AUSZEICHNUNGSSPRACHEN

VON DER TEXTVERARBEITUNG ZUM „CONTENT MANAGEMENT“

Dietmar Rösner

Textverarbeitung ist aus den meisten Büros nicht mehr wegzudenken. Es ist damit sehr einfach geworden, Textdokumente zu erstellen, zu ändern, hübsch zu formatieren, in Druckqualität auszugeben und in Papierform oder elektronisch zu verteilen. Wer intensiv mit Texten arbeiten muss, wird aber auch schon auf Probleme gestoßen sein:

– Es ist schwierig, Texte auszutauschen und wiederzuverwenden, wenn sie auf unterschiedlichen Systemen erstellt wurden.

– Es ist mühsam, als Text erstellte Informationen für unterschiedliche Präsentationsmedien zu nutzen, sie z. B. nicht nur auf Papier drucken zu können, sondern sie auch sofort als WWW-Seite oder für ein WAP-Handy zu Verfügung zu haben.

– Es ist aufwendig, in großen Dokumentbeständen und auch in großen Dokumenten gezielt nach Informationen zu suchen.

Auszeichnungssprachen haben das Potential, diese und andere Probleme der Dokumentverarbeitung lösen zu helfen. In diesem Beitrag werden zunächst die Grundideen von Auszeichnungssprachen – insbesondere von XML, der Extensible Markup Language – vorgestellt. Dann wird über eigene Arbeiten im Rahmen eines EU-Projektes berichtet, bei denen es darum ging, eine Methodik für den Autorenprozess zu entwickeln, die es – zusammen mit unterstützenden Software-Werkzeugen – ermöglichen soll, das Potenzial von Auszeichnungssprachen wirklich nutzbar zu machen, um die geschilderten Probleme beim Umgang mit Dokumenten zu lösen.

PROBLEME MIT TRADITIONELLER TEXTVERARBEITUNG

Jeder, der ein Textverarbeitungssystem verwendet, dürfte in der einen oder anderen Form schon auf eines der typischen mit der bisherigen Art von Dokumentverarbeitung verbundenen Probleme gestoßen sein.

Textverarbeitungssysteme verwenden meist proprietäre Formate. Will man ein Dokument oder Teile davon in einem anderen System wiederverwenden oder weiterverarbeiten, so geht dies oft nur mit erheblichem Informationsverlust oder großem manuellen Aufwand.

Systeme zur Textverarbeitung bieten meist nur die Möglichkeit, direkt das Layout eines Dokuments zu beeinflussen. Typische Mittel, die das Layout zur Verfügung stellt, um Intentionen des Autors zu verdeutlichen, sind etwa unterschiedliche Zeichensätze, die Möglichkeit zu Unterstreichungen, zur Hervorhebung durch Umrahmung usw. Oft wird nun mit ein und demselben Element des Layouts auch in ein und demselben Dokument durchaus Unterschiedliches ausgedrückt. So könnte sich ein Autor dafür entscheiden, Hervorhebungen durch kursive Schrift kenntlich zu machen, gleichzeitig aber auch kursive Schrift zu nutzen, um Wörter mit einer bestimmten Bedeutung (z. B. Namen von Medikamenten in einem medizinischen Text) vom umgebenden Text abzuheben.

Menschliche Leser haben meist wenig Schwierigkeiten, die unterschiedlichen Funktionen dieser Layout-Elemente problemlos zu erfassen. Schließlich können sie beim Lesen ihr gesamtes Hintergrundwissen einbringen und tun dies auch meist ohne bewusste Anstrengung. Computerprogramme haben diese Fähigkeiten derzeit nicht oder nur in einem sehr eingeschränkten Maße. Stoßen sie auf eine Layout-Markierung, so haben sie nur eine geringe Chance, die jeweilige textuelle oder kommunikative Funktion zweifelsfrei zu klassifizieren. Dass Layout-Angaben nicht zweifelsfrei erkennen lassen, welche kommunikativen Absichten zum jeweiligen Layout-Element geführt haben, trifft dann auch den Autor. Entscheidet er sich nach einiger Zeit des Arbeitens an einem Manuskript, seine ursprüngliche Layout-Entscheidung zu revidieren und nur noch Hervorhebungen kursiv gesetzt haben zu wollen, aber die Wörter einer bestimmten Bedeutung durch ein anderes Layout-Mittel (z. B. Fettdruck) kenntlich zu machen, so hat er keine Möglichkeit einer automatischen Ersetzung, sondern muss durch alle Vorkommen dieser Layout-Markierungen durchgehen, um die jeweilige Änderungsentscheidung selber zu treffen.

GRUNDIDEEN VON AUSZEICHNUNGSSPRACHEN

Das Layout hat die Funktion, dem Leser die inhaltliche und logische Struktur eines Dokuments leichter zugänglich zu machen. Was der Autor beabsichtigte und was der menschliche Leser mehr oder weniger leicht erschließen kann,

HTML
Hypertext Markup Language; die Auszeichnungssprache, mit der derzeit die meisten Seiten im Internet gestaltet werden; HTML ist eine SGML-Anwendung

Metasprache
Sprache, mit deren Hilfe andere Sprachen definiert werden

Ontologie
in der Wissensrepräsentation: die grundlegenden Sprachmittel für die Modellierung eines Sachgebiets (im Unterschied zur Bedeutung in Philosophie und Geisteswissenschaften)

SGML
Standard Generalized Markup Language; international normierte Auszeichnungssprache

Taxonomie
Begriffshierarchie; meist organisiert nach der Beziehung allgemeiner Begriff – spezieller Begriff

XML
Extensible Markup Language; von einer Arbeitsgruppe des W3C definierte Auszeichnungssprache

W3C
World Wide Web Consortium; ein Zusammenschluss von Firmenvertretern und Wissenschaftlern, die Vorschläge zur Weiterentwicklung des Internet erarbeiten

bleibt aber bei einer rein Layout-orientierten Textrepräsentation – wie in bisherigen Systemen zur Textverarbeitung – den meisten Computerprogrammen unzugänglich.

Die Grundidee von Auszeichnungssprachen ist, die inhaltlichen und logischen Strukturelemente explizit in der Textrepräsentation kenntlich zu machen. Technisch geschieht dies dadurch, dass der natürlichsprachliche Text eines Dokuments, der in ASCII-Zeichen vorliegt, ergänzt wird um so genannte Tags, d. h. Markierungen.

Tags treten dabei paarweise auf. Das Start-Tag markiert den Beginn des Textbereichs, dessen Funktion durch den Namen des Tags ausgezeichnet wird, das zugehörige Ende-Tag markiert entsprechend die Stelle im Fließtext, an welcher der bezeichnete Abschnitt endet.

Vielen ist die Syntax solcher Auszeichnungen aus ihrer Erfahrung mit HTML geläufig (zu beachten ist allerdings, dass HTML weniger strikt ist; so wird etwa beim P-Tag auf die Forderung nach einem schließenden Tag verzichtet, ebenso beim LI-Tag innerhalb von Listenstrukturen).

Innerhalb des durch ein Paar aus einem Start-Tag und einem korrespondierenden Ende-Tag markierten Textbereichs kann es eingebettet weitere markierte Bereiche geben.

<BEISPIEL> **Dies** <TAG1> **ist** <TAG2> **kein** </TAG1> **wohlgeformtes Dokument** </TAG2> </BEISPIEL>

Dies ist kein wohlgeformtes Dokument

Wohlgeformtheit erlaubt keine Überlappungen von Elementen, sondern nur Enthaltensein

Eine wichtige Forderung ist die nach der so genannten Wohlgeformtheit. Sie lässt sich so formulieren: Jeder innerhalb eines Paars korrespondierender Tags mit einem Start-Tag begonnene, eingebettete Textabschnitt muss innerhalb des umfassenden Textabschnittes enden, mit anderen Worten, sich überschneidende Textelemente sind nicht zulässig. Diese Art der „wohlgeformten" Strukturierung führt zu einer hierarchischen Zerlegung des Textes in Abschnitte mit unterschiedlicher Funktion.

Beispiel eines wohlgeformten Dokuments. (Anmerkung: Mit TAG2 könnten hier Nominalphrasen, mit TAG1 Verbalphrasen markiert sein.)

<BEISPIEL>
<TAG2> **Diese Art der Einbettung** </TAG2>
<TAG1> **ergibt** <TAG2> **ein wohlgeformtes Dokument** </TAG2>
</TAG1> </BEISPIEL>

Diese Art der Einbettung ergibt ein wohlgeformtes Dokument

Einen Fließtext, der mit Tags angereichert wurde, die der Wohlgeformtheitsbedingung genügen, bezeichnet man auch als Textinstanz. Den Bereich zwischen zwei korrespondierenden Tags, inklusive dieser Tags, nennt man auch ein Element. Je nach der Art des Inhalts, d. h. der Art dessen, was zwischen dem Paar korrespondierender Tags steht, unterscheidet man zwischen drei Arten von Elementen: solche mit reinem Textinhalt, solche bei denen auf der nächsten Einbettungsstufe wiederum nur (durch Paare korrespondierender Tags gekennzeichnete) Elemente stehen und solche mit so genanntem gemischten Inhalt, bei denen Fließtext und Elemente sich abwechseln.

Auszeichnungssprachen unterstützen die Abstraktion, da sie erlauben, Klassen von Dokumenten zu beschreiben. Zwar gibt es Textinstanzen, die einmalig sind (z. B. literarische Werke oder historische Dokumente), in der Regel gehören Textinstanzen aber einer Klasse von Dokumenten mit verwandter oder identischer logischer Struktur an. Wir alle kennen eine große Zahl solcher Klassen. Einige Beispiele sind Geschäftsbriefe, wissenschaftliche Publikationen, Kolloquiumseinladungen, Jahresberichte, Bilanzen.

Empfänger
Betreff
Anrede
Inhalt
Grußformel
Anlagen

Beispiel: Elemente der Makrostruktur von Geschäftsbriefen

Die erste international genormte Auszeichnungssprache für Dokumente war SGML, die Standard Generalized Markup Language. Diese Sprache ist sowohl eine internationale Norm (ISO) als auch durch europäische und deutsche Normungsgremien bestätigt (EN, DIN).

SGML bietet unter anderem Sprachmittel, um Dokumentklassen mit sog. Dokumenttyp-definitionen (DTD) zu definieren. Es kann z. B. ausgedrückt werden, welche Elemente sich aus welchen anderen Elementen zusammensetzen (so kann sich ein Buch aus einer Folge von Kapiteln zusammensetzen). Manche Elemente müssen vorhanden sein, andere sind fakultativ (ein Buch kann ein Vorwort vor den eigentlichen Kapiteln haben und ein Glossar am Ende – beides ist aber fakultativ). Weitere Sprachmittel: Elemente können alternativ sein, Elemente müssen auftreten, aber die Abfolge ist beliebig, Elemente müssen in einer bestimmten Mindestzahl auftreten usw.

Obwohl SGML schon seit vielen Jahren verfügbar ist, hat es sich nur langsam verbreitet und hat hauptsächlich Anwendungen im Verlagswesen und in der technischen Dokumentation, etwa in der Luftfahrtindustrie.

DIE ENTWICKLUNG VON XML

Auf der Basis von SGML ist HTML, die Hypertext Markup Language, entwickelt worden. Diese Auszeichnungssprache hat ohne Zweifel sehr wesentlich dazu beigetragen, dass sich das World Wide Web zu einem alltäglichem Medium entwickeln konnte. Die Sprache ist schnell zu lernen und es gibt unterstützende Werkzeuge (z. B. Netscape-Composer) mit denen es sehr einfach ist, HTML-basierte Internetdokumente zu gestalten.

Für die einfache Gestaltung von Internetseiten mag HTML ausreichen, für eine inhaltsbezogene automatische Verarbeitung und Verwaltung von (Internet-)Dokumenten, das so genannte „content management", ist diese Sprache aber nicht flexibel genug. Vor diesem Hintergrund hat sich eine Arbeitsgruppe des World Wide Web Consortium (W3C) im Jahre 1996 Gedanken darüber gemacht, wie eine zukunftsfähige Sprache für das Web aussehen müsse. Diese Arbeitsgruppe hat insbesondere die Erfahrungen mit SGML einerseits und HTML andererseits ausgewertet und daraus die Entwurfsziele für die neu zu definierende Sprache abgeleitet. Arbeitstitel waren zunächst Bezeichnungen, wie „SGML for the Web" oder auch „SGML light", bevor dann der auch noch heute gültige Name gefunden wurde: XML für „Extensible Markup Language" oder „erweiterbare Auszeichnungssprache" /3/.
Was waren nun die Lehren aus den vorangegangenen Auszeichnungssprachen?

Schwächen von HTML:
– Auszeichnungen sind Layout-orientiert
– Benutzer kann keine anwendungsspezifischen Tags definieren

Warum HTML für das „content management" nicht ausreicht.

HTML ist einfach, beschränkt sich aber weitgehend auf Layout-bezogene Sprachmittel und ist für den Benutzer nicht erweiterbar. SGML ist zwar erweiterbar und erlaubt benutzerdefinierte Auszeichnungen, der SGML-Standard ist aber aufgrund seiner vielen zusätzlichen Möglichkeiten so voluminös geraten, dass es nur wenige mit dem Gesamtumfang der Norm übereinstimmende Implementationen gibt. So enthält der SGML-Standard Möglichkeiten, wie etwa die Optionen zur so genannten Minimierung von Tags, die noch aus einer Zeit knappen Speicherplatzes stammen und die die Verarbeitung von SGML-Dokumenten unnötig kompliziert machen. Der von der W3C-Arbeitsgruppe vorgeschlagene Sprachentwurf für XML war so gestaltet, dass XML-Dokumente eine Untermenge der SGML-Dokumente sind, aber auf viel Ballast aus der SGML-Definition verzichtet werden konnte.

XML wurde kurz nach der Veröffentlichung schon von vielen Initiativen unterschiedlichster Art aufgegriffen und stellt seither die Basis für viele Projekte und Weiterentwicklungen dar.

XML ist als Teilmenge von SGML ebenfalls eine so genannte Metasprache, mit der anwendungsspezifische Auszeichnungssprachen definiert werden können. Dies ist auch in der Folge geschehen. So gibt es etwa Math-ML für mathematische Texte, EML (Environmental Markup Language) für Dokumente zu Umweltfragen, EBML für elektronische Geschäftsabwicklung.

<APPEAL> Aus diesem Grunde sollte jeder – <TARGETGROUP> spaetestens ab dem Alter von 40 Jahren aufwaerts </TARGETGROUP> – seine Haut regelmaessig auf Veraenderungen hin untersuchen. <BODYPART> Die dem Licht ausgesetzten Teile </BODYPART> sind dabei besonders wichtig. </APPEAL>

Mit Auszeichnungen kann die Semantik von Textabschnitten explizit und für Computer verarbeitbar gemacht werden.

SELBSTBESCHREIBENDE DATEN

XML ist weit mehr als ein Mittel zur Dokumentrepräsentation. Es ist ein allgemeines Format für strukturierte Informationen und damit auch auf „klassische" Daten anwendbar.

Nehmen wir ein einfaches Beispiel: Wenn wir eine Ziffernfolge wie etwa 24 12 03 lesen, werden wir sie vielleicht als Datum interpretieren. Wenn wir keine Verabredung über die Bedeutung der einzelnen Bestandteile dieser Folge haben, könnten wir sie anders interpretieren als sie von demjenigen gedacht war, der sie niedergeschrieben hat. Genau eine solche Situation liegt im Internet häufig vor. Wir müssten stets wissen, wie die Information tatsächlich codiert ist. Sogenannte selbstbeschreibende Daten tragen die Angaben zur Interpretation mit sich, im Falle von XML in Form von Tags. In unserem Beispiel könnte dies dann so aussehen:
<d>24</d> <m>12</m> <y>03</y>

Unter der naheliegenden Interpretation, dass **d** für Tag, **m** für Monat und **y** für Jahr steht, handelt es sich in diesem Fall um ein Datum aus dem Jahre 2003. Die Bedeutung hätte aber auch eine andere sein können:
<y>24</y> <m>12</m> <d>03</d>

In beiden Fällen handelt es sich um zusammengehörige Datenelemente. Es ist daher sinnvoll, diese in einem Element Datum mit dem Tag <date> zusammenzufassen, also z. B.:
<date><d>24</d> <m>12</m> <y>03</y></date>

In welcher Form ein auf diese Weise eindeutig codiertes Datum dann dem Benutzer präsentiert wird, ist eine völlig separate Entscheidung und bei der Verarbeitung von XML – z. B. mit so genannten Stylesheets – ein völlig separater

Schritt. Einige mögliche Varianten:
24.12.03,
24. Dezember 2003,
2003-12-24
u. v. a. m.

FALLSTUDIE: XML-BASIERTE AUTORENUNTERSTÜTZUNG FÜR DIE ERSTELLUNG UND DAS MANAGEMENT VON MULTILINGUALEN INFORMATIONSRESSOURCEN

CATCH-II, das Akronym steht für „Citizen Advisory System based on Telematics for Communication and Health", war ein europäisches Projekt, das zum so genannten Telematics Application Programme gehörte. Es wurde von einem europäischen Konsortium in den Jahren 1998 bis 2000 unter Federführung der Arbeitsgruppe „Wissensbasierte Systeme und Dokumentverarbeitung" (Institut für Wissens- und Sprachverarbeitung, Fakultät für Informatik, Otto-von-Guericke-Universität Magdeburg) durchgeführt.[1)] Sein Ziel war, eine Methodik und ein Rahmensystem für telematik-basierte Gesundheitsinformationssysteme zu entwickeln.

Die Projektkonstellation verlangte und erlaubte einen grundlegend neuen Ansatz für die Unterstützung des Lebenszyklus von Dokumenten. Am Projekt arbeiteten Experten aus der Medizin verschiedener europäischer Länder mit, die als Autoren informative Texte über Risikofaktoren, Vorbeugungsmaßnahmen, Empfehlungen usw. für die flexible Nutzung in verschiedenen Formen von Informationssystemen konzipierten. Als exemplarische Gegenstandsbereiche wurden die Themen „Hautkrebs" und „Kardiovaskuläre Krankheiten" gewählt. Diese Auswahl erfolgte auch wegen der besonderen Relevanz von Präventionsmaßnahmen in diesen Bereichen. Die Autoren sollten entlastet werden von Fragen der Layout-Gestaltung und Problemen mit unterschiedlichen Zielformaten und sollten sich auf die Inhalte und auf die logische Strukturierung konzentrieren können. Um dies zu ermöglichen, wurden im CATCH-II-Projekt mit Hilfe von XML die Bereiche Inhaltserstellung und Inhaltsspeicherung vollständig getrennt von allen Aspekten der Präsentation und des Layouts.

Ein zentraler Punkt: Autorenunterstützung

Dies sieht im Einzelnen folgendermaßen aus: Die Autoren erstellen ihre Texte und werden gebeten, sie zum einen mit Meta-Daten /1/ zu ergänzen, zum anderen die Textstrukturen durch XML-Markup zu charakterisieren und wichtige Terme in den Texten durch sogenannte XML-Inline-Tags zu kennzeichnen /2, 3/. Um diesen Vorgang so komfortabel wie möglich zu gestalten, ist CEdit implementiert worden. Dieses Werkzeug erlaubt das bequeme Erfassen von Meta-Daten und die komfortable Eingabe von XML-Tags. CEdit stellt die zentrale Komponente der Autorenumgebung des CATCH-II-Systems dar.

Die von den Autoren ausgezeichneten Beiträge werden dann im zentralen CATCH-II Repository (auch CATCH-II-Informationspool genannt) abgespeichert. Aus dieser Datenbasis mit XML-getagten Informationsobjekten werden unterschiedliche Versionen des CATCH-II-Systems abgeleitet und auch aktualisiert.

DER NUTZEN VON AUSZEICHNUNGEN UND META-DATEN

Wenn Dokumente mit Meta-Daten und logischem Markup annotiert werden sollen, dann ist dafür zunächst zusätzlicher Zeit- und Arbeitsaufwand erforderlich. Auf der anderen Seite können Meta-Daten und logische und semantische Auszeichnungen eine zentrale Rolle für das Management, die Aktualisierung, die Wiederverwendbarkeit und die Nutzung von Informationsressourcen spielen. Einige Beispiele aus CATCH-II sollen den zusätzlichen Nutzen dieses Ansatzes verdeutlichen:

- Da CATCH-II idealerweise alle europäischen Bürger erreichen sollte, müssen Texte (z. B. über Krankheiten, Präventionsmöglichkeiten, Anpassungen des Lebenstils usw.), parallel in verschiedenen europäischen Sprachen vorgehalten werden. Im Projekt werden die englischen Versionen als sogenannte Masterversion genutzt. Meta-Daten helfen dabei, die textuellen Ressourcen in verschiedenen Sprachen konsistent zu halten; mit anderen Worten, wenn es zu Änderungen in der Masterversion eines Textes kommt, kann automatisch die Aktualisierung der betroffenen Teile der parallelen Texte angestoßen werden.
- Die Auslieferung der Informationen im CATCH-II-System kann sowohl über das Internet als auch mit Hilfe so genannter Informationsterminals oder Kioske erfolgen, die dann auf ihre jeweilige Umgebung zugeschnitten sind (z. B. Kioske in Krankenhäusern). Die abgeleiteten Systeme sollen möglichst einfach aus dem gesamten Informationspool konfigurierbar sein. Wenn in den Meta-Daten die Information über den Hauptinhalt eines Informationsobjektes abgelegt ist, ist es wesentlich einfacher, automatisch solche Informationen aus dem Pool herauszuziehen, die jeweils für eine bestimmte Anwendung relevant sind (z. B. alle hautbezogenen Texte für den Kiosk in der dermatologischen Klinik).
- Wenn in den Texten wichtige Terme durch so genannte Inline-Tags semantisch klassifiziert werden können, lässt sich die automatische Verknüpfung textueller Ressourcen mit bereits existierenden erheblich vereinfachen. (So könnte etwa die Erwähnung einer Krankheit in einem Text mit dem Definitionstext dieser Krankheit verknüpft werden, die Erwähnung eines Arzneimittels mit einer Beschreibung dieses Arzneimittels usw.)

TAGING IN CATCH-II IM DETAIL

Die XML-Tags, die in den Informationsressourcen von CATCH-II verwendet werden, gehören zu den folgenden Kategorien:

– Tags zur Erfassung von Meta-Daten, die u. a. bibliografischer Natur sind,

1) Die Arbeiten im Projekt CATCH-II wurden durch die EU Kommission (Generaldirektorat Information Society) unter Vertrag HC 4004 gefördert. Zu den Partnern im Projekt CATCH gehörten neben dem Institut für Wissens- und Sprachverarbeitung (Prof. Dr. Dietmar Rösner) als Koordinator u. a. die AOK Magdeburg, die Universitätsklinik für Dermatologie und Venerologie (Prof. Dr. Harald Gollnick) der Otto-von-Guericke-Universität, die University of Ulster, Hewlett-Packard Italy und das portugiesische Nationalinstitut für kardiologische Prävention (INCP) in Lissabon.

– Tags zur Markierung struktureller Einheiten in den Texten,
– so genannte Inline-Tags zur semantischen Klassifikation natürlichsprachlicher Terme in den Texten.

Am besten lassen sich diese Kategorien durch Beispiele veranschaulichen. Wir wollen dieses am folgenden Ausschnitt aus einem Beispieltext über Hautkrebs tun (dabei bezeichnen „...“ Auslassungen).

```
<?xml version="1.0"?>
<CATCH-INFO-ELEMENT>
<META authors="Dr. Schramm, Luckert" supervisor="Prof. Gollnick" copyright="UDV, 1999"/>
<META translated-by="DR" translation-date="March-15-99" time-to-translate="50min"/>
...
<Body>
<Question-Answer-Pair>
<Question>
How to perform <PROC-Diagnostic> self examination and self diagnosis!
</PROC-Diagnostic>
</Question>
<Answer>
In prevention (of skin cancer) we distinguish between <DIS-Prevention> primary prevention </DIS-Prevention> through information and early detection as <DIS-Prevention> secondary prevention </DIS-Prevention>.
<Appeal>
There is no doubt: In addition to primary prevention early detection plays the most significant role in the fight against cancer. Don't forget: you are the most important factor in early detection!
</Appeal>
It is a big advantage that the <Organ System> skin </Organ System> -- in contrast to many other organs -- is visible and can be examined without technical devises and <PROC-Diagnostic> without invasive examination methods </PROC-Diagnostic>. We thus have the basis for an examination method applicable by everybody.

For all those <DIS-Etiology> malignant diseases of the skin </DIS-Etiology> that develops visibly regular self examination offers a big chance to detect <Disease> cancer </Disease> already in an early stage.
...
```

BIBLIOGRAFISCHE META-DATEN

Mit bibliografischen Meta-Daten werden Informationen über den Prozess des Erstellens, Übersetzens und Aktualisierens von Informationsressourcen erfasst. In CATCH-II wird hierfür ein Satz von Tags verwendet, die durch den sogenannten Dublin Core /4/ angeregt, jedoch mit Tags für eigene Zwecke erweitert worden sind. Im Beispieltext stellen etwa die Meta-Daten über die Autoren, den Betreuer, das Copyright und die Übersetzung bibliografische Meta-Daten dar.

AUSWEISEN VON STRUKTURELEMENTEN

Die so genannten strukturellen Tags erlauben den Autoren, deutlich zu machen, welche Art von Informationseinheit jeweils vorliegt und was ihr jeweiliger Zweck ist. Solche Informationen sind durch Prozesse, die mit den Informationsobjekten umgehen, interpretierbar.

Den Beispieltext kennzeichnen wir als Struktureinheit mit dem Namen „Question-Answer-Pair“. Eine solche Struktureinheit setzt sich aus zwei Elementen namens „Question“ und „Answer“ zusammen. Diese Analyse stützt sich auf die Beobachtung, dass der einleitende Satz eine (rhetorische) Frage darstellt, die durch den Rest des Textes beantwortet wird. Man beachte hierbei, dass eine solche Analyse den zugrundeliegenden Sprechakten mehr Bedeutung beimisst als der oberflächlichen syntaktischen Erscheinungsform.

Die als „Answer“ markierte Untereinheit umfasst hier den Rest des Informationselements.

Für die im Beispiel gewählte strukturelle Organisation könnte das Layout dann unterstreichen, dass die Frage auch als Titel der Informationseinheit fungiert. Liegen – wie in der Anwendung hier – eine ganze Reihe von Frage-Antwort-Paaren vor, so könnten alle Frageteile als Index für die entsprechenden Antworttexte verwendet werden und in einer dynamisch kreierten Überblicksseite als Einstieg angeboten werden.

Im Beispieltext sind noch weitere Einheiten entsprechend ihrer Funktion markiert.
– Es gibt einen expliziten Appell (Appeal):
  Mit einem Appell versucht ein Autor die Einstellungen des Lesers und seine Motive im Hinblick auf einen bestimmten Themenbereich zu beeinflussen (im Beispiel: regelmäßige Eigenuntersuchung der Haut).
– Es gibt eine Empfehlung (Recommendation):
  Empfehlungen beziehen sich auf Aspekte des Verhaltens des Lesers. Der Autor schlägt vor, bestimmte Handlungen durchzuführen oder sie in einer bestimmten Weise durchzuführen. Empfehlungen können auch als negative Versionen vorliegen und dann darauf hinweisen, bestimmte Handlungen zu vermeiden.

Im Folgenden eine Liste der derzeit in CEdit verwendeten strukturellen Tags:

```
Advice              List
Appeal              Newsletter
Comparison          Overview
Conclusion          Prohibition
Definition          Question-Answer-
                      Pair
Description         Quotation
Explanation         Recommendation
Interpretation      Summary
Introduction        Warning
```

Mit diesen Tags wird die Intention eines Textabschnitts explizit gemacht. Weitere Beispiele sind:
– Definition
  Eine Definition umfasst den zu definierenden Term (das Definiendum) und den definierenden Text (das Definiens).
– Warning
  Eine Warnung hat den Zweck, das Bewusstsein des Lesers für ein mögliches Risiko oder eine Gefahr zu wecken oder erneut zu wecken.

MEDIZINISCHE TERMINOLOGIE

Wichtige Terme in den Texten können von den Autoren semantisch klassifiziert werden. Die ihnen zur Verfügung stehenden semantischen Klassen (z. B. Disease für Krankheiten, PROC-Diagnostic für diagnostische Verfahren, ...) sind in einer Ontologie organisiert. Diese Ontologie ist weniger aus einer rein wissenschaftlichen Fachperspektive der Medizin, sondern vielmehr aus der Perspektive von Laien organisiert, stützt sich dabei aber auch auf professionelle Klassifikationssysteme und Nomenklaturen, wie z. B. MESH, SNOMED, UMLS (Unified Medical Language System, /5/).

DIE UNTERSTÜTZUNG VON AUTOREN IN CATCH-II

Immer wieder wird die Frage gestellt: Was haben die Autoren genau zu tun, wie werden sie dabei unterstützt?

Im einfachsten Fall könnten die Autoren ihre Texte in ihrem Lieblings-Text-Editor erstellen und als unformatierte ASCII-Datei ohne jegliche Tags den Administratoren des CATCH-II-Informationspools übergeben. Diese Autoren würden dann allerdings auf die Vorteile der Strukturierung und des Inline-Tagings ihres Textes völlig verzichten.

Nach unseren Erfahrungen schätzen es Autoren, wenn sie das Ergebnis ihrer Arbeit mit Hilfe von XML-Tags besser strukturieren können. Sie betrachten es als wichtige Arbeitserleichterung, wenn das Auszeichnen in die Textproduktion und das Editieren integriert ist. Die Werkzeuge in der Autorenumgebung von CATCH-II sind dafür geschaffen worden, dies in einer benutzerfreundlichen Weise zu ermöglichen.

CEDIT: XML-BASIERTE AUTORENUNTERSTÜTZUNG

Die CEdit-Anwendung steht allen Inhaltslieferanten aus den vier im CATCH-II-Projekt beteiligten Ländern zur Verfügung. CEdit hat eine einheitliche Benutzerschnittstelle, die aber in den verschiedenen Sprachen lokalisiert angeboten werden kann. /6/

Mit der Hilfe dieses Werkzeuges können die Autoren medizinische Texte kreieren, den Inhalt mittels Meta-Daten spezifizieren und sowohl strukturelle Auszeichnungen als auch semantische Typisierung von Termen in den Text einbringen.

CEdit ist in Form eines Java-Applets implementiert worden. Die Benutzung durch die Autoren erfolgt über das Internet. Es sind keine Installationsprozeduren oder andere Verwaltungsaktivitäten auf Seiten des Nutzers erforderlich. Bei jedem Einloggen steht jeweils die neueste Version von CEdit zur Verfügung.

ZUSAMMENFASSUNG DER FALLSTUDIE

Bei unserem Ansatz werden die Autoren von Dokumenten ermuntert, Meta-Daten bereitzustellen, ihre Dokumente logisch zu strukturieren und für relevante Terme im Text semantische Typisierungen zu liefern. Diese Zusatzarbeit der Autoren wird durch ein Werkzeug unterstützt, das die Annotation sehr vereinfacht. Die vorgeschlagenen Tags sind in einer Taxonomie organisiert und werden in einer menübasierten Benutzerschnittstelle verfügbar gemacht.

ZUR ÜBERTRAGBARKEIT DES ANSATZES

Die implementierten Werkzeuge wurden zwar für die Autoren von CATCH-II entwickelt, sind aber nicht durch diese Anwendung eingeschränkt. Von ihrer Konzeption her können sie auch in jeder anderen Umgebung genutzt werden, in der eine Bearbeitung strukturierter Informationsressourcen erforderlich ist. Mögliche Anwendungsgebiete sind etwa die Verwaltung multilingualer technischer Dokumentationen oder andere Anwendungen im kommerziellen Bereich. Die wichtigste Änderung, die für einen Transfer auf ein anderes Anwendungsgebiet notwendig ist, besteht darin, eine für das neue Anwendungsgebiet passende Ontologie bereitzustellen.

VERFÜGBARKEIT DES SYSTEMS:

Über das Internet kann auf CEdit zu Testzwecken zugegriffen werden:
URL: http://catch.cs.uni-magdeburg.de/CEdit
Benutzername: guest
Passwort: visitor
Die Autoren freuen sich über neue Anregungen und Rückmeldung von Testnutzern.

Literaturangaben

/1/ Murtha Baca. Introduction to Metadata – Pathways to Digital Information}. Getty Information Institute, 1998.

/2/ Jon Bosak. XML, Java, and the Future of the web.
http://sunsite.unc.edu/pub/sun-info/standards/xml/why/xmlapps.htm, 1996.

/3/ Tim Bray, Jean Paoli, and C.M. Sperberg-McQueen. Extensible Markup Language (XML) 1.0.
http://www.w3.org/TR/1998/REC-xml-19980210, 1998.

/4/ Dublin Core. Metadata Initiative. homepage of the Dublin Core Metadata Initiative. http://purl.oclc.org/dc/, 1999.

/5/ National Library of Medicine. homepage of the Unified Medical Language System (UMLS).
http://www.nlm.nih.gov/research/umls/, 1999.

/6/ Rösner, D., U. Dürer, M. Krüger, S. Neils: ‚XML-basierte Autorenunterstützung für die Erstellung und das Management von multilingualen Informationsressourcen', in: Turowski, K. u. Fellner, Kl. J. (Eds.): XML in der betrieblichen Praxis: Standards, Möglichkeiten, Praxisbeispiele; dpunkt-Verlag, ISBN 3-932588-91-6, 2001

Prof. Dr. Dietmar Rösner

Studium der Mathematik mit Nebenfach Informatik an der Universität Stuttgart, Stipendiat der Studienstiftung des Deutschen Volkes; Tätigkeit als wissenschaftlicher Mitarbeiter; Promotion 1986 in Informatik; von 1988 bis 1994 leitender Wissenschaftler für Mensch-Maschine-Kommunikation/Assistenzsysteme am Forschungsinstitut für anwendungsorientierte Wissensverarbeitung (FAW) an der Universität Ulm; 1994 Habilitation an der Fakultät Informatik der Universität Stuttgart; von 1994 bis 1995 Professor an der TU Bergakademie Freiberg (Sachsen); seit 1. 7. 1995 Professor (C4) für Angewandte Informatik/Wissensbasierte Systeme und Dokumentverarbeitung am Institut für Wissens- und Sprachverarbeitung (IWS), Fakultät für Informatik, Otto-von-Guericke-Universität Magdeburg.

Arbeitsschwerpunkte: Im Zentrum der Forschungsarbeiten steht die Frage, wie Wissen so repräsentiert werden kann, dass es vielseitig nutzbar und für verschiedenste Anwendungen einsetzbar ist. Ein wichtiges Arbeitsgebiet ist die automatische Generierung von Dokumenten aus repräsentiertem Wissen sowie die Analyse und Nutzung von Dokumenten auf der Basis von repräsentiertem Wissen. Auszeichnungssprachen wie XML spielen als Formalismus eine wichtige Rolle in den aktuellen Arbeiten, da sie das Potenzial haben, zum Zusammenwachsen „klassischer" Datenverarbeitung mit Dokumentverarbeitung und Wissensrepräsentation beizutragen.