrated Circuit
PIN . . . Personal Identification Number
```

Hinweis Für die Darstellung von Abkürzungen, Punkte inklusive, wird im Abkürzungsverzeichnis die fest eingestellte Breite von 20% der aktuellen Textbreite verwendet. In den meisten Fällen ist dies auch völlig ausreichend. Sollte die voreingestellte Standardbreite jedoch einmal nicht ausreichend sein, kann sie mit Hilfe des Kommandos

```
\setlength{\nomlabelwidth}{\langle Breite \rangle}
```

neu eingestellt werden. Diese Anpassung kann sowohl in der Präambel als auch direkt vor Aufruf des Makros `\PrintAbbrev` erfolgen.

Die Änderung des Layouts der im Abkürzungsverzeichnis eingetragenen Abkürzungen ist ebenfalls möglich. Hierfür wird dann das folgende Kommando verwendet:

```
\renewcommand{\nomlabel}[1]{\langle Formatierung \rangle}
```

Möchte man zum Beispiel die Abkürzungen des Abkürzungsverzeichnisses als schräg gestellten Text und ohne Füllpunkte darstellen, dann kann dies durch Anwendung des Kommandos

```
\renewcommand{\nomlabel}[1]{\textsl{#1}}
```

erreicht werden. Auch hierbei gilt, dass die Anpassung entweder in der Präambel oder direkt vor Aufruf des Makros `\PrintAbbrev` erfolgen sollte.

Unter Umständen kann es sinnvoll sein, die im Abkürzungsverzeichnis verwendete Standardseitenreferenz zu ändern. Soll nun beispielsweise die Seitenreferenz in **, page** (*Seite*) geändert werden, dann kann dies durch die Anwendung des folgenden Kommandos erfolgen:

```
\renewcommand{\pagedeclaration}[1]{, page~#1}
```

Wie oben bereits angesprochen, wird zur Erzeugung des Abkürzungsverzeichnisses ein spezielles \LaTeX -Paket verwendet. Dieses Paket wird *nur* zusammen mit dem Programm `makeindex`, in Verbindung mit speziellen Aufrufparametern, verwendet. Daher an dieser Stelle die Syntax zur Verwendung von `makeindex` zur Erstellung des Abkürzungsverzeichnisses.

```
$ makeindex <file>.nlo -s nomencl.ist -o <file>.nls
```

7 Dokumentenende

Jede wissenschaftliche Arbeit endet mit einer Reihe von Verzeichnissen, die dem Leser als zusätzlich Informationsquelle dienen. Die Bereitstellung dieser Verzeichnisse ist Aufgabe von `fomdes`, welche in der Datei `fomdes.sty` implementiert ist.

`\usepackage` Die Verwendung des Stils `fomdes` erfolgt analog der Verwendung aller anderen \LaTeX -Stildateien. Somit wird in der Präambel der Hauptdatei eines neuen Dokuments die Zeile `\usepackage[(Optionen)]{fomdes}` eingetragen.

Die Angabe mehrerer Parameter im Optionsteil des Kommandos `\usepackage` ist möglich, wobei einzelne Optionen durch ein Komma voneinander getrennt werden.

7.1 Optionen

Zur Wahl der Verzeichnisse am Ende einer wissenschaftlichen Arbeit verfügt `fomdes` über die im Folgenden gezeigten Optionen. Bei unbekanntem oder nicht unterstütztem Optionwert wird eine Warnmeldung ausgegeben, die zur Information des Benutzers dient.

- `listings` Diese Option aktiviert die Verwendung des Verzeichnisses für Codelistings. Der Standardwert für die Verwendung des Verzeichnisses für Codelistings ist ***aus***.
- `glossary` Diese Option aktiviert die Verwendung des Glossars beziehungsweise des Sachwortverzeichnisses. Der Standardwert für die Verwendung des Glossars ist ***aus***. Wird diese Option verwendet, dann wird während des Kompilierens eine Meldung ausgegeben, die den Benutzer darauf hinweist, dass zur Erstellung des Glossars das Kommando `bibtex <file>.gls` aufgerufen werden muss.
- `glossrefpage` Diese Option legt fest, ob im Sachwortverzeichnis eine Seitenreferenz für das jeweils erste Auftreten eines Glossareintrages enthalten sein soll. Sie ist aber nur dann wirksam, wenn `fomdes` ebenfalls mit der Option `glossary` geladen wird. Anderenfalls hat die Option `glossrefpage` keine Auswirkungen. Als Standardwert wird ***keine Seitenreferenz*** im Glossar verwendet.
- `nobibliography` Diese Option verhindert die Aktivierung des Literaturverzeichnisses. Der Standardwert für die Verwendung des Literaturverzeichnisses ist ***an***. Wird diese Option *nicht* verwendet, dann wird während des Kompilierens eine Meldung ausgegeben, die den Benutzer darauf hinweist, dass zur Erstellung des Literaturverzeichnisses das Kommando `bibtex <file>.aux` aufgerufen werden muss.
- `index` Diese Option aktiviert die Verwendung des Indexes beziehungsweise des Stichwortverzeichnisses. Der Standardwert für die Indexerzeugung ist ***aus***. Wird diese Option verwendet,

dann wird während des Kompilierens eine Meldung ausgegeben, die den Benutzer darauf hinweist, dass zur Erstellung des Indexes das Kommando `makeindex <file>.idx` aufgerufen werden muss.

`nopagebreak` Diese Option verhindert den erzwungenen Seitenwechsel nach jeweils jedem der oben beschriebenen Verzeichnisse. Sie kann besonders bei kurzen Verzeichnissen nützlich sein. Der Standardwert ist **Seitenumbruch**.

7.2 Makros

`\PrintListings` Das Makro `\PrintListings[⟨Überschrift⟩]` dient der Erstellung des Verzeichnisses für Codelistings. Hierbei ist es durch den optionalen Parameter möglich, die Überschrift des Listingverzeichnisses zu ändern. Der Standardwert für die Überschrift des Listingverzeichnisses ist **Listings**.

Zur Aktivierung des Makros `\PrintListings` ist es notwendig, den Stil `fomdes` zuvor mit der Option `listings` zu laden. Anderenfalls bleibt die Verwendung von `\PrintListings` wirkungslos.

`\PrintGlossary` Das Makro `\PrintGlossary [⟨Überschrift⟩] [a11] {⟨Glossardatei⟩}` dient der Erstellung des Glossars oder auch Sachwortverzeichnisses. Dieses Makro verfügt, im Gegensatz zu dem zuvor beschriebenen Makro, nicht nur über optionale Parameter, sondern auch über einen Pflichtparameter!

Mit Hilfe der Option `[⟨Überschrift⟩]` kann die Überschrift des Glossars individuell verändert werden. Der Standardwert für die Glossarüberschrift ist **Glossar**.

Der zweite optionale Parameter `[a11]` erzwingt die Ausgabe aller in der Bibliothek vorhandenen Glossareinträge. Dieser Parameter ist besonders dann hilfreich, wenn in der Glossarbibliothek zusätzliche Datensätze vorhanden sind, die zuvor im Text noch nicht verwendet wurden.

Achtung! Die Angabe des optionalen Parameters `[a11]` führt zur Warnmeldung `Package gloss Warning: Term '*' on page ⟨n⟩ not defined in glossary 'default' on input line ⟨m⟩!`

Empfehlung Von der Verwendung des Parameters [a11] sollte *nur* zu Testzwecken und in Ausnahmefällen Gebrauch gemacht werden, da Sachwortverzeichnisse in wissenschaftlichen Arbeiten ausschließlich Begriffe enthalten, die auch im Text verwendet werden!

Auf Grund der Tatsache, dass `fomdes` zur Erzeugung eines Glossars das Paket `gloss` verwendet, müssen die einzelnen Glossareinträge in einer externen Datei gespeichert werden. Die Angabe dieses Dateinamens ist Aufgabe des Pflichtparameters `{\Glossardatei}`. Diese Angabe erfolgt jedoch immer *ohne* die Dateiergänzung!

Da die Erstellung eines Glossars etwas komplexer ist, soll das folgende Beispiel der Darstellung aller notwendigen Schritte zur Erstellung des Glossars dienen.

- Als erstes wird unter dem Dateinamen `glsbsp.bib` die Bibliothek des Glossars angelegt und in dieser Datei der Datensatzes „gls1“ wie folgt eingetragen:

```
@GLOSSDEF{gls1,
  word      = {Lange Beschreibung},
  short     = {kurz},
  definition = {Hier ist beschrieben,
               was "kurz" bedeutet.}}
```

- Im nächsten Schritt wird ein \LaTeX -Dokument mit dem Dateinamen `glsbsp.tex` erzeugt, welches über den folgenden Inhalt verfügt:

```
\documentclass{fomsdt}
\usepackage[glossary,nopagebreak]{fomdes}
\begin{document}
In diesem Text befindet sich
"\gloss[short]{gls1}" ein Hinweis
auf {\bf\gloss{gls1}}.
% glsbsp.bib OHNE .bib einbinden!
\PrintGlossary{glsbsp}
\end{document}
```

- Nachdem zuvor alle Dateien gespeichert wurden, kann mit dem Kompilervorgang begonnen werden. Hierfür sind die folgenden Kommandos der Reihe nach auszuführen:

```
$ pdflatex glsbsp.tex
$ bibtex glsbsp.gls
$ pdflatex glsbsp.tex
$ pdflatex glsbsp.tex
```

Hinweis Werden in der Glossarbibliothek Datensätze hinzugefügt oder geändert, ist ein erneuter Aufruf des Programms `bibtex` erforderlich!

Unter Umständen kann es sinnvoll sein, die im Glossar verwendete Standardseitenreferenz zu ändern. Soll nun beispielsweise die voreingestellte Seitenreferenz (**Seite** *<Seite>*) in zum Beispiel (**see page** *<Seite>*) geändert werden, dann kann dies durch die Anwendung des folgenden Kommandos erfolgen:

```
\renewcommand{\glosspage}[1]{\space(see page~#1)}
```

An dieser Stelle noch die vom Paket `gloss` unterstützten Schlüsselwörter zur Definition von Datensätzen in einer Glossarbibliothek.

```
@GLOSSDEF{label,           % Pflichtangabe
  word      = "word",      % Pflichtangabe
  definition = "definition", % Pflichtangabe
  short     = "short",     % Optional
  sort-word = "sort-word", % Optional
  group     = "group",     % Optional
  heading   = "heading"    % Optional
}
```

`\PrintBibliography` Das Makro `\PrintBibliography[<Überschrift>][<Stil>]{<Literaturdatei>}` dient der Erstellung des Literaturverzeichnisses. Es verfügt ebenfalls über zwei optionale Parameter und auch über einen Pflichtparameter!

Mit Hilfe der Option [*<Überschrift>*] kann die Überschrift des Literaturverzeichnisses individuell verändert werden. Der

Standardwert für die Überschrift des Literaturverzeichnisses ist **Literatur**.

Der zweite optionale Parameter [*Stil*] ermöglicht die Änderung des Layouts des zu erstellenden Literaturverzeichnisses. Als Angabe wird hier der Name einer Datei mit der Endung `.bst` erwartet! Der Standardwert für das Layout des Literaturverzeichnisses ist das DIN-gerechte Layout **dinat**.

Empfehlung Für die Erstellung wissenschaftlicher Arbeiten wird von der Verwendung der Option [*Stil*] abgeraten, da dies ein Abweichen vom DIN-gerechten Standardlayout zur Folge hat!

Auf Grund der Tatsache, dass `fomdes` zur Erzeugung des Literaturverzeichnisses das Paket `natbib` verwendet, müssen die einzelnen Literaturverweise in einer externen Datei gespeichert werden. Die Angabe dieses Dateinamens ist Aufgabe des Pflichtparameters $\{\langle\textit>Literaturdatei}\}$. Diese Angabe erfolgt jedoch immer *ohne* die Dateiendung!

Da die Erstellung eines Literaturverzeichnisses etwas komplexer ist, soll das folgende Beispiel der Darstellung aller notwendigen Schritte zur Erstellung des Literaturverzeichnisses dienen.

- Als erstes wird unter dem Dateinamen `litbsp.bib` die Literaturbibliothek angelegt und in dieser Datei der Datensatz „bib1“ wie folgt eingetragen:

```
@BOOKLET{bib1,
  title   = {The Art of Computer Programming},
  author  = {Donald~E. Knuth},
  year    = {1999},
  month   = {Januar},
  url     = {http://www.amazon.de/},
  urldate = {2007-08-01}}
```

Der hier gezeigten Datensatz dient ausschließlich zur Demonstration! Aus diesem Grund wird an dieser Stelle auf die umfangreich vorhandene Literatur zur Erstellung von Literaturverzeichnissen verwiesen, da eine ausführliche Beschreibung der vielfältigen Variationsmöglichkeiten den Rahmen dieser Anleitung sprengen würde.

- Im nächsten Schritt wird ein L^AT_EX-Dokument mit dem Dateinamen `litbsp.tex` erzeugt, welches über den folgenden Inhalt verfügt:

```

\documentclass{fomsdt}
\usepackage[nopagebreak]{fomdes}
\begin{document}
In diesem Text befindet sich ein Verweis
auf eine Literaturquelle~\cite{bib1}.
% litbsp.bib OHNE .bib einbinden!
\PrintBibliography{litbsp}
\end{document}

```

- Nachdem zuvor alle Dateien gespeichert wurden, kann mit dem Kompilervorgang begonnen werden. Hierfür sind die folgenden Kommandos der Reihe nach auszuführen:

```

$ pdflatex litbsp.tex
$ bibtex litbsp.aux
$ pdflatex litbsp.tex
$ pdflatex litbsp.tex

```

Zu bemerken bleibt noch, dass einige Versionen des Programms `bibtex` die Angabe der Dateiendung `.aux` erlauben und andere Implementierungen die Annahme dieser Dateiendung verweigern. In diesem Fall würde dann das Kommando `bibtex <file>.aux` zu einer Fehlermeldung führen, die in etwa `I couldn't open file name 'litbsp.aux.aux'` lautet. In einem solchen Fall sollte dann die Dateiendung einfach weggelassen werden.

Hinweis Werden in der Literaturbibliothek Datensätze hinzugefügt oder geändert, ist ein erneuter Aufruf des Programms `bibtex` erforderlich!

`\PrintIndex` Das Makro `\PrintIndex[⟨Überschrift⟩][⟨Spalten⟩]` dient der Erstellung des Indexes. Hierbei ist es möglich, über die beiden optionalen Parameter, das Aussehen des Stichwortverzeichnis zu beeinflussen.

Mit Hilfe der Option [*Überschrift*] kann die Überschrift des Indexes individuell verändert werden. Der Standardwert für die Indexüberschrift ist **Index**.

Mit Hilfe der Option [*Spalten*] ist es möglich, die Anzahl der Spalten des Stichwortverzeichnisses zu variieren. Gültige Werte für die Spaltenzahl sind:

single Wird `\PrintIndex[...][single]` verwendet, dann besteht das Stichwortverzeichnisses aus einer einzelnen Spalte.

double Wird `\PrintIndex[...][double]` verwendet, dann besteht das Stichwortverzeichnisses aus zwei Spalten.

triple Wird `\PrintIndex[...][triple]` verwendet, dann besteht das Stichwortverzeichnisses aus drei Spalten.

Wird der optionale Parameter [*Spalten*] nicht angegeben, dann wird **double** als Standardwert für die Anzahl der Spalten des Stichwortverzeichnisses verwendet.

Das folgende kleine Beispiel demonstriert die Verwendung des Makros `\PrintIndex` in Verbindung mit der zur Aktivierung notwendigen Paketoption `index`.

```
\documentclass{fomdsd}
\usepackage[index]{fomdes}
\begin{document}
Dieser Text\index{Text} verwendet einen
Index\index{Index\see{Text}} mit zwei
Einträgen sowie einem Querverweis im
Index selbst.
\PrintIndex
\end{document}
```

Besonderheit Im Zusammenhang mit der Erstellung eines Stichwortverzeichnisses sind es noch einige Besonderheiten zu beachten. Grundsätzlich gilt hierbei, dass die alleinige Aktivierung des Indexes, zum Beispiel durch die Verwendung von `\PrintIndex`, nicht

ausreicht, um den Index auch tatsächlich in das erzeugte Dokument zu übertragen. Hierfür ist zusätzlich der Aufruf des Programms `makeindex` erforderlich!

Weiterhin ist die Einhaltung eine bestimmte Reihenfolge der Schritte zur Dokumentenerstellung notwendig. Diese Reihenfolge soll nun näher erläutert werden:

- Das Dokument wird zuerst einmal kompiliert. Dies kann durch das Kommando `pdflatex <file>.tex` erfolgen.
- Nun *muss* das Dokument ein zweites mal kompiliert werden! Somit erfolgt eine erneute Anwendung des Kommandos `pdflatex <file>.tex`. Dieser zweite Kompilierungsvorgang ist erforderlich, da erst hiernach alle Querverweise auf die „richtigen“ Seiten verweisen!
- Erst jetzt kann das Stichwortverzeichnis richtig erstellt werden. Dies geschieht durch Anwendung des Kommandos `makeindex <file>.idx`.
- Abschließend wird das Dokument ein weiteres mal mit dem Befehl `pdflatex <file>.tex` kompiliert.

Empfehlung Da die Indexerstellung mit der zuvor gezeigten Methode einige Probleme, wie beispielsweise die fehlerhafte Anzeige deutscher Umlaute, verursacht, wird anstelle des Kommandos `makeindex <hauptdatei>.idx` die Erzeugung des Stichwortverzeichnisses wie folgt empfohlen:

```
$ makeindex <file>.idx -g -s fomidx.ist
```

`\SeeAlso` Das Makro `\SeeAlso{<Verweis>}` ist für die einheitliche Formatierung von Querverweisen innerhalb des Indexes oder auch für Glossareinträge sehr hilfreich. Es erzeugt einen kleinen Pfeil mit dahinter befindlichen schräg gestelltem Text. Diese Art der Formatierung dient der besseren Hervorhebung.

Mit Hilfe der Option `{<Verweis>}` kann nicht nur reiner Text verarbeitet werden, vielmehr besteht auch die Möglichkeit, weitere Makros mit einzubinden, wie folgende kleine Beispiele demonstrieren.

Das erste Beispiel veranschaulicht die Verwendung des Makros `\SeeAlso` im Zusammenhang mit Querverweisen innerhalb eines Indexes. Nimmt man zum Beispiel an, dass im Index des Dokuments bereits ein Eintrag für „Bildschirm“ existiert und möchte man nun den Eintrag „Monitor“ mit einem Querverweis auf „Bildschirm“ zum Index hinzufügen, dann würde man im Quelltext des Dokuments folgendes eingeben:

```
... Dieser Monitor\index{Monitor\SeeAlso{Bildschirm}}
zeigt die Zustandsparameter des Gesamtsystems, ...
```

Nach erneuter Erstellung des Ausgabedokuments stellt sich das Ergebnis im Index wie folgt dar:

Monitor → *Bildschirm* (Seite)

Das zweite Beispiel zeigt die Verwendung des Makros `\SeeAlso` zusammen mit den Einträgen eines Glossars. Hierfür wird angenommen, dass in der Glossardatei `glsbsp.bib` die beiden Datensätze „pin“ und „pinpad“ bereits eingefügt wurden. Nun wird unter Anwendung des Makros `\SeeAlso` im Feld `definition` des Datensatzes `pinpad` der Querverweis auf den Datensatz `pin` wie hier dargestellt eingefügt:

```
@GLOSSDEF{pinpad,
  word      = {PIN"~Pad},
  definition = {Das PIN"~Pad wird zur Eingabe
               der \SeeAlso{\gloss[short]{pin}}
               verwendet.}}
```

Nach erneuter Erstellung des Ausgabedokuments stellt sich das Ergebnis im Glossar wie folgt dar:

PIN-Pad Das PIN-Pad wird zur Eingabe der → *PIN* verwendet.

8 Absatzdefinition

Die spezielle Formatierung von Absätzen ist in \LaTeX eine nicht-triviale Angelegenheit. Jedoch ist es in wissenschaftlichen Arbeiten üblich, Textpassagen besonders hervorzuheben. Daher bietet die Absatzdefinition `fompar`, die in der Datei `fompar.sty` implementiert ist, dem Benutzer einige hilfreiche Makros zur Hervorhebung von Absätzen.

`\usepackage` Die Verwendung des Stils `fompar` erfolgt analog der Verwendung aller anderen \LaTeX -Stildateien. Somit wird in der Präambel der Hauptdatei die Zeile `\usepackage{fompar}` eingefügt, wobei die Angabe von Optionen nicht erforderlich ist.

8.1 Makros und Umgebungen

Nachdem der Stil `fompar` geladen wurde, stehen dem Anwender die hier beschriebenen Makros zur Verfügung.

`Citation` Längere Zitate werden in wissenschaftlichen Arbeiten in der Regel in eigenen, zusammenhängenden Absätzen dargestellt. Dabei werden diese Zitate nicht nur durch doppelte Anführungszeichen hervorgehoben, sondern oft auch mit einem zusätzlichen linken und rechten Einzug versehen.

Die Formatierung dieser Art Textpassagen ist Aufgabe der Umgebung `\begin{Citation} ... \end{Citation}`.

Neben der optionalen Angabe des [*Abstands*] zum vorherigen und nachfolgenden Text, müssen zur Verwendung dieser Umgebung der *{linke}* und *{rechte}* Absatzezug unbedingt angegeben werden. Die genaue Anwendung der Umgebung `Citation` zeigt folgendes Beispiel.

```
\begin{Citation} % Begin of the environment
  [2\parskip] % Leading/following gap (optional)
  {12mm} % Left indent
  {1.5cm} % Right indent
  Text des Zitats...
\end{Citation} % End of the environment
```

Der Standardwert des optionalen Arguments [*Abstand*] ist auf den aktuellen Wert von `\parskip` eingestellt.

`\Remark` Das Makro `\Remark[⟨Einzug⟩][⟨Abstand⟩]{⟨Label⟩}{⟨Text⟩}` erzeugt einen linksseitig eingezogenen Absatz, dem ein linksbündig ausgerichteter Bezeichner vorangestellt ist. Daher eignet sich dieses Makro gut für die Hervorhebung wichtiger Informationen innerhalb eines Textes.

Die optionalen Parameter `[⟨Einzug⟩]` und `[⟨Abstand⟩]` ermöglichen sowohl die Anpassung des linksseitigen Einzugs als auch die Änderung des Abstandes zum vorherigen und nachfolgenden Text. Als Standardwert des linken Einzugs werden **3cm** und für den Abstand zum umgebenden Text wird der aktuell eingestellte Wert von `\parskip` verwendet.

An dieser Stelle nun ein kleines Beispiel zur Illustration der Verwendung des Makros `\Remark`.

```
\Remark[2cm]          % left indent
      [2\parskip] % gap before/after
      {\bf Label} % label, bold
      {Text...} % body text
```

`\Item` Das Makro `\Item[⟨Einzug⟩][⟨Abstand⟩]{⟨Label⟩}{⟨Text⟩}` erzeugt, genau wie das Makro `\Remark`, einen linksseitig eingezogenen Absatz. Jedoch wird hier dem Absatz ein rechtsbündiger Bezeichner vorangestellt. Somit eignet sich dieses Makro gut für die Hervorhebung von zum Beispiel Befehlen oder Kommandos eines Programms.

Der Standardwert des Parameters `[⟨Einzug⟩]` verwendet **4cm** und der Parameter `[⟨Abstand⟩]` ist auf den aktuellen Wert von `\parskip` eingestellt. Ansonsten erfolgt die Verwendung des Makros `\Item` analog zur Verwendung des Makros `\Remark`.

Werden innerhalb einer wissenschaftlichen Arbeit beide zuvor beschriebenen Makros `\Remark` und `\Item` häufiger verwendet und weichen dabei der Einzug und/oder der Textabstand von den Standardwerten ab, bietet sich eine grundsätzliche Anpassung der Standardwerten an. Zu diesem Zweck werden von `fompar` die im Folgenden beschriebenen Makros angeboten.

`\RemarkIndent` Das Makro `\RemarkIndent{⟨Einzug⟩}` ändert den Standardwert des linksseitigen Einzugs des Makros `\Remark` auf den im Argument `[⟨Einzug⟩]` angegebenen Wert.

<code>\DefaultRemarkIndent</code>	Das Makro <code>\DefaultRemarkIndent</code> setzt den Standardwert des linksseitigen Einzugs des Makros <code>\Remark</code> auf den Standard von 3cm zurück.
<code>\RemarkOffset</code>	Das Makro <code>\RemarkOffset{⟨Abstand⟩}</code> ändert den Standardwert des Abstandes zum umgebenden Text des Makro <code>\Remark</code> auf den im Argument [<code>⟨Abstand⟩</code>] angegebenen Wert.
<code>\DefaultRemarkOffset</code>	Das Makro <code>\DefaultRemarkOffset</code> setzt den Standardwert des Abstandes zum umgebenden Text des Makros <code>\Remark</code> auf den Standard <code>\parskip</code> zurück.
<code>\ItemIndent</code>	Das Makro <code>\ItemIndent{⟨Einzug⟩}</code> ändert den Standardwert des linksseitigen Einzugs des Makros <code>\Item</code> auf den im Argument [<code>⟨Einzug⟩</code>] angegebenen Wert.
<code>\DefaultItemIndent</code>	Das Makro <code>\DefaultItemIndent</code> setzt den Standardwert des linksseitigen Einzugs des Makros <code>\Item</code> auf den Standard von 4cm zurück.
<code>\ItemOffset</code>	Das Makro <code>\ItemOffset{⟨Abstand⟩}</code> ändert den Standardwert des Abstandes zum umgebenden Text des Makro <code>\Item</code> auf den im Argument [<code>⟨Abstand⟩</code>] angegebenen Wert.
<code>\DefaultItemOffset</code>	Das Makro <code>\DefaultItemOffset</code> setzt den Standardwert des Abstandes zum umgebenden Text des Makros <code>\Item</code> auf den Standard <code>\parskip</code> zurück.

Zur Verbesserung des Verständnisses der Anwendung der zuvor beschriebenen Makros an dieser Stelle noch ein Kurzbeispiel. Dabei sei nochmals daran erinnert, dass die Makros `\Remark` und `\Item` sich nur in der Ausrichtung ihrer Bezeichner unterscheiden.

```

\ItemIndent{25mm}
\ItemOffset{1.5\parskip}
\Item{\tt copy(){Der Befehl kopiert ...}
\Item{\tt move(){Der Befehl verschiebt ...}
\Item{\tt list(){Der Befehl listet ...}
\DefaultItemIndent
\DefaultItemOffset

```

Achtung! Zu Beginn des Dokuments werden von `fompar` die Standardwerte der Makros `\Remark` und `\Item` neu eingestellt. Aus diesem Grund ist eine Verwendung der Makros `\RemarkIndent`, `\RemarkOffset`, `\ItemIndent` und `\ItemOffset` in der Präambel nicht empfehlenswert!

`\Description` Das Makro `\Description[⟨Format⟩]{⟨Überschrift⟩}{⟨Text⟩}` generiert einen Absatz, der über eine Überschrift verfügt und ist als Alternative zum L^AT_EX-eigenen Makro `\paragraph` gedacht. Dabei unterscheidet sich `\Description` in zwei grundlegenden Eigenschaften vom Makro `\paragraph`.

- Der erste Unterschied besteht darin, dass die im Parameter `{⟨Überschrift⟩}` übergebene Überschrift in einer eigenen Zeile dargestellt wird, ohne dabei den gesamten Absatz zu „zerreißen“.
- Der zweite Unterschied zum Standardmakro `\paragraph` ist dadurch gegeben, dass die im Parameter `{⟨Überschrift⟩}` übergebene Überschrift *niemals* im Inhaltsverzeichnis der Ausarbeitung auftauchen wird, wie dies eventuell bei einer Verwendung von `\paragraph` der Fall sein könnte.

Neben den Pflichtparametern verfügt das Makro `\Description` auch über einen optionalen Parameter mit dessen Hilfe die Absatzüberschrift formatiert werden kann. Die Standardformatierung der Überschrift ist **fett**. Weiterhin ist die Angabe folgender Optionen möglich:

- `b` Die Überschrift ist fett (bold) (Standard).
- `n` Die Überschrift nicht (not) formatiert.
- `s` Die Überschrift ist schräg gestellt (slanted).
- `u` Die Überschrift ist unterstrichen (underlined).

Die beispielhafte Erzeugung eines Absatzes mit unterstrichener Überschrift zeigt der folgende kurze Codeschnipsel.

```
\Description[u]{Überschrift}{Text...}
```

9 Gleitobjektdefinition

Die Besonderheiten der Formatierung so genannter Gleitobjekte, dazu zählen Abbildungen, Tabellen und im weiteren Sinne auch Codelistings, werden in wissenschaftlichen Ausarbeitungen oft unterschätzt. Daher bietet die Gleitobjektdefinition `fomfod`, die in der Datei `fomfod.sty` implementiert ist, entsprechende Makros an, die der einfachen und einheitlichen Integration von Gleitobjekten in wissenschaftlichen Arbeiten dienen.

`\usepackage` Die Verwendung des Stils `fomfod` erfolgt analog der Verwendung aller anderen \LaTeX -Stildateien. Somit wird die Zeile `\usepackage[Optionen]{fomfod}` in der Präambel der Hauptdatei eingefügt.

Die Angabe mehrerer Parameter im Optionsteil des Kommandos `\usepackage` ist möglich, wobei einzelne Optionen durch ein Komma voneinander getrennt werden.

9.1 Optionen

Der Stil `fomfod` bietet Optionen zur grundsätzlichen Anpassung der Gleitobjekte einer wissenschaftlichen Arbeit an, die im Folgenden näher beschrieben sind. Bei unbekanntem oder nicht unterstützten Optionen wird eine Warnmeldung ausgegeben, die der Information des Benutzers dient.

`classiccaption` Diese Option aktiviert die Verwendung der Serifenschrift für die Beschriftungen der einzelnen Gleitobjekte. Als Standardeinstellung für Objektbeschriftungen wird eine ***serifenlose Schriftart*** verwendet.

`listingcolored` Diese Option aktiviert die Unterstützung der farblichen Hervorhebung von Schlüsselwörtern und Kommentaren innerhalb von Codelistings. Der Standardwert für die farbliche Hervorhebung ist ***nicht farbig***.

9.2 Makros und Umgebungen

Die von fomfod angebotenen Makros zur Einbindung von Gleitobjekten lassen sich in die drei Kategorien Abbildungen, Tabellen und Codelistings unterteilen, die im Folgenden auch in dieser Reihenfolge vorgestellt werden.

`\SinglePicture` Die Abbildung stellt in wissenschaftlichen Arbeiten die wohl am häufigsten verwendete Form von Gleitobjekten dar. Daher bietet das Makro `\SinglePicture` [*Breite*] [*box*] {*Beschriftung*} {*Referenzlabel*} {*Bilddatei*} {*Quellenangabe*} einem Benutzer die Möglichkeit, auf einfache Weise die Integration von Abbildungen zu realisieren.

Der optionale Parameter [*Breite*] ermöglicht eine Anpassung der Abbildungsbreite in Abhängigkeit von der aktuellen Textbreite `\textwidth`. Die Höhe der Abbildung wird vom \LaTeX -System automatisch angepasst. Der Standardwert für die Abbildungsbreite ist **0.9**, also 90% von `\textwidth`.

Der optionale Parameter [*box*] erzeugt einen Rahmen um die eingebundene Abbildung. Der Standardwert für Abbildungsrahmen ist **aus**.

Empfehlung Von der Darstellung eines Rahmens um Abbildungen sollte nur in seltenen Fällen gebrauch gemacht werden, da sich Rahmen und Linien jeglicher Art störend auf den Lesefluss auswirken können.

Der Pflichtparameter {*Beschriftung*} gestattet die Angabe der Abbildungsunterschrift.

Der Pflichtparameter {*Referenzlabel*} ermöglicht die Definition eines Bezeichners, mit dessen Hilfe auf die entsprechende Abbildung Bezug genommen werden kann. Üblicherweise folgt das Label einer Abbildung dem Schema `fig:<Label>`.

Der Pflichtparameter {*Bilddatei*} verwendet den Namen der Abbildungsdatei. Dabei können Pfade zu unterschiedlichen Abbildungsquellverzeichnissen mit Hilfe des \LaTeX -eigenen Makros `\graphicspath`{*relativer Pfad*} gesteuert werden.

Der Parameter {*Quellenangabe*} gehört ebenfalls zu den Pflichtangaben des Makros `\SinglePicture` und stellt im Hin-

blick auf wissenschaftliche Ausarbeitungen einen eher wichtigen Parameter dar.

Hinweis Es ist nicht nur höflich sondern auch zwingend erforderlich, in wissenschaftlichen Arbeiten das geistige Eigentum Dritter entsprechend zu Kennzeichnen! Daher wird bei Abbildungen, die einer anderen als die eigene Geistesleistung entspringen *immer* die Quelle angegeben. Diese Quellenangabe erfolgt typischerweise rechtsbündig und direkt unterhalb der jeweiligen Abbildung.

Zur Illustration und zur Verbesserung des Verständnisses nun ein Beispiel zur Verwendung des Makros `\SinglePicture`.

```
\SinglePicture[1.2] % 120% of text width
                    [box] % Including an image box
                    {Die Darstellung von Spiralen}
                    {fig:SpiraleA}
                    {spirale} % image file name
                    {Quelle: www.spiralen.de}
```

`\DoublePicture` In wissenschaftlichen Arbeiten kommt es häufig vor, dass Abbildungen zum Beispiel zum Vergleich gegenüber gestellt werden. Hierfür bietet das Makro `\DoublePicture [box] {\langle Hauptbeschriftung \rangle} {\langle Hauptreferenzlabel \rangle} {\langle linke Abbildung \rangle} {\langle rechte Abbildung \rangle}` den notwendigen Rahmen.

Der optionale Parameter `[box]` erzeugt einen Rahmen um die jeweilige Unterabbildung. Der Standardwert für Abbildungsrahmen ist **aus**.

Empfehlung Von der Darstellung eines Rahmens um Abbildungen sollte nur in seltenen Fällen gebrauch gemacht werden, da sich Rahmen und Linien jeglicher Art störend auf den Lesefluss auswirken können.

Der Pflichtparameter `{\langle Hauptbeschriftung \rangle}` gestattet die Angabe der Hauptunterschrift der Doppelabbildung.

Der Pflichtparameter `{\langle Hauptreferenzlabel \rangle}` ermöglicht die Definition eines Bezeichners, mit dessen Hilfe auf die entsprechende Doppelabbildung Bezug genommen werden kann. Üblicherweise folgt das Label einer Doppelabbildung dem Schema `fig:\langle Hauptlabel \rangle`.

Die beiden Pflichtparameter $\{\langle linke\ Abbildung\rangle\}$ und $\{\langle rechte\ Abbildung\rangle\}$ dienen der Definition der linken und rechten Unterabbildung und sind an dieser Stelle als Platzhalter für das im Folgende beschriebene Makro `\DoubleSubPicture` gedacht.

`\DoubleSubPicture` Die Definition der linken oder rechten Unterabbildung obliegt dem Makro `\DoubleSubPicture` $\{\langle Unterbeschriftung\rangle\}$ $\{\langle Unterreferenzlabel\rangle\}$ $\{\langle Unterbilddatei\rangle\}$ $\{\langle Unterquellenangabe\rangle\}$.

Der Pflichtparameter $\{\langle Unterbeschriftung\rangle\}$ gestattet die Angabe der Beschriftung der linken beziehungsweise der rechten Unterabbildung.

Der Pflichtparameter $\{\langle Unterreferenzlabel\rangle\}$ ermöglicht die vom Hauptreferenzlabel unabhängige Definition eines Bezeichners, mit dessen Hilfe auf die entsprechende Unterabbildung Bezug genommen werden kann. Üblicherweise folgt das Label einer Abbildung dem Schema `fig:\Label`.

Der Pflichtparameter $\{\langle Unterbilddatei\rangle\}$ verwendet den Namen der linken oder rechten Unterabbildungsdatei. Dabei können Pfade zu unterschiedlichen Abbildungsquellverzeichnissen mit Hilfe des L^AT_EX-eigenen Makros `\graphicspath{\langle relativer Pfad\rangle}` gesteuert werden.

Der Parameter $\{\langle Unterquellenangabe\rangle\}$ gehört ebenfalls zu den Pflichtangaben des Makros `\DoubleSubPicture` und stellt im Hinblick auf wissenschaftliche Ausarbeitungen einen eher wichtigen Parameter dar.

An dieser Stelle bitte ebenfalls den Hinweis zur Quellenangabe bei Abbildungen auf Seite 36 beachten!

Achtung! Die individuelle Anpassung der Abbildungsbreite ist weder bei `\DoublePicture` noch bei `\DoubleSubPicture` möglich! Vielmehr werden beide Abbildungen so platziert, dass jede in etwa 42% der aktuellen Textbreite verwendet. Zuzüglich einer kleinen Lücke zwischen beiden Abbildungen ist somit die Breite der gesamten Doppelabbildung auf circa 90% der verfügbaren Textbreite begrenzt.

Zur Illustration und zur Verbesserung des Verständnisses nun ein Beispiel zur Verwendung des Makros `\DoublePicture` zusammen mit dem Makro `\DoubleSubPicture`.

```

\DoublePicture
  [box] % Including image boxes
  {Der Vergleich von Linien} % Main caption
  {fig:LinienA} % Main label
  {\DoubleSubPicture % Left sub picture
    {Gerade Linien} % Left caption (a)
    {fig:LinienAleft} % Left label
    {linien} % Left image file.
    {Quelle: www.linien.de}}
  {\DoubleSubPicture % Right sub picture
    {Gekrümmte Linien} % Right caption (b)
    {fig:LinienAright} % Right label
    {kurven} % Right image file.
    {Quelle: www.kurven.de}}

```

Bemerkung Manch Einer wird sich fragen: 'Warum nur die Aufteilung in zwei voneinander unabhängige Makros?'. Die Antwort ist einfach! Das Makro `\DoublePicture` würde für eine vollständige Verarbeitung aller Informationen insgesamt elf Parameter benötigen. In \LaTeX ist jedoch die maximale Parameteranzahl auf insgesamt neun beschränkt. Aus diesem Grund erfolgte eine Unterteilung in zwei unabhängige Makros!

PlainFigure Eine weitere Variante zur Einbindung von Abbildungen in eine wissenschaftliche Arbeit bietet sich durch die Verwendung der frei konfigurierbaren Umgebung `\begin{PlainFigure}` [*Breite*] [`box`] [*Beschriftung*] [*Referenzlabel*] [*Quellenangabe*] ... `\end{PlainFigure}`

Der optionale Parameter [*Breite*] ermöglicht eine Anpassung der Abbildungsbreite in Abhängigkeit von der aktuellen Textbreite `\textwidth`. Die Höhe der Abbildung wird vom \LaTeX -System automatisch angepasst. Der Standardwert für die Abbildungsbreite ist **0.9**, also 90% von `\textwidth`.

Der optionale Parameter [`box`] erzeugt einen Rahmen um den eingebundenen Inhalt. Der Zwischenraum zwischen Rahmen und Inhalt beträgt **5pt**, also fünf Punkte. Dieser Zwischenraum kann von außen nicht geändert werden. Der Standardwert für Abbildungsrahmen ist **aus**.

Der Pflichtparameter [*Beschriftung*] gestattet die Angabe der Abbildungsunterschrift.

Der Pflichtparameter $\{\langle\textit{Referenzlabel}\rangle\}$ ermöglicht die Definition eines Bezeichners, mit dessen Hilfe auf die entsprechende Abbildung Bezug genommen werden kann. Üblicherweise folgt das Label einer Abbildung dem Schema `fig: $\langle\textit{Label}\rangle$` .

Der Parameter $\{\langle\textit{Quellenangabe}\rangle\}$ gehört ebenfalls zu den Pflichtangaben der Umgebung `PlainFigure` und stellt im Hinblick auf wissenschaftliche Ausarbeitungen einen eher wichtigen Parameter dar.

An dieser Stelle bitte ein weiteres mal den Hinweis zur Quellenangabe bei Abbildungen auf Seite 36 beachten!

Zur Illustration und zur Verbesserung des Verständnisses nun ein Beispiel zur Verwendung der Umgebung `PlainFigure`.

```
\begin{PlainFigure}[0.7] % 70% of text width
    [box] % Including an image box
    {Darstellung einer Textabbildung}
    {fig:TextFigure}
    {Quelle: www.texte.de}
    Hier befindet sich der frei
    wählbare Inhalt der Abbildung...
\end{PlainFigure}
```

Table Tabellen bilden ebenfalls einen wichtigen Bestandteil wissenschaftlicher Arbeiten. Da es jedoch hin und wieder vorkommt, dass Tabellen nicht standardgerecht formatiert werden, bietet `fomfod` die entsprechend konfigurierte Umgebung `\begin{Table}` $\{\langle\textit{Beschriftung}\rangle\}$ $\{\langle\textit{Referenzlabel}\rangle\}$ $\{\langle\textit{Quellenangabe}\rangle\}$... `\end{Table}` an.

Diese Umgebung ist in der Lage, die gängigen, zum \LaTeX -System gehörenden Umgebungen für Tabellendefinitionen zu verarbeiten.

Der Pflichtparameter $\{\langle\textit{Beschriftung}\rangle\}$ gestattet die Angabe der Überschrift der Tabelle.

Der Pflichtparameter $\{\langle\textit{Referenzlabel}\rangle\}$ ermöglicht die Definition eines Bezeichners, mit dessen Hilfe auf die entsprechende Tabelle Bezug genommen werden kann. Üblicherweise folgt das Label einer Tabelle dem Schema `tab: $\langle\textit{Label}\rangle$` .

Der Parameter `{\langle Quellenangabe \rangle}` gehört ebenfalls zu den Pflichtangaben der Umgebung `Table` und stellt im Hinblick auf wissenschaftliche Ausarbeitungen einen eher wichtigen Parameter dar.

Hinweis Auch bei Tabellen muss, sofern sie beziehungsweise deren Inhalt geistiges Eigentum Dritter sind, der Ursprung entsprechend kenntlich gemacht werden! Hierbei erfolgt die Quellenangabe typischerweise zentriert und direkt unterhalb der jeweiligen Tabelle.

Aufgrund der vielfältigen Möglichkeiten, eine Tabelle zu gestalten, ist die weitere Abstraktion und Vereinfachung der Erstellung von Tabellen nur schwer möglich. Daher soll das folgende Beispiel nur stellvertretend für die Tabellenerstellung sein.

```
\begin{Table}
  {Eine kleine Tabelle}      % Table caption
  {tab:small1}              % Reference label
  {Quelle: www.tabellen.de} % Table source
  \begin{tabular}{lcr}      % Table body
    links & mitte & rechts \\
  \end{tabular}
\end{Table}
```

Empfehlung Zur besseren Kontrolle der Formatierung einzelner Tabellenspalte wird die Verwendung der Umgebung `tabularx` des gleichnamigen Pakets `tabularx` empfohlen.

Wird nun eine Tabelle mit der Umgebung `tabularx` erstellt, dann stellt `fomfod` spezielle Spaltenformatierungsmakros bereit, auf die an dieser Stelle näher eingegangen wird.

- R Erzeugt eine rechts ausgerichtete Tabellenspalte.
- C Erzeugt eine zentriert ausgerichtete Tabellenspalte.
- L Erzeugt eine links ausgerichtete Tabellenspalte.
- B Erzeugt eine im Blocksatz ausgerichtete Tabellenspalte.

Zur Veranschaulichung an dieser Stelle noch ein kleines Beispiel zur Verwendung der fomfod-eigenen Umgebung `Table`, wie sie gemeinsam mit der Umgebung `tabularx` genutzt werden kann.

```

\begin{Table}
  {Eine weitere Tabelle}
  {tab:tabX}
  {Quelle: www.tabularx.de}
  \begin{tabularx}{8cm}
    {>{\hspace=2cm}L>{\hspace=1cm}C>{\hspace=5cm}R}
    links 2cm & mitte 1cm & rechts 5cm \\
  \end{tabularx}
\end{Table}

```

`CodeListing` Gerade auf dem Gebiet der Informatik kommt es immer wieder vor, dass so genannte Codelistings in wissenschaftliche Arbeiten integriert werden müssen. Die Einbindung dieser Quelltexte, sowie deren einheitliche Gestaltung, ist die vornehmliche Aufgabe der Umgebung `\begin{CodeListing}` [*Einzug*] `{\Beschriftung}` `{\Referenzlabel}` ... `\end{CodeListing}`

Der optionale Parameter [*Einzug*] kann verwendet werden, um den linken und rechten Einzug des Listings zu variieren. Dabei ist der linke und rechte Standardeinzug auf eine Breite von **7pt** voreingestellt.

Der Pflichtparameter `{\Beschriftung}` erlaubt die Angabe der Überschrift des Listings.

Der Pflichtparameter `{\Referenzlabel}` ermöglicht die Verwendung eines Bezeichners, mit dessen Hilfe auf das entsprechende Codelisting Bezug genommen werden kann. Üblicherweise folgt das Quelltextlabel dem Schema `1st:Label`.

Hier noch ein kleines Anwendungsbeispiel der Codelistingumgebung zur Anzeige des Quelltextes des C-Programms „Hallo Welt!“, inklusive einer farblichen Syntaxhervorhebung.

```

\documentclass{fomsdt}
\usepackage[listingcolored]{fomfod}
\begin{document}
\begin{CodeListing}[35mm]{"Hallo Welt!"}{1st:hello}

```

```

#include <stdio.h>
void main(void) {
    printf("Hallo Welt!\n");
}
\end{CodeListing}
\end{document}

```

Hinweis Die Umgebung `CodeListing` ist speziell für kurze Listings gedacht, da hierfür der gesamte Quellcode zwischen `\begin` und `\end` stehen muss. Bei längeren Quelltexten kann dies jedoch zu einer nicht zu unterschätzenden Unübersichtlichkeit führen!

`\CodeListingFile` Im Gegensatz zur zuvor beschriebenen Umgebung für Code-listings bietet das Makro `\CodeListingFile` [*Einzug*] {*Beschriftung*} {*Referenzlabel*} {*Quellcodedatei*} die Möglichkeit, den Inhalt ganzer Codedateien in eine wissenschaftliche Arbeit einzubinden.

Der optionale Parameter [*Einzug*] kann verwendet werden, um den linken und rechten Einzug des Listings zu variieren. Dabei ist linke und rechte Standardeinzug auf eine Breite von **7pt** voreingestellt.

Der Pflichtparameter {*Beschriftung*} erlaubt die Angabe der Überschrift des Listings.

Der Pflichtparameter {*Referenzlabel*} ermöglicht die Verwendung eines Bezeichners, mit dessen Hilfe auf das entsprechende Codelisting Bezug genommen werden kann. Üblicherweise folgt das Quelltextlabel dem Schema `1st:Label`.

Der Pflichtparameter {*Quellcodedatei*} ermöglicht die Angabe des Dateinamens der einzubindenden Quelltextdatei.

Hier nochmals das obige Beispiel für das C-Programm „Hallo Welt!“, jedoch nun mit der Einbindung der entsprechenden Quellcodedatei `hello.c`.

```

\documentclass{fomsdt}
\usepackage[listingcolored]{fomfod}
\begin{document}
\CodeListingFile{"'Hallo Welt!'"}{1st:hello}{hello.c}
\end{document}

```

`\ListingFontSize` Eventuell kann es vorkommen, dass die Darstellung der Zeichen eines Codelistings zu groß oder gar zu klein erscheinen. Dies könnte dann unter Umständen zu dem ungewollten Effekt führen, dass einzelnen Zeilen des Quelltextes umgebrochen werden. Das könnte durch die Anpassung der Schriftgröße behoben werden. Die Einstellung dieser Schriftgröße ist Aufgabe des Makros `\ListingFontSize` [*⟨Schriftgröße⟩*].

Der optionale Parameter [*⟨Schriftgröße⟩*] erlaubt hierbei die Verwendung vordefinierter Größenangaben zur Einstellung der bevorzugten Schriftgröße wie folgt:

`normal` Diese Option verwendet `\footnotesize` als Größe der Schrift und erlaubt damit die Darstellung von etwa 60 Zeichen pro Codezeile.

`large` Diese Option verwendet `\small` als Größe der Schrift und erlaubt damit die Darstellung von etwa 55 Zeichen pro Codezeile.

`small` Diese Option verwendet `\scriptsize` als Größe der Schrift und erlaubt damit die Darstellung von etwa 74 Zeichen pro Codezeile.

`\DefaultListingConfig` Das Makro `\DefaultListingConfig` dient der Einstellung aller Standardwerte für Codelistings. Einige dieser Einstellungen sind von besonderem Interesse und sollen aus diesem Grund etwas näher betrachtet werden.

Quelltextsprache Die Quelltextsprache ist nur dann von besonderem Interesse, wenn `fomfod` zuvor mit der Option `listingcolored` geladen wurde, das heißt, wenn auch tatsächlich die farbliche Hervorhebung für Quellcodes aktiviert wurde. In diesem Fall wird die Syntaxhervorhebung für Quelltexte in C++ verwendet!

Soll durch ein Listing die Syntaxhervorhebung einer anderen Quelltextsprache verwenden, dann kann die Standardeinstellung C++ mit Hilfe des Kommandos `\lstset{language=⟨Sprache⟩}`, das in `listings` definiert ist, geändert werden. Alle durch `listings` unterstützten Sprachen, inklusiver diverser Dialekte, werden ausführlich in der zu diesem Paket gehörenden Dokumentation beschrieben.

Beschriftung Für die Positionierung der Beschriftung eines Codelistings kann keine verbindliche Aussage getroffen werden. Sinnvoll erscheint jedoch, die Beschriftung an den Anfang des Quelltextes zu setzen, da ein Leser bei einer Textunterbrechung durch zum Beispiel ein Codelisting zumindest einen Hinweis darauf erwartet, warum sein Lesefluss gestört wurde.

Sollte man aus irgendeinem Grund von der vorkonfigurierten Beschriftungsposition oberhalb des Listings abweichen müssen, dann kann dies durch den im Paket `listings` definierten Befehl `\lstset{captionpos=b}` erreicht werden.

Zeilenumbruch Unter Umständen kann es vorkommen, dass Codezeilen eines eingebundenen Listings trotz der Anpassung des Schriftgrades nicht in eine einzelne Textzeile passen. Für diesen Fall wurde `fomfod` so konfiguriert, dass das Codelisting selbstständig einen Umbruch der betroffenen Zeilen durchführt und eine so umgebrochene Codezeile dann mit `...` beendet und in der nächsten Zeile mit einer Einrückung und weiteren `...` fortgeführt wird.

10 Vermischtes

Als Sammelbecken für weiter nützliche und hilfreiche Makros dient `fomisc`, das in der Datei `fomisc.sty` implementiert ist. Hierin befinden sich Makros, die von ihrem Sinnzusammenhang in keiner anderen Datei des Pakets untergebracht werden können.

`\usepackage` Die Verwendung des Stils `fomisc` erfolgt analog der Verwendung aller anderen \LaTeX -Stildateien. Somit wird in der Präambel der Hauptdatei die Zeile `\usepackage{fomisc}` eingefügt, wobei die Angabe von Optionen nicht erforderlich ist.

10.1 Makros

`\Raise` Grundsätzlich kann hochgestellter Text durch den mathematischen Satz erzeugt werden. Jedoch führt die Verwendung des mathematischen Satzes nicht immer zum erwarteten Ergebnis.

Beispielsweise wird hierbei hochgestellter Text kursiv dargestellt. Dies kann durchaus irritierend auf den Leser einer wissenschaftlichen Arbeit wirken. Daher wird zur verbesserten Darstellung hochgestellter Texte die Verwendung des Makros `\Raise{<Hochtext>}` empfohlen.

`\TodayShort` Manchmal kann es sehr hilfreich sein, das aktuelle Datum in einer kurzen Variante anzugeben. Zu diesem Zweck wird durch `fomisc` das Makro `\TodayShort` [`<Format>`] [`<Trennzeichen>`] bereitgestellt.

Dabei bestimmt der optionale Parameter [`<Format>`] das grundsätzlich Aussehen des erzeugten Datums. Als Standardwert für diesen Parameters wird ***din*** verwendet. Weiterhin sind die im Folgenden gezeigten vordefinierten Schlüsselwörter für diesen Parameter verfügbar:

din Diese Option verwendet `DD.MM.YYYY` zur Formatierung des aktuellen Datums und entspricht damit den Empfehlungen gemäß Norm DIN 5008.

iso Diese Option verwendet `YYYY-MM-DD` zur Formatierung des aktuellen Datums und entspricht damit den Empfehlungen gemäß der Normen ISO 8601:2004 und EN 28601.

Die Zeichen, die zur Trennung der einzelnen Bestandteile des Datums verwendet werden, sind bereits normengerecht! Sollte dennoch eine Anpassung der Trennzeichen erfolgen müssen, zum Beispiel für eine US-amerikanische Formatierung mit Schrägstrichen als Trennzeichen, dann kann dieses vom Standard abweichende Trennzeichen durch den optionalen Parameter [`<Trennzeichen>`] geändert werden. Dieser Parameter verwendet ***def*** als Standardwert.

def Diese Option bewirkt die Verwendung des Standardtrennzeichens für Datumsangaben in Abhängigkeit vom gewählten Datumsformat.

Im Falle einer ISO-gerechten Formatierung, also bei Angabe von ***iso*** im Parameter [`<Format>`], wird der Gedankenstrich (`'-'`) zur Trennung verwendet.

Für den Fall der DIN-gerechten Formatierung, also bei Angabe von `din` im Parameter [$\langle Format \rangle$], wird ein Punkt (‘.’) als Trennzeichen verwendet.

... Das angegebene, beliebige Zeichens, oder auch eine Zeichenkette, wird zur Trennung der Datumsbestandteile verwendet. Die Verwendung eines „leeren“ Zeichens ist ebenfalls möglich.

Folgende Übersicht zeigt beispielhaft die Anwendungsmöglichkeiten des Makros `\TodayShort` sowie die hieraus resultierenden Ergebnisse:

<code>\TodayShort</code>	01.08.2007	⇒ DIN-gerecht
<code>\TodayShort[din]</code>	01.08.2007	⇒ DIN-gerecht
<code>\TodayShort[din][def]</code>	01.08.2007	⇒ DIN-gerecht
<code>\TodayShort[din] []</code>	01082007	⇒ DIN ohne Trennung
<code>\TodayShort[din] [/]</code>	01/08/2007	⇒ US-amerikanisch
<code>\TodayShort[iso]</code>	2007-08-01	⇒ ISO-konform
<code>\TodayShort[iso][def]</code>	2007-08-01	⇒ ISO-konform
<code>\TodayShort[iso] []</code>	20070801	⇒ ISO ohne Trennung
<code>\TodayShort[iso] [/]</code>	2007/08/01	⇒ US-amerikanisch

Autoren wissenschaftlicher Arbeiten beziehen sich häufig auf Produkte und Marken von Herstellern, die mit dem Vermerk „Registered“, „Copyright“ oder „Trademark“ belegt sind. Hierfür werden in der Regel spezielle Symbole verwendet, die von `fomisc` bereitgestellt werden.

- `\Registered` Trägt ein Produkt oder eine Marke den Vermerk „Registered“, dann kann das Makro `\Registered` für die Erzeugung des entsprechenden Symbols verwendet werden.
- `\Copyright` Trägt ein Produkt oder eine Marke den Vermerk „Copyright“, dann kann das Makro `\Copyright` für die Erzeugung des entsprechenden Symbols verwendet werden.
- `\Trademark` Trägt ein Produkt oder eine Marke den Vermerk „Trademark“, dann kann das Makro `\Trademark` für die Erzeugung des entsprechenden Symbols verwendet werden.
- `\Axel` Das Makro `\Axel` dient der Illustration der Erzeugung von Texten, deren einzelne Buchstaben „aus der Reihe tanzen“. Beispiele solcher Texte sind die Eigennamen `TEX` und `LATEX`.

Bekannte Probleme

Dieses Kapitel verschafft einen Überblick über bekannte Probleme und erläutert deren Zusammenhang oder auch deren Umgehung. Treten während der Verwendung des Pakets `fomdsd` weitere, bisher nicht bekannte Probleme, Konflikte oder Fehler auf, dann sollten die Benutzer nicht vor einer entsprechenden Email zurückschrecken.

`\PrintIndex` Dieses durch die Datei `fomdes.sty` definiert Kommando wird ebenfalls durch die Datei `doc.sty` definiert und ist somit nach der Installation des Pakets `fomdsd` doppelt im \LaTeX -System verfügbar. Hierdurch sollten jedoch keinerlei Beeinträchtigungen im Umgang mit dem Paket `fomdsd` entstehen, da die Datei `doc.sty` in aller Regel nur im Zusammenhang mit der Erstellung der Quellcodedokumentation von \LaTeX -Paketen verwendet wird.

Leeres Verzeichnis Die Anwendung der Makros `\Print<Verzeichnis>` führen unter Umständen zu dem Effekt, dass die korrespondierenden und automatisch erstellten Verzeichnisse *nur* aus ihrer Überschrift bestehen und somit über keinerlei sinnvollen Inhalt verfügen.

Tritt dieser Effekt auf, dann sollte zunächst einmal überprüft werden, ob die benötigten Querverweise überhaupt im Dokument vorhanden sind. Als Beispiel hierfür sei das Fehlen von `\index{<Begriff>}` bei gleichzeitiger Anwendung von `\PrintIndex` genannt.

Als nächstes sollte geprüft werden, ob zur Erstellung des jeweiligen Verzeichnisses nicht noch weitere Kompilierungsvorgänge notwendig sind. Gegebenenfalls ist auch eine Anwendung der Programme `makeindex` und/oder `bibtex` erforderlich.

cygwin Von der Erstellung der \LaTeX -Dokumente zusammen mit dem Paket `fomdsd` wird unter *cygwin*, einer Linux-artigen Umgebung für Windowssysteme, dringend abgeraten, da hierbei gleich mehrere Probleme auftreten können.

Eines dieser Probleme ist, dass die Erstellung des Indexes zusammen mit dem Stil `fomidx.ist` fehlschlägt, da in der unter *cygwin* verfügbaren \LaTeX -Umgebung einige Abhängigkeiten für die DIN-gerechte, deutschsprachige Indexerstellung standardmäßig nicht installiert sind!

Seitenvorschub Das Erzwingen eines Seitenvorschubes für einzelne Kapitel durch Änderung des Kommandos `\section` führt eventuell zu dem unerwünschten Nebeneffekt, dass die Verzeichnisse der Stile `fomdss.sty` und `fomdes.sty` trotz angegebener Option `nopagebreak` nicht fortlaufend ausgegeben werden. Dieses Problem kann durch den hier gezeigten Workaround umgangen werden:

```
...
\begin{document}
...
\PrintContents
\PrintAbbrev
\PrintFigures
\PrintTables
...
% Seitenvorschub zu Kapitelbeginn erzwingen!
\let\tempsection\section
\renewcommand\section{\clearpage\tempsection}
... the writing ...
% Seitenvorschub zurücksetzen!
\let\section\tempsection
...
\PrintListings
\PrintBibliography
\PrintGlossary
\PrintIndex
...
\end{document}
```

beamer-Klasse Wird `fomfod` zusammen mit der Dokumentenklasse `beamer` geladen, dann treten hierbei verschiedentliche Konflikte und Fehler in den Paketen `caption` und `subfig` auf. Zwar sind diese Probleme den entsprechenden Paketentwicklern bekannt, jedoch existiert derzeit noch keine geeignete Fehlerbehebung.

Aus diesem Grund wird das Laden des Pakets `subfig` unterdrückt, wenn `fomfod` zusammen mit der Klasse `beamer` verwendet wird. Entsprechende Paketwarnungen werden während des Kompilervorgangs durch `fomfod` ausgegeben.

Diese Unterdrückung hat leider auch zur Folge, dass eine Verwendung des Makros `\DoublePicture` nicht mehr möglich ist!

Wird `\DoublePicture` unter diesen Umständen dennoch verwendet, dann wird in der Präsentation anstelle der einzelnen Bilder eine entsprechende Textmeldung angezeigt.

Wohlgemerkt, die zuvor beschriebenen Probleme treten nur dann auf, wenn `fomfod` zusammen mit der Dokumentenklasse `beamer` verwendet wird!

Abhängigkeiten

Dieser Abschnitt verschafft einen Überblick über die Abhängigkeiten des Paket `fomsdt` zu anderen \LaTeX -Pakete.

- `a4wide` Das Paket zur Verwendung der gesamten verfügbaren Seitengröße für DIN A4 wird ohne Optionen durch `fomsdt` geladen.
 - `caption` Das Paket zur Formatierung von Unter- und Überschriften für Abbildungen, Tabellen und Listings wird ohne Optionen durch `fomfod` geladen.
 - `color` Das Paket zur Unterstützung von farbigen Texten, die in Dokumenten unter anderem für Hyperlinks verwendet werden, wird ohne Optionen durch `fomsdt` und `fomfod` geladen.
 - `fancyhdr` Das Paket zur Unterstützung der Formatierung von Kopf- und Fußzeilen wird ohne Optionen durch `fomsdt` geladen.
 - `fontenc` Das Paket zur Unterstützung verschiedener Schriftarten wird mit der Option `T1` durch `fomsdt` geladen.
 - `footmisc` Das Paket zur Unterstützung der Formatierung von Fußnoten mit hängendem Einzug wird mit den Optionen `hang` und `stable` durch `fomsdt` geladen.
 - `gloss` Das Paket zur Erzeugung des Sachwortverzeichnisses wird mit der Option `german` durch `fomdes` geladen, wenn dieser Stil zuvor mit der Option `glossary` geladen wurde. Wurde die Option `glossrefpage` ebenfalls mit angegeben, dann wird das Paket `gloss` zusätzlich mit der Option `refpages` geladen.
- Achtung!** Eventuell ist das Paket `gloss` *nicht* im \LaTeX -System installiert und muss deshalb nachinstalliert werden, wenn in der wissenschaftlichen Arbeit ein Glossars verwendet werden soll! Empfohlen wird die Installation des Pakets `gloss` in der Version v1.5.2 vom 26. 07. 2002 (oder höher).

- `graphicx` Das Paket zur Grafikerunterstützung wird ohne Optionen durch `fomfod` und `fomisc` geladen.
- `hyperref` Das Paket zur Unterstützung von Hyperlinks wird mit den Optionen `colorlinks = true`, `bookmarks = true` und `plainpages = false` geladen. Es wird durch `fomsdt` nur dann geladen, wenn ein Dokument mit dem Programm `pdflatex` erstellt wird.
- `ifpdf` Das Paket zur Prüfung des Ausgabeformats PDF bei Verwendung des Programms `pdflatex` wird ohne Optionen durch `fomsdt`, `fomdss` und `fomdes` geladen.
- `ifthen` Das Paket zur Unterstützung von Bedingungsabfragen wird zur intern Auswertung von Übergabeparameter verwendet und durch `fomsdt`, `fomfod`, `fomdes` und `fomisc` geladen.
- `inputenc` Das Paket zur Unterstützung des Zeichensatzes für Eingabedaten wird mit der Option `latin1` durch `fomsdt` geladen.
- `listings` Das Paket zur Unterstützung von Codelistings wird ohne Optionen durch `fomfod` geladen.
- `makeidx` Das Paket zur Unterstützung der Indexerstellung wird ohne Optionen durch `fomdes` geladen, wenn dieser Stil mit zuvor der Option `index` geladen wurde.
- `multicol` Das Paket zur Unterstützung der mehrerer Spalten wird ohne Optionen durch `fomdes` geladen, wenn dieser Stil zuvor mit der Option `index` geladen wurde.
- `natbib` Das Paket zur Erzeugung des Literaturverzeichnisses wird mit den Optionen `square` und `numbers` durch `fomdes` geladen, wenn dieser Stil zuvor *nicht* mit der Option `nobibliography` geladen wurde. In diesem Fall erfolgt die Erzeugung eines Literaturverzeichnisses unter Verwendung des DIN-gerechten Verzeichnistils ***dinat***.
- Achtung!** Eventuell ist die Stildatei `dinat.bst` zur DIN-gerechten Literaturverzeichniserstellung *nicht* im \LaTeX -System installiert und muss deshalb nachinstalliert werden!
- `ngerman` Das Paket zur Unterstützung der Besonderheiten der deutschen Sprache wird ohne Optionen durch `fomsdt` geladen.
- `nomenc` Das Paket zur Erstellung des Abkürzungsverzeichnisses wird nur dann geladen, wenn `fomdss` zuvor mit der Option `abbreviation` geladen wurde. In diesem Fall wird `nomenc` mit der

Option `german` geladen. Wurde `fomdss` ebenfalls mit der Option `abbrevrefpage` geladen, dann wird `nomencl` zusätzlich mit der Option `refpage` geladen.

- Achtung!** `nomencl` ist grundsätzlich *nicht* im \LaTeX -System installiert und muss deshalb eventuell nachinstalliert werden! Auf einigen Systemen ist dieses Paket bereits in einer älteren und leider auch inkompatiblen Version installiert. In diesem Fall sollte die vorinstallierte Version des Pakets `nomencl` mindestens durch die Version v4.2 vom 22. 09. 2005 (oder höher) ersetzt werden!
- `rotating` Das Paket zur Unterstützung von rotierten Texten wird ohne Optionen durch `fomsdt` geladen.
- `subfig` Das Paket Unterstützung von Untergrafiken wird ohne Optionen durch `fomfod` geladen.
- `tabularx` Das Paket zur Unterstützung von Tabellen mit fester Spaltenbreite wird ohne Optionen durch `fomfod` geladen.
- `textfit` Das Paket zur Skalierung von Texten wird ohne Optionen durch `fomisc` geladen.
- `textpos` Das Paket zur Unterstützung spezieller Textboxen wird mit der Option `overlay` durch `fomsdt` geladen, wenn die Klasse zuvor entweder mit der Option `moderntitle` oder der Option `classictitle` geladen wurde.
- `ulem` Das Paket zur Unterstützung von Unterstreichungen im Abkürzungsverzeichnisses wird mit der Option `normalem` geladen. Dieses Paket wird nur dann geladen, wenn `fomdss` mit der Option `abbreviation` geladen wurde. Weiterhin wird `ulem` ohne weitere Einschränkung durch `fompar`, ebenfalls mit der Option `normalem`, geladen.

Weiterführende Literatur

- [1] BROOKS, Moses; CARSTEN, Heinz: The `Listings` Package. Januar 2007 (Link zu: `listings.pdf`).
- [2] CARLISLE, David: The `tabularx` package. Januar 1999 (Link zu: `tabularx.pdf`).

- [3] DALY, Patrick W.: Natural Sciences Citations and References. Februar 2007 (Link zu: [natbib.pdf](#)).
- [4] DÍAZ, Jose Luis; BEZOS, Javier: The gloss Package. Juli 2002 (Link zu: [gloss.pdf](#)).
- [5] GONZATO, Guido: L^AT_EX for Word Processor Users. Dezember 2003 (Link zu: [latex4wp.pdf](#)).
- [6] GOOSSENS, Michel; SAMARIN, Alexander; MITTELBACH, Frank: The L^AT_EX companion. Addison-Wesley Professional; Auflage: 2nd ed.; Mai 2004.
- [7] JÜRGENS, Manuela: L^AT_EX - eine Einführung und ein bisschen mehr. März 2000 (Link zu: [a026.pdf](#)).
- [8] JÜRGENS, Manuela: L^AT_EX - Fortgeschrittene Anwendungen. Oktober 1995 (Link zu: [a027.pdf](#)).
- [9] KOPKA, Helmut; DALY, Patrick W.: A guide to L^AT_EX. Addison-Wesley Professional; 3 Sub edition; Januar 1999.
- [10] LAMPORT, Leslie: *MakeIndex*: An Index Processor For L^AT_EX. Februar 1987 (Link zu: [makeindex.pdf](#)).
- [11] MÖSGEN, Peter: Makeindex, Sachregister erstellen mit L^AT_EX. Mai 1998 (Link zu: [makeidx.pdf](#)).
- [12] OETIKER, Tobias; PARTL, Hubert; SCHLEGL, Elisabeth; HYNA, Irene: The Not So Short Introduction to L^AT_EX 2_ε. Juni 2007 (Link zu: [lshort.pdf](#)).
- [13] RECKDAHL, Keith: Using Imported Graphics in L^AT_EX and pdfL^AT_EX. Januar 2006 (Link zu: [epslatex.pdf](#)).
- [14] SCHMIDT, Walter; KNAPPEN, Jörg; PARTL, Hubert; HYNA, Irene: L^AT_EX 2_ε-Kurzbeschreibung. April 2003 (Link zu: [12kurz.pdf](#)).
- [15] VEYTSMAN, Boris; SCHANDL, Bernd: nomencl: A Package to Create a Nomenclature. September 2005 (Link zu: [nomencl.pdf](#)).

Index

- A**
- a4wide *Abhängigkeit* 49
 - Abbildung → *Gleitobjekt* 35
 - Abbildungsverzeichnis 18
 - \Abbrev *Makro* 18
 - abbreviation *Option* 17
 - abbrevrefpage *Option* 17
 - Abhängigkeiten
 - a4wide 49
 - article 8
 - beamer 48
 - caption 48, 49
 - color 49
 - fancyhdr 49
 - fontenc 49
 - footmisc 49
 - gloss 23, 24, 49
 - graphicx 50
 - hyperref 50
 - ifpdf 50
 - ifthen 50
 - inputenc 50
 - listings 43, 44, 50
 - makeidx 50
 - multicol 50
 - natbib 25, 50
 - ngerman 50
 - nomencl 17, 50
 - rotating 51
 - subfig 48, 51
 - tabularx 40, 51
 - textfit 51
 - textpos 51
 - ulem 51
 - Abkürzungsverzeichnis . . . 18
 - Achtung 51
 - Hinweis 19
 - Absatzformatierung 30
 - affidavit *Option* 9
 - \AffidavitContent *Makro* . 13
 - \AffidavitDate *Makro* 14
 - \AffidavitHeader *Makro* . . 15
 - \AffidavitSigLeft *Makro* . 15
 - \AffidavitSigRight *Makro* . 15
 - article *Abhängigkeit* 8
 - \Axel *Makro* 46
- B**
- beamer *Abhängigkeit* 48
 - Befehle
 - bibtex 21, 24, 26, 47
 - initexmf 5
 - latex 11
 - makeindex 7
 - makeindex 17, 20, 22, 28, 47
 - pdflatex 3, 5, 6, 11, 28, 50
 - texhash 5, 6
 - Beispiel
 - Abbildung 36, 37, 39
 - Abkürzungsverzeichnis . 19
 - Absatz 30–33
 - Glossar 23, 29
 - Hallo Welt 8, 41, 42
 - Index 27, 29
 - Inhaltsverzeichnis 16
 - Kurzdatum 46
 - Literaturverzeichnis . . . 25
 - PDF-Eigenschaften 11
 - Sperrvermerk 15
 - Tabelle 40, 41
 - Titelblatt 11
 - Versicherung (eidesst.) . 15
 - bibtex *Befehl* 21, 24, 26, 47
- C**
- caption *Abhängigkeit* 48, 49
 - Citation *Umgebung* 30
 - classiccaption *Option* . . . 34
 - classictitle *Option* 9
 - CodeListing *Umgebung* . 41, 42
 - \CodeListingFile *Makro* . . 42
 - color *Abhängigkeit* 49
 - coloredlinks *Option* 9

<code>\Copyright</code> Makro	46	<code>fomisc.sty</code> Datei	7, 44
<code>\Course</code> Makro	10	<code>fompar.sty</code> Datei	7, 30
D			
<code>\Date</code> Makro	10	<code>fomsdt.cls</code> Datei	7, 8
Dateien		<code>fomsdt.dtx</code> Datei	2, 4
<code>dinat.bst</code>	50	<code>fomsdt.ins</code> Datei	2, 4
<code>doc.sty</code>	47	<code>fomsdt.pdf</code> Datei	3, 4
<code>fomdes.sty</code>	7, 20, 47, 48	<code>fontenc</code> Abhängigkeit	49
<code>fomdss.sty</code>	7, 16, 48	<code>footmisc</code> Abhängigkeit	49
<code>fomfod.sty</code>	7, 34	G	
<code>fomidx.ist</code>	7, 28, 47	<code>\General</code> Makro	10, 13, 14
<code>fomisc.sty</code>	7, 44	Gleitobjekt	34
<code>fompar.sty</code>	7, 30	Abbildung	35, 36, 38
<code>fomsdt.cls</code>	7, 8	Empfehlung	35, 36
<code>fomsdt.dtx</code>	2, 4	Beschriftung	34
<code>fomsdt.ins</code>	2, 4	Codelisting	21, 41
<code>fomsdt.pdf</code>	3, 4	Tabelle	39
<code>readme.txt</code>	2, 4	Empfehlung	40
Zielverzeichnis	5	<code>gloss</code> Abhängigkeit	23, 24, 49
<code>\DefaultItemIndent</code> Makro	32	Glossar	21
<code>\DefaultItemOffset</code> Makro	32	Achtung	22, 49
<code>\DefaultListingConfig</code> Ma-		Empfehlung	23
kro	43	Hinweis	24
<code>\DefaultRemarkIndent</code> Ma-		<code>glossary</code> Option	21
kro	32	<code>glossrefpage</code> Option	21
<code>\DefaultRemarkOffset</code> Ma-		<code>\graphicspath</code> Makro	35, 37
kro	32	<code>graphicx</code> Abhängigkeit	50
<code>\Description</code> Makro	33	H	
<code>dinat.bst</code> Datei	50	<code>hyperref</code> Abhängigkeit	50
<code>doc.sty</code> Datei	47	I	
<code>\documentclass</code> Makro	8	<code>ifpdf</code> Abhängigkeit	50
Dokumentenbeginn	16	<code>ifthen</code> Abhängigkeit	50
Dokumentenende	20	Index	21
<code>\DoublePicture</code> Makro		Besonderheit	27
.....	36, 37, 48, 49	Empfehlung	28
<code>\DoubleSubPicture</code> Makro	37	Reihenfolge	28
F			
<code>fancyhdr</code> Abhängigkeit	49	<code>index</code> Option	21, 27
<code>fomdes.sty</code> Datei	7, 20, 47, 48	Inhaltsverzeichnis	18
<code>fomdss.sty</code> Datei	7, 16, 48	<code>initexmf</code> Befehl	5
<code>fomfod.sty</code> Datei	7, 34	<code>inputenc</code> Abhängigkeit	50
<code>fomidx.ist</code> Datei	7, 28, 47	Installation	
		Kurzfassung	6

Verzeichnisses	4	\DefaultListingConfig	43
\Institute <i>Makro</i>	10	\DefaultRemarkIndent	32
\Item <i>Makro</i>	31	\DefaultRemarkOffset	32
\ItemIndent <i>Makro</i>	32	\Description	33
\ItemOffset <i>Makro</i>	32	\documentclass	8
		\DoublePicture
		36, 37, 48, 49
		\DoubleSubPicture	... 37
		\General 10, 13, 14
		\graphicspath	... 35, 37
		\Institute 10
		\Item 31
		\ItemIndent 32
		\ItemOffset 32
		\ListingFontSize 43
		\LockflagCompany	. 12, 13
		\LockflagContent 12
		\LockflagDate	... 12, 13
		\LockflagHeader 13
		\LockflagSigLeft 13
		\LockflagSigRight	... 13
		\Mark 18
		\PDFAuthor 11
		\PDFProducer 11
		\PDFSubject 11
		\PDFTitle 11
		\PrintAbbrev 18–20
		\PrintBibliography	.. 24
		\PrintContents 18
		\PrintFigures 18
		\PrintGlossary 22
		\PrintIndex	.. 26, 27, 47
		\PrintListings 22
		\PrintTables 18
		\Professor 10
		\Raise 44
		\Registered 46
		\Remark 31
		\RemarkIndent 31
		\RemarkOffset 32
		\section 48
		\SeeAlso 28, 29
		\Semester 10
L			
latex <i>Befehl</i>	11		
Linux	4, 47		
listingcolored <i>Option</i>	... 34		
\ListingFontSize <i>Makro</i>	.. 43		
listings <i>Abhängigkeit</i>	. 43, 44, 50		
listings <i>Option</i> 21, 22		
Literaturverzeichnis 21		
Achtung 50		
Empfehlung 25		
Hinweis 26		
lockflag <i>Option</i> 9		
\LockflagCompany <i>Makro</i>	12, 13		
\LockflagContent <i>Makro</i>	.. 12		
\LockflagDate <i>Makro</i>	. 12, 13		
\LockflagHeader <i>Makro</i>	... 13		
\LockflagSigLeft <i>Makro</i>	.. 13		
\LockflagSigRight <i>Makro</i>	. 13		
M			
makeindex <i>Befehl</i> 7		
makeidx <i>Abhängigkeit</i> 50		
makeindex <i>Befehl</i>		
.....	17, 20, 22, 28, 47		
Makros			
\Abbrev 18		
\AffidavitContent	... 13		
\AffidavitDate 14		
\AffidavitHeader 15		
\AffidavitSigLeft	... 15		
\AffidavitSigRight	.. 15		
\Axel 46		
\CodeListingFile 42		
\Copyright 46		
\Course 10		
\Date 10		
\DefaultItemIndent	.. 32		
\DefaultItemOffset	.. 32		

<code>\Remark Makro</code>	31	<code>textfit</code> <i>Abhängigkeit</i>	51
<code>\RemarkIndent Makro</code>	31	<code>textpos</code> <i>Abhängigkeit</i>	51
<code>\RemarkOffset Makro</code>	32	<code>\textwidth Makro</code>	35, 38
<code>rotating</code> <i>Abhängigkeit</i>	51	Titelblatt	8, 9
S			
Sachwortverzeichnis → <i>Glossar</i>		<code>\Title Makro</code>	10, 13
.	21	<code>toctotoc</code> <i>Option</i>	17
<code>\section Makro</code>	48	<code>\TodayShort Makro</code>	45, 46
<code>\SeeAlso Makro</code>	28, 29	<code>\Trademark Makro</code>	46
<code>\Semester Makro</code>	10	U	
<code>\SinglePicture Makro</code>	35, 36	<code>ulem</code> <i>Abhängigkeit</i>	51
Sperrvermerk	9	Umgebungen	
<i>Abhängigkeit</i>	13	<code>Citation</code>	30
<i>Signatur</i>	12	<code>CodeListing</code>	41, 42
<i>Text</i>	12	<code>PlainFigure</code>	38, 39
Stichwortverzeichnis → <i>Index</i>	21	<code>Table</code>	39, 41
<code>\Student Makro</code>	10	<code>tabularx</code>	40, 41
<code>subfig</code> <i>Abhängigkeit</i>	48, 51	<code>\usepackage Makro</code>	
<code>\Subject Makro</code>	10	16, 20, 30, 34, 44
T			
Tabelle → <i>Gleitobjekt</i>	39	V	
Tabellenverzeichnis	18	Versicherung (<i>eidesstattlich</i>)	9
<code>Table</code> <i>Umgebung</i>	39, 41	<i>Abhängigkeit</i>	14
<code>tabularx</code> <i>Abhängigkeit</i>	40, 51	<i>Signatur</i>	14
<code>tabularx</code> <i>Umgebung</i>	40, 41	<i>Text</i>	13
<code>texhash</code> <i>Befehl</i>	5, 6	W	
		Windows	4, 47