Grundlagen der Dokumentenverarbeitung

Reinhard Wilhelm
Reinhold Heckmann

Grundlagen der Dokumentenverarbeitung

 **ADDISON-WESLEY**

An imprint of Addison Wesley Longman, Inc.

Bonn • Reading, Massachusetts • Menlo Park, California • New York • Harlow, England
Don Mills, Ontario • Sydney • Mexico City • Madrid • Amsterdam

Die Deutsche Bibliothek – CIP-Einheitsaufnahme

Wilhelm, Reinhard:
Grundlagen der Dokumentenverarbeitung
/ Reinhard Wilhelm; Reinhold Heckmann. –
Bonn: Addison-Wesley-Longman, 1996
ISBN 3-89319-877-6
NE: Heckmann, Reinhold:

Lektorat: Fernando Pereira
Korrektorat: Friederike Daenecke, Reinhold Heckmann
Produktion: Claudia Lucht, Bonn
Satz: Reinhold Heckmann. Gesetzt in Palatino 10/12 Pkt.
Belichtung, Druck und Bindung: Bercker Graphischer Betrieb, Kevelaer
Umschlaggestaltung: Tandem Design, Berlin

Das verwendete Papier ist aus chlorfrei gebleichten Rohstoffen hergestellt und alterungsbeständig. Die Produktion erfolgt mit Hilfe umweltschonender Technologien und unter strengsten Auflagen in einem geschlossenen Wasserkreislauf unter Wiederverwendung unbedruckter, zurückgeführter Papiere.

Vorwort

Dokumentenverarbeitung, oder etwas profaner Textverarbeitung, ist bei weitem die häufigste Anwendung von PC's und Arbeitsplatzrechnern. Im kommerziellen Bereich wird der gesamte Briefverkehr mit Hilfe von Textverarbeitungssystemen abgewickelt. Personalisierte Massendrucksachen werden als Serienbriefe auf ihnen erstellt. Firmeninterne Notizen werden teilweise schon aus dem Textverarbeitungssystem heraus elektronisch an ihre Adressaten geschickt. Wissenschaftliches Publizieren stellt sich mehr und mehr auf rechnerunterstützte Arbeitsweise um. Die verwendeten Systeme übernehmen halbautomatisch die Erstellung von Bibliographien, indem sie auf bibliographische Datenbanken zugreifen. Sie produzieren Glossare, wandeln Querverweise auf Kapitel, Abschnitte oder Tabellen in Seitennummern um und halten sie bei Änderungen konsistent. Auch im persönlichen Bereich tritt der Wordprozessor immer mehr an die Stelle der Schreibmaschine und sogar der Handschrift. Man mag bedauern, daß man in ferner Zukunft nicht mehr die handschriftlichen Äußerungen unserer geistigen Größen bewundern können wird. Hier sei nur festgestellt, was offenkundig ist.

Angesichts dieser Beobachtung ist es überraschend, daß, anders als in England oder Frankreich, wenige Informatikfachbereiche in Deutschland eine Lehrveranstaltung zum Thema Dokumentenverarbeitung in ihrem Curriculum haben. Über die Gründe kann man spekulieren. Zum einem könnte es sein, daß das Gebiet als zu profan gilt. Wie kann ein Gebiet interessant sein, wenn jeder Laie eines der aus diesem Gebiet stammenden Systeme bedienen kann. Wäre dem so, so steckte dahinter natürlich der Irrglaube, daß eine Technik, die jeder bedienen kann, für den Fachmann uninteressant sein muß.

Zum zweiten könnte es sein, daß die Allgegenwärtigkeit von Textverarbeitungssystemen den Eindruck hervorruft, daß dieses Gebiet uninteressant wäre, weil es keine offenen Probleme mehr gibt; „es klappt ja offensichtlich alles". Dabei werden allerdings einige neuere Entwicklungen übersehen, wie etwa im Bereich strukturierter Dokumente. Da Informatiker diese teilweise verschlafen haben, haben anders vorgebildete Leute, etwa Juristen, diese Entwicklungen initiiert, was man den Ergebnissen dann auch ansieht.

Eine für die Zunft der Informatiker weniger abträgliche Interpretation ist die folgende: Die meisten Gebiete der praktischen Informatik beschäftigen sich mit einer relativ kleinen, abgeschlossenen Menge von Konzepten und das meist auf einer ziemlich homogenen theoretischen Grundlage. Der Übersetzerbauer beschäftigt sich mit Programmiersprachen, deren Semantik und Typsystemen und mit den Widrigkeiten der neuesten Errungenschaften der Rechnerarchitekten. Seine theoretischen Grundlagen stammen meist aus der Theorie der formalen Sprachen und der Automaten. Dazu braucht er noch einige Kenntnisse von Datenstrukturen und effizienten Algorithmen.

Der Betriebssystemspezialist kennt Konzepte wie Prozeß, Nebenläufigkeit, Kommunikation und Synchronisation, Virtualisierung von Betriebsmitteln, Scheduling usw. Seine theoretischen Grundlagen sind ein bißchen Warteschlangentheorie und Konzepte wie Lebendigkeit und Fairness.

Das Gebiet der Dokumentenverarbeitung ist dagegen ein Querschnittsgebiet. Es borgt seine Techniken und seine Grundlagen aus sehr vielen Gebieten der Informatik. Das muß aber kein Nachteil sein; wir meinen, es bietet Vorteile, wie sie kaum ein anderes Gebiet der praktischen Informatik anzubieten hat.

Im Gebiet Dokumentenverarbeitung lassen sich zumindest die folgenden Informatikinhalte vermitteln:

- *Datenstrukturen:*
  Man kann beschreiben, welche Datenstrukturen ein Texteditor zur Darstellung des Inhalts einer Textdatei verwenden sollte, damit er effizient einfügen und löschen kann.
- *Algorithmen:*
  Die meisten Editoren bieten eine assoziative Suche nach Textmustern an. Dafür gibt es gleichermaßen elegante wie effiziente Algorithmen. Ein anderes Beispiel ist das Problem des Zeilenumbruchs. Dieser kann mit dynamischer Programmierung oder mit einem Algorithmus zum Finden kürzester Wege durchgeführt werden.
- *Komplexität:*
  An einigen Problemen der Dokumentenverarbeitung, z. B. dem Seitenumbruch, kann man sehr schön zeigen, daß das Finden optimaler Lösungen schon unter halbwegs realitätsnahen Anforderungen NP-vollständig ist.
- *Formale Sprachen und Automaten:*
  Die Grundlagen der neueren, strukturierten Dokumentenmodelle sind Grammatiken und formale Sprachen. Zur Verarbeitung bieten sich die Automaten aus den korrespondierenden Automatenklassen an.
- *Übersetzerbau:*
  Bei der Dokumentenverarbeitung fallen Aufgaben wie die syntaktische Analyse, die Übersetzung in Zwischendarstellung und die Allokation von Ressourcen an, die man in Vorlesungen zum Übersetzerbau vermittelt.
- *Rechnen mit Termen:*
  Einige der Aufgaben bei der Verarbeitung strukturierter Dokumente lassen sich gut als Berechnungen auf Termen und als Transformation von Termen darstellen. Für die Beschreibung solcher Berechnungen bzw. Transformationen eignen sich funktionale Programmiersprachen besonders gut.
- *Abstrakte Maschinen, Interpreter:*
  Am Beispiel des *PostScript*-Interpreters lernt man die Funktionsweise einer in Software realisierten abstrakten Maschine, hier einer Kellermaschine kennen.

- *Gerätetechnik:*
  Das Gebiet hat eine faszinierende Entwicklung aufgrund der rasanten technologischen Fortschritte bei den Ein-/Ausgabegeräten gemacht. Die wesentlichen Eigenschaften dieser Technologien sollte man kennen. Vor allem aber muß man lernen, wie man Anwendungen geräteunabhängig macht.

- *Bedienoberflächen:*
  Textverarbeitung ist nur deshalb zu der populärsten Anwendung von PC's geworden, weil die entsprechenden Werkzeuge mit Bedienoberflächen versehen wurden, die sie Laien zugänglich gemacht haben. Die Prinzipien des Entwurfs von Bedienoberflächen lassen sich deshalb gut an diesen Anwendungen erklären.

- *Systemmodelle:*
  Hinter einer polierten Bedienoberfläche steckt ein vom Entwickler konzipiertes Systemmodell, etwa der Wordprozessor als simulierte Schreibmaschine. Ist dieses Systemmodell nicht konsequent durchgehalten, oder hat der Benutzer ein anderes Systemmodell im Kopf als der Entwickler, so wird sich der Benutzer mit dem System schwertun.

- *Sinn und Zweck von Standards:*
  Dokumente werden im allgemeinen nicht geschrieben und ausgedruckt, um anschließend im Archiv zu verschwinden. Meist sind sie dazu bestimmt, gelesen zu werden. Heutzutage werden viele Dokumente nicht mehr auf Papier, sondern elektronisch übermittelt. Da ist es unabdingbar, daß der Empfänger weiß, wie er ein Dokument sichtbar machen kann, etwa durch die Interpretation einer *PostScript*-Datei auf einem dieser Sprache mächtigen Drucker. Soll der Empfänger sogar an diesem Dokument mitarbeiten, so empfiehlt es sich, das Dokument in einer Form zu verschicken, die seinem Texteditor oder Wordprozessor zugänglich ist. Die Grundlagen dazu sind normierte Darstellungen.

- *Software-Engineering:*
  Viele Dokumente, z.B. dieses Buch, entstehen in einem langwierigen Prozeß, der sich über Jahre hin erstrecken kann. Das ist bei nichttrivialen Softwaresystemen nicht anders. Im Software-Engineering hat man schon lange begriffen, daß man beim Softwareentwurf die Weiterentwicklung des Systems bereits berücksichtigen muß. Genauso ist es beim „Schreiben" eines großen Dokuments.

  Ebenso gibt es bei vielen Dokumenten aufeinanderfolgende Auflagen oder verschiedensprachige Varianten. Dies verursacht Probleme der Versionen- und Variantenkontrolle, die aus dem Software-Engineering bekannt und teilweise gelöst sind. Die verschiedenen Versionen und Varianten bestehen aus gemeinsamen und nicht gemeinsamen Teilen, die jeweils für jede Ausgabe konsistent zusammengefügt werden müssen.

- *Abstraktion:*
  Eines der wichtigsten Prinzipien bei der Arbeit des Informatikers ist die Abstraktion. Ja, die Fähigkeit zur Abstraktion, der Betonung des Wesentlichen unter Vernachlässigung des Irrelevanten, wird sogar als ein Kriterium für Intelli-

genz angesehen. Abstraktion im Bereich der Dokumentenverarbeitung tritt besonders deutlich bei den modernen, strukturierten Ansätzen auf. Die Spezifikation der logischen Struktur läßt sich als die Spezifikation eines abstrakten Datentyps ansehen, die Angabe der physikalischen Struktur und der Layoutregeln als seine Implementierung. Das Erstellen eines Dokuments einer bestimmten Struktur entspricht dann der Kreation einer Instanz des Datentyps. Das klingt erst einmal nur wie eine Sichtweise; es ist aber unter den oben erwähnten Software-Engineering-Gesichtspunkten sehr wichtig, da dadurch der Entwicklungsprozeß und das Herstellen von verschiedenen Darstellungen unterstützt wird.

Deshalb denken wir, daß Lehrveranstaltungen im Bereich Dokumentenverarbeitung die Chance bieten, Studenten eine Menge wichtiger Informatikkonzepte zu vermitteln, da dieses Gebiet wie kaum ein anderes die Ergebnisse vieler anderer Gebiete der Informatik integriert. Wir haben in diesem Buch zweierlei versucht. Erstens haben wir die wichtigsten Grundlagen der Dokumentenverarbeitung dargestellt, da dieses Gebiet an sich hoch relevant ist. Zweitens möchten wir den oben geschilderten Anspruch teilweise realisieren. Dazu haben wir jeweils bei der Darstellung eines Konzepts der Dokumentenverarbeitung die Informatikinhalte vertiefend dargestellt. Im Inhaltsverzeichnis sind die Abschnittsüberschriften dieser Teile jeweils *kursiv* gesetzt.

Das Buch richtet sich an Studenten der Informatik und Studenten von Studiengängen, die sich im weiteren Sinne mit computerunterstütztem Publizieren befassen. Aus den oben genannten Gründen eignet es sich in besonderer Weise für Studenten der Informatik im Nebenfach, welche in *einer* Lehrveranstaltung bzw. beim Lesen *eines* Buches einige wichtige Konzepte, Methoden und Verfahren der Informatik kennenlernen wollen.

Wir möchten dem Verlag Addison-Wesley Deutschland und dort insbesondere den Herren Dr. Hundt und Pereira dafür danken, daß sie dieses etwas ungewöhnliche Buchprojekt mit uns realisiert haben, sowie Frau Daenecke für die Betreuung während der Fertigstellung des Buches. Herrn Stephan Diehl danken wir dafür, daß er das Manuskript gelesen und wertvolle Verbesserungsvorschläge gemacht hat, und Herrn Peter Bouillon für einige Tips im Umgang mit TEX.

Saarbrücken, im August 1996 *Reinhard Wilhelm, Reinhold Heckmann*