

Vom Papier- zum Online-Dokument

Dr. Andreas Baumert

Zusammenfassung

Auf welche Weise Online-Dokumentation erstellt werden kann, ist abhängig von Produkten und Märkten. In diesem Text wird ein Verfahren vorgestellt, mit dem für ein hoch spezialisiertes und umfangreiches Software-Produkt die Online-Dokumentation aus der papiernen Dokumentation halbautomatisch erzeugt wird. Das Ausgangsdokument wird mit PageMaker auf dem Macintosh erstellt, das Online-Dokument steht für VMS, Ultrix und AIX zur Verfügung.

Einleitung

In der Literatur und in Vorträgen über die Online-Dokumentation wird darauf hingewiesen, daß die papierne Dokumentation sich erheblich von der elektronischen unterscheiden müsse, die unterschiedlichen Medien auch unterschiedlicher Texte bedürften. Ein Beispiel gibt William Horton, der wohl bekannteste Autor zu diesem Thema:

Online documentation is not word-processing files "dumped" online.¹

Auch auf Fachkonferenzen in Deutschland wird der Unterschied zwischen der Papier- und der Online-Dokumentation betont, beispielsweise bei Bürdek und Schupbach:

Die on-line-Dokumentation muß innerhalb eines speziellen Computerprogramms ein eigenständiges Softwareprogramm darstellen, da sie auch nach anderen Primärfaktoren - z.B. Didaktik - aufgebaut werden muß.²

Diese Auffassung ist für jeden, der professionell mit Sprache arbeitet, zunächst einleuchtend. Auch wenn wir didaktische Aspekte völlig unberücksichtigt lassen, wissen wir doch, daß jeder Text dem Leser Orientierungsmöglichkeiten und Verweisstrukturen bietet, die sich an der Art der Darbietung orientieren. Man könnte diesen Gedanken zuspitzen und behaupten, daß Textdesign immer medienabhängig sein müsse.

Die pointierte Form „Textdesign ist medienabhängig“ steht in einem bizarren Kontrast zu sehr unterschiedlichen Bemühungen, Texte in Bausteinen vorzuhalten, diese zu standardisieren und den technischen Text ganz allgemein nicht nur von den Medien der Wiedergabe zu lösen. Vielmehr sollen Texte auch losgelöst werden von den Instrumenten der Erfassung und Verarbeitung, etwa Textverarbeitungs- oder DTP-Programmen. Die verschiedenen Normen der ISO, die meist unter dem Stichwort SGML diskutiert werden, lassen eine ganz andere Entwicklungsrichtung vermuten, aus der ich für die Software-Industrie die folgenden Forderungen für die Zukunft ablese:

- Texte müssen in Modulen verarbeitet werden können. Solche Module werden in unterschiedlichen Texten und mit nur geringfügiger Modifikation zu unterschiedlichen Zwecken miteinander kombiniert.
- Auch komplexe Anwendungen wie Hypertext müssen, so weit es eben möglich ist, auf solche Module zugreifen.
- Medienabhängige Komponenten von Texten müssen automatischen Selektionsverfahren zugänglich sein. Solche Komponenten sind beispielsweise Bildlegenden oder Seitenverweise.

Die Einschränkung auf die Software-Industrie ist aus mehreren Gründen geboten. Man kann vielleicht einen Idealfall annehmen, in dem

- die Grenzen zwischen Online-Dokumentation und Produkt verschwimmen,
- das Produkt einer umfangreichen Erklärung bedarf und
- häufig aktualisiert wird.

In diesem Idealfall wissen die Anwender um die Komplexität des Produktes und sind auch bereit, eine Dokumentation zu benutzen, um Funktionalitäten sachgerecht anwenden zu können. Diese Anwender sind auch darin geübt, sich mit Texten auseinanderzusetzen.

Für andere Szenarien, in denen andere Produkte eingesetzt werden, etwa eine Straßenbaumaschine, ein wehrtechnisches Gerät, eine Reinigungsmaschine, ein Kaffeeautomat usw. gelten völlig andere Voraussetzungen.

Wenn wir uns auf den genannten Idealfall einigen, sind dennoch weitere Einschränkungen nötig. Der Aufwand für eine Online-Dokumentation muß auch betriebswirtschaftlich gerechtfertigt sein. Die Forderungen der Theoretiker des elektronischen Publizierens technischer Dokumentation sind mitunter nur schwer mit den Gegebenheiten des Marktes in Einklang zu bringen. Das hier vorgestellte Verfahren wird für ein Software-Produkt hoher Komplexität eingesetzt, wobei dieses Produkt nur in einem sehr engen Marktsegment platziert werden kann.

Der Markt

Banken und Finanzdienstleister, die mit derivativen Instrumenten, etwa Optionen und Futures handeln.

Das Produkt

Ein Back office, basierend auf einer Ingres-Datenbank mit über 120 Tabellen und über 2000 Feldern. Das Handbuch für dieses Produkt besteht aus 6 Bänden.

Die Aufgabe

Vier Bände des Referenzhandbuches sollen als Online-Dokument zur Verfügung gestellt werden. Dieses Dokument soll außerdem kontextsensitiv vorgehalten werden (sogenannte ItemHelp).

Einschränkung

Gefordert wird, daß **nur der Text**, nicht aber Grafik in die Online-Dokumentation übernommen werden soll. Diese Anforderung wird gestellt, weil einige Anwender zeichenorientierte Ausgabegeräte (Terminals) nutzen, in denen Grafiken nicht abgebildet werden könnten. Die Übertragung von Grafiken ist mit dem hier geschilderten Verfahren nicht möglich.

Der Einsatz von Grafiken in der Online-Dokumentation wird zwar in der Theorie gefordert, ist aber in der Praxis schwer zu verwirklichen, wenn Software-Produkte für unterschiedliche Betriebssysteme, Systemoberflächen und Ausgabegeräte geschrieben werden.

Voraussetzungen

Das Handbuch

Die vorhandene Arbeitsumgebung für das Schreiben der Handbücher sind Macintosh-Rechner, das DTP-Programm PageMaker 4.1, die Grafikprogramme Superpaint 2.0 und FreeHand 2.0. Zu Beginn der Aufgabe steht fest, daß wechselnde Redakteure über einen Zeitraum von etwa einem Jahr an dem Projekt arbeiten werden.

Das Handbuch ist in sechs Teile gegliedert:

- | | |
|---|----------------------------|
| 1. Einführung | - Allgemeine Erläuterungen |
| 2. Tutorial | - handlungsorientiert |
| 3. Abfragebildschirme | - bloße Referenz |
| 4. Eingabebildschirme | - bloße Referenz |
| 5. Bildschirme zur Berichts-anforderung | - bloße Referenz |
| 6. Anhang | |

Als Online-Dokumentation werden nur die Teile 3 bis 5 benötigt werden. Diese Teile füllen vier Ringordner im Format DIN A4. Sie bestehen aus über 250 Kapiteln. Jedes dieser Kapitel beschreibt einen Bildschirm für die Datenabfrage oder -eingabe oder die Berichts-anforderung. Der Text soll später mit der Hilfetaste unter VMS, Ultrix oder AIX am Bildschirm angezeigt werden.

Für die Kapitel muß zunächst eine Struktur gefunden werden, die auf **alle** Kapitel in gleicher Weise angewandt werden kann. Diese Struktur ergibt sich fast zwangsläufig, wenn die Beschreibung nicht handlungsorientiert ist, sondern sich auf eine bloße Referenz beschränkt:

- Name des Bildschirms
- Abbildung des Bildschirms
- Aufgabenbeschreibung
- Beschreibung der Felder
- Beschreibung der Funktionstasten

Nachdem eine allgemeine Struktur für Referenzkapitel festgelegt ist, müssen Regeln etabliert werden, die für die Textformatierung gelten. Diese Regeln gelten für die Anwendung von Druckformatvorlagen.

Druckformatvorlagen

Druckformatvorlagen (DFV) sind das Schlüsselkonzept. Eine DFV enthält alle Formatierungsanweisungen für den Absatz, dem sie zugewiesen wird. Sie erhält einen Namen, der vom PageMaker mit einem ganzen Paket von Formatspezifikationen, etwa Schrifttyp, -grad, Absatzspezifikationen usw. verknüpft wird.

Jedem Absatz wird eine DFV zugewiesen. Ihre Anwendung ist verbindlich. Die DFV müssen von allen Redakteuren in der gleichen Weise benutzt werden, die Disziplin in ihrer Anwendung ist die Voraussetzung für den Erfolg.

Vorbereitungen auf dem Macintosh

Die Texte der Kapitel werden aus den PageMaker-Dokumenten in ASCII-Dateien exportiert. Dabei ist die Option „Formatmarken exportieren“ eingeschaltet. PageMaker stellt dann jedem Absatz den Namen der DFV in spitzen Klammern voran:

```
<Überschrift 12er>Vorbereitungen auf dem Macintosh
<Normal>Die Texte der Kapitel werden aus den PageMaker-Dokumenten
exportiert. Dabei ist die Option „Formatmarken exportieren“
eingeschaltet. PageMaker stellt dann jedem Absatz den Namen der
DFV in spitzen Klammern voran:
<Normal>Der Text soll anschließend unter MS-DOS weiterverarbeitet
werden. Das Betriebssystem des Macintosh stellt die nötigen Werkzeuge
zur Verfügung, um die Dateien nach MS-DOS zu übertragen.
```

Der Text soll anschließend unter MS-DOS weiterverarbeitet werden. Das Betriebssystem des Macintosh stellt die nötigen Werkzeuge zur Verfügung, um die Dateien nach MS-DOS zu übertragen.

Die Weiterverarbeitung muß nicht zwingend unter MS-DOS erfolgen. Für dieses Projekt wurde MS-DOS allerdings als Mittler zwischen den verschiedenen Betriebssystemen gewählt, weil in der Redaktion vorhandene Werkzeuge die Entwicklung von Programmen besonders erleichterten.

Die Online-Dokumentation

Unter MS-DOS ist dann jedes der ca. 250 Kapitel eine ASCII-Datei, die ausschließlich aus Fließtext und Formatmarken besteht. Etwa 50 unterschiedliche DFV sind in diesem Projekt genutzt worden.

Im nächsten Schritt muß ein Programm

- die Dateien einlesen,
- die Formatmarken erkennen und
- mit dem Text hinter einer Formatmarke in bestimmter Weise verfahren.

Nach einigen Versuchen konnte festgelegt werden, daß mit dem Fließtext aus dem Page-Maker-Dokument für dieses Projekt in 3 grob unterscheidbaren Arten zu verfahren ist:

Grob: Der Text wird

- für die Bildschirmanzeige neu umbrochen oder
- ersatzlos gestrichen (z.B. Bildlegenden) oder
- für eine fortlaufende Numerierung genutzt.

Fein: Bei einem zeichenorientierten Bildschirm, der nur sehr einfache Auszeichnungen zuläßt, sind 10 unterschiedliche Arten des Umbruchs mehr als genug. Als Formatierung bieten sich an:

Linker Rand
Rechter Rand
Abstand nach oben
Abstand nach unten
Unterstreichen (mit einem von drei Zeichen = - .)
Negativer Erstzeileneinzug

Diese Arten des Umbruchs nennen wir mit dem englischen Wort für Druckformatvorlage „Styles“, um sie von den Druckformatvorlagen des PageMaker unterscheidbar zu machen. Fügt man zu den 10 Styles noch die Styles „ersatzlos streichen“ und „numerieren“ hinzu, muß jede DFV des PageMaker in einen von 12 Styles übersetzt werden. Für diese Übersetzung wird eine Tabelle angefertigt, die als Datei gespeichert wird:

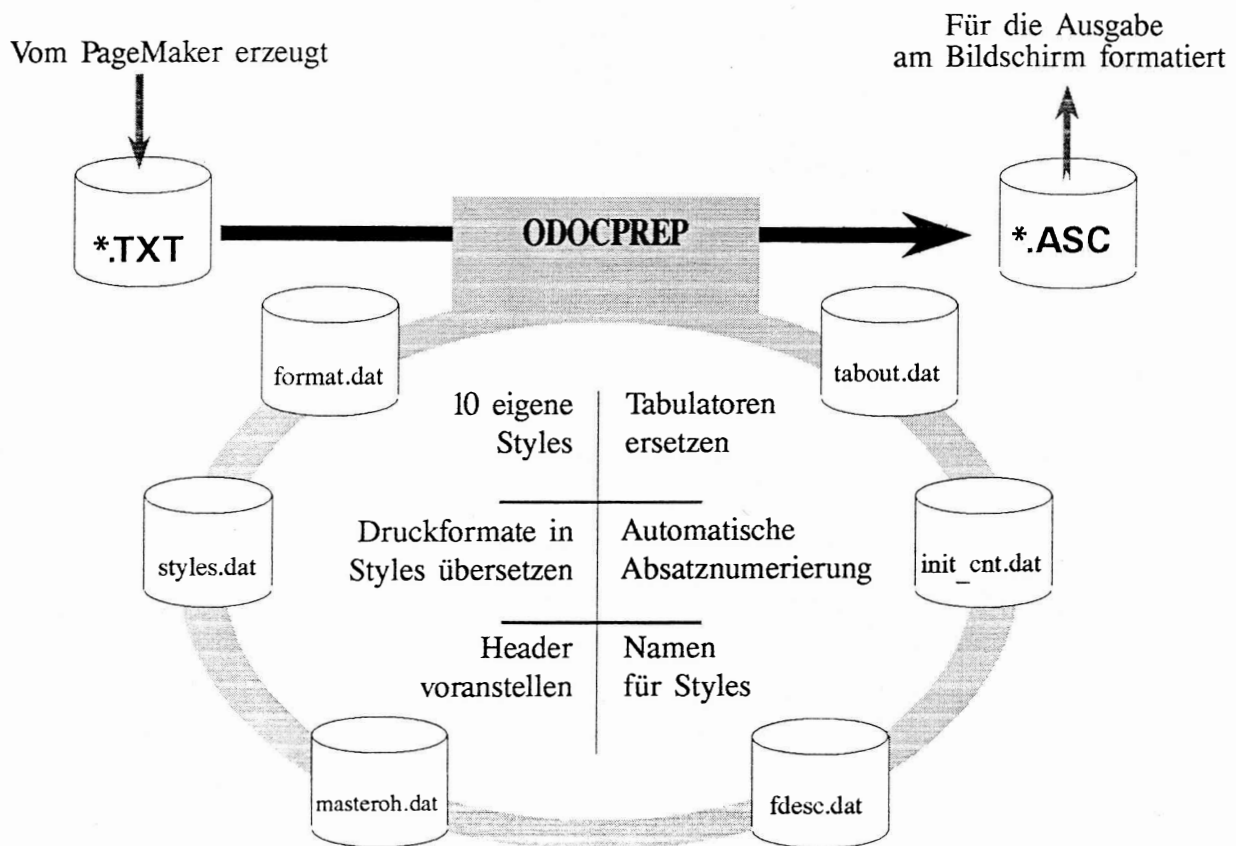
<Überschrift 18er>00	
<Legende>12	
<Description Header>01	
<Description>04	
<Data type>12	
<Field length>12	
<Table>12	
<Computed>12	
<Hinweis>05	
<Normal>04	
<Warnung>09	...

Auf der linken Seite steht jeweils der Name der DFV, rechts davon die Nummer des Styles. Den DFV „Legende“, „Data type“, „Field length“, „Table“ und „Computed“ wird Style-Nr. 12 zugewiesen: Ersatzlos streichen. Die DFV „Description“ und „Normal“ werden in diesem Beispiel für die Bildschirmausgabe gleich umbrochen.

Diese Datei (styles.dat) ist eine von sechsen, in denen Informationen für die Konvertierung der ASCII-Dateien in solche der Online-Dokumentation vorgehalten werden:

styles.dat	Tabelle DFV - Style
masteroh.dat	Kopf, der jeder Datei der Online-Dokumentation vorangestellt wird.
format.dat	Formatdefinitionen der Styles
fdesc.dat	Namen der Styles
init_cnt.dat	Initialisiert den Absatzzähler
tabout.dat	Anzahl der Leerzeichen, durch die ein Tabulator ersetzt wird.

Weil die Wartung aller Informationen, die für die Konvertierung und das Erstellen der Online-Dokumentation benötigt werden, recht aufwendig ist, wurde ein Programm geschrieben, das die Pflege der meisten dieser Dateien durch Eingabebildschirme erlaubt. In einer neueren Version, die nun auch für andere Zwecke eingesetzt wird, heißt dieses Programm ODOCPREP.



Odocprep ist in der Programmiersprache C geschrieben, nutzt die einfache Ein-/Ausgabe (read()/write()), Größe des Lesespeichers: 10KByte) für die Dateikonvertierung und ist dadurch außerordentlich schnell. Natürlich kann man ein solches Programm auch in den anderen dafür geeigneten Sprachen schreiben. Es hat sich gezeigt, daß eine menügesteuerte Benutzerführung und Eingabemasken die Arbeit erheblich erleichtern.

Der Quellcode ist in dreizehn Module gegliedert, von denen eines für die Pflege der Styles (styleset.c) und ein anderes für die Konvertierung selbst zuständig ist (onlindoc.c). Die anderen Module regeln Bildschirmausgabe, Steuerung, Hilfefunktionen, Systemmeldungen, Menüaufbau usw. An dieser Stelle reicht es aus, auf die Funktionsweise von onlindoc.c kurz einzugehen.

Parameter

Im ersten Schritt werden alle Parameter geladen, die das Programm benötigen wird: format.dat, init_zlr.dat, tabout.dat, styles.dat.

Dateinamen

Der Redakteur gibt den Namen der vom PageMaker exportierten Datei ein.

Dateien öffnen

Diese Datei und die Ausgabedatei werden geöffnet. Beide unterscheiden sich durch die Extensionen, die „fest verdrahtet“ sind, um die Möglichkeit von Eingabefehlern zu reduzieren und Schreibarbeit zu sparen.

Header schreiben

In die Ausgabedatei wird der Text der Datei masteroh.dat geschrieben. Damit hat jede Datei der Online-Dokumentation den gleichen „Vorspann“. Für den Anwender müssen darin wichtige Informationen untergebracht werden, mindestens die Release-Nummer. Sollte eine Installation nicht korrekt sein, kann der Anwender an der Release-Nummer und dem Datum sofort erkennen, ob er das zu seiner Installation gehörige Dokument liest. Im Anschluß an den Text der Datei masteroh.dat wird immer ein Zeitstempel (Datum und Uhrzeit) hinzugefügt.

Konvertieren

Die vom PageMaker exportierte Datei wird Absatz für Absatz bearbeitet:

Feststellen, welchem Style die am Beginn des Absatzes stehende Formatmarke entspricht. Ist es Style 12 (ersatzlos streichen), wird der nächste Absatz geladen und nichts gespeichert.

Für die Styles 1 bis 10 wird der Text formatiert (Style 11 „numerieren“ ist ein Sonderfall und kann ausgespart werden). Formatieren heißt in diesem Fall:

- entfernen der Formatmarke,
- umbrechen,
- wenn nötig, unterstreichen.

Der umbrochene Text wird gespeichert.

Redigieren

Die Ausgabedatei wird sofort danach in einen Editor geladen, damit von Hand redigiert werden kann. Bei mehreren Hundert Dateien würde schnell eine Übersehen, wäre nicht der Zwang zur Nachbearbeitung ein fester Bestandteil des Programms. In unserem Projekt waren nur wenig Korrekturen nötig. Schwachstellen sind gelegentlich tabellarische Darstellungen, die Anwendung einer falschen DFV im Ausgangstext und Verweise. Die letztgenannten waren oft mit einem eigenen Format versehen worden und konnten automatisch ausgesondert werden.

Kontextsensitive Hilfe

Auch die sogenannte ItemHelp konnte mit etwas Programmieraufwand erzeugt werden. Die Spezifika der Ingres-Datenbank, auf die zur Lösung dieser Aufgabe zurückgegriffen wird, können übergangen werden. Entscheidend sind die Voraussetzungen, die so oder ähnlich auch in anderen Datenbanksystemen vorliegen dürften:

Der Anwender drückt die Taste ItemHelp und löst einen Prozeß aus, der auf folgende Voraussetzungen trifft:

Das **Datenbanksystem** „kennt“

- das Eingabefeld, in dem der Cursor gerade steht,
- die Beschriftung dieses Eingabefeldes im Eingabebildschirm (z.B. „Konto-Nr.“),
- den internen Namen des Eingabebildschirms.

Mit dieser Relation wird die Aufgabe gelöst. Man benötigt eine Tabelle, in der für jedes Eingabefeld ein Datensatz angelegt wird, der diese Informationen und eine weitere vorhält: den Namen der Datei, die in den Bildschirm geladen wird, wenn ein Anwender die Taste ItemHelp drückt.

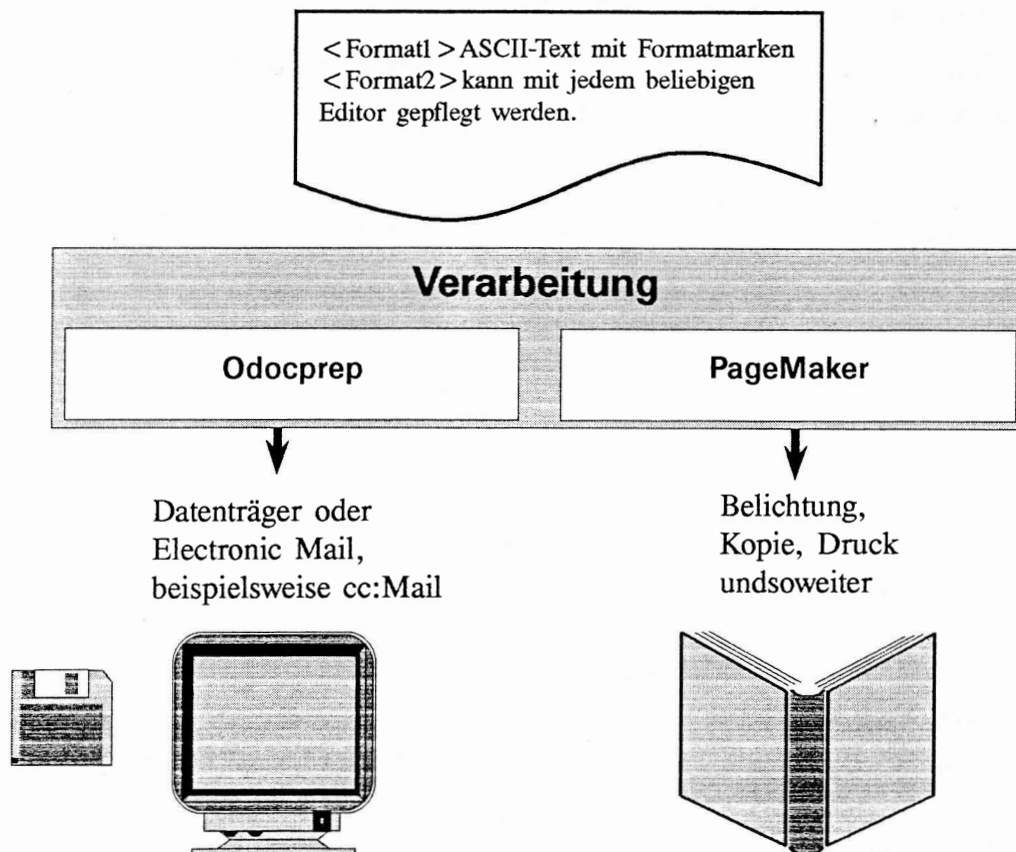
Auf der Ebene der **Online-Dokumentation** liegt für jeden Eingabebildschirm eine Datei vor, die ebenfalls den Namen der Eingabefelder enthält. Wenn „Konto-Nr.“ die Beschriftung eines Eingabefeldes auf dem Bildschirm ist, ist das natürlich auch eine Überschrift im PageMaker- wie auch im Online-Dokument.

Die Lösung kann über diese Voraussetzungen gefunden werden. Sie besteht in einer Reihe von Programmen, die die Dateien der Online-Dokumentation zerlegen und die Relationen zwischen den zerlegten Dateien und den Eingabefeldern der Bildschirme in Datensätzen erfassen. Mit einem SQL-Befehl werden diese Datensätze dann in die dafür zuständige Tabelle der Ingres-Datenbank geladen. Die ItemHelp ist vollständig!

Ergebnisse

Mit diesem Verfahren lassen sich Einsatzmöglichkeiten finden, die über das Erstellen einer Online-Dokumentation hinausgehen. ASCII-Texte, die durch vorangestellte Formatmarken ausgezeichnet sind, können beispielsweise von Odocprep bearbeitet und dann innerbetrieblich über electronic Mail ausgetauscht werden. Erst die endgültige Version wird schließlich vom PageMaker oder einem anderen DTP-Programm, das ähnliche Leistungsmerkmale zeigt, verarbeitet. Erste Versuche mit einem Pflichtenheft, das solcherart gepflegt und anderen Abteilungen zur Kenntnis gebracht wird, verlaufen positiv.

Der mit Formatmarken versehene ASCII-Text ist ein „Zwischenformat“, das einfache elektronische Ausgabemedien mit Printmedien verknüpft.



Dieses Verfahren bietet eine sehr einfache Möglichkeit, Texte medienunabhängig zu bearbeiten und vorzuhalten. Es ist noch weit von dem allgemein anerkannten Verfahren SGML entfernt, verbaut aber auch keine künftige Textkonvertierung in diese Richtung.

Negative Aspekte dieser Anwendung sollen nicht verschwiegen werden:

Kontextsensitive Hilfe läßt sich so nicht 100%ig realisieren. Das Verfahren versagt zum Beispiel, wenn die Überschrift im PageMaker-Dokument vom Text im Eingabebildschirm abweicht. So etwas kann sachlich geboten sein, es kann eine kurzfristige Änderung im Programm oder auch ein Fehler des Redakteurs sein.

Ein automatisches Verfahren ist fehleranfällig.

Die Qualitätssicherung ist aufwendig, weil Fehler entstehen können, die Redakteuren nicht unterlaufen würden.

Trotz dieser Mängel ist das Ergebnis positiv. Wir konnten feststellen, daß unsere Online-Dokumentation

- vom Anwender akzeptiert wird,
- leicht zu aktualisieren ist und
- problemlos zwischen Betriebssystemen portiert werden kann.

Mit diesem Verfahren konnten etwa 2300 Dateien zur Verfügung gestellt werden, die Informationen für den Benutzer vorhalten und auf Tastendruck in den Bildschirm geladen werden.

Entgegen den Bedenken der Theorie ist das Resultat sehr ermutigend.

Anmerkungen

- 1 Horton, William K.: Designing & Writing Online Documentation. Help Files to Hypertext. New York [u.a.] 1989, S.10
- 2 Bürdek, Bernhard E.; Schupbach, Stephan: Das Human Interface Design der On-line Dokumentation. In: Arbeitskreis Online-Dokumentation innerhalb der tekom. Online-Dokumentation, Manuskripte der gleichn. Fachtagung 24.,25. 11. 1992, S.10

Frankfurt am Main, 22. Januar 1994